SafeFLASH NAND Driver for Spansion S34ML01Gx00 User Guide

Version 1.00

For use with SafeFLASH NAND Driver for Spansion® S34ML01Gx00 versions 1.05 and above

Exported on 01/22/2019

Table of Contents

1 System Overview

This chapter contains the fundamental information for this module.

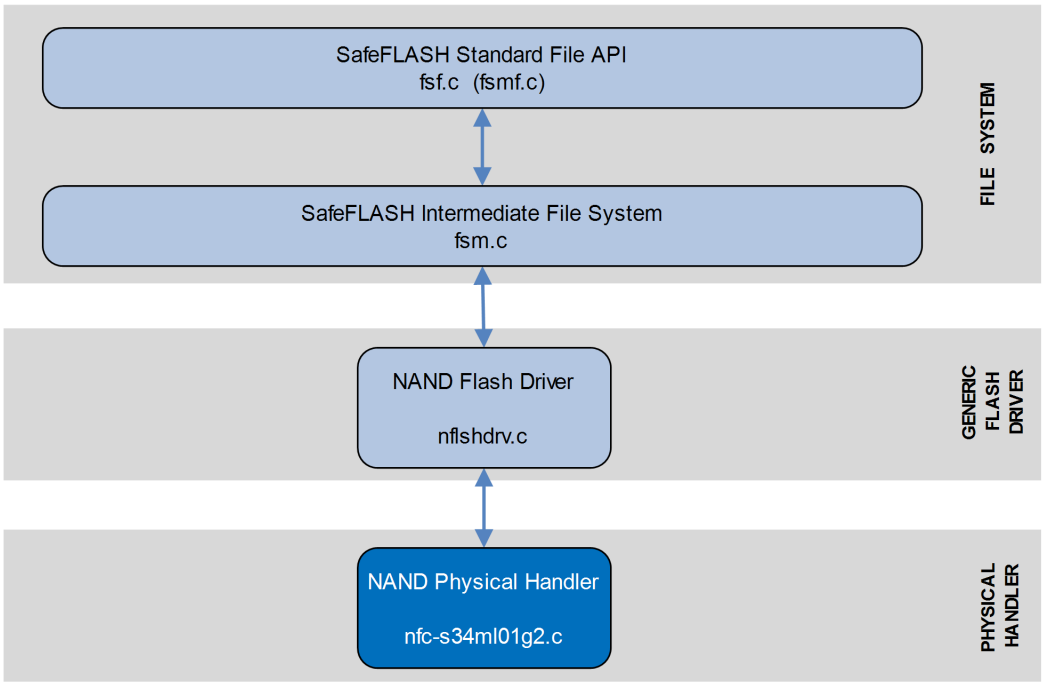
The component sections are as follows:

- 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.

1.1 Introduction

This guide is for those who want to implement an HCC Embedded SafeFLASH NAND driver for S34ML01G100 and S34ML01G200 devices from Spansion®. 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 Spansion® S34ML01G100 and S34ML01G200 flash and is easily configurable for similar NAND flash parts.
- Supports static and dynamic wear leveling.
- Provides bad block management.

1.3 Device Description

This table summarizes the properties of the two device types:

	S34ML01G100	S34ML01G200
Size (Gb)	1	2
Bytes per page	2112	2112
Pages per block	64	64
Blocks	1K	2K
Planes	1	2

Error-Correcting Code (ECC) Requirement

1 bit per 528 bytes. The sample driver includes a software ECC algorithm. This can be modified to use hardware ECC if this is provided by the host microcontroller.

1.4 Fail-safety

This driver for S34ML01G100 and S34ML01G200 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
hcc_base_doc	This contains the two guides that will help you get started.
fs_safe_nand	The SafeFLASH NAND flash driver.
fs_safe_nand_drv_nfc_s34ml01g2	The SafeFLASH NAND driver for Spansion® S34ML01Gx00 package described in this document.

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 Spansion S34ML01Gx00 driver.

HCC SafeFLASH NAND Driver for Spansion S34ML01Gx00 User Guide

This is this document.

1.6 Change History

To view or download manuals, see File System PDFs.

For the history of changes made to the package code itself, see History: fs_safe_nand_drv_nfc_s34ml01g2.

The current version of this manual is 1.00.

Manual version	Date	Software version	Reason for change
1.00	2019-01-22	1.05	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.

2.1 Configuration File

The file **src/config/config_safe_nand_s34ml01g2_nfc.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.

2.2 System Files

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

File	Description
nfc-s34ml01g2.c	Driver source code.
nfc-s34ml01g2.h	Driver header file.

2.3 Version File

The file **src/version/ver_safe_nand_s34ml01g2_nfc.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.

2.4 Platform Support Package (PSP) Files

These files in the directory **src/psp/target** provide the **psp_nfc_init()** function that configures the hardware. Modify these files as required for your hardware.

File	Description
include/hcc_mk70f_reg.h	Register values.
nand-nfc/psp_nand-nfc.c	Low level initialization function source.
nand-nfc/psp_nand-nfc.h	Header file.

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. See PSP Porting for details.
- In the package these files are offset to avoid overwriting an existing implementation. Copy them to the root **hcc** directory for use.

The PSP also has two version files:

File	Description
ver_psp_nand-nfc-k70f120.h	PSP version.
ver_psp_proc_reg.h	Register file version.

3 Configuration Options

Set the system configuration options in the file **src/config/config_safe_nand_s34ml01g2_nfc.h**. This section lists the available options and their default values.

SAFE_NAND_NFC_CS

The chip select for the flash device. This has two possible values:

- 0 - CS0 pin is used.
- 1 - CS1 pin is used. This is the default.

FS_SIGNATURE_OFFSET

The file system signature offset within a page. The default is 0x804.

ECC_ST_OFFSET

The NFC ECC offset within a page. The default is 0x808.

This must be multiple of 8; see the NFC_CFG_ECCAD value in the PSP's register file.

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 functions:

Function	Package	Element	Description
psp_memcmp()	psp_base	psp_string	Compares two blocks of memory.
psp_memcpy()	psp_base	psp_string	Copies a block of memory. The result is a binary copy of the data.

The files **psp_nand-nfc.c** and **psp_nand-nfc.h** define the psp_nfc_init() function that configures the hardware. Modify these files as required for your hardware.

4.1 psp_nfc_init

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

Format

```
extern void psp_nfc_init ( void )
```

Arguments

None.

Return Values

None.