



SafeFLASH NAND Driver for MT29FxG01 SPI User Guide

Version 2.00

For use with SafeFLASH NAND Driver for Micron®
MT29FxG01 SPI versions 1.03 and above

Table of Contents

1. System Overview	3
1.1. Introduction	4
1.2. Feature Check	5
1.3. Device Description	6
1.4. Fail-safety	6
1.5. Packages and Documents	7
1.6. Change History	8
2. Source File List	9
3. Configuration Options	11
4. PSP Porting	12
4.1. psp_safe_nand_mt29fxg01_spi_init	13
5. Version	14

1. System Overview

This chapter contains the fundamental information for this module.

The component sections are as follows:

- [Introduction](#) - describes the main elements of the module.
- [Feature Check](#) - summarizes the main features of the module as bullet points.
- [Device Description](#) - summarizes the Micron devices supported.
- [Fail-safety](#) - defines fail-safety and describes the quality of service that SafeFLASH provides.
- [Packages and Documents](#) - the *Packages* section lists the packages that you need in order to use this module. The *Documents* section lists the relevant user guides.
- [Change History](#) - lists the earlier versions of this manual, giving the software version that each manual describes.

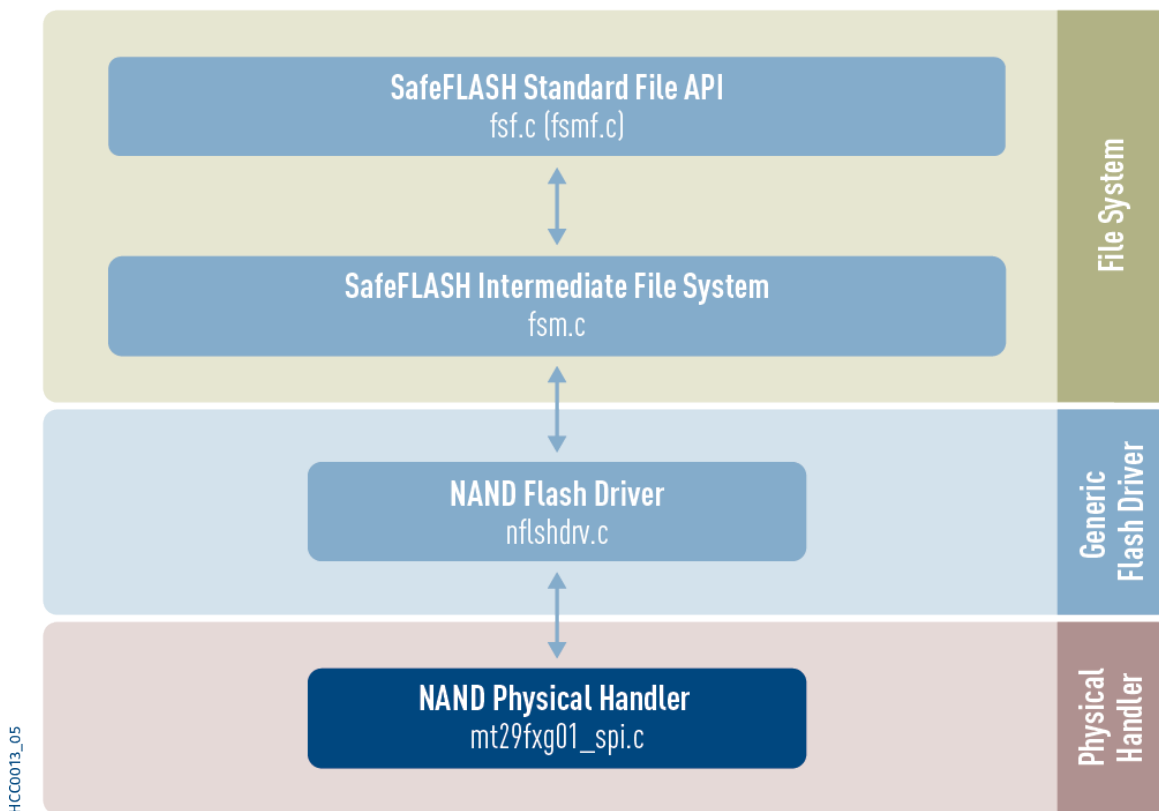
All rights reserved. This document and the associated software are the sole property of HCC Embedded. Reproduction or duplication by any means of any portion of this document without the prior written consent of HCC Embedded is expressly forbidden.

HCC Embedded reserves the right to make changes to this document and to the related software at any time and without notice. The information in this document has been carefully checked for its accuracy; however, HCC Embedded makes no warranty relating to the correctness of this document.

1.1. Introduction

This guide is for those who want to implement an HCC Embedded SafeFLASH NAND driver for MT29FxG01 Serial Peripheral Interface (SPI) flash and similar devices from Micron® Technology Inc. This is for use with HCC's SafeFLASH file system.

The following diagram illustrates the structure of the file system software:



The main SafeFLASH package provides the file API and intermediate file system. This is described in the [HCC SafeFLASH File System User Guide](#). This design is highly portable and gives excellent performance. The basic device architecture includes a high level driver for each general media type that shares some common properties. This driver handles issues of FAT maintenance, wear leveling, and so on.

Also shown in this diagram are the following:

- The NAND Flash driver - the device driver for the NAND flash, provided by the base NAND package. This is described in the [HCC SafeFLASH File System NAND Drive User Guide](#).
- The NAND physical handler - provided by this module, this performs the translation between the driver and the physical flash hardware.

Note: HCC Embedded offers hardware and firmware development consultancy to assist developers with the implementation of flash file systems.

1.2. Feature Check

The main features of the module are the following:

- Conforms to the HCC Advanced Embedded Framework.
- Designed for integration with both RTOS and non-RTOS based systems.
- Supports Micron® MT29FxG01 SPI NAND flash and is easily configurable for similar NAND flash parts.
- Supports static and dynamic wear leveling.
- Provides bad block management.

1.3. Device Description

For Micron devices, the digit after the "F" gives the size in Gb, that is 2 Gb.

The address lines are multiplexed with the data input/output signals on a multiplexed x16 input/output bus.

1.4. Fail-safety

This driver for MT29FxG01 Serial NAND flash is designed as part of HCC's SafeFLASH file system. SafeFLASH guarantees a defined level of fail-safety (see the [SafeFLASH File System User Guide](#)). For the system to be able to guarantee fail-safety, each component must provide a defined quality of service.

For this driver the following must be guaranteed to ensure the system is fail-safe:

- All write operations must be committed to flash in the sequence in which they are provided to the driver.
- Any write operation that fails must return an error.
- Any erase operation that fails must return an error.
- The system must ensure that there is at most one partially complete write or erase operation. At this point the file system should be restarted so that it can be recovered.

To achieve this, the hardware should ensure that, in the event of a falling voltage approaching the specified minimum programming level of the flash, the system either resets or provides a signal to the software to block access to the flash.

An alternative solution is to add capacitance to the design. This must provide sufficient power that, after a hardware error or reset condition is detected, the active operation on the flash can be completed.

Only by using one of these techniques can the system guarantee correct operation even after an unexpected system reset.

1.5. Packages and Documents

Packages

The table below lists the packages that you need in order to use this module:

Package	Description
<code>hcc_base_doc</code>	This contains the two guides that will help you get started.
<code>fs_safe_nand</code>	The SafeFLASH NAND flash driver.
<code>fs_safe_nand_drv_mt29fxg01_spi</code>	The SafeFLASH NAND driver for Micron MT29FxG01 SPI package described in this document.
<code>psp_template_spi</code>	The SPI Platform Support Package (PSP).

Documents

For an overview of HCC file systems and guidance on choosing a file system, see [Product Information](#) on the main HCC website.

Readers should note the points in the [HCC Documentation Guidelines](#) on the HCC documentation website.

HCC Firmware Quick Start Guide

This document describes how to install packages provided by HCC in the target development environment. Also follow the *Quick Start Guide* when HCC provides package updates.

HCC Source Tree Guide

This document describes the HCC source tree. It gives an overview of the system to make clear the logic behind its organization.

HCC SafeFLASH File System User Guide

This document describes the base SafeFLASH System.

HCC SafeFLASH File System NAND Drive User Guide

This document describes the SafeFLASH NAND driver that is used with the MT29FxG01 SPI driver.

HCC SafeFLASH NAND Driver for Micron MT29FxG01 SPI User Guide

This is this document.

1.6. Change History

This section describes past changes to this manual.

- To view or download earlier manuals, see [File System PDFs](#).
- For the history of changes made to the package code itself, see [History: fs_safe_nand_drv_mt29fxg01_spi](#).

The current version of this manual is 2.00. The full list of versions is as follows:

Manual version	Date	Software version	Reason for change
2.00	2020-02-03	1.03	New template.
1.40	2019-10-11	1.03	Added standard PSP function and macros to <i>PSP Porting</i> . Added SPI PSP functions to <i>PSP Porting</i> .
1.30	2018-07-06	1.03	Added second note to <i>PSP Files</i> section in <i>Source Files</i> .
1.20	2017-08-31	1.03	Corrected <i>Packages</i> list.
1.10	2017-06-26	1.03	New <i>Change History</i> format.
1.00	2017-04-24	1.03	First online version.

2. Source File List

The following sections describe all the source code files included in the system. These files follow the HCC Embedded standard source tree system, described in the [HCC Source Tree Guide](#). All references to file pathnames refer to locations within this standard source tree, not within the package you initially receive.

Note: Do not modify any files except the configuration file and PSP files.

Configuration File

The file `src/config/config_safe_nand_mt29fxg01_spi.h` contains the configurable parameters of the system. Configure these as required. This is the only file in the module that you should modify. For details of the options, see [Configuration Options](#).

System Files

These files are in the directory `src/safe-flash/nand/phy/micron`. **These files should only be modified by HCC.**

File	Description
<code>mt29fxg01_spi.c</code>	Driver source code.
<code>mt29fxg01_spi.h</code>	Driver header file.

Version File

The file `src/version/ver_safe_nand_mt29fxg01_spi.h` contains the version number of this module. This version number is checked by all modules that use this module to ensure system consistency over upgrades.

Platform Support Package (PSP) Files

These files in the directory `src/psp/target/safe_nand_mt29fxg01_spi` provide the [psp_safe_nand_mt29fxg01_spi_init\(\)](#) function that configures the hardware. Modify these files as required for your hardware.

Note:

- These are PSP implementations for the specific micro-controller and board; you may need to modify these to work with a different micro-controller and/or development board.
- In the package these files are offset to avoid overwriting an existing implementation. Copy them to the root `hcc` directory for use.

The files are as follows:

File	Description
psp_safe_nand_mt29fxg01_spi.c	Low level initialization function source.
psp_safe_nand_mt29fxg01_spi.h	Header file.

The PSP also has a version file, **ver_psp_safe_nand_mt29fxg01_spi.h**.

3. Configuration Options

Set the system configuration options in the file **src/config/config_safe_nand_mt29fxg01_spi.h**. This section lists the available options and their default values.

MT29FXG01_SPI_BAUDRATE

The SPI baud rate in Hz. The default is 50000000.

MT29FXG01_SPI_UNIT_ID

The unit ID of the SPI to be used to communicate with the flash chip. The default is 0.

4. PSP Porting

The Platform Support Package (PSP) is designed to hold all platform-specific functionality, either because it relies on specific features of a target system, or because this provides the most efficient or flexible solution for the developer.

The module makes use of the following standard PSP function:

Function	Package	Component	Description
psp_memcmp()	psp_base	psp_string	Compares two blocks of memory.

The module makes use of the following standard PSP macros:

Macro	Package	Element	Description
PSP_RD_BE32	psp_base	psp_endianness	Reads a 32 bit value stored as big-endian from a memory location.
PSP_WR_BE32	psp_base	psp_endianness	Writes a 32 bit value to be stored as big-endian to a memory location.

The module makes use of the following standard PSP SPI functions. These are described in the [HCC SPI Driver PSP User Guide](#).

Function	Package	Element	Description
psp_spi_init()	psp_base	psp_spi	Initializes the SPI port.
psp_spi_cs_hi()	psp_base	psp_spi	Sets chip select high.
psp_spi_cs_lo()	psp_base	psp_spi	Sets chip select low.
psp_spi_set_baudrate()	psp_base	psp_spi	Sets the baud rate.
psp_spi_rx()	psp_base	psp_spi	Receives a number of bytes.
psp_spi_tx()	psp_base	psp_spi	Transmits a number of bytes.
psp_spi_tx1()	psp_base	psp_spi	Transmits one byte.
psp_spi_lock()	psp_base	psp_spi	Locks the SPI for the specific unit. This can be useful if multiple units are attached to the same SPI bus.
psp_spi_unlock()	psp_base	psp_spi	Unlocks the SPI for the specific unit. This can be useful if multiple units are attached to the same SPI bus.

The files **psp_safe_nand_mt29fxg01_spi.c** and **psp_safe_nand_mt29fxg01_spi.h** define the [psp_safe_nand_mt29fxg01_spi_init\(\)](#) function that configures the hardware. Modify these files as required for your hardware.

4.1. psp_safe_nand_mt29fxg01_spi_init

Use this function to initialize the hardware. This configures pins and clocks.

Format

```
void psp_safe_nand_mt29fxg01_spi_init ( void )
```

Arguments

None.

Return Values

None.

5. Version

Version 2.00

For use with SafeFLASH NAND Driver for Micron® MT29FxG01 SPI versions 1.03 and above