TMS320C5515/14/05/04 DSP
External Memory Interface (EMIF)

User's Guide

Literature Number: SPRUGU6
March 2010

Contents

List of Figures

List of Tables

Read This First

About This Manual

This document describes the operation of the external memory interface (EMIF). The purpose of the EMIF is to provide a means to connect to a variety of external asynchronous devices including NOR Flash, NAND Flash, and SRAM.

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure designate a bit that is used for future device expansion.

Related Documentation From Texas Instruments

The following documents describe the TMS320C5515/14/05/04 Digital Signal Processor (DSP) Digital Signal Processor (DSP). Copies of these documents are available on the internet at http://www.ti.com.

SWPU073 — TMS320C55x 3.0 CPU Reference Guide. This manual describes the architecture, registers, and operation of the fixed-point TMS320C55x digital signal processor (DSP) CPU.

SPRU652 — TMS320C55x DSP CPU Programmer's Reference Supplement. This document describes functional exceptions to the CPU behavior.

SPRUFO1A — TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) Inter-Integrated Circuit (I2C) Peripheral User's Guide. This document describes the inter-integrated circuit (I2C) peripheral in the TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) devices. The I2C peripheral provides an interface between the device and other devices compliant with Phillips Semiconductors Inter-IC bus (I2C-bus) specification version 2.1 and connected by way of an I2C-bus. This document assumes the reader is familiar with the I2C-bus specification.

SPRUFO2 — TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) Timer/Watchdog Timer User's Guide. This document provides an overview of the three 32-bit timers in the TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) devices. The 32-bit timers of the device are software programmable timers that can be configured as general-purpose (GP) timers. Timer 2 can be configured as a GP, a Watchdog (WD), or both simultaneously.

SPRUFO3 — TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) Serial Peripheral Interface (SPI) User's Guide. This document describes the serial peripheral interface (SPI) in the TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) devices. The SPI is a high-speed synchronous serial input/output port that allows a serial bit stream of programmed length (1 to 32 bits) to be shifted into and out of the device at a programmed bit-transfer rate. The SPI supports multi-chip operation of up to four SPI slave devices. The SPI can operate as a master device only.

SPRUFO4 — TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) General-Purpose Input/Output (GPIO) User's Guide. This document describes the general-purpose input/output (GPIO) on the TMS320C5515/14/05/04/VC05/VC04 digital signal processor (DSP) devices. The GPIO peripheral provides dedicated general-purpose pins that can be configured as either inputs or outputs. When configured as an input, you can detect the state of an internal register. When configured as an output you can write to an internal register to control the state driven on the output pin.

SPRUFO5 — TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) Universal Asynchronous Receiver/Transmitter (UART) User's Guide. This document describes the universal asynchronous receiver/transmitter (UART) peripheral in the TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) devices. The UART performs serial-to-parallel conversions on data received from a peripheral device and parallel-to-serial conversion on data received from the CPU.

SPRUFP1 — TMS320C5515/05/VC05 Digital Signal Processor (DSP) Successive Approximation (SAR) Analog to Digital Converter (ADC) User's Guide. This document provides an overview of the Successive Approximation (SAR) Analog to Digital Converter (ADC) on the TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) devices. The SAR is a 10-bit ADC using a switched capacitor architecture which converts an analog input signal to a digital value.

SPRUFP3 — TMS320C5515/05/VC05 Digital Signal Processor (DSP) Liquid Crystal Display Controller (LCDC) User's Guide. This document describes the liquid crystal display controller (LCDC) in the TMS320C5515/14/05/04/VC05/VC04 Digital Signal Processor (DSP) devices. The LCD controller includes a LCD Interface Display Driver (LIDD) controller.

SPRUGU6— TMS320C5515/14/05/04 DSP External Memory Interface (EMIF) User's Guide. This document describes the operation of the external memory interface (EMIF) in the TMS320C5515/14/05/04 Digital Signal Processor (DSP) devices. The purpose of the EMIF is to provide a means to connect to a variety of external devices.

External Memory Interface (EMIF)

This document describes the operation of the External Memory Interface (EMIF) in the Digital Signal Processor (DSP).

Topic **Page**

1.1 Introduction

1.1.1 Purpose of the External Memory Interface

The purpose of the EMIF is to provide a means to connect to a variety of external asynchronous devices including:

- NOR Flash, NAND Flash, and SRAM
- Mobile single data rate (SDR) SDRAM and SDRAM devices

NOTE: Non-mobile SDRAM can be supported under certain circumstances. The C5504/05/14/15 always use mobile SDRAM initialization but are able to support SDRAM memories that ignore the BA0 and BA1 pins for the *load mode register* command. During the mobile SDRAM initialization, the device issues the *load mode register* initialization command to two different addresses that differ in only the BA0 and BA1 address bits. These registers are the Extended Mode register and the Mode register. The extended mode register exists only in mSDRAM and not in non-mSDRAM. If a non-mobile SDRAM memory ignores bits BA0 and BA1, the second loaded register value overwrites the first, leaving the desired value in the mode register and the non-mobile SDRAM will work with the device.

Section 1.3 contains examples of connecting the EMIF to mobile SDRAM devices.

1.1.2 Features

The EMIF has the following features:

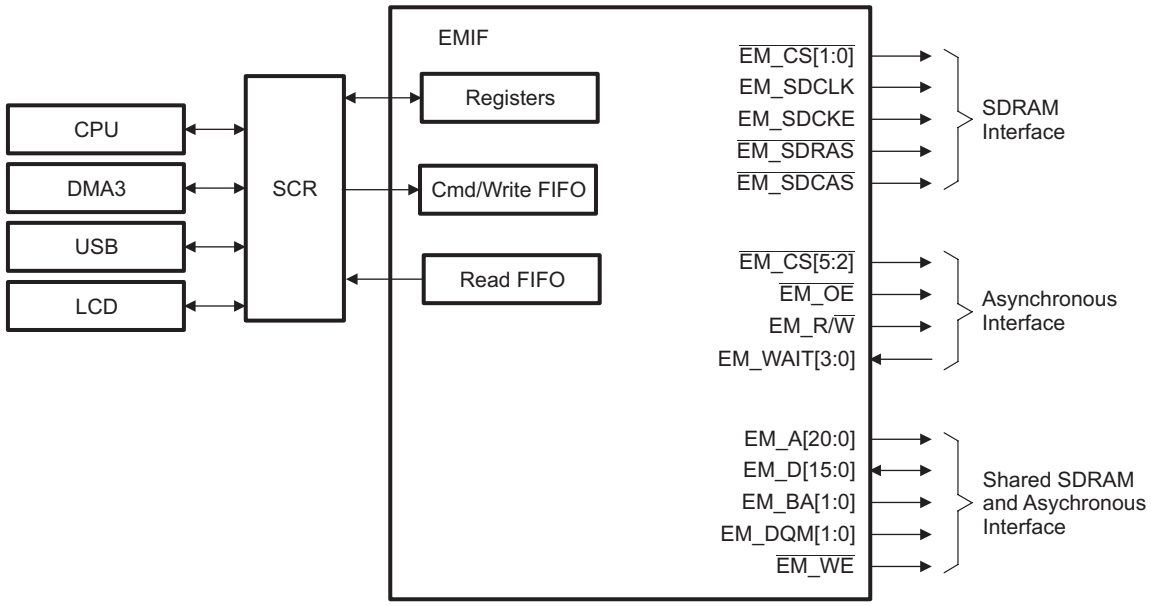
- Supports asynchronous devices (e.g., RAM, ROM, NOR Flash).
  - Up to 8 MB asynchronous address range over 4 chip selects.
  - Supports 8 and 16-bit data bus widths.
  - Programmable cycle timings for each chip select.
  - Page mode for NOR Flash.
  - Supports extended wait cycles.
  - Supports select strobe mode.
- Supports NAND Flash on 4 asynchronous chip selects.
  - Supports 8 and 16-bit data bus widths.
  - Programmable cycle timings for each chip select.
  - Supports 1-bit ECC for 8 and 16-bit NAND Flash.
  - Supports 4-bit ECC for 8-bit and 16-bit NAND Flash.
  - Does not perform error correction.

- Supports mobile SDR SDRAM devices.
  - 8MB SDRAM address range over two chip select spaces.
  - 16-bit data bus width.
  - Supports CAS latencies of 2 and 3 (4 not supported).
  - Supports 1, 2, and 4 internal banks.
  - Supports 256, 512, 1024, 2048-word page sizes.
  - Supports 4 and 8 burst lengths (1 and 2 not supported).
  - Supports sequential burst type (interleave burst type not supported).
  - SDRAM auto initialization from reset or configuration change.
  - Bank interleaving across both the chip selects.
  - Self refresh and pre-charge power down modes for low power.
  - Partial array self-refresh and temperature controlled self refresh modes for low power in mobile SDR.
    - Temperature controlled self-refresh is only supported for mobile SDR having on-chip temperature sensor.
  - Supports prioritized refresh.
  - Programmable SDRAM refresh rate and backlog counter.
  - Programmable SDRAM timing parameters.
  - Auto pre-charge not supported for better bank interleaving performance.
  - Clock Suspend not supported for SDR SDRAM.

1.1.3 Functional Block Diagram

Figure 1-1 illustrates a high-level view of the EMIF and its connections within the device. The CPU and DMA controller access the EMIF through a switched central resource. Section 1.2.3 describes the EMIF external pins and summarizes their purpose when interfacing with external devices.

Figure 1-1. EMIF Functional Block Diagram



1.2 Architecture

This section provides details about the architecture and operation of the EMIF. Both SDRAM and asynchronous interface are covered, along with other system-related topics such as clock control and pin multiplexing.

1.2.1 Clock Control

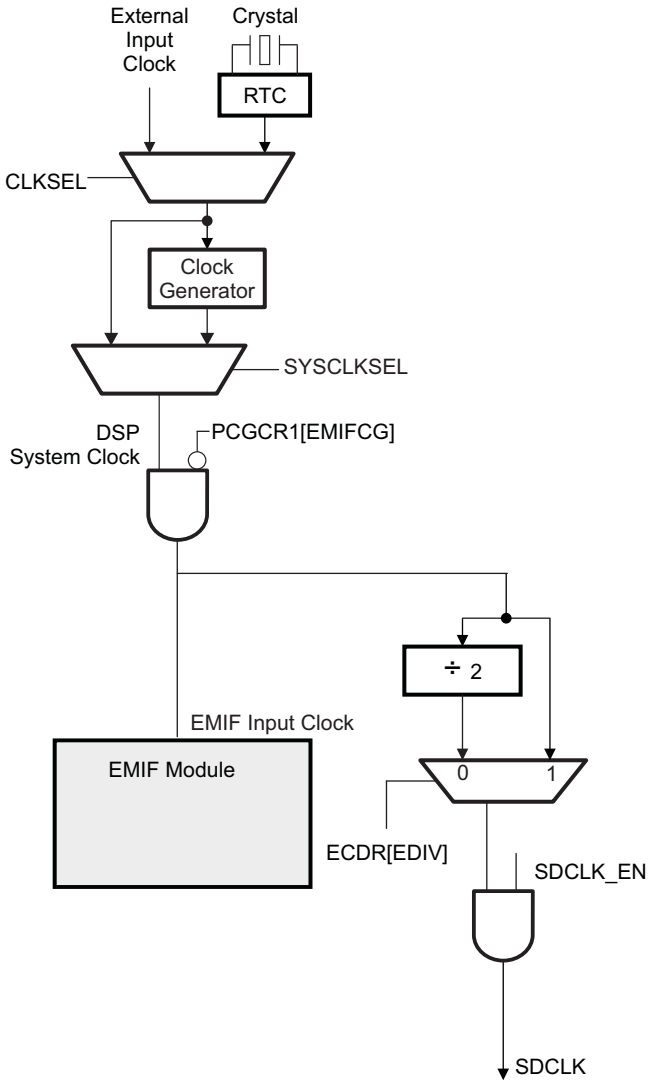
As shown in Figure 1-2, the clock generator receives either the real-time clock (RTC) or a signal from an external clock source and produces the DSP system clock. This clock is used by the DSP CPU and peripherals. The EMIF input clock is used to source the interface clock in synchronous mode and to generate the access cycles in asynchronous mode. The SDCLK can be generated from the EMIF CLK or a divide by two version of the EMIF CLK through EMIF Clock Divide Register (ECDR) register.

The C5515/14/05/04 has limitations to the clock frequency on the EM_SDCLK pin (driven by SDCLK) based on the CV_{DD} and DV_{DDEMIF}.

- When CV_{DD} = 1.3 V, and DV_{DDEMIF} = 3.3 V, 2.75 V, or 2.5 V, the maximum clock frequency on the EM_SDCLK pin is limited to 100 MHz (EM_SDCLK ≤ 100 MHz). Therefore, if SYSCLK ≤ 100 MHz, the EM_SDCLK can be configured either as SYSCLK or SYSCLK/2, but if SYSCLK > 100 MHz, the EM_SDCLK must be configured as SYSCLK/2.
- When CV_{DD} = 1.05 V, and DV_{DDEMIF} = 3.3 V, 2.75 V, or 2.5 V, the maximum clock frequency on the EM_SDCLK pin is limited to 60 MHz (EM_SDCLK ≤ 60 MHz). Therefore, if SYSCLK ≤ 60 MHz, the EM_SDCLK can be configured as either SYSCLK or SYSCLK/2, but if SYSCLK > 60 MHz, the EM_SDCLK must be configured as SYSCLK/2.
- When DV_{DDEMIF} = 1.8 V, regardless of the CV_{DD} voltage, the clock frequency on the EM_SDCLK pin must be configured as SYSCLK/2.

The device includes logic which can be used to gate the clock to its on-chip peripherals, including the EMIF. The input clock to the EMIF can be enabled and disabled through the peripheral clock gating configuration register 1 (PCGCR1).

Figure 1-2. Clocking Diagram for the EMIF



1.2.2 EMIF Requests

Different modules within the DSP can make requests to the EMIF. These requests consist of accesses to SDRAM memory, asynchronous memory, and EMIF registers. Because the EMIF can process only one request at a time, a switch central resource (SCR) exists within the DSP to provide prioritized requests from the different sources to the EMIF.

1. CPU (peripheral register access)
2. CPU (data access)
3. CPU (instruction fetch)
4. DMA Controller 3
5. USB
6. LCD

If a request is submitted to the EMIF from two or more sources simultaneously, the SCR will arbitrate between the difference sources using a round-robin approach. Upon completion of a request, the SCR again evaluates the pending requests and forwards the next pending request to the EMIF.

When the EMIF receives a request, it may or may not be immediately processed. In some cases, the EMIF will perform one or more auto refresh cycles before processing the request. For details on the internal arbitration of the EMIF between performing requests and performing auto refresh cycles, see Section 1.2.10.

CAUTION

The EMIF does not support constant addressing mode. All request serviced by the EMIF must use a linear (incrementing) addressing mode.

1.2.3 Memory Map

External memory is divided into several chip select spaces. On the device CPU, DMA controller 3 (DMA3) and USB and LCD can access external memory. The starting address for each of the chip select spaces in external memory is different from the point-of-view of these modules. The memory map as seen by these modules are shown in Table 1-1.

Table 1-1. EMIF Memory Map

DSP Memory Map	Size (Bytes)	CPU Start *Word* Address	DMA3 Start *Byte* Address	USB Start *Byte* Address	LCD Start *Byte* Address
EMIF CS0	8M(1)	02 8000h	0100 0000h	0100 0000h	0100 0000h
EMIF CS2	4M	40 0000h	0200 0000h	0200 0000h	0200 0000h
EMIF CS3	2M	60 0000h	0300 0000h	0300 0000h	0300 0000h
EMIF CS4	1M	70 0000h	0400 0000h	0400 0000h	0400 0000h
EMIF CS5 (2)	1M	78 0000h	0500 0000h	0500 0000h	0500 0000h

(1) This is an approximate value.
(2) When MP/MC = 0 the upper 128K bytes of EMIF CS5 is used for ROM data.

1.2.4 Signal Descriptions

Table 1-2 describes the EMIF pins that are used to interface to external devices.

Table 1-2. EMIF Pins Used to Access Both SDRAM and Asynchronous Devices

Pin(s)	Type	Description
EM_A[20:0]	Output	EMIF address bus. When interfacing to an SDRAM device, these pins are primarily used to provide the row and column address to the SDRAM. When interfacing to an asynchronous device, these pins are used in conjunction with the EM_BA[1:0] pins to form the address that is sent to the device.
EM_D[15:0]	Input/Output	EMIF data bus.
EM_BA[1:0]	Output	When interfacing to an SDRAM device, these pins are used to provide the bank address inputs to the SDRAM. When interfacing to an asynchronous device, these pins are used in conjunction with the EM_A[20:0] pins to form the address that is sent to the device.
$\overline{\text{EM\_DQM[1:0]}}$	Output	Active-low byte enables. When interfacing to SDRAM, these pins are connected to the $\overline{\text{DQM}}$ pins of the SDRAM to individually enable/disable each of the bytes in a data access. When interfacing to an asynchronous device, these pins are connected to byte enables.
$\overline{\text{EM\_WE}}$	Output	Active-low write enable. When interfacing to SDRAM, this pin is connected to the WE pin of the SDRAM and is used to send commands to the device. When interfacing to an asynchronous device, this pin provides a signal that is active-low during the strobe period of an asynchronous write access cycle.

Table 1-3. EMIF Pins Specific to SDRAM

Pin(s)	Type	Description
$\overline{\text{EM\_CS[1:0]}}$	Output	Active-low chip select pin for SDRAM devices. This pin is connected to the chip-select pin of the attached SDRAM device and is used for enabling/disabling commands. By default, the EMIF keeps this SDRAM chip select active, even if the EMIF is not interfaced with an SDRAM device. This pin is deactivated when accessing the asynchronous memory bank and is reactivated on completion of the asynchronous assess.
$\overline{\text{EM\_SDRAS}}$	Output	Active-low row address strobe pin. This pin is connected to the RAS pin of the attached SDRAM device and is used for sending commands to the device.
$\overline{\text{EM\_SDCAS}}$	Output	Active-low column address strobe pin. This pin is connected to the CAS pin of the attached SDRAM device and is used for sending commands to the device.
EM_SDCKE	Output	Clock enable pin. This pin is connected to the CKE pin of the attached SDRAM device and is used for issuing the SELF REFRESH command which places the device in self refresh mode.
EM_SDCLK	Output	SDRAM clock pin. This pin is connected to the CLK pin of the attached SDRAM device.

Table 1-4. EMIF Pins Specific to Asynchronous Devices

Pin(s)	Type	Description
$\overline{\text{EM\_CS[5:2]}}$	Output	Active-low chip select pins for asynchronous devices. These pins are meant to be connected to the chip-select pins of the attached asynchronous device. These pins are active only during accesses to the asynchronous memory.
$\overline{\text{EM\_OE}}$	Output	Active-low pin enable for asynchronous devices. This pin provides a signal which is active-low during the strobe period of an asynchronous read access cycle.
EM_R/$\overline{\text{W}}$	Output	Active-low read/write select pin. This pin is high for the duration of an asynchronous read access cycle and low for the duration of an asynchronous write cycle.
EM_WAIT[3:0]	Input	Wait input with programmable polarity / NAND Flash ready input. An asynchronous device can extend the strobe period of an access cycle by asserting the wait input pins. When connected to NAND flash devices these pins function as NAND Flash ready inputs.

1.2.5 Pin Multiplexing

The EMIF address pins EM_A[20:15] are multiplexed with GPIO pins GPIO[26:21]. The external bus selection register (EBSR) controls the functionality of these pins. For more details on this register, see the device-specific data manual.

1.2.6 SDRAM Controller and Interface

The EMIF can gluelessly interface to mobile SDR SDRAM devices and supports such features as self refresh mode and prioritized refresh. In addition, it provides flexibility through programmable parameters such as the refresh rate, CAS latency, and many SDRAM timing parameters. The following sections include details on how to interface and properly configure the EMIF to perform read and write operations to externally connected SDR SDRAM devices. The SDCLK_EN bit must be set in the Clock Configuration register 1 (CCR1) 0x1C1E to turn on the SDRAM clock.

Non-mobile SDRAM can be supported under certain circumstances. The C5504/05/14/15 always use mobile SDRAM initialization but are able to support SDRAM memories that ignore the BA0 and BA1 pins for the *load mode register* command. During the mobile SDRAM initialization, the device issues the *load mode register* initialization command to two different addresses that differ in only the BA0 and BA1 address bits. These registers are the Extended Mode register and the Mode register. The extended mode register exists only in mSDRAM and not in non-mSDRAM. If a non-mobile SDRAM memory ignores bits BA0 and BA1, the second loaded register value overwrites the first, leaving the desired value in the mode register and the non-mobile SDRAM will work with device.

1.2.6.1 SDRAM Commands

The EMIF supports the SDRAM commands described in Table 1-5. Table 1-6 shows the truth table for the SDRAM commands, and an example timing waveform of the PRE command is shown in Figure 1-3. The address pin EM_A[10] is pulled low in this example to deactivate only the bank specified by the EM_BA[1:0] pins.

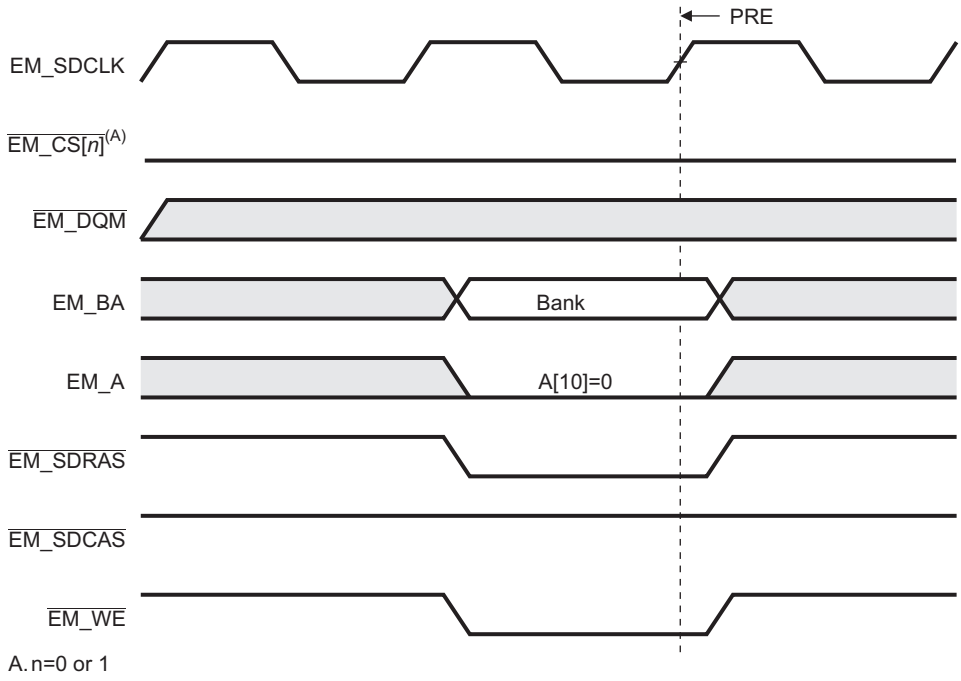
Table 1-5. EMIF SDRAM Commands

Command	Description
PRE	Pre-charge command. Depending on the value of EM_A[10], the PRE command either deactivates the open row in all banks (EM_A[10] = 1) or only the bank specified by the EM_BA[1:0] pins (EM_A[10] = 0).
ACTV	Activate command. The ACTV command activates the selected row in a particular bank for the current access.
READ	Read command. The READ command outputs the starting column address and signals the SDRAM to begin the burst read operation. Address EM_A[10] is always pulled low to avoid auto pre-charge. This allows for better bank interleaving performance.
WRT	Write command. The WRT command outputs the starting column address and signals the SDRAM to begin the burst write operation. Address EM_A[10] is always pulled low to avoid auto pre-charge. This allows for better bank interleaving performance.
BT	Burst-terminate command. The BT command is used to truncate the current read or write burst request.
LMR	Load mode register command. The LMR command sets the mode register of the attached SDRAM devices and is only issued during the SDRAM initialization sequence described in Section 1.2.6.4.
REFR	Auto-refresh command. The REFR command signals the SDRAM to perform an auto refresh according to its internal address.
SLFR	Self-refresh command. The self refresh command places the SDRAM into self-refresh mode, during which it provides its own clock signal and auto refresh cycles.
NOP	No operation command. The NOP command is issued during all cycles in which one of the above commands is not issued.
POWER DOWN	Power-down command. The POWER DOWN command signals the SDRAM to deactivate its input and output buffers, excluding CKE. The EMIF closes (pre-charge) all internal banks of the SDRAM prior to issuing the POWER DOWN command, therefore, the EMIF only supports pre-charge power down.

Table 1-6. Truth Table for SDRAM Commands

SDRAM Pins:	CKE	$\overline{CS}$	$\overline{RAS}$	$\overline{CAS}$	$\overline{WE}$	BA[1:0]	A[12:11]	A[10]	A[9:0]
EMIF Pins:	EM_SDCKE	$\overline{EM\_CS[n]}$	$\overline{EM\_SDRAS}$	$\overline{EM\_SDCAS}$	$\overline{EM\_WE}$	EM_BA[1:0]	EM_A[12:11]	EM_A[10]	EM_A[9:0]
PRE	H	L	L	H	L	Bank/X	X	L/H	X
ACTV	H	L	L	H	H	Bank	Row	Row	Row
READ	H	L	H	L	H	Bank	Column	L	Column
WRT	H	L	H	L	L	Bank	Column	L	Column
BT	H	L	H	H	L	X	X	X	X
LMR	H	L	L	L	L	X	Mode	Mode	Mode
REFR	H	L	L	L	H	X	X	X	X
SLFR	L	L	L	L	H	X	X	X	X
NOP	H	L	H	H	H	X	X	X	X

Figure 1-3. Timing Waveform for SDRAM PRE Command



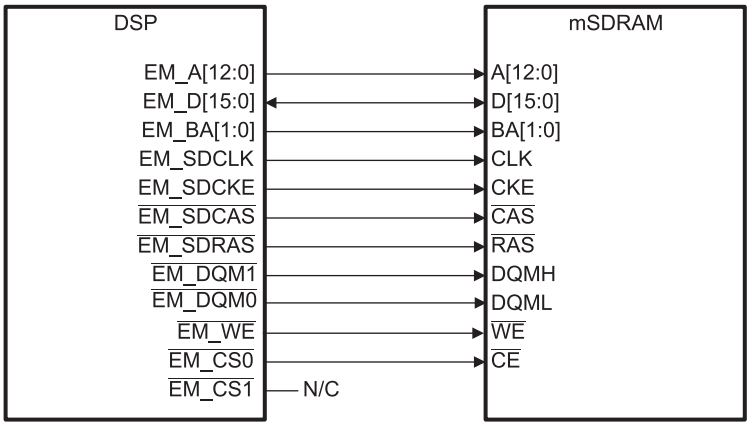
1.2.6.2 Interfacing to Mobile SDRAM

The EMIF supports a glueless interface to mobile SDRAM devices with the following characteristics:

- Pre-charge bit is A[10]
- The number of column address bits is 8, 9,10, or 11
- The number of row address bits is less than or equal to 14
- The number of internal banks is 1, 2, or 4

Figure 1-4 shows an interface between the EMIF and a single mobile SDRAM device. The mobile SDRAM device can be connected to either $\overline{EM\_CS0}$ or $\overline{EM\_CS1}$, however the proper external memory address must be used to activate the correct chip select pin. For more details, see Section 1.2.6.11. Note that when operated in SDRAM mode, the EMIF bus is always 16 bits wide.

Figure 1-4. EMIF to mSDRAM Connection



1.2.6.3 SDRAM Configuration Registers

The operation of the EMIF's SDRAM interface is controlled by programming the appropriate configuration registers. This section describes the purpose and function of each configuration register, but Section 1.4 should be referenced for a more detailed description of each register, including the default registers values and bit-field positions. The following tables list the main SDRAM configuration registers, along with a description of each of their programmable fields.

NOTE: Writing to SDCR1 or the lower byte of SDCR2 causes the EMIF to abandon whatever it is currently doing and trigger the SDRAM initialization procedure described in Section 1.2.6.4 .

Table 1-7. Description of Bit Fields in SDRAM Configuration Registers (SDCR1 and SDCR2)

Parameter	Description
SR	This bit controls entering and exiting of the Self-Refresh mode. The field should be written using a byte-write to the upper byte of SDCR2 to avoid triggering the SDRAM initialization sequence.
PD	This bit controls entering and exiting of the Power down mode. The field should be written using a byte-write to the upper byte of SDCR2 to avoid triggering the SDRAM initialization sequence. If both SR and PD bits are set, the EMIF will go into Self Refresh. See Section 1.2.6.4
PDWR	Perform refreshes during Power Down. Writing a 1 to this bit will cause the EMIF to exit the power down state and issue an AUTO REFRESH command every time Refresh May level is set.
CL	**CAS latency.** This field defines the number of clock cycles between when an SDRAM issues a READ command and when the first piece of data appears on the bus. The value in this field is sent to the attached SDRAM device via the LOAD MODE REGISTER command during the SDRAM initialization procedure as described in Section 1.2.6.4. Only values of 2h (CAS latency = 2) and 3h (CAS latency = 3) are supported and should be written to this field. A 1 must be simultaneously written to the BIT11_9LOCK bit field of SDCR1 in order to write to the CL bit field.
IBANK	**Number of Internal SDRAM Banks.** This field defines the number of banks inside the attached SDRAM devices in the following way: • When IBANK = 0, 1 internal bank is used • When IBANK = 1h, 2 internal banks are used • When IBANK = 2h, 4 internal banks are used This field value affects the mapping of logical addresses to SDRAM row, column, and bank addresses.
PAGESIZE	**Page Size.** This field defines the internal page size of the attached SDRAM devices in the following way: • When PAGESIZE = 0, 256-word pages are used • When PAGESIZE = 1h, 512-word pages are used • When PAGESIZE = 2h, 1024-word pages are used • When PAGESIZE = 3h, 2048-word pages are used This field value affects the mapping of logical addresses to SDRAM row, column, and bank addresses.

Table 1-8. Description of Bit Fields in SDRAM Refresh Control Register (SDRCR)

Parameter	Description
REFRATE	Refresh Rate. This field controls the rate at which attached SDRAM devices will be refreshed. The equation below can be used to determine the required value of REFRATE for an SDRAM device: REFRATE <= f_{EM_SDCLK} / (Required SDRAM Refresh Rate). More information about the operation of the SDRAM refresh controller can be found in Section 1.2.6.6.

Table 1-9. Description of Bit Fields in SDRAM Timing Registers (SDTIMR1 and SDTIMR2)

Parameter	Description
T_RFC T_RP T_RCD T_WR T_RAS T_RC T_RRD	SDRAM Timing Parameters. These fields configure the EMIF to comply with the AC timing constraints and to more efficiently schedule its operations. More details about each of these parameters can be found in the register description in section 4. These parameters should be set to satisfy the corresponding timing requirements found in the SDRAM's datasheet.

Table 1-10. Description of Bit Fields in SDRAM Self Refresh Exit Timing Register (SDSRETR)

Parameter	Description
T_XS	Self Refresh Exit Parameter. The T_XS field of this register informs the EMIF about the minimum number of EM_SDCLK cycles required between exiting Self Refresh and issuing any command. This parameter should be set to satisfy the t_{XSR} value for the attached SDRAM device.

1.2.6.4 Mobile SDRAM Auto-Initialization Sequence

The EMIF automatically performs an mobile SDRAM initialization sequence, regardless of whether it is interfaced to an SDRAM device. No memory accesses to the SDRAM and asynchronous interfaces are performed until this auto-initialization is complete. The auto-initialization sequence is performed by the EMIF when either of the following two events occur:

- The EMIF is taken out of reset. See Section 1.2.12 for more information on EMIF reset.
- A write is performed to SDRAM configuration register 1 (SDCR1) or the least significant byte of SDRAM configuration register 2 (SDCR2). In either of these cases the EMIF starts from step 2 below, without waiting for the eight SDRAM refresh intervals.

An SDRAM initialization sequence consists of the following steps:

1. The EMIF drives EM_SDCKE high and begins continuously issuing NOP commands.
2. After eight SDRAM refresh intervals, the EMIF issues a PRE command with EM_A[10] held high to indicate all banks. The SDRAM refresh interval is as defined by the REFRATE field in the SDRAM refresh control register (SDRCR).
3. After eight REFR commands, the EMIF issues an LMR command to the extended mode register with the settings as described in Table 1-11. (This step is needed for mobile SDRAM; because the chip does not have a pin to select if mobile SDRAM or SDRAM is used, this command is always sent during initialization. Therefore, SDRAM support is limited to memories that can handle this extra initialization command.)
4. The EMIF issues another LMR command, this time to the mode register with the settings as described in Table 1-12.
5. Finally, the EMIF performs an auto refresh cycle (see Section 1.2.6.6).

Table 1-11. Extended Mode Register Settings During Auto-Init Sequence

Mode Register Bit	Mode Register Field	Init Value	Description
15-7	Reserved	00h	Reserved
6-5	Drive Strength	SDRAM_DRIVE	Sets the drive strength of the memory data pins. The setting for these bits is taken from the SDRAM_DRIVE bits of the SDCR2 register.

Table 1-11. Extended Mode Register Settings During Auto-Init Sequence (continued)

Mode Register Bit	Mode Register Field	Init Value	Description
4-3	Internal Temperature Compensated Self-Refresh	0h	Specifies the maximum case temperature for self-refresh interval adjustment.
2-0	Partial Array Self-Refresh	PASR	Specifies the amount of memory that will be refreshed during a SLFR command. The setting for these bits is taken from the PASR bits of the SDCR2 register.

Table 1-12. Mode Register Settings During Auto-Init Sequence

Mode Register Bit	Mode Register Field	Init Value	Description
15:10	Reserved	00h	Reserved.
9	Write Burst Mode	0h	Specifies the write burst mode. This bit is set to 0 to indicate the burst type setting specified by bits 2-0 applies to both reads and writes.
8-7	Standard Operating Mode	0h	Selects standard operating mode.
6-4	CAS Latency	CL	Specifies the CAS latency. The value for these bits is taken from the CL bits of the SDCR1 register.
3	Burst Type	0h	Specifies the burst type. This bit is set to 0 to indicate sequential bursts.
2-0	Burst Length	3h	Specifies the burst length. These bits are set to 3 to indicate a burst length of 8.

1.2.6.5 SDRAM Configuration Procedure

The EMIF automatically performs the mSDRAM initialization sequence described in Section 1.2.6.4. However, you must follow the procedure listed below before performing any SDRAM requests.

The steps in the initialization procedure are as follows:

1. Turn SD clock on by writing 0x0001 to **Clock Configuration Register 1 (CCR1) [0x1C1Eh]**.
2. Program SDTIMR1, SDTIMR2, and SDSRETR to satisfy the timing requirements for the attached SDRAM device. The timing parameters should be taken from the SDRAM datasheet.
3. Program the REFRATE field of SDRCR such that the following equation is satisfied: (REFRATE x 8) / (f_{EM_SDCLK}) > 100 microseconds or 200 microseconds, depending on the power-up constraint specified by the SDRAM memory. The SDRAM power-up constraint specifies that 200 microseconds (sometimes 100 microseconds) should exist between receiving stable VDD and CLK and the issuing of a PRE command. For example, an EM_SDCLK frequency of 100 MHz would require setting REFRATE to 1251 (4E3h) or higher to meet a 100 microseconds constraint.
4. Program SDCR1 and SDCR2 to match the characteristics of the attached SDRAM device. This will cause the auto-initialization sequence in Section 1.2.6.4 to be re-run with the new value of REFRATE.
5. Perform a read from the SDRAM to guarantee that step 3 will occur after the initialization process has completed. Alternatively, wait for 200 microseconds.
6. Finally, program the REFRATE field to match that of the attached device's refresh interval. See Section 1.2.6.6.2 for details on determining the appropriate value.

After following the above procedure, the EMIF is ready to perform accesses to the attached SDRAM device. See Section 1.3 for an example of configuring the SDRAM interface.

NOTE: You must follow the steps in Section 1.2.13 before starting the steps outlined in this section.

1.2.6.6 SDRAM Refresh Controller

An SDRAM device requires that each of its rows be refreshed at a minimum required rate. The EMIF can meet this constraint by performing auto refresh cycles at or above this required rate. An auto refresh cycle consists of issuing a PRE command to all banks of the SDRAM device followed by issuing a REFR command. To inform the EMIF of the required rate for performing auto refresh cycles, the REFRATE field of the SDRAM Refresh Control Register (SDRCR) must be programmed. The EMIF will use this value along with two internal counters to automatically perform auto refresh cycles at the required rate. The auto refresh cycles cannot be disabled, even if the EMIF is not interfaced with an SDRAM. The remainder of this section details the EMIF's refresh scheme and provides an example for determining the appropriate value to place in the REFRATE field of SDRCR.

1.2.6.6.1 EMIF Refresh Scheme

The EMIF uses two counters to schedule REFR command: a 13-bit decrementing refresh interval counter and a 4-bit refresh backlog counter. The interval counter is loaded with the REFRATE field value at reset. The interval counter decrements by one each cycle until it reaches zero at which point it reloads from REFRATE and restarts decrementing. The counter also reloads and restarts decrementing whenever the REFRATE field is updated.

The refresh backlog counter records the number of REFR commands the EMIF currently has outstanding. The backlog counter increments by one each time the interval counter reloads (unless it has reached its maximum value of 15). The backlog counter decrements by one each time the EMIF issues a REFR command (unless it is already at zero).

The EMIF uses the refresh backlog counter to determine the urgency with which an auto refresh cycle should be performed. The four levels of urgency are described in Table 1-13. This refresh scheme allows the required refreshes to be performed with minimal impact on memory access requests.

Table 1-13. Refresh Urgency Levels

Urgency Level	Refresh Backlog Counter Range	Action Taken
Refresh May	1-3	An auto-refresh cycle is performed only if the EMIF has no requests pending and none of the SDRAM banks are open.
Refresh Release	4-7	An auto-refresh cycle is performed if the EMIF has no requests pending, regardless of whether any SDRAM banks are open.
Refresh Need	8-11	An auto-refresh cycle is performed at the completion of the current access unless there are read requests pending.
Refresh Must	12-15	Multiple auto-refresh cycles are performed at the completion of the current access until the Refresh Release urgency level is reached. At that point, the EMIF can begin servicing any new read or write requests.

The refresh counters do not operate when SDRAM has been put into self-refresh mode.

The EMIF issues REFR commands within auto refresh cycles. An auto refresh cycle consists of issuing an REFR command and waiting T_RFC (see the SDRAM timing register 2) cycles before re-checking the refresh levels. If the Refresh Release level is not reached, the EMIF starts another auto refresh cycle, otherwise it returns to the idle state.

1.2.6.6.2 Determining the Appropriate Value for the REFRATE Field

The value that should be programmed into the refresh-rate (REFRATE) field of SDRCR can be calculated by using the frequency of the EM_SDCLK signal (f_{EM_SDCLK}) and the required refresh rate of the SDRAM ($f_{Refresh}$).

The following formula can be used to calculate REFRATE:

REFRATE $\leq f_{EM\_SDCLK} / f_{Refresh}$

The SDRAM datasheet often communicates the required SDRAM Refresh Rate in terms of the number of REFR commands required in a given time interval. The required SDRAM Refresh Rate in the formula above can be therefore be calculated by dividing the number of required cycles per time interval (n_{cycles}) by the time interval given in the datasheet ($t_{Refresh\ Period}$):

$f_{Refresh} = n_{cycles} / t_{Refresh\ Period}$

Combining these formulas, the value that should be programmed into the REFRATE field can be computed as:

$REFRATE \leq f_{EM\_SDCLK} \times t_{Refresh\ Period} / n_{cycles}$

The following example illustrates calculating the value of REFRATE. Given that:

- f_{EM_SDCLK} = 100 MHz
- $t_{Refresh\ Period}$ = 64 ms (required refresh interval of the SDRAM)
- n_{cycles} = 8192 (number of cycles in a refresh interval for the SDRAM)

REFRATE can be calculated as:

$REFRATE \leq 100\ MHz \times 64\ ms/8192$

$REFRATE \leq 781.25$

REFRATE = 782 cycles = 30Eh cycles

1.2.6.7 Self Refresh Mode

To reduce the power consumption of the SDRAM device, you can request the EMIF to issue a SLRF command by setting the SR bit of SDCR2 to 1. When this bit is set, the EMIF will continue normal operation until all outstanding memory access requests have been serviced and the SDRAM refresh backlog has been cleared.

NOTE: The SR bit should be set and cleared using a byte-write to the upper byte of the SDCR2 to avoid triggering the SDRAM initialization sequence. The BYTEMODE bits in the EMIF system control register can be used to limit writes to EMIF registers to the upper or lower byte. See Section 1.2.8 for more details.

While in the self-refresh state, the EMIF continues to service asynchronous bank requests and register accesses as normal with one caveat: the EMIF will not park the data bus following a read to asynchronous memory while in the self-refresh state; instead, the EMIF tri-states the data bus. Therefore, it is not recommended to perform asynchronous read operations while the EMIF is in the self-refresh state in order to prevent floating inputs on the data bus. More information about data bus parking can be found in Section 1.2.9.

The EMIF will come out of the self-refresh state if the SR bit is cleared or an SDRAM access is requested while in the self-refresh state and T_RAS + 1 cycles have elapsed since the SLFR command was issued. The value of T_RAS is taken from SDRAM timing register 2. The EMIF exits from the self-refresh state by driving EM_SDCKE high, waiting T_XS + 1 cycles, and performing an auto refresh cycle. The value for T_XS is taken from the SDRAM self refresh exit timing register.

While in self-refresh mode, the SDRAM device consumes a minimal amount of power while performing its own refresh cycles. The SDRAM device should also be placed into self-refresh mode when changing the frequency of EM_SDCLK. If the frequency of EM_SDCLK changes while the SDRAM is not in self-refresh mode, the memory must be re-initialized following the procedure described in Section 1.2.6.5.

To use partial array self-refresh for mobile SDRAM, the PASR bits in the SDCR2 register must be appropriately programmed. The EMIF performs bank interleaving when IBANK_POS = 0 in SDCR2. Since the SDRAM is partially refreshed during partial array self refresh, it is recommended that IBANK_POS be set to 1 for software ease. If IBANK_POS is set to 0, it is the responsibility of software to move critical data into the banks that are going to be refreshed during partial array self-refresh.

1.2.6.8 Power Down Mode

To support low power modes, the EMIF can be requested to issue a POWER DOWN command to the SDRAM by setting the PD bit in SDCR2. When this bit is set, the EMIF will continue normal operation until all outstanding memory access requests have been serviced and the SDRAM refresh backlog has been cleared. At this point the EMIF will enter the power down state.

Upon entering this state, the EMIF will issue a POWER DOWN command (same as a NOP command but driving EM_SDCKE low on the same cycle). The EMIF then maintains EM_SDCKE low until it exits the power down state.

Since the EMIF services the refresh backlog before it enters the power down state, all internal banks of the SDRAM are closed (pre-charged) prior to issuing the POWER DOWN command. Therefore, the EMIF only supports pre-charge power down. The EMIF does not support active power down where internal banks of the SDRAM are open (active) before the POWER DOWN command is issued.

During the power down state, the EMIF services the SDRAM, asynchronous memory, and register accesses as normal, returning to the power down state upon completion.

The PDWR bit in the SDRAM configuration register indicates whether the EMIF should perform refreshes in power down state. If the PDWR bit is set, the EMIF exits power down state every time the Refresh Must level is reached, it performs REFR commands to the SDRAM, and then it returns back to the power down state. This evenly distributes the refreshes to the SDRAM in power down state. If the PDWR bit is not set, the EMIF does not perform any refreshes to the SDRAM. Therefore the data integrity of the SDRAM is not guaranteed upon power down exit if the PDWR bit is not set.

If the PD bit is cleared while in the power down state, the EMIF will come out of the power down state. To exit the power down state the EMIF drives EM_SDCKE high and enters its idle state.

1.2.6.9 SDRAM Read Operation

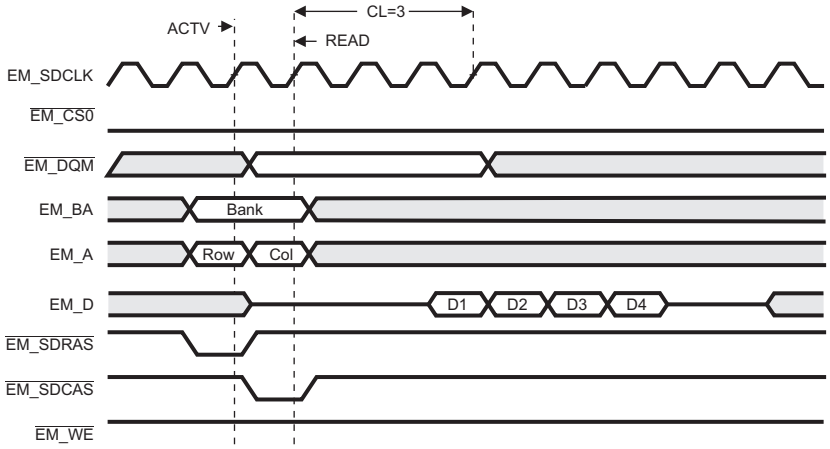
When the EMIF receives a read request to SDRAM, it performs one or more read access cycles. A read access cycle begins with the issuing of the ACTV command to select the desired bank and row of the SDRAM device. After the row has been opened, the EMIF proceeds to issue a READ command while specifying the desired bank and column address. The address pin EM_A[10] is held low during the READ command to avoid auto-pre-charging. The READ command signals the SDRAM device to start bursting data from the specified address while the EMIF issues NOP commands. Following a READ command, the CL field of SDCR1 defines how many delay cycles will be present before the read data appears on the data bus. This is referred to as the CAS latency.

Figure 1-5 shows the signal waveforms for a basic SDRAM read operation in which a burst of data is read from a single page. A burst size of eight is always used for SDRAM accesses.

The EMIF will truncate a series of bursting data if the remaining addresses of the burst are not required to complete the request. The EMIF can truncate the burst in three ways:

- By issuing another READ to the same page in the same bank.
- By issuing a PRE command in order to prepare for accessing a different page of the same bank.
- By issuing a BT command in order to prepare for accessing a page in a different bank.

Figure 1-5. Timing Waveform for Basic SDRAM Read Operation



Several other pins are also active during a read access. The EM_DQM[1:0] pins are driven low during the READ commands and are kept low during the NOP commands that correspond to the burst request. The state of the other EMIF pins during each command can be found in Table 1-6.

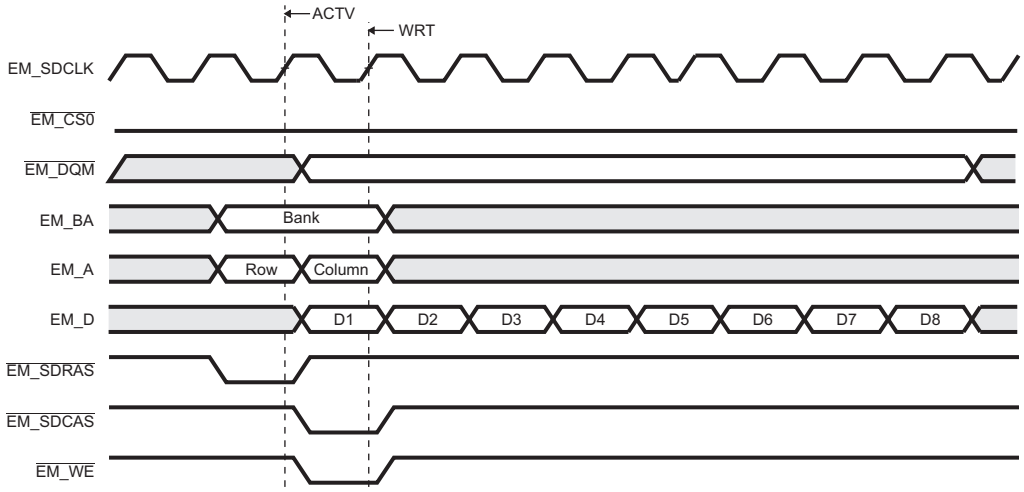
The EMIF schedules its commands based on the timing information that is provided to it in the SDRAM timing registers (SDTIMR1 and SDTIMR2). The values for the timing parameters in these registers should be chosen to satisfy the timing requirements listed in the SDRAM datasheet. The EMIF uses this timing information to avoid violating any timing constraints related to issuing commands. This is commonly accomplished by inserting NOP commands between various commands during an access. For more details on the various timing parameters, see the register description of SDTIMR1 and SDTIMR2 in Section 1.4.7.

1.2.6.10 SDRAM Write Operation

When the EMIF receives a write request to SDRAM it performs one or more write-access cycles. A write-access cycle begins with the issuing of the ACTV command to select the desired bank and row of the SDRAM device. After the row has been opened, the EMIF proceeds to issue a WRT command while specifying the desired bank and column address. The address pin EM_A[10] is held low during the WRT command to avoid automatic pre-charging. The WRT command signals the SDRAM device to start writing a burst of data to the specified address while the EMIF issues NOP commands. The associated write data will be placed on the data bus in the cycle concurrent with the WRT command and with subsequent burst continuation NOP commands.

Figure 1-6 shows the signal waveforms for a basic SDRAM write operation in which a burst of data is written to a single page. A burst size of eight is always used for SDRAM accesses.

Figure 1-6. Timing Waveform for Basic SDRAM Write Operation



The EMIF will truncate a series of bursting data if the remaining addresses of the burst are not part of the write request. The EMIF can truncate the burst in three ways:

- By issuing another WRT to the same page.
- By issuing a PRE command in order to prepare for accessing a different page of the same bank.
- By issuing a BT command in order to prepare for accessing a page in a different bank.

Several other pins are also active during a write access. The EM_DQM pins are driven to select which bytes of the data word will be written to the SDRAM device. They are also used to mask out entire undesired data words during a burst access. The state of the other EMIF pins during each command can be found in Table 1-6.

The EMIF schedules its commands based on the timing information that is provided to it in the SDRAM timing registers (SDTIMR1 and SDTIMR2). The values for the timing parameters in these registers should be chosen to satisfy the timing requirements listed in the SDRAM datasheet. The EMIF uses this timing information to avoid violating any timing constraints related to issuing commands. This is commonly accomplished by inserting NOP commands during various cycles of an access. For more details on the various timing parameters, see the register description of SDTIMR1 and SDTIMR2 in Section 1.4.7.

1.2.6.11 Mapping from Logical Address to SDRAM Address

The EMIF interleaves the internal banks for SDRAM connected to both the chip selects. From the system point of view, the external SDRAM is seen as one block of SDRAM. If smaller devices are used, the memory is seen to rollover.

When addressing SDRAM, if the IBANK_POS bit in SDCR2 is set to 0 (recommended), the EMIF uses the three fields IBANK, EBANK and PAGESIZE in SDCR1 to determine the mapping from source address to SDRAM row, column, bank and chip select. If the IBANK_POS bit is set to 1, the EMIF uses the four fields IBANK, EBANK, PAGESIZE, and ROWSIZE in SDCR1 and SDCR2 to determine the mapping from source address to SDRAM row, column, bank, and chip select. In all cases the EMIF considers its SDRAM address space to be a single logical block regardless of the number of physical devices or whether the devices are mapped across one or two EMIF chip selects.

For IBANK_POS = 0, Table 1-14 shows which logical address bits map to the SDRAM row, column, bank and chip select bits for all combinations of IBANK, EBANK, and PAGESIZE. For IBANK_POS = 1, Table 1-15 shows which logical address bits map to the SDRAM row, column, bank, and chip select bits for all combinations of IBANK, EBANK, PAGESIZE, and ROWSIZE.

Table 1-14. Mapping of Logical Address to SDRAM Address (IBANK_POS = 0)[1] [2]

EBANK	IBANK	PAGESIZE	Logical Address								
			30:23:00	22:15	14	13	12	11	10	9	8:01
0	0	0	x	nrb=14							ncb=8
1	0	0	x	nrb=13						nce=1	ncb=8
0	1	0	x	nrb=13						nbb=1	ncb=8
1	1	0	x	nrb=12					nce=1	nbb=1	ncb=8
0	2	0	x	nrb=12					nbb=2		ncb=8
1	2	0	x	nrb=11				nce=1	nbb=2		ncb=8
0	0	1	x	nrb=13						ncb=9	
1	0	1	x	nrb=12					nce=1	ncb=9	
0	1	1	x	nrb=12					nbb=1	ncb=9	
1	1	1	x	nrb=11				nce=1	nbb=1	ncb=9	
0	2	1	x	nrb=11				nbb=2		ncb=9	
1	2	1	x	nrb=10			nce=1	nbb=2		ncb=9	
0	0	2	x	nrb=12					ncb=10		
1	0	2	x	nrb=11				nce=1	ncb=10		
0	1	2	x	nrb=11				nbb=1	ncb=10		
1	1	2	x	nrb=10			nce=1	nbb=1	ncb=10		
0	2	2	x	nrb=10			nbb=2		ncb=10		
1	2	2	x	nrb=9		nce=1	nbb=2		ncb=10		
0	0	3	x	nrb=11				ncb=11			
1	0	3	x	nrb=10			nce=1	ncb=11			
0	1	3	x	nrb=10			nbb=1	ncb=11			
1	1	3	x	nrb=9		nce=1	nbb=1	ncb=11			
0	2	3	x	nrb=9		nbb=2		ncb=11			
1	2	3	x	nrb=8		nce=1	nbb=2	ncb=11			

(1) LEGEND: nrb = number of row address bits; ncb = number of column address bits; nbb = number of bank address bits; nce = number of chip select bits.

(2) Not all combinations of IBANK, ROWSIZE, EBANK, and PAGESIZE are valid. Table 1-14 shows only those combinations that are valid.

For IBANK_POS = 0, the effect of the above address-mapping scheme is that as the logical address increments across SDRAM page boundaries, the EMIF moves onto the same page in the next bank in the current device ($\overline{\text{EM\_CS0}}$). This movement along the banks of the current device continues until the page has been accessed in all banks in the current device. The EMIF then proceeds to the same page in the next device (if EBANK = 1, $\overline{\text{EM\_CS1}}$) and proceeds through the same page in all its banks before moving over to the next page in the first device ($\overline{\text{EM\_CS0}}$). The EMIF exploits this traversal across internal banks and chip selects while remaining on the same page to maximize the number of open SDRAM banks within the overall SDRAM space. Thus, the EMIF can keep a maximum of 8 banks (4 internal banks across 2 chip selects) open at a time.

Table 1-15. Mapping of Logical Address to SDRAM Address (IBANK_POS = 1) (1) (2)

IBANK	ROWSIZE	EBANK	PAGESIZE	Logical Address											
				30:23:00	22	21	20	19	18	17:13	12	11	10	9	8:01
0	0	0	0	x	x	x	x	x	x	nrb=9					ncb=8
0	0	1	0	x	x	x	x	x	nrb=9					nce=1	ncb=8
0	1	0	0	x	x	x	x	x	nrb=10						ncb=8
0	1	1	0	x	x	x	x	nrb=10						nce=1	ncb=8
0	2	0	0	x	x	x	x	nrb=11							ncb=8
0	2	1	0	x	x	x	nrb=11							nce=1	ncb=8
0	3	0	0	x	x	x	nrb=12								ncb=8
0	3	1	0	x	x	nrb=12								nce=1	ncb=8
0	4	0	0	x	x	nrb=13									ncb=8
0	4	1	0	x	nrb=13									nce=1	ncb=8
0	5	0	0	x	nrb=14										ncb=8
1	0	0	0	x	x	x	x	x	nbb=1	nrb=9					ncb=8
1	0	1	0	x	x	x	x	nbb=1	nrb=9					nce=1	ncb=8
1	1	0	0	x	x	x	x	nbb=1	nrb=10						ncb=8
1	1	1	0	x	x	x	nbb=1	nrb=10						nce=1	ncb=8
1	2	0	0	x	x	x	nbb=1	nrb=11							ncb=8
1	2	1	0	x	x	nbb=1	nrb=11							nce=1	ncb=8
1	3	0	0	x	x	nbb=1	nrb=12								ncb=8
1	3	1	0	x	nbb=1	nrb=12								nce=1	ncb=8
1	4	0	0	x	nbb=1	nrb=13									ncb=8
2	0	0	0	x	x	x	x	x	nbb=2	nrb=9					ncb=8
2	0	1	0	x	x	x	x	nbb=2	nrb=9					nce=1	ncb=8
2	1	0	0	x	x	x	x	nbb=2	nrb=10						ncb=8
2	1	1	0	x	x	x	nbb=2	nrb=10						nce=1	ncb=8
2	2	0	0	x	x	x	nbb=2	nrb=11							ncb=8
2	2	1	0	x	nbb=2	nrb=11								nce=1	ncb=8
2	3	0	0	x	nbb=2	nrb=12									ncb=8
0	0	0	1	x	x	x	x	x	nrb=9					ncb=9	
0	0	1	1	x	x	x	x	nrb=9					nce=1	ncb=9	
0	1	0	1	x	x	x	x	nrb=10						ncb=9	
0	1	1	1	x	x	x	nrb=10						nce=1	ncb=9	
0	2	0	1	x	x	x	nrb=11							ncb=9	
0	2	1	1	x	x	nrb=11							nce=1	ncb=9	
0	3	0	1	x	x	nrb=12								ncb=9	
0	3	1	1	x	nrb=12								nce=1	ncb=9	
0	4	0	1	x	nrb=13									ncb=9	
1	0	0	1	x	x	x	x	nbb=1	nrb=9					ncb=9	
1	0	1	1	x	x	x	nbb=1	nrb=9					nce=1	ncb=9	
1	1	0	1	x	x	x	nbb=1	nrb=10						ncb=9	
1	1	1	1	x	x	nbb=1	nrb=10						nce=1	ncb=9	
1	2	0	1	x	x	nbb=1	nrb=11							ncb=9	
1	2	1	1	x	nbb=1	nrb=11							nce=1	ncb=9	
1	3	0	1	x	nbb=1	nrb=12								ncb=9	
2	0	0	1	x	x	x	nbb=2	nrb=9						ncb=9	
2	0	1	1	x	x	nbb=2	nrb=9						nce=1	ncb=9	
2	1	0	1	x	x	nbb=2	nrb=10							ncb=9	
2	1	1	1	x	nbb=2	nrb=10							nce=1	ncb=9	
2	2	0	1	x	nbb=2	nrb=11								ncb=9	
0	0	0	2	x	x	x	x	nrb=9					ncb=10		
0	0	1	2	x	x	x	nrb=9					nce=1	ncb=10		
0	1	0	2	x	x	x	nrb=10						ncb=10		
0	1	1	2	x	x	nrb=10						nce=1	ncb=10		
0	2	0	2	x	x	nrb=11							ncb=10		

(1) LEGEND: nrb = number of row address bits; ncb = number of column address bits; nbb = number of bank address bits; nce = number of chip select bits.

(2) Not all combinations of IBANK, ROWSIZE, EBANK, and PAGESIZE are valid. Table 1-15 shows only those combinations that are valid.

Table 1-15. Mapping of Logical Address to SDRAM Address (IBANK_POS = 1) [1] [2] (continued)

IBANK	ROWSIZE	EBANK	PAGESIZE	Logical Address										
				30:23:00	22	21	20	19	18	17:13	12	11	10	9	8:01
0	2	1	2	x	nrb=11							nce=1	ncb=10		
0	3	0	2	x	nrb=12								ncb=10		
1	0	0	2	x	x	x	nbb=1	nrb=9					ncb=10		
1	0	1	2	x	x	nbb=1	nrb=9					nce=1	ncb=10		
1	1	0	2	x	x	nbb=1	nrb=10						ncb=10		
1	1	1	2	x	nbb=1	nrb=10						nce=1	ncb=10		
1	2	0	2	x	nbb=1	nrb=11							ncb=10		
2	0	0	2	x	x	nbb=2	nrb=9						ncb=10		
2	0	1	2	x	nbb=2	nrb=9						nce=1	ncb=10		
2	1	0	2	x	nbb=2	nrb=10							ncb=10		
0	0	0	3	x	x	x	nrb=9					ncb=11			
0	0	1	3	x	x	nrb=9					nce=1	ncb=11			
0	1	0	3	x	x	nrb=10						ncb=11			
0	1	1	3	x	nrb=10						nce=1	ncb=11			
0	2	0	3	x	nrb=11							ncb=11			
1	0	0	3	x	x	nbb=1	nrb=9					ncb=11			
1	0	1	3	x	nbb=1	nrb=9					nce=1	ncb=11			
1	1	0	3	x	nbb=1	nrb=10						ncb=11			
2	0	0	3	x	nbb=2	nrb=9						ncb=11			

For IBANK_POS = 1, the effect of the address-mapping scheme is that as the source address increments across SDRAM page boundaries, the EMIF moves onto the same page in the same bank in the next device (if EBANK = 1, $\overline{\text{EM\_CS1}}$). The EMIF then proceeds to the next page in the same bank in the first device ($\overline{\text{EM\_CS0}}$). This movement along the same banks of both devices continues until all the pages have been accessed in the same bank in both the devices. The EMIF then proceeds to the next bank in the first device ($\overline{\text{EM\_CS0}}$). Thus, the EMIF can keep a maximum of 2 banks across 2 chip selects open at a time. Since the EMIF can keep less number of banks open, this case is lower in performance than the IBANK_POS = 0 case. Thus this case is only recommended to be used along with Partial Array Self Refresh for mobile SDR SDRAM where performance can be traded off for power savings.

1.2.7 Asynchronous Controller and Interface

The EMIF easily interfaces to a variety of asynchronous devices including: NOR Flash, NAND Flash, and SRAM. It can be operated in two major modes (see Table 1-16):

- Normal Mode
- Select Strobe Mode

The default mode of operation is Normal Mode, in which the $\overline{\text{EM\_DQM}}$ pins of the EMIF function as byte enables. In this mode, the $\overline{\text{EM\_CS[5:2]}}$ pins behave as typical chip select signals, remaining active for the duration of the asynchronous access. See Section 1.2.7.1 for an example interface with multiple 8-bit devices.

In Select Strobe Mode the $\overline{\text{EM\_CS[5:2]}}$ pins act as a strobe-active only during the strobe period of an access. In this mode, the $\overline{\text{EM\_DQM}}$ pins of the EMIF function as standard byte enables for reads and writes.

A summary of the differences between the two modes of operation are shown in Table 1-16. For the details of asynchronous operations in Normal Mode, see Section 1.2.7.4 and for the details of asynchronous operations in Select Strobe Mode, see Section 1.2.7.5.

The EMIF hardware defaults to Normal Mode for each chip select space, but can be manually switched to Select Strobe Mode by setting the SS bit of the Asynchronous CSn Configuration Register 2 (ACSnCR2).

Table 1-16. Normal Mode vs. Select Strobe Mode

Mode	Function of $\overline{\text{EM\_DQM}}$ Pins	Operation of $\overline{\text{EM\_CS[5:2]}}$ Pins
Normal Mode	Byte enables	Active during the entire asynchronous access cycle
Select Strobe Mode	Byte enables	Active only during the strobe period of an access cycle

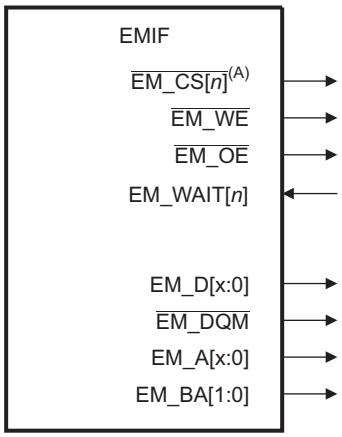
In both Normal Mode and Select Strobe Mode, the EMIF can be configured to operate in a sub-mode called NAND Flash Mode. In NAND Flash Mode, the EMIF is able to calculate an error correction code (ECC) for transfers up to 512 bytes.

The EMIF also provides configurable cycle timing parameters and an Extended Wait Mode that allows the connected device to extend the strobe period of an access cycle. The following sections describe the features related to interfacing with external asynchronous devices.

1.2.7.1 Interfacing to Asynchronous Memory

Figure 1-7 shows the EMIF's external pins used in interfacing with an asynchronous device. The pin $\overline{\text{EM\_CS[n]}}$ can be any of these chip select pins: $\overline{\text{EM\_CS[5:2]}}$.

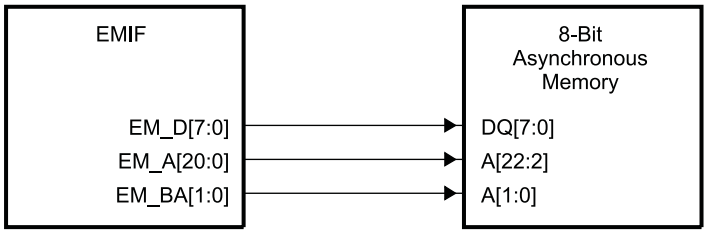
Figure 1-7. EMIF Asynchronous Interface



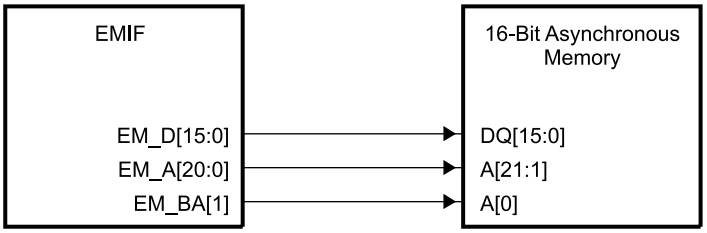
A. n=2, 3, 4, or 5

Of special note is the connection between the EMIF and the external device's address bus. The EMIF address pins EM_A[20:0] always provide the least significant bits of a double-word (32-bit) address. The EM_BA[1:0] pins provide word (16-bit) and byte selection functionality according to the data bus width configured in the Asynchronous CSn Configuration Register 1 (ACSnCR1). Figure 1-8 shows the mapping between the EMIF's and the connected device's data and address pins for an 8- and 16-bit data bus configuration.

Figure 1-8. Connecting Data and Address Bus to Asynchronous Memory Devices



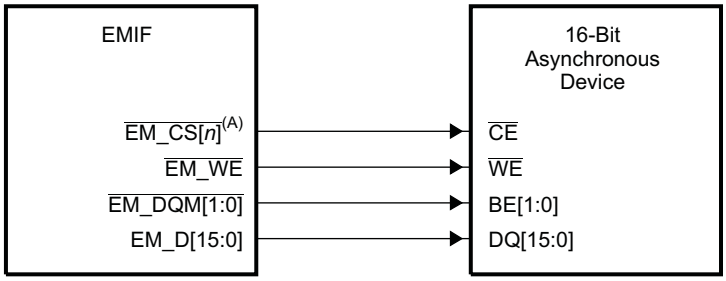
a) EMIF to 8-bit memory interface



b) EMIF to 16-bit memory interface

Figure 1-9 shows a common interface between the EMIF and an external asynchronous memory with byte enables. The EMIF should be operated in either Normal Mode or Select Strobe Mode when using this interface so that the $\overline{\text{EM\_DQM}}$ signals operate as byte enables.

Figure 1-9. Common Asynchronous Interface



A. n=2, 3, 4, or 5

1.2.7.2 Accessing Larger Asynchronous Memories

The external memory address space within the DSP's memory map and the number of dedicated EMIF address pins limits the maximum size of any asynchronous memory device connected to the EMIF. If an asynchronous memory device with a size larger than that allowed by these restrictions is needed, then GPIO pins may be used to control the upper address lines of the memory device.

An approach like this is useful in a system which requires that application code be loaded from slower/larger flash memory into faster/smaller internal or SDRAM memory. In this type of system, the DSP's ROM Bootloader can load a secondary Bootloader from flash. The secondary Bootloader can then finish the boot process by loading the actual application code.

When the ROM Bootloader copies the secondary Bootloader from the lower portion of the flash it does not need to manipulate the upper address lines. Only the secondary Bootloader, which is board-specific and is stored in the external flash, needs to know which GPIO pins have been assigned to the function of upper address lines. Therefore, the secondary Bootloader can perform the task of configuring the selected pins as GPIO and loading the remainder of the code from the upper flash memory.

The ROM Bootloader assumes that any GPIO pins used to control the upper address lines of the flash memory will be pulled to ‘0’ after reset. This means that normally the GPIO pins selected for this function will be either spare or used as outputs only by the application, and therefore can be pulled to ‘0’ at reset with an external pull-down resistor. The GPIO pins chosen should be tri-stated by default on device reset. For details on which GPIO-capable pins are tri-stated on device reset, see the device Data Manual.

1.2.7.3 Configuring the EMIF for Asynchronous Accesses

The operation of the EMIF's asynchronous interface can be configured by programming the appropriate register fields. The tables below list the register fields that can be programmed and describe the purpose of each field. These registers must be programmed prior to accessing the external memory. A transfer following a write to these registers will use the new configuration.

Note: Section 1.4 provides the reset value and bit position for each register field. However, the Bootloader documentation should be consulted to determine if the fields are programmed during boot.

Table 1-17. Description of the Asynchronous CSn Configuration Registers (ACSnCR1 and ACSnCR2)

Parameter	Description
SS	**Select Strobe mode**. This bit selects the EMIF’s mode of operation in the following way: • SS = 0h selects Normal Mode – $\overline{\text{EM\_DQM}}$ pins function as byte enables – $\overline{\text{EM\_CS}}$[5:2] pins are active for duration of access • SS = 1h selects Select Strobe Mode – $\overline{\text{EM\_DQM}}$ pins function as byte enables – $\overline{\text{EM\_CS}}$[5:2] pins acts as a strobe
EW	**Extended Wait Mode enable**. • EW = 0h disables Extended Wait Mode • EW = 1h enables Extended Wait Mode When set to 1, the EMIF enables its Extended Wait Mode in which the strobe width of an access cycle can be extended in response to the assertion of the wait pins (EM_WAIT[3:0]). The WPn bits in the Asynchronous Wait Cycle Configuration Register 2 (AWCCR2) controls to polarity of wait pins. **NOTE:** Extended Wait Mode should not be used while in NAND Flash Mode. See Section 1.2.7.6 for more details on this mode of operation
W_SETUP/R_SETUP	**Ready/Write setup widths**. These fields define the number of EMIF clock cycles of setup time for the address pins (EM_A[20:0] and EM_BA[1:0]), byte enables ($\overline{\text{EM\_DQM}}$[1:0]), and asynchronous chip select pins ($\overline{\text{EM\_CS}}$[5:2]) before the read strobe pin ($\overline{\text{EM\_OE}}$) or write strobe pin ($\overline{\text{EM\_WE}}$) falls, minus one cycle. For writes, the W_SETUP field also defines the setup time for the data pins (EM_D[15:0]). See the datasheet of the external asynchronous device to determine the appropriate setting for this field.
W_STROBE/R_STROBE	**Read/Write strobe widths**. These fields define the number of EMIF clock cycles between the falling and rising of the read strobe pin ($\overline{\text{EM\_OE}}$) or write strobe pin ($\overline{\text{EM\_WE}}$), minus one cycle. If Extended Wait Mode is enabled (EW = 1), these fields must be set to a value greater than zero. See the datasheet of the external asynchronous device to determine the appropriate setting for this field.
W_HOLD/R_HOLD	**Read/Write hold widths**. These fields define the number of EMIF clock cycles of hold time for the address pins (EM_A[20:0] and EM_BA[1:0]), byte enables ($\overline{\text{EM\_DQM}}$[1:0]), and asynchronous chip select pins ($\overline{\text{EM\_CS}}$[5:2]) after the read strobe pin ($\overline{\text{EM\_OE}}$) or write strobe pin ($\overline{\text{EM\_WE}}$) rises, minus one cycle. For writes, the W_HOLD field also defines the hold time for the data pins (EM_D[15:0]). See the datasheet of the external asynchronous device to determine the appropriate setting for this field.
TA	**Minimum turnaround time.** This field defines the minimum number of EMIF clock cycles between asynchronous reads and writes, minus one cycle. The purpose of this feature is to avoid contention on the bus. The value written to this field also determines the number of cycles that will be inserted between asynchronous accesses and SDRAM accesses. See the datasheet of the external asynchronous device to determine the appropriate setting for this field.

Table 1-17. Description of the Asynchronous CSn Configuration Registers (ACSnCR1 and ACSnCR2) (continued)

Parameter	Description
ASIZE	**Asynchronous device bus width.** This field determines the data bus width of the asynchronous interface in the following way: • ASIZE = 0h selects an 8-bit bus • ASIZE = 1h selects a 16-bit bus The configuration of ASIZE determines the function of the EM_A[20:0] and EM_BA[1:0] pins as described in Section 1.2.7.1. This field also determines the number of external accesses required to fulfill a request generated by one of the sources mentioned in Section 1.2.6.7 . For example, a request for a double-word (32 bits) would require four external access when ASIZE = 0h. See the datasheet of the external asynchronous device to determine the appropriate setting for this field.

Table 1-18. Description of the Asynchronous Wait Cycle Configuration Registers (AWCCR1 and AWCCR2)

Parameter	Description
WP[3:0]	**Wait pin polarity.** • WP*n* = 0h selects active-high polarity • WP*n* = 1h selects active-low polarity When set to 1, the EMIF will wait if the wait pin is high. When set to 0, the EMIF will wait if the wait pin is low. The EMIF must have the Extended Wait Mode enabled for the wait pins to affect the width of the strobe period. The polarity of the wait pins is programmable. **NOTE:** The polarity of the wait pins is not programmable in NAND Flash Mode.
MEWC	**Maximum Extended Wait Cycles.** This field configures the number of EMIF clock cycles the EMIF will wait for the wait pins (EM_WAIT[3:0]) to be deactivated during the strobe period of an access cycle. The maximum number of EMIF clock cycles it will wait is determined by the following formula: Maximum Extended Wait Cycles = (MEWC + 1) * 16 If the wait pin is not deactivated within the time specified by this field, the EMIF resumes the access cycle, registering whatever data is on the bus and preceding to the hold period of the access cycle. This situation is referred to as an Asynchronous Timeout. An Asynchronous Timeout generates an interrupt if it has been enabled in the EMIF Interrupt Mask Set Register (EIMSR). For more information about the EMIF's interrupts, see Section 1.2.14. **NOTE:** Extended Wait Mode should not be used while in NAND Flash Mode. See Section 1.2.7.6 for more details on this mode of operation.

Table 1-19. Description of the EMIF Interrupt Mask Set Register (EIMSR)

Parameter	Description
WRMSET	**Wait Rise Mask Set.** Writing a 1 to this bit enables an interrupt to be generated when a rising edge on a wait pin occurs while in NAND Flash Mode
ATMSET	**Asynchronous Timeout Mask Set.** Writing a 1 to this bit enables an interrupt to be generated when an Asynchronous Timeout occurs.

Table 1-20. Description of the EMIF Interrupt Mask Clear Register (EIMCR)

Parameter	Description
WRMCLR	**Wait Rise Mask Clear.** Writing a 1 to this bit disables the interrupt, clearing the WRMSET bit in the EMIF interrupt mask set register (EIMSR).
ATMCLR	**Asynchronous Timeout Mask Clear.** Writing a 1 to this bit enables an interrupt to be generated when an Asynchronous Timeout occurs.

1.2.7.4 Read and Write Operations in Normal Mode

Normal Mode is the asynchronous interface's default mode of operation. It is selected when the SS bit of the Asynchronous CSn Configuration Register 2 (ACSnCR2) is cleared to 0. In this mode, the EM_DQM pins operate as byte enables. Section 1.2.7.4.1 and Section 1.2.7.4.2 explain the details of read and write operations while in Normal Mode.

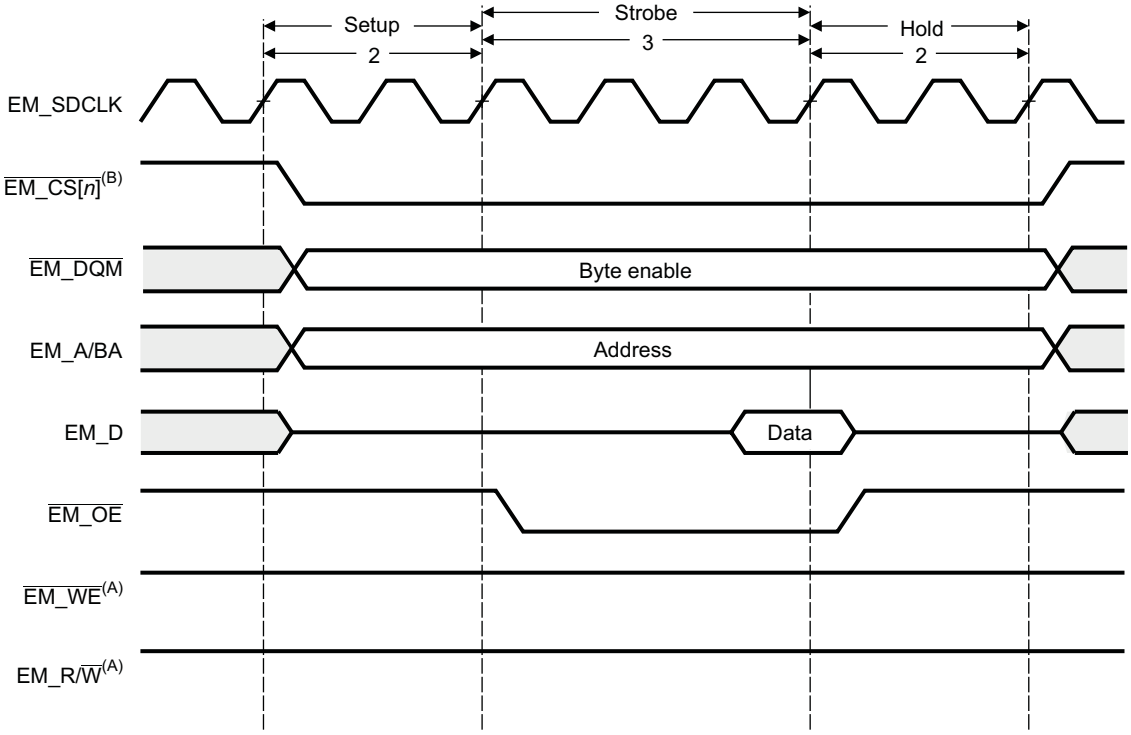
1.2.7.4.1 Asynchronous Read Operations (Normal Mode)

An asynchronous read is performed when any of the requesters mentioned in Section 1.2.2 request a read from the attached asynchronous memory. After the request is received, a read operation is initiated once it becomes the EMIF's highest priority task, according to the priority scheme detailed in Section 1.2.10. In the event that the read request cannot be serviced by a single access cycle to the external device, multiple access cycles will be performed by the EMIF until the entire request is fulfilled. The details of an asynchronous read operation in Normal Mode are described in Table 1-21. Also, Figure 1-10 shows an example timing diagram of a basic read operation.

Table 1-21. Asynchronous Read Operation in Normal Mode

Time Interval	Pin Activity in Normal Mode
Turn-around period	Once the read operation becomes the highest priority task for the EMIF, the EMIF waits for the programmed number of turn-around cycles before proceeding to the setup period of the operation. The number of wait cycles is taken directly from the TA field of the Asynchronous CSn Configuration Register 1 (ACSnCR1). There are two exceptions to this rule: • If the current read operation was directly proceeded by another read operation, no turnaround cycles are inserted. • If the current read operation was directly proceeded by a write operation and the TA field has been set to 0h, one turn-around cycle will be inserted. After the EMIF has waited for the turnaround cycles to complete, it again checks to make sure that the read operation is still its highest priority task. If so, the EMIF proceeds to the setup period of the operation. If it is no longer the highest priority task, the EMIF terminates the operation.
Start of the setup period	The following actions occur at the start of the setup period: • The setup, strobe, and hold values are set according to the R_SETUP, R_STROBE, and R_HOLD values in ACSnCR1 and ACSNCR2. • The address pins EM_A[20:0] and EM_BA[1:0] become valid and carry the values described in Figure 1-10. • EM_DQM[1:0] becomes valid. • EM_CS[n] (where n = 2, 3, 4, or 5) falls to enable the external device (if not already low from a previous operation).
Strobe period	The following actions occur during the strobe period of a read operation: 1. EM_OE falls at the start of the strobe period 2. On the rising edge of the clock which is concurrent with the end of the strobe period: • EM_OE rises • The data on the EM_D[15:0] bus is sampled by the EMIF. In Figure 1-10, the wait pins (EM_WAIT[3:0]) are inactive. If wait pins are instead activated, the strobe period can be extended by the external device to give it more time to provide the data. Section 1.2.5 contains more details on using the wait pins.
End of the hold period	At the end of the hold period: • The address pins EM_A[20:0], EM_BA[1:0], and EM_DQM[1:0] become invalid • EM_CS[n] rises (if no more operations are required to complete the current request) EMIF may be required to issue additional read operations to a device with a small data bus width in order to complete an entire word access. In this case, the EMIF immediately re-enters the setup period to begin another operation without incurring the turn-round cycle delay. The setup, strobe, and hold values are not updated in this case. If the entire word access has been completed, the EMIF returns to its previous state unless another asynchronous request has been submitted and is currently the highest priority task. If this is the case, the EMIF instead enters directly into the turnaround period for the pending read or write operation.

Figure 1-10. Timing Waveform of an Asynchronous Read Cycle in Normal Mode



A. During the entirety of an asychronous read operation, the EM_WE and EM_R/W pins are driven high.
B. n=2, 3, 4, or 5

1.2.7.4.2 Asynchronous Write Operations (Normal Mode)

An asynchronous write is performed when any of the requesters mentioned in Section 1.2.2 request a write to memory in the asynchronous bank of the EMIF. After the request is received, a write operation is initiated once it becomes the EMIF's highest priority task, according to the priority scheme detailed in Section 1.2.10. In the event that the write request cannot be serviced by a single access cycle to the external device, multiple access cycles will be performed by the EMIF until the entire request is fulfilled. The details of an asynchronous write operation in Normal Mode are described in Table 1-22. Also, Figure 1-11 shows an example timing diagram of a basic write operation.

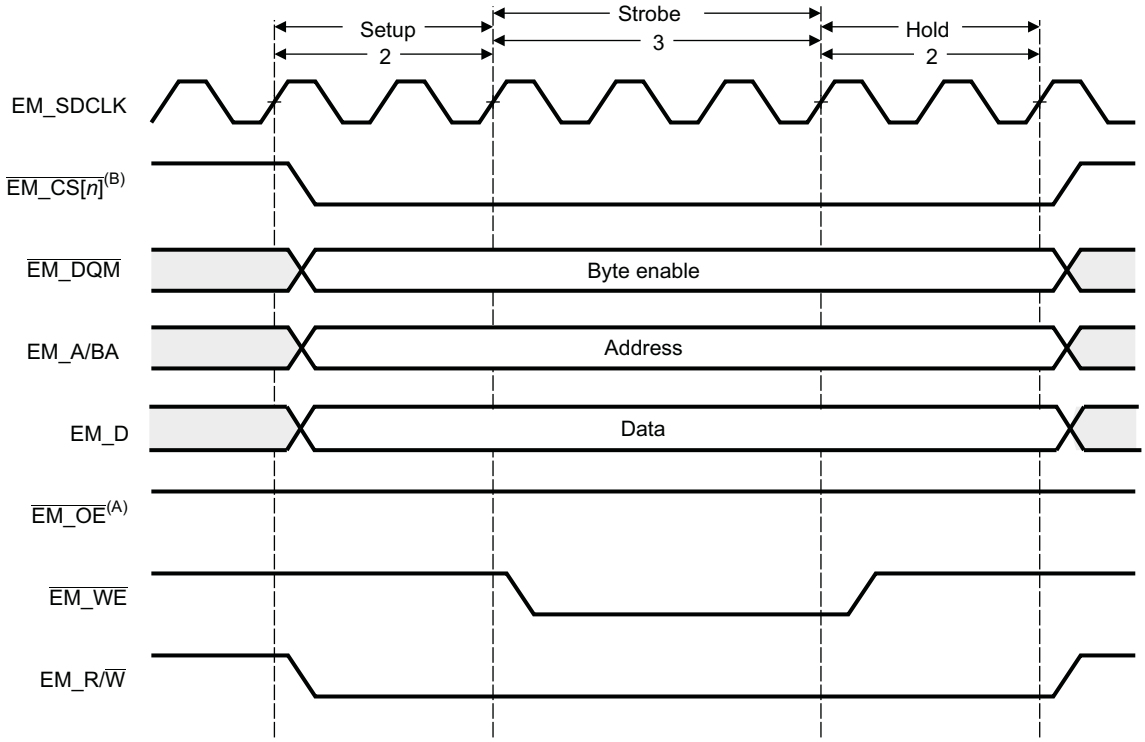
Table 1-22. Asynchronous Write Operation in Normal Mode

Time Interval	Pin Activity in Normal Mode
Turnaround period	Once the write operation becomes the highest priority task for the EMIF, the EMIF waits for the programmed number of turn-around cycles before proceeding to the setup period of the operation. The number of wait cycles is taken directly from the TA field of the Asynchronous CSn Configuration Register 1 (ACSnCR1). There are two exceptions to this rule: • If the current write operation was directly proceeded by another write operation, no turn-around cycles are inserted. • If the current write operation was directly proceeded by a read operation and the TA field has been set to 0h, one turnaround cycle will be inserted. After the EMIF has waited for the turn-around cycles to complete, it again checks to make sure that the write operation is still its highest priority task. If so, the EMIF proceeds to the setup period of the operation. If it is no longer the highest priority task, the EMIF terminates the operation.

Table 1-22. Asynchronous Write Operation in Normal Mode (continued)

Time Interval	Pin Activity in Normal Mode
Start of the setup period	The following actions occur at the start of the setup period: • The setup, strobe, and hold values are set according to the W_SETUP, W_STROBE, and W_HOLD values in ACSnCR2. • The address pins EM_A[20:0] and EM_BA[1:0] and the data pins EM_D[15:0] become valid. The EM_A[20:0] and EM_BA[1:0] pins carry the values described in Section 1.2.7.1. • $\overline{\text{EM\_DQM}}$[1:0] become valid. • The EM_R/$\overline{\text{W}}$ pin falls to indicate a write (if not already low from a previous operation). • $\overline{\text{EM\_CS}}$[n] (where n = 2, 3, 4, or 5) falls to enable the external device (if not already low from a previous operation).
Strobe period	The following actions occur at the start of the strobe period of a write operation: • $\overline{\text{EM\_WE}}$ falls • Normal Mode The following actions occur on the rising edge of the clock which is concurrent with the end of the strobe period: • $\overline{\text{EM\_WE}}$ rises • Normal Mode In Figure 1-11, the wait pins (EM_WAIT[3:0]) are inactive. If the wait pins are instead activated, the strobe period can be extended by the external device to give it more time to accept the data. Section 1.2.7.7 contains more details on using the wait pins.
End of the hold period	At the end of the hold period: • The address pins EM_A[20:0] and EM_BA[1:0] become invalid • The data pins become invalid • The EM_R/$\overline{\text{W}}$ pin rises (if no more operations are required to complete the current request) • $\overline{\text{EM\_CS}}$[n] rises (if no more operations are required to complete the current request) The EMIF may be required to issue additional write operations to a device with a small data bus width in order to complete an entire word access. In this case, the EMIF immediately re-enters the setup period to begin another operation without incurring the turnaround cycle delay. The setup, strobe, and hold values are not updated in this case. If the entire word access has been completed, the EMIF returns to its previous state unless another asynchronous request has been submitted and is currently the highest priority task. If this is the case, the EMIF instead enters directly into the turnaround period for the pending read or write operation.

Figure 1-11. Timing Waveform of an Asynchronous Write Cycle in Normal Mode



A. During the entirety of an asychronous write operation, the EMA_OE pin is driven high.
B. n=2, 3, 4, 5

1.2.7.5 Read and Write Operation in Select Strobe Mode

Select Strobe Mode is the EMIF's second mode of operation. It is selected when the SS bit of the Asynchronous CSn Configuration Register 2 (ACSnCR2) is set to 1. In this mode, the EM_DQM[1:0] pins operate as byte enables and the EM_CS[5:2] pins are only active during the strobe period of an access cycle. Section 1.2.7.5.1 and Section 1.2.7.5.2 explain the details of read and write operations while in Select Strobe Mode.

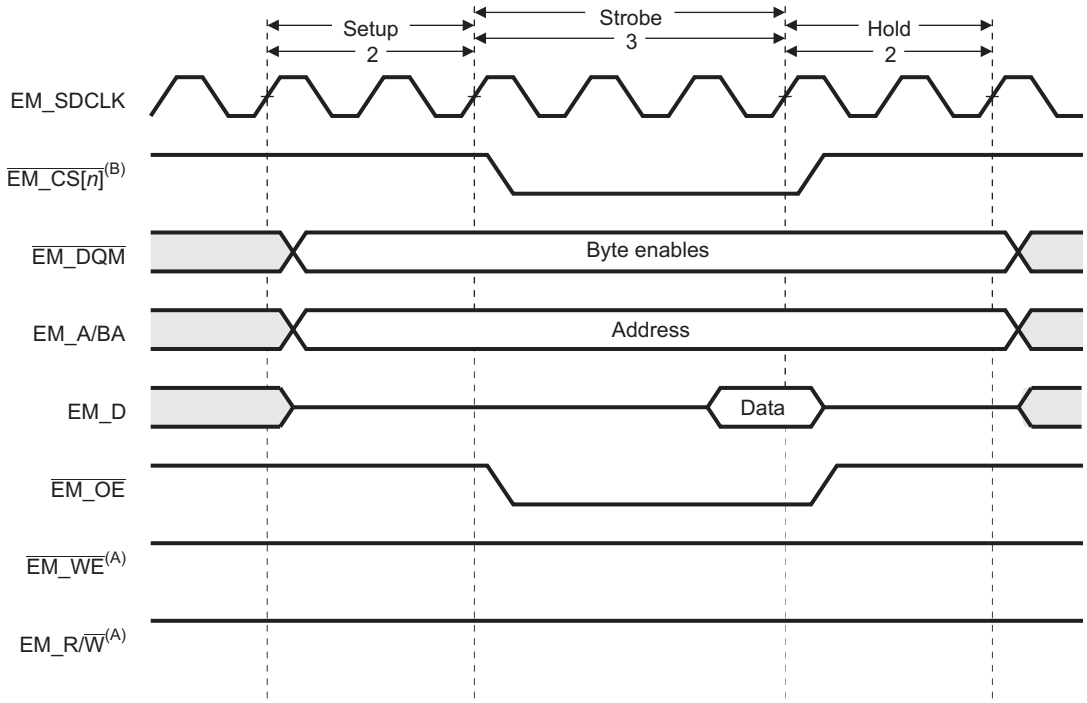
1.2.7.5.1 Asynchronous Read Operations (Select Strobe Mode)

An asynchronous read is performed when any of the requesters mentioned in Section 1.2.2 request a read from the attached asynchronous memory. After the request is received, a read operation is initiated once it becomes the EMIF's highest priority task, according to the priority scheme detailed in Section 1.2.10. In the event that the read request cannot be serviced by a single access cycle to the external device, multiple access cycles will be performed by the EMIF until the entire request is fulfilled. The details of an asynchronous read operation in Select Strobe Mode are described in Table 1-23. Also, Figure 1-12 shows an example timing diagram of a basic read operation.

Table 1-23. Asynchronous Read Operation in Select Strobe Mode

Time Interval	Pin Activity in Select Strobe Mode
Turnaround period	Once the read operation becomes the highest priority task for the EMIF, the EMIF waits for the programmed number of turn-around cycles before proceeding to the setup period of the operation. The number of wait cycles is taken directly from the TA field of the Asynchronous CSn Configuration Register 1 (ACSnCR1). There are two exceptions to this rule: • If the current read operation was directly proceeded by another read operation, no turn-around cycles are inserted. • If the current read operation was directly proceeded by a write operation and the TA field has been set to 0h, one turn-around cycle will be inserted. After the EMIF has waited for the turn-around cycles to complete, it again checks to make sure that the read operation is still its highest priority task. If so, the EMIF proceeds to the setup period of the operation. If it is no longer the highest priority task, the EMIF terminates the operation.
Start of the setup period	The following actions occur at the start of the setup period: • The setup, strobe, and hold values are set according to the R_SETUP, R_STROBE, and R_HOLD values in ACSnCR1 and ACSnCR2. • The address pins EM_A[20:0] and EM_BA[1:0] become valid and carry the values described in Section 1.2.7.1. • The $\overline{\text{EM\_DQM}}$ pins become valid as byte enables.
Strobe period	The following actions occur during the strobe period of a read operation: • $\overline{\text{EM\_CS}}$[n] (where n = 2, 3, 4, or 5) and $\overline{\text{EM\_OE}}$ fall at the start of the strobe period • On the rising edge of the clock which is concurrent with the end of the strobe period: – $\overline{\text{EM\_CS}}$[n] and $\overline{\text{EM\_OE}}$ rise – The data on the EM_D[15:0] bus is sampled by the EMIF. In Figure 1-12, the wait pins (EM_WAIT[3:0]) are inactive. If the wait pins are instead activated, the strobe period can be extended by the external device to give it more time to provide the data. Section 1.2.7.7 contains more details on using the wait pins.
End of the hold period	At the end of the hold period: • The address pins EM_A[20:0] and EM_BA[1:0] become invalid • The $\overline{\text{EM\_DQM}}$ pins become invalid The EMIF may be required to issue additional read operations to a device with a small data bus width in order to complete an entire word access. In this case, the EMIF immediately re-enters the setup period to begin another operation without incurring the turnaround cycle delay. The setup, strobe, and hold values are not updated in this case. If the entire word access has been completed, the EMIF returns to its previous state unless another asynchronous request has been submitted and is currently the highest priority task. If this is the case, the EMIF instead enters directly into the turnaround period for the pending read or write operation.

Figure 1-12. Timing Waveform of an Asynchronous Read Cycle in Select Strobe Mode



A. During the entirety of an asychronous read operation, the EM_WE and EM_R/W pins are driven high.
B. n=2, 3, 4, or 5

1.2.7.5.2 Asynchronous Write Operations (Select Strobe Mode)

An asynchronous write is performed when any of the requesters mentioned in Section 1.2.2 request a write to memory in the asynchronous bank of the EMIF. After the request is received, a write operation is initiated once it becomes the EMIF's highest priority task, according to the priority scheme detailed in Section 1.2.10 . In the event that the write request cannot be serviced by a single access cycle to the external device, multiple access cycles will be performed by the EMIF until the entire request is fulfilled. The details of an asynchronous write operation in Select Strobe Mode are described in Table 1-24 . Also, Figure 1-13 shows an example timing diagram of a basic write operation.

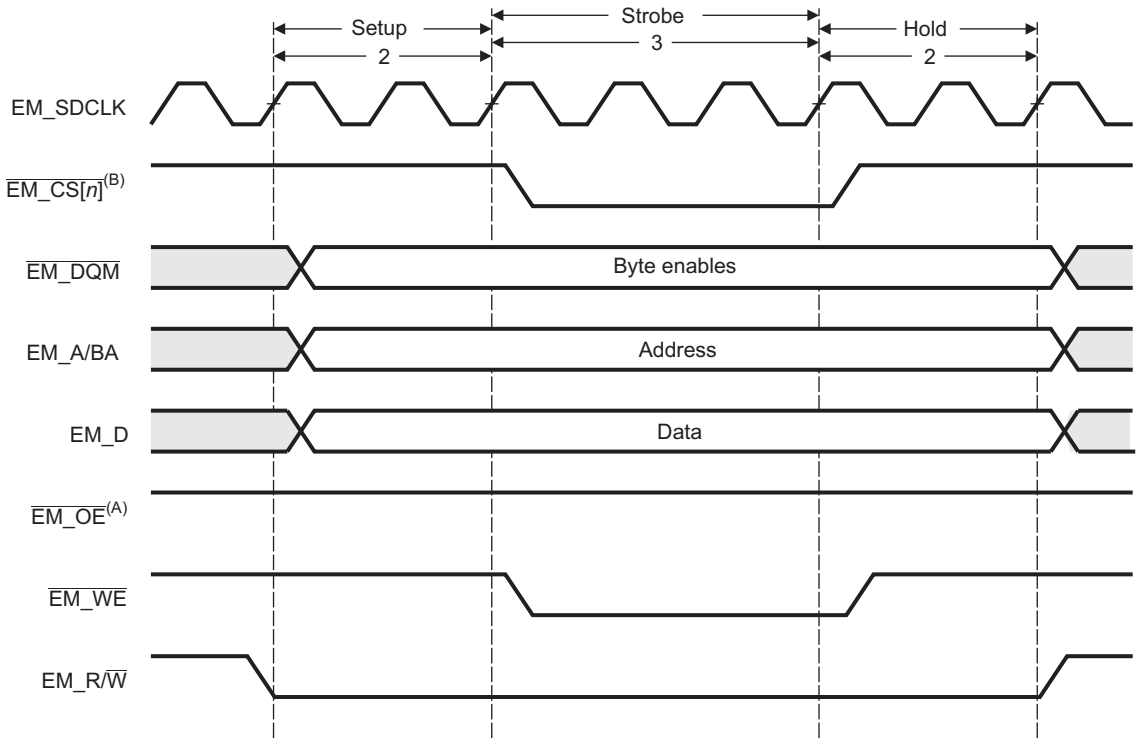
Table 1-24. Asynchronous Write Operation in Select Strobe Mode

Time Interval	Pin Activity in Select Strobe Mode
Turnaround period	Once the write operation becomes the highest priority task for the EMIF, the EMIF waits for the programmed number of turn-around cycles before proceeding to the setup period of the operation. The number of wait cycles is taken directly from the TA field of the Asynchronous CSn Configuration Register 1 (ACSnCR1). There are two exceptions to this rule: • If the current write operation was directly proceeded by another write operation, no turn-around cycles are inserted. • If the current write operation was directly proceeded by a read operation and the TA field has been set to 0h, one turnaround cycle will be inserted. After the EMIF has waited for the turn-around cycles to complete, it again checks to make sure that the write operation is still its highest priority task. If so, the EMIF proceeds to the setup period of the operation. If it is no longer the highest priority task, the EMIF terminates the operation.
Start of the setup period	The following actions occur at the start of the setup period: • The setup, strobe, and hold values are set according to the W_SETUP, W_STROBE, and W_HOLD values in ACSnCR2. • The address pins EM_A[20:0] and EM_BA[1:0] and the data pins EM_D[15:0] become valid. The EM_A[20:0] and EM_BA[1:0] pins carry the values described in Section 1.2.7.1. • The EM_R/W pin falls to indicate a write (if not already low from a previous operation). • EM_CS[n] (where n = 2, 3, 4, or 5) falls to enable the external device (if not already low from a previous operation).

Table 1-24. Asynchronous Write Operation in Select Strobe Mode (continued)

Time Interval	Pin Activity in Select Strobe Mode
Strobe period	The following actions occur at the start of the strobe period of a write operation: • $\overline{\text{EM\_WE}}$ falls • The $\overline{\text{EM\_DQM}}$ pins become active as write strobes. The following actions occur on the rising edge of the clock which is concurrent with the end of the strobe period: • $\overline{\text{EM\_WE}}$ rises • The $\overline{\text{EM\_DQM}}$ pins deactivate In Figure 1-13, the wait pins (EM_WAIT[3:0]) are inactive. If the wait pins are instead activated, the strobe period can be extended by the external device to give it more time to accept the data. Section 1.2.7.7 contains more details on using the wait pins.
End of the hold period	At the end of the hold period: • The address pins EM_A[20:0] and EM_BA[1:0] become invalid • The data pins become invalid • The EM_R/$\overline{\text{W}}$ pin rises (if no more operations are required to complete the current request) • $\overline{\text{EM\_CS[n]}}$ rises (if no more operations are required to complete the current request) The EMIF may be required to issue additional write operations to a device with a small data bus width in order to complete an entire word access. In this case, the EMIF immediately re-enters the setup period to begin another operation without incurring the turnaround cycle delay. The setup, strobe, and hold values are not updated in this case. If the entire word access has been completed, the EMIF returns to its previous state unless another asynchronous request has been submitted and is currently the highest priority task. If this is the case, the EMIF instead enters directly into the turnaround period for the pending read or write operation.

Figure 1-13. Timing Waveform of an Asynchronous Write Cycle in Select Strobe Mode



A. During the entirety of an asychronous write operation, the EM_OE pin is driven high.
B. n=2, 3, 4, or 5

1.2.7.6 NAND Flash Mode

NAND Flash Mode is a sub-mode of both Normal Mode and Select Strobe Mode. The chip select pins ($\overline{\text{EM\_CS[5:2]}}$) may be placed in NAND Flash mode by setting the CSn_USE_NAND bits in the NAND Flash control register (NANDFCR). Note that the NAND Flash Mode can be independently enabled for each chip select space. Table 1-25 displays the bit fields present in NANDFCR and briefly describes their use.

When a chip select space is configured to operate in NAND Flash mode, the EMIF hardware can calculate the error correction code (ECC) for each 512 byte data transfer to that chip select space. The EMIF hardware will not automatically generate a NAND access cycle, which includes the command, address, and data phases, necessary to complete a transfer to NAND Flash. All NAND Flash operations can be divided into single asynchronous cycles and with the help of software, the EMIF can execute a complete NAND access cycle.

NOTE: By default, the EM_CS2 pin is set to NAND Flash mode after reset. The NAND Flash mode is disabled for the other chip select pins (EM_CS[3:5]).

Table 1-25. Description of the NAND Flash Control Register (NANDFCR)

Parameter	Description
CS2_ECC_START	**NAND Flash ECC state for EM_CS2.** Set to 1 to start an ECC calculation for NAND Flash connected to EM_CS2. This bit is cleared to 0 when the NAND Flash 1-Bit ECC Registers for EMIF CS2 are read.
CS2_USE_NAND	**NAND Flash mode for EM_CS2.** Set to 1 to enable NAND Flash mode for EM_CS2.
CS3_ECC_START	**NAND Flash ECC state for EM_CS3.** Set to 1 to start an ECC calculation for NAND Flash connected to EM_CS3. This bit is cleared to 0 when the NAND Flash 1-Bit ECC Registers for EMIF CS3 are read.
CS3_USE_NAND	**NAND Flash mode for EM_CS3.** Set to 1 to enable NAND Flash mode for EM_CS3.
CS4_ECC_START	**NAND Flash ECC state for EM_CS4.** Set to 1 to start an ECC calculation for NAND Flash connected to EM_CS4. This bit is cleared to 0 when the NAND Flash 1-Bit ECC Registers for EMIF CS4 are read.
CS4_USE_NAND	**NAND Flash mode for EM_CS4.** Set to 1 to enable NAND Flash mode for EM_CS4.
CS5_ECC_START	**NAND Flash ECC state for EM_CS5.** Set to 1 to start an ECC calculation for NAND Flash connected to EM_CS5. This bit is cleared to 0 when the NAND Flash 1-Bit ECC Registers for EMIF CS5 are read.
CS5_USE_NAND	**NAND Flash mode for EM_CS5.** Set to 1 to enable NAND Flash mode for EM_CS5.

1.2.7.6.1 Configuring for NAND Flash Mode

Similar to the asynchronous accesses previously described, the EMIF's registers must be programmed appropriately to interface to a NAND Flash device. In addition to the fields listed in Table 1-25, you should set the CSn_USE_NAND bits of the NAND Flash Control Register (NANDFCR) to 1 for the chip select spaces you want to operate in NAND Flash Mode. Note that the Extended Wait Mode of any chip select space being used in NAND Flash Mode should be disabled by setting EW = 0 in the Asynchronous CSn Configuration Register 2 (ACSnCR2).

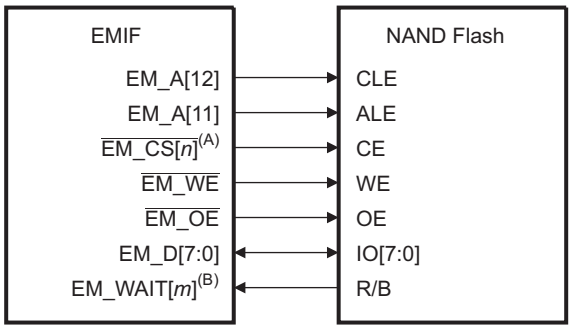
1.2.7.6.2 Connecting to NAND Flash

Figure 1-14 shows the EMIF external pins used to interface with a NAND Flash device. EMIF address lines are used to drive the NAND Flash device's command latch enable (CLE) and address latch enable (ALE) signals. Note that you can use any EMIF address lines to drive the CLE and ALE signals of the NAND Flash.

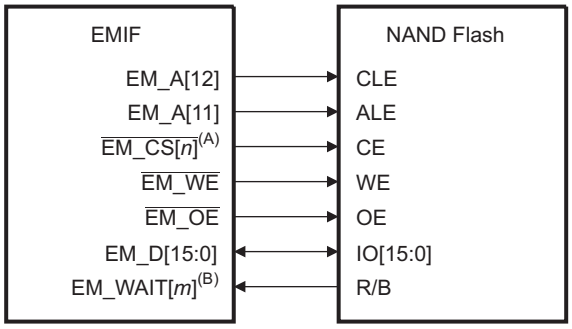
NOTE: The EMIF will not control the NAND Flash device's write protect pin. The write protect pin must be controlled outside of the EMIF.

NOTE: By default, the EM_CS2 pin is set to NAND Flash mode after reset. The NAND Flash mode is disabled for the other chip select pins (EM_CS[3:5]).

Figure 1-14. EMIF to NAND Flash Interface



a) Connection to 8-Bit NAND Device



b) Connection to 16-Bit NAND Device

A. n=2, 3, 4, or 5
B. m=0, 1, 2, or 3

1.2.7.6.3 Driving CLE and ALE

As described in Section 1.2.7.6.2, any of the EMIF address lines can be used to drive the ALE and CLE pins of the NAND Flash device. The following must be considered when using this approach:

- The CPU cannot generate 8-bit accesses to its data or I/O space. However, the EMIF can be programmed to access a single byte for every word access initiated by the CPU (see Section 1.2.8 for more details).
- As stated in Section 1.2.7.1, the EMIF always drives the least significant bit of a double-word (32-bit) address on EM_A[0].

Table 1-26 shows how a CPU word address (EM_A[21:0]) is driven on the EMIF address pins. Table 1-26 also shows how a DMA byte address is driven on the EMIF address pins.

Table 1-26. CPU and DMA Address to EMIF Address Pin Mapping

CPU Word Address	DMA Byte Address	EMIF Address Pins
A21	A22	EM_A[20]
…	…	…
A13	A14	EM_A[12]
A12	A13	EM_A[11]
…	…	…
A1	A2	EM_A[0]
A0	A1	EM_BA[1]
0	A0	EM_BA[0]

Notes:

- The EM_BA[1:0] pins are not required when interfacing to NAND Flash memory, however, they are included in this figure to provide a complete picture of CPU address to EMIF address pin mapping.
- The EM_BA[0] pin is only used if the EMIF data bus is configured as an 8-bit bus (ASIZE = 00b).

As an example, suppose that the EMIF address pins EM_A[12] and EM_A[11] are used to drive the CLE and ALE pins of the NAND Flash memory and that the $\overline{\text{EM\_CS2}}$ is used to drive the CE pin. In this situation, a write or read from the following CPU word addresses can be used to manipulate the CLE and ALE pins:

- 40 0000h (40 0000h + 00 0000h): a CPU write or read from this address drives CLE and ALE low.
- 40 2000h (40 0000h + 00 2000h): a CPU write or read from this address drives CLE high and ALE low.
- 40 1000h (40 0000h + 00 1000h): a CPU write or read from this address drives CLE low and ALE high.

Note that in this example the CPU must access EMIF CS2 to initiate accesses with the memory device. As indicated by , EMIF CS2 starts at CPU word address 40 0000h. When using other chip select spaces to interface to NAND, simply replace base address used in this example with that of chip select being used.

1.2.7.6.4 NAND Read and Program Operations

A NAND Flash access cycle is composed of a command, address, and data phase. The EMIF will not automatically generate these three phases to complete a NAND access with one transfer request. To complete a NAND access cycle, multiple single asynchronous access cycles (as described above) must be completed by the EMIF. Software must be used to request the appropriate asynchronous accesses to complete a NAND Flash access cycle. This software must be developed to the specification of the chosen NAND Flash device.

Since NAND operations are divided into single asynchronous access cycles, the chip select signal will not remain activated for the duration of the NAND operation. Instead, the chip select signal will deactivate between each asynchronous access cycle. For this reason, the EMIF does not support NAND Flash devices that require the chip select signal to remain low during the t_R time for a read. See Section 1.2.7.6.9 for workaround.

Care must be taken when performing a NAND read or write operation via the DMA controller. See Section 1.2.7.6.5 for more details.

NOTE: The EMIF does not support NAND Flash devices that require the chip select signal to remain low during the t_R time for a read. See Section 1.2.7.6.9 for workaround.

1.2.7.6.5 NAND Data Read and Write via DMA Controller

When performing NAND accesses, the DMA controller is most efficiently used for the data phase of the access. The command and address phases of the NAND access require only a few words of data to be transferred and therefore do not take advantage of the DMA controller's ability to transfer larger quantities of data with a single request. Furthermore, since the minimum amount of data the DMA controller can transfer is one double-word (32-bits), it cannot be used during NAND command and address phases. In this section we will focus on using the DMA controller for the data phase of a NAND access.

There are two conditions that require care to be taken when performing NAND reads and writes via the DMA controller. These are:

- The address lines used to drive CLE and ALE signals must be driven low.
- The EMIF does not support a constant address mode, but only supports linear incrementing address modes.

Since the EMIF does not support a constant addressing mode, when programming the DMA, a linear incrementing address mode must be used. When using a linear incrementing address mode, care must be taken not to increase the address into a range that drives CLE and/or ALE high. To prevent the address from incrementing into a range that drives CLE and/or ALE high, the DMA start address and transfer size must be carefully considered.

Consider a system in which EM_A[12] is connected to CLE and EM_A[11] is connected to ALE. In this case, per Figure 1-15, the transfer size of the DMA must stay below 8192 bytes (2^13) to avoid driving the CLE and ALE pins to 1. The DMA setup for a NAND Flash data read would look as follows:

- 4 ≤ LENGTH ≤ 8192 bytes
- DSTAMODE = 00b, automatic post increment
- SRCAMODE = 00b, automatic post increment
- SSA (source start address) = CE base address
- DSA (destination start address) = address of internal memory buffer

Similarly, the DMA setup for a NAND Flash data write would be as follows:

- 4 ≤ LENGTH ≤ 8192 bytes
- DSTAMODE = 00b, automatic post increment
- SRCAMODE = 00b, automatic post increment
- SSA (source start address) = address of internal memory buffer
- DSA (destination start address) = CE base address

1.2.7.6.6 1-Bit ECC Generation

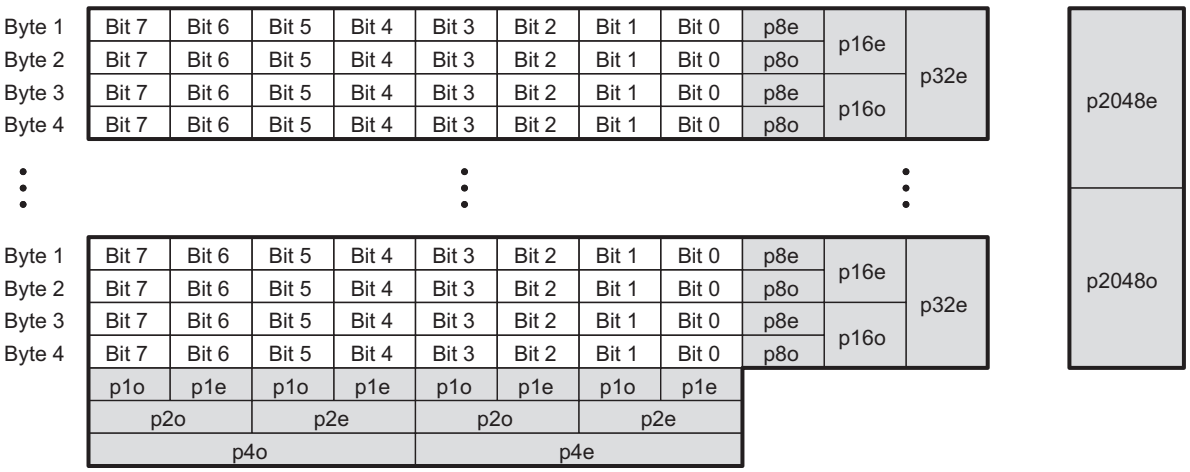
If the CSn_USE_NAND bits in the NAND Flash control register (NANDFCR) are set to 1, the EMIF supports ECC calculation for up to 512 bytes for the corresponding chip select space. To perform the ECC calculation, the corresponding CSn_ECC_START bits in NANDFCR must be set to 1. It is the responsibility of the software to start the ECC calculation prior to issuing a write or read to NAND Flash. It is also the responsibility of the software to read the calculated ECC from the NAND Flash 1-Bit ECC registers once the transfer to NAND Flash has completed. If the software writes or reads more than 512 bytes, the ECC will be incorrect.

Reading a NAND Flash 1-Bit ECC register clears the corresponding CSn_ECC_START bit in NANDFCR. Also, the NAND Flash 1-Bit ECC registers are cleared upon writing a 1 to the corresponding CSn_ECC_START bit in NANDFCR.

Figure 1-15 shows the algorithm used to calculate the ECC value for an 8-bit NAND Flash.

For an 8-bit NAND Flash, p1o through p4e are column parities and p8e through p2048o are row parities. Similarly, the algorithm can be extended to a 16-bit NAND Flash. For a 16-bit NAND Flash p1o through p8e are column parities and p16e through p2048o are row parities. The software must ignore the unwanted parity bits if ECC is desired for less than 512 bytes of data. For example, p2048e and p2048o are not required for ECC on 256 bytes of data. Similarly, p1024e, p1024o, p2048e, and p2048o are not required for ECC on 128 bytes of data.

Figure 1-15. ECC Value for 8-Bit NAND Flash



1.2.7.6.7 4-Bit ECC Generation

The EMIF supports 4-bit ECC for 8-bit and 16-bit NAND Flash. In NAND mode, if the NAND Flash 4-bit ECC start bit (4BIT_ECC_START) in the NAND Flash Control register (NANDFCR) is set, the EMIF calculates 4-bit ECC for the selected chip select. Only one chip select can be selected for the 4-bit ECC calculation at one time. The selection of the chip select is done by programming the 4-bit ECC CS select bit field (4BIT_ECC_SEL) in NANDFCR.

The calculated parity (for writes) and syndrome (for reads) can be read from the NAND Flash 4-Bit ECC registers (NAND4BITECC1[8:1]). The 4-bit ECC start bit (4BIT_ECC_START) is cleared upon reading any of the NAND Flash 4-Bit ECC registers. The NAND Flash 4-Bit ECC registers are cleared upon writing 1 to the 4-bit ECC start bit (4BIT_ECC_START).

The 4-bit ECC algorithm works on a 10-bit data bus. Since the 4-bit ECC is calculated over an 8-bit value for 8- and 16-bit NAND Flash, the EMIF zeroes the upper two bits. However, the parity and the syndrome value read from the NAND Flash 4-Bit ECC registers are 10 bits wide. It is the responsibility of software to convert 10-bit parity values to 8 or 16 bits before writing to the spare location of the NAND Flash after a write operation. Similarly, it is the responsibility of the software to convert the 8- or 16-bit parity values read from the spare location of the NAND Flash after a read operation to 10 bits before writing the NAND Flash 4-Bit ECC Load register (NAND4BITECCLOAD).

At the end of the syndrome calculation after read, the error address and the error value can be calculated by setting the address and error value calculation start bit (ADDR_CALC_ST) in the NAND Flash Control register (NANDFCR). The end of address calculation is flagged by the 4-bit ECC correction state field (CORRSTATE) in the NAND Flash Status register 1 (NANDFSR1). The number of errors can be read from the 4-bit number of errors field (ERRNUM) in the NAND Flash Status register 2 (NANDFSR2). The error address value can be read from the NAND Flash 4-Bit ECC Error Address registers (NANDERRADD[4:1]). The error value can be read from the NAND Flash 4-Bit ECC Error Value registers (NANDERRVAL1[4:1]).

The address and error value start bit (ADDR_CALC_ST) are cleared upon reading any of the NAND Flash 4-Bit ECC Error Address registers or the NAND Flash 4-Bit ECC Error Value registers. The EMIF records the syndrome value internally before the error address and error value calculation. Therefore, a new read operation can be performed simultaneously with the error address calculation.

The EMIF supports 4-bit ECC calculation up to 518 bytes. The software needs to follow the following procedure for 4-bit ECC calculation:

For writes:

1. Set the (4BIT_ECC_START) bit in the NAND Flash Control register (NANDFCR) to 1.
2. Write 518 bytes of data to the NAND Flash.
3. Read the parity from the NAND Flash 4-Bit ECC registers (NAND4BITECC1[8:1]).
4. Convert the 10-bit parity values to 8 or 16 bits depending on the width of the connected NAND Flash memory. All 10-bit parity values can be concatenated together with ECC value 1 (4BIT_ECC_VAL1) as the LSB and ECC value 8 (4BIT_ECC_VAL8) as the MSB. Then the concatenated value can be broken down into ten 8-bit or five 16-bit values.
5. Store the parity to a spare location in the NAND Flash.

For reads:

1. Set the (4BIT_ECC_START) bit in the NAND Flash Control register (NANDFCR) to 1.
2. Read 518 bytes of data from the NAND Flash.
3. Clear the (4BIT_ECC_START) bit in NANDFCR by reading any of the NAND Flash 4-Bit ECC registers.
4. Read the parity stored in the spare location in the NAND Flash.
5. Convert the 8-bit or 16-bit parity values to 10-bits. Reverse of the conversion that was done during writes.
6. Write the parity values in the NAND Flash 4-bit ECC Load register (NAND4BITECCLOAD). Write each parity value one at a time starting from 4BIT_ECC_VAL8 to 4BIT_ECC_VAL1.
7. Perform a dummy read to the EMIF revision register (REV). This is only required to ensure time for syndrome calculation after writing the ECC values in step 6.

8. Read the syndrome from the NAND Flash 4-Bit ECC registers (NAND4BITECC1[8:1]). A syndrome value of 0 means no bit errors. If the syndrome is non-zero continue to step 9.
9. Set the (ADDR_CALC_ST) it in NANDFCR to 1.
10. Start another read from NAND if required (a new thread from step 1).
11. Wait for the 4-bit ECC correction state field (CORRSTATE) in NANDFSR1 to be equal to 1h, 2h, or 3h.
12. Read the number of errors from the 4-bit number of errors field (ERRNUM) in the NANDFSR2.
13. Read the error address from the NAND Flash 4-Bit ECC Error Address registers (NANDERRADD[4:1]). The address for the word in error is equal to: total_words_read + 7 - address_value. For 518 bytes, the address will be equal to: 525 - address_value.
14. Read the error value from NAND Flash 4-Bit ECC Error Value registers (NANDERRVAL[4:1]). The values from these registers can be XORed with the errored word to correct the errors.

NOTE: You must perform step 13 or 14. Otherwise the ADDR_CALC_ST bit will not be cleared and you will not be able to start a new address and value error calculation.

1.2.7.6.8 NAND Flash Status Registers (NANDFSR1 and NANDFSR2)

The NAND Flash status register 1 (NANDFSR1) indicates the raw status of the EM_WAIT[3:0] pins while in NAND Flash Mode. A wait pin should be connected to the NAND Flash device's R/B signal, so that it can indicate whether or not the NAND Flash device is busy.

During a read, the R/B signal will transition and remain low while the NAND Flash retrieves the data requested. Once the R/B signal transitions high, the requested data is ready and should be read by the EMIF. During a write/program operation, the R/B signal transitions and remains low while the NAND Flash is programming the Flash with the data it has received from the EMIF. Once the R/B signal transitions high, the data has been written to the Flash and the next phase of the transaction may be performed.

From this explanation, you can see that the NAND Flash status register is useful to the software for indicating the status of the NAND Flash device and determining when to proceed to the next phase of a NAND Flash operation.

When a rising edge occurs on a wait pin, the EMIF sets the corresponding WR (wait rise) bit in the EMIF Interrupt Raw Register (EIRR). Therefore, the EMIF Wait Rise interrupt may be used to indicate the status of the NAND Flash device. The WPn bits in the Asynchronous Wait Cycle Configuration Register (AWCCR) do not affect the NAND Flash status register (NANDFSR) or the WR bits in EIRR. See Section 1.2.14 for more a detailed description of the wait rise interrupt.

1.2.7.6.9 Interfacing to a Non-CE Don't Care NAND Flash

As explained in Section 1.2.7.6.4, the EMIF does not support NAND Flash devices that require the chip select signal to remain low during the t_R time for a read. One way to work around this limitation is to use a GPIO pin to drive the CE signal of the NAND Flash device. If this work around is implemented, software will configure the selected GPIO to be low, then begin the NAND Flash operation, starting with the command phase. Once the NAND Flash operation has completed the software can then configure the selected GPIO to be high.

1.2.7.7 Extended Wait Mode and the Wait Pins

The EMIF supports the Extend Wait Mode. This is a mode in which the external asynchronous device may assert control over the length of the strobe period. The Extended Wait Mode can be enabled on a per chip select basis by setting the EW bit in the Asynchronous CSn Configuration register 2 (ACSnCR2). When this bit is set, the EMIF monitors the wait pin corresponding to the chip select being accessed to determine if the attached device wishes to extend the strobe period of the current access cycle beyond the programmed number of clock cycles.

If the EMIF detects that the wait pin has been asserted, it will begin inserting extra strobe cycles into the operation until the wait pin is deactivated by the external device. The EMIF will then return to the last cycle of the programmed strobe period and the operation will proceed as usual from this point. Please refer to the device data manual for details on the timing requirements of the wait pins.

The wait pins cannot be used to extend the strobe period indefinitely. The programmable MEWC field in the Asynchronous Wait Cycle Configuration Register 1 (AWCCR1) determines the maximum number of EMIF clock cycles the strobe period may be extended beyond the programmed length. When the counter expires, the EMIF proceeds to the hold period of the operation regardless of the state of the wait pin. The EMIF can also generate an interrupt upon expiration of this counter. See Section 1.2.14 for details on enabling this interrupt.

For the EMIF to function properly in the Extended Wait mode, the wait pin polarity bits (WPn) of AWCCR2 must be programmed to set the polarity of the wait pins. In its reset state of 1, the EMIF will insert wait cycles when the wait pin is sampled high. When set to 0, the EMIF will insert wait cycles only when wait pin is sampled low. This programmability allows for a glueless connection to larger variety of asynchronous devices.

By default, a wait pin is assigned to each chip select space. You can change the default wait pin assignment using the CSn_WAIT bits of AWCCR2. You are allowed to assign a single wait pin to more than one chip select space.

Finally, a restriction is placed on the strobe period timing parameters when operating in Extended Wait mode. Specifically, the W_STROBE and R_STROBE fields must not be set to 0 for proper operation.

1.2.8 BYTEMODE Bits of The EMIF System Control Register

WARNING

The BYTEMODE bits affect CPU program and data accesses to external memory as well as CPU accesses to the EMIF registers. Therefore, to avoid data corruption issues you must always set BYTEMODE = 00b (16-bit access) after you have completed all 8-bit CPU accesses to external memory or EMIF registers. External memory accesses by other modules are not affected by these bits.

The CPU cannot generate 8-bit accesses to its data or I/O space. This presents a problem given that it is often necessary to access a single byte when communicating with NAND Flash devices or, in the case of the EMIF SDRAM self-refresh mode, you must only access a single byte in the SDRAM Configuration Register 2 (SDCR2) to avoid triggering the SDRAM auto-initialization procedure.

For these situations, the BYTEMODE bits of the EMIF System Control Register (ESCR) can be used to program the DSP switched central resource (SCR) such that a CPU word access generates a single byte access when reading or writing from external memory or when accessing the EMIF registers. Table 1-27 summarizes the effect of the BYTEMODE bits for different CPU operations. For more details on ESCR, see the *DSP System User's Guide* chapter.

NOTE: The BYTEMODE bits should only be used for controlling CPU accesses to NAND Flash devices and EMIF registers.

Table 1-27. Effect of BYTEMODE Bits on EMIF Accesses

BYTEMODE Setting	CPU Access to EMIF Register	CPU Access To External Memory
BYTEMODE = 00b (16-bit word access)	Entire register contents are accessed	ASIZE = 01b (16-bit data bus): EMIF generates a single 16-bit access to external memory for every CPU word access. ASIZE = 00b (8-bit data bus): EMIF generates two 8-bit accesses to external memory for every CPU word access.
BYTEMODE = 01b (8-bit access with high byte selected)	Only the upper byte of the register is accessed.	ASIZE = 01b (16-bit data bus): EMIF generates a 16-bit access to external memory for every CPU word access; only the high byte of the EMIF data bus is used. ASIZE = 00b (8-bit data bus): EMIF generates a single 8-bit access to external memory for every CPU word access.

Table 1-27. Effect of BYTEMODE Bits on EMIF Accesses (continued)

BYTEMODE Setting	CPU Access to EMIF Register	CPU Access To External Memory
BYTEMODE = 10b (8-bit access with low byte selected)	Only the lower byte of the register is accessed.	ASIZE = 01b (16-bit data bus): EMIF generates a 16-bit access to external memory for every CPU word access; only the low byte of the EMIF data bus is used. ASIZE = 00b (8-bit data bus): EMIF generates a single 8-bit access to external memory for every CPU word access.

1.2.9 Data Bus Parking

The EMIF always drives the data bus to the previous write data value when it is idle. This feature is called data bus parking. Only when the EMIF issues a read command to the external memory does it stop driving the data bus. After the EMIF latches the last read data, it immediately parks the data bus again.

The one exception to this behavior occurs after performing an asynchronous read operation while the EMIF is in the self-refresh state. In this situation, the read operation is not followed by the EMIF parking the data bus, instead the EMIF tri-states the data bus. Therefore, it is not recommended to perform asynchronous read operations while the EMIF is in the self-refresh state, in order to prevent floating inputs on the data bus. External pull-up or pull-down resistors should be placed on the EMIF data bus pins if it is required to perform reads in this situation. The precise resistor value should be chosen so that the worst case combined off-state leakage currents do not cause the voltage levels on the associated pins to drop below the high-level input voltage requirement.

More information about the self-refresh state can be found in Section 1.2.6.7.

1.2.10 Priority and Arbitration

Section 1.2.2 of this document describes the external prioritization and arbitration among requests from different sources within the DSP. The result of this external arbitration is that only one request is presented to the EMIF at a time. Once the EMIF completes a request, the external arbiter then provides the EMIF with the next pending request.

Internally, the EMIF undertakes memory device transactions according to a strict priority scheme. The highest priority events are:

- A hardware reset (see Section 1.2.12).
- A write to SDRAM configuration register 1 (SDCR1) or the least significant byte of SDRAM configuration register 2 (SDCR2).

Either of these will cause the EMIF to immediately commence its initialization sequence as described in Section 1.2.6.4 .

Once the EMIF has completed its initialization sequence, it performs memory transactions according to the following priority scheme (highest priority listed first):

1. If the EMIF's backlog refresh counter is at the Refresh Must urgency level, the EMIF performs multiple SDRAM auto refresh cycles until the Refresh Release urgency level is reached.
2. If an SDRAM or asynchronous read has been requested, the EMIF performs a read operation.
3. If the EMIF's backlog refresh counter is at the Refresh Need urgency level, the EMIF performs an SDRAM auto refresh cycle.
4. If an SDRAM or asynchronous write has been requested, the EMIF performs a write operation.
5. If the EMIF's backlog refresh counter is at the Refresh May or Refresh Release urgency level, the EMIF performs an SDRAM auto refresh cycle.
6. If the value of the SR bit in SDCR2 has been set to 1, the EMIF will enter the self-refresh state as described in Section 1.2.6.7.

After taking one of the actions listed above, the EMIF then returns to the top of the priority list to determine its next action.

Because the EMIF does not issue auto-refresh cycles when in the self-refresh state, the above priority scheme does not apply when in this state. See Section 1.2.6.7 for details on the operation of the EMIF when in the self-refresh state.

1.2.11 System Considerations

In a system that interfaces to both SDRAM and asynchronous memory, the asynchronous requests must not take longer than the smaller of the following two values:

- t_{RAS} (typically 120us): to avoid violating the maximum time allowed between issuing an ACTV and PRE command to the SDRAM.
- $t_{Refresh\_Rate}$ x 11 (typically 15.7 ms = 11 = 172.7 ms): to avoid refresh violations on the SDRAM.

The length of an asynchronous request is controlled by multiple factors, the primary factor being the number of access cycles required to complete the request. For example, an asynchronous request for 4 bytes will require four access cycles using an 8-bit data bus configuration and only two access cycles using a 16-bit data bus configuration.

The length of the individual access cycles that make up the asynchronous request is determined by the programmed setup, strobe, hold, and turnaround values, but can also be extended with the assertion of the EM_WAIT[3:0] input signals up to a programmed maximum limit. It is up to the user to make sure that an entire asynchronous request does not exceed the timing values listed above when also interfacing to an SDRAM device. This can be done by limiting the asynchronous timing parameters.

1.2.12 Reset Considerations

The EMIF has one reset source: a hardware reset. This reset is always initiated during a full chip reset. Alternatively, software can force a hardware reset on the EMIF through the PG1_RST bit of the peripheral reset control register (PRCR). See the device data manual for more details on PRCR. Please note that the EMIF input clock must be enabled for PG1_RST to have an affect on the EMIF (see Section 1.2.1). Also note that the PG1_RST bit resets other modules in the DSP.

When a hardware reset occurs, the EMIF state machine and registers are reset. Command and data stored in the EMIF memory controller FIFOs is lost.

NOTE: External memory accesses and EMIF register accesses should not be performed while PG1_RST is asserted.

When the EMIF is taken out of reset it automatically begins running the SDRAM initialization sequence described in Section 1.2.6.4. Note that even though the initialization procedure is automatically started, you must still follow the configuration procedure described Section 1.2.6.5.

1.2.13 Initialization

The following initialization procedure describes the basic setup of the EMIF.

1. Perform the necessary device pin multiplexing setup (see Section 1.2.5 for more details).
2. Program the DSP clock generator to provide the desired EM_SDCLK clock frequency. For details on programming the DSP clock generator, see the *DSP System Guide* chapter .
3. Set the EDIV bit of the EMIF clock divider register (ECDR) to select the frequency of EM_SDCLK: half the CPU clock rate or equal to the CPU clock rate. The frequency of EM_SDCLK must meet the timing requirements in the SDRAM manufacturer's documentation and the timing limitations described in electrical specifications of the device data manual. For C5515/14/05/04 device EM_SDCLK limitation, see section Section 1.2.1.
4. Reset the EMIF using the PG1_RST bit of the peripheral reset control register (PRCR). See Section 1.2.12 for more details on this bit.
5. If you want to use interrupts, you can enable them in this step. The EMIF can generate interrupts on three conditions: the rising edge of the wait signals (EM_WAIT[3:0]), asynchronous memory access time-outs, and addressing errors. More information on interrupts is given in Section 1.2.14.
6. If using the EMIF to access SDRAM, follow the steps outlined in Section 1.2.6.5 (including turning on the SDCLK as outline in that section)..
7. If using the EMIF to access asynchronous memories or other devices, configure the EMIF as described in Section 1.2.7.3.

1.2.14 Interrupt Support

The EMIF supports a single interrupt to the CPU. Section 1.2.14.1 details the generation and internal masking of EMIF interrupts, and Section 1.2.14.2 describes how the EMIF interrupts are sent to the CPU.

1.2.14.1 Interrupt Events and Requests

There are three conditions that may cause the EMIF to generate an interrupt to the CPU. These conditions are:

- A rising edge on the EM_WAIT[3:0] signals (wait interrupt).
- An asynchronous memory access time out.
- Usage of unsupported addressing mode (line trap interrupt).

The wait interrupt is not affected by the wait pin polarity configuration bits (WPn) in the asynchronous wait cycle configuration register 2 (AWCCR2). The asynchronous time out interrupt condition occurs when the attached asynchronous device fails to dessert the EM_WAIT pin within the number of cycles defined by the MEWC bits in the asynchronous wait cycle configuration register 1 (AWCCR1).

The EMIF supports only linear incrementing addressing mode. If an access request for an unsupported addressing mode is received, the EMIF will set the LT bit in interrupt raw register and treat the request as a linear incrementing request.

Only when the interrupt is enabled by setting the appropriate bit (WRMASKSET/ATMASKSET/LTMASKSET) in the EMIF interrupt mask set register (EIMSR) to 1, will the interrupt be sent to the CPU. Once enabled, the interrupt may be disabled by writing a 1 to the corresponding bit in the EMIF interrupt mask clear register (EIMCR). The bit fields in both the EIMSR and EIMCR may be used to indicate whether the interrupt is enabled. When the interrupt is enabled, the corresponding bit field in both the EIMSR and EIMCR will have a value of 1; when the interrupt is disabled, the corresponding bit field will have a value of 0.

The EMIF interrupt raw register (EIRR) and the EMIF interrupt mask register (EIMR) indicate the status of each interrupt. The appropriate bit (WR/AT/LT) in EIRR is set when the interrupt condition occurs, whether or not the interrupt has been enabled. However, the appropriate bit (WRMASKED/ATMASKED/LTMASKED) in EIMR is set only when the interrupt condition occurs and the interrupt is enabled. Writing a 1 to the bit in EIRR clears the EIRR bit as well as the corresponding bit in EIMR.

Table 1-28 contains a brief summary of the interrupt status and control bit fields. See Section 1.4 for complete details on the register fields.

Table 1-28. Interrupt Monitor and Control Bit Fields

Register Name	Bit Name	Description
EMIF interrupt raw register (EIRR)	WR[3:0]	These bits are set when an rising edge on the wait signals (EM_WAIT[3:0]) occurs. Writing a 1 clears the WR bits as well as the WRMASKED bits in EIMR. Each WR bit corresponds to an EM_WAIT pin.
	AT	This bit is set when an asynchronous timeout occurs. Writing a 1 clears the AT bit as well as the ATMASKED bit in EIMR.
	LT	This bit is set when an unsupported addressing mode is used. Writing a 1 clears LT bit as well as the LTMASKED bit in EIMR.
EMIF interrupt mask register (EIMR)	WRMASKED[3:0]	These bits are set only when a rising edge on the wait signals (EM_WAIT[3:0]) occurs and the interrupt has been enabled by writing a 1 to the WRMASKSET bits in EIMSR. Each WRMASKED bit corresponds to an EM_WAIT pin.
	ATMASKED	This bit is set only when an asynchronous timeout occurs and the interrupt has been enabled by writing a 1 to the ATMASKSET bit in EIMSR.
	LTMASKED	This bit is set only when line trap interrupt occurs and the interrupt has been enabled by writing a 1 to the LTMASKSET bit in EIMSR.
EMIF interrupt mask set register (EIMSR)	WRMASKSET[3:0]	Writing a 1 to these bits enables the wait rise interrupt of the corresponding wait pin. Each WRMASKSET bit corresponds to an EM_WAIT pin.
	ATMASKSET	Writing a 1 to this bit enables the asynchronous timeout interrupt.
	LTMASKSET	Writing a 1 to this bit enables the line trap interrupt.
EMIF interrupt mask clear register (EIMCR)	WRMASKCLR[3:0]	Writing a 1 to these bits disables the wait rise interrupt of the corresponding wait pin. Each WRMASKCLR bit corresponds to an EM_WAIT pin.
	ATMASKCLR	Writing a 1 to this bit disables the asynchronous timeout interrupt.
	LTMASKCLR	Writing a 1 to this bit disables the line trap interrupt.

1.2.14.2 Interrupt Multiplexing

The EMIF interrupt to the DSP CPU is not multiplexed with any other interrupt source.

1.2.15 DMA Event Support

The EMIF does not generate any DMA events.

1.2.16 Power Management

There are several ways to reduce the power consumption of the device EMIF. First, the EMIF power domain voltage can be selected during the board design phase based on the requirements of the devices connected to the EMIF. Running at lower voltages consumes less power. The lowest voltage allowable by the memory device should be used to minimize power consumption.

Second, there are several clock options. the input clock of the EMIF can be turned off by using the peripheral clock gating configuration register (PCGCR). For detailed information on PCGCR see the device-specific data manual.

- The input clock of the EMIF can be turned off by using the peripheral clock gating configuration register (PCGCR). For detailed information on PCGCR, see the device-specific data manual.
- SDCLK can be turned off if SDRAM is not used by using the Clock Configuration Register 1 (CCR1). For detail information on CCR1, see the device-specific data manual.
- EMIF clock divider should be used if SDRAM operates at less than half the CPU frequency. The EMIF clock divider divides the EMIF clock by two for SDCLK. This is set using the EMIF Clock Divider Register (ECDR). For detail information on ECDR, see the device-specific data manual and Section 1.2.1 for C5515/14/05/04 device EM_SDCLK limitation.

Third, when connected to SDRAM, the EMIF can enter a self-refresh mode. While in the self-refresh state, the EMIF continues to service asynchronous bank requests and register accesses as normal. Self-refresh mode must be used with care as bus parking is disabled following a read to asynchronous memory while in this mode. This could lead to floating inputs which would lead to higher power consumption. For more information on the EMIF self-refresh mode please see Section 1.2.6.7.

Lastly, the EMIF can be requested to issue a power down command to an SDRAM device. When placed in the power down mode the memory device deactivates its input and output buffers, excluding CKE, to maximize power savings. While in the power down state, the EMIF services the SDRAM, asynchronous memory, and register accesses as normal, returning to the power down state upon completion. For more information on the EMIF power down mode, see Section 1.2.6.8 and Section 1.2.1 for C5515/14/05/04 device EM_SDCLK limitations.

1.2.17 Emulation Considerations

The EMIF will remain fully functional during emulation halts to allow emulation access to external memory.

1.2.18 CPU Instruction Pipeline Considerations

This section explains two special cases of pipeline operation that could impact external memory accesses.

As described in the *TMS320C55x DSP CPU Reference Guide* (SPRU371), the CPU uses instruction pipelining. Multiple instructions are processed simultaneously in the pipeline, and different instructions may access external memory during different phases of completion. The pipeline consists of a number of phases during which different, designated tasks are performed.

1.2.18.1 A Write Followed by a Read at a Different Address

The read phase of the pipeline is used to read operands and other data needed to complete the execution of an instruction. The results of an instruction may be written to external memory in the write phase of the pipeline. The read phase occurs earlier in the pipeline than the write phase. For this reason, the read and write requests made to the EMIF may occur in an order which is different than the order in which the instructions entered the pipeline. For example, consider the following code segment:

```
I1: MOV T0, *(#External_Address_1) ; Instruction 1 writes to external memory
I2: MOV *(#External_Address_2), T1 ; Instruction 2 reads from external memory
```

Although the code shows the write followed by the read, the read actually occurs first on the EMIF because of the pipelining effect, as shown in Table 1-29. The figure shows the two instructions passing through the read (R), execute (X), and write (W) phases.

For some applications, maintaining the proper write-read order is critical. In such cases, NOP (no operation) instructions (or other instructions that do not perform external memory accesses) can be inserted between the original instructions to delay the read operation. This technique is shown in Table 1-30.

Table 1-29. Partial Pipeline Diagram of Consecutive Instructions That Write and Read at Different Addresses

R	X	W	Cycle	Commnet
I1			n	
I2	I1		n+1	Read initiated by instruction 2
	I2	I1	n+2	Write initiated by instruction 1
		I2	n+3	

Table 1-30. NOP Instructions Inserted in the Code of Figure 1−3 to Make the Write Occur Before the Read

R	X	W	Cycle	Commnet
I1			n	
NOP	I1		n+1	

Table 1-30. NOP Instructions Inserted in the Code of Figure 1−3 to Make the Write Occur Before the Read (continued)

R	X	W	Cycle	Commnet
NOP	NOP	I1	n+2	Write initiated by instruction 1
I2	NOP	NOP	n+3	Read initiated by instruction 2
	I2	NOP	n+4	
		I2	n+5	

1.2.18.2 A Write Followed by a Read at the Same Address

In most cases, when a write to memory is followed immediately by a read at the same address, the data written is the same data expected back during the read. The C55x CPU takes advantage of this fact with a special memory-bypass feature. During the read, the CPU gets a copy of the data directly from the write bus(es) instead of accessing memory.

When the CPU is accessing external devices, there may be cases in which the memory-bypass feature would lead to unwanted results. For example, suppose two physical memory locations X and Y are mapped to the same address. Writing modifies location X, and reading gets data from location Y. If the memory-bypass feature takes effect, location Y is not read.

To prevent the memory bypass, insert three or more NOP instructions (or other instructions) between the instructions that perform the write and the read. For example:

```
MOVT0,*(#External_Address_1)  ; Write to address 1
NOP                           ; 3-cycle delay
NOP
NOP
MOV*(#External_Address_1), T1 ; Read from address 1
```

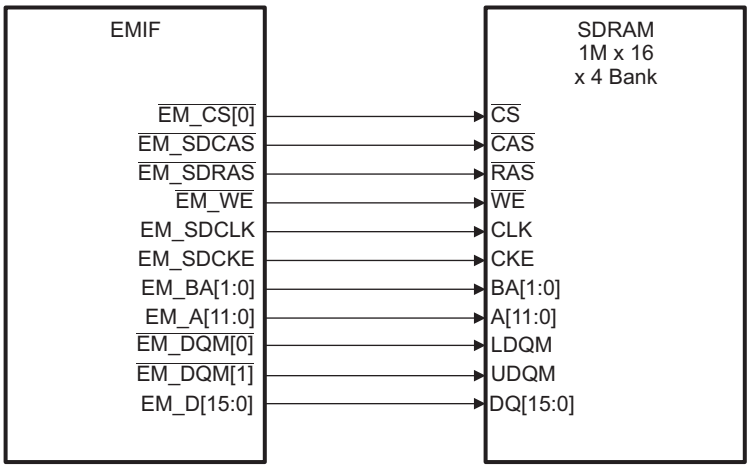
1.3 Interfacing the EMIF to Mobile SDRAM

This section presents an example of interfacing the EMIF to a mobile SDRAM device. The 16Mbit mobile SDRAM Micron MT48H4M16LF-8 device is used for this example.

1.3.1 Hardware Interface

Figure 1-16 shows the hardware interface between the EMIF and the 16Mbit mobile SDRAM Micron MT48H4M16LF-8 device.

Figure 1-16. Hardware Interface to Mobile SDRAM Device



1.3.2 SW Configuration

This section describes how to configure the EMIF registers to interface with the Micron MT48H4M16LF-8 device. The SDRAM clock frequency is assumed to be 100 MHz (f_{EM_SDCLK} = 100 MHz).

1.3.2.1 Settings for SDRAM Timing Registers (SDTIMR1 and SDTIMR2)

The fields of SDTIMR1 and SDTIMR2 control various timing parameters used by the EMIF. Table 1-31 shows the values that should be used when interfacing to the Micron MT48H4M16LF device. These values were derived using timing specifications obtained from the memory's datasheet. Figure 1-17 and Figure 1-18 show a graphical representation of the contents of SDTIMR1 and SDTIMR2 after being programmed with the derived values.

Table 1-31. SDTIMR1 and SDTIMR2 Field Calculations

Field Name	Formula	Value from MT48H4M16LF-8 Datasheet	Value Calculated for Field
T_RFC	T_RFC ≥ (t_{RFC}*f_{EM_SDCLK}) - 1	t_{RFC} = 80 ns (min)	7
T_RP	T_RP ≥ (t_{RP}*f_{EM_SDCLK}) – 1	t_{RP} = 19 ns (min)	1
T_RCD	T_RCD ≥ (t_{RCD}*f_{EM_SDCLK}) – 1	t_{RCD} = 19 ns (min)	1
T_WR	T_WR ≥ (t_{WR}*f_{EM_SDCLK}) – 1	t_{RDL} **(1)** = 2 clocks = 20ns (min)	1
T_RAS	T_RAS ≥ (t_{RAS}*f_{EM_SDCLK}) - 1	t_{RAS} = 48 ns (min)	4
T_RC	T_RC ≥ (t_{RC}*f_{EM_SDCLK}) - 1	t_{RFC} = 80 ns (min)	7
T_RRD	T_RRD ≥ (t_{RRD}*f_{EM_SDCLK}) - 1	t_{RRD} = 16 ns (min)	1

(1) The Micron MT48H4M16LF data sheet does not specify a t_{WR}, instead is specifies t_{RDL} as the last data in to PRECHARGE command.

Figure 1-17. SDTIMR1 Contents

15–12	11–8	7	6–4	3–0
0100	0111	0	001	0000
TRAS	TRC	Reserved	TRRD	Reserved

Figure 1-18. SDTIMR2 Contents

15–11	10–8	7	6–4	3	2–0
0 0111	001	0	001	0	001
TRFC	TRP	Reserved	TRCD	Reserved	TWR

1.3.2.2 Settings for SDRAM Self Refresh Exit Timing Register (SDSRETR)

The TXS field of SDSRETR should be programmed satisfy the t_{XSR} timing requirement of the memory. Table 1-32 shows the calculation of the value for TXS and Figure 1-19 shows a graphical representation of SDSRETR after being programmed with this value.

Table 1-32. SDSRETR Field Calculations

Field Name	Formula	Value from MT48H4M16LF-8 Datasheet	Value Calculated for Field
TXS	TXS ≥ (t_{XSR}*f_{EM_SDCLK}) - 1	t_{XSR} = 80 ns (min)	7

Figure 1-19. SDRETR Contents

15–5	4–0
0	0 0111
Reserved	TXS

1.3.2.3 Settings for SDRAM Refresh Control Register (SDRCR)

SDRCR should next be programmed to satisfy the required refresh rate of the memory device. Table 1-33 shows the calculation of the proper value to program into the REFRATE field of this register. Based on this calculation, a value of 61Ah should be written to SDRCR. Figure 1-20 shows a graphical representation of the value that should be programmed into SDRCR.

Table 1-33. SDRCR Field Calculations

Field Name	Formula	Value From MT48H4M16LF-8 Data Sheet	Value Calculated for Field
REFRATE	REFRATE ≤ f_{EM_SDCLK} * $t_{Refresh\_Period}$ / ncycles	$t_{Refresh\_Period}$ = 64 ms; ncycles = 4096	1562 = 61Ah

Figure 1-20. SDRCR Contents

15 – 13	12 – 0
0	0 0110 0001 1010 (61Ah)
Reserved	REFRATE

1.3.2.4 Settings for SDRAM Configuration Registers (SDCR1 and SDCR2)

The fields of SDCR1 and SDCR2 should be programmed as described in Table 1-34 to properly interface with the memory device. Based on these settings, a value of 4720h should be written to SDCR1 and a value of 0001h should be written to SDCR2. Figure 1-21 and Figure 1-22 show a graphical representation of the values that should be programmed into SDCR1 and SDCR2.

Table 1-34. SDCR1 and SDCR2 Field Calculations

Field Name	Value	Purpose
SR	0	To avoid placing the EMIF in Self-Refresh Mode.
PD	0	To avoid placing the EMIF in Power Down Mode.
PDWR	0	To disable refreshes during power down.
PASR	0	To select 4 banks for refresh during SLFR command.
ROWSIZE	Don't care	This field is only used when IBANK_POS = 1.
IBANK_POS	0	To configure the EMIF to access the page in all banks before moving on to the next page. You can set this bit to 1 if you want the EMIF to access all the pages in a single bank before moving on to the next page.
SDRAM_DRIVE	0	To select full drive strength when initializing the memory device.
BIT_9_1_UNLOCK	1	To allow writes to PASR, ROWSIZE, IBANK_POS, and SDRAM_DRIVE.
NM	1	This bit should always be set to 1.
CL	3	To select a CAS latency of 3.
BIT_11_9_LOCK	1	To allow the CL field to be written.
IBANK	2	To select 4 internal SDRAM banks.
EBANK	0	To use a single chip select when communicating with the memory device.
PAGESIZE	0	To select a page size of 256 words.

Figure 1-21. SDCR1 Contents

15	14	13 – 12	11 – 9	8
0	1	0	011	1
Reserved	NM	Reserved	CL	BIT_11_9_LOCK

7	6 – 4	3	2 – 0
0	010	0	000
Reserved	IBANK	EBANK	PAGESIZE

Figure 1-22. SDCR2 Contents

15	14	13	12	10	9	8
0	0	0	000		000	
SR	PD	PDWR	Reserved		PASR	

7	6	4	3	2	1	0
0	000		0	00		1
PASR	ROWSIZE		IBANK_POS	SDRAM_DRIVE		BIT_9_1_LOCK

1.3.2.5 Software Sequence for Programming EMIF Registers

There are several steps involved in setting up the EMIF for SDRAM accesses. These steps are described in Section 1.2.6.4 (steps applicable for all types of memories) and Section 1.2.6.5 (steps specific for SDRAM accesses). For reference, Table 1-35 summarizes the values that should be written to the EMIF registers when interfacing to a 16 Mbit mobile SDRAM Micron MT48H4M16LF-8 device using an EM_SDCLK frequency of 100 MHz.

Table 1-35. EMIF Register Values for Micron MT48H4M16LF-8 Mobile SDRAM Device

Register	Value
SDTIMR1	4710h
SDTIMR2	3911h
SDRETR	0007h
SDRCR	061Ah
SDCR1	4720h
SDCR2	0001h
CCR1	0001h

1.4 Registers

Table 1-36 list the registers associated with the device EMIF. The EMIF registers can be accessed by the CPU at the 16-bit addresses specified in Table 1-36. Note that the CPU accesses all peripheral registers through its I/O space. All other register addresses not listed in Table 1-36 should be considered as reserved locations and the register contents should not be modified.

Two additional registers—the EMIF clock divider register and the EMIF system control clock register—affect the operation of the EMIF, see Table 1-37. These registers are not part of the EMIF module; they are part of the DSP system. For more information on these registers, see the *DSP System Guide* chapter.

WARNING

All EMIF registers, except SDCR2, support only 16-bit accesses; performing byte accesses to these registers results in undefined behavior. SDCR2 is byte writable to allow the setting of the SR bit without triggering the SDRAM initialization sequence. The CPU cannot generate 8-bit accesses to its data or I/O space. However, you can program the BYTEMODE bits of ESCR such that CPU writes affect only the upper or lower byte of SDCR2. The BYTEMODE bits should be set to 00b (16-bit access) when accessing any register other than SDCR2. See Section 1.2.8 for more details on the BYTEMODE bits.

Table 1-36. EMIF Registers

CPU Word Address	Acronym	Register Description	Section
1000h	REV	Revision Register	Section 1.4.1
1001h	STATUS	Status Register	Section 1.4.2
1004h	AWCCR1	Asynchronous Wait Cycle Configuration Register 1	Section 1.4.3
1005h	AWCCR2	Asynchronous Wait Cycle Configuration Register 2	Section 1.4.3
1008h	SDCR1	SDRAM Configuration Register 1	Section 1.4.4
1009h	SDCR2	SDRAM Configuration Register 2	Section 1.4.4
100Ch	SDRCR	SDRAM Refresh Control Register	Section 1.4.5
1010h	ACS2CR1	Asynchronous CS2 Configuration Register 1	Section 1.4.6
1011h	ACS2CR2	Asynchronous CS2 Configuration Register 2	Section 1.4.6
1014h	ACS3CR1	Asynchronous CS3 Configuration Register 1	Section 1.4.6
1015h	ACS3CR2	Asynchronous CS3 Configuration Register 2	Section 1.4.6
1018h	ACS4CR1	Asynchronous CS4 Configuration Register 1	Section 1.4.6
1019h	ACS4CR2	Asynchronous CS4 Configuration Register 2	Section 1.4.6
101Ch	ACS5CR1	Asynchronous CS5 Configuration Register 1	Section 1.4.6
101Dh	ACS5CR2	Asynchronous CS5 Configuration Register 2	Section 1.4.6
1020h	SDTIMR1	SDRAM Timing Register 1	Section 1.4.7
1021h	SDTIMR2	SDRAM Timing Register 2	Section 1.4.7
103Ch	SDSRETR	SDRAM Self Refresh Exit Timing Register	Section 1.4.8
1040h	EIRR	EMIF Interrupt Raw Register	Section 1.4.9
1044h	EIMR	EMIF Interrupt Mask Register	Section 1.4.10
1048h	EIMSR	EMIF Interrupt Mask Set Register	Section 1.4.11
104Ch	EIMCR	EMIF Interrupt Mask Clear Register	Section 1.4.12
1060h	NANDFCR	NAND Flash Control Register	Section 1.4.13

Table 1-36. EMIF Registers (continued)

CPU Word Address	Acronym	Register Description	Section
1064h	NANDFSR1	NAND Flash Status Register 1	Section 1.4.14
1065h	NANDFSR2	NAND Flash Status Register 2	Section 1.4.14
1068h	PGMODECTRL1	Page Mode Control Register 1	Section 1.4.15
1069h	PGMODECTRL2	Page Mode Control Register 2	Section 1.4.15
1070h	NCS2ECC1	NAND Flash CS2 1-Bit ECC Register 1	Section 1.4.16
1071h	NCS2ECC2	NAND Flash CS2 1-Bit ECC Register 2	Section 1.4.16
1074h	NCS3ECC1	NAND Flash CS3 1-Bit ECC Register 1	Section 1.4.16
1075h	NCS3ECC2	NAND Flash CS3 1-Bit ECC Register 2	Section 1.4.16
1078h	NCS4ECC1	NAND Flash CS4 1-Bit ECC Register 1	Section 1.4.16
1079h	NCS4ECC2	NAND Flash CS4 1-Bit ECC Register 2	Section 1.4.16
107Ch	NCS5ECC1	NAND Flash CS5 1-Bit ECC Register 1	Section 1.4.16
107Dh	NCS5ECC2	NAND Flash CS5 1-Bit ECC Register 2	Section 1.4.16
10BCh	NAND4BITECCLOAD	NAND Flash 4-Bit ECC Load Register	Section 1.4.17
10C0h	NAND4BITECC1	NAND Flash 4-Bit ECC Register 1	Section 1.4.18
10C1h	NAND4BITECC2	NAND Flash 4-Bit ECC Register 2	Section 1.4.19
10C4h	NAND4BITECC3	NAND Flash 4-Bit ECC Register 3	Section 1.4.20
10C5h	NAND4BITECC4	NAND Flash 4-Bit ECC Register 4	Section 1.4.21
10C8h	NAND4BITECC5	NAND Flash 4-Bit ECC Register 5	Section 1.4.22
10C9h	NAND4BITECC6	NAND Flash 4-Bit ECC Register 6	Section 1.4.23
10CCh	NAND4BITECC7	NAND Flash 4-Bit ECC Register 7	Section 1.4.24
10CDh	NAND4BITECC8	NAND Flash 4-Bit ECC Register 8	Section 1.4.25
10D0h	NANDERRADD1	NAND Flash 4-Bit ECC Error Address Register 1	Section 1.4.26
10D1h	NANDERRADD2	NAND Flash 4-Bit ECC Error Address Register 2	Section 1.4.27
10D4h	NANDERRADD3	NAND Flash 4-Bit ECC Error Address Register 3	Section 1.4.28
10D5h	NANDERRADD4	NAND Flash 4-Bit ECC Error Address Register 4	Section 1.4.29
10D8h	NANDERRVAL1	NAND Flash 4-Bit ECC Error Value Register 1	Section 1.4.30
10D9h	NANDERRVAL2	NAND Flash 4-Bit ECC Error Value Register 2	Section 1.4.31
10DCh	NANDERRVAL3	NAND Flash 4-Bit ECC Error Value Register 3	Section 1.4.32
10DDh	NANDERRVAL4	NAND Flash 4-Bit ECC Error Value Register 4	Section 1.4.33

Table 1-37. EMIF System Registers

CPU Word Address	Acronym	Register Description
1C26h	ECDR (1)	EMIF Clock Divider Register
1C33h	ESCR (1)	EMIF System Control Register
1C1Eh	CCR1 (1)	SD Clock ON/OFF

(1) For more details on this register, see the *DSP System Guide* chapter.

1.4.1 Revision Register (REV)

This is a read only register indicating the major and minor revision of the EMIF. The revision register (REV) is shown in Figure 1-23 and described in Table 1-38.

Figure 1-23. Revision Register (REV)

15	8	7	0
MAJOR_REVISION		MINOR_REVISION	
R-02h		R-05h	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-38. Revision Register (REV) Field Descriptions

Bit	Field	Value	Description
15-8	MAJOR_REVISION	0-FFh	Major revision code for EMIF.
7-0	MINOR_REVISION	0-FFh	Minor revision code for EMIF.

1.4.2 Status Register (STATUS)

This is a read only register showing the module ID and clock rate configuration of the EMIF. The status register (STATUS) is shown in Figure 1-24 and described in Table 1-39.

Figure 1-24. Status Register (STATUS)

15	14	13	8
BE	FULLRATE	MODULE_ID	
R-1	R-1	R-0	

7	0
MODULE_ID	
R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-39. Status Register (STATUS) Field Descriptions

Bit	Field	Value	Description
15	BE		Big Endian. Reflects the endianess of the EMIF. This can not be changed.
		0	EMIF is in little endian mode
		1	EMIF is in big endian mode
14	FULLRATE		Full rate status bit. This bit reflects the clock rate being used for the EMIF SDRAM clock.
		0	EM_SDCLK is running at half the CPU clock frequency.
		1	EM_SDCLK is running at the same speed as the CPU clock.
13-0	MODULE_ID	0-3FFFh	Module ID code for EMIF.

1.4.3 Asynchronous Wait Cycle Configuration Registers (AWCCR1 and AWCCR2)

The asynchronous wait cycle configuration registers (AWCCR1 and AWCCR2) are used to configure the parameters for extended wait cycles. Both the polarity of the wait pins (EM_WAIT[3:0]) and the maximum allowable number of extended wait cycles can be configured through these registers. AWCCR1 and AWCCR2 are shown in Figure 1-25 and Figure 1-26 and described in Table 1-40 and Table 1-41.

Figure 1-25. Asynchronous Wait Cycle Configuration Register 1 (AWCCR1)

15	8	7	0
Reserved		MEWC	
R-0		RW-40h	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-26. Asynchronous Wait Cycle Configuration Register 2 (AWCCR2)

15	14	13	12	11	8
WP3	WP2	WP1	WP0	Reserved	
RW-1	RW-1	RW-1	RW-1	R-0	

7	6	5	4	3	2	1	0
CS5_WAIT		CS4_WAIT		CS3_WAIT		CS2_WAIT	
RW-3		RW-2		RW-1		RW-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-40. Asynchronous Wait Cycle Configuration Register 1 (AWCCR1) Field Descriptions

Bit	Field	Value	Description
15-8	Reserved	0	Reserved.
7-0	MEWC	0-FFh	Maximum extended wait cycles. The EMIF will wait for a maximum of (MEWC + 1) x 16 clock cycles before it stops inserting asynchronous wait cycles and proceeds to the hold period of the access. This setting applies to all of the wait pins.

Table 1-41. Asynchronous Wait Cycle Configuration Register 2 (AWCCR2) Field Descriptions

Bit	Field	Value	Description
15	WP3		EM_WAIT3 polarity bit. This bit defines the polarity of the EM_WAIT3 pin.
		0	Insert wait cycles if EM_WAIT3 is low.
		1	Insert wait cycles if EM_WAIT3 is high.
14	WP2		EM_WAIT2 polarity bit. This bit defines the polarity of the EM_WAIT2 pin.
		0	Insert wait cycles if EM_WAIT2 is low.
		1	Insert wait cycles if EM_WAIT2 is high.
13	WP1		EM_WAIT1 polarity bit. This bit defines the polarity of the EM_WAIT1 pin.
		0	Insert wait cycles if EM_WAIT1 is low.
		1	Insert wait cycles if EM_WAIT1 is high.
12	WP0		EM_WAIT0 polarity bit. This bit defines the polarity of the EM_WAIT0 pin.
		0	Insert wait cycles if EM_WAIT0 is low.
		1	Insert wait cycles if EM_WAIT0 is high.
11-8	Reserved	0	Reserved
7-6	CS5_WAIT		Wait pin mapping bits for EMIF CS5. By default, each asynchronous chip select space is assigned a wait input pin. You can use the wait pin mapping bits to change the default assignment.
		0	Use the EM_WAIT0 pin.
		1h	Use the EM_WAIT1 pin.
		2h	Use the EM_WAIT2 pin.
		3h	Use the EM_WAIT3 pin.

Table 1-41. Asynchronous Wait Cycle Configuration Register 2 (AWCCR2) Field Descriptions (continued)

Bit	Field	Value	Description
5-4	CS4_WAIT		Wait pin mapping bits for EMIF CS4. By default, each asynchronous chip select space is assigned a wait input pin. You can use the wait pin mapping bits to change the default assignment.
		0	Use the EM_WAIT0 pin.
		1h	Use the EM_WAIT1 pin.
		2h	Use the EM_WAIT2 pin.
		3h	Use the EM_WAIT3 pin.
3-2	CS3_WAIT		Wait pin mapping bits for EMIF CS3. By default, each asynchronous chip select space is assigned a wait input pin. You can use the wait pin mapping bits to change the default assignment.
		0	Use the EM_WAIT0 pin.
		1h	Use the EM_WAIT1 pin.
		2h	Use the EM_WAIT2 pin.
		3h	Use the EM_WAIT3 pin.
1-0	CS2_WAIT		Wait pin mapping bits for EMIF CS2. By default, each asynchronous chip select space is assigned a wait input pin. You can use the wait pin mapping bits to change the default assignment.
		0	Use the EM_WAIT0 pin.
		1h	Use the EM_WAIT1 pin.
		2h	Use the EM_WAIT2 pin.
		3h	Use the EM_WAIT3 pin.

1.4.4 SDRAM Configuration Registers (SDCR1 and SDCR2)

The SDRAM configuration registers (SDCR1 and SDCR2) are used to configure various parameters of the SDRAM controller such as the number of internal banks, the internal page size, and the CAS latency to match those of the attached SDRAM device. In addition, these registers are used to put the attached SDRAM device into self-refresh mode. SDCR1 and SDCR2 are shown in Figure 1-27 and Figure 1-28 and described in Table 1-42 and Table 1-43.

Figure 1-27. SDRAM Configuration Register 1 (SDCR1)

15	14	13 12	11 9	8
Reserved	NM	Reserved	CL	BIT_11_9_LOCK
R-0	RW-0	R-0	RW-3	RW-0

7	6 4	3	2 0
Reserved	IBANK	EBANK	PAGESIZE
R-0	RW-2	RW-0	RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-28. SDRAM Configuration Register 2 (SDCR2)

15	14	13	12 10	9 8
SR	PD	PDWR	Reserved	PASR
RW-0	RW-0	RW-0	R-0	RW-0

7	6 4	3	2	1	0
PASR	ROWSIZE	IBANK_POS	SDRAM_DRIVE		BIT_9_1_LOCK
RW-0	RW-0	RW-0	RW-0		RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-42. SDRAM Configuration Register 1 (SDCR1) Field Descriptions

Bit	Field	Value	Description
15	Reserved	0	Reserved.
14	NM		Narrow mode bit. This bit should always be set to 1. A write to this field will cause the EMIF to start the SDRAM initialization sequence.
13-12	Reserved	0	Reserved. The reserved bit location is always read as 0. If writing to this field, always write the default value of 0.
11-9	CL		CAS latency. This field defines the CAS latency to be used when accessing connected SDRAM devices. A 1 must be simultaneously written to the BIT_11_9_LOCK bit field of this register in order to write to the CL bit field. Writing to this field triggers the SDRAM initialization sequence.
		0	Reserved.
		1h	Reserved.
		2h	CAS latency set to 2 EM_SDCLK cycles.
		3h	CAS latency set to 3 EM_SDCLK cycles.
		4h	Reserved.
		5h	Reserved.
		6h	Reserved.
		7h	Reserved.
8	BIT_11_9_LOCK		Bits 11 to 9 lock. CL can only be written if BIT_11_9_LOCK is simultaneously written with a 1. BIT_11_9_LOCK is always read as 0.
		0	Writes to bits 11 through 9 are disabled.
		1	Writes to bits 11 through 9 are enabled.
7	Reserved	0	Reserved.

Table 1-42. SDRAM Configuration Register 1 (SDCR1) Field Descriptions (continued)

Bit	Field	Value	Description
6-4	IBANK		Internal SDRAM Bank size. This field defines number of banks inside the connected SDRAM devices. Writing to this field triggers the SDRAM initialization sequence.
		0	1 bank SDRAM devices.
		1h	2 bank SDRAM devices.
		2h	4 bank SDRAM devices.
		3h	Reserved.
		4h	Reserved.
		5h	Reserved.
		6h	Reserved.
		7h	Reserved.
3	EBANK		External chip select setup. Defines whether SDRAM accesses will use 1 or 2 chip select lines. A write to this field will cause the EMIF to start the SDRAM initialization sequence.
		0	EMIF will use $\overline{\text{EM\_CS0}}$ for all SDRAM accesses.
		1	EMIF will use $\overline{\text{EM\_CS0}}$ and $\overline{\text{EM\_CS1}}$ for all SDRAM accesses.
2-0	PAGESIZE		Page size. This field defines the internal page size of connected SDRAM devices. Writing to this field triggers the SDRAM initialization sequence.
		0	8 column address bits (256 elements per row)
		1h	9 column address bits (512 elements per row)
		2h	10 column address bits (1024 elements per row)
		3h	11 column address bits (2048 elements per row)
		4h	Reserved.
		5h	Reserved.
		6h	Reserved.
		7h	Reserved.

Table 1-43. SDRAM Configuration Register 2 (SDCR2) Field Descriptions

Bit	Field	Value	Description
15	SR		Self-Refresh mode bit. This bit controls entering and exiting of the Self-Refresh mode. The field should be written using a byte-write to the upper byte of SDCR to avoid triggering the SDRAM initialization sequence.
		0	Writing a 0 to this bit will cause connected SDRAM devices and the EMIF to exit the self-refresh mode.
		1	Writing a 1 to this bit will cause connected SDRAM devices and the EMIF to enter the self-refresh mode.
14	PD		Power down bit. This bit controls entering and exiting of the power down mode. The field should be written using a byte-write to the upper byte of SDCR to avoid triggering the SDRAM initialization sequence. If both SR and PD bits are set, the EMIF will go into Self Refresh.
		0	Writing a 0 to this bit will cause connected SDRAM devices and the EMIF to exit the power down mode.
		1	Writing a 1 to this bit will cause connected SDRAM devices and the EMIF to enter the power down mode.
13	PDWR		Perform refreshes during power down bit. Writing a 1 to this bit will cause EMIF to exit power down state and issue and AUTO REFRESH command every time Refresh May level is set.
		0	Do not perform refreshes during power down.
		1	Perform refreshes during power down.
12-10	Reserved	0	Reserved.

Table 1-43. SDRAM Configuration Register 2 (SDCR2) Field Descriptions (continued)

Bit	Field	Value	Description
9-7	PASR		Partial array self refresh. These bits get loaded into the Extended Mode Register of a mobile SDR during initialization. A write to this field will cause the EMIF to start the SDRAM initialization sequence. To write to this field you must simultaneously set BIT_9_1_LOCK to 1.
		0	4 banks will be refreshed.
		1h	2 banks will be refreshed.
		2h	1 bank will be refreshed.
		3h	Reserved.
		4h	Reserved.
		5h	Half a bank will be refreshed.
		6h	One quarter of a bank will be refreshed.
		7h	Reserved.
6-4	ROWSIZE		Row size. Defines the number of row address bits of connected SDRAM devices. This field is only used when IBANK_POS is set to 1. A write to this field will cause the EMIF to start the SDRAM initialization sequence. To write to this field you must simultaneously set BIT_9_1_LOCK to 1.
		0	9 row address bits.
		1h	10 row address bits.
		2h	11 row address bits.
		3h	12 row address bits.
		4h	13 row address bits.
		5h	14 row address bits.
		6h	Reserved.
		7h	Reserved.
3	IBANK_POS	0-1h	Internal bank position. This bit specifies the mapping of the logical address to the SDRAM address as described in Section 1.2.6.11. Mapping of Logical Address to SDRAM address. A write to this field will cause the EMIF to start the SDRAM initialization sequence. To write to this field you must simultaneously set BIT_9_1_LOCK to 1. Setting this bit to 0 is recommended.
2-1	SDRAM_DRIVE		SDRAM drive strength. These bits specify the SDRAM drive strength configuration the EMIF uses while initializing the SDRAM device. A write to this field will cause the EMIF to start the SDRAM initialization sequence. To write to this field you must simultaneously set BIT_9_1_LOCK to 1.
		0	Full drive strength.
		1h	Half drive strength.
		2h	One fourth drive strength.
		3h	One eighth drive strength.
0	BIT_9_1_LOCK		Bits 9 to 1 lock. Bits 9 through 1 can only be written if BIT_9_1_LOCK is simultaneously written with a 1. BIT_9_1_LOCK is always read as a 0.
		0	Writes to bits 9 through 1 are disabled.
		1	Writes to bits 9 through 1 are enabled.

1.4.5 SDRAM Refresh Control Register (SDRCR)

The SDRAM refresh control register (SDRCR) is used to configure the rate at which connected SDRAM devices will be automatically refreshed by the EMIF. Refer to Section 1.2.6.6.1 for more details. The SDRCR is shown in Figure 1-29 and described in Table 1-44.

Figure 1-29. SDRAM Refresh Control Register (SDRCR)

15 13	12 0
Reserved	REFRATE
R-0	RW-07Dh

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-44. SDRAM Refresh Control Register (SDRCR) Field Descriptions

Bit	Field	Value	Description
15-13	Reserved	0	Reserved.
12-0	REFRATE	0-1FFFh	Refresh Rate. This field is used to define the SDRAM refresh period in terms of EM_SDCLK cycles. Writing a value < 0x0020 to this field will cause it to be loaded with (2 * T_RFC) + 1 value from SDRAM timing register.

1.4.6 Asynchronous Configuration Registers (ACSnCR1 and ACSnCR2)

The asynchronous CSn configuration registers (ACSnCR1 and ACSnCR2) are used to configure the shaping of the address and control signals during an access to asynchronous memory connected to EM_CS2, EM_CS3, EM_CS4, and EM_CS5. They are also used to program the width of asynchronous interface and to select from various modes of operation. These registers can be written prior to any transfer, and any asynchronous transfer following the write will use the new configuration. There are two registers per chip select pin. The ACSnCR1 and ACSnCR2 are shown in the following figures and described in Table 1-45 and Table 1-46.

Figure 1-30. Asynchronous CS2 Configuration Register 1 (ACS2CR1)

15 13	12 7	6 4	3 2	1 0
RSETUPLSB	RSTROBE	RHOLD	TA	ASIZE
RW-3	RW-3Fh	RW-7	RW-3	RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-31. Asynchronous CS2 Configuration Register 2 (ACS2CR2)

15	14	13 11	10 9	8
SS	EW	WSETUP		WSTROBE
RW-0	RW-0	RW-Fh		RW-3Fh

7 4	3 1	0
WSTROBE	WHOLD	RSETUPMSB
RW-3Fh	RW-7	RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-32. Asynchronous CS3 Configuration Register 1 (ACS3CR1)

15 13	12 7	6 4	3 2	1 0
RSETUPLSB	RSTROBE	RHOLD	TA	ASIZE
RW-3	RW-3Fh	RW-7	RW-3	RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-33. Asynchronous CS3 Configuration Register 2 (ACS3CR2)

15	14	13	11	10	9	8
SS	EW	WSETUP			WSTROBE	
RW-0	RW-0	RW-Fh			RW-3Fh	

7	4	3	1	0
WSTROBE		WHOLD		RSETUPMSB
RW-3Fh		RW-7		RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-34. Asynchronous CS4 Configuration Register 1 (ACS4CR1)

15	13	12	7	6	4	3	2	1	0
RSETUPLSB		RSTROBE		RHOLD		TA		ASIZE	
RW-3		RW-3Fh		RW-7		RW-3		RW-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-35. Asynchronous CS4 Configuration Register 2 (ACS4CR2)

15	14	13	11	10	9	8
SS	EW	WSETUP			WSTROBE	
RW-0	RW-0	RW-Fh			RW-3Fh	

7	4	3	1	0
WSTROBE		WHOLD		RSETUPMSB
RW-3Fh		RW-7		RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-36. Asynchronous CS5 Configuration Register 1 (ACS5CR1)

15	13	12	7	6	4	3	2	1	0
RSETUPLSB		RSTROBE		RHOLD		TA		ASIZE	
RW-3		RW-3Fh		RW-7		RW-3		RW-1	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-37. Asynchronous CS5 Configuration Register 2 (ACS5CR2)

15	14	13	11	10	9	8
SS	EW	WSETUP			WSTROBE	
RW-0	RW-0	RW-Fh			RW-3Fh	

7	4	3	1	0
WSTROBE		WHOLD		RSETUPMSB
RW-3Fh		RW-7		RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-45. Asynchronous CSn Configuration Register 1 (ACSnCR1) Field Descriptions

Bit	Field	Value	Description
15-13	RSETUPLSB	0-7h	These bits in conjunction with RSETUPMSB in ACE1CR2 define the read setup timing in EM_SDCLK cycles, minus one cycle.
12-7	RSTROBE	0-3Fh	Read strobe width in EM_SDCLK cycles, minus one cycle.
6-4	RHOLD	0-7h	Read hold width in EM_SDCLK cycles, minus one cycle.
3-2	TA	0-3h	Minimum turn-around time. This field defines the minimum number of EM_SDCLK cycles between reads and writes, minus one cycle.
1-0	ASIZE	0-3h	Asynchronous data bus width. This field defines the width of the asynchronous device's data bus.
		0	8- bit data bus.
		1h	16-bit data bus.

Table 1-45. Asynchronous CSn Configuration Register 1 (ACSnCR1) Field Descriptions (continued)

Bit	Field	Value	Description
		2-3h	Reserved.

Table 1-46. Asynchronous CSn Configuration Register 2 (ACSnCR2) Field Descriptions

Bit	Field	Value	Description
15	SS		Select Strobe bit. This bit defines whether the asynchronous interface operates in Normal Mode or Select Strobe Mode.
		0	Normal Mode enabled.
		1	Select Strobe Mode enabled.
14	EW		Extended wait cycles enable bit. This bit defines whether extended wait cycles will be enabled.
		0	Extended wait cycles disabled.
		1	Extended wait cycles enabled.
13-10	WSETUP	0-Fh	Write setup width in EM_SDCLK cycles, minus one cycle.
9-4	WSTROBE	0-3Fh	Write strobe width in EM_SDCLK cycles, minus one cycle.
3-1	WHOLD	0-7h	Write hold width in EM_SDCLK cycles, minus one cycle.
0	RSETUPMSB	0-1	These bits in conjunction with RSETUPLSB in ACE1CR1 define the read setup timing in EM_SDCLK cycles, minus one cycle.

1.4.7 SDRAM Timing Registers (SDTIMR1 and SDTIMR2)

The SDRAM Timing Registers (SDTIMR1 and SDTIMR2) are used to program many of the SDRAM timing parameters. Consult the SDRAM datasheet for information on the appropriate values to program into each field.

SDRAM timing registers are shown in Figure 1-38 and Figure 1-39 and described in Table 1-47 and Table 1-48.

Figure 1-38. SDRAM Timing Register 1 (SDTIMR1)

15 12	11 8	7	6 4	3 0
TRAS	TRC	Rsvd	TRRD	Reserved
RW-5	RW-8	R-0	RW-2	R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-39. SDRAM Timing Register 1 (SDTIMR2)

15 11	10 8	7	6 4	3	2 0
TRFC	TRP	Rsvd	TRCD	Rsvd	TWR
RW-8	RW-2	R-0	RW-2	R-0	RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-47. SDRAM Timing Register 1 (SDTIMR1) Field Descriptions

Bit	Field	Value	Description
15-12	TRAS	0-Fh	Specifies the t_{RAS} value of the SDRAM. This defines the minimum number of EM_SDCLK clock cycles from Activate (ACTV) to Precharge (PRE), minus 1: T_RAS = (t_{RAS}/t_{EM_SDCLK}) - 1
11-8	TRC	0-Fh	Specifies the t_{RC} value of the SDRAM. This defines the minimum number of EM_SDCLK clock cycles from Activate (ACTV) to Activate (ACTV), minus 1: T_RC = (t_{RC}/t_{EM_SDCLK}) - 1
7	Reserved	0	Reserved
6-4	TRRD	0-7h	Specifies the t_{RRD} value of the SDRAM. This defines the minimum number of EM_SDCLK clock cycles from Activate (ACTV) to Activate (ACTV) for a different bank, minus 1: T_RRD = (t_{RRD}/t_{EM_SDCLK}) - 1
3-0	Reserved	0	Reserved

Table 1-48. SDRAM Timing Register 1 (SDTIMR2) Field Descriptions

Bit	Field	Value	Description
15-11	TRFC	0-1Fh	Specifies the t_{RFC} value of the SDRAM. This defines the minimum number of EM_SDCLK cycles from Refresh (REFR) to Refresh (REFR), minus 1: T_RFC = (t_{RFC}/t_{EM_SDCLK}) - 1
10-8	TRP	0-7h	Specifies the t_{RP} value of the SDRAM. This defines the minimum number of EM_SDCLK cycles from Precharge (PRE) to Activate (ACTV) or Refresh (REFR) command, minus 1: T_RP = (t_{RP}/t_{EM_SDCLK}) - 1
7	Reserved	0	Reserved
6-4	TRCD	0-7h	Specifies the t_{RCD} value of the SDRAM. This defines the minimum number of EM_SDCLK cycles from Active (ACTV) to Read (READ) or Write (WRT), minus 1: T_RCD = (t_{RCD}/t_{EM_SDCLK}) - 1
3	Reserved	0	Reserved
2-0	TWR	0-7h	Specifies the t_{WR} value of the SDRAM. This defines the minimum number of EM_SDCLK cycles from last Write (WRT) to Precharge (PRE), minus 1: T_WR = (t_{WR}/t_{EM_SDCLK}) - 1

1.4.8 SDRAM Self Refresh Exit Timing Register (SDSRETR)

The SDRAM Self Refresh Exit Timing Register (SDSRETR) is used to program the amount of time between when the SDRAM exits self-refresh mode and when the EMIF issues another command.

The SDRAM self refresh exit timing register is shown in Figure 1-40 and described in Table 1-49.

Figure 1-40. SDRAM Self Refresh Exit Timing Register (SDSRETR)

15	5	4	0
Reserved		TXS	
R-0		RW-5	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-49. SDRAM Self Refresh Exit Timing Register (SDSRETR) Field Descriptions

Bit	Field	Value	Description
15-5	Reserved	0	Reserved.
4-0	TXS	0-1Fh	This field specifies the minimum number of EM_SDCLK cycles from Self-Refresh exit to any command, minus one. T_XS = t_{XSR} / t_{EM_SDCLK} - 1

1.4.9 Interrupt Raw Register (EIRR)

The EMIF Interrupt Raw Register (EIRR) is used to monitor and clear the EMIF's hardware-generated interrupts. The bits in this register will be set when an interrupt condition occurs regardless of the status of the interrupt mask set register and interrupt mast clear register. Writing a 1 to the bits of this register will clear them. The EMIF interrupt raw register is shown in Figure 1-41 and described in Table 1-50.

Figure 1-41. Interrupt Raw Register (EIRR)

15	8
Reserved	
R-0	

7	6	5	2	1	0
Reserved		WR		LT	AT
R-0		R/W1C-0		R/W1C-0	R/W1C-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset; W1C = write 1 to clear (writing 0 has no effect)

Table 1-50. Interrupt Raw Register (EIRR) Field Descriptions

Bit	Field	Value	Description
15-6	Reserved	0	Reserved
5-2	WR[3:0]	0-Fh	Wait rise bits. These bits are set to 1 by hardware when a rising edge is detected on the wait pins. The polarity bits of AWCCR2 have not effect on these bits. Writing a 1 will clear these bits as well as the WRMASKED bits in the interrupt masked register (EMIR). Writing a 0 has no effect.
1	LT	0	Line trap. This bit is set to 1 by hardware to indicate illegal memory access. Writing a 1 will clear this bit as well as the LTMASKED bit in the interrupt masked register (EMIR). Writing a 1 has no effect.
0	AT	0	Asynchronous timeout. This bit is set to 1 by hardware to indicate that during an extended asynchronous memory access cycle, a wait pin did not go inactive within the number of cycles defined by the MEWC field in asynchronous wait cycle configuration register 1 (AWCCR1). Writing a 1 will clear these bits as well as the ATMASKED bit in the interrupt masked register (EMIR). Writing a 0 has no effect.

1.4.10 Interrupt Mask Register (EIMR)

Like the EMIF Interrupt Raw Register (EIRR), the EMIF Interrupt Mask Register (EIMR) is used to monitor and clear the status of the EMIF's hardware-generated interrupts. The main difference between the two registers is that when the bits in this register are set, an active-high pulse will be sent to the CPU interrupt controller. Also, the bit fields in EIMR are only set to 1 if the associated interrupt has been enabled in the EMIF interrupt mask set register (EIMSR). The interrupt mask register is shown in Figure 1-42 and described in Table 1-51.

Figure 1-42. Interrupt Mask Register (EIMR)

15	8
Reserved	
R-0	

7	6	5	2	1	0
Reserved		WRMASKED		LTMASKED	ATMASKED
R-0		R/W1C-0		R/W1C-0	R/W1C-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset; W1C = write 1 to clear (writing 0 has no effect)

Table 1-51. Interrupt Mask Register (EIMR) Field Descriptions

Bit	Field	Value	Description
15-6	Reserved	0	Reserved
5-2	WRMASKED[3:0]	0-Fh	Masked wait rise bits. These bits are set to 1 by hardware when a rising edge is detected on the wait pins. The polarity bits of AWCCR2 have not effect on these bits. Writing a 1 will clear these bits as well as the WR bits in the interrupt raw register (EIRR). Writing a 0 has no effect.
1	LTMASKED	0	Masked linetrap. This bit is set to 1 by hardware to indicate illegal memory access, only if the LTMASKSET bit in the interrupt mask set register (EMISR) is set to 1. Writing a 1 will clear this bit as well as the LT bit in the interrupt raw register (EIRR). Writing a 0 has no effect.
0	ATMASKED	0	Masked asynchronous timeout. This bit is set to 1 by hardware to indicate that during an extended asynchronous memory access cycle, a wait pin did not go inactive within the number of cycles defined by the MEWC field in asynchronous wait cycle configuration register 1 (AWCCR1), only if the ATMASKSET bit in the interrupt mask set register (EIMSR) is set to 1. Writing a 1 will clear these bits as well as the AT bit in the interrupt raw register (EIRR). Writing a 0 has no effect.

1.4.11 Interrupt Mask Set Register (EIMSR)

The EMIF Interrupt Mask Set Register (EIMSR) is used to enable the EMIF interrupts. If read as 1, the corresponding bit in the EMIF Interrupt Mask Register (EIMR) will be set and an interrupt will be generated when the interrupt condition occurs. If read as 0, the corresponding bit in EIMR will always read 0 and no interrupt will be generated when the interrupt condition occurs. Writing a 1 to the bits of EIMSR enables the EMIF interrupts. The interrupt mask set register is shown in Figure 1-43 and described in Table 1-52.

Figure 1-43. Interrupt Mask Set Register (EIMSR)

15	8
Reserved	
R-0	

7	6	5	2	1	0
Reserved		WRMASKSET		LTMASKSET	ATMASKSET
R-0		R/W1S-0		R/W1S-0	R/W1S-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset; W1S = write 1 to set (writing 0 has no effect)

Table 1-52. Interrupt Mask Set Register (EIMSR) Field Descriptions

Bit	Field	Value	Description
15-6	Reserved	0	Reserved
5-2	WRMASKSET[3:0]	0-Fh	Mask set for WRMASKED bits in interrupt mask register (EIMR). Writing a 1 will enable the wait rise interrupts and set these bits as well as the WRMASKCLR bits in the interrupt mask clear register (EIMCR). Writing a 0 has no effect.
1	LTMASKSET	0	Mask set for LTMASKED bits in interrupt mask register (EIMR). Writing a 1 will enable the interrupt and set these bits as well as the LTMASKCLR bits in the interrupt mask clear register (EIMCR). Writing a 0 has no effect.
0	ATMASKSET	0	Mask set for ATMASKED bit in interrupt mask register (EIMR). Writing a 1 will enable the interrupt and set this bit as well as the ATMASKCLR bit in the interrupt mask clear register (EIMCR). Writing a 0 has no effect.

1.4.12 Interrupt Mask Clear Register (EIMCR)

The EMIF Interrupt Mask Clear Register (EIMCR) is used to disable the EMIF interrupts. Writing a 1 to the bits of this register disables the interrupt. Writing a 0 has no effect. The interrupt mask clear register is shown in Figure 1-44 and described in Table 1-53.

Figure 1-44. Interrupt Mask Clear Register (EIMCR)

15	8
Reserved	
R-0	

7	6	5	2	1	0
Reserved		WRMASKCLR		LTMASKCLR	ATMASKCLR
R-0		R/W1C-0		R/W1C-0	R/W1C-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset; W1C = write 1 to clear (writing 0 has no effect)

Table 1-53. Interrupt Mask Clear Register (EIMCR) Field Descriptions

Bit	Field	Value	Description
15-6	Reserved	0	Reserved
5-2	WRMASKCLR[3:0]	0-Fh	Mask clear for WRMASKED bits in interrupt mask register (EIMR). Writing a 1 will disable the wait rise interrupts and clear these bits as well as the WRMASKSET bits in the interrupt mask set register (EIMSR). Writing a 0 has no effect.
1	LTMASKCLR	0	Mask clear for LTMASKED bits in interrupt mask register (EIMR). Writing a 1 will disable the interrupt and clear this bits as well as the LTMASKSET bits in the interrupt mask set register (EIMSR). Writing a 0 has no effect.
0	ATMASKCLR	0	Mask clear for ATMASKED bit in interrupt mask register (EIMR). Writing a 1 will disable the interrupt and clear this bit as well as the ATMASKSET bit in the interrupt mask set register (EIMSR). Writing a 0 has no effect.

1.4.13 NAND Flash Control Register (NANDFCR)

The NAND Flash Control Register (NANDFCR) is shown in Figure 1-45 and described in Table 1-54.

Figure 1-45. NAND Flash Control Register (NANDFCR)

15	14	13	12	11	10	9	8
Reserved		ADDR_CALC_ST	4BIT_ECC_START	CS5_ECC_START	CS4_ECC_START	CS3_ECC_START	CS2_ECC_START
R-0		RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

7	6	5	4	3	2	1	0
Reserved		4BIT_ECC_SEL		CS5_USE_NAND	CS4_USE_NAND	CS3_USE_NAND	CS2_USE_NAND
R-0		RW-0		RW-0	RW-0	RW-0	RW-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-54. NAND Flash Control Register (NANDFCR) Field Descriptions

Bit	Field	Value	Description
15-14	Reserved	0	Reserved
13	ADDR_CALC_ST		NAND flash 4-bit ECC error address and error value calculation start. Set to 1 to start 4-bit ECC error address and error value calculation on read syndrome. This bit is cleared when any of the NAND flash error address registers or NAND flash error value registers are read. Writing a 0 has no effect.
12	4BIT_ECC_START		NAND flash 4-bit ECC start for the selected chip select. Set to 1 to start 4-bit ECC calculation on data for NAND flash on chip select selected by 4BIT_ECC_SEL field. This bit is cleared when any of the NAND flash 4-Bit ECC registers are read. Writing a 0 has no effect.
11	CS5_ECC_START		NAND flash 1-bit ECC start for EMIF CS5. Set to 1 to start 1-bit ECC calculation on data for NAND flash on EMIF CS5. This bit is cleared when NAND flash CS5 1-Bit ECC register is read. Writing a 0 has no effect.

Table 1-54. NAND Flash Control Register (NANDFCR) Field Descriptions (continued)

Bit	Field	Value	Description
10	CS4_ECC_START		NAND flash 1-bit ECC start for EMIF CS4. Set to 1 to start 1-bit ECC calculation on data for NAND flash on EMIF CS4. This bit is cleared when NAND flash CS4 1-Bit ECC register is read. Writing a 0 has no effect.
9	CS3_ECC_START		NAND flash 1-bit ECC start for EMIF CS3. Set to 1 to start 1-bit ECC calculation on data for NAND flash on EMIF CS3. This bit is cleared when NAND flash CS3 1-Bit ECC register is read. Writing a 0 has no effect.
8	CS2_ECC_START		NAND flash 1-bit ECC start for EMIF CS2. Set to 1 to start 1-bit ECC calculation on data for NAND flash on EMIF CS2. This bit is cleared when NAND flash CS2 1-Bit ECC register is read. Writing a 0 has no effect.
7-6	Reserved	0	Reserved
5-4	4BIT_ECC_SEL		NAND flash 4-bit ECC chip select selection. This field selects the chip select on which the 4-bit ECC will be calculated.
		0	Select EMIF CS2 for 4-bit ECC calculation.
		1h	Select EMIF CS3 for 4-bit ECC calculation.
		2h	Select EMIF CS4 for 4-bit ECC calculation.
		3h	Select EMIF CS5 for 4-bit ECC calculation.
3	CS5_USE_NAND		NAND flash mode for EMIF CS5. Set to 1 if using NAND flash on $\overline{\text{EM\_CS5}}$.
		0	Not using NAND flash.
		1	Using NAND flash on chip select pin.
2	CS4_USE_NAND		NAND flash mode for EMIF CS4. Set to 1 if using NAND flash on $\overline{\text{EM\_CS4}}$.
		0	Not using NAND flash.
		1	Using NAND flash on chip select pin.
1	CS3_USE_NAND		NAND flash mode for EMIF CS3. Set to 1 if using NAND flash on $\overline{\text{EM\_CS3}}$.
		0	Not using NAND flash.
		1	Using NAND flash on chip select pin.
0	CS2_USE_NAND		NAND flash mode for EMIF CS2. Set to 1 if using NAND flash on $\overline{\text{EM\_CS2}}$.
		0	Not using NAND flash.
		1	Using NAND flash on chip select pin.

1.4.14 NAND Flash Status Registers (NANDFSR1) and (NANDFSR1)

The NAND Flash Status Registers (NANDFSR1) and (NANDFSR1) are shown in Figure 1-46 and Figure 1-47 and described in Table 1-55 and Table 1-56.

Figure 1-46. NAND Flash Status Register 1 (NANDFSR1)

15 12	11 8	7 4	3 0
Reserved	CORRSTATE	Reserved	WAITSTAT
R-0	R-0	R-0	R-EM_WAIT[3:0]

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-47. NAND Flash Status Register 2 (NANDFSR2)

15 2	1 0
Reserved	ERRNUM
R-0	R-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-55. NAND Flash Status Register 1 (NANDFSR1) Field Descriptions

Bit	Field	Value	Description
15-12	Reserved	0	Reserved
11-8	CORRSTATE		4-bit ECC state value when performing error address and error value calculation.
		0	No error.
		1h	Errors cannot be corrected (five or more errors).
		2h	Error correction complete (errors on bit 8 or 9).
		3h	Error correction complete (error exists).
		4h	Reserved.
		5h	Calculating number of errors.
		6h	Preparing for error search.
		7h	Preparing for error search.
		8h	Searching for errors.
		9h	Reserved.
		10h	Reserved.
		11h	Reserved.
		12h	Calculating error value.
		13h	Calculating error value.
		14h	Calculating error value.
		15h	Calculating error value.
7-4	Reserved	0	Reserved
3-0	WAITSTAT	0-Fh	These bits shows the raw status of the four wait input signals (EM_WAIT[3:0]). WAITSTAT[0] corresponds to EM_WAIT[0], WAITSTAT[1] corresponds to EM_WAIT[1], and so on. The polarity bits in the asynchronous wait cycle configuration register 2 (AWCCR2) have no effect on these bits.

Table 1-56. NAND Flash Status Register 2 (NANDFSR2) Field Descriptions

Bit	Field	Value	Description
15-2	Reserved	0	Reserved
1-0	ERRNUM		4-bit ECC error number. This field shows the number for errors found after the error address calculation and error value calculation is done.
		0	1 error found.
		1h	2 errors found.
		2h	3 errors found.
		3h	4 errors found.

1.4.15 Page Mode Control Registers (PAGEMODCTRL1) and (PAGEMODCTRL2)

The Page Mode Control Registers (PAGEMODCTRL1 and PAGEMODCTRL2) are shown in Figure 1-48 and Figure 1-49 and described in Table 1-57 and Table 1-58.

Figure 1-48. Page Mode Control Register 1 (PAGEMODCTRL1)

15	10	9	8
CS3_PAGE_DELAY		CS3_PAGE_SIZE	CS3_PAGEMOD_EN
RW-3Fh		RW-0	RW-0
7	2	1	0
CS2_PAGE_DELAY		CS2_PAGE_SIZE	CS2_PAGEMOD_EN
RW-3Fh		RW-1	RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-49. Page Mode Control Register 2 (PAGEMODCTRL2)

15	10	9	8
CS5_PAGE_DELAY		CS5_PAGE_SIZE	CS5_PAGEMOD_EN
RW-3Fh		RW-0	RW-0
7	2	1	0
CS4_PAGE_DELAY		CS4_PAGE_SIZE	CS4_PAGEMOD_EN
RW-3Fh		RW-0	RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-57. Page Mode Control Register 1 (PAGEMODCTRL1) Field Descriptions

Bit	Field	Value	Description
15-10	CS3_PAGE_DELAY	0-3Fh	Page access delay for NOR flash connected on $\overline{\text{EM\_CS3}}$. Number of EMIF clock cycles required for the page read data to be valid, minus one cycle. This value must not be set to 0.
9	CS3_PAGE_SIZE		Page size for NOR flash connected on $\overline{\text{EM\_CS3}}$.
		0	4-word page.
		1	8-word page.
8	CS3_PAGEMOD_EN		Page mode enable for NOR flash connected on $\overline{\text{EM\_CS3}}$.
		0	Disable page mode.
		1	Enable page mode.
7-2	CS2_PAGE_DELAY	0-3Fh	Page access delay for NOR flash connected on $\overline{\text{EM\_CS2}}$. Number of EMIF clock cycles required for the page read data to be valid, minus one cycle. This value must not be set to 0.
1	CS2_PAGE_SIZE		Page size for NOR flash connected on $\overline{\text{EM\_CS2}}$.
		0	4-word page.
		1	8-word page.
0	CS2_PAGEMOD_EN		Page mode enable for NOR flash connected on $\overline{\text{EM\_CS2}}$.
		0	Disable page mode.
		1	Enable page mode.

Table 1-58. Page Mode Control Register 2 (PAGEMODCTRL2) Field Descriptions

Bit	Field	Value	Description
15-10	CS5_PAGE_DELAY	0-3Fh	Page access delay for NOR flash connected on $\overline{\text{EM\_CS5}}$. Number of EMIF clock cycles required for the page read data to be valid, minus one cycle. This value must not be set to 0.
9	CS5_PAGE_SIZE		Page size for NOR flash connected on $\overline{\text{EM\_CS5}}$.
		0	4-word page.
		1	8-word page.
8	CS5_PAGEMOD_EN		Page mode enable for NOR flash connected on $\overline{\text{EM\_CS5}}$.
		0	Disable page mode.
		1	Enable page mode.
7-2	CS4_PAGE_DELAY	0-3Fh	Page access delay for NOR flash connected on $\overline{\text{EM\_CS4}}$. Number of EMIF clock cycles required for the page read data to be valid, minus one cycle. This value must not be set to 0.
1	CS4_PAGE_SIZE		Page size for NOR flash connected on $\overline{\text{EM\_CS4}}$.
		0	4-word page.
		1	8-word page.
0	CS4_PAGEMOD_EN		Page mode enable for NOR flash connected on $\overline{\text{EM\_CS4}}$.
		0	Disable page mode.
		1	Enable page mode.

1.4.16 NAND Flash CSn 1-Bit ECC Registers (NCSnECC1) and (NCSnECC2)

The NAND Flash CSn 1-Bit ECC Registers (NCSnECC1) and (NCSnECC2) are shown in Figure 1-50 and Figure 1-51 and described in Table 1-59 and Table 1-60.

Figure 1-50. NAND Flash CSn 1-Bit ECC Register 1 (NCSnECC1)

15			12	11	10	9	8
Reserved				P2048E	P1024E	P512E	P256E
R-0				RW-0	RW-0	RW-0	RW-0
7	6	5	4	3	2	1	0
P128E	P64E	P32E	P16E	P8E	P4E	P2E	P1E
RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 1-51. NAND Flash CSn 1-Bit ECC Register 2 (NCSnECC2)

15			12	11	10	9	8
Reserved				P2048O	P1024O	P512O	P256O
R-0				RW-0	RW-0	RW-0	RW-0
7	6	5	4	3	2	1	0
P128O	P64O	P32O	P16O	P8O	P4O	P2O	P1O
RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-59. NAND Flash CSn 1-Bit ECC Register 1 (NCSnECC1) Field Descriptions

Bit	Field	Value	Description
15-12	Reserved	0	Reserved
11	P2048E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
10	P1024E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
9	P512E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
8	P256E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
7	P128E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
6	P64E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
5	P32E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
4	P16E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
3	P8E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
2	P4E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.

Table 1-59. NAND Flash CSn 1-Bit ECC Register 1 (NCSnECC1) Field Descriptions (continued)

Bit	Field	Value	Description
1	P2E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.
0	P1E		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1E, P2E, and P4E are column parities and P8E to P2048E are row parities. For 16-bit NAND Flash, P1E, P2E, P4E, and P8E are column parities and P16E to P2048E are row parities.

Table 1-60. NAND Flash CSn 1-Bit ECC Register 2 (NCSnECC2) Field Descriptions

Bit	Field	Value	Description
15-12	Reserved	0	Reserved
11	P2048O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
10	P1024O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
9	P512O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
8	P256O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
7	P128O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
6	P64O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
5	P32O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
4	P16O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
3	P8O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
2	P4O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
1	P2O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.
0	P1O		1-Bit ECC code calculated while reading/writing NAND Flash. For 8-bit NAND Flash, P1O, P2O, and P4O are column parities and P8O to P2048O are row parities. For 16-bit NAND Flash, P1O, P2O, P4O, and P8O are column parities and P16O to P2048O are row parities.

1.4.17 NAND Flash 4-Bit ECC Load Register (NAND4BITECCLOAD)

NAND Flash 4-Bit ECC Load Register (NAND4BITECCLOAD) is shown in Figure 1-52 and described in Table 1-61.

Figure 1-52. NAND Flash 4-Bit ECC Load Register (NAND4BITECCLOAD)

15	10	9	0
Reserved		4BIT_ECC_LOAD	
R-0		RW-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-61. NAND Flash 4-Bit ECC Load Register (NAND4BITECCLOAD) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_LOAD	0-3FFh	4-Bit ECC Load. This register is used to load 4-bit ECC values when performing syndrome calculation during NAND Flash reads.

1.4.18 NAND Flash 4-Bit ECC Register 1 (NAND4BITECC1)

The NAND Flash 4-bit ECC Register 1 (NAND4BITECC1) is shown in Figure 1-53 and described in Table 1-62.

Figure 1-53. NAND Flash 4-Bit ECC Load Register 1 (NAND4BITECC1)

15	10	9	0
Reserved		4BIT_ECC_VAL1	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-62. NAND Flash 4-Bit ECC Load Register 1 (NAND4BITECC1) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL1	0-3FFh	4-Bit ECC or syndrome value 1 calculated while writing or reading NAND Flash.

1.4.19 NAND Flash 4-Bit ECC Register 2 (NAND4BITECC2)

The NAND Flash 4-bit ECC Register 2 (NAND4BITECC2) is shown in Figure 1-54 and described in Table 1-63.

Figure 1-54. NAND Flash 4-Bit ECC Register 2 (NAND4BITECC2)

15	10	9	0
Reserved		4BIT_ECC_VAL2	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-63. NAND Flash 4-Bit ECC Register 2 (NAND4BITECC2) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL2	0-3FFh	4-Bit ECC or syndrome value 2 calculated while writing or reading NAND Flash.

1.4.20 NAND Flash 4-Bit ECC Register 3 (NAND4BITECC3)

The NAND Flash 4-bit ECC Register 3 (NAND4BITECC3) is shown in Figure 1-55 and described in Table 1-64.

Figure 1-55. NAND Flash 4-Bit ECC Register 3 (NAND4BITECC3)

15	10	9	0
Reserved		4BIT_ECC_VAL3	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-64. NAND Flash 4-Bit ECC Register 3 (NAND4BITECC3) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL3	0-3FFh	4-Bit ECC or syndrome value 3 calculated while writing or reading NAND Flash.

1.4.21 NAND Flash 4-Bit ECC Register 4 (NAND4BITECC4)

The NAND Flash 4-bit ECC Register 4 (NAND4BITECC4) is shown in Figure 1-56 and described in Table 1-65.

Figure 1-56. NAND Flash 4-Bit ECC Register 4 (NAND4BITECC4)

15	10	9	0
Reserved		4BIT_ECC_VAL4	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-65. NAND Flash 4-Bit ECC Register 4 (NAND4BITECC4) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL4	0-3FFh	4-Bit ECC or syndrome value 4 calculated while writing or reading NAND Flash.

1.4.22 NAND Flash 4-Bit ECC Register 5 (NAND4BITECC5)

The NAND Flash 4-bit ECC Register 5 (NAND4BITECC5) is shown in Figure 1-57 and described in Table 1-66.

Figure 1-57. NAND Flash 4-Bit ECC Register 5 (NAND4BITECC5)

15	10	9	0
Reserved		4BIT_ECC_VAL5	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-66. NAND Flash 4-Bit ECC Register 5 (NAND4BITECC5) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL5	0-3FFh	4-Bit ECC or syndrome value 5 calculated while writing or reading NAND Flash.

1.4.23 NAND Flash 4-Bit ECC Register 6 (NAND4BITECC6)

The NAND Flash 4-bit ECC Register 6 (NAND4BITECC6) is shown in Figure 1-58 and described in Table 1-67.

Figure 1-58. NAND Flash 4-Bit ECC Register 6 (NAND4BITECC6)

15	10	9	0
Reserved		4BIT_ECC_VAL6	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-67. NAND Flash 4-Bit ECC Register 6 (NAND4BITECC6) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL6	0-3FFh	4-Bit ECC or syndrome value 6 calculated while writing or reading NAND Flash.

1.4.24 NAND Flash 4-Bit ECC Register 7 (NAND4BITECC7)

The NAND Flash 4-bit ECC Register 7 (NAND4BITECC7) is shown in Figure 1-59 and described in Table 1-68.

Figure 1-59. NAND Flash 4-Bit ECC Register 7 (NAND4BITECC7)

15	10	9	0
Reserved		4BIT_ECC_VAL7	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-68. NAND Flash 4-Bit ECC Register 7 (NAND4BITECC7) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL7	0-3FFh	4-Bit ECC or syndrome value 7 calculated while writing or reading NAND Flash.

1.4.25 NAND Flash 4-Bit ECC Register 8 (NAND4BITECC8)

The NAND Flash 4-bit ECC Register 8 (NAND4BITECC8) is shown in Figure 1-60 and described in Table 1-69.

Figure 1-60. NAND Flash 4-Bit ECC Register 8 (NAND4BITECC8)

15	10	9	0
Reserved		4BIT_ECC_VAL8	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-69. NAND Flash 4-Bit ECC Register 8 (NAND4BITECC8) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	4BIT_ECC_VAL8	0-3FFh	4-Bit ECC or syndrome value 8 calculated while writing or reading NAND Flash.

1.4.26 NAND Flash 4-Bit ECC Error Address Register 1 (NANDERRADD1)

The NAND Flash 4-bit ECC Error Register 1 (NANDERRADD1) is shown in Figure 1-61 and described in Table 1-70.

Figure 1-61. NAND Flash 4-Bit ECC Error Address Register 1 (NANDERRADD1)

15	10	9	0
Reserved		ERR_ADDR1	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-70. NAND Flash 4-Bit ECC Error Address Register 1 (NANDERRADD1) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_ADDR1	0-3FFh	4-bit ECC error address 1.

1.4.27 NAND Flash 4-Bit ECC Error Address Register 2 (NANDERRADD2)

The NAND Flash 4-bit ECC Error Register 2 (NANDERRADD2) is shown in Figure 1-62 and described in Table 1-71.

Figure 1-62. NAND Flash 4-Bit ECC Error Address Register 2 (NANDERRADD2)

15	10	9	0
Reserved		ERR_ADDR2	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-71. NAND Flash 4-Bit ECC Error Address Register 2 (NANDERRADD2) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_ADDR2	0-3FFh	4-bit ECC error address 2.

1.4.28 NAND Flash 4-Bit ECC Error Address Register 3 (NANDERRADD3)

The NAND Flash 4-bit ECC Error Register 3 (NANDERRADD3) is shown in Figure 1-63 and described in Table 1-72.

Figure 1-63. NAND Flash 4-Bit ECC Error Address Register 3 (NANDERRADD3)

15	10	9	0
Reserved		ERR_ADDR3	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-72. NAND Flash 4-Bit ECC Error Address Register 3 (NANDERRADD3) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_ADDR3	0-3FFh	4-bit ECC error address 3.

1.4.29 NAND Flash 4-Bit ECC Error Address Register 4 (NANDERRADD4)

The NAND Flash 4-bit ECC Error Register 4 (NANDERRADD4) is shown in Figure 1-64 and described in Table 1-73.

Figure 1-64. NAND Flash 4-Bit ECC Error Address Register 4 (NANDERRADD4)

15	10	9	0
Reserved		ERR_ADDR4	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-73. NAND Flash 4-Bit ECC Error Address Register 4 (NANDERRADD4) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_ADDR4	0-3FFh	4-bit ECC error address 4.

1.4.30 NAND Flash 4-Bit ECC Error Value Register 1 (NANDERRVAL1)

The NAND Flash 4-bit ECC Error Value Register 1 (NANDERRVAL1) is shown in Figure 1-65 and described in Table 1-74.

Figure 1-65. NAND Flash 4-Bit ECC Error Value Register 1 (NANDERRVAL1)

15	10	9	0
Reserved		ERR_VALUE1	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-74. NAND Flash 4-Bit ECC Error Value Register 1 (NANDERRVAL1) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_VALUE1	0-3FFh	4-bit ECC error value 1.

1.4.31 NAND Flash 4-Bit ECC Error Value Register 2 (NANDERRVAL2)

The NAND Flash 4-bit ECC Error Value Register 2 (NANDERRVAL2) is shown in Figure 1-66 and described in Table 1-75.

Figure 1-66. NAND Flash 4-Bit ECC Error Value Register 2 (NANDERRVAL2)

15	10	9	0
Reserved		ERR_VALUE2	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-75. NAND Flash 4-Bit ECC Error Value Register 2 (NANDERRVAL2) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_VALUE2	0-3FFh	4-bit ECC error value 2.

1.4.32 NAND Flash 4-Bit ECC Error Value Register 3 (NANDERRVAL3)

The NAND Flash 4-bit ECC Error Value Register 3 (NANDERRVAL3) is shown in Figure 1-67 and described in Table 1-76.

Figure 1-67. NAND Flash 4-Bit ECC Error Value Register 3 (NANDERRVAL3)

15	10	9	0
Reserved		ERR_VALUE3	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-76. NAND Flash 4-Bit ECC Error Value Register 3 (NANDERRVAL3) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_VALUE3	0-3FFh	4-bit ECC error value 3.

1.4.33 NAND Flash 4-Bit ECC Error Value Register 4 (NANDERRVAL4)

The NAND Flash 4-bit ECC Error Value Register 4 (NANDERRVAL4) is shown in Figure 1-68 and described in Table 1-77.

Figure 1-68. NAND Flash 4-Bit ECC Error Value Register 4 (NANDERRVAL4)

15	10	9	0
Reserved		ERR_VALUE4	
R-0		R-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 1-77. NAND Flash 4-Bit ECC Error Value Register 4 (NANDERRVAL4) Field Descriptions

Bit	Field	Value	Description
15-10	Reserved	0	Reserved
9-0	ERR_VALUE4	0-3FFh	4-bit ECC error value 4.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Communications and Telecom	www.ti.com/communications
DSP	dsp.ti.com	Computers and Peripherals	www.ti.com/computers
Clocks and Timers	www.ti.com/clocks	Consumer Electronics	www.ti.com/consumer-apps
Interface	interface.ti.com	Energy	www.ti.com/energy
Logic	logic.ti.com	Industrial	www.ti.com/industrial
Power Mgmt	power.ti.com	Medical	www.ti.com/medical
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Space, Avionics & Defense	www.ti.com/space-avionics-defense
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video and Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless-apps

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265