Code 582
Flight Software Branch

CORE FLIGHT EXECUTIVE BUILD 6.5.0

FLIGHT SOFTWARE BUILD VERIFICATON TEST REPORT

Flight Software Branch – Code 582

Version 1.0

SIGNATURES

Submitted by:

WALTER MOLESKI

Digitally signed by WALTER MOLESKI
DN: c=US, o=U.S. Government, ou=NASA, ou=PIV, cn=WALTER MOLESKI, 0.9.2342.19200300.100.1.1=wmoleski
Date: 2016.06.06 06:54:53 -04'00'

_____ _____
Walt Moleski/GSFC-582.0 Date
core Flight System (cFS) Flight Software Tester

Approved by:

Susanne Strege

SUSANNE STREGE

Digitally signed by SUSANNE STREGE
DN: c=US, o=U.S. Government, ou=NASA, ou=PIV, cn=SUSANNE STREGE, 0.9.2342.19200300.100.1.1=sstrege
Date: 2016.06.02 14:34:16 -04'00'

_____ _____
Susanne Strege/GSFC-582.0 Date
core Flight System (cFS) Flight Software Product Development Lead

PLAN UPDATE HISTORY

Version	Date	Description	Affected Pages
1.0		cFE build 6.5.0.0 verification test report	all

TABLE OF CONTENTS

1 INTRODUCTION

1.1 DOCUMENT PURPOSE

This Test Report describes the test results from the core Flight Executive (cFE) Flight Software (FSW) Test Team build 6.5.0.0 verification testing. It is used to verify that the cFE FSW has been tested in a manner that validates that it satisfies the functional and performance requirements defined within the cFE FSW Requirements Specification and all Discrepancy/Change Request (DCR)/Trac Ticket fixes and code updates assigned to build 6.5.0.0. This Test Report summarizes the FSW test history, the build verification process, the build test configuration, and the test execution and results

1.2 APPLICABLE DOCUMENTS

Unless otherwise stated, these documents refer to the latest version.

Parent Documents (Mission and FSW)

- 582-2000-012 FSB Flight Software TestBed Requirements Guidelines

Reference Documents

All of the references below can be found on the Code 582 internal website at https://fsb.gsfc.nasa.gov/

- 582-2003-001 FSB FSW Test Plan Template
- 582-2004-001 FSB FSW Test Description Template
- 582-2004-002 FSB FSW Test Scenario Template
- 582-2004-003 FSB FSW Test Procedure Template
- 582-2004-004 FSB FSW Test Execution Summary Template
- 582-2004-005 FSB Test Product Peer Review Form
- 582-2000-002 FSB FSW Unit Test Standard
- 582-2007-040 FSB Test Analysis Summary Template
- 582-2008-006 FSB Testbed Validation Description

1.3 DOCUMENT ORGANIZATION

Section 1 of this document presents some introductory material.

Section 2 provides a flight software overview and context along with the test history and testing overview.

Section 3 describes the build verification process including procedure development and execution and test products produced.

Section 4 describes the build test configuration which includes an overview of the testbed and the requirements verification matrix.

Section 5 describes the test execution and results by subsystem.

5.6.1 provides the Requirements Traceability Matrix

Appendix A - provides the Command, Telemetry, and Events Verification Matrix

1.4 DEFINITIONS

There were 3 verifications methods used during build verification testing. They were:

- Demonstration: Show compliance with system requirement by exhibiting the required capability (e.g. by demonstrating interactive capability, display capability, print capability, etc.
- Inspection: Show compliance with a system requirement by visual verification of the software (e.g. verifying preparation for delivery, proper interfacing)
- Analysis: Perform detailed analysis of code, generated data (both intermediate data and final output data), etc., to determine compliance with system requirements.

The fields in the Requirements Verification Matrix in Section 4.3 are defined as follows:

- Requirements Tested Passed: Requirement was fully tested in a build test procedure and passed all tests.
- Requirements Tested Failed: Requirement was fully tested in a build test procedure and failed one or more aspect of the testing.
- Requirements Tested Partially: Requirement was tested partially in a build test procedure. To be fully tested, the partially tested requirement is either tested additionally in one or more other test procedures within the same build **and/or** other aspects of the requirement must be tested in a later build, due to capabilities not present in the current build
- Total Tested: Total number of requirements fully tested in a build test procedure. Includes total passed and total failed, but does **not** include requirements tested partially, **unless** (included as a separate entry) testing in multiple procedures within the same build constitutes total testing of a particular requirement. Total Requirements Tested is computed this way in order to avoid multiple counting of individual requirements that are tested partially in more than one procedure.
- Deferred: Number of requirements that were planned to be tested in current build, but were not tested due to some FSW capability or necessary system component not being present.
- Total: Total Requirements Tested + Number of Requirements Deferred

In each software test section in Section 5 there is a table of DCR's. The state definitions are as follows:

- Opened: The DCR is currently being addressed
- Assigned: The DCR was accepted and the modification is being addressed
- InTest: The DCR was corrected and is currently in test
- Validated: The DCR was corrected and tested and have been validated, needs to have a CCB to close the DCR
- Closed: The DCR is closed and have been resolved and tested to satisfaction
- Closed with Defect: The DCR is closed and the defect is most likely assigned a differed DCR number associated with another subsystem.

2 OVERVIEW

2.1 FLIGHT DATA SYSTEM CONTEXT

Build verification was performed using cFE in a single flight processor context, as depicted in Figure 2-1. The ground system interfaces with the lab Applications Command Ingest (CI), Scheduler (SCH), and Telemetry Output (TO) and not directly with the cFE. Spacecraft operators send Commands and Files to the cFE and receive Files, Events, and Telemetry from the cFE. Note that this context is relative to the cFE and does not show ground communications with other Applications. For example, a typical spacecraft has a Stored Command (SC) Application that receives stored command loads from the ground and sends stored command dumps to the ground.

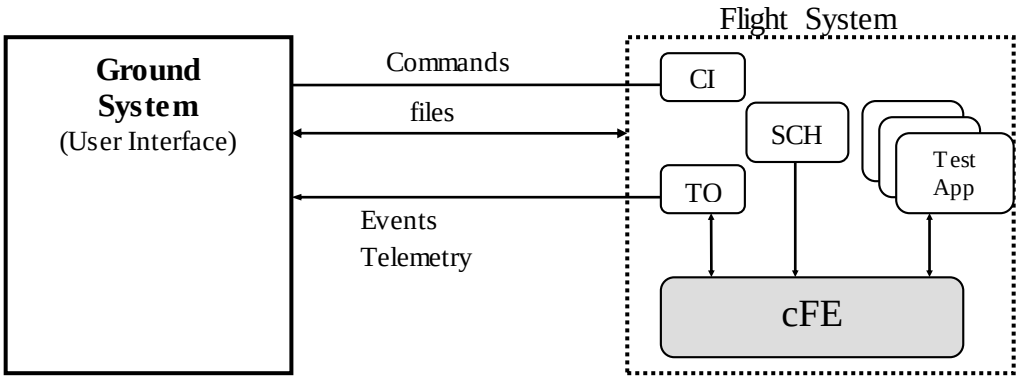

Figure 2-1 cFE Single Flight Processor Context

2.2 TEST HISTORY

cFE 3.3 – Build Verification Testing completed 9/2006 by Walt Moleski
cFE 4.0.0 – Regression Testing completed 12/2006 by Walt Moleski
cFE 4.0.0 – Build Verification Testing completed 3/2007 by Walt Moleski
cFE 4.0.1 – Build Verification Testing completed 4/2007 by Walt Moleski
cFE 4.1.0 – Build Verification Testing completed 7/6/2007 by Walt Moleski
cFE 4.2.0 - Build Verification Testing completed 8/16/2007 by Walt Moleski
cFE 4.2.1 - Build Verification Testing completed 9/24/2007 by Walt Moleski
cFE 5.0.0 - Build Verification Testing completed 11/7/2007 by Walt Moleski
cFE 5.2.0 - Build Verification Testing completed 10/6/2008 by Walt Moleski
cFE 6.0.0 – Build Verification Testing completed 8/18/2009 by Walt Moleski
cFE 6.1.1.0 – Build Verification Testing completed 11/30/2010 by Walt Moleski
cFE 6.2.2.0 – Build Verification Testing completed 10/3/2011 by Walt Moleski
cFE 6.3.1.0 – Build Verification Testing completed 2/21/12 by Walt Moleski
cFE 6.3.2.0 – Build Verification Testing completed 5/1/12 by Walt Moleski
cFE 6.4.0.0 – Build Verification Testing completed 9/24/14 by Walt Moleski
cFE 6.4.1.0 – Build Verification Testing completed 12/4/14 by Walt Moleski
cFE 6.4.2.0 – Build Verification Testing completed 6/16/15 by Walt Moleski
cFE 6.5.0.0 – Build Verification Testing completed 5/26/16 by Walt Moleski

2.3 TESTING OVERVIEW

There are 5 cFE core subsystems that are tested during Build Verification testing. There are a total of 20 test procedures that could be executed. cFE 6.5.0.0 executed all of these test procedures. Refer to the tables below for these procedures for more information on what they test. These test procedures are modified to test any new capabilities developed in a build as well as DCR fixes that were contained in a build.

For each build prior to cFE 6.0.0, a new test account was created for the testers to use. As of cFE 6.0.0, a single test account is used. This account runs the Advanced Spacecraft Integration and System Test (ASIST) software and is setup to contain all the files needed to test the cFE. These files are extracted from MKS, the source repository tool. Included in these files are test utilities. These utilities can be located in 2 places depending upon whether they are “local” or “global” utilities. The local utilities are extracted into the working prc directory ($WORK/prc). The global utilities are pointed to by ASIST in the global area defined on the test system. Additional tools utilized by the test procedures are located in the $TOOLS directory.

The following utilities were used during testing:

Name	Description
$sc_$cpu_check_sb_msgcnt	Checks if the change in the message count per msg id is as expected.
$sc_$cpu_print_sb_pipes	Prints the status of all the test app pipes.
$scx_$cpu_print_all_pipes	Prints the SB routing table.
CFE_startup	Directive combines the "start_data_center", "open_tlm", and "open cmd <cpu>" ASIST startup commands.
CFE_shutdown	Directive combines the "close_data_center" and "exit" ASIST shutdown commands.
create_tbl_file_from_cvt	Procedure that creates a load file from the specified arguments and cvt
evs_app_unreg	Procedure that request the generation of one event message which is registered for filtering and one which is not.
evs_ctr_check	To verify application evt msg sent counter EVS msg sent counter and App bin filter ctr.
evs_fltrinfo	To output evt msg filter info.
evs_gen_dis_ty	To request generation of event messages while all Evt Msg Tupes are DISABLED
evs_gen_evts	To request generation of evt msgs when requirement cEVS3103 is fully met
evs_gen_no_evts	To request generation of evt msgs while Event Message Generation is DISABLED
evs_mskd_evt	To request generation of evt msgs after change of binary filter mask from 0 to ffff (always filter) for the event message registered for filtering
evs_test_app_info	To provide test application information
ftp_file	To ftp a file to/from the FSW/GSW.
get_file_to_cvt	Procedure to write some specified FSW data to a file and then FTP the file from the FSW hardware to ASIST hardware and load file to the CVT.
get_tbl_to_cvt	Procedure that dumps the specified table from the processor and loads it into the cvt
load_start_app	Procedure to load and start a user application from the /s/opr/accounts/cfebx/apps/cpux directory.
load_table	Procedure that takes the specified file and transfers the file to the specified processor and then issues a TBL_LOAD command using the file.
tst_tbl_apps_start	Procedure that checks if the TST_TBL and TST_TBL2 applications are running and starts them if they are not.
ut_pfindicate	Directive to print the pass fail status of a particular requirement number.
ut_runproc	Directive to formally run the procedure and capture the log file.
ut_sendcmd	Directive to send EVS commands Verifies command processed and command error counters.
ut_sendrawcmd	Send raw commands to the spacecraft. Verifies command processed and command error counters.
ut_setrequirements	A directive to set the status of the cFE requirements array.

ut_setupevents	Directive to look for multiple events and increment a value for each event to indicate receipt.
ut_tlmupdate	Procedure to wait for a specified telemetry point to update.
ut_tlmwait	Directive that waits for the specified telemetry condition to be met

3 BUILD VERIFICATION TEST PREPARATION

3.1 SCENERIO DEVELOPMENT

There were no new scenarios developed for build verification test 6.5.0.0. All scenarios are stored on the MKS server, in cfe-project test-and-ground directory within the test-review-packages subdirectory in the Scenarios folder.

3.2 PROCEDURE DEVELOPMENT AND EXECUTION

This build test was completed by running 20 test procedures, 3 for Executive Services (ES), 2 for Time Services (TIME), 5 for Event Services (EVS), 4 for Software Bus (SB), 3 for Table Services (TBL), and 3 procedures that required the cFE Core software to be modified. All test procedures were written using the Spacecraft Test and Operations Language (STOL). The naming convention for files output from these test procedures was: scx_cpu<#>_<procedure name>_GMT.<ext>.

3.3 TEST PRODUCTS

Five log files were generated for every procedure that was run. They are defined as follows:

- Logs with the .loge extension list all events sent by the flight software
- Logs with the .logr extension list all requirements that passed validation by demonstration
- Logs with the .logp extension lists all prints that are generated by the test procedure
- Logs with the .logf extension lists everything from the other logs along with the steps in the test procedure
- Logs with the .logs extension lists the Standard Formatted Data Unit (SFDU) information (if applicable) contained in the full log.

A Test Report is developed by the tester after build testing is completed. The log files are stored on the test machine in the $WORK/test_logs/cFE6.5.0 folder. The data files generated are stored in the $WORK/image folder. All test products are maintained on MKS in the cfe-project test-and-ground directory.

4 BUILD VERIFICATION TEST EXECUTION

4.1 TESTBED OVERVIEW

The cFE build verification testbed consists of two ASIST workstations running ASIST version 20.2 and two MPC750 CPU boards running VxWorks 6.4 and VxWorks 6.9. CPU1 was primarily used for the development and build verification testing of the older cFE releases. CPU2 is currently under development and is not being used. CPU3 was used for cFE 6.5.0 build verification testing. Figure 4-1 depicts the testbed.

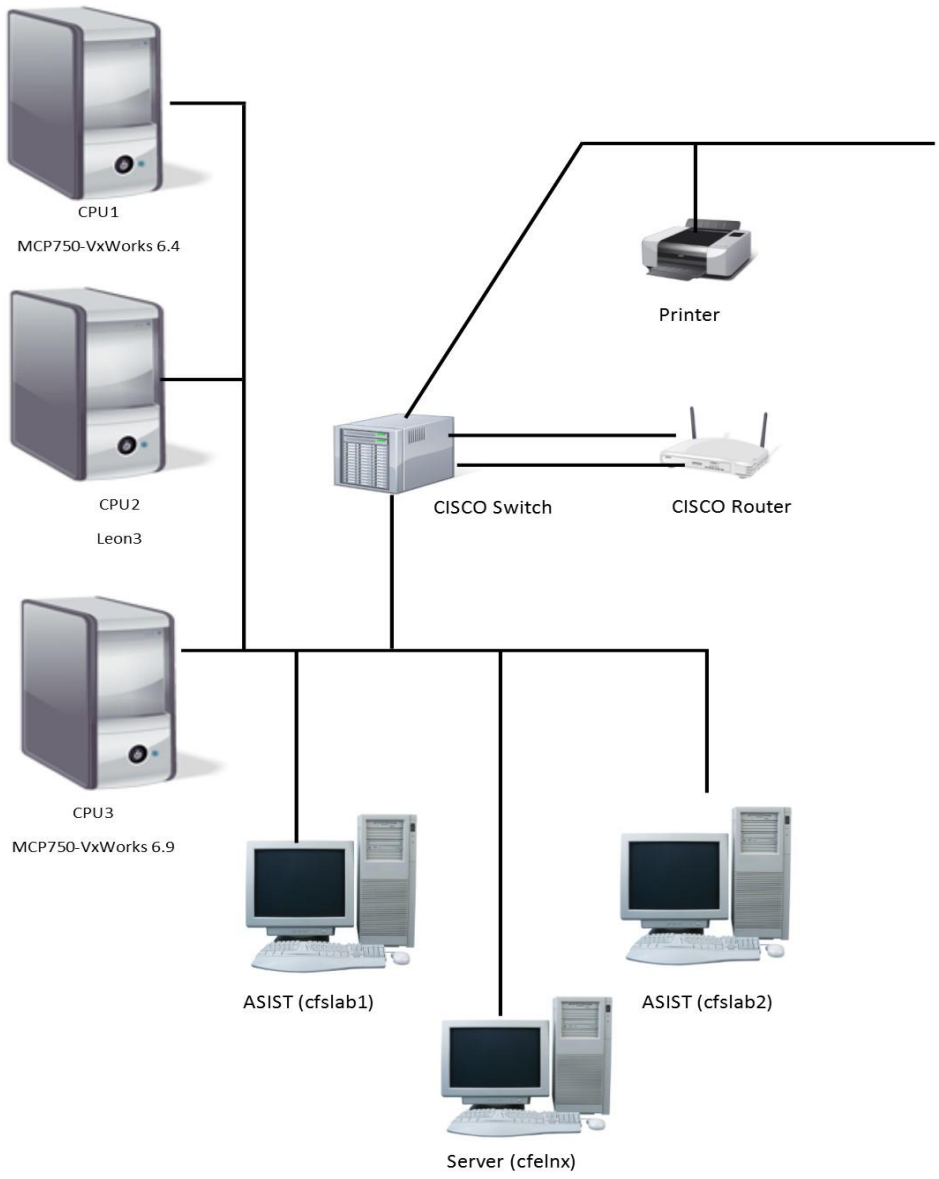

Figure 4-1: cFE Build Verification Testbed

4.2 REQUIREMENTS VERIFICATION MATRIX

Subsystem	Requirements Tested Passed	Requirements Tested Failed	Requirements Tested Partially	Total Tested	Deferred	Total
Executive Services (ES)	136	5	0	141	7	148
Time Services (TIME)	33	0	0	33	6	39
Event Services (EVS)	65	0	0	65	0	65
Software Bus (SB)	35	0	0	35	0	35
Tables (TBL)	51	0	0	51	0	51

4.3 REQUIREMENTS PARTIALLY TESTED

No requirements were partially tested.

4.4 REQUIREMENTS DEFERRED

The rational for why these requirements are deferred is contained in the Requirements to Test Traceability Matrix (RTTM). Please refer to that document for additional information.

Requirement	Description
cES1324	Upon receipt of a Request, the cFE shall load and initialize a hardware device driver and connect it with the specified hardware handshaking and device processing code.
cES1325	Upon receipt of a Request, the cFE shall unload a specified hardware device driver and de-allocate all previously allocated resources used by the driver.
cES1326	Upon receipt of a Request, the cFE shall disable a specified hardware device driver.
cES1326.1	If the specified hardware device driver is not loaded, then the cFE shall record the error in the System Log, and return an error code.
cES1327	Upon receipt of a Request, the cFE shall re-enable a specified hardware device driver.
cES1327.1	If the specified hardware device driver is not loaded, then the cFE shall record the error in the System Log, and return an error code.
cES1508.3	The cFE shall create and initialize cFE Device Drivers according to the entry in the cFE Startup File.
cTIME2012.1	The cFE shall ignore Time Updates while in Flywheel state.
cTIME2013	Upon receipt of Command the cFE shall adjust the spacecraft time by adding the Command specified value (seconds and subseconds) to spacecraft time
cTIME2014	Upon receipt of Command the cFE shall adjust the spacecraft time by subtracting the Command specified value (seconds and subseconds) from spacecraft time
cTIME2701	The cFE Time Services Server shall send a "time at the tone" Software Bus message within a <MISSION_DEFINED> period of time preceding or following the tone.
cTIME2702	The cFE Time Services Server shall update its MET using the timer hardware interface defined in the cFE Application Developer's Guide.
cTIME2703	The cFE shall define a MET with a <MISSION_DEFINED> resolution.

5 BUILD VERFICIATON TEST RESULTS

5.1 EXECUTIVE SERVICES (ES)

5.1.1 Overall Assessment

During this build test of the ES subsystem:

- 111 requirements passed demonstration
- 25 requirements were validated by analysis.
- 5 requirements failed.
- 7 requirements were deferred for Mission testing
- Two new DCRs/Trac Tickets were generated

5.1.2 Procedure Description

Procedure	Description	Requirements tested
es_appctrl	The purpose of this test is to verify the cFE Executive Services (ES) software meets the requirements defined in the SRS for the de/fined Executive Services logs (System, Exception and Reset, and Logic Analyzer Capture).	cES1005, cES1005.1, cES1005.2, cES1005.3, cES1005.4, cES1006, cES1006.1, cES1007, cES1007.1, cES1007.2, cES1007.3, cES1008, cES1008.1, cES1008.2, cES1008.3, cES1011, cES1012, cES1012.1, cES1013, cES1013.1, cES1026, cES1027, cES1300, cES1302, cES1303, cES1304, cES1305, cES1306, cES1307, cES1309, cES1309.1, cES1310, cES1310.1, cES1310.2, cES1310.3, cES1311, cES1311.1, cES1311.2, cES1312, cES1312.1, cES1313, cES1314, cES1314.1, cES1315, cES1315.1, cES1315.2, cES1316, cES1316.1, cES1316.2, cES1319, cES1320, cES1320.1, cES1320.2, cES1321, cES1321.1, cES1321.2, cES1321.3, cES1322, cES1322.1, cES1323, cES1328, cES1328.1, cES1328.2, cES1700, cES1708
es_logging	The purpose of this test is to verify the cFE Executive Services (ES) software meets the requirements defined in the SRS for the defined Executive Services logs (System, Exception and Reset, and Logic Analyzer Capture).	cES1005, cES1005.1, cES1009, cES1010, cES1014, cES1014.1, cES1014.2, cES1014.2.1, cES1014.2.2, cES1015, cES1016, cES1016.1, cES1017, cES1018, cES1019, cES1021, cES1022, cES1022.1, cES1022.2, cES1023, cES1023.1, cES1024, cES1025, cES1028, cES1509, cES1510, cES1511, cES1512, cES1520, cES1522, cES1702, cES1702.1, cES1702.2, cES1703, cES1703.1, cES1703.2, cES1706, cES1707, cES1709

Procedure	Description	Requirements tested
es_reset	The purpose of this test is to verify the cFE Executive Services (ES) software meets the requirements defined in the SRS for power-on and processor resets.	cES1000, cES1001, cES1002, cES1003, cES1004, cES1005, cES1005.1, cES1009, cES1010, cES1012, cES1016, cES1016.1, cES1017, cES1019, cES1301, cES1301.1, cES1317, cES1318, cES1500, cES1501, cES1502, cES1503, cES1504, cES1505, cES1506, cES1507, cES1508, cES1508.1, cES1508.2, cES1509, cES1510, cES1511, cES1512, cES1513, cES1514, cES1515, cES1516, cES1517, cES1518, cES1518.1, cES1518.2, cES1519, cES1519.1, cES1519.2, cES1520, cES1521
CFE_AltImage	The purpose of this test is to verify four (4) cFE requirements that require a modification to the cFE Core software. The following changes were made to the fsw: • cfe_es_start.c - Modified CFE_ES_InitializeFileSystems to force the failure of the volatile file system. • cfe_es_task.c - Modified the CFE_ES_NoopCmd function to perform a floating point divide by zero in order to cause an exception to be generated in the CORE FSW. • cfe_time_utils.c - Modified the CFE_TIME_QueryResetVars function to set the DataStoreStatus for the reset area to BAD.	cES1517.1, cES1702.3, cES1703.3, cTIME2502.1
CFE_OSObjFailure	The purpose of this test is to verify cFE requirement ES1515.1. In order to verify this requirement, the cFE Core software requires a modification. The modification was to the cfe_es_objtab.c file to have an OS Object creation failure. The modification made was to change the priority of a CFE Core task entry from what was specified to 300. This is a size that is larger than the maximum (255) priority specified.	cES1515.1
CFE_MyEH	The purpose of this test is to verify that cFE requirements ES1702.3 and ES1703.3 allow a user-defined exception handler to be created and used when exceptions occur. This procedure is a result of an update to these requirements.	cES1702.3; cES1703.3

.

5.1.3 Failed Requirements

The following requirements failed during testing.

Requirement	Description	Reason for Failure
cES1515.1	If the creation of the operating system object fails, the cFE shall perform a power on reset.	The power on reset did not occur. This was found to be due to a failure to call the CFE_PSP_Restart function from the CFE_ES_CreateObjects function when the return from OS_TaskCreate != OS_SUCCESS. Instead the CFE_PSP_Panic function is being called resulting in the processor hanging after the call to exit(-1).
cES1702.2	If the CPU exception was caused by a cFE Application, the cFE shall restart the cFE Application that caused the exception.	The CPU resets rather than just restarting the application. The reason for this failure is due to the VxWorks 6.9 kernel configuration on the platform being used to test and verify cFE 6.5.0. The unexpected behavior of performing a processor reset (rather than a task suspension) following a CPU exception was reproduced via a simple test function performing a divide by zero. This test function was loaded and run independently from the cFE core. Note: This requirement was tested and verified producing passing results on the older mcp750/VxWorks 6.4 platform. See Figure 4-1: cFE Build Verification Testbed.
cES1702.3	If the CPU exception was caused by the Operating System or cFE Core then the cFE shall initiate a <PLATFORM_DEFINED> response.	The <PLATFORM_DEFINED> exception handler did not get called. In cFE version 6.4.2, this requirement was new. This requirement was being satisfied in the PSP rather than the cFE. Changes to the PSP (in PSP version 1.3.0) resulted in the failure of this requirement. cFE requirements must be satisfied by the cFE.
cES1703.2	If the Floating Point exception was caused by a cFE Application, the cFE shall restart the cFE Application that caused the exception.	The CPU resets rather than just restarting the application. The reason for this failure is due to the VxWorks 6.9 kernel configuration on the platform being used to test and verify cFE 6.5.0. The unexpected behavior of performing a processor reset (rather than a task suspension) following a Floating Point exception was reproduced via a simple test function using floats to perform a divide by zero. This test function was loaded and run independently from the cFE core. Note: This requirement was tested and verified producing passing results on the older mcp750/VxWorks 6.4 platform. See Figure 4-1: cFE Build Verification Testbed.

Requirement	Description	Reason for Failure
cES1703.3	If the Floating Point exception was caused by the OS or cFE Core then the cFE shall initiate a <PLATFORM_DEFINED> response.	The <PLATFORM_DEFINED> exception handler did not get called. In cFE version 6.4.2, this requirement was new. This requirement was being satisfied in the PSP rather than the cFE. Changes to the PSP (in PSP version 1.3.0) resulted in the failure of this requirement. cFE requirements must be satisfied by the cFE.

5.1.4 Analysis Requirements Verification

The following ES requirements were verified using analysis.

Requirement	Description	Status	Justification
cES1014.1	Each entry in the Executive Services System Log shall be time tagged with the time that the event happened.	Pass	There are several system log files dumped to the ground that can verify this requirement. The scx_cpu3_es_syslog15.log was viewed and it contained time-stamped entries.
cES1014.2	The cFE shall calculate the number of bytes used and number of entries in Executive Services System Log	Pass	The ES Housekeeping display page in ASIST contains this information. Steps 1.11 of the ES_Logging test procedure attempt to fill the ES System Log and utilize the bytes used and print the number of entries contained in the System Log.
cES1014.2.1	If the Executive Services System Log is full and the System Log Mode is set to OVERWRITE then the cFE shall write all new entries from the top of the log	Pass	The system log dump file scx_cpu3_es_syslog1117.log verifies this requirement by showing a new entry in the system log at the top of the file.
cES1014.2.2	If the Executive Services System Log is full and the System Log Mode is set to DISCARD then the cFE shall discard all new entries	Pass	Step 1.11.4 writes a system log message when the mode is DISCARD. The files scx_cpu3_es_syslog1113.log and scx_cpu3_es_syslog1115.log were viewed. Both logs contained the same entries and the entry written in Step 1.11.4 was not contained in the scx_cpu3_es_syslog1115.log file.
cES1017	The cFE shall maintain an Executive Services Exception and Reset Log which will log critical system data for exceptions and resets including: • A time stamp • Processor Context information • Critical system variables • ASCII string stating the reason for the reset	Pass	The Exception and Reset Log contained the stated components. This was verified by viewing the ASIST display page after transferring the scx_cpu3_er13.log file to the ground.

Requirement	Description	Status	Justification
cES1022.1	The cFE shall store a timestamp along with the specified Logic Analyzer Capture Tag.	Pass	There are 2 performance log files generated by the ES_Logging test procedure. Viewing these files in the Software Timing Analyzer tool verified that each entry contained a timestamp.
cES1022.2	If the Logic Analyzer Capture Log is full, then the cFE shall write all new entries from the top of the log	Pass	The imported performance analysis file scx_cpu3_perf37.dat file indicates that the starting point is non-zero. This means that the file has overlapped data contained in it.
cES1311.2	In the event a child task attempts to create another child task, the cFE shall record the error in the System Log, and return an error code.	Pass	Step 3.4 of the es_appctrl procedure starts a child task that attempts to start another child task. The required system log messages were included in the scx_cpu3_es_app33syslog.log file indicating that a child cannot start a child task.
cES1314	Upon receipt of a Request, the cFE shall end execution of the calling cFE Child Task.	Pass	Step 3.8 of the es_appctrl procedure tests this requirement. The uart dump was captured and it contained the required message to verify that the child task has ended.
cES1314.1	If the calling task is the cFE Application Main Task, the cFE shall record the error in the System Log, and return an error code.	Pass	Step 3.7 of the es_appctrl procedure tests this requirement. The scx_cpu3_es_app36syslog.log file clearly contains the appropriate message indicating that a main task cannot be stopped with the CFE_ES_ExitChildTask API.
cES1321.2	If the specified Memory Pool identifier is invalid then the cFE shall record the error in the System Log, and return an error code.	Pass	Step 4.8 of the es_appctrl procedure tests this requirement by trying to allocate a memory block for a non-existing memory pool. The scx_cpu3_es_app48syslog.log file contains the required system log entry to verify this requirement.
cES1501	Upon a Power-On Reset, the cFE shall clear the Executive Services System Log.	Pass	Step 4.5 in the ES_Reset test procedure dumps the system log to the scx_cpu3_es_syslog45.log after performing a Power-On reset. This log contained the system startup information.

Requirement	Description	Status	Justification
cES1502	Upon a Power-On Reset, the cFE shall clear the Executive Services Exception and Reset Log.	Pass	Step 4.5 of the ES_Reset test procedure dumps the Exception and Reset log to the scx_cpu3_es_erlog45.log file after a Power-On reset. This file contains a single entry for the Power-On reset.
cES1505	Upon a Power-on Reset, the cFE shall create all operating system objects required by the cFE.	Pass	There are two system log files dumped by the ES_Reset test procedure that verify this requirement. The files scx_cpu3_es_syslog145.log and scx_cpu3_es_syslog45.log contain an entry indicating that the system objects were created.
cES1508.2	The cFE shall create and initialize cFE Shared Libraries according to the entry in the cFE Startup File.	Pass	The scx_cpu3_es_syslog145.log file contains an entry indicating that the cFE Test Library was initialized. This is the library contained in the startup script used when the system is started or reset.
cES1511	Upon a Processor Reset, the cFE shall preserve the Executive Services System Log.	Pass	The scx_cpu3_es_syslog1.log is dumped by the ES_Reset test procedure when a Processor Reset occurs. This file contained the previous entries and thus was preserved.
cES1512	Upon a Processor Reset, the cFE shall preserve the Executive Services Exception and Reset Log.	Pass	The Exception and Reset log was dumped after performing two Processor Resets in the ES_Reset test procedure. The files scx_cpu3_es_erlog35.log and scx_cpu3_es_erlog55.log contained the previous entries and thus were preserved.
cES1515	Upon a Processor Reset, the cFE shall create all operating system objects required by the cFE.	Pass	The scx_cpu3_es_syslog1.log file generated by the ES_Reset test procedure when a Processor Reset occurs contains an entry indicating that the system objects were created.
cES1518.2	The cFE shall create and initialize Shared Libraries according to the entry in the cFE Startup File.	Pass	Step 3.5 in the ES_Reset test procedure dumps the System Log to the scx_cpu3_es_syslog1.log file. This file contains and entry indicating that the cFE shared Library was initialized.
cES1519.2	The cFE shall create and initialize Shared Libraries according to the entry in the cFE Startup File.	Pass	Step 5.5 of the ES_Reset test procedure dumps the System log to scx_cpu3_es_syslog1.log. This file contained an entry indicating the cFE shared library was initialized.

Requirement	Description	Status	Justification
cES1520	Upon a Processor Reset, the cFE shall make an entry in the Executive Services Exception and Reset Log recording the Processor Reset.	Pass	The ES_Logging test procedure dumps the Exception and Reset log to files after a Processor Reset occurs. The scx_cpu3_er110.log and scx_cpu3_er25.log files contain entries indicating the Processor Reset occurred.
cES1702.1	Upon detection of a CPU exception, the cFE shall add an entry in the Executive Services Exception And Reset Log.	Pass	The ES_Logging test procedure generates an exception using a test application in Step 2.3. The exception added an entry into the Exception and Reset log and can be verified with the scx_cpu3_er23.log file.
cES1703.1	Upon detection of an unmasked Floating Point exception, the cFE shall add an entry in the Executive Services Exception and Reset Log.	Pass	The ES_Logging test procedure generates an exception using a test application in Step 2.3. The exception added an entry into the Exception and Reset log and can be verified with the scx_cpu3_er23.log file.
cES1704	The cFE shall support a <PLATFORM_DEFINED,TBD> byte volatile file system.	Pass	This requirement was tested manually from the ASIST console by uploading a large file to the volatile file system and then attempting to generate another file. When the file system is full, the additional file creation command fails. I then removed the large file and issued the command again. This time the command passed and created the file. Although the uart output was not captured, the errors as well as the successful writes were contained in the uart.
cES1705	The cFE shall support a <PLATFORM_DEFINED,TBD> byte non-volatile file system.	Pass	The non-volatile file system was inspected and verified on the test CPU.

5.1.5 DCRs/Trac Tickets

Two new DCRs/Trac Tickets were generated (#164 and #168) in response to the failure of requirements cES1515.1, cES1702.3, and cES1703.3.

5.1.6 Notes

Other than the failed requirements mentioned above, there were no significant findings and/or anomalies reported during testing.

5.2 TIME SERVICES (TIME)

5.2.1 Overall Assessment

During this build test of the TIME subsystem:

- 32 requirements passed demonstration
- 1 requirement was validated by analysis
- 6 requirements were deferred for later testing
- No new DCRs/Trac Tickets were generated during testing

5.2.2 Procedure Description

Procedure	Description	Requirements tested
time_command_server_tai	The purpose of this test is to verify the Core Flight Executive (cFE) Time Services (TIME) common subsystem commands, time adjustment commands, clock selection commands, current time access requests, and time utility requests.	cTIME2000, cTIME2001, cTIME2002, cTIME2003, cTIME2004, cTIME2005, cTIME2006, cTIME2007, cTIME2008, cTIME2009, cTIME2010, cTIME2011, cTIME2012, cTIME2012.1, cTIME2013, cTIME2014, cTIME2300, cTIME2301, cTIME2302, cTIME2303, cTIME2304, cTIME2305, cTIME2306, cTIME2307, cTIME2309, cTIME2310, cTIME2311, cTIME2312, cTIME2313, cTIME2314
time_resets_server_tai	The purpose of this test is to verify the Core Flight Executive (cFE) Time Services (TIME) processor reset requirements.	cTIME2005, cTIME2006, cTIME2012, cTIME2306, cTIME2307, cTIME2308, cTIME2500, cTIME2501, cTIME2502, cTIME2700
CFE_AltImage	The purpose of this test is to verify four (4) cFE requirements that require a modification to the cFE Core software. The following changes were made to the fsw: • cfe_es_start.c - Modified CFE_ES_InitializeFileSystems to force the failure of the volatile file system. • cfe_es_task.c - Modified the CFE_ES_NoopCmd function to perform a floating point divide by zero in order to cause an exception to be generated in the CORE FSW. • cfe_time_utils.c - Modified the CFE_TIME_QueryResetVars function to set the DataStoreStatus for the reset area to BAD.	cES1517.1, cES1702.3, cES1703.3, cTIME2502.1

5.2.3 Analysis Requirements Verification

The following TIME requirements were verified using analysis.

Requirement	Description	Status	Justification
cTIME2314	Upon receipt of a Request the cFE shall return the provided system time in the following format; yyyy-ddd-hh:mm:ss.xxxxx\0	Pass	This requirement can be verified by looking at any ES System Log dump file generated by the cFE 6.5.0.0 test procedures. This was done and the time format was present in the system log.

5.2.4 DCRs/Trac Tickets

No new DCRs/Trac Tickets were generated during 6.5.0.0 testing.

5.2.5 Notes

There were no significant findings and/or anomalies reported during testing.

5.3 EVENT SERVICES (EVS)

5.3.1 Overall Assessment

During this build testing of the EVS subsystem:

- 56 requirements were validated by demonstration
- 9 requirements were validated by analysis
- No new DCRs/Trac Tickets were generated during testing

5.3.2 Procedure Description

Procedure	Description	Requirements tested
evs_evt_msg_gen	The purpose of this test is to verify the functionality of the cFE Event Message generation software for Events Messages that are registered for filtering as well as Event Messages that are not registered for filtering.	cEVS3004, cEVS3007, cEVS3008, cEVS3012, cEVS3018, cEVS3100, cEVS3100.1, cEVS3100.2, cEVS3100.3, cEVS3101, cEVS3102, cEVS3103, cEVS3103.1, cEVS3103.2, cEVS3103.3, cEVS3103.4.1, cEVS3103.6, cEVS3103.7, cEVS3104, cEVS3105, cEVS3109
evs_cmds	The purpose of this test is to verify the CFE_EVS Command functionality for the Event Service (CFE_EVS) function of the Core Flight Executive (cFE). The operation of all CFE_EVS commands will be verified for valid and invalid commands.	cEVS3000, cEVS3002, cEVS3003, cEVS3004, cEVS3004.1, cEVS3005,cEVS3006, cEVS3007, cEVS3008, cEVS3009, cEVS3010, cEVS3011, cEVS3017, cEVS3018, cEVS3300

Procedure	Description	Requirements tested
evs_log	The purpose of this test is to verify the EVS log requirements for the Event Service (EVS) function of the Core Flight Executive (cFE) software. The operation of EVS Log will be verified in both the Overwrite and Discard modes. The Local Event Log Full flag, Local Event Log Overflow Counter, Event Logging Mode flag, and Event Format flag will be examined for proper value(s) during the execution of the test scenario. The contents of the Event Log will be periodically dumped from the FSW to the ASIST box for examination using telemetry pages and off-line analysis. The TST_EVS test application will be used to send multiple event messages. The supplied event text / event time will serve to uniquely identify each event message.	cEVS3001, cEVS3013, cEVS3014, cEVS3015, cEVS3015.1, cEVS3016, cEVS3018, cEVS3103.4, cEVS3103.5, cEVS3108, cEVS3108.1, cEVS3108.2, cEVS3108.3, cEVS3301
evs_bin_fltr	The purpose of bin_fltr test is to verify the correct functionality of the Binary Filter Process in the cFE FSW.	cEVS3003, cEVS3004, cEVS3009, cEVS3010, cEVS3011, cEVS3012, cEVS3019, cEVS3019.1, cEVS3019.2, cEVS3020, cEVS3020.1, cEVS3100, cEVS3100.1, cEVS3103, cEVS3013.3, cEVS3013.3.1, cEVS3104, cEVS3104.1, cEVS3105, cEVS3105.1, cEVS3106, cEVS3107, cEVS3302
evs_reset	The purpose of evs_reset is to verify Event Message Services EVS behavior upon Power on and Processor Reset.	cEVS3017, cEVS3104, cEVS3110, cEVS3200, cEVS3201, cEVS3202, cEVS3203, cEVS3207, cEVS3208, cEVS3209, cEVS3210

5.3.3 Analysis Requirements Verification

The following EVS requirements were verified using analysis.

Requirement	Description	Status	Justification
cEVS3015	<OPTIONAL> Upon receipt of Command, the cFE shall write the contents of the Local Event Log to the Command specified file.	Pass	Steps 4.5.1 and 4.5.2 of the evs_log test procedure sent commands specifying a filename and using the default filename for writing the contents of the Local Event Log. These files were transferred to the ground and displayed in the EVS_LOG ASIST display page. Both commands displayed the contents of the files.
cEVS3015.1	If a file is not specified, the cFE shall use the <PLATFORM_DEFINED> filename.	Pass	Steps 4.5.1 and 4.5.2 of the evs_log test procedure sent commands specifying a filename and using the default filename for writing the contents of the Local Event Log. These files were transferred to the ground and displayed in the EVS_LOG ASIST display page. Both commands displayed the contents of the files.
cEVS3016	<OPTIONAL> The cFE shall write each Event Message from the earliest logged message to the most recently logged message.	Pass	Step 7.5.1 of the evs_log test procedure verifies this requirement. The step dumps the local event log and then prints it in the procedure log file. The entries of the log were in earliest to latest order.
cEVS3100	Upon receipt of Request, the cFE shall register an Application for event service, enabling the Application Event Service Enable Status and storing the following request specified Application data: Application Event IDs (for events to be filtered) Application Binary Filter Masks (one per registered Event ID)	Pass	The EVS Housekeeping, EVS_App_Data_Main and EVS_App_Data display pages were used to verify this requirement. All the listed applications in this display page were registered for event services. The event filter masks and messages were viewed in the EVS_App_Data display page.
cEVS3103.6	The requester shall be able to specify the Application ID to be used in the Event Message	Pass	This requirement was verified by viewing the log file and verifying that the event message contained the specified item.
cEVS3103.7	The requester shall be able to specify the time to be used in the Event Message.	Pass	This requirement was verified by viewing the log file and verifying that the event message contained the specified item.

Requirement	Description	Status	Justification
cEVS3108.3	<OPTIONAL> If the Local Event Log is full, the cFE shall either (1) overwrite the oldest Event Message if the Event Logging Mode is overwrite, or (2) discard the Event Message if the Event Logging Mode is discard.	Pass	Steps 3.3.3, 3.4.1 and 4.2.1 of the evs_log test procedure verify this requirement. The local event log is written and displayed in the EVS_Log window as well as printed in the procedure log file. The analysis verifies that in the first two steps the log messages were overwritten and the last step verifies that the log remained the same.
cEVS3109	For each created Event Message, the cFE shall route the Event Message, formatted as an ASCII text string, to each enabled Event Message Output Port.	Pass	The uart window displayed multiple messages for a single event when multiple output ports were enabled. The uart log was not captured but the multiple events were viewed by the tester as the test executed.
cEVS3207	<OPTIONAL> Upon a Processor Reset, the cFE shall preserve or overwrite the contents of the Local Event Log based upon the setting of the Event Logging Mode configuration parameter.	Pass	Step 3.1 of the evs_reset test procedure dumps and displays the local EVS log both before and after a Processor Reset. The file rst_284.log file is the contents before the reset and the rst_301.log is the contents after the reset. Verification of these files finds that the information was preserved after the reset since the configuration parameter was set to DISCARD.

5.3.4 DCRs/Trac Tickets

No DCRs/Trac Tickets were generated during build testing.

5.3.5 Notes

There were no significant findings and/or anomalies reported during testing.

5.4 SOFTWARE BUS SERVICES (SB)

5.4.1 Overall Assessment

During SB build verification testing

- 33 requirements were validated by demonstration
- 2 requirements were validated by analysis
- No new DCRs/Trac Tickets were generated

5.4.2 Procedure Description

Procedure	Description	Requirements tested
sb_enapipes	The purpose of this test is to verify that the flight software satisfies the requirements relating to enabling pipes.	cSB4000, cSB4003, cSB4004, cSB4005, cSB4007, cSB4007.1, cSB4300, cSB4301, cSB4302, cSB4303, cSB4304, cSB4305, cSB4305.5, cSB4305.6, cSB4306, cSB4307, cSB4308, cSB4309, cSB4701, cSB4704, cSB4705
sb_dispipes	The purpose of this test is to verify that the flight software satisfies the requirements relating to disabling pipes.	cSB4001, cSB4002, cSB4003, cSB4003.1, cSB4005, cSB4008, cSB4008.1, cSB4301, cSB4303, cSB4305.1, cSB4305.3, cSB4305.4, cSB4500, cSB4700, cSB4705, cSB4706
sb_cmds_err	The purpose of this test is to verify that the flight software will reject SB commands with bad data in the command fields.	cSB4004, cSB4005, cSB4305.6, cSB4701
sb_reset	The purpose of this test is to verify that the SB flight software handles a Power-On and Processor reset according to the requirements.	cSB4303, cSB4303.1, cSB4310, cSB4311, cSB4311.1, cSB4500, cSB4501

5.4.3 Analysis Requirements Verification

The following SB requirements were verified using analysis.

Requirement	Description	Status	Justification
cSB4300	The cFE shall provide a zero-copy message transfer mode for intra-processor communication.	Pass	Step 11.0 of the sb_enapipes procedure tests this requirement. The TST_SB application generates an event message that prints the pointer of the SB zero copy message being sent and also generates an event message when the zero copy message is received. The pointers were identical.

cSB4310	Upon receipt of Request the cFE shall free resources allocation for the specified Application	Pass	Step 7.2 of the SB_Reset test procedure sends a command to stop the TST_SB application. When this command executes, there are numerous events generated and contained in the log file indicating that the TST_SB resources were "freed".

5.4.4 DCRs/Trac Tickets

No DCRs/Trac Tickets were generated during build testing.

5.4.5 Notes

There were no significant findings and/or anomalies reported during testing.

5.5 TABLE SERVICES (TBL)

5.5.1 Overall Assessment

During this build testing of the TB subsystem:

- 49 requirements were validated by demonstration
- 2 requirements were validated by analysis
- No new DCRs/Trac Tickets were generated during testing

5.5.2 Procedure Description

Procedure	Description	Requirements tested
tbl_func	The purpose of this test is to verify the functionality of the cFE Table Services commands.	cTBL6000, cTBL6000.5, cTBL6001, cTBL6002, cTBL6002.1, cTBL6002.2, cTBL6003, cTBL6003.1, cTBL6003.1.1, cTBL6003.1.2, cTBL6005, cTBL6005.1, cTBL6006, cTBL6011, cTBL6012, cTBL6012.1, cTBL6012.2, cTBL6012.3, cTBL6300, cTBL6300.1, cTBL6301, cTBL6302, cTBL6302.1, cTBL6302.2, cTBL6303, cTBL6304, cTBL6305, cTBL6305.1, cTBL6305.2, cTBL6306, cTBL6308, cTBL6308.1, cTBL6309, cTBL6310, cTBL6311, cTBL6311.1, cTBL6311.2, cTBL6312, cTBL6700, cTBL6701
tbl_cmding	The purpose of this test is to verify the Table Services commands.	cTBL6000, cTBL6000.1, cTBL6000.2, cTBL6000.3, cTBL6000.4, cTBL6001, cTBL6003, cTBL6007, cTBL6008, cTBL6009, cTBL6010, cTBL6011
tbl_reset	The purpose of this test is to verify that the cFE Table Services (TBL) software meets the requirements defined in the SRS for Power-	cTBL6500, cTBL6501, cTBL6501.1

	On and Processor Resets	

5.5.3 Analysis Requirements Verification

The following TBL requirements were verified using analysis.

Requirement	Description	Status	Justification
cTBL6308.1	If a Table is locked when an update Request is made, an appropriate error code shall be returned to the calling Application and the update shall not occur.	Pass	The uart output captured for the tbl_func test procedure contained an error indicating that the table was locked. Once the lock was removed, the table was updated appropriately.
cTBL6311.1	Upon providing a calling Application with the addresses of a Tables' data, the cFE shall lock the contents of the Tables to prevent modification.	Pass	Step 18.2 of the tbl_func test procedure attempts to update a table that is shared by another application. The error message displayed indicating that the table did not have any working buffers available to perform the update.

5.5.4 DCRs/Trac Tickets

No new DCRs/Trac Tickets were generated.

5.5.5 Notes

There were no significant findings and/or anomalies reported during testing.

5.6 DCRS/TRAC TICKETS VERIFIED

The following DCRs/Trac Tickets were explicitly tested and verified during cFE 6.5.0.0 Build Verification testing. Build test procedures were not adequate for testing/verifying these DCRs/Trac Tickets.

DCR/ Ticket #	High Level Description of Functionality/Bug Report	Test Method	Test Approach
#2	Compiler errors/warnings on EVS_SendEvent() calls on some architectures	Demonstration	Verified no warnings were generated when EVS was compiled.
#5	Reentrant version of decompress routine in CFE FS. The decompression routine keeps its internal state in global variables which makes it non-reentrant. To protect against concurrent usage a mutex is used but this has a major performance impact. The global variables have been replaced with a state structure so that it can be multithreaded like all other parts of CFS. A global state object has been implemented in order to preserve API compatibility.	Demonstration	A test application was gzipped and uploaded to the test CPU. The ES_StartApp command was sent specifying the gzipped file and the application started.
#33	Fix "no return" warning on CFE_SB_ReadQueue() function	Demonstration	Verified no warnings were generated when SB was compiled.
#52	CFE_TIME fails to build with CFE_TIME_CFG_SRC_MET set to TRUE	Demonstration	Verified the cfe-core.o was built when setting this parameter to TRUE and other related parameters that depend upon this setting.
#106	CCSDS header file macro CCSDS_INC_SEQ generates a compiler warning when referenced (GSFC DCR 22932)	Demonstration	Added this macro into a test application and compiled it. Verified no warnings were generated.
#120	resolve "-m32" CFE classic build issues	Demonstration	This issue was resolved by building the cFE. Verified successful build of cFE 6.5.0 via classic build makefiles.

5.6.1 Outstanding DCRs/Trac Tickets

Information on currently open Trac tickets is available at:

https://babelfish.arc.nasa.gov/trac/cfs_cfe/

Note this is a restricted website that require a server account. Additional DCRs and/or Trac Tickets may have been submitted after preparation of this report. A cFE DCR and/or Trac Ticket report containing a listing of open DCRs and/or Track Tickets is available on request for customers who do not have access to the babelfish server. Please contact Susanne Strege, susie.strege@nasa.gov for detailed information on currently open Trac tickets if access to the babelfish server is restricted.

No.	Trac Ticket #	Description	Component	Status	Planned Delivery	Type	Priority
1	#14	CFE_TIME_GetTime() should not return a structure	other	new	Not Determined	enhancement	minor
2	#15	CFE_TIME_GetReference() has insufficient protection against update while reading	time	new	Not Determined	defect	major
3	#17	Implement "bootstrap" script	build	new	Not Determined	enhancement	major
4	#25	Consolidate CDS and generic/ram mempool code into single implementation	es	on_hold	Not Determined	enhancement	minor
5	#30	Review use of CFE_PSP_MemCpy/CFE_PSP_MemSet	other	review	Not Determined	task	minor
6	#34	update cFE unit tests	test	assigned	Not Determined	defect	major
7	#38	Update CFE_FS_InitHeader to to Handle Error/Invalid Length Conditions	fs	new	Not Determined	enhancement	major
8	#39	Enforce Strict ASCII	other	new	Not Determined	defect	minor
9	#43	CFE TIME uses OSAL IntLock/IntUnlock for mutual exclusion	time	review	Not Determined	defect	major
10	#44	CFE_ES_WriteToSyslog() is not multi-thread safe	es	review	Not Determined	defect	major
11	#45	CFE_ES_ProcessCoreException() is not interrupt-safe	other	review	Not Determined	defect	major
12	#46	Application Startup Race Conditions (GSFC DCR 22819)	es	new	Not Determined	defect	major
13	#47	CFE TIME fails to build when CFE_TIME_CFG_SIGNAL set to TRUE	time	review	Not Determined	defect	minor
14	#49	Extend CMake app search path	build	new	Not Determined	enhancement	minor
15	#53	File operations in CFE_ES_ShellOutputCommand() need cleanup	es	review	Not Determined	defect	minor
16	#54	Pre-CMake fallback build script needs updating.	build	in_work	Not Determined	task	minor
17	#58	Exiting an Application Creates an Application with an Unknown State (GSFC DCR 23035)	es	new	Not Determined	defect	major
18	#61	CFE_SB_GetMsgTime() and CFE_SB_TimeStampMsg() do not handle byte-swapping on _EL platforms	sb	new	Not Determined	defect	major
19	#62	Clean up EVS_SendViaPorts() function	evs	new	Not Determined	enhancement	minor
20	#63	EVS "output ports" should be a function of the PSP	evs	review	Not Determined	enhancement	minor
21	#64	Suspicious implementation of SHORT_FORMAT mode in EVS_SendPacket()	other	review	Not Determined	defect	major
22	#69	SB Pipes are not protected.	sb	review	Not Determined	defect	major

23	#70	SB Only Increments Message Sequence Count Where There are Subscribers	other	new	Not Determined	defect	major
24	#78	cFE TIME unit tests break when different configuration options are used	time	new	Not Determined	defect	major
25	#83	Default Configuration Setting for CFE_ES_STARTUP_SCRIPT_TIMEOUT_MSEC is Too Big	other	new	Not Determined	enhancement	major
26	#85	Add UT assert stubs to CFE	test	on_hold	Not Determined	enhancement	major
27	#86	Correction of an infinite loop in cfe_sb_task.c	sb	new	Not Determined	defect	minor
28	#89	ES Does Not Check CFE_PSP_MemRead8 Return Code	es	new	Not Determined	defect	major
29	#90	ES - Invalid Memory Handle When Restarting/Deleting an Application with Tables (GSFC DCR 14483)	tbl	new	Not Determined	defect	major
30	#92	cFE Time subsystem has calls to OS functions that do not exist	time	new	Not Determined	defect	major
31	#93	Executive Services always creates tasks with floating point enabled (GSFC DCR 21293)	es	new	Not Determined	defect	major
32	#94	ES - Add Ability to Recreate the RAM Disk via Command (GSFC DCR 21594)	es	new	Not Determined	defect	major
33	#95	ES - Registered Tasks Counter Does Not Decrement When Child Tasks are Exited (GSFC DCR 21771)	es	new	Not Determined	defect	major
34	#96	Add support to allow SBN to pass sender information across the network (GSFC DCR 22063)	other	new	Not Determined	defect	major
35	#97	EVS - Add Configuration To Output Events Upon Command Message vs. Function Call (GSFC DCR 22080)	evs	new	Not Determined	defect	major
36	#98	SB - Add Last Pipe ID and Msg ID to Routine Telemetry for Diagnosing Message Limit Error and Buffer Overrun Errors (GSFC DCR 22081)	sb	new	Not Determined	defect	major
37	#99	TBL - Update Table Services to Send Messages to Notify Applications of Pending Table Updates (GSFC DCR 22622)	tbl	new	Not Determined	defect	major
38	#100	Update CFE_ES_SYSTEM_LOG_SIZE Verify to Allow Larger Sys Log Files (GSFC DCR 22684)	es	new	cfe_next	defect	major
39	#101	Table Services Name Buffer Overflow	tbl	new	Not Determined	defect	major
40	#102	ES Creates Redundant Sys Log Entries When Creating ER Log Entries (GSFC DCR 22768)	es	new	Not Determined	defect	major
41	#104	MMS-IVV-013 (OBS-1238) - Static Code Analysis: Possible Buffer Underrun in cfe_fs_decompress.c (GSFC DCR 22838)	fs	new	Not Determined	defect	major
42	#105	cFE Does Not Meet Double Floating Point Alignment Requirements on PPC440x5 Book E Architecture (GSFC DCR 22813)	common	new	Not Determined	defect	major
43	#107	SB - Duplicate Pipe Creation Causes Failure to Delete Pipe (GSFC DCR 22934)	sb	new	Not Determined	defect	major

44	#108	TBL - Dump Table Registry Data Command Can Hog CPU (GSFC DCR 23031)	tbl	new	Not Determined	defect	major
45	#110	ES - Recursive Exit Application Error Message	es	new	Not Determined	defect	major
46	#111	Naming convention for macros in cfe_mission_cfg and cfe_platform_cfg	other	new	Not Determined	enhancement	major
47	#112	Simplify Function Pointer Manipulations	other	new	Not Determined	task	minor
48	#115	Standardize Version Numbering (in CFE)	other	new	Not Determined	defect	major
49	#116	printf format specs need to be cleaned up	other	new	Not Determined	defect	minor
50	#117	CFE_ES_GetAppName() undefined output when failure occurs	es	new	cfe_next	defect	major
51	#118	Improve cppcheck configuration for CFE	cppcheck	new	Not Determined	enhancement	major
52	#119	cppcheck a vxworks build	cppcheck	in_work	Not Determined	task	major
53	#133	CFE_ES_AppCreate does not unload an object file if the entry point is not found	other	new	cfe_next	defect	major
54	#135	SB: "cfe_sb.h" should not depend on cfe_platform_cfg.h	sb	new	cfe_next	defect	minor
55	#137	Possible buffer overrun in format strings used for scanf	es	new	cfe_next	defect	major
56	#138	FS - ExtractFilenameFromPath Function Needs Revision	other	new	cfe_next	enhancement	major
57	#140	The ES "LoadLibrary()" call - avoid duplicates and pass ID	es	new	cfe_next	enhancement	minor
58	#141	Macro Parameters need Parens	common	new	cfe_next	defect	minor
59	#142	Refactor CFE_ES_AppCreate and CFE_ES_LoadLibrary	es	new	cfe_next	enhancement	minor
60	#143	ES does not check target file existence before attempting to reload an application (GSFC DCR 145460)	other	new	cfe_next	defect	major
61	#144	Thread safety issues in CFE_TIME around the Sync Callbacks	time	new	cfe_next	defect	major
62	#145	use the OSAL configuration file loader library	es	new	cfe_next	enhancement	minor
63	#147	ES - CreateChildTask API Function Does Not Use "Flags" Input Parameter	es	new	Not Determined	defect	major
64	#152	Redundant Assignments and Unread Variables	other	new	Not Determined	defect	minor
65	#156	Incorrect leap seconds in docs	time	new	Not Determined	defect	minor
66	#158	EVS Unit Test Code Coverage Incomplete in Task.c (GSFC DCR 8492)	evs	new	Not Determined	enhancement	minor
67	#159	ES Unit Test Code Coverage Incomplete in apps.c	es	new	Not Determined	enhancement	minor
68	#161	CFE_ES_DeleteChildTask SysLog Message/Comments are Misleading	es	new	cfe_next	defect	minor
69	#164	cFE cES1702.3 and cES1703.3 Requirement Failures	es	new	cfe_next	defect	major
70	#168	cFE cES1515.1 Requirement Failure	es	new	cfe_next	defect	major

RTTM

The cFE 6.5.0.0 RTTM can be found on the MKS server, in CFE-Repository test-and-ground directory Results folder.

APPENDIX A - COMMAND, TELEMETRY, AND EVENTS VERIFICATION MATRIX

Command	Test Procedure(s)	Notes/Comments
ES_NOOP	ES_Reset	
ES_ResetCtrs	ES_Reset	
ES_ProcessorReset	ES_Logging, ES_Reset	
ES_PowerOnReset	ES_Logging, ES_Reset, ES_App_Ctrl	
ES_Shell	ES_App_Ctrl	
ES_StartApp	ES_Logging, ES_Reset, ES_App_Ctrl	
ES_DeleteApp	ES_App_Ctrl	
ES_RestartApp	ES_App_Ctrl	
Es_ReloadApp	ES_App_Ctrl	
ES_QueryApp	ES_App_Ctrl	
ES_WriteAppInfo2File	ES_App_Ctrl	
ES_ClearSysLog	ES_Logging	
ES_WriteSysLog2File	ES_Logging, ES_Reset	
ES_ClearERLog	ES_Logging	
ES_WriteERLog2File	ES_Logging, ES_Reset	
ES_StartPerf	ES_Logging	
ES_StopPerf	ES_Logging	
ES_PerfFltrMask	ES_Logging	
ES_PerfTrigMask	ES_Logging	
ES_OverwriteSysLogMode	ES_App_Ctrl	
ES_ResetPRCnt	ES_Logging	
ES_SetMAXPRCnt	ES_Logging	
ES_DeleteCDS	ES_App_Ctrl	
ES_PoolStats	ES_App_Ctrl	
ES_WriteCDS2File	ES_App_Ctrl	
ES_WriteTaskInfo2File	ES_App_Ctrl	
EVS_NOOP	EVS_BinFilter, EVS_Cmd, EVS_Reset	
EVS_ResetCtrs	EVS_Cmd	
EVS_EnaEventType	ES_App_Ctrl, ES_Logging, ES_Reset. EVS_BinFilter, EVS_Cmd, EVS_Reset, EVS_EvtGen, SB_DisablePipe, SB_EnablePipe, SB_Reset, TBL_Cmd, TBL_Reset, TBL_Functionality, TIME_CmdTlm	
EVS_EnaEventTypeMask	EVS_Cmd, TIME_CmdTlm	
EVS_DisEventType	EVS_Cmd, EVS_Reset	
EVS_DisEventTypeMask	EVS_Cmd	
EVS_SetEvtFmt	EVS_Log, EVS_Reset	
EVS_EnaAppEvtType	EVS_BinFilter, EVS_Cmd, EVS_EvtGen	
EVS_EnaAppEvtTypeMask	EVS_Cmd	
EVS_DisAppEvtType	EVS_BinFilter, EVS_Cmd, EVS_EvtGen	
EVS_DisAppEvtTypeMask	EVS_Cmd	

Command	Test Procedure(s)	Notes/Comments
EVS_EnaAppEvGen	EVS_Cmd, EVS_EvtGen	
EVS_DisAppEvGen	EVS_Cmd, EVS_EvtGen, EVS_Reset	
EVS_RstAppCtrs	EVS_BinFilter, EVS_Cmd	
EVS_SetBinFltrMask	EVS_BinFilter, EVS_Cmd, EVS_EvtGen	
EVS_EnaPort	EVS_Cmd, EVS_Reset	
EVS_EnaPortMask	EVS_Cmd	
EVS_DisPort	EVS_Cmd, EVS_Reset	
EVS_DisPortMask	EVS_Cmd	
EVS_RstBinFltrCtr	EVS_BinFilter, EVS_Cmd	
EVS_RstAllFltrs	EVS_BinFilter, EVS_Cmd	
EVS_AddEvtFltr	EVS_BinFilter	
EVS_DelEvtFltr	EVS_BinFilter	
EVS_WriteAppData2File	EVS_BinFilter, EVS_Cmd, EVS_EvtGen, EVS_Reset	
EVS_WriteLog2File	EVS_EvtGen, EVS_Log, EVS_Reset	
EVS_SetLogMode	EVS_Log, EVS_Reset	
EVS_ClrLog	EVS_Log	
SB_NOOP	SB_EnablePipe	
SB_ResetCtrs	SB_DisablePipe	
SB_DumpStats	SB_DisablePipe	
SB_WriteRouting2File	SB_Reset, SB_DisablePipe, SB_EnablePipe	
SB_EnaRoute	SB_CmdsErr, SB_Reset, SB_DisablePipe, SB_EnablePipe	
SB_DisRoute	SB_CmdsErr, SB_DisablePipe, SB_EnablePipe	
SB_DumpNetwork	SB_DisablePipe	
SB_WritePipe2File	SB_EnablePipe	
SB_WriteMap2File	SB_DisablePipe	
SB_EnaSubRptg		
SB_DisSubRptg		
SB_SendPrevSubs		
TBL_NOOP	TBL_CMD	
TBL_ResetCtrs	TBL_CMD	
TBL_Load	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_Dump	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_Validate	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_Activate	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_WriteReg2File	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_TLMReg	TBL_Functionality	
TBL_DeleteCDS	TBL_Reset	
TBL_LoadAbort	TBL_CMD, TBL_Functionality	
TIME_NOOP	TIME_CmdTlm	

Command	Test Procedure(s)	Notes/Comments
TIME_ResetCtrs	TIME_CmdTlm	
TIME_RequestDiag	TIME_Reset	
TIME_SetSource	TIME_CmdTlm	
TIME_SetState	TIME_CmdTlm, TIME_Reset	
TIME_AddClockLat	TIME_CmdTlm	
TIME_SubClockLat	TIME_CmdTlm	
TIME_SetClock	TIME_CmdTlm	
TIME_SetClockMET	TIME_CmdTlm	
TIME_SetClockSTCF	TIME_CmdTlm, TIME_Reset	
TIME_SetClockLeap	TIME_CmdTlm, TIME_Reset	
TIME_AddSTCFAdj	TIME_CmdTlm	
TIME_SubSTCFAdj	TIME_CmdTlm	
TIME_Add1HzSTCF	TIME_CmdTlm	
TIME_Sub1HzSTCF	TIME_CmdTlm	
TIME_StopAdd1Hz	TIME_CmdTlm	
TIME_StopSub1Hz	TIME_CmdTlm	
TIME_SetSignal	TIME_CmdTlm	

Telemetry	Test Procedure(s)	Notes/Comments
ES_CMDPC	ES_App_Ctrl, ES_Logging, ES_Reset	
ES_CMDEC	ES_App_Ctrl, ES_Logging, ES_Reset	
ES_CKSUM	ut_runproc	
ES_CFEMAJORVER	ut_runproc	
ES_CFEMINORVER	ut_runproc	
ES_CFEREVISION	ut_runproc	
ES_CFEMSNREV	ut_runproc	
ES_OSMAJORVER	ut_runproc	
ES_OSMINORVER	ut_runproc	
ES_OSREVISION	ut_runproc	
ES_OSMISSIONREV	ut_runproc	
ES_SYSLOGBYTEUSED	ES_Logging, ES_Reset	
ES_SYSLOGSIZE	ES_Logging	
ES_SYSLOGENTRIES	ES_Logging, ES_Reset	
ES_SYSLOGMODE	ES_Logging	
ES_ERLOGINDEX	ES_Logging	
ES_ERLOGENTRIES	ES_Logging	
ES_RegCoreApps	ES_Reset, ES_App_Ctrl	
ES_RegExtApps	ES_Reset, ES_App_Ctrl	
ES_RegTasks	ES_Reset	
ES_RegLibs	ES_Reset	
ES_ResetType	ES_Logging; ES_Reset	
ES_ResetSubtype	ES_Logging; ES_Reset	
ES_ProcResetCnt	ES_Logging; ES_Reset	
ES_MaxProcResets	ES_Logging	
ES_BootSource	ES_Reset	
ES_PerfState	ES_Logging	
ES_PerfMode		
ES_PerfTrigCnt		

ES_PerfFltrMask	ES_Logging	
ES_PerfTrigMask	ES_Logging	
ES_PerfDataStart		
ES_PerfDataEnd		
ES_PerfDataCnt	ES_Logging	
ES_PerfData2Write		
ES_HeapBytesFree		
ES_HeapBlocksFree		
ES_HeapMaxBlkSize		
ES_AppID	ES_App_Ctrl	
ES_AppType	ES_App_Ctrl	
ES_AppName	ES_App_Ctrl	
ES_AppEntryPt	ES_App_Ctrl	
ES_AppFilename	ES_App_Ctrl	
ES_StackSize	ES_App_Ctrl	
ES_ModuleID	ES_App_Ctrl	
ES_AddrsValid	ES_App_Ctrl	
ES_CodeAddress	ES_App_Ctrl	
ES_CodeSize	ES_App_Ctrl	
ES_DataAddress	ES_App_Ctrl	
ES_DataSize	ES_App_Ctrl	
ES_BSSAddress	ES_App_Ctrl	
ES_BSSSize	ES_App_Ctrl	
ES_StartAddr	ES_App_Ctrl	
ES_ExceptnActn	ES_App_Ctrl	
ES_Priority	ES_App_Ctrl	
ES_MainTaskId	ES_App_Ctrl	
ES_ExecutionCtr	ES_App_Ctrl	
ES_MainTaskName	ES_App_Ctrl	
ES_ChildTasks	ES_App_Ctrl	
ES_PooHandle	ES_App_Ctrl	
ES_PoolSize	ES_App_Ctrl	
ES_BlksREQ	ES_App_Ctrl	
ES_BlkErrCTR	ES_App_Ctrl	
ES_FreeBytes	ES_App_Ctrl	
ES_BlockStats.BlockSize	ES_App_Ctrl	
ES_BlockStats.BlocksCreated	ES_App_Ctrl	
ES_BlockStats.BlocksFree	ES_App_Ctrl	
EVS_APPNAME	pseudo tlm	
EVS_EVENTID	pseudo tlm	
EVS_EVENTTYPE	pseudo tlm	
EVS_SCID	pseudo tlm	
EVS_PROCESSORID	pseudo tlm	
EVS_EVENT	pseudo tlm	
EVS_CMDPC	EVS_BinFltr; EVS_Cmds	
EVS_CMDEC	EVS_BinFltr; EVS_Cmds	
EVS_MSGFMTMODE	EVS_BinFltr; EVS_Log; EVS_Reset	
EVS_MSGTRUNC	EVS_Cmds; EVS_EvtGen	

EVS_UNREGAPPC	EVS_Cmds; EVS_EvtGen	
EVS_OUTPUTPORT	EVS_Cmds; EVS_Reset	
EVS_LOGFULL	EVS_Log; EVS_Reset	
EVS_LOGMODE	EVS_BinFltr; EVS_Log; EVS_Reset	
EVS_MSGSENTC	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
EVS_LOGOVERFLOWC	EVS_Log; EVS_Reset	
EVS_LogState		
EVS_APP.APPID	EVS_Reset	
EVS_APP.APPMSGSENTC	EVS_BinFltr; EVS_Reset	
EVS_APP.APPENASTAT	EVS_BinFltr; EVS_EvtGen; EVS_Reset	
SB_CMDPC	SB_DisablePipe; SB_Reset	
SB_CMDEC	SB_DisablePipe; SB_Reset	
SB_NoSubEC	SB_DisablePipe; SB_EnablePipe; SB_Reset	
SB_MsgSndEC	SB_DisablePipe; SB_EnablePipe	
SB_MsgRecEC	SB_DisablePipe; SB_Reset	
SB_InternalEC		
SB_NewPipeEC	SB_DisablePipe	
SB_SubscrEC	SB_Reset	
SB_DupSubCnt	SB_Reset	
SB_PipeOvrEC	SB_DisablePipe	
SB_MsgLimEC	SB_DisablePipe	
SB_MemPoolHdl		
SB_MemInUse		
SB_UnmarkedMem		
SB_Stat.SB_SMMIDIU	SB_DisablePipe	
SB_Stat.SB_SMPMIDIU	SB_DisablePipe	
SB_Stat.SB_SMMMIDALW	SB_DisablePipe	
SB_Stat.SB_SMPIU	SB_DisablePipe	
SB_Stat.SB_SMPPIU	SB_DisablePipe	
SB_Stat.SB_SMMPALW	SB_DisablePipe	
SB_Stat.SB_SMBMIU	SB_DisablePipe	
SB_Stat.SB_SMPBMIU	SB_DisablePipe	
SB_Stat.SB_SMMBMALW	SB_DisablePipe	
SB_Stat.SB_SMSIU	SB_DisablePipe	
SB_Stat.SB_SMPSIU	SB_DisablePipe	
SB_Stat.SB_SMMSALW	SB_DisablePipe	
SB_Stat.SB_SMSBBIU	SB_DisablePipe	
SB_Stat.SB_SMPSBBIU	SB_DisablePipe	
SB_Stat.SB_SMMPDALW	SB_DisablePipe	

SB_Stat.SB_SMPDS.SB_PDPIPEID	SB_DisablePipe	
SB_Stat.SB_SMPDS.SB_PDDEPTH	SB_DisablePipe	
SB_Stat.SB_SMPDS.SB_PDINUSE	SB_DisablePipe	
SB_Stat.SB_SMPDS.SB_PDPKINUSE	SB_DisablePipe	
TBL_CMDPC	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_CMDEC	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_NumTables	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_NumUpdatesPend		No real way to test this
TBL_ValCompltdCtr	TBL_CMD	
TBL_LastValCRC	TBL_Functionality	
TBL_LastValS	TBL_Reset, TBL_Functionality	
TBL_LastValBuf	TBL_CMD, TBL_Functionality	
TBL_LastValTblName	TBL_Functionality	
TBL_ValSuccessCtr	TBL_CMD	
TBL_ValFailedCtr	TBL_CMD	
TBL_ValReqCtr	TBL_CMD	
TBL_NumFreeShrBuf	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_MemPoolHdl		
TBL_LastUpdTime.TBL_Seconds	TBL_CMD, TBL_Functionality	
TBL_LastUpdTime.TBL_SubSeconds	TBL_CMD, TBL_Functionality	
TBL_LastUpdTblName	TBL_CMD, TBL_Functionality	
TBL_LastFileLoaded	TBL_CMD, TBL_Functionality	
TBL_LastFileDumped	TBL_CMD, TBL_Functionality	
TBL_Size	TBL_Functionality, TBL_Reset	
TBL_CRC		
TBL_ActBufAdd	TBL_Functionality	
TBL_IActBufAdd	TBL_Functionality	
TBL_ValFuncPtr	TBL_Functionality	
TBL_TimeLastUpd.TBL_TLUSeconds	TBL_Functionality	
TBL_TimeLastUpd.TBL_TLUSubSeconds	TBL_Functionality	
TBL_FILECSECONDS	TBL_Functionality	
TBL_FILECSUBSECONDS	TBL_Functionality	
TBL_LoadedOnce	TBL_Functionality	
TBL_UpdatePending	TBL_Functionality	
TBL_DumpOnly	TBL_Reset, TBL_Functionality	
TBL_DblBuffered	TBL_Functionality	
TBL_Name	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_LastFileUpd	TBL_CMD, TBL_Reset, TBL_Functionality	
TBL_OwnerApp	TBL_Functionality	

TBL_CritFlag	TBL_Functionality	
TIME_CMDPC	TIME_CmdTlm	
TIME_CMDEC	TIME_CmdTlm	
TIME_FlagSet	TIME_Reset	
TIME_FlagFly	TIME_CmdTlm; TIME_Reset	
TIME_FlagSrc		
TIME_FlagPri	cFE_AltImage	
TIME_FlagSfly	TIME_Reset	
TIME_FlagCfly	TIME_CmdTlm; TIME_Reset	
TIME_FlagAdjd		
TIME_Flag1Hzd	TIME_CmdTlm	
TIME_FlagClat		
TIME_FlagSorC		
TIME_APIState	TIME_Reset	
TIME_LeapSecs	TIME_CmdTlm; TIME_Reset	
TIME_METSecs	TIME_CmdTlm	
TIME_METSubsecs	TIME_CmdTlm	
TIME_STCFSecs	TIME_Reset	
TIME_STCFSubsecs	TIME_Reset	
TIME_1HzAdjSecs	TIME_CmdTlm	
TIME_1HzAdjSSecs	TIME_CmdTlm	
TIME_DTMETS	TIME_CmdTlm	
TIME_DTMETSs		
TIME_DSTCFS	TIME_CmdTlm; TIME_Reset	
TIME_DSTCFSS	TIME_CmdTlm; TIME_Reset	
TIME_DLatentS	TIME_Reset	
TIME_DLatentSs	TIME_Reset	
TIME_DTValidS		
TIME_DTValidSs		
TIME_DLeapS	TIME_CmdTlm; TIME_Reset	
TIME_DAPIState	TIME_Reset	
TIME_DElapsedS		
TIME_DElapsedSS		
TIME_DLocalS		
TIME_DLocalSS		
TIME_DMETS	TIME_CmdTlm; TIME_Reset	
TIME_DMETSS	TIME_CmdTlm; TIME_Reset	
TIME_DTAIS	TIME_CmdTlm	
TIME_DTAISS	TIME_CmdTlm	
TIME_DUTCS	TIME_CmdTlm	
TIME_DUTCSS	TIME_CmdTlm	
TIME_DValid		
TIME_DFlywheel		
TIME_Dsource		
TIME_Dsignal		

TIME_DSrvFly		
TIME_DCMD2Fly		
TIME_DFlagSet	TIME_Reset	
TIME_DFlagFly	TIME_CmdTlm; TIME_Reset	
TIME_DFlagSrc	TIME_CmdTlm	
TIME_DFlagPri	TIME_CmdTlm; TIME_Reset	
TIME_DFlagSfly	TIME_Reset	
TIME_DFlagCfly	TIME_CmdTlm; TIME_Reset	
TIME_DFlagAdjd		
TIME_DFlag1Hzd		
TIME_DFlagClat		
TIME_DFlagSorC		
TIME_DAdjustDir	TIME_CmdTlm	
TIME_D1HzAdjDir	TIME_CmdTlm	
Time_DLatentDir		
Time_DAdjustS	TIME_CmdTlm	
Time_DAdjustSS	TIME_CmdTlm	
Time_D1HzAdjS	TIME_CmdTlm	
Time_D1HzAdjSS	TIME_CmdTlm	
TIME_DTTS		
TIME_DTTSS		
TIME_DTDS		
TIME_DTDSS		
Time_DVerifyCNT	TIME_CmdTlm	
Time_DVerifyER	TIME_CmdTlm	
Time_DTSDetCNT	TIME_CmdTlm	
Time_DTatTCNT	TIME_CmdTlm	
Time_DTsISRCNT		
Time_DTsISRERR		
Time_DTsTaskCNT	TIME_CmdTlm	
Time_DVersionCNT	TIME_CmdTlm	
Time_D1HzISRCNT	TIME_CmdTlm	
Time_D1HzTaskCNT	TIME_CmdTlm	
Time_DLogicalMET		
Time_DMinWindow		
Time_DMaxWindow		
Time_DWrapS		
Time_DWrapSS		
Time_DMaxSS		
Time_DMinSS		
Time_DataStStat		

File Telemetry	**Test Procedure(s)**	**Notes/Comments**
RF.TBL_Size	TBL_Functionality	
RF.TBL_SysTime.TBL_ST_Seconds	TBL_CMD, TBL_Functionality	
RF.TBL_SysTime.TBL_ST_Subseconds	TBL_CMD, TBL_Functionality	
RF.TBL_NumUsers	TBL_Functionality	

RF.TBL_LoadBufferID	TBL_CMD, TBL_Reset, TBL_Functionality	
RF.TBL_FileCreateSeconds		
RF.TBL_FileCreateSubseconds		
RF.TBL_RegCRC		
RF.TBL_ValFuncPresent	TBL_Functionality	
RF.TBL_LoadedOnce	TBL_Functionality	
RF.TBL_UpdatePndng	TBL_Functionality	
RF.TBL_DumpOnly	TBL_Reset, TBL_Functionality	
RF.TBL_DblBuffered	TBL_Functionality	
RF.TBL_Name	TBL_CMD, TBL_Reset, TBL_Functionality	
RF.TBL_LastFileUpd	TBL_CMD, TBL_Reset, TBL_Functionality	
RF.TBL_OwnerAppName		
RF.TBL_Critical	TBL_Functionality	
SB_RouteEntry.SB_MsgId	SB_DisablePipe; SB_EnablePipe; SB_Reset	
SB_RouteEntry.SB_PipeId	SB_DisablePipe; SB_EnablePipe; SB_Reset	
SB_RouteEntry.SB_State	SB_EnablePipe; SB_Reset	
SB_RouteEntry.SB_MsgCnt	SB_DisablePipe; SB_EnablePipe;	
SB_RouteEntry.SB_AppName	SB_Reset	
SB_RouteEntry.SB_PipeName	SB_DisablePipe; SB_EnablePipe; SB_Reset	
PE.SBPF_InUse		
PE.SBPF_PipeID		
PE.SBPF_PipeName	SB_EnablePipe	
PE.SBPF_AppName		
PE.SBPF_TaskId		
PE.SBPF_SysQId		
PE.SBPF_LastSender		
PE.SBPF_Qdepth		
PE.SBPF_SendErrs		
PE.SBPF_Buffer		
SB_MsgMapEntry.SB_MM_MID		
SB_MsgMapEntry.SB_MM_INDEX		
EVS_LOG.EvtLogEntry.AppName	EVS_Log	
EVS_LOG.EvtLogEntry.EvtId	EVS_Log	
EVS_LOG.EvtLogEntry.EvtType	EVS_Log	
EVS_LOG.EvtLogEntry.ScId	EVS_Log	
EVS_LOG.EvtLogEntry.PrcId	EVS_Log	
EVS_Log.EvtMsg	EVS_Log	
EVS_AppData.AppName	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	

EVS_AppData.ActiveFlag	EVS_Cmds; EVS_EvtGen; EVS_Reset	
EVS_AppData.EvtTypeAF	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
EVS_AppData.EventCounter	EVS_Cmds; EVS_EvtGen	
EVS_AppData.BinFltr.EvtId	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
EVS_AppData.BinFltr.Msk	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
EVS_AppData.BinFltr.Ctr	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
ES_ERLE.ERLog_EntryType		
ES_ERLE.ERLog_ResetType	ES_Reset	
ES_ERLE.ERLog_ResetSubType	ES_Reset	
ES_ERLE.ERLog_BootSource		
ES_ERLE.ERLog_ProcessorResetCnt		
ES_ERLE.ERLog_MaxProcResetCnt		
ES_ERLE.ERLog_DebugFlag		
ES_ERLE.ERLog_WatchDogWriteFlag		
ES_ERLE.ERLog_PrintfEnabledFlag		
ES_ERLE.ERLog_LastAppID		
ES_ERLE.ERLog_Seconds		
ES_ERLE.ERLog_Subseconds		
ES_ERLE.ERLog_Description		
ES_ERLE.ERLog_ContextPresent		
ES_ERLE.ERLog_AppID		
ES_ERLE.ERLog_Context		
ES_ALE.ES_AL_AppId	ES_Logging, ES_Reset, ES_App_Ctrl	
ES_ALE.ES_AL_AppType	ES_App_Ctrl	
ES_ALE.ES_AL_AppName	ES_Logging, ES_Reset, ES_App_Ctrl	
ES_ALE.ES_AL_EntryPoint	ES_App_Ctrl	
ES_ALE.ES_AL_FileName	ES_App_Ctrl	
ES_ALE.ES_AL_StackSize	ES_App_Ctrl	
ES_ALE.ES_AL_ModuleID	ES_App_Ctrl	
ES_ALE.ES_AL_AddrsValid	ES_App_Ctrl	
ES_ALE.ES_AL_CodeAddr	ES_App_Ctrl	
ES_ALE.ES_AL_CodeSize	ES_App_Ctrl	
ES_ALE.ES_AL_DataAddr	ES_App_Ctrl	
ES_ALE.ES_AL_DataSize	ES_App_Ctrl	
ES_ALE.ES_AL_BSSAddr	ES_App_Ctrl	

ES_ALE.ES_AL_BSSSize	ES_App_Ctrl	
ES_ALE.ES_AL_StartAddr	ES_App_Ctrl	
ES_ALE.ES_AL_ExceptionAction	ES_App_Ctrl	
ES_ALE.ES_AL_Priority	ES_App_Ctrl	
ES_ALE.ES_AL_TaskId	ES_Logging, ES_Reset, ES_App_Ctrl	
ES_ALE.ES_AL_ExecutionCtr	ES_App_Ctrl	
ES_ALE.ES_AL_TaskName	ES_Logging, ES_Reset, ES_App_Ctrl	
ES_ALE.ES_AL_ChildTasks	ES_Reset	
ES_CDSReg.CDSHandle		
ES_CDSReg.CDSSize	ES_App_Ctrl	
ES_CDSReg.CriticalTBL	ES_Reset; TBL_Reset	
ES_CDSReg.CDSName	ES_App_Ctrl: TBL_Reset	
ES_TL.TaskId	ES_App_Ctrl	
ES_TL.ExecutionCtr		
ES_TL.TaskName	ES_App_Ctrl	
ES_TL.AppId	ES_App_Ctrl	
ES_TL.AppName	ES_App_Ctrl	

Id	Event Message	Test Procedure(s)	Notes/Comments
1	CFE_ES_INIT_INF_EID	Generated at cFE Startup	
2	CFE_ES_INITSTATS_INF_EID	Generated at cFE Startup	
3	CFE_ES_NOOP_INF_EID	ES_Reset; EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
4	CFE_ES_RESET_INF_EID	ES_Reset	
5	CFE_ES_SHELL_INF_EID	ES_AppCtrl	
6	CFE_ES_START_INF_EID	ES_AppCtrl; ES_Logging; ES_Reset; EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Log; EVS_Reset; SB_DisablePipe; SB_EnablePipe; SB_Reset; TBL_Cmd; TBL_Functionality; TBL_Reset; TIME_CmdTlm; TIME_Reset	
7	CFE_ES_STOP_DBG_EID	ES_AppCtrl; ES_Logging; TBL_Functionality; TBL_Reset;	
8	CFE_ES_STOP_INF_EID	ES_AppCtrl; ES_Logging; TBL_Functionality; TBL_Reset;	
9	CFE_ES_RESTART_APP_DBG_EID	ES_AppCtrl	
10	CFE_ES_RESTART_APP_INF_EID	ES_AppCtrl; ES_Logging;	
11	CFE_ES_RELOAD_APP_DBG_EID	ES_AppCtrl	
12	CFE_ES_RELOAD_APP_INF_EID	ES_AppCtrl	
13	CFE_ES_EXIT_APP_INF_EID		
14	CFE_ES_ERREXIT_APP_INF_EID		
15	CFE_ES_ONE_APP_EID	ES_AppCtrl	

Id	Event Message	Test Procedure(s)	Notes/Comments
16	CFE_ES_ALL_APPS_EID	ES_AppCtrl; ES_Logging; ES_Reset; EVS_Cmds; EVS_EvtGen; SB_Reset; TBL_Cmd; TBL_Functionality; TBL_Reset;	
17	CFE_ES_SYSLOG1_INF_EID	ES_Logging	
18	CFE_ES_SYSLOG2_EID	ES_AppCtrl, ES_Logging, ES_Reset	
19	CFE_ES_ERLOG1_INF_EID	ES_Logging	
20	CFE_ES_ERLOG2_EID	ES_Logging, ES_Reset; cFE_AltImage	
21	CFE_ES_MID_ERR_EID		
22	CFE_ES_CC1_ERR_EID	ES_Reset; EVS_Cmds; EVS_EvtGen;	
23	CFE_ES_LEN_ERR_EID		
24	CFE_ES_BOOT_ERR_EID		
25	CFE_ES_SHELL_ERR_EID		
26	CFE_ES_START_ERR_EID	ES_AppCtrl	
27	CFE_ES_START_INVALID_FILENAME_ERR_EID	ES_AppCtrl	
28	CFE_ES_START_INVALID_ENTRY_POINT_ERR_EID		
29	CFE_ES_START_NULL_APP_NAME_ERR_EID		
30	CFE_ES_START_STACK_ERR_EID	ES_AppCtrl	
31	CFE_ES_START_PRIORITY_ERR_EID		
32	CFE_ES_START_EXC_ACTION_ERR_EID		
33	CFE_ES_ERREXIT_APP_ERR_EID		
35	CFE_ES_STOP_ERR1_EID	ES_AppCtrl	
36	CFE_ES_STOP_ERR2_EID	ES_AppCtrl	
37	CFE_ES_STOP_ERR3_EID		
38	CFE_ES_RESTART_APP_ERR1_EID	ES_AppCtrl	
39	CFE_ES_RESTART_APP_ERR2_EID	ES_AppCtrl	
40	CFE_ES_RESTART_APP_ERR3_EID	ES_AppCtrl	
41	CFE_ES_RESTART_APP_ERR4_EID		
42	CFE_ES_RELOAD_APP_ERR1_EID	ES_AppCtrl	
43	CFE_ES_RELOAD_APP_ERR2_EID	ES_AppCtrl	
44	CFE_ES_RELOAD_APP_ERR3_EID	ES_AppCtrl	
45	CFE_ES_RELOAD_APP_ERR4_EID		
46	CFE_ES_EXIT_APP_ERR_EID		
47	CFE_ES_PCR_ERR1_EID		
48	CFE_ES_PCR_ERR2_EID		
49	CFE_ES_ONE_ERR_EID		
50	CFE_ES_ONE_APPID_ERR_EID	ES_AppCtrl	
51	CFE_ES_OSCREATE_ERR_EID	ES_AppCtrl	
52	CFE_ES_WRHDR_ERR_EID		
53	CFE_ES_TASKWR_ERR_EID		
55	CFE_ES_SYSLOG2_ERR_EID	ES_Logging;	
56	CFE_ES_ERLOG2_ERR_EID	ES_Logging;	
57	CFE_ES_PERF_STARTCMD_EID	ES_Logging;	
58	CFE_ES_PERF_STARTCMD_ERR_EID		
59	CFE_ES_PERF_STARTCMD_TRIG_ERR_EID		
60	CFE_ES_PERF_STOPCMD_EID	ES_Logging;	
61	CFE_ES_PERF_STOPCMD_ERR1_EID		

Id	Event Message	Test Procedure(s)	Notes/Comments
62	CFE_ES_PERF_STOPCMD_ERR2_EID		
63	CFE_ES_PERF_FILTMSKCMD_EID	ES_Logging;	
64	CFE_ES_PERF_FILTMSKERR_EID		
65	CFE_ES_PERF_TRIGMSKCMD_EID	ES_Logging;	
66	CFE_ES_PERF_TRIGMSKERR_EID		
67	CFE_ES_PERF_LOG_ERR_EID	ES_Logging;	
68	CFE_ES_PERF_DATAWRITTEN_EID	ES_Logging;	
69	CFE_ES_CDS_REGISTER_ERR_EID		
70	CFE_ES_SYSLOGMODE_EID	ES_AppCtrl; ES_Logging;	
71	CFE_ES_ERR_SYSLOGMODE_EID		
72	CFE_ES_RESET_PR_COUNT_EID	ES_Logging;	
73	CFE_ES_SET_MAX_PR_COUNT_EID	ES_Logging;	
74	CFE_ES_FILEWRITE_ERR_EID		
75	CFE_ES_RST_ACCESS_EID		
76	CFE_ES_CDS_DELETE_ERR_EID		
77	CFE_ES_CDS_NAME_ERR_EID	ES_AppCtrl	
78	CFE_ES_CDS_DELETED_INFO_EID	ES_AppCtrl	
79	CFE_ES_CDS_DELETE_TBL_ERR_EID	ES_AppCtrl	
80	CFE_ES_CDS_OWNER_ACTIVE_EID	ES_AppCtrl	
81	CFE_ES_TLM_POOL_STATS_INFO_EID	ES_AppCtrl	
82	CFE_ES_INVALID_POOL_HANDLE_ERR_EID	ES_AppCtrl	
83	CFE_ES_CDS_REG_DUMP_INF_EID	ES_AppCtrl; TBL_Reset;	
84	CFE_ES_CDS_DUMP_ERR_EID		
85	CFE_ES_WRITE_CFE_HDR_ERR_EID		
86	CFE_ES_CREATING_CDS_DUMP_ERR_EID	ES_AppCtrl;	
87	CFE_ES_TASKINFO_EID	ES_AppCtrl;	
88	CFE_ES_TASKINFO_OSCREATE_ERR_EID	ES_AppCtrl;	
89	CFE_ES_TASKINFO_WRHDR_ERR_EID		
90	CFE_ES_TASKINFO_WR_ERR_EID		
0	CFE_EVS_NOOP_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset	
1	CFE_EVS_STARTUP_EID		
2	CFE_EVS_ERR_WRLOGFILE_EID		
3	CFE_EVS_ERR_CRLOGFILE_EID	EVS_Log	
5	CFE_EVS_ERR_MSGID_EID		
6	CFE_EVS_ERR_EVTIDNOREGS_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; SB_DisablePipe; SB_Reset;	
7	CFE_EVS_ERR_APPNOREGS_EID		
8	CFE_EVS_ERR_ILLAPPIDRANGE_EID		
9	CFE_EVS_ERR_NOAPPIDFOUND_EID	EVS_Cmds; EVS_EvtGen;	
10	CFE_EVS_ERR_ILLEGALFMTMOD_EID		
11	CFE_EVS_ERR_MAXREGSFILTER_EID	EVS_BinFltr	
12	CFE_EVS_ERR_WRDATFILE_EID		
13	CFE_EVS_ERR_CRDATFILE_EID	EVS_Cmds	
15	CFE_EVS_ERR_CC_EID		
16	CFE_EVS_RSTCNT_EID		
17	CFE_EVS_SETFILTERMSK_EID	EVS_BinFltr; EVS_EvtGen	
18	CFE_EVS_ENAPORT_EID	EVS_Cmds; EVS_Reset;	
19	CFE_EVS_DISPORT_EID	EVS_Cmds; EVS_Reset;	

Id	Event Message	Test Procedure(s)	Notes/Comments
20	CFE_EVS_ENAEVTTYPE_EID	ES_AppCtrl; ES_Logging; ES_Reset; EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; SB_CmdsErr; SB_DisablePipe; SB_EnablePipe; SB_Reset; TBL_Cmd; TBL_Functionality; TBL_Reset; TIME_CmdTlm; cFE_AltImage	
21	CFE_EVS_DISEVTTYPE_EID	EVS_Cmds; EVS_Reset;	
22	CFE_EVS_SETEVTFMTMOD_EID	EVS_Reset;	
23	CFE_EVS_ENAAPPEVTTYPE_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen;	
24	CFE_EVS_DISAPPENTTYPE_EID	EVS_EvtGen;	
25	CFE_EVS_ENAAPPEVT_EID	EVS_Cmds; EVS_EvtGen;	
26	CFE_EVS_DISAPPEVT_EID	EVS_EvtGen;	
27	CFE_EVS_RSTEVTCNT_EID	EVS_Cmds;	
28	CFE_EVS_RSTFILTER_EID	EVS_BinFltr; EVS_Cmds;	
29	CFE_EVS_RSTALLFILTER_EID	EVS_BinFltr	
30	CFE_EVS_ADDFILTER_EID	ES_AppCtrl; ES_Logging; EVS_BinFltr; SB_DisablePipe; SB_EnablePipe; SB_Reset; TBL_Functionality;	
31	CFE_EVS_DELFILTER_EID	EVS_BinFltr; SB_DisablePipe; SB_EnablePipe; SB_Reset;	
32	CFE_EVS_WRDAT_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; SB_EnablePipe;	
33	CFE_EVS_WRLOG_EID	ES_Reset; EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; SB_EnablePipe;	
34	CFE_EVS_NO_LOGSET_EID		
35	CFE_EVS_NO_LOGCLR_EID		
36	CFE_EVS_NO_LOGWR_EID		
37	CFE_EVS_EVT_FILTERED_EID	EVS_BinFltr; SB_DisablePipe; SB_EnablePipe;	
38	CFE_EVS_LOGMODE_EID	EVS_Reset;	
39	CFE_EVS_ERR_LOGMODE_EID	EVS_EvtGen;	
40	CFE_EVS_ERR_INVALID_BITMASK_EID	EVS_Cmds;	
41	CFE_EVS_ERR_UNREGISTERED_EVS_APP	EVS_Cmds; EVS_EvtGen;	
42	CFE_EVS_FILTER_MAX_EID	EVS_BinFltr	
43	CFE_EVS_LEN_ERR_EID	EVS_Cmds	
1	CFE_SB_INIT_EID		
2	CFE_SB_CR_PIPE_BAD_ARG_EID	SB_DisablePipe;	
3	CFE_SB_MAX_PIPES_MET_EID	SB_DisablePipe; SB_EnablePipe; SB_Reset;	
4	CFE_SB_CR_PIPE_ERR_EID	SB_DisablePipe; SB_EnablePipe; SB_Reset	

Id	Event Message	Test Procedure(s)	Notes/Comments
5	CFE_SB_PIPE_ADDED_EID	ES_AppCtrl; ES_Logging; ES_Reset; EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; SB_DisablePipe; SB_EnablePipe; SB_Reset; TBL_Cmd; TBL_Functionality; TBL_Reset;	
6	CFE_SB_SUB_ARG_ERR_EID		
7	CFE_SB_DUP_SUBSCRIP_EID	ES_Logging; SB_Reset; TBL_Functionality;	
8	CFE_SB_MAX_MSGS_MET_EID	SB_DisablePipe;	
9	CFE_SB_MAX_DESTS_MET_EID	ES_AppCtrl; SB_DisablePipe; SB_EnablePipe; SB_Reset;	
10	CFE_SB_SUBSCRIPTION_RCVD_EID	ES_AppCtrl; ES_Logging; ES_Reset; EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; SB_DisablePipe; SB_EnablePipe; SB_Reset; TBL_Cmd; TBL_Functionality; TBL_Reset;	
11	CFE_SB_UNSUB_ARG_ERR_EID		
12	CFE_SB_UNSUB_NO_SUBS_EID	SB_Reset	
13	CFE_SB_SEND_BAD_ARG_EID		
14	CFE_SB_SEND_NO_SUBS_EID	ES_AppCtrl; SB_EnablePipe; SB_CmdsErr; SB_EnablePipe; SB_Reset	
15	CFE_SB_MSG_TOO_BIG_EID	SB_CmdsErr; SB_DisablePipe; SB_EnablePipe	
16	CFE_SB_GET_BUF_ERR_EID		
17	CFE_SB_MSGID_LIM_ERR_EID	ES_AppCtrl; ES_Logging; SB_DisablePipe; SB_EnablePipe	
18	CFE_SB_RCV_BAD_ARG_EID	SB_DisablePipe; SB_Reset;	
19	CFE_SB_BAD_PIPEID_EID	SB_EnablePipe;	
20	CFE_SB_DEST_BLK_ERR_EID		
21	CFE_SB_SEND_INV_MSGID_EID		
22	CFE_SB_SUBSCRIPTION_RPT_EID		
24	CFE_SB_UNSUBSCRIPTION_RPT_EID		
25	CFE_SB_Q_FULL_ERR_EID	SB_DisablePipe	
26	CFE_SB_Q_WR_ERR_EID		
27	CFE_SB_Q_RD_ERR_EID		
28	CFE_SB_CMD0_RCVD_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; SB_CmdsErr; SB_EnablePipe;	
29	CFE_SB_CMD1_RCVD_EID	SB_DisablePipe	
30	CFE_SB_LSTSNDER_ERR1_EID		
31	CFE_SB_LSTSNDER_ERR2_EID		
32	CFE_SB_SND_STATS_EID	SB_DisablePipe; SB_EnablePipe	
33	CFE_SB_ENBL_RTE1_EID	SB_CmdsErr; SB_EnablePipe	
34	CFE_SB_ENBL_RTE2_EID	SB_DisablePipe; SB_EnablePipe	

Id	Event Message	Test Procedure(s)	Notes/Comments
35	CFE_SB_ENBL_RTE3_EID	SB_CmdsErr; SB_Reset;	
36	CFE_SB_DSBL_RTE1_EID	SB_CmdsErr; SB_EnablePipe	
37	CFE_SB_DSBL_RTE2_EID	SB_DisablePipe; SB_EnablePipe	
38	CFE_SB_DSBL_RTE3_EID	SB_CmdsErr	
39	CFE_SB_SND_RTG_EID	SB_DisablePipe; SB_EnablePipe; SB_Reset;	
40	CFE_SB_SND_RTG_ERR1_EID	SB_DisablePipe; SB_EnablePipe	
41	CFE_SB_GLS_INV_CALLER_EID		
42	CFE_SB_BAD_CMD_CODE_EID	EVS_Cmds; EVS_EvtGen; SB_CmdsErr;	
43	CFE_SB_BAD_MSGID_EID		
44	CFE_SB_FULL_SUB_PKT_EID		
45	CFE_SB_PART_SUB_PKT_EID		
46	CFE_SB_DEL_PIPE_ERR1_EID		
47	CFE_SB_PIPE_DELETED_EID	ES_AppCtrl; ES_Logging; SB_EnablePipe; SB_Reset; TBL_Functionality; TBL_Reset;	
48	CFE_SB_SUBSCRIPTION_REMOVED_EID	ES_AppCtrl; ES_Logging; SB_EnablePipe; SB_Reset; TBL_Functionality; TBL_Reset;	
49	CFE_SB_FILEWRITE_ERR_EID		
50	CFE_SB_SUB_INV_PIPE_EID	SB_Reset;	
51	CFE_SB_SUB_INV_CALLER_EID		
52	CFE_SB_UNSUB_INV_PIPE_EID		
53	CFE_SB_UNSUB_INV_CALLER_EID		
54	CFE_SB_DEL_PIPE_ERR2_EID		
1	CFE_TBL_INIT_INF_EID		
10	CFE_TBL_NOOP_INF_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; TBL_Cmd	
11	CFE_TBL_RESET_INF_EID	TBL_Cmd	
12	CFE_TBL_FILE_LOADED_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
13	CFE_TBL_OVERWRITE_DUMP_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
14	CFE_TBL_WRITE_DUMP_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
15	CFE_TBL_OVERWRITE_REG_DUMP_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
16	CFE_TBL_VAL_REQ_MADE_INF_EID	TBL_Cmd: TBL_Functionality; TBL_Reset	
17	CFE_TBL_LOAD_PEND_REQ_INF_EID	TBL_Cmd: TBL_Functionality; TBL_Reset	
18	CFE_TBL_TLM_REG_CMD_INF_EID	TBL_Functionality	
21	CFE_TBL_LOAD_ABORT_INF_EID	TBL_Cmd; TBL_Functionality	
22	CFE_TBL_WRITE_REG_DUMP_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
23	CFE_TBL_ASSUMED_VALID_INF_EID	TBL_Functionality	

Id	Event Message	Test Procedure(s)	Notes/Comments
35	CFE_TBL_LOAD_SUCCESS_INF_EID	ES_AppCtrl; TBL_Cmd; TBL_Functionality; TBL_Reset;	
36	CFE_TBL_VALIDATION_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
37	CFE_TBL_UPDATE_SUCCESS_INF_EID	TBL_Cmd; TBL_Functionality; TBL_Reset	
38	CFE_TBL_CDS_DELETED_INFO_EID	TBL_Reset;	
50	CFE_TBL_MID_ERR_EID		
51	CFE_TBL_CC1_ERR_EID	EVS_EvtGen; EVS_Cmds; TBL_Cmd;	
52	CFE_TBL_LEN_ERR_EID		
53	CFE_TBL_FILE_ACCESS_ERR_EID	TBL_Cmd; TBL_Functionality	
54	CFE_TBL_FILE_STD_HDR_ERR_EID		
55	CFE_TBL_FILE_TBL_HDR_ERR_EID		
56	CFE_TBL_FAIL_HK_SEND_ERR_EID		
57	CFE_TBL_NO_SUCH_TABLE_ERR_EID	TBL_Functionality; TBL_Reset	
58	CFE_TBL_FILE_TYPE_ERR_EID		
59	CFE_TBL_FILE_SUBTYPE_ERR_EID		
60	CFE_TBL_NO_WORK_BUFFERS_ERR_EID	TBL_Functionality	
61	CFE_TBL_INTERNAL_ERROR_ERR_EID		
62	CFE_TBL_CREATING_DUMP_FILE_ERR_EID	TBL_Functionality	
63	CFE_TBL_WRITE_CFE_HDR_ERR_EID		
64	CFE_TBL_WRITE_TBL_HDR_ERR_EID		
65	CFE_TBL_WRITE_TBL_IMG_ERR_EID		
66	CFE_TBL_NO_INACTIVE_BUFFER_ERR_EID	TBL_Functionality	
67	CFE_TBL_TOO_MANY_VALIDATIONS_ERR_EID		
68	CFE_TBL_WRITE_TBL_REG_ERR_EID		
69	CFE_TBL_LOAD_ABORT_ERR_EID		
70	CFE_TBL_ACTIVATE_ERR_EID	TBL_Cmd; TBL_Functionality	
71	CFE_TBL_FILE_INCOMPLETE_ERR_EID		
72	CFE_TBL_LOAD_EXCEEDS_SIZE_ERR_EID	TBL_Cmd; TBL_Functionality	
73	CFE_TBL_ZERO_LENGTH_LOAD_ERR_EID		
74	CFE_TBL_PARTIAL_LOAD_ERR_EID		
75	CFE_TBL_FILE_TOO_BIG_ERR_EID	TBL_Cmd	
76	CFE_TBL_TOO_MANY_DUMPS_ERR_EID		
77	CFE_TBL_DUMP_PENDING_ERR_EID		
78	CFE_TBL_ACTIVATE_DUMP_ONLY_ERR_EID	TBL_Functionality	
79	CFE_TBL_LOADING_A_DUMP_ONLY_ERR_EID	TBL_Functionality	
80	CFE_TBL_ILLEGAL_BUFF_PARAM_ERR_EID	TBL_Functionality;	
81	CFE_TBL_UNVALIDATED_ERR_EID	TBL_Functionality	
82	CFE_TBL_IN_REGISTRY_ERR_EID	TBL_Reset	
83	CFE_TBL_NOT_CRITICAL_TBL_ERR_EID		
84	CFE_TBL_NOT_IN_CRIT_REG_ERR_EID	TBL_Reset	
85	CFE_TBL_CDS_NOT_FOUND_ERR_EID		
86	CFE_TBL_CDS_DELETE_ERR_EID		
87	CFE_TBL_CDS_OWNER_ACTIVE_ERR_EID	TBL_Reset	
88	CFE_TBL_LOADING_PENDING_ERR_EID		
89	CFE_TBL_FAIL_NOTIFY_SEND_ERR_EID		
90	CFE_TBL_REGISTER_ERR_EID	TBL_Functionality; TBL_Reset	
91	CFE_TBL_SHARE_ERR_EID		
92	CFE_TBL_UNREGISTER_ERR_EID		

Id	Event Message	Test Procedure(s)	Notes/Comments
93	CFE_TBL_LOAD_ERR_EID	TBL_Functionality	
94	CFE_TBL_LOAD_TYPE_ERR_EID		
95	CFE_TBL_UPDATE_ERR_EID		
96	CFE_TBL_VALIDATION_ERR_EID	TBL_Cmd; TBL_Functionality;	
97	CFE_TBL_SPACECRAFT_ID_ERR_EID	TBL_Validate	
98	CFE_TBL_PROCESSOR_ID_ERR_EID	TBL_Validate	
1	CFE_TIME_INIT_EID		
4	CFE_TIME_NOOP_EID	EVS_BinFltr; EVS_Cmds; EVS_EvtGen; EVS_Reset; TIME_CmdTlm	
5	CFE_TIME_RESET_EID	TIME_CmdTlm	
6	CFE_TIME_DIAG_EID	TIME_CmdTlm	
7	CFE_TIME_STATE_EID	TIME_CmdTlm; TIME_Reset	
8	CFE_TIME_SOURCE_EID		
9	CFE_TIME_SIGNAL_EID		
11	CFE_TIME_DELAY_EID		
12	CFE_TIME_TIME_EID	TIME_CmdTlm	
13	CFE_TIME_MET_EID	TIME_CmdTlm	
14	CFE_TIME_STCF_EID	TIME_CmdTlm; TIME_Reset	
15	CFE_TIME_DELTA_EID	EVS_Log; TIME_CmdTlm	
16	CFE_TIME_1HZ_EID	TIME_CmdTlm; cFE_AltImage;	
17	CFE_TIME_LEAPS_EID	TIME_CmdTlm; TIME_Reset	
20	CFE_TIME_FLY_ON_EID		
21	CFE_TIME_FLY_OFF_EID		
25	CFE_TIME_EXIT_ERR_EID		
26	CFE_TIME_ID_ERR_EID		
27	CFE_TIME_CC_ERR_EID	EVS_Cmds; EVS_EvtGen;	
30	CFE_TIME_STATE_ERR_EID		
31	CFE_TIME_SOURCE_ERR_EID	TIME_CmdTlm	
32	CFE_TIME_SIGNAL_ERR_EID		
33	CFE_TIME_DELAY_ERR_EID		
34	CFE_TIME_TIME_ERR_EID		
35	CFE_TIME_MET_ERR_EID		
36	CFE_TIME_STCF_ERR_EID		
37	CFE_TIME_DELTA_ERR_EID		
38	CFE_TIME_1HZ_ERR_EID		
40	CFE_TIME_SOURCE_CFG_EID	TIME_CmdTlm	
41	CFE_TIME_SIGNAL_CFG_EID	TIME_CmdTlm	
42	CFE_TIME_DELAY_CFG_EID	TIME_CmdTlm	
43	CFE_TIME_TIME_CFG_EID		
44	CFE_TIME_MET_CFG_EID		
45	CFE_TIME_STCF_CFG_EID		
46	CFE_TIME_LEAPS_CFG_EID		
47	CFE_TIME_DELTA_CFG_EID		
48	CFE_TIME_1HZ_CFG_EID		

APPENDIX B - TEST STATUS MATRIX

Test Name	Status	Date	Seconds	Minutes	Comments
sb_cmds_err	Passed	5/23/2016	134.068	2.23447	
sb_dispipes	Passed	5/23/2016	3175.84	52.9307	
sb_enapipes	Passed	5/24/2016	2880.84	48.0122	
sb__reset	Passed	5/24/2016	1130.29	18.8382	
tbl_cmding	Passed	5/23/2016	3222.55	52.7092	
tbl_func	Passed	5/23/2016	8029.45	133.824	
tbl_reset	Passed	5/23/2016	3810.68	63.5147	
time_command_server_tai	Passed	5/24/2016	1251.14	20.8523	
time_resets_server_tai	Passed	5/23/2016	353.749	5.89581	
evs_log	Passed	5/23/2016	978.78	16.313	
evs_cmds	Passed	5/24/2016	3356.74	55.9457	
evs_evt_msg_gen	Passed	5/23/2016	2070.73	34.5122	
evs_reset	Passed	5/23/2016	1267.09	21.1181	
evs_bin_fltr	Passed	5/23/2016	13394.8	223.246	
es_appctrl	Passed	5/24/2016	1935.75	32.2625	
es_logging	Failed	5/24/2016	1770.5	29.5083	CPU rebooted when RestartApp was expected
es_reset	Passed	5/23/2016	1748.41	29.1401	
cfe_altimage	Passed	5/23/2016	289.01	4.81684	
cfe_myeh	Failed	5/25/2016	280.364	4.67274	User-defined exception did not get called
cfe_osobjfailure	Failed	5/26/2016	104.014	1.73357	CPU did not reboot as expected on failure
		Total Time:	51184.685	853.078	