SmartFusion2 and IGLOO2

Embedded Nonvolatile Memory (eNVM) Simulation

Microsemi

Table of Contents

Introduction

The SmartFusion2 MSS has an on-chip embedded non-volatile memory (eNVM). You can access the eNVM using the eNVM controller. This controller is a slave of the MSS AHB Switch Matrix and can receive commands from a master located either inside the MSS (i.e., Cortex M3) or a master located in the fabric via the Fabric Interface Controllers (FIC_0 or FIC_1).

This document describes the steps required to simulate eNVM operation.

1 – NVM Configuration

You can configure the eNVM using the MSS eNVM configurator. Using the eNVM configurator, you can:

- Add Serialization and Data Storage clients
- Supply data files that will be used to initialize the eNVM block when you program your device

For details, and to learn about the eNVM configurator options, refer to the SmartFusion2 MSS eNVM Configuration Guide.

2 – eNVM Organization

This chapter describes the internal organization of the eNVM, and how it is accessed by Masters in the MSS and the FPGA fabric.

The eNVM is a nonvolatile (flash) memory that is divided into pages. Each page of the eNVM contains 128 bytes (accessible as 32 words).

The total capacity of the eNVM varies with the SmartFusion2 or IGLOO2 device you are using. Table 2-1 provides a list of devices and eNVM capacities.

***Table 2-1* • eNVM Capacity by Device**

Device	eNVM capacity (bytes)
SmartFusion2	
M2S005	1 x 128KB
M2S010, M2S025, M2S050, M2S060	1 x 256KB
M2S090, M2S150	2 x 256KB
IGLOO2	
M2GL005	1 x 128KB
M2GL010, M2GL025, M2GL050, M2GL060	1 x 256KB
M2GL090, M2GL150	2 x 256KB

Note: On the larger (M2S/M2GL090/150) devices, the eNVM is composed of two blocks (eNVM_0 and eNVM_1), which are accessed separately

eNVM Internal Organization

Each eNVM is divided into pages. One page is a 128-byte section of the eNVM. The eNVM is word-addressable. Table 2-2 and Table 2-3 list the ranges of the eNVM pages for different devices.

***Table 2-2* • SmartFusion2 eNVM Page Ranges**

Device	Capacity	Total Pages (User + Reserved)	Total Reserved Pages	User Page Range (Available to User)	Reserved Page Range (Unavailable to User)
M2S005	128KB	1024	16	0-1007	1008-1023
M2S010, M2S025, M2S050	256KB	2048	16	0-2031	2032-2047
M2S050T_ES	256KB	2048	33	0-2014	2015-2047

***Table 2-2* • SmartFusion2 eNVM Page Ranges (continued)**

Device	Capacity	Total Pages (User + Reserved)	Total Reserved Pages	User Page Range (Available to User)	Reserved Page Range (Unavailable to User)
M2S060	256KB	2048	64	0-1983	1984-2047
M2S090, M2S150	512KB	4096	64	0-4031	4032-4095

***Table 2-3* • IGLOO2 eNVM Page Ranges**

Device	Capacity	Total Pages (User + Reserved)	Total Reserved Pages	User Page Range (Available to User)	Reserved Page Range (Unavailable to User)
M2GL005	128KB	1024	48	0-975	976-1023
M2GL010, M2GL025, M2GL050	256KB	2048	48	0-1999	2000-2047
M2GL060	256KB	2048	96	0-1951	1952-2047
M2GL090, M2GL150	512KB	4096	96	0-3999	4000-4095

Note: For IGLOO2, Reserved Pages are used by the HPMS to store Certificate/Digest and Peripheral configuration data for SERDES, FDDR and MDDR. For SmartFusion2, Reserved Pages are used by the MSS to store Certificate/Digest only. Reserved pages are for internal use only and not available to the user.

eNVM Access

The SmartFusion2 eNVM is part of the MSS. It is accessed via the eNVM Controller, which is a slave of the MSS AHB Switch Matrix (Figure 2-1). Masters of the AHB Switch Matrix (MSS Cortex-M3), a Fabric Master (via the FIC32_0/1 interfaces) can read from and write to the eNVM.

- All eNVM accesses are performed using AHB read and write transactions
- Irrespective of the master initiating the access, the procedure to read and write the eNVM remains the same

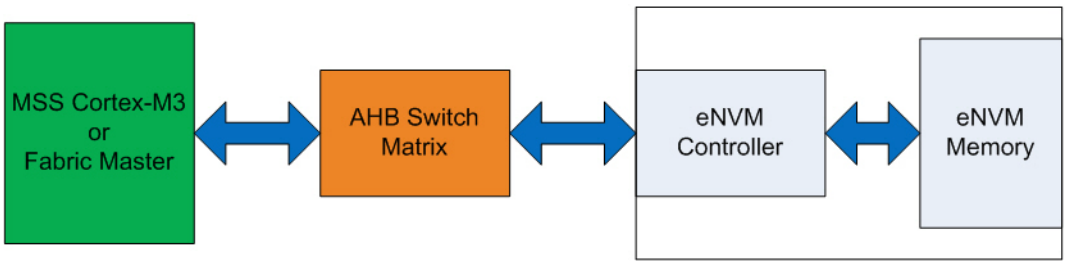

***Figure 2-1* • eNVM Access**

3 – eNVM Simulation

The eNVM simulation model fully models the commands and bus transactions required to access the eNVM on silicon.

To access the eNVM, you must initiate AMBA transactions using either the Cortex-M3 Master or a Fabric Master (Using the FIC Slave Interface).

Before accessing the eNVM, Microsemi recommends that you poll bit #0 of the eNVM status register (address: 0x60080120). If this bit is 0, the eNVM is busy. Wait until this bit becomes 1 to access the eNVM.

Writing to the eNVM

You can simulate writing to eNVM from the following bus masters:

- Cortex-M3 (SmartFusion2 only)
- Fabric AHB Master (via FIC_0 or FIC_1)
- Fabric APB Master (via FIC_0 or FIC_1)

Writes to the eNVM are buffered. You must first write your data into the write data buffer (WDB) and then use a single command to commit (program) your data into one page of the eNVM.

The sequence of transactions required to program the eNVM is:

1. Request exclusive access to the eNVM control register set. This is necessary to ensure that no other Master can write to the eNVM at the same time, and is done by writing 0x1 to the REQACCESS register (address: 0x600801FC)

   The Master that is requesting exclusive access must then check that the request has been granted by reading back from the REQACCESS register.

   – On read back, check bit #2 (counting up from 0). If it is 1, the request was successful.

   – If bit #2 is 0, the request for exclusive access was denied, and the eNVM cannot be written at this time.
2. Write your data into the WDB; the WDB is a byte-addressable 1024-bit buffer. Its base address is: 0x60080080 for eNVM_0 and at 0x600C0080 for eNVM_1.
3. Compute values of bits that will be written into the eNVM Command Register:

   – Bits 31-24 should be 0x80 (Hex) to specify the ProgramADS command code.

   – Bits [17:7] corresponds to the eNVM page address to be written

   – Bits [23:18] and [6:0] are not relevant for the ProgramADS command and can be written 0x0 (Hex)

For details about what values to use, refer to Table 4-7 in the SmartFusion2 Microcontroller Subsystem User's Guide.

4. Write eNVM Command Register (address: 0x60080148) with the data computed in Step 3 above

   Note that the eNVM will not respond to further commands until the write is complete

   Note that on silicon, writing a page of the eNVM may take up to 8ms, but in simulation, this step completes in a few clock cycles
5. Release exclusive access to the eNVM control register set by writing 0x0 to the REQACCESS register.

The following is an example of a Cortex-M3 BFM script configured to write a block of data to eNVM0. Assume that you want to write two 32-bit words 0xaaaaaaaa and 0xbbbbbbbb into page 25 of the eNVM.

```
#1. Wait for bit 0 of status register to become 1
pollbit w 0x60080120 0x0 0 1
#2. Request exclusive access to the eNVM control register set
write w 0x600801fc 0x0 0x1
```

```
#2b. Readcheck to see if access has been granted
readcheck w 0x600801fc 0x0 0x5 (for MSS master)
#The simulation will fail if access has not been granted
#3. Write data to the WDB
write w 0x60080080 0x0 0xaaaaaaaa
write w 0x60080080 0x4 0xbbbbbbbb
#4. Compute the value of the command register: Bits[31-19]: '0000 1000 0000 0'
#Bits[18-7]: '000 0000 1100 1' (25 in decimal)
#Bits[6-0]: '000 0000'
#Complete string: 0x08000c80
#5. Write the command register
write w 0x60080148 0x0 0x08000c80
#6. Release exclusive access to the eNVM
write w 0x600801fc 0x0 0x0
```

Refer to the SmartFusion2 FPGA Microcontroller Subsystem BFM Simulation Guide for general guidelines on BFM simulations for SmartFusion2 designs.

Reading from the eNVM

The eNVM can be read as a byte-addressable random access memory. The address range for reads is given in Table 3-1.

***Table 3-1* • eNVM Read Address Ranges**

	ENVM0	**ENVM1**
Base Address	0x60000000	0x60040000
Max read address (005)	0x6002FFFF	N/A
Max read address (010,025,050, 060)	0x6003FFFF	N/A
Max read address (090,150)	0x6003FFFF	0x6007FFFF

The eNVM is accessible directly, similar to a random access memory. The address ranges of the eNVMs are given in Table 3-1. To read any location in the eNVM, first compute the offset address as follows:

Offset Address = (Page #) * 0x80 + (Address of Word in Page)

The Base Address will be either 0x60000000 or 0x60040000, depending on whether you are accessing eNVM_0 or eNVM_1.

The following is an example that demonstrates how to read data from eNVM_0. In the example above, two 32-bit words "0xaaaaaaaa" and "0xbbbbbbbb" were written into addresses 0x0 and 0x4 of Page# 25 of eNVM_0. The example below shows an attempt to read the same two words back.

```
#1. Wait for bit 0 of status register to become 1
pollbit w 0x60080120 0x0 0 1
#2  Read first word
#2a Base Address = 0x60000000
#2b Word in Page = 0x0 (first word). Page Number = 25.
#   Offset Address = 0x80 * 25 + 0x0 = 0xc80
#2c Read and compare word to what was written in Fig. 2
readcheck w 0x60000000 0xc80 0xaaaaaaaa

#3  Read second word
#3a Base Address = 0x60000000
#3b Word in Page = 0x4 (second word). Page Number = 25.
#   Offset Address = 0x80 * 25 + 0x4 = 0xc84
#3c Read and compare word to what was written in Fig. 2
readcheck w 0x60000000 0xc84 0xbbbbbbbb
```

Erasing the eNVM

You can also erase the contents of the eNVM, one page at a time using the following steps:

1. Request exclusive access to the eNVM control register set. This is necessary to ensure that no other Master can write to the eNVM at the same time. This is done by writing 0x1 to the REQACCESS register (address: 0x600801FC). The Master that is requesting exclusive access must then check that the request has been granted by reading back from the REQACCESS register.
   - On read back, check bit #2 (counting up from 0). If it is 1, the request was successful.
   - If bit #2 is 0, the request for exclusive access was denied, and the eNVM cannot be written at this time.
2. Compute values of bits that will be written into the eNVM Command Register:
   - Bits 31-20: "0x020"
   - Bit 19: 0
   - Bits 18-7 corresponds to the number of the page to be written
   - Bits 6-0: “0x0"
3. Write eNVM Command Register (address: 0x60080148) with the data computed in the previous step.

   The eNVM will not respond to further commands until the erase is complete.

   Note that on silicon, erasing a page of the eNVM may take up to 8ms, but in simulation, this step completes in a few clock cycles.
4. Release exclusive access to the eNVM control register set by writing 0x0 to the REQACCESS register.

The example below shows a sequence of instructions to erase page #25 of eNVM_0.

```
#1. Wait for bit 0 of status register to become '1'
pollbit w 0x60080120 0x0 0 1
#2. Request exclusive access to the eNVM control register set
write w 0x600801fc 0x0 0x1
#2c. Readcheck to see if access has been granted
readcheck w 0x600801fc 0x0 0x5 (for MSS master)
#The simulation will fail if access has not been granted
#3. Compute the value of the command register: Bits[31-19]: '0000 0010 0000 0'
#Bits[18-7]: '000 0000 1100 1' (25 in decimal)
#Bits[6-0]: '000 0000'
#Complete string: 0x02000c80
#4. Write the command register
write w 0x60080148 0x0 0x02000c80
#5. Release exclusive access to the eNVM
write w 0x600801fc 0x0 0x0
```

4 – Product Support

Microsemi SoC Products Group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, electronic mail, and worldwide sales offices. This appendix contains information about contacting Microsemi SoC Products Group and using these support services.

Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From North America, call 800.262.1060
From the rest of the world, call 650.318.4460
Fax, from anywhere in the world, 408.643.6913

Customer Technical Support Center

Microsemi SoC Products Group staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions about Microsemi SoC Products. The Customer Technical Support Center spends a great deal of time creating application notes, answers to common design cycle questions, documentation of known issues, and various FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

Technical Support

Visit the Customer Support website (www.microsemi.com/soc/support/search/default.aspx) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the website.

Website

You can browse a variety of technical and non-technical information on the SoC home page, at www.microsemi.com/soc.

Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center. The Technical Support Center can be contacted by email or through the Microsemi SoC Products Group website.

Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is soc_tech@microsemi.com.

My Cases

Microsemi SoC Products Group customers may submit and track technical cases online by going to My Cases.

Outside the U.S.

Customers needing assistance outside the US time zones can either contact technical support via email (soc_tech@microsemi.com) or contact a local sales office. Sales office listings can be found at www.microsemi.com/soc/company/contact/default.aspx.

ITAR Technical Support

For technical support on RH and RT FPGAs that are regulated by International Traffic in Arms Regulations (ITAR), contact us via soc_tech_itar@microsemi.com. Alternatively, within My Cases, select **Yes** in the ITAR drop-down list. For a complete list of ITAR-regulated Microsemi FPGAs, visit the ITAR web page.

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA

Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996

E-mail: sales.support@microsemi.com

Microsemi Corporation (MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense & security, aerospace, and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs, and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif. and has approximately 3,600 employees globally. Learn more at **www.microsemi.com**.