

SafeFLASH NOR Driver for Macronix MX29GL128F User Guide

Version 1.20

For use with SafeFLASH NOR Driver for Macronix
MX29GL128F versions 1.02 and above

Date: 31-Aug-2017 12:55

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.

Table of Contents

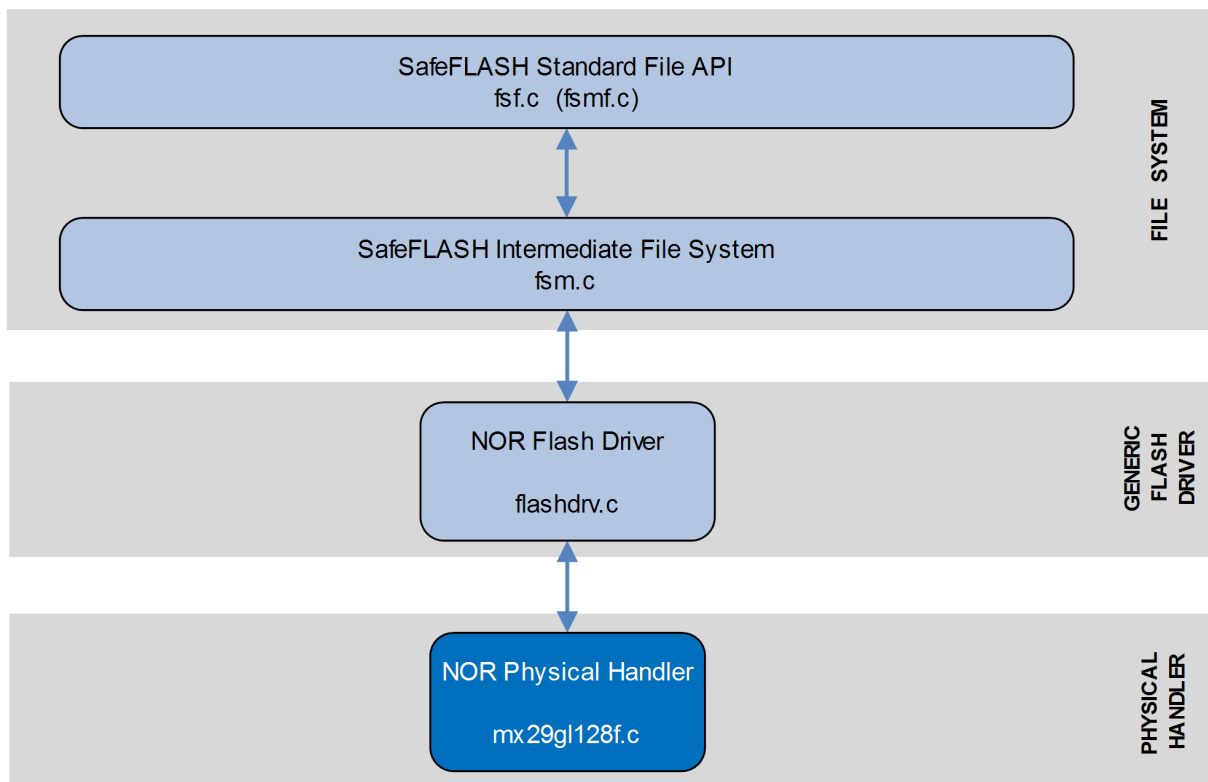
System Overview	3
Introduction	3
Feature Check	4
Device Description	5
Fail-safety	6
Packages and Documents	7
Packages	7
Documents	7
Change History	8
Source File List	9
Configuration File	9
System Files	9
Platform Support Package (PSP) Files	9
Version File	10
Configuration Options	11
PSP Porting	13
psp_mx29gl128f_hw_init	13

1 System Overview

1.1 Introduction

This guide is for those who want to implement a SafeFLASH NOR driver for Macronix MX29GL128F and similar devices from Macronix International Co. This is for use with HCC's SafeFLASH file system.

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



This diagram shows:

- The main SafeFLASH package – this provides the file API and intermediate file system. This is described in the [HCC SafeFLASH File System User Guide](#).
- The NOR flash driver – the generic device driver for NOR flash, provided by the base NOR package. This driver handles issues of FAT maintenance, wear leveling, and so on. It is described in the [HCC SafeFLASH File System NOR Drive User Guide](#).
- The NOR 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 Macronix MX29GL128F flash and is easily configurable for similar NOR flash parts.
- Supports static and dynamic wear leveling.
- Provides bad block management.
- A sample driver is available with a porting description.

1.3 Device Description

This driver is specifically tested and validated with the MX29GL128F device. In addition, note that:

- With minor modification, the driver should work with all MX29xyyyz devices: MX29GL512F, MX29GL512G, MX29GA512F, and MX29GL512E.

This table summarizes the properties of the devices:

Name	Description
MX29	Family prefix
GA,GL,VS	GA = Standard Read Access GL = Page Mode Read Access VS = Burst mode read access.
128/256/512	Density 128Mb/256Mb/512Mb.

	MX29GL128F
Density	128Mb
Bus Width	x8 or x16

The devices also have:

- 64KW/128KB uniform sector architecture - 128 equal sectors.
- 16-byte/8-word page read buffer.
- 64-byte/32-word write buffer.
- Extra 128-word sector for security.

1.4 Fail-safety

This driver for MX29G128F NOR 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 in practice, the target hardware should ensure that in the event of a falling voltage the system resets or signals when the level approaches the specified programming level of the flash chip and inhibits further flash access.

There are other ways to manage this, for instance by adding a capacitance to ensure power is still available to complete an operation after a hardware error or reset condition is detected.

By using these techniques, the system can 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_nor</code>	The SafeFLASH NOR flash driver.
<code>fs_safe_nor_drv_mx29gl128f</code>	The low level driver 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 NOR Drive User Guide

This document describes the SafeFLASH NOR generic driver.

HCC SafeFLASH NOR Driver for Macronix MX29GL128F 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 [Archive: SafeFLASH NOR Driver for Macronix MX29GL128F User Guide](#).
- For the history of changes made to the package code itself, see [History: fs_safe_nor_drv_mx29gl128f](#).

The current version of this manual is 1.20. The full list of versions is as follows:

Manual version	Date	Software version	Reason for change
1.20	2017-08-31	1.02	Corrected <i>Packages</i> list.
1.10	2017-06-26	1.02	New <i>Change History</i> format.
1.00	2017-04-24	1.02	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_nor_mx29gl128f.h` contains the configurable system parameters. 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/nor/phy/macronix`. **These files should only be modified by HCC.**

File	Description
<code>mx29gl128f.c</code>	Driver source code.
<code>mx29gl128f.h</code>	Driver header file.

2.3 Platform Support Package (PSP) Files

These files in the directory `src/psp/target/macronix` define the `psp_mx29gl128f_hw_init()` function that configures the 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.

File	Description
<code>psp_mx29gl128f_hw.c</code>	Low level initialization function.
<code>psp_mx29gl128f_hw.h</code>	Header file.

2.4 Version File

The file `src/version/ver_safe_nor_drv_mx29gl128f.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.

3 Configuration Options

Set the system configuration options in the file `src/config/config_safe_nor_mx29gl128f.h`

Note: The following four options are manufacturer and device IDs that identify this device. Do not change any of these values unless you port this driver to support another similar device.

NOR_MAN_ID

The identifier of the flash manufacturer. This is fixed at 0xC2 for this device, representing Macronix.

NOR_DEV_ID

The default is 0x227E.

NOR_DEV_ID2

The default is 0x2221.

NOR_DEV_ID3

The default is 0x2201.

NOR_BLOCKSTART

The first block used by the file system. All blocks before this are free to be used by other applications. The blocks are numbered from 0, which is the default.

NOR_SECTORSIZE

The logical sector size. Set this to a value less than or equal to the block size (`NOR_BLOCK_SIZE`). The default is 4096.

NOR_DESCSIZE

The descriptor size. Set this to a value less than or equal to the block size (`NOR_BLOCK_SIZE`). The default is (16 * 1024).

NOR_CACHEDESCSIZE

Set the cache size to a value less than the descriptor size (`NOR_DESCSIZE`). The default is (4 * 1024).

Note: Set the following parameters according to the flash used. For more details of the block start, block number and size parameters, see the [HCC SafeFLASH File System NOR User Guide](#).

NOR_BLOCK_SIZE

The block size. Set this to match the flash device used. The default is (128 * 1024).

NOR_NUM_BLOCKS

The number of blocks used by the file system. The default is 128.

NOR_BLOCKSTART + NOR_NUM_BLOCKS must be less than or equal to the number of physical blocks on the device (128). Blocks outside this range are free for use by other applications.

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 files `psp_mx29gl128f_hw.c` and `psp_mx29gl128f_hw.h` define the [`psp_mx29gl128f_hw_init\(\)`](#) function that configures the hardware. Modify these files as required for your hardware.

4.1 `psp_mx29gl128f_hw_init`

Use this function to initialize the device.

Format

```
void psp_mx29gl128f_hw_init ( void )
```

Arguments

None.

Return Values

None.