OMAP5910 Dual-Core Processor Memory Interface Traffic Controller Reference Guide

Literature Number: SPRU673
October 2003

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 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.

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.

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
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
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
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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

Preface

Read This First

About This Manual

This document describes the OMAP5910 multimedia processor memory interface traffic controller (TC).

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.

Related Documentation From Texas Instruments

The following documents describe the OMAP5910 device and related peripherals. 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.

OMAP5910 Dual-Core Processor MPU Subsystem Reference Guide (literature number SPRU671)

OMAP5910 Dual-Core Processor DSP Subsystem Reference Guide (literature number SPRU672)

OMAP5910 Dual-Core Processor Memory Interface Traffic Controller Reference Guide (literature number SPRU673)

OMAP5910 Dual-Core Processor System DMA Controller Reference Guide (literature number SPRU674)

OMAP5910 Dual-Core Processor LCD Controller Reference Guide (literature number SPRU675)

OMAP5910 Dual-Core Processor Universal Asynchronous Receiver/Transmitter (UART) Devices Reference Guide (literature number SPRU676)

OMAP5910 Dual-Core Processor Universal Serial Bus (USB) and Frame Adjustment Counter (FAC) Reference Guide (literature number SPRU677)

OMAP5910 Dual-Core Processor Clock Generation and System Reset Management Reference Guide (literature number SPRU678)

OMAP5910 Dual-Core Processor General-Purpose Input/Output (GPIO) Reference Guide (literature number SPRU679)

OMAP5910 Dual-Core Processor MMC/SD Reference Guide (literature number SPRU680)

OMAP5910 Dual-Core Processor Inter-Integrated Circuit (I2C) Controller Reference Guide (literature number SPRU681)

OMAP5910 Dual-Core Processor Timer Reference Guide (literature number SPRU682)

OMAP5910 Dual-Core Processor Inter-Processor Communication Reference Guide (literature number SPRU683)

OMAP5910 Dual-Core Processor Camera Interface Reference Guide (literature number SPRU684)

OMAP5905 Dual-Core Processor Multichannel Serial Interface (MCSI) Reference Guide (literature number SPRU685)

OMAP5910 Dual-Core Processor Micro-Wire Interface Reference Guide (literature number SPRU686)

OMAP5910 Dual-Core Processor Real-Time Clock (RTC) Reference Guide (literature number SPRU687)

OMAP5910 Dual-Core Processor HDQ/1-Wire Interface Reference Guide (literature number SPRU688)

OMAP5910 Dual-Core Processor PWL, PWT, and LED Peripheral Reference Guide (literature number SPRU689)

OMAP5910 Dual-Core Processor Multichannel Buffered Serial Port (McBSP) Reference Guide (literature number SPRU708)

Trademarks

OMAP and the OMAP symbol are trademarks of Texas Instruments.

Contents

Figures

Tables

Memory Interface Traffic Controller

This document describes the OMAP5910 multimedia processor memory interface traffic controller (TC).

1 Introduction

The memory interface traffic controller (TC) manages all accesses by the MPU, the TMS320C55x DSP, the system DMA, and the local bus to the OMAP5910 system memory resources (SRAM, SDRAM, flash, ROM, etc.). The TC also manages accesses by the MPU or the USB host. The USB host is an internal OMAP5910 peripheral connected on the local bus, so the TC contributes to managing USB host accesses.

Figure 1 shows the OMAP5910 device with the traffic controller highlighted. Figure 2 shows the traffic controller in more detail. Table 1 lists the access modes and data access width of the controllers (MPU, C55x DSP, system DMA, and local bus).

Figure 1. TC Block Diagram

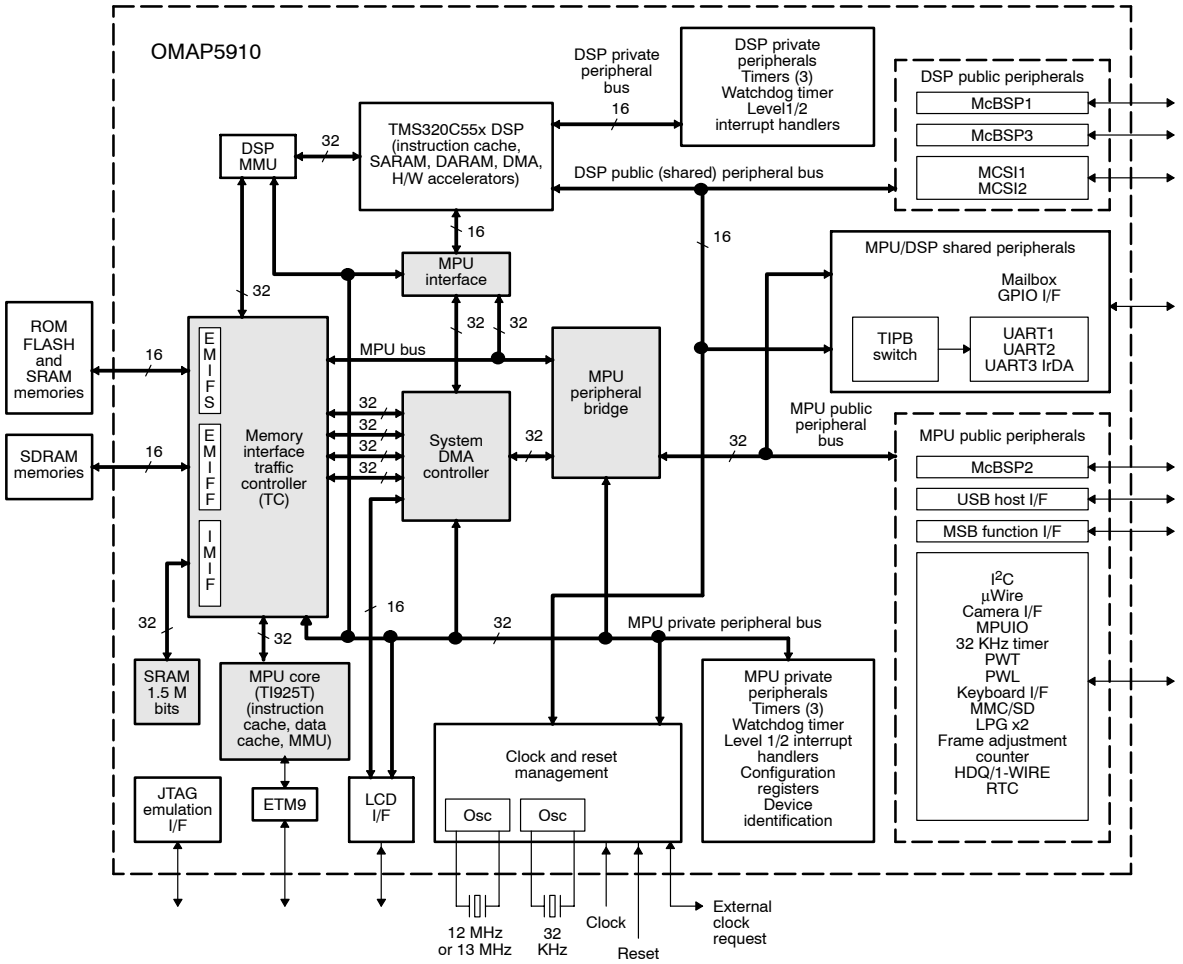


Figure 2. Traffic Controller

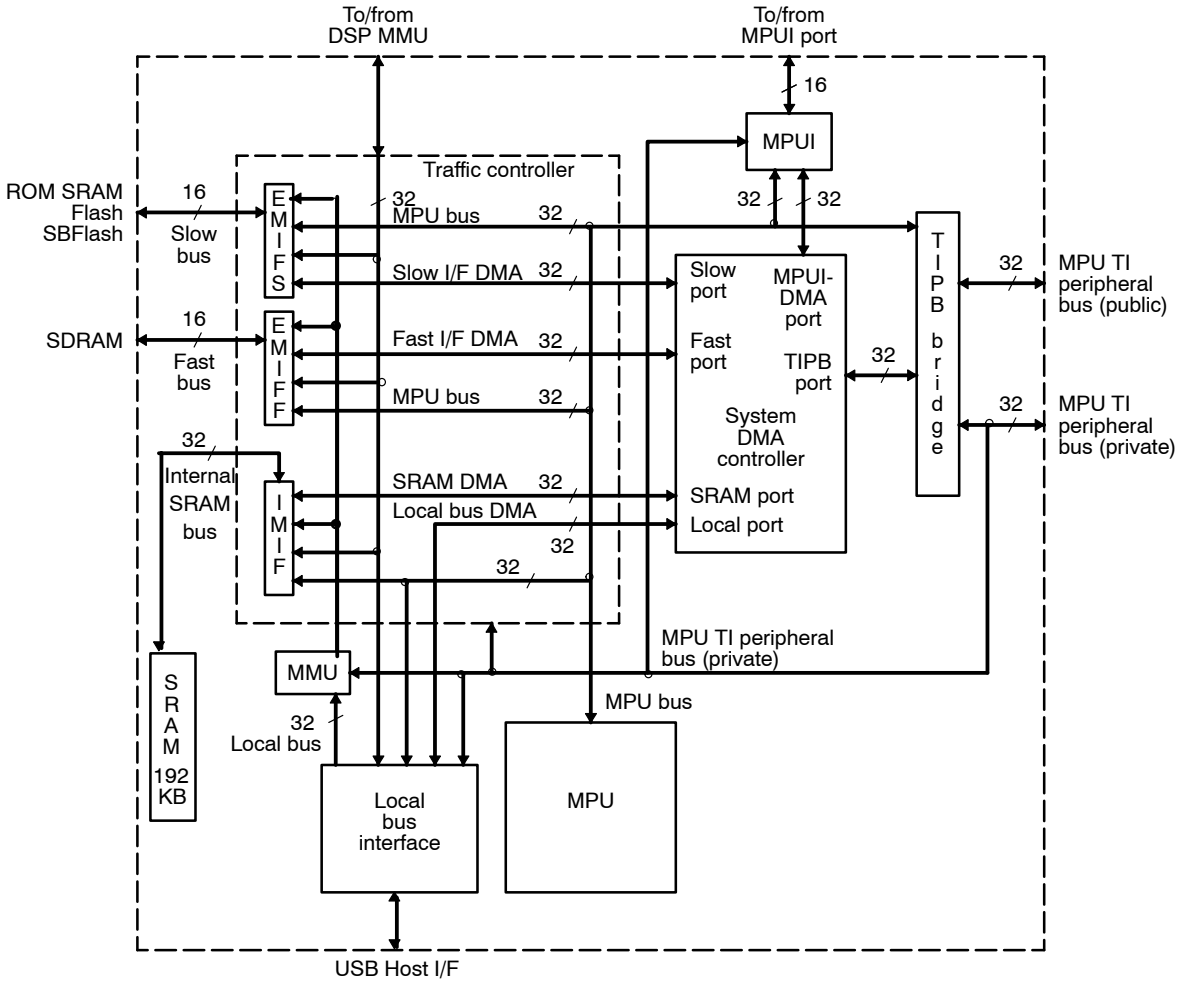


Table 1. Controller Access Mode and Data Access Width

Controllers	Single Access Mode Data Access Width (Bits)	Burst Access Mode Data Access Width (Bits)	Remarks
MPU	8, 16, 32	32	Single and burst access
C55x DSP	8, 16, 32	32	Single and burst access
System DMA controller	8, 16, 32	16, 32	Single and burst access; the 16-bit burst access is reserved for the LCD controller channel
Local bus	8, 16, 32	32	Single and burst access

The memories accessed by the TC are separated into two groups:

- External memory is memory that is not part of the OMAP5910 device. It can be SDRAM, flash, ROM, RAM, etc. External memory is accessed using the external memory interface (EMIF). The TC has two separate memory interfaces to access the external memories.
  - External memory interface fast (EMIFF): A fast synchronous interface for SDRAM
  - External memory interface slow (EMIFS): An asynchronous/ synchronous interface to handle flash, ROM, RAM, etc.
- Internal memory is memory that is part of the OMAP5910 device and consists of 192K bytes (1.5MB) of SRAM. The TC accesses the internal memory using an internal memory interface (IMIF) that is part of the TC.

Four hosts access the system resources using the TC.

- MPU: The MPU is connected to the TC via the MPU bus. The MPU can access memories that are connected to the IMIF, EMIFF, and EMIFS.
- DSP: The DSP is connected to the TC via the DSP MMU bus. The C55x DSP can access memories connected to IMIF, EMIFF, and EMIFS.
- System DMA: The system DMA is connected to the TC using four separate 32-bit buses, providing the system DMA controller with concurrent
  access to memories connected to the IMIF, EMIFF, and EMIFS.
- Internal local bus interface: An internal local bus interface is connected to the TC to allow access to memories connected to IMIF, EMIFF, and EMIFS. In OMAP5910, the USB host controller interfaces to (and is master of) the local bus.

The TC provides each of the four hosts with:

- 8-/16-/32-bit single access and burst access, except the LCD controller channel. 16- or 32-bit access must start from the 16- or 32-bit boundary address.
- Size adaptation for 8-, 16-, or 32-bit words, with the requirement that address must be aligned on the correct bit boundary. For example, 32-bit access must be aligned on 32-bit boundary, 16-bit access must be aligned on 16-bit boundary, and so forth.
- Access duration management (wait state insertion) to enable the connection of slow memory devices
- Memory control signal generation (chip-select, memory-specific protocol generation)
- 16-bit burst access for the LCD controller channels

2 Memory Map

Four external chip-selects and a series of internal address decodes are provided for external and internal memories and for peripherals attached to the TI peripheral bus (see Table 2). CS0, CS1, CS2, and CS3 each has an address range of 32M bytes; the external SDRAM space has an address range of 64M bytes; the internal SRAM space has an address range of 512K bytes.

EMIFS memory spaces corresponding to CS0 and CS3 are swapped if the MPU_BOOT pin is high during reset. MPU_BOOT pin 11 is multiplexed with MCBSP3.DR and USBI_SUSP. This multiplex configuration can be changed after the reset. The state of this pin is reflected in the BM bit field of the EMIF slow interface configuration register. For details, see Table 13, *EMIF Slow Interface Configuration Register (EMIFS_CONFIG_REG).*

Table 2. Device Types Associated With Chip-Select

CS	Device
CS0	External asynchronous RAM, ROM or flash External synchronous burst flash
CS1	External asynchronous RAM, ROM or flash External synchronous burst flash
CS2	External asynchronous RAM, ROM, or flash External synchronous burst flash
CS3	External asynchronous RAM, ROM, or flash External synchronous burst flash

Table 2. Device Types Associated With Chip-Select (Continued)

CS	Device
None†	External synchronous dynamic RAM
None†	Internal SRAM

† The interface to these memory devices is activated via internal address decoding. There is no external chip select.

The OMAP5910 peripherals are mapped on the MPU memory space in two different segments: through STROBE0 (public peripherals) and STROBE1 (private peripherals). Each peripheral has a range of 2K bytes.

Table 3 shows the MPU memory map.

Table 3. MPU Memory Map

Device Name	Start Address	End Address	Size in Bytes	Data Access†
System Memory Address Space				
External Slow Memory Interface (Flash)				
FLASH CS0	0000:0000	01FF:FFFF	32M bytes	8/16/32 R/W
Reserved	0200:0000	03FF:FFFF		
FLASH CS1	0400:0000	05FF:FFFF	32M bytes	8/16/32 R/W
Reserved	0600:0000	07FF:FFFF		
FLASH CS2	0800:0000	09FF:FFFF	32M bytes	8/16/32 R/W
Reserved	0A00:0000	0BFF:FFFF		
FLASH CS3	0C00:0000	0DFF:FFFF	32M bytes	8/16/32 R/W
Reserved	0E00:0000	0FFF:FFFF		
External Fast Memory Interface (SDRAM)				
SDRAM	1000:0000	13FF:FFFF	64M bytes	8/16 R/W
Reserved	1400:0000	1FFF:FFFF		
Internal Memory Interface (SRAM)				
Internal RAM	2000:0000	2002:FFFF	192K bytes	8/16/32 R/W
Reserved	2003:0000	2FFF:FFFF		

† Each register must always be accessed using the appropriate data access width as indicated in this table. Failure to do so may result in unexpected behavior including a TIPB bus error condition with an associated interrupt. Reserved address locations should never be accessed.

Table 3. MPU Memory Map (Continued)

Device Name	Start Address	End Address	Size in Bytes	Data Access†
DSP Processor Address Space				
DSP MPUI Interface				
MPUI Port RAM	E000:0000	E0FF:FFFF	16M bytes	16/32 R/W
MPUI DSP Peripherals I/O Space	E100:0000	E101:FFFF	128K bytes	16 R/W
DSP Private TIPB Peripherals (Strobe0)				
DSP TI peripheral bus	E100:0000	E100:07FF	2K bytes	16 R/W
Reserved	E100:0800	E100:7FFF	30K bytes	
DSP CLKM (clock control)	E100:8000	E100:87FF	2K bytes	16 R/W
Reserved	E100:8800	E100:8FFF	2K bytes	
DSP Shared TIPB Peripherals (Strobe1)				
UART1	E101:0000	E101:07FF	2K bytes	8 R/W
UART2	E101:0800	E101:0FFF	2K bytes	8 R/W
Reserved	E101:1000	E101:17FF	2K bytes	
McBSP1	E101:1800	E101:1FFF	2K bytes	16 R/W
MCSI2	E101:2000	E101:27FF	2K bytes	16 R/W
MCSI1	E101:2800	E101:2FFF	2K bytes	16 R/W
Reserved	E101:3000	E101:6FFF	16K bytes	
McBSP3	E101:7000	E101:77FF	2K bytes	16 R/W
Reserved	E101:7800	E101:97FF	8K bytes	
UART3	E101:9800	E101:9FFF	2K bytes	8 R/W
Reserved	E101:A000	E101:DFFF	16K bytes	
GPIOs	E101:E000	E101:E7FF	2K bytes	16 R/W
Reserved	E101:E800	E101:EFFF	6K bytes	
MPU Address Space				
MPUI port interrupt, control and status registers	E102:0000	E102:0003	4 bytes	16 R/W

† Each register must always be accessed using the appropriate data access width as indicated in this table. Failure to do so may result in unexpected behavior including a TIPB bus error condition with an associated interrupt. Reserved address locations should never be accessed.

Table 3. MPU Memory Map (Continued)

Device Name	Start Address	End Address	Size in Bytes	Data Access†
MPU Address Space (Continued)				
Reserved	E102:0004	EFFF:FFFF		
Reserved	F000:0000	FFFD:0000		
MPU Public TIPB Peripherals (Strobe 0)				
UART1	FFFB:0000	FFFB:07FF	2K bytes	8 R/W
UART2	FFFB:0800	FFFB:0FFF	2K bytes	8 R/W
McBSP2	FFFB:1000	FFFB:17FF	2K bytes	16 R/W
Reserved	FFFB:1800	FFFB:2FFF	6K bytes	
µWire	FFFB:3000	FFFB:37FF	2K bytes	16 R/W
I^2C	FFFB:3800	FFFB:3FFF	2K bytes	16 R/W
USB function	FFFB:4000	FFFB:47FF	2K bytes	16 R/W
RTC	FFFB:4800	FFFB:4FFF	2K bytes	8 R/W
MPUIO	FFFB:5000	FFFB:57FF	2K bytes	16 R/W
PWL	FFFB:5800	FFFB:5FFF	2K bytes	8 R/W
PWT	FFFB:6000	FFFB:67FF	2K bytes	8 R/W
Camera IF	FFFB:6800	FFFB:6FFF	2K bytes	32 R/W
Reserved	FFFB:7000	FFFB:77FF	2K bytes	
MMC	FFFB:7800	FFFB:7FFF	2K bytes	16 R/W
Reserved	FFFB:8000	FFFB:8FFF	4K bytes	
32-kHz timer	FFFB:9000	FFFB:97FF	2K bytes	32 R/W
UART3	FFFB:9800	FFFB:9FFF	2K bytes	8 R/W
USB host	FFFB:A000	FFFB:A7FF	2K bytes	32 R/W
FAC	FFFB:A800	FFFB:AFFF	2K bytes	16 R/W
Reserved	FFFB:B000	FFFB:BFFF	4K bytes	
HDQ/1-Wire	FFFB:C000	FFFB:C7FF	2K bytes	8 R/W

† Each register must always be accessed using the appropriate data access width as indicated in this table. Failure to do so may result in unexpected behavior including a TIPB bus error condition with an associated interrupt. Reserved address locations should never be accessed.

Table 3. MPU Memory Map (Continued)

Device Name	Start Address	End Address	Size in Bytes	Data Access†
MPU Public TIPB Peripherals (Strobe 0) (continued)				
TIPB switches	FFFB:C800	FFFB:CFFF	2K bytes	16 R/W
LED1	FFFB:D000	FFFB:D7FF	2K bytes	8 R/W
LED2	FFFB:D800	FFFB:DFFF	2K bytes	8 R/W
Reserved	FFFB:E000	FFFB:FFFF	8K bytes	
MPU Public TIPB Peripherals (Strobe 1)				
Reserved	FFFC:0000	FFFC:DFFF	56K bytes	
GPIOs	FFFC:E000	FFFC:E7FF	2K bytes	32 R/W
Reserved	FFFC:E800	FFFC:EFFF	2K bytes	
Mailbox	FFFC:F000	FFFC:F7FF	2K bytes	16 R/W
Reserved	FFFC:F800	FFFC:FFFF	2K bytes	
MPU Private TIPB Peripherals (Strobe 0)				
Reserved	FFFD:0000	FFFD:FFFF	2K bytes	
MPU Private TIPB Peripherals (Strobe 1)				
MPU level 2 interrupt handler	FFFE:0000	FFFE:07FF	2K bytes	32 R/W
ULPD power management	FFFE:0800	FFFE:0FFF	2K bytes	16 R/W
OMAP5910 configuration	FFFE:1000	FFFE:17FF	2K bytes	32 R/W
Die ID	FFFE:1800	FFFE:1FFF	2K bytes	32 R/W
Reserved	FFFE:2000	FFFE:BFFF	40K bytes	
LCD controller	FFFE:C000	FFFE:C0FF	256 bytes	32 R/W
Local bus interface	FFFE:C100	FFFE:C1FF	256 bytes	32 R/W
Local bus MMU	FFFE:C200	FFFE:C2FF	256 bytes	32 R/W
Reserved	FFFE:C300	FFFE:C4FF	512 bytes	
MPU Timer 1	FFFE:C500	FFFE:C5FF	256 bytes	32 R/W
MPU Timer 2	FFFE:C600	FFFE:C6FF	256 bytes	32 R/W

† Each register must always be accessed using the appropriate data access width as indicated in this table. Failure to do so may result in unexpected behavior including a TIPB bus error condition with an associated interrupt. Reserved address locations should never be accessed.

Table 3. MPU Memory Map (Continued)

Device Name	Start Address	End Address	Size in Bytes	Data Access†
MPU Private TIPB Peripherals (Strobe 1) (Continued)				
MPU Timer 3	FFFE:C700	FFFE:C7FF	256 bytes	32 R/W
MPU watchdog timer	FFFE:C800	FFFE:C8FF	256 bytes	32 R/W
MPUI	FFFE:C900	FFFE:C9FF	256 bytes	32 R/W
MPU private TIPB bridge	FFFE:CA00	FFFE:CAFF	256 bytes	32 R/W
MPU level 1 interrupt handler	FFFE:CB00	FFFE:CBFF	256 bytes	32 R/W
Traffic controller	FFFE:CC00	FFFE:CCFF	256 bytes	32 R/W
Reserved	FFFE:CD00	FFFE:CDFF	256 bytes	
MPU CLKM (clock control)	FFFE:CE00	FFFE:CEFF	256 bytes	32 R/W
DPLL1	FFFE:CF00	FFFE:CFFF	256 bytes	32 R/W
Reserved	FFFE:D000	FFFE:D0FF	256 bytes	
Reserved	FFFE:D100	FFFE:D1FF	256 bytes	
DSP MMU	FFFE:D200	FFFE:D2FF	256 bytes	32 R/W
MPU public TIPB bridge	FFFE:D300	FFFE:D3FF	256 bytes	16 R/W
JTAG ID code	FFFE:D400	FFFE:D4FF	256 bytes	32 R/W
Reserved	FFFE:D500	FFFE:D7FF		
System DMA controller	FFFE:D800	FFFE:DFFF	2K bytes	16 R/W
Reserved	FFFE:E000	FFFE:FFFF	2K bytes each	

† Each register must always be accessed using the appropriate data access width as indicated in this table. Failure to do so may result in unexpected behavior including a TIPB bus error condition with an associated interrupt. Reserved address locations should never be accessed.

3 Memory Interfaces

The TC has three memory interfaces:

- ❏ Internal memory interface (IMIF)
- ❏ External memory interface slow (EMIFS)
- ❏ External memory interface fast (EMIFF)

3.1 Internal Memory Interface

The IMIF interfaces to an on-chip 192K byte (1.5 MB) block of SRAM. The interface handles all single and burst requests from the MPU, the DSP, the system DMA controller, and the local bus.

3.1.1 IMIF Priority Handler

This memory interface has two software-selectable priority algorithms for resolving simultaneous access requests: least recently used and dynamic priority. The priority scheme is shared with the EMIFS and EMIFF and is set in the OMAP5910 configuration registers (bit 20, LRU_SEL in FUNC_MUX_CTRL_0). See SPRU671, *MPU Subsystem Reference Gide*, for details on configuration registers.

- Least recently used
  - A round-robin arbitration scheme. The highest priority requestor is the one that least recently accessed the memory.
- Dynamic priority
  - Dynamic priority uses high- and low-priority queues.
  - Each requestor, except the MPU, has a time-out register allocated to it (see *Time-Out Registers* in Section 4). These registers hold the number of clock cycles that a low-priority queue request must wait before it is moved from the low-priority queue to the high-priority queue.
  - At reset, all requestors are initially in the low-priority queue and the time-out registers are set to minimum value for each requestor. You must program these registers before using dynamic priority.
  - The low-priority queue order is:
    - MPU
    - DSP
    - Local bus
    - DMA (all channels including LCD)
  - The high-priority queue order is:
    - DMA transfer to LCD controller
    - DSP
    - Local bus
    - DMA transfer involving channels other than LCD channel
- Fixed priority is a special case of dynamic priority. To create a fixed priority, all time-out registers must have a value of 0. This way any request made goes into the high-priority queue after one clock cycle. Then the high-priority queue provides a fixed priority.

3.1.2 IMIF Operation

The 192K bytes of internal SRAM are selected by an internal chip select based on the appropriate address decode. The interface to the SRAM is 32 bits wide and provides support for single and burst accesses. The SRAM operates at the frequency of the traffic controller.

3.2 External Memory Interface Slow

The EMIFS interfaces traditional and synchronous flash, asynchronous SRAM, and ROM. The interface can drive up to four devices with one of four chip-select pins. Each chip-select pin has a corresponding register to specify the protocol for the associated external device.

Table 4 shows the EMIFS signal list.

Table 4. External Memory Interface Slow Signal List

Signal Name	I/O	Bus	Description
FLASH.RDY	I	–	Ready/busy signal from device
$\overline{\text{FLASH.WP}}$	O	–	Write protection
FLASH.CLK	O	–	Clock signal for flash device
$\overline{\text{FLASH.RP}}$	O	–	Flash power-down/reset
$\overline{\text{FLASH.CS0}}$	O	–	Active-low chip-select for device
$\overline{\text{FLASH.CS1}}$	O	–	Active-low chip-select for device
$\overline{\text{FLASH.CS2}}$†	O	–	Active-low chip-select for device
$\overline{\text{FLASH.CS3}}$	O	–	Active-low chip-select for device
$\overline{\text{FLASH.BAA}}$†	O	–	Active-low burst advance acknowledge for Advanced Micro Devices (AMD) burst flash
$\overline{\text{FLASH.OE}}$	O	–	Active-low output enable
$\overline{\text{FLASH.WE}}$	O	–	Active-low write enable
$\overline{\text{FLASH.ADV}}$	O	–	Active-low address valid
FLASH.D[15:0]	I/O	15 – 0	Flash data bus from external device

† $\overline{\text{FLASH.CS2}}$ and $\overline{\text{FLASH.BAA}}$ are multiplexed on the same device pin. Pin function is selected using the OMAP5910 configuration register, FUNC_MUX_CRTL_0. The $\overline{\text{FLASH.CS2}}$ functionality is default.

Table 4. External Memory Interface Slow Signal List (Continued)

Signal Name	I/O	Bus	Description
FLASH.A[24:1]	O	24 – 1	Flash data bus to external device
$\overline{\text{FLASH.BE}}$	O	3 – 0	External byte enable

† $\overline{\text{FLASH.CS2}}$ and $\overline{\text{FLASH.BAA}}$ are multiplexed on the same device pin. Pin function is selected using the OMAP5910 configuration register, FUNC_MUX_CRTL_0. The $\overline{\text{FLASH.CS2}}$ functionality is default.

Note:

OMAP5910 multiplexes the $\overline{\text{FLASH.CS2}}$ and $\overline{\text{FLASH.BAA}}$ pin functionality to the same device pin. Selecting the $\overline{\text{FLASH.BAA}}$ function to enable burst flash advance acknowledge disables $\overline{\text{FLASH.CS2}}$ functionality. In this case, capability of the EMIFS interface is reduced from a maximum of four external devices to a maximum of three external devices.

3.2.1 EMIFS Priority Handler

This memory interface has two software-selectable priority algorithms for resolving simultaneous access requests: least recently used and dynamic priority. The priority scheme is shared with the IMIF and EMIFF and is set in the OMAP5910 configuration registers (bit 20, LRU_SEL in FUNC_MUX_CTRL_0). See Chapter 6, *MPU Private Peripherals*, for details on configuration registers.

- Least recently used
  - A round-robin arbitration scheme. The highest priority requestor is the one that least recently accessed the memory.
- Dynamic priority
  - Dynamic priority uses high- and low-priority queues
  - Each requestor, except the MPU, has a time-out register allocated to it (see *Time-Out Registers* in Section 4). These registers hold the number of clock cycles that a low-priority queue request must wait before it is moved from the low priority queue to the high-priority queue.
  - At reset, all requestors are initially in the low-priority queue and the time-out registers are set to minimum value for each requestor. Users must program these registers before using dynamic priority.

- The low-priority queue order is:
  - MPU
  - DSP
  - Local bus
  - DMA (all channels excluding LCD)
- The high-priority queue order is:
  - DSP
  - Local bus
  - DMA transfer involving channels other than LCD channel

❏ Fixed priority is a special case of dynamic priority. To create a fixed priority, all time-out registers must have a value of 0. This way any request made goes into the high-priority queue after one clock cycle. Then the high-priority queue provides a fixed priority.

3.2.2 EMIFS Operation

This interface generates the appropriate signal timings to drive the following types of devices or compatible devices:

❏ Intel fast boot block flash (23FxxxF3)
❏ AMD simultaneous read/write boot sector flash (AM29DLxxxG)
❏ AMD burst mode flash (AM29BLxxxC)
❏ Intel StrataFlash memory (28FxxxJ3A)
❏ Intel synchronous StrataFlash memory (28FxxxK3/K18)
❏ Intel wireless flash memory (28FxxxW18)
❏ Asynchronous SRAM

Every flash command (read array, program, clear status register) is sent to the flash memory controller by the MPU. The MPU writes in the flash, followed by a read or a write, to set up the flash in the correct mode.

File/boot block flash basic operations supported are:

❏ Asynchronous read, including specific reads like manufacturer ID
❏ Burst read emulation (by multiple asynchronous reads) in 32-bit width
❏ Reset or power down
❏ Asynchronous write with $\overline{WE}$ in 16-bit width

The following operations are also supported for burst flash devices:

❏ Synchronous burst read mode (for Intel and AMD flashes)

An additional read mode is provided that supports burst read on page mode ROM devices.

Figure 3 through Figure 7 show the external timing of the protocols used by the EMIF slow interface.

3.2.3 Device Initialization

Depending on the flash memory or RAM device associated with each chip-select, the EMIFS interface must be initialized. If the device used is a flash, the flash may have to be initialized in the correct protocol to achieve maximum performance.

To use the external flash device with the synchronous flash burst protocol, the following configuration must be set in the flash device and in the EMIFS chip-select configuration registers (see Table 14, *EMIF Slow Chip-Select Configuration Registers*):

- ❏ Read mode
- ❏ Frequency configuration
- ❏ Data output configuration
- ❏ Burst order. The EMIFs only supports linear burst order.
- ❏ Burst length
- ❏ CLK configuration
- ❏ Flash mode operation. Some flash modules use multiple signals for burst operations (see Section 3.2.7, *Burst Read Operation*, for more information).

After reset, each of the EMIF slow chip-select configuration registers is configured in the asynchronous mode with 15 wait cycles and a clock divider of 6 (relative to the traffic controller clock). This configuration ensures maximum compatibility with many existing devices.

3.2.4 EMIFS Memory Timing Control

In both asynchronous and synchronous modes all EMIFS-to-memory control signals are referenced to an internal EMIFS reference clock. The internal EMIFS reference clock is divided from the TC clock by a programmable value in the FCLKDIV bit field of the EMIFS chip select configuration register (EMIFS_CSx_CONFIG). This allows the EMIFS to accommodate timing constraints of slow devices, even with high system clock rate. Table 5 shows FCLKDIV settings and resulting EMIFS reference clock values.

Table 5. FCLKDIV Settings and Resulting EMIFS Reference Clock

FCLKDIV	EMIFS Reference
00	TC clock/1
01	TC clock/2

Table 5. FCLKDIV Settings and Resulting EMIFS Reference Clock (Continued)

10	TC clock/4
11	TC clock/6

In the synchronous mode, the active EMIFS clock is output on the FLASH.CLK pin. In asynchronous mode, the pin is driven inactive low.

In synchronous modes a selectable retiming feature enables read data to be latched by a delayed EMIFS reference clock. The retiming feature accounts for delays through the OMAP5910 input/output pins by feeding back FLASH.CLK to offer optimum data and clock alignment. You can select the retiming mode using the RT bit in the EMIFS chip-select configuration registers.

3.2.5 Asynchronous Read Operation

Asynchronous read mode is selected by programming the RDMODE bit field to 000 in the corresponding EMIF slow chip-select configuration register. This is the default mode at reset.

The following characteristics describe asynchronous read mode operations:

- The chip-select pulse width depends on the RDWST bit field of the EMIFS chip-select configuration register. Pulse width equals:

  (RDWST + 2) x EMIFS_Ref (shown as N cycles in Figure 3)

  Chip-select minimum pulse width is (2 x EMIFS_Ref).

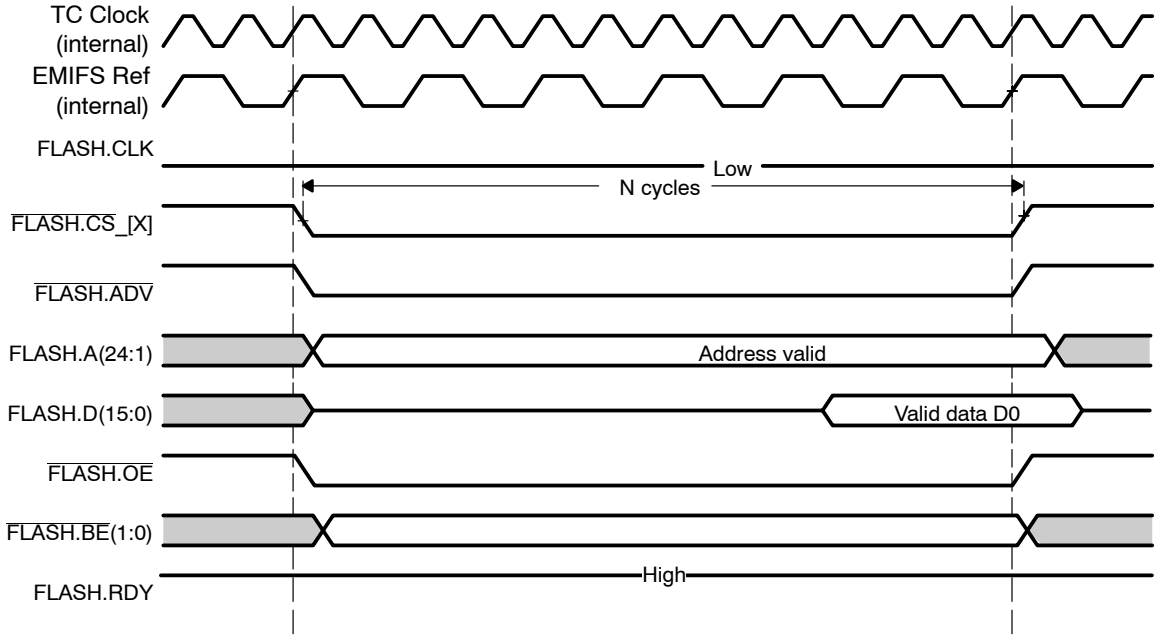
- Address drive time follows $\overline{\text{FLASH.CS}}$_[X] activation. The $\overline{\text{FLASH.ADV}}$ output is asserted with the address for use with Intel and AMD burst flash protocols.

- Read data is latched on the same TC clock rising edge that deactivates the $\overline{\text{FLASH.OE}}$ signal.

- In asynchronous mode, the internal EMIFS reference clock is not provided outside the EMIFS. The FLASH.CLK signal remains low.

- Figure 3 shows typical timing for an asynchronous 16-bit read operation on a 16-bit width device with RDWST = 4, FCLKDIV = 01.

Figure 3. Asynchronous 16-Bit Read Operation on a 16-Bit Width Device



3.2.6 Asynchronous Page Mode Read Operation

The asynchronous page mode read operation is similar to the asynchronous read, except that the number of wait states is different between the first access and the subsequent accesses within the page.

This mode of operation is selected by programming the following fields of the EMIF slow chip-select configuration registers (see Table 14, *EMIF Slow Chip-Select Configuration Registers*).

- RDMODE selects the memory type and number of words per page for page mode devices; supported values for words per page are 4, 8, or 16.
- RDWST sets the delay to insert prior to latching the first data word read from a page (range 0-15). The resulting delay is equal to (RDWST+2) x EMIFS_ref. This is represented by N cycles in Figure 4 and Figure 5. When crossing a page boundary, as in Figure 5, the RDWST parameter is used again for the first access on the new page.
- PGWST sets the delay between subsequent words in the page (range 0-15). The resulting delay is equal to (PGWST+1) x EMIFS_ref. This is represented by P cycles in Figure 4 and Figure 5.
- BW defines the word length of the access, which is equal to the memory data bus width.

As in asynchronous mode, device interface signals are referenced to the internal EMIFS reference clock, which is divided from the TC clock using FCLKDIV in the EMIF slow interface configuration register. The FLASH.CLK signal is not externally driven in asynchronous page operating mode.

Figure 4 shows typical timing for an asynchronous page mode 8x16-bit read operation on a 16-bit width device with RDWST = 2, PGWST = 0, FCLKDIV = 01, and RDMODE = 2.

Figure 5 shows typical timing for an asynchronous page mode 8x16-bit read with page crossing on a 16-bit width device with RDWST = 2, PGWST = 0, FCLKDIV = 01, and RDMODE = 2.

Figure 4. Asynchronous Page Mode 8x16-Bit Read Operation on a 16-Bit Width Device (8 Words per Page)

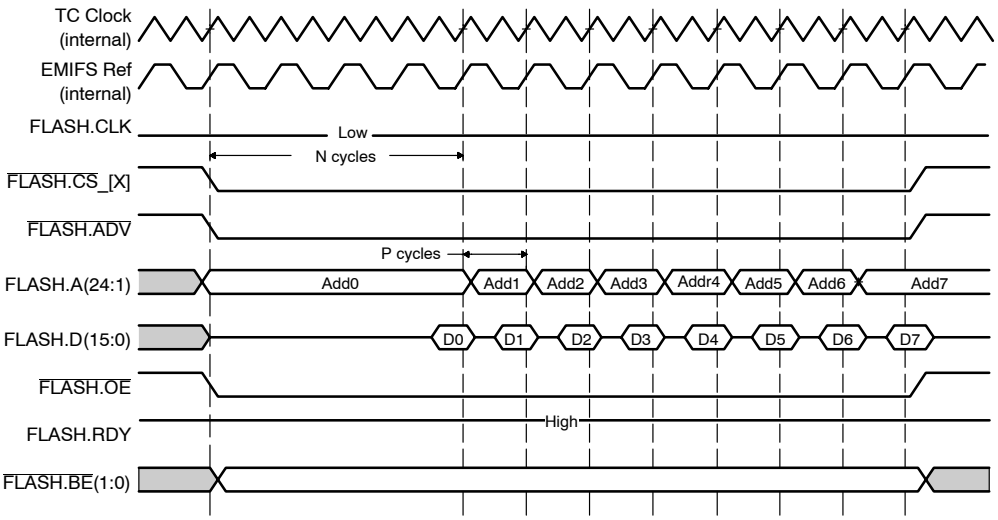
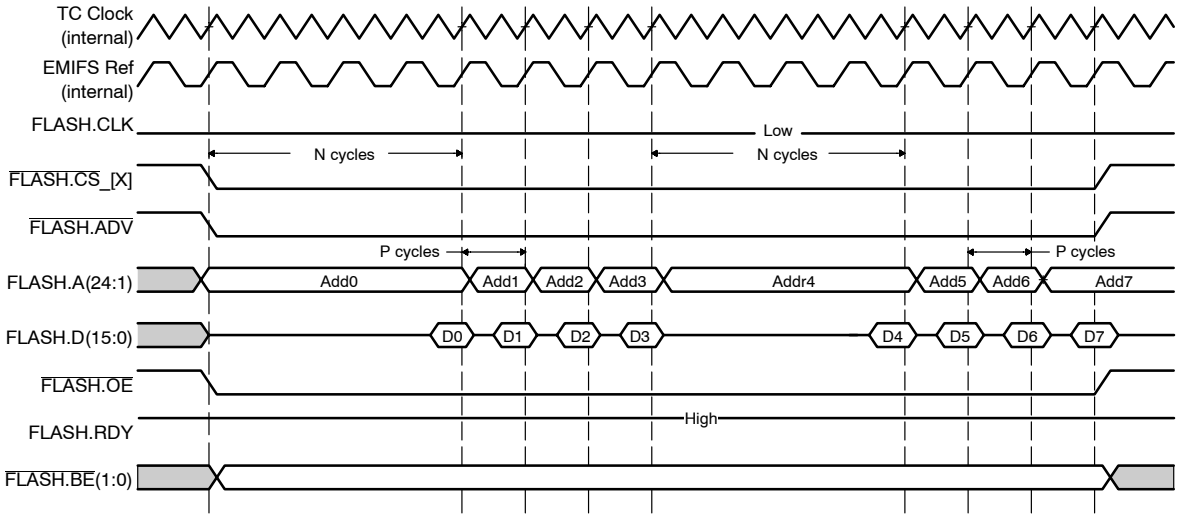


Figure 5. Asynchronous Page Mode 8x16-Bit Read With Page Crossing on 16-Bit Width Device (4 Words per Page)



3.2.7 Burst Read Operation

The synchronous read mode is selected for each device by setting the RDMODE configuration bit field to 100b.

In this mode of operation, FLASH.CLK is driven on the OMAP5910 device pin.

Both AMD burst flash and Intel burst flash have three modes of operation:

- ❏ Asynchronous single read mode (default)
- ❏ Synchronous single read or burst read mode
- ❏ Asynchronous write

Asynchronous single read mode and asynchronous write modes are compatible with operation described in Section 3.2.5, *Asynchronous Read Operation*, and Section 3.2.8, *Asynchronous Write With WE Operation*.

Figure 6 shows the timing view of synchronous burst read mode operation.

On the AMD device, $\overline{\text{LBA}}$ is directly connected to the $\overline{\text{FLASH.ADV}}$ OMAP5910 pin.

The address is latched on the rising edge of $\overline{\text{FLASH.ADV}}$ with a specified hold time of 3 ns. This is easily met by maintaining the address during two cycles.

Data output of the device is stable on the rising edge of FLASH.CLK (specified with a setup and hold time referenced to this edge).

Two configuration registers are used in this operating mode:

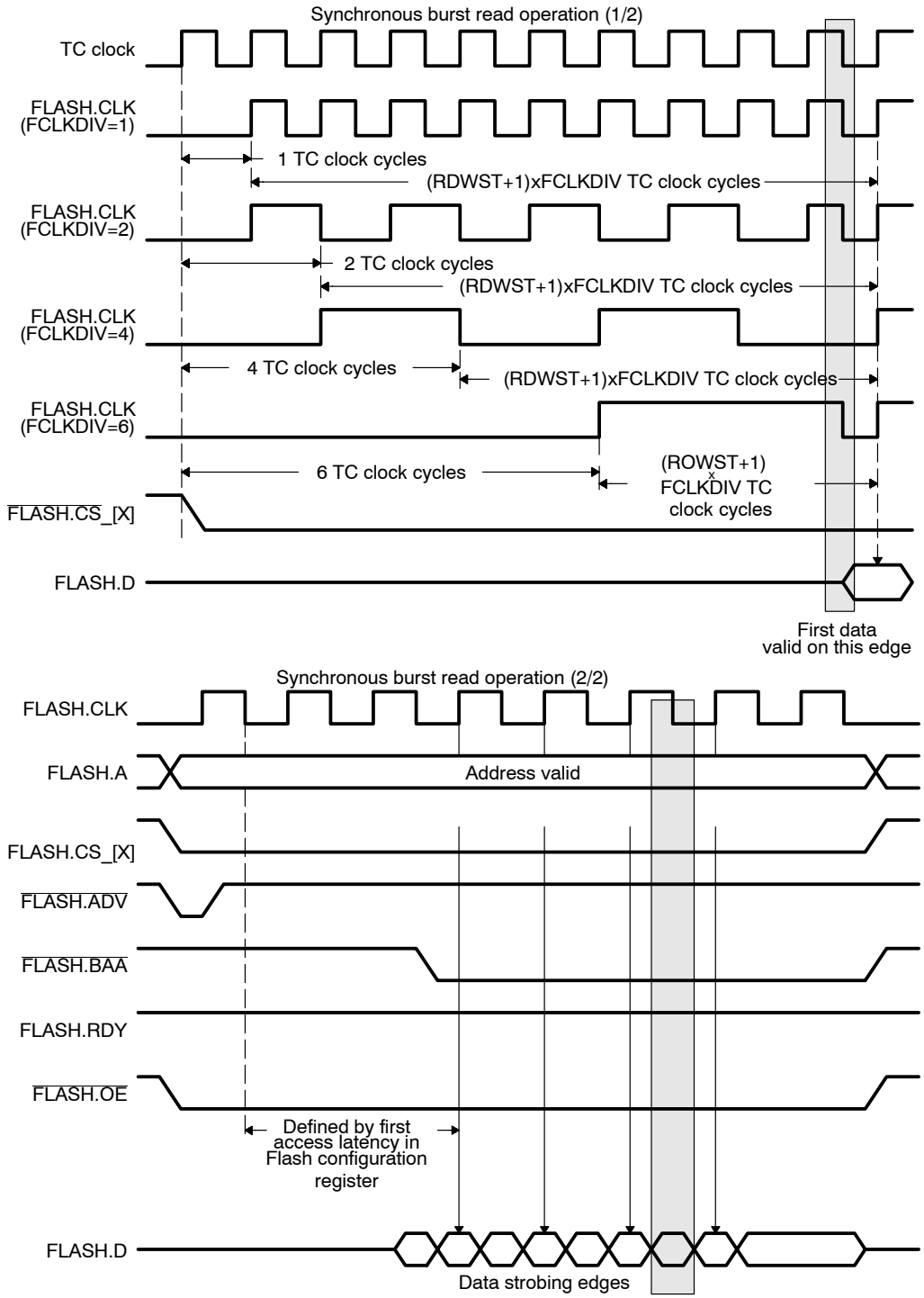
- FCLKDIV. Specifies the frequency ratio between the TC clock and FLASH.CLK (see Table 14, *EMIF Slow Chip-Select Configuration Registers*).
- RDWST. Specifies the number of FLASH.CLK cycles between the falling edge of $\overline{\text{FLASH.ADV}}$ and the edge at which first data is valid (see Table 14, *EMIF Slow Chip-Select Configuration Registers*).

The FLASH.RDY signal is not used in this mode: however, it is used during flash program and erase operations.

Note: Intel Burst Flash Operation

Intel burst flash (such as the Intel 28FxxxK3, 28FxxxK18, and 28FxxxW18), requires the OMAP5910 FLASH.RDY pin to be pulled up instead of being tied to the flash's WAIT pin. The traffic controller properly handles all timing requirements without the WAIT pin assertion. Connecting these two pins may cause a performance penalty.

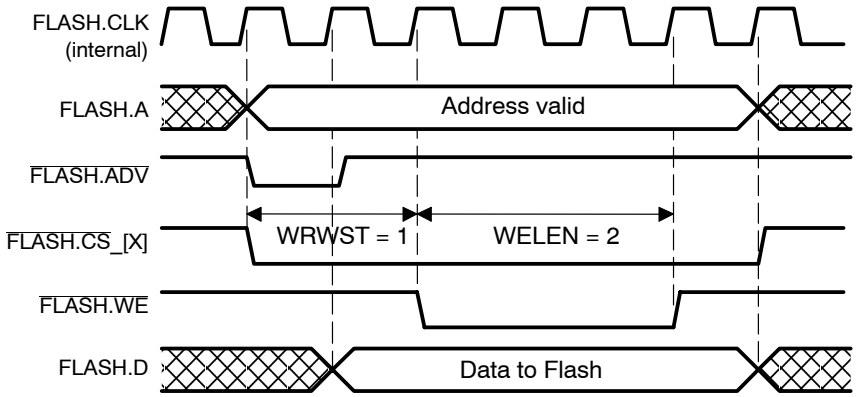
Figure 6. Synchronous Burst Read With Page Alignment



3.2.8 Asynchronous Write With WE Operation

The asynchronous write is used for both file flash and burst flash devices. Figure 7 shows the timing diagram. Burst write operation is not supported.

Figure 7. Asynchronous Write With WE Operation



The flash device latches the data on the rising edge of FLASH.WE. The FLASH.WE low pulse duration is programmable for each device through the WELEN field in the flash configuration register. The number of wait states between write operations is programmable for each device through the WRWST field in the EMIFS_C5x_CONFIG register.

The duration from falling FLASH.CS to falling FLASH.WE (shown in Figure 7) is equal to the programmed value of WRWST + 1, and the duration for which FLASH.WE is asserted active low is equal to the programmed value of WELEN + 1.

The FLASH.CLK signal is not driven inactive low in the asynchronous write mode.

The chip-select pulse duration equals:

(WRWST + WELEN + 3) X EMIFS_Ref\

Table 6. EMIFS CS Active Widths for Asynchronous Reads/Writes

FCLK DIV	CS Active Width Read (TC Cycles)	CS Active Width Write (TC Cycles)
/1	1 * (RDWST + 1) + 1	1 * (WRWST + WELEN + 1) + 2
/2	2 * (RDWST + 1) + 2	2 * (WRWST + WELEN + 1) + 4

Table 6. EMIFS $\overline{CS}$ Active Widths for Asynchronous Reads/Writes (Continued)

FCLK DIV	$\overline{CS}$ Active Width Read (TC Cycles)	$\overline{CS}$ Active Width Write (TC Cycles)
/4	4 * (RDWST + 1) + 4	4 * (WRWST + WELEN + 1) + 8
/6	6 * (RDWST + 1) + 6	6 * (WRWST + WELEN + 1) + 12

3.2.9 EMIFS Dual-Port RAM Interface Mode

The OMAP5910 EMIFS includes a programmable mode associated with the $\overline{\text{FLASH.CS2}}$ chip select pin to support external devices that require a valid flash address before chip select is active. An example of such a device is a dual port RAM (DPRAM). When DPRAM mode is enabled, the low transition of $\overline{\text{FLASH.CS2}}$ is delayed to ensure that address is valid. The low to high transition of $\overline{\text{FLASH.CS2}}$ is not changed regardless of the mode setting. EMIFS DPRAM mode is programmed in the OMAP5910 configuration registers using bit 22, CONF_MOD_DPRAM_ENABLE_R, in register MOD_CONF_CTRL_0. See Chapter 6, *MPU Private Peripherals*, for details on configuration registers.

DPRAM interface mode is only applicable to EMIFS chip-select $\overline{\text{FLASH.CS2}}$. Also note that the $\overline{\text{FLASH.CS2}}$ pin multiplexes the $\overline{\text{FLASH.BAA}}$ function (see Table 4), To activate the DPRAM interface mode, the user must first ensure that OMAP5910 pin multiplexing has $\overline{\text{FLASH.CS2}}$ selected, then program for DPRAM interface configuration as described above. The FLASH.CS2 pin asserts only once for 32-bit access, even though it is divided into two 16-bit accesses.

3.3 External Memory Interface Fast

The EMIFF can interface with synchronous DRAM (SDRAM). The interface directs all the transactions to the SDRAM device. The bus width is 16 bits.

Table 7 shows the EMIFF signal list.

Table 7. External Memory Interface Fast Signal List NIL

Signal Name	I/O	Description
SDRAM.A[12:0]	O	SDRAM address bus
SDRAM.D[15:0]	I/O	Data from SDRAM
SDRAM.CLK	O	Clock to SDRAM
SDRAM.BA[1:0]	O	SDRAM bank select
SDRAM.CKE	O	SDRAM clock enable
$\overline{\text{SDRAM.RAS}}$	O	SDRAM RAS
$\overline{\text{SDRAM.CAS}}$	O	SDRAM CAS
$\overline{\text{SDRAM.WE}}$	O	SDRAM write enable
$\overline{\text{SDRAM.DQML}}$	O	Lower byte 3-state
$\overline{\text{SDRAM.DQMU}}$	O	Upper byte 3-state

3.3.1 EMIFF Priority Handler

This memory interface has two software-selectable priority algorithms for resolving simultaneous access requests: least recently used and dynamic priority. The priority scheme is shared with the EMIFS and IMIF and it is set in the OMAP5910 configuration registers (bit 20, LRU_SEL in FUNC_MUX_CTRL_0). See SPRU671, *MPU Subsystem Reference Guide*, for details on configuration registers.

- Least recently used
  - A round-robin arbitration scheme. The highest priority requestor is the one that least recently accessed the memory.
- Dynamic priority
  - Dynamic priority uses high- and low-priority queues.
  - Each requestor, except the MPU, has a time-out register allocated to it (see *Time-Out Registers* in Section 4). These registers hold the number of clock cycles that a low-priority queue request has to wait before it is moved from the low-priority queue to the high-priority queue.
  - At reset, all requestors are initially in the low-priority queue and the time-out registers are set to minimum value for each requestor. You must program these registers before using dynamic priority.

- The low-priority queue order is:
  - MPU
  - DSP
  - Local bus
  - DMA (all channels including LCD)
- The high-priority queue order is:
  - DMA transfer involving LCD channel
  - DSP
  - Local bus
  - DMA transfer involving channels other than LCD channel

❑ Fixed priority is a special case of dynamic priority. To create a fixed priority, all time-out registers must have a value of 0. This way any request made goes into the high-priority queue after one clock cycle. Then the high-priority queue provides a fixed priority.

3.3.2 EMIFF Operation

The EMIFF controller can support up to two devices for up to 64M bytes of memory. The following devices are supported:

❑ 256M bit, 128M bit, 64M bit
❑ 2 or 4 banks for 64M byte device
❑ x8 or x16 data bus configurations

Table 8 shows the possible SDRAM configurations.

Table 8. Possible SDRAM Configurations

Memory Size (Bytes)	Bus Size	Number of Devices	Type of Device
64M	1 x16	1	512M bits organized in 32M x 16
	2 x 8	2	256M bits organized in 32M x 8
32M	1 x 16	1	256M bits organized in 16M x 16
	2 x 8	2	128M bits organized in 16M x 8
16M	1 x 16	1	128M bits organized in 8M x 16
	2 x 8	2	64M bits organized in 8M x 8
8M	1 x 16	1	64M bits organized in 4M x 16
4M	2 x 8	2	16M bits organized in 2M x 8
2M	1 x 16	1	16M bits organized in 1M x 16

The LCD channel transfer and MPU cache refill use burst transfers from SDRAM. LCD channel transfer has a higher priority, and its burst length is always 8 x 16 bits. SDRAM burst transfers are supported for a maximum of 8 x 16 bits, which can be utilized for DSP and MPU cache line fills and system DMA. The burst length through mode register initialization would be (EMIFF_MRS[2:0] PGBL=111b)

The SDRAM controller supports:

- The self-refresh mode (idle) and auto-refresh (normal operation)
- Automatic generation of MRS and EMRS commands to the SDRAM by writing to a mirror configuration register within the OMAP5910 device
- Burst sizes of 1x8, 1x16, 1x32, and 4x32 for all accesses and 8x16 burst access for LCD.
- Burst across page boundary (local address increment coupled with current address register)

3.3.3 SDRAM Mode and Extended Mode Register Initialization

To make SDRAM memory accessible, its internal mode register must first be configured. The MRS register contains the protocol information used to communicate with the OMAP5910 device (burst size, latency, write burst, etc.). The EMRS register enables certain low-power characteristics for the SDRAM.

- Writing to the EMIF fast interface SDRAM MRS register (EMIFF_MRS) automatically forces the generation of an MRS command on the pins of the SDRAM interface. When the command is issued, the content of the OMAP5910 MRS register is placed on the SDRAM address bus and latched by the SDRAM into its internal MRS register.
- OMAP5910 uses the same EMIF fast interface SDRAM MRS register, combined with a control bit setting, to write EMRS commands to the SDRAM. When the CONF_MOD_EMRS_CTRL bit field in the MOD_CONF_CTRL_0 register is set, the OMAP configures SDRAM banks to write out the EMIFF_MRS register as EMRS commands instead of MRS commands.
- Reading from the EMIF fast interface SDRAM MRS register does not generate any external transactions.

Note:

Typically, SDRAM requires 100 μs to stabilize after power up. Software is responsible for performing the initial setup of SDRAM. For more information refer to the SDRAM manufacturer's data sheet.

3.3.4 SDRAM Autorefresh Initialization

To increase SDRAM bus availability, it is preferable to subdivide the SDRAM into smaller sections and then evenly distribute the refresh of each of these subsections instead of performing a single autorefresh for the entire SDRAM. The OMAP5910 device can support subdividing the autorefresh of the SDRAM into bursts of 1, 4, or 8 rows. It is recommended to set this parameter to 8 rows.

A 16-bit timer is used to track the interval between autorefresh burst requests to the SDRAM. An autorefresh request is issued when the timer reaches a user-defined value based on the following parameters:

- SDRAM frequency
- Refresh rate
- Number of SDRAM rows

The following formula is used to determine the refresh counter value that must be programmed in the EMIF fast interface configuration register 1 (EMIFF_SDRAM_CONFIG):

$$\text{Counter Value} = \frac{\left(\frac{\text{SDRAM refresh rate}}{T_f}\right) - 400}{\text{Number of SDRAM Rows}}$$

where T_f = (1 / traffic controller frequency) and the 400 cycles take into account the worst-case priority scenario where the SDRAM refresh is at the bottom of the priority queue.

Example: 64-ms refresh rate, 75MHz traffic controller frequency, 4096 rows to be refreshed:

T_f = 13.3 ns

$$\text{Counter Value} = \frac{\left(\frac{64000000\ \text{ns}}{13.3\ \text{ns}}\right) - 400}{4096} = 1172\ \text{cycles}$$

This ensures a 64-millisecond refresh period for the full SDRAM.

3.3.5 SDRAM Self-Refresh Protection

In idle mode, the TC clock is stopped. If the clock will not be restarted for greater than the maximum refresh period (i.e. 64 ms), the SDRAM should be placed into the self-refresh mode before the TC enters idle to avoid corruption of data.

A similar SDRAM data corruption can occur in the event of a warm global system reset from external device pin. Since the reset event is likely to extend

beyond 64 milliseconds, and the SDRAM controller does not autorefresh during reset, data is corrupted. Setting the RFRSH_RST bit in the EMIF fast interface SDRAM configuration register 2 (EMIFF_SDRAM_CONFIG_2) avoids SDRAM data corruption for this case by automatically placing the SDRAM in self-refresh mode prior to warm reset being applied to the traffic controller. The SDRAM controller continues in self-refresh mode until the reset is unasserted. Note that RFRSH_RST applies only in the case of warm reset. For cold reset, SDRAM is not set to self-refresh regardless of the state of RFRSH_RST.

Caution: Self-Refresh Mode

When the EMIFF SDRAM is in self-refresh mode, the EMIFF does not respond to TIPB requests including MRS writes. To respond, the SLRF bit must be cleared by firmware. Writes to TC registers which would normally cause EMIFF to perform an action have no effect while EMIFF is in self-refresh mode. If an MRS write is attempted while EMIFF is in self-refresh mode, there is a pending MRS request. This prevents the traffic controller from idling and therefore prevents the device from entering deep sleep mode. The MRS request is not serviced until the SLRF bit is cleared.

3.3.6 SDRAM Clock Disable

The EMIF fast SDRAM clock signal (SDRAM.CLK) is disabled using these steps:

1) Set the PDE bit field of the EMIF slow interface configuration register.
2) Set one (or both) of the following bit fields in the EMIF fast SDRAM configuration register 1
   a) Set the SLRF to place the SDRAM into self-refresh mode
   b) Set the PWD to place the SDRAM into power-down mode
3) Set the CLK bit field of the EMIF fast interface SDRAM configuration register 1 to stop the clock

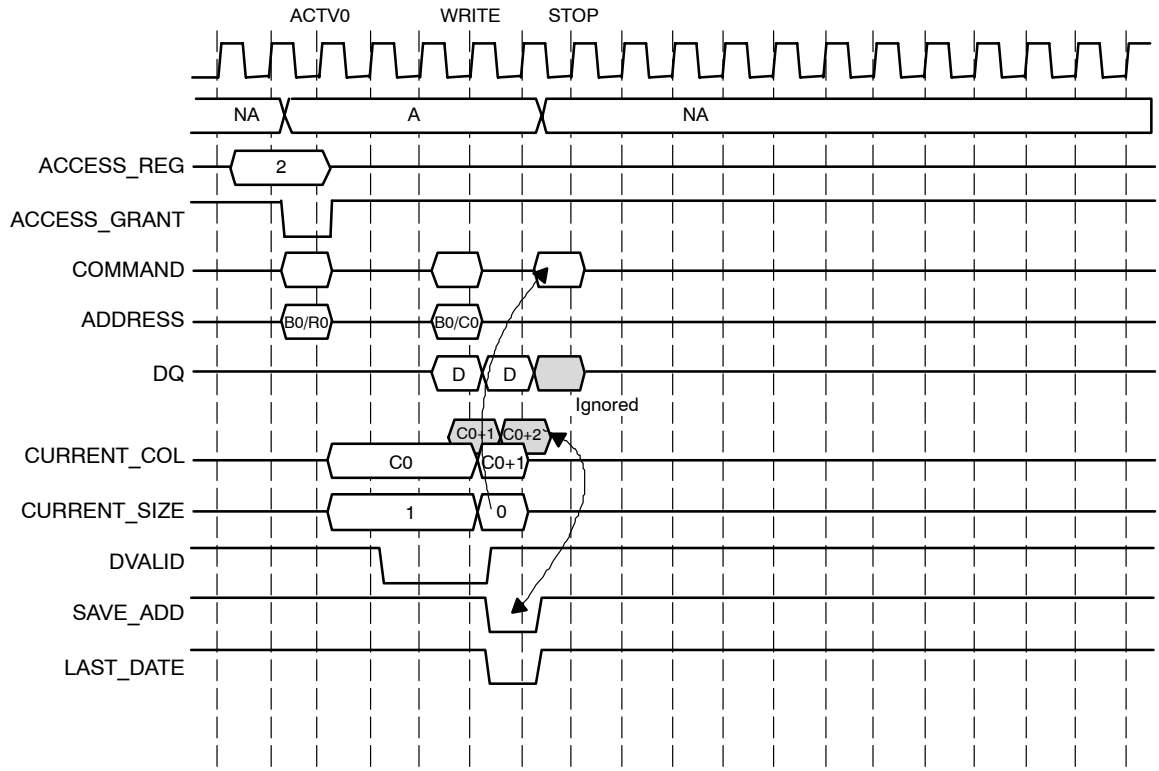
3.3.7 Endian Conversion Control

The traffic controller registers include a register to control endian conversion in the DSP memory management unit. For details, see Table 26, *Endianism Register (ENDIANISM)*.

3.3.8 SDRAM Access Timing Diagrams

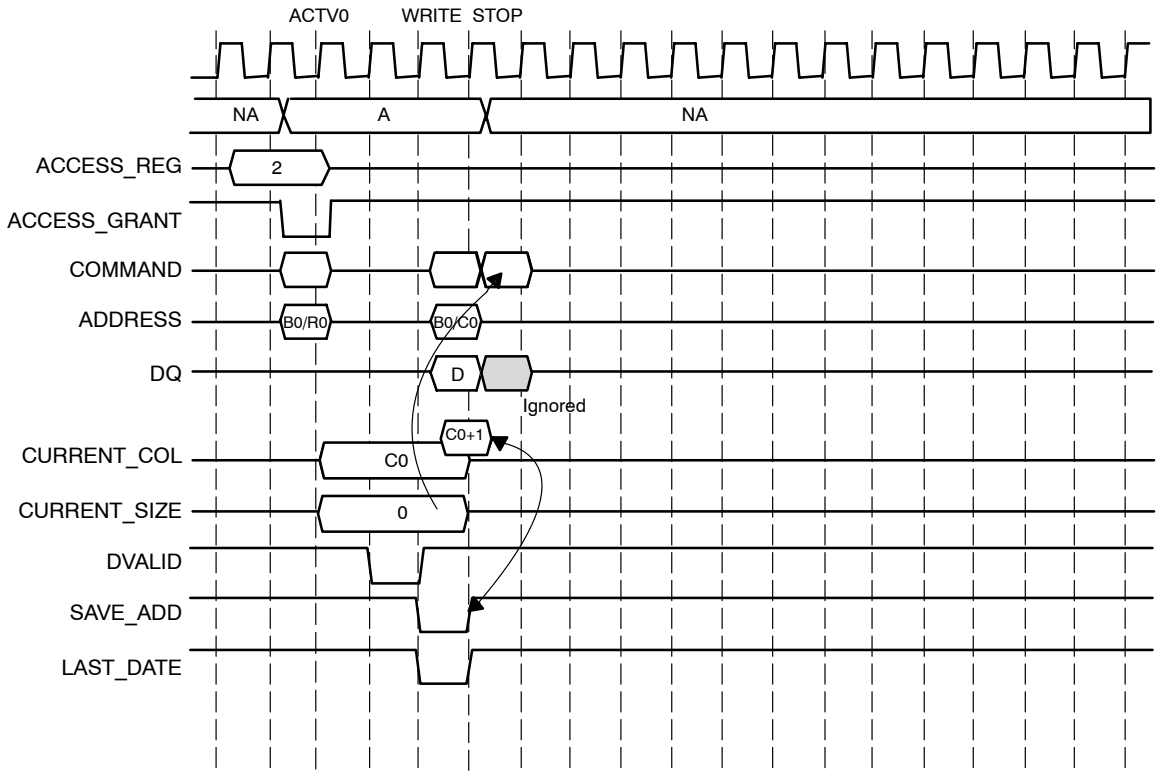
Figure 8 through Figure 18 show the SDRAM timing diagrams. Burst accesses shown here might not be achievable by all initiators of EMIFF transactions. See Section 3.3.2 for more detail on bursting behavior.

Figure 8. SDRAM Write Single 32-Bit Word With Burst Stop



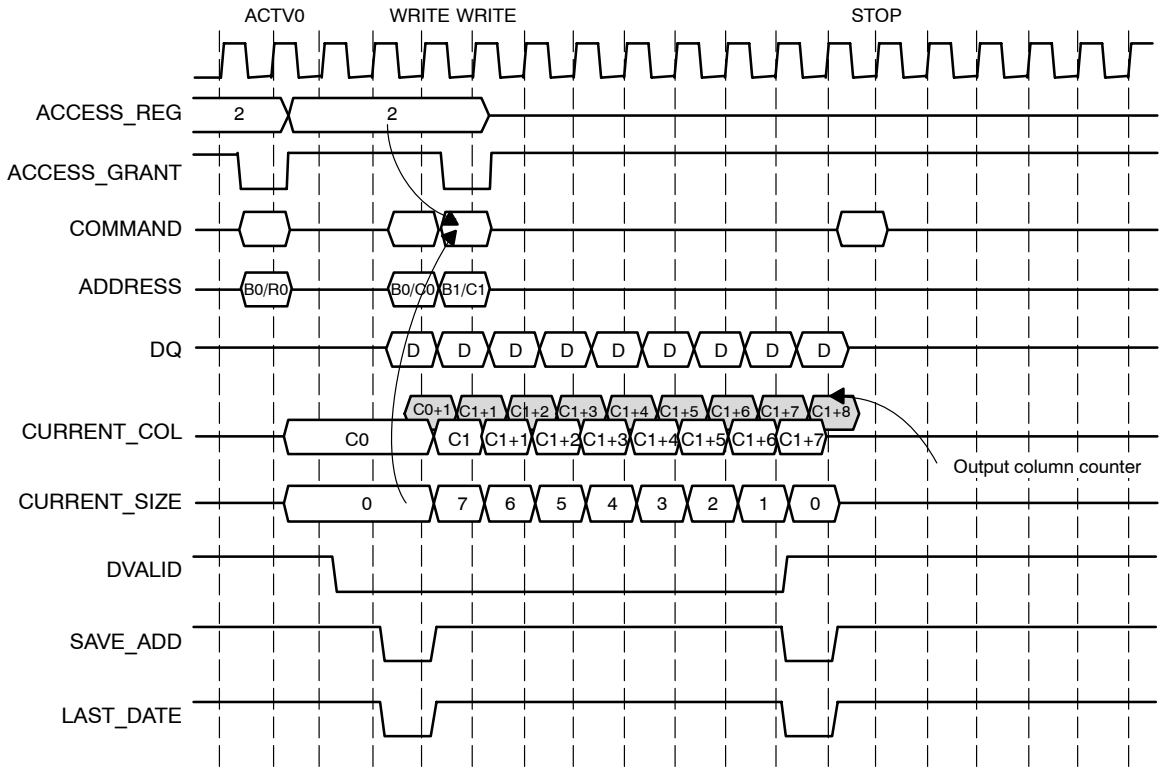
Note: WRITE (burst reduced to 2) is interrupted by a STOP command because no new request is pending.

Figure 9. SDRAM Write Single 16-Bit Half-Word With Burst Stop



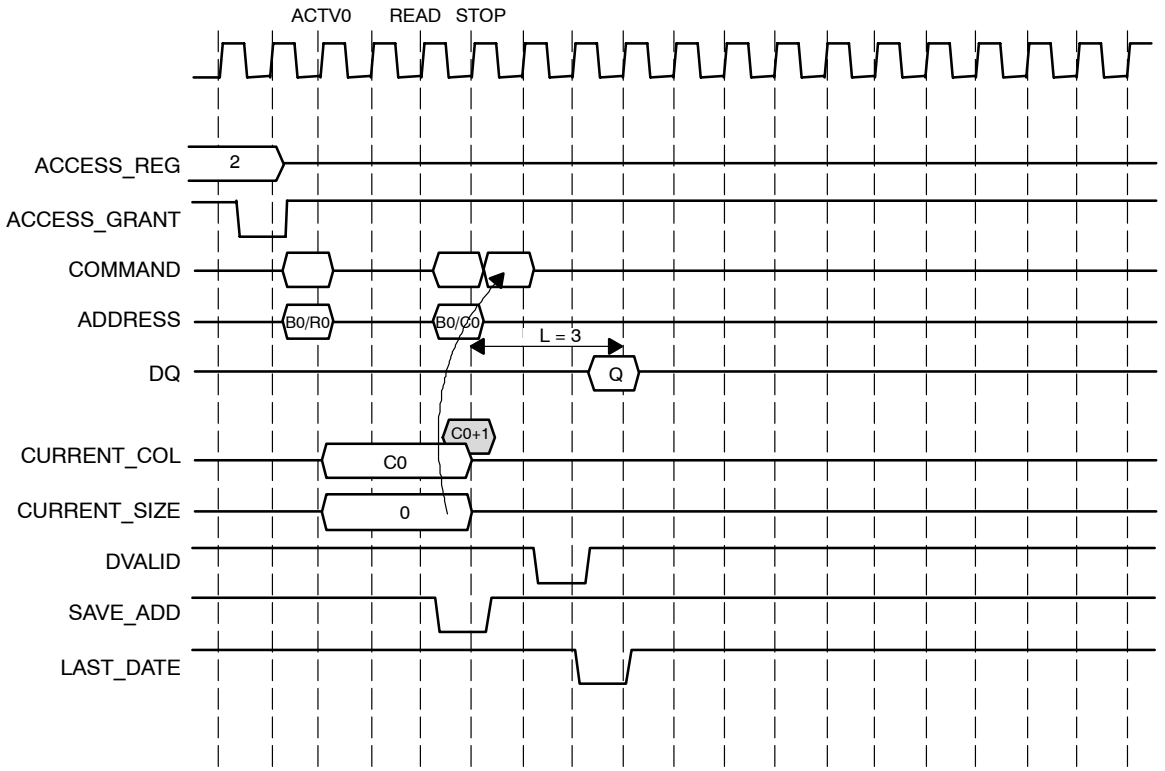
Note: WRITE (burst reduced to 1) is interrupted by a STOP command because no new request is pending.

Figure 10. SDRAM Write Single 16-Bit Half-Word Followed by Write Burst 8



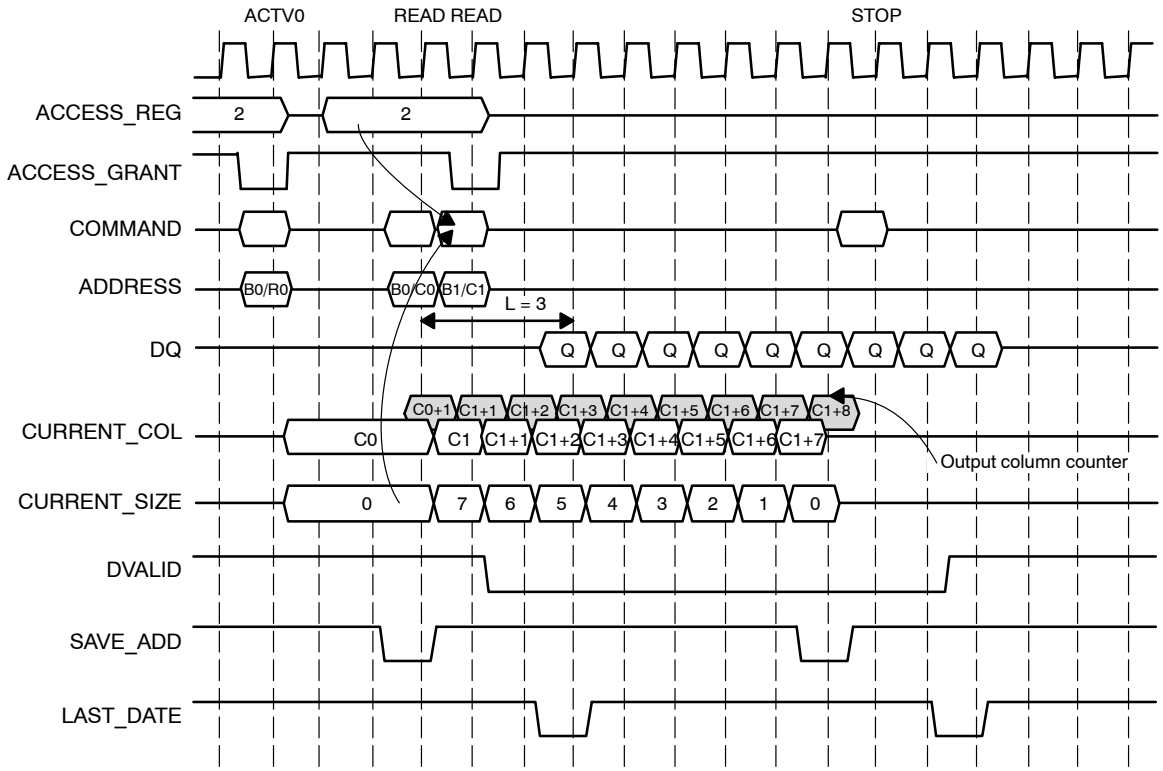
Note: WRITE (burst reduced to 1) is followed by a WRITE (8) in a different bank and in a page already active.

Figure 11. SDRAM Read Single 16-Bit Half-Word With Burst Stop



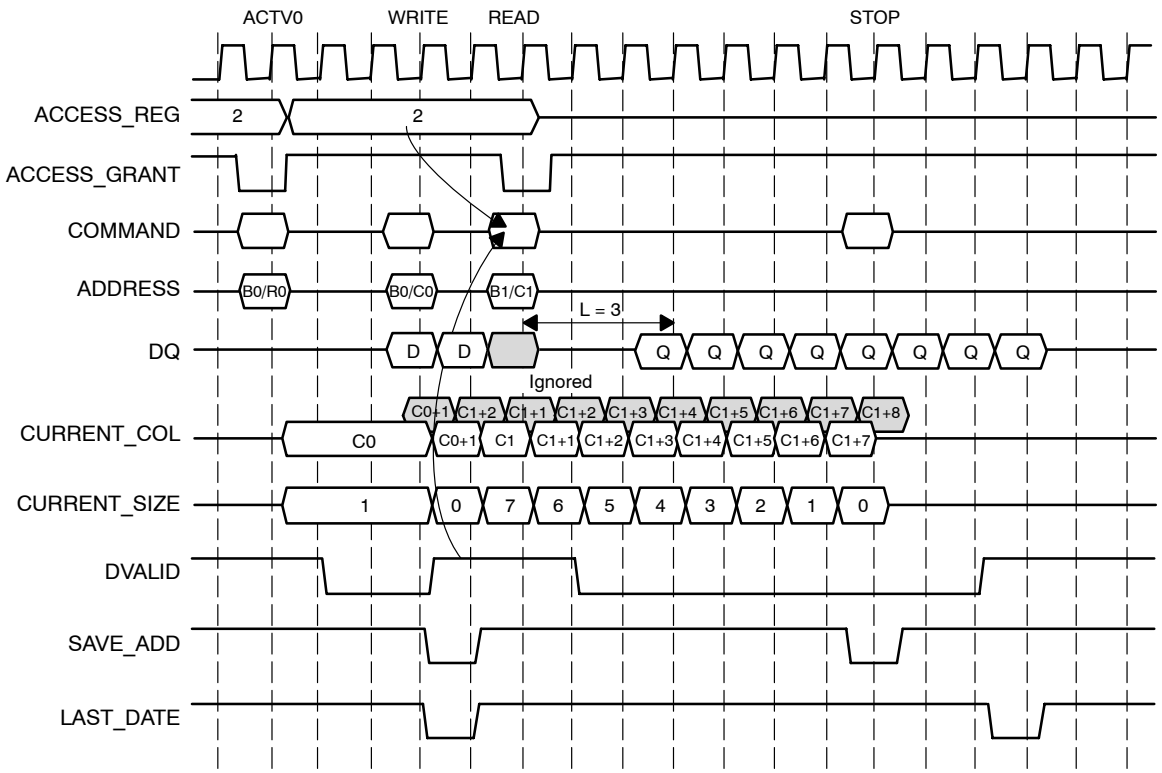
Note: READ (burst reduced to 1) is interrupted by a STOP command because no new request is pending.

Figure 12. SDRAM Read Single 16-Bit Half-Word Followed by Read Burst 8 Half-Word



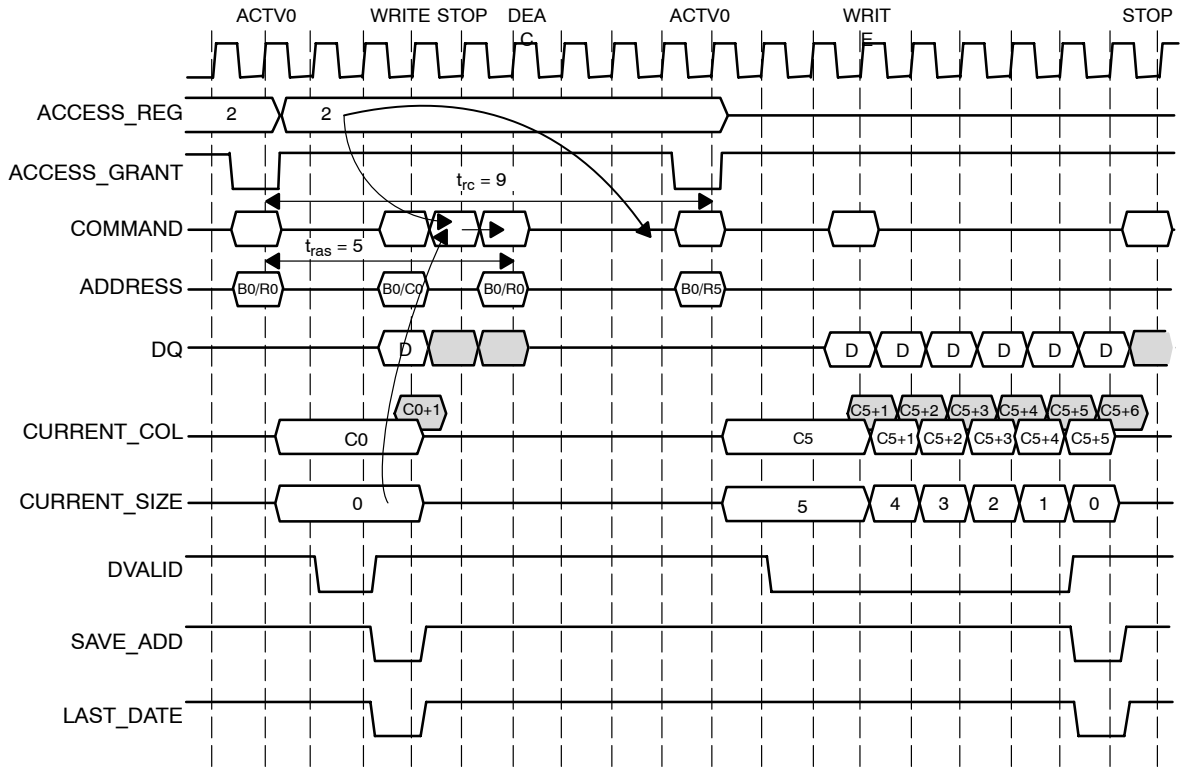
Note: READ (burst reduced to 1) is followed by a READ burst (8) in a different bank and in a page already active.

Figure 13. SDRAM Write Burst 32-Bit Word Followed by Read Burst 8 Half-Word



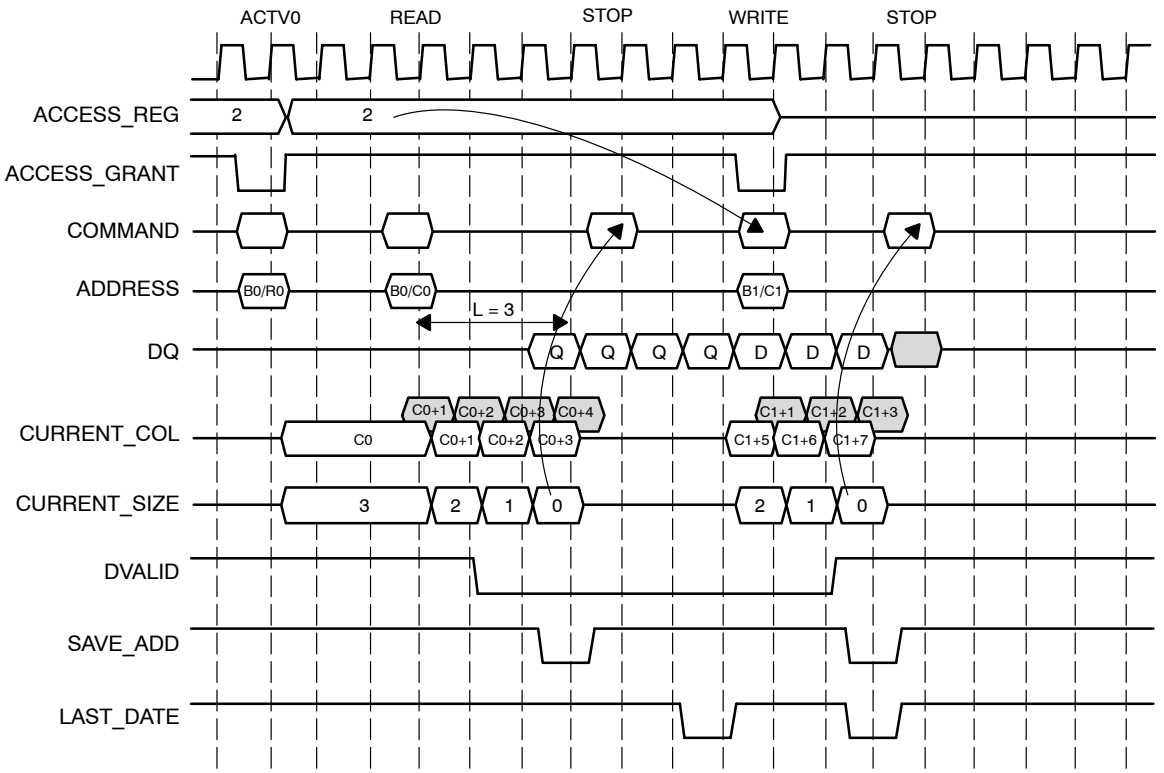
Note: WRITE (burst reduced to 2) is interrupted by a READ request pending on a bank and row already active.

Figure 14. SDRAM Single Half-Word Followed by a Read Burst 6 Half-Words



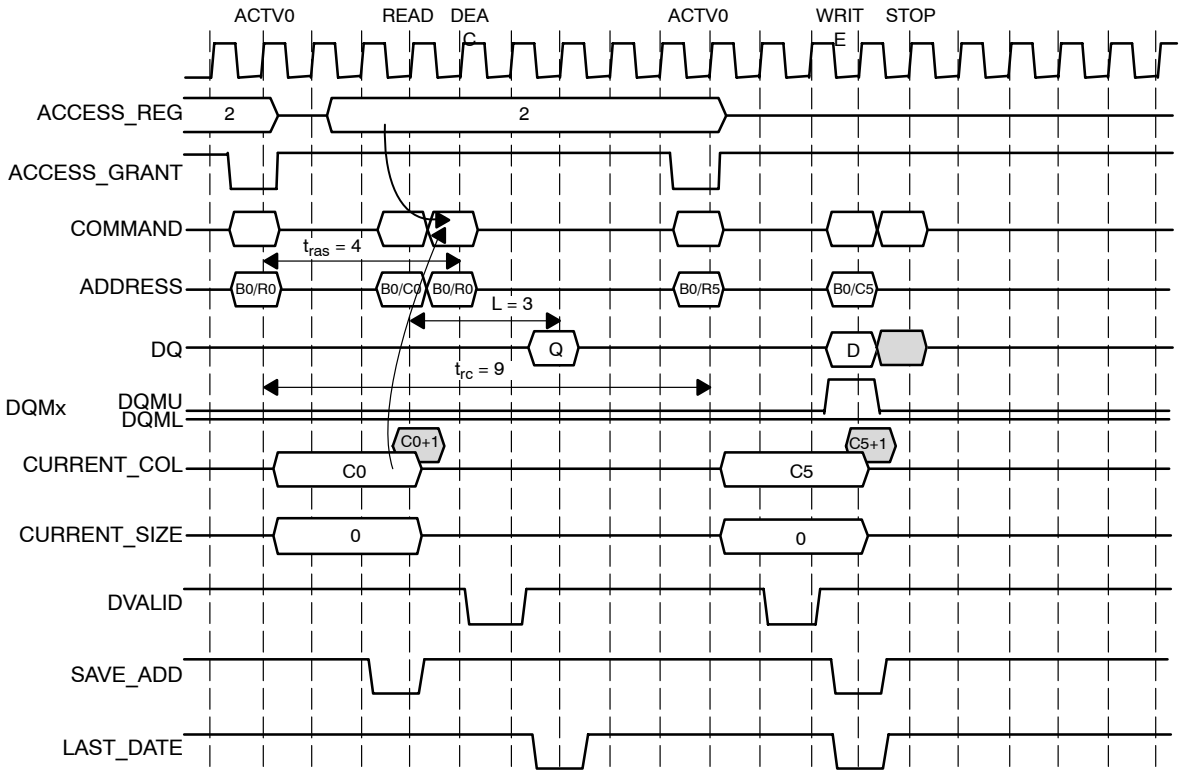
Note: WRITE (burst reduced to 1) is followed by a WRITE (6) in the same bank but on a different page..

Figure 15. SDRAM Read Burst 4 Half-Words Followed by a Write Burst 3 Half-Words



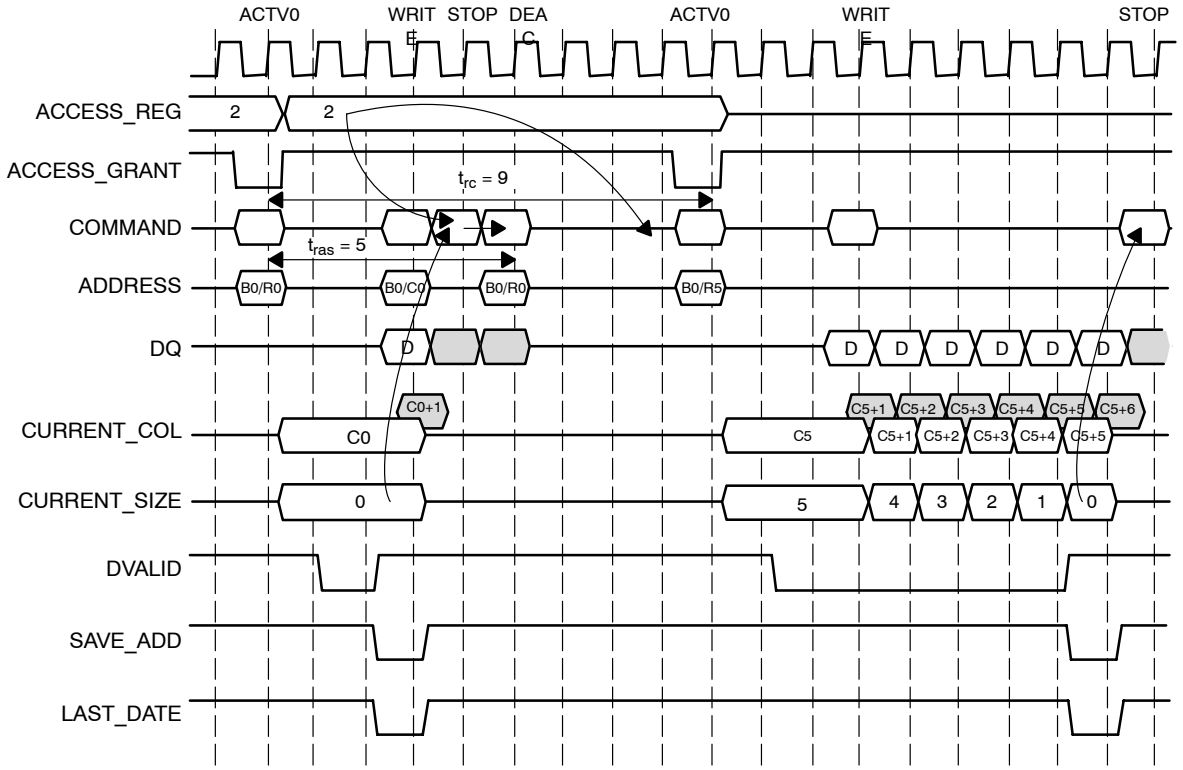
Note: READ (burst reduced to 4) is interrupted by a WRITE request (reduced to 3) pending on a bank and row already active.

Figure 16. SDRAM Read Single Half-Word Followed by a Write Byte



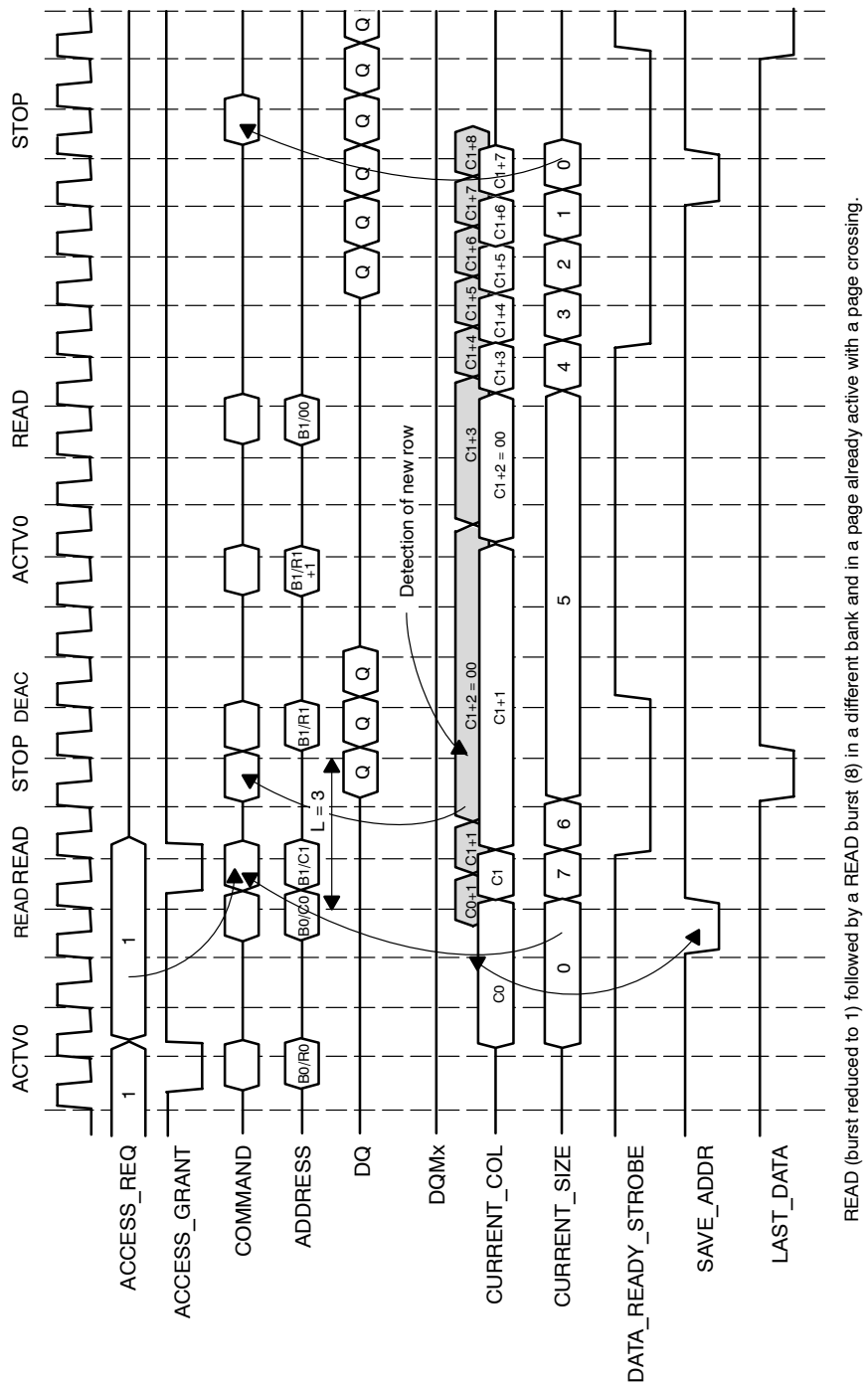
Note: READ (burst reduced to 1) is followed by a single-byte WRITE in the same bank but on a different page.

Figure 17. SDRAM Write Single Followed by Write Burst 6 on the Same Bank and Different Page



Note: WRITE (burst reduced to 1) is followed by a WRITE (6) in the same bank but on a different page.

Figure 18. SDRAM Read Single Half-Word Followed by a Read Burst 8 With Page Crossing



4 Traffic Controller Memory Interface Registers

The OMAP5910 traffic controller base address is 0xFFFE:CC00.

Table 9 lists the traffic controller registers. Table 10 through Table 28 describe the register bits.

The EMIF slow interface configuration register provides access to EMIFS boot, operation, and power-down options (see Table 13).

Table 9. Traffic Controller Registers

Name	Description	R/W	Size	Address	Reset Value
IMIF_PRIO	IMIF priority register	R/W	32 bits	0xFFFE:CC00	0x0000 0000
EMIFS_PRIO	EMIF slow priority register	R/W	32 bits	0xFFFE:CC04	0x0000 0000
EMIFF_PRIO	EMIF fast priority register	R/W	32 bits	0xFFFE:CC08	0x0000 0000
EMIFS_CONFIG_REG	EMIF slow interface configuration register	R/W	32 bits	0xFFFE:CC0C	0x0000 00yy (See Table 13 for details on the y values.)
EMIFS_CS0_CONFIG	EMIF slow interface chip-select configuration register nCS0	R/W	32 bits	0xFFFE:CC10	0x0000 FFFB
EMIFS_CS1_CONFIG	EMIF slow interface chip-select configuration register nCS1	R/W	32 bits	0xFFFE:CC14	0x0010 FFFB
EMIFS_CS2_CONFIG	EMIF slow interface chip-select configuration register nCS2	R/W	32 bits	0xFFFE:CC18	0x0010 FFFB
EMIFS_CS3_CONFIG	EMIF slow interface chip-select configuration register nCS3	R/W	32 bits	0xFFFE:CC1C	0x0000 FFFB
EMIFF_SDRAM_CONFIG	EMIF fast interface SDRAM configuration register 1	R/W	32 bits	0xFFFE:CC20	0x0061 8800
EMIFF_MRS	EMIF fast interface SDRAM MRS register	R/W	32 bits	0xFFFE:CC24	0x0000 0037
TIMEOUT1	Timeout1	R/W	32 bits	0xFFFE:CC28	0x0000 0000
TIMEOUT2	Timeout2	R/W	32 bits	0xFFFE:CC2C	0x0000 0000
TIMEOUT3	Timeout3	R/W	32 bits	0xFFFE:CC30	0x0000 0000

Table 9. Traffic Controller Registers (Continued)

Name	Description	R/W	Size	Address	Reset Value
ENDIANISM	Endianism	R/W	32 bits	0xFFFE:CC34	0x0000 0000
	Location not used			0xFFFE:CC38	
EMIFF_SDRAM_CONFIG_2	EMIF fast interface SDRAM configuration register 2	R/W	32 bits	0xFFFE:CC3C	0x0000 0003
EMIFS_CFG_DYN_WAIT	EMIF slow wait-state configuration register	R/W	32 bits	0xFFFE:CC40	0x0000 0000

Table 10. IMIF Priority Register (IMIF_PRIO)

Bit	Field	Description	Access	Reset Value
31–16	Reserved	Reserved. These pins must always be written as 0.	R	All 0s
15–12		LB host: Number of consecutive transfers while other ports are waiting + 1.	R/W	0000
11–8		System DMA: Number of consecutive transfers while other ports are waiting + 1.	R/W	0000
7	Reserved		R	0
6–4		DSP (CPU or DMA): Number of consecutive transfers while other ports are waiting + 1.	R/W	000
3	Reserved		R	0
2–0		ARM: Number of consecutive transfers while other ports are waiting + 1.	R/W	000

Table 11. EMIF Slow Priority Register (EMIFS_PRIO)

Bit	Field	Description	Access	Reset Value
31–16	Reserved	Reserved. These pins must always be written as 0.	R	All 0s
15–12		LB host: Number of consecutive transfers while other ports are waiting + 1.	R/W	0000
11–8		System DMA: Number of consecutive transfers while other ports are waiting + 1.	R/W	0000
7	Reserved			0
6–4		DSP (CPU or DMA): Number of consecutive transfers while other ports are waiting + 1.	R/W	000
3	Reserved			0
2–0		ARM: Number of consecutive transfers while other ports are waiting + 1.	RW	000

Table 12. EMIF Fast Priority Register (EMIFF_PRIO)

Bit	Field	Description	Access	Reset Value
31–16	Reserved	Reserved. These pins must always be written as 0.	R	All 0s
15–12		LB host: Number of consecutive transfers while other ports are waiting + 1.	R/W	0000
11–8		System DMA: Number of consecutive transfers while other ports are waiting + 1.	R/W	0000
7	Reserved			0
6–4		DSP (CPU or DMA): Number of consecutive transfers while other ports are waiting + 1.	R/W	000
3	Reserved			0
2–0		ARM: Number of consecutive transfers while other ports are waiting + 1.	R/W	000

Table 13. EMIF Slow Interface Configuration Register (EMIFS_CONFIG_REG)

Bit	Field	Value	Description	Access	Reset Value
31–5	Reserved		Read is undefined. Writes must be zero.	R	All 0
4	FR		Ready signal. This bit is a copy of the FLASH.RDY input pin as sampled by TC clock.	R	x
		0	FLASH.RDY pin is low.		
		1	FLASH.RDY pin is high.		
3	PDE		Global power-down enable. This bit is used by EMIFS, EMIFF, and IMIF as an enable for dynamic power down, clock auto-gating. Note, however, that PDE must be set in conjunction with individual power down bits for IMIF and SDRAM before clocks will be cut.	R/W	0
		0	Power down not enabled		
		1	Power down enabled		
2	PWD_EN		IMIF power-down enable. Controls IMIF internal clock enable:	R/W	0
		0	IMIF power down not enabled		
		1	IMIF power down enabled		
			Also note that PWD_EN is one of the prerequisites to meet TC idle. PWD_EN must be set before the memory interface can acknowledge a TC idle request.		
1	BM		MPU boot mode. This bit is sampled at reset from the MPU_BOOT device pin. BM enables CS0 and CS3 address decode swapping.	R/W	x
		0	CS0 [0000:0000 – 01FF:FFFF] CS3 [0C00:0000 – 0DFF:FFFF]		
		1	CS0 [0C00:0000 – 0DFF:FFFF] CS3 [0000:0000 – 01FF:FFFF]		
			Since BM is read/write, care must be exercised not to write the bit since there is potential to inadvertently modify EMIFS memory mapping.		
0	WP		Write protect bit. Enables write protection for all flash devices.	R/W	0
		0	$\overline{\text{FLASH.WP}}$ output pin is set low.		
		1	$\overline{\text{FLASH.WP}}$ output pin is set high.		

The four EMIF slow chip-select configuration registers (see Table 14) are used to select the protocols and timings to be used for handshake with devices

connected to CS0-CS3 (corresponding to device pins $\overline{\text{FLASH.CS0}}$ - $\overline{\text{FLASH.CS3}}$). Table 15 describes the memory types, and Table 16 describes the wait cycles insertion.

Table 14. EMIF Slow Chip-Select Configuration Registers (EMIFS_CS0_CONFIG...EMIFS_CS3_CONFIG)

Bit	Field	Value	Description	Access	Reset Value
31–22	Reserved		Read is undefined. Writes must be zero.	R	All 0
21	FL		Specifies how EMIFS handles addressing when performing 32-bit writes to the OMAP5910 16-bit data bus.	R/W	0
		0	The address is incremented for the second 16-bit access (default).		
		1	The address is not incremented for the second 16-bit access. This bit is valid only when EMIFS is configured for 16-bit data bus width (BW = 0). This bit has no effect for read operations.		
20	BW		Specifies EMIFS data bus width.	R/W	
		0	16-bit bus. This is the appropriate setting for OMAP5910.		
		1	Reserved. Do not use this setting on OMAP5910. (BW bit reset value depends on the chip-select: For CS0 and CS3, BW = 0; For CS1 and CS2, BW = 1. If CS1 or CS2 is to be used, BW must first be written to 0 since OMAP5910 only supports 16-bit bus.)		
19	Reserved		Read is undefined. Writes must be zero.	R	0
18:16	RDMODE		Read mode select (see Table 15)	R/W	000
15:12	PGWST/WELEN		For read accesses, number of wait states for page mode ROM reads within a page. For write accesses, the length of $\overline{\text{WE}}$ pulse duration.	R/W	1111
11:8	WRWST		Numbers of wait states for write operation	R/W	1111
7:4	RDWST		Number of wait states for asynchronous read operation (see Table 16). Number of inserted clock cycles in protocol (value matches the value programmed in Intel flash devices).	R/W	1111

Table 14. EMIF Slow Chip-Select Configuration Registers (EMIFS_CS0_CONFIG...EMIFS_CS3_CONFIG) (Continued)

Bit	Field	Value	Description	Access	Reset Value
3	Reserved		Read is undefined. Writes must be zero.	R/W	U
2	RT		Retiming control register:	R/W	0
		0	The data is not retimed.		
		1	The data coming from the external bus is retimed with the CLK.		
1:0	FCLKDIV		EMIFS internal reference clock divider:	R/W	11
		00	Reference clock = TC clock divided by 1		
		01	Reference clock = TC clock divided by 2		
		10	Reference clock = TC clock divided by 4		
		11	Reference clock = TC clock divided by 6		

Table 15. Memory Type

RDMODE	Memory
000	Asynchronous read
001	Page mode ROM read—4 words per page
010	Page mode ROM read—8 words per page
011	Page mode ROM read—16 words per page
100	Synchronous burst read
Others	Reserved. Do not use.

Table 16. Wait Cycles Insertion

RDWST	Number of Cycles Inserted
0	2
1	3
2	4
3	5
4	6
5	7

There is no automatic hardware adjustment of the programmed latencies when the system clock frequency changes.

The following restrictions apply when synchronous burst read Intel protocol is selected:

- ❏ Only full-page burst mode is supported
- ❏ Only sequential data access order is supported
- ❏ Only 1 clock cycle data duration mode is supported (there is no advantage in supporting 2 clock cycle duration since FLASH.CLK may be divided).

Page crossing is supported in page mode ROM burst read.

In asynchronous read mode, $\overline{\text{FLASH.ADV}}$ is activated during one FLASH.CLK cycle in order to ensure compatibility with burst flash.

Table 17. EMIF Fast Interface SDRAM Configuration Register 1 (EMIFF_SDRAM_CONFIG)

Bit	Field	Value	Description	Access	Reset Value
31–28	Reserved		Read is undefined. Writes must be zero.	R	All 0
27	CLK		SDRAM clock disable. See section 4.3.3.6, *SDRAM Clock Disable*, for details related to disabling the SDRAM clock.	R/W	0
		0	Clock is not disabled.		
		1	Clock is disabled.		
			CLK is one of the prerequisites to meet TC idle. CLK must be set before the memory interface can acknowledge a TC idle request.		

Table 17. EMIF Fast Interface SDRAM Configuration Register 1 (EMIFF_SDRAM_CONFIG) (Continued)

Bit	Field	Value	Description	Access	Reset Value
26	PWD		SDRAM power-down enable. Controls power-down state of SDRAM interface:	R/W	0
		0	SDRAM interface is not powered down.		
		1	SDRAM interface is powered down.		
			PWD is one of the prerequisites to meet TC idle. PWD must be set before the memory interface can acknowledge a TC idle request.		
25–24	SDRAM_ FREQUENCY		SDRAM frequency range. Selects one of four SDRAM timing configurations based on clock latencies. See Table 19.	R/W	00
		00	SDF0 (reset value)		
		01	SDF1		
		10	SDF2		
		11	SDF3		
23–8	ARCV		Autorefresh counter register value. Sets the interval between partial refresh requests to the SDRAM. See Section 3.3.4, *SDRAM Autorefresh Initialization*, for formula and example.	R/W	0x6188
7–4	SDRAM_TYPE		Set the SDRAM internal organization (see Table 18)	R/W	0000
3–2	ARE		Autorefresh enable is set, the EMIF issues a REFRESH command for the SDRAM when the autorefresh counter expires. Burst refresh can be 1,4, or 8. There are separate burst and auto-refresh counters.	R/W	00
		00	Autorefresh disable		
		01	Autorefresh enable		
		10	Autorefresh by burst of 4 commands		
		11	Autorefresh by burst of 8 commands		

Table 17. EMIF Fast Interface SDRAM Configuration Register 1 (EMIFF_SDRAM_CONFIG) (Continued)

Bit	Field	Value	Description	Access	Reset Value
1	SD_RET		SDRAM retiming:	R/W	0
		0	Data is single buffered with the return clock from SDRAM.		
		1	Data from SDRAM is double buffered. Data is first clocked on return clock from SDRAM, then with the OMAP5910 internal SDRAM clock.		
0	SLRF		When set, places the SDRAM in self-refresh mode. Mode is automatically exited upon the generation of any SDRAM access.	R/W	0

This register is used to configure the SDRAM, interface timing, autorefresh setup, and powerdown modes of the EMIFF interface. Table 18 describes the internal organization. Table 19 describes the frequency range.

Table 18. SDRAM Internal Organization

SDRAM_TYPE	Memory Size (M Bits)	Size Of Data Bus	Number Of Banks
0000	16	8	2
0001		8	4†
0010		16	2
0011		16	4†
0100	64	8	2
0101		8	4
0110		16	2
0111		16	4
1000	128	8	2†
1001		8	4

† Unavailable bank number (not supported). Do not use this setting.

Table 18. SDRAM Internal Organization (Continued)

SDRAM_TYPE	Memory Size (M Bits)	Size Of Data Bus	Number Of Banks
1010		16	2†
1011		16	4
1100	256	8	2†
1101		8	4
1110		16	2†
1111		16	4

† Unavailable bank number (not supported). Do not use this setting.

*Table 19. Frequency Range(*see parameter definitions below Table 1−19)

ac Parameters	SDF0 (Cycles)	SDF1 (Cycles)	SDF2 (Cycles)	SDF3 (Cycles)
t_{rc}	9	5	3	2
t_{ras}	5	3	2	2
t_{rp}	3	2	2	2
t_{rcd}	3	2	2	2
t_{rrd}†	2	2	2	2
t_{dpl}(trwl)‡	–	–	–	–
t_{dal}	–	–	–	–
t_{rsc}	2	2	2	–

† Write is never interrupted by precharge command directly.
‡ Neither read or write with auto-precharge is supported.

Table 20. SDRAM Timing Requirements (see parameter definitions below)

ac Parameters	SDRAM Timing Requirements (ns)	Meeting this Timing With SDRAM.CLK = 60 MHz (16.7 ns Period)
t_{rc}	80	5
t_{ras}	48	3
t_{rp}	24	2
t_{rcd}	24	2
t_{rrd}†	16	1
t_{dpl}(trwl)‡	8	–
t_{dal}	27	–
t_{rsc}	2	1

† Write is never interrupted by precharge command directly.
‡ Neither read or write with autoprecharge is supported.

ac Parameter definitions

Trl Delay time, ACTV, MRS, REFR, or SLFR to ACTV, MRS, REFR, or SLFR command

Tras Active to precharge command period

Trp Precharge to active command period

Trcd Active command to column command (same bank)

Trrd Active command to active (b) command period

Tdpl (Trwl) Write recovery of data-in to precharge lead time

Tdal Last data into active latency

Trsc Mode register set cycle time

For 60 MHz, timing can be met by using the SDF1 timing configuration.

This register, when written, programs the SDRAM MRS (default) and EMRS configuration registers. In default mode, a write to the register initiates an MRS

request to the SDRAM. In EMRS mode, a write to this same register initiates an EMRS request. Reading this register does not issue an external transaction. Table 21 describes the bits for the MRS mode. Table 22 describes the bits for the EMRS mode.

Table 21. EMIF Fast Interface SDRAM MRS Register—Default (EMIFF_MRS)

Bit	Field	Value	Description	Access	Reset Value
31–10	Reserved		Read is undefined. Writes must be zero.	R	All 0
9	WBST		Reserved (must be set to 0, single location access is not supported).	R/W	0
8–7	Reserved		Read is undefined. Writes must be zero.	R/W	00
6–4	CASL		CAS latency:	R/W	011
		001	CAS latency = 1		
		010	CAS latency = 2		
		011	CAS latency = 3 (default at reset)		
3	S/I		Reserved (must be set to 0, interleave burst-type is not supported).	R/W	0
2–0	PGBL		Reserved (must be set to 111, only full page burst length is supported).	R/W	111

Note: When the CONF_MOD_EMRS_CTRL bit field (bit 13) of the OMAP5910 control register (MOD_CONF_CTRL_0) is set, the device reconfigures bank settings to write out the EMIFF_MRS register as EMRS commands (see Table 22).

Table 22. EMIF Fast Interface SDRAM MRS Register—EMRS Mode (EMIFF_MRS)

Bit	Field	Value	Description	Access	Reset Value
31–5	Reserved		Read is undefined. Writes must be zero.	R	See Note 1
4–3	TCSR		SDRAM EMRS register temperature compensated self-refresh setting:	R/W	See Note 1
		00	70 degrees Celsius maximum case temperature		
		01	45 degrees Celsius maximum case temperature		
		10	15 degrees Celsius maximum case temperature		
		11	85 degrees Celsius maximum case temperature		
			Bit descriptions are given with respect to standard SDRAM devices and must be verified with the actual SDRAM chosen for the application.		
2–0	PASR		SDRAM EMRS register partial array self-refresh coverage setting:	R/W	See Note 1
		000	All banks		
		001	Half array		
		010	Quarter array		
		011	Reserved		
		100	Reserved		
		101	Reserved		
		110	Reserved		
		111	Reserved		
			Bit descriptions are given with respect to standard SDRAM devices and must be verified with the actual SDRAM chosen for the application.		

Notes: 1) Reset value is defined by the default mode of the register (see Table 21).

The three time-out registers store the number of clock cycles before DSP, DMA, LCD, LB requests are made high-priority in dynamic priority scheme for the TC (see Table 23 through Table 25).

Table 23. Time-Out 1 Register (TIMEOUT1)

Bit	Field	Description	Access	Reset Value
31–24	Reserved	Read is undefined. Writes must be zero.	R	All 0
23–16	Local bus		R/W	0x00
15:8	Reserved	Read is undefined. Writes must be zero.	R/W	All 0
7:0	DMA		R/W	0x00

Table 24. Time-Out 2 Register (TIMEOUT2)

Bit	Field	Description	Access	Reset Value
31–24	Reserved	Read is undefined. Writes must be zero.	R	All 0
23–16	DSP		R/W	0x00
15–8	Reserved	Read is undefined. Writes must be zero.	R/W	All 0
7–0	LCD		R/W	0x00

Table 25. Time-Out 3 Register (TIMEOUT3)

Bit	Field	Description	Access	Reset Value
31–0	Reserved	Read is undefined. Writes must be zero.	R	All 0

The endianism register (ENDIANISM) is used to control endian conversion in the DSP memory management unit endianism block.

Table 26. Endianism Register (ENDIANISM)

Bit	Field	Value	Description	Access	Reset Value
31–2	Reserved		Read is undefined. Writes must be zero.	R	All 0
1	SWAP	0	Byte swap (8 bits)	R/W	0
		1	Word swap (16 bits)		
0	EN	0	Endian conversion is disabled (default).	R/W	0
		1	Endianism is enabled.		

Table 27. EMIF Fast Interface SDRAM Configuration Register 2 (EMIFF_SDRAM_CONFIG_2)

Bit	Field	Value	Description	Access	Reset Value
31–2	Reserved		Read is undefined. Writes must be zero.	R	All 0
1	RFRSH_ RST		SDRAM self-refresh on warm reset. RFRSH_RST determines what action the TC SDRAM controller takes toward setting SDRAM to self-refresh mode in the event of a warm system reset.	R/W	1
		0	SDRAM is not entered to self-refresh mode.		
		1	SDRAM is entered to self-refresh mode upon warm system reset.		
0	RFRSH_ STBY		SDRAM self-refresh on standby. After the TC receives an idle request from the clock generation module, RFRSH_STBY determines what action the TC SDRAM controller takes toward setting SDRAM to self-refresh mode prior to acknowledging the idle request.	R/W	1
		0	SDRAM enters self-refresh mode.		
		1	SDRAM enters self-refresh mode prior to the TC acknowledging an idle request.		

Table 28. EMIF Slow Wait State Configuration (EMIFS_CFG_DYN_WAIT)

Bit	Field	Value	Description	Access	Reset Value
31–4	Reserved		Read is undefined. Writes must be zero.	R	All 0
3	DYNW_CS3		Specifies function of FLASH.RDY for CS3.	R/W	0
		0	Enable classic not-ready for EMIFS CS3.		
		1	Enable dynamic not-ready for EMIFS CS3.		
2	DYNW_CS2		Specifies function of FLASH.RDY for CS2.	R/W	0
		0	Enable classic not-ready for EMIFS CS2.		
		1	Enable dynamic not-ready for EMIFS CS2.		
1	DYNW_CS1		Specifies function of FLASH.RDY for CS1.	R/W	0
		0	Enable classic not-ready for EMIFS CS1.		

Table 28. EMIF Slow Wait State Configuration (EMIFS_CFG_DYN_WAIT)

Bit	Field	Value	Description	Access	Reset Value
0	DYNW_CS0	1	Enable dynamic not-ready for EMIFS CS1. Specifies function of FLASH.RDY for CS0.	R/W	0
		0	Enable classic not-ready for EMIFS CS0.		
		1	Enable dynamic not-ready for EMIFS CS0.		

5 Interfacing Memories With the OMAP5910 Device

This section provides two examples of how to connect memories to the OMAP5910 device. Many scenarios can be considered using different kinds of memories. For flash memories, Intel and Hitachi products are used. For SDRAM and SRAM, Hitachi and Toshiba products are used, respectively.

The Intel flash memory has a total capacity of 160M bits (8M x 8 x 2 chips and 2M x 16). Program code uses two Intel 28F64J3A memories, and data uses Intel 28F32J3A. The power supply voltages for these memories range from 2.7V to 3V. Hitachi memory has a total capacity of 96M bits in this example. Two flash memories are used for program code, and one flash memory is used for data. The power supply voltage also ranges from 2.7V to 3V.

Hitachi SDRAM (HM52Y64165F) has a total capacity of 64M bits (4M x 16). Its power supply voltage ranges from 2.5V to 2.8V. Toshiba SRAM (JT5MM6A-AD) has 8M bit capacity and 2.7V to 3V power supply voltage ranges.

Figure 19 shows external memory interconnection using Intel flash memory, and Figure 20 shows external memory interconnection using Hitachi flash memory.

Figure 19. External Memory Interconnection Using Intel Flash Memory

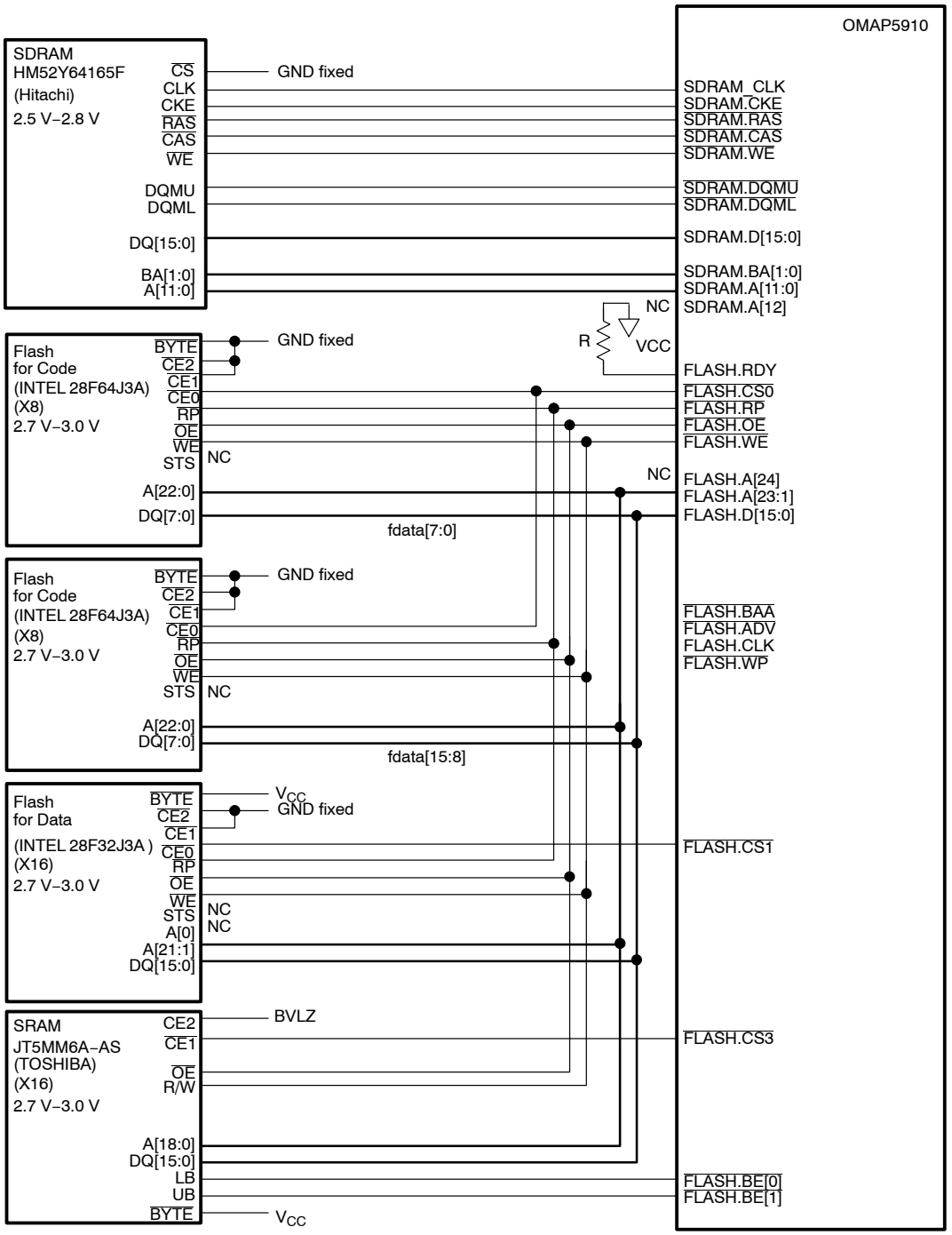
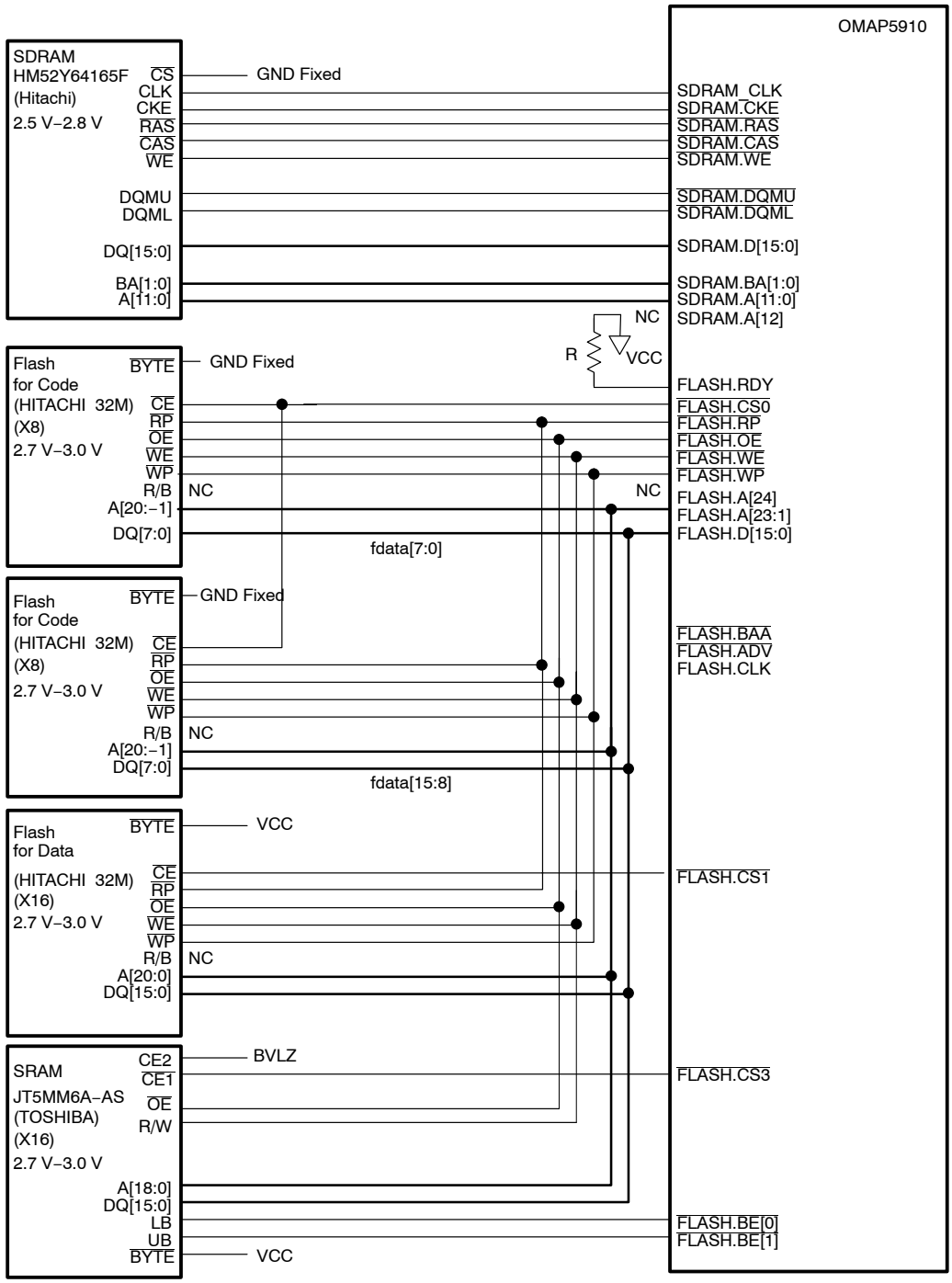


Figure 20. External Memory Interconnection Using Hitachi Flash Memory



Index

A

B

C

D

E

F

H

I

L

M

O

P

R

S

T

W

Invalid Literature requested. Please click on TI Home Page and browse through to get the desired page.

SEMICONDUCTOR Products

eXpressDSP™ SOFTWARE AND TOOLS
Current software and development tools information to support your DSP development including a multimedia eXpressDSP™ guided tour.

OTHER TI PRODUCTS
- Educational & Productivity Solutions
- DLP™ Products
- Sensors and Controls
- TI-RFid™

SEMICONDUCTORS

Digital Signal Processors (DSPs)
Analog and Mixed-Signal
- Amplifiers and Comparators
- Data Converters
- Interface Products
- Power Management
- Audio
- Clock and Timers
- Control and Monitoring
- MSP430 Ultra-Low-Power MCU
- RF
- Telecom
- Video and Imaging

OMAP™ Platform
Digital Logic
ASIC
Microcontrollers
Programmable Logic
Military
Automotive
Space Products
FIFOs

TOOLS AND SOFTWARE

All TI Tools
- Software
- Hardware Tools
- Software Tools

SEMICONDUCTOR RESOURCES

Technical Documentation
- Datasheets
- Application Notes
- Block Diagrams

Distributors
Sales Representatives
Samples
Pricing/Availability
Packaging
Glossary
Training
Webcasts