4-Megabit RDRAM (512K x 9)

Description

The 4-Megabit Rambus™ DRAM (RDRAM™) is an extremely-high-speed CMOS DRAM organized as 512K words by 9 bits and capable of bursting up to 256 bytes of data at 2 nanoseconds per byte. The use of Rambus Signaling Logic (RSL) technology makes this 500 MHz transfer rate achievable while using conventional system and board design methodologies. Low latency is attained by using the RDRAM's large internal sense amplifier arrays as high speed caches.

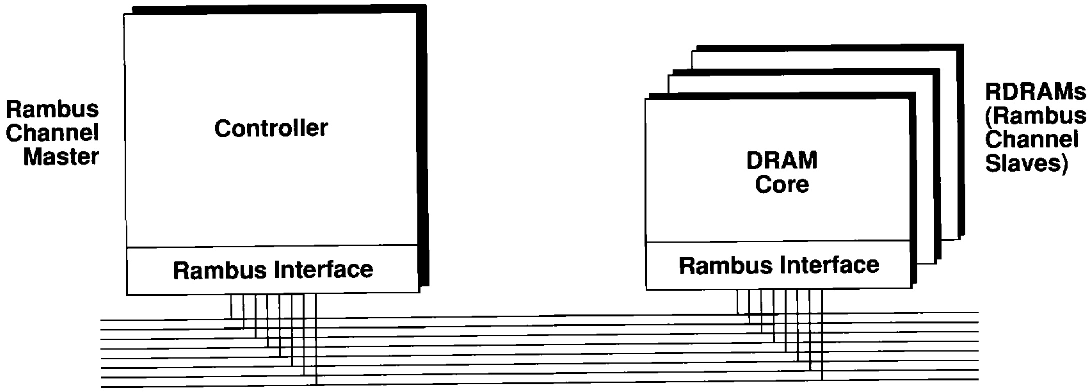
RDRAMs are general purpose high-performance memory devices suitable for use in a broad range of applications including main memory, graphics, video, and any other application where high-performance and low cost are required.

Features

- Rambus Interface:
  - 500 MB/sec peak transfer rate per RDRAM
  - RSL interface
  - Synchronous protocol for fast block-oriented transfers
  - Direct connection to Rambus ASICs, MPUs, and Peripherals
- 48 ns from start of read request to first byte; 2 ns per byte thereafter
- Dual 1KByte sense amplifiers act as caches for low latency accesses
- Control and refresh logic entirely self-contained
- EIAJ standard plastic vertical surface mount package (SVP)
- On-chip registers for flexible addressing and timing
- Low pincount – only 15 active signals
- Standardized pinout across multiple generations of RDRAMs
- 5 volt operation

System Benefits

- Can eliminate second-level caches in main memory applications
- Narrow 9-bit Channel simplifies and shrinks board layout
- Decreases system part count by eliminating caches, buffers, address decoders, extra buses, etc.
- Full 500MB/second bandwidth from just one RDRAM
- Multiple Rambus Channels provide an additional 500MB/second bandwidth each
- 3 to 5 times the performance of VRAMs at lower cost
- 5 to 15 times the performance of DRAMs
- Sufficient bandwidth to support unified graphic and main memory system designs
- Systems are modular – faster MPUs and larger Rambus memories can be installed without changing board layout or logic design



Rambus Channel = 9 bits every 2 ns

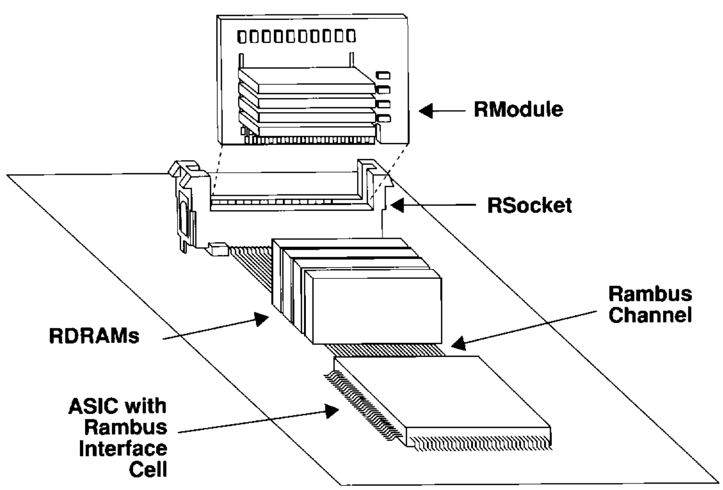
Rambus System Overview

A typical Rambus memory system has three main elements: the Rambus Channel, the RDRAMs, and a Rambus Interface on a controller. The logical representation of this is shown in the figure above.

The Rambus Channel is a synchronous, high-speed, byte-wide bus that is used to directly connect Rambus devices together. Using only 13 high-speed signals, the Channel carries all address, data, and control information to and from devices through the use of a high level block-oriented protocol.

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

The following figure 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.

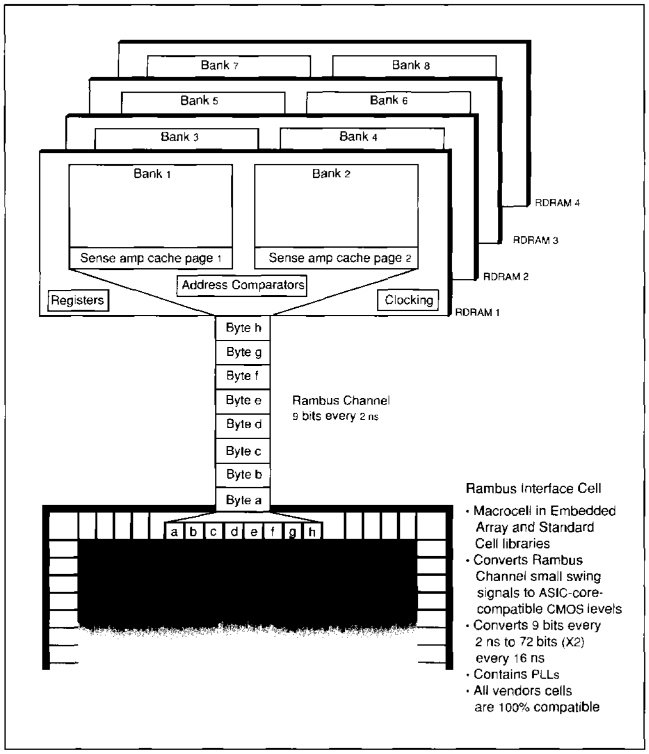


A Rambus System Example

Rambus Signaling Logic

RSL technology is the key to attaining the high data rates available in Rambus systems. By employing high quality transmission lines, current-mode drivers, low capacitive loading, low-voltage signaling, and precise clocking, systems reliably transfer data at 2 nano-second intervals on a Rambus Channel with signal quality that is superior to TTL or GTL-based interfaces.

All Rambus Interfaces incorporate special logic to convert signals from RSL to CMOS levels for internal use. In addition, these interfaces convert the Channel data rate of one byte every 2 nanoseconds to an internal data rate of 8 bytes every 16 nanoseconds as shown in the figure below. Although the bandwidth remains the same, the use of a wide internal bus eases internal timing requirements for chip designers.



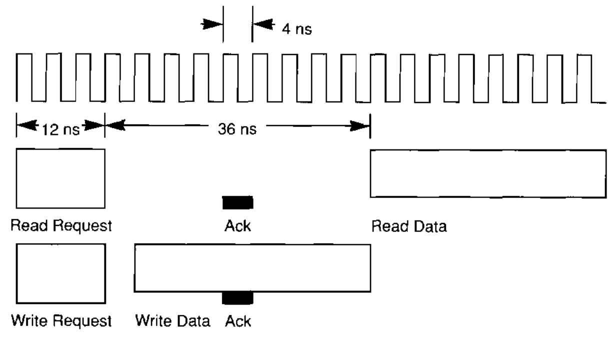
Protocol

The high-level transaction protocol used in Rambus systems is built from several types of information packets. These include the request, acknowledge, serial mode, and data packets. A master device initiates a transaction by generating a six byte request packet containing address, control, and byte count information as shown in the figure above.

Request Packet	BusCtrl	BusData[8:0]		
0	Start	Op[0]	Adr[9:2]	
1	Op[1]	Op[3]	Adr[17:10]	
2	OpX[1]	Adr[26:18]		
3	Op[2]	Adr[35:27]		
4	OpX[0]	ReqUnimp[5:4]	Count[6,4,2]	ReqUnimp[3:0]
5	ReqUnimp[8:6]	Count[7,5,3]	Count[1:0]	Adr[1:0]

Request Packet Format

All slave devices constantly monitor the Channel for a request to access their assigned memory range. The device matching the address range requested then drives an acknowledge packet back to the master. The RDRAM also drives a data packet back to the master in the case of a read, or accepts a data packet from the master in the case of a write. The figure below shows example 16 byte read and write transactions. The actual timing from the end of a request packet to data and acknowledge packets is adjustable through RDRAM register settings.



Sample 16-Byte Read and Write Transactions

RDRAM Commands

The request packet generated by the master device contains an Op field used to specify the type of operation to be performed on the RDRAM being accessed. The following table describes the types of operations supported in the 4-Megabit RDRAM. Implied operations, such as non-sequential access and bit masking, are not available in this device but are supported in other Rambus products.

Op[3:0]	Name	Description
0000	Rseq	Read sequential data from memory space.
0100	WseqNpb	Write sequential data to memory space with no per-bit mask application.
0110	Rreg	Read data from register space.
0111	Wreg	Write data to register space.
1111	WregB	Broadcast write to register space of all responding devices with no acknowledge permitted.

RDRAM Overview

Conceptually, the RDRAM can be divided into three sections referred to as the physical, logical and application layers. These are shown in the block diagram on the next page.

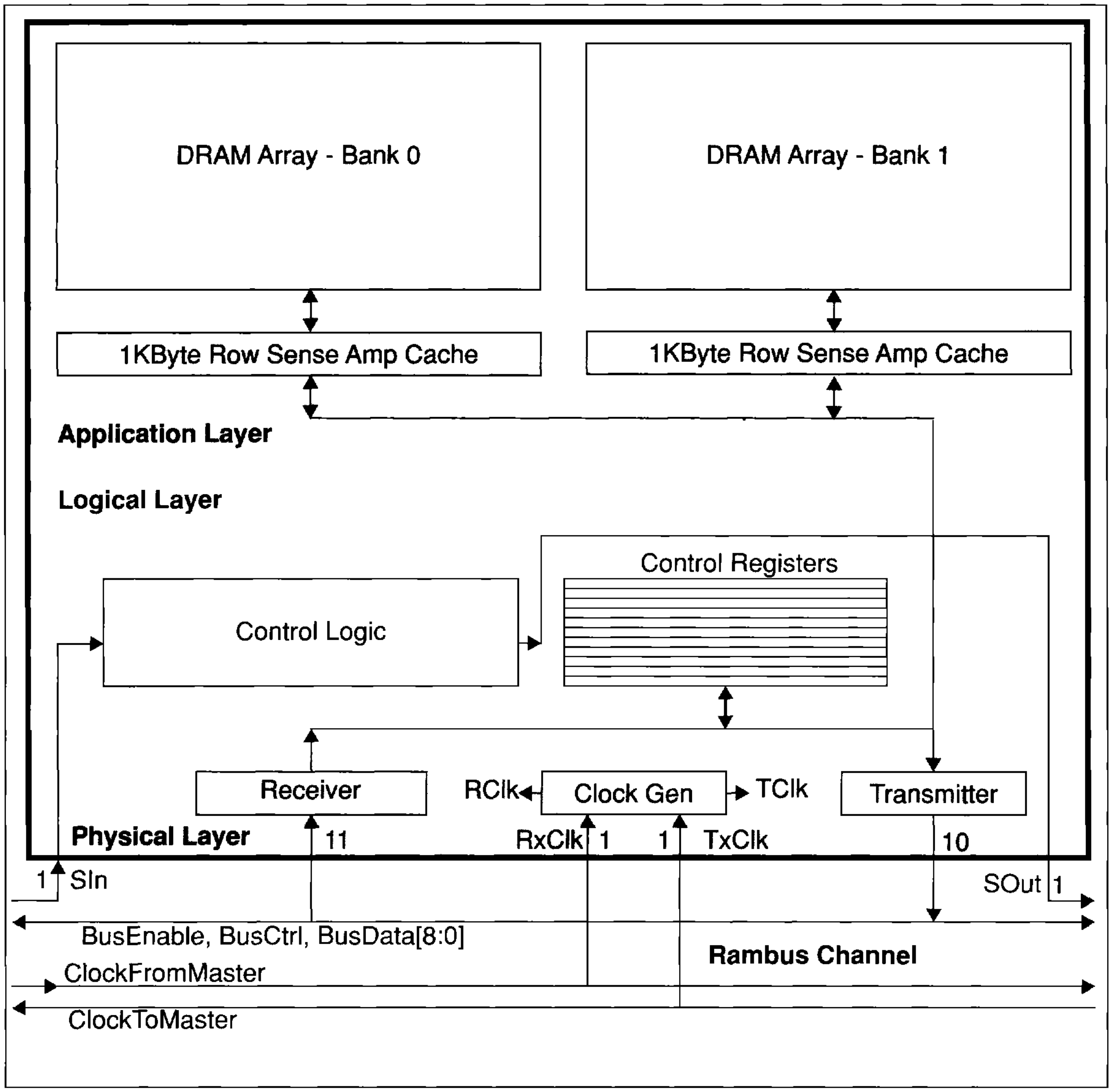
The physical layer consists of a clock generator, a receiver, and a transmitter. The clock generator uses the external clock signals RxClk and TxClk (tapped off the Channel traces ClockFromMaster and ClockTo-Master) and creates the internal signals RClk and TClk. These are used by the receiver and transmitter, 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 16 nanoseconds.

The logical layer consists of 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 the next page.

The application layer consists of a standard DRAM memory core and row sense amplifier caches.

Caches

Each RDRAM is broken down into two independent banks of memory. Each of these banks has a 1KByte cache line associated with it that is built out of large sense amplifier arrays. These caches work by holding the last accessed row of their associated bank in the sense amplifiers allowing further accesses to the same row of memory to result in cache hits. With the row already stored in the cache, data can be accessed with very low latency. Each RDRAM added to a system adds two cache lines to the memory system helping increase cache hit rates.



Register	Description
DeviceType	Readable register that defines the type and size of the device along with part version information.
DeviceId	Used to define the base address for the RDRAM.
Delay	Used to specify programmable CAS-type delay values.
Mode	Used to enable device, control refresh mode, and adjust current control settings.
RefInterval	Used to define the refresh interval for self-refresh.
RefRow	Contains the current refresh row and bank information.
RasInterval	Used to define RAS timing parameters.
MinInterval	Readable register defining minimum timings for Delay register parameters.
AddressSelect	Used to define address bit swapping to alter RDRAM address mapping.
DeviceManufacture	Readable register containing a manufacturer code and a manufacturer specific part type code.
Row	Readable register containing the row numbers for the currently sensed rows in each bank.

A cache miss results when a row is accessed that is not currently stored in one of the cache lines. 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 cache line and waits for the master to submit a retry of the previous request. The figure below shows an example of a read miss followed by a read hit for a 32 byte memory read operation.

Address Mapping

Address mapping hardware is provided to increase cache 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. By using address mapping, contiguous blocks of addresses are split across several RDRAMs, and therefore across several cache lines. In a typical system containing, for example, eight RDRAMs, miss rates could be expected to be as low as 5%. Address mapping is easily adjusted by writing a control register in each RDRAM.

Standby Mode

The RDRAM's normal state while not being accessed is standby mode. While in this low power state, each RDRAM monitors the BusEnable signal for a serial mode packet while ignoring most other activity on the remaining Channel signals. The serial mode packet is used by a master device to bring all RDRAMs temporarily out of standby and into active mode so they can respond to a request packet. Once a request packet is acknowledged, all RDRAMs return to standby mode with the exception of the one responding to the request. This device will return to standby mode once the read or write operation is completed.

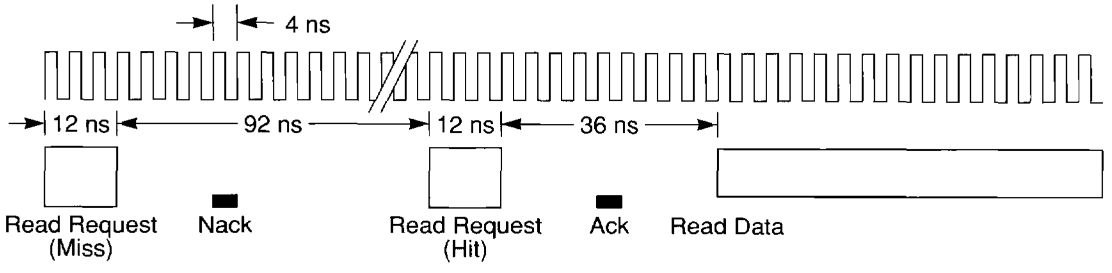
Unlike conventional DRAM memory systems where each device in an entire bank of memory must be kept active and consuming power through an entire access, Rambus memory systems allow only one active device while all others remain in a low power state.

Transaction Concurrency

Concurrent transactions can be used to optimize RDRAM utilization in high performance applications by taking advantage of available Channel bandwidth during cache miss latency periods. When a miss in one RDRAM takes place, that device will be busy loading a new row into one of its cache lines. Other than that, the Channel and all other RDRAMs will still be available for use. Instead of waiting for the first RDRAM to finish loading its cache, a transaction to another RDRAM can be initiated.

This can be used in various ways. In systems where memory accesses can be queued, a transaction can take place for any pending access residing in a different RDRAM. When that transaction is complete, the first transaction can be retried.

Pretouching can be used in systems where 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 the original RDRAM is known in advance, a dummy transaction can be first generated to cause a row miss and prepare it for its next access. Transactions to other devices can then take place while a cache fill is taking place. When the original device is next accessed, the required row of data will already be loaded in the cache line and a cache hit will take place.

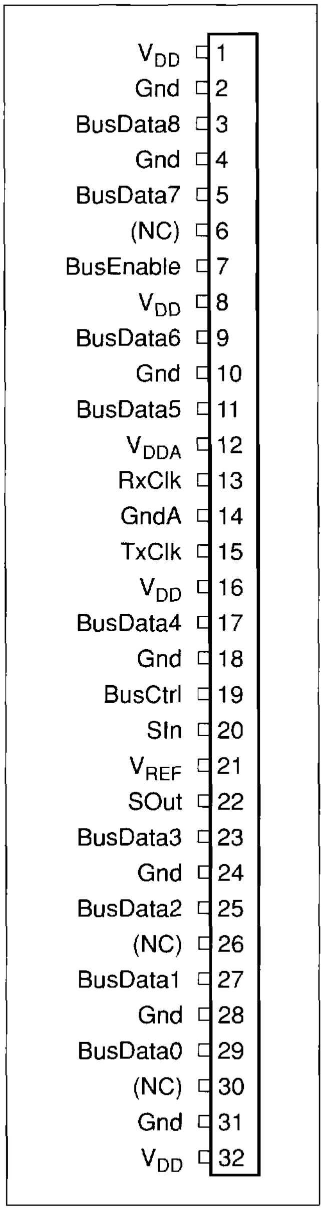


Sample 32-Byte Read Miss and Read Hit Transactions

RDRAM Interface Signals

Every RDRAM comes in a EIAJ standard (#ED-7424) 32-pin vertical surface mount package. Thirteen RSL I/O signals are used to transfer information at 2 nanosecond intervals. Two TTL signals are used for initialization and powerdown operation. The remaining 17 signals supply power and DC voltage references to the RDRAM.

32-Pin SVP SIP



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. This is a low-swing, active-low signal 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 packets are aligned with this clock. This is a low-swing, active-low signal referenced to Vref.
Vref	I	Logic threshold voltage for low swing signals.
BusCtrl	I/O	Control signal to frame packets, to transmit part of the operation code, and to acknowledge requests. Low-swing, active-low signal referenced to Vref.
BusEnable	I	Control signal to enable the bus. Long assertions of this signal will reset all devices on the Channel. This is a low-swing, active-low signal referenced to Vref.
Vdd, VddA		+5V power supply. VddA is a separate analog supply.
Gnd, GndA		Circuit ground. GndA is a separate analog ground.
SIn	I	Initialization daisy chain input. TTL levels. Active high.
SOut	O	Initialization daisy chain output. TTL levels. Active high.

Absolute Maximum Ratings

The following table represents stress ratings only, and functional operation at the maximums is not guaranteed. Extended exposure to the maximum ratings may affect device reliability. Furthermore, although 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}+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 VDD 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
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		10	pF

Power Consumption

Mode	Description	Min	Max	Unit
Standby	Device inactive		500	mW
Active	Device evaluating request packet		750	mW
Read	Data being transfered from device		2400	mW
Write	Data being transfered to device		1600	mW

Recommended Electrical Conditions

Symbol	Parameter and Conditions	Min	Max	Unit
$V_{DD, VDDA}$	Supply voltage	4.5	5.5	V
V_{REF}	Reference voltage	1.9	2.4	V
V_{IL}	Input low voltage	V_{REF} - 0.8	V_{REF} - 0.2	V
V_{IH}	Input high voltage	V_{REF} + 0.2	V_{REF} + 0.8	V
$V_{IL,TTL}$	TTL input low voltage	- 0.5	0.8	V
$V_{IH,TTL}$	TTL input high voltage	2.0	V_{DD} + 0.5	V

Electrical Characteristics

Symbol	Parameter and Conditions	Min	Max	Unit
I_{REF}	V_{REF} current @ $V_{REF,MAX=2.4V}$	-10	10	µA
I_{OH}	Output high current @ ($0 \le V_{OUT} \le VDD$)	-10	10	µA
I_{OL}	Output low current @ V_{OUT} = 1.6V		35	mA
ΔI_{OL}	Error in programmed output low current (from unit to unit)	-1.1	1.1	mA
$I_{I,TTL}$	TTL input leakage current @ ($0 \le V_{I,TTL} \le VDD$)	-10.0	10.0	µA
$V_{OL,TTL}$	TTL output voltage @ $I_{OL,TTL}$= 1.0mA	0.0	0.4	V
$V_{OH,TTL}$	TTL output high voltage @ $I_{OH,TTL}$= -0.25mA	2.4	V_{DD}	V

Recommended Timing Conditions

Symbol	Parameter	Min	Max	Unit
t_{CR}, t_{CF}	TxClk and RxClk input rise and fall times	0.3	0.7	ns
t_{CYCLE}	TxClk and RxClk cycle times	4	10	ns
t_{TICK}	Transport time per bit per pin (this timing interval is synthesized by the RDRAM's internal clock generator)	0.5 (2ns @ t_{CYCLE} = 4ns)	0.5 (2ns @ t_{CYCLE} = 4ns)	t_{CYCLE}
t_{CH}, t_{CL}	TxClk and RxClk high and low times	45%	55%	t_{CYCLE}
t_{TR}	TxClk-RxClk differential	0	t_{CYCLE} - 1.2	ns
t_{DR}, t_{DF}	Data/Control input rise and fall times	0.3	0.7	ns
t_{QR}, t_{QF}	Data/Control output rise and fall times	0.4	0.6	ns
t_S	Data/Control-to-RxClk setup time	0.3		ns
t_H	RxClk-to-Data/Control hold time	0.3		ns
t_{REF}	Refresh interval		16	ms

Symbol	Parameter	Min	Max	Unit
$t_{LOCK,ACTIVE}$	RDRAM internal clock generator lock time in active mode		3750 (15µs @ t_{CYCLE} = 4ns)	t_{CYCLE}
$t_{LOCK,STANDBY}$	RDRAM internal clock generator lock time in standby mode		27000 (108µs @ t_{CYCLE} = 4ns)	t_{CYCLE}

Timing Characteristics

Symbol	Parameter	Min	Max	Unit
t_{PIO}	SIn-to-SOut propagation delay @ $C_{LOAD,TTL=40pF}$	1	50	ns
t_Q	TClk-to-Data/Control output time	$t_{CYCLE}/4 - 0.3$	$t_{CYCLE}/4 + 0.3$	ns

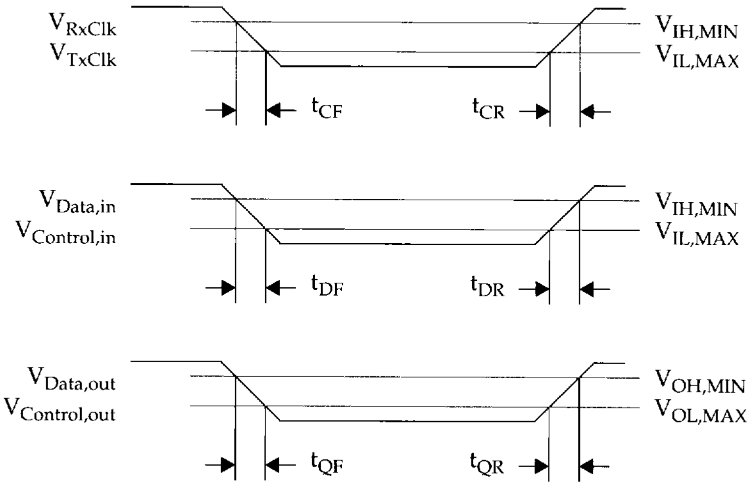
Rambus Channel Timing

The next table shows important timings on the Rambus Channel for common operations. Please refer to the RDRAM Reference Manual and 4Mb RDRAM Specification for all possible interactions that could occur on the Rambus Channel. All timings are from the point of view of the Channel master, and thus have the bus overhead delay of 4ns per bus transversal included where appropriate.

Symbol	Parameter	Min	Max	Unit
t_{CYCLE}	TxClk and RxClk cycle times	4	10	ns
$t_{RESPONSE}$	Start of request packet to start of acknowledge packet.	7 [a]	10 (1)	t_{CYCLE}
$t_{READHIT}$	Start of request packet to start of read data packet for row hit (Okay).	12 (1)	15 (1)	t_{CYCLE}
$t_{WRITEHIT}$	Start of request packet to start of write data packet for row hit (Okay).	4 (1)	7 (1)	t_{CYCLE}
$t_{RETRYSENSEDCLEAN}$	Start of request packet for row miss (Nack) to start of request packet for row hit (Okay). The previous row is unmodified.	26 [b]		t_{CYCLE}
$t_{READBURST32}$	Start of request packet to end of 32 byte read data packet for row hit (Okay).	28 [c]		t_{CYCLE}
$t_{READBURST256}$	Start of request packet to end of 256 byte read data packet for row hit (Okay).	140 (3)		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 (4)		t_{CYCLE}

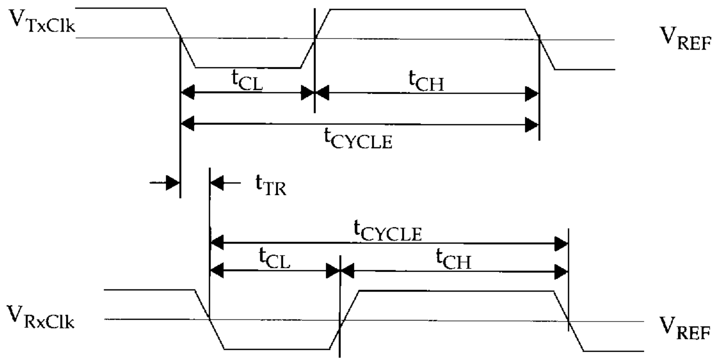
a. Programmable
b. Minimum at $t_{CYCLE,MIN}$. Delay programmable to give equivalent timings at longer t_{CYCLE}.
c. Calculated with $t_{READHIT,MIN}$
d. Calculated with $t_{WRITEHIT,MIN}$

Rise/Fall Timing

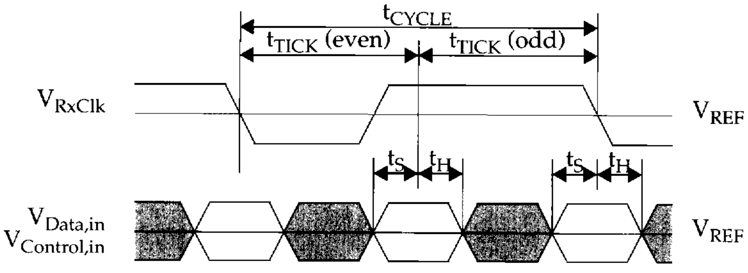


Where:
$V_{OH,MIN} = V_{TERM} - Z_O * (I_{OH,MIN})$
$V_{OL,MAX} = V_{TERM} - Z_O * (I_{OL,MAX} * \text{programmed value})$

Clock Timing

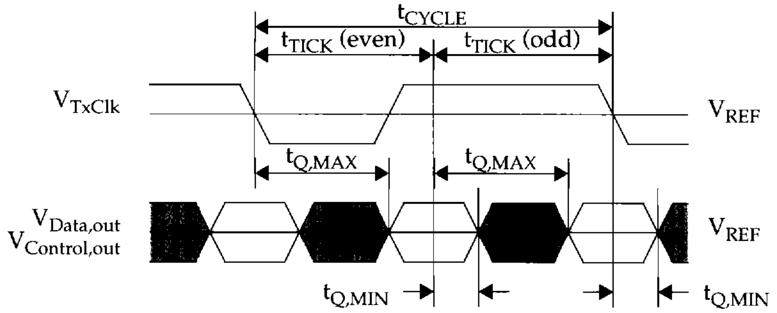


Receive Data Timing



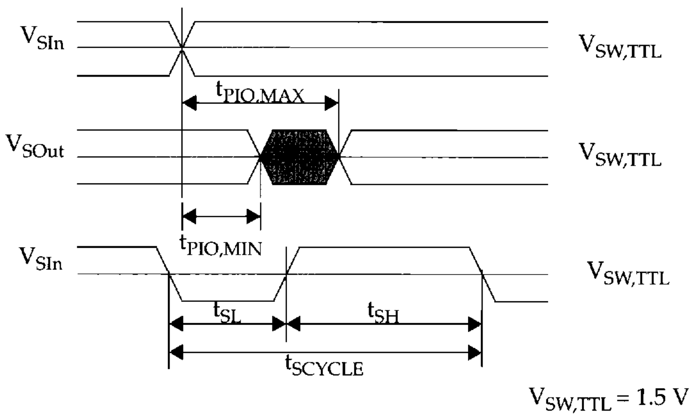
* t_{TICK} is defined as one-half t_{CYCLE}.

Transmit Data Timing

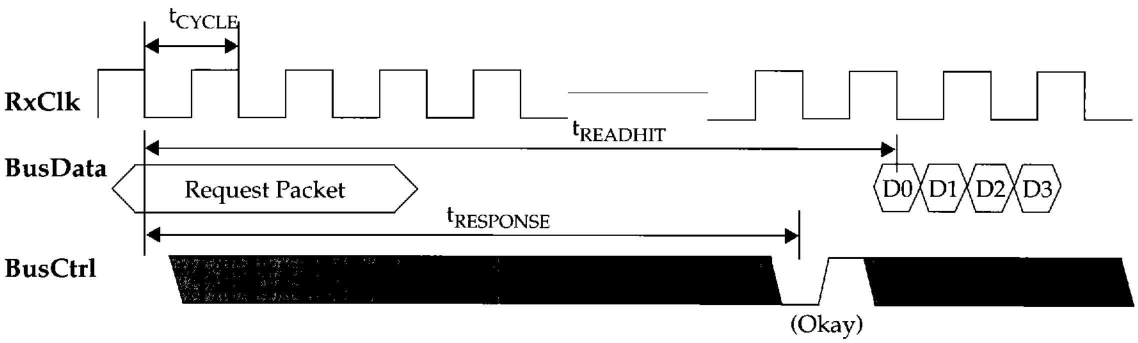


* t_{TICK} is defined as one-half t_{CYCLE}.

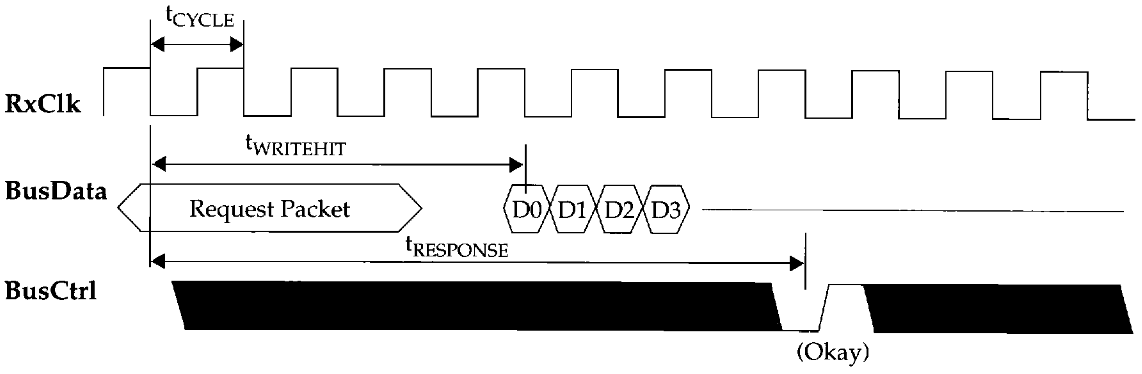
Serial Configuration Pin Timing



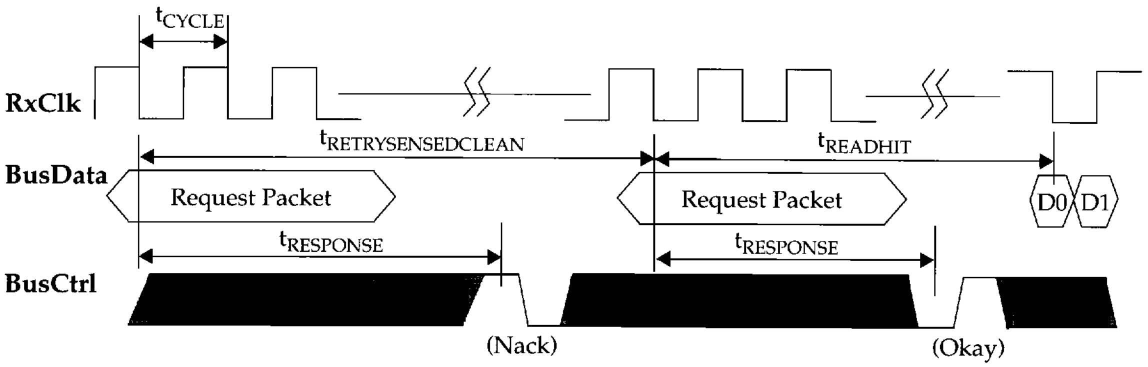
Read Hit Timing Diagram



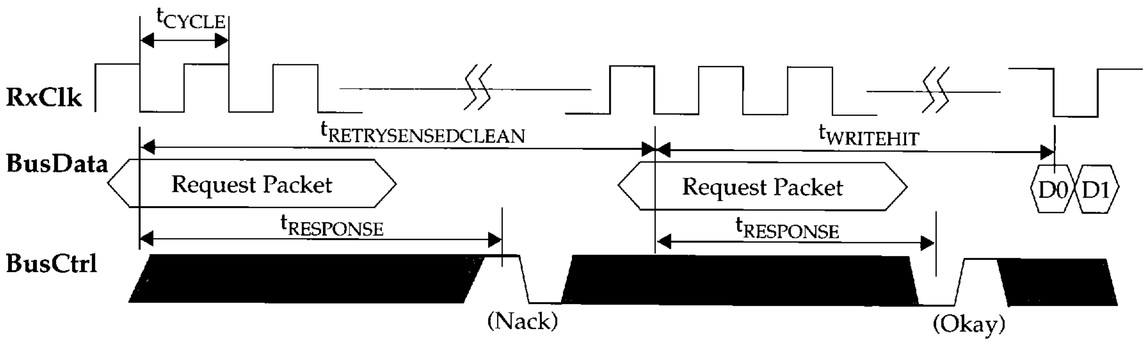
Write Hit Timing Diagram



Read Miss Timing Diagram

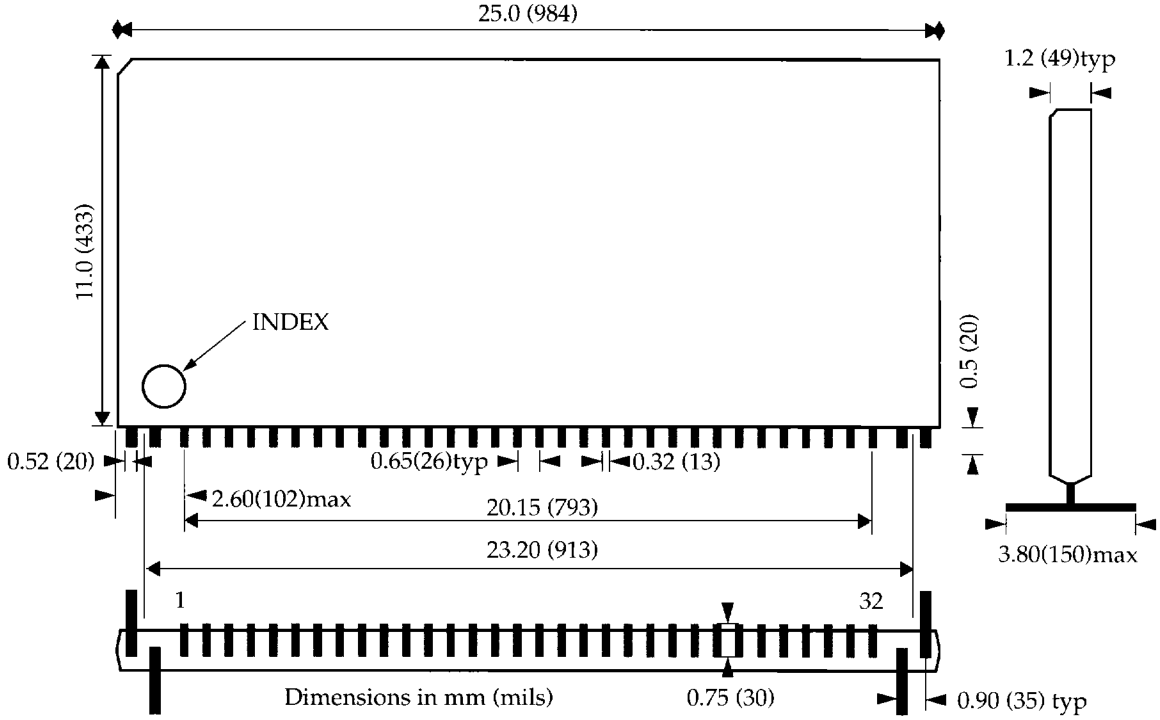


Write Miss Timing Diagram



Mechanical Drawings

The RDRAM package is a vertical surface mount package (SVP) which is an EIAJ standard (#ED-7424). Dimensions are shown in the figure below.



Surface Vertical Package (SVP) Outline mm (mils)