NASA GSFC FLIGHT SOFTWARE SYSTEMS BRANCH

FSW VERSION DESCRIPTION DOCUMENT

CFS HEALTH & SAFETY (HS) APPLICATION

BUILD: HS 2.3.2

RELEASE DATE: 10/05/2020

1.0 FSW VERSION DESCRIPTION

1.1 PURPOSE AND SUMMARY

This build is a minor build of the Health & Safety (HS) application to resolve bugs with the HS application. This build does not include any new functionality. The primary purpose of this release is to ensure compatibility between the HS application and cFS Bootes.

This document serves as the notification of the Build 2.3.2 release of the cFS HS application.

Health & Safety (HS) version 2.3.2 is compatible with cFE builds 6.8.0 and above and OSAL builds 5.0 and above.

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.

Table 1.2-1 – DCRs Implemented in this Version

Key	Summary	Description
GSFCCFS-999	Readme Updates	HS readme needs a number of updates.
GSFCCFS-1000	Possible buffer overflow	In hs_monitors, lines 568, 655, and 745, static code analysis finding - possible buffer overflow, might not leave enough space for null terminator. Tam suggests "Even though null-terminator is assigned in the next statement, recommend strncpy OS_MAX_API_NAME-1 to get rid of static code analyzer finding"
GSFCCFS-1001	Else if chain with no else clause	In following functions there are else-if chains with no else clause. HS_MonitorApplications HS_MonitorUtilization HS_HousekeepingReq
GSFCCFS-1002	Multiple instances of special math	Special math in functions: HS_ValidateEMTable HS_ValidateAMTable
GSFCCFS-1004	Uninitialized variables in HS	Some uninitialized variables throughout. In HS_ValidateAMTable, static code analysis issue with initialization of 'char BadName[OS_MAX_API_NAME] = "";'
GSFCCFS-1005	documentation updates in hs_custom.h	Several function prototypes need their doxygen documentation fleshed out.

GSFCCFS-1006	Possible buffer overflow	HS_custom.c, line 382 Static code analysis finding - buffer overflow; array index out-of-bounds; array DiagValue might use index 16. Need to check i value first before referencing it.
GSFCCFS-1007	No null pointer checks on function arguments	
GSFCCFS-1008	Bad extern designation	In hs_app.h: extern HS_AppData_t HS_AppData; Shouldn't extern in the header defining the packet, overloads where it's actually defined (hs_app.c). Cause of static analysis issue.
GSFCCFS-1009	Field sizes in HS_AppData_t don't match housekeeping packet	
GSFCCFS-1010	Add code to handle failure from HS_CustomInit	Currently, if HS_CustomInit fails in HS_AppInit, no action is taken. The return code is captured, but nothing is done.
GSFCCFS-1011	Fix return value in HS_AppInit	HS_AppInit returns hard-coded CFE_SUCCESS. Should just return status
GSFCCFS-1012	Initialize app data to zero	Recommend memset 0 HS_AppData, then set just non-zero elements (or do at start of entry to app). Doesn't look like HS8000 (power on reset initialization) is explicitly implemented. Would allow some initialization lines to be removed from HS_SbInit
GSFCCFS-1013	Update paths to work with cFE v6.6	Update table filename paths to work out-of-the-box with cFE 6.6 (change from "/cf/apps/filename" to "/cf/filename")

GSFCCFS-1015	TBDs in CFS_HS_Design.ppt	Documentation updates needed in CFS_HS_Design.ppt: --TBD item on page 6 needs to be provided. Text as follows: "How are infinite reset loops prevented? Processor Resets caused by HS are limited to a maximum of <TBD, Configuration parameter>" --TBD item on page 12 under the first bullet "HS will perform the following CPU related functions:" --TBD item on page 13 under the first bullet "HS has a fixed number of execution counter slots in its housekeeping packet" --TBD item on page 20 under the second bullet "Power-On Reset"
GSFCCFS-1106	HS CMakeLists.txt file does not build tables	
GSFCCFS-1118	HS8005.1 part B is not implemented in the code	From IV&V gap analysis: In the HS8005.1 requirement ID, “If the execution Counter Table fails validation, HS shall a) Issue an event message and b) Report 0x’FFFFFFFF’ for all <PLATFORM_DEFINED> items in the table.” The rationale behind this requirement is that if the file does not exist to populate the Execution Counter Table (or fails validation), HS should continue to execute other functions and report an identifiable number, such as x’FFFFFFFF’, for entries using the configuration parameter. On lines 566-572 of the hs_app.c file, the first part is satisfied in the CFE_TBL_Load() call and return of something other than CFE_SUCCESS. An event message “Error Loading ExeCount Table” will be sent. The second part defined in b) of the requirement is missing.
GSFCCFS-1125	Update HS Doxygen User Guide	Suggest updating the doxygen user guide file and adding a doxygen configuration file to allow users to successfully generate the doxygen guide themselves.
GSFCCFS-1138	Location of HS_CustomData prevents linking on RTEMS	HS_CustomData_t HS_CustomData in hs_custom.h prevents proper linking using RTEMS/SPARC/GCC due to multiple definitions, and so should be moved to hs_custom.c with an extern in hs_custom.h (consistent with how cFS apps handle AppData with _app.h and _app.c)
GSFCCFS-1151	HS should build with -Werror the OMIT_DEPRECATED enabled	HS should build against the latest cFE with -Werror and OMIT_DEPRECATED enabled.

GSFCCFS-1232	HS may have alignment problems on some platforms	HS 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-1246	HS Version number is not correct for Release Candidate 2.3.2	The hs_version.h file still has 2.3.0 specified as the version.

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.

Table 1.3-1 – Currently open DCRs

Key	Summary	Description
GSFCCFS-1208	Performance ID usage is inconsistent in HS app	Anytime a task goes into a system delay (for example a pend on message receipt or task delay), an app should do the following: CFE_ES_PerfLogExit(<performance_id>); <do the delay function> CFE_ES_PerfLogEntry(<performance_id>); There appear to be some cases in HS where a task delay called, but the performance monitor is not exited. Other apps should also be checked for consistent usage.
GSFCCFS-1182	HS has static code analysis findings	In analysis done on 7/10/2020, CodeSonar flagged the attached findings.

GSFCCFS-1172	HS does not appear to handle short format of events	This was an issue exposed when addressing ticket GSFCCFS-1151 (compatibility with cFE 6.8). For event monitoring, HS previously relied on long format events (subscribing to CFE_EVS_EVENT_MSG_MID in HS_AppMain and handling CFE_EVS_Packet_t in HS_MonitorEvent). In cFE 6.8, these no longer exist, and in fact were separated to handle long format and short format event messages differently. The backwards compatibility defines in cFE map these things to the event long formats. typedef CFE_EVS_LongEventTlm_t CFE_EVS_Packet_t; #define CFE_EVS_EVENT_MSG_MID CFE_EVS_LONG_EVENT_MSG_MID For purposes of resolving GSFCCFS-1151, occurrences of the deprecated items were fixed with the long format, but in the long term, HS needs updates to handle monitoring short format event messages.
GSFCCFS-1089	Migrate HS unit tests to distributed UT Assert	
GSFCCFS-1014	Scrub configuration file for items that don't change	Suggest removing non-project configuration items from this file. Would help to limit this list to only items that projects should expect to manage. Examples of things I wouldn't consider for project configuration - app name, wakeup pipe depth, table names, etc. Note open ticket on CPU utilization/idle task so avoiding comments on those parameters since I expect this implementation to change.
GSFCCFS-1003	Recommended refactoring in HS_MonitorEvent	In HS_MonitorEvent, refactor common action logic for the different monitors into one function, and pass in unique info. Would avoid repeated logic for the same action from a different trigger.

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	2.3.2	https://github.com/nasa/hs
Doxygen Documentation	Yes	N/A	https://github.com/nasa/hs
Unit Test Data	Yes	2.3.2	https://github.com/nasa/hs
FSW Make Files	Yes	2.3.2	https://github.com/nasa/hs

3.0 INSTALLATION PROCEDURES

In order to build and install the HS 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 hs)
SET(TGT1_FILELIST cfe_es_startup.scr)
```

After HS 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 HS GitHub repository (https://github.com/nasa/HS) under the tag "2.3.2".

Verification of the version can be done by sending an HS NOOP command which 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
COTS ..... Commercial Off-The-Shelf
CPU ..... Central Processing Unit
DCR ..... Discrepancy/Change Request
ETU ..... Engineering Test Unit
FSB ..... Flight Software Branch
FSW ..... Flight Software
GSFC ..... Goddard Space Flight Center
HS ..... Health & Safety
I&T ..... Integration & Test
JSC ..... Johnson Space Center
POSIX ..... Portable Operating System Interface
RTOS ..... Real-Time Operating System
SMP ..... Symmetric Multiprocessing
T&C ..... Telemetry and Command
TBD ..... To Be Determined
URL ..... Universal Resource Locator
VDD ..... Version Description Document