F021 Flash API v2.00.01

Release Notes

July 11, 2013

The Texas Instruments' F021 Flash API provides functions that can be used to erase, program and verify F021 Flash on TI Hercules 65nm devices.

The version 2.x.x series of the Flash API is the first series to follow ISO26262 development flow.

TABLE OF CONTENTS

1 New In This Release

v2.00.01

- No new features in this release

v2.00.00

- Replaced the user defined callback functions Fapi_setupEepromSectorEnable() and Fapi_setupBankSectorEnable() with the functions Fapi_enableEepromBankSectors() and Fapi_enableMainBankSectors.
- Deprecated the function Fapi_waitDelay().
- Removed the header files F021_FMC_BE.h and F021_FMC_LE.h as F021.h has been updated to automatically determine compile endianness.
- Replaced the Fapi_initializeAPI() function with Fapi_initializeFlashBanks(). With this change, all global variables have been removed from the API.
- Added the Compatibility.h header file. This file contains some backwards compatibility macros to work with projects that were previously built with v1.51 of the API. The list of functions and global variables 2.00.00 with compatibility defines are:
  - Fapi_initializeAPI()
  - Fapi_getFsmStatus()
  - Fapi_issueFsmSuspendCommand()
  - Fapi_writeEwaitValue(mEwait)
  - Fapi_checkFsmForReady()
  - Fapi_GlobalInit.m_poFlashControlRegisters
- Fapi_getBankSectors() was updated to return sector sizes in kilobytes and to support 256kB sectors, au8SectorSizes which was an array uint8_t was changed to an array of uint16_t and renamed au16SectorSizes.
- Added Fapi_remapMainAddress() to give an easy method to determine ECC address for a main flash address.
- Removed unused status' from Fapi_StatusType
  - Fapi_Status_AsyncBusy
  - Fapi_Status_AsyncComplete
  - Fapi_Error_StateMachineTimeout
  - Fapi_Error_InvalidDelayValue
  - Fapi_Error_InvalidCpu
- Removed the listing of structures. Please refer to the installed F021 Flash API headers files for these.
- Changed from the use of defined typedefs uint64, uint32, uint16, and uint8 to the standard definitions in stdint.h, uint64_t, uint32_t, uint16_t, and uint8_t. Also changed boolean to boolean_t
- Added #if defined guardbanding around the defines in Types.h that can conflict with Autosar Platform_Types.h defines.
- Added appendix to reference guide describing the PSA calculation

2 Release Contents

The following API files are distributed with the installer:

- Library Files - *(All library files were built using TI's code generation tools for ARM v5.0.1 with the following compile options: -mv7R4 --abi=eabi --strict_ansi -g -O3 --symdebug:dwarf_version=3 --diag_warning=225 --gen_func_subsections=on --enum_type=packed --code_state=16)*
  - F021_API_CortexR4_BE.lib – This is the Flash API object file for Cortex R4 Big Endian devices.
  - F021_API_CortexR4_BE_v3D16.lib – This is the Flash API object file for Cortex R4 Big Endian devices that are using floating point unit. *(In addition to the general build options, this library was built using : --float_support=VFPv3D16)*
  - F021_API_CortexR4_BE_L2FMC.lib – This is the Flash API object file for Cortex R4 Big Endian devices using the L2FMC memory controller. F021_API_CortexR4_LE.lib – This is the Flash API object file for Cortex R4 Little Endian devices. *(In addition to the general build options, this library was built using : -me)*
  - F021_API_CortexR4_LE_v3D16.lib – This is the Flash API object file for Cortex R4 Little Endian devices that are using floating point unit. *(In addition to the general build options, this library was built using : -me --float_support=VFPv3D16)*
  - F021_API_CortexR4_LE_L2FMC.lib – This is the Flash API object file for Cortex R4 Little Endian devices using the L2FMC memory controller.
- Source Files
  - Fapi_UserDefinedFunctions.c – This is file that contains the user definable functions.
- Include Files
  - F021.h – This is the master include file and includes all other include files. This should be the only include file added to the users's code.

  The following include files should not be included directly by the user's code, but are listed here for user reference:
  - Compatibility.h - A set of macros to be used for backwards compatibility for 1.x.x versions of the API.
  - Constants.h – Constant definitions used by the API.
  - FapiFunctions.h - Contains all the Fapi function prototypes.
  - Helpers.h – Set of helper defines
  - Registers.h – Definitions common to all register implementations and includes the appropriate register include file for the selected device type.
    - Registers_FMC_BE.h – Big Endian Flash memory controller registers structure for TMS570/RM4 devices.
    - Registers_FMC_LE.h – Little Endian Flash memory controller registers structure for TMS570/RM4 devices.
  - Types.h – Contains all the enumerations and structures used by the API

  Below are a set of compiler specific support header files:
  - CGT.ARM.h - Contains a set of definitions used by the ARM compiler
  - CGT.CCS.h - Contains a set of definitions used by the TI CCS compiler
  - CGT.gcc.h - Contains a set of definitions used by the gcc compiler
  - CGT.GHS.h - Contains a set of definitions used by the GreenHills compiler
  - CGT.IAR.h - Contains a set of definitions used by the IAR EWARM compiler

- Library information files
  - build_information.txt - This file contains function callgraphs, worst case stack usage for each function, function size in bytes and MD5 and SHA1 checksums for all files delivered in the installer package.
  - License_Agreement.pdf - This is library's license agreement.
  - readme.txt - This file contains release specific information.
  - Release_Notes.pdf - This file.
  - spna148.pdf- This is the application note, Advanced F021 Flash API Erase/Program Usage.
  - spnu501.pdf – This is the reference guide for the library.
  - spnz210.pdf - This is the library errata document.

3 Fixed In This Release

v2.00.01

Reference	Description
SDOCM00102084	Typo in CGT.CCS.H in GNU attribute check
SDOCM00102399	Restored FEDACSDIS and FEDACSDIS2 definitions

v2.00.00

Reference	Description
SDOCM00094147	Incorrect read in Verify functions in ECC regions on LE devices

4 Known Issues

None Known.