Technical notes on using Analog Devices DSPs, processors and development tools
Contact our technical support at dsp.support@analog.com and at dsptools.support@analog.com
Or visit our on-line resources http://www.analog.com/ee-notes and http://www.analog.com/processors

The ADSP-TS20x TigerSHARC® Processor On-chip SDRAM Controller

Contributed by Maikel Kokaly-Bannourah *Rev 1 – February 4, 2004*

1 Introduction

This Engineer-to-Engineer Note introduces characteristics of the ADSP-TS20x TigerSHARC® processor on-chip SDRAM controller. Although, this document is based on the ADSP-TS201 TigerSHARC processor, a list highlighting the differences between TigerSHARC processor family derivatives (such as ADSP-TS201, ADSP-TS202, and ADSP-TS203) is provided at the end of this document.

The internal signal chain is shown with the necessary address-mapping scheme. The command truth table gives detailed information about execution in the SDRAM. The power-up sequence summarizes detail information to start successful designs. A timing overview demonstrates the performance for different access modes. For basic understanding of SDRAM memories, refer to the application note *The ABC of SDRAMemory (EE-126)* [5].

2 Table of Contents

3 Listings

4 Signal Chain of SDRAM

Figure 1 illustrates the signal chain between the ADSP-TS201S, the on-chip SDRAM controller, and the external memory device for a 64-bit bus configuration:

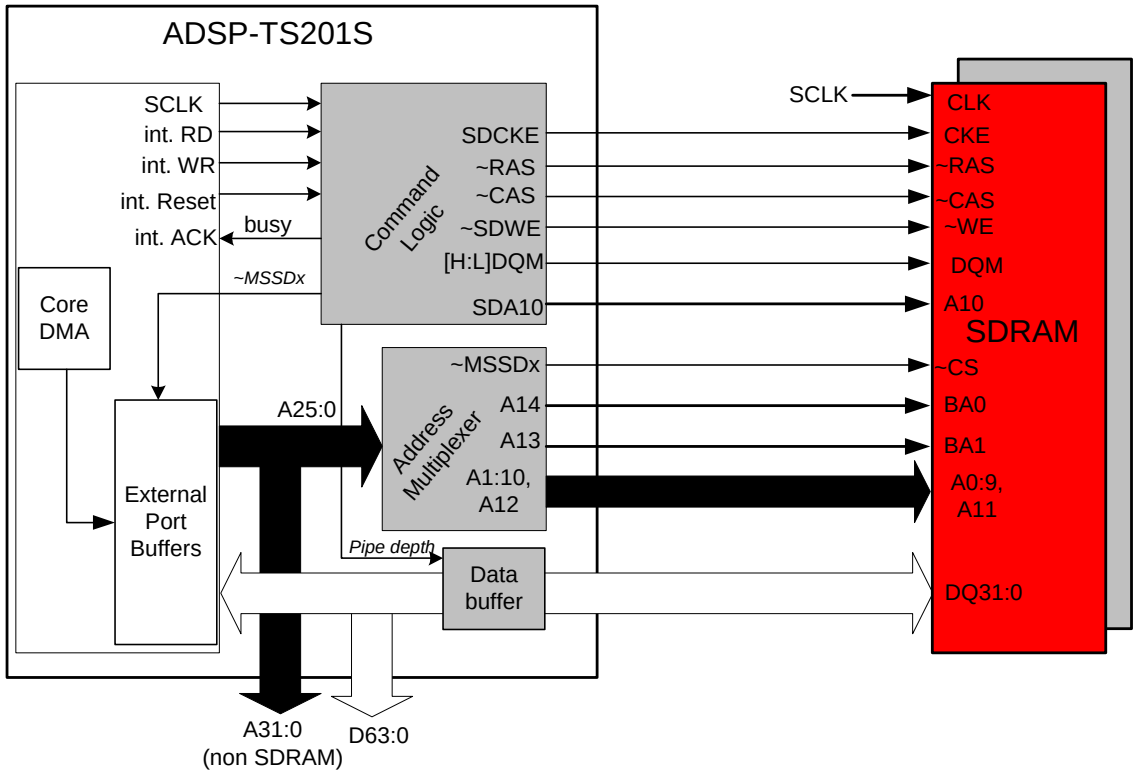


Figure 1. ADSP-TS201 to SDRAM Signal Chain (64-bit bus configuration)

Consider the three parts of the signal flow:

ADSP-TS201S (core, DMA engine, I/O processor, and the address buffer)
The SDRAM Controller (control interface, delay buffer, and address multiplexer)
SDRAM device

5 On-Chip Controller Architecture

The synchronous interface between the ADSP-TS201S processor and the on-chip controller are described in four basic parts:

5.1 Controller Command Interface

Because of the two different timing protocols, the TigerSHARC processor's internal commands are converted to comply with the JEDEC standard for SDRAMs. The 125 MHz (maximum) external clock is used for synchronous operation. The TigerSHARC's internal request lines or strobes are used to access the SDRAM with pulsed commands. The controller's internal ACK line inserts variable wait states to the processor during overhead cycles, caused by DRAM technology.

5.2 TigerSHARC Processor Output FIFO

The TigerSHARC processor's output FIFO is active for external port addresses like SDRAM. The processor's six-stage FIFO depth supports address pipelining for high-speed non-sequential read operations without performance loss.

5.3 Controller Address Multiplexer

Every first read or write action is issued in multiplexed mode. A maximum of 8192 rows (64-bit bus configuration) and 16384 rows (32-bit bus configuration) within 1024 columns can be addressed.

5.4 Controller Data Delay Buffer

If systems incorporate a heavy busload, an additional data buffer is used to decouple the input from the capacitive load. This delay buffer, in conjunction with an external buffer for SDRAM control and address lines, reduces additional logic.

5.5 SDRAM Types

The ADSP-TS201S processor's on-chip SDRAM controller interface supports various LVTTL (3.3V) as well as mobile low-power SDRAM devices (2.5V), depending on size and internal organization (I/O capability, number of rows, and page size). The following table summarizes all the supported types:

Size	I/O capability	Row x Page	Size	I/O capability	Row x Page
16 Mbits	1M x 16	2K x 256	256 Mbits	8M x 32	8K x 256
	2M x 8	2K x 512		16M x 16	8K x 512
	4M x 4	2K x 1024		32M x 8	8K x 1024
64 Mbits	2M x 32	2K x 256	512 Mbits	8M x 32	8K x 256
	4M x 16	4K x 256		16M x 16	8K x 512
	8M x 8	4K x 512		32M x 8	8K x 1024
	16M x 4	4K x 1024			
128 Mbits	4M x 32	4K x 256			
	8M x 16	4K x 512			
	16M x 8	4K x 1024			

Table 1. Supported SDRAM devices

6 Command Coding

6.1 Controller's Pin Definition

Pin	Type	Description
~MSSD[3:0]	I/O/T (pu)	SDRAM banks Memory select signals
~RAS	I/O/T (pu)	Row select signal
~CAS	I/O/T (pu)	Column select signal
~SDWE	I/O/T (pu)	Write enable signal
HDQM	O/T (pu)	Mask data high lane signal
LDQM	O/T (pu)	Mask data low lane signal
SDA10	O/T (pu)	Address10 /command select signal
SDCKE	I/O/T (pu/pd)	Clock enable signal
A[1:10,:12-15]	I/O/T	addresses for 64-bit
A[0:9,11-15]	I/O/T	addresses for 32-bit
A[11:15]	I/O/T	Bank select signal
D[63:0]	I/O/T	Data signals

I = input, O = output, T = Hi-Z, pd = pull-down, pu = pull-up

Table 2. ADSP-TS201 SDRAM Controller Pins Description

6.2 Controller Command Truth Table

Table 3 provides an overview of all commands provided by the SDRAM controller. These commands are automatically handled by the interface.

SDCKE = high								
CMD	**SDCKE(n-1)**	**SDCKE(n)**	**~MSSDx**	**~RAS**	**~CAS**	**~SDWE**	**SDA10**	**ADDR**
MRS	1	1	0	0	0	0	V	V
ACT	1	1	0	0	1	1	V	V
RD	1	1	0	1	0	1	0	V
WR	1	1	0	1	0	0	0	V

SDCKE = high, no validity of address								
CMD	**SDCKE(n-1)**	**SDCKE(n)**	**~MSSDx**	**~RAS**	**~CAS**	**~SDWE**	**SDA10**	**ADDR**
NOP	1	1	1	x	x	x	x	x
BST	1	1	0	1	1	0	x	x
REF	1	1	0	0	0	1	1	x
PREA	1	1	0	0	1	0	1	x

Commands with SDCKE transition								
CMD	**SDCKE(n-1)**	**SDCKE(n)**	**~MSSDx**	**~RAS**	**~CAS**	**~SDWE**	**SDA10**	**ADDR**
SREF En	1	0	0	0	0	1	x	x
SREF Ma	0	0	x	x	x	X	x	x
SREF Ex	0	1	1	x	x	x	x	x

x=don't care, v=valid data input, 0=logic 0, 1=logic 1, En=entry, Ma=maintain, Ex=exit

Table 3. SDRAM Commands Truth Table

Although the SDCKE line toggles in an asynchronous manner, the commands are sampled synchronous to the CLK signal.

Note that *Power-down* and *Suspend* modes are not supported, and that the controller does not allow *auto precharge*. Lastly, keep in mind that *all SDRAM commands are fully transparent to the user*.

6.3 Setup and Hold Times

The synchronous operation uses the external clock as a reference. Commands, addresses, and data are latched at the rising edge of clock. The valid time margin around the rising edge is defined as setup time (time before rising edge) and hold time (time after rising edge) to guarantee that both the controller and the SDRAM are working together reliably. Signal slew rates, propagation delays (PCB), and capacitive loads (devices) influence these parameters and should be taken into consideration. Refer to the *ADSP-TS201S Data sheet* for SDRAM interface AC signal specifications.

6.4 Simplified State Diagram

The following state diagram (Figure 2) shows all possible SDRAM commands sequences to help analyze the controller's functionality.

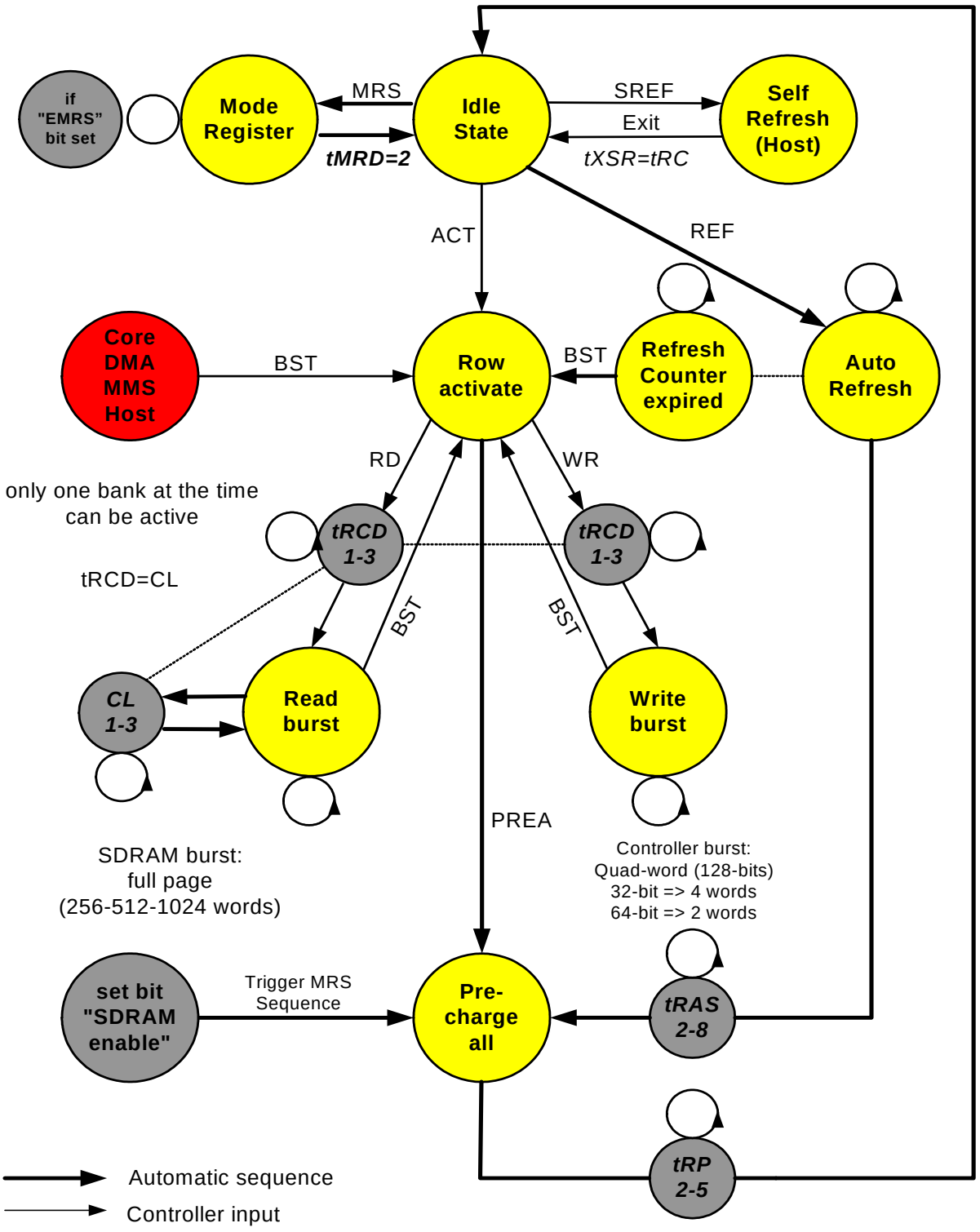


Figure 2. ADSP-TS201S SDRAM Controller Simplified State Diagram

7 SDRAM Controller Properties

Following, the ADSP-TS201S processor's on-chip SDRAM controller properties are examined:

7.1 Address Mapping Scheme

There are various possibilities when accessing the SDRAM. For instance, all rows in a bank (or all banks in a row) can be accessed sequentially. PC DIMM modules are accessed in a different manner compared to a typical DSP application. The ADSP-TS201S controller uses a hardware map scheme optimized for digital signal processing.

The address mapping scheme is decoded from the page size and the bus width (both configurable by software in the SDRCON and SYSCON registers respectively; refer to section 8 *SDRAM Programming*). For more information regarding the address mapping scheme, refer to the SDRAM chapter of the *ADSP-TS201 TigerSHARC Processor Hardware Reference* [1].

Figure 3 reproduces an example of the controller's address mapping for 64-bit data. In bank A, the SDRAM's columns are sequentially accessed until the end of the row. Similarly, the SDRAM's rows are sequentially selected until the bank's end.

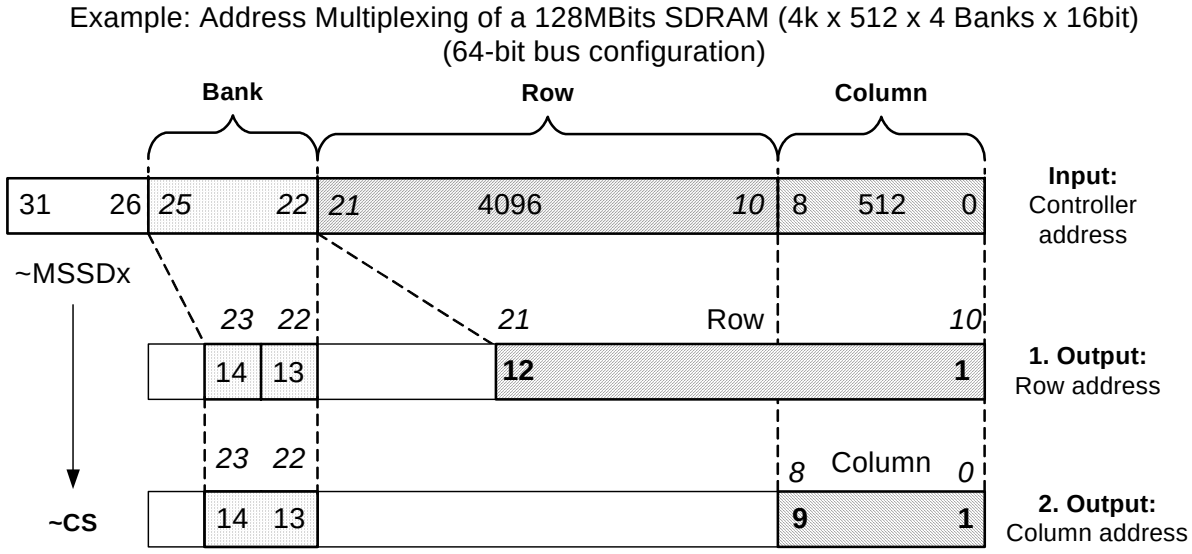


Figure 3. ADSP-TS201S SDRAM Controller Mapping Scheme Example

Note that only one bank may be active at a time, which results in overhead cycles when switching between banks (off-bank accesses). Similarly, moving from one row to another (off-page access) results in the same overhead cycles. Figure 4 shows how the ADSP-TS201S TigerSHARC processor's on-chip SDRAM controller accesses SDRAM.

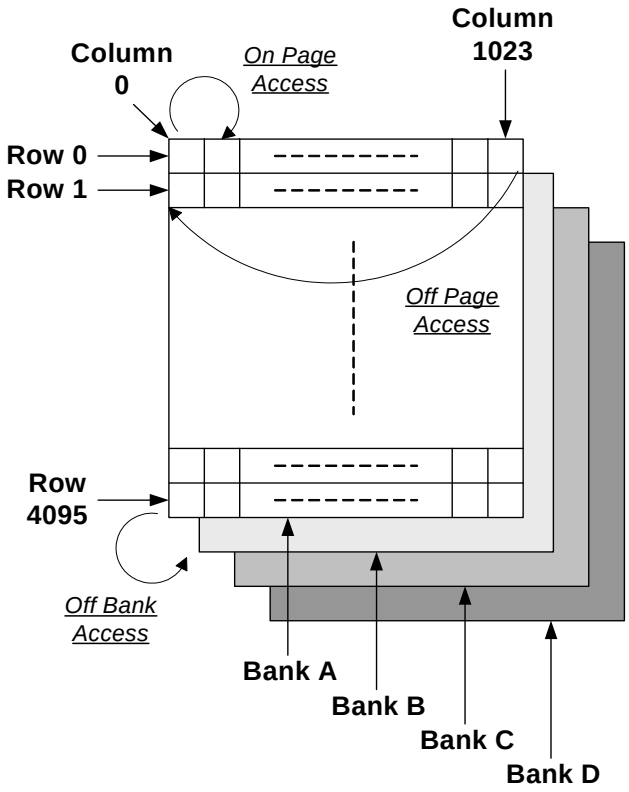


Figure 4. ADSP-TS201S Processor SDRAM Controller Access Structure

7.2 TigerSHARC Processor SDRAM Memory Select Signals (~MSSDx)

The ADSP-TS201 TigerSHARC processor has four SDRAM Memory Select lines (~MSSD[3:0]) to address the four SDRAM memory banks:

- ~MSSD0 – SDRAM address memory space: 0x4000 0000-0x4400 0000.
- ~MSSD1 – SDRAM address memory space: 0x5000 0000-0x5400 0000.
- ~MSSD2 – SDRAM address memory space: 0x6000 0000-0x6400 0000.
- ~MSSD3 – SDRAM address memory space: 0x7000 0000-0x7400 0000.

These signals can be used for SDRAM accesses only. In this memory region, the controller's address multiplexer will be active.

7.3 Burst Stop (BST)

Although the controller works in burst mode, there is one way to interrupt the burst with the burst stop command. BST is issued if the next instruction is:

- Non external SDRAM access (access to another TigerSHARC bank)
- Core access (depending on the number of accesses, delay and external port FIFOs state)
- DMA operation (external port DMA to SDRAM interrupted by a higher priority DMA)

- Refresh counter expired (refresh period counter)
- SDRAM read to write and write to read transitions
- SDRAM off page/bank access
- ~HBR asserted (host interface)
- During a Bus Transition Cycle (multiprocessing)

7.4 Data Mask Function ([H:L]DQM)

The [H:L]DQM pins are used by the controller to mask write operations. HDQM masks the SDRAM DQ buffers when performing 32-bit writes to even addresses in a 64-bit bus configuration. LDQM masks the SDRAM DQ buffers when performing writes to odd addresses in a 64-bit bus configuration. This data mask function does not apply for read operations, where the LDQM and HDQM pins are always low (inactive).

This is summarized in the following table:

Bus Width*	**64-bit**			**32-bit**
Access type	**32-bit Even**	**32-bit Odd**	**64-bit**	**32-bit Even/Odd**
HDQM	1	0	0	x
LDQM	0	1	0	0

*Bus Width bit setting in SYSCON
x = don't care, 0 = logic 0, 1 = logic 1

Table 4. [H:L]DQM Pins Functionality

7.5 SDRAM Bank Select

The connections of the address pins as bank select lines for the SDRAM device varies, depending the operational voltage of the SDRAM device (standard or low-power) and the number of banks that the part has.

Standard SDRAMs
The next tables show the address lines selection for the different banks:

2-banked access		
Banks	**A[11:15]**	**SDA10**
Bank_A	0	0
Bank_B	1	0
Banks A/B	x	1

x = don't care, 0 = logic 0, 1 = logic 1

Table 5. Bank Select Pins for 2-Banked LVTTL SDRAMs

Note: Any address line from address range A[11:15] can be used for bank select as long as it is are not driven as a row or column address.

4-banked access			
Banks	**A[11,13,15]**	**A[12,14]**	**SDA10**
Bank_A	0	0	0
Bank_B	1	0	0
Bank_C	0	1	0
Bank_D	1	1	0
All Banks	X	X	1

x = don't care, 0 = logic 0, 1 = logic 1

Table 6. Bank Select Pins for 4-Banked LVTTL SDRAMs

Note: Any address line pair from address range A[11:15] can be used for bank select as long as they are not driven as a row or column address.

Low-Power SDRAMs

The following tables show address line selection for the different banks:

2-banked access		
Banks	**A[14:15]**	**SDA10**
Bank_A	0	0
Bank_B	1	0
Banks A/B	x	1

x = don't care, 0 = logic 0, 1 = logic 1

Table 7. Bank Select Pins for 2-Banked Low-Power SDRAMs

Note: Any address line from the two address lines A[14:15] can be used for bank select as long as it is not driven as row- or column address.

4-banked access			
Banks	**A[14]**	**A[15]**	**SDA10**
Bank_A	0	0	0
Bank_B	1	0	0
Bank_C	0	1	0
Bank_D	1	1	0
All Banks	X	X	1

x = don't care, 0 = logic 0, 1 = logic 1

Table 8. Bank Select Pins for 4-Banked Low-Power SDRAMs

Note: Address lines A[14:15] must be used for bank select.

7.6 Controller Address 10 (SDA10)

This pin provides a special solution to gain control of the SDRAM, even when the processor operates as a slave (multiprocessing). The SDA10 pin allows simultaneous access to all banks during a refresh and precharge-all command. This pin must be connected to the A10 pin of the SDRAM.

Note: The SDA10 pin replaces the processor's A[10] and A[11] pins in a 32-bi and 64-bit bus width configuration, respectively. Also, during access to the ~MSSDx space, these pins are not used.

7.7 Burst Mode

Although the SDRAM device is programmed for full-page burst, the controller uses *quad-word* (128-bits) burst mode. For 32-bit bus width, the burst length is 4 words; for 64-bit width, the burst length is 2 words. Only the first read or write command is accompanied with an external address, which is driven by the controller until the burst is interrupted by another address.

Note that *the SDRAM Controller burst mode cannot be changed.*

7.8 Precharge All (PREA)

This command precharges all SDRAM banks simultaneously (SDA10 must be high to select all banks), which places the banks into the idle state.

Although only one bank may be active at a time, *the controller does not support a single bank precharge.*

7.9 Circular Access

The controller supports circular accesses during sequential read or writes within a page, performing a fixed throughput of 1 cycle/word. At the end of the page (defined in the SDRCON register), the instructions

```
xR3:0=Q[j1+=last_word];;
```

followed by

```
xR7:4=Q[j1+=first_word];;
```

are also executed with a 1-cycle/word throughput.

This functionality is similar to the IALU's circular buffering mode supported by the TigerSHARC processor's core.

7.10 Auto Refresh (REF)

After the SDRAM registers the CAS before RAS refresh command, it internally asserts CAS and delayed RAS to execute a row's refresh. The row interval (tREFI) is typically 15,6 or 7,8 µs, depending on the SDRAM device being used. The limit of refresh period is given by the *tREFmax* specification.

Note that *the controller does not support burst refresh.*

7.11 Self-Refresh (SREF)

The self-refresh is a very effective way of reducing the application's power consumption to a minimum. When a host processor gains control of the cluster bus, the TigerSHARC processor's SDRAM controller places the SDRAM into self-refresh mode before the bus is relinquished to the host. When triggered by an

internal timer, the SDRAM starts refreshing itself. The controller does not allow the software to place the SDRAM into self-refresh mode, only during host accesses.

7.12 Mode Register Set (MRS)

During the MRS command, the SDRAM controller initializes the SDRAM with the following *fixed* settings:

- Burst length is hardwired to full-page burst
- Burst type is hardwired to sequential burst
- Read latency (CL) is user programmable (1-3 cycles)

7.13 Extended Mode Register Set (EMRS)

Although, this is also an MRS command, the difference between the two sequences, MRS and EMRS, relies on the data sent via the address lines driven by the TigerSHARC processor SDRAM controller (also see section *SDRAM Bank Select*).

The Extended Mode register controls power-saving related functions and initializes the SDRAM with the following *fixed* settings:

- Partial Array Self-Refresh (PASR) –self-refresh coverage is set to 4 banks
- Temperature Compensated Self-Refresh (TCSR) - self-refresh frequency is set to its maximum (maximum case temperature 85°C)

8 SDRAM Programming

Before external bus transactions to SDRAM start, the system and SDRAM control registers must be configured accordingly.

8.1 SYSCON Register

SYSCON, the system configuration register, must be configured after hardware reset at the beginning of the source code.

This register is composed of different fields, although only the following applies for SDRAM:

Bus Width: For proper operation, make sure that the dedicated bus width bits settings is:

0: 32-bit bus
1: 64-bit bus

Note that if either the host or memory bus width is 64 bits, the multiprocessing width must also be 64 bits. Also, when using 64-bit mode, the TigerSHARC's address 0 pin becomes redundant, since its information is contained in the strobes. Therefore, this pin is not connected for SDRAM accesses in 64-bit mode.
The following diagram shows the ADSP-TS201S processor's external port data alignment for both 32- and 64-bit bus configurations:

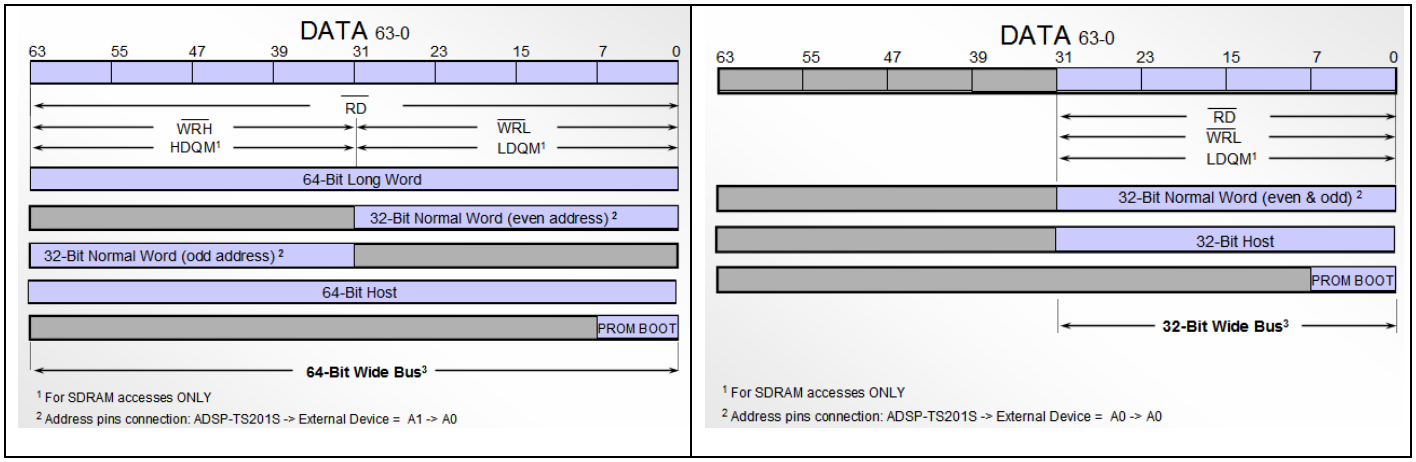


Figure 5. ADSP-TS201S Processor External Port Data Alignment

8.2 SDRCON Register

The SDRAM programming is done by the SDRCON register. Similar to SYSCON, configure SDRCON after hardware reset at the beginning of the source code before trying to access the SDRAM memory. In multiprocessor systemsprocessor , this register must be programmed to the same setup in every processor.

The SDRCON bit fields are:

SDRAM Enable: set whenever an SDRAM is present in the system.

CAS Latency: this bit defines the read latency (CL*min*) related to the vendor's device. This value can be set from 1 to 3 cycles, depending on the SDRAM characteristics and the clock frequency.

Pipe depth: this bit allows the SDRAM address and control lines to be pipelined. Setting this bit introduces a one-cycle delay during read and write accesses.

Page Boundary: this bit, which can be set to either 256, 512, or 1024 words, determines the page size of the SDRAM.

Refresh Counters: these bits enable coordination of the SOC bus clock rate (SOCCLK) with the SDRAM's required refresh rate. They select between 4 different refresh rates calculated with the following equation:

$$Cycles = \left(SOCCLK \cdot \frac{tREF}{Rows} \right) = \left(SOCCLK \cdot refresh\_rate \right)$$

The ADSP-TS201S processor supports refresh rates of 1100, 1850, 2200, or 3700 cycles. The following table illustrates how to select between the different rates based on the formula above.

SOC Clock frequency (MHz)	SDRAM Memory Refresh Rate (µsecs)	SDRAM Memory Refresh Rate (cycles)	ADSP-TS201 Supported Refresh rate (cycles)	SDRCON Bits [8:7]
SOCCLK @ 150	7.8	1170	1100	00
SOCCLK @ 250	7.8	1950	1850	01
SOCCLK @ 300	7.8	2340	2200	10
SOCCLK @ 150	15.6	2340	2200	10
SOCCLK @250	15.6	3900	3700	11
SOCCLK @300	15.6	4680	3700	11

* Note that the SOCCLK frequency equals ½ CCLK.

Table 9. Refresh Counter Values

Precharge to RAS delay (tRP): this bit defines the precharge time (*tRPmin*) related to the vendor's device. This value can be set from 2 to 5 cycles. Note that most SDRAMs specify this parameter in nanoseconds rather than number of cycles.

RAS to precharge delay (tRAS): this bit defines the row active time (*tRASmin*) related to the vendor's device. This value can be programmed from 2 to 8 cycles. Note that this parameter is also specified in nanoseconds rather than number of cycles.

Initialization Sequence: this bit determines the order of the MRS (Mode Register set) and refresh sequences. When set, MRS follows refresh in the SDRAM initialization sequence. Otherwise, the MRS precedes refresh:

Init Sequence = 0

- PREA command - places the SDRAM in the defined idle state.
- 8 REF commands - charges SDRAM's internal nodes.
- MRS command - initializes the SDRAM's working mode.

Init Sequence = 1

- PREA command - places the SDRAM in the defined idle state.
- MRS command - initializes the SDRAM's working mode.
- 8 REF commands - charges SDRAM's internal nodes.

EMR Enable: this bit, when set (=1), enables an Extended Mode Register set (EMRS) sequence, which is issued proceeding the initial MRS (see Initialization Sequence above). This bit must be set when interfacing to *low-power* SDRAM devices only. Otherwise, this bit should remain cleared.

tRAS and *tRP* are used for the refresh cycle, which can be expressed as*: tRC=tRAS+tRP*. Also, the specification requires that *tRASmin*, *tRPmin*, and *CLmin be* defined as a fraction of the *SCLK* period.

8.3 SDRAM Mode of Operation

During the MRS, address bits ADDR[13:0] of the SDRAM are used to program the device. The MRS, issued by the TigerSHARC processor with ID=000, is executed during power-up only. MRS initializes the following parameters:

- A[2:0] - Burst length is hardwired to full-page burst.
- A[3] - Burst type is hardwired to sequential burst.
- A[6:4] - Latency mode set according to the CAS latency programmed in SDRCON (1-3 cycles).
- A[13:7] - Hardwired to zero (reserved mode of operation for future needs).

Additionally, when the *EMR Enable* bit is set, a second MRS sequence (EMRS) is issued by the controller. The Extended Mode register controls power saving related functions and initializes the following parameters:

- A[2:0] - Partial array self-refresh (PASR). The self-refresh coverage is hardwired to four banks
- A[4:3] - Temperature-compensated self-refresh (TCSR). The self-refresh frequency is hardwired to its maximum case temperature, 85°C
- Bits 13–5 – Hardwired and set to 0

8.4 ADSP-TS201S Processor EZ-KIT Lite Evaluation Systems

The ADSP-TS201 TigerSHARC processor EZ-KIT Lite™ evaluation board includes 256 Mbits of SDRAM memory (32 Mbytes - 4M x 64bits). The following table shows the SDRAM control register (SDRCON) settings for the EZ-KIT Lite board:

SDRAM Device	SDRAM Clock (MHz)	SDRAM Settings	SDRCON
2x MT48LC4M32B2	100	SDRAM Enable = 1 CAS Latency = 2 Pipe Depth = 0 Page size = 256 Refresh rate = 3700 tRP = 2, tRAS = 5 Init Sequence = 1 EMR Enable = 0	0x5983

Table 10. ADSP-TS201 EZ-KIT Lite SDRAM Settings

8.5 SDRAM Setting Overview

Timing spec	Controller	Description
tCK	20–125 MHz	Clock cycle time, hardware
tREFmax	1100–3700 cycles	Refresh period
Clmin	1-3 cycles	Read latency
tRASmin	2-8 cycles	Activate to precharge
tRPmin	2-5 cycles	Precharge period
tRCD	tRCD = CL (fixed)	RAS to CAS delay
tRFCmin	tRC = tRAS+ tRP (fixed)	Activate period
tRRD	tRRD = tRC (fixed)	Activate A to activate B
tMRD	2 cycles (fixed)	Mode register to command
tXSR	tRC + tRP (fixed)	Exit self refresh to active

SDRAM (Mode Register Set)	
Power up Mode	PREA-MRS-REF or PREA-REF-MRS
Burst Mode	Sequential (fixed)
Burst Length	Full page (fixed)
CL	1-3 cycles

SDRAM (Extended Mode Register Set)	
Partial Array Self Refresh (PASR)	Self-refresh coverage set to four banks (fixed)
Temperature Compensated Self-refresh (TCSR)	Self-refresh frequency set to its maximum case temperature, 85°C (fixed)

9 Power-Up Sequence

Figure 6 shows the power-up procedure for the SDRAM interface:

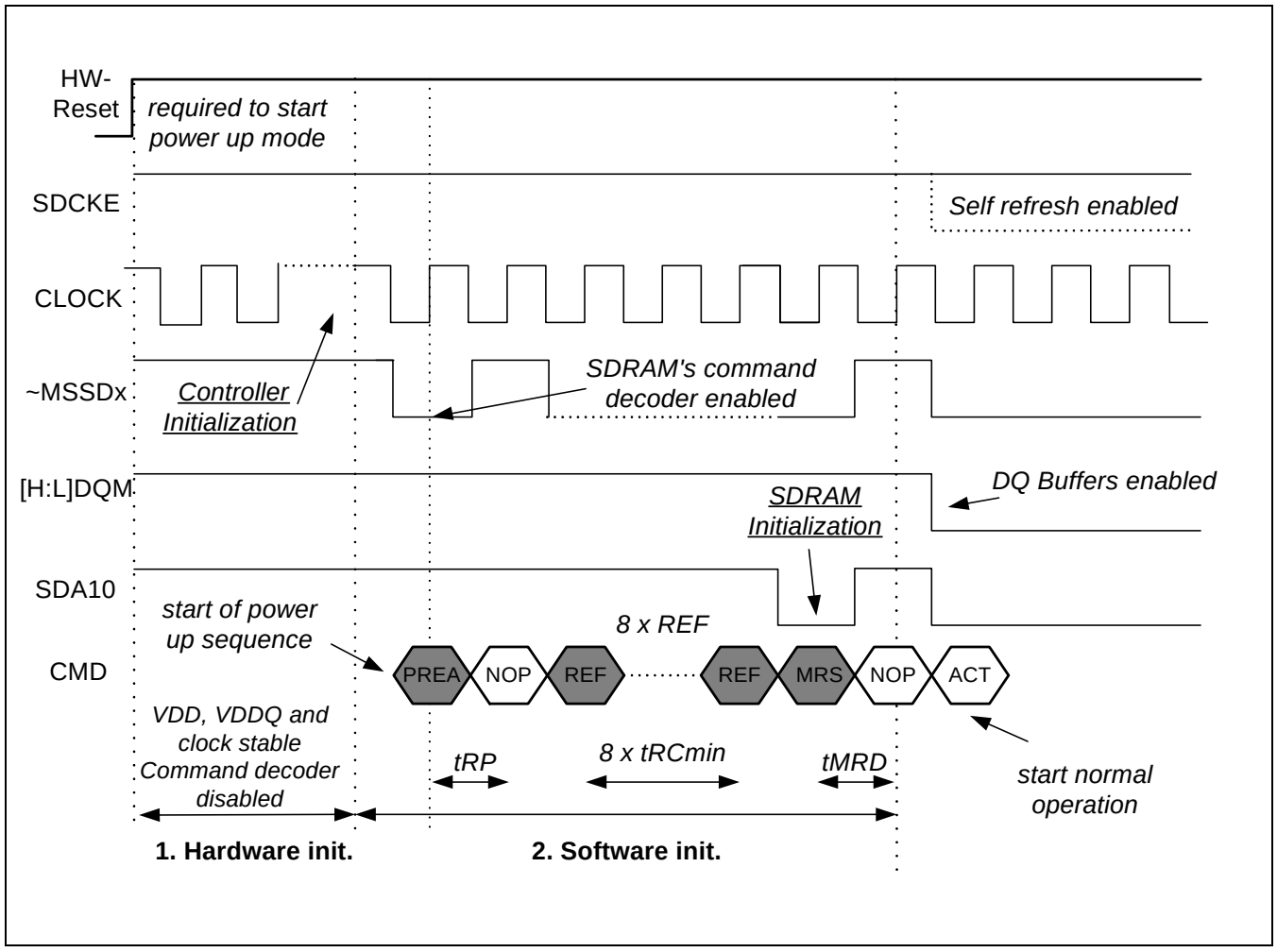


Figure 6. Power-Up and Initialization: PREA-REF-MRS

9.1 Hardware Initialization

After a hardware reset of the ADSP-TS201S processor, the clock and the SDRAM's power supply pins (VDD and VDDQ) must provide a stable signal for a typical minimum time of 200µs. The SDRCON register can be accessed only after this time has elapsed.

9.2 Software Initialization

Writing to the SDRCON register initializes the SDRAM controller. As soon as ~MSSDx is asserted, the SDRAM's command decoder is enabled.

The *Init Sequence* bit in SDRCON starts the initialization sequence (refer to section *8.2 SDRCON Register*). During this time, [H:L]DQM lines remains high, three-stating the SDRAM DQ buffers. Once

the initialization sequence is completed, buffers are enabled and the SDRAM is ready for normal operation.

Note that when the EMRS bit in the SDRCON register is set (required for interfacing to low-power SDRAM), an additional MRS command, which follows the MRS command shown in Figure 6, is executed (see 7.13 Extended Mode Register Set (EMRS) for more details).

The time elapsed before the first access to SDRAM can be represented as:

$$taccess \approx tRP + 8(tRAS + tRP) + tMRD \text{ (SCLK cycles)}$$

Note: To properly initialize the SDRAM, the first access is delayed with the internal acknowledge until the power-up sequence has finished.

9.3 SDRAM Initialization Example

This section shows how to initialize the SDRCON register according to the following device specifications.

- 2x 128Mb (4Mx32 bit) SDRAM
- 3.3 Volts (EMR bit cleared)
- 4 banks, page size: 256 words
- SDRAM clock: 100 MHz (SOCCLK: 250 MHz)
- Refresh cycles: 4K/64ms (15.6µsecs)
- Power-up mode: PRE - MRS - REF
- CLmin = 2 @100MHz, tRASmin = 42 ns, tRPmin = 18 ns
- No self-refresh and no buffering

Therefore, the SDRAM initialization code would look as follows:

```
j1 = j31 + 0x00005983;;    // SDRAM ENA=1, CL=2, pipedepth=0, page=256w
SDRCON = j1;;              // refresh rate=3700, trp=2, tras=5, init=1, EMR=0
```

Note that the minimum timing specifications for tRAS, tRP, and CL must be guaranteed. Setting these to values larger than the minimum required causes a loss of performance (i.e. longer delay cycles or unnecessary refresh cycles).

9.4 SDRAM Interface After Reset

After power-up when the ADSP-TS201S processor's reset pin is deasserted, the SDRAM-lines are in the following state:

Pin	State	Description
SDCKE	1	SDRAM Clock enabled
~MSSD[3:0]	1	Command decoder disabled
~RAS	1	Deselected
~CAS	1	Deselected
~SDWE	1	Deselected
[H:L]DQM	1	SDRAM data buffers disabled
SDA10	1	Access all banks simultaneously

Table 11. ADSP-TS201 Processor SDRAM Pin Description After Reset

10 DMA Transfers

Direct Memory Access (DMA) is a mechanism for transferring data without core involvement. Transfers can be between internal and external memory, or between external memory and an external peripheral.

10.1 Internal Memory and SDRAM

The TigerSHARC processor DMA controller can be used to transfer data from the processor's internal memory to external SDRAM. Similarly, the SDRAM can be used as the source, and the internal memory as the destination.

10.2 External Device and SDRAM (FLY-BY)

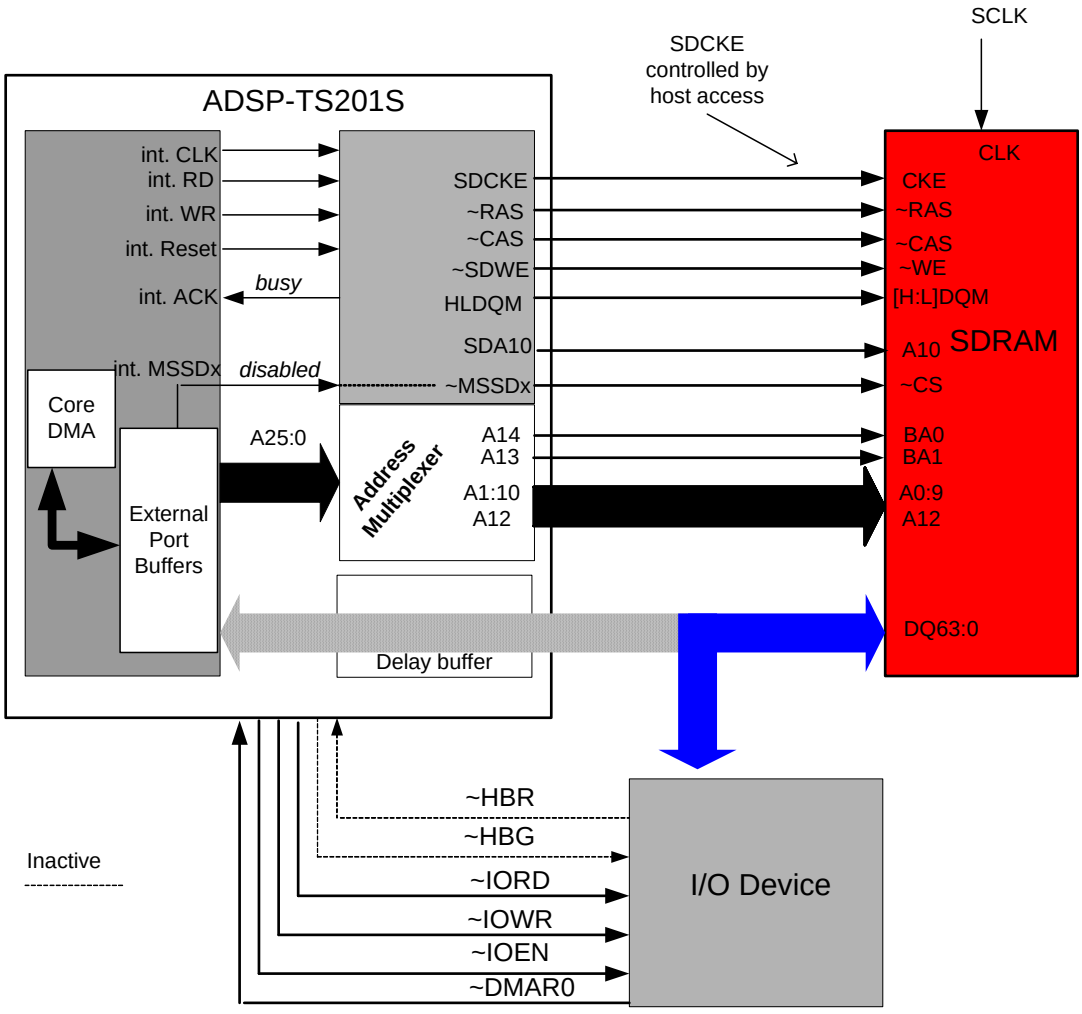


Figure 7. Signal Chain: Fly-by DMA and SDRAM

Fly-by transactions are used by the DMA controller to transfer data between an external I/O device and external SDRAM memory. The fly-by transaction type can be SDRAM memory to I/O (fly-by read) or I/O to SDRAM memory (fly-by write). A fly-by transaction is issued by a DMA channel that is programmed to execute fly-by.

In fly-by transactions, the SDRAM memory is controlled using the appropriate SDRAM controller signals (~MSSDx, ~RAS, ~CAS, and ~SDWE), and the I/O device is controlled with the ~IORD, ~IOWR, and ~IOEN pins.

The ~IOWR signal is used during fly-by read transactions (SDRAM memory read and I/O write). During fly-by write transactions (SDRAM memory write and I/O read), the signals ~IORD and ~IOEN are driven.

To transfer data, the I/O-device asserts the ~DMAR0 pin – only DMA channel 0 can be used for fly-by transactions. Each assertion (falling edge) of the ~DMAR0 signal represents one transaction. The ~IORD, ~IOWR, and ~IOEN pins are output by the current TigerSHARC processor's bus master and is used to control the I/O-device.

For more details on the TigerSHARC processor's DMA controller and fly-by mode, refer to the *"Direct Memory Access"* and *"SDRAM Interface"* chapters of the *ADSP-TS201 TigerSHARC Processor Hardware Reference* [1].

11 SDRAM Interface in Host Mode

Figure 8 shows a possible signal chain between the ADSP-TS201S processor, SDRAM, and a host processor.

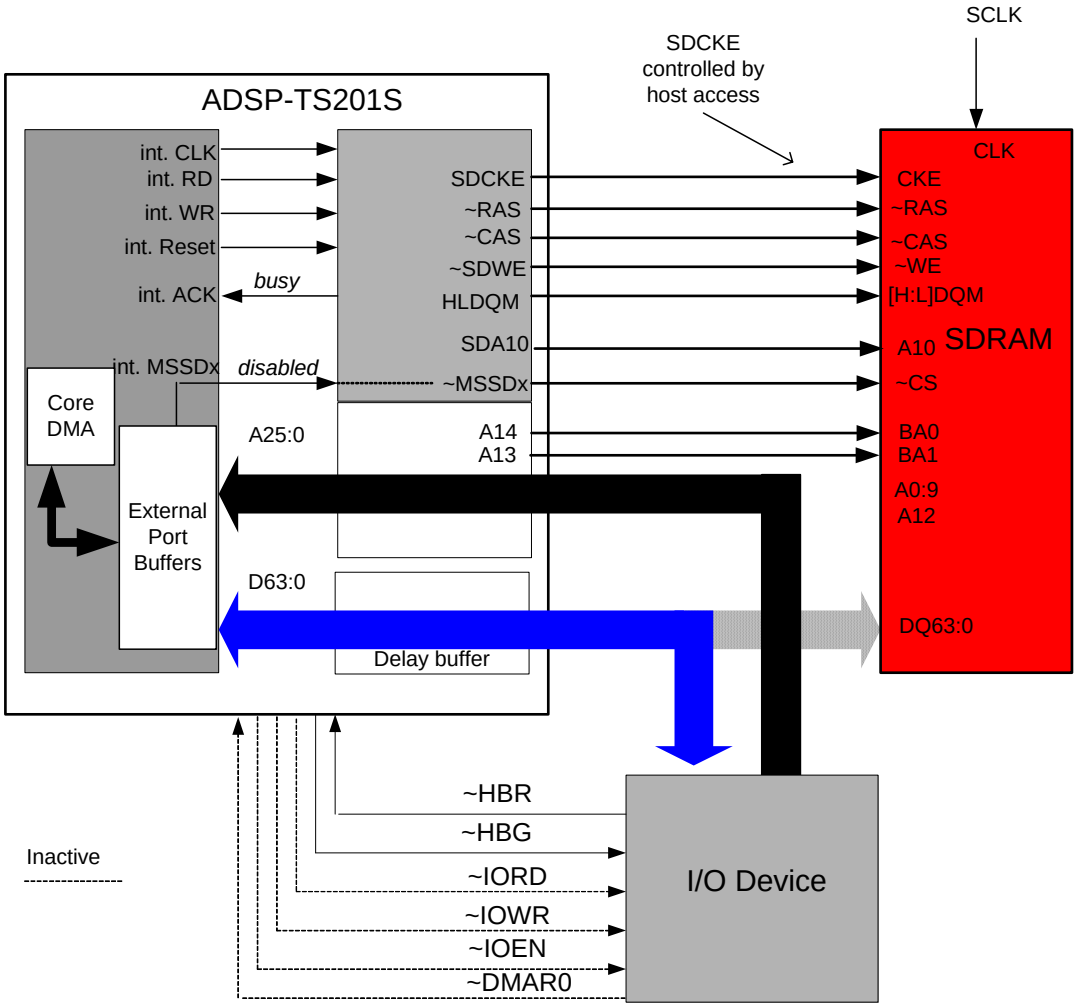


Figure 8. Signal Chain: Host to ADSP-TS201S Processor

Host accesses are initiated when the host asserts ~HBR, relinquishinging the external bus to the TigerSHARC processor. Thereafter, the TigerSHARC processor responds to this request by asserting ~HBG, granting the bus to the host. The host becomes bus master and can now access the TigerSHARC processor's internal resources through its multiprocessor memory space.

Note that *during ~HBG low, the SDRAM is in self-refresh mode.* This limits the host device to not being able to access the SDRAM using the TigerSHARC processor's SDRAM controller (fly-by mode). *Thus, during host transactions, the SDRAM can only be accessed by a host using its own SDRAM controller.*

However, a workaround for this may be possible if the host follows the given sequence:

1. Initially, the TigerSHARC processor is current bus master. To gain control of the bus, the host asserts *~HBR* and waits for the TigerSHARC processor to grant the bus *(~HBG)*
2. The host becomes bus master and is now able to access the internal resources of the TigerSHARC processor.
3. The host sets up a DMA transfer, preparing the TigerSHARC processor to transfer data to SDRAM upon host requests (DMA request line *~DMAR0*).
4. The host relinquishes the bus (de-asserts *~HBR*), which brings the SDRAM out of self-refresh mode
5. Finally, the host starts toggling *~DMAR0,* indicating to the TigerSHARC processor to start transmitting data to the SDRAM.

12 Multiprocessing

This section covers the arbitration logic used to guarantee multiprocessing systems.

12.1 Command Decoding

The ADSP-TS201S processor can be connected to a multiprocessor cluster of up to eight processors. Only one TigerSHARC processor can drive the bus at a time. To build a glueless hardware, the interface works in slave mode, as well, to detect these commands: MRS, REF, and SREF. The following pins are necessary for detection:

Pin	Type	Description
SDCKE	I/O/T	SDRAM Clock Enable
~MSSD[3:0]	I/O/T	Memory Select SDRAM (command input)
~RAS	I/O/T	Row Address Select (command input)
~CAS	I/O/T	Column Address Select (command input)
~SDWE	I/O/T	SDRAM Write Enable (command input)

I=input, O=output, T=three-state

Table 12. Multiprocessing Command Decoding

12.2 MRS Decoding

In TigerSHARC multiprocessor systems where SDRAM is used, *the Mode Register Set sequence is only issued by the processor with ID=000. Therefore, a processor with ID=000 must be present in every multiprocessor system.* The MRS sequence is only issued once before accessing the SDRAM for the first time. It is also important to note that the set-up value initialized in the SDRCON register must be identical for all processors in the cluster.

12.3 REF Decoding

This detection helps to synchronize all eight refresh counters. The slave's refresh counter is decremented each time the interface detects a refresh. This guarantees a periodic refresh, and similar to the MRS sequence, requires the exact same settings of SDRCON registers.

12.4 SREF Decoding

This detection helps to synchronize the self-refresh base. When a host requests the bus, the system master places the SDRAM into self-refresh mode. The slave processors recognize the SREF, freezing the refresh counters until self-refresh mode is exited.

12.5 Bus Transition Cycle

The bus transition (turn-over cycle) is issued when bus mastership changes between the different TigerSHARC processors sharing the cluster bus. This also implies a mastership transition between the SDRAM controllers. Since the new bus master unable to determine which row in the SDRAM is currently open, the current bus master issues a precharge-all command (PREA) automatically before the bus is

relinquished to the new bus master. This way, the new bus master can safely start accessing the SDRAM by simply issuing an activation command (ACT).

Figure 9 shows this sequence:

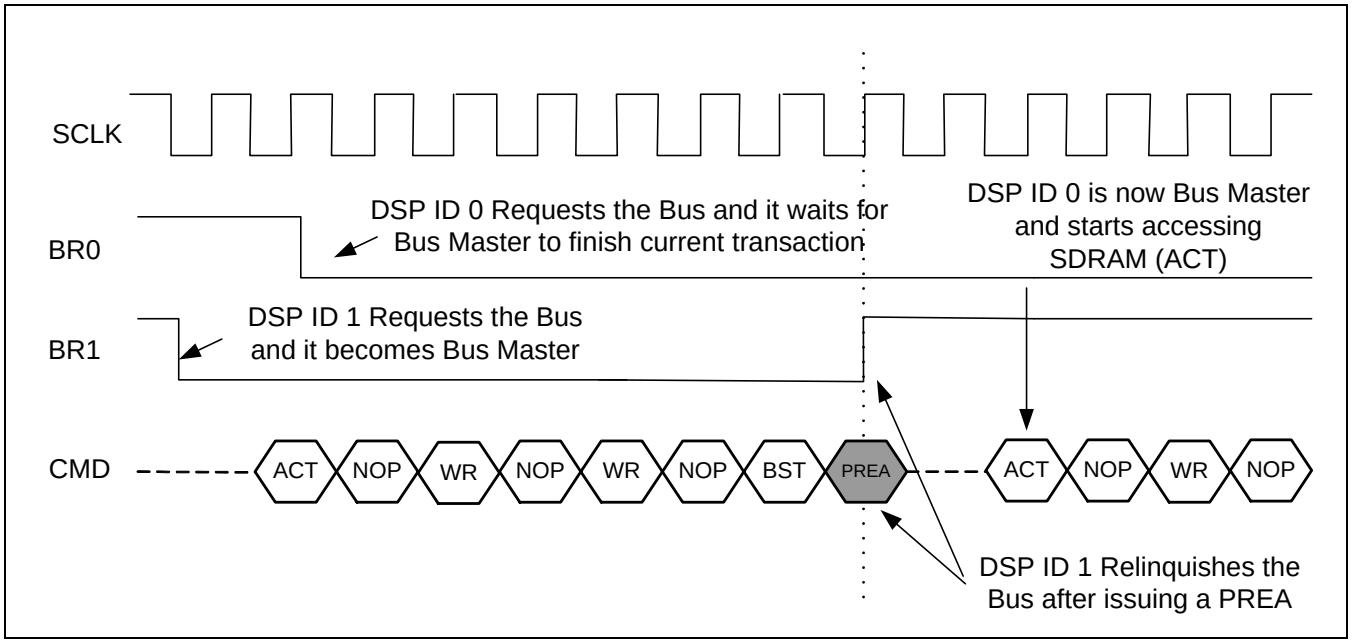


Figure 9. Bus Transition Cycle During SDRAM Accesses

13 SDRAM and Booting

13.1 Loader Kernel

Data must be loaded in SDRAM before runtime. This requires an initialized SDRAM before downloading the data during the boot scenario. The VisualDSP++™ loader utility provides this capability.

Three standard loader executables are available for booting, depending upon the boot mode (EPROM, link ports, or a host processor). *Note that independently from the selected boot mode, the SDRAM Control register (SDRCON) must be initialized before the user's application starts writing data to it. This must be done during the first 256-loader kernel words; otherwise, the data placed in the SDRAM will be corrupted.*

The steps are:

1. ~RESET must be de-asserted, ~BMS is sampled
2. Kernel (256 x 32 bit) is transmitted via DMA into the processor (0x00-0xFF) (~BMS or ~HBG continuously asserted)
3. Interrupt generation starts kernel execution, SDRAM and controller is initialized (~MSSDx asserted for SDRAM setup)
4. User's data is loaded into the processor/SDRAM (~BMS, ~HBG and ~MSSDx are toggling requesting the bus)
5. Kernel is now overwritten with user's code (~BMS or ~HBG continuously asserted)
6. Processor starts code execution at start address 0x00000000

13.2 Booting Modes

The boot mode is selected at the end of reset by sampling the ~BMS pin. At this stage, four options for beginning operation can be selected:

Mode	~BMS	DMA channel
EPROM	0 (Default)	Hardwired DMA Channel 0
Link Port	1	Link Port DMA Channels 8-11
Host	1	AutoDMA Channels 0-1
No boot	1	-

If ~BMS is sampled low during reset, EPROM boot is selected. However, if ~BMS is sampled high during reset, the processor goes into an IDLE state, waiting for host or link boot. Refer to [6]"*ADSP-TS201S TigerSHARC Processor Boot Loader Kernels Operation (EE-200)*", which describes each mode of operation in more detail.

14 SDRAM Interface Throughput

This section investigates the SDRAM controller's performance during data transfers, where the SDRAM is used as a source or destination. The following settings are used:

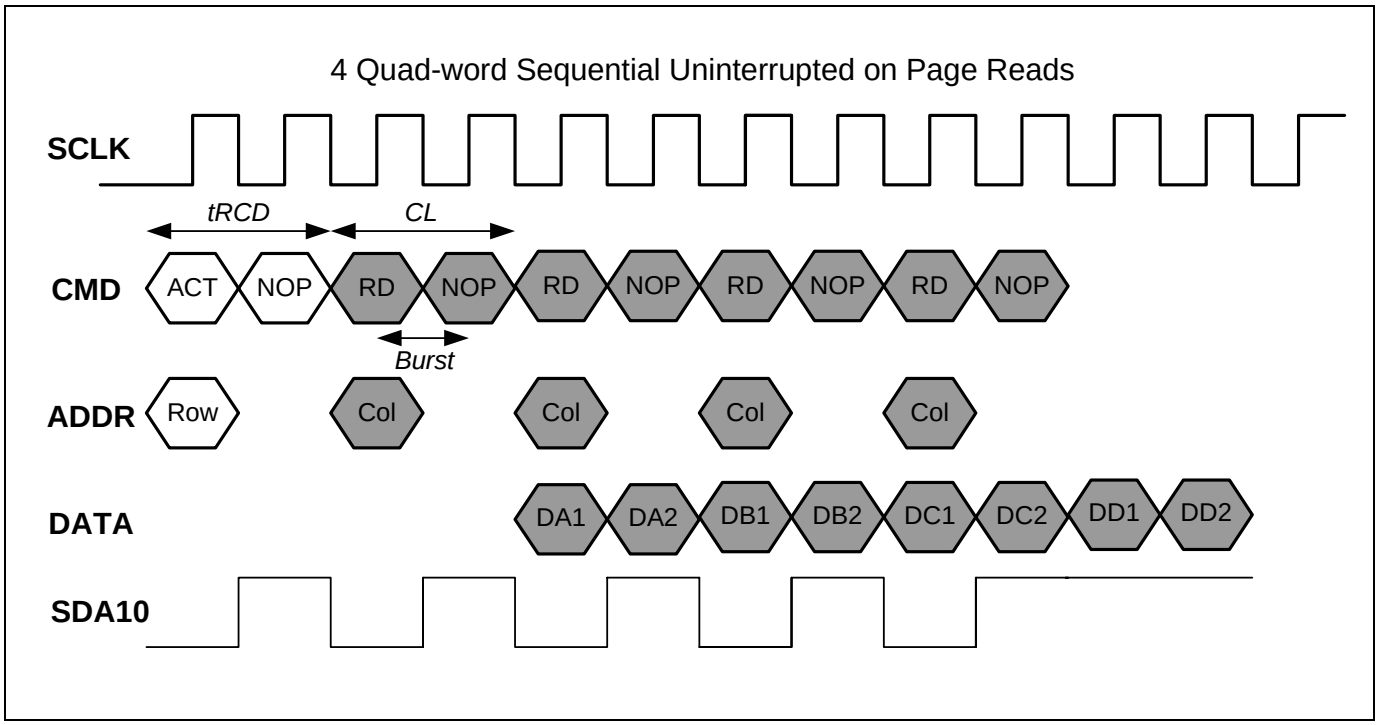
- Silicon = ADSP-TS201S Rev 0.0
- SCLK = 100 MHz (external port and SDRAM clock)
- CCLK = 500 MHz (core clock)
- SOCCLK = 250 MHz (SOC bus clock)
- SYSCON = 0x389067, for 64-bit bus configuration
- SDRCON = 0x5983, where Page Boundary = 256 words,
  Refresh rate = 3700 cycles,
  t_{RAS} = 5 cycles,
  t_{RP} = 2 cycle,
  CL = 2 cycles,

The following diagrams correspond to the different types of core/IOP accesses to SDRAM. Additionally, a list of instructions and SDRAM commands executed during each memory access is included.

Note that the external bus is configured to 64 bits wide. Therefore, every data word (e.g., DA1, DA2, or DB1.) read/written from/to the SDRAM will be 64 bits.

14.1 Sequential Reads Without Interruption

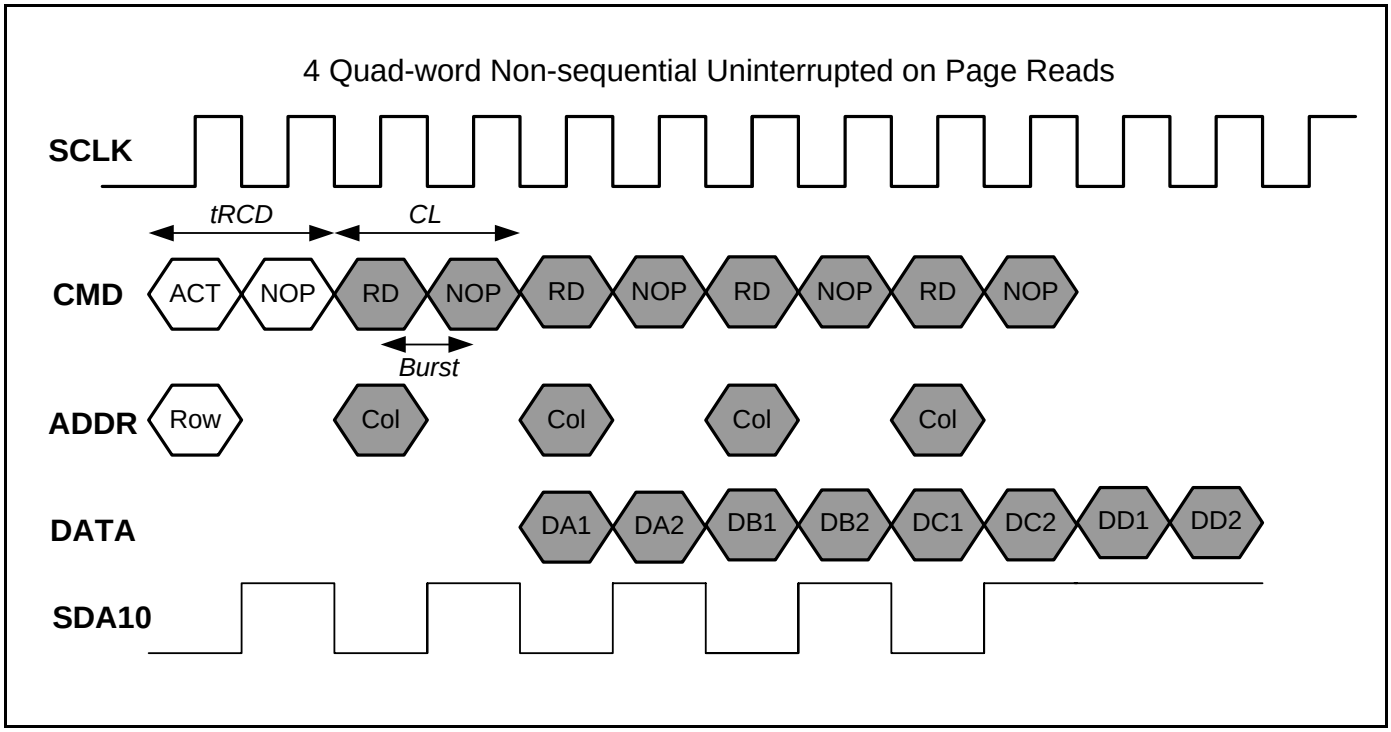
As shown below, uninterrupted sequential reads have a throughput of 1 word per cycle after the first word is read.



Nr. Cycles	Core	Controller	Data
1	**xr3:0=q[j1+=j4];;**	ACT	
2	int. ACK	NOP	
3	int. ACK	RD	
4	int. ACK	NOP	
5	**xr7:4=q[j1+=j4];;**	RD	DA1
6	int. ACK	NOP	DA2
7	**xr11:8=q[j1+=j4];;**	RD	DB1
8	int. ACK	NOP	DB2
9	**xr15:12=q[j1+=j4];;**	RD	DC1
10	int. ACK	NOP	DC2
11	int. ACK	…	DD1
12	int. ACK	…	DD2

14.2 Non Sequential Reads Without Interruption

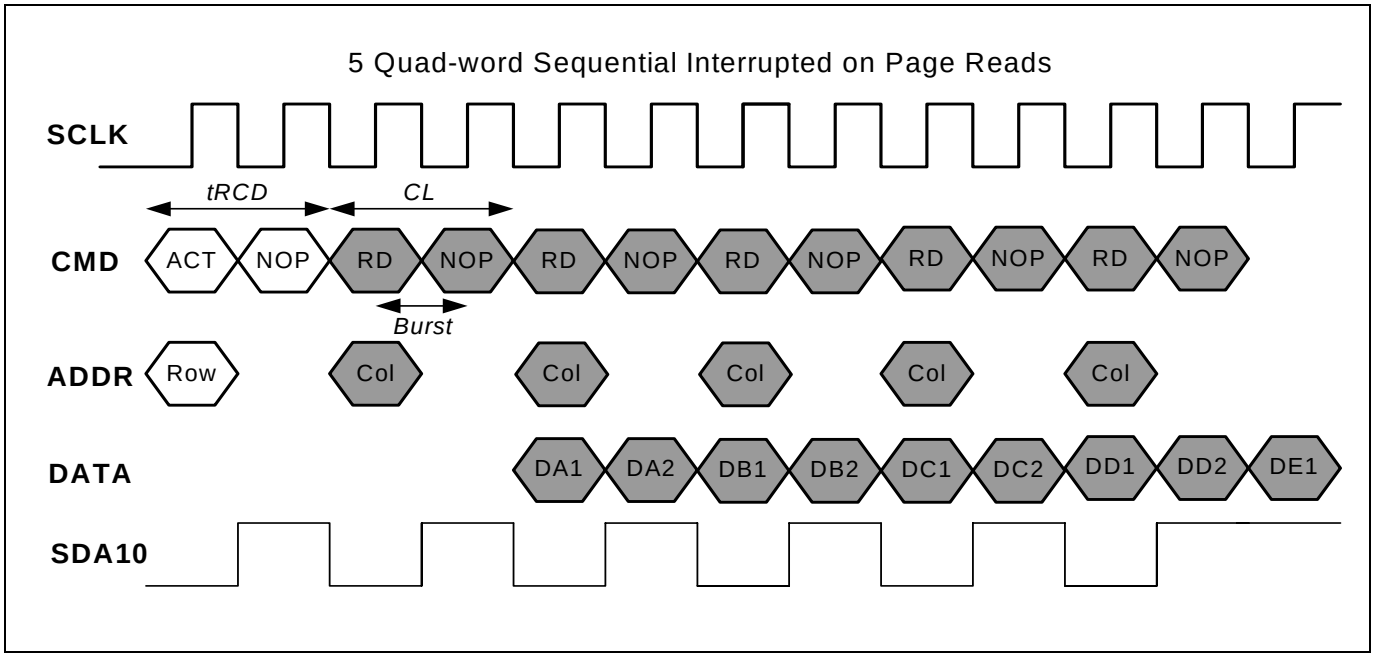
Similar to uninterrupted sequential reads, non-sequential reads also give a 1-word/cycle throughput.



Nr. Cycles	Core	Controller	Data
1	**xr3:0=q[j1+=j4];;**	ACT	
2	int. ACK	NOP	
3	int. ACK	RD	
4	int. ACK	NOP	
5	**xr7:4=q[j1+=j4];;**	RD	DA1
6	int. ACK	NOP	DA2
7	**xr11:8=q[j1+=j4];;**	RD	DB1
8	int. ACK	NOP	DB2
9	**xr15:12=q[j1+=j4];;**	RD	DC1
10	int. ACK	NOP	DC2
11	int. ACK	…	DD1
12	int. ACK	…	DD2

14.3 Sequential Reads with Minimum Interruption

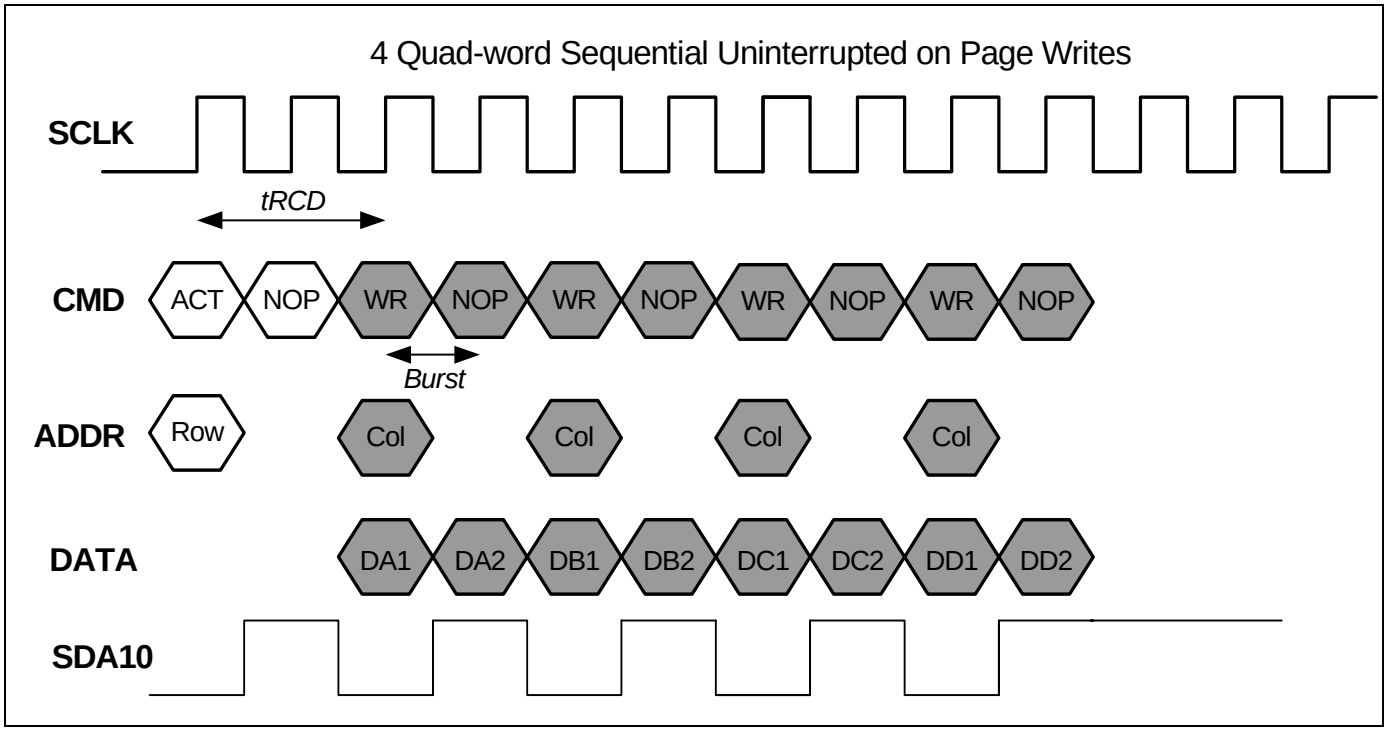
In this case, an additional core access is performed during a core/IOP read from SDRAM. As shown below, the SDRAM access does not get interrupted, and it operates similarly to uninterrupted reads (i.e., 1 word/cycle). Note that a higher number of core accesses during a read from SDRAM may cause a throughput decrease, depending upon system speed and core activity.



Nr. Cycles	Core	Controller	Data
1	**xr3:0=q[j1+=j4];;**	ACT	
2	int. ACK	NOP	
3	int. ACK	RD	
4	int. ACK	NOP	
5	**xr7:4=q[j1+=j4];;**	RD	DA1
6	**xr0 = r1 + r2;;**	NOP	DA2
7	**xr11:8=q[j1+=j4];;**	RD	DB1
8	int. ACK	NOP	DB2
9	**xr15:12=q[j1+=j4];;**	NOP	DC1
10	int. ACK	NOP	DC2

14.4 Sequential Writes Without Interruption

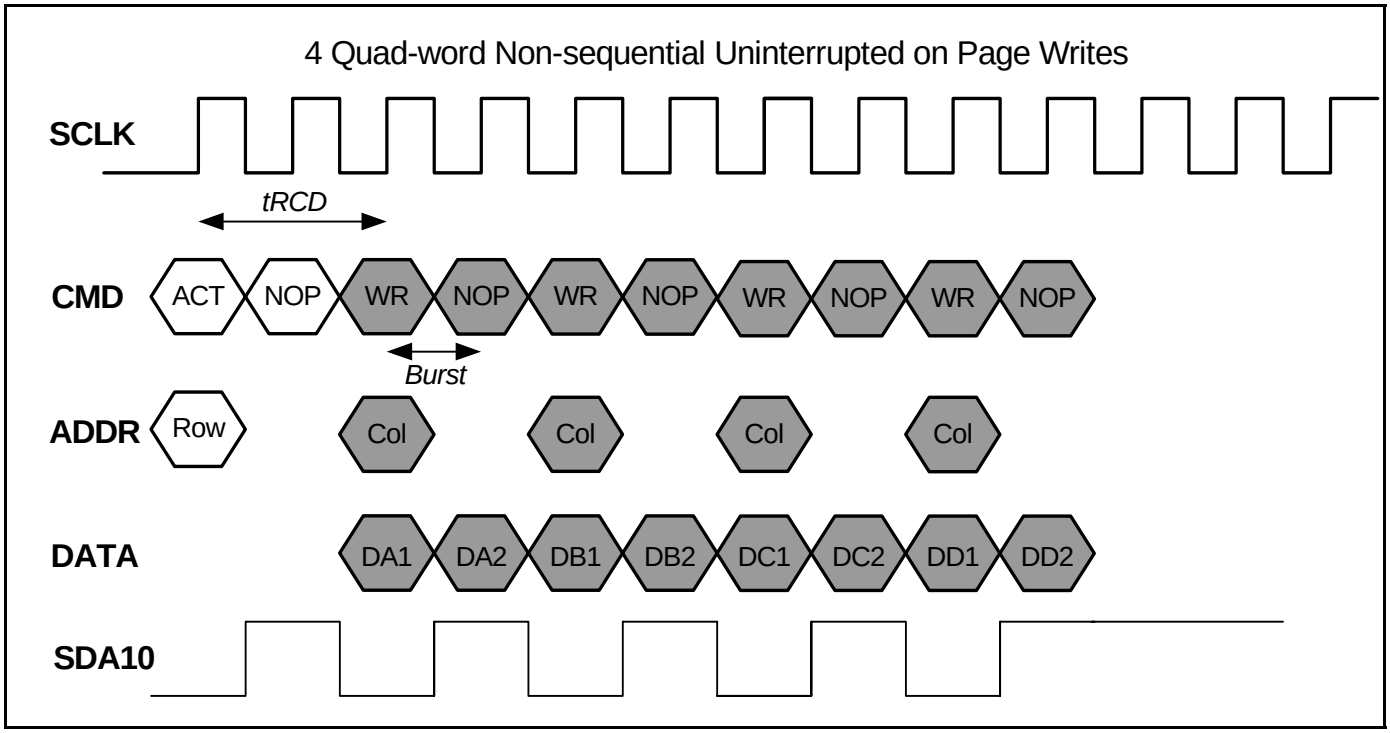
As shown below, uninterrupted sequential writes have a throughput of 1 word per cycle.



Nr. Cycles	Core	Controller	Data
1	**q[j1+=j4]=xr3:0;;**	ACT	
2	int. ACK	NOP	
3	int. ACK	WR	DA1
4	int. ACK	NOP	DA2
5	**q[j1+=j4]=xr7:4;;**	WR	DB1
6	int. ACK	NOP	DB2
7	**q[j1+=j4]=xr11:8;;**	WR	DC1
8	int. ACK	NOP	DC2
9	**q[j1+=j4]=xr15:12;;**	WR	DD1
10	int. ACK	NOP	DD2

14.5 Non Sequential Writes Without Interruption

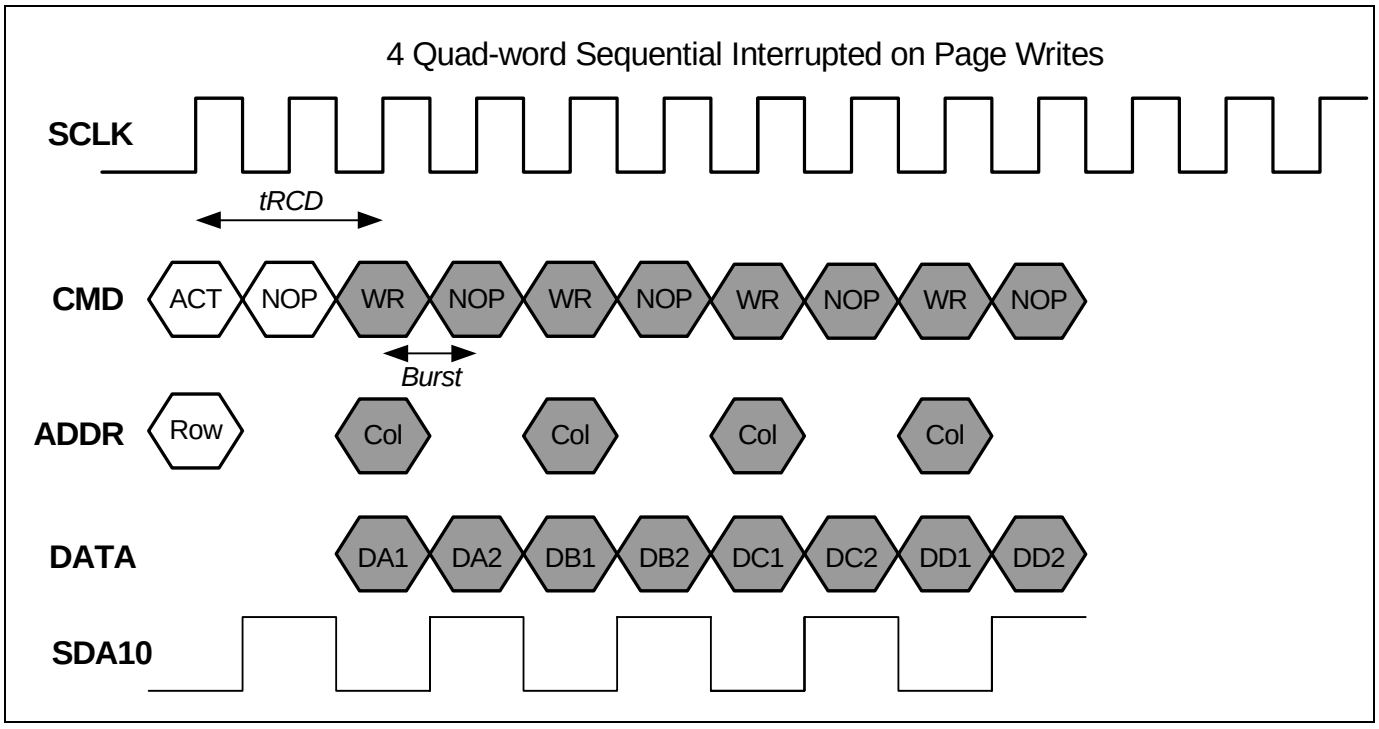
Similar to uninterrupted sequential writes, non-sequential writes also give a 1 word/cycle throughput.



Nr. Cycles	Core	Controller	Data
1	**q[j1+=j4]=xr3:0;;**	ACT	
2	int. ACK	NOP	
3	int. ACK	WR	DA1
4	int. ACK	NOP	DA2
5	**q[j1+=j4]=xr7:4;;**	WR	DB1
6	int. ACK	NOP	DB2
7	**q[j1+=j4]=xr11:8;;**	WR	DC1
8	int. ACK	NOP	DC2
9	**q[j1+=j4]=xr15:12;;**	WR	DD1
10	int. ACK	NOP	DD2

14.6 Sequential Writes with Minimum Interruption

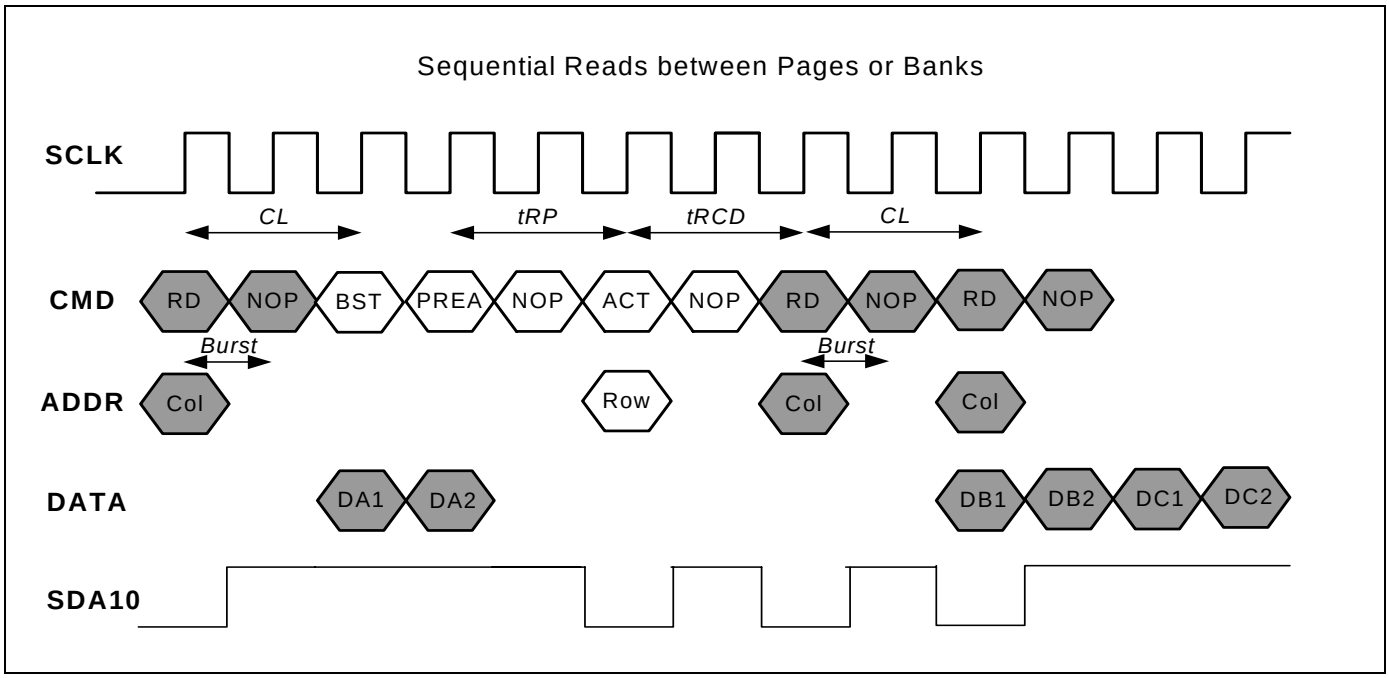
In this case, an additional core access is performed during a core/IOP write to SDRAM. As shown below, the SDRAM access does not get interrupted, and it operates similarly to uninterrupted writes (i.e., 1 word/cycle). Note that a higher number of core accesses during a write to SDRAM may cause a throughput decrease, depending upon system speed and core activity.



Nr. Cycles	Core	Controller	Data
1	**q[j1+=j4]=xr3:0;;**	ACT	
2	int. ACK	NOP	
3	int. ACK	WR	DA1
4	int. ACK	NOP	DA2
5	**q[j1+=j4]=xr7:4;;**	WR	DB1
6	**xr0 = r1 + r2;;**	NOP	DB2
7	**q[j1+=j4]=xr11:8;;**	WR	DC1
8	int. ACK	NOP	DC2
9	**q[j1+=j4]=xr15:12;;**	WR	DD1
10	int. ACK	NOP	DD2

14.7 Reads Between Page/Bank

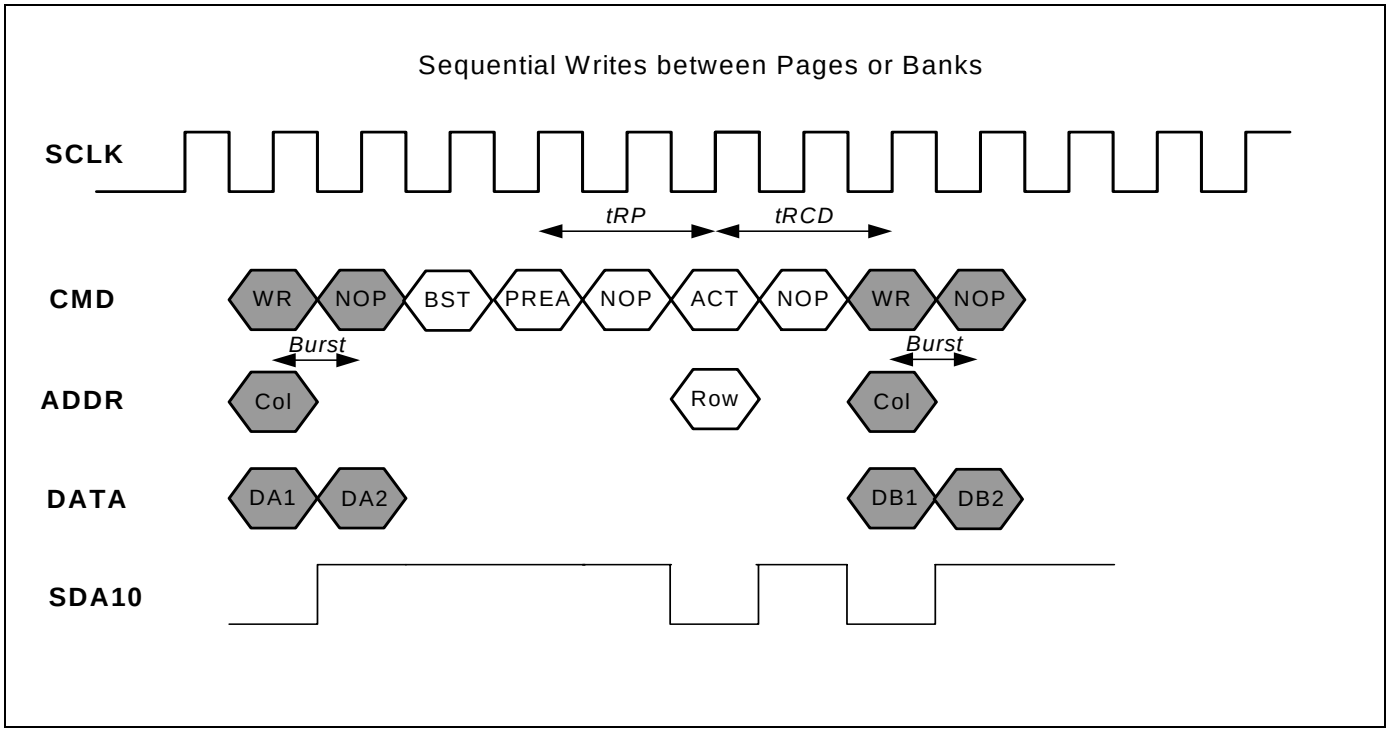
Core/IOP reads from SDRAM interrupted by off-bank/page accesses. Crossing bank/page boundaries impacts performance, giving a 1 word per 6 cycles throughput.



Nr. Cycles	Core	Controller	Data
1	x**r3:0=q[j1+=260];;**	ACT	
2	int. ACK	NOP	
3	int. ACK	RD	
4	int. ACK	NOP	
5	int. ACK	BST	DA1
6	int. ACK	PREA	DA2
7	int. ACK	NOP	
8	**xr7:4=q[j1+=j4];;**	ACT	
9	int. ACK	NOP	
10	int. ACK	RD	
11	int. ACK;	NOP	
12	**xr11:8=q[j1+=j4];;**	RD	DB1
13	int. ACK;	NOP	DB2
14	**xr15:12=q[j1+=j4];;**	RD	DC1
15	int. ACK;	NOP	DC2

14.8 Writes Between Page/Bank

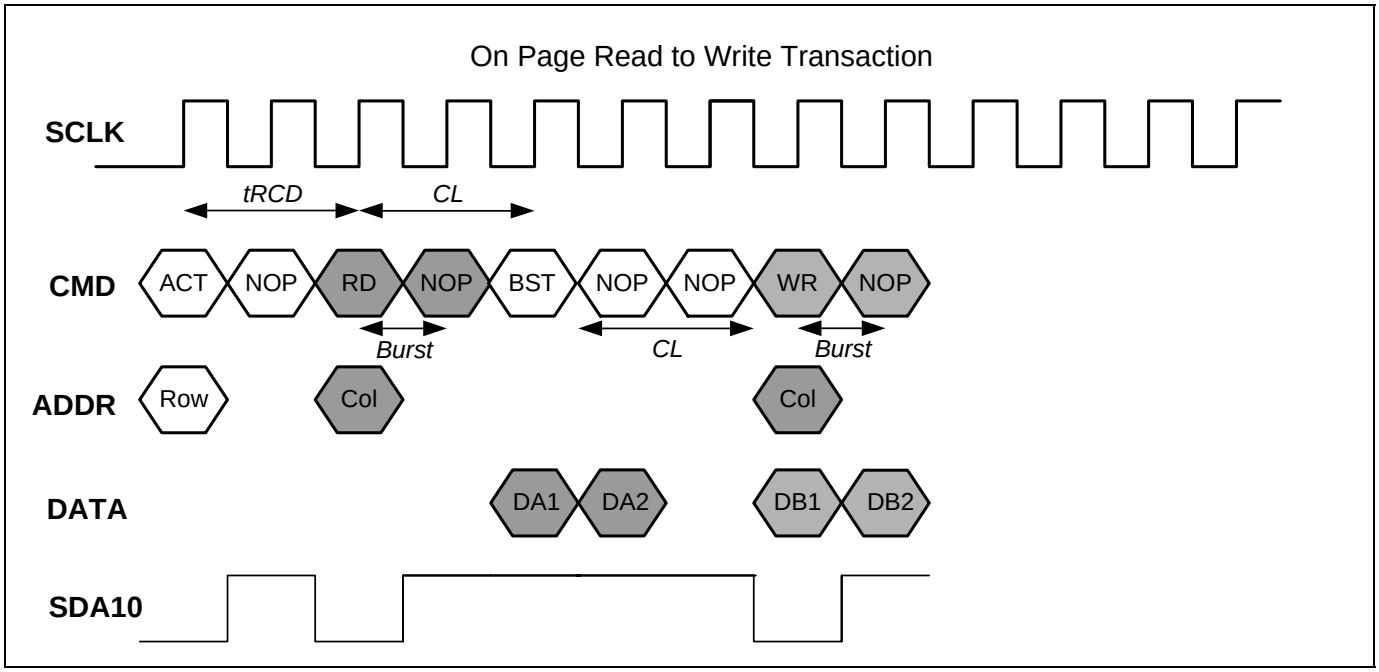
Core/IOP writes to SDRAM interrupted by off-bank/page accesses. Similar to read accesses, although smaller, crossing bank/page boundaries greatly impact performance, giving a 1 word per 6 cycles throughput.



Nr. Cycles	Core	Controller	Data
1	**q[j1+=260]=xr3:0;;**	ACT	
2	int. ACK	NOP	
3	int. ACK	WR	DA1
4	int. ACK	NOP	DA2
5	int. ACK	BST	
6	int. ACK	PREA	
7	int. ACK	NOP	
8	**q[j1+=j4]=xr7:4;;**	ACT	
9	int. ACK	NOP	
10	int. ACK	WR	DB1
11	int. ACK	NOP	DB2

14.9 Minimum Read-to-Write Interval

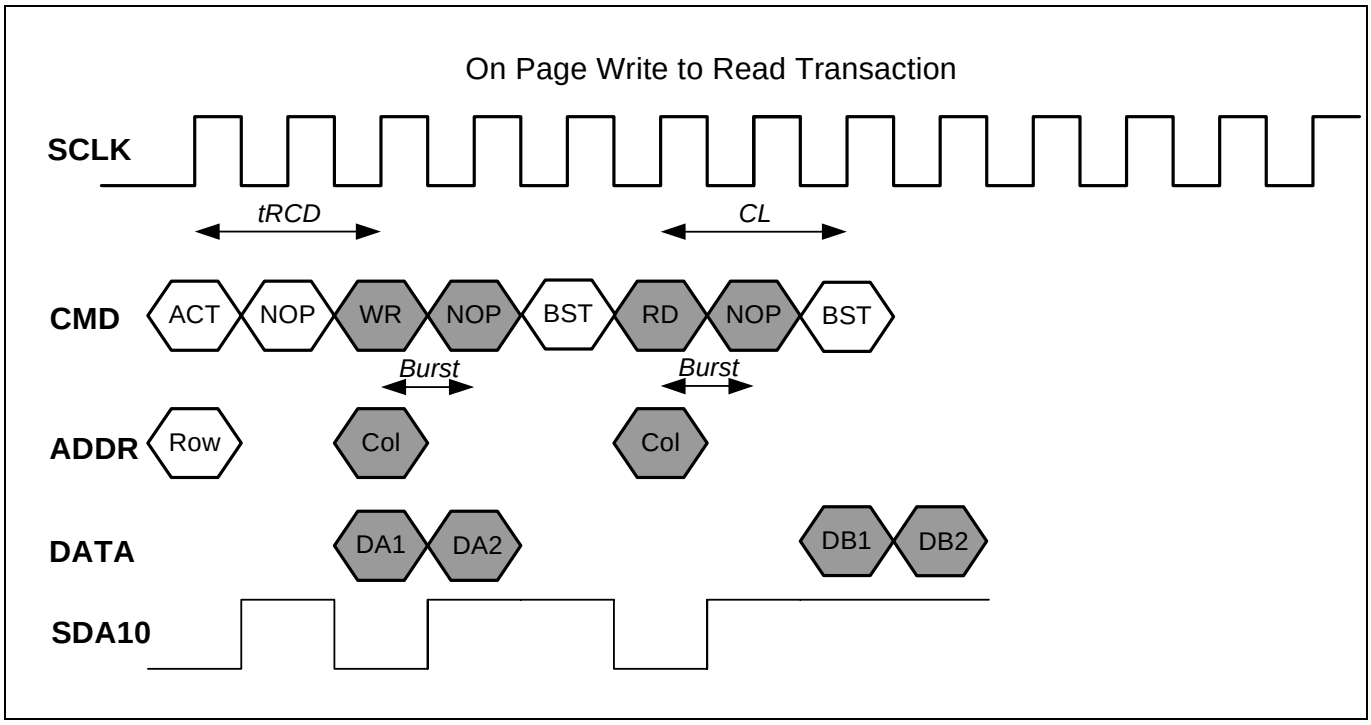
The following diagram shows an SDRAM read-to-write transaction. As shown below, the transition from one access to another introduces overhead cycles (fixed to CL value), resulting performance loss of 1 word per 2 cycles.



Nr. Cycles	Core	Controller	Data
1	**xr3:0=q[j1+=j4];;**	ACT	
2	int. ACK	NOP	
3	int. ACK	RD	
4	int. ACK	NOP	
5	int. ACK	BST	DA1
6	int. ACK	NOP	DA2
7	int. ACK	NOP	
8	**q[j1+=j4]=xr7:4;;**	WR	DB1
9	int. ACK	NOP	DB2

14.10 Minimum Write-to-Read Interval

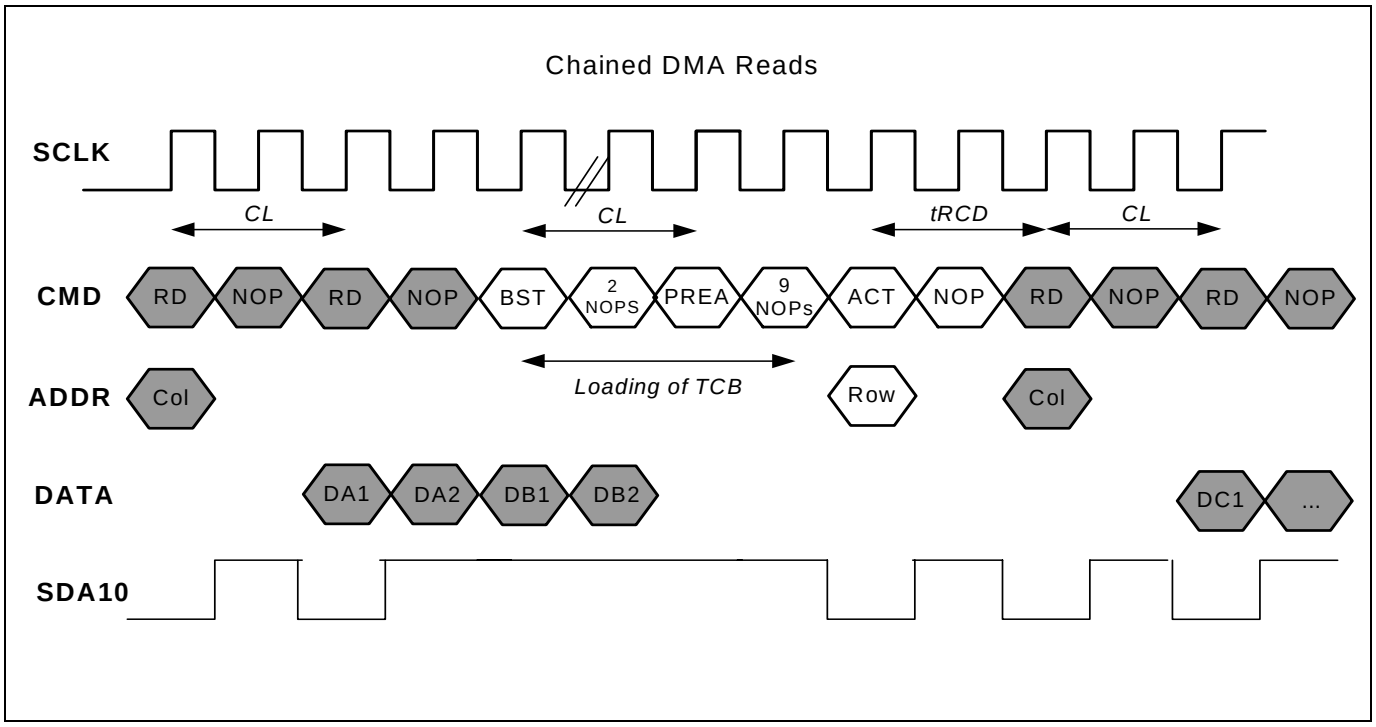
The following diagram shows an SDRAM write-to-read transaction. Simialr to read-to-write transitions, some cycles of overhead are introduced, resulting in a loss of performance of 1 word per 4 cycles.

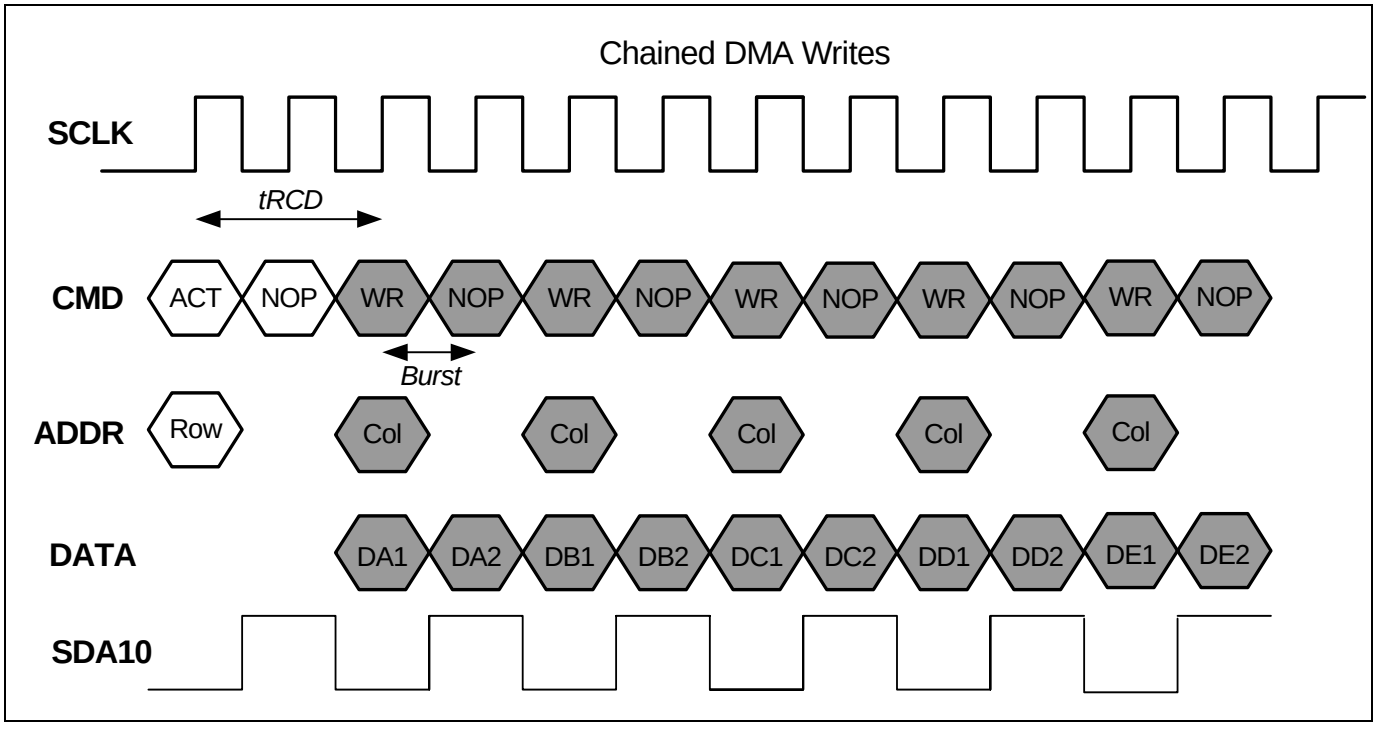


Nr. Cycles	Core	Controller	Data
1	**q[j1+=j4]=xr3:0;;**	ACT	
2	int. ACK	NOP	
3	int. ACK	WR	DA1
4	int. ACK	NOP	DA2
5	int. ACK	BST	
6	**xr7:4=q[j1+=j4];;**	RD	
7	int. ACK	NOP	
8	int. ACK	BST	DB1
9	int. ACK	NOP	DB2

14.11 Chained DMA Transfers

Higher throughput can be achieved by using chained DMAs. However, as shown below, a performance difference can be seen between chained DMA writes and reads to and from SDRAM. Loading the TCB for a chained read DMA requires additional core cycles.



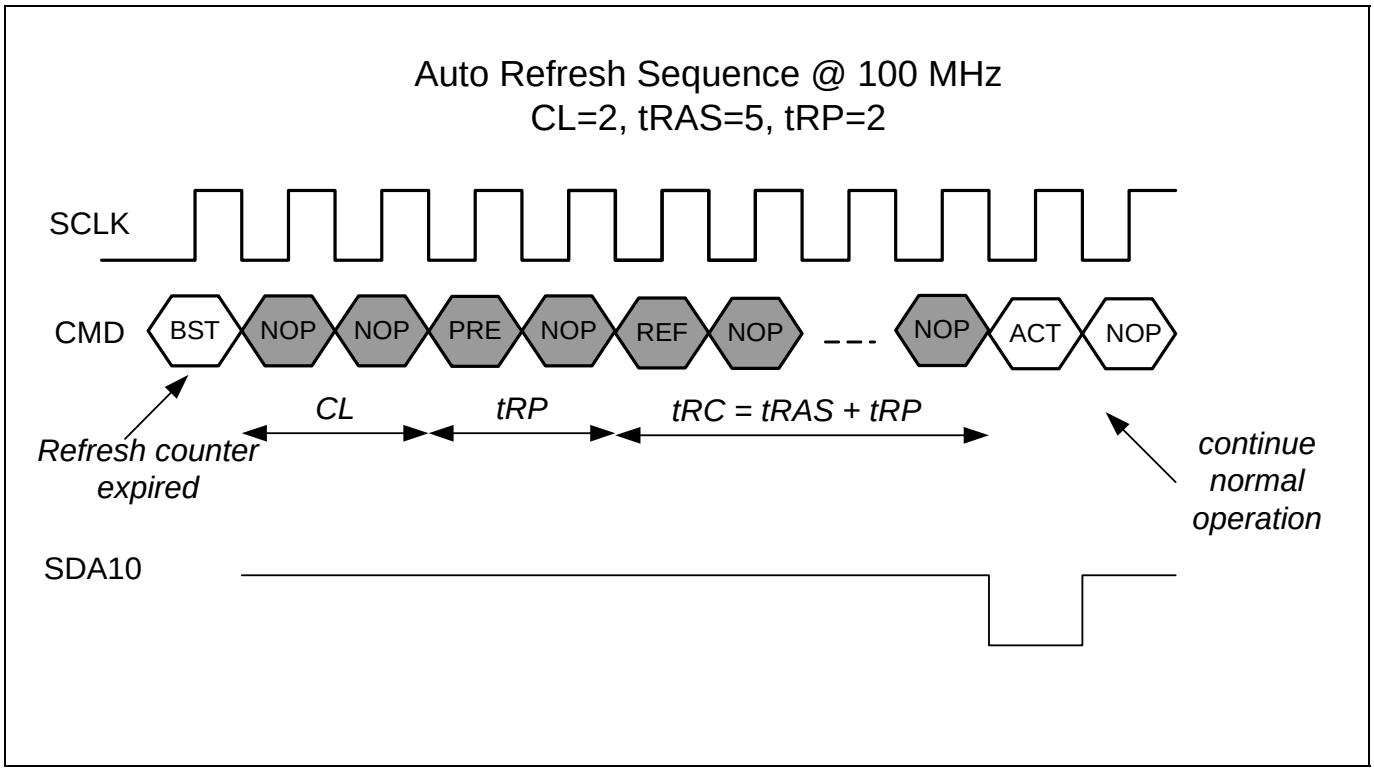


14.12 Auto-Refresh

The refresh counter triggers refresh requests when expired. The following diagram shows the refresh sequence. All banks are precharged (SDA10 is asserted high) right after the internal counter issues the refresh. A REF command starts an internal row refresh with CAS before RAS. During refresh, other commands cannot be executed. The ratio between application time and refresh time is given by:

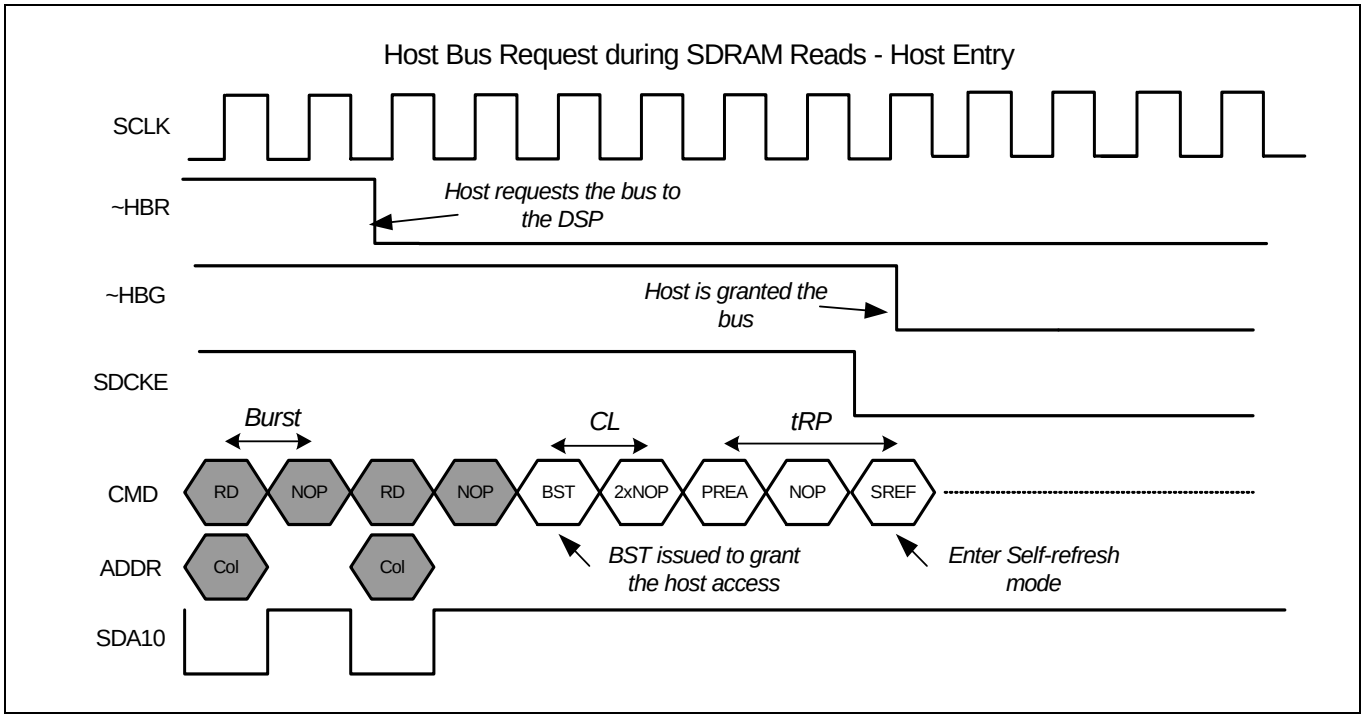
t_{RC} = 15,625 µs/row, (e.g., 16 cycles/3700 cycles x 100 = 0.43%)

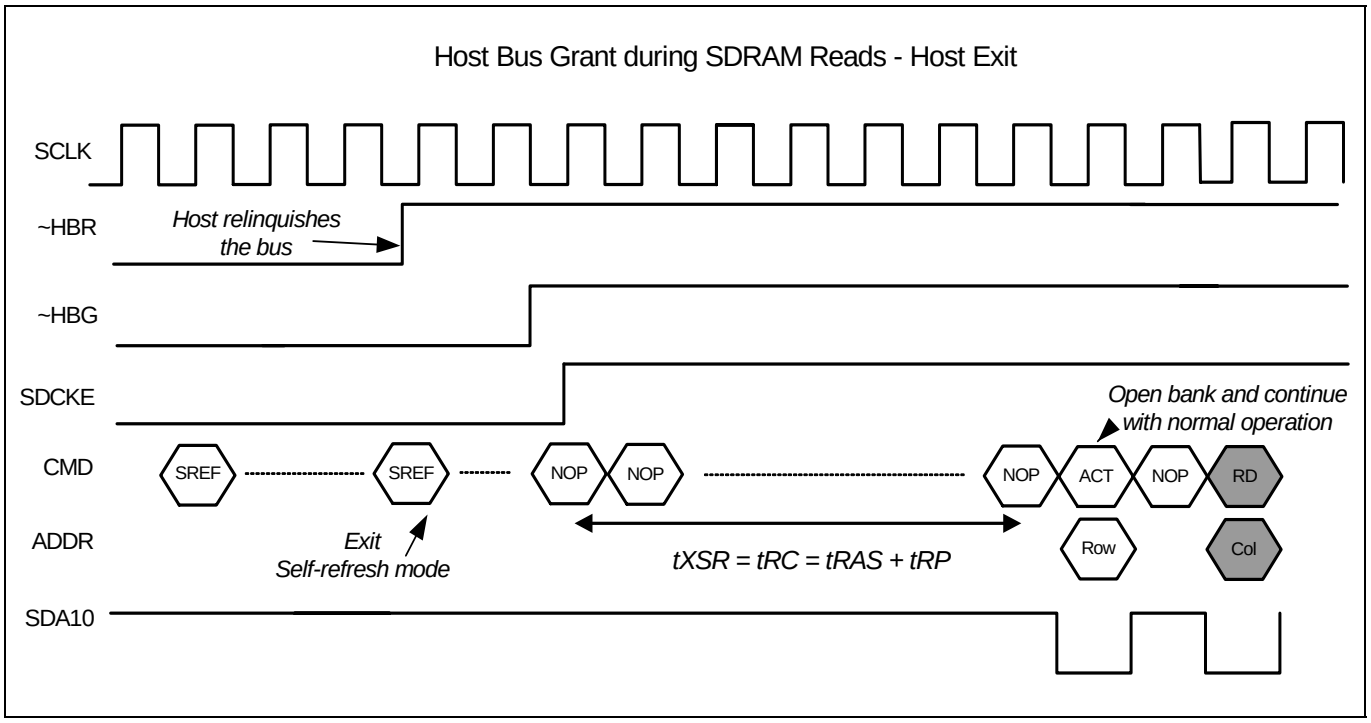
This means that the refresh sequence requires 0.43 % of the whole performance at 100 MHz. Auto-refresh with 15,625 µs/row provideds the best performance ratio. This ratio can be modified accordingly to the system needs by changing the refresh rate programmed in SDRCON.



14.13 Self-Refresh and Host Accesses

The following diagrams show a random host access during a read from SDRAM:





After detecting a host request, the current burst operation of the SDRAM controller is interrupted and frozen. The TigerSHARC processor enters self-refresh mode, in response to host bus request (~HBR),

before the bus is relinquished to the host. As shown above, the controller issues a BST command, stopping the current action. To provide current information, all banks are precharged. During the PREA and SREF commands, the SDCKE pin transitions from high to low, thus enabling self-refresh mode.

After de-assertion of ~HBG, the controller ends the self-refresh function and the SDCKE line transits high. Following, the controller executes a sequence of NOPs (*tXSREF = tRAS + tRP*). The bank is then activated and continues normal operation, reading data from where it left off.

Note that *only the SDCKE pin keeps control of the device in self-refresh mode.*

14.14 SDRAM Performance Table

The following table gives a summary of all the different SDRAM accesses previously explored.

Accesses	Operations	Page	Throughput per SDRAM Clock (64-bit words)[1]
Sequential Uninterrupted	Read	Same	1 word/1 cycle
Non-sequential Uninterrupted	Read	Same	1 word/1 cycle
Sequential Interrupted	Read	Same	1 word/ 1 cycle
Sequential Uninterrupted	Write	Same	1 word/1 cycle
Non-sequential Uninterrupted	Write	Same	1 word/1 cycle
Sequential Interrupted	Write	Same	1 word/1 cycle
Both	Read to Write	Same	1 word/2 cycles (CL)
Both	Write to Read	Same	1 word/4 cycles (2 + CL)
Nonsequential	Reads	Different	1 word/6 cycles (t_{RP} + t_{RCD} +CL)
Nonsequential	Writes	Different	1 word/5 cycles (1+ t_{RP} + t_{RCD})
Auto-refresh before read	Reads	Different	1 word/15 cycle ($2t_{RP}$ + t_{RAS} + t_{RCD} +2CL)
Auto-refresh before write	Writes	Different	1 word/12 cycle (1+ $2t_{RP}$ + t_{RAS} + t_{RCD})

CAS latency (CL) = 2 cycles, Precharge (t_{RP}) = 2 cycles, t_{RAS} = 5 cycles, t_{RCD} = CL = 2 cycles

[1] SYSCON External bus width configuration: 64-bit bus.

15 Optimizing SDRAM Performance

There are several ways to optimize SDRAM performance, depending on the system set-up and application. Some of these are explained below:

15.1 External Buffering

In parallel connections, one address and control bus feeds all the system devices. To meet the timing requirements and help against capacitive load, the TigerSHARC processor's controller provides a feature to cope with this situation.

The controller can pipeline the SDRAM accesses by making use of the internal delay buffer. In SDRCON, setting the pipe depth bit to 1 inserts this delay buffer, allowing address and command pipelining. Set this bit if the capacitive load of 30 pF/pin is exceeded. The data sheet's timing is based on a nominal load of 30 pF/pin.

Figure 10 shows a system set-up in which an external buffer is used in conjunction with the delay buffer provided by the TigerSHARC processor's controller. The external buffer reduces capacitive load and power dissipation but increases the pipeline effects.

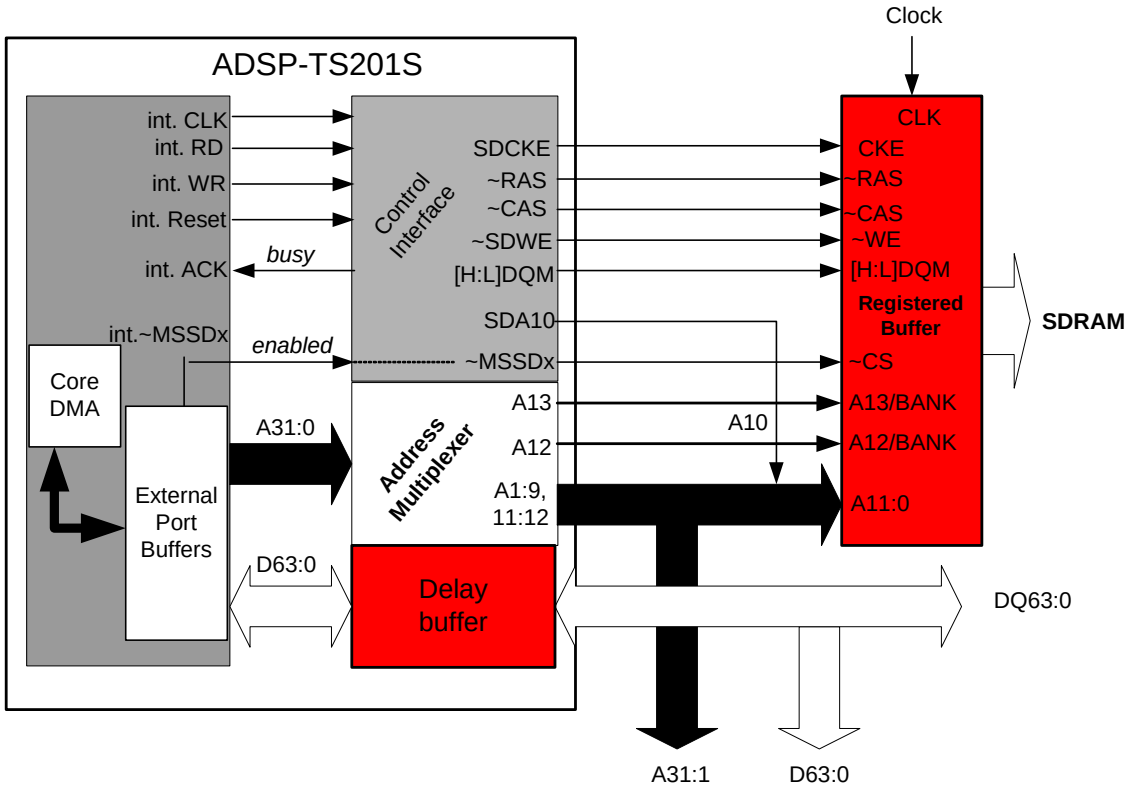


Figure 10. Signal Chain: ADSP-TS201S Processor to SDRAM Using External Buffer

15.2 Using PC Modules

The maximum addressable size is 16M x 32bits per bank (~MSSD[3:0]). The use of unbuffered PC DIMM modules (typical I/O sizes x32, x64, or x72 bits) is efficient, because most vendors offer a size of

x64 bits, which is commonly used for PCs. Moreover, depending on the size, the need for external address and control buffers (pipelining) is required.

15.3 General Rules for Optimized Performance

Depending on the SDRAM's page size and number of banks, position your data segments to minimize off-page and off-bank accesses.

Use parallel connection for SDRAMs with a large page size to obtain 32-bit or 64-bit I/O (SDRAM: the bigger the page size, the smaller the I/O structure).

Use the optimized settings for the controller's state machine (*tRAS*, *tRP*, *CL*), depending on the speed grade and on the application's speed.

16 ADSP-TS20x TigerSHARC Processor Family Derivatives

The differences between the ADSP-TS20xS processors are highlighted in Table 13.

Feature	ADSP-TS201	ADSP-TS202	ADSP-TS203
Max. Core Clock	500 / 600 MHz	500 MHz	500 MHz
On-chip Memory	24 Mbits Internal DRAM	12 Mbits Internal DRAM	4 Mbits Internal DRAM
Communications Logic Unit (CLU)	YES	NO	NO
Link Ports	4 Link Ports Total throughput of 4 Gbyte/sec	4 Link Ports Total throughput of 4 Gbyte/sec	2 Link Ports Total throughput of 1 Gbyte/sec
External Port	64/32-bits Total throughput of 1 GByte/sec	64/32-bits Total throughput of 1 GByte/sec	32-bits ONLY Total throughput of 0.5 GByte/sec

Table 13. ADSP-TS20x TigerSHARC Processor Family Product Differences

The SDRAM controller's functionality and characteristics are identical for all three derivatives, with the only exception of the ADSP-TS203S processor. This processor's external port is restricted to 32-bits, which leads to a fixed external port configuration of 32-bits ONLY. This implies a throughput degradation, decreasing from 1 GByte/sec to 0.5 GByte/sec, as well as a different pin out (/HDQM is no longer valid for the ADSP-TS203S TigerSHARC processor).

Refer to the particular data sheet for the pin-out of each part in the ADSP-TS20xS TigerSHARC processor family.

17 References

[1] *ADSP-TS201S TigerSHARC Processor Hardware Reference Manual.* Revision 0.2, September 2003. Analog Devices, Inc.

[2] *ADSP-TS201S TigerSHARC Embedded Processor Preliminary Data Sheet.* Rev PrH, January 2004. Analog Devices, Inc.

[3] *ADSP-TS202S TigerSHARC Embedded Processor Preliminary Data Sheet.* Rev PrB, January 2004. Analog Devices, Inc.

[4] *ADSP-TS203S TigerSHARC Embedded Processor Preliminary Data Sheet.* Rev PrB, January 2004. Analog Devices, Inc.

[5] *The ABC of SDRAMemory (EE-126).* Rev 1, March 2002. Analog Devices, Inc.

[6] *ADSP-TS201S TigerSHARC Processor Boot Loader Kernels Operation (EE-200).* Rev 1, February 2004. Analog Devices, Inc.

[7] *Considerations for Porting Code from the ADSP-TS101 TigerSHARC Processor to the ADSP-TS201S TigerSHARC Processor (EE-205).* September 2003, Analog Devices, Inc.

[8] *The ADSP-TS101S TigerSHARC On-chip SDRAM Controller (EE-178).* Rev 2, December, 2003. Analog Devices, Inc.

[9] *ADSP-21161N SHARC On-chip SDRAM Controller (EE-163).* Rev 2, September, 2003. Analog Devices, Inc.

[10] *The ADSP-21065L On-chip SDRAM Controller (EE-127).* February, 2002. Analog Devices, Inc.

18 Document History

Revision	Description
Rev 1 – February 4, 2004 *by Maikel Kokaly-Bannourah*	Initial Release