Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

IDT™ Interprise™ Integrated Communications Processor

RC32365

Device Overview

The RC32365 device is a member of the IDT™ Interprise™ family of integrated communications processors. This device is designed to address a range of communications applications that require the efficient processing of IPSec algorithms. These applications include gateways, wireless access points, and virtual private network (VPN) equipment. The key to the RC32365's efficient processing of IPSec algorithms is a highly progammable security engine which off-loads the CPU core of encryption/decryption, hashing, and padding tasks.

Features List

- **RC32300 32-bit CPU core**
  - *32-bit MIPS instruction set*
  - *Supports big or little endian operation*
  - *MMU*
    - *16-entry TLB*
    - *Supports variable page sizes and enhanced write algorithm*
    - *Supports variable number of locked entries*
  - *8KB Instruction Cache*
    - *2-way set associative*
    - *LRU replacement algorithm*
    - *4 word line size*
    - *Sub-block ordering*
    - *Word parity*
    - *Per line cache locking*
  - *2KB Data Cache*
    - *2-way set associative*
    - *LRU replacement algorithm*
    - *4 word line size*
    - *Sub-block ordering*
    - *Byte parity*
    - *Per line cache locking*
    - *Can be programmed on a page basis to implement write-through no write allocate, write-through write allocate, or write-back algorithms*
  - *Enhanced EJTAG and JTAG Interfaces*
    - *Compatible with IEEE Std. 1149.1-1990*
- **Security Engine**
  - *Dedicated DMA channels for high speed data transfers to and from the security engine*
  - *On-chip memory for storage of two security contexts*
  - *Supports ECB and CBC modes for the following symmetric encryption algorithms: DES, triple DES (both two key (k1=k3) and three key (k1!=k3) modes), AES-128 with 128-bit blocks, AES-192 with 128-bit blocks*
  - *Hardware support for encryption pad generation and checking using one of seven popular padding algorithms: supports pad algorithm required by IPSec ESP*
  - *Supports MD5 and SHA-1 one-way hash functions*
  - *Programmable truncation length of computed hash and HMAC on a security context basis*
  - *Supports concurrent hash and encryption operations*

Block Diagram

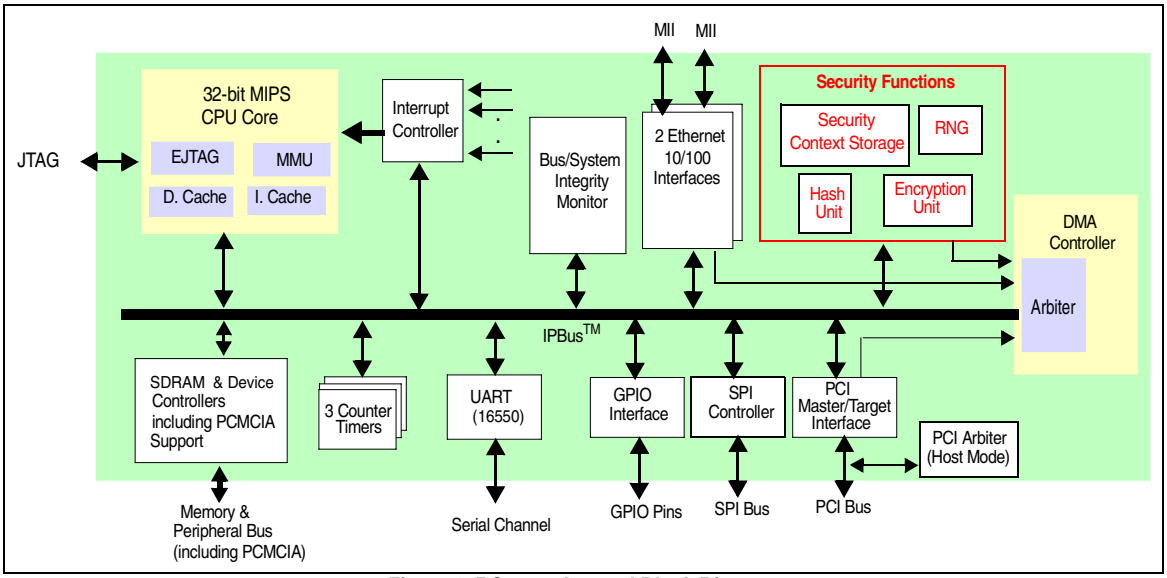


Figure 1 RC32365 Internal Block Diagram

- *Optimized for IPSec AH, ESP, and AH+ESP (single MAC) tunnel and transport mode processing: initialization Vector (IV) insertion and extraction, HMAC checking, AH mutable field processing for both IPv4 and IPv6 packets, IPSec pad generation and checking*

- **Random Number Generator**
  - *True hardware random number generator suitable for security applications: may be used to generate symmetric and public keys, initialization vectors, and nonces*
  - *Dedicated DMA engine for transferring random numbers to memory*
  - *Generates random numbers at a bit rate equal to IPBus clock frequency divided by 32*
  - *Provides 4 word (16 byte) FIFO to queue random numbers*
  - *Randomness tester continually verifies proper operation of random number generator using a randomness test defined in FIPS 140-2*
- **PCI Interface**
  - *32-bit PCI revision 2.2 compliant*
  - *Supports host or satellite operation in both master and target modes*
  - *PCI clock: supports frequencies from 16 MHz to 66 MHz, PCI clock may be asynchronous to master clock (CLK)*
  - *PCI arbiter in Host mode: supports 3 external masters, fixed priority or round robin arbitration*
  - *I_2O "like" PCI Messaging Unit*
- **Two Ethernet Interfaces**
  - *10 and 100 Mb/s ISO/IEC 8802-3:1996 compliant*
  - *Two IEEE 802.3u compatible Media Independent Interfaces (MII) with serial management interface*
  - *MII supports IEEE 802.3u auto-negotiation speed selection*
  - *Supports 64 entry hash table based multicast address filtering*
  - *512 byte transmit and receive FIFOs*
  - *Supports flow control functions outlined in IEEE Std. 802.3x-1997*
- **SDRAM Controller**
  - *Supports up to 512 MB of memory*
  - *2 chip selects (each supports 2 or 4 banks internal SDRAM banks)*
  - *32-bit data width, supports 8/16/32-bit width devices*
  - *Supports 16Mb, 64Mb, 128Mb, and 256Mb, and 512Mb devices*
  - *Automatic refresh generation*
- **Memory and Peripheral Device Controller**
  - *Provides "glueless" interface to standard SRAM, Flash, ROM, dual-port memory, and peripheral devices*
  - *Provides "glueless" interface to many 16-bit PCMCIA devices*
  - *Demultiplexed address and data buses: 32-bit data bus, 26-bit address bus, 6 chip selects, control for external data bus buffers*
  - *Supports 8-bit, 16-bit, and 32-bit width devices: automatic byte gathering and scattering*
  - *Flexible protocol configuration parameters: programmable number of wait states (0 to 63), programmable postread/post-write delay (0 to 31), supports external wait state generation, supports Intel and Motorola style peripherals*
  - *Write protect capability per chip select*
  - *Programmable bus transaction timer generates warm reset when counter expires*
  - *Supports up to 64MB of memory per chip select*
- **DMA Controller**
  - *9 DMA channels: two channels for each of the two Ethernet interfaces (transmit/receive), two channels for PCI (PCI to Memory and Memory to PCI), two channels for security engine (input/output), one channel for the hardware random number generator*
  - *Provides flexible descriptor based operation*
  - *Supports unaligned transfers (i.e., source or destination address may be on any byte boundary) with arbitrary byte length*
- **General Purpose Peripherals**
  - *Serial port compatible with 16550 Universal Asynchronous Receiver Transmitter (UART)*
  - *Three general purpose 32-bit counter/timers*
  - *Interrupt Controller*
  - *Serial Peripheral Interface (SPI) supporting host mode*
  - *16 general purpose I/O (GPIO) pins which can be configured as interrupt sources*
- **System Features**
  - *JTAG Interface (IEEE Std. 1149.1 compatible)*
  - *256 pin CABGA package*
  - *2.5V core supply and 3.3V I/O supply*

CPU Execution Core

The RC32365 is built around the RC32300 32-bit high performance microprocessor core. The RC32300 implements the enhanced MIPS-II ISA and helps meet the real-time goals and maximize throughput of communications and consumer systems by providing capabilities such as a prefetch instruction, multiple DSP instructions, and cache locking. The instruction set is largely compatible with the MIPS32 instruction set, allowing the customer to select from a broad range of software and development tools. Cache locking guarantees real-time performance by holding critical code and parameters in the cache for immediate availability. The microprocessor also implements an on-chip MMU with a TLB, making the it fully compliant with the requirements of real time operating systems.

Security Engine

The RC32365 incorporates an on-chip security engine that has been designed to accelerate IPSec performance and minimize the amount of performance required by the CPU to process secure packet traffic. The engine includes hardware support for the DES, 3DES, and AES encryption algorithms and the MD5 and SHA1 hash functions. The engine also supports hardware-assisted packet processing for the various modes of IPSec, including AH, ESP, and AH+ESP tunnel and transport modes. Two dedicated DMA channels are used to transfer data to and from the security engine, allowing the CPU to work on other tasks during this time.

PCI Interface

The PCI interface on the RC32365 is compatible with version 2.2 of the PCI specification. An on-chip arbiter supports up to three external bus masters, supporting both fixed priority and rotating priority arbitration schemes. The RC32365 can support both satellite and host PCI configurations, enabling it to act as a slave controller for a PCI add-in card application, or as the primary PCI controller in the system. The PCI interface can be operated synchronously or asynchronously to the other I/O interfaces on the RC32365 device.

PCMCIA Interface

The RC32365 provides a "glueless" connection to a single PCMCIA I/O device via the memory and peripheral device controller. The PCMCIA interface allows the RC32365 to connect to various types of I/O peripherals including fax modems, storage devices, and wireless LAN chipsets. The RC32365 implementation provides a maximum throughput of 160 Mbps through the 16-bit wide interface as specified by the PCMCIA 2.1 Standard.

Ethernet Interface

The RC32365 has two Ethernet Channels supporting 10Mbps and 100Mbps speeds and provides a standard media independent interface (MII) off-chip, allowing a wide range of external devices to be connected efficiently.

Memory and I/O Controller

The RC32365 incorporates a flexible memory and peripheral device controller providing direct support for SDRAM, Flash ROM, SRAM, PCMCIA, and other I/O devices. It can interface directly to 8-bit boot ROM for a very low cost system implementation. It also offers various trade-offs in cost / performance for the main memory architecture. The timers implemented on the RC32365 satisfy the requirements of most real time operating systems.

DMA Controller

The DMA controller off-loads the CPU core from moving data among the on-chip interfaces, external peripherals, and memory. The DMA controller supports scatter / gather DMA with no alignment restrictions, appropriate for communications and graphics systems.

Enhanced JTAG Interface

For system debugging, the RC32300 CPU core includes an Enhanced JTAG (EJTAG) interface which operates in Run-Time Mode.

Thermal Considerations

The RC32365 is guaranteed in a ambient temperature range of 0° to +70° C for commercial temperature devices and - 40° to +85° for industrial temperature devices.

Revision History

March 17, 2003: Initial publication.

May 15, 2003: Removed “write protect capability” from features of the SDRAM Controller.

July 9, 2003: In Table 6, changed values for RSTN (output). Changed values in Tables 7, 8, 9, 10, and 17.

October 3, 2003: Added 180 MHz speed grade. Changed min values in Table 7 from 1.8 to 1.2 for all signals except SDCLKINP and SDCKENP. Changed min values for Tdo 10b and 10c in Table 10 for PCIBEN, etc. and PCIGNTN/PCIREQN from 2.0 to 1.5.

February 25, 2004: Deleted reference to RNGCLK in Table 1 (GPIO[6]) and Table 22.

May 25, 2004: In Table 9, signals MIIxRXCLK and MIIxTXCLK, the Min and Max values for Thigh/Tlow_9c were changed to 140 and 260 respectively and the Min and Max values for Thigh/Tlow_9d were changed to 14.0 and 26.0 respectively.

October 5, 2005: Removed 180 MHz speed grade.

Pin Description Table

The following table lists the functions of the pins provided on the RC32365. Some of the functions listed may be multiplexed onto the same pin (indicated as alternate functions).

To define the active polarity of a signal, a suffix will be used. Signals ending with an “N” should be interpreted as being active, or asserted, when at a logic zero (low) level. All other signals (including clocks, buses, and select lines) will be interpreted as being active, or asserted, when at a logic one (high) level.

Signal	Type	Name/Description
Memory and Peripheral Bus		
BDIRN	O	**External Buffer Direction.** Memory and peripheral bus external data bus buffer direction control. If the RC32365 memory and peripheral bus is connected to the A side of a transceiver such as an IDT74FCT245, then this pin may be directly connected to the direction control (e.g., BDIR) pin of the transceiver.
BOEN[1:0]	O	**External Buffer Enable.** These signals provide output enable control for external buffers on the memory and peripheral data bus.
BWEN[3:0]	O	**Byte Write Enables.** These signals are memory and peripheral bus byte write enable signals. BWEN[0] corresponds to byte lane MDATA[7:0] BWEN[1] corresponds to byte lane MDATA[15:8] BWEN[2] corresponds to byte lane MDATA[23:16] BWEN[3] corresponds to byte lane MDATA[31:24]
CSN[5:0]	O	**Chip Selects.** These signals are used to select an external device on the memory and peripheral bus.
MADDR[21:0]	O	**Address Bus.** 22-bit memory and peripheral bus address bus. MADDR[25:22] are available as GPIO[5:2] alternate functions.
MDATA[31:0]	I/O	**Data Bus.** 32-bit memory and peripheral data bus. During a cold reset, bits 0 through 16 of this data bus function as inputs that are used to load the boot configuration vector.
OEN	O	**Output Enable.** This signal is asserted when data should be driven by an external device on the memory and peripheral bus.
RWN	O	**Read Write.** This signal indicates whether the transaction on the memory and peripheral bus is a read transaction or a write transaction. A high level indicates a read from an external device. A low level indicates a write to an external device.
WAITACKN	I	**Wait or Transfer Acknowledge.** When configured as wait, this signal is asserted during a memory and peripheral bus transaction to extend the bus cycle. When configured as a transfer acknowledge, this signal is asserted during a transaction to signal the completion of the transaction.
RASN	O	**SDRAM Row Address Strobe.** Row address strobe asserted during memory and peripheral bus SDRAM transactions.
CASN	O	**SDRAM Column Address Strobe.** Column address strobe asserted during memory and peripheral bus SDRAM transactions.
SDCSN[1:0]	O	**SDRAM Chip Selects.** These signals are used to select SDRAM device(s) on the memory and peripheral bus.
SDWEN	O	**SDRAM Write Enable.** This signal is asserted during memory and peripheral bus SDRAM write transactions.
SDCLKOUT	O	**SDRAM Clock Output.** This clock is used for all SDRAM memory and peripheral bus operations.
SDCLKINP	I	**SDRAM Clock Input.** This clock input is typically a delayed version of SDCLKOUT. Data from the SDRAMs is sampled using this clock.

Table 1 Pin Description (Part 1 of 6)

Signal	Type	Name/Description
General Purpose I/O		
GPIO[0]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: U0SOUT Alternate function: UART channel 0 serial output.
GPIO[1]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: U0SINP Alternate function: UART channel 0 serial input.
GPIO[2]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: MADDR[22] Alternate function: Memory and Peripheral bus address bit 22 (output).
GPIO[3]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: MADDR[23] Alternate function: Memory and Peripheral bus address bit 23 (output).
GPIO[4]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: MADDR[24] Alternate function: Memory and Peripheral bus address bit 24 (output).
GPIO[5]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: MADDR[25] Alternate function: Memory and Peripheral bus address bit 25 (output).
GPIO[6]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. The value of this pin may be used as a Counter Timer Clock input.
GPIO[7]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: SDCKENP Alternate function: SDRAM clock enable output The value of this pin may be used as a Counter Timer Clock input.
GPIO[8]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: CEN1 Alternate function: PCMCIA chip enable 1 (CE1#) (output).
GPIO[9]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: CEN2 Alternate function: PCMCIA chip enable 2 (CE2#) (output).
GPIO[10]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: REGN Alternate function: PCMCIA Attribute Memory Select (REG#) (output).
GPIO[11]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: IORDN Alternate function: PCMCIA IO Read (IORD#) (output).
GPIO[12]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: IOWRN Alternate function: PCMCIA IO Write (IOWR#) (output).
GPIO[13]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: PCIREQN[2] Alternate function: PCI bus request 2 (output).
GPIO[14]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: PCIGNTN[2] Alternate function: PCI bus grant 2 (output).

Table 1 Pin Description (Part 2 of 6)

Signal	Type	Name/Description
GPIO[15]	I/O	**General Purpose I/O.** This pin can be configured as a general purpose I/O pin. Alternate function pin name: PCIMUINTN Alternate function: PCI Messaging unit interrupt output.
Serial Interface		
SCK	I/O	**Serial Clock**. This signal is used as the serial SPI clock output. This pin may be used as a bit input/output port.
SDI	I/O	**Serial Data Input**. This signal is used to shift in serial SPI data. This pin may be used as a bit input/output port.
SDO	I/O	**Serial Data Output**. This signal is used to shift out serial SPI data. This pin may be used as a bit input/output port.
PCI Bus		
PCIAD[31:0]	I/O	**PCI Multiplexed Address/Data Bus**. Address is driven by a bus master during initial PCIFRAMEN assertion. Data is then driven by the bus master during writes or by the bus target during reads.
PCICBEN[3:0]	I/O	**PCI Multiplexed Command/Byte Enable Bus**. PCI command is driven by the bus master during the initial PCIFRAMEN assertion. Byte enables are driven by the bus master during subsequent data phase(s).
PCICLK	I	**PCI Clock**. Clock used for all PCI bus transactions.
PCIDEVSELN	I/O	**PCI Device Select**. This signal is driven by a bus target to indicate that the target has decoded the address as one of its own address spaces.
PCIFRAMEN	I/O	**PCI Frame**. Driven by a bus master. Assertion indicates the beginning of a bus transaction. Negation indicates the last data.
PCIGNTN[1:0]	I/O	**PCI Bus Grant**. **In PCI host mode with internal arbiter:** The assertion of these signals indicates to the agent that the internal RC32365 arbiter has granted the agent access to the PCI bus. **In PCI host mode with external arbiter:** PCIGNTN[0]: asserted by an external arbiter to indicate to the RC32365 that access to the PCI bus has been granted. PCIGNTN[1]: unused and driven high. **In PCI satellite mode:** PCIGNTN[0]: this signal is asserted by an external arbiter to indicate to the RC32365 that access to the PCI bus has been granted. PCIGNTN[1]: this signal takes on the alternate function of PCIEECS and is used as a PCI Serial EEPROM chip select.
PCIIRDYN	I/O	**PCI Initiator Ready**. Driven by the bus master to indicate that the current data can complete.
PCILOCKN	I/O	**PCI Lock**. This signal is asserted by an external bus master to indicate that an exclusive operation is occurring.
PCIPAR	I/O	**PCI Parity**. Even parity of the PCIAD[31:0] bus. Driven by the bus master during address and write data phases. Driven by the bus target during the read data phases.
PCIPERRN	I/O	**PCI Parity Error**. This signal is asserted by the receiving bus agent 2 clocks after the data is received if a parity error is detected.

Table 1 Pin Description (Part 3 of 6)

Signal	Type	Name/Description
PCIREQN[1:0]	I/O	**PCI Bus Request.** **In PCI host mode with internal arbiter:** These signals are inputs whose assertion indicates to the internal RC32365 arbiter that an agent desires ownership of the PCI bus. **In PCI host mode with external arbiter:** PCIREQN[0]: asserted by the RC32365 to request ownership of the PCI bus. PCIREQN[1]: unused and driven high. **In PCI satellite mode:** PCIREQN[0]: this signal is asserted by the RC32365 to request ownership of the PCI bus. PCIREQN[1]: function changes to PCIIDSEL and is used as a chip select during configuration read and write transactions.
PCIRSTN	I/O	**PCI Reset**. In host mode, this signal is asserted by the RC32365 to generate a PCI reset. In satellite mode, assertion of this signal initiates a warm reset.
PCISERRN	I/O	**PCI System Error**. This signal is driven by an agent to indicate an address parity error, data parity error during a special cycle command, or any other system error. Requires an external pull-up.
PCISTOPN	I/O	**PCI Stop**. Driven by the bus target to terminate the current bus transaction. For example, to indicate a retry.
PCITRDYN	I/O	**PCI Target Ready**. Driven by the bus target to indicate that the current data can complete.
Ethernet Interface		
MII0CL	I	**Ethernet 0 MII Collision Detected.** This signal is asserted by the ethernet PHY when a collision is detected.
MII0CRS	I	**Ethernet 0 MII Carrier Sense.** This signal is asserted by the ethernet PHY when either the transmit or receive medium is not idle.
MII0RXCLK	I	**Ethernet 0 MII Receive Clock.** This clock is a continuous clock that provides a timing reference for the reception of data.
MII0RXD[3:0]	I	**Ethernet 0 MII Receive Data.** This nibble wide data bus contains the data received by the ethernet PHY.
MII0RXDV	I	**Ethernet 0 MII Receive Data Valid.** The assertion of this signal indicates that valid receive data is in the MII receive data bus.
MII0RXER	I	**Ethernet 0 MII Receive Error.** The assertion of this signal indicates that an error was detected somewhere in the ethernet frame currently being sent in the MII receive data bus.
MII0TXCLK	I	**Ethernet 0 MII Transmit Clock.** This clock is a continuous clock that provides a timing reference for the transfer of transmit data.
MII0TXD[3:0]	O	**Ethernet 0 MII Transmit Data.** This nibble wide data bus contains the data to be transmitted.
MII0TXENP	O	**Ethernet 0 MII Transmit Enable.** The assertion of this signal indicates that data is present on the MII for transmission.
MII0TXER	O	**Ethernet 0 MII Transmit Coding Error.** When this signal is asserted together with MIITXENP, the ethernet PHY will transmit symbols which are not valid data or delimiters.
MII1CL	I	**Ethernet 1 MII Collision Detected.** This signal is asserted by the ethernet PHY when a collision is detected.
MII1CRS	I	**Ethernet 1 MII Carrier Sense.** This signal is asserted by the ethernet PHY when either the transmit or receive medium is not idle.
MII1RXCLK	I	**Ethernet 1 MII Receive Clock.** This clock is a continuous clock that provides a timing reference for the reception of data.

Table 1 Pin Description (Part 4 of 6)

Signal	Type	Name/Description
MII1RXD[3:0]	I	**Ethernet 1 MII Receive Data.** This nibble wide data bus contains the data received by the ethernet PHY.
MII1RXDV	I	**Ethernet 1 MII Receive Data Valid.** The assertion of this signal indicates that valid receive data is in the MII receive data bus.
MII1RXER	I	**Ethernet 1 MII Receive Error.** The assertion of this signal indicates that an error was detected somewhere in the ethernet frame currently being sent in the MII receive data bus.
MII1TXCLK	I	**Ethernet 1 MII Transmit Clock.** This clock is a continuous clock that provides a timing reference for the transfer of transmit data.
MII1TXD[3:0]	O	**Ethernet 1 MII Transmit Data.** This nibble wide data bus contains the data to be transmitted.
MII1TXENP	O	**Ethernet 1 MII Transmit Enable.** The assertion of this signal indicates that data is present on the MII for transmission.
MII1TXER	O	**Ethernet 1 MII Transmit Coding Error.** When this signal is asserted together with MIITXENP, the ethernet PHY will transmit symbols which are not valid data or delimiters.
MIIMDC	O	**MII Management Data Clock.** This signal is used as a timing reference for transmission of data on the management interface.
MIIMDIO	I/O	**MII Management Data.** This bidirectional signal is used to transfer data between the station management entity and the ethernet PHY.
EJTAG / JTAG		
JTAG_TMS	I	**JTAG Mode**. The value on this signal controls the test mode select of the boundary scan logic or JTAG Controller. When using the EJTAG debug interface, this pin should be left disconnected (since there is an internal pull-up) or driven high.
EJTAG_TMS	I	**EJTAG Mode**. The value on this signal controls the test mode select of the EJTAG Controller. When using the JTAG boundary scan, this pin should be left disconnected (since there is an internal pull-up) or driven high.
JTAG_TRST_N	I	**JTAG Reset**. This active low signal asynchronously resets the boundary scan logic, JTAG TAP Controller, and the EJTAG Debug TAP Controller. An external pull-up on the board is recommended to meet the JTAG specification in cases where the tester can access this signal. However, for systems running in functional mode, one of the following should occur: 1) actively drive this signal low with control logic 2) statically drive this signal low with an external pull-down on the board 3) clock JTAG_TCK while holding EJTAG_TMS and/or JTAG_TMS high.
JTAG_TCK	I	**JTAG Clock**. This is an input test clock used to clock the shifting of data into or out of the boundary scan logic, JTAG Controller, or the EJTAG Controller. JTAG_TCK is independent of the system and the processor clock with a nominal 50% duty cycle.
JTAG_TDO	O	**JTAG Data Output**. This is the serial data shifted out from the boundary scan logic, JTAG Controller, or the EJTAG Controller. When no data is being shifted out, this signal is tri-stated.
JTAG_TDI	I	**JTAG Data Input**. This is the serial data input to the boundary scan logic, JTAG Controller, or the EJTAG Controller.

Table 1 Pin Description (Part 5 of 6)

Signal	Type	Name/Description
Miscellaneous		
CLK	I	**Master Clock.** This is the master clock input. The processor frequency is a multiple of this clock frequency. This clock is used as the system clock for all memory and peripheral bus operations except those associated with SDRAMs.
COLDRSTN	I	**Cold Reset.** The assertion of this signal initiates a cold reset. This causes the processor state to be initialized, boot configuration to be loaded, and the internal PLL to lock onto the master clock (CLK).
RSTN	I/O	**Reset.** The assertion of this bidirectional signal initiates a warm reset. This signal is asserted by the RC32365 during a warm reset. It can also be asserted by an external device to force the RC32365 to take a warm reset exception.

Table 1 Pin Description (Part 6 of 6)

Pin Characteristics

Pin Name	Type	Buffer	I/O Type	Internal Resistor	External Resistor[1]
Memory and Peripheral Bus					
BDIRN	O	LVTTL	High Drive		
BOEN[1:0]	O	LVTTL	High Drive		
BWEN[3:0]	O	LVTTL	High Drive		
CSN[5:0]	O	LVTTL	High Drive		
MADDR[21:0]	O	LVTTL	High Drive		
MDATA[31:0]	I/O	LVTTL	High Drive		
OEN	O	LVTTL	High Drive		
RWN	O	LVTTL	High Drive		
WAITACKN	I	LVTTL	STI[2]	pull-up	
RASN	O	LVTTL	High Drive		
CASN	O	LVTTL	High Drive		
SDCSN[1:0]	O	LVTTL	High Drive		
SDWEN	O	LVTTL	High Drive		
SDCLKOUT	O	LVTTL	High Drive		
SDCLKINP	I	LVTTL	STI	pull-up	
General Purpose I/O					
GPIO[15:13]	I/O	PCI	PCI		
GPIO[12:0]	I/O	LVTTL	Low Drive	pull-up	
Serial Interface					
SCK	I/O	LVTTL	Low Drive	pull-up	pull-up on board
SDI	I/O	LVTTL	Low Drive	pull-up	pull-up on board
SDO	I/O	LVTTL	Low Drive	pull-up	pull-up on board
PCI Bus Interface					
PCIAD[31:0]	I/O	PCI	PCI		
PCICBEN[3:0]	I/O	PCI	PCI		
PCICLK	I	PCI	PCI		
PCIDEVSELN	I/O	PCI	PCI		pull-up on board

Table 2 Pin Characteristics (Part 1 of 2)

Pin Name	Type	Buffer	I/O Type	Internal Resistor	External Resistor[1]
PCIFRAMEN	I/O	PCI	PCI		pull-up on board
PCIGNTN[1:0]	I/O	PCI	PCI		pull-up on board
PCIIRDYN	I/O	PCI	PCI		pull-up on board
PCILOCKN	I/O	PCI	PCI		
PCIPAR	I/O	PCI	PCI		
PCIPERRN	I/O	PCI	PCI		
PCIREQN[1:0]	I/O	PCI	PCI		pull-up on board
PCIRSTN	I/O	PCI	PCI		pull-down on board
PCISERRN	I/O	PCI	Open Collector; PCI		pull-up on board
PCISTOPN	I/O	PCI	PCI		pull-up on board
PCITRDYN	I/O	PCI	PCI		pull-up on board
Ethernet Interfaces					
MII0CL	I	LVTTL	STI	pull-up	
MII0CRS	I	LVTTL	STI	pull-up	
MII0RXCLK	I	LVTTL	STI	pull-up	
MII0RXD[3:0]	I	LVTTL	STI	pull-up	
MII0RXDV	I	LVTTL	STI	pull-up	
MII0RXER	I	LVTTL	STI	pull-up	
MII0TXCLK	I	LVTTL	STI	pull-up	
MII0TXD[3:0]	O	LVTTL	Low Drive		
MII0TXENP	O	LVTTL	Low Drive		
MII0TXER	O	LVTTL	Low Drive		
MII1CL	I	LVTTL	STI	pull-up	
MII1CRS	I	LVTTL	STI	pull-up	
MII1RXCLK	I	LVTTL	STI	pull-up	
MII1RXD[3:0]	I	LVTTL	STI	pull-up	
MII1RXDV	I	LVTTL	STI	pull-up	
MII1RXER	I	LVTTL	STI	pull-up	
MII1TXCLK	I	LVTTL	STI	pull-up	
MII1TXD[3:0]	O	LVTTL	Low Drive		
MII1TXENP	O	LVTTL	Low Drive		
MII1TXER	O	LVTTL	Low Drive		
MIIMDC	O	LVTTL	Low Drive		
MIIMDIO	I/O	LVTTL	Low Drive	pull-up	
EJTAG / JTAG					
JTAG_TMS	I	LVTTL	STI	pull-up	See Chapters 22 and 23 of the RC32365 User Reference Manual
EJTAG_TMS	I	LVTTL	STI	pull-up	
JTAG_TRST_N	I	LVTTL	STI	pull-up	
JTAG_TCK	I	LVTTL	STI	pull-up	
JTAG_TDO	O	LVTTL	Low Drive		
JTAG_TDI	I	LVTTL	STI	pull-up	
Miscellaneous					
CLK	I	LVTTL	STI		
COLDRSTN	I	LVTTL	STI		
RSTN	I/O	LVTTL	Low Drive / STI	pull-up	pull-up on board

Table 2 Pin Characteristics (Part 2 of 2)

1. External pull-up required in most system applications. Some applications may require additional pull-ups not identified in this table.

2. Schmidt Trigger Input (STI).

Boot Configuration Vector

The boot configuration vector is read into the RC32365 during cold reset. The vector defines parameters in the RC32365 that are essential to operation when cold reset is complete.

The encoding of boot configuration vector is described in Table 3, and the vector input is illustrated in Figure 4.

Signal	Name/Description
MDATA[2:0]	**CPU Clock Multiplier**. This field specifies the value by which the PLL multiplies the master clock input (CLK) to obtain the processor clock frequency (PCLK). 0x0 - Multiply by 2 0x1 - 0x7 — Reserved
MDATA[3]	**Endian.** This bit specifies the endianness. 0x0 - little endian 0x1 - big endian
MDATA[4]	**Reserved.** This pin may be driven high or low during boot configuration and its state is recorded in the Boot Configuration Vector (BCV) field of the BCV register. This reserved bit may be used to pass boot configuration parameters to software.
MDATA[6:5]	**Boot Device Width**. This field specifies the width of the boot device (i.e., Device 0). 0x0 - 8-bit boot device width 0x1 - 16-bit boot device width 0x2 - 32-bit boot device width 0x3 - reserved
MDATA[7]	**Reset Mode**. This bit specifies the length of time the RSTN signal is driven. 0x0 - Normal reset: RSTN driven for minimum of 4096 clock cycles 0x1 - reserved
MDATA[8]	**Disable Watchdog Timer**. When this bit is set, the watchdog timer is disabled following a cold reset. 0x0 - Watchdog timer is enabled 0x1 - Watchdog timer is disabled
MDATA[11:9]	**PCI Mode**. This bit controls the operating mode of the PCI bus interface. The initial value of the EN bit in the PCIC register is determined by the PCI mode. 0x0 - Disabled (EN initial value is zero) 0x1 - PCI satellite mode with PCI target not ready (EN initial value is one) 0x2 - PCI satellite mode with suspended CPU execution (EN initial value is one) 0x3 - PCI host mode with external arbiter (EN initial value is zero) 0x4 - PCI host mode with internal arbiter using fixed priority arbitration algorithm (EN initial value is zero) 0x5 - PCI host mode with internal arbiter using round robin arbitration algorithm (EN initial value is zero) 0x6 - *reserved* 0x7 - *reserved*
MDATA[15:12]	**Reserved**. These pins may be driven high or low during boot configuration and their state is recorded in the Boot Configuration Vector (BCV) field of the BCV register. These reserved bits may be used to pass boot configuration parameters to software.

Table 3 Boot Configuration Vector Encoding

Logic Diagram

The following Logic Diagram shows the primary pin functions of the RC32365.

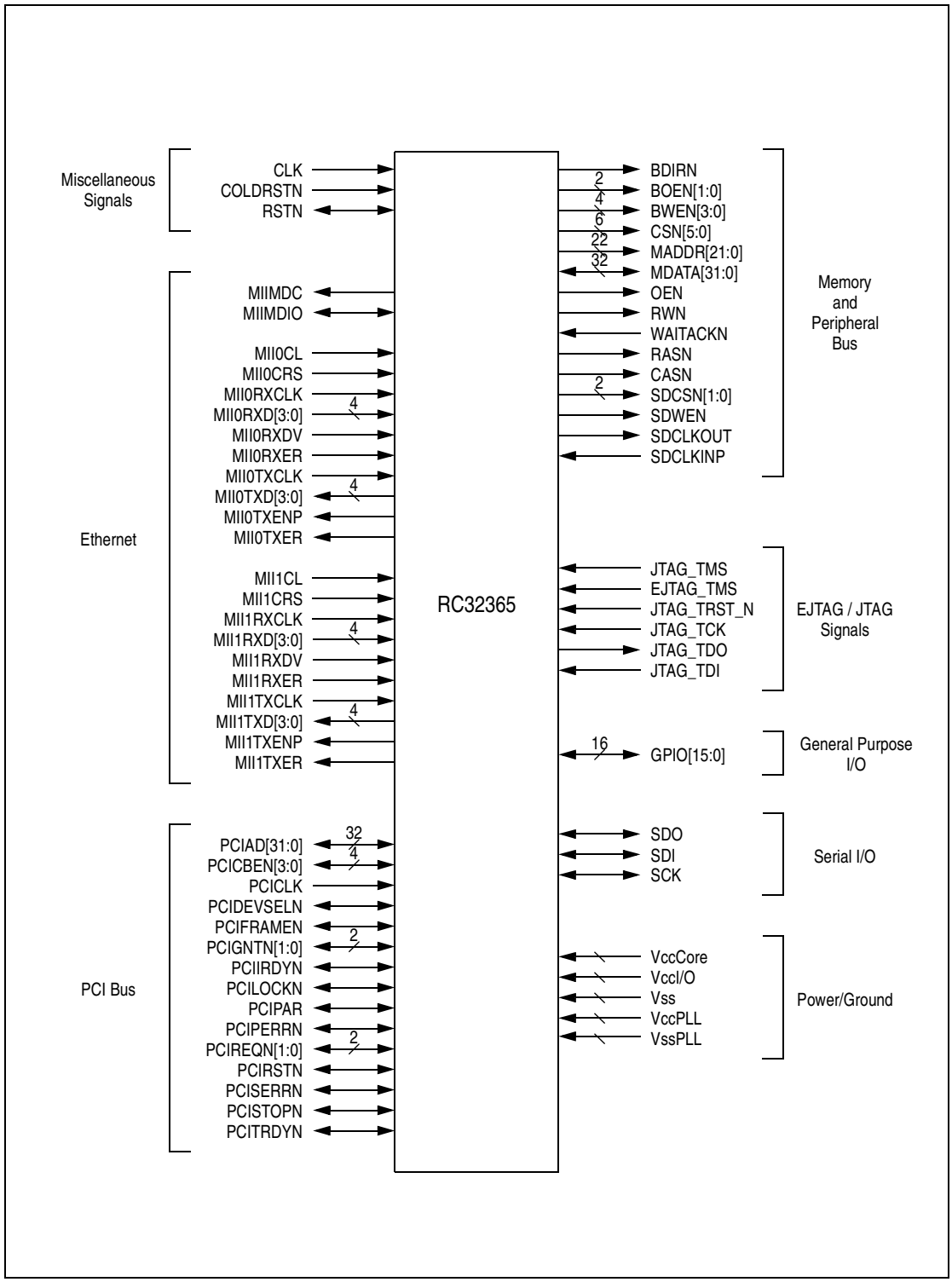


Figure 1 RC32365 Logic Diagram

AC Timing Definitions

Below are examples of the AC timing characteristics used throughout this document.

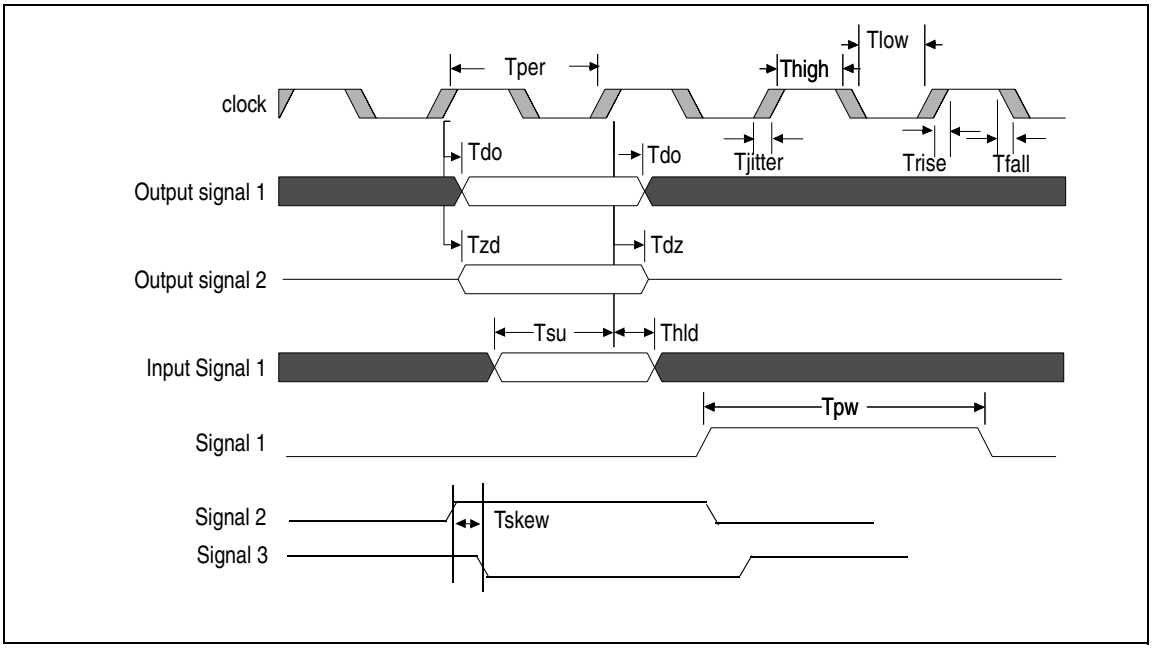


Figure 2 AC Timing Definitions Waveform

Symbol	Definition
Tper	Clock period.
Tlow	Clock low. Amount of time the clock is low in one clock period.
Thigh	Clock high. Amount of time the clock is high in one clock period.
Trise	Rise time. Low to high transition time.
Tfall	Fall time. High to low transition time.
Tjitter	Jitter. Amount of time the reference clock (or signal) edge can vary on either the rising or falling edges.
Tdo	Data out. Amount of time after the reference clock edge that the output will become valid. The minimum time represents the data output hold. The maximum time represents the earliest time the designer can use the data.
Tzd	Z state to data valid. Amount of time after the reference clock edge that the tri-stated output takes to become valid.
Tdz	Data valid to Z state. Amount of time after the reference clock edge that the valid output takes to become tri-stated.
Tsu	Input set-up. Amount of time before the reference clock edge that the input must be valid.
Thld	Input hold. Amount of time after the reference clock edge that the input must remain valid.
Tpw	Pulse width. Amount of time the input or output is active for asynchronous signals.
Tslew	Slew rate. The rise or fall rate for a signal to go from a high to low, or low to high.
X(clock)	Timing value. This notation represents a value of 'X' multiplied by the clock time period of the specified clock. Using 5(CLK) as an example: X = 5 and the oscillator clock (CLK) = 25MHz, then the timing value is 200.
Tskew	Skew. The amount of time two signal edges deviate from one another.

Table 4 AC Timing Definitions

Clock Parameters

The values given below are based on systems running at recommended supply voltages and operating temperatures, as shown in Tables 14 and 15.

Parameter	Symbol	Reference Edge	150MHz		Units	Timing Diagram Reference
			Min	Max		
PCLK[1]	Frequency	none	100	150	MHz	See Figure 3
CLK[2,3]	Frequency	none	50	75	MHz	
	Tper_5a		13.3	20	ns	
	Thigh_5a, Tlow_5a		40	60	% of Tper_5a	
	Trise_5a, Tfall_5a		—	3.0	ns	
	Tjitter_5a		—	± 250	ps	

Table 5 RC32365 Clock Parameters

1. The CPU pipeline clock (PCLK) speed is selected during cold reset by the boot configuration vector (see Table 3).
2. Ethernet clock (MIIxRXCLK and MIIxTXCLK) frequency must be less than or equal to 1/2 CLK frequency.
3. PCI clock (PCICLK) frequency must be less than or equal to two times CLK.

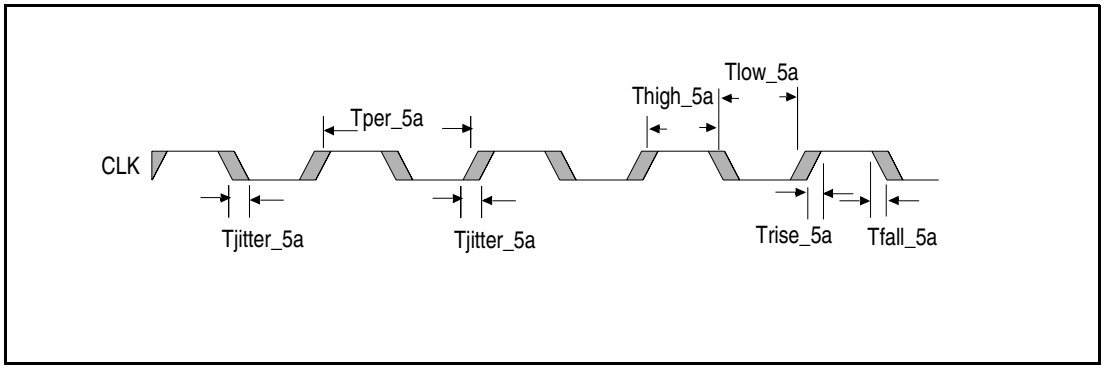


Figure 3 Clock Parameters Waveform

AC Timing Characteristics

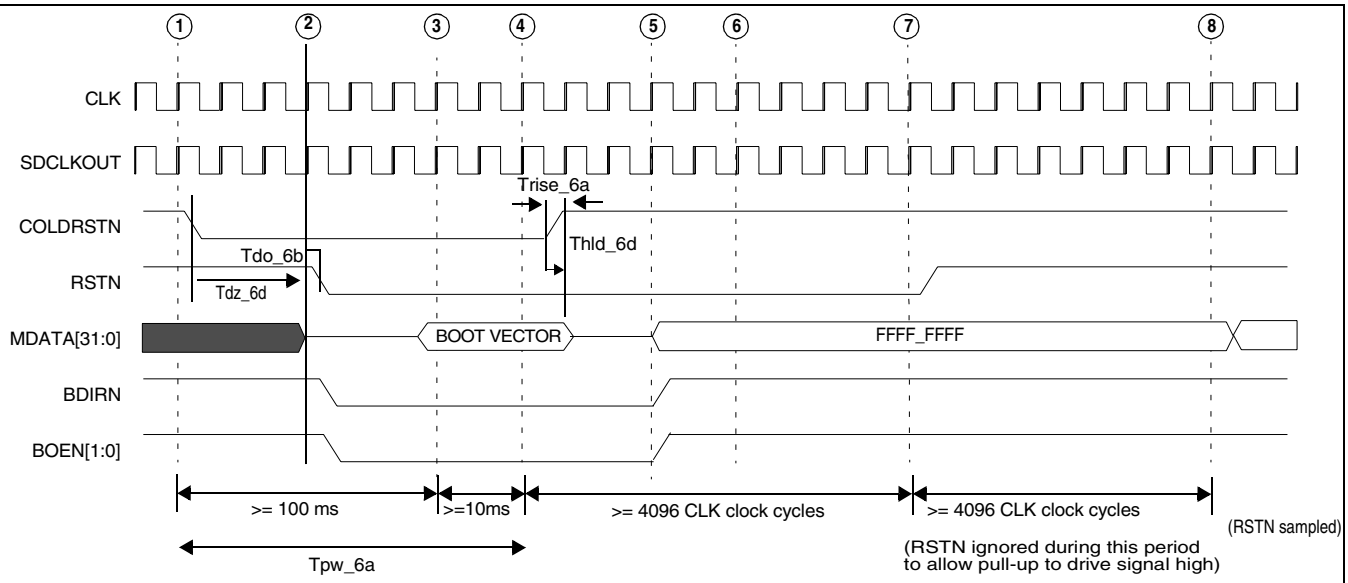
The values given below are based on systems running at recommended operating supply voltages and temperatures as shown in Tables 14 and 15.

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
Reset and System							
COLDRSTN	Tpw_6a[1]	none	110	—	ms	Cold reset	See Figures 4 and 5
	Trise_6a		—	5.0	ns	Cold reset	
RSTN[2] (output)	Tdo_6b	CLK rising	2.0	9.0	ns	Cold reset	
RSTN[2] (input)	Tpw_6c[1]	none	2(CLK)	—	ns	Cold reset	
MDATA[15:0] Boot Configuration Vector	Thld_6d	COLDRSTN rising	3.0	—	ns	Cold reset	
	Tdz_6d[1]	COLDRSTN falling	—	2(CLK)	ns	Cold reset	
	Tdz_6d[1]	RSTN falling	—	2(CLK)	ns	Warm reset	
	Tzd_6d[1]	RSTN rising	3.0	—	ns	Warm reset	

Table 6 Reset and System AC Timing Characteristics

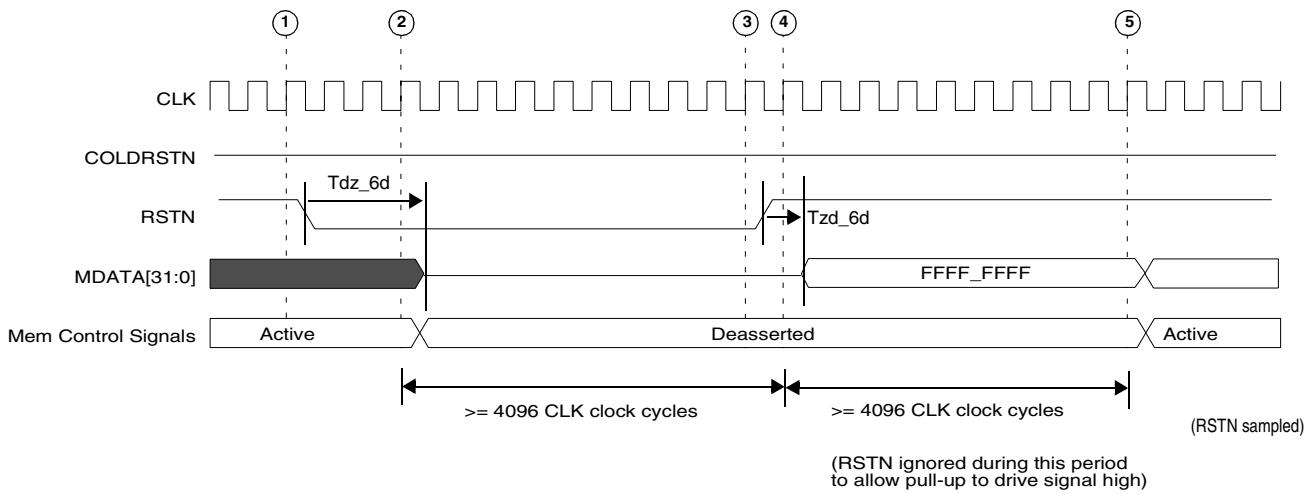
1. The values for this symbol were determined by calculation, not by testing.

2. RSTN is a bidirectional signal. It is treated as an asynchronous input.



1. COLDRSTN asserted by external logic.
2. RC32365 asserts RSTN, asserts BOEN[0] low, drives BDIRN low, and tri-states the data bus in response.
3. External logic begins driving valid boot configuration vector on the data bus, and the RC32365 starts sampling it.
4. External logic negates COLDRSTN and tri-states the boot configuration vector on MDATA[15:0]. The boot configuration vector must not be tri-stated before COLDRSTN is deasserted. The RC32365 stops sampling the boot configuration vector.
5. The RC32365 starts driving the data bus, MDATA[31:0], deasserts BOEN[0] high, and drives BDIRN high.
6. SYSCLK may be held constant after this point if Hold SYSCLK Constant is selected in the boot configuration vector.
7. RSTN negated by the RC32365.
8. CPU begins executing by taking MIPS reset exception, and the RC32365 starts sampling RSTN as a warm reset input.

Figure 4 Cold Reset AC Timing Waveform



1. Warm reset condition caused by either RSTN asserted, write to reset register, or bus transaction timer time-out. The RC32365 asserts RSTN output low in response.
2. The RC32365 tri-states the data bus, MDATA[31:0], and deasserts all memory control signals, such as RASN, CASN, RWN, OEN, etc.
3. The RC32365 deasserts RSTN.
4. The RC32365 starts driving the data bus, MDATA[31:0], again, but does not sample the RSTN input.
5. CPU begins executing by taking a MIPS soft reset exception and also starts sampling the RSTN input again.

Figure 5 Warm Reset AC Timing Waveform

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
Memory and Peripheral Bus - SDRAM Access							
MDATA[31:0]	Tsu_7a	SDCLKINP rising	1.0	—	ns		See Figures 6 and 7
	Thld_7a		1.7	—	ns		
	Tdo_7a	SDCLKOUT rising	1.2	6.0	ns		
	Tdz_7a[1]		1.2	7.0	ns		
	Tzd_7a[1]		1.2	8.0	ns		
MADDR[20:2]	Tdo_7b	SDCLKOUT rising	1.2	6.0	ns		
RASN	Tdo_7c	SDCLKOUT rising	1.2	6.0	ns		
CASN	Tdo_7d	SDCLKOUT rising	1.2	6.0	ns		
SDWEN	Tdo_7e	SDCLKOUT rising	1.2	6.0	ns		
SDCSN[1:0]	Tdo_7f	SDCLKOUT rising	1.2	6.0	ns		
BDIRN	Tdo_7g	SDCLKOUT rising	1.2	6.0	ns		
BOEN[1:0]	Tdo_7h	SDCLKOUT rising	1.2	6.0	ns		
BWEN[3:0]	Tdo_7i	SDCLKOUT rising	1.2	6.0	ns		
SDCLKINP	Tdelay_7k	SDCLKOUT rising	0.0	2.5	ns		See Figures 6 and 8
SDCKENP	Tdo_7l	SDCLKOUT rising	2.0	6.0	ns		

Table 7 Memory and Peripheral Bus AC Timing Characteristics

1. The values for this symbol were determined by calculation, not by testing.

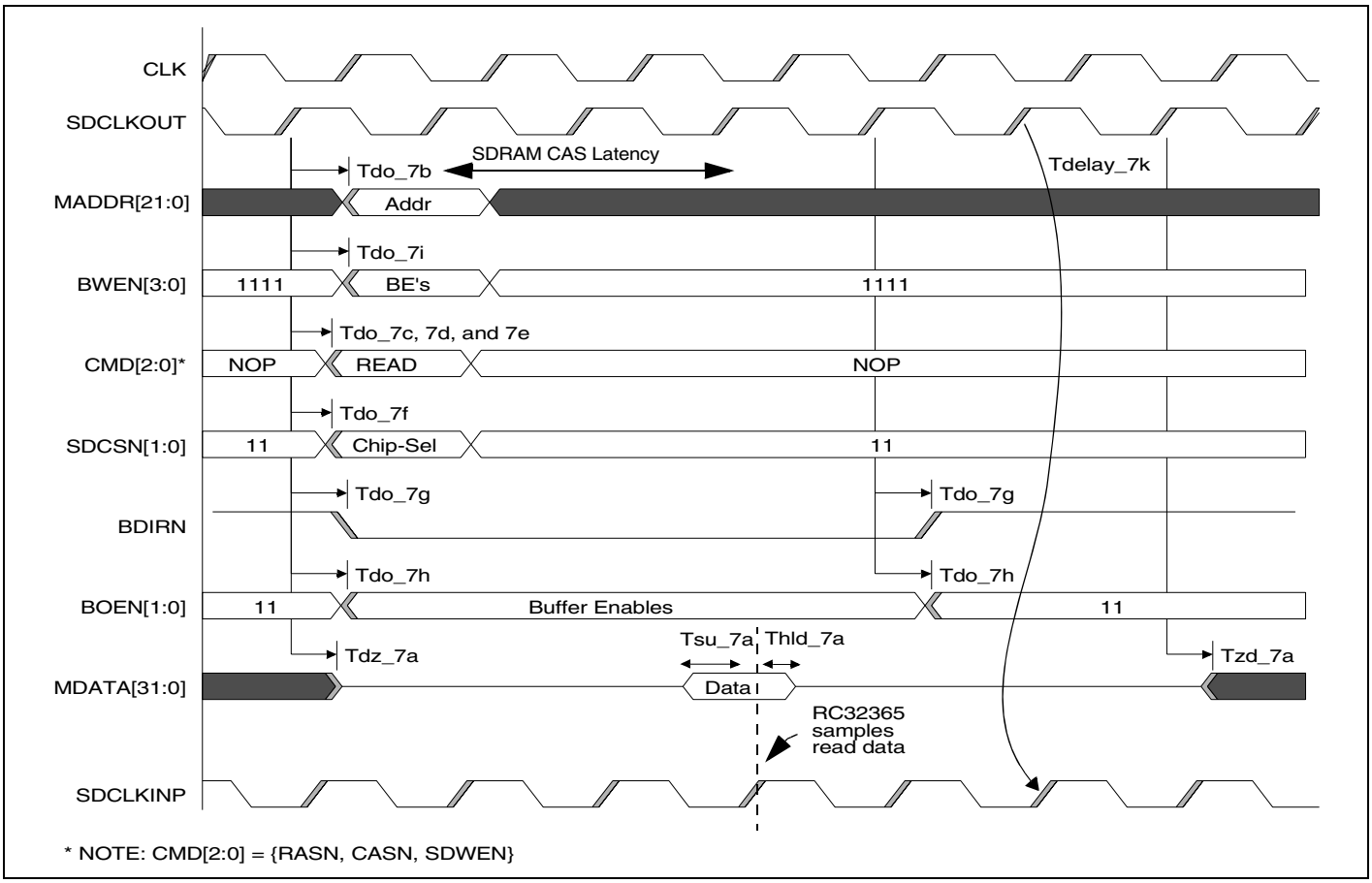


Figure 6 Memory and Peripheral Bus AC Timing Waveform - SDRAM Read Access

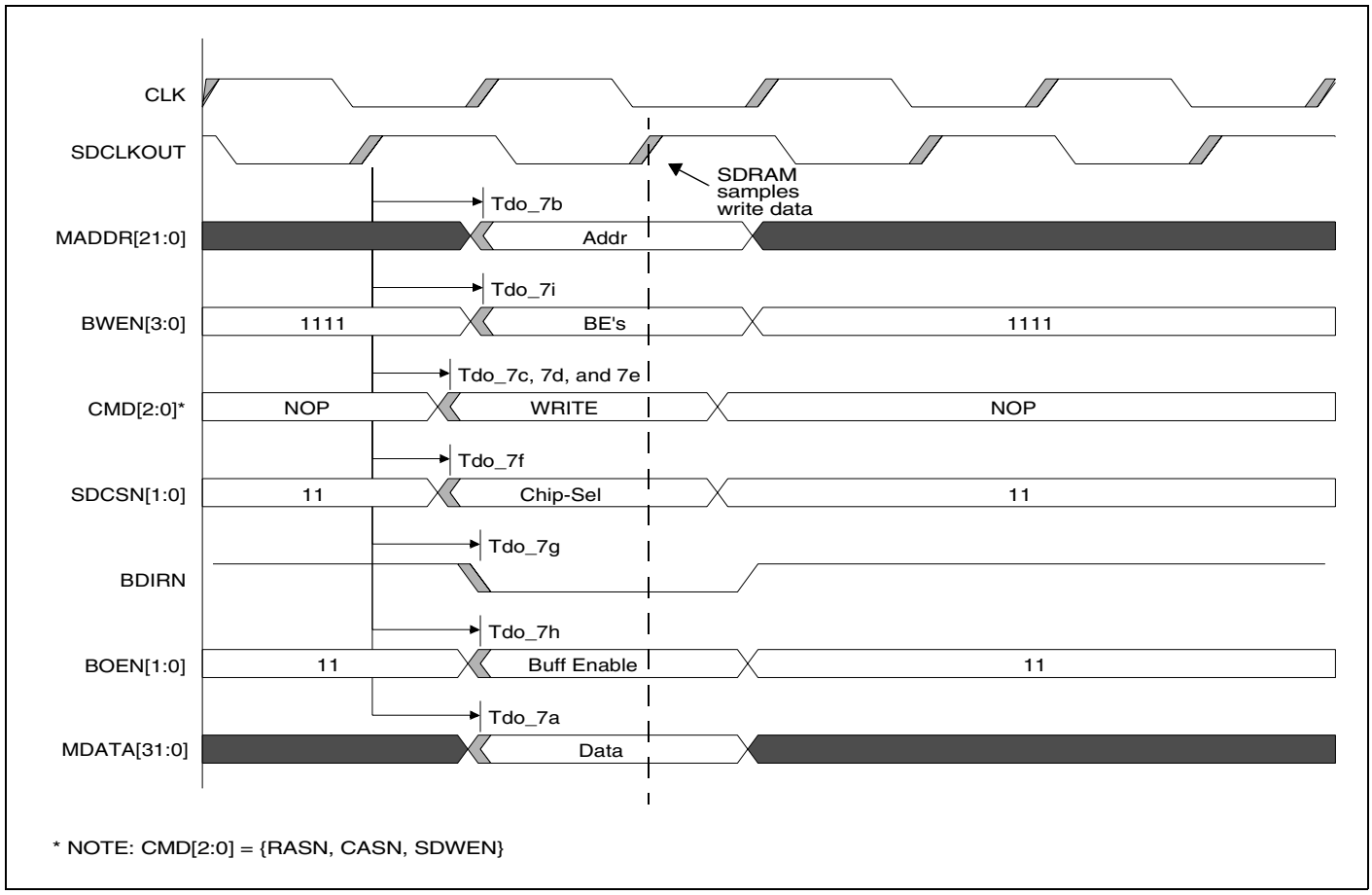


Figure 7 Memory and Peripheral Bus AC Timing Waveform - SDRAM Write Access

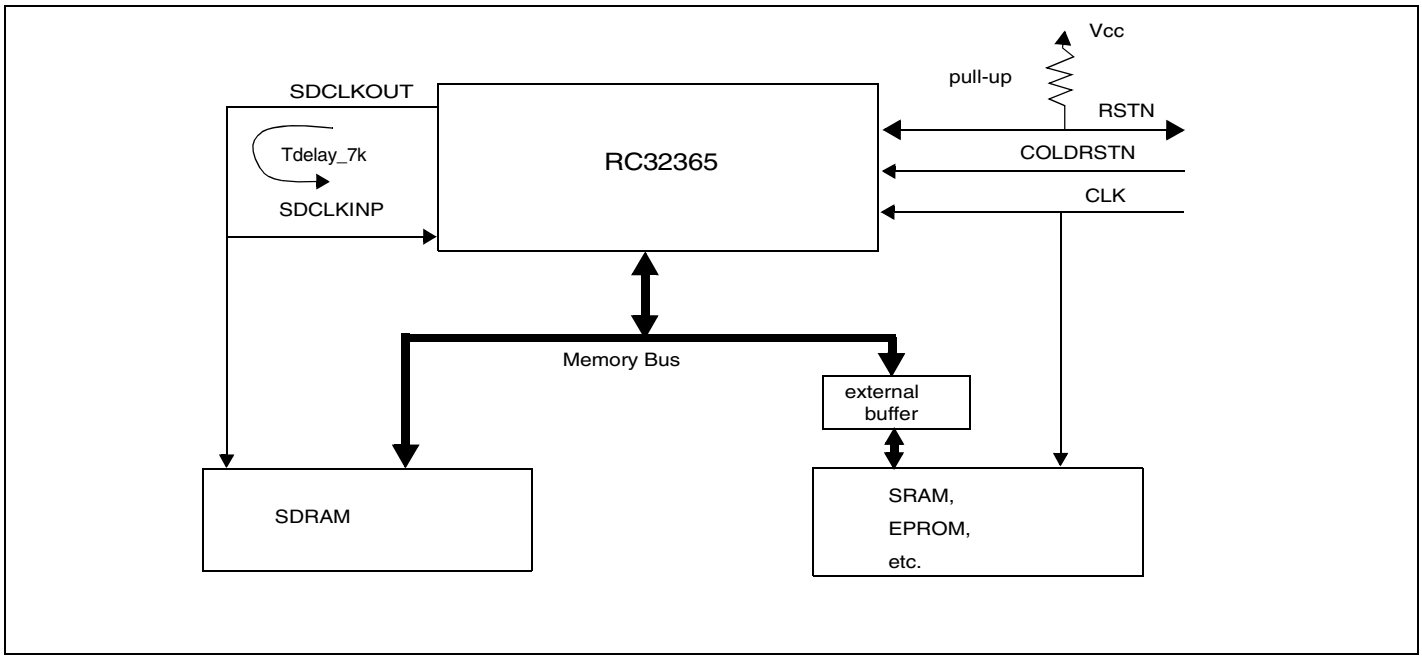


Figure 8 SDCLKOUT - SDCLKINP Relationship

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
Memory and Peripheral Bus[1] — Device Access							
MDATA[31:0]	Tsu_8a	CLK rising	2.5	—	ns		See Figures 9 and 10
	Thld_8a		1.0	—	ns		
	Tdo_8a		2.0	6.5	ns		
	Tdz_8a[2]		2.0	9.5	ns		
	Tzd_8a[2]		2.0	10.5	ns		
MADDR[21:0]	Tdo_8b	CLK rising	2.0	6.5	ns		
MADDR[25:22]	Tdo_8c	CLK rising	3.0	7.5	ns		
CSN[5:0]	Tdo_8d	CLK rising	2.0	6.5	ns		
RWN	Tdo_8e	CLK rising	2.0	6.5	ns		
OEN	Tdo_8f	CLK rising	2.0	6.5	ns		
BWEN[1:0]	Tdo_8g	CLK rising	2.0	6.5	ns		
BDIRN	Tdo_8h	CLK rising	2.0	6.5	ns		
BOEN[1:0]	Tdo_8i	CLK rising	2.0	6.5	ns		
WAITACKN[3]	Tsu_8j	CLK rising	2.0	—	ns		
	Thld_8j		0.5	—	ns		
	Tpw_8j[2]	none	2(CLK)	—	ns		

Table 8 Memory and Peripheral Bus AC Timing Characteristics — Device Access (Part 1 of 2)

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
CEN1[4], CEN2[4]	Tdo_8k	CLK rising	3.0	7.5	ns		See Figures 9 and 10 (cont.)
REGN[4]	Tdo_8l	CLK rising	3.0	7.5	ns		
IORDN[4]	Tdo_8m	CLK rising	3.0	7.5	ns		
IOWRN[4]	Tdo_8n	CLK rising	3.0	7.5	ns		

Table 8 Memory and Peripheral Bus AC Timing Characteristics — Device Access (Part 2 of 2)

1. The RC32365 provides bus turnaround cycles to prevent bus contention when going from a read to write and write to read. For example, there are no cycles where an external device and the RC32365 are both driving. See Chapter 6, Device Controller, in the RC32365 User Reference Manual.
2. The values for this symbol were determined by calculation, not by testing.
3. WAITACKN must meet the setup and hold times if it is synchronous or the minimum pulse width if it is asynchronous.
4. CEN1, CEN2, REGN, IORDN, and IOWRN are alternate functions of GPIO[12:8].

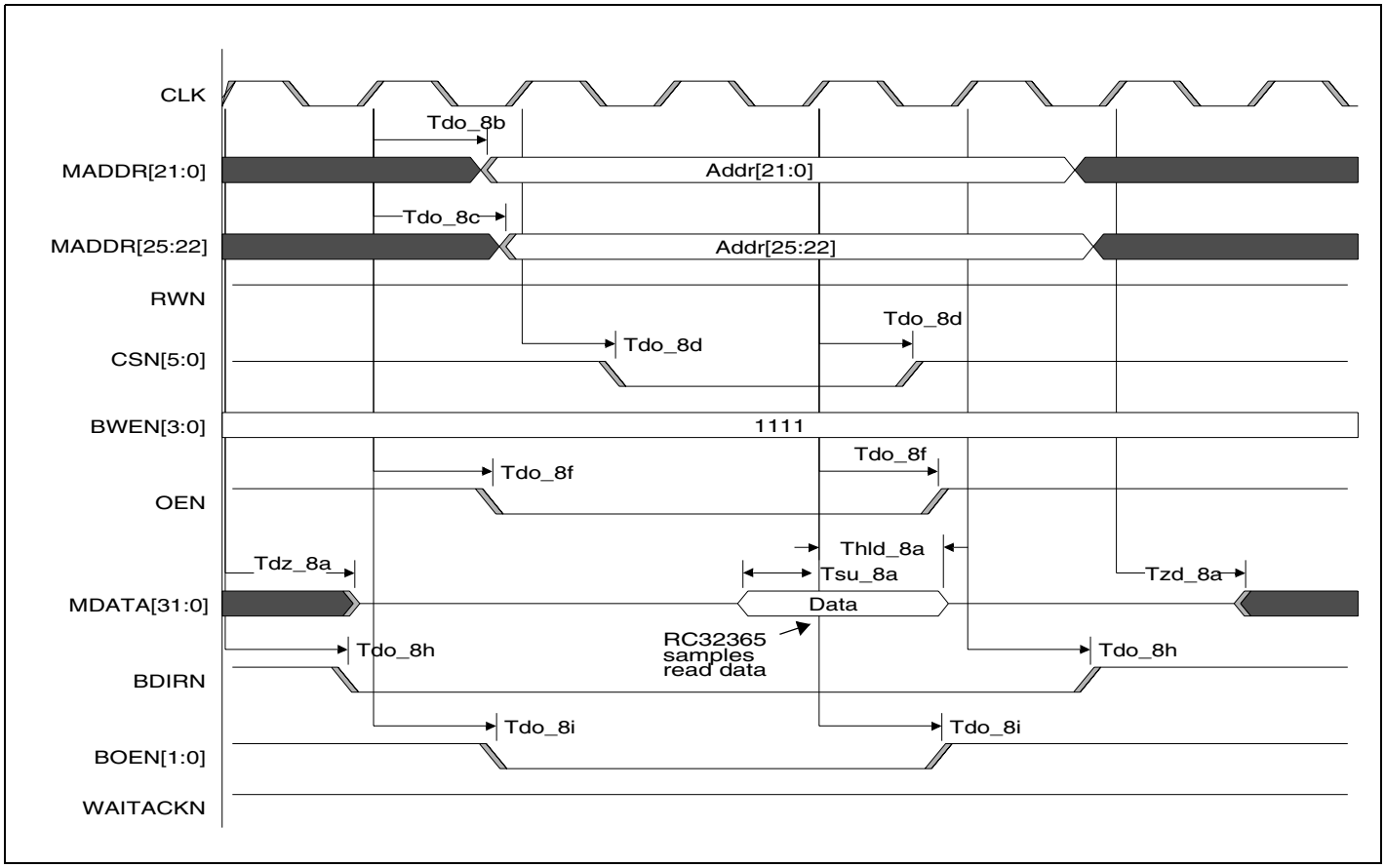


Figure 9 Memory and Peripheral Bus AC Timing Waveform - Device Read Access

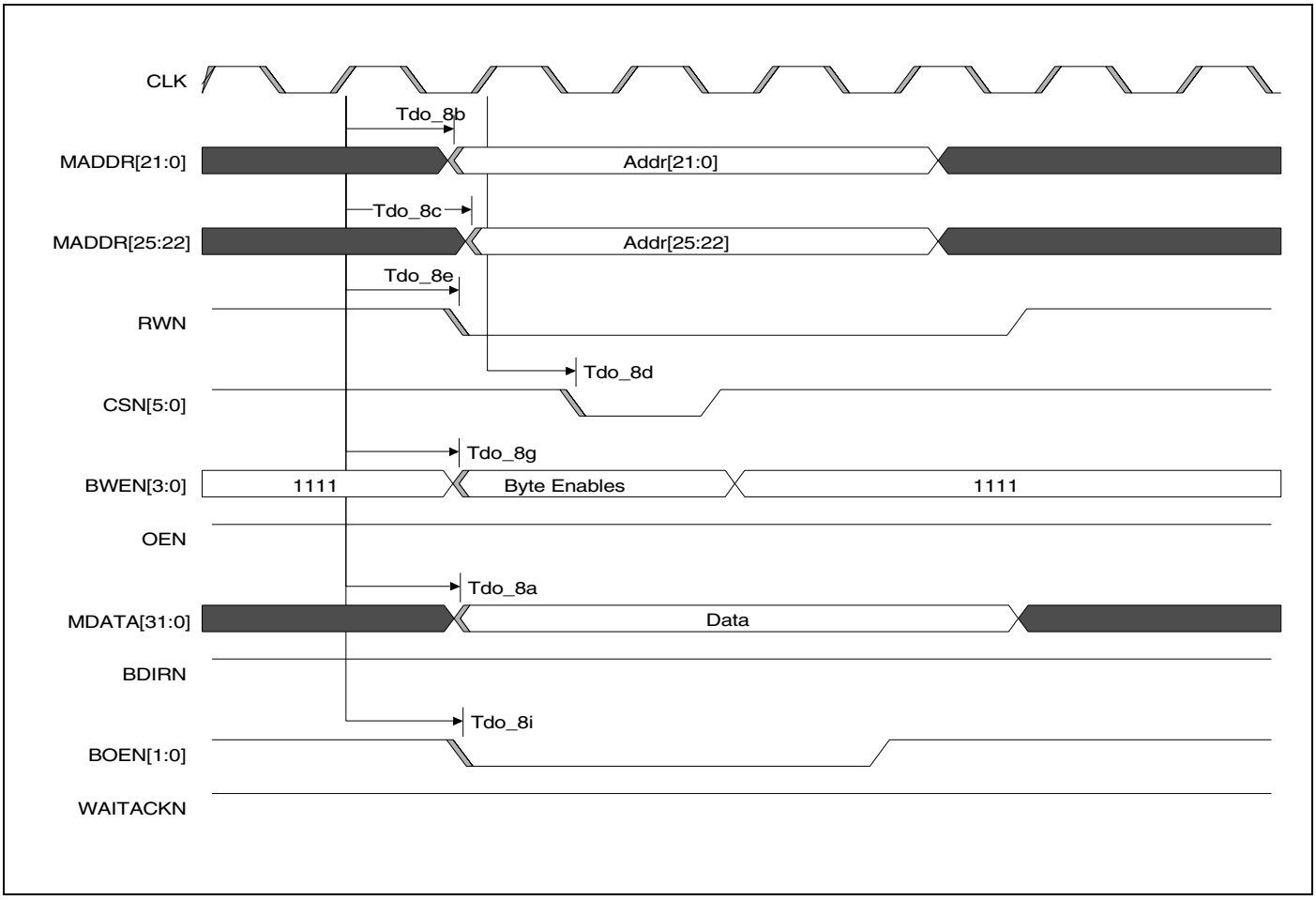


Figure 10 Memory AC and Peripheral Bus Timing Waveform - Device Write Access

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
Ethernet[1]							
MIIMDC	Tper_9a	None	53.3	—	ns		See Figure 11
	Thigh_9a, Tlow_9a		23.0	—	ns		
MIIMDIO	Tsu_9b	MIIMDC rising	10.0	—	ns		
	Thld_9b		1.0	—	ns		
	Tdo_9b		1(ICLK)	3(ICLK)	ns		
MIIxRXCLK, MIIxTX-CLK[2]	Tper_9c	None	399.96	400.4	ns	10 Mbps	
	Thigh_9c, Tlow_9c		140	260	ns		
	Trise_9c, Tfall_9c		—	3.0	ns		
MIIxRXCLK, MIIxTXCLK[2]	Tper_9d	None	39.9	40.0	ns	100 Mbps	
	Thigh_9d, Tlow_9d		14.0	26.0	ns		
	Trise_9d, Tfall_9d		—	2.0	ns		
MIIxRXD[3:0], MIIxRXDV, MIIxRXER	Tsu_9e	MIIxRXCLK rising	3.0	—	ns		
	Thld_9e		2.0	—	ns		
MIIxTXD[3:0], MIIxTXENP, MIIxTXER	Tdo_9f	MIIxTXCLK rising	5.0	13	ns		

Table 9 Ethernet AC Timing Characteristics

1. There are two MII interfaces and the timing is the same for each. “x” represents interface 0 or 1 (For example, MIIxRXCLK can be either MII0RXCLK or MII1RXCLK).

2. The ethernet clock (MIIxRXCLK and MIIxTXCLK) frequency must be equal to or less than 1/2 CLK (MIIxRXCLK and MIIxTXCLK <= 1/2(CLK)).

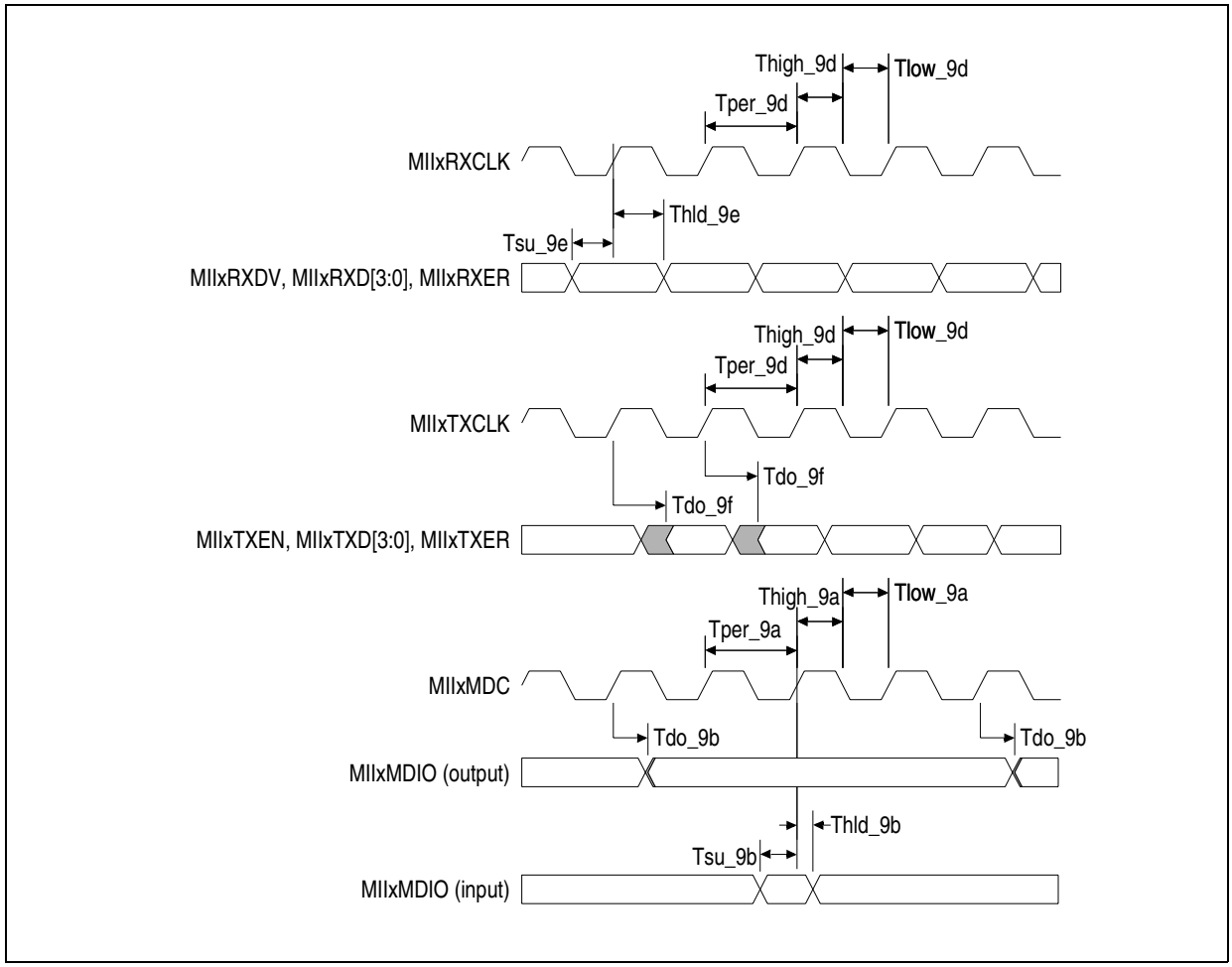


Figure 11 Ethernet AC Timing Waveform

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
PCI[1]							
PCICLK[2]	Tper_10a	none	15.0	30.0	ns	66 MHz PCI	See Figure 12
	Thigh_10a, Tlow_10a		6.0	—	ns		
	Tslew_10a		1.5	4.0	V/ns		

Table 10 PCI AC Timing Characteristics (Part 1 of 2)

Signal	Symbol	Reference Edge	150MHz		Unit	Conditions	Timing Diagram Reference
			Min	Max			
PCIAD[31:0],	Tsu_10b	PCICLK rising	3.0	—	ns		
	Thld_10b		0	—	ns		
	Tdo_10b		2.0	6.0	ns		
	Tdz_10b[3]		—	14.0	ns		
	Tzd_10b[3]		2.0	—	ns		
PCIBEN[3:0], PCIDEVSELN, PCIFRAMEN,PCIIR-DYN, PCILOCKN, PCI-PAR, PCIPERRN, PCISTOPN, PCITRDY[4]	Tsu_10b[5]	PCICLK rising	5.0	—	ns		
	Thld_10b		0	—	ns		
	Tdo_10b		1.5	6.0	ns		
PCIGNTN[2:0], PCIREQN[2:0][4,6]	Tsu_10c	PCICLK rising	5.0	—	ns		
	Thld_10c		0	—	ns		
	Tdo_10c		1.5	6.0	ns		
PCIRSTN (output)[7]	Tpw_10d[3]	None	4000 (CLK)	—	ns		See Figure 13
PCIRSTN (input)[7,8]	Tpw_10e[3]	None	2(CLK)	—	ns		See Figure 14
	Tdz_10e[3]	PCIRSTN falling	6(CLK)	—	ns		
PCISERRN[9]	Tsu_10f	PCICLK rising	3.0	—	ns		See Figure 12
	Thld_10f		0	—	ns		
	Tzd_10f[3]		2.0	6.0	ns		
PCIMUINTN[10]	Tzd_10g[3]	PCICLK rising	4.7	11.1	ns		

Table 10 PCI AC Timing Characteristics (Part 2 of 2)

1. This PCI interface conforms to the PCI Local Bus Specification, Rev 2.2 at 33MHz.

2. PCICLK must be equal to or less than two times CLK (PCICLK <= 2(CLK)).

3. The values for this symbol were determined by calculation, not by testing.

4. PCI Local Bus Specification, Rev 2.2 specifies Tval minimum = 2.0ns.

5. The 5ns minimum set-up time conforms to the PCI Local Bus Specification, Rev 2.2 at 33MHz. At 66MHz, the 5ns minimum set-up time provides a wide margin of 4ns, which is sufficient to ensure a working design at such frequency.

6. PCIGNTN[2] and PCIREQN[2] are alternate functions of GPIO[14] and GPIO[13] respectively.

7. PCIRSTN is an output in host mode and an input in satellite mode.

8. To meet the PCI delay specification from reset asserted to outputs floating, the PCI reset should be logically combined with the COLD-RSTN input, instead of input on PCIRSTN.

9. PCISERRN uses open collector I/O types.

10. PCIMUINTN is an alternate function of GPIO[15].