OKI Semiconductor

MSM5718B70

18-Megabit RDRAM (2M × 9)

DESCRIPTION

The 18-Megabit Rambus™ DRAM (RDRAM™) is an extremely high-speed CMOS DRAM organized as 2M words by 9 bits. It is capable of bursting up to 256 bytes of data at less than 2 nanoseconds per byte. The use of Rambus Signaling Logic (RSL) technology makes transfer rates greater than 500 MHz achievable while using conventional system and board design methodologies. Lower effective latency is attained by operating the dual 2KByte sense amplifiers as high speed caches, and by using random access mode to facilitate large block transfers.

RDRAMs are general purpose high-performance memory devices suitable for use in a broad range of applications including PC and consumer main memory, graphics, video, and any other application where high-performance is required.

FEATURES

- Rambus Interface:
  Over 500 MB/sec peak transfer rate per RDRAM
  Rambus Signaling Logic (RSL) interface
  Synchronous protocol for fast block-oriented transfers
  Direct connection to Rambus ASICs, MPUs, and Peripherals
  15 active signals require just 32 total pins on the controller interface (including power)
  3.3 V operation
  Additional/multiple Rambus Channels provide an additional 500 MB/second band-width each
- Dual 2KByte sense amplifiers may be operated as caches for low latency access
- Random Access mode enables any burst order at full band width
- Features for graphics include random-access mode, write-per-bit and mask-per-bit operations
- Control and refresh logic entirely self-contained
- On-chip registers for flexible addressing and timing
- Available in horizontal surface mount plastic package (SHP32-P-1125-0.65-K)
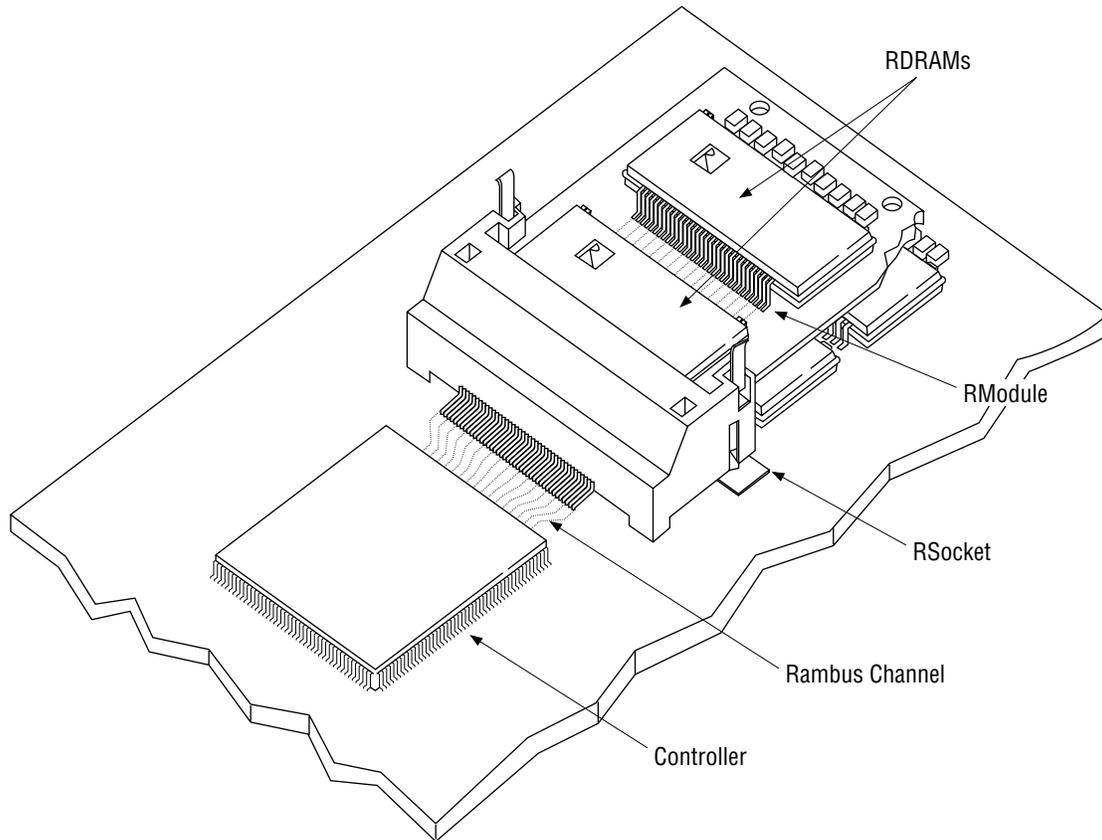
Fig. 1 Rambus Subsystem

SYSTEM BENEFITS

• Fully engineered solution includes clock chips, memory expansion sockets and simple layout
• For graphics subsystems addressing display resolutions of 1024×768×8 or above, it provides high performance, fewest controller pins, and ease of memory expansion
• For Pentium® processor class main memory, it provides fast memory subsystem, fewer components, and 2 MB granularity
• Sufficient performance for unified memory system architectures in consumer applications

PART NUMBERS

The 18-Megabit RDRAMs are available in horizontal surface mount plastic package (SHP), with a 500 MHz clock rate, a 533 MHz clock rate and a 600 MHz clock rate. The part numbers for the various options are shown in Table 1.

Table 1 Part Numbers

500 MHz	533 MHz	600 MHz
MSM5718B70-50GS-K	MSM5718B70-53GS-K	MSM5718B70-60GS-K
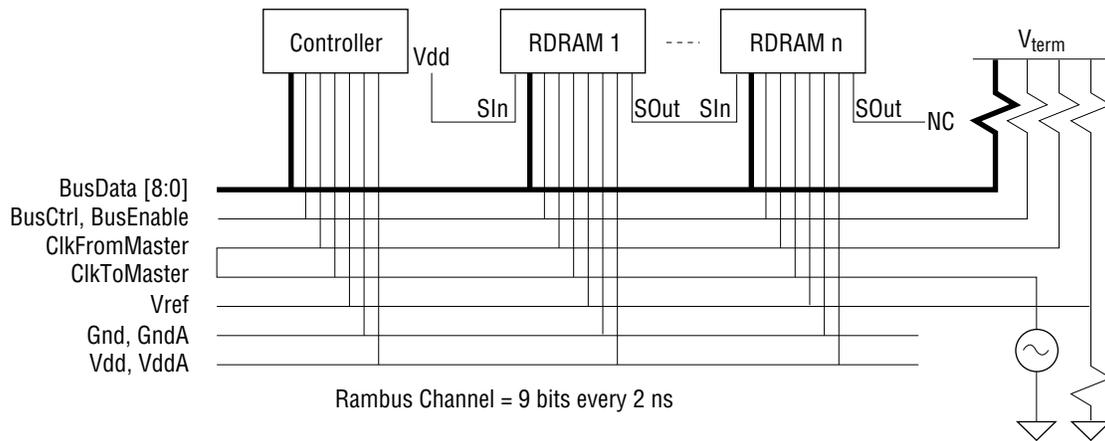
**Fig. 2 Controller and RDRAMs Connect to Terminated Transmission Lines**

RAMBUS SYSTEM OVERVIEW

A typical Rambus memory system has three main elements: the Rambus Controller, the channel, and the RDRAMs. The logical representation of this is shown in Fig. 2.

The Rambus channel is a high-speed, byte-wide, synchronous bus used to connect Rambus devices together. The channel carries all address, data, and control information to and from devices. Transfer of data on the Rambus channel is managed through the use of a high level block-oriented protocol.

3
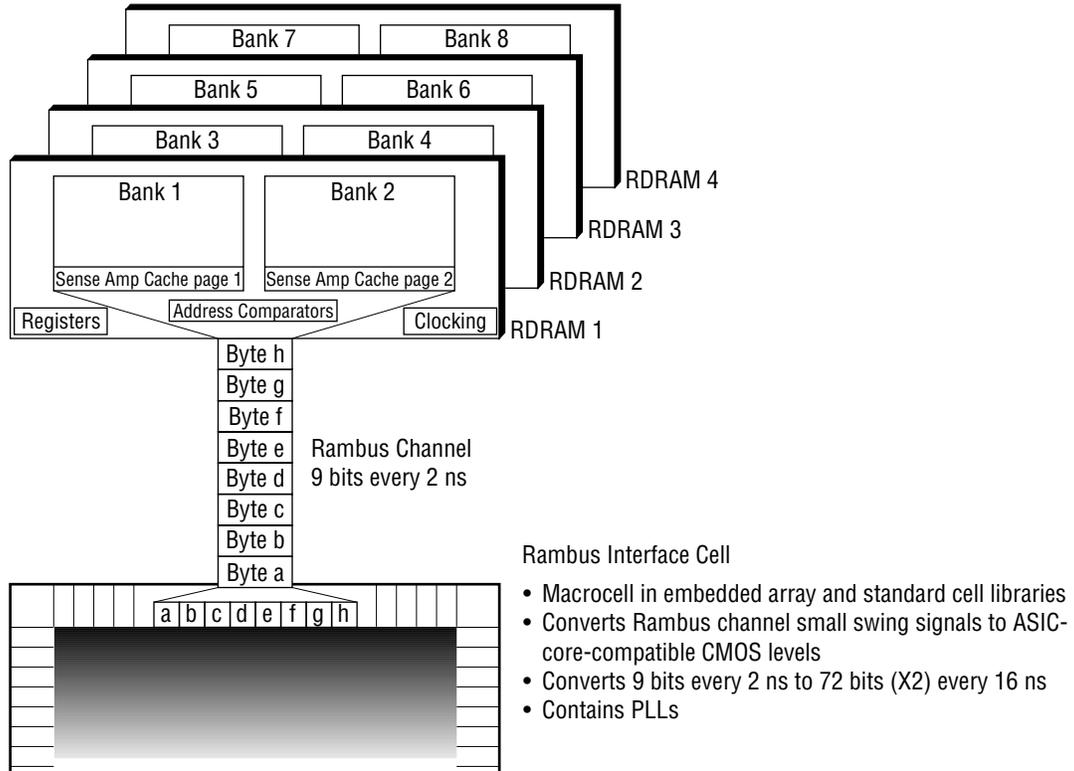
Fig. 3 Data Transfer on the Rambus Channel

The Rambus channel has thirteen high-speed Rambus Signaling Logic (RSL) I/O signals that are used to transfer information at 2 nanosecond intervals. These signals use low voltage swings (logic 0 = 2.25 V, logic 1 = 1.55 V) to achieve high bus speeds. Two TTL level signals are used for initialization and powerdown operation. Fourteen signals supply power and DC voltage references to the RDRAM, and the remaining pins are No Connects (reserved for future expansion).

The Rambus interface is implemented on both master and slave devices. Rambus masters (ASIC devices, memory controllers, graphics engines, peripheral chips, or microprocessors) are the only devices capable of generating transaction requests. RDRAMs are slave devices and can respond to requests from master devices.

Fig. 1 shows a typical physical implementation of a Rambus system. It includes a controller ASIC that acts as the Channel master and a base set of RDRAMs soldered directly to the board. An RSocket™ is included on the Channel for memory upgrade using RModule™ expansion cards.

RDRAM PACKAGES AND PINOUTS

RDRAMs are available in horizontal surface mount plastic package (SHP).
The package has 32 signal pins and four mechanical pins that provide support for the device. The mechanical pins are located on the opposite side from the signal leads in the SHP.

Signal	Pin
V_{DD}	1
Gnd	2
BusData8	3
Gnd	4
BusData7	5
(NC)	6
BusEnable	7
V_{DD}	8
BusData6	9
Gnd	10
BusData5	11
V_{DDA}	12
RxClk	13
GndA	14
TxClk	15
V_{DD}	16
BusData4	17
Gnd	18
BusCtrl	19
SIn	20
V_{REF}	21
SOut	22
BusData3	23
Gnd	24
BusData2	25
(NC)	26
BusData1	27
Gnd	28
BusData0	29
(NC)	30
Gnd	31
V_{DD}	32

Fig. 4 SHP Pin Numbering

Table 2 Pin Descriptions

| Signal	I/O	Description
BusData [8-0]	I/O	Signal lines for request, write data, and read data packets. The request packet contains the address, operation codes, and the count of the bytes to be transferred. These are low-swing active-low signals referenced to Vref.
RxClk	I	Receive clock. Incoming request and write data packets are aligned to this clock. This is a low-swing active-low signal referenced to Vref.
TxClk	I	Transmit clock. Outgoing acknowledge and read data packets are aligned with this clock. This is a low-swing active-low signal referenced to Vref.
Vref	I	Logic threshold reference voltage for low swing signals.
BusCtrl	I/O	Control signal to frame packets transmit part of the operation code to acknowledge requests, and to interrupt (terminate) pending transactions. This is a low-swing active-low signal referenced to Vref.
BusEnable	I	Control signal to manage the operating modes of the RDRAMs and to transfer column addresses for random-access (non-sequential) transactions. This is a low-swing active-low signal referenced to Vref.
Vdd, VddA		+3.3V power supply. VddA is a separate analog supply for clock recovery in the RDRAM.
Gnd, GndA		Circuit ground. GndA is a separate analog ground for clock generation in the RDRAM.
SIn	I	Initialization daisy chain input. TTL levels. Active high.
SOut	O	Initialization daisy chain output. TTL levels. Active high.
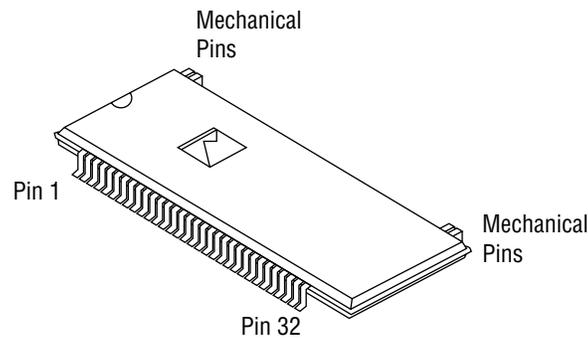
Fig. 5 SHP Package

PROTOCOL

The transaction protocol used in Rambus systems is built from several types of information packets. These include the request, acknowledge, serial mode, and data packets.

Request Packet

A master device initiates a transaction by generating a six-byte request packet containing address, control, and byte count information as shown in Fig. 6. The Op and OpX fields in the Request packet contain a command that is used to instruct the RDRAM which operation is being requested. A summary of these commands is shown in Table 3.

| Clock Cycle Number	Bus-Enable	Bus-Ctrl	BusData (8:0)				
[0] even	—	Start	Op[0]	Adr[9:2]			
[0] odd	—	Op[1]	Op[3]	Adr[17:10]			
[1] even	—	OpX[1]	Adr[26:18]				
[1] odd	—	Op[2]	Adr[35:27]				
[2] even	—	OpX[0]	ReqUnimp[5:4]	Count[6, 4, 2]		ReqUnimp[3:0]	
[2] odd	—	ReqUnimp[8:6]		Count[7, 5, 3]	Count[1:0]	Adr[1:0]	

Time
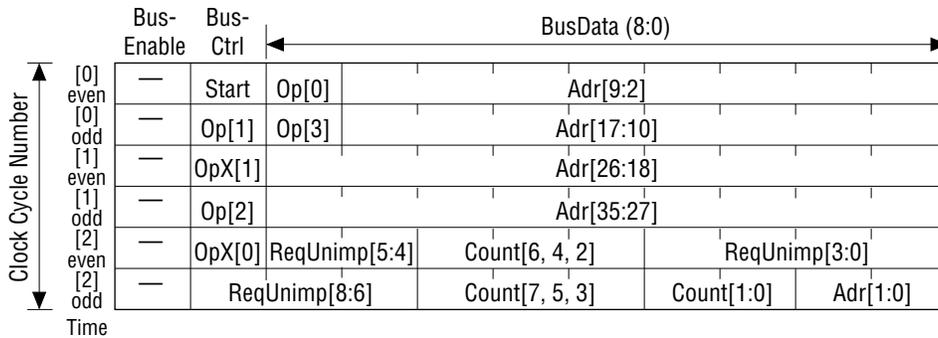
Fig. 6 Request Packet [1]

Note: 1.   A — | in this diagram signifies that this pin is not used by this packet. If it is not used by another packet, it is pulled to a logic zero value.

Table 3 Command Summary

Op[3:0]	OpX[1:0]	Name	Description
0000	00	Rseq	Read sequential data from memory space.
0000	01	Rnsq	Read random-access (non-sequential) data from memory space.
0100	00	WseqNpb	Write sequential data to memory space with no per-bit mask application.
0100	01	WseqDpb	Write sequential data to memory space with data-per-bit masking. Static bit masks are supplied by the MDReg while write data is supplied in the data packet.
0100	10	WseqBpb	Write sequential data to memory space with mask-per-bit masking. Both write data and dynamic bit masks are supplied in the data packet.
0100	11	WseqMpb	Write sequential data to memory space with mask-per-bit masking. Static write data is supplied by the MDReg while dynamic bit masks are supplied in the data packet.
0110	00	Rreg	Read data from register space.
0111	00	Wreg	Write data to register space.
1000	00	WnsqNpb	Write random-access (non-sequential) data to memory space with no per-bit mask application.
1000	01	WnsqDpb	Write random-access (non-sequential) data to memory space with data-per-bit masking. Static bit masks are supplied by the MDReg while write data is supplied in the data packet.
1000	10	WnsqBpb	Write random-access (non-sequential) data to memory space with mask-per-bit masking. Both write data and dynamic bit masks are supplied in the data packet.
1000	11	WnsqMpb	Write random-access (non-sequential) data to memory space with mask-per-bit masking. Static write data is supplied by the MDReg while dynamic bit masks are supplied in the data packet.
1100	00	WbnsNpb	Write random-access (non-sequential) data to memory space with byte masking and no per-bit mask application. Both byte masks and write data are supplied in the data packet.
1100	01	WbnsDpb	Write random-access (non-sequential) data to memory space with byte masking and data-per-bit masking. Static bit masks are supplied by the MDReg while byte masks and write data are supplied in the data packets.
1100	11	WbnsMpb	Write random-access (non-sequential) data to memory space with byte masking and mask-per-bit masking. Static write data is supplied by the MDReg while byte masks and dynamic bit masks are supplied in the data packets.
1111	00	WregB	Broadcast write to register space of all responding devices with no acknowledge permitted.

Fig. 8 Sample 16-Byte Read and Write Transactions

Serial Address Packet

The non-sequential (Random Access) commands specify the eight column address bits needed to access random octbytes within the open page. These address bits are provided using a Serial Address packet.

Clock Cycle Number	Bus-Enable	Bus-Ctrl	BusData (8:0)
[4] even	SAdr [1][3]	—	—
[4] odd	SAdr [1][4]	—	—
[5] even	SAdr [1][5]	—	—
[5] odd	SAdr [1][6]	—	—
[6] even	SAdr [1][7]	—	—
[6] odd	SAdr [1][8]	—	—
[7] even	SAdr [1][9]	—	—
[7] odd	SAdr [1][10]	—	—
⋮	⋮	⋮	
[4*n] even	SAdr [n][3]	—	—
[4*n] odd	SAdr [n][4]	—	—
[4*n+1] even	SAdr [n][5]	—	—
[4*n+1] odd	SAdr [n][6]	—	—
[4*n+2] even	SAdr [n][7]	—	—
[4*n+2] odd	SAdr [n][8]	—	—
[4*n+3] even	SAdr [n][9]	—	—
[4*n+3] odd	SAdr [n][10]	—	—

Time
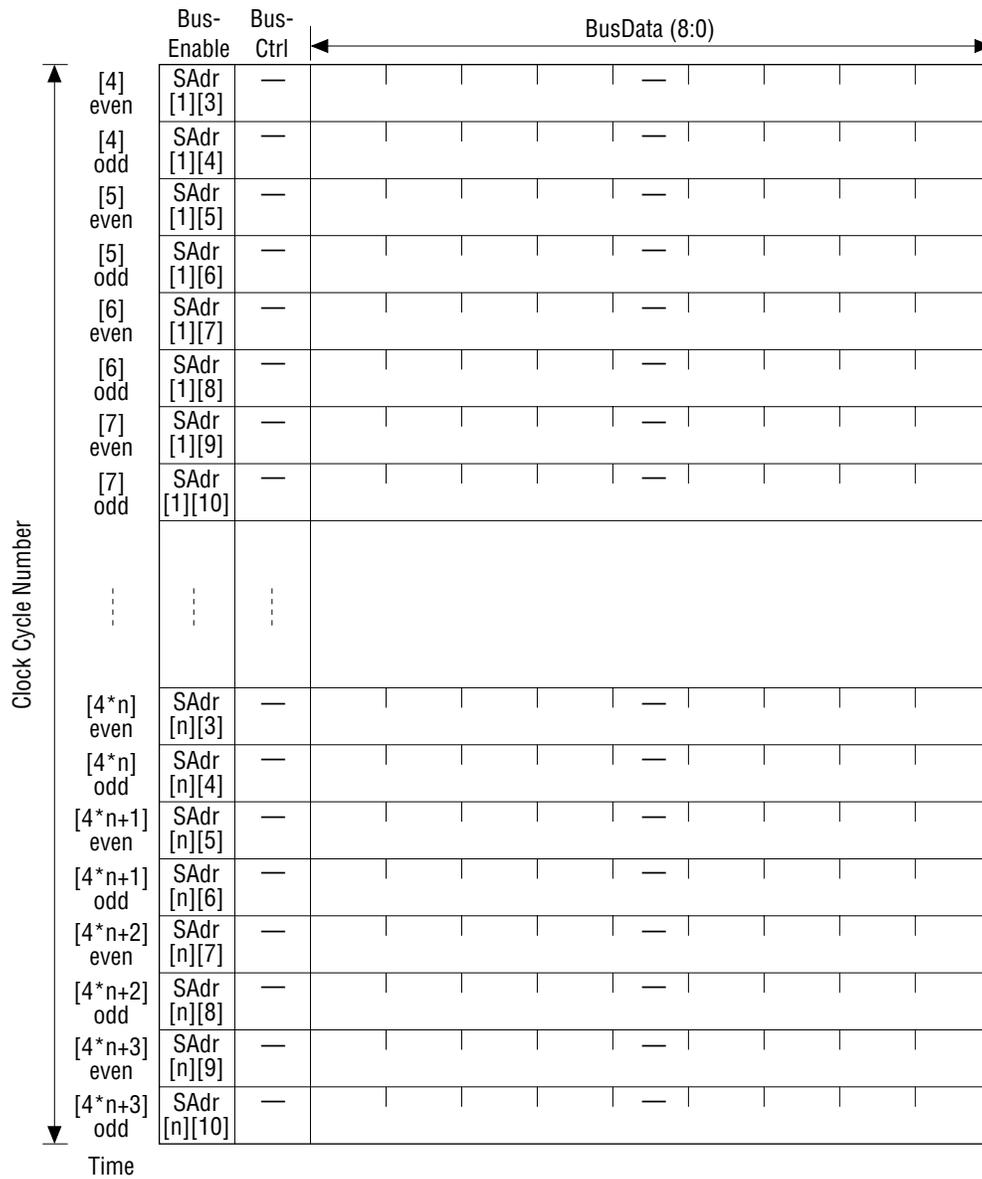
Fig. 9 Serial Address Packet Format[1]

Note: 1. A — in this diagram signifies that this pin is not used by this packet. If it is not used by another packet, it is pulled to a logic zero value.

Serial Control Packet

The protocol also allows the channel master to issue an early termination instruction for a memory read or write transaction. This is done using a Serial Control packet.
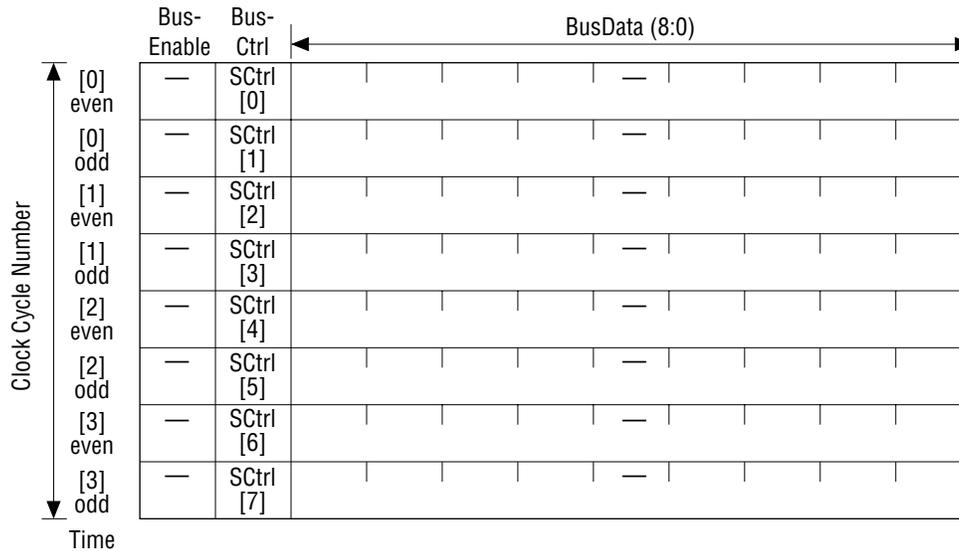
Fig. 10 Serial Control Packet Format[1]

Note: 1.   A ⎯ in this diagram signifies that this pin is not used by this packet. If it is not used by another packet, it is pulled to a logic zero value.

The value of the Serial Control field specifies whether the instruction should continue or terminate. This is shown in the table below.

Table 5 Serial Control Field

Serial Control Field	Description	Value
SCtrl[7:0]	Continue	00000000
SCtrl[7:0]	Terminate	00100000

Serial Mode Packet

Serial Mode Packets are used to instruct the RDRAM to perform a operating mode change. A Serial Mode Packet is simply a pulse on the BusEnable line.
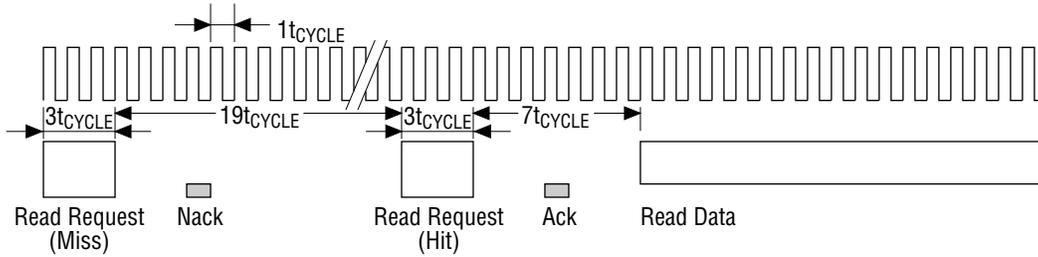
Fig. 11 Sample 32-Byte Read Miss and Read Hit Transactions
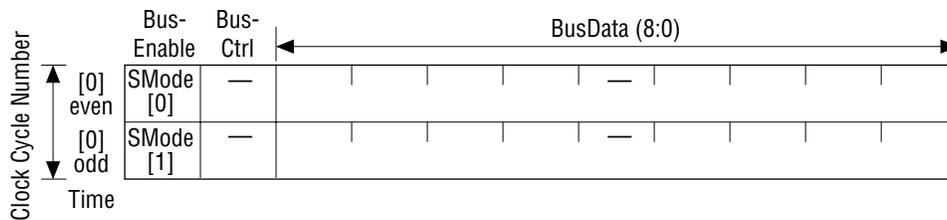
Fig. 12 Serial Mode Packet[1]

Note: 1.   A ☐ in this diagram signifies that this pin is not used by this packet. If it is not used by another packet, it is pulled to a logic zero value.

RDRAM OVERVIEW

The figure on the next page is a block diagram of the RDRAM device. The Rambus channel interface consists of a clock generator, a receiver, and a transmitter. The clock generator uses the external clock signals RxClk and TxClk (tapped off the channel traces ClockFromMaster and ClockToMaster) and creates the internal signals RClk and TClk. These are used by the receiver and transmitter, respectively, to transfer a bit every 2 nanoseconds on each wire between the RDRAM and the master device. The receiver and transmitter blocks also contain multiplexing and storage hardware to permit the internal RDRAM data paths to operate at the slower clock rate (but equivalent bandwidth) of eight bytes transferred every 15 or 16 nanoseconds (four clock cycles).

The RDRAM also contains control logic and configuration registers. The registers are read and written using special register space commands and control various aspects of RDRAM operation as described on page 19.

The remaining logic consists of a standard DRAM memory core and row sense amplifier caches.

ClockToMaster   **Rambus Channel**
ClockFromMaster

BusEnable, BusCtrl, BusData[8:0]

1 | SIn   SOut | 1

| | 11 | RxClk | 1 | 1 | TxClk | | 10 | |
| Receiver | | RClk ◄ | Clock Generator | ► TClk | | Transmitter | |

11   10

| Request, SMode, SAdr, SCtrl | | WriteData[7:0][8:0] | | Ack[1:0] | ReadData[7:0][8:0] |

37   25   8   72   72

Op[3:0]   Adr[10:3]   Internal Data Bus
OpX[1:0]
Adr[2:0]   Adr[35:11]
Count[7:0]
Start, Close   Address
SMode[1:0]   Mapping
SAdr[31:1][10:3]   Logic
SCtrl[7:0]

AddressSelect[3:0][8:0]

Device Id	DeviceId[3:0][8:0]
Compare	DeviceType[3:0][8:0]
	Delay[3:0][8:0]
Command and	Mode[3:0][8:0]
Control Logic	DeviceManufacture[3:0][8:0]
RowAddres	Row[3:0][8:0]
Compare	
CoreAddres	RefRow[3:0][8:0]
Compare	MinInterval[3:0][8:0]

RasInterval[3:0][8:0]

2   1   72
CoreAccess   CoreBankAddr
CoreRestore   9   MDReg[7:0][8:0]
                      CoreRowAddr   72   72
8
CoreColAddr   72

72   72

| 72×256 RowSenseAmpLatch | Column Decoder | 72×256 RowSenseAmpLatch |

72×256   72×256

| 72×256×512 DRAM Array - Bank 0 | Row Decoder | 72×256×512 DRAM Array - Bank 1 |

Fig. 13 RDRAM Block Diagram

14

RDRAM OPERATION

The RDRAM is composed of two independent banks of memory with each bank storing a full 1 Mbyte of data (see Fig. 13). Each of these banks has a 2KByte open page associated with it that is built out of sense amplifier arrays. These sense amplifiers hold the last accessed row of their associated bank in the sense amplifiers. This allows further accesses to the same row of memory to result in page hits. With the row already stored in the sense amplifiers, subsequent data can be accessed with very low latency. Each RDRAM added to a system adds two open pages to the memory system helping to increase hit rates.

A page miss results when a row is accessed that is not currently stored as one of the open pages. When this happens, the requesting master is sent a NACK Acknowledge packet indicating the requested row is not yet available. The RDRAM then loads the requested row into the sense amplifiers and waits for the master to submit a retry of the previous request. Fig. 11 shows an example of a read miss followed by a read hit for a 32 byte memory read operation.

The amount of time that is needed before the retry can be serviced depends on whether the data in the open page is clean or dirty. The sense amplifiers act as a "write back" cache in that data written to the open page is not written into the actual DRAM cells until the page is closed. If the data in an open page is clean (not previously written) when a new page is requested, the open page does not need to be written back into the DRAM. If the data in an open page is dirty, then additional time must be added to the miss retry delay to account for the writeback operation.

ADDRESS MAPPING

Address mapping hardware is provided to increase page hit rates by allowing system designers to easily perform n-way RDRAM interleaving. In a non-interleaved memory system, contiguous blocks of addresses follow each other in sequence in one RDRAM, which is then followed by the next RDRAM.

Using address mapping, adjacent blocks of data (2K or greater) can be separated across several RDRAMs, and therefore across several open pages. This allows a more optimal mapping of the pages as caches and creates higher effective page hit rates. In a typical system containing, for example eight RDRAMs, hit rates could be expected to be as high as 95%. Address mapping is easily adjusted by writing a control register in each RDRAM.

TRANSACTION CONCURRENCY

Concurrent transactions can be used to optimize RDRAM utilization in high performance applications by taking advantage of available channel bandwidth during page miss latency periods. When a miss in one RDRAM takes place, that device will be busy loading a new row into one of its sense amp caches. The channel and all other RDRAMs will still be available for use. While waiting for the first RDRAM to finish loading its open page, a transaction to another RDRAM can be initiated. In systems where memory accesses can be queued, a tansaction can take place for any pending access residing in a different RDRAM.

Pretouching can be used in systems where certain memory accesses are predictable, such as video applications. This is done when an application is finished with a particular RDRAM and about to access a different one. If the next access to an RDRAM is known in advance, a transaction can be first generated that will cause a row miss and prepare the RDRAM for its next access. When the device is next accessed, the required row of data will already be loaded in the open page and a page hit will take place.

15

RANDOM ACCESS MODE

Non-contiguous blocks of memory can be accessed through the use of the read and write non-sequential (Random Access) operations. With these commands multiple eight-byte blocks (octbytes) of data within an open page can be accessed in any order. To do this, the master device sends a request packet specifying a non-sequential operation along with the address of the first octbyte to be accessed. The master device also generates a serial address packet (see Fig. 9) on the BusEnable signal that specifies the address of the next octbyte. Successive serial address packets continue to specify new addresses within the open page while data is continuously transferred until the access is complete.

Random access mode can be used to satisfy the burst order of processors like the Intel™ Pentium™ at full RDRAM bandwidth.

BIT MASKING

Three forms of bit masking are available for memory write operations. These operations are referred to as data-per-bit (Dpb), mask-per-bit (Mpb), and both-per-bit (Bpb) masking. An eight-byte Mask Data register (MDReg) within the RDRAM is used to hold the static value of either mask or data information for these operations. The Mask Data register is an RDRAM internal register that is written by the Bpb commands, and is used by the Mpb and Dpb commands.

With the Dpb operation, the MDReg is used to hold a static mask that is applied to all octbytes of data written to the RDRAM core. With the Mpb operation, the MDReg is used to hold an octbyte of static data that is masked by dynamic bit masks supplied in the data packets before being written to the RDRAM core.

The Bpb operation requires data packets to alternate between mask and data octbytes. The even data packets (starting with data packet 0) carry bit masking information which is placed in the MDReg while the odd data packets carry the data to be masked by the latest contents on the MDReg. This type of operation is also used to set the MDReg for later use in Dpb and Mpb operations.

Table 6 shows the source of the mask and data for each of the write commands. The first eight columns show the Wseq and Wnsq sequential and non-sequential (fandom-access) write commands. Each has four bit mask sub-commands: Npb, Dpb, Mpb, and Bpb.

A write command consists of writing from one to 32 octbytes of data W[31:0]. Each octbyte of data is masked by an octbyte of bitmask Bit[31:0]; that is, if the bitmask bit is set, then the corresponding bit of write data is written. If the bitmask bit is clear, then the bit in memory is left unchanged.

Each of the first eight columns of Table 6 shows the source of the up-to-32 octbytes of write data and bitmask for the eight write commands. The Npb commands use no bitmask at all - effectively the bitmask is all ones. The Dpb command takes a single octbyte of bitmask from the MDReg and applies it to all data octbytes that are written. The Dpb command does the reverse and takes a single octbyte of write data from the MDReg and writes it to each octbyte of memory using a different bitmask from the data packet. The Bpb commands take an octbyte of bitmask and an octbyte of write data alternately, and writes them to a single octbyte of memory.

BYTE MASKING

Contiguous byte masking is supported by the WseqNpb command. This command uses the Adr[2:0] and Count[2:0] fields of the request packet (Fig. 6) to specify the byte masks of the first and last octbytes of a data packet. The RDRAM also supports a more general form of byte masking called non-contiguous byte masking. This is available with the Wbns write commands.

As can be seen in column nine of Table 6 for the WbnsNpb command, the first data packet, and every ninth thereafter, contain byte masking information that is applied to the eight data packets that follow. This means data packets 0, 9, 19, and 27 (gray boxes in the table) are not written to memory, but are instead used as byte masks for the eight octbytes of data that follow. This means that the data packet Data [35:0] may consist of up to 36 octbytes of information: 4 octbytes of byte mask information and 32 octbytes of data that is actually written to memory.

Each bit of the 64-bit byte mask (one octbyte) controls whether a byte of the following 64 bytes of data (eight octbytes) is written (one) or not written (zero) to memory.

Static bit masking is also available with non-contiguous byte masking. This is shown in the last two columns of Table 6 labeled WbnsDpb and WbnsMpb. The WbnsDpb command applies a static bit held in the MDReg to each octbyte of data W[31:0]. Each bit of the data octbyte is written if the corresponding bits of both the bitmask and bytemask are a one, and is not written if either is a zero. The WbnsMpb command applies a different bitmask Bit[31:0] to a static data octbyte held in the MDReg. Each bit of the data octbyte is written if the corresponding bits of both the bitmask and bytemask are a one, and is not written if either is a zero.

Figure 14 shows how the bits of each Data[i][[7:0][8:0] octbyte are flow through the wire circuitry as a function of the operation type. The data W[i][j][8:0] is written if the corresponding Bit[i][j][8:0] and Byte [i][j] mask bits are set.
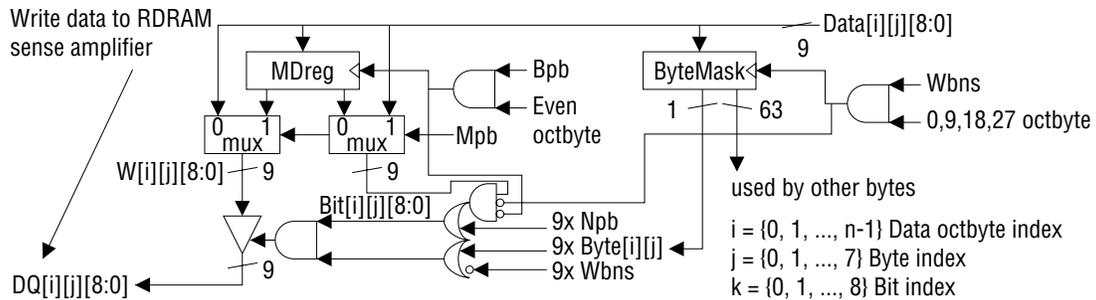

Fig. 14 MDReg with BitMask and ByteMask Logic - one byte slice

Table 6 WriteData, BitMask, and ByteMask Sources[2]

Data Octbyt	Wseq Npb	Wseq Dpb	Wseq Mpb	Wseq Bpb	Wnsq Npb	Wnsq Dpb	Wnsq Mpb	Wnsq Bpb	Wbns Npb	Wbns Dpb	Wbns Mpb
MDReg	—	Bit[35:0]	W[35:0]	—	—	Bit[35:0]	W[35:0]	—	—	Bit[35:0]	W[35:0]
Data[0]	W[0]	W[0]	Bit[0]	Bit[0]	W[0]	W[0]	Bit[0]	Bit[0]	ByteM[7:0]	ByteM[7:0]	ByteM[7:0]
Data[1]	W[1]	W[1]	Bit[1]	W[0]	W[1]	W[1]	Bit[1]	W[0]	W[0]	W[0]	Bit[0]
Data[2]	W[2]	W[2]	Bit[2]	Bit[1]	W[2]	W[2]	Bit[2]	Bit[1]	W[1]	W[1]	Bit[1]
Data[3]	W[3]	W[3]	Bit[3]	W[1]	W[3]	W[3]	Bit[3]	W[1]	W[2]	W[2]	Bit[2]
Data[4]	W[4]	W[4]	Bit[4]	Bit[2]	W[4]	W[4]	Bit[4]	Bit[2]	W[3]	W[3]	Bit[3]
Data[5]	W[5]	W[5]	Bit[5]	W[2]	W[5]	W[5]	Bit[5]	W[2]	W[4]	W[4]	Bit[4]
Data[6]	W[6]	W[6]	Bit[6]	Bit[3]	W[6]	W[6]	Bit[6]	Bit[3]	W[5]	W[5]	Bit[5]
Data[7]	W[7]	W[7]	Bit[7]	W[3]	W[7]	W[7]	Bit[7]	W[3]	W[6]	W[6]	Bit[6]
Data[8]	W[8]	W[8]	Bit[8]	Bit[4]	W[8]	W[8]	Bit[8]	Bit[4]	W[7]	W[7]	Bit[7]
Data[9]	W[9]	W[9]	Bit[9]	W[4]	W[9]	W[9]	Bit[9]	W[4]	ByteM[15:8]	ByteM[15:8]	ByteM[15:8]
Data[10]	W[10]	W[10]	Bit[10]	Bit[5]	W[10]	W[10]	Bit[10]	Bit[5]	W[8]	W[8]	Bit[8]
Data[11]	W[11]	W[11]	Bit[11]	W[5]	W[11]	W[11]	Bit[11]	W[5]	W[9]	W[9]	Bit[9]
Data[12]	W[12]	W[12]	Bit[12]	Bit[6]	W[12]	W[12]	Bit[12]	Bit[6]	W[10]	W[10]	Bit[10]
Data[13]	W[13]	W[13]	Bit[13]	W[6]	W[13]	W[13]	Bit[13]	W[6]	W[11]	W[11]	Bit[11]
Data[14]	W[14]	W[14]	Bit[14]	Bit[7]	W[14]	W[14]	Bit[14]	Bit[7]	W[12]	W[12]	Bit[12]
Data[15]	W[15]	W[15]	Bit[15]	W[7]	W[15]	W[15]	Bit[15]	W[7]	W[13]	W[13]	Bit[13]
Data[16]	W[16]	W[16]	Bit[16]	Bit[8]	W[16]	W[16]	Bit[16]	Bit[8]	W[14]	W[14]	Bit[14]
Data[17]	W[17]	W[17]	Bit[17]	W[8]	W[17]	W[17]	Bit[17]	W[8]	W[15]	W[15]	Bit[15]
Data[18]	W[18]	W[18]	Bit[18]	Bit[9]	W[18]	W[18]	Bit[18]	Bit[9]	ByteM[23:16]	ByteM[23:16]	ByteM[23:16]
Data[19]	W[19]	W[19]	Bit[19]	W[9]	W[19]	W[19]	Bit[19]	W[9]	W[16]	W[16]	Bit[16]
Data[20]	W[20]	W[20]	Bit[20]	Bit[10]	W[20]	W[20]	Bit[20]	Bit[10]	W[17]	W[17]	Bit[17]
Data[21]	W[21]	W[21]	Bit[21]	W[10]	W[21]	W[21]	Bit[21]	W[10]	W[18]	W[18]	Bit[18]
Data[22]	W[22]	W[22]	Bit[22]	Bit[11]	W[22]	W[22]	Bit[22]	Bit[11]	W[19]	W[19]	Bit[19]
Data[23]	W[23]	W[23]	Bit[23]	W[11]	W[23]	W[23]	Bit[23]	W[11]	W[20]	W[20]	Bit[20]
Data[24]	W[24]	W[24]	Bit[24]	Bit[12]	W[24]	W[24]	Bit[24]	Bit[12]	W[21]	W[21]	Bit[21]
Data[25]	W[25]	W[25]	Bit[25]	W[12]	W[25]	W[25]	Bit[25]	W[12]	W[22]	W[22]	Bit[22]
Data[26]	W[26]	W[26]	Bit[26]	Bit[13]	W[26]	W[26]	Bit[26]	Bit[13]	W[23]	W[23]	Bit[23]
Data[27]	W[27]	W[27]	Bit[27]	W[13]	W[27]	W[27]	Bit[27]	W[13]	ByteM[31:24]	ByteM[31:24]	ByteM[31:24]
Data[28]	W[28]	W[28]	Bit[28]	Bit[14]	W[28]	W[28]	Bit[28]	Bit[14]	W[24]	W[24]	Bit[24]
Data[29]	W[29]	W[29]	Bit[29]	W[14]	W[29]	W[29]	Bit[29]	W[14]	W[25]	W[25]	Bit[25]
Data[30]	W[30]	W[30]	Bit[30]	Bit[15]	W[30]	W[30]	Bit[30]	Bit[15]	W[26]	W[26]	Bit[26]
Data[31]	W[31]	W[31]	Bit[31]	W[15]	W[31]	W[31]	Bit[31]	W[15]	W[27]	W[27]	Bit[27]
Data[32]	—	—	—	—	—	—	—	—	W[28]	W[28]	Bit[28]
Data[33]	—	—	—	—	—	—	—	—	W[29]	W[29]	Bit[29]
Data[34]	—	—	—	—	—	—	—	—	W[30]	W[30]	Bit[30]
Data[35]	—	—	—	—	—	—	—	—	W[31]	W[31]	Bit[31]

Note: 2.   The shaded data packet contains byte masking information that is applied to the eight data packets that follow.

RDRAM REGISTERS

The 18M RDRAM contains ten registers. These are read and written with the Rreg, Wreg, and WregB commands. They are used to provide configuration information to the RDRAM controller (DeviceType, MinInterval, and DeviceManufacture), to control device, bank, and row addressing (DeviceId, AddressSelect, and Row), to control refresh (RefRow), to control RDRAM timing (Delay and RasInterval), and to control RDRAM operation (Mode). The following table summarizes these functions:

Table 7 Register Summary

| Register Name	Reg. #	Description
DeviceType[3:0][8:0]	0	Read-only register that defined the size and configuration of the RDRAM.
DeviceId[3:0][8:0]	1	Used to specify the base address for the RDRAM.
Delay[3:0][8:0]	2	Used to specify CAS timing parameters.
Mode[3:0][8:0]	3	Used to initialize the RDRAM and set the I_{OL} output current.
RefRow[3:0][8:0]	5	Used to specify the next row and bank of the RDRAM to be refreshed.
RasInterval[3:0][8:0]	6	Used to specify RAS timing parameters.
MinInterval[3:0][8:0]	7	Read-only register defining minimum timing parameters for CAS accesses.
AddressSelect[3:0][8:0]	8	Used to specify address bit swapping to maximize RDRAM cache hit rate.
DeviceManufacturer[3:0][8:0]	9	Read-only register containing a manufacturer code.
Row[3:0][8:0]	128	Used to specify the currently sensed row in each bank.

The following diagrams show the individual fields of the RDRAM registers. The color of a field denotes its usage: dark-gray is unimplemented, light-gray is read-only, and white is read-write. The arrow within each multi-bit field points from the least-significant bit to the most-significant bit. Bit and byte numbering use little-endien notation.

Fig. 15 Registers

DeviceType Register - 0

Read-only register that defines the size and configuration of an RDRAM.

Bns (Bonus) - 9 (1) bit bytes
ColumnBits - 1011 (number of column addr bits)[1]

RowBits - 1001 (number of row addr bits)[2]
BankBits - 0001 (number of bank addr bits)[3]

Type - 0000 (RDRAM device)
Version - 0001 (extended architecture)
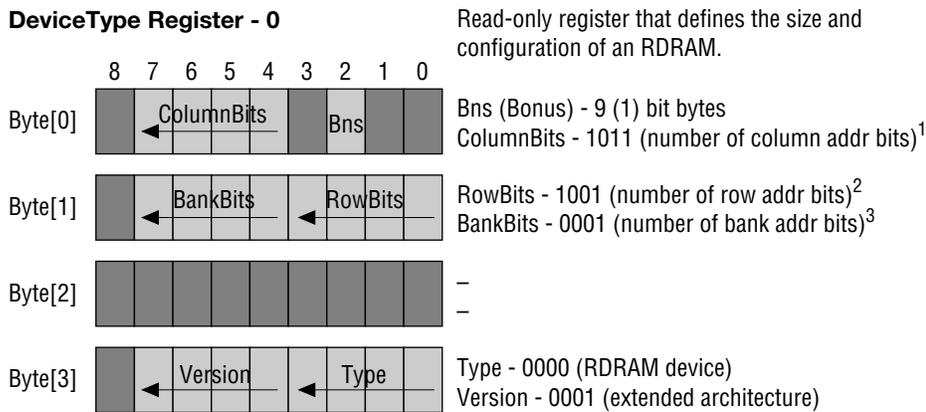
1. This value specifies the number of bytes per row ($1011=11_{10}=2^{11}$=2048 bytes). This is the address range over which accesses may be made without causing a row miss.
2. This value specifies the number of rows per bank ($1001=9_{10}=2^9$=512 rows).
3. This value specifies the number of banks per RDRAM ($1=1_{10}=2^1$=2 banks).

Deviceld Register - 1

Read-write register for setting the base address of the RDRAM.

IdField [35:21] - Compared to Adr[35:21] of request packet to select RDRAM.
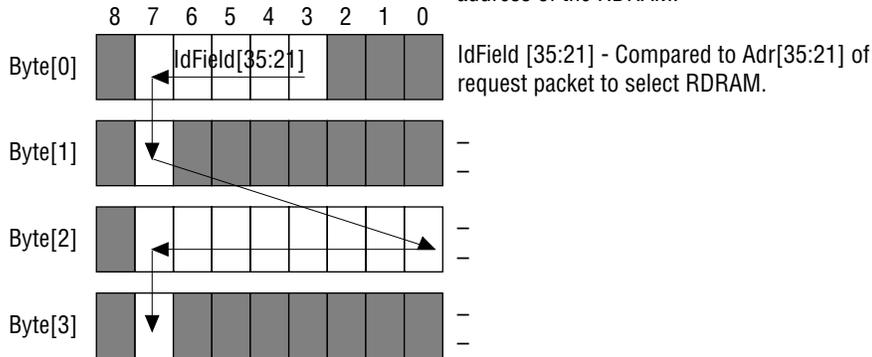
Delay Register -2

Read-write and read-only register to program the CAS access delays.

AckWinBits - 011 (Number of AckWinDelay bits)
AckWinDelay - Normally set to 101 (5 busclocks)

ReadBits - 011 (Number of ReadDelay bits)
ReadDelay - Normally set to 111 (7 busclocks)

AckBits - 010 (Number of AckDelay bits)
AckDelay - Normally set to 011 (3 busclocks)

WriteBits - 011 (Number of WriteDelay bits)
WriteDelay - Normally set to 001 (1 busclocks)
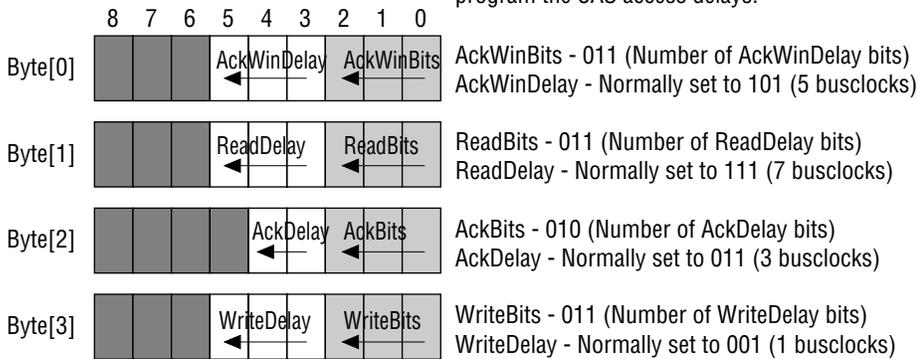
AckWinDelay adjusts the size of the acknowledge window. ReadDelay, WriteDelay, and AckDelay adjust the time from the end of the packet to the start of read data, write data, and the acknowledge packets, respectively.

Mode Register - 3

Read-write register for initializing the RDRAM and for controlling operating modes.

DE (DevEn) - Selects RDRAM at initialization
AS - Set to one

PL (PwrLng) - Selects powerdown wake-up time
X2 (CCMult) - Set to one.

CE (CCEnable) - Set to one.

C[5:0] (CCValue) - Specifies I_{OL} • 63 = minimum
0 = maximum
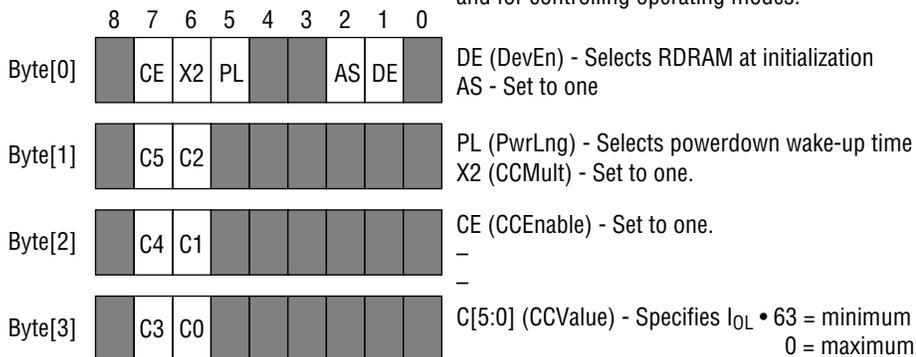
RefRow Register - 5

Read-write register for setting the next row refreshed by SetRR burst refresh.

	8	7	6	5	4	3	2	1	0
Byte[0]				RowField					
Byte[1]					Bank Field				
Byte[2]									
Byte[3]									

RowField - Next row to be refreshed by SetRR
–

BankField - Next bank to be refreshed by SetRR
–

–
–

–
–
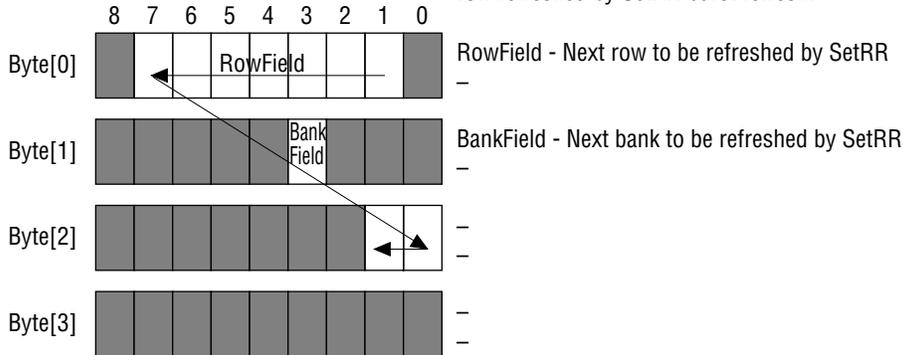
RasInterval Register - 6

Read-write register to program the RAS access intervals. Note that fields are in bit-reversed order.

	8	7	6	5	4	3	2	1	0
Byte[0]					RowPrecharge				
Byte[1]					RowSense				
Byte[2]					RowImpRestore				
Byte[3]					RowExpRestore				

RowPrecharge - Set the precharge interval
(Program to 01000)

RowSense - Set the sense interval
(Program to 01100)

RowImpRestore - Set the implicit restore interval
(Program to 10010)

RowExpRestore - Set the explicit restore interval
(Program to 00100)
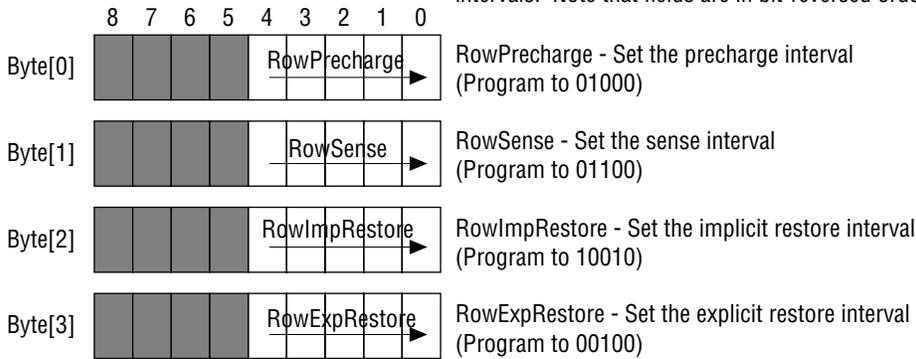
MinInterval Register - 7

Read-only register (configuration info) and write-only register (special control).

	8	7	6	5	4	3	2	1	0
Byte[0]									
Byte[1]		MinAckDly	MinReadDly	MinWriteDly					
Byte[2]									
Byte[3]					SpecFunc				

MinWriteDelay - 0001
(minimum WriteDelay of RDRAM)

MinReadDelay - 0111
(minimum ReadDelay of RDRAM)

MinAckDelay - 0011
(minimum AckDelay of RDRAM)

SpecFunc - Performs SetRR burst refresh and SetPD powerdown entry
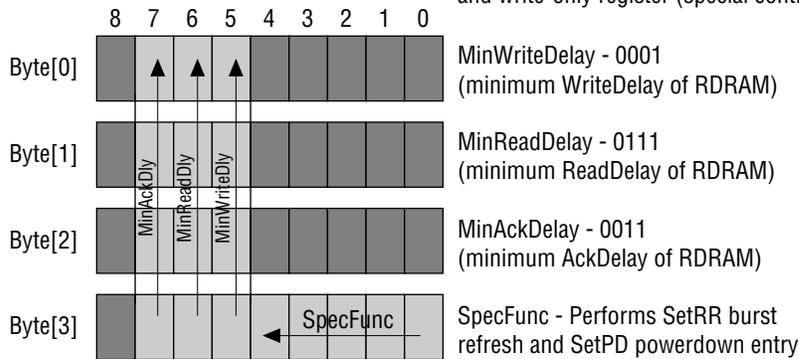
MinWriteDelay, MinReadDelay, and MinAckDelay specify the minimum number of cycles allowed between a request packet and a write data, read data, and acknowledge packet, respectively. SpecFunc is a write-only field that is used to initiate a SetRR burst refresh or to place the RDRAM into powerdown mode.

AddressSelect Register - 8

	8	7	6	5	4	3	2	1	0
Byte[0]				SwapField					
Byte[1]									
Byte[2]									
Byte[3]									

Read-write register for swapping subfields of the Adr field of request packet. This maximizes the row hit rate for many applications.
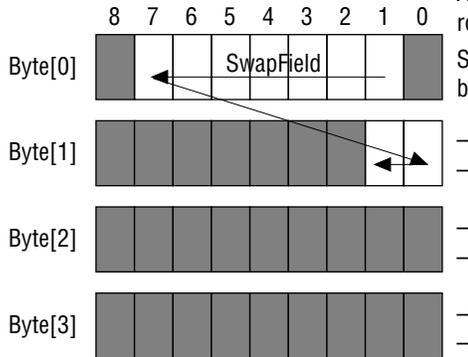
SwapField - Each bit swaps a pair of bits from Adr[28:20] and Adr[19:11]

DeviceManufacture Register - 9

	8	7	6	5	4	3	2	1	0
Byte[0]		ManufactureCode							
Byte[1]		ManufactureCode							
Byte[2]		Manufacturer							
Byte[3]		Manufacturer							

Read-only register with configuration information for RDRAM.
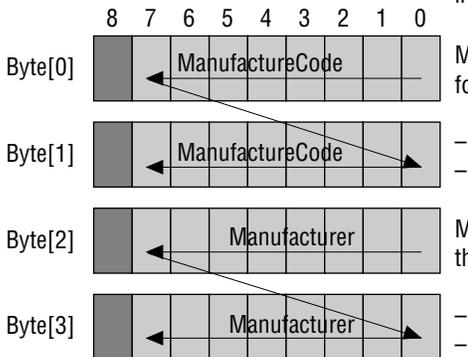
ManufactureCode - See RDRAM Design Guide for more information.

Manufacturer - Contains code specifying the manufacturing company.

Row Register - 128

	8	7	6	5	4	3	2	1	0
Byte[0]			SensedRow[0]						
Byte[1]									
Byte[2]			SensedRow[1]						
Byte[3]									

Read-write register with address of currently-sensed row in each bank of RDRAM.
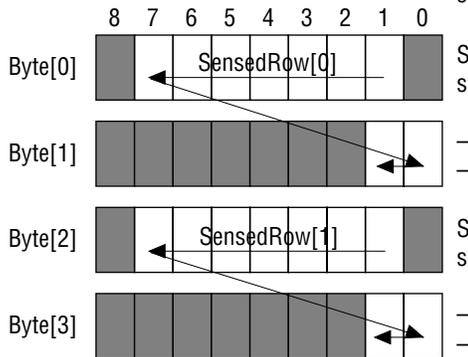
SensedRow[0] - Address of currently sensed row of bank 0.

SensedRow[1] - Address of currently sensed row of bank 1.

REFRESH

The RDRAM is a dynamic device, and the memory array must be refreshed every 17 ms (t_{REF}). The RDRAM includes all of the logic necessary to support three refresh modes to support this need. These refresh modes are:

• Manual Refresh:
  The Rambus channel master uses a register write transaction (SetRR) to initiate a single burst refresh of four rows.

• Touching:
  A single row is refreshed each time that a read or write request is made to that row in the RDRAM.

• PowerDown Mode Refresh:
  A single row is refreshed with each pulse on the SIn/SOut pins.

When the RDRAM is in PowerDown mode, it can be refreshed by passing a periodic pulse at a frequency of 60.2 kHz or greater through the SIn/SOut pins. This minimum frequency is equal to $1024/t_{REF}$.

OPERATING MODES

The RDRAM has three operating modes; Active, Standby and PowerDown. The three modes are distinguished by two factors, their power consumption, and the time that it will take the RDRAM to service a request from that mode.

The control logic within the RDRAM includes a counter that counts Serial Mode packets. It takes a specific number of packets to cause the RDRAM to transition from a low-power mode to the Active state. This counter is active in all three operating modes.

In Active mode, the RDRAM is active and ready to immediately service a request packet. Power consumption is also highest in Active mode.

An RDRAM automatically transitions to Standby mode at the end of a transaction. While in this low power state, each RDRAM monitors the BusEnable signal for a serial mode packet while ignoring other activity on the remaining channel signals. The channel master sends a serial mode packet to bring all RDRAMs temporarily out of Standby and into Active mode so they can respond to a request packet. Once the request packet is acknowledged, all of the RDRAMs return to Standby mode with the exception of the one responding to the request. That device returns to Standby mode once the read or write operation is complete.

Unlike conventional DRAM memory systems where each device in an entire bank of memory must be kept active and consumes power through an entire access, Rambus memory systems use only one active device while all others remain in a lower power state.

Power consumption may be greatly reduced by using the PowerDown mode. This mode is entered manually by setting the Special Function bit *SetPD* in the MinInterval register. Entering this mode causes the device to write back and precharge both cache lines, disable the internal clock generator, and disable most DC current sources. The BusEnable receiver is kept active to detect serial mode packets used to exit powerdown mode. The only significant power consumption in powerdown mode is due to refresh.

Since the RDRAM's internal clocks are disabled while in powerdown mode, refresh must be maintained manually by the master device. This is done by supplying a low frequency square wave on the SIn TTL signal. This propagates through each RDRAM and is used to initiate asynchronous refresh operations in each device.

Each RDRAM may be placed in either low or high threshold powerdown mode. Threshold refers to the number of serial mode packets required to wake up the RDRAM. A low threshold requires relatively few serial mode packets while a high threshold requires a larger number. The actual power dissipation is identical in both modes.

An example of where these modes are used is in a portable computer application, as shown below. The sleep mode is implemented by placing a majority of the RDRAMs in high threshold powerdown while the RDRAM that contains the frame buffer is placed in low threshold powerdown. This permits screen refresh to take place without powering up the entire memory system.
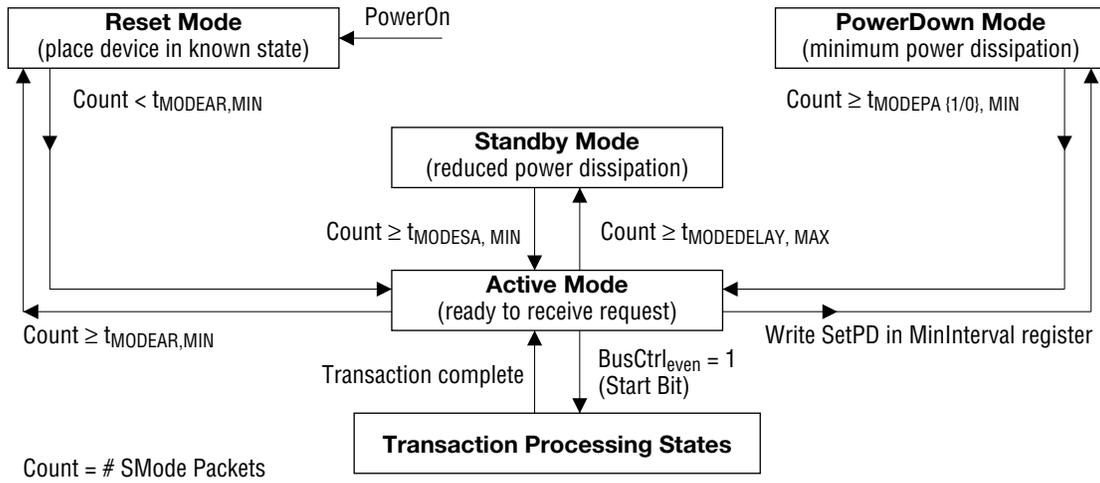
Fig. 16 RDRAM Operating Modes

INITIALIZATION

The channel master resets the RDRAM devices on the channel by asserting the BusEnable signal for $T_{MODEAR,\ MIN}$ clock cycles. After the RDRAM has been reset, the base address and register space registers revert to their default values. Because the default address of all devices is zero until initialization is complete, individual devices cannot be addressed from the channel, although the devices can and will respond to broadcast commands.

In a typical system application, the SIn pin of the first RDRAM is tied to VDD (refer to Figure 2). The SIn pin of the next RDRAM is connected to the SOut pin of the first RDRAM, and so on. SOut of the last device is left unconnected. An RDRAM will not respond to a write command (other than a broadcast write) until its SIn pin is set to 1. Note that if PowerDown mode is to be used by the application, it must be possible to connect a 60.2 kHz pulse source to SIn of the first RDRAM to provide refresh.

To start the initialization sequence, The RAC cell in the controller is reset, its DLL is allowed to lock, and its current control register is loaded. Next, the RDRAMs are put into Reset state by asserting the BusEnable wire for $t_{MODEAR,\ MIN}$ cycles. The $t_{LOCK,\ RESET}$ interval is observed to allow the RDRAM DLLs to lock.

After the RDRAMs have been placed in Reset state, broadcast writes are made to all control registers needing values different from their Reset values. Next, SIn is asserted high on the first RDRAM in the chain. This enables it. The channel master then writes the desired device address to the DeviceID register and sets the *DevEn* bit. This asserts the SOut pin (and the SIn pin on the next device in the chain) to 1. Before proceeding to the next RDRAM, these additional steps are taken on the current RDRAM:

1. Current control calibration. The value written into the CCValue field of the Mode register is finetuned to maximize signal margin. This calibration process must take place before the controller performs any register or memory reads or any acknowledge responses.

2. Check read-only fields of control registers. This confirms which type of Base RDRAMs are present. This also provides an indication of when the end of the channel is reached.

3. Set the RasInterval register fields.

4. Touch the RDRAM with eight successive memory read transactions. This settles timing circuitry.

This process continues until all of the RDRAMs have been initialized. When these steps have been completed for every device in the chain, all of the RDRAM devices will have unique, contiguous DeviceID values, and will have *DevEn* bits set.

ABSOLUTE MAXIMUM RATINGS

The following table represents stress ratings only, and functional operation at the maximum ratings is not guaranteed. Extended exposure to the maximum ratings may affect device reliability. Although these devices contain protective circuitry to resist damage from static electric discharge, always take precautions to avoid high static voltages or electric fields.

Symbol	Parameter	Min.	Max.	Unit
$V_{I,ABS}$	Voltage applied to any RSL pin with respect to Gnd	−0.5	$V_{DD,MAX}$+0.5	V
$V_{I,TTL,ABS}$	Voltage applied to any TTL pin with respect to Gnd	−0.5	V_{DD}+0.5	V
$V_{DD,ABS}$	Voltage on V_{DD} with respect to Gnd	−0.5	$V_{DD,MAX}$+1.0	V
$T_{J,ABS}$	Junction temperature under bias	−55	125	°C
T_{STORE}	Storage temperature	−55	125	°C

THERMAL PARAMETERS

Symbol	Parameter and Conditions	Min.	Max.	Unit
T_J	Junction operating temperature	0	100	°C
T_C	Package surface temperature	—	90	°C
Θ_{JC}	Junction-to-Case thermal resistance	—	5	°C/Watt

CAPACITANCE

Symbol	Parameter and Conditions	Min.	Max.	Unit
C_I	Low-swing input parasitic capacitance	—	2	pF
$C_{I,TTL}$	TTL input parasitic capacitance	—	8	pF

POWER CONSUMPTION

(Ta = 0°C to 70°C)

Mode	Description	Min.	Max.	Unit
Powerdown	Device shut down	—	20[a]/20[b]/30[c]	mW
Standby	Device inactive	—	330[a]/340[b]/360[c]	mW
Active	Device evaluating request packet	—	0.9[a]/0.95[b]/1.0[c]	W
Read	Data being transferred from device (@Burst Length = 256)	—	1.4[a]/1.5[b]/1.6[c]	W
Write	Data being transferred to device (@Burst Length = 256)	—	1.3[a]/1.4[b]/1.5[c]	W

Notes:
a. 500 MHz rank
b. 533 MHz rank
c. 600 MHz rank

RECOMMENDED ELECTRICAL CONDITIONS

(Ta = 0°C to 70°C)

Symbol	Parameter and Conditions	Min.	Max.	Unit
V_{DD}, V_{DDA}	Supply voltage	3.15	3.45	V
V_{REF}	Reference voltage	1.9	V_{DD}–0.8	V
V_{IL}	Input low voltage	V_{REF}–0.8	V_{REF}–0.35	V
V_{IH}	Input high voltage	V_{REF}+0.35	V_{REF}+0.8	V
$V_{IL,TTL}$	TTL input low voltage	−0.5	0.8	V
$V_{IH,TTL}$	TTL input high voltage	1.8	V_{DD}+0.5	V

ELECTRICAL CHARACTERISTICS

(Ta = 0°C to 70°C)

Symbol	Parameter and Conditions	Min.	Max.	Unit
I_{REF}	V_{REF} CURRENT @ $V_{REF,MAX}$	−10	10	μA
I_{OH}	Output high current @ ($0 \leq V_{OUT} \leq V_{DD}$)	−10	10	μA
$I_{OL,0}$	Output low current @ V_{OUT}=1.6 V @ C[5:0]=111111 (63_{10})[a]	0.0	4.0	mA
$I_{OL,20}$	Output low current @ V_{OUT}=1.6 V @ C[5:0]=110001 (49_{10})[a]	18.0	22.0	mA
$I_{OL,40}$	Output low current @ V_{OUT}=1.6 V @ C[5:0]=011111 (31_{10})[a]	36.0	44.0	mA
$I_{I,TTL}$	TTL input leakage current @ ($0 \leq V_{I,TTL} \leq V_{DD}$)	−10.0	10.0	μA
$V_{OL,TTL}$	TTL output voltage @ $I_{OL,TTL}$=1.0 mA	0.0	0.4	V
$V_{OH,TTL}$	TTL output high voltage @ $I_{OH,TTL}$= −0.25 mA	2.0	V_{DD}	V

Note:

a. This is the value written into the C[5:0] field of the Mode register. Values of I_{OL} in between the $I_{OL,0}$, $I_{OL,20}$, and $I_{OL,40}$ values are produced by interpolating C[5:0] to intermediate values. For example, C[5:0] = 101000 (40_{10}) produces an I_{OL} in the range of 27.0 to 33.0 mA.

RECOMMENDED TIMING CONDITIONS

(Ta = 0°C to 70°C)

Symbol	Parameter	Min.	Max.	Unit
t_{CR}, t_{CF}	TxClk and RxClk input rise and fall times	0.3	0.8	ns
t_{CYCLE}	TxClk and RxClk cycle times	3.33/3.75/4.0	4.5	ns
t_{TICK}	Transport time per bit per pin (this timing interval is synthesized by the RDRAM's internal clock generator)	0.5 (2 ns @ t_{CYCLE} = 4ns)	0.5 (2.25 ns @ t_{CYCLE} = 4.5ns)	t_{CYCLE}
t_{CH}, t_{CL}	TxClk and RxClk high and low times	45%	55%	t_{CYCLE}
t_{TR}	TxClk-RxClk differential	0	0.7	t_{CYCLE}
t_{DR}, t_{DF}	Data/Control input rise and fall times	0.3	0.6	ns
t_{QR}, t_{QF}	Data/Control output rise and fall times	0.3	0.5	ns
t_S	Data/Control-to-RxClk setup time	0.35	—	ns
t_H	RxClk-to-Data/Control hold time	0.35	—	ns
t_{REF}	Refresh interval	—	17	ms
$t_{LOCK,RESET}$	RDRAM internal clock generator lock time from Reset mode	—	750 (3 μs @ t_{CYCLE} = 4ns)	t_{CYCLE}
$t_{LOCK,POWERUP}$	RDRAM internal clock generator lock time from PowerUp mode	—	750 (3 μs @ t_{CYCLE} = 4ns)	t_{CYCLE}

TIMING CHARACTERISTICS

(Ta = 0°C to 70°C)

Symbol	Parameter	Min.	Max.	Unit
t_{PIS}	SIn-to-SOut propagation delay @ $C_{LOAD,TTL}$ = 40 pF	1	25	ns
t_Q	TClk-to-Data/Control output time	$\dfrac{t_{CYCLE}}{4} - 0.4$	$\dfrac{t_{CYCLE}}{4} + 0.4$	ns

RAMBUS CHANNEL TIMING

The next table shows important timings on the Rambus channel for common operations. All timings are from the point of view of the channel master, and thus have the bus overhead delay of 4 ns per bus transversal included where appropriate.

(Ta = 0°C to 70°C)

Symbol	Parameter	Min.	Max.	Unit
$t_{RESPONSE}$	Start of request packet to start of acknowledge packet	6[a]	9[a]	t_{CYCLE}
$t_{READHIT}$	Start of request packet to start of read data packet for row hit (Okay)	10[a]	17[a]	t_{CYCLE}
$t_{WRITEHIT}$	Start of request packet to start of write data packet for row hit (Okay)	4[a]	11[a]	t_{CYCLE}
$t_{RETRYSENSED}$ CLEAN (no restore)	Start of request packet for row miss (Nack) to start of request packet for row hit (Okay). The previous row had not been written	22[b]	—	t_{CYCLE}
$t_{RETRYSENSED}$ DIRTY (restore)	Start of request packet for row miss (Nack) to start of request packet for row hit (Okay). The previous row had been written	30[b]	—	t_{CYCLE}
$t_{RETRYREFRESH}$ CLEAN (no restore)	Start of request that performs a burst refresh (SetRR) until the start of a request that will not have a Nack acknowledge due to the pending refresh. The previously sensed row had not been written	209	—	t_{CYCLE}
$t_{RETRYREFRESH}$ DIRTY (restore)	Start of request that performs a burst refresh (SetRR) until the start of a request that will not have a Nack acknowledge due to the pending refresh. The previously sensed row had been written	217	—	t_{CYCLE}

| Symbol	Parameter	Min.	Max.	Unit
$t_{READBURST32}$	Start of request packet to end of 32 byte read data packet for row hit (Okay)	26^c	—	t_{CYCLE}
$t_{READBURST256}$	Start of request packet to end of 256 byte read data packet for row hit (Okay)	138	—	t_{CYCLE}
$t_{WRITEBURST32}$	Start of request packet to end of 32 byte write data packet for row hit (Okay)	20^d	—	t_{CYCLE}
$t_{WRITEBURST256}$	Start of request packet to end of 256 byte write data packet for row hit (Okay)	132	—	t_{CYCLE}
$t_{READDELAY}$	End of request packet to beginning of read data packet	7^a	14	t_{CYCLE}
$t_{WRITEDELAY}$	End of request packet to beginning of write data packet	1^a	8	t_{CYCLE}
$t_{ACKDELAY}$	End of request packet to beginning of acknowledge packet	3^a	6	t_{CYCLE}
$t_{ACKWINDELAY}$	Window in which an acknowledge packet will be sent	5^a	12	t_{CYCLE}
$t_{SERIALREADOFFSET}$	Delay from the beginning of a serial address packet or serial control packet to the beginning of the corresponding read data subpacket	13	13	t_{CYCLE}
$t_{SERIALWRITEOFFSET}$	Delay from the beginning of a serial address packet or serial control packet to the beginning of the corresponding write data subpacket	5	5	t_{CYCLE}
$t_{POSTMEMWRITEDELAY}$	Delay from the end of the current memory space transaction to the beginning of the next memory space transaction	2	—	t_{CYCLE}
$t_{POSTREGWRITEDELAY}$	Delay from the end of the current register space transaction to the beginning of the next register space transaction	4	—	t_{CYCLE}
$t_{MODEOFFSET}$	Offset from the beginning of SMode packet to request packet for standby to active transaction	4	4	t_{CYCLE}
t_{MODESA}	Number of SMode packets to cause a transition from StandbyMode to ActiveMode	1	4	t_{CYCLE}
$t_{MODEPA[0]}$	Number of SMode packets to cause a transition from PowerDownMode[0] to ActiveMode	20	—	t_{CYCLE}
$t_{MODEPA[1]}$	Number of SMode packets to cause a transition from PowerDownMode[1] to ActiveMode	176	—	t_{CYCLE}
t_{MODEAR}	Number of SMode packets necessary to cause a transition from ActiveState to ResetState	254	—	t_{CYCLE}
$t_{MODEDELAY}$	Delay time after a transaction is complete for the RDRAM to enter the StandbyState	—	20	t_{CYCLE}
$t_{INTERREQUEST}$	Offset from the beginning of the request of the current transaction to the beginning of the request packet of the next transaction.	6	—	t_{CYCLE}
$t_{RAS, MAX}$	Time that a row may remained sensed within a bank.	—	125	μs

Notes:

a. Programmable - All RDRAMS will operate across the full programming range.

b. Minimum at $t_{CYCLE,MIN}$. The delay is programmable to give equivalent timings at longer t_{CYCLE}.

c. Calculated with $t_{READHIT,MIN}$

d. Calculated with $t_{WRITEHIT,MIN}$

RISE/FALL TIMING
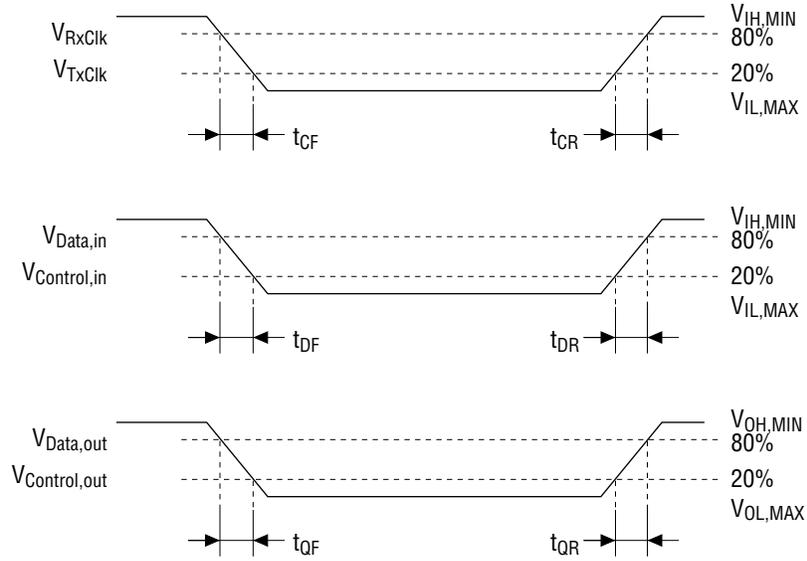
Where:
$V_{OH,MIN} = V_{TERM,MIN}$
$V_{OL,MAX} = V_{TERM} - Z_O{}^* (I_{OL,MAX}{}^*$ programmed value)

CLOCK TIMING
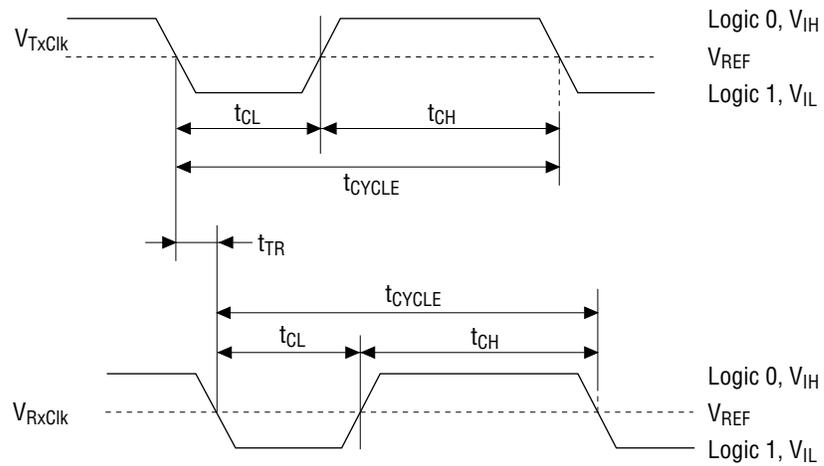
RECEIVE DATA TIMING
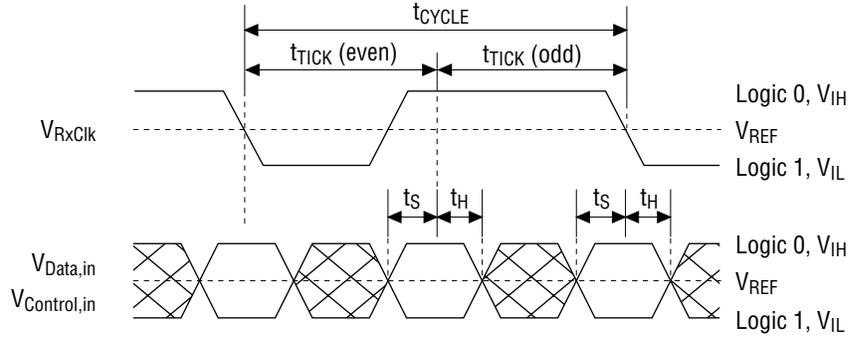
* t_{TICK} is defined as one-half t_{CYCLE}.

TRANSMIT DATA TIMING
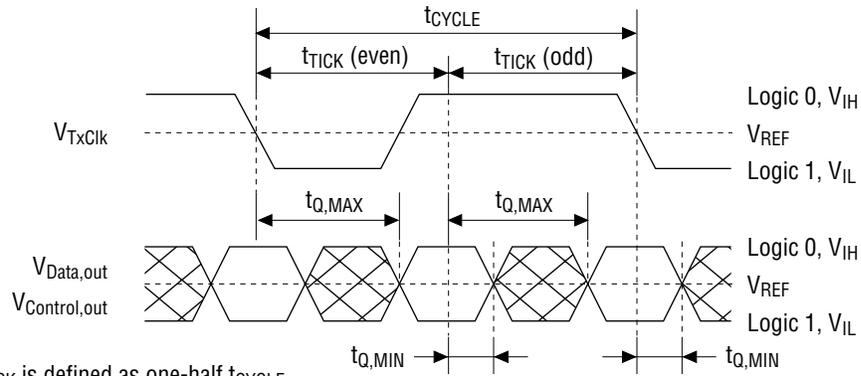

* t_{TICK} is defined as one-half t_{CYCLE}.

SERIAL CONFIGURATION PIN TIMING

$V_{SW,TTL}$=1.5 V
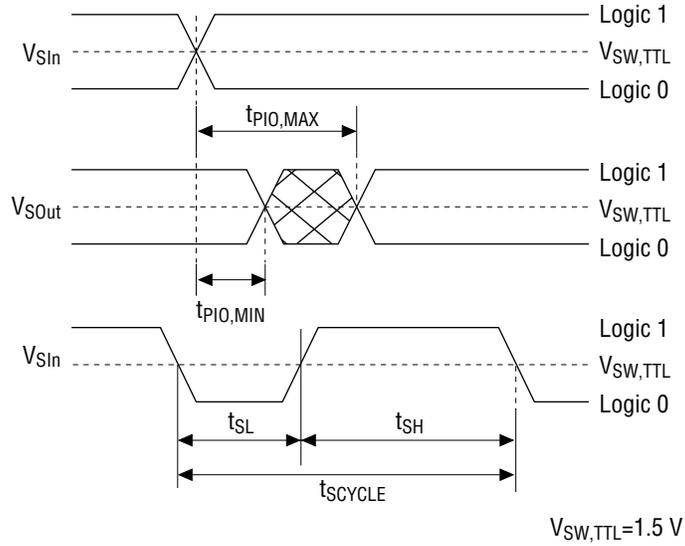
STANDBY MODE TO ACTIVE MODE TIMING
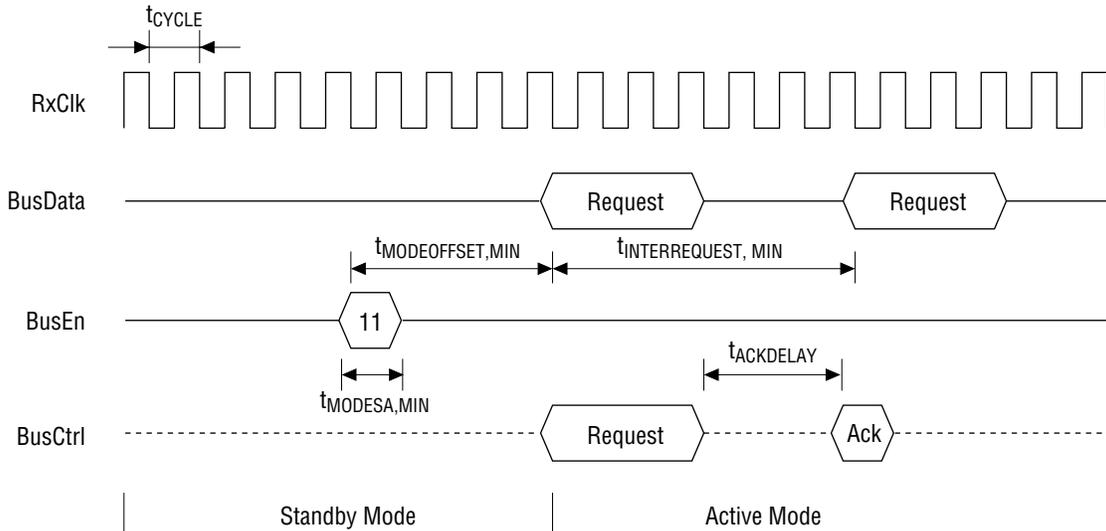
RESET TIMING
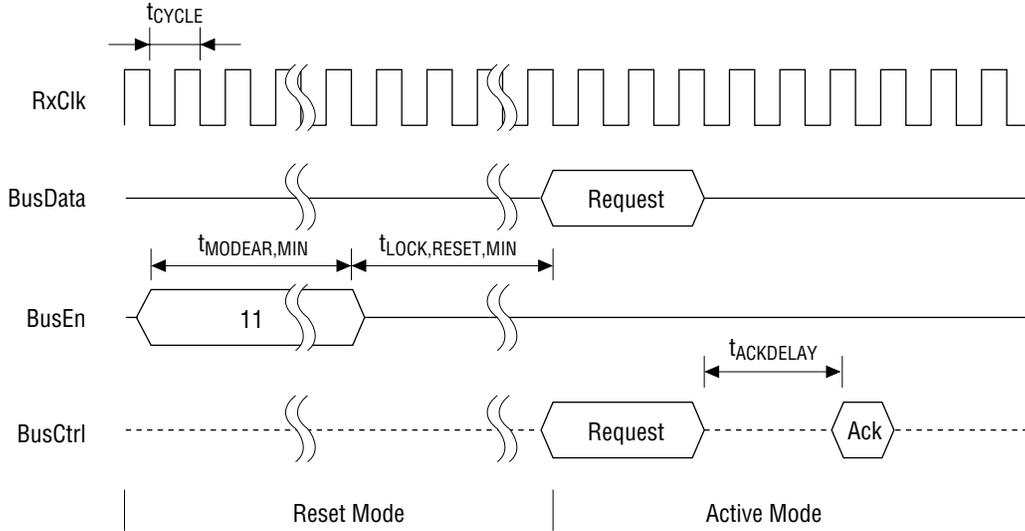
READ HIT TIMING DIAGRAM

WRITE HIT TIMING DIAGRAM

t_{CYCLE}

RxClk

$t_{WRITEHIT}$

BusData: Request | Data(0) | Data(n-1) | Data(n)

$t_{WRITEDELAY}$

$t_{SERIALWRITEOFFSET}$

BusEn: SAdr(1) | SAdr(2) | SAdr(n)

$t_{ACKDELAY}$

$t_{RESPONSE}$

BusCtrl: Ack(1:0) | SCtrl(n+1)

READ MISS TIMING DIAGRAM

t_{CYCLE}

RxClk

BusData: Request | Request | Data(0)

$t_{READHIT}$

BusEn

$t_{RESPONSE}$ $t_{RESPONSE}$

BusCtrl: Ack(1:0) = Nack   Ack(1:0) = Ack

$t_{RETRYSENSEDCLEAN}$ or $t_{RETRYSENSEDDIRTY}$
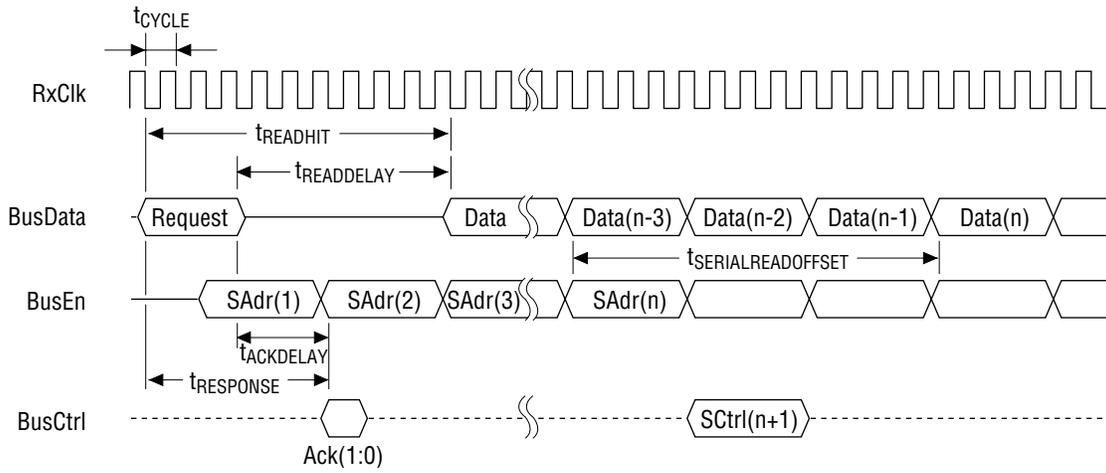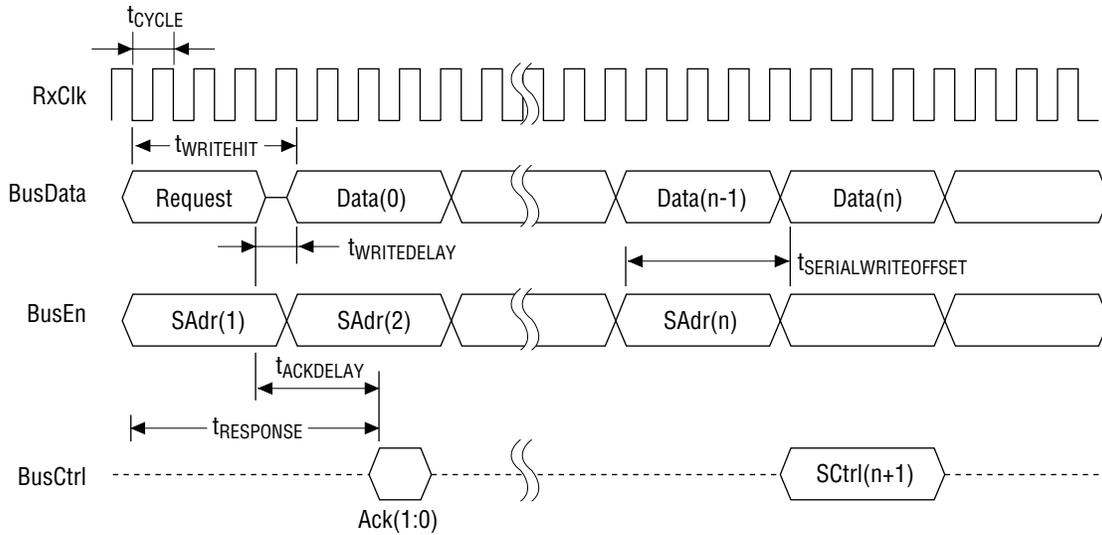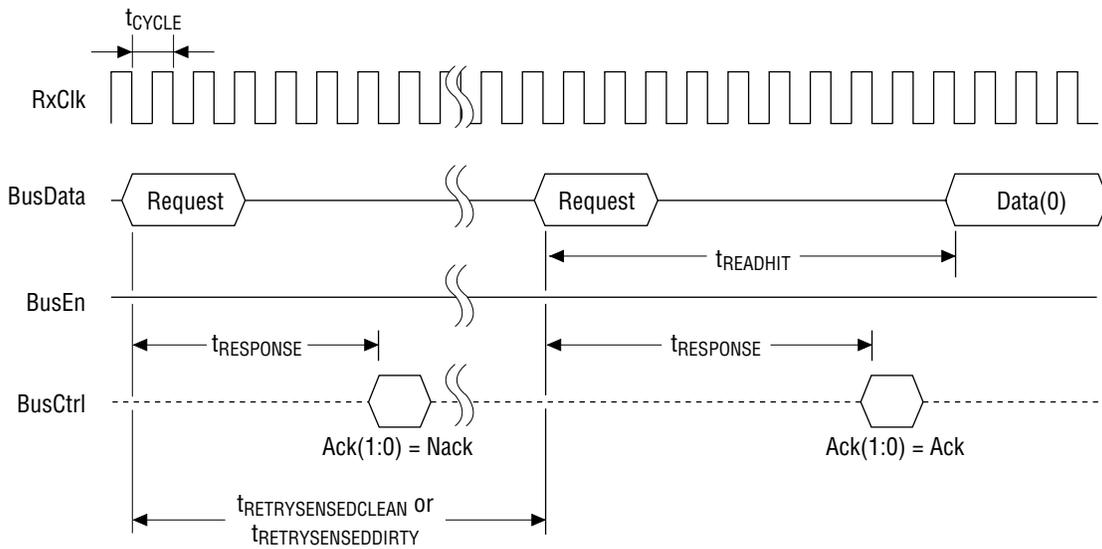
WRITE MISS TIMING DIAGRAM
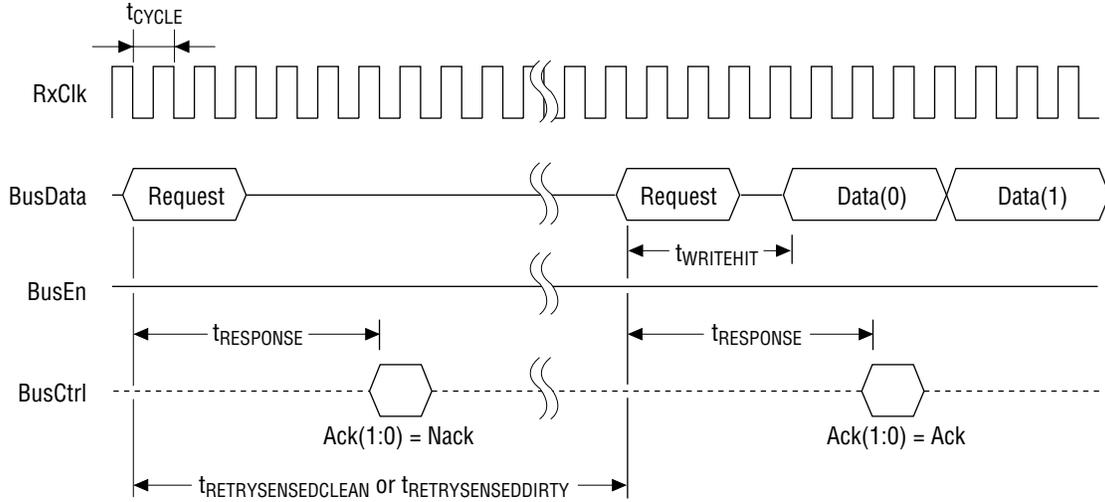
REGISTER READ TIMING DIAGRAM
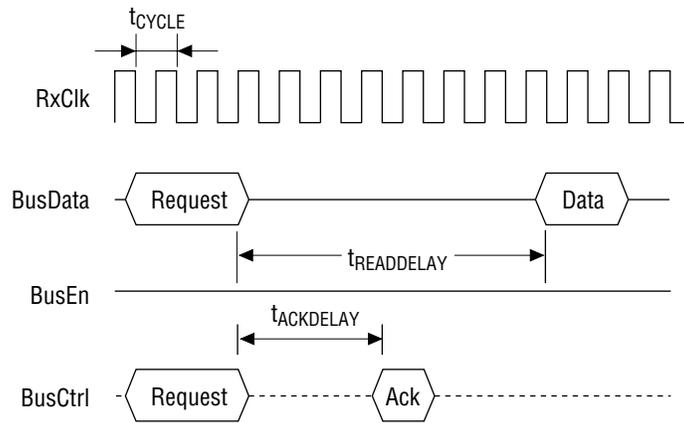
REGISTER WRITE TIMING DIAGRAM

t_{CYCLE}

RxClk

$t_{WRITEDELAY}$

BusData — Request — Data — Request

BusEn

$t_{ACKDELAY}$ $t_{POSTREGWRITEDELAY}$

BusCtrl — Request — Ack — Request
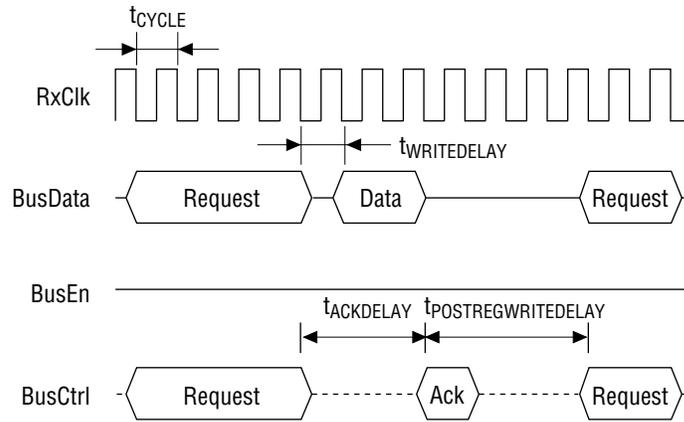
MANUAL REFRESH USING SETRR

t_{CYCLE}

RxClk

$t_{WRITEDELAY}$

BusData — Request — Data — Request

$t_{RETRYREFRESH}$

BusEn

$t_{ACKDELAY}$

BusCtrl — Ack
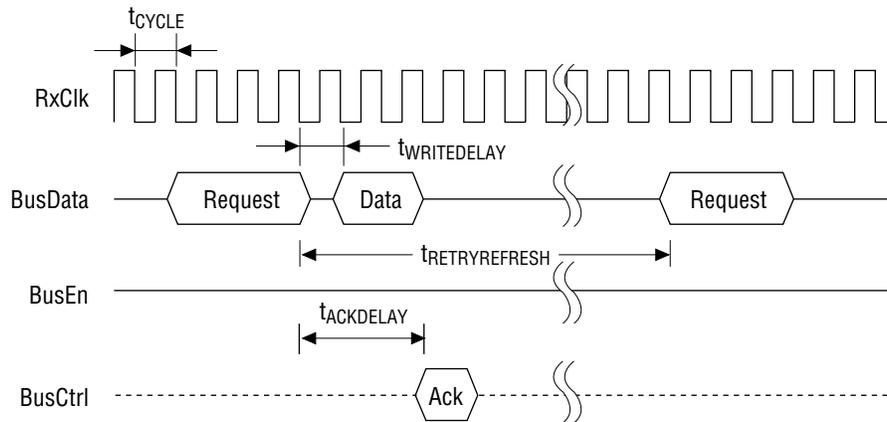
REFRESH TIMING FOR POWERDOWN MODE USING SIN, SOUT

$t_{REF}/1024$

SIn
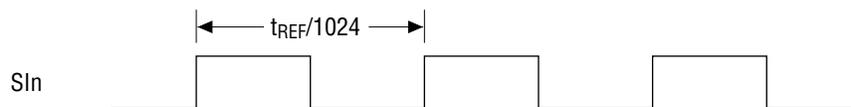
POWER DOWN TIMING (REGISTER WRITE)

t_{CYCLE}

RxClk

BusData — Request — Data

BusEn

$t_{ACKDELAY}$

BusCtrl — Request — Ack

Active Mode | PowerDown Mode

POWER UP TIMING

t_{CYCLE}

RxClk

BusData — Request

$t_{MODEPA[0/1],MIN}$ $t_{LOCK,POWERUP,MIN}$

BusEn — 11

$t_{ACKDELAY}$

BusCtrl — Request — Ack

PowerDown Mode | Active Mode
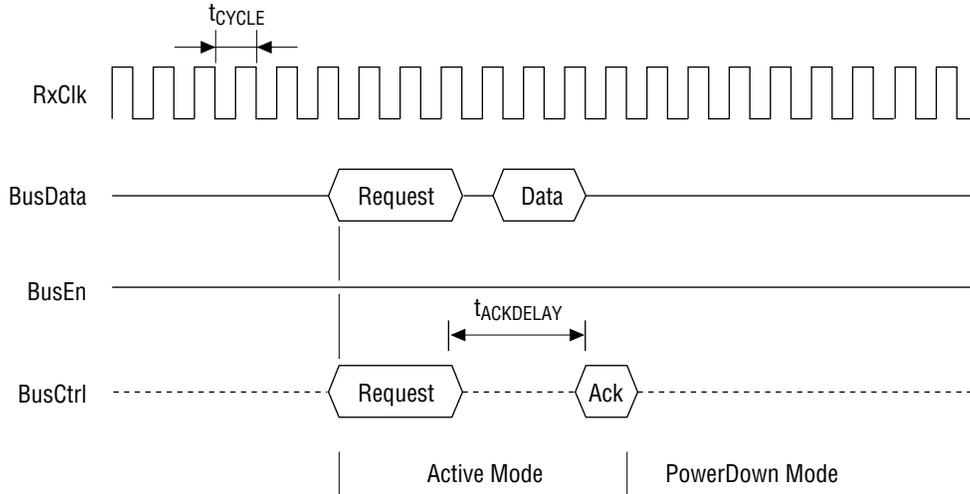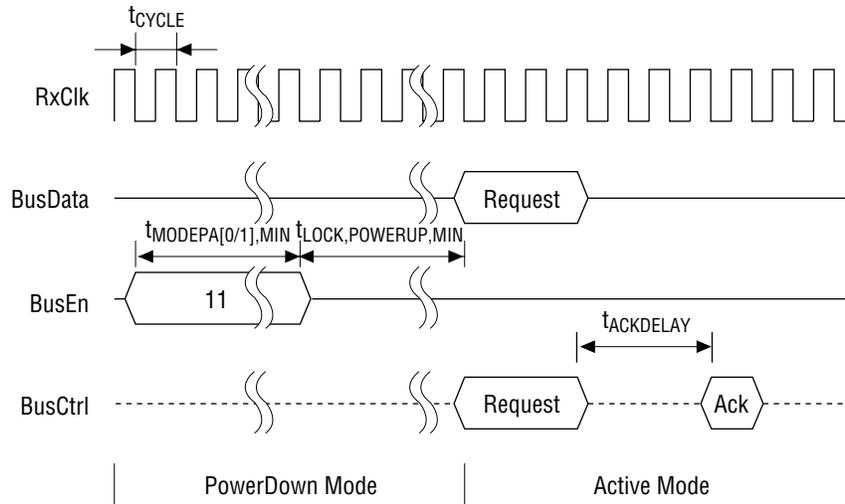
MECHANICAL DRAWINGS

The RDRAM is available in a horizontal surface mount plastic package. Dimensions for the horizontal surface mount plastic package are shown below.
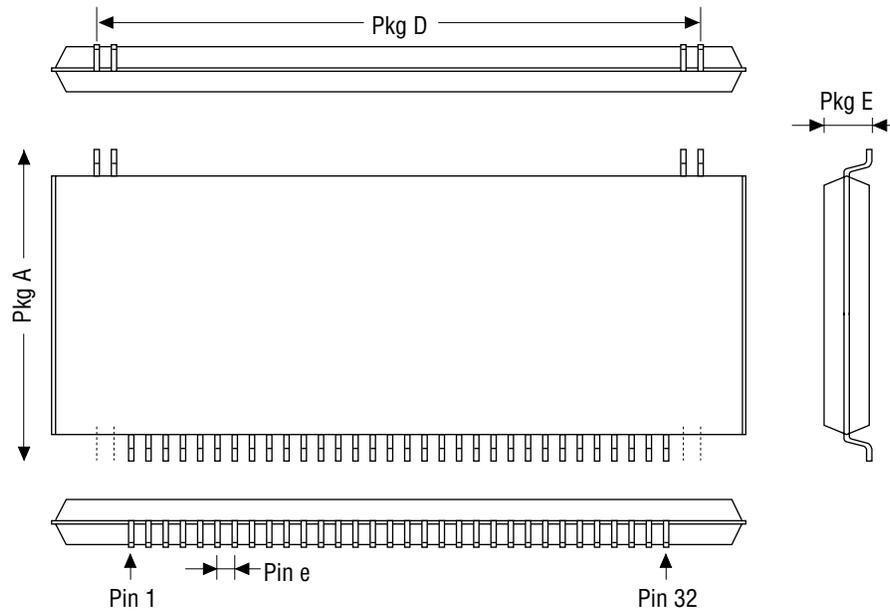
Fig. 17 SHP-32 Package

The next figure shows the footprint of the SHP-32 package. Plane R-R is the electrical reference plane of the device on the center line of the SMT pads.
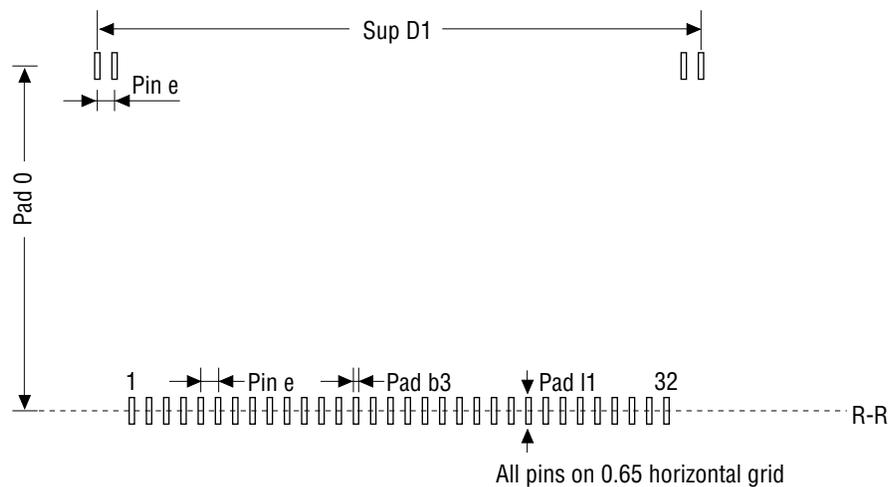
Fig. 18 SHP-32 Footprint

This table summarizes the values of the package and footprint dimensions.

Table 8 SHP-32 Package Dimensions

Symbol	Parameter	Min.	Typ.	Max.	Unit
Pin e	Pin pitch	—	0.65	—	mm
Pkg D	Package width	24.9	25.0	25.1	mm
Pkg A	Package total height	12.8	13.0	13.2	mm
Pkg E	Package thickness	—	—	1.75	mm
Pad b3	SMT pad width	0.30	—	0.40	mm
Pad l1	SMT pad length	1.2	—	1.4	mm
Sup D1	Support pad outer pitch	—	22.75	—	mm
Pad 0	SMT pad offset	—	12.5	—	mm