TMS320DM647/DM648 DSP
External Memory Interface (EMIF)

User's Guide

Literature Number: SPRUEK6B
December 2008

Contents

List of Figures

List of Tables

Read This First

About This Manual

This document describes the operation of the external memory interface (EMIF) in the TMS320DM647/DM648 Digital Signal Processor (DSP).

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.

Note: Acronyms 3PSW, CPSW, CPSW_3G, and 3pGSw are interchangeable and all refer to the 3 port gigabit switch.

Related Documentation From Texas Instruments

The following documents describe the TMS320DM647/DM648 Digital Signal Processor (DSP). Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

SPRS372 — ***TMS320DM647/DM648 Digital Media Processor Data Manual*** describes the signals, specifications and electrical characteristics of the device.

SPRU732 — ***TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide*** describes the CPU architecture, pipeline, instruction set, and interrupts for the TMS320C64x and TMS320C64x+ digital signal processors (DSPs) of the TMS320C6000 DSP family. The C64x/C64x+ DSP generation comprises fixed-point devices in the C6000 DSP platform. The C64x+ DSP is an enhancement of the C64x DSP with added functionality and an expanded instruction set.

SPRUEK5 — ***TMS320DM647/DM648 DSP DDR2 Memory Controller User's Guide*** describes the DDR2 memory controller in the TMS320DM647/DM648 Digital Signal Processor (DSP). The DDR2/mDDR memory controller is used to interface with JESD79D-2A standard compliant DDR2 SDRAM devices and standard Mobile DDR SDRAM devices.

SPRUEK6 — ***TMS320DM647/DM648 DSP External Memory Interface (EMIF) User's Guide*** describes the operation of the asynchronous external memory interface (EMIF) in the TMS320DM647/DM648 Digital Signal Processor (DSP). The EMIF supports a glueless interface to a variety of external devices.

SPRUEK7 — ***TMS320DM647/DM648 DSP General-Purpose Input/Output (GPIO) User's Guide*** describes the general-purpose input/output (GPIO) peripheral in the TMS320DM647/DM648 Digital Signal Processor (DSP). 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 the input by reading 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.

SPRUEK8 — ***TMS320DM647/DM648 DSP Inter-Integrated Circuit (I2C) Module User's Guide*** describes the inter-integrated circuit (I2C) peripheral in the TMS320DM647/DM648 Digital Signal Processor (DSP). The I2C peripheral provides an interface between the DSP and other devices compliant with the I2C-bus specification and connected by way of an I2C-bus. External components attached to this 2-wire serial bus can transmit and receive up to 8-bit wide data to and from the DSP through the I2C peripheral. This document assumes the reader is familiar with the I2C-bus specification.

SPRUEL0 — ***TMS320DM647/DM648 DSP 64-Bit Timer User's Guide*** describes the operation of the 64-bit timer in the TMS320DM647/DM648 Digital Signal Processor (DSP). The timer can be configured as a general-purpose 64-bit timer, dual general-purpose 32-bit timers, or a watchdog timer.

SPRUEL1 — ***TMS320DM647/DM648 DSP Multichannel Audio Serial Port (McASP) User's Guide*** describes the multichannel audio serial port (McASP) in the TMS320DM647/DM648 Digital Signal Processor (DSP). The McASP functions as a general-purpose audio serial port optimized for the needs of multichannel audio applications. The McASP is useful for time-division multiplexed (TDM) stream, Inter-Integrated Sound (I2S) protocols, and intercomponent digital audio interface transmission (DIT).

SPRUEL2 — ***TMS320DM647/DM648 DSP Enhanced DMA (EDMA) Controller User's Guide*** describes the operation of the enhanced direct memory access (EDMA3) controller in the TMS320DM647/DM648 Digital Signal Processor (DSP). The EDMA3 controller's primary purpose is to service user-programmed data transfers between two memory-mapped slave endpoints on the DSP.

SPRUEL4 — ***TMS320DM647/DM648 DSP Peripheral Component Interconnect (PCI) User's Guide*** describes the peripheral component interconnect (PCI) port in the TMS320DM647/DM648 Digital Signal Processor (DSP). The PCI port supports connection of the C642x DSP to a PCI host via the integrated PCI master/slave bus interface. The PCI port interfaces to the DSP via the enhanced DMA (EDMA) controller. This architecture allows for both PCI master and slave transactions, while keeping the EDMA channel resources available for other applications.

SPRUEL5 — ***TMS320DM647/DM648 DSP Host Port Interface (UHPI) User's Guide*** describes the host port interface (HPI) in the TMS320DM647/DM648 Digital Signal Processor (DSP). The HPI is a parallel port through which a host processor can directly access the CPU memory space. The host device functions as a master to the interface, which increases ease of access. The host and CPU can exchange information via internal or external memory. The host also has direct access to memory-mapped peripherals. Connectivity to the CPU memory space is provided through the enhanced direct memory access (EDMA) controller.

SPRUEL8 — ***TMS320DM647/DM648 DSP Universal Asynchronous Receiver/Transmitter (UART) User's Guide*** describes the universal asynchronous receiver/transmitter (UART) peripheral in the TMS320DM647/DM648 Digital Signal Processor (DSP). The UART peripheral performs serial-to-parallel conversion on data received from a peripheral device, and parallel-to-serial conversion on data received from the CPU.

SPRUEL9 — ***TMS320DM647/DM648 DSP VLYNQ Port User's Guide*** describes the VLYNQ port in the TMS320DM647/DM648 Digital Signal Processor (DSP). The VLYNQ port is a high-speed point-to-point serial interface for connecting to host processors and other VLYNQ compatible devices. It is a full-duplex serial bus where transmit and receive operations occur separately and simultaneously without interference.

SPRUEM1 — ***TMS320DM647/DM648 DSP Video Port/VCXO Interpolated Control (VIC) Port User's Guide*** discusses the video port and VCXO interpolated control (VIC) port in the TMS320DM647/DM648 Digital Signal Processor (DSP). The video port can operate as a video capture port, video display port, or transport channel interface (TCI) capture port. The VIC port provides single-bit interpolated VCXO control with resolution from 9 bits to up to 16 bits. When the video port is used in TCI mode, the VIC port is used to control the system clock, VCXO, for MPEG transport channel.

SPRUEM2 — ***TMS320DM647/DM648 DSP Serial Port Interface (SPI) User's Guide*** discusses the Serial Port Interface (SPI) in the TMS320DM647/DM648 Digital Signal Processor (DSP). This reference guide provides the specifications for a 16-bit configurable, synchronous serial peripheral interface. The SPI is a programmable-length shift register, used for high speed communication between external peripherals or other DSPs.

SPRUEU6 — **TMS320DM647/DM648 DSP Subsystem User's Guide** describes the subsystem in the TMS320DM647/DM648 Digital Signal Processor (DSP). The subsystem is responsible for performing digital signal processing for digital media applications. The subsystem acts as the overall system controller, responsible for handling many system functions such as system-level initialization, configuration, user interface, user command execution, connectivity functions, and overall system control.

SPRUF57 — **TMS320DM647/DM648 DSP 3 Port Switch (3PSW) Ethernet Subsystem User's Guide** describes the operation of the 3 port switch (3PSW) ethernet subsystem in the TMS320DM647/DM648 Digital Signal Processor (DSP). The 3 port switch gigabit ethernet subsystem provides ethernet packet communication and can be configured as an ethernet switch (DM648 only). It provides the serial gigabit media independent interface (SGMII), the management data input output (MDIO) for physical layer device (PHY) management.

DSP External Memory Interface (EMIF)

This document describes the operation and registers of the External Memory Interface (EMIF) in the device.

1 Overview

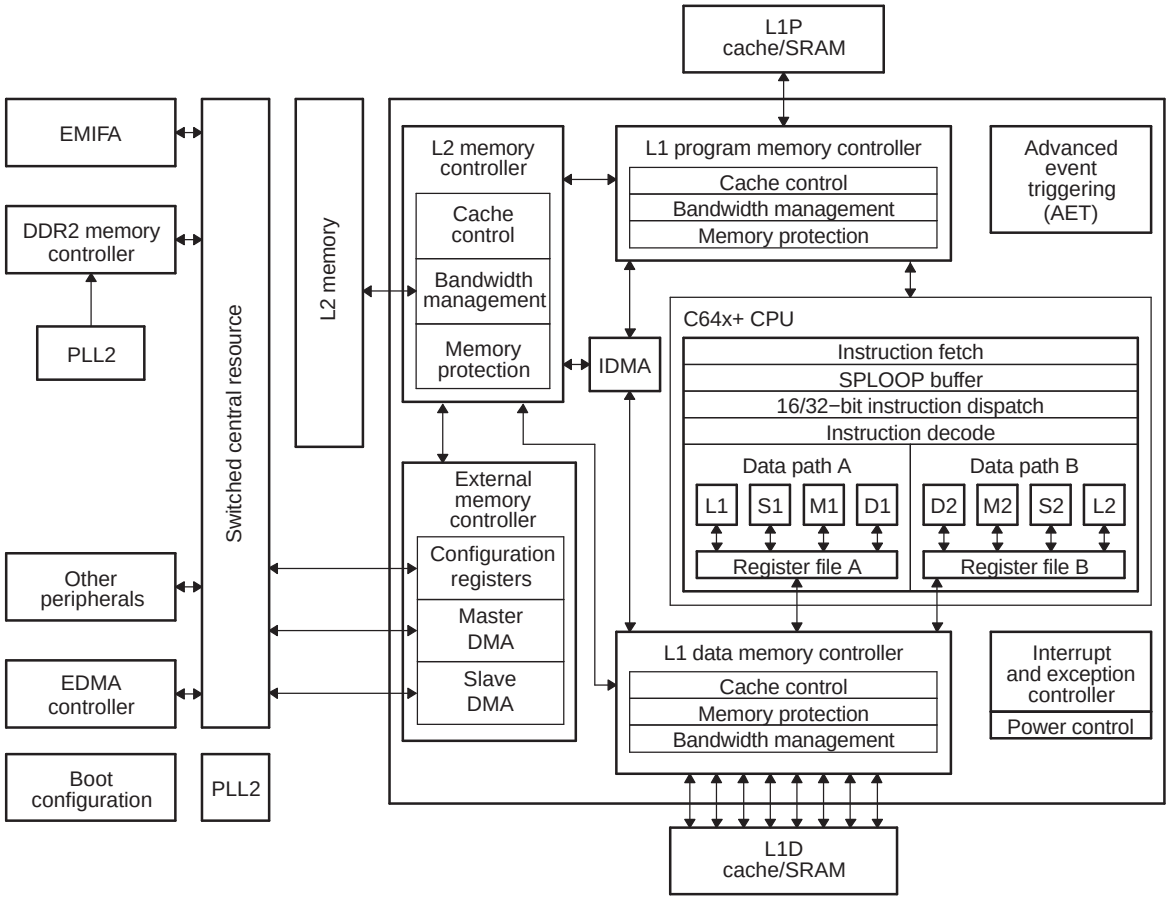
The DSP External Memory Interface (EMIF) can interface to a variety of external devices, including:

- Supports industry standard synchronous devices (Pipelined and Flow Through SBSRAM, ZBT SRAM, Late Write SRAM, Sync FIFO, FWFT FIFO) and asynchronous devices (RAM, ROM and Flash).
- Up to 128MB asynchronous address range over 2chip selects
- Programmable read and write latencies for each synchronous chip select
- Programmable asynchronous cycle timings for each asynchronous memory chip select
- Supports extended waits for asynchronous devices
- Supports Select Strobe mode for asynchronous devices
- Supports TI DSP HPI interface on asynchronous interface
- Supports hold interface
- Supports little endian
- Supports Module Enable/Disable capability

A block diagram of the DSP is shown in Figure 1. In this document, the term EMIF refers to the EMIFA of the device.

The EMIF services requests of the external bus from on-chip masters such as the enhanced direct-memory access (EDMA) controller and the C64x+ Megamodule. On-chip masters place requests to the EMIF through the switched control resource (SCR). For more information on the SCR, see the device data manual.

Figure 1. DSP Block Diagram



2 EMIF Interface Signals

The EMIF signals of the DSP are shown in Figure 2 and described in the following tables. The EMIF has the following features:

- A 64-bit Internal data bus which is used to configure the module's registers and access data
- A 16-bit External data bus that is used to interface 8 or 16 bit wide memory devices
- An output clock, AECLKOUT, generated internally based on the EMIF input clock. You can select one of the following two clocks as the EMIF input clock source at device reset: internal SYSCLK4 or external AECLKIN. All of the memories interfacing with the EMIF should operate using AECLKOUT(EMIF clock cycle). The AECLKOUT frequency equals the EMIF input clock frequency.
- A programmable synchronous interface allowing glueless interfaces to synchronous devices such as ZBT SRAM, Late Write SRAM, and Pipelined and Flow-Through SBSRAM devices. Interfaces to synchronous FIFOs are also supported with the addition of external logic.
- A configurable asynchronous interface allowing interfaces to asynchronous devices such as SRAM, EPROM, and Flash, as well as FPGA and ASIC designs.
- Two EMIF spaces (CE2-3) reserved for either asynchronous or synchronous memory accesses.

Note: When the EMIF input clock source is chosen an internal SYSCLK4, the actual clock frequency would be SYSCLK4 divide by 2.

Figure 2. EMIF Interface Signals

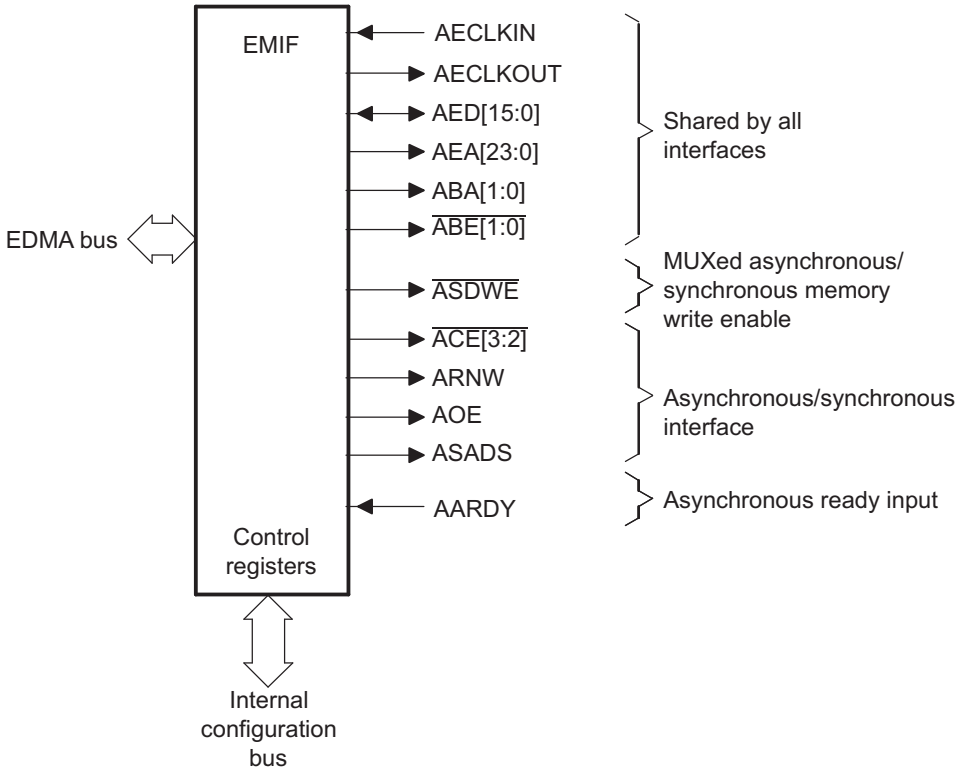


Table 1. EMIF Pins Used to Access All Device Types

Pin	I/O/Z	Description
AECLKIN	I	EMIF external input clock
AED[15:0]	I/O/Z	EMIF 16-bit data bus I/O
AEA[23:0]	O/Z	External address output for EMIF
ABA[1:0]	O/Z	Address outputs for async/sync interface when the data bus is configured as 8- or 16-bits wide.
$\overline{\text{ABE[1:0]}}$	O/Z	Active-low byte enables. Byte enables go active for only the appropriate byte lane for both writes and reads.

Table 2. EMIF Pins Specific to Asynchronous Devices

Pin	I/O/Z	Description
$\overline{\text{ACE2}}$	O/Z	Active-low chip select for memory space CE2
$\overline{\text{ACE3}}$	O/Z	Active-low chip select for memory space CE3
AARDY	I	Active-high asynchronous ready input used to insert wait states for slow memories and peripherals
ARNW	O/Z	Read/write enable for asynchronous memory interface
AOE	O/Z	Active-low output enable for asynchronous memory interface
$\overline{\text{ASDWE}}$	O/Z	Active-low write strobe for asynchronous memory interface

Table 3. EMIF Pins Specific to Synchronous Devices

Pin	I/O/Z	Description
AECLKOUT	O/Z	EMIF output clock at EMIF input clock (AECLKIN or SYSCLK4) frequency
$\overline{\text{ACE2}}$	O/Z	Active-low chip select for memory space CE2
$\overline{\text{ACE3}}$	O/Z	Active-low chip select for memory space CE3
ARNW	O/Z	Read/write enable for asynchronous memory interface
AOE	O/Z	Programmable synchronous interface output enable
ASADS	O/Z	Synchronous memory address strobe or read enable. The R_ENABLE field in the CE*n* Configuration Register (CE*n*CFG) selects ASADS: If R_ENABLE = 0, then signal functions as ASADS. If R_ENABLE = 1, then signal functions as ASRE.
$\overline{\text{ASDWE}}$	O/Z	Synchronous memory write enable

3 Memory Width and Byte Alignment

The EMIF supports memory widths of 8 and 16 bits. The EMIF automatically performs packing and unpacking for accesses to external memories of less than the requested transfer length. Only little-endian formats are supported.

Figure 3 shows the byte lane used by the EMIF for each of the bus size configurations. The external memory is always right aligned to the AED[7:0] side of the bus.

Table 4 summarizes the addressable memory ranges, as well as the internal address bus to external address bus translation for each of the supported memory types.

Figure 3. Byte Alignment By Bus Size Configuration

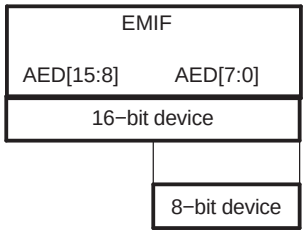


Table 4. Addressable Memory Ranges and Internal to External Address Bus Translation

Memory Type	Memory Width	Maximum Addressable Units per CE Space	Internal Address Bus to External Address Bus Translation	Represents
Async memory	×8	64M	AEA[23:0] = A[25:2] ABA[1:0] = A[1:0]	Byte address
	×16	32M	AEA[23:0] = A[24:1] ABA1 = A0	Halfword address
Programmable sync memory	×8	64M	AEA[23:0] = A[25:2] ABA[1:0] = A[1:0]	Byte address
	×16	32M	AEA[23:0] = A[24:1] ABA1 = A1	Halfword address

4 Asynchronous Interface

The asynchronous interface offers configurable memory cycle types to interface to a variety of memory and peripheral types, including SRAM, EPROM, and flash memory, as well as FPGA and ASIC designs.

The EMIF allows access widths of 8 or 16 bits on CE2 space and only 16-bits on CE3 space, as shown in the ASIZE description of the CE*n* Configuration Register (CE*n*CFG) register. The asynchronous interface signals on the EMIF are combined with the SBSRAM memory interface. To avoid bus contention, a programmable turnaround time also allows you to control the minimum number of cycles between a read followed by a write (same or different CE spaces), or between reads from different CE spaces (see Section 6).

Table 5 lists the asynchronous interface pins. Figure 4, Figure 5 show EMIF interfaces to 8-bit and 16-bit ROMs, respectively.

Table 5. Asynchronous Interface Signal Descriptions

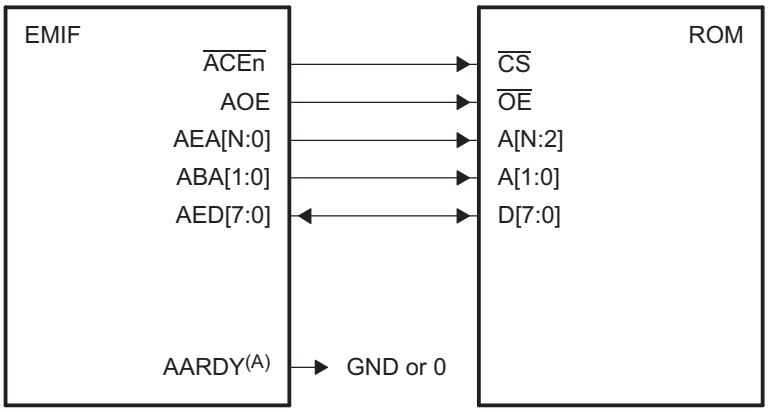
EMIF Signal	Function
$\overline{\text{ACE}}$[3:2]	Active-low chip select
AED[15:0]	16-bit data bus I/O
AEA[23:0]	External address output
ABA[1:0]	Bank select outputs or address outputs
AOE	Active-low output enable for asynchronous memory interface
$\overline{\text{ASDWE}}$	Active-low write strobe for asynchronous memory interface
ARNW	Read/write control for asynchronous memory interface
AARDY	Asynchronous ready. Input used to insert wait states into the memory cycle.

A See Section 4.1 for information on memory addressing.

B For interface to a 16-bit data bus: $\overline{\text{ABE}}$[1:0] and AED[15:0] are used.
For interface to an 8-bit data bus: $\overline{\text{ABE0}}$ and AED[7:0] are used.

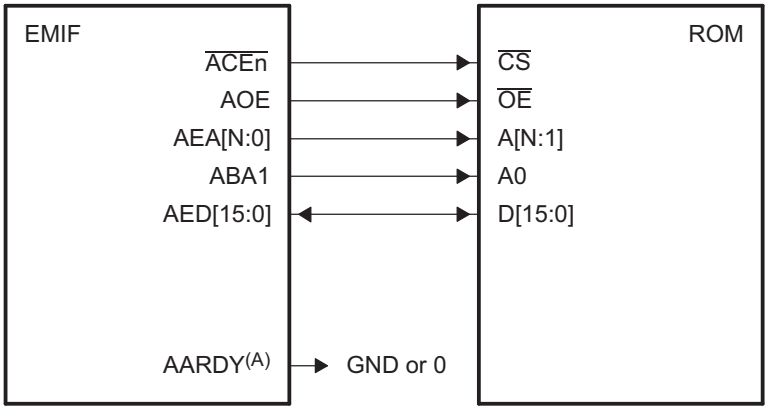
C The disabled state of the read input (AARDY) can be configured using the WP bit of the Asynchronous Wait Cycle Configuration register.

Figure 4. EMIF-to-8-Bit ROM Interface Block Diagram



A The disabled state of the read input (AARDY) can be configured using the WP bit of the Asynchronous Wait Cycle Configuration register.

Figure 5. EMIF-to-16-Bit ROM Interface Block Diagram



A The disabled state of the read input (AARDY) can be configured using the WP bit of the Asynchronous Wait Cycle Configuration register.

4.1 Asynchronous Interface Addressing

The EMIF uses the AEA[23:0] and ABA[1:0] pins to define the address bus that connects to memory devices. For 8 and 16-bit devices, AEA[23:0] always carries the least significant bits of a 32-bit address.

The functionality of ABA[1:0] depends on the width of the addressed device, as follows:

-
- For 16-bit devices, ABA1 defines bit 0 of the device's address; ABA0 is not used and should be left unconnected.
- For 8-bit devices, ABA[1:0] define bits 1 and 0 of the device's address.

4.2 Programmable ASRAM Parameters

The EMIF allows a high degree of programmability for shaping asynchronous accesses. The programmable parameters are:

- **Setup:** The time between the beginning of a memory cycle ($\overline{ACE}$ low, address valid) and the activation of $\overline{ASDWE}$ (writes) or AOE (reads).
- **Strobe:** The time between the activation and deactivation of $\overline{ASDWE}$ (writes) or AOE (reads). The read and write strobe period must not be programmed to be less than two AECLKOUT cycles when AE = 1 (AARDY extends the strobe cycle).
- **Hold:** The time between the deactivation of $\overline{ASDWE}$ (writes) or AOE (reads) and the end of the cycle, which can be either an address change or the deactivation of the $\overline{ACE}$ signal.

These parameters are programmed in terms of AECLKOUT cycles. Separate setup, strobe, and hold timing parameters are available for read and write accesses. Minimum values for ASRAM are:

- SETUP ≥1
- STROBE ≥1 (must be greater than or equal to 2 when AARDY is used)
- HOLD ≥1

4.3 Asynchronous Reads

Figure 6 shows an asynchronous read with the AARDY signal always disabled. The disabled state of AARDY depends on the setting of WP. The R_SETUP, R_STROBE, and R_HOLD parameters are programmed with the values 1, 2, and 0, respectively. An asynchronous read proceeds as follows:

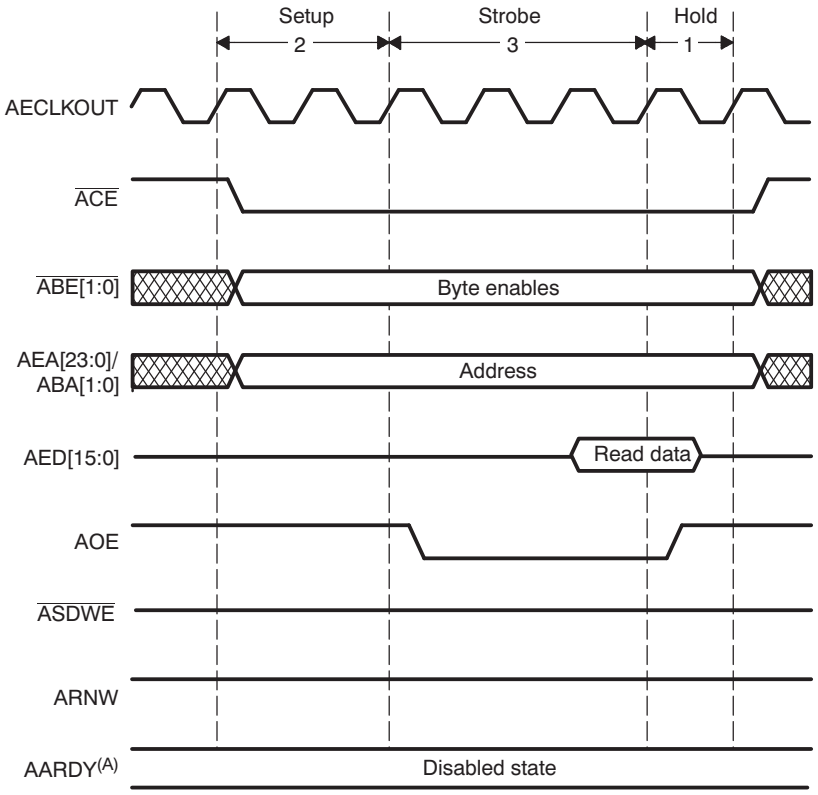
- At the beginning of the setup period:
  - $\overline{ACEn}$ becomes active, if not already active from a previous access.
  - ABE[1:0] become active.

  - AEA[23:0] and ABA[1:0] become valid.
- At the beginning of a strobe period, AOE becomes active.
- At the beginning of a hold period:
  - AOE becomes inactive (high).
  - Data is sampled on the AECLKOUT rising edge concurrent with the beginning of the hold period (the end of the strobe period).
- At the end of the hold period:
  - $\overline{\text{ACEn}}$ becomes inactive only if another read or write access to the same ACEn space is not pending.
  - ABE[1:0] become inactive.
  - AEA[23:0] and ABA[1:0] become invalid.

The AARDY pin can be activated by the external device to extend the strobe period, giving it more time to provide the data. See Section 4.5 for details on using the AARDY pin.

To avoid bus contention, a programmable turnaround time also allows you to control the minimum number of cycles between a read followed by a write (same or different CE spaces), or between reads from different CE spaces (see Section 6).

Figure 6. Asynchronous Read Timing Diagram



A The disabled state of the AARDY pin depends on the setting of the WP bit in the Asynchronous Wait Cycle Configuration Register.

4.4 Asynchronous Writes

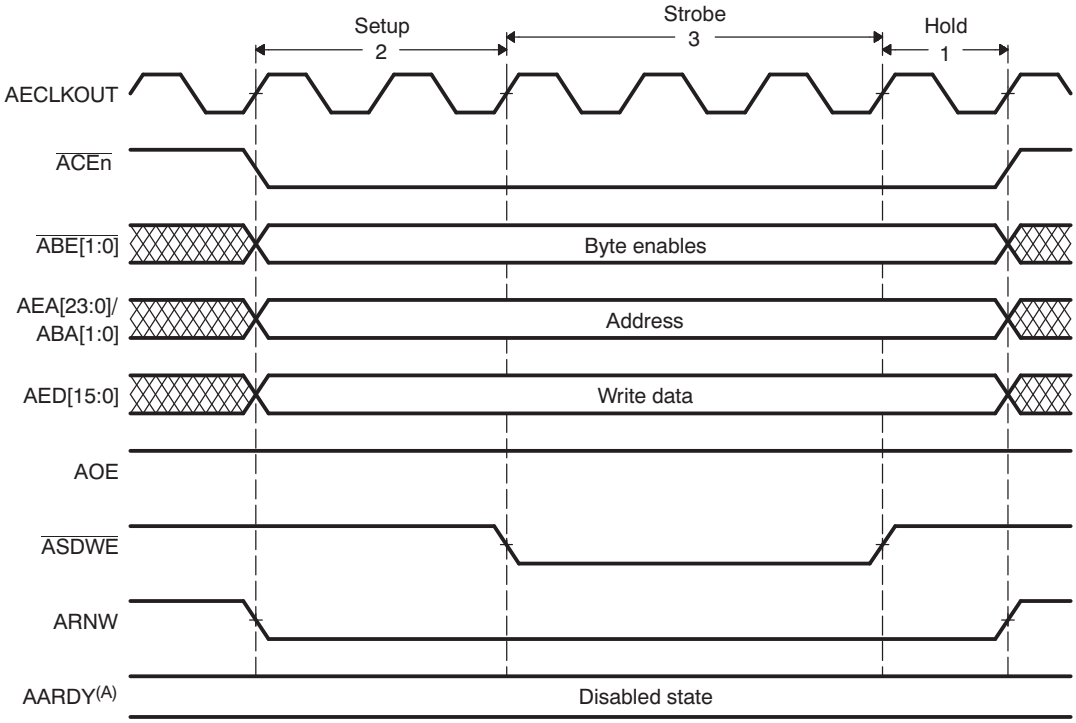
Figure 7 shows an asynchronous write cycle with the AARDY signal always disabled. The disabled state of AARDY depends on the setting of WP. The R_SETUP, R_STROBE, and R_HOLD parameters are programmed to 1, 2, and 0, respectively. An asynchronous write proceeds as:

- At the beginning of the setup period:
  - $\overline{\text{ACE[n]}}$ becomes active, if not already active from a previous access.
  - $\overline{\text{ABE[1:0]}}$ become valid.

  - AEA[23:0] and ABA[1:0] become valid.
  - AED is driven.
  - R/$\overline{W}$ becomes active (low).
- At the beginning of a strobe period, $\overline{ASDWE}$ becomes active.
- At the beginning of a hold period, $\overline{ASDWE}$ becomes inactive.
- At the end of the hold period:
  - AEA[23:0] and ABA[1:0] become invalid.
  - AED[15:0] becomes invalid.
  - $\overline{ACEn}$ becomes inactive (if no additional read or write accesses to the same CE*n* space are pending).

To avoid bus contention, a programmable turnaround time also allows you to control the minimum number of cycles between a read followed by a write (same or different CE spaces), or between reads from different CE spaces (see Section 6).

Figure 7. Asynchronous Write Timing Diagram



A The disabled state of the AARDY pin depends on the setting of the WP bit in the Asynchronous Wait Cycle Configuration Register.

4.5 Ready Input

The EMIF external asynchronous devices may assert control over the length of the strobe period through the use of the ready input (AARDY) pin. The AARDY pin can be activated by setting the AE bit in the CE*n* Configuration Register (CE*n*CFG). When this bit is set, the EMIF monitors the AARDY pin to determine if the attached device wishes to extend the strobe period of the current access cycle beyond the programmed number of clock cycles.

The AARDY pin must be asserted by the second rising edge of the AECLKOUT pin before the end of the programmed strobe period to be registered by the EMIF. When the EMIF detects that the AARDY pin has been asserted, it will begin inserting extra strobe cycles into the operation until the AARDY pin is

deactivated by the external device. The AARDY pin must be held in the asserted and de-asserted states for a minimum of two AECLKOUT cycles to be synchronized inside of the EMIF. In addition to the two cycles of internal synchronization to de-assert AARDY, the EMIF will insert two more wait cycles before returning to the last cycle of the programmed strobe period. The operation will proceed as usual at this point.

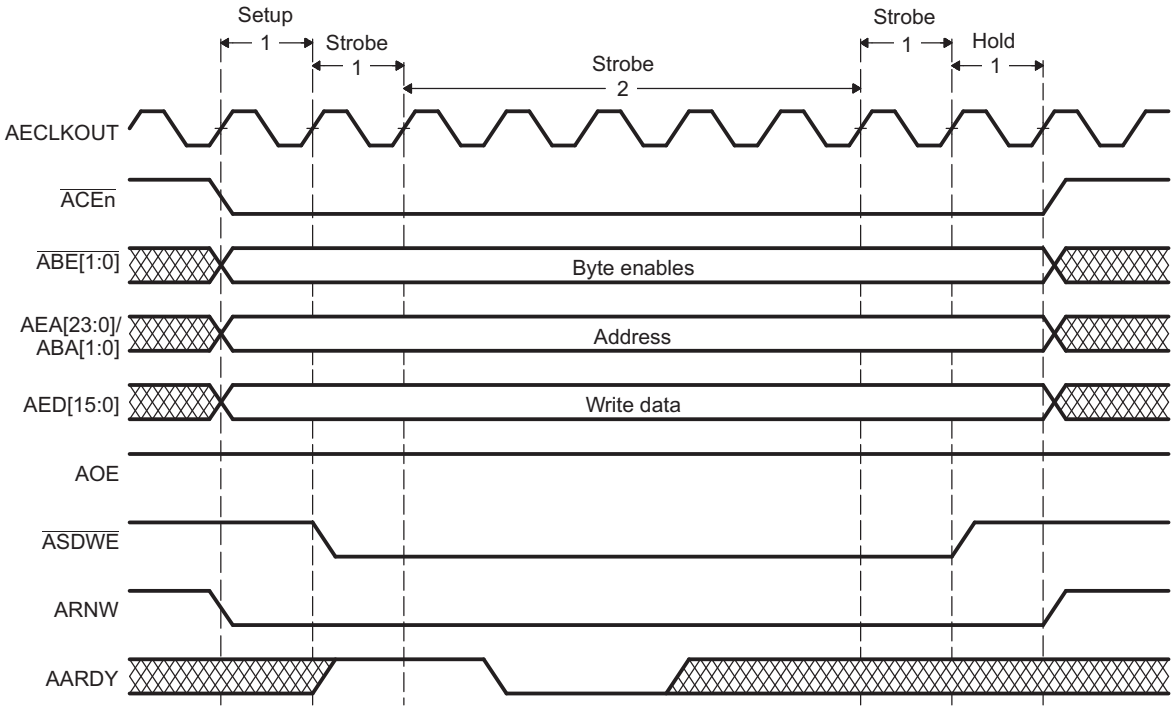
The AARDY pin cannot be used to extend the strobe period indefinitely. The programmable MAX_EXT_WAIT field in the Asynchronous Wait Cycle Configuration register (AWCC) determines the maximum number of AECLKOUT cycles the strobe period may be extended beyond the programmed length. When the counter expires, the EMIF proceeds to the hold period of the operation, regardless of the state of the AARDY pin. The EMIF can also generate an interrupt upon expiration of this counter. See Section 4.8 for details on enabling this interrupt.

For the AARDY pin to function properly, the WP bit of AWCC must be programmed to match the polarity used by the external device. In its reset state of 1, the EMIF will insert wait cycles when the AARDY pin is sampled high. When set to 0, the EMIF will insert wait cycles only when AARDY is sampled low. This programmability allows for a glueless connection to a larger variety of synchronous devices.

Finally, a restriction is placed on the strobe period timing parameters when using the AARDY pin. Specifically, the W_STROBE and R_STROBE fields must not be set to 0 for proper operation.

Figure 8 shows an example of extending a write operation using the AARDY pin, and Figure 9 shows a similar case for a read operation.

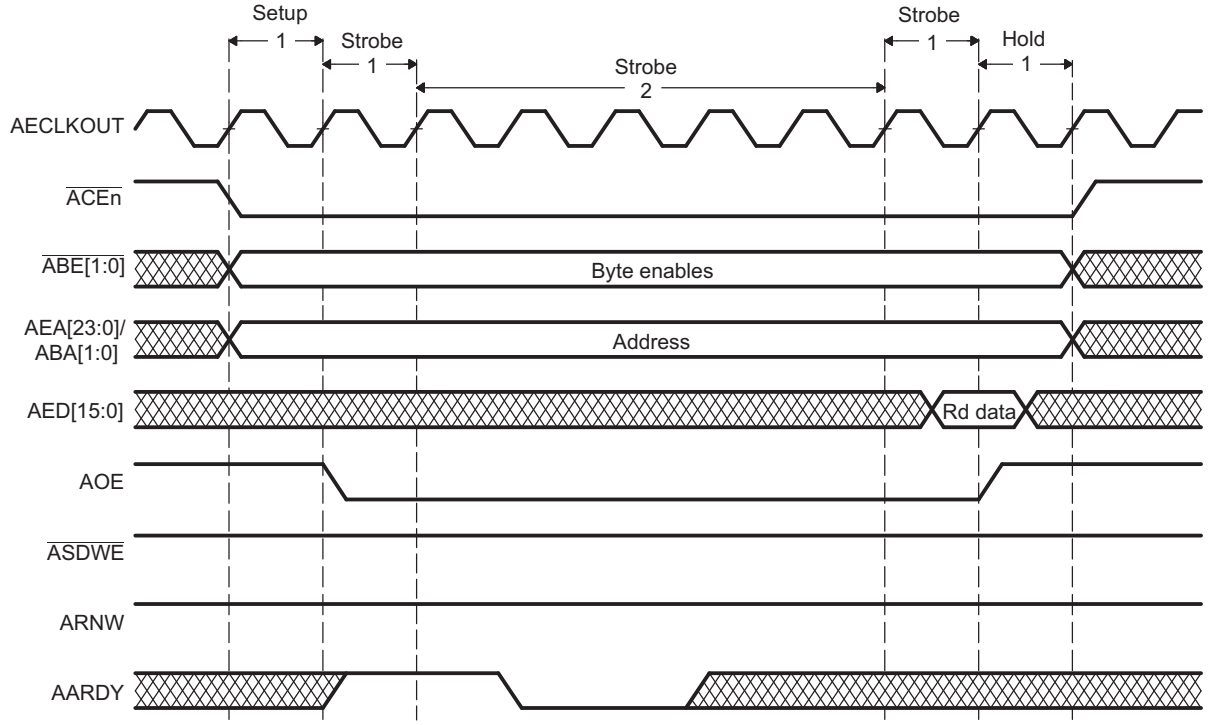
Figure 8. Asynchronous Write Timing Diagram Using Ready Input



A In this figure:

- SSEL = 0, SS = 0, BWEM = 0, and AE = 1 in CE*n* Configuration register
- WP = 1 in Asynchronous Wait Cycle Configuration register
- W_SETUP = 0, W_STROBE = 1, and W_HOLD = 0 in CE*n* Configuration register

Figure 9. Asynchronous Read Timing Diagram Using Ready Input



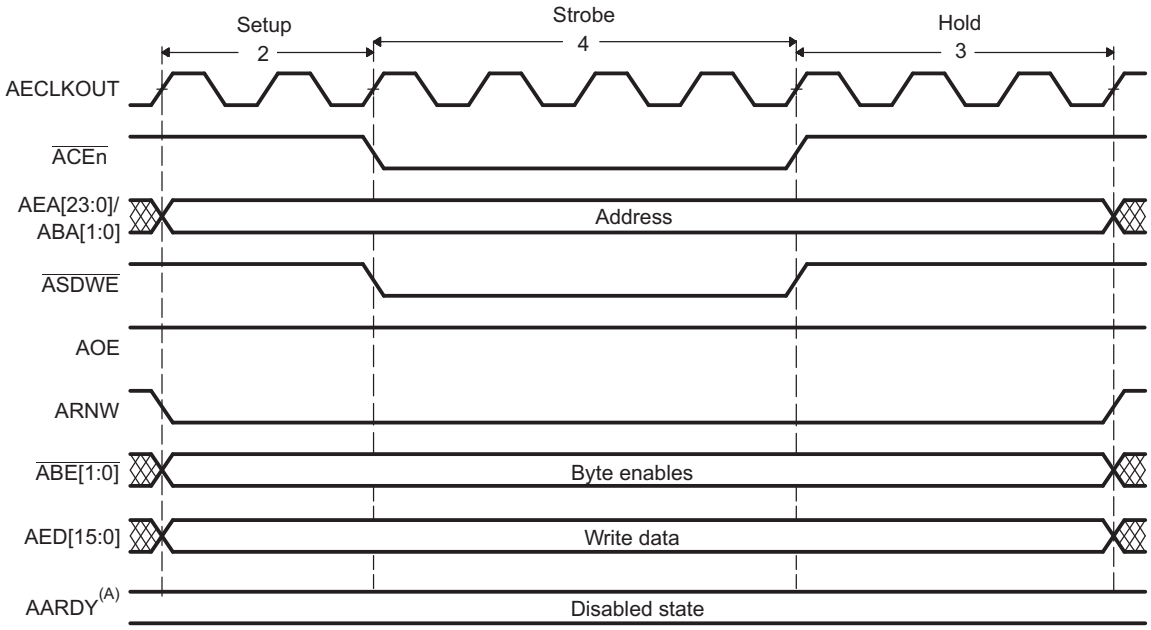
A In this figure:

- SSEL = 0, SS = 0, BWEM = 0, and AE = 1 in CE*n* Configuration register
- WP = 1 in Asynchronous Wait Cycle Configuration register
- W_SETUP = 0, W_STROBE = 1, and W_HOLD = 0 in CE*n* Configuration register

4.6 Asynchronous Memory Access in Select Strobe Mode

If the SS bit in the Asynchronous Wait Cycle Configuration register for a particular chip select is set, that chip select acts as a strobe. In other words, the timing of ACEn is the same as ASDWE and AOE. The value of the BWEM field in the Asynchronous Wait Cycle Configuration register is ignored in this mode; i.e., the ABE[1:0] pins act as byte enables. See Figure 10 and Figure 11.

Figure 10. Asynchronous Write in Select Strobe Mode

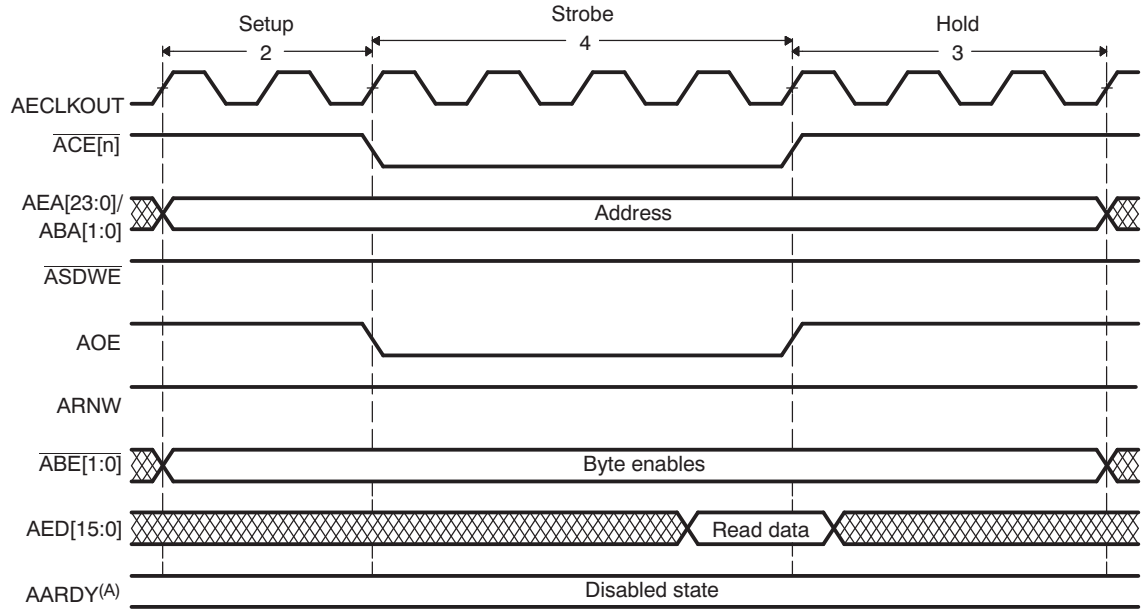


A The disabled state of the AARDY pin depends on the setting of the WP bit in the Asynchronous Wait Cycle Configuration Register.

B In this figure:

- SSEL = 0, SS = 1, BWEM = 0, and AE = 0 in CE*n* Configuration register
- W_SETUP = 1, W_STROBE = 3, and W_HOLD = 2 in CE*n* Configuration register
- AARDY is set to its inactive state

Figure 11. Asynchronous Read in Select Strobe Mode



A The disabled state of the AARDY pin depends on the setting of the WP bit in the Asynchronous Wait Cycle Configuration Register.

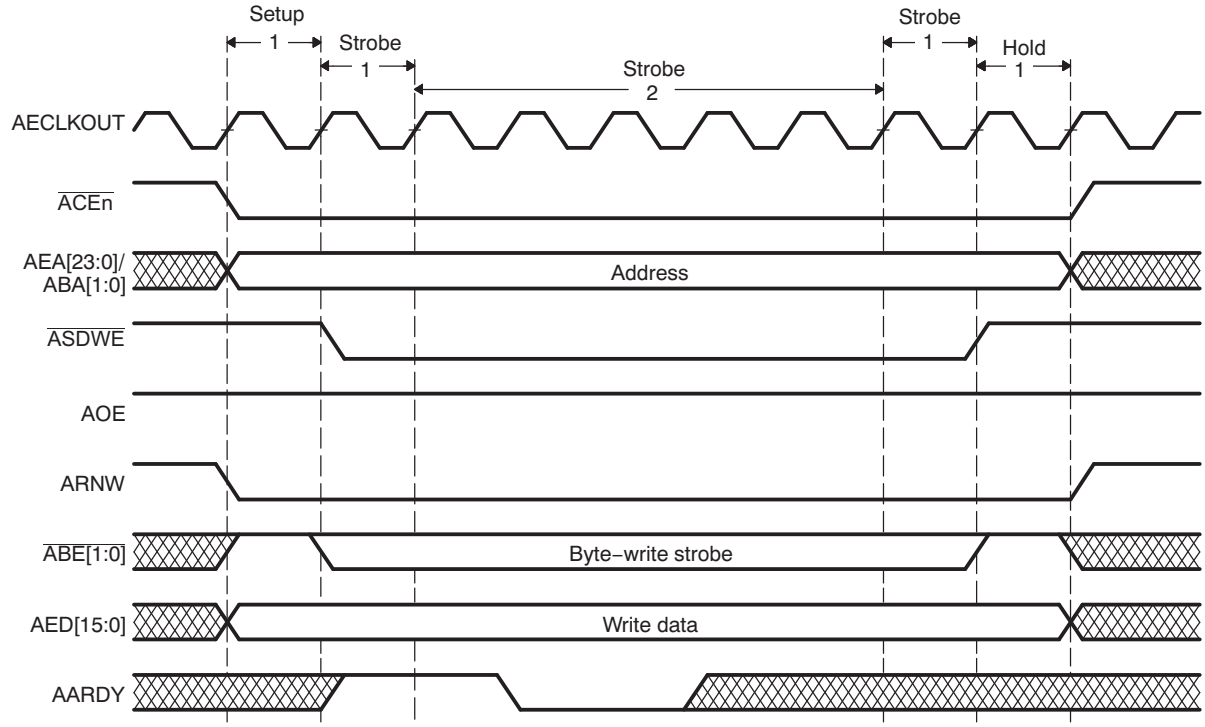
B In this figure:

- SSEL = 0, SS = 1, BWEM = 0, and AE = 0 in CE*n* Configuration register
- R_SETUP = 1, R_STROBE = 3, and R_HOLD = 2 in CE*n* Configuration register
- AARDY is set to its inactive state

4.7 Asynchronous Memory Access in WE Strobe Mode

The $\overline{\text{ABE}}$[1:0] pins act as write strobes when the WE Strobe Mode is enabled (the BWEM bit in Asynchronous Wait Cycle Configuration register is set). See Figure 12 and Figure 13. The WE Strobe Mode is useful when combining multiple 8-bit devices to create a 16-bit data bus. This mode allows the EMIF to perform byte writes to a group of 8-bit devices which do not have byte enable inputs. In this configuration, the byte enable pins are connected to the write strobes of the two 8-bit devices. This mode cannot be used when in the Select Strobe mode, as the Select Strobe mode overrides this mode.

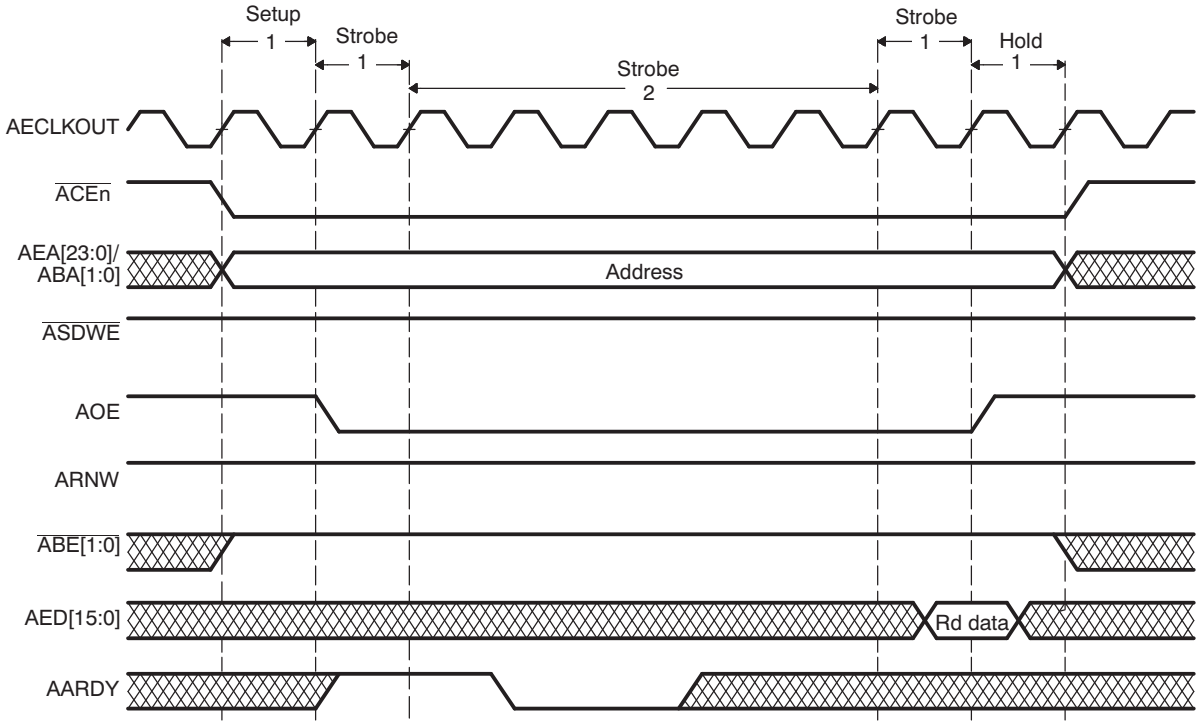
Figure 12. Asynchronous Write in WE Strobe Mode



A In this figure:

- SSEL = 0, SS = 0, BWEM = 1, and AE = 1 in CE*n* Configuration register
- WP = 1 in Asynchronous Wait Cycle Configuration register
- W_SETUP = 0, W_STROBE = 1, and W_HOLD = 0 in CE*n* Configuration register

Figure 13. Asynchronous Read in WE Strobe Mode



A In this figure:

- SSEL = 0, SS = 0, BWEM = 1, and AE = 1 in CE*n* Configuration register
- WP = 1 in Asynchronous Wait Cycle Configuration register
- R_SETUP = 0, R_STROBE = 1, and R_HOLD = 0 in CE*n* Configuration register

4.8 Asynchronous Time-Out Interrupt

The EMIF can generate an asynchronous time-out interrupt to the CPU when the attached device fails to de-assert the AARDY pin within the number of cycles defined in the MAX_EXT_WAIT field of the Asynchronous Wait Cycle Configuration register (AWCC). This interrupt is enabled by writing a 1 to the AT_MASK_SET bit of the Interrupt Mask Set register (INTMSKSET) and is disabled by writing a 2 to the AT_MASK_CLR field of the Interrupt Mask Clear register (INTMSKCLR). Both AT_MASK_SET and AT_MASK_CLR bits read 1 if the interrupt is enabled and read 0 if the interrupt is disabled.

Two other bits monitor the status of each interrupt. The AT bit of the Interrupt Raw register (INTRAW) is set when an asynchronous time-out occurs, regardless of whether or not the interrupt has been enabled. The AT_MASKED bit of the Interrupt Masked register (INTMSK) is set when an asynchronous time-out occurs and the interrupt has been enabled. The AT_MASKED bit will not be set if the interrupt is disabled. Writing a 1 to either the AT bit or the AT_MASKED bit will clear both bits.

Table 6 contains a summary of the interrupt monitor and control bit fields. See Section 12 for complete details on the register fields.

Table 6. Interrupt Monitor and Control Bit Fields

Bit Name	Register Name	Description
AT	Interrupt Raw register (INTRAW)	This bit is always set when an asynchronous time-out occurs.
AT_MASKED	Interrupt Masked register (INTMSK)	This bit is only set when an asynchronous time-out occurs and the interrupt has been enabled by writing a 1 to AT_MASK_SET.
AT_MASK_SET	Interrupt Mask Set register (INTMSKSET)	Writing a 1 to this bit enables the asynchronous time-out interrupt.

Table 6. Interrupt Monitor and Control Bit Fields (continued)

Bit Name	Register Name	Description
AT_MASK_CLR	Interrupt Mask Clear register (INTMSKCLR)	Writing a 1 to this bit disables the asynchronous time-out interrupt.

5 Programmable Synchronous Interface

The programmable synchronous interface of the EMIF supports glueless interfaces to the following devices:

- Pipelined and flow-through SBSRAM
- Zero bus turnaround (ZBT) synchronous pipeline SRAM, Late Write SRAM

The programmable synchronous interface can also interface to Standard Synchronous FIFOs with the addition of glue logic.

The bit fields in the CE*n* Configuration Registers (CE*n*CFG) control the cycle timing parameters for programmable synchronous interface synchronization. See Section 12.4 for a description of the CE*n*CFG registers. To avoid bus contention, a programmable turnaround time also allows you to control the minimum number of cycles between a read followed by a write (same or different CE spaces), or between reads from different CE spaces (see Section 6).

Table 7 shows the programmable synchronous interface pins.

Table 7. Programmable Synchronous Interface Pins

EMIF Signal	Signal Function
$\overline{\text{ACE}}$[3:2]	Chip select
AED[15:0]	64-bit data bus I/O
AEA[23:0]	External address output
ABA[1:0]	Bank select outputs or address outputs
$\overline{\text{ABE}}$[1:0]	Byte enable
ASADS	Synchronous memory address strobe or read enable
AOE	Synchronous memory output enables
$\overline{\text{ASDWE}}$	Synchronous memory write enable
AECLKOUT	EMIF output clock at EMIF input clock (AECLKIN or SYSCLK4) clock frequency

5.1 Programmable Synchronous Interface Addressing

The EMIF uses the AEA[23:0] and ABA[1:0] pins to define the address bus used to connect to memory devices.

The functionality of ABA[1:0] depends on the width of the device being addressed, as follows:

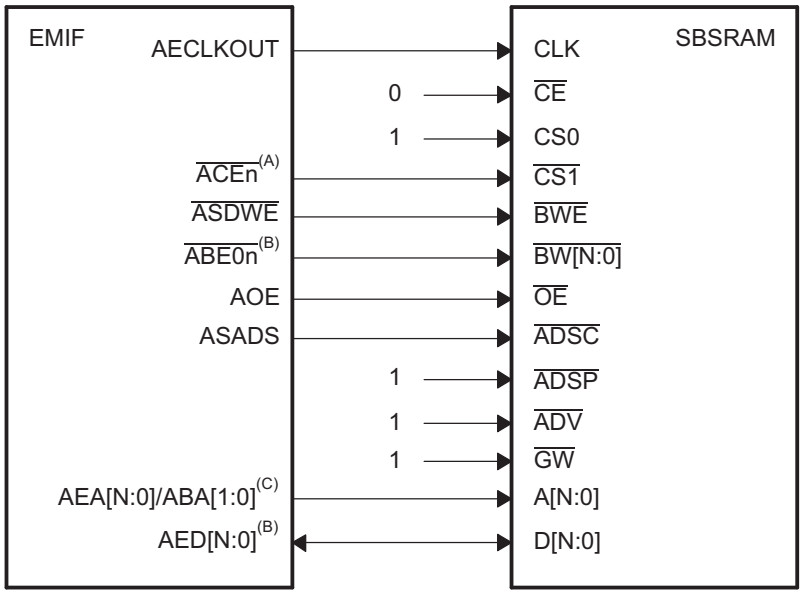
- For 16-bit devices, ABA1 defines bit 0 of the device's address; ABA0 is not used and should be left unconnected.
- For 8-bit devices, ABA[1:0] define bits 1 and 0 of the device's address.

5.2 SBSRAM Interface

The programmable synchronous mode supports the SBSRAM interface shown in Figure 14. The EMIF interface does not explicitly make use of the burst mode of the SBSRAM (the ADV signal on the SBSRAM is tied high). Instead, the EMIF performs SBSRAM bursts by issuing a new command every cycle. At the end of a burst where no accesses are pending in that CE space, the EMIF issues a deselect cycle. The R_ENABLE field in CE*n*CFG should be cleared for the SBSRAM interface to enable the ASADS signal.

The EMIF also supports programmable read and write latency, which allows it to interface with different types of synchronous memories.

Figure 14. EMIF-to-SBSRAM Interface Block Diagram



A Only ACE[3:2] can be used for synchronous memory interfaces.
For interface to a 16-bit data bus: ABE[1:0] and AED[15:0] are used.
For interface to an 8-bit data bus: ABE0 and AED[7:0] are used.

B See Section 5.1 for information on memory addressing.

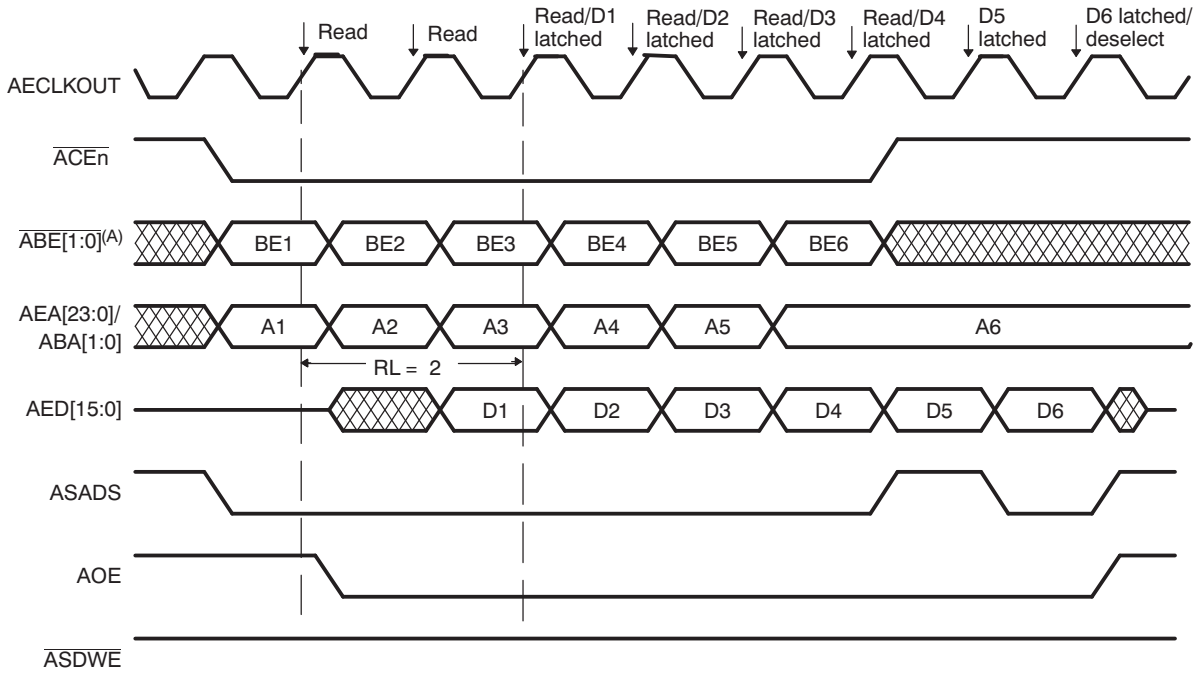
5.2.1 SBSRAM Read

Figure 15 shows an SBSRAM read cycle. The EMIF issues a read command to the SBSRAM by asserting ACEn and ASADS low. The EMIF provides the memory address on AEA/ABA on each successive cycle. The data mask on ABE is always driven high. The AOE is asserted one cycle before the programmed read latency for the chip select. A deselect command is issued by driving ACEn high and ASADS low on the clock cycle during the last data out. The AOE pin is de-asserted after the deselect cycle.

For the standard SBSRAM interface, set the following fields in the CE*n* Configuration Register (CE*n*CFG, Section 12.4):

- SSEL = 1; to configure chip select for synchronous memory
- R_LTNCY = 10b; 2 cycle read latency
- W_LTNCY = 00b; 0 cycle write latency
- CE_EXT = 0; ACEn goes inactive after the final command has been issued
- R_ENABLE = 0; ASADS pin acts as ASADS signal.

Figure 15. SBSRAM Six-Element Read Timing Diagram



A ABE[1:0] is driven during reads only if the RD_BE_EN bit of the CE*n* Configuration register is set to 1. ABE[1:0] stays high during reads if RD_BE_EN = 0.

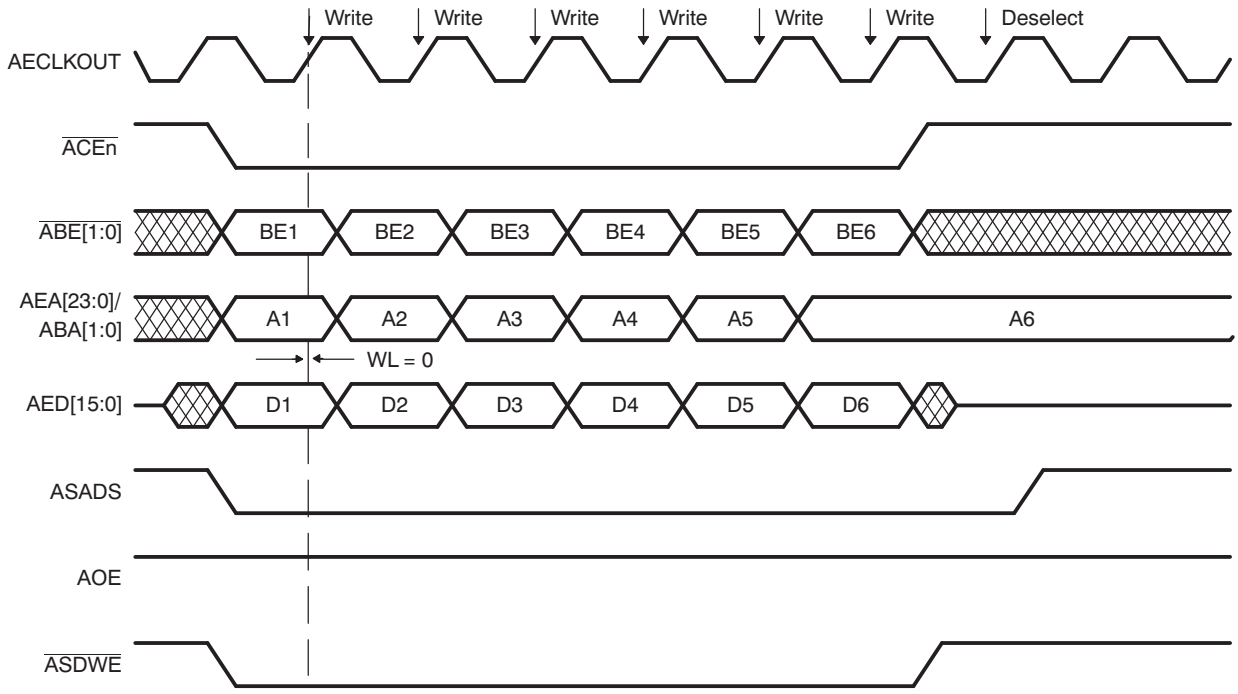
5.2.2 SBSRAM Write

Figure 16 shows an SBSRAM write cycle. The EMIF issues a write command to the SBSRAM by asserting ACEn, ASADS, and ASDWE low. The address, byte enables, and write data are presented to the memory on AEA/ABA, ABE, and AED, respectively, on each cycle without any latency on the write data. A deselect command is issued by de-asserting the ACE signal after the write burst is complete.

For the standard SBSRAM interface, set the following fields in CE*n*CFG to their default state:

- SSEL = 1; to configure chip select for synchronous memory
- R_LTNCY = 10b; 2 cycle read latency
- W_LTNCY = 00b; 0 cycle write latency
- CE_EXT = 0b; ACEn goes inactive after the final command has been issued
- R_ENABLE = 0; ASADS pin acts as ASADS signal.

Figure 16. SBSRAM Six-Element Write Timing Diagram

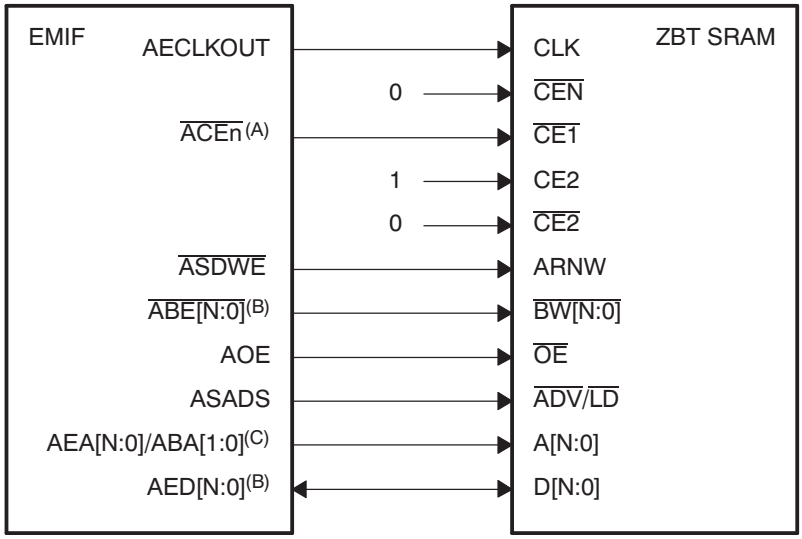


5.3 Zero Bus Turnaround (ZBT) SRAM Interface

The programmable synchronous mode supports the zero bus turnaround (ZBT) SRAM interface shown in Figure 17. For the ZBT SRAM interface, set the following fields in the CE*n* Configuration Register (CE*n*CFG):

- SSEL = 1; to configure chip select for synchronous memory
- R_LTNCY = 10b; 2 cycle read latency
- W_LTNCY = 10b; 2 cycle write latency
- CE_EXT = 0b; $\overline{\text{ACEn}}$ goes inactive after the final command has been issued
- R_ENABLE = 0b; ASADS signal

Figure 17. EMIF-to-Zero Bus Turnaround (ZBT) SRAM Interface Block Diagram



A Only ACE[3:2] can be used for synchronous memory interfaces.

B
For 16-bit interface, ABE[1:0] and AED[15:0] are used.
For 8-bit interface, ABE[0] and AED[7:0] are used.

C See Section 5.1 for information on memory addressing.

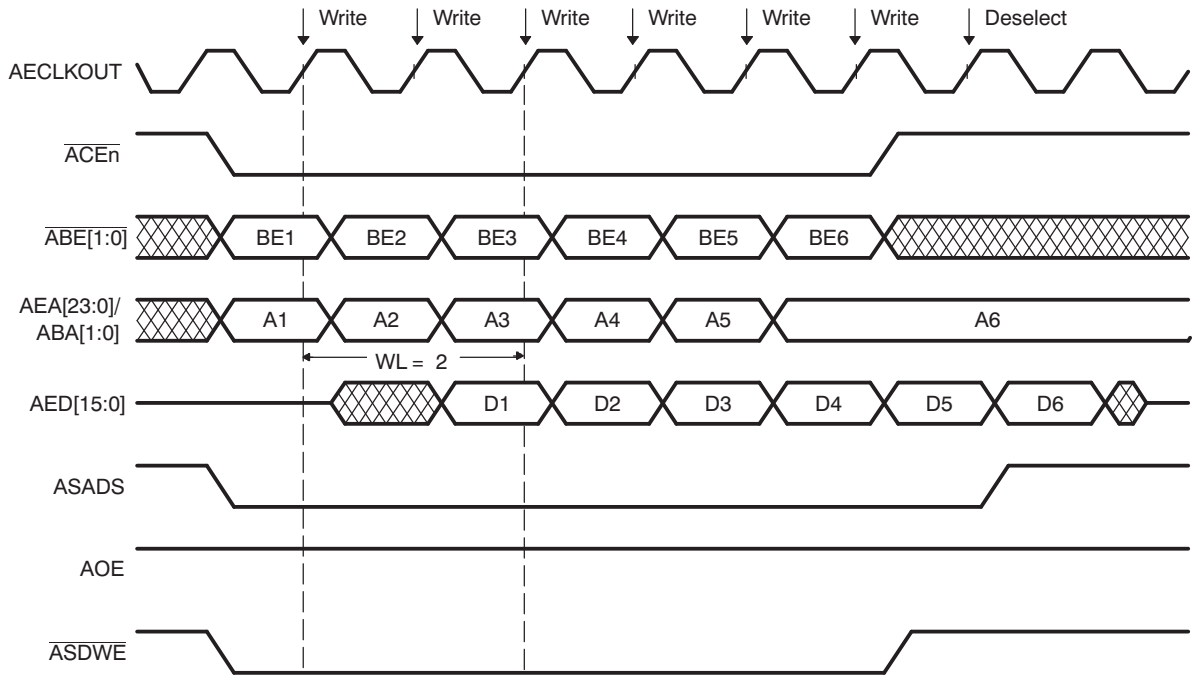
5.3.1 ZBT SRAM Read

The ZBT SRAM read waveforms are identical to the SBSRAM read waveforms, as the register settings corresponding to the reads are identical. See Section 5.2.1 for details.

5.3.2 ZBT SRAM Write

For ZBT SRAM writes, the control signal waveforms are the same as standard SRAM writes. However, the write data is delayed by two cycles, as controlled by W_LTNCY = 10b. Figure 18 shows the ZBT SRAM write timing.

Figure 18. Zero Bus Turnaround (ZBT) SRAM Six-Element Write Timing Diagram



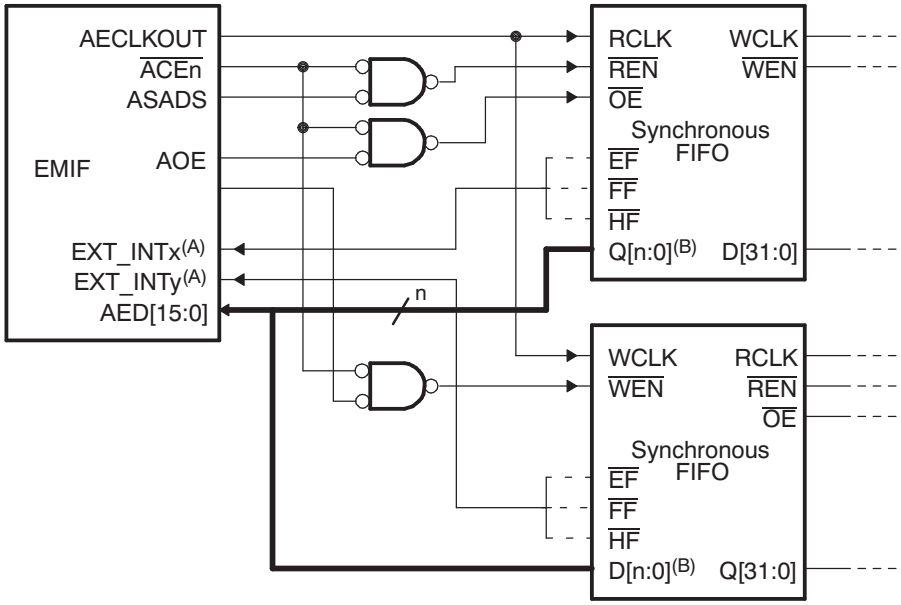
5.4 Synchronous FIFO Interface

The programmable synchronous mode supports interface to standard timing synchronous FIFOs with the addition of glue logic as shown in Figure 19. For a synchronous FIFO interface, set the following fields in the CE*n* Configuration Register (CE*n*CFG):

- R_ENABLE = 1b; ASADS pin acts as ASRE signal.

Figure 19 shows the synchronous FIFO interface with glue.

Figure 19. Read and Write Synchronous FIFO Interface With Glue Block Diagram



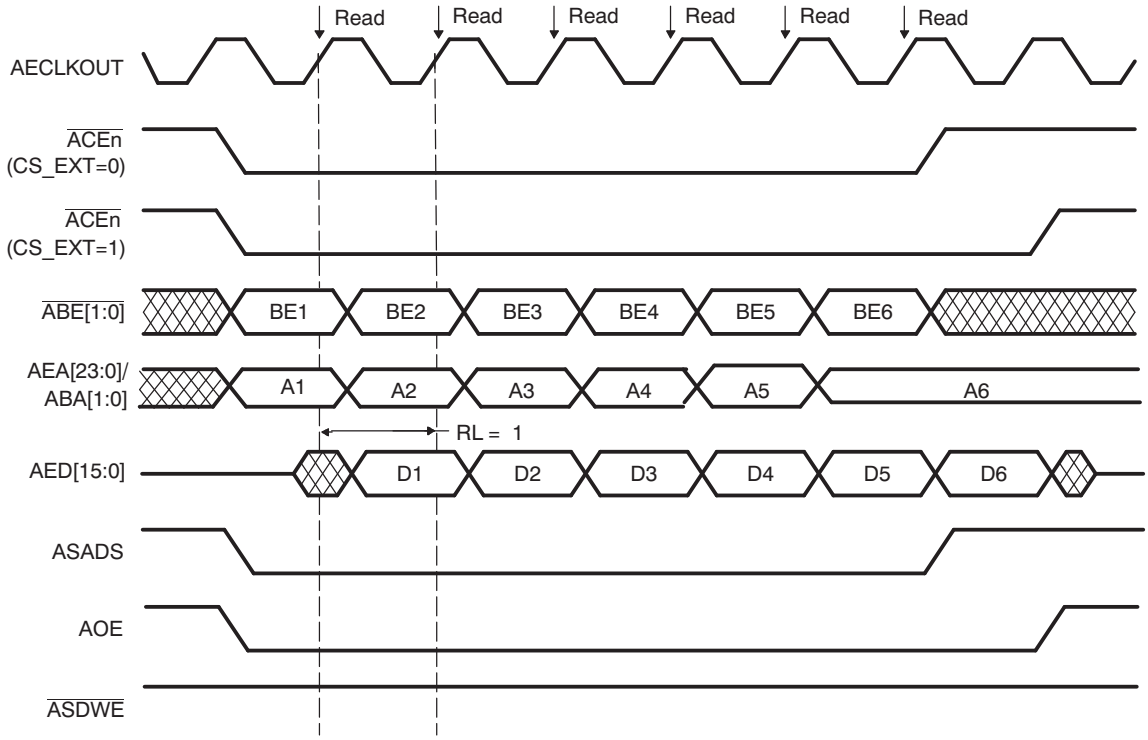
A GPIO pins on the device may be configured as external interrupt sources to the CPU. See the DSP General-Purpose Input/Output (GPIO) User's Guide for more details.

5.4.1 Standard Synchronous FIFO Read

Figure 20 shows a six-word read from a standard synchronous FIFO. The CE*n*CFG settings are:

- SSEL = 1; to configure chip select for synchronous memory
- R_LTNCY = 01b; 1 cycle read latency
- CE_EXT = 0; $\overline{\text{ACEn}}$ goes inactive after the final command has been issued
- CE_EXT = 1; during reads, $\overline{\text{ACEn}}$ goes active when AOE goes active and will stay active until AOE goes inactive
- R_ENABLE = 1; ASADS pin acts as ASRE signal.

Figure 20. Standard Synchronous FIFO Read Timing Diagram

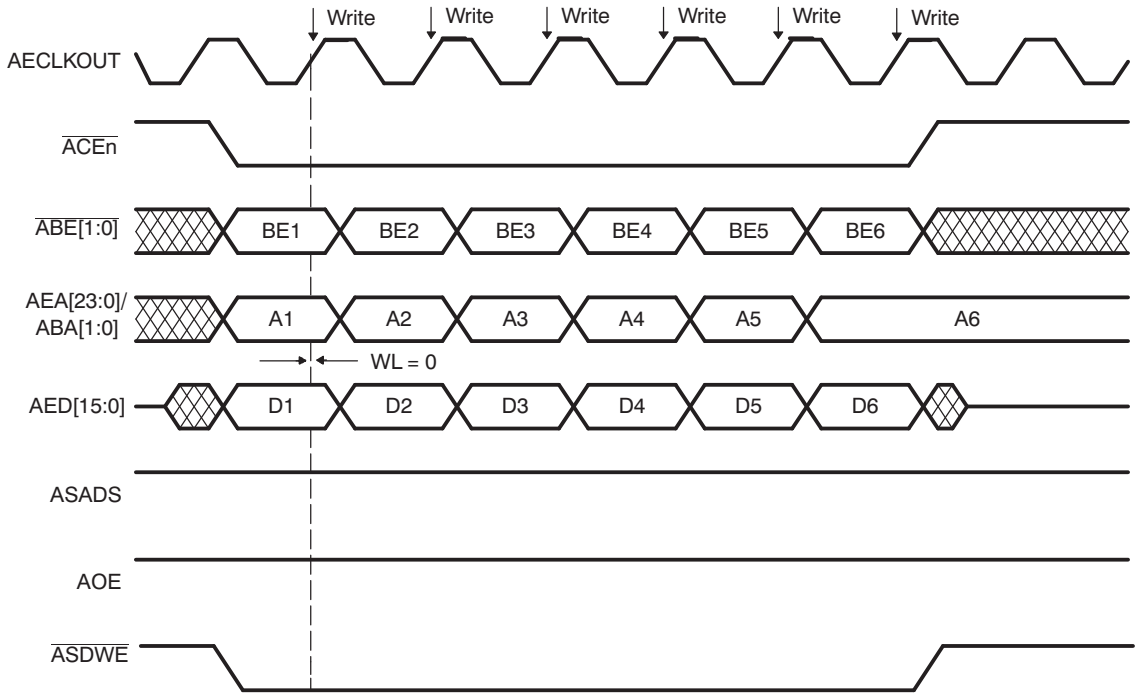


5.4.2 Standard Synchronous FIFO Write

Figure 21 shows a six-word write to a standard synchronous FIFO. The CE*n*CFG settings are:

- SSEL = 1; to configure chip select for synchronous memory
- CE_EXT = 0; $\overline{\text{ACEn}}$ goes inactive after the final command has been issued
- W_LTNCY = 00b; 0 cycle write latency
- R_ENABLE = 1b; ASADS pin acts as ASRE signal

Figure 21. Standard Synchronous FIFO Write Timing Diagram



6 Turnaround Time

To avoid bus contention, a programmable turnaround time also allows you to control the minimum number of cycles between a read followed by a write (same or different CE spaces), or between reads from different CE spaces. Table 8 shows the turnaround time introduced by the EMIF on the data bus for various back-to-back accesses. The turnaround time is programmed through the TA bits of the Asynchronous Wait Cycle Configuration register (AWCC).

Table 8. Turnaround Time

Previous Access	Next Access	Turnaround Time (Number of AECLKOUT cycles)
Async read/write	Any read/write	TA from Asynchronous Wait Cycle Configuration register
Sync read	Async read/write	TA from Asynchronous Wait Cycle Configuration register
	Sync read to same chip select	0
	Sync read to different chip select	1
	Sync write	1
Sync write	Async read/write	TA from Asynchronous Wait Cycle Configuration register
	Sync read	1
	Sync write	0

7 Command FIFO and Scheduling

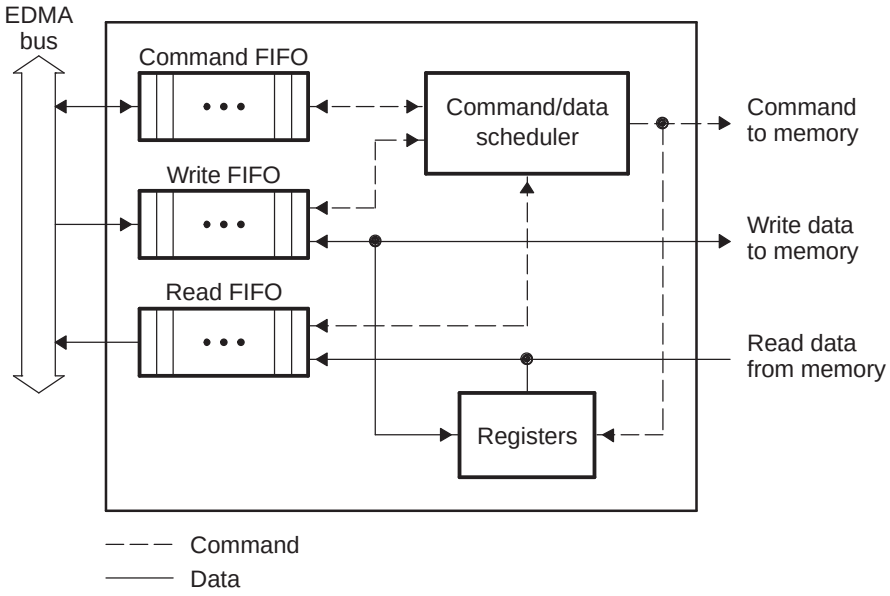
To move data efficiently from on-chip resources to external memory and vice versa, the EMIF makes use of six FIFO's, and two schedulers namely Command and Data. Table 9 describes the purpose of each FIFO.

Table 9. EMIF FIFO Description

FIFO	Description	Number of Entries (64-Bit Wide)
Command FIFO	Stores all commands coming from on-chip requesters	7
Write Data FIFO	Stores write data coming from on-chip requesters to be written to external memory	11
Read Data FIFO	Stores read data coming from external memory to be sent to on-chip requester	15
Write Status FIFO	Stores the write status information to be sent back to on-chip resources for each write transaction	7
Read Command FIFO	Stores all the read transactions to be issued to the on-chip resources	20
Reg Read FIFO	Stores the data read from memory mapped registers	4

Figure 22 shows the block diagram of the EMIF FIFOs. Commands, write data, and read data arrive at the EMIF parallel to each other. The VBUSM Interface is used to write and read data from external memory as well as internal memory-mapped registers.

Figure 22. EMIF FIFO Block Diagram



7.1 Command Ordering and Scheduling

The EMIF performs command re-ordering and scheduling in an attempt to achieve efficient transfers with maximum throughput. The goal is to maximize the utilization of external memory bandwidth.

The EMIF looks at all the commands stored in the Command FIFO to schedule commands to the external memory. For each master the EMIF reorders the commands based on the following rules:

- Selects the oldest command

- A read command is advanced before an older write command if the read is to a different block address (2048 bytes) and the read priority is equal to or greater than the write priority

Note: Most masters issue commands on a single priority level. Also, EDMA Transfer Controller Read and Write ports are considered different masters, and thus the above rule does not apply.

Once the commands from each master are reordered, the EMIF will have at most one pending read or write from each master. The EMIF then selects the highest priority read from the pending reads, and the highest priority write from the pending writes. If two or more commands have the highest priority, the EMIF selects the oldest command.

As a result, the EMIF may have a final read and a final write command. If the Read FIFO is not full, then the read command will be performed before the write command, otherwise the write command will be performed before the read command.

The following results from the above scheduling algorithm:

- All writes from a single master will complete in order
- All reads from a single master will complete in order
- From the same master, any read to the same location (or within 2048 bytes) as a previous write will complete in order

7.2 Command Starvation

The reordering and scheduling rules listed above may lead to command starvation, which is the prevention of certain commands from being processed by the EMIF. Command starvation can result when a continuous stream of high-priority read commands blocks a low-priority write command.

To avoid this condition, the EMIF momentarily raises the priority of the oldest command in the command FIFO after a set number of transfers have been made. The PRIO_RAISE field in the Burst Priority Register (BPRIO) sets the number of the transfers that must be made before the EMIF will raise the priority of the oldest command.

Note: Leaving the PRIO_RAISE bits at their default value (FFh) disables this feature of the EMIF. This means commands can stay in the command FIFO indefinitely. Therefore, these bits should be set to FEh immediately following reset to enable this feature with the highest level of allowable memory transfers. It is suggested that system-level prioritization be set to avoid placing high-bandwidth masters on the highest priority levels. These bits can be left as FEh unless advanced bandwidth/prioritization control is required.

7.3 Possible Race Condition

A race condition may exist when certain masters write data to the EMIF. For example, if master A passes a software message via a buffer in external memory and does not wait for indication that the write completes, when master B attempts to read the software message, the Master B Read may bypass the Master A write, and thus Master B may read stale data and therefore receive an incorrect message.

Some master peripherals (e.g., EDMA3 controller) will always wait for the write to complete before signaling an interrupt to the system, thus avoiding this race condition. For masters who do not have hardware guarantee of write-read ordering, it may be necessary to guarantee data ordering via software. If master A does not wait for indication that a write is complete, it must perform the following workaround:

1. Perform the required write.
2. Perform a dummy write to the EMIF module ID and revision register.
3. Perform a dummy read to the EMIF module ID and revision register.
4. Indicate to master B that the data is ready to be read after completion of the read in step 3. The completion of the read in step 3 ensures that the previous write was done.

For a list of the master peripherals that need this workaround, please see the device-specific data manual.

8 Resetting the EMIF

The EMIF can be reset through a hard reset or a soft reset. A hard reset resets the state machine, the FIFOs, and the internal registers. A soft reset only resets the state machine and the FIFOs. A soft reset does not reset the internal registers except for the interrupt registers. Register accesses cannot be performed while either reset is asserted.

The EMIF hard and soft reset are derived from device-level resets. There are several types of device-level resets: power-on reset, warm reset, max reset, system reset, and CPU reset. Table 10 shows the relationship between the device-level resets and the EMIF resets. See the device data manual for more information on the device-level resets.

Table 10. Device and EMIF Reset Relationship

EMIF Reset	Effect	Initiated By:
Hard reset	Resets control logic and all EMIF registers	Power on reset Warm reset Max reset
Soft reset	Resets control logic and interrupt registers	System reset CPU reset

9 EMIF Emulation

The EMIF continues operating during emulation halts in order to allow emulation access to external memory.

10 EMIF Pin Muxing

EMIF pins are multiplexed with VideoPort3(VP3),VideoPort4(VP4) and some BootConfig pins. User needs to program the Pin Mux Register appropriately in order to use the EMIF interface.

11 EMIF Clocking

The EMIF module has a LPSC associated with it that controls the EMIF module reset and clock gating to the EMIF module. On power-up the LPSC associated with the EMIF does not gate the clock to the EMIF module. User would need to program the registers of the LPSC associated with EMIF appropriately to enable the clock to be gated to the EMIF module, before trying a data transfer using EMIF interface.

12 EMIF Registers

Control of the EMIF and the memory interfaces it supports is maintained through memory-mapped registers within the EMIF. Table 11 lists the memory-mapped registers of the EMIF. See the device-specific datasheet for the memory address of these registers.

Table 11. EMIF Registers

Offset	Acronym	Register Name	Section
0000	MIDR	Module ID and Revision Register	Section 12.1
0004	STAT	Status Register	Section 12.2
0020	BPRIO	Burst Priority Register	Section 12.3
0080-008C	CE*n*CFG	CE*n* Configuration Registers (SSEL = 0)	Section 12.4
0080-008C	CE*n*CFG	CE*n* Configuration Registers (SSEL = 1)	Section 12.5
00A0	AWCC	Asynchronous Wait Cycle Configuration Register	Section 12.6
00C0	INTRAW	Interrupt RAW Register	Section 12.7
00C4	INTMSK	Interrupt Masked Register	Section 12.8
00C8	INTMSKSET	Interrupt Mask Set Register	Section 12.9

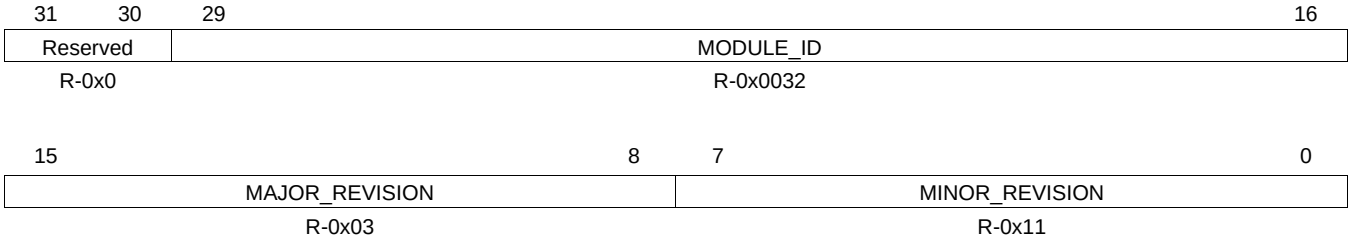
Table 11. EMIF Registers (continued)

Offset	Acronym	Register Name	Section
00CC	INTMSKCLR	Interrupt Mask Clear Register	Section 12.10

12.1 Module ID and Revision Register (MIDR)

This register reflects the latest changes made to the memory controller. The Module ID and Revision Register (MIDR) is shown in Figure 23 and described in Table 12.

Figure 23. Module ID and Revision Register (MIDR)



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

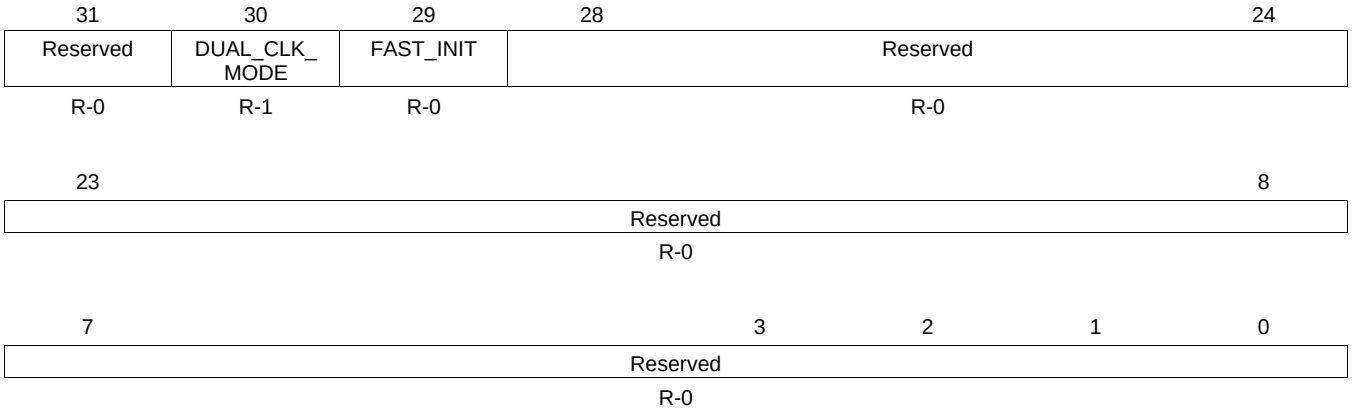
Table 12. Module ID and Revision Register (MIDR) Field Descriptions

Bit	Field	Value	Description
31-30	Reserved	0	Reserved
29-16	MODULE_ID		EMIF module ID
15-8	MAJOR_REVISION		Major revision
7-0	MINOR_REVISION		Minor revision

12.2 Status Register (STAT)

This register reflects the configuration of the EMIF. The Status Register (STAT) is shown in Figure 24 and described in Table 13.

Figure 24. Status Register (STAT)



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

Table 13. Status Register (STAT) Field Descriptions

Bit	Field	Value	Description
31	Reserved		Reserved
30	DUAL_CLK_MODE		Dual Clock Mode. Reflects the value on dual_clk_mode port that defines whether mclk and vclk are asynchronous
		1	mclk and vclk are asynchronous
		0	mclk and vclk are synchronous
29	FAST_INIT		Reflects the value of FAST_INIT port that defines if EMIFA is initialized for fast init mode
		1	EMIFA initialized for fast init mode
		0	EMIFA is not initialized for fast init mode
28-0	Reserved		Reserved

12.3 Burst Priority Register (BPRIO)

This register is used to temporarily raise priority of old commands. The Burst Priority Register (BPRIO) is shown in Figure 25 and described in Table 14.

Figure 25. Burst Priority Register (BPRIO)

31	16
Reserved	
R-0x0	

15	8	7	0
Reserved		PR_OLD_COUNT	
R-0x0		RW-0xFF	

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

Table 14. Burst Priority Register (BPRIO) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved		Reserved.
7-0	PR_OLD_COUNT		Number of memory transfers after which the EMIF will elevate the priority of the oldest command in the command FIFO. Setting this field to FFh disables this feature, thereby allowing old commands to stay in the FIFO indefinitely.
		0	1 memory transfer
		1	2 memory transfers
		2	3 memory transfers
		3-FEh	4-FFh memory transfers
		FFh	Feature disabled, commands can stay in command FIFO indefinitely.

12.4 CE*n* Configuration Registers (CE*n*CFG) if SSEL = 0

These registers select the access type (asynchronous or synchronous) and configure the access parameters for the CE*n* space. The contents of the CE*n* Configuration Registers (CE*n*CFG) are interpreted according to the SSEL bit. Figure 26 and Table 15 describe the CE*n*CFG registers when SSEL = 0.

Figure 26. CE*n* Configuration Registers (CE*n*CFG) if SSEL = 0

31	30	29	28	27 24	23 18	17 16
SSEL	SS	BWEM	AE	W_SETUP	W_STROBE	W_HOLD
RW-0x0	RW-0x0	RW-0x0	RW-0x0	RW-0xF	RW-0x3F	RW-0x7

15	14 11	10 5	4 2	1 0
W_HOLD	R_SETUP	R_STROBE	R_HOLD	ASIZE
RW-0x7	RW-0xF	RW-0x3F	RW-0x7	RW-0x0

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

Table 15. CE*n* Configuration Registers (CE*n*CFG) if SSEL = 0 Field Descriptions

Bit	Field	Value	Description
31	SSEL		Synchronous/asynchronous memory select. This bit specifies whether CE*n* is configured for synchronous or asynchronous memory accesses. The other fields in this table define the bits in the CE*n*CFG when SSEL is cleared to 0 (asynchronous memory mode).
		1	Synchronous memory mode
		0	Asynchronous memory mode
30	SS		Select Strobe mode enable. When set to 1, the CE*n* pin will have read and write strobe timing.
		1	Select strobe mode enabled
		0	Select strobe mode disabled
29	BWEM		WE Strobe mode enable. When set to 1, the ABE[7:0] output pins will act as active low byte write enables when accessing CE*n* space. When cleared to 0, the ABE[7:0] output pins will act as active low byte enables when accessing CE*n*.
		1	WE Strobe mode enabled
		0	WE Strobe mode disabled
28	AE		Asynchronous ready input enable. Set to 1 to enable the asynchronous ready (AARDY) input pin during accesses to the CE*n* space. When enabled, the AARDY pin can be used to extend the strobe period during asynchronous accesses.
		1	AARDY pin enabled
		0	AARDY pin disabled
27-24	W_SETUP		Write setup width. Number of AECLKOUT cycles from AEA[19:0], ABA[1:0], D[63:0], ABE[7:0], and CE*n* being set to $\overline{\text{ASDWE}}$ asserted, minus one cycle.
23-18	W_STROBE		Write strobe width. Number of AECLKOUT cycles for which $\overline{\text{ASDWE}}$ is held active, minus one cycle.
17-15	W_HOLD		Write hold width. Number of AECLKOUT cycles for which AEA[19:0], ABA[1:0], D[63:0], ABE[7:0], and $\overline{\text{ACE}}$n are held after $\overline{\text{ASDWE}}$ has been de-asserted, minus one cycle.
14-11	R_SETUP		Read setup width. Number of AECLKOUT cycles from AEA[19:0], ABA[1:0], ABE[7:0], and CE*n* being set to AOE asserted, minus one cycle.
10-5	R_STROBE		Read strobe width. Number of AECLKOUT cycles for which AOE is held active, minus one cycle.
4-2	R_HOLD		Read hold width. Number of AECLKOUT cycles for which AEA[19:0], ABA[1:0], ABE[7:0], and CE*n* are held after AOE has been de-asserted, minus one cycle.
1-0	ASIZE		Asynchronous Memory Size. Defines the width of the asynchronous device's data bus.
		00	8-bit data bus
		01	16-bit data bus
		10	32-bit data bus
		11	64-bit data bus

12.5 CE*n* Configuration Registers (CE*n*CFG) Field Descriptions if SSEL = 1

These registers select the access type (asynchronous or synchronous) and configure the access parameters for the CE*n* space. The contents of the CE*n* Configuration Registers (CE*n*CFG) are interpreted according to the SSEL bit. Figure 27 and Table 16 describe the CE*n*CFG registers when SSEL = 1.

Figure 27. CE*n* Configuration Registers (CE*n*CFG) Field Descriptions if SSEL = 1

31	16
SSEL	Reserved
RW-0x0	R-0x0

15	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		RD_BE_EN	CE_EXT	R_ENABLE	W_LTNCY		Reserved		R_LTNCY		SBSIZE	
R-0x0		RW-0x0	RW-0x0	RW-0x0	RW-0x0		R-0x0		RW-0x0		RW-0x0	

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

Table 16. CE*n* Configuration Registers (CE*n*CFG) Field Descriptions if SSEL = 1

Bit	Field	Value	Description
31	SSEL		Synchronous/asynchronous memory select. This bit specifies whether $\overline{\text{ACEn}}$ is configured for synchronous or asynchronous memory accesses. The other fields in this table define the bits in the CE*n*CFG when SSEL is set to 1 (synchronous memory mode).
		1	Synchronous memory mode
		0	Asynchronous memory mode
30-11	Reserved		Reserved
10	RD_BE_EN		Read Byte Enable. If set to 1, the byte enable pins (ABE[7:0]) are driven during asynchronous memory reads. If cleared to 0, ABE[7:0] stay high during synchronous memory reads. Not supported for R_LTNCY = 0.
		1	Byte enables are driven during synchronous memory reads.
		0	Byte enables stay high during synchronous memory reads.
9	CE_EXT		Synchronous Memory Chip Enable Extend. Defines the behavior of the $\overline{\text{ACEn}}$ pin during synchronous accesses.
		1	ACEn goes active when AOE goes active and will stay active until AOE goes inactive. The timing of AOE is controlled by R_LTNCY. Used for synchronous FIFO interfaces where $\overline{\text{ACE}}$ gates AOE.
		0	ACE*n* goes inactive after final command has been issued.
8	R_ENABLE		Synchronous Memory Address Strobe or Read Enable. Defines the behavior of the ASADS pin during synchronous accesses.
		1	The ASADS pin acts as ASRE. The ASADS pin goes low only for reads. No deselect command is issued. Used for FIFO interfaces.
		0	The ASADS pin acts as the ASADS signal. The ASADS pin goes active low for reads and writes. A deselect command is issued by driving ASADS active high after a command if no new command is pending for the same CE space. Used for SBSRAM and ZBT devices.
7-6	W_LTNCY		Synchronous Memory Write Latency. Defines the synchronous device's write latency in EMIF clock cycles.
		00	0 cycle write latency
		01	1 cycle write latency
		10	2 cycle write latency
		11	3 cycle write latency
5-4	Reserved		Reserved

Table 16. CE*n* Configuration Registers (CE*n*CFG) Field Descriptions if SSEL = 1 (continued)

Bit	Field	Value	Description
3-2	R_LTNCY		Synchronous Memory Read Latency. Defines the synchronous device's read latency in EMIF clock cycles. Read latency of 0 is not supported.
		01	1 cycle read latency
		10	2 cycle read latency
		11	3 cycle read latency
1-0	SBSIZE		Synchronous Memory Device Size. Defines the width of the synchronous device's data bus.
		00	8-bit data bus
		01	16-bit data bus
		10	32-bit data bus
		11	64-bit data bus

12.6 Asynchronous Wait Cycle Configuration Register (AWCC)

The Asynchronous Wait Cycle Configuration Register (AWCC) controls asynchronous memory access features such as the turn-around time between accesses and the asynchronous ready pin (AARDY) polarity. AWCC is shown in Figure 28 and described in Table 17.

Figure 28. Asynchronous Wait Cycle Configuration Register (AWCC)

31	30	29	16
Res	WP	Reserved	
R-0x0	RW-0x1	R-0x0	

15	11	10	8	7	0
Reserved		TA		MAX_EXT_WAIT	
R-0x0		RW-0x3		RW-0x80	

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

Table 17. Asynchronous Wait Cycle Configuration Register (AWCC) Field Descriptions

Bit	Field	Value	Description
31	Reserved		Reserved.
30	WP		Asynchronous ready (AARDY) pin polarity. Defines the polarity of the AARDY pin.
		1	AARDY pin is active-high (strobe period extended when AARDY is high)
		0	AARDY pin is active-low (strobe period extended when AARDY is low)
29-11	Reserved		Reserved.
10-8	TA		Turn Around cycles. Number of AECLKOUT cycles between the end of one asynchronous memory access and the start of another asynchronous memory access, minus one cycle.
7-0	MAX_EXT_WAIT		Maximum Extended Wait cycles. The value in this field defines the number of 16 EMIF cycle periods the EMIF will wait for an extended asynchronous cycle before the cycle is terminated.

12.7 Interrupt RAW Register (INTRAW)

This register displays the status of the EMIF interrupt regardless of whether or not the interrupt is enabled. The interrupt RAW register (INTRAW) is shown in Figure 29 and described in Table 18.

Figure 29. Interrupt RAW Register (INTRAW)

31	16
Reserved	
R-0x0	

15	2	1	0
Reserved	LT	Res	AT
R-0x0	RW-0x0	R-0x0	RW-0x0

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

Table 18. Interrupt RAW Register (INTRAW) Field Descriptions

Bit	Field	Value	Description
31-3	Reserved		Reserved.
2	LT		Line Trap. Set to 1 by hardware to indicate illegal memory access. Writing 1 will clear this bit as well as the lt_masked bit in Interrupt Masked Register. Writing 0 will have no effect.
		1	Line Trap, illegal memory access has occurred.
		0	Illegal memory access has not occurred.
1	Reserved		Reserved
0	AT		Asynchronous timeout interrupt. Set to 1 by the EMIF to indicate that the AARDY pin did not go inactive within the number of cycles defined by the MAX_EXT_WAIT field in the Async Wait Cycle Configuration register (AWCC). Writing a 1 will clear this bit as well as the AT_MASKED bit in the Interrupt Masked register (INTMSK). Writing a 0 has no effect.
		1	An asynchronous access timeout has occurred.
		0	An asynchronous access timeout has not occurred.

12.8 Interrupt Masked Register (INTMSK)

This register displays the status of the EMIF interrupt only when the interrupt is enabled. The interrupt masked register (INTMSK) is shown in Figure 30 and described in Table 19.

Figure 30. Interrupt Masked Register (INTMSK)

31	16
Reserved	
R-0	

15	8
Reserved	
R-0x0	

7	3	2	1	0
Reserved		LT_MASKED	Reserved	AT_MASKED
R-0		R/W-0	R-0	R/W-0

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

Table 19. Interrupt Masked Register (INTMSK) Field Descriptions

Bit	Field	Value	Description
31-3	Reserved		Reserved

Table 19. Interrupt Masked Register (INTMSK) Field Descriptions (continued)

Bit	Field	Value	Description
2	LT_MASKED		Masked Line Trap. Set to 1 by hardware to indicate illegal memory access type, only if lt_mask_set bit in Interrupt Mask Set Register is set to 1. Writing a 1 will clear this bit as well as lt bit in Interrupt Raw Register. Writing 0 has no effect.
		1	Line Trap, illegal memory access occurred.
		0	No illegal memory access occurred.
1	Reserved		Reserved
0	AT_MASKED		Asynchronous timeout interrupt masked. Set to 1 by the EMIF to indicate that the AARDY pin did not go inactive within the number of cycles defined by the MAX_EXT_WAIT field in the Asynchronous Wait Cycle Configuration register (AWCC) only if the AT_MASK_SET bit in the Interrupt Mask Set register is set to 1. Writing a 1 will clear this bit as well as the AT bit in the Interrupt Raw register (INTRAW). Writing a 0 has no effect.
		1	An asynchronous access timeout has occurred.
		0	An asynchronous access timeout has not occurred.

12.9 Interrupt Mask Set Register (INTMSKSET)

The interrupt mask set register (INTMSKSET) enables the EMIF interrupt. It is shown in Figure 31 and described in Table 20.

Figure 31. Interrupt Mask Set Register (INTMSKSET)

31	16
Reserved	
R-0	

15	8
Reserved	
R-0	

7	3	2	1	0
Reserved		LT_MASK_SET	Reserved	AT_MASK_SET
R-0		R/W-0	R-0	R/W-0

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

Table 20. Interrupt Mask Set Register (INTMSKSET) Field Descriptions

Bit	Field	Value	Description
31-3	Reserved		Reserved.
2	LT_MASK_SET		Mask set for LT_MASKED bit in the Interrupt Masked register. Writing a 1 to this bit will enable the interrupt and set this bit as well as the LT_MASK_CLR bit in the Interrupt Mask Clear register. Writing a 0 has no effect.
		1	Line Trap Interrupt is enabled
		0	No effect
1	Reserved		Reserved
0	AT_MASK_SET		Mask set for AT_MASKED bit in the Interrupt Masked register. Writing a 1 to this bit will enable the interrupt and set this bit as well as the AT_MASK_CLR bit in the Interrupt Mask Clear register. Writing a 0 has no effect.
		1	The asynchronous access timeout interrupt is enabled.
		0	No effect

12.10 Interrupt Mask Clear Register (INTMSKCLR)

The interrupt mask clear register (INTMSKCLR) disables the EMIF interrupt. It is shown in Figure 32 and described in Table 21.

Figure 32. Interrupt Mask Clear Register (INTMSKCLR)

31	16
Reserved	
R-0x0	

15	8
Reserved	
R-0x0	

7	3	2	1	0
Reserved		LT_MASK_CLR	Reserved	AT_MASK_CLR
R-0		R/W-0	R-0	R/W-0

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

Table 21. Interrupt Mask Clear Register (INTMSKCLR) Field Descriptions

Bit	Field	Value	Description
31-3	Reserved		Reserved
2	LT_MASK_CLR		Mask clear for LT_MASKED bit in the Interrupt Masked register. Writing a 1 to this bit will disable the interrupt and clear this bit as well as the LT_MASK_SET bit in the Interrupt Mask Set register. Writing a 0 has no effect.
		1	Disable the Line Trap interrupt .
		0	No effect
1	Reserved		Reserved
0	AT_MASK_CLR		Mask clear for AT_MASKED bit in the Interrupt Masked register. Writing a 1 to this bit will disable the interrupt and clear this bit as well as the AT_MASK_SET bit in the Interrupt Mask Set register. Writing a 0 has no effect.
		1	The asynchronous access timeout interrupt is disabled.
		0	No effect

Appendix A Revision History

Table A-1 lists the changes made since the previous version of this document.

Table A-1. Document Revision History

Reference	Additions/Modifications/Deletions	
Global	Changed signal names as follows:	
	Former Name	Current Name
	BA[1:0]	ABA[1:0]
	BE[1:0]	ABE[1:0]
	CE	$\overline{\text{ACE}}$
	EA[23:0]	AEA[23:0]
	ED[15:0]	AED[15:0]
	SDAS~/SRE~	ASADS
	SRE	ASRE
	AWE~/SWE~	$\overline{\text{ASDWE}}$
	R/W~	ARNW
	ECLKIN	AECLKIN
	ECLKOUT	AECLKOUT
	SDRAS	$\overline{\text{DDR\_RAS}}$
	SDCAS	$\overline{\text{DDR\_CAS}}$
	CS	$\overline{\text{DDR\_CS}}$
Section 2	Added two notes to Section 2.	
Figure 2	Changed Figure 2.	
Figure 3	Changed Figure 3.	
Figure 24	Changed Figure 24.	
Table 13	Changed Table 13.	
Figure 29	Changed Figure 29.	
Table 18	Changed Table 18.	
Figure 30	Changed Figure 30.	
Table 19	Changed Table 19.	
Figure 31	Changed Figure 31.	
Table 20	Changed Table 20.	
Figure 32	Changed Figure 32.	
Table 21	Changed Table 21.	
Table 4	Changed column head in Table 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
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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