PRODUCT PREVIEW

80960RP
INTELLIGENT I/O MICROPROCESSOR

- **HIGH PERFORMANCE 80960JF CORE**
  - Sustained One Instruction/Clock Execution
  - 4 Kbyte Two-Way Set-Associative Instruction Cache
  - 2 Kbyte Direct-Mapped Data Cache
  - Sixteen 32-Bit Global Registers
  - Sixteen 32-Bit Local Registers
  - Programmable 8-, 16-, 32-Bit Bus Widths
  - 1 Kbyte Internal Data RAM
  - Local Register Cache (8 Available Stack Frames)
  - Two 32-Bit On-Chip Timer Units

- **DMA CONTROLLER**
  - Three Independent Channels
  - PCI Memory Controller Interface
  - 32-Bit Local Bus Addressing
  - 64-Bit PCI Bus Addressing
  - Independent Interface to Primary and Secondary PCI Buses
  - 132 Mbyte/sec Burst Transfers to PCI and Local Buses
  - Direct Addressing to and from PCI Buses
  - Unaligned Transfers Supported in Hardware
  - Two Channels Dedicated to Primary PCI Bus

- **MESSAGING UNIT**
  - Four Message Registers
  - Two Doorbell Registers
  - Four Circular Queues
  - 1004 Index Registers

- **I^2C BUS INTERFACE UNIT**
  - Serial Bus
  - Master/Slave Capabilities
  - System Management Functions

- **PCI-to-PCI BRIDGE UNIT**
  - Primary and Secondary PCI Interfaces
  - Two 64-Byte Posting Buffers
  - Supports Delayed and Posted Transactions
  - Forwards Memory, I/O and Configuration Commands from PCI Bus to PCI Bus

- **TWO ADDRESS TRANSLATION UNITS**
  - Connects Local Bus to PCI Buses
  - Supports Inbound/Outbound Address Translation
  - Supports Direct Outbound Addressing

- **SUPPORT FOR PRIVATE PCI DEVICES**

- **INTEGRATED MEMORY CONTROLLER**
  - 256 Mbytes of 32- or 36-Bit DRAM
  - Interleaved or Non-Interleaved DRAM
  - Fast Page-Mode DRAM Supported
  - Extended Data Out and Burst Extended Data Out DRAM Supported
  - Two Independent Banks for SRAM/ROM/Flash
  - 16 Mbytes/Bank of 8- or 32-Bit SRAM/ROM/Flash

- **SELECTABLE CLOCKING**
  - Halt Mode Reduces Power Consumption
  - Fully Synchronous Operation
  - Optional Asynchronous Mode for Primary PCI Bus

- **I/O APIC BUS INTERFACE UNIT**
  - Multiprocessor Interrupt Management for Intel Architecture CPU's (Pentium™ Processors)
  - Dynamic Interrupt Distribution
  - Multiple I/O Subsystem Support

May 1995 Order Number: 272737-001

CONTENTS

1.0 PURPOSE 1
2.0 OVERVIEW 1
2.1 80960RP Features 3
2.1.1 PCI-to-PCI Bridge Unit 3
2.1.2 Private PCI Devices 3
2.1.3 DMA Controller 4
2.1.4 Address Translation Unit 4
2.1.5 Messaging Unit 4
2.1.6 Integrated Memory Controller 4
2.1.7 I^2C Bus Interface Unit 4
2.1.8 I/O APIC Bus Interface Unit 4
2.2 80960JF Processor Specific Features 4
2.2.1 80960 Processor Core 4
2.2.2 Burst Bus 5
2.2.3 Timer Unit 5
2.2.4 Priority Interrupt Controller 5
2.2.5 Faults and Debugging 6
2.2.6 Low Power Operation 6
2.2.7 Test Features 6
2.2.8 Memory-Mapped Control Registers 6
2.2.9 Instructions, Data Types and Memory Addressing Modes 6
3.0 PACKAGE INFORMATION 9
3.1 Package Introduction 9
3.2 Pin Descriptions 9
3.2.1 Functional Pin Definitions 9
3.2.2 80960RP 352-Lead BGA Pinout 20
4.0 BUS FUNCTIONAL WAVEFORMS 22

Figure 1. 80960RP Block Diagram 2
Figure 2. 80960JF Core Block Diagram 3
Figure 3. 352-Lead Ball Grid Array Package - Top View 20
Figure 4. 352-Lead Ball Grid Array Package - Bottom View 21
Figure 5. Non-Burst Read and Write Transactions Without Wait States, 32-Bit 80960 Local Bus 22
Figure 6. Burst Read and Write Transactions Without Wait States, 32-Bit 80960 Local Bus 23
Figure 7. Burst Write Transactions With 2,1,1,1 Wait States, 32-Bit 80960 Local Bus 24
Figure 8. Burst Read and Write Transactions Without Wait States, 8-Bit 80960 Local Bus 25
Figure 9. Burst Read and Write Transactions With 1, 0 Wait States and Extra Tr State on Read, 16-Bit 80960 Local Bus 26
Figure 10. Bus Transactions Generated by Double Word Read Bus Request, Misaligned One Byte From Quad Word Boundary, 32-Bit 80960 Local Bus, Little Endian 27
Figure 11. HOLD/HOLDA Waveform For Bus Arbitration 28
Figure 12. PCI Delayed Read Request 29
Figure 13. PCI Delayed Read Completion 29
Figure 14. PCI Delayed Write Request 30
Figure 15. PCI Delayed Write Completion 30
Figure 16. PCI Delayed Memory Read Line Completion Up to 4 DWORD Cacheline Boundary 31
Figure 17. PCI Posted Memory Write and Invalidate Up to 4 DWORD Cacheline Boundary 31
Figure 18. PCI Delayed Write with Target-Abort 32
Figure 19. PCI Delayed Read with Target-Abort 32
Figure 20. PCI Three Cycle Bridge Latency 33
Figure 21. 32-Bit Bus, Burst Flash Memory, Read Access with 2,1,1,1 Wait States 34
Figure 22. 32-Bit Bus, Burst Flash Memory Write Access with 2,1,1,1, Wait States 35
Figure 23. 32-Bit Bus, SRAM Read Accesses with 0 Wait States 36
Figure 24. 32-Bit Bus, SRAM Write Access With 1,1,1,1 Wait States 37
Figure 25. 32-Bit Bus, Write Access With 1 Wait State and Extended MWE3:0# 38
Figure 26. Fast Page-Mode DRAM, Read Cycle, CAS# Characteristics 39
Figure 27. Fast Page-Mode DRAM, Write Cycle, CAS# Characteristics 39
Figure 28. EDO DRAM, Read Cycle, CAS# Characteristics 40
Figure 29. EDO DRAM, Write Cycle, CAS# Characteristics 40
Figure 30. BEDO DRAM, Read Cycle, CAS# Characteristics 41
Figure 31. BEDO DRAM, Write Cycle, CAS# Characteristics 41
Figure 32. FPM DRAM System Read Access, Non-Interleaved, 3,1,1,1, Wait States 42
Figure 33. FPM DRAM System Write Access, Non-Interleaved, 2,1,1,1 Wait States 43
Figure 34. FPM DRAM System Read Access, Interleaved, 2,0,0,0 Wait States (Even Bank Accessed First) 44
Figure 35. FPM DRAM System Read Access, Interleaved, 2,0,0,0 Wait States (Odd Bank Accessed First) 45
Figure 36. FPM DRAM System Write Access, Interleaved, 1,0,0,0 Wait States (1 Extra Recovery State) 46

Figure 37. EDO DRAM System Read Access, 2,0,0,0, Wait States 47
Figure 38. EDO DRAM System Write Access, 1,0,0,0 Wait States 48
Figure 39. BEDO DRAM System Read Access, 2,0,0,0, Wait States 49
Figure 40. BEDO DRAM System Write Access, 1,0,0,0 Wait States 50
Figure 41. BEDO WBCR Program Cycle 51

Table 1. 80960JF Instruction Set 8
Table 2. Pin Description Nomenclature 9
Table 3. Pin Description — 80960 Processor Signals 10
Table 4. Pin Description — Processor Control Signals, Test Signals and Power 13
Table 5. Pin Description — Interrupt Unit Signals 14
Table 6. Pin Description— PCI Signals 14
Table 7. 80960RP Pin Description— Memory Controller Signals 17
Table 8. Pin Description— 80960RP Processor Signals 19

1.0 PURPOSE

This document provides a preview of the 80960RP microprocessor. Future editions will provide targeted electrical characteristics and package pinout information. Detailed functional descriptions other than parametric performance will be published in the *i960® RP Microprocessor User's Guide* (272736).

2.0 OVERVIEW

The 80960RP processor is a multi-function device that integrates an 80960JF processor, PCI-to-PCI bus bridge, PCI to 80960 Address Translation Unit, Direct Memory Access (DMA) controller, memory controller, secondary PCI bus arbitration unit, I^2C bus interface unit, and I/O APIC Bus interface unit into a single system. It is an integrated processor that addresses the needs of intelligent I/O applications and helps reduce intelligent I/O system costs.

The PCI bus is an industry standard, high performance, low latency system bus. The PCI-to-PCI bridge provides a connection path between two independent 32-bit PCI buses and provides the ability to overcome PCI electrical loading limits. The addition of the i960 JF processor brings intelligence to the PCI bus bridge. Figure 1 shows a block diagram of the 80960RP.

The 80960RP is a multi-function PCI device. Function 0 is the PCI-to-PCI Bridge Unit. Function 1 is the Address Translation Unit. The 80960RP contains PCI configuration space accessible through the primary PCI bus.

The 80960JF provides high performance to cost-sensitive 32-bit embedded applications. The 80960JF is object code compatible with the 80960 Core Architecture and is capable of sustained execution at the rate of one instruction per clock. This processor's features include generous instruction cache, data cache and data RAM. It also boasts a fast interrupt mechanism, dual programmable timer units and new instructions.

Memory subsystems for cost-sensitive embedded applications often impose substantial wait state penalties. The 80960JF integrates considerable storage resources on-chip to decouple CPU execution from the external bus.

The 80960JF includes a 4 Kbyte instruction cache, a 2 Kbyte data cache and 1 Kbyte data RAM.

The 80960JF rapidly allocates and deallocates local register sets during context switches. The processor needs to flush a register set to the stack only when it saves more than seven sets to its local register cache.

A 32-bit multiplexed burst bus provides a high-speed interface to system memory and I/O. A full complement of control signals simplifies the connection of the 80960JF to external components. The user programs physical and logical memory attributes through memory-mapped control registers (MMRs), an extension not found on the i960 Kx, Sx or Cx processors. Physical and logical configuration registers enable the processor to operate with all combinations of bus width and data object alignment. The processor supports a homogeneous byte ordering model.

The 80960JF processor integrates two important peripherals: a timer unit and an interrupt controller. These and other hardware resources are programmed through memory-mapped control registers, an extension to the familiar 80960 architecture.

The Timer Unit offers two independent 32-bit timers for use as real-time system clocks and general purpose system timing. These operate in either single-shot or auto-reload mode and can generate interrupts.

The Interrupt Controller Unit (ICU) provides a flexible, low-latency means for requesting interrupts. It provides full programmability of up to 240 interrupt sources into 31 priority levels. The ICU takes advantage of a cached priority table and optional routine caching to minimize interrupt latency. Local registers may be dedicated to high-priority interrupts to further reduce their latency. Acting independently from the core, the ICU compares the priorities of posted interrupts with the current process priority, off-loading this task from the core. The ICU also supports the integrated timer interrupts.

The 80960JF features a Halt mode designed to support applications where low power consumption is critical. The **halt** instruction lets you shut down instruction execution.

PRODUCT PREVIEW

The 80960JF's testability features, including ONCE (On-Circuit Emulation) mode and Boundary Scan (JTAG), provide a powerful environment for design debug and fault diagnosis.

The *Solutions960*® program features a wide variety of development tools that support the i960 processor family. Many of these tools are developed by partner companies; some are developed by Intel, such as profile-driven optimizing compilers. For more information on these products, contact your local Intel representative.

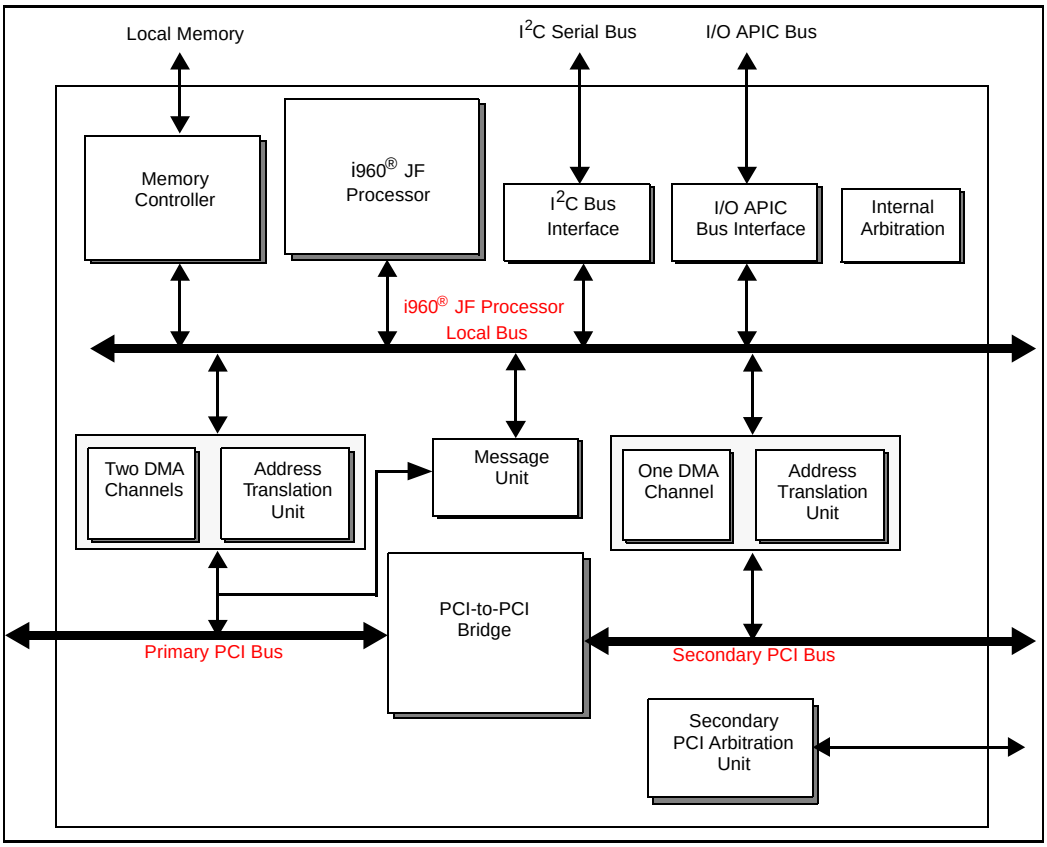


Figure 1. 80960RP Block Diagram

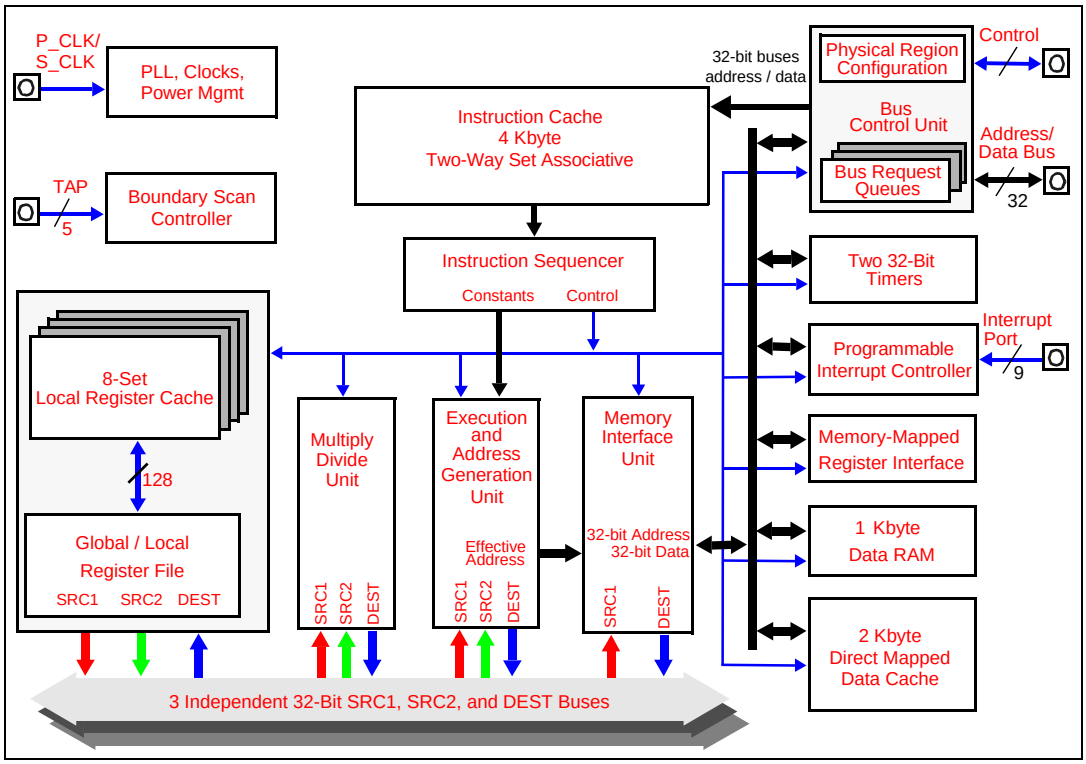


Figure 2. 80960JF Core Block Diagram

2.1 80960RP Features

The 80960RP adds many features onto an 80960JF core to create an intelligent I/O processor. These features are outlined in the following sections.

2.1.1 PCI-to-PCI Bridge Unit

The PCI-to-PCI bridge unit connects two independent PCI buses. The bridge allows certain bus transactions on one PCI bus to be forwarded to the other PCI bus. It also allows fully independent PCI bus operation, including independent clocks. Dedicated data queues support high performance bandwidth on the PCI buses. The 80960RP supports PCI 64-bit Dual Address Cycle (DAC) addressing.

The PCI-to-PCI bridge has dedicated PCI configuration space that is accessible through the primary PCI bus.

The PCI-to-PCI bridge in the 80960RP is fully compliant with the *PCI-to-PCI Bridge Architecture Specification*, published by the PCI Special Interest Group.

2.1.2 Private PCI Devices

A key feature of 80960RP's design is that it explicitly supports private PCI devices that can use the secondary PCI bus without being detected by the PCI configuration software. The PCI-to-PCI Bridge and the Address Translation Unit work together to

hide private devices from PCI configuration cycles and to allow these devices to utilize a private PCI address space. These devices can be configured by the Address Translation Unit through normal PCI configuration cycles.

2.1.3 DMA Controller

The DMA Controller allows low-latency, high-throughput data transfers between PCI bus agents and 80960 local memory.

There are three separate DMA channels to accommodate data transfers. Two channels are dedicated to primary PCI bus data transfers and one channel is dedicated to secondary PCI bus data transfers. The DMA Controller supports chaining and unaligned data transfers. It is programmable only through the i960 JF processor.

2.1.4 Address Translation Unit

The Address Translation Unit allows PCI transactions direct access to the 80960 local memory. The i960 JF processor has direct access to both PCI buses. Address translation is provided for transactions between the PCI address space and i960 processor address space. Address translation is controlled through programmable registers accessible from both the PCI interface and the i960 JF processor. Dual access to registers allows flexibility in mapping the two address spaces.

2.1.5 Messaging Unit

The Messaging Unit (MU) provides data transfer between the PCI system and the 80960JF processor. It also notifies the respective system of the arrival of new data through an interrupt.

The MU has four distinct messaging mechanisms. Each allows a host processor or external PCI agent and the 80960RP to communicate through message passing and interrupt generation.

The four mechanisms are:

- Message Registers
- Doorbell Registers
- Circular Queues
- Index Registers

2.1.6 Integrated Memory Controller

The Integrated Memory Controller provides direct control for external memory systems. Support is provided for DRAM, SRAM, ROM and Flash Memory. The Integrated Memory Controller provides a direct connect interface to memory that typically does not require external logic. It features programmable chip selects, a wait state generator and byte parity.

The external memory can be configured as PCI addressable memory or as private i960 JF processor memory.

2.1.7 I^2C Bus Interface Unit

The I^2C (Inter-Integrated Circuit) Bus Interface Unit allows the i960 JF processor to serve as a master and slave device residing on the I^2C bus. The I^2C bus is a serial bus developed by Philips Corporation consisting of a two pin interface. The bus allows the 80960RP to interface to other I^2C peripherals and microcontrollers for system management functions. It requires a minimum of hardware for an economical system to relay status and reliability information on the I/O subsystem to an external device.

2.1.8 I/O APIC Bus Interface Unit

The I/O APIC Bus Interface Unit provides an interface to the three-wire Advanced Programmable Interrupt Controller (APIC) bus that allows I/O APIC emulation in software. Interrupt messages can be sent on the bus and EOI messages can be received.

2.2 80960JF Processor Specific Features

The processing power of the 80960RP comes from the 80960JF Processor core. The features of the 80960JF are detailed in the following sections.

2.2.1 80960 Processor Core

The 80960JF microprocessor is a new, scalar implementation of the 80960 Core Architecture. Intel designed it to be a very high performance device that is also cost-effective.

Factors that contribute to the core's performance include:

- Single-clock execution of most instructions
- Independent Multiply/Divide Unit
- Efficient instruction pipeline minimizes pipeline break latency
- Register and resource scoreboarding allow overlapped instruction execution
- 128-bit register bus speeds local register caching
- 4 Kbyte two-way set-associative, integrated instruction cache
- 2 Kbyte direct-mapped, integrated data cache
- 1 Kbyte integrated data RAM delivers zero wait state program data

The local processor operates out of its own 32-bit address space, which is independent of the PCI address space. Memory on the i960 JF processor bus can be:

- Made visible to the PCI address space
- Kept private to the local processor
- Allocated as a combination of the two.

2.2.2 Burst Bus

A 32-bit high-performance bus controller interfaces the 80960JF to external memory and peripherals. The Bus Control Unit fetches instructions and transfers data on the 80960JF local bus at the rate of up to four 32-bit words per six clock cycles. The external address/data bus is multiplexed.

Users may configure the 80960JF's bus controller to match an application's fundamental memory organization. Physical bus width is register programmed for up to eight regions. Byte ordering and data caching are programmed through a group of logical memory templates and a defaults register.

The Bus Control Unit's features include:

- Multiplexed external bus to minimize pin count
- 32-, 16- and 8-bit bus widths to simplify I/O interfaces
- External ready control for address-to-data, data-to-data and data-to-next-address wait state types
- Support for big or little endian byte ordering to facilitate the porting of existing program code
- Unaligned bus accesses performed transparently
- Three-deep load/store queue to decouple the bus from the core

Upon reset, the 80960JF conducts an internal self test. Then, before executing its first instruction, it performs an external bus confidence test by performing a checksum on the first words of the Initialization Boot Record.

2.2.3 Timer Unit

The timer unit (TU) contains two independent 32-bit timers that are capable of counting at several clock rates and generating interrupts. Each is programmed by use of the Timer Unit registers. These memory-mapped registers are addressable on 32-bit boundaries. The timers have a single-shot mode and auto-reload capabilities for continuous operation. Each timer has an independent interrupt request to the 80960JF's interrupt controller. The TU can generate a fault when unauthorized writes from user mode are detected.

2.2.4 Priority Interrupt Controller

A programmable interrupt controller manages up to 240 external sources through an 8-bit external interrupt port. Alternatively, the interrupt inputs may be configured for individual edge- or level-triggered inputs. The Interrupt Unit also accepts interrupts from the two on-chip timer channels and a single Non-Maskable Interrupt (NMI#) pin. Interrupts are serviced according to their priority levels relative to the current process priority.

Low interrupt latency is critical to many embedded applications. As part of its highly flexible interrupt mechanism, the 80960JF exploits several techniques to minimize latency:

- Interrupt vectors and interrupt handler routines can be reserved on-chip
- Register frames for high-priority interrupt handlers can be cached on-chip
- The interrupt stack can be placed in cacheable memory space

2.2.5 Faults and Debugging

The 80960JF employs a comprehensive fault model. The processor responds to faults by making implicit calls to a fault handling routine. Specific information collected for each fault allows the fault handler to diagnose exceptions and recover appropriately.

The processor also has built-in debug capabilities. Via software, the 80960JF may be configured to detect as many as seven different trace event types. Alternatively, **mark** and **fmark** instructions can generate trace events explicitly in the instruction stream. Hardware breakpoint registers are also available to trap on execution and data addresses.

2.2.6 Low Power Operation

Intel fabricates the 80960JF using an advanced sub-micron manufacturing process. The processor's sub-micron topology provides the circuit density for optimal cache size and high operating speeds while dissipating modest power. The processor also uses dynamic power management to turn off clocks to unused circuits.

Users may program the 80960JF to enter Halt mode for maximum power savings. In Halt mode, the processor core stops completely but the integrated peripherals continue to function. Processor execution resumes from internally or externally generated interrupts.

2.2.7 Test Features

The 80960JF incorporates numerous features that enhance the user's ability to test both the processor and the system to which it is attached. These features include ONCE (On-Circuit Emulation) mode and Boundary Scan (JTAG).

The 80960JF provides testability features compatible with IEEE Standard Test Access Port and Boundary Scan Architecture (IEEE Std. 1149.1).

One of the boundary scan instructions, HIGHZ, forces the processor to float all its output pins (ONCE mode). ONCE mode can also be initiated at reset without using the boundary scan mechanism.

ONCE mode is useful for board-level testing. This feature allows a mounted 80960JF to electrically "remove" itself from a circuit board. This mode allows system-level testing where a remote tester, such as an In-Circuit Emulator (ICE) system, can exercise the processor system.

The provided test logic does not interfere with component or circuit board behavior and ensures that components function correctly, connections between various components are correct, and various components interact correctly on the printed circuit board.

The JTAG Boundary Scan feature is an attractive alternative to conventional "bed-of-nails" testing. It can examine connections that might otherwise be inaccessible to a test system.

2.2.8 Memory-Mapped Control Registers

The 80960JF, though compliant with 80960 series processor core, has the added advantage of memory-mapped, internal control registers not found on the 80960Kx, Sx or Cx processors. These give software the interface to easily read and modify internal control registers.

Each of these registers is accessed as a memory-mapped, 32-bit register. Access is accomplished through regular memory-format instructions. The processor ensures that these accesses do not generate external bus cycles.

2.2.9 Instructions, Data Types and Memory Addressing Modes

As with all 80960 family processors, the 80960JF instruction set supports several different data types and formats:

- Bit
- Bit fields
- Integer (8-, 16-, 32-, 64-bit)
- Ordinal (8-, 16-, 32-, 64-bit unsigned integers)
- Triple word (96 bits)
- Quad word (128 bits)

The 80960JF provides a full set of addressing modes for C and assembly:

- Two Absolute modes
- Five Register Indirect modes
- Index with displacement
- IP with displacement

Table 1 shows all of the 80960JF instructions that are available.

Table 1. 80960JF Instruction Set

Data Movement	Arithmetic	Logical	Bit, Bit Field and Byte
Load	Add	And	Set Bit
Store	Subtract	Not And	Clear Bit
Move	Multiply	And Not	Not Bit
*Conditional Select	Divide	Or	Alter Bit
Load Address	Remainder	Exclusive Or	Scan For Bit
	Modulo	Not Or	Span Over Bit
	Shift	Or Not	Extract
	Extended Shift	Nor	Modify
	Extended Multiply	Exclusive Nor	Scan Byte for Equal
	Extended Divide	Not	*Byte Swap
	Add with Carry	Nand	
	Subtract with Carry		
	*Conditional Add		
	*Conditional Subtract		
	Rotate		
Comparison	**Branch**	**Call/Return**	**Fault**
Compare	Unconditional Branch	Call	Conditional Fault
Conditional Compare	Conditional Branch	Call Extended	Synchronize Faults
Compare and Increment	Compare and Branch	Call System	
Compare and Decrement		Return	
Test Condition Code		Branch and Link	
Check Bit			
Debug	**Processor Management**	**Atomic**	
Modify Trace Controls	Flush Local Registers	Atomic Add	
Mark	Modify Arithmetic Controls	Atomic Modify	
Force Mark	Modify Process Controls		
	*Halt		
	System Control		
	*Cache Control		
	*Interrupt Control		

* Denotes 80960JF instructions unavailable on 80960CA/CF, 80960KA/KB and 80960SA/SB implementations.

3.0 PACKAGE INFORMATION

NOTICE: *This document contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product is available.*

3.1 Package Introduction

The 80960RP will be offered in a 352-pin perimeter Ball Grid Array (BGA) device of the Over Molded Pad Array Carrier (OMPAC) type. This is a perimeter array package with four rows of ball connections in the outer area of the package. The package is 35mm on a side with a ball spacing of 1.27mm. The Theta JA (Junction to Ambient) for this package is currently estimated at 25°C/Watt with no airflow. The FW80960RP-33 will be specified for operation at V_{cc}=5.0V±5% at 33.33 MHz.

3.2 Pin Descriptions

This section describes the pins for the 80960RP in the 352-pin plastic Ball Grid Array (BGA) package.

Section 3.2.1, Functional Pin Definitions describes pin function; **Section 3.2.2, 80960RP 352-Lead BGA Pinout** defines the signal and pin locations.

3.2.1 Functional Pin Definitions

Table 2 presents the legend for interpreting the pin descriptions which follow. Pins associated with the bus interface are described in Table 3. Pins associated with basic control and test functions are described in Table 4. Pins associated with the Interrupt Unit are described in Table 5. PCI signal pins are described in Table 6. Memory Controller pins are described in Table 7. Pins associated specifically with functions on the 80960RP are described in Table 8.

Table 2. Pin Description Nomenclature

Symbol	Description
I	Input pin only.
O	Output pin only.
I/O	Pin can be either an input or output.
OD	Open Drain pin.
–	Pin must be connected as described.
S (...)	Synchronous. Inputs must meet setup and hold times relative to P_CLK or S_CLK, based on the status of the SYNC pin at reset, for proper operation. S(E) Edge sensitive input S(L) Level sensitive input
A (...)	Asynchronous. Inputs may be asynchronous relative to P_CLK or S_CLK, based on the status of the SYNC pin at reset. A(E) Edge sensitive input A(L) Level sensitive input
R (...)	While the P_RST# pin is asserted, the pin: R(1) is driven to V_{CC} R(0) is driven to V_{SS} R(Q) is a valid output R(Z) Floats R(H) is pulled up to V_{CC}
H (...)	While the 80960RP processor is in the hold state, the pin: H(1) is driven to V_{CC} H(0) is driven to V_{SS} H(Q) Maintains previous state or continues to be a valid output H(Z) Floats
P (...)	While the 80960RP processor is halted, the pin: P(1) is driven to V_{CC} P(0) is driven to V_{SS} P(Q) Maintains previous state or continues to be a valid output
K (...)	While the Secondary PCI Bus is in park mode, the pin: K(Z) Floats K(Q) Maintains previous state or continues to be a valid output

Table 3. Pin Description — 80960 Processor Signals (Sheet 1 of 4)

NAME	TYPE	DESCRIPTION
AD31:0	**I/O** S(L) R(Z) H(Z) P(Q)	**ADDRESS / DATA BUS** carries 32-bit physical addresses and 8-, 16- or 32-bit data to and from memory. During an address (T_a) cycle, bits 2-31 contain a physical word address (bits 0-1 indicate SIZE; see below). During a data (T_d) cycle, read or write data is present on one or more contiguous bytes, comprising AD31:24, AD23:16, AD15:8 and AD7:0. During write operations, unused pins are driven to determinate values. SIZE, which comprises bits 0-1 of the AD lines during a T_a cycle, specifies the number of data transfers during the bus transaction on the 80960RP local bus. When the DMA or ATU's initiate data transfers the transfer size shown below is not valid. **AD1 AD0 Bus Transfers** 0 0 1 Transfer 0 1 2 Transfers 1 0 3 Transfers 1 1 4 Transfers When the processor enters Halt mode, if the previous bus operation was a: • write — AD31:2 are driven with the last data value on the AD bus. • read — AD31:2 are driven with the last address value on the AD bus. Typically, AD1:0 reflect the SIZE information of the last bus transaction (either instruction fetch or load/store) that was executed before entering Halt mode.
ALE	**O** R(0) H(Z) P(0)	**ADDRESS LATCH ENABLE** indicates the transfer of a physical address. ALE is asserted during a T_a cycle and deasserted before the beginning of the T_d state. It is active HIGH and floats to a high impedance state during a hold cycle (T_h).
ADS#	**O** R(1) H(Z) P(1)	**ADDRESS STROBE** indicates a valid address and the start of a new bus access. The processor asserts ADS# for the entire T_a cycle. External bus control logic typically samples ADS# at the end of the cycle.

Table 3. Pin Description — 80960 Processor Signals (Sheet 2 of 4)

NAME	TYPE	DESCRIPTION
BE3:0#	**O** R(1) H(Z) P(1)	**BYTE ENABLES** select which of up to four data bytes on the bus participate in the current bus access. Byte enable encoding is dependent on the bus width of the memory region accessed: *32-bit bus:* BE3# enables data on AD31:24 BE2# enables data on AD23:16 BE1# enables data on AD15:8 BE0# enables data on AD7:0 *16-bit bus:* BE3# becomes Byte High Enable (enables data on AD15:8) BE2# is not used (state is high) BE1# becomes Address Bit 1 (A1) BE0# becomes Byte Low Enable (enables data on AD7:0) *8-bit bus:* BE3# is not used (state is high) BE2# is not used (state is high) BE1# becomes Address Bit 1 (A1) BE0# becomes Address Bit 0 (A0) The processor asserts byte enables, byte high enable and byte low enable during T_a. Since unaligned bus requests are split into separate bus transactions, these signals do not toggle during a burst. They remain active through the last T_d cycle.
WIDTH/ HLTD1/ RETRY	**I/O** R(H) H(Z) P(1)	**WIDTH/HALTED1/RETRY** denotes the physical memory attributes for a bus transaction in conjunction with the WIDTH/HLTD0/SYNC PIN: WIDTH/HLTD1/RETRY — WIDTH/HLTD0/SYNC 0 — 0 — 8 Bits Wide 0 — 1 — 16 Bits Wide 1 — 0 — 32 Bits Wide 1 — 1 — Processor Halted The processor floats the WIDTH/HLTD pins whenever it relinquishes the bus in response to a HOLD request, regardless of prior operating state. The RETRYsignal is sampled at Primary PCI bus reset to determine if the Primary PCI interface will be disabled. If high, the Primary PCI interface will disable PCI configuration cycles by signaling a Retry until the Configuration Cycle Disable bit is cleared in the Extended Bridge Control Register. If low, the Primary PCI interface allow configuration cycles to occur.
WIDTH/ HLTD0/ SYNC	**I/O** R(H) H(Z) P(1)	**WIDTH/HALTED0/SYNC** denotes the physical memory attributes for a bus transaction (see WIDTH/HLTD1 description above) and is used to select clock synchronization. The SYNC signal is sampled at Primary PCI Reset to determine the clock synchronization of the Primary PCI Bus clock. If SYNC is high, both the primary and secondary PCI bus clocks are synchronous with each other. If SYNC is low, the Primary PCI Bus clock is asynchronous with respect to the Secondary PCI Bus clock.

Table 3. Pin Description — 80960 Processor Signals (Sheet 3 of 4)

NAME	TYPE	DESCRIPTION
D/C#/ RST_MODE#	**I/O** R(H) H(Z) P(Q)	**DATA/CODE/RESET_MODE** indicates that a bus access is a data access (1) or an instruction access (0). D/C# has the same timing as W/R#. The RST_MODE signal is sampled at Primary PCI bus reset to determine if the 80960RP Processor is to be held in reset. If asserted, the 80960RP Processor will be held in reset until the 80960 Processor Reset bit is cleared in the Extended Bridge Control Register.
W/R#	**O** R(0) H(Z) P(Q)	**WRITE/READ** specifies, during a T_a cycle, whether the operation is a write (1) or read (0). It is latched on-chip and remains valid during T_d cycles.
DT/R#	**O** R(0) H(Z) P(Q)	**DATA TRANSMIT / RECEIVE** indicates the direction of data transfer to and from the address/data bus. It is low during T_a and T_w/T_d cycles for a read; it is high during T_a and T_w/T_d cycles for a write. DT/R# never changes state when DEN# is asserted.
DEN#	**O** R(1) H(Z) P(1)	**DATA ENABLE** indicates data transfer cycles during a bus access. DEN# is asserted at the start of the first data cycle in a bus access and deasserted at the end of the last data cycle. DEN# is used with DT/R# to provide control for data transceivers connected to the data bus.
BLAST#	**O** R(1) H(Z) P(1)	**BURST LAST** indicates the last transfer in a bus access. BLAST# is asserted in the last data transfer of burst and non-burst accesses. BLAST# remains active as long as wait states are inserted via the RDYRCV# pin. BLAST# becomes inactive after the final data transfer in a bus cycle.
RDYRCV#	**I** S(L)	**READY/RECOVER** indicates that data on AD lines can be sampled or removed. If RDYRCV# is not asserted during a T_d cycle, the T_d cycle is extended to the next cycle by inserting a wait state (T_w). The RDYRCV# pin has an alternate function during the recovery (T_r) state. The processor continues to insert additional recovery states until it samples the pin HIGH. This function allows slow external devices longer to float their buffers before the processor begins to drive address again.
LRDYRCV#	**O** R(1) H(Q) P(Q)	**LOCAL READY/RECOVER** is an output version of the READY/RECOVER (RDYRCV#) signal generated by the integrated memory controller.
LOCK#/ ONCE#	**I/O** S(L) R(H) H(Z) P(Q)	**BUS LOCK** indicates that an atomic read-modify-write operation is in progress. The LOCK# output is asserted in the first clock of an atomic operation and deasserted in the last data transfer of the sequence. The processor does not grant HOLDA while it is asserting LOCK#. This prevents external agents from accessing memory involved in semaphore operations. **ONCE MODE:** The processor samples the ONCE# input during reset. If it is asserted LOW at the end of reset, the processor enters ONCE mode. In ONCE mode, the processor stops all clocks and floats all output pins. The pin has a weak internal pullup which is active during reset to ensure normal operation if the pin is left unconnected.

Table 3. Pin Description — 80960 Processor Signals (Sheet 4 of 4)

NAME	TYPE	DESCRIPTION
HOLD	**I** S(L)	**HOLD**: A request from an external bus master to acquire the bus. When the processor receives HOLD and grants bus control to another master, it asserts HOLDA, floats the address/data and control lines and enters the T_h state. When HOLD is deasserted, the processor deasserts HOLDA and enters either the T_i or T_a state, resuming control of the address/data and control lines.
HOLDA	**O** R(Q) H(1) P(Q)	**HOLD ACKNOWLEDGE** indicates to an external bus master that the processor has relinquished control of the bus. The processor can grant HOLD requests and enter the T_h state during reset and while halted as well as during regular operation.

Table 4. Pin Description — Processor Control Signals, Test Signals and Power (Sheet 1 of 2)

NAME	TYPE	DESCRIPTION
STEST	**I** S(L)	**SELF TEST** enables or disables the processor's internal self-test feature at initialization. STEST is examined at the end of reset. If STEST is asserted, the processor performs its internal self-test and the external bus confidence test. If STEST is deasserted, the processor performs only the external bus confidence test.
FAIL#	**O** R(0) H(Q) P(1)	**FAIL** indicates a failure of the processor's built-in self-test performed during initialization. FAIL# is asserted immediately upon reset and toggles during self-test to indicate the status of individual tests: • If self-test passes, the processor deasserts FAIL# and commences operation from user code. • If self-test fails, the processor asserts FAIL# and then stops executing. This does not, however, cause the bridge to stop execution.
TCK	**I**	**TEST CLOCK** is a CPU input which provides the clocking function for IEEE 1149.1 Boundary Scan Testing (JTAG). State information and data are clocked into the processor on the rising edge; data is clocked out of the processor on the falling edge.
TDI	**I** S(L)	**TEST DATA INPUT** is the serial input pin for JTAG. TDI is sampled on the rising edge of TCK, during the SHIFT-IR and SHIFT-DR states of the Test Access Port.
TDO	**O** R(Q) H(Q) P(Q)	**TEST DATA OUTPUT** is the serial output pin for JTAG. TDO is driven on the falling edge of TCK during the SHIFT-IR and SHIFT-DR states of the Test Access Port. At other times, TDO floats.
TRST#	**I** A(L)	**TEST RESET** asynchronously resets the Test Access Port (TAP) controller function of IEEE 1149.1 Boundary Scan testing (JTAG). When using the Boundary Scan feature, connect a pulldown resistor between this pin and V_{SS}. If you are not using TAP, this pin must be connected to V_{SS}; however, no resistor is required.
TMS	**I** S(L)	**TEST MODE SELECT** is sampled at the rising edge of TCK to select the operation of the test logic for IEEE 1149.1 Boundary Scan testing.

Table 4. Pin Description — Processor Control Signals, Test Signals and Power (Sheet 2 of 2)

NAME	TYPE	DESCRIPTION
V_{CC}	–	**POWER** leads intended for external connection to a V_{CC} board plane.
V_{SS}	–	**GROUND** leads intended for external connection to a V_{SS} board plane.
N.C.	–	**NO CONNECT** leads. Do not make any system connections to these leads.

Table 5. Pin Description — Interrupt Unit Signals

NAME	TYPE	DESCRIPTION
S_INTA#/ XINT0#	**I** A(L)	**EXTERNAL INTERRUPT** can be directed to **P_INTA#** or the 80960RP Processor XINT0# input.
S_INTB#/ XINT1#	**I** A(L)	**EXTERNAL INTERRUPT** can be directed to **P_INTB#** or the 80960RP Processor XINT1# input.
S_INTC#/ XINT2#	**I** A(L)	**EXTERNAL INTERRUPT** can be directed to **P_INTC#** or the 80960RP Processor XINT2# input.
S_INTD#/ XINT3#	**I** A(L)	**EXTERNAL INTERRUPT** can be directed to **P_INTD#** or the 80960RP Processor XINT3# input.
XINT7:4#	**I** A(L)	**EXTERNAL INTERRUPT** pins are used to request interrupt service. The XINT7:0# pins operate in dedicated mode, where each pin is assigned a dedicated interrupt level.
NMI#	**I** A(E)	**NON-MASKABLE INTERRUPT** causes a non-maskable interrupt event to occur. NMI# is the highest priority interrupt source and is falling edge-triggered. If NMI# is unused, it should be connected to V_{CC}.

Table 6. Pin Description— PCI Signals (Sheet 1 of 4)

NAME	TYPE	DESCRIPTION
P_AD31:0	**I/O** K(Q) R(Z)	**PRIMARY PCI ADDRESS/DATA** is the primary multiplexed PCI address and data bus.
P_PAR	**I/O** K(Q) R(Z)	**PRIMARY PCI BUS PARITY** is even parity across P_AD31:0 and P_C/BE3:0.
P_C/BE3:0#	**I/O** K(Q) R(Z)	**PRIMARY PCI BUS COMMAND and BYTE ENABLES** are multiplexed on the same PCI pins. During the address phase, they define the bus command. During the data phase, they are used as byte enables.
P_REQ#	**O** K(Q) R(Z)	**PRIMARY PCI BUS REQUEST** indicates to the primary PCI bus arbiter that the processor desires use of the primary PCI bus.
P_GNT#	**I** R(Z)	**PRIMARY PCI BUS GRANT** indicates that access to the primary PCI bus has been granted.

Table 6. Pin Description— PCI Signals (Sheet 2 of 4)

NAME	TYPE	DESCRIPTION
P_FRAME#	**I/O** K(Z) R(Z)	**PRIMARY PCI BUS CYCLE FRAME** is asserted to indicate the beginning and duration of an access on the Primary PCI bus.
P_IRDY#	**I/O** K(Z) R(Z)	**PRIMARY PCI BUS INITIATOR READY** indicates the initiating agent's ability to complete the current data phase of the transaction. During a write, it indicates that valid data is present on the primary Address/Data bus. During a read, it indicates the processor is ready to accept the data.
P_TRDY#	**I/O** K(Z) R(Z)	**PRIMARY PCI BUS TARGET READY** indicates the target agent's ability to complete the current data phase of the transaction. During a read, it indicates that valid data is present on the primary Address/Data bus. During a write, it indicates the target is ready to accept the data.
P_STOP#	**I/O** K(Z)	**PRIMARY PCI BUS STOP** indicates the a request to stop the current transaction on the primary PCI bus.
P_DEVSEL#	**I/O** K(Z) R(Z)	**PRIMARY PCI BUS DEVICE SELECT** is driven by a target agent that has successfully decoded the address. As an input, it indicates whether or not an agent has been selected.
P_SERR#	**I/O** **OD** K(Z)	**PRIMARY PCI BUS SYSTEM ERROR** is driven for address parity errors on the Primary PCI bus.
P_CLK	**I**	**PRIMARY PCI BUS CLOCK** provides the timing for all primary PCI transactions.
P_RST#	**I** A(L)	**PRIMARY PCI BUS RESET** initializes the 80960RP processor and clears its internal logic.
P_PERR#	**I/O** K(Z) R(Z)	**PRIMARY PCI BUS PARITY ERROR** is asserted when a data parity error during a primary PCI bus transaction.
P_LOCK#	**I/O** K(Z) R(Z)	**PRIMARY PCI BUS LOCK** indicates the need to perform an atomic operation on the primary PCI bus.
P_IDSEL	**I** S(L)	**PRIMARY PCI BUS INITIALIZATION DEVICE SELECT** is used to select the 80960RP processor during a Configuration Read or Write command on the primary PCI bus.
P_INTA#	**O** **OD** R(Z)	**PRIMARY PCI BUS INTERRUPT A** is used to request a PCI interrupt.
P_INTB#	**O** **OD** R(Z)	**PRIMARY PCI BUS INTERRUPT B** is used to request a PCI interrupt.
P_INTC#	**O** **OD** R(Z)	**PRIMARY PCI BUS INTERRUPT C** is used to request a PCI interrupt.

Table 6. Pin Description— PCI Signals (Sheet 3 of 4)

NAME	TYPE	DESCRIPTION
P_INTD#	**O** **OD** R(Z)	**PRIMARY PCI BUS INTERRUPT D** is used to request a PCI interrupt.
S_AD31:0	**I/O** K(Q) R(Z)	**SECONDARY PCI ADDRESS/DATA** is the secondary multiplexed PCI address and data bus.
S_PAR	**I/O** K(Q) R(Z)	**SECONDARY PCI BUS PARITY** is even parity across S_AD31:0 and S_C/BE3:0.
S_C/BE3:0#	**I/O** K(Q) R(Z)	**SECONDARY PCI BUS COMMAND and BYTE ENABLES** are multiplexed on the same PCI pins. During the address phase, they define the bus command. During the data phase, they are used as byte enables.
S_FRAME#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS CYCLE FRAME** is asserted to indicate the beginning and duration of an access.
S_IRDY#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS INITIATOR READY** indicates the initiating agent's ability to complete the current data phase of the transaction. During a write, it indicates that valid data is present on the secondary Address/Data bus. During a read, it indicates the processor is ready to accept the data.
S_TRDY#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS TARGET READY** indicates the target agent's ability to complete the current data phase of the transaction. During a read, it indicates that valid data is present on the secondary Address/Data bus. During a write, it indicates the target is ready to accept the data.
S_STOP#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS STOP** indicates the a request to stop the current transaction on the secondary PCI bus.
S_DEVSEL#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS DEVICE SELECT** is driven by a target agent that has successfully decoded the address. As an input, it indicates whether or not an agent has been selected.
S_SERR#	**I/O** **OD** K(Z) R(Z)	**SECONDARY PCI BUS SYSTEM ERROR** is driven for address parity errors.
S_CLK	**I**	**SECONDARY PCI BUS CLOCK** provides the timing for all secondary PCI transactions and the 80960 JF processor.
S_RST#	**O** R(Q) K(Q)	**SECONDARY PCI BUS RESET** is an output based on P_RST#. It resets all devices connected to the secondary PCI bus.

Table 6. Pin Description— PCI Signals (Sheet 4 of 4)

NAME	TYPE	DESCRIPTION
S_IDSEL	**I** S(L)	**SECONDARY PCI BUS INITIALIZATION DEVICE SELECT** is used to select the 80960RP processor during a Configuration Read or Write command on the secondary PCI bus.
S_PERR#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS PARITY ERROR** is asserted when a data parity error occurs during a secondary PCI bus transaction.
S_LOCK#	**I/O** K(Z) R(Z)	**SECONDARY PCI BUS LOCK** indicates the need to perform an atomic operation on the secondary PCI bus.
S_REQ0#/ S_GNT#	**I**	**SECONDARY PCI BUS REQUEST0** is a request signal from device 0 on the secondary PCI bus when the internal Secondary PCI Bus Arbiter is enabled. **Secondary PCI BUS GRANT** is the grant signal for the 80960RP processor when the arbiter is disabled.
S_GNT0#/ S_REQ#	**O** K(Q) R(Z)	**SECONDARY PCI BUS GRANT0** is a grant signal sent to device 0 on the secondary PCI bus when the internal Secondary PCI Bus Arbiter is enabled. **Secondary PCI BUS REQUEST** is the request signal for the 80960RP processor when the arbiter is disabled.
S_REQ5:1#	**I** S(L)	**SECONDARY PCI BUS REQUEST** are request signals from devices 1-5 on the secondary PCI bus.
S_GNT5:1#	**O** K(Q) R(Q)	**SECONDARY PCI BUS GRANT** are grant signals sent to devices 1-5 on the secondary PCI bus.

Table 7. 80960RP Pin Description— Memory Controller Signals (Sheet 1 of 2)

NAME	TYPE	DESCRIPTION
MA11:0	**O** R(X) H(Q) P(Q)	**MULTIPLEXED ADDRESS BUS** carries the multiplexed row and column addresses to the DRAM memory banks.
DP3:0	**I/O** R(X) H(Q) P(Q)	**DATA PARITY** provides one parity bit for each byte of data in the DRAM banks.
RAS3:0#	**O** R(1) H(Q) P(Q)	**ROW ADDRESS STROBES** indicate the presence of a valid row address on the Multiplexed Address Bus.
CAS7:0#	**O** R(1) H(Q) P(Q)	**COLUMN ADDRESS STROBES** indicate the presence of a valid column address on the Multiplexed Address Bus.

Table 7. 80960RP Pin Description— Memory Controller Signals (Sheet 2 of 2)

NAME	TYPE	DESCRIPTION
MWE3:0#	**O** R(1) H(Q) P(Q)	**MEMORY WRITE ENABLES** provide the write strobe for the selected ROM or SRAM memory bank.
DWE1:0#	**O** R(1) H(Q) P(Q)	**DRAM WRITE ENABLES** indicate the direction data is to be transferred to/from DRAM and controls the WE# input on the DRAM device.
DALE1:0	**O** R(0) H(Q) P(Q)	**DRAM ADDRESS LATCH ENABLES** are used to control external address latches in an interleaved DRAM system.
CE1:0#	**O** R(1) H(Q) P(Q)	**CHIP ENABLE** is used to enable the ROM and SRAM devices during a memory access.
LEAF1:0#	**O** R(1) H(Q) P(Q)	**LEAF** signals are used to control interleaved DRAM banks.

Table 8. Pin Description— 80960RP Processor Signals

NAME	TYPE	DESCRIPTION
DREQ#	**I** S(L)	**DMA DEMAND MODE REQUEST** is used by external devices to indicate it has new data transfer to the DMA Controller or has available buffers to receive data from the DMA Controller.
DACK#	**O** R(1) H(Q) P(Q)	**DMA DEMAND MODE ACKNOWLEDGE** is asserted by the DMA Controller to indicate that it can receive new data from an external device or it has data to send to an external device.
PICCLK	**I**	**APIC BUS CLOCK** provides synchronous operation of the APIC bus.
PICD1:0	**I/O** **OD** R(Z) H(Q) P(Q)	**APIC DATA** is for incoming and outgoing APIC bus messages.
SDA	**I/O** **OD** R(Z) H(Q) P(Q)	**I^2C DATA** is used for data transfer and arbitration on the I^2C bus.
SCL	**I/O** **OD** R(Z) H(Q) P(Q)	**I^2C CLOCK** provides synchronous operation of the I^2C bus.
WAIT#	**O** R(1) H(Q) P(Q)	**WAIT** is an output that allows the DMA Controller to insert wait states during DMA accesses to an external memory system.

3.2.2 80960RP 352-Lead BGA Pinout

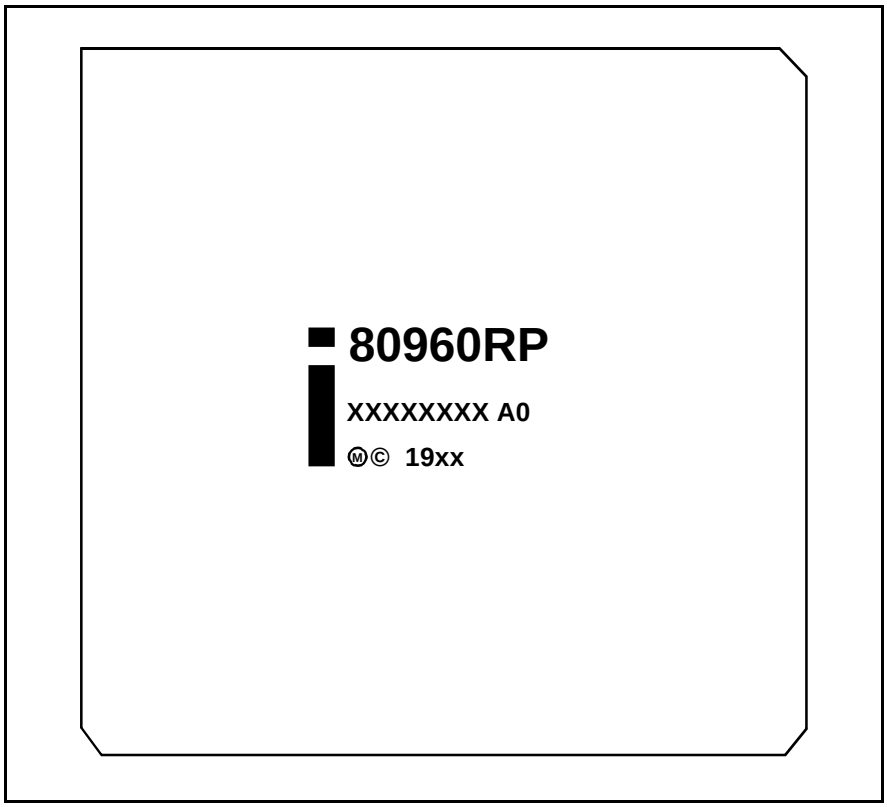


Figure 3. 352-Lead Ball Grid Array Package - Top View

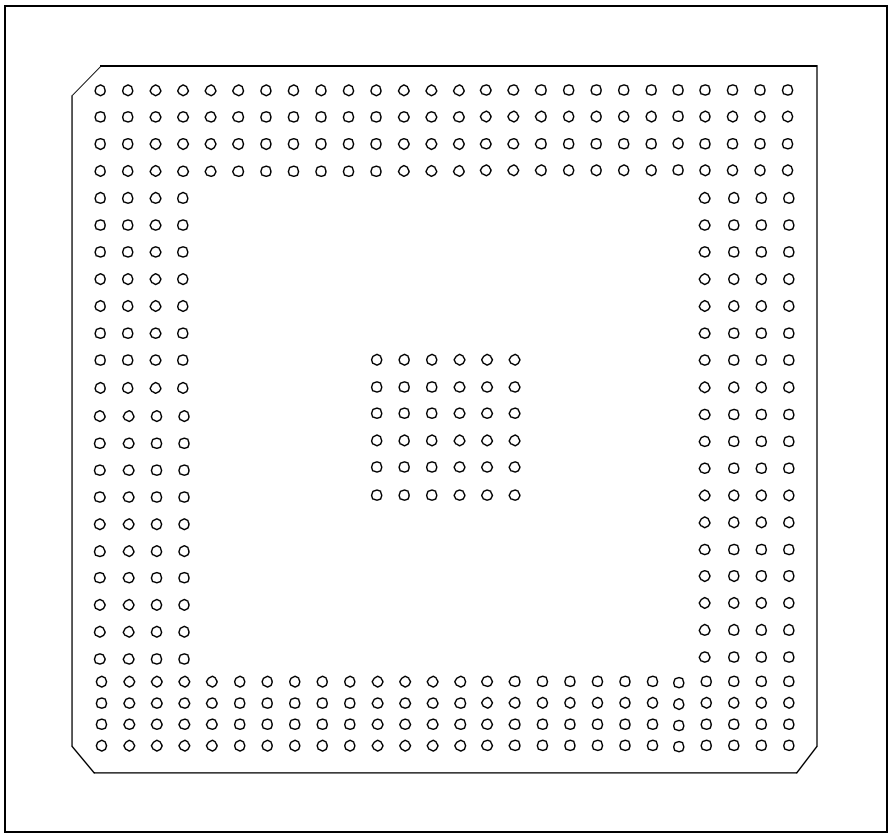


Figure 4. 352-Lead Ball Grid Array Package - Bottom View

4.0 BUS FUNCTIONAL WAVEFORMS

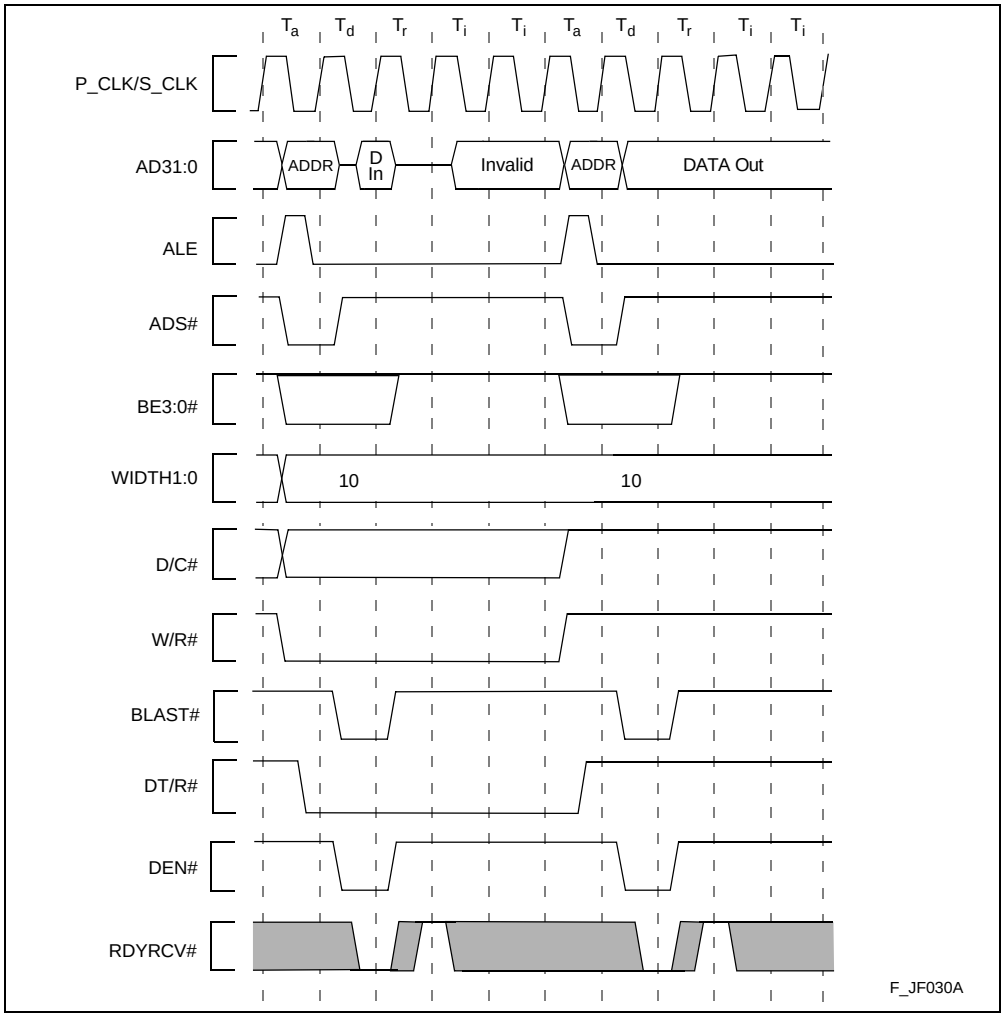


Figure 5. Non-Burst Read and Write Transactions Without Wait States, 32-Bit 80960 Local Bus

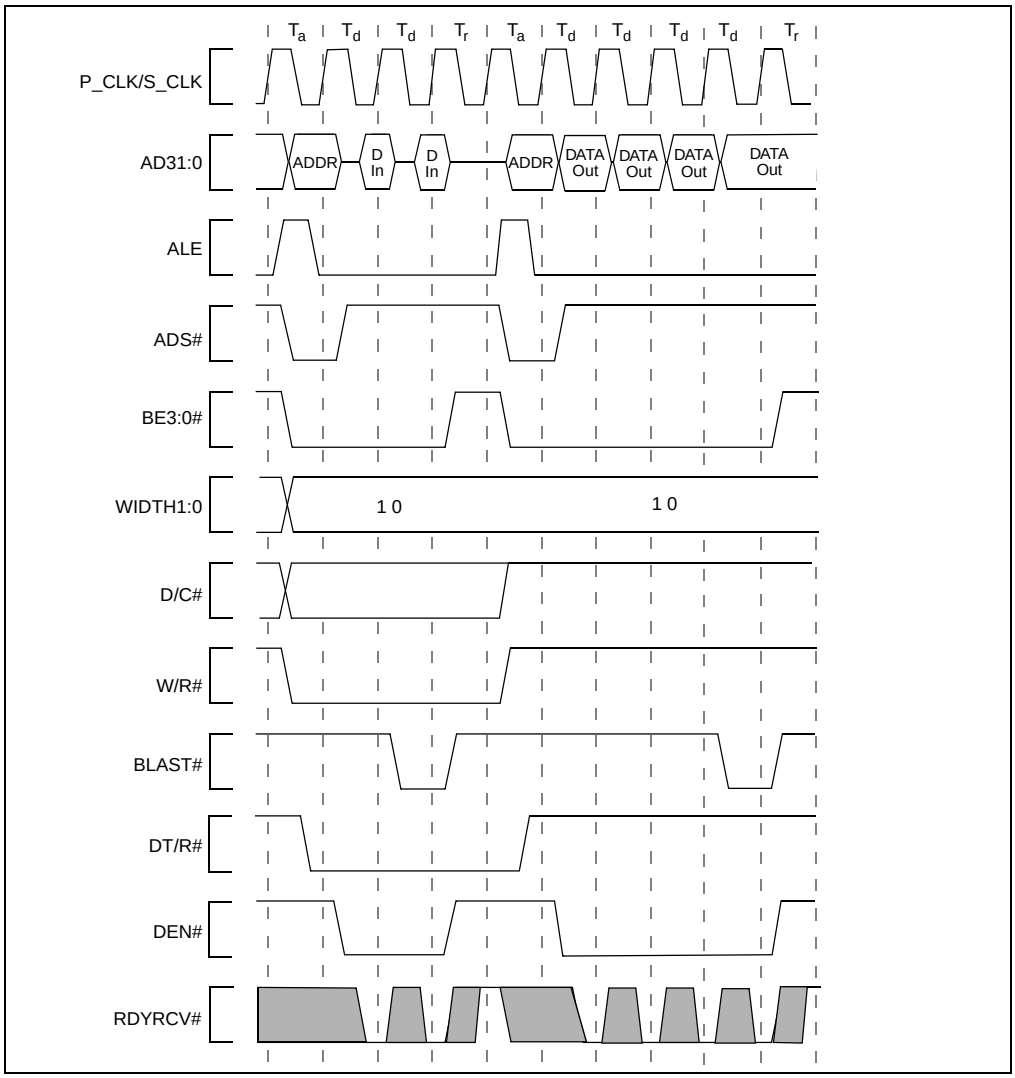


Figure 6. Burst Read and Write Transactions Without Wait States, 32-Bit 80960 Local Bus

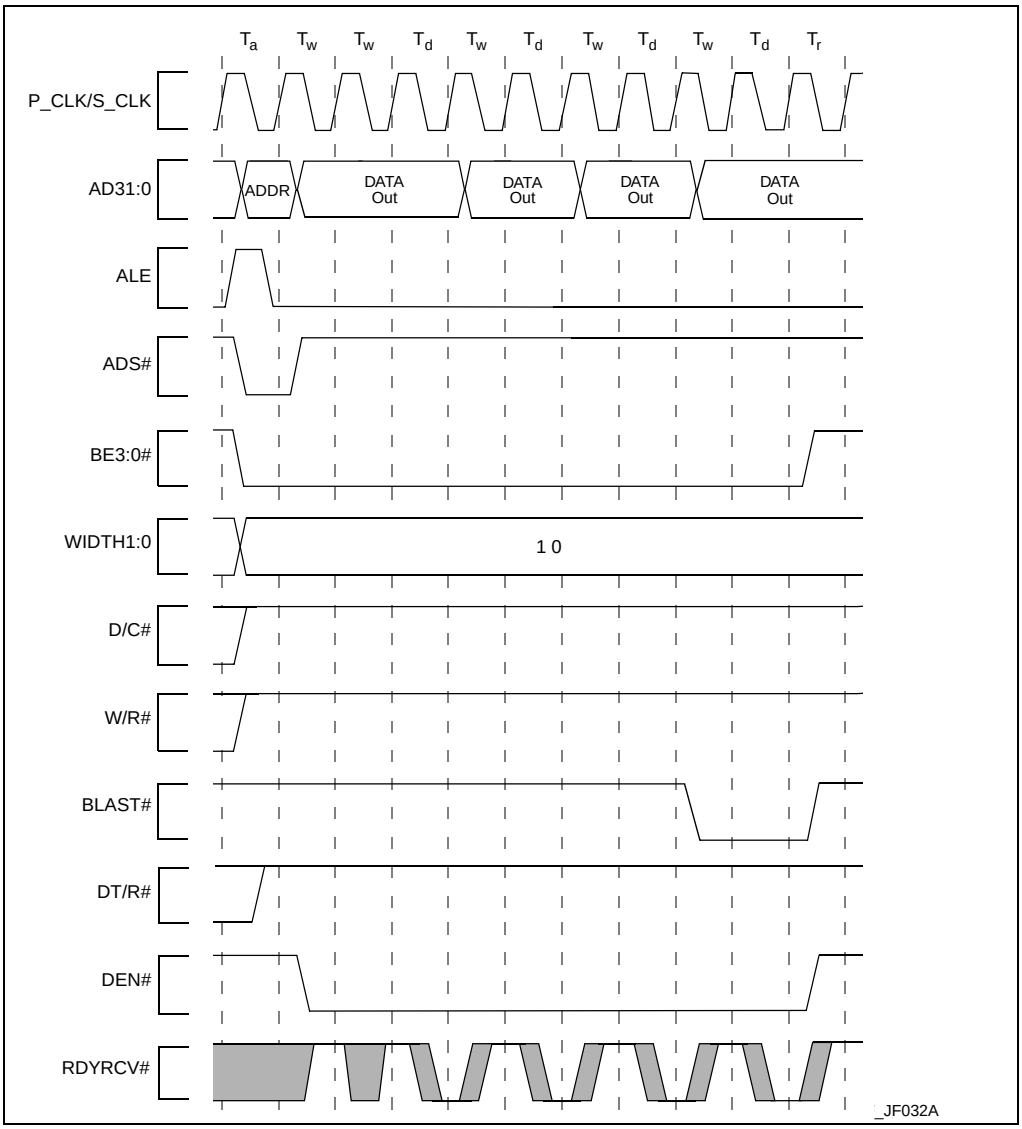


Figure 7. Burst Write Transactions With 2,1,1,1 Wait States, 32-Bit 80960 Local Bus

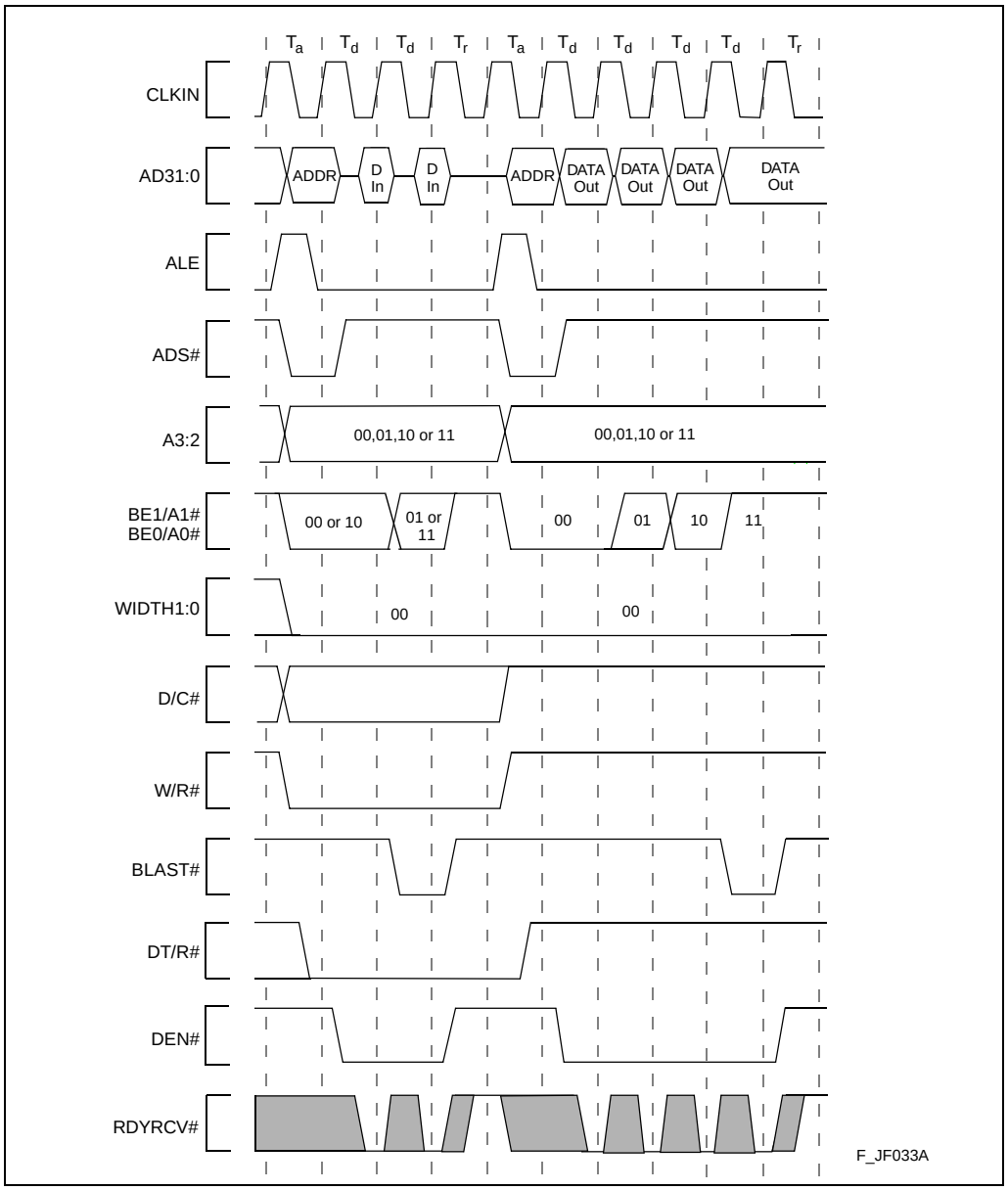


Figure 8. Burst Read and Write Transactions Without Wait States, 8-Bit 80960 Local Bus

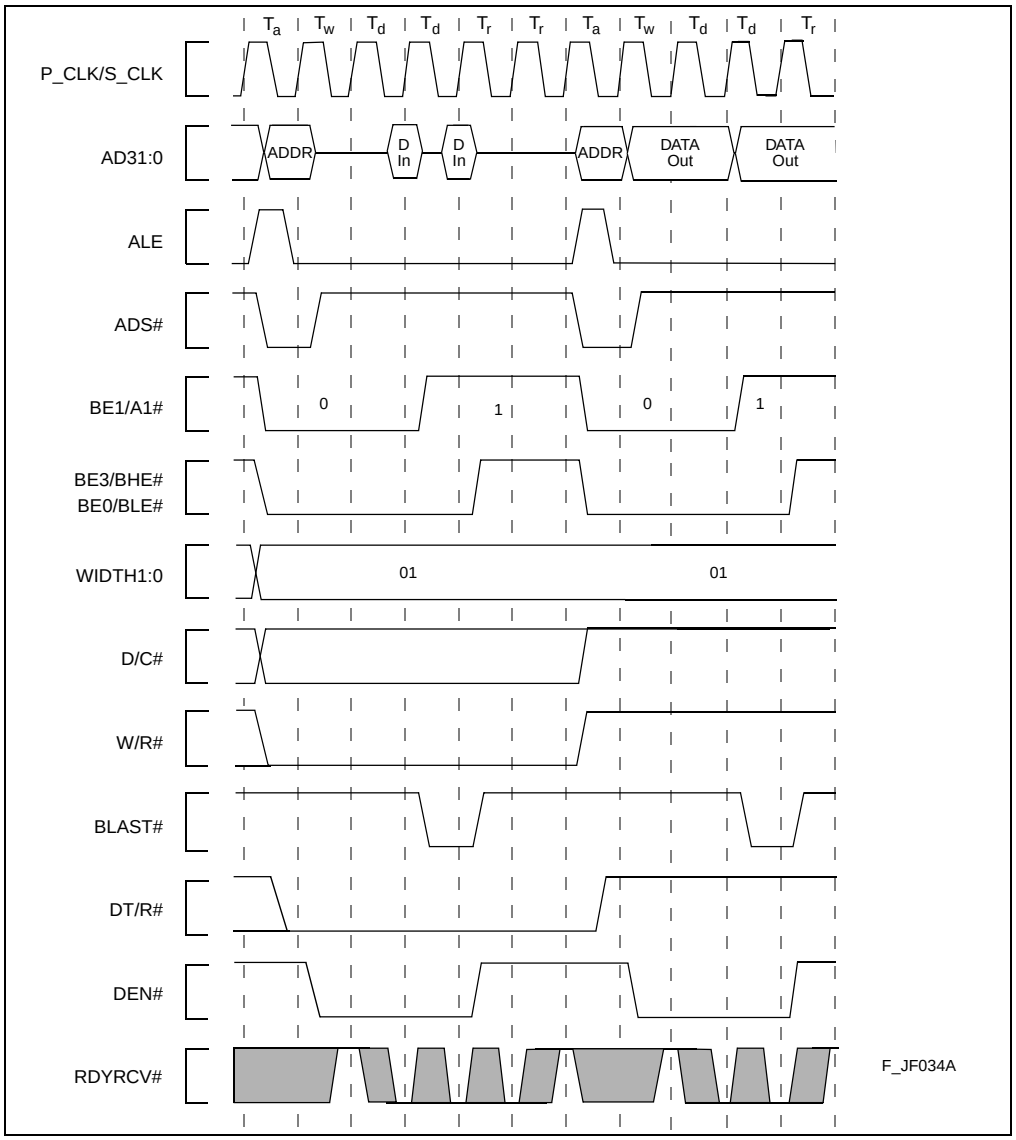


Figure 9. Burst Read and Write Transactions With 1, 0 Wait States and Extra Tr State on Read, 16-Bit 80960 Local Bus

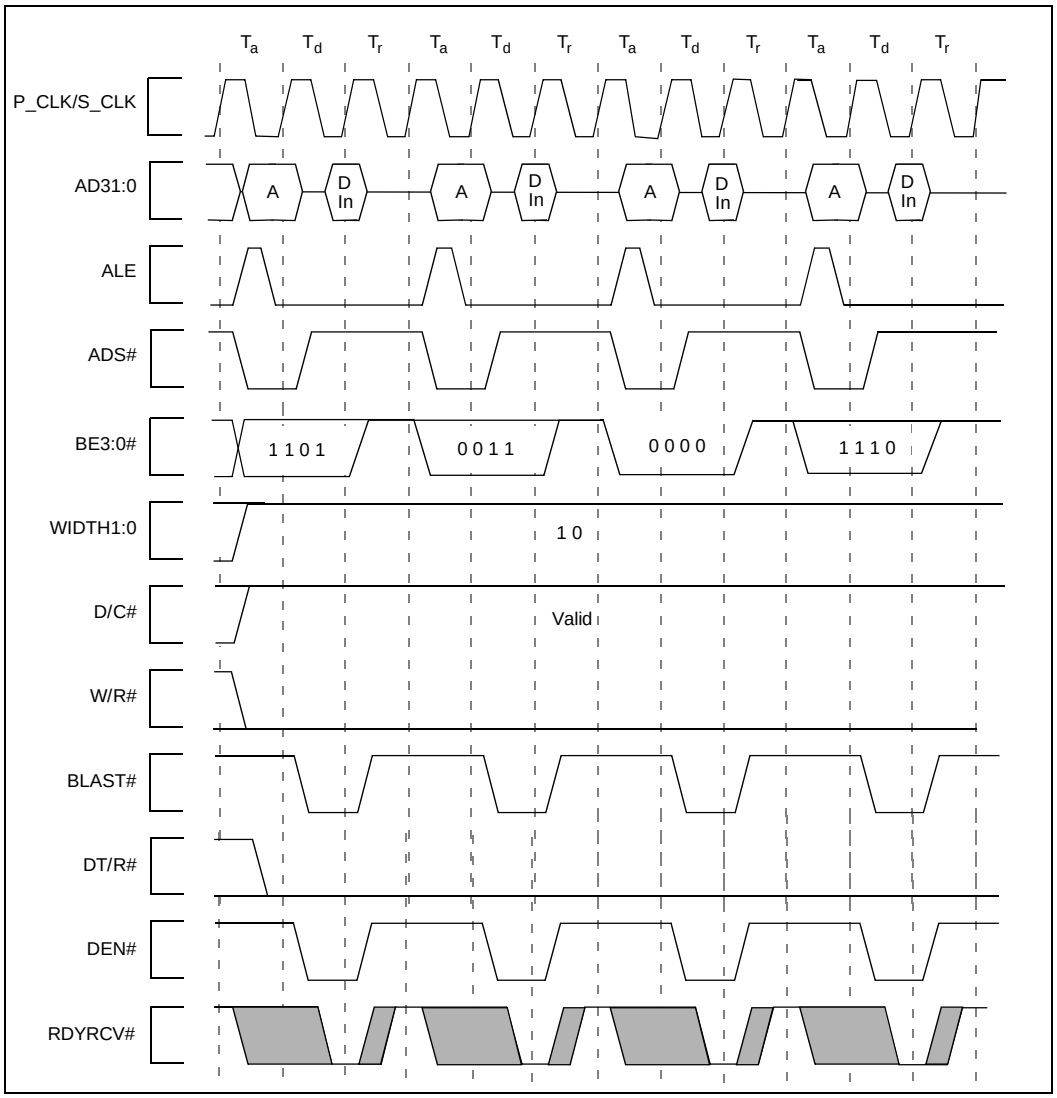
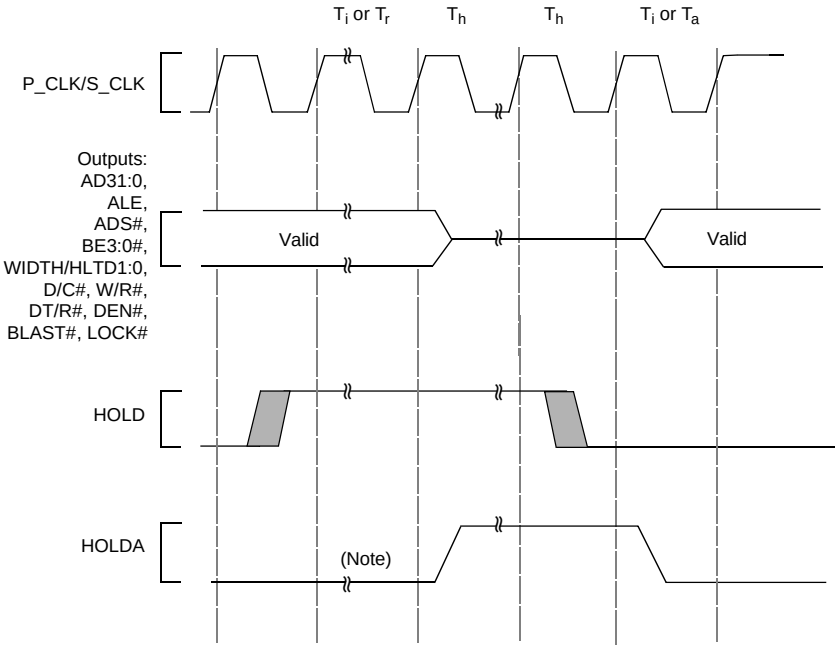


Figure 10. Bus Transactions Generated by Double Word Read Bus Request, Misaligned One Byte From Quad Word Boundary, 32-Bit 80960 Local Bus, Little Endian



NOTE: HOLD is sampled on the rising edge of CLKIN. The processor asserts HOLDA to grant the bus on the same edge in which it recognizes HOLD if the last state was T_i or the last T_r of a bus transaction. Similarly, the processor deasserts HOLDA on the same edge in which it recognizes the deassertion of HOLD.

Figure 11. HOLD/HOLDA Waveform For Bus Arbitration

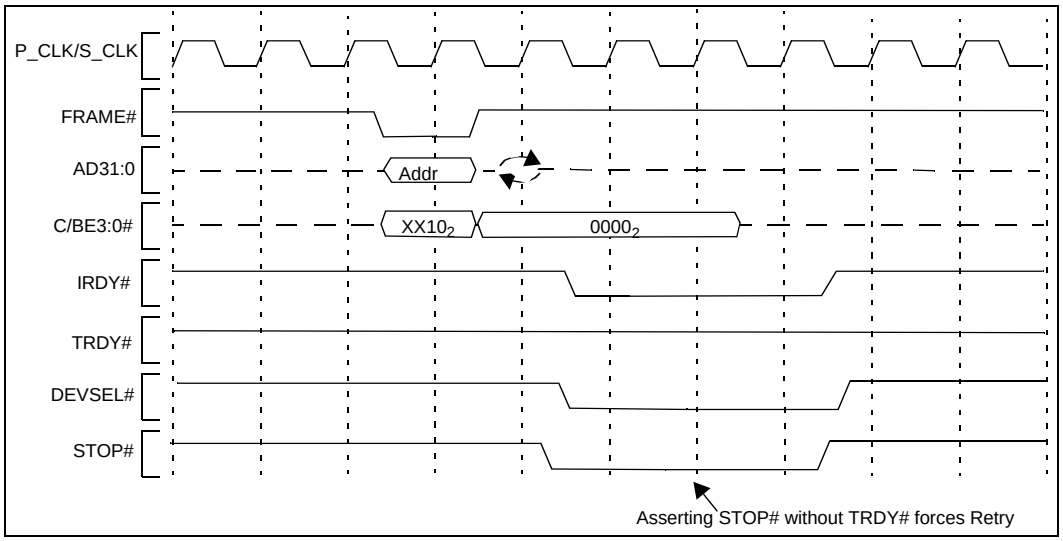


Figure 12. PCI Delayed Read Request

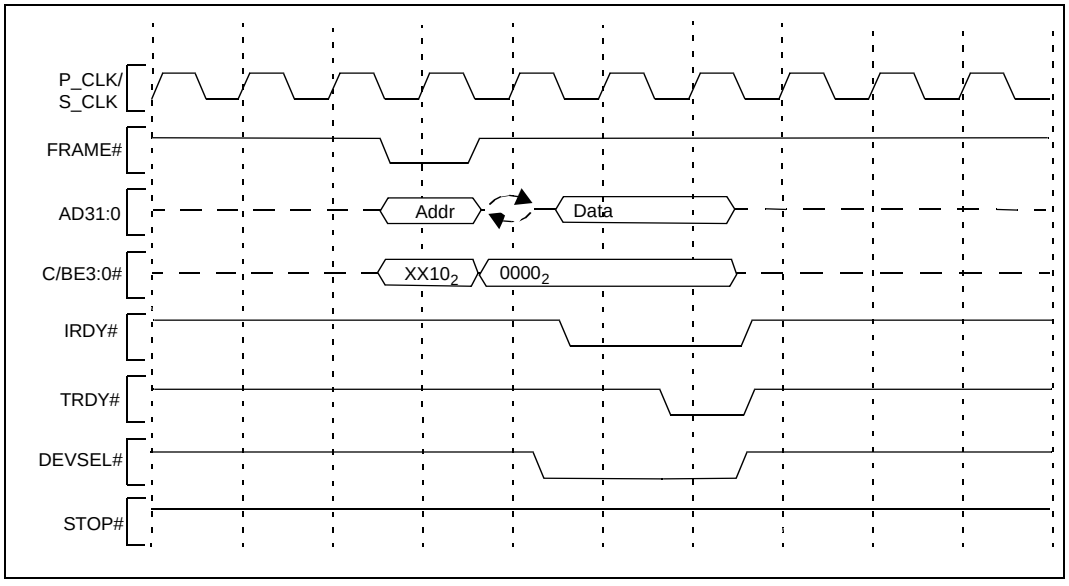


Figure 13. PCI Delayed Read Completion

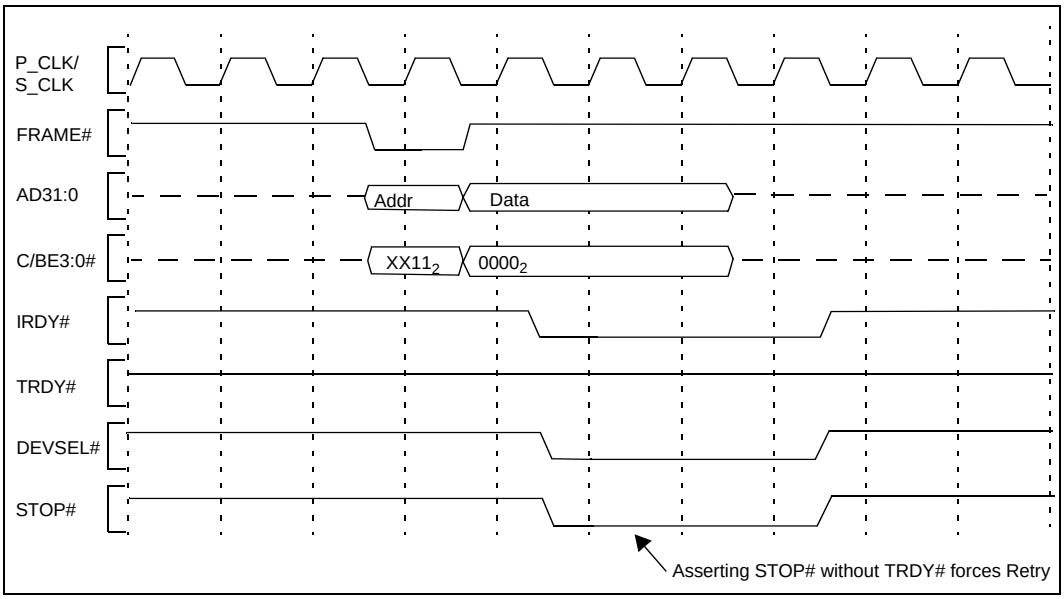


Figure 14. PCI Delayed Write Request

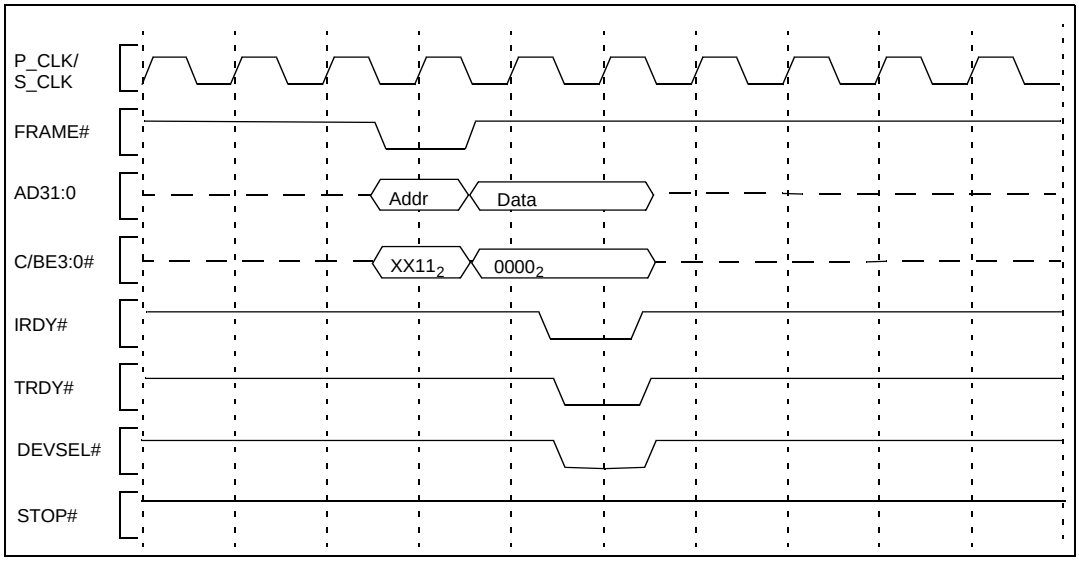


Figure 15. PCI Delayed Write Completion

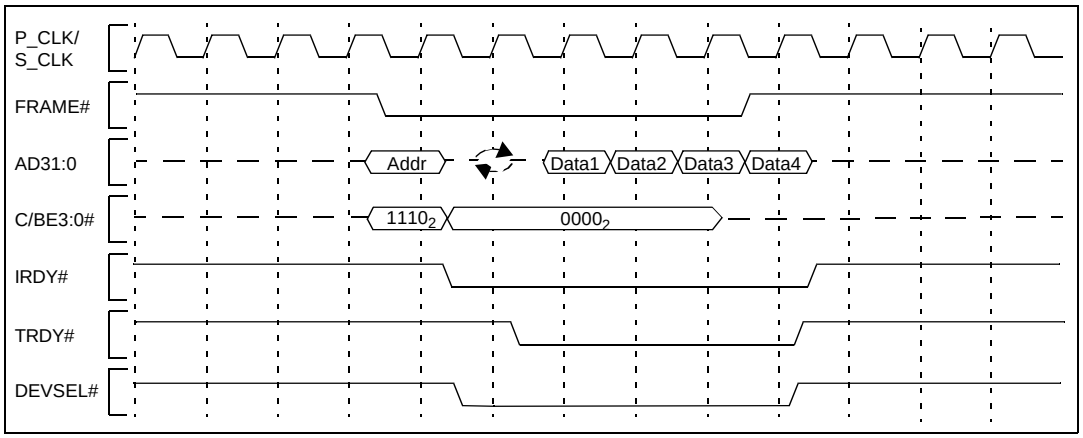


Figure 16. PCI Delayed Memory Read Line Completion Up to 4 DWORD Cacheline Boundary

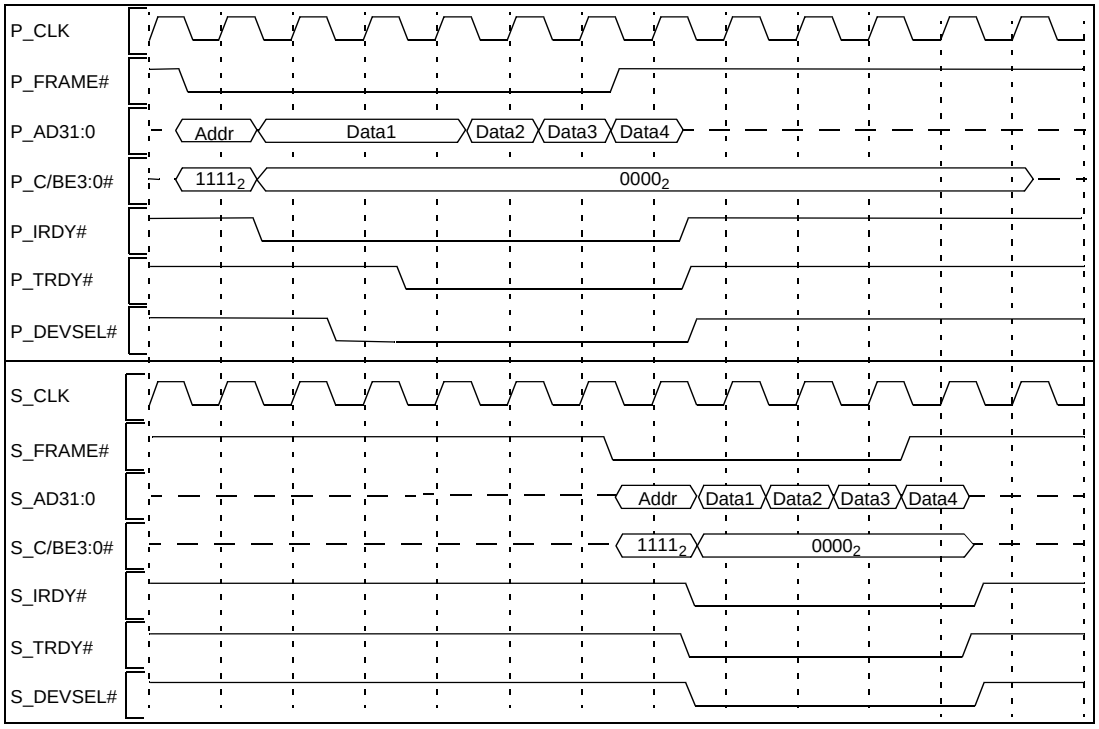


Figure 17. PCI Posted Memory Write and Invalidate Up to 4 DWORD Cacheline Boundary

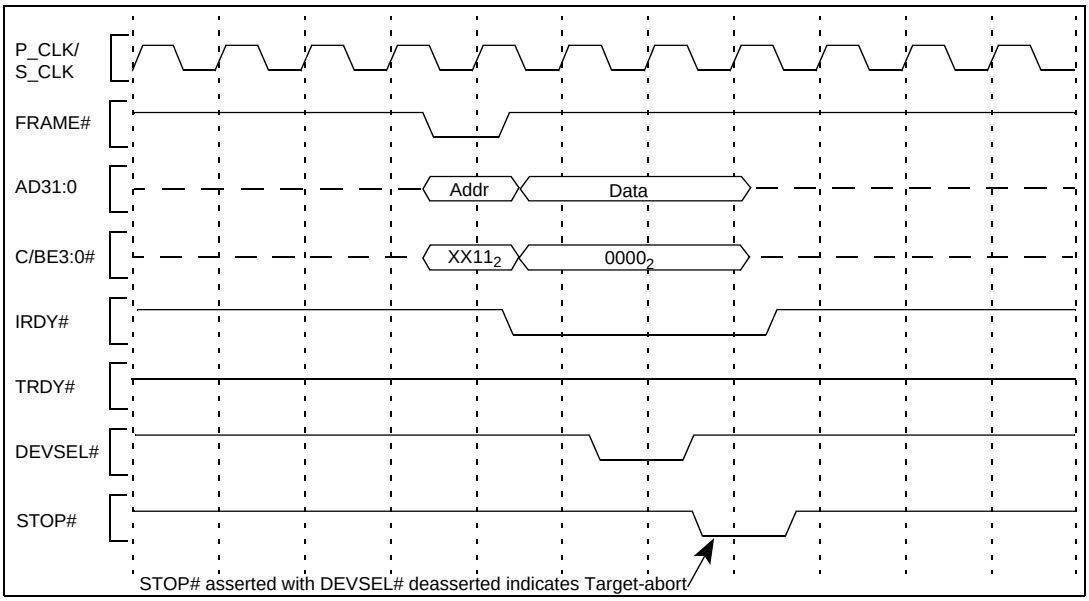


Figure 18. PCI Delayed Write with Target-Abort

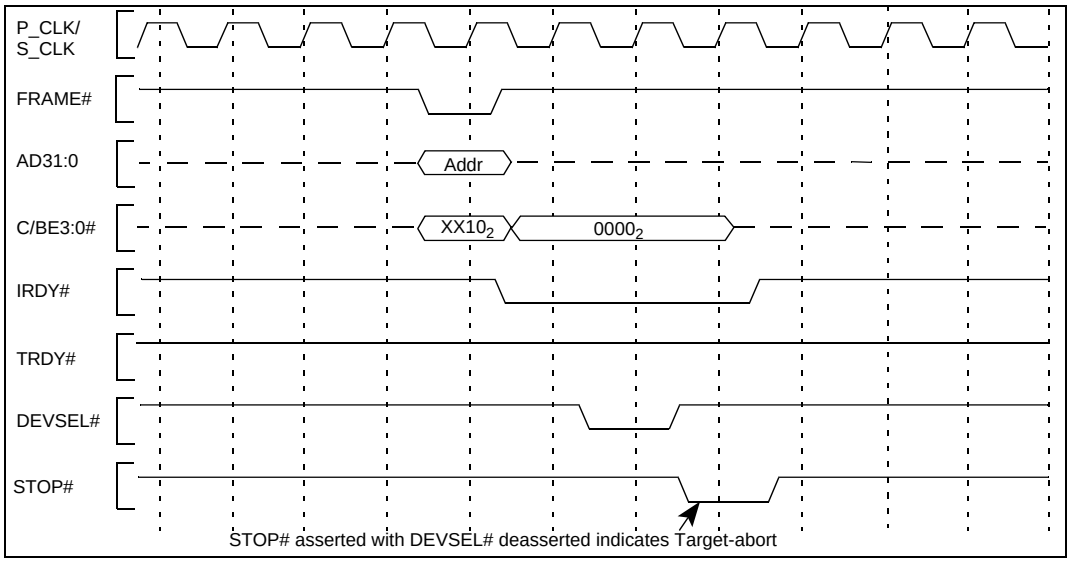


Figure 19. PCI Delayed Read with Target-Abort

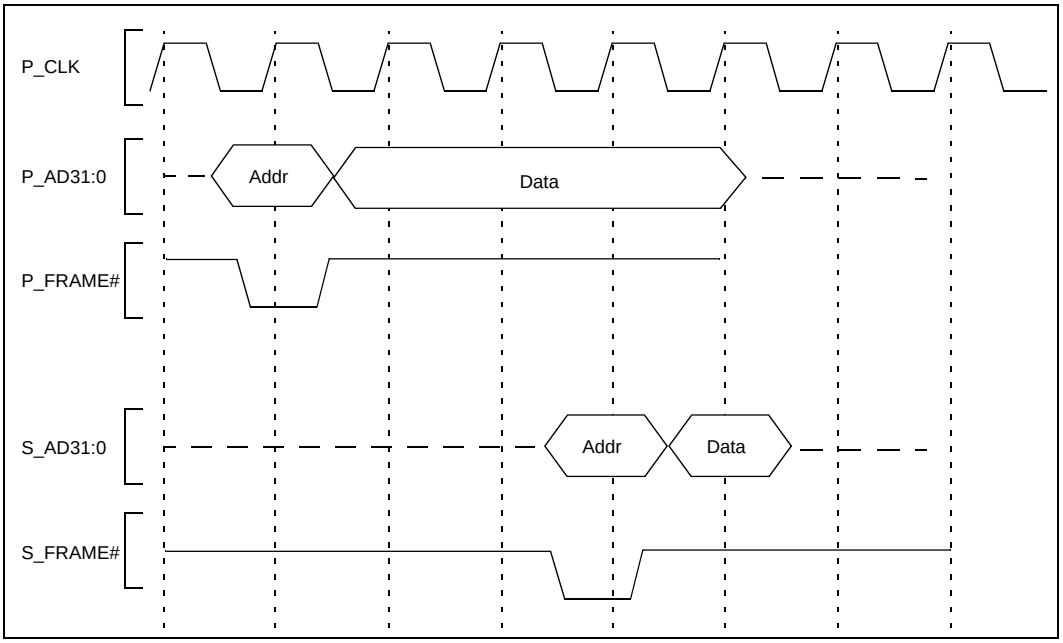


Figure 20. PCI Three Cycle Bridge Latency

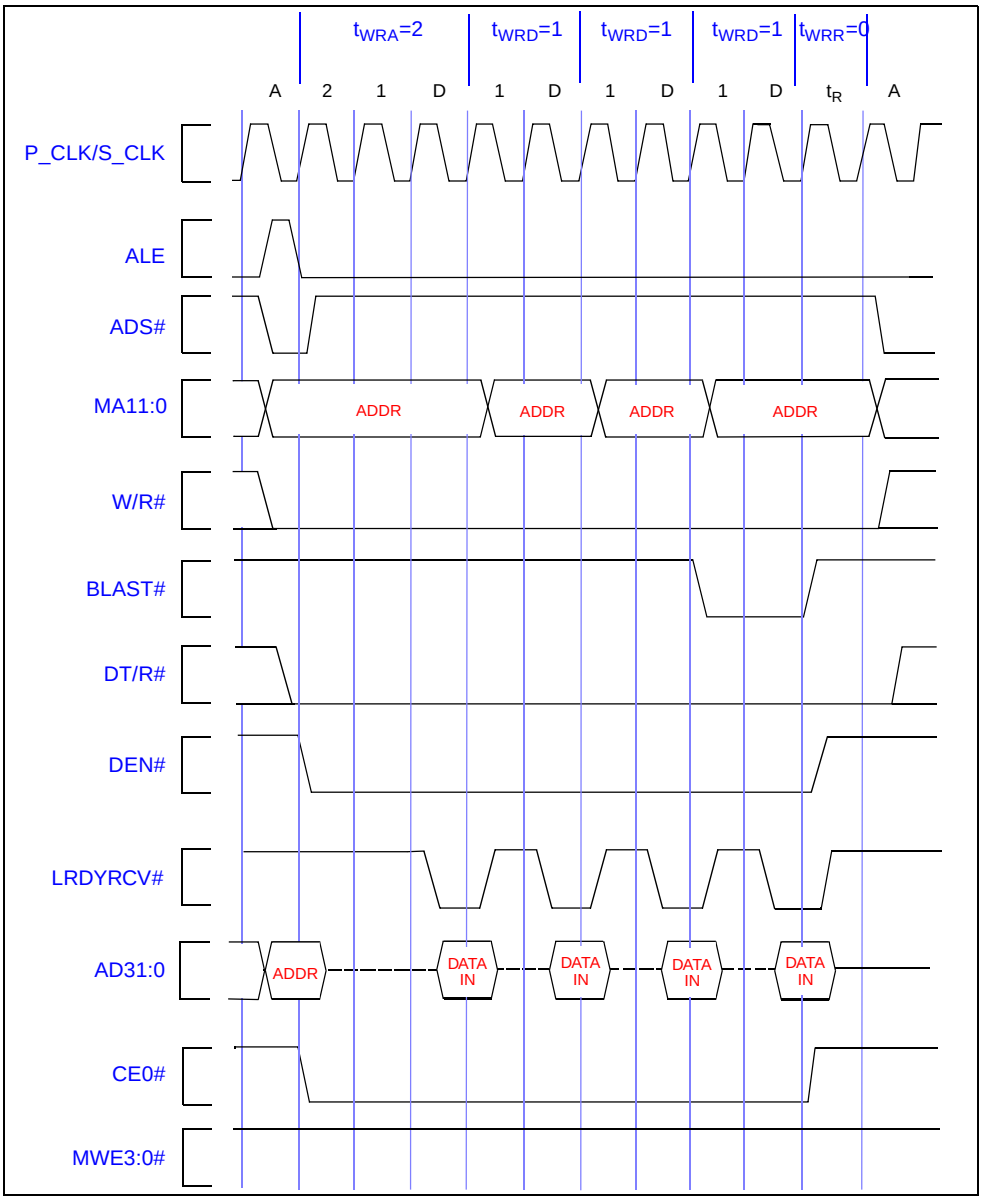


Figure 21. 32-Bit Bus, Burst Flash Memory, Read Access with 2,1,1,1 Wait States

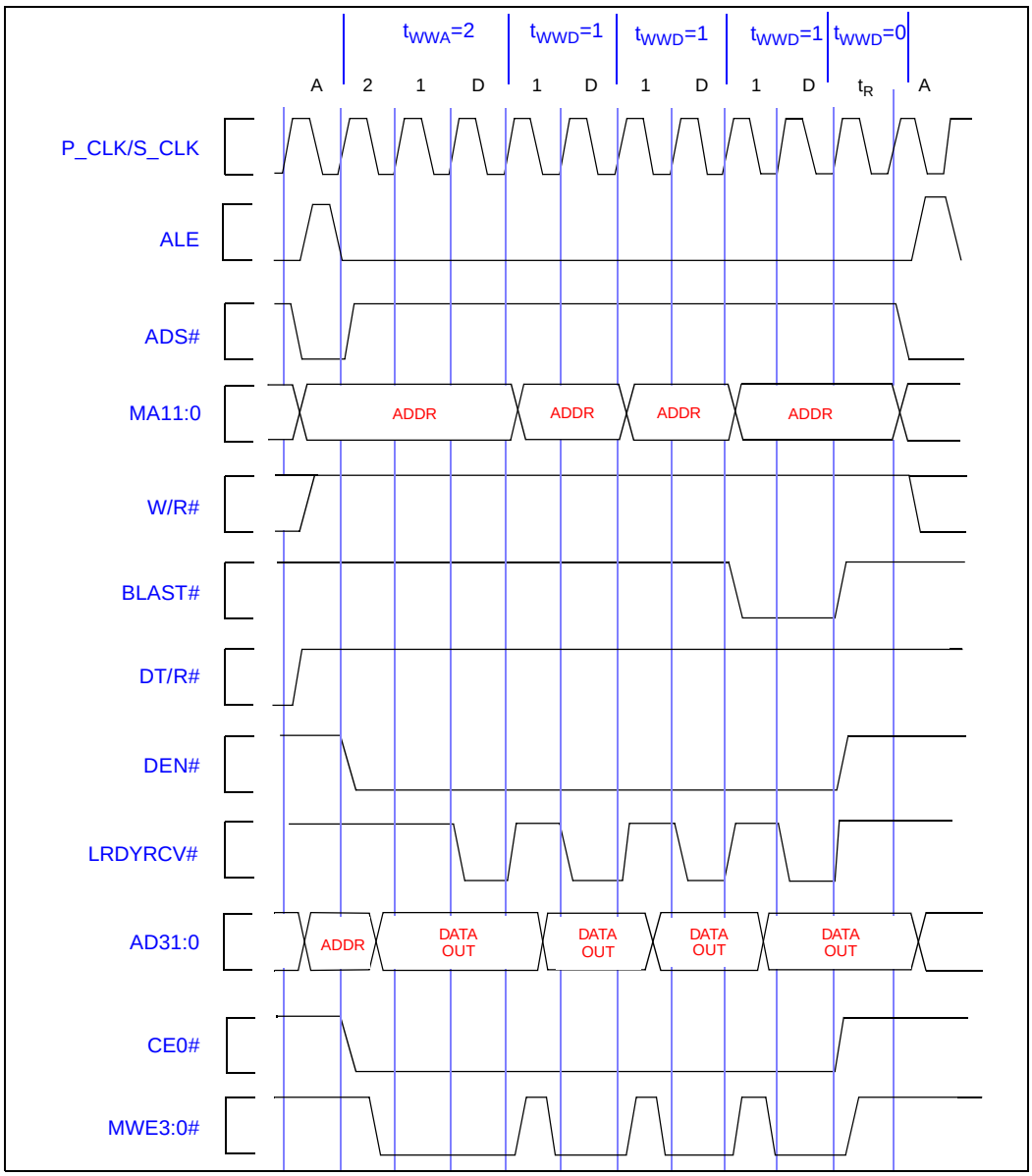


Figure 22. 32-Bit Bus, Burst Flash Memory Write Access with 2,1,1,1, Wait States

.

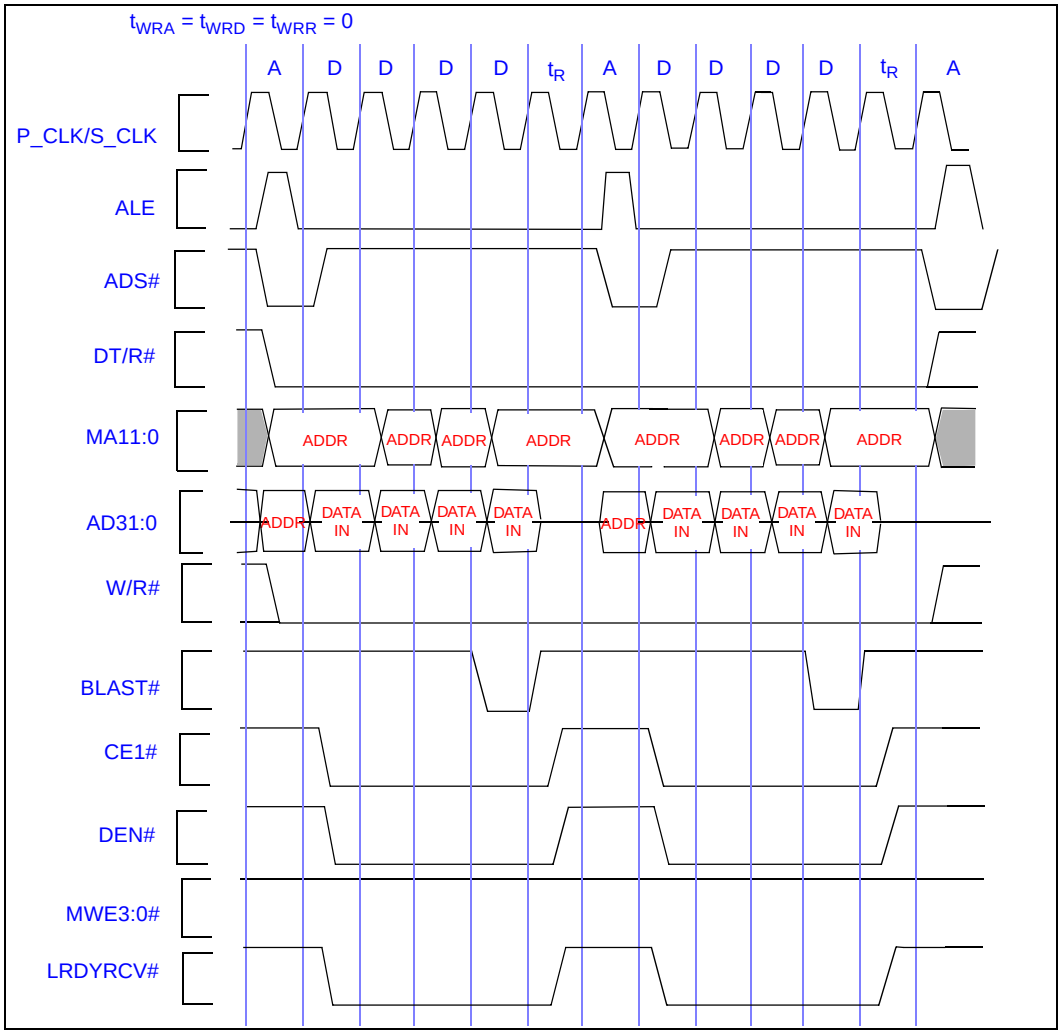


Figure 23. 32-Bit Bus, SRAM Read Accesses with 0 Wait States

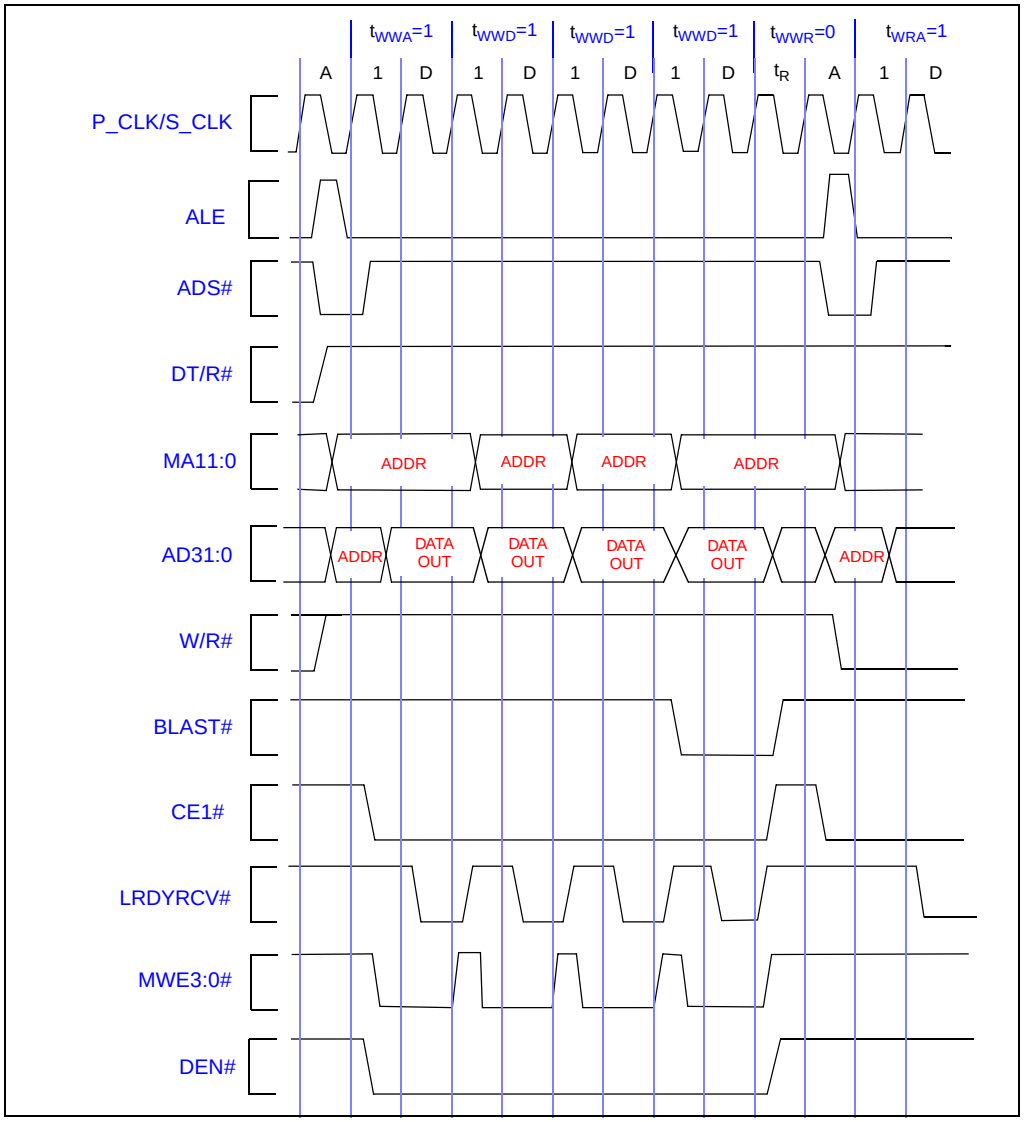


Figure 24. 32-Bit Bus, SRAM Write Access With 1,1,1,1 Wait States

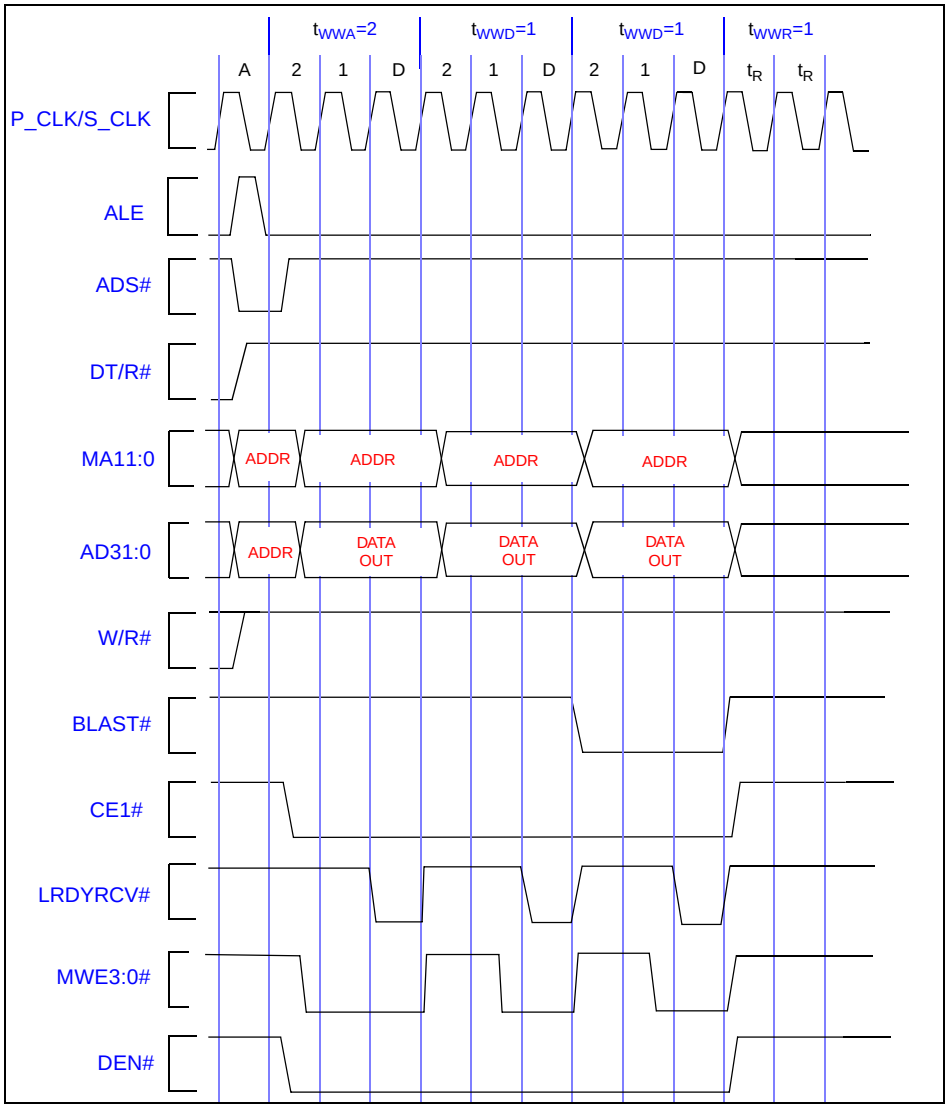


Figure 25. 32-Bit Bus, Write Access With 1 Wait State and Extended MWE3:0#

.

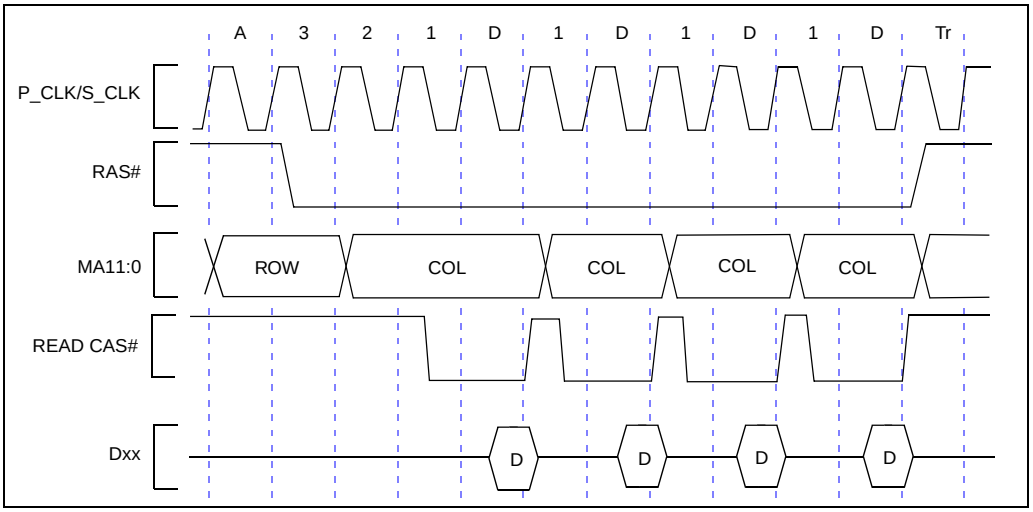


Figure 26. Fast Page-Mode DRAM, Read Cycle, CAS# Characteristics

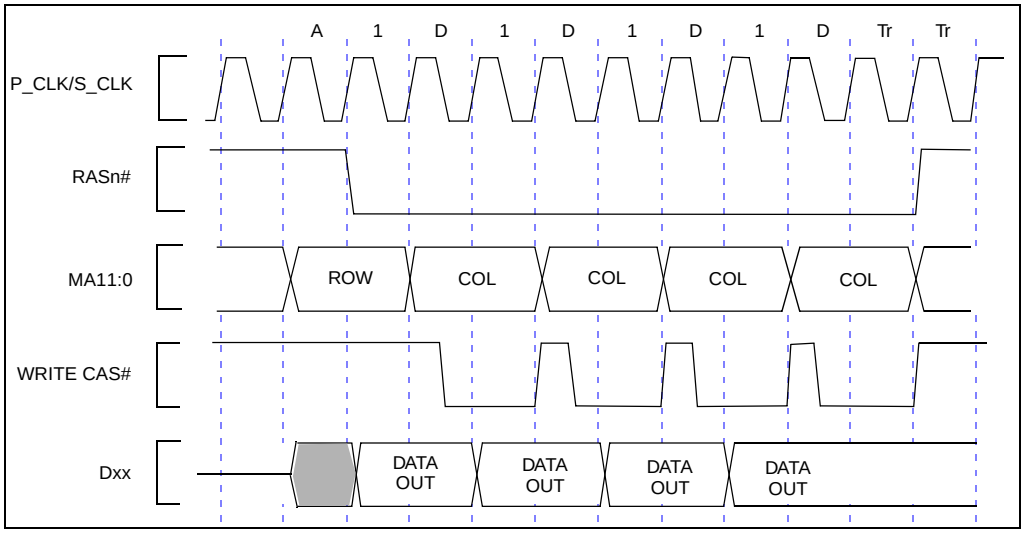


Figure 27. Fast Page-Mode DRAM, Write Cycle, CAS# Characteristics

.

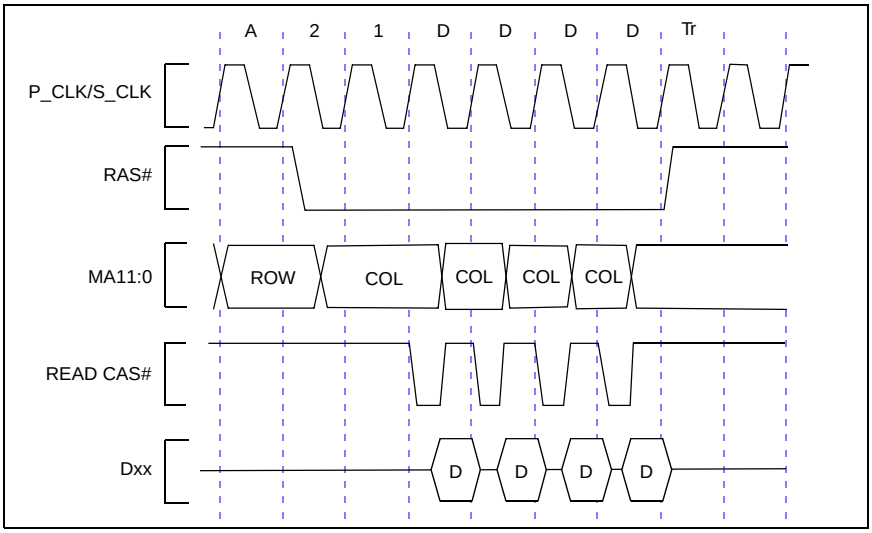


Figure 28. EDO DRAM, Read Cycle, CAS# Characteristics

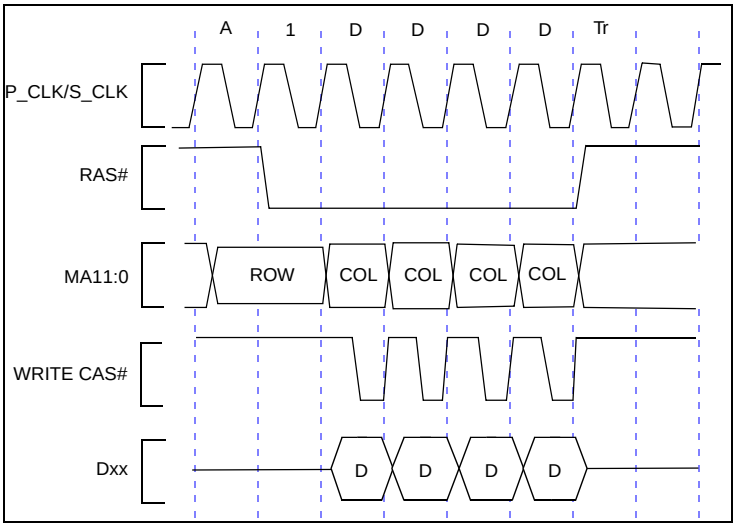


Figure 29. EDO DRAM, Write Cycle, CAS# Characteristics

.

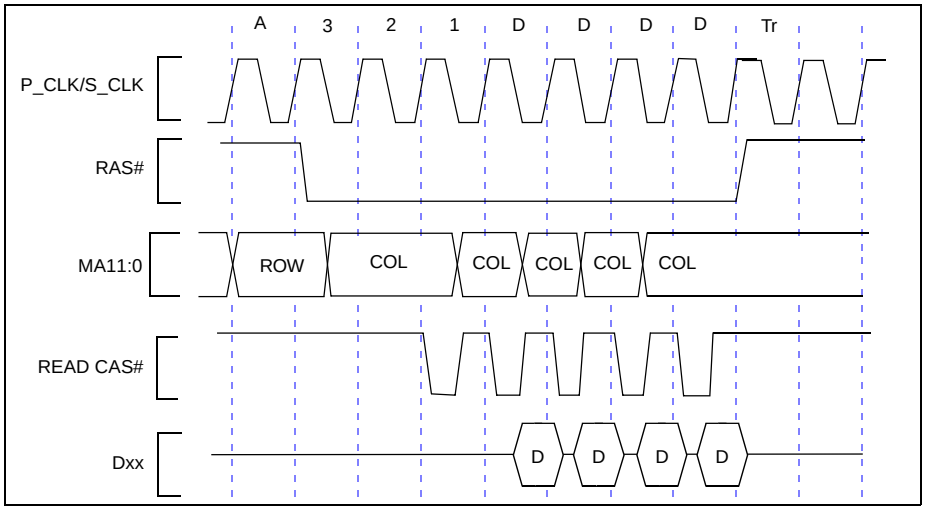


Figure 30. BEDO DRAM, Read Cycle, CAS# Characteristics

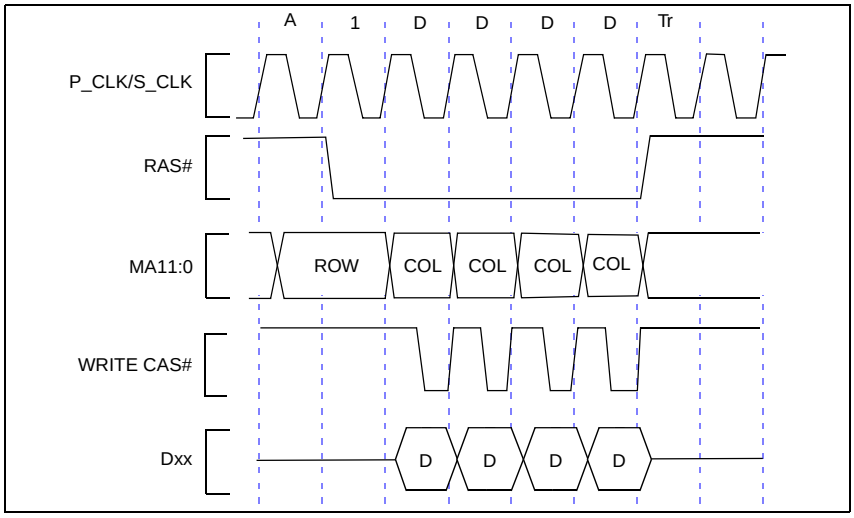


Figure 31. BEDO DRAM, Write Cycle, CAS# Characteristics

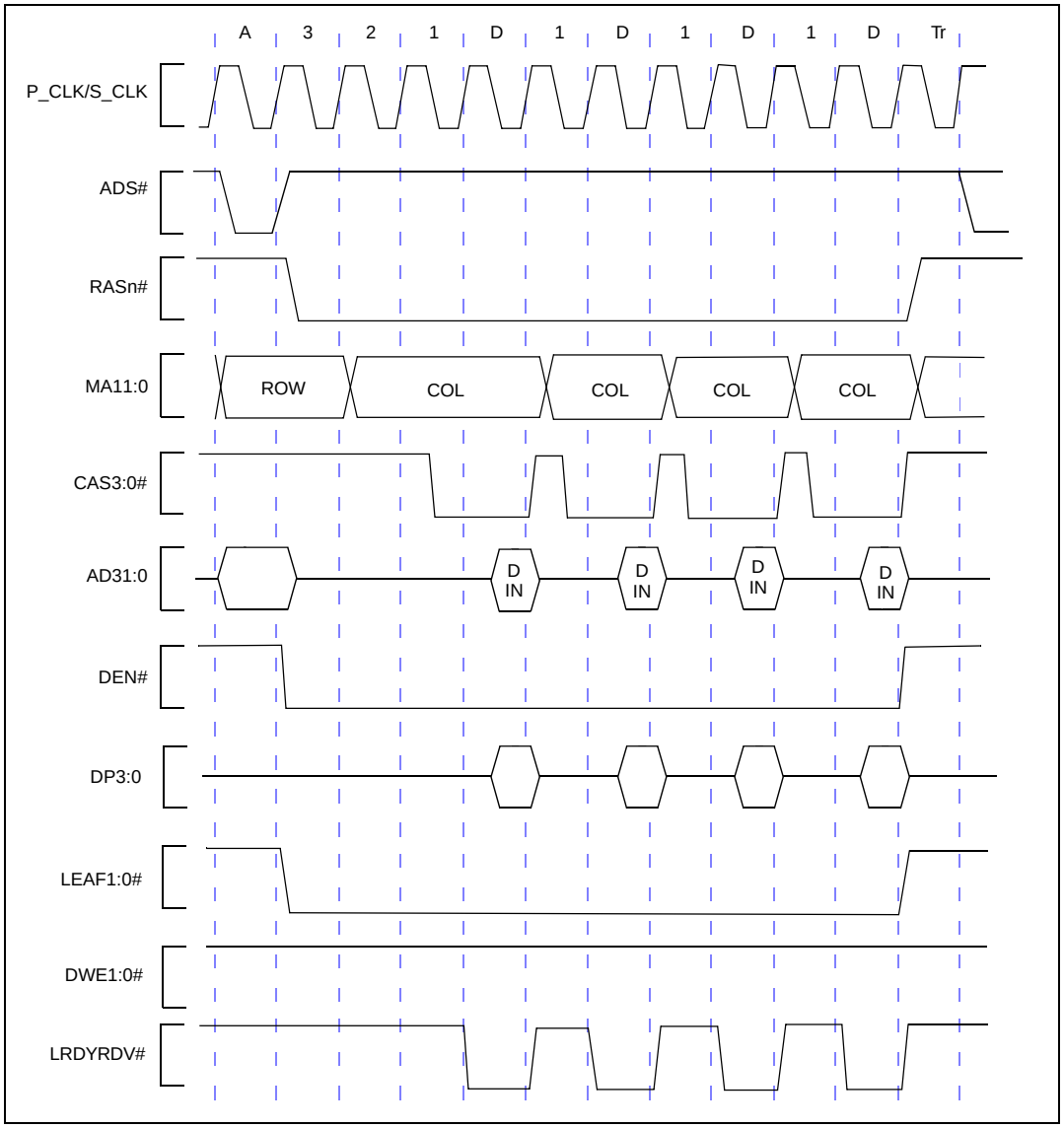


Figure 32. FPM DRAM System Read Access, Non-Interleaved, 3,1,1,1, Wait States

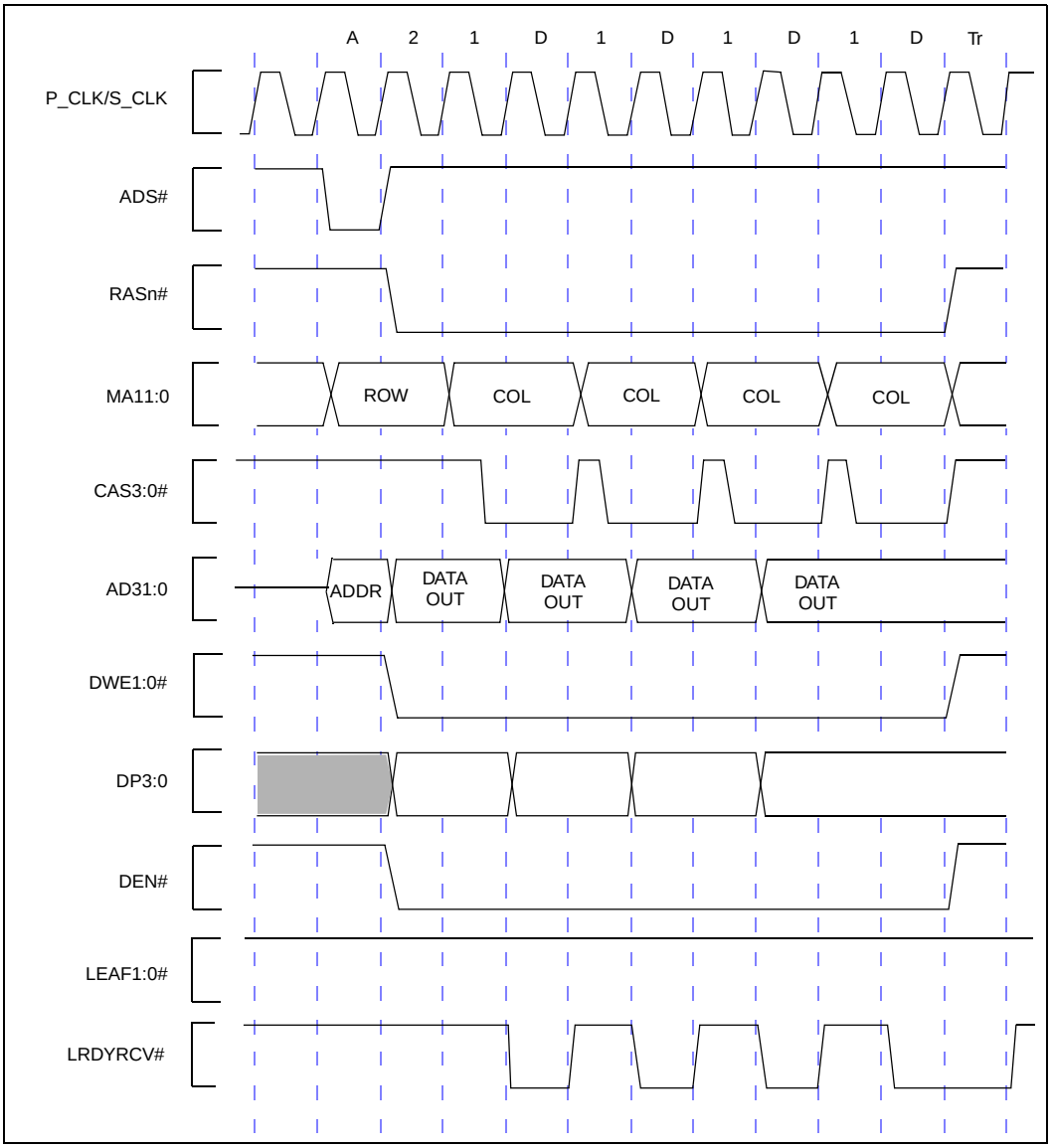


Figure 33. FPM DRAM System Write Access, Non-Interleaved, 2,1,1,1 Wait States

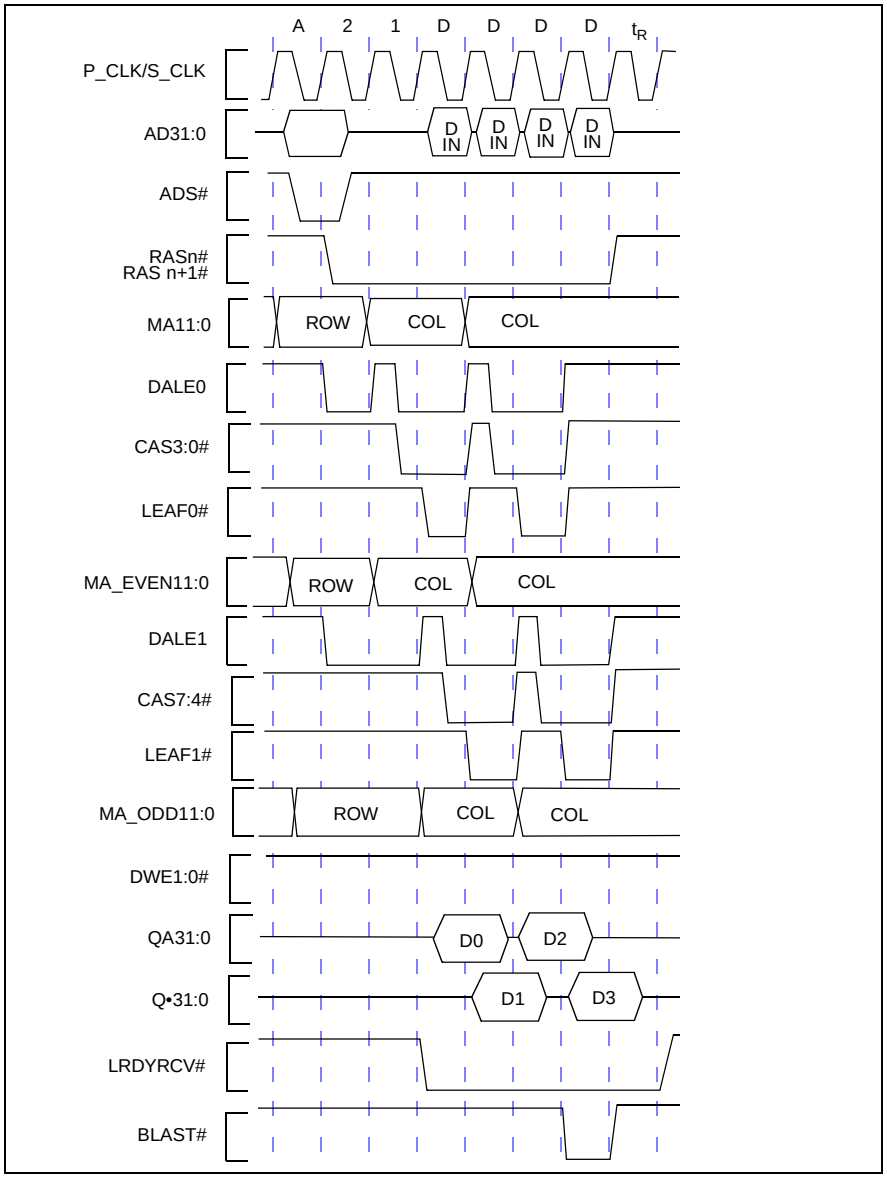


Figure 34. FPM DRAM System Read Access, Interleaved, 2,0,0,0 Wait States (Even Bank Accessed First)

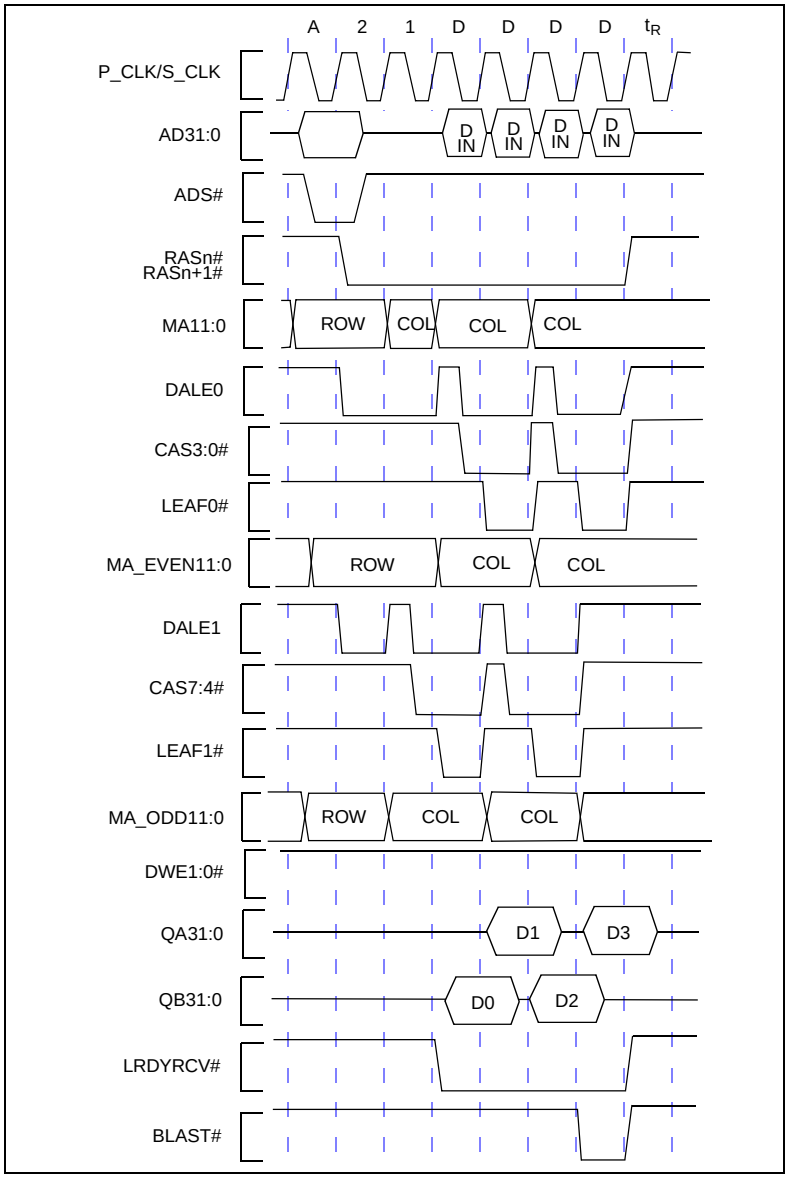


Figure 35. FPM DRAM System Read Access, Interleaved, 2,0,0,0 Wait States (Odd Bank Accessed First)

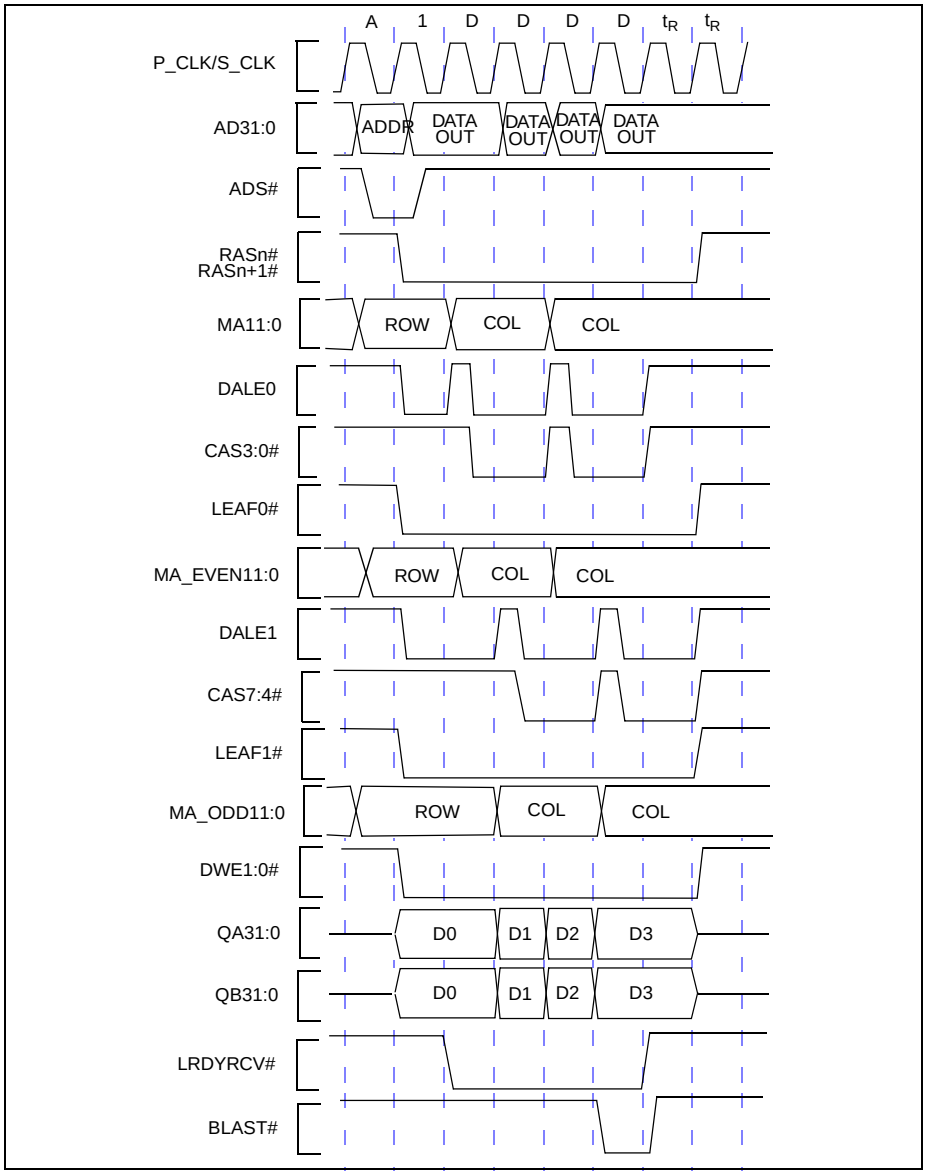


Figure 36. FPM DRAM System Write Access, Interleaved, 1,0,0,0 Wait States (1 Extra Recovery State)

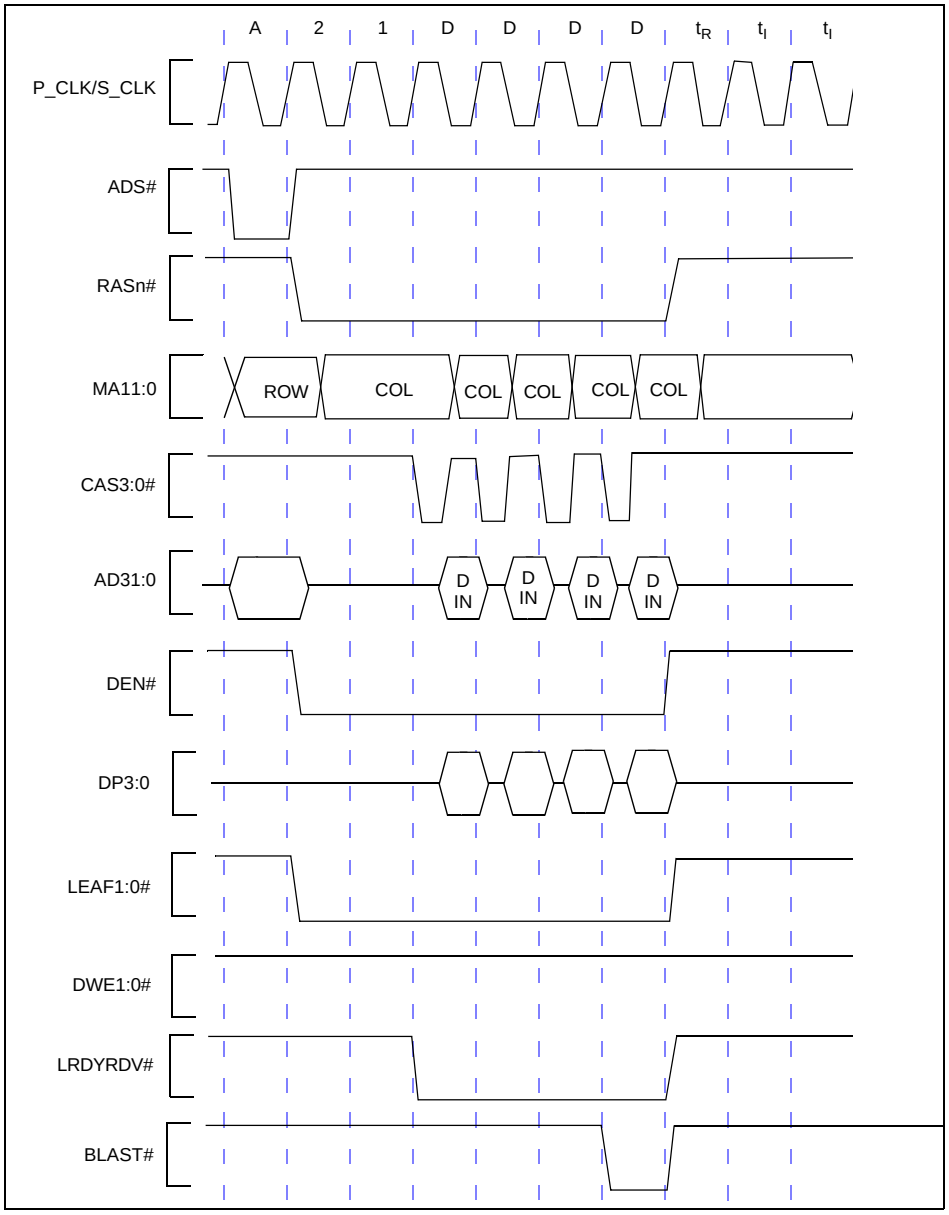


Figure 37. EDO DRAM System Read Access, 2,0,0,0, Wait States

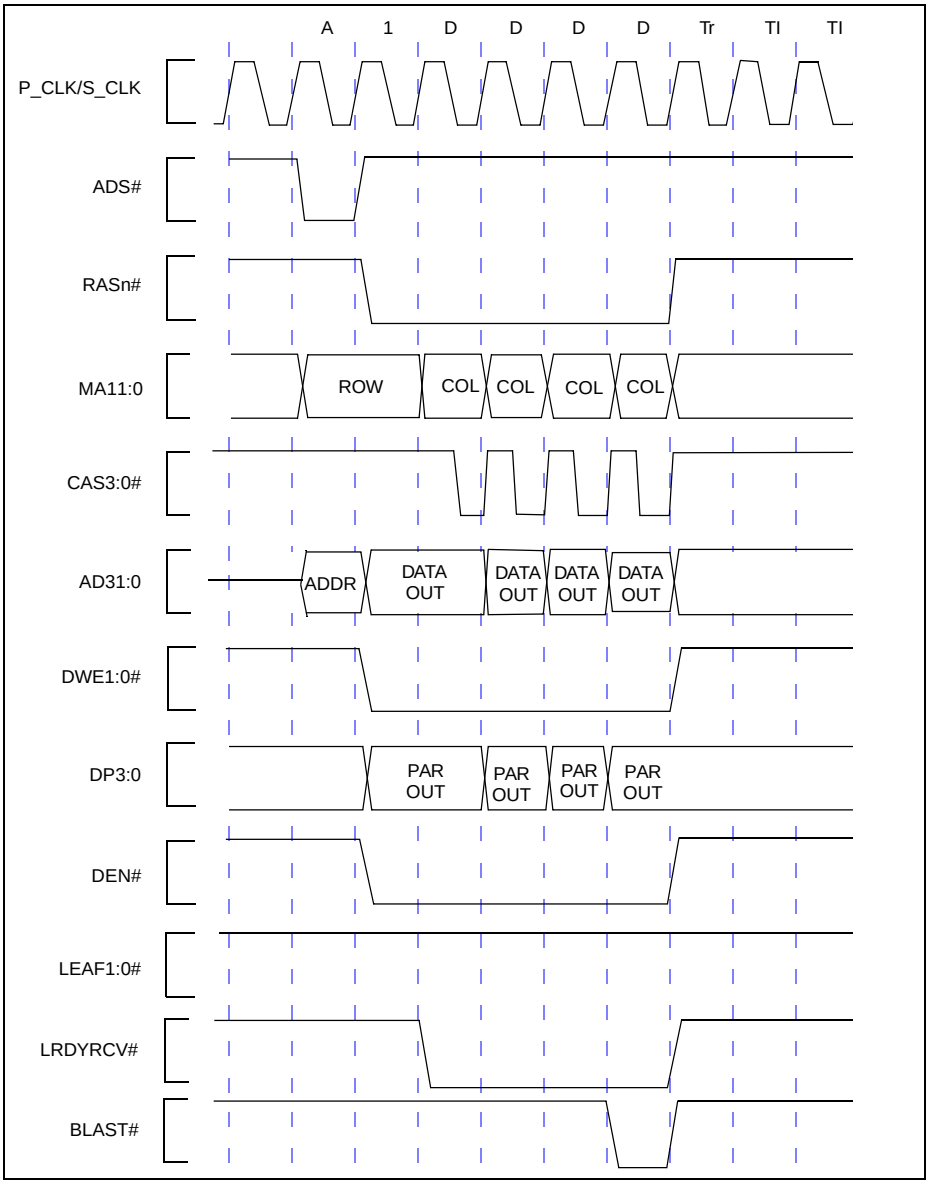


Figure 38. EDO DRAM System Write Access, 1,0,0,0 Wait States

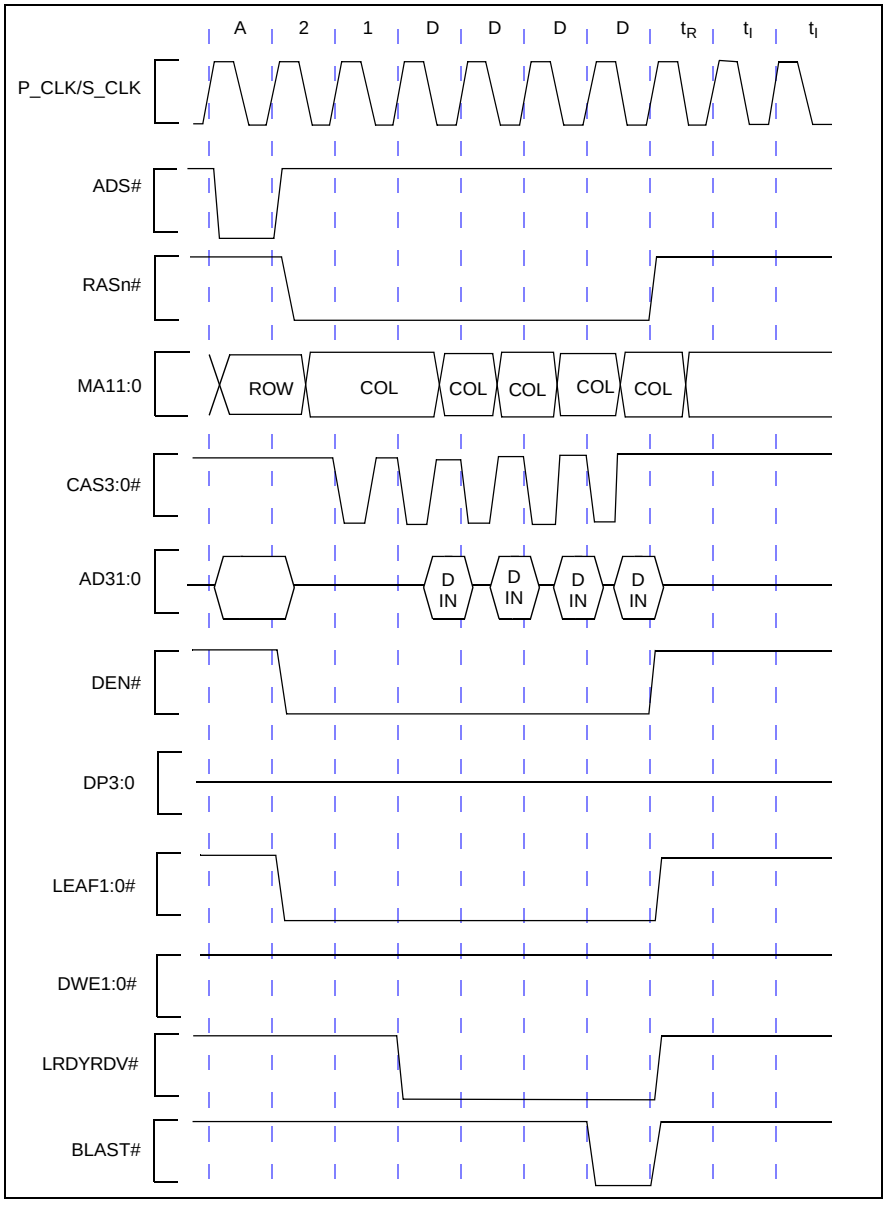


Figure 39. BEDO DRAM System Read Access, 2,0,0,0, Wait States

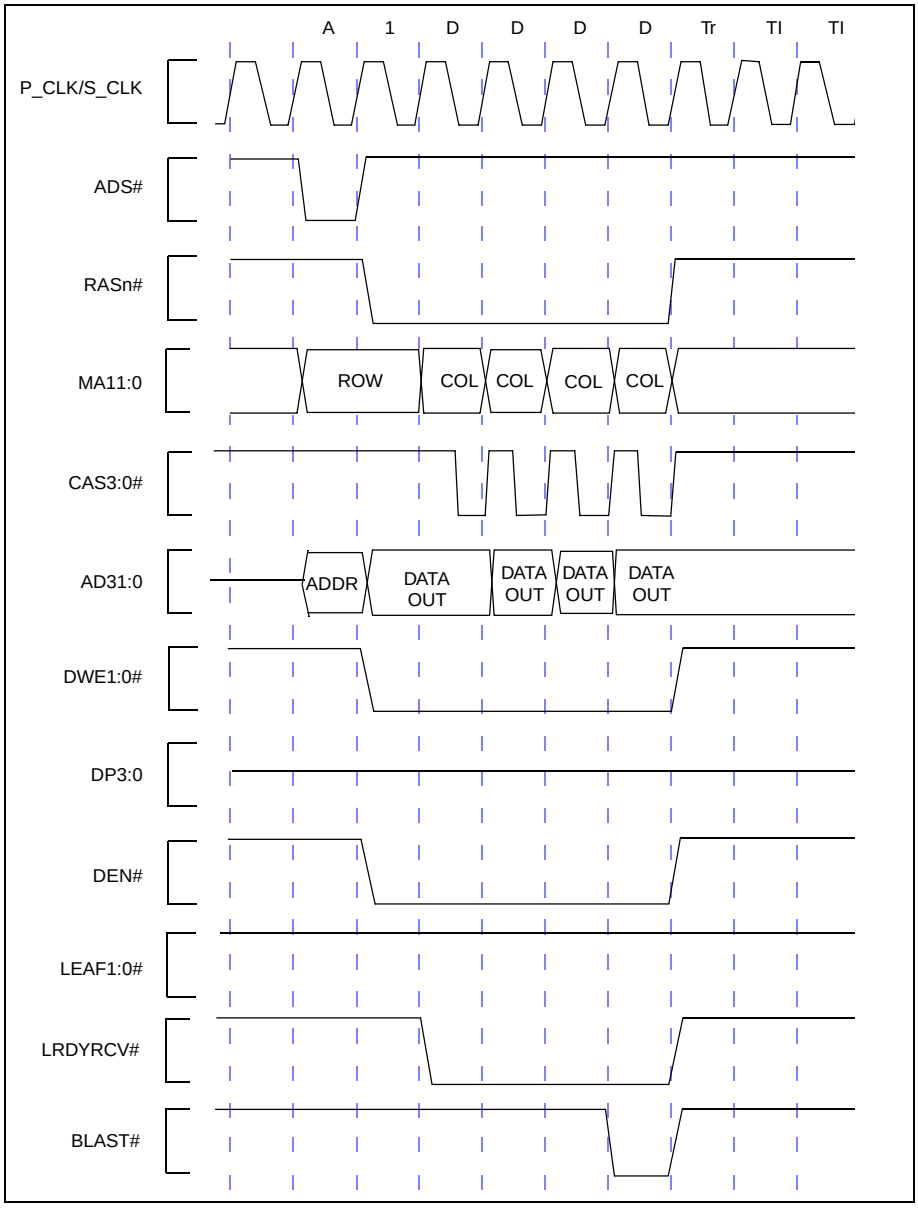


Figure 40. BEDO DRAM System Write Access, 1,0,0,0 Wait States

.

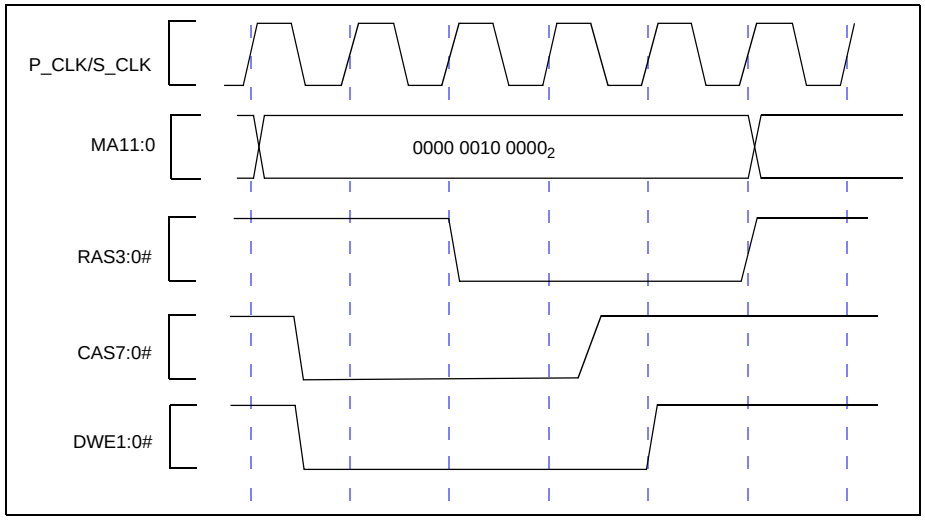


Figure 41. BEDO WBCR Program Cycle