



DCN No.
LAT-XR-07131-01

LAT PROJECT DOCUMENT CHANGE NOTICE (DCN)

SHEET 1 OF 2

ORIGINATOR: Mike DeKlotz **PHONE:** 650-926-4752 **DATE:** 8/16/05

CHANGE TITLE: DCN for LAT Flight Software Test Procedures **ORG.:**

DOCUMENT NUMBER	TITLE	NEW REV.
LAT-TD-07126	LAT FSW Qualification Test Procedure: DCMODE_001: ACD Diagnostics & Calibration	01
LAT-TD-07127	LAT FSW Qualification Test Procedure: DCMODE_002: ACD Diagnostics & Calibration	01
LAT-TD-07128	LAT FSW Qualification Test Procedure: DCMODE_003: TKR Diagnostics & Calibration	01
LAT-TD-07129	LAT FSW Qualification Test Procedure: NBTLMV_001: Narrowband Telemetry Housekeeping & Low-rate Science Data Verificaiton	01
LAT-TD-07130	LAT FSW Qualification Test Procedure: NBTLMV_002: Diagnostic Telemetry Verification	01

CHANGE DESCRIPTION (FROM/TO):

LAT-TD-07132-01- LAT FSW Qualification Test Procedure: FSWINI_001: FSW Initialization-SIU Primary Boot
 LAT-TD-07133-01- LAT FSW Qualification Test Procedure: FSWINI_002: FSW & LAT Initialization- Boot Self-Test & Boot Housekeeping Telemetry
 LAT-TD-07134-01- LAT FSW Qualification Test Procedure: FSWINI_003: FSW & LAT Initialization- Multiple Boot Images
 LAT-TD-07135-01- LAT FSW Qualification Test Procedure: FSWINI_004: FSW & LAT Initialization- SIU Hardware Reboot in response to the Signal on the Discrete Lines
 LAT-TD-07136-01- LAT FSW Qualification Test Procedure: FSWINI_005: FSW Initialization- EPU Primary Boot
 LAT-TD-07137-01- LAT FSW Qualification Test Procedure: FSWINI_006: FSW & LAT initialization- Reset Source
 LAT-TD-07138-01- LAT FSW Qualification Test Procedure: FSWINI_007: FSW & LAT Initialization- Storage & Retrieval of System Errors During SIU Primary Boot
 LAT-TD-07139-01- LAT FSW Qualification Test Procedure: FSWINI_008: FSW & LAT Initialization- Storage & Retrieval of System Errors During EPU Primary Boot
 LAT-TD-07140-01- LAT FSW Qualification Test Procedure: FSWINI_009: FSW & LAT Initialization- SIU Boot Status on Discrete Lines & SISU Boot Housekeeping Telemetry
 LAT-TD-07141-01- LAT FSW Qualification Test Procedure: FSWINI_010: FSW & LAT Initialization- SIU & EPU Secondary Boot
 LAT-TD-07142-01- LAT FSW Qualification Test Procedure: FSWINI_011: FSW & LAT Initialization- SIU & EPU Secondary Boot Error Mitigation
 LAT-TD-07143-01- LAT FSW Qualification Test Procedure: FSWINI_012: FSW & LAT Initialization- LAT SEU Protection
 LAT-TD-07144-01- LAT FSW Qualification Test Procedure: FSWINI_013: FSW & LAT Initialization- LAT Memory Scrubbing
 LAT-TD-07145-01- LAT FSW Qualification Test Procedure: FSWINI_014: FSW & LAT Initialization- Watchdog Management During Boot
 LAT-TD-07146-01- LAT FSW Qualification Test Procedure: FSWINI_015: FSW & LAT Initialization- Soft Reset
 LAT-TD-07152-01 - LAT FSW Qualification Test Procedure: FECALB_001: Charge Injection Calibration – TOT Measurements
 LAT-TD-07153-01- LAT FSW Qualification Test Procedure: FECALB_002: Charge Injection Calibration – TKR Threshold and Charge Scans
 LAT-TD-07154-01- LAT FSW Qualification Test Procedure: FECALB_003: Charge Injection Calibration – TKR Trigger Check
 LAT-TD-07155-01- LAT FSW Qualification Test Procedure: FECALB_004: Charge Injection Calibration – ACD Charge Injection
 LAT-TD-07156-01- LAT FSW Qualification Test Procedure: FECALB_005: Charge Injection Calibration – CAL Charge Injection
 LAT-TD-07157-01- LAT FSW Qualification Test Procedure: WBTLMV_001: Wideband Telemetry Verification – Science Data Format and Volume

REASON FOR CHANGE:

ACTION TAKEN: Change(s) included in new release DCN attached to document(s), changes to be included in next revision
 Other (specify):

DISPOSITION OF HARDWARE (IDENTIFY SERIAL NUMBERS):	DCN DISTRIBUTION:
<input checked="" type="checkbox"/> No hardware affected (record change only)	
<input type="checkbox"/> List S/Ns which comply already:	
<input type="checkbox"/> List S/Ns to be reworked or scrapped:	
<input type="checkbox"/> List S/Ns to be built with this change:	
<input type="checkbox"/> List S/Ns to be retested per this change:	
<input type="checkbox"/>	

SAFETY, COST, SCHEDULE, REQUIREMENTS IMPACT? YES NO
 If yes, CCB approval is required. Enter change request number:

APPROVALS	DATE	OTHER APPROVALS (specify):	DATE
ORIGINATOR: M. DeKlotz (signature on file)	8/16/05		
ORG. MANAGER: D. Horn (signature on file)	8/16/05		
PSA- K. Burlingham (signature on file)	8/16/05		
DCC RELEASE: Natalie Cramar (signature on file)	8/16/05	Doc. Control Level: <input checked="" type="checkbox"/> Subsystem <input type="checkbox"/> LAT IPO <input type="checkbox"/> GLAST Project	

DCN No: LAT-XR-07131-01




DCN No.
LAT-XR-07131-01

SHEET 2 OF 2

LAT PROJECT DOCUMENT CHANGE NOTICE

Continuation:

LAT-TD-07158-01- LAT FSW Qualification Test Procedure: FILMGT_001: File Management Verification
LAT-TD-07159-01- LAT FSW Qualification Test Procedure: MEMMGT_002: Memory Load Data
LAT-TD-07160-01- LAT FSW Qualification Test Procedure: NBTLMV_003: ACD HSK Anomaly Response and Alert Telemetry Verification
LAT-TD-07161-01- LAT FSW Qualification Test Procedure: OPMODE_001: LAT Operational Mode Control
LAT-TD-07162-01- LAT FSW Qualification Test Procedure: THRMCS_001: LAT Thermal Control System
LAT-TD-07163-01- LAT FSW Qualification Test Procedure: VSGIFV_001: Discrete Signal Interfaces
LAT-TD-07164-01- LAT FSW Qualification Test Procedure: CMDFNC_003: 1553 Interface and Command Functional Verification
LAT-TD-07054-01- LAT FSW Qualification Test Procedure: EVTPMO_001: Event Performance Monitoring and Verification – Software-Related Deadtime
LAT-TD-07055-01- LAT FSW Qualification Test Procedure: EVTPMO_002: Event Performance Monitoring and Verification –Acquisition of VETO Rates from the GEM
LAT-TD-07056-01- LAT FSW Qualification Test Procedure: EVTPMO_003: Event Performance Monitoring and Verification – Level 1 Trigger Rates
LAT-TD-07057-01- LAT FSW Qualification Test Procedure: EVTPMO_004: Event Performance Monitoring and Verification – Monitor CNO Rates
LAT-TD-07112-01- LAT FSW Qualification Test Procedure: EVTFIL_001: Event Filtering – Interface from the Event Builder
LAT-TD-07113-01- LAT FSW Qualification Test Procedure: EVTFIL_002: Event Filtering – Event Filter Rates and Capacity
LAT-TD-07114-01- LAT FSW Qualification Test Procedure: EVTFIL_003: Event Filtering – Event Filter Reprogramming
LAT-TD-07115-01- LAT FSW Qualification Test Procedure: EVTFIL_004: Event Filtering – Event Filter Bypass

	Document # LAT-TD-07158-01	Date effective 8/16/05
	Author(s) Shantha Condamoor Mark Arndt	Supersedes
	Subsystem/Office Electronics & DAQ Subsystem	
Document Title LAT FSW Qualification Test Procedure: FILMGT_001: File Management Verification		

CHANGE HISTORY LOG

Revision	Effective Date	Description of Changes
01	8/16/05	Original

TABLE OF CONTENTS

Table of Contents	3
1. Scope	5
1.1 Test Suite	5
1.2 Test ID	7
1.3 Requirement(s) Tested	7
2. DEFINITIONS AND ACRONYMS	12
2.1 Definitions	12
2.2 Acronyms	12
3. REFERENCES	13
3.1 Applicable Documents	13
4. REQUIREMENTS	14
4.1 Test Data, Equipment and Software	14
4.2 Quality Assurance	14
4.3 Safety	14
4.4 Warnings, Cautions, and Notes	15
4.5 General Instructions	15
5. SETUP	16
5.1 Hardware Setup	16
5.2 Software Setup	17
5.3 Setup Validation	19
6. TEST PROCEDURE FOR FILMGT_001	20
6.1 Test Objective	20
6.2 Test Input Files	23

6.3	Test Preparation	23
6.4	Test Procedure and Test Analysis.....	24
7.	TEST POST_CONDITIONS AND OVERALL OUTCOME.....	49
7.1	Test Post-Conditions.....	49
7.2	Overall Outcome of FILMGT_001.....	49
8.	CERTIFICATION	50
	Appendix A: Deviations from the Qualification Test Procedure.....	51

1. **SCOPE**

This document describes the procedure followed for qualification testing of the LAT FSW. This Qualification Test Procedure document describes one of the qualification tests executed to verify compliance with the requirements defined in the “Flight Software Specification – Level III” (LAT-SS-00399).

1.1 **Test Suite**

FILMGT

The Test Suite is a logical grouping of a set of tests based on common functionalities and can be executed as a suite of tests in some order when the pre-conditions and post-conditions for each test within the suite have been met successfully.

This test procedure *FILMGT_001* belongs in the Test Suite *FILMGT* under the *FST* Project.

The *FILMGT_001* test verifies the capability of FSW, on command, to successfully upload files into SIU or EPU and dump the contents of files from SIU or EPU file systems. Additionally, *FILMGT_001* verifies that FSW can create and delete directories and dump file system status data.

1.1.1 **Relationship of *FILMGT_001* to Other Test Suites**

Responsibility for verifying 1553 interface and telecommand validation and processing requirements is shared between the *FILMGT_001* test and the *CMDFNC* test suite. In particular, *FILMGT_001* verifies that the subset of telecommands and telemetry associated with the file management (FILE and LFS) FSW have the correct format, can be received over the 1553 interface and processed, and that file management telecommands with valid parameters are correctly processed. By contrast, the *CMDFNC* suite tests these file management telecommands and telemetry when packet structures and parameters are *invalid*. The specific requirements shared between these test suites are identified in Section 1.3 below. See the *CMDFNC* test procedure documents for a complete discussion of how responsibility for testing the entire telecommand and telemetry interface is mapped among various qualification test suites.

Responsibility for verifying the ability of FSW to transmit diagnostic telemetry is shared between the FILMGT_001 test and the NBTLMV test suite. The specific requirements shared between these tests are identified in Section 1.3 below. Again, see the NBTLMV test procedure documents for a complete discussion of how responsibility for testing diagnostic telemetry is partitioned among various qualification test suites.

The following FILE and LFS telecommands and LFS telemetry are used in test FILMGT_001.

FILE Telecommands

APID	FC Cmd. Packet (L)	Description (L)
0x641	0 LFILUPLSTART	File Upload Start
0x641	1 LFILUPLCANCEL	File Upload Cancel
0x641	2 LFILUPLCOMMIT	File Upload Commit
0x641	3 LFILUPLDATA	File Upload Data
0x641	4 LFILUPLLEPU	File Upload to EPU

LFS Telecommands

APID	FC Cmd. Packet (L)	Description (L)
0x648	0 LLFSFILEDELETE	File Delete
0x648	1 LLFSFILECOPY	File Copy Local
0x648	2 LLFSDIRCREATE	Directory Create
0x648	3 LLFSDIRDELETE	Directory Delete
0x648	4 LLFSFILEDUMPC	File Dump CTDB
0x648	5 LLFSDIRDUMP	Directory Dump
0x648	6 LLFSSYSSTATUS	File System Status

LFS Telemetry

APID	Tlm. Packet (L)	Description (L)
0x318	LLFSDIRLIST	Directory Listing Report

0x319 LLFSROOTLIST	Root Listing Report
0x31A LLFSSYSLIST	File System Status Report
0x31B LLFSDUMPCTDB	File Dump Data CTDB

1.2 Test ID

FILMGT_001

Test *FILMGT_001* contains one main script, *FILMGT_001*, which can be run when the SIU or an EPU is in the Boot or Application Mode. This test verifies file/directory creations/deletions and dumps.

1.3 Requirement(s) Tested

The Qualification Test Procedure described herein is performed to verify that the FSW satisfies the following requirement(s), quoted from the Flight Software Specification – Level III:

Requirement Number	Requirement Name	Requirement	Level of Requirements Verification in This Test
5.3.7.1	Directory Create	In order to create a directory, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes directory identifier, device identifier, and unit identifier.	Full
5.3.7.2	Directory Delete	In order to delete a directory, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes directory identifier, device identifier, and unit identifier.	Full
5.3.7.3	File Delete	In order to delete a file, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes file identifier, directory identifier, device identifier, and unit identifier.	Full

Requirement Number	Requirement Name	Requirement	Level of Requirements Verification in This Test
5.3.7.4	File Copy within a Processor	In order to copy a file within a processor, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes source file identifier, source directory identifier, source device identifier, unit identifier, destination file identifier, destination directory identifier, and destination device identifier.	Full
5.3.7.5	File Copy from SIU to EPU	In order to copy a file from the SIU to an EPU processor, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes source file identifier, source directory identifier, source device identifier, and destination unit identifier. Note that a separate commit command is sent to the EPU to move the file from EPU RAM into the file system.	Full
5.3.7.6.1	File Dump Command	In order to dump a file, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes file identifier, directory identifier, device identifier, and unit identifier.	Full
5.3.7.6.2	File Dump Data	In response to receiving a File Dump command, the FSW shall transmit the requested file data to the spacecraft.	Full
5.3.7.7.1	File Directory Dump Command	In order to dump a file, the FSW shall receive as input, from the spacecraft via the CTDB, a command that includes directory identifier, device identifier, and unit identifier.	Full
5.3.7.7.2	File Directory Dump Data	In response to receiving a File Directory Dump command, the FSW shall transmit the requested data to the spacecraft, including: device identifier, directory identifier, file identifier, setting of read-only flag, archive flag setting, most recent update time, byte size, and number of blocks for each file and subdirectory in the requested directory.	Full

Requirement Number	Requirement Name	Requirement	Level of Requirements Verification in This Test
5.3.7.8.1	File System Status Dump Command	The FSW shall receive as input, from the spacecraft via the CTDB, a command to dump the file system status for a specific unit.	Full
5.3.7.8.2	File System Status Dump Data	In response to receiving a File System Status Dump command, the FSW shall transmit the requested data to the spacecraft, including: total block size, blocks used, and blocks free for the requested unit.	Full
5.3.7.9	File Loads	After entering the boot shell, the FSW processors shall be commandable to perform file loads.	Full
5.3.7.10	File Upload Cancellation	The FSW shall be commandable to cancel an active file load.	Full
5.3.3.8	Multiple Block Commands	The FSW shall execute block commands for uploads or dumps serially (i.e., not simultaneously, not interleaved with other block commands).	Partial (FILMGT_001 tests block command handling of file management telecommands. Handling of memory management telecommand blocks is covered in the MEMMGT suite)

This test also shares verification of certain requirements with the NBTLMV suite. The requirements partially verified in this test are listed below.

Requirement Number	Requirement Name	Requirement	Level of Requirements Verification in This Test
5.3.5.2	Diagnostic Telemetry	[Derived] The FSW shall send diagnostic telemetry messages, as defined in [9], to the SC via the CTDB. Diagnostic telemetry is sent in response to a command and downlinked immediately while the SC is in contact with the ground. If not in contact, the telemetry is written to the solid-state recorder on the spacecraft.	Partial (FILMGT_001 verifies that telemetry packets transmitted by the file management FSW have APIDs in the diagnostic telemetry APID range and that these packets are delivered over the CTDB, as required. Transmission of other diagnostic telemetry is covered in the NBTLMV test suite)

This test also shares verification of certain requirements with the CMDFNC suite. The requirements partially verified in this test are listed below.

Requirement Number	Requirement Name	Requirement	Level of Requirements Verification in This Test
5.2.2.1.1	Command, Telemetry, and Data Bus Protocol – SC	[2] (3.2.5.1.1) [3] (5.3.4) [4] (5.3.1) The SIU FSW shall exchange commands, low rate telemetry, time messages, and ancillary data with the SC C&DH across a MIL-STD-1553B bus, using the MIL-STD-1553B physical layer protocol.	Partial (Also see the CMDFNC suite)
5.2.2.1.3	Data Format	[1] (3.1.2.5.1.1) [2] (3.2.5.4.6) All data exchanged across the CTDB interface shall be formatted as CCSDS telecommand or telemetry packets, as specified in [7].	Partial (Also see the CMDFNC suite)
5.3.3.5	Command Format	[Derived] The FSW shall be able to process commands of the format described in [9].	Partial (Also see the CMDFNC suite)
5.3.3.9.1	Data Integrity Errors	[Derived] To reduce the chance of executing a corrupted command, the FSW shall validate commands prior to execution.	Partial (Also see the CMDFNC suite)
5.3.3.9.2	Parameter Validation	[Derived] The SIU FSW shall validate command parameters prior to execution.	Partial (Also see the CMDFNC suite)
5.3.3.1	SC Command Processing by SIU	[2] (3.2.6.1.1.2) [4] (5.3.1) The SIU FSW shall process LAT commands from the SC, received via the CTDB, in the order that they are received.	Partial (Also see the CMDFNC suite)
5.3.3.2	Command Execution Notification	[Derived] For all LAT commands from the SC, the FSW shall be capable of generating time-tagged telemetry notification of their dispatch for execution.	Partial (Also see the CMDFNC suite)

Requirement Number	Requirement Name	Requirement	Level of Requirements Verification in This Test
5.3.3.3	Command Completion Status	<p>[Derived]</p> <p>The FSW shall be capable of generating time-tagged telemetry notification of command completion status (success or general cause of failure) for ground-initiated commands at the time such information becomes available.</p> <p>As a goal, in addition to completion status, each successful command should result in a modification of a telemetry parameter (e.g., Off to On, Open to Closed, etc.) to further confirm success.</p>	<p>Partial</p> <p>(Also see the CMDFNC suite)</p>
5.3.3.6	Command Execution	<p>[4] (5.3.2)</p> <p>The SIU FSW shall be able to reconfigure and direct the operation of the instrument through a combination of direct execution of commands from the SC.</p>	<p>Partial</p> <p>(Also see the CMDFNC suite)</p>

If the requirement(s) quoted above cite external documents (e.g., "...Further details are provided in [11]"), consult LAT-SS-00399 for the list of citations.

2. DEFINITIONS AND ACRONYMS

The following terms, abbreviations, and acronyms are used in this document:

2.1 Definitions

Hz	Hertz, unit of frequency
s, sec	seconds
V	Volt
W	Watt

2.2 Acronyms

CAL	Calorimeter
EGSE	Electrical Ground Support Equipment
GASU	Global trigger Anti-collision Spacecraft Unit
PTR	Post Test Review
TEM	Tower Electronics Module
TKR	Tracker
TPS	Tower Power Supply
TRR	Test Readiness Review
QAE	Quality Assurance Engineer
SC	Spacecraft
TE	Test Engineer

3. REFERENCES

The list below provides documents that are to be used as references for this procedure:

3.1 Applicable Documents

<u>Document Number</u>	<u>Description</u>
<u>SPECIFICATIONS</u>	
LAT-SS-00399	LAT Flight Software Level III Specification
1196 EI-S46310-000	GLAST 1553 Bus Protocol Interface Control
LAT-TD-02659	LAT Flight Software Telecommand and Telemetry Formats
LAT-TD-0561	The Virtual Spacecraft (VSC)
<u>PROCEDURES</u>	
LAT-TD-TBD	CMDFNC_003: 1553 Interface and Command Functional Verification
LAT-TD-TBD	NBTLMV_002: TBD
<u>PLANS</u>	
LAT-MD-00039	Performance Assurance Implementation Plan
LAT-MD-00078	GLAST LAT System Safety Program Plan
LAT-MD-00404	LAT Contamination Control Plan
LAT-MD-00408	LAT Program Instrument Performance Verification Plan
LAT-SS-00296	T & DF Test Plan
LAT-TD-00297	LAT Electronics Test Plan
LAT-TD-00786	LAT Flight Software Test Plan
LAT-MD-00104	LAT FSW Management Plan
<u>DRAWINGS</u>	
N/A	
<u>OTHER</u>	
LAT-MD-00091	GLAST Quality Manual
LAT-MD-00471	Control of Nonconforming Product
LAT-MD-00472	Corrective and Preventative Action

Hard copies of this document are for REFERENCE ONLY and should not be considered the latest revision.

4. REQUIREMENTS

This section lists the requirements that shall be followed during the LAT FSW Qualification Testing process.

The Performance Assurance Implementation Plan, LAT-MD-00039, shall be utilized to ensure that the products produced by the GLAST LAT project intended for design qualification, flight and critical ground support equipment usage meet the required levels of quality and functionality for their intended purposes.

The LAT Program Instrument Performance Verification Plan, LAT-MD-00408, shall be utilized to address the testing to be performed at the unit/subsystem and instrument level for flight qualification, proto-flight and acceptance testing phases. Also included are the necessary processes/procedures and systems assurance activities.

4.1 Test Data, Equipment and Software

This procedure shall follow the requirements found in the Control of Nonconforming Product, LAT-MD-00471. This document establishes the method to identify and control nonconforming product developed by the LAT project team.

4.2 Quality Assurance

This procedure shall follow the requirements found in the Corrective and Preventative Action document, LAT-MD-00472 and the GLAST Quality Manual, LAT-MD-00091.

The Corrective and Preventative Action document establishes the method to be used to initiate, implement, evaluate and record corrective and preventive actions. The GLAST Quality Manual defines the methods implemented by the GLAST LAT project to ensure consistent quality of all processes for procurement, design, development and production of flight hardware, flight software and all associated ground support equipment interfacing with flight hardware and software.

4.3 Safety

This procedure shall follow the requirements found in the GLAST LAT System Safety Program Plan, LAT-MD-00078. This document defines all phases of the LAT program including: design, development, fabrication, handling, transportation, storage, test, assembly and operation.

WARNING: When high voltages are present extreme care should be exercised.

4.4 Warnings, Cautions, and Notes

The following SAFETY ALERTS are intended to create awareness of the potential safety hazards and the steps that must be taken to avoid accidents. These same alerts are used throughout this document to identify specific hazards that may endanger personnel and/or equipment.

Identification of every conceivable hazardous situation is impossible. Therefore, all personnel have the responsibility to diligently exercise safe practices whenever exposed to this equipment.

WARNING: Indicates a potential hazardous situation which, if not avoided, could result in death or injury.

CAUTION: Indicates a potential hazardous situation which, if not avoided, could result in damage to equipment.

Note: Indicates a notification of information that is important, but not hazard related.

4.5 General Instructions

This qualification test procedure shall be conducted on a formal basis to its latest approved and released version. The designated Software QAE shall be notified 24 hours prior to the start of this procedure. Software QAE may monitor the execution of all or part of this procedure should they elect to do so.

The Test Engineer conducting this test shall read this document in its entirety and resolve any apparent ambiguities before beginning the procedures described herein.

Deviations from the procedures described in this document and breaks in hardware or software configuration can only be initiated by the Test Engineer, must be approved by QA, and must be documented in Appendix A.

Any nonconformance/defect/anomaly is to be reported in JIRA. Refer to the LAT Flight Software Test Plan LAT-TD-00786 for guidance. Do not alter or break configuration if a failure occurs. Notify Software Quality Assurance.

All success conditions for a test must be met for the test to pass.

5. SETUP

This section describes the hardware and software configuration used for the qualification test described later in this document. Any break from configuration or deviation from a particular procedure must be authorized by the Quality Assurance Engineer and documented in Appendix A.

5.1 Hardware Setup

The list below indicates the equipment that is used to execute the tests described in this document.

Hardware Unit	Manufacturer	GLAT Number (and Hardware Sub-Units by GLAT Number)	Firmware Version (where applicable)
Virtual Spacecraft (VSC)	SLAC		
Spacecraft Interface Unit (SIU)	SLAC		
Event Processing Unit (EPU): 2 (EPU0 and EPU1)	SLAC		
Global trigger, AEM Signal Distribution Unit (GASU)	SLAC		
Power Distribution Unit (PDU)	SLAC		
1553 cables and couplers	SLAC		
Unix or Linux Host Establishes connection between VSC and the terminal from which the test is run.	Sun Microsystems		
Power supply for SIU	SLAC		

The Figure below depicts the Testbed on which this qualification test is performed. The particular hardware units utilized in this test are itemized by GLAT number and firmware version in the

preceding table.

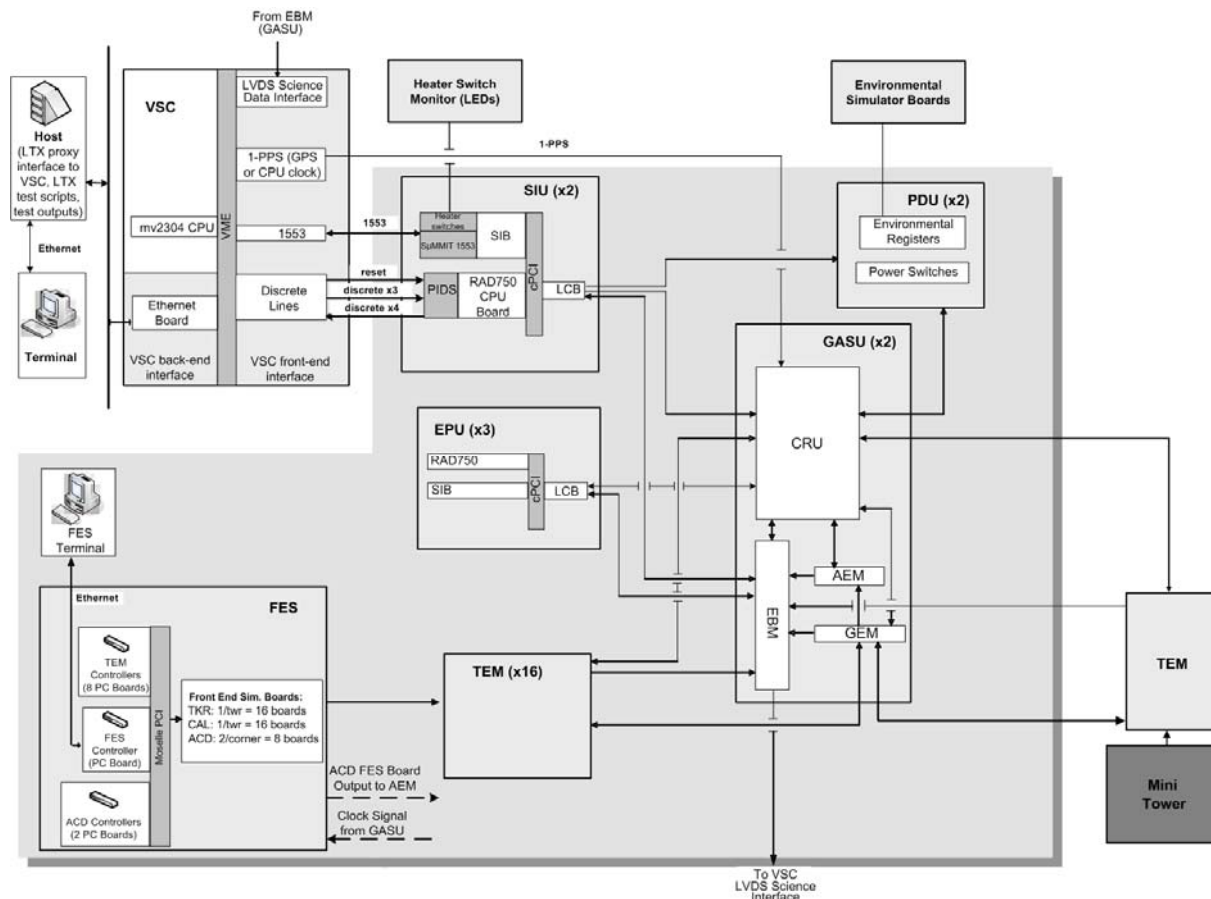


Figure 1. FSW Testbed

5.2 Software Setup

The software required to prepare for and execute the tests described in this Qualification Test Procedure document is itemized in this section.

5.2.1 Test Tools

The following table specifies the test executive used to run this qualification test, and identifies the other software tools used to support the execution of the test. The “Software Version Number” column identifies the version number of the test tool being used; alternatively, this column identifies the hardcopy attachment to this document that records the version of the tool being used (e.g., “Attachment 1”). The “Path to Attachment” column identifies the directory in which the electronic copy of any hardcopy attachment is saved (if applicable).

Software	Description of Software	Software Version Number (or Specify Attachment Number)	Path to Attachment (If Applicable)
LTX	LAT Test Executive		
FMX	File Management Extra. FMX is used prior to this test to prepare RTOS image files and other files for upload and storage to EEPROM.		
VSC	Virtual Spacecraft system software		
VPI	VSC Python/Proxy Interface		

5.2.2 Test Scripts

The following table identifies the test scripts that are run to execute this qualification test. The “Script Version Number” column identifies which version of the script is being used; alternatively, this column identifies the hardcopy attachment to this document that records the version of the script being used (e.g., “Attachment 1”). The “Path to Attachment” column identifies the directory in which the electronic copy of any hardcopy attachment is saved (if applicable).

Test Script	Description of Test Script	Script Version Number (or Specify Attachment Number)	Path to Attachment (If Applicable)
FILMGT_001.py	This script, the main test execution script, can be run in the Boot or Application Modes. It executes tests of SIU FSW, EPU0 FSW, and EPU1 FSW. The script controls the entire test process, detecting the state of the hardware and FSW and advancing SIU and EPU FSW through the different operational modes covered during the test.		

5.2.3 Flight Software

This qualification test is performed on a complete, integrated Candidate Release of FSW. All FSW libraries under test are final Flight Unit Candidate versions. The test described in this document is designed to evaluate the particular FSW packages and constituents listed in the following table; for clarity, only those constituents that are the focus of the test are listed below.

FSW Package	Constituent(s)
FILE	
LFS	
PBC	

The Candidate Release into which these FSW constituents are integrated is identified in the following table. The “Candidate Release Build Number” column specifies the build number of the FSW Candidate Release under test (e.g., “B0-0-1”). The “Candidate Release Attachment Number” column identifies which hardcopy attachment to this document confirms the name and version number of the full set of FSW packages and constituents used in the test (e.g., “Attachment 2”). The “Path to Attachment” column specifies where in the test repository an electronic copy of the hardcopy attachment has been saved.

Candidate Release Build Number	Candidate Release Attachment Number	Path to Attachment

5.3 Setup Validation

5.3.1 Hardware Validation

The following signatures confirm that the Test Engineer and Quality Assurance Engineer have verified the GLAT numbers, firmware version numbers, and the proper connection of all hardware listed in the table in Section 5.1.

 Date Time Test Engineer QAE

5.3.2 Software Validation

The Test Engineer performs the following procedure to validate the software setup for this qualification test and records completion of the setup validation steps in the space provided.

Step No.	Description of Step	Step Outcome
1	Record the version numbers of all test tools used to perform this qualification test in the table in Section 5.2.1	Complete/ Not Complete
2	Record the version numbers of all test scripts used to perform this qualification test in the table in Section 5.2.2	Complete/ Not Complete
3	Record the version numbers of the FSW constituents and the Candidate Release on which this test is performed in Section 5.2.3.	Complete/ Not Complete

The following signatures confirm that, using the procedure described in the previous table, the Test Engineer and Quality Assurance Engineer have verified that all versions of test support software, test scripts, and FSW constituents match those identified in Section 5.2.

 Date Time Test Engineer QAE

6. TEST PROCEDURE FOR FILMGT 001

6.1 Test Objective

The following are the objectives of this test when tested in the Boot or Application Modes.

This test verifies that the SIU or EPU FSW is commandable by ground to perform File Management Operations listed in the Requirement(s) Tested section.

This Test Objective is broken down into the following Test Sub-Objectives.

Number	Test Sub-Objective
1	Verify file management functionality on SIU, EPU0 and EPU1.
2	Verify functionality in both the Boot and Application Modes.
3	Verifies File System Status Dump command in the Application Mode by issuing the LLFSSYSSTATUS command and verifying the resultant File System Status Report Telemetry LLFSSYSLIST.
4	Verifies Directory Dump command in the Application Mode by issuing the LLFSDIRDUMP command and verifying the resultant the Directory Listing Report Telemetry LLFSDIRLIST and Root Listing Report LLFSROOTLIST. Verifies Root Directory Dumps for all devices. Verifies that Directory Dump from an invalid file id (non-existent directory) results in error.
5	Verifies File Dump command in the Application Mode by issuing the LLFSFILEDUMPC command and verifying the resultant File Dump Data CTDB Telemetry LLFSDUMPCTDB. Verifies File Dumps of small, medium and big files. Verifies the dump of boot partition RTOS files that were uploaded in the Boot Mode as well as dump of Application files that were uploaded in the Application Mode.

Number	Test Sub-Objective
	. Verifies that File Dump from an invalid file id (non-existent file) results in error.
6	<p>Verifies Directory Creation in the Application Mode only by issuing the LLFSDIRCREATE command.</p> <p>Verifies Directory Creation on both EEPROM banks: EEPROM0, EEPROM1</p> <p>Verifies that Directory Creation of an existing directory results in error.</p>
7	<p>Verify file uploads on SIU by issuing the LFILUPLSTART, LFILUPLDATA and LFILUPLCOMMIT commands. Verified in both the Boot and Application Modes.</p> <p>Verify File uploads for EPUs by issuing File Copy from SIU to EPU command LFILUPLEPU and following it with LFILUPLCOMMIT commands. Verified only when SIU is in the Application Mode. EPU can be in Boot or Application Modes.</p> <p>Verify on both banks: EEPROM0 and EEPROM1</p> <p>Verify for file uploads of medium size file (few telecommand packets) and large size file (several telecommand packets).</p> <p>Verify RTOS file uploads to Boot Partition Areas as well as Application file uploads.</p> <p>Verifies that a File Upload to a destination file number that already exists results in an error. Verifies that File Upload to an incorrect destination file id (invalid directory number), results in error.</p>
8	<p>Verify File Upload Cancellation on SIU by issuing the LFILUPLSTART, LFILUPLDATA followed by LFILUPLCANCEL commands. Verified in both the Boot and Application Modes.</p> <p>Verify File upload cancellation for EPUs by issuing File Copy from SIU to EPU command LFILUPLEPU and following it with and LFILUPLCANCEL commands. Verified only when SIU is in the Application Mode. EPU can be in Boot or Application Modes.</p> <p>Verify Upload Cancellation on both banks: EEPROM0 and EEPROM1</p> <p>Verify for file upload cancellation of a large size RTOS file (several telecommand packets) in the Boot Mode as well as large size App mode files.</p>
9	<p>Verifies File Copying within a processor in the Application Mode by issuing the LLFSFILECOPY command.</p> <p>Verifies File Copying within SIU and EPUs.</p> <p>Verifies File Copying on both EEPROM banks: EEPROM0, EEPROM1</p> <p>Verifies that File Copying to a destination file number that already exists results in error. Verifies that File Copying from a source file that does not exist results in error. Verifies that File Copying to an incorrect destination file id (invalid directory number) results in error.</p>
10	<p>Verifies File Copying from SIU to EPU in the Application Mode only by sending the command LFILUPLEPU and following it with and LFILUPLCOMMIT commands. Verified only when SIU is in the Application Mode. EPU can be in Boot or Application Modes.</p> <p>Verifies File Copying on both EEPROM banks: EEPROM0, EEPROM1</p> <p>Verifies that File Copying to a destination file number that already exists, results in error. Verifies File Copying from a source file that does not exist, results in error. Verifies that File Copying to an incorrect destination file id (invalid directory number) results in error.</p>

Number	Test Sub-Objective
11	<p>Verifies File Deletion in the Application Mode only by issuing the LLFSFILEDELETE command.</p> <p>Verifies File Deletion on both EEPROM banks: EEPROM0, EEPROM1</p> <p>Verifies that File Deletion of a non-existent file, results in error.</p>
12	<p>Verifies Directory Deletion in the Application Mode only by issuing LLFSDIRDELETE command.</p> <p>Verifies Directory Deletion on both EEPROM banks: EEPROM0, EEPROM1</p> <p>Verifies that Directory Deletion of an empty directory, results in success. Verifies that Directory Deletion of a non-existent directory, results in error. Verifies that Directory Deletion of a directory that still contains some files, results in error.</p>
13	<p>Verifies that FSW executes block commands for uploads or dumps serially (i.e., not simultaneously, not interleaved with other block commands).</p> <p>Verifies interleaved File Upload Commands are rejected. Verifies interleaved File Dumps are rejected. Verifies interleaving non-block and block commands are successful.</p>

As explained in Section 1, the FILMGT_001 test is related to tests in the NBTLMV suite. FILMGT_001 partially verifies the diagnostic telemetry requirements, thus the following are also listed as Test Objectives and are verified in this test.

Number	Test Sub-Objective
1	<p>Verify that file management FSW transmits telemetry packets with APIDs in the diagnostic telemetry range and that this telemetry is delivered over the CTDB interface.</p>

In addition, the FILMGT_001 test is related to tests in the CMDFNC suite. FILMGT_001 partially verifies some of the 1553 interface and command validation requirements, thus the following are also listed as Test Objectives and are verified in this test.

Number	Test Sub-Objective
1	<p>Verify that the FILE and LFS Telecommands and Telemetry exchanged across the 1553 Interface conform to CCSDS packet structure:</p>
2	<p>Verify that all supported valid FILE and LFS Telecommands issued by S/C with LAT as their destination are received correctly and processed.</p>
3	<p>Verify that SIU FSW processes LAT commands from the SC in the order that they are received.</p>
4	<p>Verify that for all FILE and LFS commands from SC, the FSW is capable of generating time-tagged telemetry notification of their dispatch for execution.</p>
5	<p>Verifies that FSW is capable of generating time-tagged telemetry notification of command completion status (success</p>

Number	Test Sub-Objective
	or general cause of failure) for all FILE and LFS commands.
6	Verifies that SIU FSW receives commands at the rate of up to 20 commands per second.

Analysis of results is performed as and when data arrives in telemetry. Typically analysis includes verifying the telemetry values against expected values and tagging the sub-objectives as either “PASS” or “FAIL”.

6.2 Test Input Files

The following table identifies all auxiliary files (e.g., Front End Simulator data files, GLEAM data files) used as inputs to this qualification test. Note that not all qualification tests use input data of this type. The “Input File Version Number” column identifies the version number of the auxiliary file being used; alternatively, this column identifies the hardcopy attachment to this document that records the version of the file being used (e.g., “Attachment 1”). The “Path to Attachment” column specifies where in the test repository an electronic copy of the hardcopy attachment has been saved .

Input File	Description of Input File	Input File Version Number (or Specify Attachment Number)	Path to Attachment (If Applicable)
TBD1.f	RTOS image file 1		
TBD2.f	Upload file 1 (small size, spans one LFILEUPDATA packet)		
TBD3.f	Upload file 1 (medium size, spans a few LFILEUPDATA file upload packets)		
TBD4.f	Upload file 2 (large size, spans several LFILEUPDATA file upload packets)		

6.3 Test Preparation

After the hardware and software setup has been validated, steps may be required to place the hardware and FSW in an operational mode in which the qualification test can be performed or otherwise complete preparations for the test to begin.

The Test Engineer carries out the following procedure to prepare for qualification test and records completion of the test preparation steps in the space provided.

Step No.	Description of Step	Step Outcome
1	Confirm that the VSC is powered up.	Complete/ Not Complete

The Test Engineer and Quality Assurance Engineer verify that all test preparation steps are complete.

Date

Time

Test Engineer

QAE

6.4 Test Procedure and Test Analysis

This section describes the step by step procedure performed once the test preparation is complete. The test as well as analysis for each of the sub-objectives is conducted by the main test script FILMGT_001.

The Test Engineer proceeds with the qualification test procedure itself, as described below, and records the outcome of each step during test execution. The outcome of each step is either “Complete” or “Not Complete” (for steps which involve no analysis or verification); or, “Pass” or “Fail” (for steps involved in verifying completion of test objectives and sub-objectives).

The test procedure is executed on the SIU and EPUs, in both Boot Mode and Application Mode, in 7 major parts:

- SIU Boot Mode Testing
- SIU Application Mode Testing
- EPU0 Boot Mode Testing
- EPU0 Application Mode Testing
- EPU1 Boot Mode Testing
- EPU1 Application Mode Testing
- Review of Test Outputs

6.4.1 Part 1: SIU Boot Mode Testing

This part of the test exercises Boot Mode file upload (FILE) functionality on the SIU.

Step No.	Description of Step	Step Outcome
1	At the test terminal, run the script FILMGT_001 under LTX through the VSC with the following command: \$ ltx run <FILMGT_001	Complete/ Not Complete
2	The test script determines whether the SIU is powered on by checking whether telemetry is being transmitted. Regardless of the SIU's current operational mode, the script sends the SIU the LPBCRESET telecommand to reboot the unit. The script then checks whether the SIU FSW is now operating in Boot Mode by detecting if boot housekeeping is being transmitted. If not, the script sends the LPBCRESET command and checks again. If the SIU FSW cannot be placed in Boot Mode, LTX prints an error message to the screen and exits, aborting the test. If the test is NOT aborted, mark "Complete" for the Step Outcome.	Complete/ Not Complete
3	With FSW on the SIU advanced to the Boot Mode, the test script begins executing the main FILMGT test script, starting with the SIU Boot Mode File Uploads phase. The test script uploads the RTOS image file(s) listed in Section 6.2 to the SIU by sending the LFILUPLSTART, LFILUPLDATA, and LFILUPLCOMMIT commands to the SIU. The file is only uploaded to the boot device. The script examines the LPBCTOTALERRCNT, LPBCNEXTERRWORD, LPBCTCREVCNT, LPBCTCACCCNT, and LPBCLASTERRWORD fields in SIU boot housekeeping telemetry (LBTHKP packets) to confirm success or failure of the File Upload commands. The script also examines the LPBCFILESTAT and LPBCFILEPKTCNT fields in SIU boot housekeeping to confirm a successful upload.	N/A
4	The test script next executes its SIU Boot Mode File Upload Cancellation phase. The test script re-issues the LFILUPLSTART and LFILUPLDATA commands, then sends the LFILUPLCANCEL command to cancel the upload. The script examines the LPBCTOTALERRCNT, LPBCNEXTERRWORD, LPBCTCREVCNT, LPBCTCACCCNT, and LPBCLASTERRWORD fields in the SIU boot housekeeping telemetry to confirm success or failure. The script also checks the LPBCFILESTAT and LPBCFILEPKTCNT fields in SIU boot housekeeping telemetry to confirm success or failure of the cancellation.	N/A
5	The test script executes its SIU Boot Mode Multiple Block Commands phase. The script sends LFILUPLSTART command while a file upload is in progress, and reviews the same fields in boot housekeeping mentioned in the previous steps to confirm that the second LFILUPLSTART command packet is rejected. This verifies that interleaved block commands are rejected. Note: Correct handling of non-interleaved (sequential) block commands is verified in Steps 3 and 4, described above. In those steps, the test confirms that sequentially transmitted File Upload telecommands are properly executed, and that a UPLSTART/UPLCANCEL/UPLSTART sequence is also properly executed. The complete analysis of block command handling based on the results of Steps 3 and 4 is recorded in the output log printed and reviewed at the end of the test.	N/A

Step No.	Description of Step	Step Outcome
6	<p>Next, the test script executes its SIU Boot Mode FILE Command Data Format and Integrity Errors phase.</p> <p>The script confirms that the SIU boot housekeeping telemetry (LBTHKP) packets conform to the CCSDS standard.</p> <p>Note: In Steps 3 and 4, the test verifies that all Boot Mode FILE telecommands transmitted in those steps have the proper data format and integrity and conform to the CCSDS standard. The complete analysis of SIU Boot Mode command and telemetry formatting based on the results of Steps 3, 4, and 6 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
7	<p>Next, the test script executes its SIU Boot Mode FILE Command Format and Parameter Validation phase.</p> <p>Proper execution of telecommands that have the correct format and valid parameters is tested in Steps 3 and 4. The complete analysis of command parameter validation based on the results of Steps 3 and 4 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
8	<p>Next, the test script executes its SIU Boot Mode FILE Command Processing Order phase.</p> <p>In step 3, the script sent all Boot Mode file upload telecommands (LFILUPLSTART, LFILUPLDATA, LFILUPLCOMMIT, LFILUPLCANCEL) in sequence to the SIU. The script examines the LPBCLASTFUNC, LPBCLASTAPID fields and other fields of SIU boot housekeeping telemetry (LBTHKP packets) to confirm that the commands were received and processed in the order in which they were sent. The output of this is reviewed at the end of the test.</p>	N/A
9	<p>Next, the test script executes its SIU Boot Mode FILE Command Completion Status phase.</p> <p>In Steps 3 and 4, the test checks that command completion status is reported by FSW, using the telemetry fields described in those steps. The complete analysis of command completion status notification based on the results of Steps 3 and 4 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A

6.4.2 Part 2: SIU Application Mode Testing

Part 2 of the test exercises Application Mode file upload (FILE) and file management (LFS) functionality on the SIU.

For each of the LFS-related commands, a Command Verification (CmdConfirm) telemetry packet is received, which is further analyzed to verify the status of the reception and execution of these commands. The following fields in the CmdConfirm telemetry are used for analysis: ITC_NODE, ITC_TASK, and ITC_EXESTATUS fields and the reflected CCSDS header of the FILE or LFS command just sent out.

Additionally, the test script sends the ReqDiagPacket (APID:0x650, FC:0) command to request a “demand” version of File System Statistics Housekeeping Packet, DiagFileStats (APID: 0x28A),

which contains statistics about the file management commands just executed. The various LHKD* fields in this packet are used for analysis.

Step No.	Description of Step	Step Outcome
10	<p>The test script advances the SIU FSW to Application Mode by sending the LPBCRTOSEXEC telecommand, which commands the SIU to perform a secondary boot.</p> <p>The script checks that the SIU is operating in Application Mode by detecting whether Application Mode housekeeping telemetry is being transmitted. If not, the script sends the LPBCRTOSEXEC command and checks again.</p> <p>If the SIU FSW cannot be advanced to Application Mode, LTX prints an error message to the screen and exits, aborting the test.</p> <p>If the test is NOT aborted, mark "Complete" for the Step Outcome.</p>	Complete/ Not Complete
11	<p>The test script begins executing its SIU Application Mode File System Status Dump phase.</p> <p>The script sends the LLFSSYSTATUS telecommand to the SIU 4 times, once for each file system device (boot, ram, eeprom0, and eeprom1). The script examines the ITC_EXESTATUS field in CmdConfirm telemetry to confirm that telecommands were properly handled. The script also examines the LLFSSYSBLKTOT (total block size), LLFSSYSBLKSIZE (block size), and LLFSSYSBLKFREE (blocks free) fields in the resulting LLFSSYSLIST telemetry packets to verify that the fields contain valid values.</p>	N/A
12	<p>Next, the test script executes its SIU Application Mode Directory Create phase.</p> <p>The script starts by sending the LLFSDIRDUMP telecommand to the SIU to dump the baseline contents of the root directory on device eeprom0 and the LLFSSYSSTATUS telecommand to independently confirm the initial directory layout of the device. These two commands result in transmission of LLFSROOTLIST and LLFSSYSLIST telemetry packets, respectively, which are examined by the script. Next the script sends the LLFSDIRCREATE telecommand to the SIU to create a directory on eeprom0. The script repeats the LLFSDIRDUMP (on the root directory) and LLFSSYSSTATUS commands to verify that the directory was in fact created. Finally, the script uses the telecommands to attempt to create a directory that already exists, and reviews telemetry to verify that such an operation results in an error.</p> <p>The script repeats this process on device eeprom1.</p> <p>While all of these commands are being sent, the script examines CmdConfirm telemetry to verify that commands are received, dispatched, and executed.</p>	N/A

Step No.	Description of Step	Step Outcome
13	<p>Next, the test script executes its SIU Application Mode File Uploads phase.</p> <p>The test script issues an LFILDIRDUMP command to request the contents of directory created in step 12 on device eeprom0 and confirms in LLFSDIRLIST telemetry that the directory is empty. Next, the test script uploads the file(s) (TBD2.f, TBD3.f, then TBD4.f) listed in Section 6.2 to the SIU eeprom0 to the above mentioned directory by sending the upload sequence of LFILUPLSTART, LFILUPLDATA, and LFILUPLCOMMIT telecommands. The script then sends the LFILDIRDUMP command and reviews the returned LLFSDIRLIST telemetry to verify that the above mentioned directory now in fact contains the files sent with the Upload command sequence.</p> <p>Note: As an additional verification for this step, the script also sends the LLFSFILEDUMPC command to dump the uploaded files back in LLFSDUMPCTDB telemetry packets, confirming that the files were created in the file system.</p> <p>Finally, the script uses the telecommands to attempt to upload a file using a file identifier for a file that is already stored on the device and directory, and reviews telemetry to verify that such an operation results in an error.</p> <p>The sizes of TBD2.f, TBD3.f and TBD4.f are chosen so that the file data spans a small number of upload packets as well as a large number of upload packets.</p> <p>The script repeats this process on device eeprom1.</p> <p>While all of these commands are being sent, the script examines the ITC_EXESTATUS field in CmdConfirm telemetry to verify that the commands are properly handled.</p>	N/A
14	<p>Next, the test script executes its SIU Application Mode File Upload Cancellation phase.</p> <p>The script sends an LLFSDIRDUMP command to confirm the current contents of directory created in step 12.. The test script then sends the LFILUPLSTART and LFILUPLDATA commands to begin a file upload to device eeprom0, followed by the LFILUPLCANCEL command. Next, the script re-sends the LLFSDIRDUMP command and examines the resulting directory listing to verify that the file does not appear in the directory, meaning the upload was cancelled and the file was NOT in fact uploaded.</p> <p>Note: As an additional verification for this step,, the script also sends an LLFSFILEDUMPC command to attempt to dump the contents of the file after the upload sequence is cancelled; if the file upload process was indeed cancelled, this operation should result in an error.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p> <p>The script repeats this process on device eeprom1.</p>	N/A

Step No.	Description of Step	Step Outcome
15	<p>Next, the test script executes its SIU Application Mode File Copying within a Processor phase.</p> <p>As in previous test steps, the script uses LLFSDIRDUMP commands and resulting telemetry to check the contents of a selected destination directory, the directory created in eeprom1 in step 12.. The script then sends the LLFSFILECOPY command to the SIU to copy file TBD2.f from directory created in eeprom0 in step 12 to directory created in eeprom1 in step 12. Next, the script issues LLFSDIRDUMP to confirm that the file now appears in the destination directory,</p> <p>Note: As an additional verification for this step, the script sends LLFSFILEDUMPC to dump and confirm the contents of the copied file.</p> <p>The script also uses LLFSFILECOPY commands to attempt to copy a source file that does not exist, create a destination file that already exists, and copy a file to a non-existent directory, and confirm that such operations result in an error.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
16	<p>Next, the script executes its SIU Application Mode Directory Dump Command phase.</p> <p>The script begins by sending a series of LLFSDIRDUMP commands to dump the contents of the root directory on all devices (boot, ram, eeprom0, and eeprom1) on the SIU. The script examines the resulting LLFSROOTLIST telemetry packets for correct outcomes. Next, the script sends a series of LLFSDIRDUMP commands to dump the contents of various directories created in eeprom0 and eeprom1 during previous steps in the course of the test itself and examines the resulting LLFSDIRLIST telemetry. In all LLFSROOTLIST and LLFSDIRLIST telemetry, the LLFSRDMPFDEV (device identifier), LLFSRDMPFDIR (directory identifier), LLFSRDMPFFILE (file identifier), setting of LLFSRDMPROFLG (read-only flag), LLFSRDMPARCFLG (archive flag setting), LLFSRDMPFSTIME (most recent update time), LLFSRDMPFSSIZE (byte size), and LLFSRDMPFSBLKS (number of blocks) for each subdirectory or file in the requested directory are also verified to contain valid values.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
17	<p>Next, the script executes its SIU Application Mode File Dump Command phase.</p> <p>The script sends a series of LLFSFILEDUMPC telecommands to dump a variety of files, of various sizes, that are uploaded, copied, and deleted in other steps of test execution. The script examines the resulting LLFSDUMPCTDB telemetry to confirm the expected contents of the files; or, in cases where deleted files are being dumped, the scripts confirms that an error is reported. The test covers dumps of boot partition RTOS files that were uploaded in the Boot Mode as well as dumps of Application files that were uploaded in the Application Mode.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	

Step No.	Description of Step	Step Outcome
18	<p>Next, the test script executes its SIU Application Mode File Delete Commands phase.</p> <p>Again, the script sends the LLFSDIRDUMP command to establish the contents of a specified directory.</p> <p>Next, the script sends the LLFSFILEDELETE command to delete all the application-level files created in the previous steps from the directory. The script re-sends the LLFSDIRDUMP command to verify that the file no longer appears in the directory. It also sends the LLFSFILEDUMPC command to attempt to dump the contents of the file that was just deleted; if the delete command was executed properly, this operation should result in an error. The script also confirms that if LLFSFILEDELETE is used to attempt to delete a file that does not exist, an error results.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p> <p>The script verifies correct file deletions on both EEPROM banks, eeprom0 and eeprom1. Only Application Mode files are deleted from the file system. File in the boot device are NOT deleted.</p>	N/A
19	<p>Next, the test script executes its SIU Application Mode Directory Deletion Commands phase.</p> <p>The script begins by sending the LLFSDIRDUMP command to retrieve a root directory listing and determine the current set of directories present on device eeprom0. It also sends the LLFSSYSSTATUS command to confirm which directories currently exist. Next, the script sends the LLFSDIRDELETE command to delete all the directories created in the previous steps, and re-sends the LLFSDIRDUMP and LLFSSYSSTATUS commands to verify the directories have been deleted. The script repeats the process to confirm that use of the LLFSDIRDELETE command on a non-empty directory results in an error, and that use of the command on a non-existent directory results in an error.</p> <p>The script repeats this process on device eeprom1.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
20	<p>Next, the script executes its SIU Application Mode Multiple Block Commands phase.</p> <p>The script then tests interleaved block command handling for file dumps by sending two successive overlapping LLFSFILEDUMPC telecommands and verifying that the second command is rejected; the LLFSDUMPCTDB (file dump telemetry) packets and the ITC_EXESTATUS field of CmdConfirm telemetry are examined to make this determination. Finally, the script tests that interleaving non-block commands with block commands is successful. The script sends an LLFSFILEDUMPC command to EPU0 followed by LLFSDIRCREATE and LLFSDIRDELETE commands and checks the ITC_EXESTATUS field in the CmdConfirm telemetry stream.</p> <p>Note: Correct handling of non-interleaved block commands in Application Mode is verified in Steps 11-19 described above. In those steps, the test confirms that sequentially transmitted (non-overlapping) FILE and LFS telecommands are properly executed and that COMMAND/CANCEL/COMMAND sequences are also properly processed. The complete analysis of block command handling based on the results of Steps 11-19 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A

Step No.	Description of Step	Step Outcome
21	<p>Next, the script executes its SIU Application Mode LFS Telecommand and LFS Telemetry Data Format and Integrity Errors phase.</p> <p>In Steps 11-20, the test verifies that all Application Mode FILE and LFS telecommands transmitted in those steps have the proper data format and conform to the CCSDS standard. Those steps also confirm that the LLFSDIRLIST, LLFSROOTLIST, LLFSSYSLIST, and LLFSDUMPCTDB telemetry packets returned in response to those commands conform to the CCSDS standard. The complete analysis of command and telemetry data format and integrity based on the results of Steps 11-20 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
22	<p>The test script executes its SIU Application Mode LFS Command Format and Parameter Validation phase.</p> <p>Proper execution of telecommands with correct format and valid parameters is tested in Steps 11-20. The complete analysis of command formatting and parameter validation based on the results of Steps 11-20 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
23	<p>Next, the test script executes its SIU Application Mode LFS Command Processing Order phase.</p> <p>The script sends all Application Mode file management telecommands to the SIU. The script verifies the Deque time (ITC_DEQTIMHI and ITC_DEQTIMLO fields) returned in the CmdConfirm telemetry is in the same increasing order as the commands sent from SC.</p>	N/A
24	<p>Next, the test script executes its SIU Application Mode LFS Command Execution Notification phase.</p> <p>In Steps 11-20, after sending each telecommand, the script verifies that FSW sends notification of command execution by printing the Deque (ITC_DEQTIMHI and ITC_DEQTIMLO fields) and Dispatch for Execution (ITC_EXETIMEHI and ITC_EXETIMELO) timestamps received in the CmdConfirm telemetry. The complete analysis of command execution notification based on the results of Steps 11-20 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
25	<p>Next, the test script executes its SIU Application Mode Command Completion Status phase.</p> <p>In Steps 11-20, the test script checks that FSW generates time-tagged telemetry notification of command completion status (success or general cause of failure) by examining the ITC_EXESTATUS field in CmdConfirm telemetry as well as the Timestamp field in the CCSDS header of CmdConfirm telemetry. The complete analysis of command completion status reporting based on the results of Steps 11-20 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
26	<p>Next, the test script executes its SIU Application Mode Command Rates phase.</p> <p>The script issues all Application Mode LFS commands at the specified rate of 20 Hz in conjunction with Magic 7 commands (LLSMSIATTITUDE, LLSMSIANCILLARY, and LLSMSITIMETONE) at the specified rate of 20 Hz and checks the Deque and Execution timestamps (ITC_DEQTIMHI, ITC_DEQTIMLO, ITC_EXETIMEHI, and ITC_EXETIMELO fields) in CmdConfirm telemetry. The script also checks that all commands were successfully received and dispatched for execution by noting the ITC_EXESTATUS field in this telemetry.</p>	N/A

6.4.3 Part 3: EPU0 Boot Mode Testing

The test continues with Boot Mode file management functionality on EPU0. As described below, at this stage, the test procedure moves through a series of steps to place the GASU, PDU , EPU, and instrument power in a “clean” state for testing. Once this is accomplished, the file management testing continues.

Step No.	Description of Step	Step Outcome
27	The test script issues an LLIMLOADSHED command. The SIU reboots and the GASU, PDU, EPU, and instrument power are turned off for a fresh test start.	N/A
28	The test script determines whether the SIU is now operating in Boot Mode as a result of the reboot by detecting whether boot housekeeping telemetry is being transmitted. If not, the script sends the LPBCRESET command and checks again. If the SIU FSW cannot be placed in Boot Mode, LTX prints an error message to the screen and exits, aborting the test. If the test is NOT aborted, mark “Complete” for the Step Outcome.	Complete/ Not Complete
29	The test script advances the SIU FSW to Application Mode by sending the LPBCRTOSEXEC telecommand, which commands the SIU to perform a secondary boot. The script checks that the SIU is operating in Application Mode by detecting whether Application Mode housekeeping telemetry is being transmitted. If not, the script sends the LPBCRTOSEXEC command and checks again. If the SIU FSW cannot be advanced to Application Mode, LTX prints an error message to the screen and exits, aborting the test. If the test is NOT aborted, mark “Complete” for the Step Outcome.	Complete/ Not Complete
30	The script sends the LLIMMAINFEEDON telecommand to power up the PDU and GASU.	Complete/ Not Complete
31	The script then sends the LLIMPOWERON command to power up EPU0. In response to a powerup, EPU0 should begin a primary boot. The test script determines whether EPU0 is now operating in Boot Mode as a result of the powerup and primary boot by detecting whether the EPU is transmitting boot housekeeping telemetry. If not, the script sends the LPBCRESET command and checks again. If the EPU0 FSW cannot be placed in Boot Mode, LTX prints an error message to the screen and exits, aborting the test. If the test is NOT aborted, mark “Complete” for the Step Outcome.	Complete/ Not Complete

Step No.	Description of Step	Step Outcome
32	<p>With FSW on EPU0 advanced to Boot Mode, the test continues with the EPU0 Boot Mode File Uploads phase.</p> <p>The test script uploads the RTOS image file(s) listed in Section 6.2 to the EPU0 by copying the file from the SIU. To do so, the script sends the LFILUPLEPU telecommand, followed by the LFILUPLCOMMIT command. The file is only uploaded to the boot device. The script examines the LPBC0TOTALERRCNT, LPBC0NEXTERRWORD, LPBC0TCREVCNT, LPBC0CACCCNT, and LPBC0LASTERRWORD fields in EPU0 boot housekeeping telemetry (LBTEPU0HKP packets) to confirm success or failure of the copy commands. The script also examines the LPBC0FILESTAT and LPBC0FILEPKTCNT fields in EPU0 boot housekeeping to confirm a successful load.</p>	N/A
33	<p>The test script next executes its EPU0 Boot Mode File Upload Cancellation phase.</p> <p>The test script re-issues the LFILUPLEPU command, then sends the LFILUPLCANCEL command to cancel the copy process. The script examines the LPBC0TOTALERRCNT, LPBC0NEXTERRWORD, LPBC0TCREVCNT, LPBC0CACCCNT, and LPBC0LASTERRWORD fields in EPU0 boot housekeeping telemetry to confirm success or failure. The script also checks the LPBC0FILESTAT and LPBC0FILEPKTCNT fields in EPU0 boot housekeeping telemetry to confirm a success or failure of the cancellation.</p>	N/A
34	<p>The test script executes its EPU0 Boot Mode Multiple Block Commands phase.</p> <p>Correct handling of non-interleaved (sequential) block commands is verified in Steps 32 and 33, described above. In those steps, the test confirms that sequentially transmitted File Upload telecommands are properly executed, and that a UPLEPU/CANCEL/UPLEPU sequence is also properly executed. The complete analysis of block command handling based on the results of Steps 32 and 33 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
35	<p>Next, the test script executes its EPU0 Boot Mode FILE Command Data Format and Integrity Errors phase.</p> <p>The script confirms that EPU0 boot housekeeping telemetry (LBTEPU0HKP) packets conform to the CCSDS standard.</p> <p>Note: In Steps 32 and 33, the test verifies that all Boot Mode FILE telecommands transmitted in those steps have the proper data format and integrity and conform to the CCSDS standard. The complete analysis of EPU0 Boot Mode command and telemetry formatting and integrity based on the results of Steps 32 and 33 is recorded in the output log printed and reviewed at the end of the test</p>	N/A
36	<p>Next, the test script executes its EPU0 Boot Mode FILE Command Format and Parameter Validation phase.</p> <p>Proper execution of telecommands that have the correct format and valid parameters is tested in Steps 32 and 33. The complete analysis of command parameter validation based on the results of Steps 32 and 33 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
37	<p>Next, the test script executes its EPU0 Boot Mode FILE Command Processing Order phase.</p> <p>In Step 32, the script sent all Boot Mode file upload telecommands (LFILUPLSTART, LFILUPLDATA, LFILUPLCOMMIT, LFILUPLCANCEL) in sequence to EPU0. The script examines the LPBC0LASTFUNC, LPBC0LASTAPID fields and other fields of EPU0 boot housekeeping telemetry (LBTEPU0HKP packets) to confirm that the commands were received and processed in the order in which they were sent.. The output of this is reviewed at the end of the test.</p>	N/A

Step No.	Description of Step	Step Outcome
38	<p>Next, the test script executes its EPU0 Boot Mode FILE Command Completion Status phase.</p> <p>In Steps 32 and 33, the test checks that command completion status is reported by FSW, using the telemetry fields described in those steps. The complete analysis of command completion status notification based on the results of Steps 32 and 33 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A

6.4.4 Part 4: EPU0 Application Mode Testing

The test continues, exercising the Application Mode file upload (FIL) and file management (LFS) functionality on EPU0.

For each of the LFS-related commands, a Command Verification (CmdConfirm) telemetry packet is received, which is further analyzed to verify the status of the reception and execution of these commands. The following fields in the CmdConfirm telemetry are used for analysis: ITC_NODE, ITC_TASK, and ITC_EXESTATUS fields and the reflected CCSDS header of the FILE or LFS command just sent out.

Additionally, the test script sends the ReqDiagPacket (APID:0x650, FC:0) command to request a “demand” version of File System Statistics Housekeeping Packet, DiagFileStats (APID: 0x28A), which contains statistics about the file management commands just executed. The various LHKDE0FIL* fields in this packet are used for analysis.

Step No.	Description of Step	Step Outcome
39	<p>The test script advances EPU0 FSW to Application Mode by sending the LPBCRTOSEXEC telecommand, which commands the EPU to perform a secondary boot.</p> <p>The script checks that EPU0 is operating in Application Mode by detecting whether the EPU is transmitting Application Mode housekeeping telemetry. If not, the script sends the LPBCRTOSEXEC command and checks again.</p> <p>If the EPU0 FSW cannot be advanced to Application Mode, LTX prints an error message to the screen and exits, aborting the test.</p> <p>If the test is NOT aborted, mark “Complete” for the Step Outcome.</p>	Complete/ Not Complete

Step No.	Description of Step	Step Outcome
39	<p>The test script begins executing its EPU0 Application Mode File System Status Dump phase.</p> <p>The script sends the LLFSSYSTATUS telecommand to EPU0 4 times, once for each file system device (boot, ram, eeprom0, and eeprom1). The script examines the ITC_EXESTATUS field in CmdConfirm telemetry to confirm that telecommands were properly handled. The script also examines the LLFSSYSSBLKTOT (total block size), LLFSSYSSBLKSIZE (block size), and LLFSSYSSBLKFREE (blocks free) fields in the resulting LLFSSYSLIST telemetry packets to verify that the fields contain valid values.</p> <p>Finally, the script sends an LLFSSYSTATUS telecommand and, while that command is executing, sends a second LLFSSYSTATUS command. The script checks CmdConfirm telemetry to verify that the second command was rejected.</p>	N/A
40	<p>Next, the test script executes its EPU0 Application Mode Directory Create phase.</p> <p>The script starts by sending the LLFSDIRDUMP telecommand to EPU0 to dump the baseline contents of the root directory on device eeprom0 and the LLFSSYSSTATUS telecommand to independently confirm the initial directory layout of the device. These two commands result in transmission of LLFSROOTLIST and LLFSSYSLIST telemetry packets, respectively, which are examined by the script. Next the script sends the LLFSDIRCREATE telecommand to EPU0 to create a directory on eeprom0. The script repeats the LLFSDIRDUMP (on the root directory) and LLFSSYSSTATUS commands to verify that the directory was created. Finally, the script uses the telecommands to attempt to create a directory that already exists, and reviews telemetry to verify that such an operation results in an error.</p> <p>The script repeats this process on device eeprom1.</p> <p>While all of these commands are being sent, the script examines CmdConfirm telemetry to verify that commands are received, dispatched, and executed.</p>	N/A
41	<p>Next, the test script executes its EPU0 Application Mode File Uploads phase.</p> <p>The test script issues an LFILDIRDUMP command to EPU0 to request the contents of directory created in step 40 on device eeprom0 and confirms in LLFSDIRLIST telemetry that the directory is empty. Next, the test script copies the file(s) (TBD2.f, TBD3.f then TBD4.f) listed in Section 6.2 from the SIU to EPU0 by sending the LFILUPEPU and LFILUPLCOMMIT telecommands. The script then re-sends the LFILDIRDUMP command and reviews the returned LLFSDIRLIST telemetry to verify that the dir TBD now in fact contains the files sent with the Copy from SIU to EPU command sequence.</p> <p>Note: As an additional verification for this step, the script also sends the LLFSFILEDUMPC command to dump the uploaded files back in LLFSDUMPCTDB telemetry packets, confirming that the files were created in the file system.</p> <p>Finally, the script uses the telecommands to attempt to upload a file using a file identifier for a file that is already stored on the device and directory, and reviews telemetry to verify that such an operation results in an error.</p> <p>The sizes of TBD2.f, TBD3.f and TBD4.f are chosen so that the file data spans a small number of upload packets as well as a large number of upload packets.</p> <p>The script repeats this process on device eeprom1.</p> <p>While all of these commands are being sent, the script examines the ITC_EXESTATUS field in CmdConfirm telemetry to verify that the commands are properly handled.</p>	N/A

Step No.	Description of Step	Step Outcome
42	<p>Next, the test script executes its EPU0 Application Mode File Upload Cancellation phase.</p> <p>The script sends an LLFSDIRDUMP command to confirm the current contents of directory created on eeprom0 on EPU0. The test script then sends the LFILUPLEPU command to begin a file copy/upload to device eeprom0, followed by the LFILUPLCANCEL command.</p> <p>Next, the script re-sends the LLFSDIRDUMP command and examines the resulting directory listing to verify that the file does not appear in the directory, meaning the copy/upload was cancelled and the file was NOT in fact uploaded to EPU0.</p> <p>Note: As an additional verification for this step,, the script also sends an LLFSFILEDUMPC command to attempt to dump the contents of the file after the upload sequence is cancelled; if the file upload process was indeed cancelled, this operation should result in an error.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p> <p>The script repeats this process on device eeprom1.</p>	N/A
43	<p>Next, the test script executes its EPU0 Application Mode File Copying within a Processor phase.</p> <p>As in previous test steps, the script uses LLFSDIRDUMP commands and resulting telemetry to check the contents of a selected destination directory, the directory created in eeprom1 in step 40.. The script then sends the LLFSFILECOPY command to EPU0 to copy file TBD2.f created in step 41 in directory created in eeprom0 to directory created in eeprom1 in step 40. Next, the script issues LLFSDIRDUMP to confirm that the file now appears in the destination directory,</p> <p>Note: As an additional verification for this step, the script sends LLFSFILEDUMPC to dump and confirm the contents of the copied file.</p> <p>The script also uses LLFSFILECOPY commands to attempt to copy a source file that does not exist, create a destination file that already exists, and copy a file to a non-existent directory, and confirm that such operations result in an error.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
44	<p>Next, the test script executes its EPU0 Application Mode File Copying from SIU to EPU phase.</p> <p>As in previous test steps, the script uses LLFSDIRDUMP commands and resulting telemetry to check the contents of a selected destination directory, created in step 40, on device eeprom0 on EPU0. The scripts then copies the file TBD3.f from the SIU to EPU0 by sending the LFILUPLEPU command and following it with the LFILUPLCOMMIT command. The script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled. The script then issues another LLFSDIRDUMP command to confirm that the file appears in the destination directory. It also sends the LLFSFILEDUMPC command to dump the contents of the copied file over the 1553 interface, and checks the resulting telemetry to verify that the file contents match those of the original file stored on the SIU. The scripts sends additional LFILUPLEPU commands to verify that copying a file to a destination file number that already exists, copying from a source file that does not exist, and copying to an invalid destination directory all result in errors.</p> <p>The script repeats this process copying a file from the SIU to device eeprom1 on EPU0.</p>	N/A

Step No.	Description of Step	Step Outcome
45	<p>Next, the script executes its EPU0 Application Mode Directory Dump Command phase.</p> <p>The script begins by sending a series of LLFSDIRDUMP commands to dump the contents of the root directory on all devices (boot, ram, eeprom0, and eeprom1) on EPU0. The script examines the resulting LLFSROOTLIST telemetry packets for correct outcomes. Next, the script sends a series of LLFSDIRDUMP commands to dump the contents of various directories created in eeprom0 and eeprom1 during the course of the test itself and examines the resulting LLFSDIRLIST telemetry. In all LLFSROOTLIST and LLFSDIRLIST telemetry, the LLFSRDMPFDEV (device identifier), LLFSRDMPFDIR (directory identifier), LLFSRDMPFFILE (file identifier), setting of LLFSRDMPROFLG (read-only flag), LLFSRDMPARCFLG (archive flag setting), LLFSRDMPFSTIME (most recent update time), LLFSRDMPFSSIZE (byte size), and LLFSRDMPFSBLKS (number of blocks) for each subdirectory or file in the requested directory are also verified to contain valid values.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
46	<p>Next, the script executes its EPU0 Application Mode File Dump Command phase.</p> <p>The script sends a series of LLFSFILEDUMPC telecommands to dump a variety of files, of various sizes, that are uploaded, copied, and deleted in other steps of test execution. The script examines the resulting LLFSDUMPCTDB telemetry to confirm the expected contents of the files. The test verifies the dump of boot partition RTOS files that were uploaded in the Boot Mode as well as dump of Application files that were uploaded in the Application Mode.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
47	<p>Next, the test script executes its EPU0 Application Mode File Delete Commands phase.</p> <p>Again, the script sends the LLFSDIRDUMP command to determine the contents of the directories created in the previous steps, on EPU0. Next, the script sends the LLFSFILEDELETE command to delete all the files created in the previous steps from the directory. The script re-sends the LLFSDIRDUMP command to verify that the file no longer appears in the directory. It also sends the LLFSFILEDUMPC command to attempt to dump the contents of the file that was just deleted; if the delete command was executed properly, this operation should result in an error. The script also confirms that if LLFSFILEDELETE is used to attempt to delete a file that does not exist, an error results.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p> <p>The script verifies correct file deletions on both EEPROM banks, eeprom0 and eeprom1. Only Application Mode files are deleted from the file system. Files from the boot device are NOT deleted.</p>	N/A

Step No.	Description of Step	Step Outcome
48	<p>Next, the test script executes its EPU0 Application Mode Directory Deletion Commands phase.</p> <p>The script begins by sending the LLFSDIRDUMP command to retrieve a root directory listing on device eeprom0 on EPU0 and determine the current set of directories present on device eeprom0. It also sends the LLFSSYSSTATUS command to confirm which directories currently exist. Next, the script sends the LLFSDIRDELETE command to delete all directories created in the previous steps on EPU0, and re-sends the LLFSDIRDUMP and LLFSSYSSTATUS commands to verify the directories have been deleted. The script repeats the process to confirm that use of the LLFSDIRDELETE command on a non-empty directory results in an error, and use of the command on a non-existent directory results in an error.</p> <p>The script repeats this process on device eeprom1.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
49	<p>Next, the script executes its EPU0 Application Mode Multiple Block Commands phase.</p> <p>The script then tests interleaved (overlapping) block command handling for file dumps by sending two successive LLFSFILEDUMPC telecommands to EPU0 and verifying that the second command is rejected; the LLFSDUMPCTDB (file dump telemetry) packets and the ITC_EXESTATUS field of CmdConfirm telemetry are examined to make this determination. Finally, the script tests that interleaving non-block commands with block commands is successful. The script sends an LLFSFILEDUMPC command to EPU0 followed by LLFSDIRCREATE and LLFSDIRDELETE commands and checks the ITC_EXESTATUS field in the CmdConfirm telemetry stream.</p> <p>Note: Correct handling of non-interleaved block commands in Application Mode is verified in Steps 39-48 described above. In those steps, the test confirms that sequentially transmitted (non-overlapping) FILE and LFS telecommands are properly executed and that COMMAND/CANCEL/COMMAND sequences are also properly processed. The complete analysis of block command handling based on the results of Steps 39-48 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
50	<p>Next, the script executes its EPU0 Application Mode LFS Telecommand and LFS Telemetry Data Format and Integrity Errors phase.</p> <p>In Steps 39-48, the test verifies that all Application Mode FILE and LFS telecommands transmitted in those steps have the proper data format and conform to the CCSDS standard. Those steps also confirm that the LLFSDIRLIST, LLFSROOTLIST, LLFSSYSLIST, and LLFSDUMPCTDB telemetry packets returned in response to those commands conform to the CCSDS standard. The complete analysis of command and telemetry data format and integrity based on the results of Steps 39-48 is recorded in the output log printed and reviewed at the end of the test</p>	N/A
51	<p>The test script executes its EPU0 Application Mode LFS Command Format and Parameter Validation phase.</p> <p>Proper execution of telecommands with correct format and valid parameters is tested in Steps 39-48. The complete analysis of command formatting and parameter validation based on the results of Steps 39-48 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A

Step No.	Description of Step	Step Outcome
52	<p>Next, the test script executes its EPU0 Application Mode LFS Command Processing Order phase.</p> <p>The script sends all Application Mode file management telecommands to EPU0. The script verifies the Deque time (ITC_DEQTIMHI and ITC_DEQTIMLO fields) returned in the CmdConfirm telemetry is in the same increasing order as the commands sent from SC.</p>	N/A
53	<p>Next, the test script executes its EPU0 Application Mode LFS Command Execution Notification phase.</p> <p>In Steps 39-48, after sending each telecommand, the script verifies that FSW sends notification of command execution by printing the Deque (ITC_DEQTIMHI and ITC_DEQTIMLO fields) and Dispatch for Execution (ITC_EXETIMEHI and ITC_EXETIMELO) timestamps received in the CmdConfirm telemetry. The complete analysis of command execution notification based on the results of Steps 39-48 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
54	<p>Next, the test script executes its EPU0 Application Mode Command Completion Status phase.</p> <p>In Steps 39-48, the test script checks that FSW is generates time-tagged telemetry notification of command completion status (success or general cause of failure) by examining the ITC_EXESTATUS field in CmdConfirm telemetry as well as the Timestamp field in the CCSDS header of CmdConfirm telemetry. The complete analysis of command completion status reporting based on the results of Steps 39-48 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
55	<p>The test script sends the LPBCRESET command to EPU0 to reboot that unit. It then powers down EPU0 by sending the LLIMPOWEROFF command.</p>	N/A

6.4.5 Part 5: EPU1 Boot Mode Testing

The test continues with Boot Mode file management functionality on EPU1. EPU0 is rebooted for a clean exit from the test and is then shut down.

Step No.	Description of Step	Step Outcome
56	<p>The script then sends the LLIMPOWERON command to power up EPU1. In response to a powerup, EPU1 should begin a primary boot.</p> <p>The test script determines whether EPU1 is now operating in Boot Mode as a result of the powerup and primary boot by detecting whether EPU1 is transmitted boot housekeeping telemetry. If not, the script sends the LPBCRESET command and checks again.</p> <p>If the EPU1 FSW cannot be placed in Boot Mode, LTX prints an error message to the screen and exits, aborting the test.</p> <p>If the test is NOT aborted, mark "Complete" for the Step Outcome.</p>	Complete/ Not Complete

Step No.	Description of Step	Step Outcome
57	<p>With FSW on EPU1- advanced to Boot Mode, the test continues with the the test script executes its EPU1 Boot Mode File Uploads phase.</p> <p>The test script uploads the RTOS image file(s) listed in Section 6.2 to the EPU1 by copying the file from the SIU. To do so, the script sends the LFILUPLEPU telecommand, followed by the LFILUPLCOMMIT command. The file is only uploaded to the boot device. The script examines the LPBC1TOTALERRCNT, LPBC1NEXTERRWORD, LPBC1TCREVCNT, LPBC1TACCNT, and LPBC1LASTERRWORD fields in EPU1 boot housekeeping telemetry (LBTEPU1HKP packets) to confirm success or failure of the copy commands. The script also examines the LPBC1FILESTAT and LPBC1FILEPKTCNT fields in EPU1 boot housekeeping to confirm a successful load.</p>	N/A
58	<p>The test script next executes its EPU1 Boot Mode File Upload Cancellation phase.</p> <p>The test script re-issues the LFILUPLEPU command, then sends the LFILUPLCANCEL command to cancel the copy process. The script examines the LPBC1TOTALERRCNT, LPBC1NEXTERRWORD, LPBC1TCREVCNT, LPBC1TACCNT, and LPBC1LASTERRWORD fields in EPU1 boot housekeeping telemetry to confirm success or failure. The script also checks the LPBC1FILESTAT and LPBC1FILEPKTCNT fields in EPU1 boot housekeeping telemetry to confirm a success or failure of the cancellation.</p>	N/A
59	<p>The test script executes its EPU1 Boot Mode Multiple Block Commands phase.</p> <p>Correct handling of non-interleaved (sequential) block commands is verified in Steps 57 and 58, described above. In those steps, the test confirms that sequentially transmitted File Upload telecommands are properly executed, and that a UPLEPU/CANCEL/UPLEPU sequence is also properly executed. The complete analysis of block command handling based on the results of Steps 57 and 58 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
60	<p>Next, the test script executes its EPU1 Boot Mode FILE Command Data Format and Integrity Errors phase.</p> <p>The script confirms that EPU1 boot housekeeping telemetry (LBTEPU1HKP) packets conform to the CCSDS standard.</p> <p>Note: In Steps 57 and 58, the test verifies that all Boot Mode FILE telecommands transmitted in those steps have the proper data format and integrity and conform to the CCSDS standard. The complete analysis of EPU1 Boot Mode command and telemetry formatting and integrity based on the results of Steps 57 and 58 is recorded in the output log printed and reviewed at the end of the test</p>	N/A
61	<p>Next, the test script executes its EPU1 Boot Mode FILE Command Format and Parameter Validation phase.</p> <p>Proper execution of telecommands that have the correct format and valid parameters is tested in Steps 57 and 58. The complete analysis of command parameter validation based on the results of Steps 57 and 58 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
62	<p>Next, the test script executes its EPU1 Boot Mode FILE Command Processing Order phase.</p> <p>In Step 57, the script sent all Boot Mode file upload telecommands (LFILUPLSTART, LFILUPLDATA, LFILUPLCOMMIT, LFILUPLCANCEL) in sequence to EPU1. The script examines the LPBC1LASTFUNC, LPBC1LASTAPID fields and other fields of EPU1 boot housekeeping telemetry (LBTEPU1HKP packets) to confirm that the commands were received and processed in the order in which they were sent. The output of this is reviewed at the end of the test.</p>	N/A

Step No.	Description of Step	Step Outcome
63	<p>Next, the test script executes its EPU1 Boot Mode FILE Command Completion Status phase.</p> <p>In Steps 57 and 58, the test checks that command completion status is reported by FSW, using the telemetry fields described in those steps. The complete analysis of command completion status notification based on the results of Steps 57 and 58 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A

6.4.6 Part 6: EPU1 Application Mode Testing

The test continues, exercising the Application Mode file upload (FIL) and file management (LFS) functionality on EPU1.

For each of the LFS-related commands, a Command Verification (CmdConfirm) telemetry packet is received, which is further analyzed to verify the status of the reception and execution of these commands. The following fields in the CmdConfirm telemetry are used for analysis: ITC_NODE, ITC_TASK, and ITC_EXESTATUS fields and the reflected CCSDS header of the FILE or LFS command just sent out.

Additionally, the test script sends the ReqDiagPacket (APID:0x650, FC:0) command to request a “demand” version of File System Statistics Housekeeping Packet, DiagFileStats (APID: 0x28A), which contains statistics about the file management commands just executed. The various LHKDE1FIL* fields in this packet are used for analysis.

Step No.	Description of Step	Step Outcome
64	<p>The test script advances EPU1 FSW to Application Mode by sending the LPBCRTOSEXEC telecommand, which commands the EPU to perform a secondary boot.</p> <p>The script checks that EPU1 is operating in Application Mode by detecting whether EPU1 is transmitting Application Mode housekeeping telemetry. If not, the script sends the LPBCRTOSEXEC command and checks again.</p> <p>If the EPU1 FSW cannot be advanced to Application Mode, LTX prints an error message to the screen and exits, aborting the test.</p> <p>If the test is NOT aborted, mark “Complete” for the Step Outcome.</p>	Complete/ Not Complete

Step No.	Description of Step	Step Outcome
65	<p>The test script begins executing its EPU1 Application Mode File System Status Dump phase.</p> <p>The script sends the LLFSSYSTATUS telecommand to EPU1 4 times, once for each file system device (boot, ram, eeprom0, and eeprom1). The script examines the ITC_EXESTATUS field in CmdConfirm telemetry to confirm that telecommands were properly handled. The script also examines the LLFSSYSSBLKTOT (total block size), LLFSSYSSBLKSIZE (block size), and LLFSSYSSBLKFREE (blocks free) fields in the resulting LLFSSYSLIST telemetry packets to verify that the fields contain valid values.</p> <p>Finally, the script sends an LLFSSYSTATUS telecommand and, while that command is executing, sends a second LLFSSYSTATUS command. The script checks CmdConfirm telemetry to verify that the second command was rejected.</p>	N/A
66	<p>Next, the test script executes its EPU1 Application Mode Directory Create phase.</p> <p>The script starts by sending the LLFSDIRDUMP telecommand to EPU1 to dump the baseline contents of the root directory on device eeprom0 and the LLFSSYSSTATUS telecommand to independently confirm the initial directory layout of the device. These two commands result in transmission of LLFSROOTLIST and LLFSSYSLIST telemetry packets, respectively, which are examined by the script. Next the script sends the LLFSDIRCREATE telecommand to EPU1 to create a directory on eeprom0. The script repeats the LLFSDIRDUMP (on the root directory) and LLFSSYSSTATUS commands to verify that the directory was created. Finally, the script uses the telecommands to attempt to create a directory that already exists, and reviews telemetry to verify that such an operation results in an error.</p> <p>The script repeats this process on device eeprom1.</p> <p>While all of these commands are being sent, the script examines CmdConfirm telemetry to verify that commands are received, dispatched, and executed.</p>	N/A
67	<p>Next, the test script executes its EPU1 Application Mode File Uploads phase.</p> <p>The test script issues an LFILDIRDUMP command to EPU1 to request the contents of directory created in step 66 on device eeprom0 and confirms in LLFSDIRLIST telemetry that the directory is empty. Next, the test script copies the file(s) (TBD2.f, TBD3.f then TBD4.f) listed in Section 6.2 from the SIU to EPU0 by sending the LFILUPEPU and LFILUPLCOMMIT telecommands. The script then re-sends the LFILDIRDUMP command and reviews the returned LLFSDIRLIST telemetry to verify that the dir TBD now in fact contains the files sent with the Copy from SIU to EPU command sequence.</p> <p>Note: As an additional verification for this step, the script also sends the LLFSFILEDUMPC command to dump the uploaded files back in LLFSDUMPCTDB telemetry packets, confirming that the files were created in the file system.</p> <p>Finally, the script uses the telecommands to attempt to upload a file using a file identifier for a file that is already stored on the device and directory, and reviews telemetry to verify that such an operation results in an error.</p> <p>The sizes of TBD2.f, TBD3.f and TBD4.f are chosen so that the file data spans a small number of upload packets as well as a large number of upload packets.</p> <p>The script repeats this process on device eeprom1.</p> <p>While all of these commands are being sent, the script examines the ITC_EXESTATUS field in CmdConfirm telemetry to verify that the commands are properly handled.</p>	N/A

Step No.	Description of Step	Step Outcome
68	<p>Next, the test script executes its EPU1 Application Mode File Upload Cancellation phase.</p> <p>The script sends an LLFSDIRDUMP command to confirm the current contents of directory created on eeprom0 on EPU1. The test script then sends the LFILUPLEPU command to begin a file copy/upload to device eeprom0, followed by the LFILUPLCANCEL command.</p> <p>Next, the script re-sends the LLFSDIRDUMP command and examines the resulting directory listing to verify that the file does not appear in the directory, meaning the copy/upload was cancelled and the file was NOT in fact uploaded to EPU1.</p> <p>Note: As an additional verification for this step,, the script also sends an LLFSFILEDUMPC command to attempt to dump the contents of the file after the upload sequence is cancelled; if the file upload process was indeed cancelled, this operation should result in an error.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p> <p>The script repeats this process on device eeprom1.</p>	N/A
69	<p>Next, the test script executes its EPU1 Application Mode File Copying within a Processor phase.</p> <p>As in previous test steps, the script uses LLFSDIRDUMP commands and resulting telemetry to check the contents of a selected destination directory, the directory created in eeprom1 in step 66.. The script then sends the LLFSFILECOPY command to EPU1 to copy file TBD2.f created in step 67 in directory created in eeprom0 to directory created in eeprom1 in step 66. Next, the script issues LLFSDIRDUMP to confirm that the file now appears in the destination directory,</p> <p>Note: As an additional verification for this step, the script sends LLFSFILEDUMPC to dump and confirm the contents of the copied file.</p> <p>The script also uses LLFSFILECOPY commands to attempt to copy a source file that does not exist, create a destination file that already exists, and copy a file to a non-existent directory, and confirm that such operations result in an error.</p> <p>The script repeats this process on device eeprom1, the repeats the process to copy between device eeprom0 and eeprom1.</p>	N/A
70	<p>Next, the test script executes its EPU1 Application Mode File Copying from SIU to EPU phase.</p> <p>As in previous test steps, the script uses LLFSDIRDUMP commands and resulting telemetry to check the contents of a selected destination directory, created in step 66, on device eeprom0 on EPU1. The scripts then copies the file TBD3.f from the SIU to EPU1 by sending the LFILUPLEPU command and following it with the LFILUPLCOMMIT command. The script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled. The script then issues another LLFSDIRDUMP command to confirm that the file appears in the destination directory. It also sends the LLFSFILEDUMPC command to dump the contents of the copied file over the 1553 interface, and checks the resulting telemetry to verify that the file contents match those of the original file stored on the SIU. The scripts sends additional LFILUPLEPU commands to verify that copying a file to a destination file number that already exists, copying from a source file that does not exist, and copying to an invalid destination directory all result in errors.</p> <p>The script repeats this process copying a file from the SIU to device eeprom1 on EPU1.</p>	N/A

Step No.	Description of Step	Step Outcome
71	<p>Next, the script executes its EPU1 Application Mode Directory Dump Command phase.</p> <p>The script begins by sending a series of LLFSDIRDUMP commands to dump the contents of the root directory on all devices (boot, ram, eeprom0, and eeprom1) on EPU1. The script examines the resulting LLFSROOTLIST telemetry packets for correct outcomes. Next, the script sends a series of LLFSDIRDUMP commands to dump the contents of various directories created in eeprom0 and eeprom1 during the course of the test itself and examines the resulting LLFSDIRLIST telemetry. In all LLFSROOTLIST and LLFSDIRLIST telemetry, the LLFSRDMPFDEV (device identifier), LLFSRDMPFDIR (directory identifier), LLFSRDMPFFILE (file identifier), setting of LLFSRDMPROFLG (read-only flag), LLFSRDMPARCFLG (archive flag setting), LLFSRDMPFSTIME (most recent update time), LLFSRDMPFSSIZE (byte size), and LLFSRDMPFSBLKS (number of blocks) for each subdirectory or file in the requested directory are also verified to contain valid values.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
72	<p>Next, the script executes its EPU1 Application Mode File Dump Command phase.</p> <p>The script sends a series of LLFSFILEDUMPC telecommands to dump a variety of files, of various sizes, that are uploaded, copied, and deleted in other steps of test execution. The script examines the resulting LLFSDUMPCTDB telemetry to confirm the expected contents of the files. The test verifies the dump of boot partition RTOS files that were uploaded in the Boot Mode as well as dump of Application files that were uploaded in the Application Mode.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
73	<p>Next, the test script executes its EPU1 Application Mode File Delete Commands phase.</p> <p>Again, the script sends the LLFSDIRDUMP command to determine the contents of a directories created in the previous steps, on EPU1. Next, the script sends the LLFSFILEDELETE command to delete all files created in the previous steps on EPU1 from the directory. The script re-sends the LLFSDIRDUMP command to verify that the file no longer appears in the directory. It also sends the LLFSFILEDUMPC command to attempt to dump the contents of the file that was just deleted; if the delete command was executed properly, this operation should result in an error. The script also confirms that if LLFSFILEDELETE is used to attempt to delete a file that does not exist, an error results.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p> <p>The script verifies correct file deletions on both EEPROM banks, eeprom0 and eeprom1. Only Application Mode files are deleted from the file system. Files from the boot device are NOT deleted.</p>	N/A

Step No.	Description of Step	Step Outcome
74	<p>Next, the test script executes its EPU1 Application Mode Directory Deletion Commands phase.</p> <p>The script begins by sending the LLFSDIRDUMP command to retrieve a root directory listing on device eeprom0 on EPU1 and determine the current set of directories present on device eeprom0. It also sends the LLFSSYSSTATUS command to confirm which directories currently exist. Next, the script sends the LLFSDIRDELETE command to delete all directories created in the previous steps on EPU1 and re-sends the LLFSDIRDUMP and LLFSSYSSTATUS commands to verify the directories have been deleted. The script repeats the process to confirm that use of the LLFSDIRDELETE command on a non-empty directory results in an error, and use of the command on a non-existent directory results in an error.</p> <p>The script repeats this process on device eeprom1.</p> <p>In all cases, the script checks for the SUCCESS status code in the ITC_EXESTATUS field of CmdConfirm telemetry to verify that commands were properly handled.</p>	N/A
75	<p>Next, the script executes its EPU1 Application Mode Multiple Block Commands phase.</p> <p>The script then tests interleaved (overlapping) block command handling for file dumps by sending two successive LLFSFILEDUMPC telecommands to EPU1 and verifying that the second command is rejected; the LLFSDUMPCTDB (file dump telemetry) packets and the ITC_EXESTATUS field of CmdConfirm telemetry are examined to make this determination. Finally, the script tests that interleaving non-block commands with block commands is successful. The script sends an LLFSFILEDUMPC command to EPU1 followed by LLFSDIRCREATE and LLFSDIRDELETE commands and checks the ITC_EXESTATUS field in the CmdConfirm telemetry stream.</p> <p>Note: Correct handling of non-interleaved block commands in Application Mode is verified in Steps 65-74 described above. In those steps, the test confirms that sequentially transmitted (non-overlapping) FILE and LFS telecommands are properly executed and that COMMAND/CANCEL/COMMAND sequences are also properly processed. The complete analysis of block command handling based on the results of Steps 64-73 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
76	<p>Next, the script executes its EPU1 Application Mode LFS Telecommand and LFS Telemetry Data Format and Integrity Errors phase.</p> <p>In Steps 65-74, the test verifies that all Application Mode FILE and LFS telecommands transmitted in those steps have the proper data format and conform to the CCSDS standard. Those steps also confirm that the LLFSDIRLIST, LLFSROOTLIST, LLFSSYSLIST, and LLFSDUMPCTDB telemetry packets returned in response to those commands conform to the CCSDS standard. The complete analysis of command and telemetry data format and integrity based on the results of Steps 65-74 is recorded in the output log printed and reviewed at the end of the test</p>	N/A
77	<p>The test script executes its EPU1 Application Mode LFS Command Format and Parameter Validation phase.</p> <p>Proper execution of telecommands with correct format and valid parameters is tested in Steps 65-74. The complete analysis of command formatting and parameter validation based on the results of Steps 65-74 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A

Step No.	Description of Step	Step Outcome
78	<p>Next, the test script executes its EPU1 Application Mode LFS Command Processing Order phase.</p> <p>The script sends all Application Mode file management telecommands to EPU1. The script verifies the Deque time (ITC_DEQTIMHI and ITC_DEQTIMLO fields) returned in the CmdConfirm telemetry is in the same increasing order as the commands sent from SC.</p>	N/A
79	<p>Next, the test script executes its EPU1 Application Mode LFS Command Execution Notification phase.</p> <p>In Steps 64-73, after sending each telecommand, the script verifies that FSW sends notification of command execution by printing the Deque (ITC_DEQTIMHI and ITC_DEQTIMLO fields) and Dispatch for Execution (ITC_EXETIMEHI and ITC_EXETIMELO) timestamps received in the CmdConfirm telemetry. The complete analysis of command execution notification based on the results of Steps 65-74 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
80	<p>Next, the test script executes its EPU1 Application Mode Command Completion Status phase.</p> <p>In Steps 64-73, the test script checks that FSW is generates time-tagged telemetry notification of command completion status (success or general cause of failure) by examining the ITC_EXESTATUS field in CmdConfirm telemetry as well as the Timestamp field in the CCSDS header of CmdConfirm telemetry. The complete analysis of command completion status reporting based on the results of Steps 65-74 is recorded in the output log printed and reviewed at the end of the test.</p>	N/A
81	<p>The test script sends the LPBCRESET command to EPU1 to reboot that unit. It then powers down EPU1 by sending the LLIMPOWEROFF command.</p>	N/A
82	<p>The script next sends the LLIMLOADSHED telecommand to power off the GASU and PDU.</p>	N/A
83	<p>The script then checks whether the SIU FSW is now operating in Boot Mode. If not, the script sends the LPBCRESET command and checks again.</p> <p>If the SIU FSW cannot be placed in Boot Mode, LTX exits.</p> <p>Confirm that the script has placed the SIU in Boot Mode ready for another test.</p>	Complete/ Not Complete

6.4.7 Part 7: Review of Test Outputs

This final part of the test covers detailed review of the FILMGT script output files and verification of PASS/FAIL criteria.

Step No.	Description of Step	Step Outcome
83	<p>After the FILMGT test script has executed all of the sub-tests described in the previous sections, the script sends a single, large log file to the printer.</p> <p>Retrieve the output file from the printer, label it, and attach it to this test procedure. Record the label written on the printed log file: _____</p> <p>Record the full path and filename of the soft copy of this output file: _____</p>	Complete/ Not Complete
84	<p>Review the log file. The file contains detailed outputs of the test script, with the names of telecommands being issued, telemetry packets being received, telemetry fields being examined, and status summaries of each major step in the test clearly called out for review.</p> <p>The following log file entries appear interspersed through the log file, followed by "PASS" or "FAIL".</p> <p>Boot Mode File Upload to SIU, EPU0 and EPU1</p> <p>Boot Mode File Upload Cancellations to SIU, EPU0 and EPU1</p> <p>Boot Mode Multiple Block Commands</p> <p>Boot Mode FILE command Data Format and Integrity Errors</p> <p>Boot Mode FILE Command Format and Parameter Validation</p> <p>Boot Mode FILE Command Processing order</p> <p>Boot Mode FILE Command Completion Status</p> <p>FILMGT_001(1): Passed</p> <p>Confirm that each of the above entries appears in the log file with a "PASS" indication.</p> <p>This test verifies completion of all test sub-objectives in the Boot Mode.</p>	Pass/ Fail

Step No.	Description of Step	Step Outcome
85	<p>Review the log file. The file contains detailed outputs of the test script, with the names of telecommands being issued, telemetry packets being received, telemetry fields being examined, and status summaries of each major step in the test clearly called out for review.</p> <p>The following log file entries appear interspersed through the log file, followed by “PASS” or “FAIL”.</p> <p>Application Mode File System Status Dump Command</p> <p>Application Mode Directory Dump Command</p> <p>Application Mode File Dump Command</p> <p>Application Mode File Upload Command</p> <p>Application Mode File Upload Cancellation Command</p> <p>Application Mode File Copy within Processor Command</p> <p>Application Mode File Copy from SIU to EPU0 and SIU to EPU1 commands.</p> <p>Application Mode File Delete command.</p> <p>Application Mode Directory Delete Command.</p> <p>Application Mode Multiple Block Commands</p> <p>Application Mode File System Status Dump Command</p> <p>Application Mode LFS command and LFS Telemetry Data Format and Integrity Errors</p> <p>Application Mode LFS Command Format and Parameter Validation</p> <p>Application Mode LFS Command Processing order</p> <p>Application Mode LFS Command Execution Notification</p> <p>Application Mode LFS Command Completion Status</p> <p>FILMGT_001(2): Passed</p> <p>Confirm that each of the above entries appears in the log file with a “PASS” indication.</p> <p>This test verifies completion of all test sub-objectives in the Application Mode</p>	Pass/ Fail

Initial to confirm.

Date Time Test Engineer QAE

7. TEST POST CONDITIONS AND OVERALL OUTCOME

7.1 Test Post-Conditions

The following post-conditions are analyzed and verified by the test script as described in “Test Procedure and Test Analysis”:

No.	Post-Condition	Post-Condition Met? (Yes/No)
1	SIU in Boot Mode	
2	File System unchanged. Boot Partition File TBD uploaded. At the end of this test procedure, the file system is restored to its original state. All files and directories that were created are deleted. The boot mode RTOS images are modified, however.	

The Test Engineer and Quality Assurance Engineer verify that all test post-conditions are met.

Date

Time

Test Engineer

QAE

7.2 Overall Outcome of FILMGT_001

Based on the analysis of the test results, the overall outcome of Test FILMGT_001 is as follows:

Passed - all of the expected outcomes for the test were confirmed

Failed - one or more of the test outcomes were not confirmed

Date

Test Engineer

QAE

8. CERTIFICATION

I certify that the information obtained under this test procedure is as represented and the information recorded in this document is complete and correct. Any deviations from test procedures described herein are identified in Appendix A.

Date Test Engineer (Print Name) Test Engineer (Signature)

I certify that the information obtained through execution of this test procedure is as represented and the information recorded in this document is complete and correct. Execution of the test, storage of the results, and verification of outcomes were carried out in accordance with quality standards defined in the GLAST Quality Manual (LAT-MD-00091).

Date Software QA Engineer (Print Name) Software QA Engineer (Sign)

I certify that the information obtained under this test procedure is as represented and the information recorded in this document is complete and correct. The test procedure, as designed and executed, does indeed verify that the FSW functionality under test satisfies the corresponding requirements from the Flight Software Specification – Level III.

Date FSW Manager (Print Name) FSW Manager (Signature)

APPENDIX A: DEVIATIONS FROM THE QUALIFICATION TEST PROCEDURE

This section details any deviations from the hardware configuration, software configuration, or test procedure followed during the execution of the test or tests described in this Qualification Test Procedure document. All deviations from the approved procedure are agreed to by the Test Engineer and the Software Quality Engineer during the test execution session. All deviations must be reported during the Post Qualification Test Review, where their impact on the test results will be evaluated.

Hardware Deviations

Describe any deviations from the hardware configuration defined in Section 5.1. Name the hardware that was modified and describe the modifications. If hardware is *replaced* during execution of the test, name the replaced hardware, the manufacturer, and list an identification number (e.g., GLAT ID number).

Software Deviations

Describe any changes made to the software configuration under test or the software configuration used to support test execution, as defined in Section 5.2. Give version numbers of all FSW packages and test packages that were modified. Describe how the contents of the modified software load were verified. Describe these deviations for each test that was modified.

Procedural Deviations

Specify any deviations from the test procedure for the test being executed. If this document contains more than one test procedure, list the procedure by number (e.g., “FILMGT_001”). List by number the steps modified or skipped. Provide a numbered sequence listing any added steps. Describe these deviations for each test that was modified.
