



NASA GSFC FLIGHT SOFTWARE SYSTEMS BRANCH

FSW VERSION DESCRIPTION DOCUMENT

CFS MEMORY MANAGER (MM) APPLICATION

BUILD: MM 2.4.2

RELEASE DATE: 10/5/2020

1.0 FSW VERSION DESCRIPTION

1.1 PURPOSE AND SUMMARY

The purpose of this build is to continue to refine the cFS Memory Manager (MM) application product. This build provides various bug fixes and enhancements, but does not include any new functionality. The primary purpose of this release is to ensure compatibility between the MM application and cFS Bootes.

This document serves as the notification of the Build 2.4.2 release of the cFS MM application.

Memory Manager (MM) version 2.4.2 is compatible with cFE builds 6.8.0 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-906	MM must be tested with cFE 6.6	
GSFCCFS-923	MM functions have incorrect stride when dumping memory	MM_DumpMem16ToFile() and MM_DumpMem32ToFile() have incorrect stride when dumping memory. (MM_DumpMemToFile() did work, but strictly-aligned access is required for some platforms.) -- You don't get what you want -- You can get a segfault or illegal access -- The MM unit tests didn't actually test this functionality. They didn't confirm if the correct pattern was actually dumped. Likely this is pervasive for all the unit tests?
GSFCCFS-1072	MM Readme file needs updates	Readme file includes out of date information on sources for cFE and OSAL.
GSFCCFS-1137	MM Performance Logging doesn't exit	Memory Manger (MM) calls CFE_ES_PerfLogEntry(MM_EEPROM_FILELOAD_PERF_ID); but it doesn't call CFE_ES_PerfLogExit(MM_EEPROM_FILELOAD_PERF_ID);
GSFCCFS-1144	MM uses OS_FS_SUCCESS Codes (soon deprecated)	MM uses OS_FS_SUCCESS code which may soon be deprecated in cFE (details: https://github.com/nasa/osal/issues/262)
GSFCCFS-1156	MM does not build against cFE 6.8 with OMIT_DEPRECATED=true and -Werror	
GSFCCFS-1214	Add files to generate MM doxygen user guide	MM needs a configuration file to generate HTML documentation with doxygen.
GSFCCFS-1231	MM may have alignment problems on some platforms	MM 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-1247	MM_Fill command generates Write errors for MEM32 memory type	When sending a MM_Fill command using the MEM32 memory type, an error is generated indicating a PSP memory write error. Also, when attempting to dump memory to a file, an OS Write error was generated.

1.2.1 CHANGED FUNCTIONALITY

One potential functionality change is introduced in this version of MM. In cFS Bootes, the OSAL functions OS_IntLock and OS_IntUnlock were deprecated (see <https://github.com/nasa/osal/issues/385> and <https://github.com/nasa/PSP/issues/180> for details). These functions were used in the MM “Write Memory with Interrupts Disabled” (MM_WriteMemWID). In MM version 2.4.2, these functions have simply been removed from the MM_WriteMemWID function. **Users should be aware that this function no longer disables interrupts before writing to memory.** Users may or may not see a difference in functionality as the OS_IntLock and OS_IntUnlock functions did not previously work on all platforms. A longer-term approach to interrupt locking for this command will be investigated for the next release (ticket GSFCCFS-1132 – see Section 1.3).

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.

Information on currently open DCRs is available at:

<https://etdjira.gsfc.nasa.gov/projects/GSFCCFS/issues>

Note that this is a restricted website that requires a server account. Additional DCRs may have been submitted after preparation of this VDD. A cFS MM DCR report containing a listing of open DCRs is available upon request for customers who do not have access to the restricted server. Please contact Elizabeth Timmons, elizabeth.timmons@nasa.gov.

Table 1.3-1 – Currently open DCRs

Key	Summary	Description
GSFCCFS-1132	Reevaluate how MM uses intlock	<p>Memory Manager uses OS_IntLock() and OS_IntUnlock(). That's the only place any of the OS_Int* functions are used in the GSFC apps. These functions will likely be deprecated in the next OSAL release.</p> <p>I'm guessing it does this with the intent of loading the mem block "atomically" with the hope of preventing another task from writing to it while this happens?</p> <p>IntLock/Unlock has always been a no-op on POSIX, and furthermore even on platforms where it does something, it will not achieve that exclusivity effect on Multi-Core CPUs, as it only locks the interrupts on the core which calls it, and other cores continue to run anyway, interrupt or not.</p> <p>Probably worth re-evaluating what MM is trying to achieve with the intlock... might be able to simply take it out with no loss of function.</p>

GSFCCFS-925	MM Unclear app configuration assumptions	Suggest adding comments to the config file that some app configuration values have assumptions on modulo-2 or modulo-4 sizes. For example, these 16 and 32-bit dump routines would fail if these config #defines are not set with the appropriate values.
GSFCCFS-924	MM Unit tests are fragile	Unit tests are fragile because they assume a particular app configuration. This makes them very troublesome for reuse (can't unit test your particular deployed config).
GSFCCFS-1184	MM has static code analysis findings	In analysis done on 7/10/2020, CodeSonar flagged several findings.

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/MM
Doxygen Documentation	Yes	N/A	https://github.com/nasa/MM
Unit Test Data	Yes	2.4.2	https://github.com/nasa/MM
FSW Make Files	Yes	2.4.2	https://github.com/nasa/MM

3.0 INSTALLATION PROCEDURES

In order to build and install the MM 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 mm)
SET(TGT1_FILELIST cfe_es_startup.scr)
```

After MM 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 MM GitHub repository (<https://github.com/nasa/MM>) under the tag “2.4.2”.

Verification of the version can be done by sending an MM 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
I&T.....	Integration & Test
JSC	Johnson Space Center
MM	Memory Manager
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