# Chapter 11
# Synchronous/Asynchronous DRAM Controller Module

This chapter describes configuration and operation of the synchronous/asynchronous DRAM controller component of the system integration module (SIM). It begins with a general description and brief glossary, and includes a description of signals involved in DRAM operations. The remainder of the chapter consists of the two following parts:

- Section 11.3, “Asynchronous Operation,” describes the programming model and signal timing for the four basic asynchronous modes.
  - — Non-page mode
  - — Burst page mode
  - — Continuous page mode
  - — Extended data-out mode
- Section 11.4, “Synchronous Operation,” describes the programming model and signal timing, as well as the command set required for synchronous operations. This section also includes extensive examples the designer can follow to better understand how to configure the DRAM controller for synchronous operations.

## 11.1 Overview

The DRAM controller module provides glueless integration of DRAM with the ColdFire product. The key features of the DRAM controller include the following:

- Support for two independent blocks of DRAM
- Interface to standard synchronous/asynchronous dynamic random access memory (ADRAM/SDRAM) components
- Programmable $\overline{\text{SRAS}}$, $\overline{\text{SCAS}}$, and refresh timing
- Support for page mode
- Support for 8-, 16-, and 32-bit wide DRAM blocks
- Support for synchronous and asynchronous DRAMs, including EDO DRAM, SDRAM, and fast page mode

## 11.1.1 Definitions

The following terminology is used in this chapter:

- A/SDRAM block—Any group of DRAM memories selected by one of the MCF5307 $\overline{\text{RAS}}$[1:0] signals. Thus, the MCF5307 can support two independent memory blocks. The base address of each block is programmed in the DRAM address and control registers (DACR0 and DACR1).
- SDRAM—RAMs that operate like asynchronous DRAMs but with a synchronous clock, a pipelined, multiple-bank architecture, and faster speed.
- SDRAM bank—An internal partition in an SDRAM device. For example, a 64-Mbit SDRAM component might be configured as four 512K x 32 banks. Banks are selected through the SDRAM component's bank select lines.

## 11.1.2 Block Diagram and Major Components

The basic components of the DRAM controller are shown in Figure 11-1.

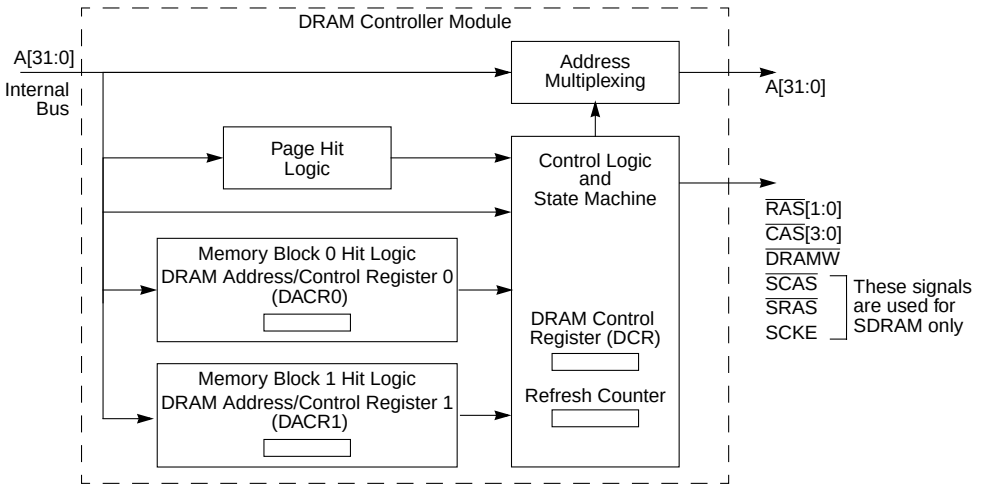



**Figure 11-1. Asynchronous/Synchronous DRAM Controller Block Diagram**

The DRAM controller's major components, shown in Figure 11-1, are described as follows:

- DRAM address and control registers (DACR0 and DACR1)—The DRAM controller consists of two configuration register units, one for each supported memory block. DACR0 is accessed at MBAR + 0x0108; DACR1 is accessed at 0x010. The register information is passed on to the hit logic.

- Control logic and state machine—Generates all DRAM signals, taking bus cycle characteristic data from the block logic, along with hit information to generate DRAM accesses. Handles refresh requests from the refresh counter.
  - — DRAM control register (DCR)—Contains data to control refresh operation of the DRAM controller. Both memory blocks are refreshed concurrently as controlled by DCR[RC].
  - — Refresh counter—Determines when refresh should occur, determined by the value of DCR[RC]. It generates a refresh request to the control block.
- Hit logic—Compares address and attribute signals of a current DRAM bus cycle to both DACRs to determine if a DRAM block is being accessed. Hits are passed to the control logic along with characteristics of the bus cycle to be generated.
- Page hit logic—Determines if the next DRAM access is in the same DRAM page as the previous one. This information is passed on to the control logic.
- Address multiplexing—Multiplexes addresses to allow column and row addresses to share pins. This allows glueless interface to DRAMs.

## 11.2 DRAM Controller Operation

The DRAM controller mode is programmed through DCR[SO]. Asynchronous mode (SO = 0) includes support for page mode and EDO DRAMs. Synchronous mode is designed to work with industry-standard SDRAMs. These modes act very differently from one another, especially regarding the use of DRAM registers and pins. Memory blocks cannot operate in different modes; both are either synchronous or asynchronous.

### 11.2.1 DRAM Controller Registers

The DRAM controller registers memory map, Table 11-1, is the same regardless of whether asynchronous or synchronous DRAM is used, although bit configurations may vary.

**Table 11-1. DRAM Controller Registers**

| MBAR Offset | [31:24] | [23:16] | [15:8] | [7:0] |
|---|---|---|---|---|
| 0x100 | DRAM control register (DCR) [p. 11-4] | | Reserved | |
| 0x104 | Reserved | | | |
| 0x108 | DRAM address and control register 0 (DACR0) [p. 11-5] | | | |
| 0x10C | DRAM mask register block 0 (DMR0) [p. 11-7] | | | |
| 0x110 | DRAM address and control register 1 (DACR1) [p. 11-5] | | | |
| 0x114 | DRAM mask register block 1 (DMR1) [p. 11-7] | | | |

**NOTE:**

External masters cannot access MCF5307 on-chip memories or MBAR, but they can access DRAM controller registers.

# 11.3 Asynchronous Operation

The DRAM controller supports asynchronous DRAMs for cost-effective systems. Typical access times for the DRAM controller interfacing to ADRAM are 4-3-3-3. The DRAM controller supports the following four asynchronous modes:

- Non-page mode
- Burst page mode
- Continuous page mode
- Extended data-out mode

In asynchronous mode, $\overline{RAS}$ and $\overline{CAS}$ always transition at the falling clock edge. As summarized previously, operation and timing of each ADRAM block is controlled by separate registers, but refresh is the same for both. All ADRAM accesses should be terminated by the DRAM controller. There is no priority encoding between memory blocks, so programming blocks to overlap with other blocks or with other internal resources causes undefined behavior.

## 11.3.1 DRAM Controller Signals in Asynchronous Mode

Table 11-2 summarizes DRAM signals used in asynchronous mode.

**Table 11-2. SDRAM Signal Summary**

| Signal | Description |
|---|---|
| $\overline{RAS}$[1:0] | Row address strobes. Interface to $\overline{RAS}$ inputs on industry-standard ADRAMs. When SDRAMs are used, these signals interface to the chip-select lines within an SDRAM's memory block. Thus, there is one $\overline{RAS}$ line for each of the two blocks. |
| $\overline{CAS}$[3:0] | Column address strobes. Interface to $\overline{CAS}$ inputs on industry-standard DRAMs. These provide $\overline{CAS}$ for a given ADRAM block. When SDRAMs are used, $\overline{CAS}$[3:0] control the byte enables (DQMx) for standard SDRAMs. $\overline{CAS}$[3:0] strobes data in least-to-most significant byte order ($\overline{CAS0}$ is MSB). |
| $\overline{DRAMW}$ | DRAM read/write. Asserted when a DRAM write cycle is underway. Negated for read bus cycles. |

## 11.3.2 Asynchronous Register Set

The following register configurations apply when DCR[SO] is 0, indicating the DRAM controller is interfacing to asynchronous DRAMs.

### 11.3.2.1 DRAM Control Register (DCR) in Asynchronous Mode

The DCR provides programmable options for the refresh logic as well as the control bit to determine if the module is operating with synchronous or asynchronous DRAMs. The DCR is shown in Figure 11-2.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Field | SO | — | NAM | RRA | | RRP | | RC | |
| Reset | 0 | Uninitialized | | | | | | | |
| R/W | R/W | | | | | | | | |
| Address | MBAR + 0x100 | | | | | | | | |

**Figure 11-2. DRAM Control Register (DCR) (Asynchronous Mode)**

Table 11-3 describes DCR fields.

**Table 11-3. DCR Field Descriptions (Asynchronous Mode)**

| Bits | Name | Description |
|---|---|---|
| 15 | SO | Synchronous operation. Selects synchronous or asynchronous mode. A DRAM controller in synchronous mode can be switched to ADRAM mode only by resetting the MCF5307.<br>0 Asynchronous DRAMs. Default at reset.<br>1 Synchronous DRAMs |
| 14 | — | Reserved, should be cleared. |
| 13 | NAM | No address multiplexing. Some implementations require external multiplexing. For example, when linear addressing is required, the DRAM should not multiplex addresses on DRAM accesses.<br>0 The DRAM controller multiplexes the external address bus to provide column addresses.<br>1 The DRAM controller does not multiplex the external address bus to provide column addresses. |
| 12–11 | RRA | Refresh $\overline{RAS}$ asserted. Determines how long $\overline{RAS}$ is asserted during a refresh operation.<br>00 2 clocks<br>01 3 clocks<br>10 4 clocks<br>11 5 clocks |
| 10–9 | RRP | Refresh $\overline{RAS}$ precharge. Controls how many clocks $\overline{RAS}$ is precharged after a refresh operation before accesses are allowed to DRAM.<br>00 1 clock<br>01 2 clocks<br>10 3 clocks<br>11 4 clocks |
| 8–0 | RC | Refresh count. Controls refresh frequency. The number of bus clocks between refresh cycles is (RC + 1) * 16. Refresh can range from 16–8192 bus clocks to accommodate both standard and low-power DRAMs with bus clock operation from less than 2 MHz to greater than 50 MHz.<br>The following example calculates RC for an auto-refresh period for 4096 rows to receive 64 mS of refresh every 15.625 µs for each row (625 bus clocks at 40 MHz).<br># of bus clocks = 625 = (RC field + 1) * 16<br>RC = (625 bus clocks/16) -1 = 38.06, which rounds to 38; therefore, RC = 0x26. |

### 11.3.2.2 DRAM Address and Control Registers (DACR0/DACR1)

DACR0 and DACR1, Figure 11-3, contain the base address compare value and the control bits for memory blocks 0 and 1. Address and timing are also controlled by these registers. Memory areas defined for each block should not overlap; operation is undefined for accesses in overlapping regions.

| | 31 18 | 17 16 | 15 | 14 | 13 12 | 11 10 | 9 | 8 | 7 | 6 | 5 4 | 3 2 | 1 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | BA | — | RE | — | CAS | RP | RNCN | RCD | — | EDO | PS | PM | — |
| Reset | Unitialized | | 0 | Unitialized | | | | | | | | | |
| R/W | R/W | | | | | | | | | | | | |
| Addr | MBAR + 0x10C (DACR0); 0x110 (DACR1) | | | | | | | | | | | | |

**Figure 11-3. DRAM Address and Control Registers (DACR0/DACR1)**

Table 11-4 describes DACR*n* fields.

**Table 11-4. DACR0/DACR1 Field Description**

| Bits | Name | Description |
|---|---|---|
| 31–18 | BA | Base address. Used with DMR[BAM] to determine the address range in which the associated DRAM block is located. Each BA bit is compared with the corresponding address of the bus cycle in progress. If each bit matches, or if bits that do not match are masked in the BAM, the address selects the associated DRAM block. |
| 17–16 | — | Reserved, should be cleared. |
| 15 | RE | Refresh enable. Determines whether the DRAM controller generates a refresh to the associated DRAM block. DRAM contents are not preserved during hard reset or software watchdog reset.<br>0 Do not refresh associated DRAM block. (Default at reset)<br>1 Refresh associated DRAM block. |
| 14 | — | Reserved, should be cleared. |
| 13–12 | CAS | $\overline{CAS}$ timing. Determines how long $\overline{CAS}$ is asserted during a DRAM access.<br>00 1 clock cycle<br>01 2 clock cycles<br>10 3 clock cycles<br>11 4 clock cycles |
| 11–10 | RP | $\overline{RAS}$ precharge timing. Determines how long $\overline{RAS}$ is precharged between accesses. Note that RP is different from DCR[RRP].<br>00 1 clock cycle<br>01 2 clock cycles<br>10 3 clock cycles<br>11 4 clock cycles |
| 9 | RNCN | $\overline{RAS}$-negate-to-$\overline{CAS}$-negate. Controls whether $\overline{RAS}$ and $\overline{CAS}$ negate concurrently or one clock apart. RNCN is ignored if $\overline{CAS}$ is asserted for only one clock and both $\overline{RAS}$ and $\overline{CAS}$ are negated. RNCN is used only for non-page-mode accesses and single accesses in page mode.<br>0 $\overline{RAS}$ negates concurrently with $\overline{CAS}$.<br>1 $\overline{RAS}$ negates one clock before $\overline{CAS}$. |
| 8 | RCD | $\overline{RAS}$-to-$\overline{CAS}$ delay. Determines the number of system clocks between assertions of $\overline{RAS}$ and $\overline{CAS}$.<br>0 1 clock cycle<br>1 2 clock cycles |
| 7 | — | Reserved, should be cleared. |
| 6 | EDO | Extended data out. Determines whether the DRAM block operates in a mode to take advantage of industry-standard EDO DRAMs. Do not use EDO mode with non-EDO DRAM.<br>0 EDO operation disabled.<br>1 EDO operation enabled. |

**Table 11-4. DACR0/DACR1 Field Description (Continued)**

| Bits | Name | Description |
|---|---|---|
| 5–4 | PS | Port size. Determines the port size of the associated DRAM block. For example, if two 16-bit wide DRAM components form one DRAM block, the port size is 32 bits. Programming PS allows the DRAM controller to execute dynamic bus sizing for associated accesses.<br>00 32-bit port<br>01 8-bit port<br>1x 16-bit port |
| 3–2 | PM | Page mode. Configures page-mode operation for the memory block.<br>00 No page mode<br>01 Burst page mode (page mode for bursts only)<br>10 Reserved<br>11 Continuous page mode |
| 1–0 | — | Reserved, should be cleared. |

### 11.3.2.3 DRAM Controller Mask Registers (DMR0/DMR1)

The DRAM controller mask registers (DMR0 and DMR1), shown in Figure 11-4, include mask bits for the base address and for address attributes.

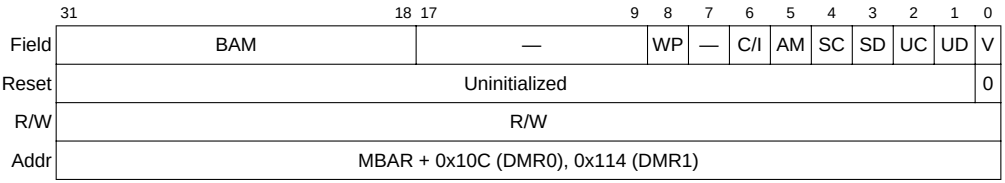



**Figure 11-4. DRAM Controller Mask Registers (DMR0 and DMR1)**

Table 11-5 describes DMR*n* fields.

**Table 11-5. DMR0/DMR1 Field Descriptions**

| Bits | Name | Description |
|---|---|---|
| 31–18 | BAM | Base address mask. Masks the associated DACRn[BA]. Lets the DRAM controller connect to various DRAM sizes. Mask bits need not be contiguous (see Section 11.5, “SDRAM Example.”)<br>0 The associated address bit is used in decoding the DRAM hit to a memory block.<br>1 The associated address bit is not used in the DRAM hit decode. |
| 17–9 | — | Reserved, should be cleared. |
| 8 | WP | Write protect. Determines whether the associated block of DRAM is write protected.<br>0 Allow write accesses<br>1 Ignore write accesses. The DRAM controller ignores write accesses to the memory block and an address exception occurs. Write accesses to a write-protected DRAM region are compared in the chip select module for a hit. If no hit occurs, an external bus cycle is generated. If this external bus cycle is not acknowledged, an access exception occurs. |
| 7 | — | Reserved, should be cleared. |

**Table 11-5. DMR0/DMR1 Field Descriptions (Continued)**

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th></tr>
<tr><td>6–1</td><td>AMx</td><td>Address modifier masks. Determine which accesses can occur in a given DRAM block.<br>0 Allow access type to hit in DRAM<br>1 Do not allow access type to hit in DRAM
<table>
<tr><th>Bit</th><th>Associated Access Type</th><th>Access Definition</th></tr>
<tr><td>C/I</td><td>CPU space/interrupt acknowledge</td><td>MOVEC instruction or interrupt acknowledge cycle</td></tr>
<tr><td>AM</td><td>Alternate master</td><td>External or DMA master</td></tr>
<tr><td>SC</td><td>Supervisor code</td><td>Any supervisor-only instruction access</td></tr>
<tr><td>SD</td><td>Supervisor data</td><td>Any data fetched during the instruction access</td></tr>
<tr><td>UC</td><td>User code</td><td>Any user instruction</td></tr>
<tr><td>UD</td><td>User data</td><td>Any user data</td></tr>
</table></td></tr>
<tr><td>0</td><td>V</td><td>Valid. Cleared at reset to ensure that the DRAM block is not erroneously decoded.<br>0 Do not decode DRAM accesses.<br>1 Registers controlling the DRAM block are initialized; DRAM accesses can be decoded.</td></tr>
</table>

## 11.3.3 General Asynchronous Operation Guidelines

The DRAM controller provides control for $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, and $\overline{\text{DRAMW}}$ signals, as well as address multiplexing and bus cycle termination. Whether the mode is synchronous or asynchronous determines signal control and termination. To reduce complexity, multiplexing is the same for both modes. Table 11-6 shows the scheme for DRAM configurations. This scheme works for symmetric configurations (in which the number of rows equals the number of columns) as well as asymmetric configurations (in which the number of rows and columns are different).

**Table 11-6. Generic Address Multiplexing Scheme**

| Address Pin | Row Address | Column Address | Notes Relating to Port Sizes |
|---|---|---|---|
| 17 | 17 | 0 | 8-bit port only |
| 16 | 16 | 1 | 8- and 16-bit ports only |
| 15 | 15 | 2 | |
| 14 | 14 | 3 | |
| 13 | 13 | 4 | |
| 12 | 12 | 5 | |
| 11 | 11 | 6 | |
| 10 | 10 | 7 | |
| 9 | 9 | 8 | |
| 17 | 17 | 16 | 32-bit port only |
| 18 | 18 | 17 | 16-bit port only or 32-bit port with only 8 column address lines |
| 19 | 19 | 18 | 16-bit port only when at least 9 column address lines are used |

**Table 11-6. Generic Address Multiplexing Scheme (Continued)**

| Address Pin | Row Address | Column Address | Notes Relating to Port Sizes |
|---|---|---|---|
| 20 | 20 | 19 | |
| 21 | 21 | 20 | |
| 22 | 22 | 21 | |
| 23 | 23 | 22 | |
| 24 | 24 | 23 | |
| 25 | 25 | 24 | |

Note the following:

- Each MCF5307 address bit drives both a row address and a column address bit.
- As the user upgrades ADRAM, corresponding MCF5307 address bits must be connected. This multiplexing scheme allows various memory widths to be connected to the address bus.
- Some differences exist for each of the three possible port sizes. Note that only 8-bit ports use an A0 address from the MCF5307. Because 16- and 32-bit ports issue either words or longwords when accessed, they do not use the MCF5307 A0 signal.

  Likewise, the configuration for 32-bit ports uses neither A0 or A1. This presents a slight problem because DRAM address signal A0 is issued on physical pin A17 of the MCF5307 along with the ADRAM address signal A17. Although A0 is not used for larger ports, A17 is still needed. The MCF5307 DRAM controller provides for this by changing the column address that appears on physical pin A17 of the processor whenever an 8-bit port is not selected. This is determined by the DACR*n*[PS] settings. For 8-bit ports, MCF5307 physical pin A17 drives logical address A0 during the CAS cycle. When 16- or 32-bit port sizes are programmed, the CAS cycle pin A17 drives logical address A16, as indicated in the generic connection scheme.
- If a 32-bit port is used with only eight column address lines, A18 must drive DRAM address bit A18. Otherwise, in 32-bit port configurations, the MCF5307 physical address line is not connected with more than eight column address lines.
- All ADRAM blocks have a fixed page size of 512 bytes for page-mode operation. The addresses are connected differently for various width combinations.

Table 11-7, Table 11-8, and Table 11-9 show how 8-, 16-, and 32-bit symmetrical ADRAM memories are connected to the address bus. The memory sizes show what DRAM size is accessed if the corresponding bits are connected to the memory. In each case, there is a base memory size. This limitation exists to allow simple page-mode multiplexing. Notice also that MCF5307 pin 17 is treated differently in byte-wide operations. In byte-wide operations, address bits 16 and 17 are driven on MCF5307 physical address pins 16 and 17, rather than the two bits being driven solely on A17, as they are for 32-wide memories.

**Table 11-7. DRAM Addressing for Byte-Wide Memories**

| MCF5307 Address Pin | MCF5307 Address Bit Driven for RAS | MCF5307 Address Bit Driven when CAS is Asserted | Memory Size |
|---|---|---|---|
| 17 | 17 | 0 | Base memory size of 256 Kbytes |
| 16 | 16 | 1 | |
| 15 | 15 | 2 | |
| 14 | 14 | 3 | |
| 13 | 13 | 4 | |
| 12 | 12 | 5 | |
| 11 | 11 | 6 | |
| 10 | 10 | 7 | |
| 9 | 9 | 8 | |
| 19 | 19 | 18 | 1 Mbyte |
| 21 | 21 | 20 | 4 Mbytes |
| 23 | 23 | 22 | 16 Mbytes |
| 25 | 25 | 24 | 64 Mbytes |

Note that in Table 11-8, MCF5307 pin A19 is not connected because DRAM address bit 18 is already provided on MCF5307 pin A18; thus, the next MCF5307 pin used should be A20.

**Table 11-8. DRAM Addressing for 16-Bit Wide Memories**

| MCF5307 Address Pin | MCF5307 Address Bit Driven for RAS | MCF5307 Address Bit Driven when CAS is Asserted | Memory Size |
|---|---|---|---|
| 16 | 16 | 1 | Base memory size of 128 Kbytes |
| 15 | 15 | 2 | |
| 14 | 14 | 3 | |
| 13 | 13 | 4 | |
| 12 | 12 | 5 | |
| 11 | 11 | 6 | |
| 10 | 10 | 7 | |
| 9 | 9 | 8 | |
| 18 | 18 | 17 | 512 Kbytes |
| 20 | 20 | 19 | 2 Mbytes |
| 22 | 22 | 21 | 8 Mbytes |
| 24 | 24 | 23 | 32 Mbytes |

**Table 11-9. DRAM Addressing for 32-Bit Wide Memories**

| MCF5307 Address Pin | MCF5307 Address Bit Driven for $\overline{RAS}$ | MCF5307 Address Bit Driven when CAS is Asserted | Memory Size |
|---|---|---|---|
| 15 | 15 | 2 | Base Memory Size of 64 Kbytes |
| 14 | 14 | 3 | |
| 13 | 13 | 4 | |
| 12 | 12 | 5 | |
| 11 | 11 | 6 | |
| 10 | 10 | 7 | |
| 9 | 9 | 8 | |
| 17 | 17 | 16 | 256 Kbytes |
| 19 | 19 | 18 | 1 Mbyte |
| 21 | 21 | 20 | 4 Mbytes |
| 23 | 23 | 22 | 16 Mbytes |
| 25 | 25 | 24 | 64 Mbytes |

### 11.3.3.1 Non-Page-Mode Operation

In non-page mode, the simplest mode, the DRAM controller provides termination and runs a separate bus cycle for each data transfer. Figure 11-5 shows a non-page-mode access in which a DRAM read is followed by a write. Addresses for a new bus cycle are driven at the rising clock edge.

For a DRAM block hit, the associated $\overline{RAS}$ is driven at the next falling edge. Here DACR*n*[RCD] = 0, so the address is multiplexed at the next rising edge to provide the column address. The required $\overline{CAS}$ signals are then driven at the next falling edge and remain asserted for the period programmed in DACR*n*[CAS]. Here, DACR*n*[RNCN] = 1, so it is precharged one clock before $\overline{CAS}$ is negated. On a read, data is sampled on the last rising edge of the clock that $\overline{CAS}$ is valid.

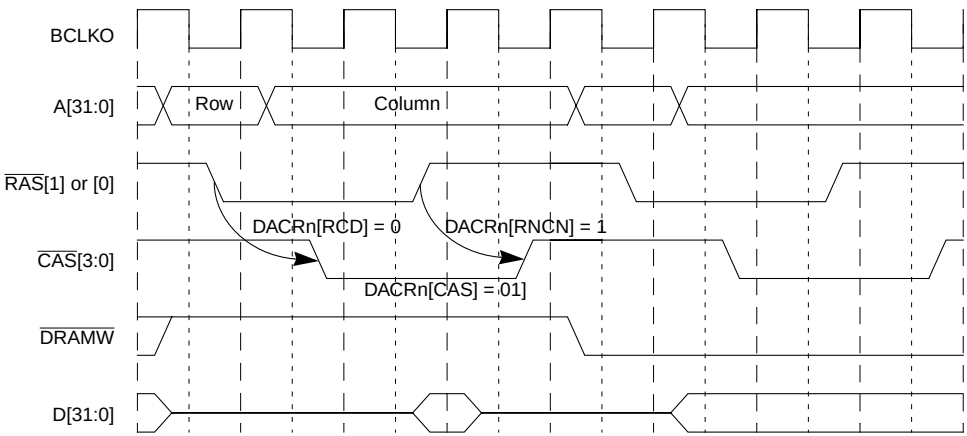



**Figure 11-5. Basic Non-Page-Mode Operation RCD = 0, RNCN = 1 (4-4-4-4)**

Figure 11-6 shows a variation of the basic cycle. In this case, RCD is 1, so there are two clocks between $\overline{RAS}$ and $\overline{CAS}$. Note that the address is multiplexed on the rising clock immediately before $\overline{CAS}$ is asserted. Because RNCN = 0, $\overline{RAS}$ and $\overline{CAS}$ are negated together. The next bus cycle is initiated, but because DACR*n*[RP] requires $\overline{RAS}$ to be precharged for two clocks, $\overline{RAS}$ is delayed for a clock in the bus cycle. Note that this does not delay the address signals, only $\overline{RAS}$.

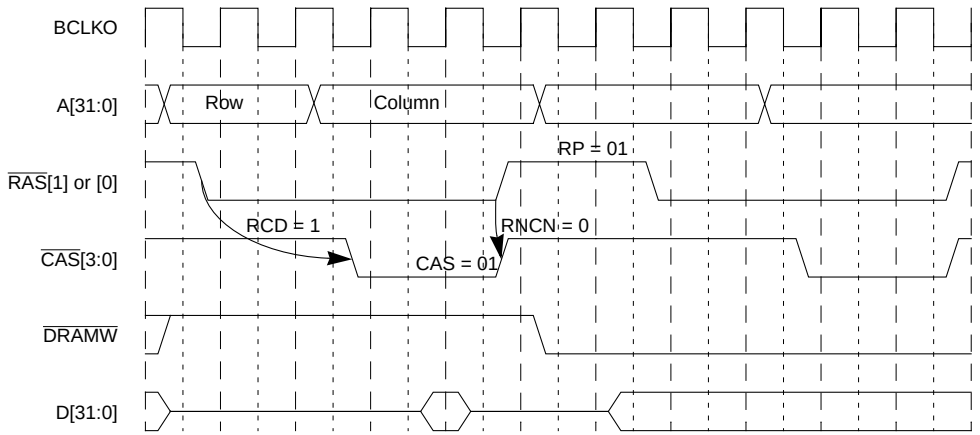



**Figure 11-6. Basic Non-Page-Mode Operation RCD = 1, RNCN = 0 (5-5-5-5)**

### 11.3.3.2 Burst Page-Mode Operation

Burst page-mode operation (DACRn[PM] = 01) optimizes memory accesses in page mode by allowing a row address to remain registered in the DRAM while accessing data in different columns. This eliminates the setup and hold times associated with the need to precharge and assert RAS. Therefore, only the first bus cycle in the page takes the full access time; subsequent accesses are streamlined. Single accesses look the same as non-page-mode accesses.

Burst page-mode accesses of any size—byte, word, longword, or line—are assumed to reside in the same page. In this mode, the DRAM controller generates a burst transfer only when the operand is larger than the DRAM block port size (such as, a line transfer to a 32-bit port or a longword transfer to an 8-bit port). The primary cycle asserts $\overline{RAS}$ and $\overline{CAS}$; subsequent cycles assert only $\overline{CAS}$. At the end of the access, $\overline{RAS}$ is precharged. The DRAM controller increments addresses between cycles.

Figure 11-7 shows a read access in burst page mode. Four accesses take place, which could be a 32-bit access to an 8-bit port or a line access to a 32-bit port. Other burst page-mode operations may be from 2 to 16 accesses long, depending on the access and port sizes. In those cases, timing is similar with more or fewer accesses.

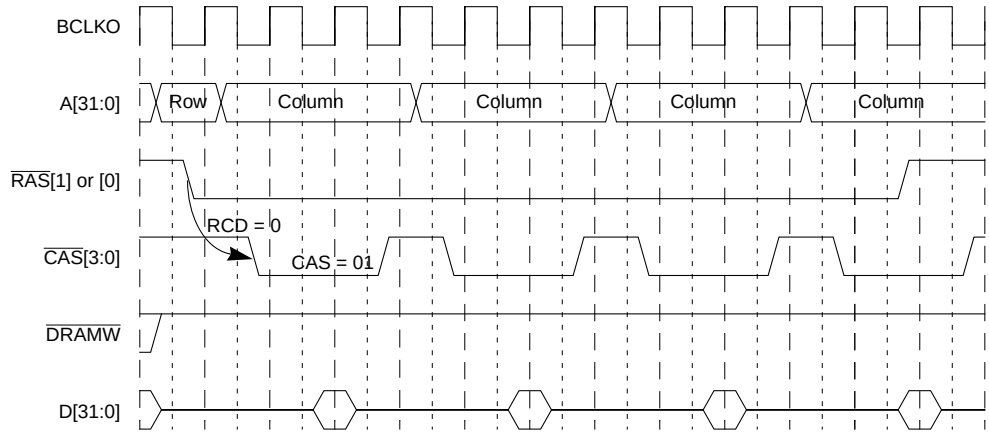



**Figure 11-7. Burst Page-Mode Read Operation (4-3-3-3)**

Figure 11-8 shows the write operation with the same configuration.

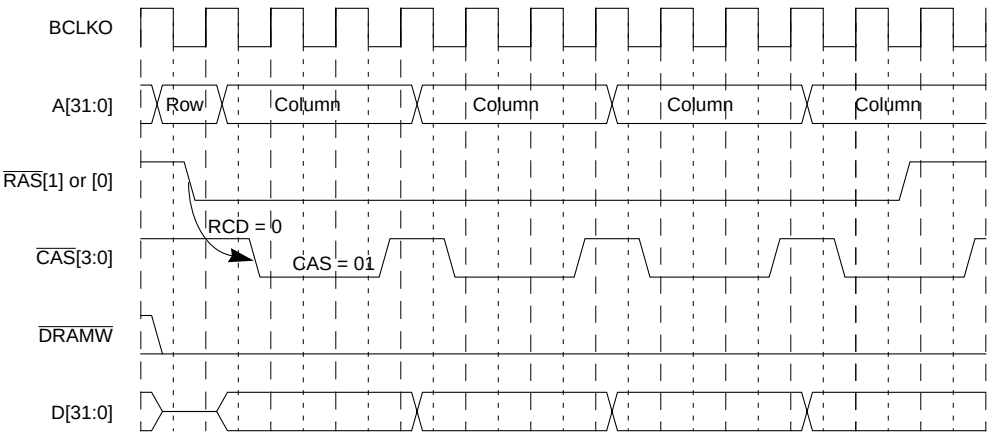



**Figure 11-8. Burst Page-Mode Write Operation (4-3-3-3)**

### 11.3.3.3 Continuous Page Mode

Continuous page mode (DACRn[PM] = 11) is a type of page mode that balances performance, complexity, and size. In typical page-mode implementations, sequential addresses are checked for multiple hits in a DRAM block. On a hit, $\overline{\text{RAS}}$ remains asserted and $\overline{\text{CAS}}$ is asserted with the new column address. On a miss, $\overline{\text{RAS}}$ must be precharged again before the bus cycle begins.

Continuous page mode supports page-mode operation without requiring an address holding register per memory block and eliminates the delay for a miss-to-precharge $\overline{\text{RAS}}$ for the upcoming bus cycle. Because the internal MCF5307 address bus is pipelined, addresses for

the next bus cycle are often available before the current cycle completes. The two addresses are compared at the end of the cycle to determine if the next address hits the same page. If so, $\overline{RAS}$ remains asserted. If not, or if no access is pending, $\overline{RAS}$ is precharged before the next bus cycle is active on the external bus. As a result, a page miss suffers no penalty. Single accesses not followed by a hit in the page look like non-page-mode accesses.

Figure 11-9 shows a write cycle followed by a read cycle in continuous page mode. The read hits in the same page as the write so $\overline{RAS}$ is not negated before the second cycle. Note that the row address does not appear on the pins for a bus cycle that hits in the page. Column addresses are immediately multiplexed onto the pins. The third bus cycle is a page miss, so $\overline{RAS}$ is precharged before the end of the bus cycle and no extra precharge delay is incurred.

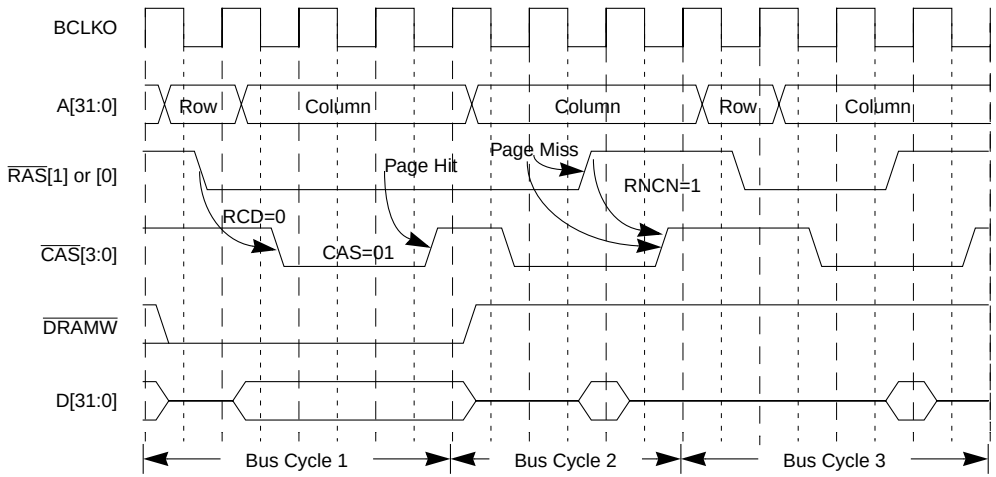



**Figure 11-9. Continuous Page-Mode Operation**

If a write cycle hits in the page, $\overline{CAS}$ must be delayed by one clock to allow data to become valid, as shown in Figure 11-10.

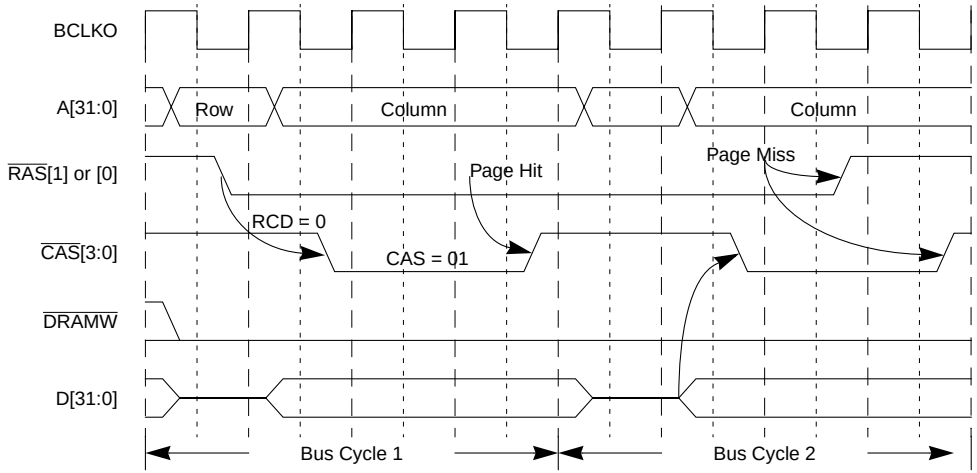



**Figure 11-10. Write Hit in Continuous Page Mode**

### 11.3.3.4 Extended Data Out (EDO) Operation

EDO is a variation of page mode that allows the DRAM to continue driving data out of the device while $\overline{CAS}$ is precharging. To support EDO DRAMs, the DRAM controller delays internal termination of the cycle by one clock so data can continue to be captured as $\overline{CAS}$ is being precharged. For data to be driven by the DRAMs, $\overline{RAS}$ is held after $\overline{CAS}$ is negated. EDO operation does not affect write operations. EDO DRAMs can be used in continuous page or burst page modes. Single accesses not followed by a hit in the page look like non-page-mode accesses.

Figure 11-11 shows four consecutive EDO accesses. Note that data is sampled after $\overline{CAS}$ is negated and that on the last page access, $\overline{CAS}$ is held until after data is sampled to assure that the data is driven. This allows $\overline{RAS}$ to be precharged before the end of the cycle.

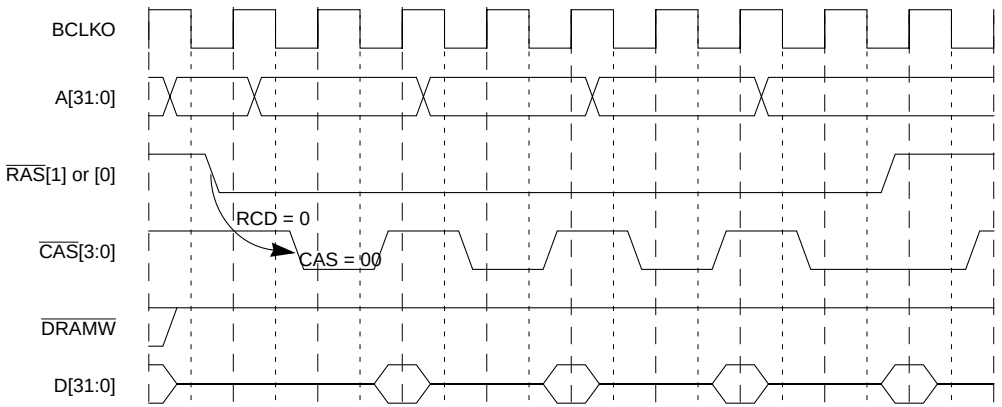



**Figure 11-11. EDO Read Operation (3-2-2-2)**

### 11.3.3.5 Refresh Operation

The DRAM controller supports $\overline{\text{CAS}}$-before-$\overline{\text{RAS}}$ refresh operations that are not synchronized to bus activity. A special $\overline{\text{DRAMW}}$ pin is provided so refresh can occur regardless of the state of the processor bus.

When the refresh counter rolls over, it sets an internal flag to indicate that a refresh is pending. If that happens during a continuous page-mode access, the page is closed ($\overline{\text{RAS}}$ precharged) when the data transfer completes to allow the refresh to occur. The flag is cleared when the refresh cycle is run. Both memory blocks are simultaneously refreshed as determined by the DCR. DRAM accesses are delayed during refresh. Only an active bus access to a DRAM block can delay refresh.

Figure 11-12 shows a bus cycle delayed by a refresh operation. Notice that $\overline{\text{DRAMW}}$ is forced high during refresh. The row address is held until the pending DRAM access.

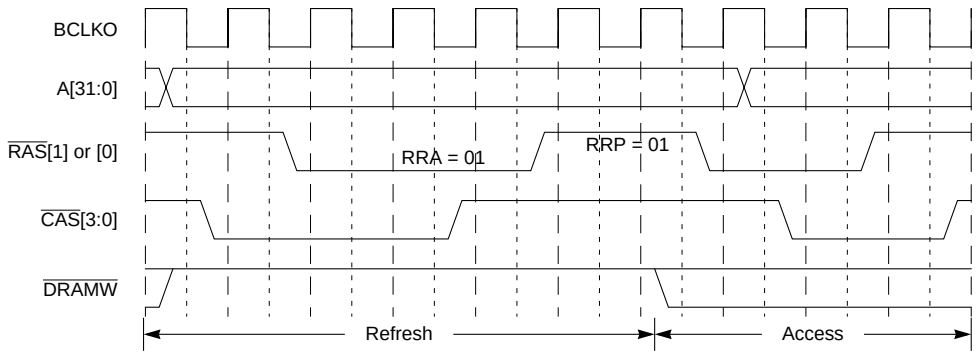



**Figure 11-12. DRAM Access Delayed by Refresh**

## 11.4 Synchronous Operation

By running synchronously with the system clock instead of responding to asynchronous control signals, SDRAM can (after an initial latency period) be accessed on every clock; 5-1-1-1 is a typical MCF5307 burst rate to SDRAM.

Note that because the MCF5307 cannot have more than one page open at a time, it does not support interleaving.

SDRAM controllers are more sophisticated than asynchronous DRAM controllers. Not only must they manage addresses and data, but they must send special commands for such functions as precharge, read, write, burst, auto-refresh, and various combinations of these functions. Table 11-10 lists common SDRAM commands.

**Table 11-10. SDRAM Commands**

| Command | Definition |
|---|---|
| ACTV | Activate. Executed before READ or WRITE executes; SDRAM registers and decodes row address. |
| MRS | Mode register set. |
| NOP | No-op. Does not affect SDRAM state machine; DRAM controller control signals negated; $\overline{\text{RAS}}$ asserted. |
| PALL | Precharge all. Precharges all internal banks of an SDRAM component; executed before new page is opened. |
| READ | Read access. SDRAM registers column address and decodes that a read access is occurring. |
| REF | Refresh. Refreshes internal bank rows of an SDRAM component. |
| SELF | Self refresh. Refreshes internal bank rows of an SDRAM component when it is in low-power mode. |
| SELFX | Exit self refresh. This command is sent to the DRAM controller when DCR[IS] is cleared. |
| WRITE | Write access. SDRAM registers column address and decodes that a write access is occurring. |

SDRAMs operate differently than asynchronous DRAMs, particularly in the use of data pipelines and commands to initiate special actions. Commands are issued to memory using specific encodings on address and control pins. Soon after system reset, a command must be sent to the SDRAM mode register to configure SDRAM operating parameters. Note that, after synchronous operation is selected by setting DCR[SO], DRAM controller registers reflect the synchronous operation and there is no way to return to asynchronous operation without resetting the processor.

## 11.4.1 DRAM Controller Signals in Synchronous Mode

Table 11-11 shows the behavior of DRAM signals in synchronous mode.

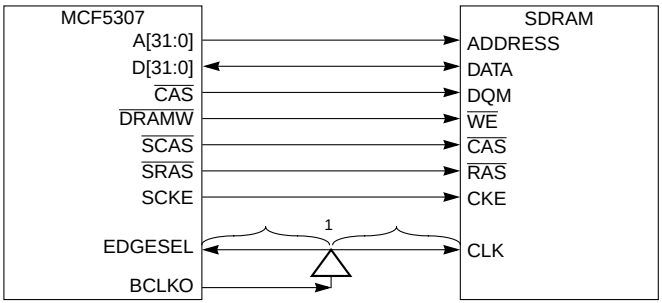
**Table 11-11. Synchronous DRAM Signal Connections**

| Signal | Description |
|---|---|
| $\overline{\text{SRAS}}$ | Synchronous row address strobe. Indicates a valid SDRAM row address is present and can be latched by the SDRAM. $\overline{\text{SRAS}}$ should be connected to the corresponding SDRAM $\overline{\text{SRAS}}$. Do not confuse $\overline{\text{SRAS}}$ with the DRAM controller's $\overline{\text{RAS}}$[1:0], which should not be interfaced to the SDRAM $\overline{\text{SRAS}}$ signals. |
| $\overline{\text{SCAS}}$ | Synchronous column address strobe. Indicates a valid column address is present and can be latched by the SDRAM. $\overline{\text{SCAS}}$ should be connected to the corresponding signal labeled $\overline{\text{SCAS}}$ on the SDRAM. Do not confuse $\overline{\text{SCAS}}$ with the DRAM controller's $\overline{\text{CAS}}$[3:0] signals. |
| $\overline{\text{DRAMW}}$ | DRAM read/write. Asserted for write operations and negated for read operations. |
| $\overline{\text{RAS}}$[1:0] | Row address strobe. Select each memory block of SDRAMs connected to the MCF5307. One $\overline{\text{RAS}}$ signal selects one SDRAM block and connects to the corresponding $\overline{\text{CS}}$ signals. |
| SCKE | Synchronous DRAM clock enable. Connected directly to the CKE (clock enable) signal of SDRAMs. Enables and disables the clock internal to SDRAM. When CKE is low, memory can enter a power-down mode where operations are suspended or they can enter self-refresh mode. SCKE functionality is controlled by DCR[COC]. For designs using external multiplexing, setting COC allows SCKE to provide command-bit functionality. |
| $\overline{\text{CAS}}$[3:0] | Column address strobe. For synchronous operation, $\overline{\text{CAS}}$[3:0] function as byte enables to the SDRAMs. They connect to the DQM signals (or mask qualifiers) of the SDRAMs. |

**Table 11-11. Synchronous DRAM Signal Connections (Continued)**

| Signal | Description |
|---|---|
| BCLKO | Bus clock output. Connects to the CLK input of SDRAMs. |
| EDGESEL | Synchronous edge select. Provides additional output hold time for signals that interface to external SDRAMs. EDGESEL supports the three following modes for SDRAM interface signals:<br>• Tied high. Signals change on the rising edge of BCLKO.<br>• Tied low. Signals change on the falling edge of BCLKO.<br>• Tied to buffered BCLKO. Signals change on the rising edge of the buffered clock.<br>EDGESEL can provide additional output hold time for SDRAM interface signals, however the SDRAM clock and BCLKO frequencies must be the same. See Section 11.4.2, “Using Edge Select (EDGESEL).” |

Figure 11-13 shows a typical signal configuration for synchronous mode.




[1] Trace length from buffer to CLK must equal length from buffer to EDGESEL.

**Figure 11-13. MCF5307 SDRAM Interface**

## 11.4.2 Using Edge Select (EDGESEL)

EDGESEL can ease system-level timings (note that the optional buffer in Figure 11-13 is for memories that need extra delay). The clock at the input to the SDRAM is monitored and data is held until the next edge of the bus clock, adding required output hold time to the address, data, and control signals.

To generate SDRAM interface timing, address, data, and control signals are clocked through a two-stage shift register. The first stage is clocked on the rising edge of BCLKO; the second is clocked on the falling edge. This makes the signal available for up to an additional half bus clock cycle, of which only a small amount is needed for proper timing.

Using the connection shown in Figure 11-13 ensures that data remains held for a longer time after the rising edge of the SDRAM clock input. This helps to match the MCF5307 output timing with the SDRAM clock.

Figure 11-14 shows the output wave forms for the interface signals changing on the rising edge (A) and falling edge (B) of BCLKO as determined by whether EDGESEL is tied high or low. It also shows timing (C) with EDGESEL tied to buffered BCLKO.

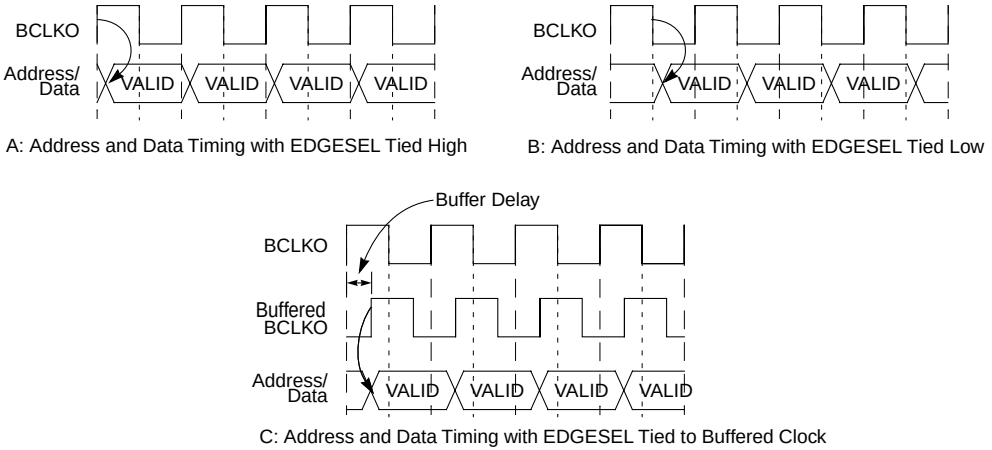



**Figure 11-14. Using EDGESEL to Change Signal Timing**

## 11.4.3 Synchronous Register Set

The memory map in Table 11-1 is the same for both synchronous and asynchronous operation. However, some bits are different, as noted in the following sections.

### 11.4.3.1 DRAM Control Register (DCR) in Synchronous Mode

The DRAM control register (DCR), Figure 11-15, controls refresh logic.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | SO | — | NAM | COC | IS | RTIM | | RC | | | | | | | | |
| Reset | 0 | Uninitialized | | | | | | | | | | | | | | |
| R/W | R/W | | | | | | | | | | | | | | | |
| Addr | MBAR + 0x100 | | | | | | | | | | | | | | | |

**Figure 11-15. DRAM Control Register (DCR) (Synchronous Mode)**

Table 11-12 describes DCR fields.

**Table 11-12. DCR Field Descriptions (Synchronous Mode)**

| Bits | Name | Description |
|---|---|---|
| 15 | SO | Synchronous operation. Selects synchronous or asynchronous mode. When in synchronous mode, the DRAM controller can be switched to ADRAM mode only by resetting the MCF5307.<br>0 Asynchronous DRAMs. Default at reset.<br>1 Synchronous DRAMs |
| 14 | — | Reserved, should be cleared. |
| 13 | NAM | No address multiplexing. Some implementations require external multiplexing. For example, when linear addressing is required, the DRAM should not multiplex addresses on DRAM accesses.<br>0 The DRAM controller multiplexes the external address bus to provide column addresses.<br>1 The DRAM controller does not multiplex the external address bus to provide column addresses. |

**Table 11-12. DCR Field Descriptions (Synchronous Mode) (Continued)**

| Bits | Name | Description |
|---|---|---|
| 12 | COC | Command on SDRAM clock enable (SCKE). Implementations that use external multiplexing (NAM = 1) must support command information to be multiplexed onto the SDRAM address bus.<br>0 SCKE functions as a clock enable; self-refresh is initiated by the DRAM controller through DCR[IS].<br>1 SCKE drives command information. Because SCKE is not a clock enable, self-refresh cannot be used (setting DCR[IS]). Thus, external logic must be used if this functionality is desired. External multiplexing is also responsible for putting the command information on the proper address bit. |
| 11 | IS | Initiate self-refresh command.<br>0 Take no action or issue a SELFX command to exit self refresh.<br>1 If DCR[COC] = 0, the DRAM controller sends a SELF command to both SDRAM blocks to put them in low-power, self-refresh state where they remain until IS is cleared, at which point the controller sends a SELFX command for the SDRAMs to exit self-refresh. The refresh counter is suspended while the SDRAMs are in self-refresh; the SDRAM controls the refresh period. |
| 10–9 | RTIM | Refresh timing. Determines the timing operation of auto-refresh in the DRAM controller. Specifically, it determines the number of clocks inserted between a REF command and the next possible ACTV command. This same timing is used for both memory blocks controlled by the DRAM controller. This corresponds to $t_{RC}$ in the SDRAM specifications.<br>00 3 clocks<br>01 6 clocks<br>1x 9 clocks |
| 8–0 | RC | Refresh count. Controls refresh frequency. The number of bus clocks between refresh cycles is (RC + 1) * 16. Refresh can range from 16–8192 bus clocks to accommodate both standard and low-power DRAMs with bus clock operation from less than 2 MHz to greater than 50 MHz.<br>The following example calculates RC for an auto-refresh period for 4096 rows to receive 64 mS of refresh every 15.625 µs for each row (625 bus clocks at 40 MHz). This operation is the same as in asynchronous mode.<br># of bus clocks = 625 = (RC field + 1) * 16<br>RC = (625 bus clocks/16) -1 = 38.06, which rounds to 38; therefore, RC = 0x26. |

## 11.4.3.2 DRAM Address and Control Registers (DACR0/DACR1) in Synchronous Mode

The DRAM address and control registers (DACR0 and DACR1), shown in Figure 11-16, contain the base address compare value and the control bits for both memory blocks 0 and 1 of the DRAM controller. Address and timing are also controlled by bits in DACR*n*.

| | 31 … 18 | 17 16 | 15 | 14 | 13 12 | 11 | 10 9 8 | 7 | 6 | 5 4 | 3 | 2 | 1 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | BA | — | RE | — | CASL | — | CBM | — | IMRS | PS | IP | PM | — |
| Reset | Uninitialized | | 0 | Uninitialized | | | | | 0 | Uninitialized | | | |
| R/W | R/W | | | | | | | | | | | | |
| Addr | MBAR+0x108 (DACR0); 0x110(DACR1) | | | | | | | | | | | | |

**Figure 11-16. DACR0 and DACR1 Registers (Synchronous Mode)**

Table 11-13 describes DACR*n* fields.

**Table 11-13. DACR0/DACR1 Field Descriptions (Synchronous Mode)**

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th></tr>
<tr><td>31–18</td><td>BA</td><td>Base address register. With DCMR[BAM], determines the address range in which the associated DRAM block is located. Each BA bit is compared with the corresponding address of the current bus cycle. If all unmasked bits match, the address hits in the associated DRAM block. BA functions the same as in asynchronous operation.</td></tr>
<tr><td>17–16</td><td>—</td><td>Reserved, should be cleared.</td></tr>
<tr><td>15</td><td>RE</td><td>Refresh enable. Determines when the DRAM controller generates a refresh cycle to the DRAM block.<br>0 Do not refresh associated DRAM block<br>1 Refresh associated DRAM block</td></tr>
<tr><td>14</td><td>—</td><td>Reserved, should be cleared.</td></tr>
<tr><td>13–12</td><td>CASL</td><td>$\overline{CAS}$ latency. Affects the following SDRAM timing specifications. Timing nomenclature varies with manufacturers. Refer to the SDRAM specification for the appropriate timing nomenclature:
<table>
<tr><th rowspan="2">Parameter</th><th colspan="4">Number of Bus Clocks</th></tr>
<tr><th>CASL= 00</th><th>CASL = 01</th><th>CASL= 10</th><th>CASL= 11</th></tr>
<tr><td>$t_{RCD}$—$\overline{SRAS}$ assertion to $\overline{SCAS}$ assertion</td><td>1</td><td>2</td><td>3</td><td>3</td></tr>
<tr><td>$t_{CASL}$—$\overline{SCAS}$ assertion to data out</td><td>1</td><td>2</td><td>3</td><td>3</td></tr>
<tr><td>$t_{RAS}$—ACTV command to precharge command</td><td>2</td><td>4</td><td>6</td><td>6</td></tr>
<tr><td>$t_{RP}$—Precharge command to ACTV command</td><td>1</td><td>2</td><td>3</td><td>3</td></tr>
<tr><td>$t_{RWL}$,$t_{RDL}$—Last data input to precharge command</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>$t_{EP}$—Last data out to precharge command)</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table></td></tr>
<tr><td>11</td><td>—</td><td>Reserved, should be cleared.</td></tr>
<tr><td>10–8</td><td>CBM</td><td>Command and bank MUX [2:0]. Because different SDRAM configurations cause the command and bank select lines to correspond to different addresses, these resources are programmable. CBM determines the addresses onto which these functions are multiplexed.<br>CBM Command Bit Bank Select Bits<br>000 17 18 and up<br>001 18 19 and up<br>010 19 20 and up<br>011 20 21 and up<br>100 21 22 and up<br>101 22 23 and up<br>110 23 24 and up<br>111 24 25 and up<br>This encoding and the address multiplexing scheme handle common SDRAM organizations. Bank select bits include a base bit and all address bits above for SDRAMs with multiple bank select bits.</td></tr>
<tr><td>7</td><td>—</td><td>Reserved, should be cleared.</td></tr>
</table>

**Table 11-13. DACR0/DACR1 Field Descriptions (Synchronous Mode) (Continued)**

| Bit | Name | Description |
|---|---|---|
| 6 | IMRS | Initiate mode register set (MRS) command. Setting IMRS generates a MRS command to the associated SDRAMs. In initialization, IMRS should be set only after all DRAM controller registers are initialized and PALL and REFRESH commands have been issued. After IMRS is set, the next access to an SDRAM block programs the SDRAM's mode register. Thus, the address of the access should be programmed to place the correct mode information on the SDRAM address pins. Because the SDRAM does not register this information, it doesn't matter if the IMRS access is a read or a write or what, if any, data is put onto the data bus. The DRAM controller clears IMRS after the MRS command finishes.<br>0 Take no action<br>1 Initiate MRS command |
| 5–4 | PS | Port size. Indicates the port size of the associated block of SDRAM, which allows for dynamic sizing of associated SDRAM accesses. PS functions the same in asynchronous operation.<br>00 32-bit port<br>01 8-bit port<br>1x 16-bit port |
| 3 | IP | Initiate precharge all (PALL) command. The DRAM controller clears IP after the PALL command is finished. Accesses via IP should be no wider than the port size programmed in PS.<br>0 Take no action.<br>1 A PALL command is sent to the associated SDRAM block. During initialization, this command is executed after all DRAM controller registers are programmed. After IP is set, the next write to an appropriate SDRAM address generates the PALL command to the SDRAM block. |
| 2 | PM | Page mode. Indicates how the associated SDRAM block supports page-mode operation.<br>0 Page mode on bursts only. The DRAM controller dynamically bursts the transfer if it falls within a single page and the transfer size exceeds the port size of the SDRAM block. After the burst, the page closes and a precharge is issued.<br>1 Continuous page mode. The page stays open and only $\overline{\text{SCAS}}$ needs to be asserted for sequential SDRAM accesses that hit in the same page, regardless of whether the access is a burst. |
| 1–0 | — | Reserved, should be cleared. |

## 11.4.3.3 DRAM Controller Mask Registers (DMR0/DMR1)

The DMR*n*, Figure 11-17, include mask bits for the base address and for address attributes. They are the same as in asynchronous operation.

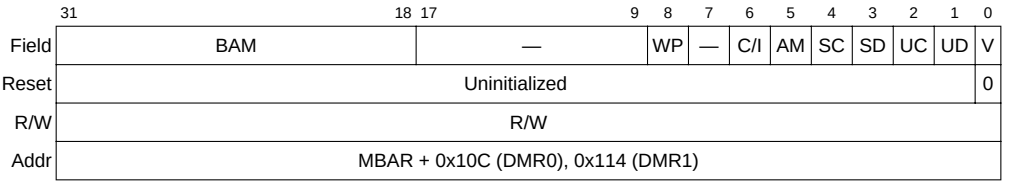



**Figure 11-17. DRAM Controller Mask Registers (DMR0 and DMR1)**

Table 11-14 describes DMR*n* fields.

**Table 11-14. DMR0/DMR1 Field Descriptions**

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th></tr>
<tr><td>31–18</td><td>BAM</td><td>Base address mask. Masks the associated DACRn[BA]. Lets the DRAM controller connect to various DRAM sizes. Mask bits need not be contiguous (see Section 11.5, “SDRAM Example.”)<br>0 The associated address bit is used in decoding the DRAM hit to a memory block.<br>1 The associated address bit is not used in the DRAM hit decode.</td></tr>
<tr><td>17–9</td><td>—</td><td>Reserved, should be cleared.</td></tr>
<tr><td>8</td><td>WP</td><td>Write protect. Determines whether the associated block of DRAM is write protected.<br>0 Allow write accesses<br>1 Ignore write accesses. The DRAM controller ignores write accesses to the memory block and an address exception occurs. Write accesses to a write-protected DRAM region are compared in the chip select module for a hit. If no hit occurs, an external bus cycle is generated. If this external bus cycle is not acknowledged, an access exception occurs.</td></tr>
<tr><td>7</td><td>—</td><td>Reserved, should be cleared.</td></tr>
<tr><td>6–1</td><td>AMx</td><td>Address modifier masks. Determine which accesses can occur in a given DRAM block.<br>0 Allow access type to hit in DRAM<br>1 Do not allow access type to hit in DRAM
<table>
<tr><th>Bit</th><th>Associated Access Type</th><th>Access Definition</th></tr>
<tr><td>C/I</td><td>CPU space/interrupt acknowledge</td><td>MOVEC instruction or interrupt acknowledge cycle</td></tr>
<tr><td>AM</td><td>Alternate master</td><td>External or DMA master</td></tr>
<tr><td>SC</td><td>Supervisor code</td><td>Any supervisor-only instruction access</td></tr>
<tr><td>SD</td><td>Supervisor data</td><td>Any data fetched during the instruction access</td></tr>
<tr><td>UC</td><td>User code</td><td>Any user instruction</td></tr>
<tr><td>UD</td><td>User data</td><td>Any user data</td></tr>
</table></td></tr>
<tr><td>0</td><td>V</td><td>Valid. Cleared at reset to ensure that the DRAM block is not erroneously decoded.<br>0 Do not decode DRAM accesses.<br>1 Registers controlling the DRAM block are initialized; DRAM accesses can be decoded.</td></tr>
</table>

## 11.4.4 General Synchronous Operation Guidelines

To reduce system logic and to support a variety of SDRAM sizes, the DRAM controller provides SDRAM control signals as well as a multiplexed row address and column address to the SDRAM.

When SDRAM blocks are accessed, the DRAM controller can operate in either burst or continuous page mode. The following sections describe the DRAM controller interface to SDRAM, the supported bus transfers, and initialization.

### 11.4.4.1 Address Multiplexing

Table 11-6 shows the generic address multiplexing scheme for SDRAM configurations. All possible address connection configurations can be derived from this table.

The following tables provide a more comprehensive, step-by-step way to determine the correct address line connections for interfacing the MCF5307 to SDRAM. To use the

tables, find the one that corresponds to the number of column address lines on the SDRAM and to the port size as seen by the MCF5307, which is not necessarily the SDRAM port size. For example, if two 1M x 16-bit SDRAMs together form a 2M x 32-bit memory, the port size is 32 bits. Most SDRAMs likely have fewer address lines than are shown in the tables, so follow only the connections shown until all SDRAM address lines are connected.

**Table 11-15. MCF5307 to SDRAM Interface (8-Bit Port, 9-Column Address Lines)**

| **MCF5307 Pins** | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A18 | A19 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Row** | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| **Column** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | | | | | | | | | | | | | |
| **SDRAM Pins** | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 | A21 | A22 |

**Table 11-16. MCF5307 to SDRAM Interface (8-Bit Port,10-Column Address Lines)**

| **MCF5307 Pins** | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A19 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Row** | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| **Column** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 18 | | | | | | | | | | | | |
| **SDRAM Pins** | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 | A21 |

**Table 11-17. MCF5307 to SDRAM Interface (8-Bit Port,11-Column Address Lines)**

| **MCF5307 Pins** | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A19 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Row** | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 19 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| **Column** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 18 | 20 | | | | | | | | | | |
| **SDRAM Pins** | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 |

**Table 11-18. MCF5307 to SDRAM Interface (8-Bit Port,12-Column Address Lines)**

| **MCF5307 Pins** | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A19 | A21 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Row** | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 19 | 21 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| **Column** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 18 | 20 | 22 | | | | | | | | |
| **SDRAM Pins** | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 |

### Table 11-19. MCF5307 to SDRAM Interface (8-Bit Port,13-Column Address Lines)

| MCF5307 Pins | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A19 | A21 | A23 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 19 | 21 | 23 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 18 | 20 | 22 | 24 | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 |

### Table 11-20. MCF5307 to SDRAM Interface (16-Bit Port, 8-Column Address Lines)

| MCF5307 Pins | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A18 | A19 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | | | | | | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 | A21 | A22 |

### Table 11-21. MCF5307 to SDRAM Interface (16-Bit Port, 9-Column Address Lines)

| MCF5307 Pins | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A18 | A19 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 17 | | | | | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 | A21 |

### Table 11-22. MCF5307 to SDRAM Interface (16-Bit Port, 10-Column Address Lines)

| MCF5307 Pins | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A18 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 18 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 17 | 19 | | | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 |

### Table 11-23. MCF5307 to SDRAM Interface (16-Bit Port, 11-Column Address Lines)

| MCF5307 Pins | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A18 | A20 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 18 | 20 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 17 | 19 | 21 | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 |

### Table 11-24. MCF5307 to SDRAM Interface (16-Bit Port, 12-Column Address Lines)

| MCF5307 Pins | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A18 | A20 | A22 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 18 | 20 | 22 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 17 | 19 | 21 | 23 | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 |

### Table 11-25. MCF5307to SDRAM Interface (16-Bit Port, 13-Column-Address Lines)

| MCF5307 Pins | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A18 | A20 | A22 | A24 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 18 | 20 | 22 | 24 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 17 | 19 | 21 | 23 | 25 | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 |

### Table 11-26. MCF5307 to SDRAM Interface (32-Bit Port, 8-Column Address Lines)

| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A18 | A19 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 16 | | | | | | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 | A21 |

### Table 11-27. MCF5307 to SDRAM Interface (32-Bit Port, 9-Column Address Lines)

| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A19 | A20 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 17 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 16 | 18 | | | | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 | A20 |

### Table 11-28. MCF5307 to SDRAM Interface (32-Bit Port, 10-Column Address Lines)

| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A19 | A21 | A22 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 17 | 19 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 16 | 18 | 20 | | | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 | A19 |

**Table 11-29. MCF5307 to SDRAM Interface (32-Bit Port, 11-Column Address Lines)**

| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A19 | A21 | A23 | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 17 | 19 | 21 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 16 | 18 | 20 | 22 | | | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 | A18 |

**Table 11-30. MCF5307 to SDRAM Interface (32-Bit Port, 12-Column Address Lines)**

| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A19 | A21 | A23 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Row | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 17 | 19 | 21 | 23 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| Column | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 16 | 18 | 20 | 22 | 24 | | | | | | |
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | A16 | A17 |

### 11.4.4.2 Interfacing Example

The tables in the previous section can be used to configure the interface in the following example. To interface one 2M x 32-bit x 4 bank SDRAM component (8 columns) to the MCF5307, the connections would be as shown in Table 11-31.

**Table 11-31. SDRAM Hardware Connections**

| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 = CMD | BA0 | BA1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A18 | A19 | A20 | A21 | A22 |

### 11.4.4.3 Burst Page Mode

SDRAM can efficiently provide data when an SDRAM page is opened. As soon as $\overline{\text{SCAS}}$ is issued, the SDRAM accepts a new address and asserts $\overline{\text{SCAS}}$ every clock for as long as accesses are in that page. In burst page mode, there are multiple read or write operations for every ACTV command in the SDRAM if the requested transfer size exceeds the port size of the associated SDRAM. The primary cycle of the transfer generates the ACTV and READ or WRITE commands; secondary cycles generate only READ or WRITE commands. As soon as the transfer completes, the PALL command is generated to prepare for the next access.

Note that in synchronous operation, burst mode and address incrementing during burst cycles are controlled by the MCF5307 DRAM controller. Thus, instead of the SDRAM enabling its internal burst incrementing capability, the MCF5307 controls this function. This means that the burst function that is enabled in the mode register of SDRAMs must be disabled when interfacing to the MCF5307.

Figure 11-18 shows a burst read operation. In this example, DACR[CASL] = 01, for an $\overline{\text{SRAS}}$-to-$\overline{\text{SCAS}}$ delay ($t_{RCD}$) of 2 BCLKO cycles. Because $t_{RCD}$ is equal to the read CAS

latency ($\overline{SCAS}$ assertion to data out), this value is also 2 BCLKO cycles. Notice that NOPs are executed until the last data is read. A PALL command is executed one cycle after the last data transfer.

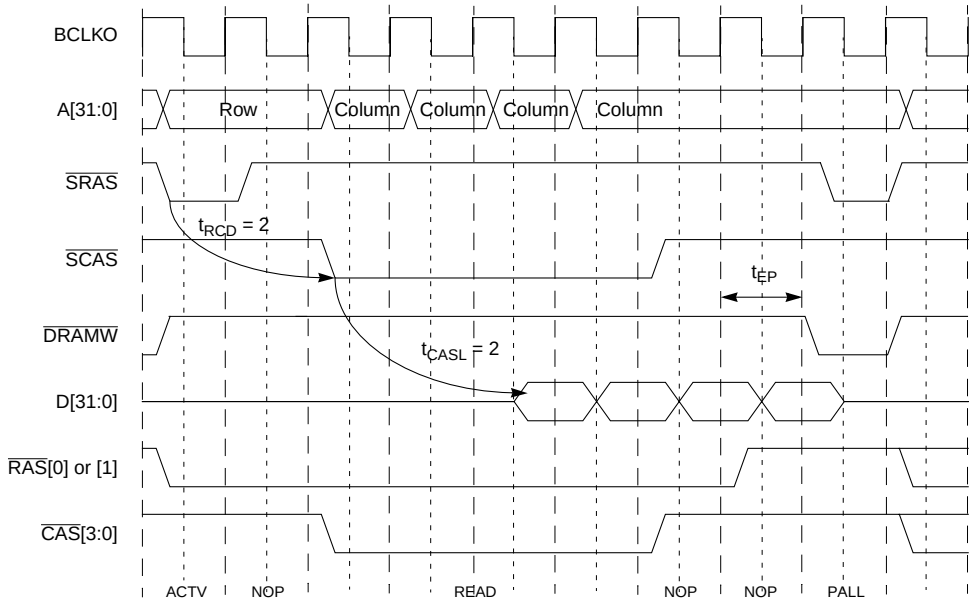



**Figure 11-18. Burst Read SDRAM Access**

Figure 11-19 shows the burst write operation. In this example, DACR[CASL] = 01, which creates an $\overline{SRAS}$-to-$\overline{SCAS}$ delay ($t_{RCD}$) of 2 BCLKO cycles. Note that data is available upon $\overline{SCAS}$ assertion and a burst write cycle completes two cycles sooner than a burst read cycle with the same $t_{RCD.}$ The next bus cycle is initiated sooner, but cannot begin an SDRAM cycle until the precharge-to-ACTV delay completes.

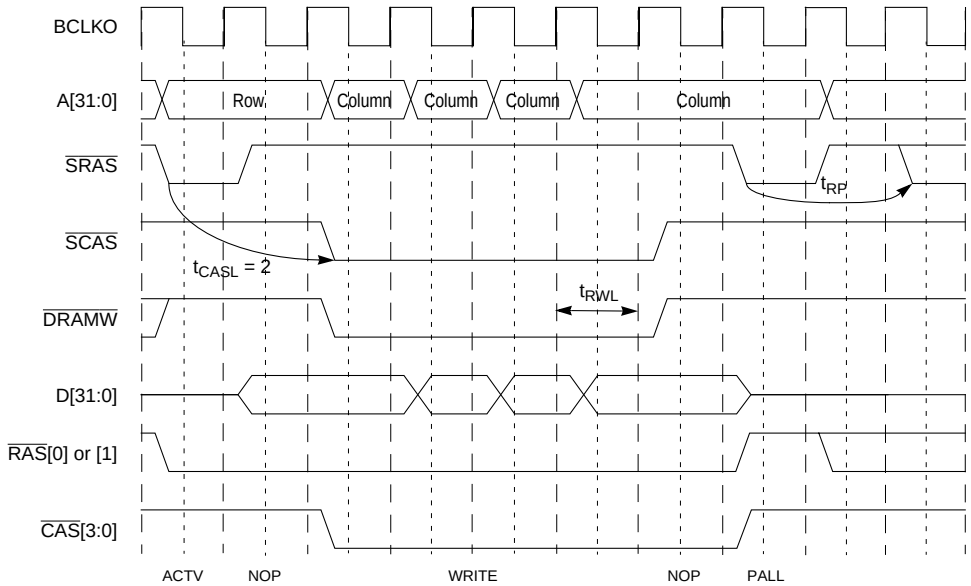



**Figure 11-19. Burst Write SDRAM Access**

Accesses in synchronous burst page mode always cause the following sequence:

1. ACTV command
2. NOP commands to assure $\overline{\text{SRAS}}$-to-$\overline{\text{SCAS}}$ delay (if $\overline{\text{CAS}}$ latency is 1, there are no NOP commands).
3. Required number of READ or WRITE commands to service the transfer size with the given port size.
4. Some transfers need more NOP commands to assure the ACTV-to-precharge delay.
5. PALL command
6. Required number of idle clocks inserted to assure precharge-to-ACTV delay.

### 11.4.4.4 Continuous Page Mode

Continuous page mode is identical to burst page mode, except that it allows the processor core to handle successive bus cycles that hit the same page without having to close the page. When the current bus cycle finishes, the MCF5307 core internal pipelined bus can predict whether the upcoming cycle will hit in the same page.

- If the next bus cycle is not pending or misses in the page, the PALL command is generated to the SDRAM.
- If the next bus cycle is pending and hits in the page, the page is left open, and the next SDRAM access begins with a READ or WRITE command.

- Because of the nature of the internal CPU pipeline this condition does not occur often; however, the use of continuous page mode is recommended because it can provide a slight performance increase.

Figure 11-20 shows two read accesses in continuous page mode. Note that there is no precharge between the two accesses. Also notice that the second cycle begins with a read operation with no ACTV command.

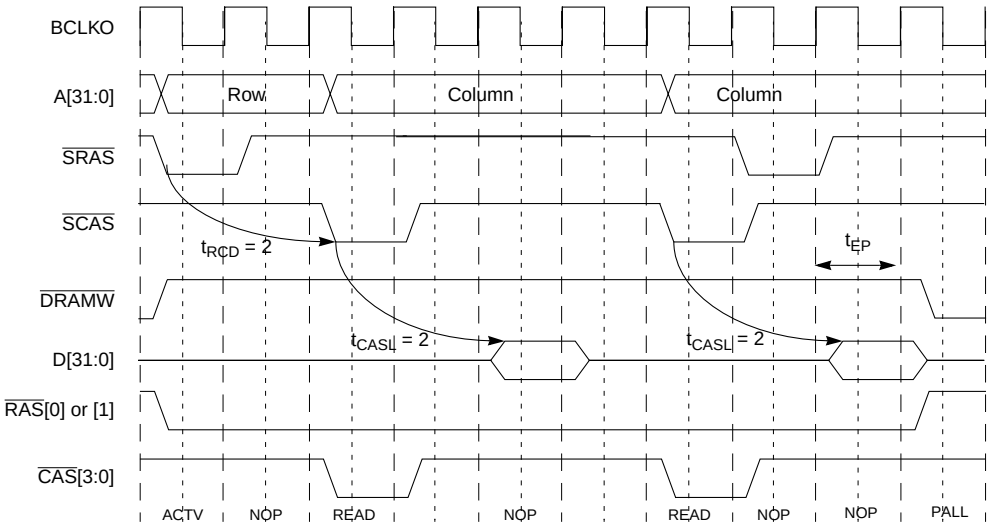



**Figure 11-20. Synchronous, Continuous Page-Mode Access—Consecutive Reads**

Figure 11-21 shows a write followed by a read in continuous page mode. Because the bus cycle is terminated with a WRITE command, the second cycle begins sooner after the write than after the read. A read requires data to be returned before the bus cycle can terminate. Note that in continuous page mode, secondary accesses output the column address only.

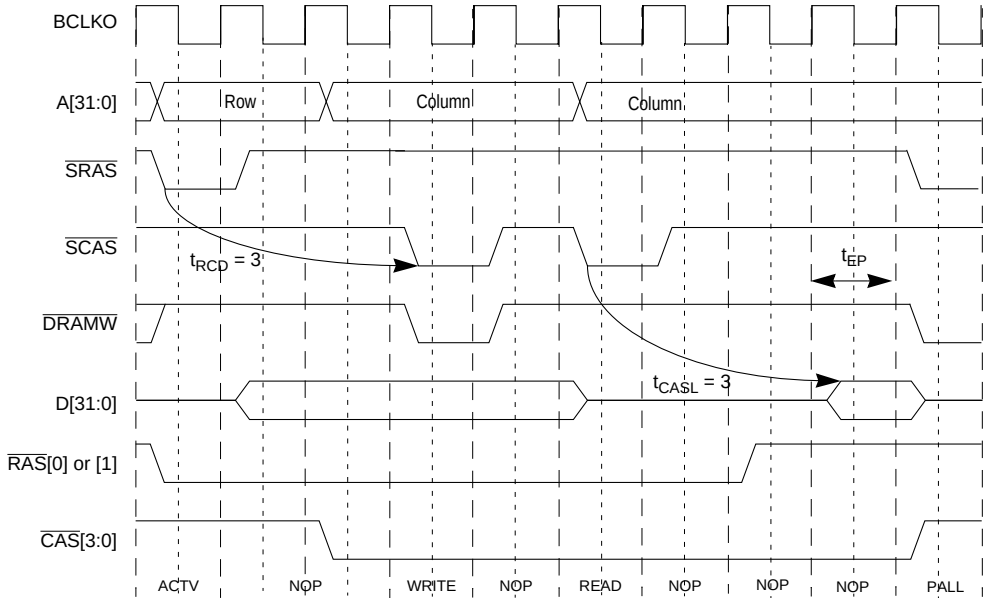



**Figure 11-21. Synchronous, Continuous Page-Mode Access—Read after Write**

### 11.4.4.5 Auto-Refresh Operation

The DRAM controller is equipped with a refresh counter and control. This logic is responsible for providing timing and control to refresh the SDRAM. Once the refresh counter is set, and refresh is enabled, the counter counts to zero. At this time, an internal refresh request flag is set and the counter begins counting down again. The DRAM controller completes any active burst operation and then performs a PALL operation. The DRAM controller then initiates a refresh cycle and clears the refresh request flag. This refresh cycle includes a delay from any precharge to the auto-refresh command, the auto-refresh command, and then a delay until any ACTV command is allowed. Any SDRAM access initiated during the auto-refresh cycle is delayed until the cycle is completed.

Figure 11-22 shows the auto-refresh timing. In this case, there is an SDRAM access when the refresh request becomes active. The request is delayed by the precharge to ACTV delay programmed into the active SDRAM bank by the CAS bits. The REF command is then generated and the delay required by DCR[RTIM] is inserted before the next ACTV command is generated. In this example, the next bus cycle is initiated, but does not generate an SDRAM access until $T_{RC}$ is finished. Because both chip selects are active during the REF command, it is passed to both blocks of external SDRAM.

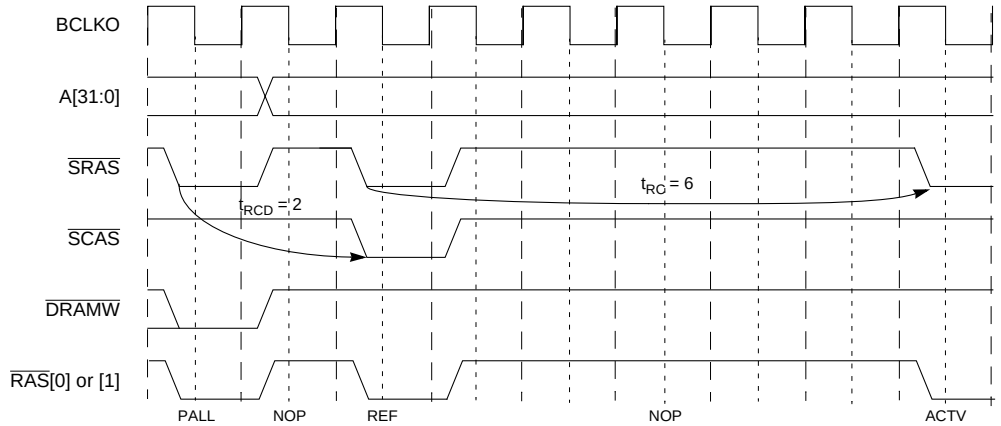



**Figure 11-22. Auto-Refresh Operation**

## 11.4.4.6 Self-Refresh Operation

Self-refresh is a method of allowing the SDRAM to enter into a low-power state, while at the same time to perform an internal refresh operation and to maintain the integrity of the data stored in the SDRAM. The DRAM controller supports self-refresh with DCR[IS]. When IS is set, the SELF command is sent to the SDRAM. When IS is cleared, the SELFX command is sent to the DRAM controller. Figure 11-23 shows the self-refresh operation.

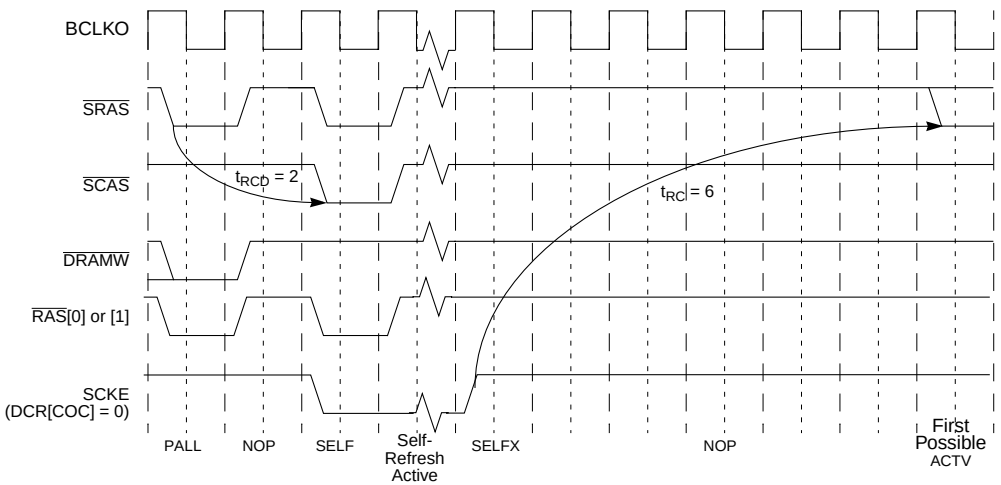



**Figure 11-23. Self-Refresh Operation**

## 11.4.5 Initialization Sequence

Synchronous DRAMs have a prescribed initialization sequence. The DRAM controller supports this sequence with the following procedure:

1. SDRAM control signals are reset to idle state. Wait the prescribed period after reset before any action is taken on the SDRAMs. This is normally around 100 µs.
2. Initialize the DCR, DACR, and DMR in their operational configuration. Do not yet enable PALL or REF commands.
3. Issue a PALL command to the SDRAMs by setting DCR[IP] and accessing a SDRAM location. Wait the time (determined by $t_{RP}$) before any other execution.
4. Enable refresh (set DACR[RE]) and wait for at least 8 refreshes to occur.
5. Before issuing the MRS command, determine if the DMR mask bits need to be modified to allow the MRS to execute properly
6. Issue the MRS command by setting DACR[IMRS] and accessing a location in the SDRAM. Note that mode register settings are driven on the SDRAM address bus, so care must be taken to change DMR[BAM] if the mode register configuration does not fall in the address range determined by the address mask bits. After the mode register is set, DMR mask bits can be restored to their desired configuration.

### 11.4.5.1 Mode Register Settings

It is possible to configure the operation of SDRAMs, namely their burst operation and $\overline{\text{CAS}}$ latency, through the SDRAM component's mode register. $\overline{\text{CAS}}$ latency is a function of the speed of the SDRAM and the bus clock of the DRAM controller. The DRAM controller operates at a $\overline{\text{CAS}}$ latency of 1, 2, or 3.

Although the MCF5307 DRAM controller supports bursting operations, it does not use the bursting features of the SDRAMs. Because the MCF5307 can burst operand sizes of 1, 2, 4, or 16 bytes long, the concept of a fixed burst length in the SDRAMs mode register becomes problematic. Therefore, the MCF5307 DRAM controller generates the burst cycles rather than the SDRAM device. Because the MCF5307 generates a new address and a READ or WRITE command for each transfer within the burst, the SDRAM mode register should be set either to a burst length of one or to not burst. This allows bursting to be controlled by the MCF5307 instead.

The SDRAM mode register is written by setting the associated block's DACR[IMRS]. First, the base address and mask registers must be set to the appropriate configuration to allow the mode register to be set. Note that improperly set DMR mask bits may prevent access to the mode register address. Thus, the user should determine the mapping of the mode register address to the MCF5307 address bits to find out if an access is blocked. If the DMR setting prohibits mode register access, the DMR should be reconfigured to enable the access and then set to its necessary configuration after the MRS command executes.

The associated CBM bits should also be initialized. After DACR[IMRS] is set, the next access to the SDRAM address space generates the MRS command to that SDRAM. The address of the access should be selected to place the correct mode information on the SDRAM address pins. The address is not multiplexed for the MRS command. The MRS access can be a read or write. The important thing is that the address output of that access needs the correct mode programming information on the correct address bits.

Figure 11-24 shows the MRS command, which occurs in the first clock of the bus cycle.

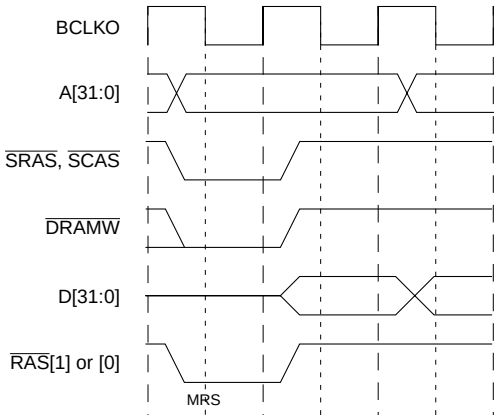



**Figure 11-24. Mode Register Set (MRS) Command**

# 11.5 SDRAM Example

This example interfaces a 2M x 32-bit x 4 bank SDRAM component to a MCF5307 operating at 40 MHz. Table 11-32 lists design specifications for this example.

**Table 11-32. SDRAM Example Specifications**

| Parameter | Specification |
|---|---|
| Speed grade (-8E) | 40 MHz (25-nS period) |
| 10 rows, 8 columns | |
| Two bank-select lines to access four internal banks | |
| ACTV-to-read/write delay ($t_{RCD}$) | 20 nS (min.) |
| Period between auto refresh and ACTV command ($t_{RC}$) | 70 nS |
| ACTV command to precharge command ($t_{RAS}$) | 48 nS (min.) |
| Precharge command to ACTV command ($t_{RP}$) | 20 nS (min.) |
| Last data input to PALL command ($t_{RWL}$) | 1 bus clock (25 nS) |
| Auto refresh period for 4096 rows ($t_{REF}$) | 64 mS |

## 11.5.1 SDRAM Interface Configuration

To interface this component to the MCF5307 DRAM controller, use the connection table that corresponds to a 32-bit port size with 8 columns (Table 11-26). Two pins select one of four banks when the part is functional. Table 11-33 shows the proper hardware hook-up.

**Table 11-33. SDRAM Hardware Connections**

| MCF5307 Pins | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A17 | A18 | A19 | A20 | A21 | A22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SDRAM Pins | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 = CMD | BA0 | BA1 |

## 11.5.2 DCR Initialization

At power-up, the DCR has the following configuration if synchronous operation and SDRAM address multiplexing is desired.

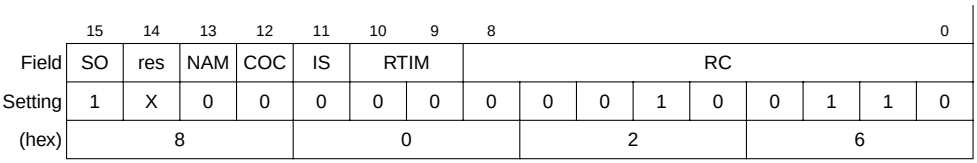



**Figure 11-25. Initialization Values for DCR**

This configuration results in a value of 0x8026 for DCR, as shown in Table 11-34.

**Table 11-34. DCR Initialization Values**

| Bits | Name | Setting | Description |
|---|---|---|---|
| 15 | SO | 1 | Indicating synchronous operation |
| 14 | — | x | Don't care (reserved) |
| 13 | NAM | 0 | Indicating SDRAM controller multiplexes address lines internally |
| 12 | COC | 0 | SCKE is used as clock enable instead of command bit because user is not multiplexing address lines externally and requires external command feed. |
| 11 | IS | 0 | At power-up, allowing power self-refresh state is not appropriate because registers are being set up. |
| 10–9 | RTIM | 00 | Because $t_{RC}$ value is 70 nS, indicating a 3-clock refresh-to-ACTV timing. |
| 8–0 | RC | 0x26 | Specification indicates auto-refresh period for 4096 rows to be 64 mS or refresh every 15.625 µs for each row, or 625 bus clocks at 40 MHz. Because DCR[RC] is incremented by 1 and multiplied by 16, RC = (625 bus clocks/16) -1 = 38.06 = 0x38 |

## 11.5.3 DACR Initialization

As shown in Figure 11-26, in this example the SDRAM is programmed to access only the second 512-Kbyte block of each 1-Mbyte partition in the SDRAM (each 16 Mbytes). The starting address of the SDRAM is 0xFF80_0000. Continuous page mode feature is used.

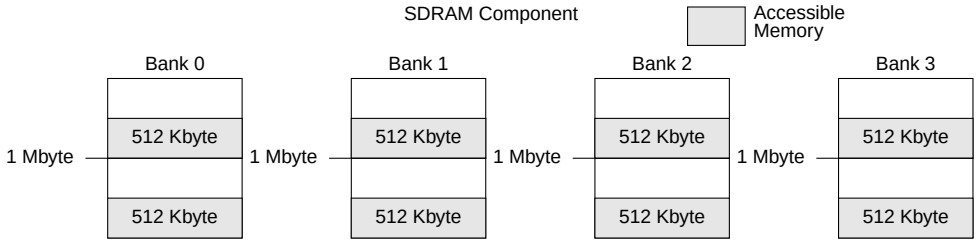



**Figure 11-26. SDRAM Configuration**

The DACRs should be programmed as shown in Figure 11-27.

| | 31 … 18 | 17 | 16 |
|---|---|---|---|
| Field | BA | — | |
| Setting | 1111_1111_1000_10 | xx | |
| (hex) | 15 | 15 | 8 | 8 |

| | 15 | 14 | 13 12 | 11 | 10 … 8 | 7 | 6 | 5 … 4 | 3 | 2 | 1 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | RE | — | CASL | — | CBM | — | IMRS | PS | IP | PM | — |
| Setting | 0 | X | 00 | X | 011 | X | 0 | 00 | 0 | 1 | xx |
| (hex) | 0 | | 3 | | 0 | | 4 | | | | |

**Figure 11-27. DACR Register Configuration**

This configuration results in a value of DACR0 = 0xFF88_0304, as described in Table 11-35. DACR1 initialization is not needed because there is only one block. Subsequently, DACR1[RE,IMRS,IP] should be cleared; everything else is a don't care.

**Table 11-35. DACR Initialization Values**

| Bits | Name | Setting | Description |
|---|---|---|---|
| 31–18 | BA | | Base address. So DACR0[31–16] = 0xFF88, which places the starting address of the SDRAM accessible memory at 0xFF88_0000. |
| 17–16 | — | | Reserved. Don't care. |
| 15 | RE | 0 | 0, which keeps auto-refresh disabled because registers are being set up at this time. |
| 14 | — | | Reserved. Don't care. |
| 13–12 | CASL | 00 | Indicates a delay of data 1 cycle after $\overline{\text{CAS}}$ is asserted |
| 11 | — | | Reserved. Don't care. |
| 10–8 | CBM | 011 | Command bit is pin 20 and bank selects are 21 and up. |
| 7 | — | | Reserved. Don't care. |
| 6 | IMRS | 0 | Indicates MRS command has not been initiated. |
| 5–4 | PS | 00 | 32-bit port. |
| 3 | IP | 0 | Indicates precharge has not been initiated. |

**Table 11-35. DACR Initialization Values**

| Bits | Name | Setting | Description |
|---|---|---|---|
| 2 | PM | 1 | Indicates continuous page mode |
| 1–0 | — | | Reserved. Don't care. |

## 11.5.4 DMR Initialization

In this example, again, only the second 512-Kbyte block of each 1-Mbyte space is accessed in each bank. In addition the SDRAM component is mapped only to readable and writable supervisor and user data. The DMRs have the following configuration.

| | 31 | | | | | | | | | | | | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | BAM | | | | | | | | | | | | | — | |
| Setting | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | X | X |
| (hex) | 0 | | | | 0 | | | | 7 | | | | 4 | | | |

| | 15 | | | | | | | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | — | | | | | | | | WP | — | C/I | AM | SC | SD | UC | UD | V |
| Setting | X | X | X | X | X | X | X | 0 | X | 1 | 1 | 1 | 0 | 1 | 0 | 1 | |
| (hex) | 0 | | | | 0 | | | | 7 | | | | 5 | | | | |

**Figure 11-28. DMR0 Register**

With this configuration, the DMR0 = 0x0074_0075, as described in Table 11-36.

**Table 11-36. DMR0 Initialization Values**

| Bits | Name | Setting | Description |
|---|---|---|---|
| 31–16 | BAM | | With bits 17 and 16 as don't cares, BAM = 0x0074, which leaves bank select bits and upper 512K select bits unmasked. Note that bits 22 and 21 are set because they are used as bank selects; bit 20 is set because it controls the 1-Mbyte boundary address. |
| 15–9 | — | | Reserved. Don't care. |
| 8 | WP | 0 | Allow reads and writes |
| 7 | — | | Reserved |
| 6 | C/I | 1 | Disable CPU space access |
| 5 | AM | 1 | Disable alternate master access |
| 4 | SC | 1 | Disable supervisor code accesses |
| 3 | SD | 0 | Enable supervisor data accesses |
| 2 | UC | 1 | Disable user code accesses |
| 1 | UD | 0 | Enable user data accesses |
| 0 | V | 1 | Enable accesses. |

## 11.5.5 Mode Register Initialization

When DACR[IMRS] is set, a bus cycle initializes the mode register. If the mode register setting is read on A[10:0] of the SDRAM on the first bus cycle, the bit settings on the corresponding MCF5307 address pins must be determined while being aware of masking requirements.

Table 11-37 lists the desired initialization setting:

**Table 11-37. Mode Register Initialization**

| MCF5307 Pins | SDRAM Pins | Mode Register Initialization | |
|---|---|---|---|
| A20 | A10 | Reserved | X |
| A19 | A9 | WB | 0 |
| A18 | A8 | Opmode | 0 |
| A17 | A7 | Opmode | 0 |
| A9 | A6 | CASL | 0 |
| A10 | A5 | CASL | 0 |
| A11 | A4 | CASL | 1 |
| A12 | A3 | BT | 0 |
| A13 | A2 | BL | 0 |
| A14 | A1 | BL | 0 |
| A15 | A0 | BL | 0 |

Next, this information is mapped to an address to determine the hexadecimal value.

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | | | | | | | | | | | | | | | | |
| Setting | X | X | X | X | X | X | X | X | X | X | X | X | 0 | 0 | 0 | X |
| (hex) | 0 | | | | 0 | | | | 0 | | | | 0 | | | |

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Field | | | | | | | | | | | | | | | | V |
| Setting | 0 | 0 | 0 | 0 | 1 | 0 | 0 | X | X | X | X | X | X | X | X | X |
| (hex) | 0 | | | | 8 | | | | 0 | | | | 0 | | | |

**Figure 11-29. Mode Register Mapping to MCF5307 A[31:0]**

Although A[31:20] corresponds to the address programmed in DACR0, according to how DACR0 and DMR0 are initialized, bit 19 must be set to hit in the SDRAM. Thus, before the mode register bit is set, DMR0[19] must be set to enable masking.

## 11.5.6 Initialization Code

The following assembly code initializes the SDRAM example.

Power-Up Sequence:

```
move.w  #0x8026, d0           //Initialize DCR
move.w  d0, DCR
move.l  #0xFF880300, d0       //Initialize DACR0
move.l  d0, DACR0
move.l  #0x00740075, d0       //Initialize DMR0
move.l  d0, DMR0
```

Precharge Sequence:

```
move.l  #0xFF880308, d0       //Set DACR0[IP]
move.l  d0, DACR0
move.l  #0xBEADDEED, d0       //Write to memory location to init. precharge
move.l  d0, 0xFF880000
```

Refresh Sequence:

```
move.l  #0xFF888300, d0       //Enable refresh bit in DACR0
move.l  d0, DACR0
```

Mode Register Initialization Sequence:

```
move.l  #0x00600075, d0       //Mask bit 19 of address
move.l  d0, DMR0
move.l  #0xFF888340, d0       //Enable DACR0[IMRS]; DACR0[RE] remains set
move.l  d0, DACR0
move.l  #0x00000000, d0       //Access  SDRAM  address  to  initialize  mode
register
move.l  d0, 0xFF800800
```