Freescale Semiconductor
Application Note

Document Number: AN4302
Rev. 0, 04/2011

Introduction to the S12G Family EEPROM

Including a Comparison between the S08DZ, S12XE, and S12P Families

by: Victor Hugo Osornio López
Guadalajara, Mexico

1 Introduction

The MC9S12G family is intended as a bridge between high-end 8-bit microcontrollers, such as the S08DZ family, and high-performance 16-bit microcontrollers, such as the MC9S12XS family. It uses features found on the MC9S12XS and MC9S12P families, including error correction code (ECC) on flash memory, a fast analog-to-digital converter (ADC), and a frequency modulated phase locked loop (IPLL) that improves the EMC performance.

This application note explains the differences in nonvolatile memory (NVM) storage, more specifically regarding EEPROM storage, between the four families, and introduces the user to the S12G family EEPROM.

A simple driver is included to provide the basic tools to manage a S12G family EEPROM.

Contents

freescale™ semiconductor

2 Device Comparison

The main feature of an EEPROM is its small sector size; an EEPROM sector is the smallest portion of the EEPROM memory that can be erased. For the S12G family, this portion is 4 bytes long.

Table 1. Differences between MCU families

	S12G	S12XE [1]	S12P	S08DZ
D-FLASH	No	Yes	Yes	No
EEPROM	Yes	No	No	Yes
Alternate Scheme	No	Hardware & software supported	Software supported	No
Sector Size (Bytes)	4	256	256	4/8 selectable
Maximum size (bytes)	4000	4000	4000	8000
Programming Scheme	FCCOB	FCCOB	FCCOB	FCMD
Program /erase cycles	500K typical, 100K min	500K typical, 50K min	500K typical, 50K min	100K typical, 50K min
Operating Frequency (MHz)	Min: 0.8 Typ: 1.0 Max: 1.05	Min: 0.8 Max: 1.05	Min: 0.8 Typ: 1.0 Max: 1.05	Min: .150 Max: .200
Word Programming time (us)	Typ: 90 Max: 100	Typ: 88 Max: 95	Typ: 100 Max: 107	Min: ~90 Max: ~120.06
Sector Erase time (us)	Typ: 5 Max: 26	Typ: 5.2 Max: 21	Typ: 5 Max: 26	NA
Memory Protection	Yes	Yes	Yes	Yes

[1] Not using emulated EEPROM support.

The S12G, S12XE, and S12P use the same indexed command execution scheme. They are also equal in terms of erase/program speeds, operating frequency, and memory protection scheme. The S12XE and S12P families do not have an EEPROM. They have a D-flash with a sector size of 256 bytes; that means that every time the user needs to erase a single byte, 255 extra bytes are erased; therefore, an additional memory management scheme is recommended. The S12G family erases 4 bytes instead of 256, decreasing the need for a memory management algorithm/scheme.

The S08DZ family does not share the indexed command scheme with the other compared families. It is also different in terms of writing speed, programming cycles, and operating frequency. As with the S12G family, S08DZ devices contain an EEPROM with a configurable sector size of 4/8 bytes.

In order to change EEPROM contents, the S12G, S12XE, and S12P use an indexed FCCOB register to provide a command code and its relevant parameters to the memory controller. The S08DZ requires an attempt to write data to a memory address. The address determines where the command is going to be

executed, and the data written is taken as a parameter for the command; after providing that data, the flash command register needs to be updated and then the command is launched.

Please refer to Section 4, "Migrating to the S12G," for details on how to use the FCCOB for EEPROM command execution.

3 S12G EEPROM Overview

3.1 EEPROM features

- S12G family EEPROM sizes range from 512 bytes to 4 Kbytes
- 4 bytes sector size
- Single-bit fault correction and double-bit fault detection within a word during read operations
- Automated program and erase algorithm with verify and generation of ECC parity bits
- Fast sector erase and word program operation
- Protection scheme to prevent accidental program or erase of EEPROM memory
- Ability to program up to four words in a burst sequence

3.2 Launching a flash command

The FCCOB is a six word array addressed via the CCOBIX index found in the FCCOBIX register.

To launch a flash command, first set up all required FCCOB fields and then initiate the command's execution by writing a 1 to the CCIF bit in the FSTAT register (a 1 written by the user clears the CCIF command completion flag to 0). When the CCIF bit in the FSTAT register has been cleared, all FCCOB parameter fields are locked and cannot be changed until the command completes (the completion is indicated by the memory controller returning CCIF to 1).

Table 2 shows the generic flash command format. For details on specific command format, refer to Section 3.2.2, "EEPROM commands."

Table 2. Generic command format table

CCOBIX	BYTE	PARAMETER	PARAMETER FIELD
0	HI	1	Flash Command ID
	LOW		Global Address [17:16]
1	HI	2	Global Address [15:8]
	LOW		Global Address [7:0]
2	HI	3	Data Word 0 [15:8]
	LOW		Data Word 0 [7:0]
3	HI	4	Data Word 1 [15:8]
	LOW		Data Word 1 [7:0]

Table 2. Generic command format table (continued)

CCOBIX	BYTE	PARAMETER	PARAMETER FIELD
4	HI	5	Data Word 2 [15:8]
	LOW		Data Word 2 [7:0]
5	HI	6	Data Word 3 [15:8]
	LOW		Data Word 3 [7:0]

3.2.1 Flash command description

All flash commands are launched using the following command write sequence:

1. Prior to launching any command, software must write the FCLKDIV register to divide the main bus clock (FBUS) to a frequency ranging from 0.8 to 1.05 MHz. Setting flash frequency below 0.8 MHz may overstress and damage the device. Setting it higher than 1.05 MHz may cause incomplete erasure or programming of flash memory cells. This step needs to be executed after a reset event; once executed, the FDIVLCK bit in FCLKDIV register will be set and flash frequency will be locked for writing until next reset.
2. Evaluate CCIF value status; if CCIF is 0, the flash module is still executing a command.
3. Clear the ACCERR and FPVIOL bits in FSTAT register. This will clear previous access or protection violation errors.
4. Set CCOBIX to 0. Write the FCCOB high byte with the command ID to execute.
5. If the command requires more parameters, write the low byte of FCCOB. (Global address [17:16], Parameter 1).
6. If the command requires more parameters, increase CCOBIX and write the next parameter to FCCOB; keep doing so until all parameters are passed to the memory controller. CCOBIX final value depends on the command that is being launched; refer to Section 3.2.2, “EEPROM commands,” to verify command-specific CCOBIX values.
7. Clear the CCIF bit to start the command execution; the CCIF bit will be set by the memory controller once the command execution ends.

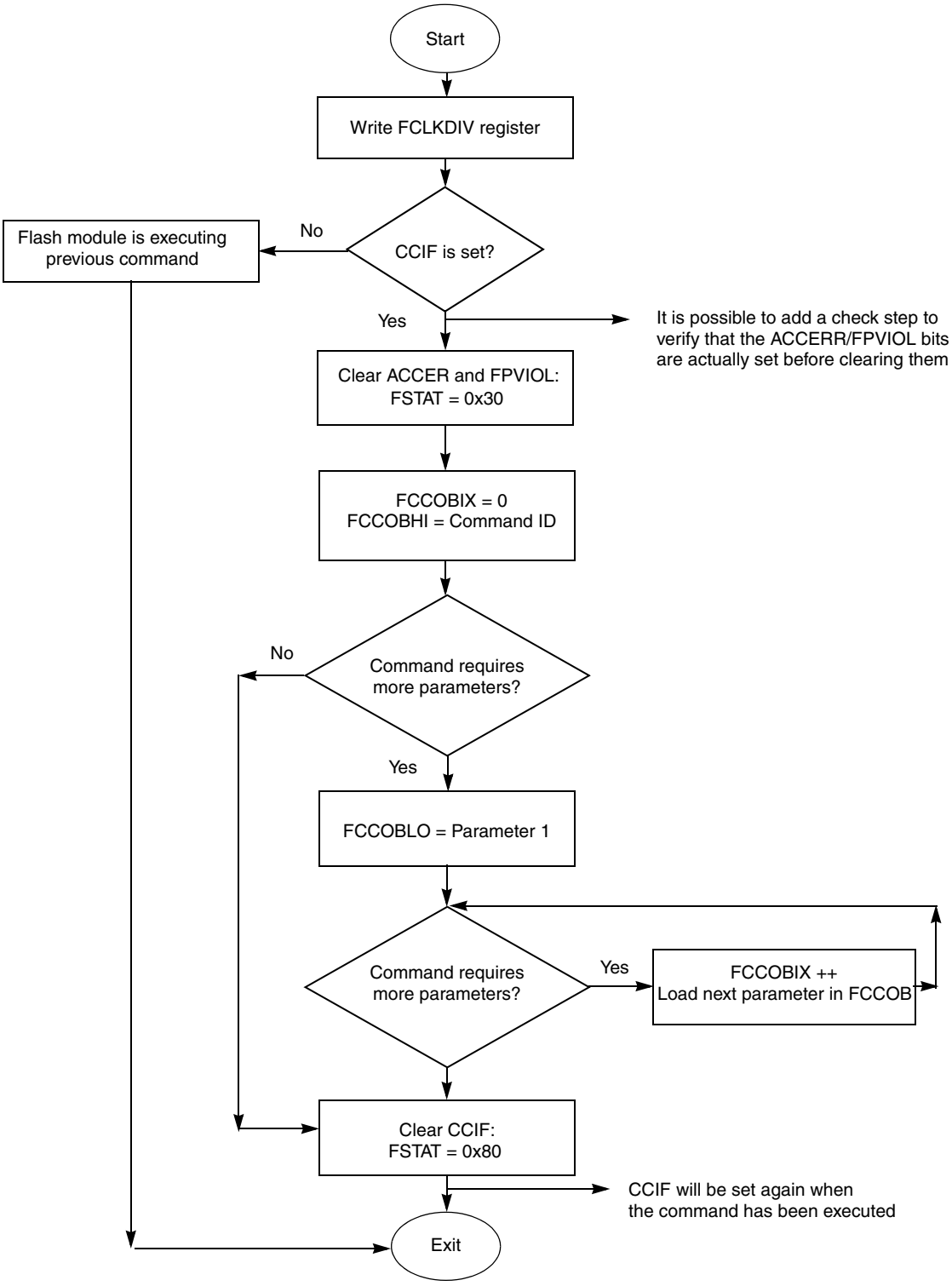

Figure 1. Launch command flowchart

3.2.2 EEPROM commands

3.2.2.1 Erase verify all blocks

- Verifies that all flash memory and EEPROM blocks have been erased.
- CCOBIX value must be 0 when executing the command.

Table 3. Erase verify all blocks command format

Erase Verify All Blocks				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Flash Command ID	0x01
	LOW		NOT REQUIRED	NOT REQUIRED

3.2.2.2 Erase verify block

- Verifies that an EEPROM/flash memory block has been erased.
- CCOBIX value must be 0 when executing the command.

Table 4. Erase verify block command format

Erase Verify Block				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Flash Command ID	0x02
	LOW		Flash Block selection code: (00 to verify EEPROM, 01, 10 and 11 to verify P-Flash)	0x0

3.2.2.3 Erase all blocks

- Erases EEPROM and P-flash memory spaces.
- CCOBIX value must be 0 when executing the command.

Table 5. Erase all blocks command format

Erase All Blocks				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Flash Command ID	0x08
	LOW		NOT REQUIRED	NOT REQUIRED

3.2.2.4 Erase flash block

- Erases all addresses within a specified block.
- CCOBIX value must be 1 when executing the command.

Table 6. Erase flash block command format

Erase Flash Block				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Flash Command ID	0x09
	LOW		Global Address [17:16] (EEPROM Global Address starts at 0x000400)	0x00
1		2	Global Address [15:0]	0x0400

3.2.2.5 Erase verify EEPROM section

- Verifies that a size-adjustable EEPROM section is erased. The verified size depends on Parameter 3 value.
- CCOBIX value must be 2 when executing the command.
- Example verifies that 2 words (4 bytes) starting from address 0x000400 are in erased state.

Table 7. Erase verify EEPROM section command format

Erase Verify EEPROM Section				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Command ID	0x10
	LOW		Global Address[17:16]: EEPROM global address starts at 0x00_0400	0x00
1	HI	2	Global address [15:0] of first word to be verified.	0x400
	LOW			
2	HI	3	Number of words to be verified	0x02
	LOW			

3.2.2.6 Program EEPROM command

- Programs between one and four previously erased words in the EEPROM block. CCOBIX value determines how many words will be programmed.
- CCOBBIX value must be greater than 1 and less than 6 when the command is executed.
- Example stores 0x1111 at 0x400–0x401, 0x2222 at 0x402–0x403, 0x3333 at 0x404–0x405, and 0x4444 at 0x406–0x407.

Table 8. Program EEPROM command format

Program EEPROM Command				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Command ID	0x11
	LOW		Global Address [17:16] of word to be programmed	0x00
1	HI	2	Global address[15:0] of word to be programmed	0x400
	LOW			
2	HI	3	Word 0 value (mandatory)	0x1111
	LOW			
3	HI	4	Word 1 value (Optional)	0x2222
	LOW			
4	HI	5	Word 2 value (Optional)	0x3333
	LOW			
5	HI	6	Word 3 value (Optional)	0x4444
	LOW			

3.2.2.7 Erase EEPROM sector

- Erases 4 bytes within EEPROM block.
- CCOBBIX value must be 1 when command is executed.
- Example erases address range 0x408–0x409–0x40A–0x40B.

Table 9. Erase EEPROM sector format

Erase EEPROM Sector				
CCOBIX	BYTE	PARAMETER	PARAMETER FIELD	VALUE
0	HI	1	Command ID	0x12
	LOW		Global Address [17:16] of sector to be erased	0x00
1	HI	2	Global Address [15:0] of sector to be erased.	0x408
	LOW			

4 Migrating to the S12G

S12XE (D-flash) and S12P use the same memory controller as the S12G; flash registers, command sets, and command schemes are similar among these devices. Migrating from a D-flash to an EEPROM is seamless in terms of low level development.

The S08DZ uses a different memory controller, with different registers and command scheme. Despite this fact, both the S12G and S08DZ families have similar functions. Table 10 and Table 11 show available commands and registers for the S08DZ and their corresponding match for the S12G.

Table 10. Command comparison

S08DZ Command	S12G Match Command
Blank Check	EraseVerifyBlock EraseVerifySector
Byte Program	Program EEPROM
Burst Program	Program EEPROM
Sector Erase	Erase Sector
Mass Erase	Erase All Blocks
Sector Erase Abort	NA

Other than the Sector Erase Abort command, each S08DZ flash command and register has an S12G counterpart that matches or enhances its behavior, thus making it possible to migrate from one device to the other without modifying high level application code.

Table 11. Register comparison

S08DZ Register	S12G Match Register	Register Function
FCDIV	FCLKDIV	Configures flash clock
FOPT	FSEC	Contains KEY enable, EEPROM sector Size, security status
FPROT	EEPROT/DPROT	Defines protected state and protected area size
FSTAT	FSTAT	Memory controller state, access and violation errors, command completion flag, blank check

5 EEPROM Driver

A simple EEPROM driver was developed to show FCCOB and CCOBIX usage to execute flash commands and manage EEPROM content.

The EEPROM driver consists of three main files:

Devices.h contains:

- Device specific information; Bus clock.
- S12G_DEVICE macro will select currently used device and its respective EEPROM size.

Flash.c contains:

- **LaunchFlashCommand** function used by every link function to interface with the Memory controller. Its execution flowchart corresponds to the command sequence suggested by Figure 1.
- **FlashConfigFLClock** function used to configure flash clock to run at 1 MHz.
- **Link Functions:**
  - FlshProgramEEPROM: function used to program 1 to 4 words in a specified address.
  - FlashEraseVerifyEEPROM: function used to verify if EEPROM block is erased.
  - FlashEraseEEPROMSector: function used to erase 4 bytes starting from a specified address.
  - FlashEraseVerifyEEPROMSection: Function used to verify a given number of words starting from a specified address.
  - FlashEraseEEPROMBlock: Function used to erase EEPROM block, this command cannot be executed while reading data from P-flash, LaunchFlashCommand function has to be transferred to RAM before the command is executed.

Flash.h contains macros for command IDs, required parameter numbers for each command and error masks to be able to isolate errors reported on FSTAT register.

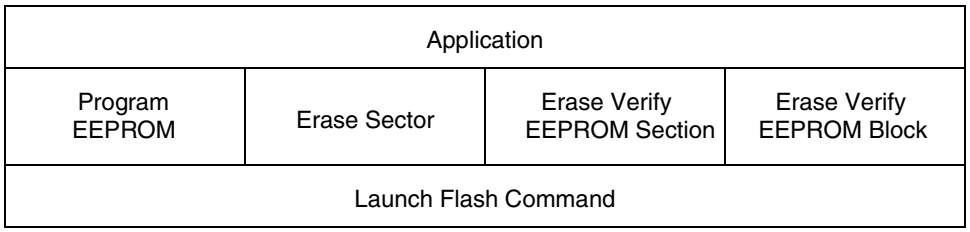

Figure 2. Driver Overview

THIS PAGE IS INTENTIONALLY BLANK

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Number: AN4302
Rev. 0
04/2011

freescale™
semiconductor