NASA GSFC FLIGHT SOFTWARE SYSTEMS BRANCH

FSW VERSION DESCRIPTION DOCUMENT

CFS CHECKSUM (CS) APPLICATION

BUILD: CS 2.4.2

RELEASE DATE: 10/7/2020

1.0 FSW VERSION DESCRIPTION

1.1 PURPOSE AND SUMMARY

This build is a minor build of the Checksum (CS) application to provide compatibility with cFS Bootes. This build provides compatibility with cFS Bootes and cFE 6.8 as well as some minor code cleanup.

1.2 NEW/CHANGED FUNCTIONALITY IN THIS VERSION

Table 1.2-1 identifies the DCRs that have been implemented in this FSW version. For each DCR the "Key" column shows the corresponding DCR in the GSFC cFS tracking system.

NOTE: As of this writing, there is an open cFE Framework ticket that affects the execution of CS. The module name used to find the cFE Text Segment is incorrect, and as a result the cFE baseline CRC calculation will fail. Additional details and status on the cFE Framework ticket are here: https://github.com/nasa/PSP/issues/111

Table 1.2-1 – DCRs Implemented in this Version

Key	Summary	Description
GSFCCFS-754	CS: Fix compiler error using with the latest build scripts	The "CS" application attempts to verify that CS_MAX_NUM_APP_TABLE_ENTRIES is not greater than CFE_ES_MAX_APPLICATIONS and a similar check for CS_MAX_NUM_TABLES_TABLE_ENTRIES. This verification breaks if CFE_ES_MAX_APPLICATIONS is not known at compile time, which is an issue when "cfe_platform_cfg.h" is not available to apps. However, this is really not needed at all because the runtime code already checks this and it fails with a very obvious error message if limits are exceeded.
GSFCCFS-927	CS Build Warnings	Submitted by Allen Brown: Fixed build warnings: o CS_ComputeTables() TblInfo local variable needs to be initialized to {0}; o CS_ComputeApp() AppInfo local var also needs to be init to {0}; o CS_ProcessNewTablesDefinitionTable() name string local variables need an initial "\0".
GSFCCFS-1253	CS Does not check return value of CFE_PSP_GetCFETextSegmentInfo	In CS_AppInit, CS does not check the return value of CFE_PSP_GetCFETextSegmentInfo. The values of CFEAddress and CFESize that are retrieved from this function are used in subsequent calculations of the CFE baseline CRC. That CRC computation should not be enabled if the function fails. The return value of CFE_PSP_GetCFETextSegmentInfo should be checked and handled following the same pattern as CFE_PSP_GetKernelTextSegmentInfo.

GSFCCFS-963	CS Unit tests run failure because of stack smashing	While attempting to run CS unit tests, this is the output: ./cs_testrunner.exe *** stack smashing detected ***: ./cs_testrunner.exe terminated makefile:165: recipe for target 'run' failed make: *** [run] Aborted Tracking down which test causes this ended up with line 917 in cs_cmds_test.c: UtTest_Add(CS_BackgroundCheckCmd_Test_InvalidMsgLength, CS_Test_Setup, CS_Test_TearDown, "CS_BackgroundCheckCmd_Test_InvalidMsgLength"); Commenting out this line allowed the tests to run without issue: ./cs_testrunner.exe Tests Executed: 318 Assert Pass Count: 1456 Assert Fail Count: 0 There appears to be some issue in CS_BackgroundCheckCmd_Test_InvalidMsgLength that is causing the failure.
GSFCCFS-1067	CS Readme needs updates	Readme file contains out of date information on sources for cFE and OSAL.
GSFCCFS-1143	CS uses OS_FS_* Error Codes (soon deprecated)	CS uses OS_FS_* error codes which will soon be deprecated in cFE (details: https://github.com/nasa/osal/issues/262)
GSFCCFS-1157	CS does not build against cFE 6.8 with OMIT_DEPRECATED=TRUE and -Werror	
GSFCCFS-1174	CS has uninitialized variable	In analysis on 7/10/2020, CodeSonar flagged the following red warning. Uninitialized Variable help TblInfo was not initialized. TblInfo was defined at cs_compute.c:139.
GSFCCFS-1212	Add files to generate CS doxygen documentation	add config file that generates CS doxygen documentation
GSFCCFS-1233	CS may have alignment problems on some platforms	CS uses uint8[] for command and telemetry packet headers. This can cause alignment issues (this has been experienced with other apps). Instead of the uint8[], the command and telemetry packets should use the actual header types to ensure alignment.
GSFCCFS-1237	CS does not build tables "out of the box"	
GSFCCFS-1238	CS does not compute unique cFE Core Checksum	While running the cs_corecode test with CS 2.4.2.0 and cFE Bootes (6.8), the initial cFE Baseline CRC is calculated but every subsequent background CRC calculation generates a different CRC.

No new functionality was added in this build.

1.3 MISSING PLANNED FEATURES AND KNOWN PROBLEMS

Table 1.3-1 identifies currently open DCRs that are not addressed in this build.

Any workarounds that may apply are identified.

Refer to the Delivery Letter for any additional DCRs submitted after preparation of this VDD.

Table 1.3-1 – Currently open DCRs

Key	Summary	Description
GSFCCFS-1178	CS has static code analysis findings	In analysis on 7/10/2020, CodeSonar flagged the attached findings.

GSFCCFS-1154	Incorrect data type in CS_HousekeepingCmd_Test_Nominal()	The CS_HousekeepingCmd_Test_Nominal() function in the cs_app_test.c file sets:
		CS_AppData.RecomputeInProgress = 10;
		CS_AppData.OneShotInProgress = 11;
		The associated member attributes in the HK message are uint8, but the member attribute in the CS_AppData struct is type boolean. This still compiles and passed with older versions of the OSAL, but with OSAL 6.7.3-bv this compiles but fails unit test. I recommend changing assignments from 10 and 11 to both true, and the equality check from 10 and 11 to 1. When this change is made, CS passes unit testing.

GSFCCFS-964	CS test CS_BackgroundTables_Test_Miscompare failing when some tests do not run	Fail results occurred while tracking down another issue in the unit tests. When all tests in cs_cmds_test.c were commented out, there were no failures; however, while tracking down an issue in this file, certain combinations of tests not being added (commented out) would cause this result: FAIL: CS_AppData.TablesCSErrCounter == 1, File: cs_utils_test.c, Line: 929 FAIL: Checksum Failure: Table name, Expected: 0x00000001, Calculated: 0x00000002, File: cs_utils_test.c, Line: 933 FAIL: Ut_CFE_EVS_GetEventQueueDepth() == 1, File: cs_utils_test.c, Line: 939 One example: UtTest_Add(CS_BackgroundCheckCmd_Test_InvalidMsgLength, CS_Test_Setup, CS_Test_TearDown, "CS_BackgroundCheckCmd_Test_InvalidMsgLength"); // UtTest_Add(CS_BackgroundCheckCmd_Test_BackgroundCfeCore, CS_Test_Setup, CS_Test_TearDown, "CS_BackgroundCheckCmd_Test_BackgroundCfeCore"); Although this was not the only time this occurred. Unit tests should be independent and not be affected by other tests whether they run or not.

GSFCCFS-952	Checksum errors after restarting app	This description is from an email chain with multiple teams describing a bug in CS with downstream impacts on HS. A short summary is included here and the full email chain is attached to this issue. 1. Application gets an exception 2. ES restarts the application 3. There is a high probability that the application checksum has changed due to the restart. 4. Checksum is not notified to recompute the application checksum. When it finds the checksum has changed, it puts out an event message to that effect. 5. HS is monitoring for application checksum messages, and does a processor reset as a consequence.
GSFCCFS-951	ES - RegisteredTasks Counter Does Not Decrement When Child Tasks are Exited	From babelfish (ticket #95): Project team reported that the ES "RegisteredTasks" counter was not decrementing when a child task exited. The CS code was examined to ensure the needed ES API calls were being made. It was confirmed the CS child tasks make the ES API call to CFE_ES_ExitChildTask. The CFE_ES_ExitChildTask API function does decrement the RegisteredTasks counter on line 1337 however, this step is performed only if conditional checks are passed.
GSFCCFS-929	CS CRC values may lead to confusion	The CS application relies on the CFE_ES_CalculateCRC function to report the CRCs that it calculates. However, this function calculates the CRC as an int16 value and returns that value as a uint32 value. This causes the sign bit of the int16 (bit 15) to be propagated to the upper 16 bits of the uint32. If this bit is set, it can cause confusion since a CRC of 0x845E would result in the CS application having a value of 0xFFFF845E.

2.0 DELIVERED PRODUCTS

Table 2-1 identifies the locations of FSW products relevant to this FSW Build. The version or date of the Build and where the product can be located are provided. Changes from a previous VDD are identified.

Table 2-1 – Delivered Products and their Locations

Software Element	Changed with this Version?	New Version or Date	Location
Source Code of this FSW Build	Yes	2.4.2	https://github.com/nasa/cs
Doxygen Documentation	Yes	N/A	https://github.com/nasa/cs

Software Element	Changed with this Version?	New Version or Date	Location
Unit Test Data	Yes	2.4.2	https://github.com/nasa/cs
FSW Make Files	Yes	2.4.2	https://github.com/nasa/cs

3.0 INSTALLATION PROCEDURES

In order to build and install the CS application, it must be added to the cFE CMake build system. This is done by modifying the TGTX_APPLIST in the cFE targets.cmake file. This is shown in the trivial example below.

```
SET(TGT1_NAME cpu1)
SET(TGT1_APPLIST cs)
SET(TGT1_FILELIST cfe_es_startup.scr)
```

After CS is added to the targets.cmake file, it is built and installed using the standard cFE CMake build instructions. These instructions are available in cFE CMake documentation:

https://github.com/nasa/cFE/blob/main/cmake/README.md

4.0 CONFIGURATION SUMMARY AND VERSION IDENTIFICATION

This software can be found in the CS GitHub repository (https://github.com/nasa/CS) under the tag “CS-2.4.2”.

Verification of the version can be done by sending a CS NOOP command that produces an event message containing the version information. In addition, the initialization event message generated during the application startup provides the version information.

ACRONYMS

ACS	Attitude Control System
C&DH	Command and Data Handling
cFS	Core Flight System
CM	Configuration Management
CS	Checksum
COTS	Commercial Off-The-Shelf
CRC	Cyclic Redundancy Check
DCR	Discrepancy/Change Request
ETU	Engineering Test Unit
FSB	Flight Software Branch
FSW	Flight Software
H&S	Health & Safety
I&T	Integration & Test
PSP	Platform Support Package
RTOS	Real-Time Operating System
T&C	Telemetry and Command
URL	Universal Resource Locator
VDD	Version Description Document