80960HA/HD/HT 32-Bit High-Performance Superscalar Processor

Datasheet

Product Features

- 32-Bit Parallel Architecture
  - — Load/Store Architecture
  - — Sixteen 32-Bit Global Registers
  - — Sixteen 32-Bit Local Registers
  - — 1.28 Gbyte Internal Bandwidth (80 MHz)
  - — On-Chip Register Cache
- Processor Core Clock
  - — 80960HA is 1x Bus Clock
  - — 80960HD is 2x Bus Clock
  - — 80960HT is 3x Bus Clock
- Binary Compatible with Other 80960 Processors
- Issue Up To 150 Million Instructions per Second
- High-Performance On-Chip Storage
  - — 16 Kbyte Four-Way Set-Associative Instruction Cache
  - — 8 Kbyte Four-Way Set-Associative Data Cache
  - — 2 Kbyte General Purpose RAM
- Separate 128-Bit Internal Paths For Instructions/Data
- 3.3 V Supply Voltage
  - — 5 V Tolerant Inputs
  - — TTL Compatible Outputs
- Guarded Memory Unit
  - — Provides Memory Protection
  - — User/Supervisor Read/Write/Execute
- 32-Bit Demultiplexed Burst Bus
  - — Per-Byte Parity Generation/Checking
  - — Address Pipelining Option
  - — Fully Programmable Wait State Generator
  - — Supports 8-, 16- or 32-Bit Bus Widths
  - — 160 Mbyte/s External Bandwidth (40 MHz)
- High-Speed Interrupt Controller
  - — Up to 240 External Interrupts
  - — 31 Fully Programmable Priorities
  - — Separate, Non-maskable Interrupt Pin
- Dual On-Chip 32-Bit Timers
  - — Auto Reload Capability and One-Shot
  - — CLKIN Prescaling, divided by 1, 2, 4 or 8
  - — JTAG Support - IEEE 1149.1 Compliant

Order Number: 272495-009
August 2004

Contents

Figures

Tables

Revision History

Date	Revision	History
August 2004	009	To address the fact that many of the package prefix variables have changed, all package prefix variables in this document are now indicated with an "x".
September 2002	008	Formatted the datasheet in a new template. In "32-Bit Parallel Architecture" on page 1: • Removed operating frequency of 16/32 (bus/core) from 80960HD. • Removed operating frequency of 20/60 (bus/core) from 80960HT. In Table 5 "80960HA/HD/HT Package Types and Speeds" on page 14: • Removed core speed of 32 MHz and bus speed of 16 MHz, and order number A80960HD32-S-L2GG from the 168L PGA package, 80960HD device. • Removed core speed of 60 MHz and bus speed of 20 MHz, and order number A80960HT60 from the 168L PGA package, 80960HT device. • Removed core speed of 32 MHz and bus speed of 16 MHz, and order number FC80960HD32-S-L2GL from the 208L PQFP package, 80960HD device. • Removed core speed of 60 MHz and bus speed of 20 MHz, and order number FC80960HT60-S-L2G2 from the 208L PQFP package, 80960HT device.
July 1998	007	In "32-Bit Parallel Architecture" on page 1: • Revised 1.2 Gbyte Internal Bandwidth (75 MHz) to 1.28 Gbyte Internal Bandwidth (80 MHz). In Section 3.0, "Package Information" on page 14: • Added paragraph two and Table 5 "80960HA/HD/HT Package Types and Speeds" on page 14. In Table 7 "80960Hx Processor Family Pin Descriptions" on page 16: • Corrected minor typeset and spacing errors. • BREQ; Revised description. • $\overline{\text{ONCE}}$; last sentence, changed 'low' to 'high'. • TDI and TMS; removed last sentence stating, "Pull this pin low when not in use." In Figure 2 "80960Hx 168-Pin PGA Pinout—View from Top (Pins Facing Down)" on page 20: • Added insert package marking diagram. In Figure 4 "80960Hx 208-Pin PQ4 Pinout" on page 26: • Added insert package marking diagram. In Table 10 "80960Hx PQ4 Pinout—Signal Name Order" on page 27: • Corrected TDO ('O' was zero) and revised alphabetical ordering. In Table 11 "80960Hx PQ4 Pinout—Pin Number Order" on page 29: • Corrected TDO ('O' was zero) and revised alphabetical ordering. In Section 4.1, "Absolute Maximum Ratings" on page 37: • Revised V_{CC} to VCC5 for Voltage on Other Pins with respect to V_{SS}. In Section 4.5, "VCCPLL Pin Requirements" on page 39: • Added section. In Table 22 "80960Hx DC Characteristics" on page 40: • Added footnote (1) to I_{LO} notes column for TDO pin. • Added footnote (10) to C_{IN}, C_{OUT} and $C_{I/O}$ pin.

Date	Revision	History
July 1998 (continued)	007 (continued)	In Table 23 “80960Hx AC Characteristics” on page 42: • Added overbars where required. • Modified T_{DVNH} to list separate specifications for 3.3 V and 5 V. • Modified T_{OV2}, T_{OH2} and T_{TVEL} to reflect specific 80960HA, 80960HD and 80960HT values. In Figure 23 “ICC Active (Power Supply) vs. Frequency” on page 51: • Changed ‘5’ to ‘0’ on the CLKIN Frequency axis. In Figure 49 “BREQ and BSTALL Operation” on page 74: • Added figure and following text.
August 1997	006	Fixed several font and format issues.

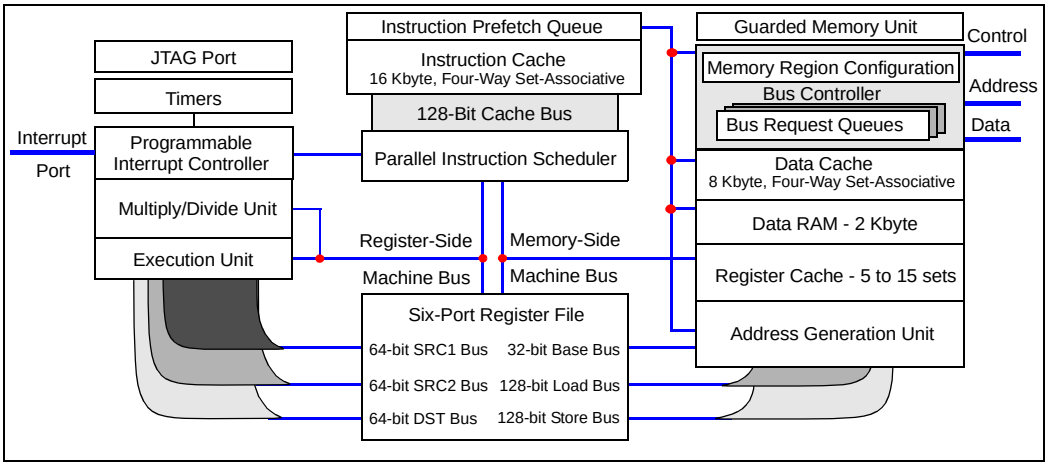
This page intentionally left blank.

1.0 About This Document

This document describes the parametric performance of Intel's 80960Hx embedded superscalar microprocessors. Detailed descriptions for functional topics, other than parametric performance, are published in the *i960® Hx Microprocessor User's Guide* (272484).

In this document, '80960Hx' and 'i960 Hx processor' refer to the products described in Table 1. Throughout this document, information that is specific to each is clearly indicated.

Figure 1. 80960Hx Block Diagram



2.0 Intel 80960Hx Processor

The Intel 80960Hx processor provides new performance levels while maintaining backward compatibility (pin[1] and software) with the i960 CA/CF processor. This newest member of the family of i960 32-bit, RISC-style, embedded processors allows customers to create scalable designs that meet multiple price and performance points. This is accomplished by providing processors that may run at the bus speed or faster using Intel's clock multiplying technology (see Table 1). The 80960Hx core is capable of issuing 150 million instructions per second, using a sophisticated instruction scheduler that allows the processor to sustain a throughput of two instructions every core clock, with a peak performance of three instructions per clock. The 80960Hx-series comprises three processors, which differ in the ratio of core clock speed to external bus speed.

Table 1. 80960Hx Product Description

Product	Core	Voltage	Operating Frequency (bus/core)
80960HA	1x	3.3 V†	25/25, 33/33, 40/40
80960HD	2x	3.3 V†	25/50, 33/66, 40/80
80960HT	3x	3.3 V†	25/75

† Processor inputs are 5 V tolerant.

1. The 80960Hx is not "drop-in" compatible in an 80960Cx-based system. Customers may design systems that accept either 80960Hx or Cx processors.

In addition to expanded clock frequency options, the 80960Hx provides essential enhancements for an emerging class of high-performance embedded applications. Features include a larger instruction cache, data cache, and data RAM than any other 80960 processor to date. It also boasts a 32-bit demultiplexed and pipelined burst bus, fast interrupt mechanism, guarded memory unit, wait state generator, dual programmable timers, ONCE and IEEE 1149.1-compliant boundary scan test and debug support, and new instructions.

2.1 The i960® Processor Family

The i960® processor family is a 32-bit RISC architecture created by Intel to serve the needs of embedded applications. The embedded market includes applications as diverse as industrial automation, avionics, image processing, graphics and communications.

Because all members of the i960 processor family share a common core architecture, i960 applications are code-compatible. Each new processor in the family adds its own special set of functions to the core to satisfy the needs of a specific application or range of applications in the embedded market.

2.2 Key 80960Hx Features

2.2.1 Execution Architecture

Independent instruction paths inside the processor allow the execution of multiple, out-of-sequence instructions per clock. Register and resource scoreboarding interlocks maintain the logical integrity of sequential instructions that are being executed in parallel. To sustain execution of multiple instructions in each clock cycle, the processor decodes multiple instructions in parallel and simultaneously issues these instructions to parallel processing units. The various processing units are then able to independently access instruction operands in parallel from a common register set.

Local Register Cache integrated on-chip provides automatic register management on call/return instructions. Upon a call instruction, the processor allocates a set of local registers for the called procedure, then stores the registers for the previous procedure in the on-chip register cache. As additional procedures are called, the cache stores the associated registers such that the most recently called procedure is the first available by the next return (**ret**) instruction. The processor may store up to fifteen register sets, after which the oldest sets are stored (spilled) into external memory.

The 80960Hx supports the 80960 architecturally-defined branch prediction mechanism. This allows many branches to execute with no pipeline break. With the 80960Hx's efficient pipeline, a branch may take as few as zero clocks to execute. The maximum penalty for an incorrect prediction is two core clocks.

2.2.2 Pipelined, Burst Bus

A 32-bit high performance bus controller interfaces the 80960Hx core to the external memory and peripherals. The Bus Control Unit features a maximum transfer rate of 160 Mbytes per second (at a 40 MHz external bus clock frequency). A key advantage of this design is its versatility. The user may independently program the physical and logical attributes of system memory. Physical attributes include wait state profile, bus width, and parity. Logical attributes include cacheability and Big or Little Endian byte order. Internally programmable wait states and 16 separately configurable physical memory regions allow the processor to interface with a variety of memory

subsystems with minimum system complexity. To reduce the effect of wait states, the bus design is decoupled from the core. This lets the processor execute instructions while the bus performs memory accesses independently.

The Bus Controller's key features include:

- Demultiplexed, Burst Bus to support most efficient DRAM access modes
- Address Pipelining to reduce memory cost while maintaining performance
- 32-, 16- and 8-bit modes to facilitate I/O interfacing
- Full internal wait state generation to reduce system cost
- Little and Big Endian support
- Unaligned Access support implemented in hardware
- Three-deep request queue to decouple the bus from the core
- Independent physical and logical address space characteristics

2.2.3 On-Chip Caches and Data RAM

As shown in Figure 1, the 80960Hx provides generous on-chip cache and storage features to decouple CPU execution from the external bus. The processor includes a 16 Kbyte instruction cache, an 8 Kbyte data cache and 2 Kbytes of Data RAM. The caches are organized as 4-way set associative. Stores that hit the data cache are written through to memory. The data cache performs write allocation on cache misses. A fifteen-set stack frame cache allows the processor to rapidly allocate and deallocate local registers. All of the on-chip RAM sustains a 4-word (128-bit) access every clock cycle.

2.2.4 Priority Interrupt Controller

The interrupt unit provides the mechanism for the low latency and high throughput interrupt service essential for embedded applications. A priority interrupt controller provides full programmability of 240 interrupt sources with a typical interrupt task switch (latency) time of 17 core clocks. The controller supports 31 priority levels. Interrupts are prioritized and signaled within 10 core clocks of the request. When the interrupt has a higher priority than the processor priority, the context switch to the interrupt routine would typically complete in another seven bus clocks.

External agents post interrupts through the 8-bit external interrupt port. The Interrupt unit also handles the two internal sources from the Timers. Interrupts may be level- or edge-triggered.

2.2.5 Guarded Memory Unit

The Guarded Memory Unit (GMU) provides memory protection without the address translation found in Memory Management Units. The GMU contains two memory protection schemes: one prevents illegal memory accesses, the other detects memory access violations. Both signal a fault to the processor. The programmable protection modes are: user read, write or execute; and supervisor read, write or execute.

2.2.6 Dual Programmable Timers

The processor provides two independent 32-bit timers, with four programmable clock rates. The user configures the timers through the Timer Unit registers. These registers are memory-mapped within the 80960Hx, 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 processor's interrupt controller.

2.2.7 Processor Self Test

When a system error is detected, the $\overline{\text{FAIL}}$ pin is asserted, a fail code message is driven onto the address bus, and the processor stops execution at the point of failure. The only way to resume normal operation is to perform a RESET operation. Because System Error generation may occur sometime after the bus confidence test and even after initialization during normal processor operation, the $\overline{\text{FAIL}}$ pin is HIGH (logic "1") before the detection of a System Error.

The processor uses only one read bus-transaction to signal the fail code message; the address of the bus transaction is the fail code itself. The fail code is of the form: **0xfeffff***nn*; bits 6 to 0 contain a mask recording the possible failures. Bit 7, when set to 1, indicates that the mask contains failures from the internal Built-In Self-Test (BIST); when 0, the mask indicates other failures.

Ignore reserved bits 0 and 1. Also ignore bits 5 and 6 when bit 7 is clear (=0).

The mask is shown in Table 2 and Table 3.

Table 2. Fail Codes For BIST (bit 7 = 1)

Bit	When Set
6	On-chip Data-RAM failure detected by BIST.
5	Internal Microcode ROM failure detected by BIST.
4	Instruction cache failure detected by BIST.
3	Data cache failure detected by BIST.
2	Local-register cache or processor core failure detected by BIST.
1	Reserved. Always zero.
0	Reserved. Always zero.

Table 3. Remaining Fail Codes (bit 7 = 0)

Bit	When Set
6	Reserved. Always one.
5	Reserved. Always one.
4	A data structure within the IMI is not aligned to a word boundary.
3	A System Error during normal operation has occurred.
2	The Bus Confidence test has failed.
1	Reserved. Always zero.
0	Reserved. Always zero.

2.3 Instruction Set Summary

Table 4 summarizes the 80960Hx instruction set by logical groupings.

Table 4. 80960Hx Instruction Set

Data Movement	Arithmetic	Logical	Bit / Bit Field / Byte
Load Store Move Load Address Conditional Select[2]	Add Subtract Multiply Divide Remainder Modulo Shift Extended Shift Extended Multiply Extended Divide Add with Carry Subtract with Carry Rotate Conditional Add[2] Conditional Subtract[2]	And Not And And Not Or Exclusive Or Not Or Or Not Nor Exclusive Nor Not Nand	Set Bit Clear Bit Not Bit Alter Bit Scan For Bit Span Over Bit Extract Modify Scan Byte for Equal Byte Swap[2]
Comparison	**Branch**	**Call/Return**	**Fault**
Compare Conditional Compare Compare and Increment Compare and Decrement Compare Byte[2] Compare Short[2] Test Condition Code Check Bit	Unconditional Branch Conditional Branch Compare and Branch	Call Call Extended Call System Return Branch and Link	Conditional Fault Synchronize Faults
Debug	**Processor Mgmt**	**Atomic**	**Cache Control**
Modify Trace Controls Mark Force Mark	Flush Local Registers Modify Arithmetic Controls Modify Process Controls Interrupt Enable/ Disable[1, 2] System Control[1]	Atomic Add Atomic Modify	Instruction Cache Control[1, 2] Data Cache Control[1, 2]

NOTES:
1. 80960Hx extensions to the 80960 core instruction set.
2. 80960Hx extensions to the 80960Cx instruction set.

3.0 Package Information

This section describes the pins, pinouts and thermal characteristics for the 80960Hx in the 168-pin ceramic Pin Grid Array (PGA) package, 208-pin PowerQuad2* (PQ4). For complete package specifications and information, see the Intel *Packaging* Handbook (Order# 240800).

The 80960HA/HD/HT is offered with eight speeds and two package types (Table 5). Both the 168-pin ceramic Pin Grid Array (PGA) and the 208-pin PowerQuad2* (PQ4) devices are specified for operation at V_{CC} = 3.3 V ± 0.15 V over a case temperature range of 0 °C to 85 °C.

Table 5. 80960HA/HD/HT Package Types and Speeds

Package/Name	Device	Core Speed (MHz)	Bus Speed (MHz)	Order #
168L PGA	80960HA	25		x80960HA25 S L2GX
		33		x80960HA33 S L2GY
		40		x80960HA40 S L2GZ
	80960HD	50	25	x80960HD50 S L2GH
		66	33	x80960HD66 S L2GJ
		80	40	x80960HD80 S L2GK
	80960HT	75	25	x80960HT75 S L2GP
208L PQFP (also known as PQ4)	80960HA	25		x80960HA25 S L2GU
		33		x80960HA33 S L2GV
		40		x80960HA40 S L2GW
	80960HD	50	25	x80960HD50 S L2GM
		66	33	x80960HD66 S L2GN
		80	40	x80960HD80 S L2LZ
	80960HT	75	25	x80960HT75 S L2GT

NOTE: To address the fact that many of the package prefix variables have changed, all package prefix variables in this document are now indicated with an "x".

3.1 Pin Descriptions

This section defines the 80960Hx pins. Table 6 presents the legend for interpreting the pin descriptions in Table 7. All pins float while the processor is in the ONCE mode, except TDO, which may be driven active according to normal JTAG specifications.

Table 6. Pin Description Nomenclature

Symbol	Description
I	Input only pin.
O	Output only pin.
I/O	Pin may be input or output.
-	Pin must be connected as indicated for proper device functionality.
S(E)	Synchronous edge sensitive input. This input must meet the setup and hold times relative to CLKIN to ensure proper operation of the processor.
S(L)	Synchronous level sensitive input. This input must meet the setup and hold times relative to CLKIN to ensure proper operation of the processor.
A(E)	Asynchronous edge-sensitive input.
A(L)	Asynchronous level-sensitive input.
H(...)	While the processor bus is in the HOLD state (HOLDA asserted), the pin: H(1) is driven to V_{CC} H(0) is driven to V_{SS} H(Z) floats H(Q) continues to be a valid output
B(...)	While the processor is in the bus backoff state ($\overline{BOFF}$ asserted), the pin: B(1) is driven to V_{CC} B(0) is driven to V_{SS} B(Z) floats B(Q) continues to be a valid output
R(...)	While the processor's $\overline{RESET}$ pin is asserted, the pin: R(1) is driven to V_{CC} R(0) is driven to V_{SS} R(Z) floats R(Q) continues to be a valid output

Table 7. 80960Hx Processor Family Pin Descriptions (Sheet 1 of 4)

Name	Type	Description
A31:2	**O** H(Z) B(Z) R(Z)	**ADDRESS BUS** carries the upper 30 bits of the physical address. A31 is the most significant address bit and A2 is the least significant. During a bus access, A31:2 identify all external addresses to word (4-byte) boundaries. The byte enable signals indicate the selected byte in each word. During burst accesses, A3 and A2 increment to indicate successive addresses.
D31:0	**I/O** S(L) H(Z) B(Z) R(Z)	**DATA BUS** carries 32, 16, or 8-bit data quantities depending on bus width configuration. The least significant bit of the data is carried on D0 and the most significant on D31. The lower eight data lines (D7:0) are used when the bus is configured for 8-bit data. When configured for 16-bit data, D15:0 are used.
DP3:0	**I/O** S(L) H(Z) B(Z) R(Z)	**DATA PARITY** carries parity information for the data bus. Each parity bit is assigned a group of eight data bus pins as follows: DP3 generates/checks parity for D31:24 DP2 generates/checks parity for D23:16 DP1 generates/checks parity for D15:8 DP0 generates/checks parity for D7:0 Parity information is generated for a processor write cycle and is checked for a processor read cycle. Parity checking and polarity are programmable. Parity generation/checking is only performed for the size of the data accessed.
$\overline{\text{PCHK}}$	**O** H(Q) B(Q) R(1)	**PARITY CHECK** indicates the result of a parity check operation. An asserted PCHK indicates that the previous bus read access resulted in a parity check error.
$\overline{\text{BE3:0}}$	**O** H(Z) B(Z) R(1)	**BYTE ENABLES** select which of the four bytes addressed by A31:2 are active during a bus access. Byte enable encoding is dependent on the bus width of the memory region accessed: *32-bit bus:* BE3 enables D31:24 BE2 enables D23:16 BE1 enables D15:8 BE0 enables D7:0 *16-bit bus:* BE3 becomes Byte High Enable (enables D15:8) BE2 is not used (state is undefined) BE1 becomes Address Bit 1 (A1) BE0 becomes Byte Low Enable (enables D7:0) *8-bit bus:* BE3 is not used (state is undefined) BE2 is not used (state is undefined) BE1 Address Bit 1 (A1) BE0 Address Bit 0 (A0)
W/$\overline{\text{R}}$	**O** H(Z) B(Z) R(0)	**WRITE/READ** is low for read accesses and high for write accesses. W/$\overline{\text{R}}$ becomes valid during the address phase of a bus cycle and remains valid until the end of the cycle for non-pipelined accesses. For pipelined accesses, W/$\overline{\text{R}}$ changes state when the next address is presented. 0= Read 1= Write
D/$\overline{\text{C}}$	**O** H(Z) B(Z) R(0)	**DATA/CODE** indicates that a bus access is a data access or an instruction access. D/$\overline{\text{C}}$ has the same timing as W/$\overline{\text{R}}$. 0 = Code 1 = Data

Table 7. 80960Hx Processor Family Pin Descriptions (Sheet 2 of 4)

Name	Type	Description
$\overline{\text{SUP}}$	O H(Z) B(Z) R(1)	**SUPERVISOR ACCESS** indicates whether the current bus access originates from a request issued while in supervisor mode or user mode. $\overline{\text{SUP}}$ may be used by the memory subsystem to isolate supervisor code and data structures from non-supervisor access. 0 = Supervisor Mode 1 = User Mode
$\overline{\text{ADS}}$	O H(Z) B(Z) R(1)	**ADDRESS STROBE** indicates a valid address and the start of a new bus access. $\overline{\text{ADS}}$ is asserted for the first clock of a bus access.
$\overline{\text{READY}}$	I S(L)	**$\overline{\text{READY}}$**, when enabled for a memory region, is asserted by the memory subsystem to indicate the completion of a data transfer. $\overline{\text{READY}}$ is used to indicate that read data on the bus is valid, or that a write transfer has completed. $\overline{\text{READY}}$ works in conjunction with the internal wait state generator to accommodate various memory speeds. $\overline{\text{READY}}$ is sampled after any programmed wait states: During each data cycle of a burst access During the data cycle of a non-burst access
$\overline{\text{BTERM}}$	I S(L)	**BURST TERMINATE**, when enabled for a memory region, is asserted by the memory subsystem to terminate a burst access in progress. When $\overline{\text{BTERM}}$ is asserted, the current burst access is terminated and another address cycle occurs.
$\overline{\text{WAIT}}$	O H(Z) B(Z) R(1)	**$\overline{\text{WAIT}}$** indicates the status of the internal wait-state generator. $\overline{\text{WAIT}}$ is asserted when the internal wait state generator generates N_{WAD}, N_{RAD}, N_{WDD} and N_{RDD} wait states. $\overline{\text{WAIT}}$ may be used to derive a write data strobe.
$\overline{\text{BLAST}}$	O H(Z) B(Z) R(1)	**BURST LAST** indicates the last transfer in a bus access. $\overline{\text{BLAST}}$ is asserted in the last data transfer of burst and non-burst accesses after the internal wait-state generator reaches zero. $\overline{\text{BLAST}}$ remains active as long as wait states are inserted through the $\overline{\text{READY}}$ pin. $\overline{\text{BLAST}}$ becomes inactive after the final data transfer in a bus cycle.
DT/$\overline{\text{R}}$	O H(Z) B(Z) R(0)	**DATA TRANSMIT/RECEIVE** indicates direction for data transceivers. DT/$\overline{\text{R}}$ is used with $\overline{\text{DEN}}$ to provide control for data transceivers connected to the data bus. DT/$\overline{\text{R}}$ is driven low to indicate the processor expects data (a read cycle). DT/$\overline{\text{R}}$ is driven high when the processor is “transmitting” data (a store cycle). DT/$\overline{\text{R}}$ only changes state when $\overline{\text{DEN}}$ is high. 0 = Data Receive 1 = Data Transmit
$\overline{\text{DEN}}$	O H(Z) B(Z) R(1)	**DATA ENABLE** indicates data transfer cycles during a bus access. $\overline{\text{DEN}}$ is asserted at the start of the first data cycle in a bus access and de-asserted at the end of the last data cycle. $\overline{\text{DEN}}$ remains asserted for an entire bus request, even when that request spans several bus accesses. For example, a ldq instruction starting at an unaligned quad word boundary is one bus request spanning at least two bus accesses. $\overline{\text{DEN}}$ remains asserted throughout all the accesses (including $\overline{\text{ADS}}$ states) and de-asserts when the lqd instruction request is satisfied. $\overline{\text{DEN}}$ is used with DT/$\overline{\text{R}}$ to provide control for data transceivers connected to the data bus. $\overline{\text{DEN}}$ remains asserted for sequential reads from pipelined memory regions.
$\overline{\text{LOCK}}$	O H(Z) B(Z) R(1)	**BUS LOCK** indicates that an atomic read-modify-write operation is in progress. $\overline{\text{LOCK}}$ may be used by the memory subsystem to prevent external agents from accessing memory that is currently involved in an atomic operation (e.g., a semaphore). $\overline{\text{LOCK}}$ is asserted in the first clock of an atomic operation and de-asserted when $\overline{\text{BLAST}}$ is deasserted in the last bus cycle.

Table 7. 80960Hx Processor Family Pin Descriptions (Sheet 3 of 4)

Name	Type	Description
HOLD	**I** S(L)	**HOLD REQUEST** signals that an external agent requests access to the processor's address, data, and control buses. When HOLD is asserted, the processor: Completes the current bus request. Asserts HOLDA and floats the address, data, and control buses. When HOLD is deasserted, the HOLDA pin is deasserted and the processor reassumes control of the address, data, and control pins.
HOLDA	**O** H(1) B(0) R(Q)	**HOLD ACKNOWLEDGE** indicates to an external master that the processor has relinquished control of the bus. The processor grants HOLD requests and enters the HOLDA state while the RESET pin is asserted. HOLDA is never granted while LOCK is asserted.
BOFF	**I** S(L)	**BUS BACKOFF** forces the processor to immediately relinquish control of the bus on the next clock cycle. When READY/BTERM is enabled and: When BOFF is asserted, the address, data, and control buses are floated on the next clock cycle and the current access is aborted. When BOFF is deasserted, the processor resumes by regenerating the aborted bus access. See Figure 16 on page 48 for BOFF timing requirements.
BREQ	**O** H(Q) B(Q) R(0)	**BUS REQUEST** indicates that a bus request is pending in the bus controller. BREQ does not indicate whether or not the processor is stalled. See BSTALL for processor stall status. BREQ may be used with BSTALL to indicate to an external bus arbiter the processor's bus ownership requirements.
BSTALL	**O** H(Q) B(Q) R(0)	**BUS STALL** indicates that the processor has stalled pending the result of a request in the bus controller. When BSTALL is asserted, the processor must regain bus ownership to continue processing (i.e., it may no longer execute strictly out of on-chip cache memory).
CT3:0	**O** H(Z) B(Z) R(Z)	**CYCLE TYPE** indicates the type of bus cycle currently being started or processor state. CT3:0 encoding follows: **Cycle Type** — **ADSCT3:0** Program-initiated access using 8-bit bus — 00000 Program-initiated access using 16-bit bus — 00001 Program-initiated access using 32-bit bus — 00010 Event-initiated access using 8-bit bus — 00100 Event-initiated access using 16-bit bus — 00101 Event-initiated access using 32-bit bus — 00110 Reserved — 00X11 Reserved for future products — 01XXX Reserved — 1XXXX
XINT7:0	**I** A(E) A(L)	**EXTERNAL INTERRUPT** pins are used to request interrupt service. These pins may be configured in three modes: *Dedicated Mode:* Each pin is assigned a dedicated interrupt level. Dedicated inputs may be programmed to be level (low or high) or edge (rising or falling) sensitive. *Expanded Mode:* All eight pins act as a vectored interrupt source. The interrupt pins are level sensitive in this mode. *Mixed Mode:* The XINT7:5 pins act as dedicated sources and the XINT4:0 pins act as the five most significant bits of a vectored source. The least significant bits of the vectored source are set to "010" internally.
NMI	**I** A(E)	**NON-MASKABLE INTERRUPT** causes a non-maskable interrupt event to occur. NMI is the highest priority interrupt source. NMI is falling edge triggered.

Table 7. 80960Hx Processor Family Pin Descriptions (Sheet 4 of 4)

Name	Type	Description
CLKIN	**I**	**CLOCK INPUT** provides the time base for the 80960Hx. All internal circuitry is synchronized to CLKIN. All input and output timings are specified relative to CLKIN. For the 80960HD, the 2x internal clock is derived by multiplying the CLKIN frequency by two. For the 80960HT, the 3x internal clock is derived by multiplying the CLKIN frequency by three.
$\overline{\text{RESET}}$	**I** A(L)	**$\overline{\text{RESET}}$** forces the device into reset. $\overline{\text{RESET}}$ causes all external and internal signals to return to their reset state (when defined). The rising edge of $\overline{\text{RESET}}$ starts the processor boot sequence.
STEST	**I** S(L)	**SELF TEST**, when asserted during the rising edge of $\overline{\text{RESET}}$, causes the processor to execute its built in self-test.
$\overline{\text{FAIL}}$	**O** H(Q) B(Q) R(0)	**$\overline{\text{FAIL}}$** indicates a failure of the processor's built-in self-test performed during initialization. $\overline{\text{FAIL}}$ is asserted immediately out of reset and toggles during self-test to indicate the status of individual tests. When self-test passes, $\overline{\text{FAIL}}$ is de-asserted and the processor branches to the user's initialization code. When self-test fails, the $\overline{\text{FAIL}}$ pin asserts and the processor ceases execution.
$\overline{\text{ONCE}}$	**I**	**ON-CIRCUIT EMULATION** control: the processor samples this pin during reset. When 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 except the TDO pin. $\overline{\text{ONCE}}$ uses an internal pull-up resistor; see R_{PU} definition in Table 22, "80960Hx DC Characteristics" on page 40. Pull this pin high when not in use.
TCK	**I**	**TEST CLOCK** provides the clocking function for IEEE 1149.1 Boundary Scan testing.
TDI	**I**	**TEST DATA INPUT** is the serial input pin for IEEE 1149.1 Boundary Scan testing. TDI uses an internal pull-up resistor; see R_{PU} definition in Table 22, "80960Hx DC Characteristics" on page 40.
TDO	**O**	**TEST DATA OUTPUT** is the serial output pin for IEEE 1149.1 Boundary Scan testing. ONCE does not disable this pin.
$\overline{\text{TRST}}$	**I**	**TEST RESET** asynchronously resets the Test Access Port (TAP) controller. $\overline{\text{TRST}}$ must be held low at least 10,000 clock cycles after power-up. One method is to provide $\overline{\text{TRST}}$ with a separate power-on-reset circuit. $\overline{\text{TRST}}$ includes an internal pull-up resistor; see R_{PU} definition in Table 22, "80960Hx DC Characteristics" on page 40. Pull this pin low when not in use.
TMS	**I**	**TEST MODE SELECT** is sampled at the rising edge of TCK. TCK controls the sequence of TAP controller state changes for IEEE 1149.1 Boundary Scan testing. TMS uses an internal pull-up resistor; see R_{PU} definition in Table 22, "80960Hx DC Characteristics" on page 40.
VCC5	**I**	**5 V REFERENCE VOLTAGE** input is the reference voltage for the 5 V-tolerant I/O buffers. Connect this signal to +5 V for use with inputs which exceed 3.3 V. When all inputs are from 3.3 V components, connect this signal to 3.3 V.
VCCPLL	**I**	**PLL VOLTAGE** is the +3.3 VDC analog input for the PLL.
VOLDET	**O**	**VOLTAGE DETECT** signal allows external system logic to distinguish between a 5 V 80960Cx processor and the 3.3 V 80960Hx processor. This signal is active low for a 3.3 V 80960Hx (it is high impedance for 5 V 80960Cx). This pin is available only on the PGA version. 0 = 80960Hx 1 = 80960Cx

3.2 80960Hx Mechanical Data

3.2.1 80960Hx PGA Pinout

Figure 2 depicts the complete 80960Hx PGA pinout as viewed from the top side of the component (i.e., pins facing down). Figure 3 shows the complete 80960Hx PGA pinout as viewed from the pin-side of the package (i.e., pins facing up). Table 9 lists the 80960Hx pin names with package location. See Section 4.3, "Recommended Connections" on page 38 for specifications and recommended connections.

Figure 2. 80960Hx 168-Pin PGA Pinout—View from Top (Pins Facing Down)

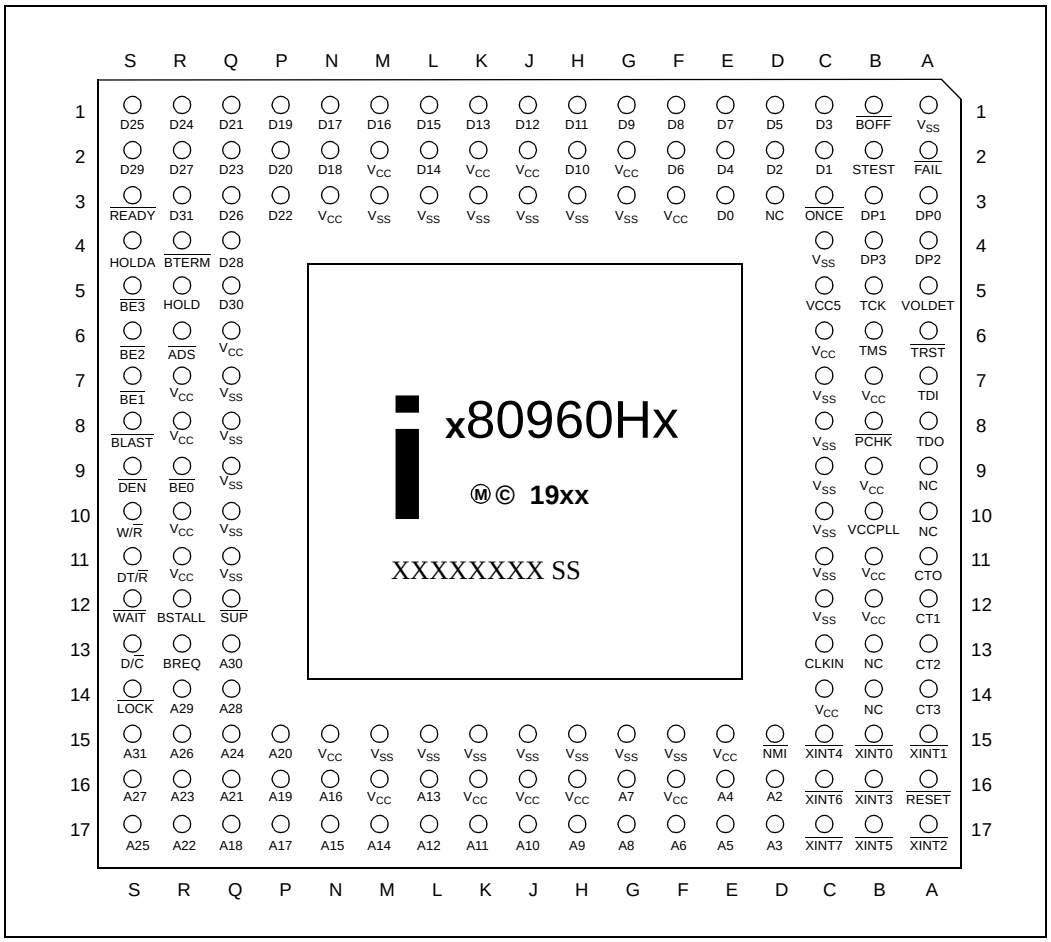


Figure 3. 80960Hx 168-Pin PGA Pinout—View from Bottom (Pins Facing Up)

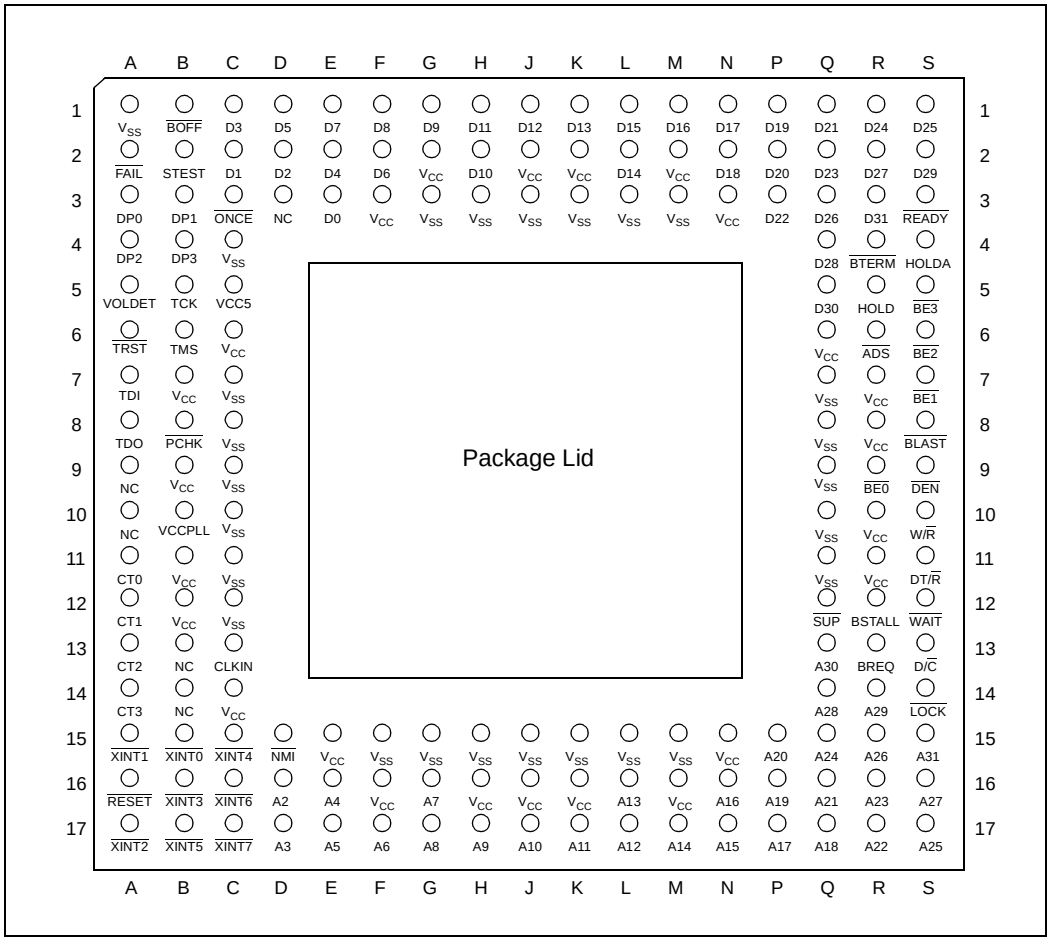


Table 8. 80960Hx 168-Pin PGA Pinout—Signal Name Order (Sheet 1 of 2)

Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin
A2	D16	**ADS**	R6	**D14**	L2	**LOCK**	S14
A3	D17	**BE0**	R9	**D15**	L1	**NC**	A9
A4	E16	**BE1**	S7	**D16**	M1	**NC**	A10
A5	E17	**BE2**	S6	**D17**	N1	**NC**	B13
A6	F17	**BE3**	S5	**D18**	N2	**NC**	B14
A7	G16	**BLAST**	S8	**D19**	P1	**NC**	D3
A8	G17	**BOFF**	B1	**D20**	P2	**NMI**	D15
A9	H17	**BREQ**	R13	**D21**	Q1	**ONCE**	C3
A10	J17	**BSTALL**	R12	**D22**	P3	**PCHK**	B8
A11	K17	**BTERM**	R4	**D23**	Q2	**READY**	S3
A12	L17	**CLKIN**	C13	**D24**	R1	**RESET**	A16
A13	L16	**CT0**	A11	**D25**	S1	**STEST**	B2
A14	M17	**CT1**	A12	**D26**	Q3	**SUP**	Q12
A15	N17	**CT2**	A13	**D27**	R2	**TCK**	B5
A16	N16	**CT3**	A14	**D28**	Q4	**TDI**	A7
A17	P17	**D/C**	S13	**D29**	S2	**TDO**	A8
A18	Q17	**D0**	E3	**D30**	Q5	**TMS**	B6
A19	P16	**D1**	C2	**D31**	R3	**TRST**	A6
A20	P15	**D2**	D2	**DEN**	S9	**V_{CC}**	B7
A21	Q16	**D3**	C1	**DP0**	A3	**V_{CC}**	B9
A22	R17	**D4**	E2	**DP1**	B3	**V_{CC}**	B11
A23	R16	**D5**	D1	**DP2**	A4	**V_{CC}**	B12
A24	Q15	**D6**	F2	**DP3**	B4	**V_{CC}**	C6
A25	S17	**D7**	E1	**DT/R**	S11	**V_{CC}**	C14
A26	R15	**D8**	F1	**FAIL**	A2	**V_{CC}**	E15
A27	S16	**D9**	G1	**—**	**—**	**V_{CC}**	F3
A28	Q14	**D10**	H2	**—**	**—**	**V_{CC}**	F16
A29	R14	**D11**	H1	**—**	**—**	**V_{CC}**	G2
A30	Q13	**D12**	J1	**HOLD**	R5	**V_{CC}**	H16
A31	S15	**D13**	K1	**HOLDA**	S4	**V_{CC}**	J2

Table 8. 80960Hx 168-Pin PGA Pinout—Signal Name Order (Sheet 2 of 2)

Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin
V_{CC}	J16	VCCPLL	B10	V_{SS}	H3	V_{SS}	Q10
V_{CC}	K2	VOLDET	A5	V_{SS}	H15	V_{SS}	Q11
V_{CC}	K16	V_{SS}	A1	V_{SS}	J3	W/R	S10
V_{CC}	M2	V_{SS}	C4	V_{SS}	J15	WAIT	S12
V_{CC}	M16	V_{SS}	C7	V_{SS}	K3	XINT0	B15
V_{CC}	N3	V_{SS}	C8	V_{SS}	K15	XINT1	A15
V_{CC}	N15	V_{SS}	C9	V_{SS}	L3	XINT2	A17
V_{CC}	Q6	V_{SS}	C10	V_{SS}	L15	XINT3	B16
V_{CC}	R7	V_{SS}	C11	V_{SS}	M3	XINT4	C15
V_{CC}	R8	V_{SS}	C12	V_{SS}	M15	XINT5	B17
V_{CC}	R10	V_{SS}	F15	V_{SS}	Q7	XINT6	C16
V_{CC}	R11	V_{SS}	G3	V_{SS}	Q8	XINT7	C17
VCC5	C5	V_{SS}	G15	V_{SS}	Q9	—	—

Table 9. 80960Hx 168-Pin PGA Pinout—Pin Number Order (Sheet 1 of 2)

PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name
A1	V_{SS}	**B14**	NC	**E15**	V_{CC}	**K15**	V_{SS}
A2	$\overline{FAIL}$	**B15**	$\overline{XINT0}$	**E16**	A4	**K16**	V_{CC}
A3	DP0	**B16**	$\overline{XINT3}$	**E17**	A5	**K17**	A11
A4	DP2	**B17**	$\overline{XINT5}$	**F1**	D8	**L1**	D15
A5	VOLDET	**C1**	D3	**F2**	D6	**L2**	D14
A6	$\overline{TRST}$	**C2**	D1	**F3**	V_{CC}	**L3**	V_{SS}
A7	TDI	**C3**	$\overline{ONCE}$	**F15**	V_{SS}	**L15**	V_{SS}
A8	TDO	**C4**	V_{SS}	**F16**	V_{CC}	**L16**	A13
A9	NC	**C5**	VCC5	**F17**	A6	**L17**	A12
A10	NC	**C6**	V_{CC}	**G1**	D9	**M1**	D16
A11	CT0	**C7**	V_{SS}	**G2**	V_{CC}	**M2**	V_{CC}
A12	CT1	**C8**	V_{SS}	**G3**	V_{SS}	**M3**	V_{SS}
A13	CT2	**C9**	V_{SS}	**G15**	V_{SS}	**M15**	V_{SS}
A14	CT3	**C10**	V_{SS}	**G16**	A7	**M16**	V_{CC}
A15	$\overline{XINT1}$	**C11**	V_{SS}	**G17**	A8	**M17**	A14
A16	$\overline{RESET}$	**C12**	V_{SS}	**H1**	D11	**N1**	D17
A17	$\overline{XINT2}$	**C13**	CLKIN	**H2**	D10	**N2**	D18
B1	$\overline{BOFF}$	**C14**	V_{CC}	**H3**	V_{SS}	**N3**	V_{CC}
B2	STEST	**C15**	$\overline{XINT4}$	**H15**	V_{SS}	**N15**	V_{CC}
B3	DP1	**C16**	$\overline{XINT6}$	**H16**	V_{CC}	**N16**	A16
B4	DP3	**C17**	$\overline{XINT7}$	**H17**	A9	**N17**	A15
B5	TCK	**D1**	D5	**J1**	D12	**P1**	D19
B6	TMS	**D2**	D2	**J2**	V_{CC}	**P2**	D20
B7	V_{CC}	**D3**	NC	**J3**	V_{SS}	**P3**	D22
B8	$\overline{PCHK}$	**D15**	$\overline{NMI}$	**J15**	V_{SS}	**P15**	A20
B9	V_{CC}	**D16**	A2	**J16**	V_{CC}	**P16**	A19
B10	VCCPLL	**D17**	A3	**J17**	A10	**P17**	A17
B11	V_{CC}	**E1**	D7	**K1**	D13	**Q1**	D21
B12	V_{CC}	**E2**	D4	**K2**	V_{CC}	**Q2**	D23
B13	NC	**E3**	D0	**K3**	V_{SS}	**Q3**	D26

Table 9. 80960Hx 168-Pin PGA Pinout—Pin Number Order (Sheet 2 of 2)

PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name	PGA Pin	Signal Name
Q4	D28	**Q16**	A21	**R11**	V_{CC}	**S6**	$\overline{BE2}$
Q5	D30	**Q17**	A18	**R12**	BSTALL	**S7**	$\overline{BE1}$
Q6	V_{CC}	**R1**	D24	**R13**	BREQ	**S8**	$\overline{BLAST}$
Q7	V_{SS}	**R2**	D27	**R14**	A29	**S9**	$\overline{DEN}$
Q8	V_{SS}	**R3**	D31	**R15**	A26	**S10**	W/$\overline{R}$
Q9	V_{SS}	**R4**	$\overline{BTERM}$	**R16**	A23	**S11**	DT/$\overline{R}$
Q10	V_{SS}	**R5**	HOLD	**R17**	A22	**S12**	$\overline{WAIT}$
Q11	V_{SS}	**R6**	$\overline{ADS}$	**S1**	D25	**S13**	D/$\overline{C}$
Q12	$\overline{SUP}$	**R7**	V_{CC}	**S2**	D29	**S14**	$\overline{LOCK}$
Q13	A30	**R8**	V_{CC}	**S3**	$\overline{READY}$	**S15**	A31
Q14	A28	**R9**	$\overline{BE0}$	**S4**	HOLDA	**S16**	A27
Q15	A24	**R10**	V_{CC}	**S5**	$\overline{BE3}$	**S17**	A25

3.2.2 80960Hx PQ4 Pinout

Figure 4. 80960Hx 208-Pin PQ4 Pinout

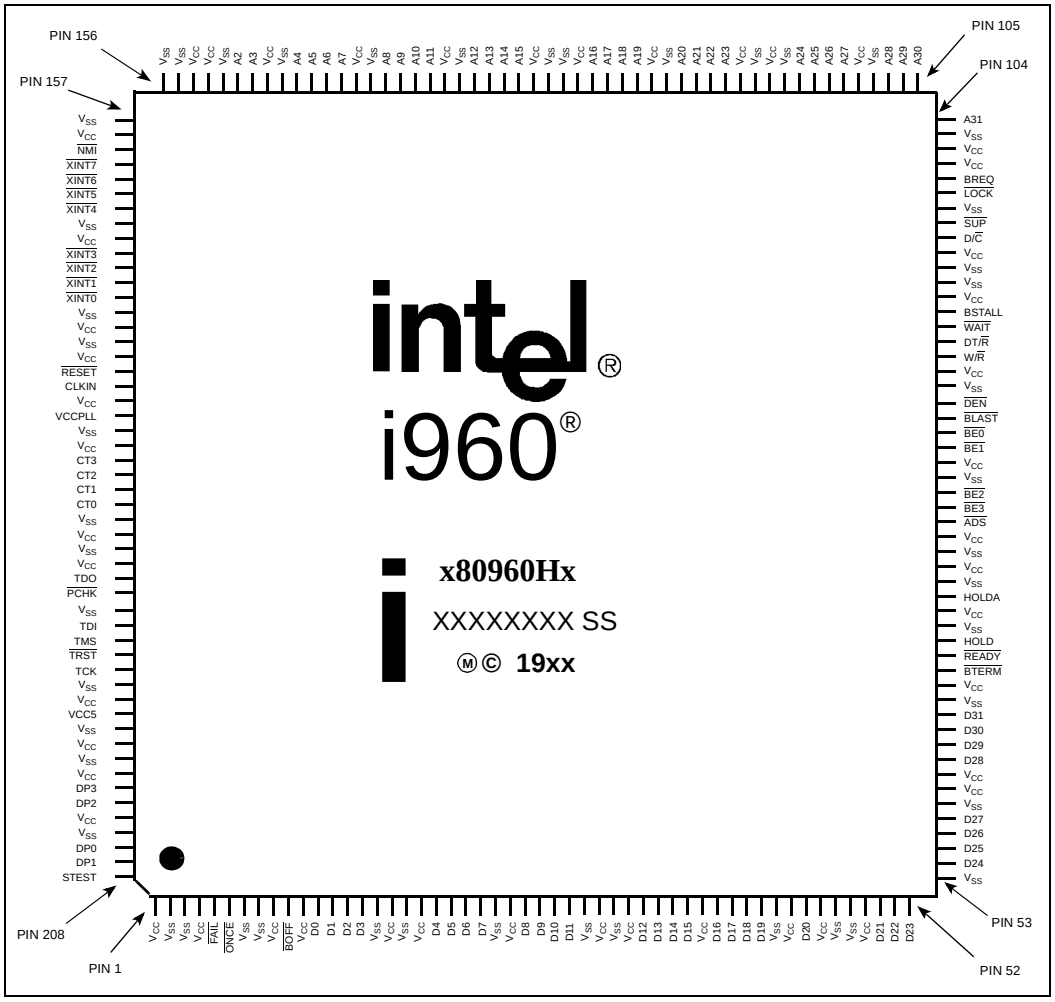


Table 10. 80960Hx PQ4 Pinout—Signal Name Order (Sheet 1 of 2)

Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin
A2	151	$\overline{BE0}$	83	D16	39	$\overline{PCHK}$	189
A3	150	$\overline{BE1}$	82	D17	40	$\overline{READY}$	68
A4	147	$\overline{BE2}$	79	D18	41	$\overline{RESET}$	174
A5	146	$\overline{BE3}$	78	D19	42	STEST	208
A6	145	$\overline{BLAST}$	84	D20	45	$\overline{SUP}$	97
A7	144	$\overline{BOFF}$	10	D21	50	TCK	194
A8	141	BREQ	100	D22	51	TDI	191
A9	140	BSTALL	91	D23	52	TDO	188
A10	139	$\overline{BTERM}$	67	D24	54	TMS	192
A11	138	CLKIN	175	D25	55	$\overline{TRST}$	193
A12	135	CT0	183	D26	56	V_{CC}	1
A13	134	CT1	182	D27	57	V_{CC}	4
A14	133	CT2	181	D28	61	V_{CC}	9
A15	132	CT3	180	D29	62	V_{CC}	11
A16	127	$\overline{D/C}$	96	D30	63	V_{CC}	17
A17	126	D0	12	D31	64	V_{CC}	19
A18	125	D1	13	$\overline{DEN}$	85	V_{CC}	25
A19	124	D2	14	DP0	206	V_{CC}	31
A20	121	D3	15	DP1	207	V_{CC}	33
A21	120	D4	20	DP2	203	V_{CC}	38
A22	119	D5	21	DP3	202	V_{CC}	44
A23	118	D6	22	$\overline{DT/R}$	89	V_{CC}	46
A24	113	D7	23	$\overline{FAIL}$	5	V_{CC}	49
A25	112	D8	26	—	—	V_{CC}	59
A26	111	D9	27	—	—	V_{CC}	60
A27	110	D10	28	—	—	V_{CC}	66
A28	107	D11	29	HOLD	69	V_{CC}	71
A29	106	D12	34	HOLDA	72	V_{CC}	74
A30	105	D13	35	$\overline{LOCK}$	99	V_{CC}	76
A31	104	D14	36	$\overline{NMI}$	159	V_{CC}	81
$\overline{ADS}$	77	D15	37	$\overline{ONCE}$	6	V_{CC}	87

Table 10. 80960Hx PQ4 Pinout—Signal Name Order (Sheet 2 of 2)

Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin
V_{CC}	92	V_{CC}	187	V_{SS}	70	V_{SS}	164
V_{CC}	95	V_{CC}	196	V_{SS}	73	V_{SS}	170
V_{CC}	101	V_{CC}	199	V_{SS}	75	V_{SS}	172
V_{CC}	102	V_{CC}	201	V_{SS}	80	V_{SS}	178
V_{CC}	109	V_{CC}	204	V_{SS}	86	V_{SS}	184
V_{CC}	115	**VCC5**	197	V_{SS}	93	V_{SS}	186
V_{CC}	117	**VCCPLL**	177	V_{SS}	94	V_{SS}	190
V_{CC}	123	V_{SS}	2	V_{SS}	98	V_{SS}	195
V_{CC}	128	V_{SS}	3	V_{SS}	103	V_{SS}	198
V_{CC}	131	V_{SS}	**7**	V_{SS}	108	V_{SS}	200
V_{CC}	137	V_{SS}	8	V_{SS}	114	V_{SS}	205
V_{CC}	143	V_{SS}	16	V_{SS}	116	**W/$\overline{R}$**	88
V_{CC}	149	V_{SS}	18	V_{SS}	122	**$\overline{WAIT}$**	90
V_{CC}	153	V_{SS}	24	V_{SS}	129	**$\overline{XINT0}$**	169
V_{CC}	154	V_{SS}	30	V_{SS}	130	**$\overline{XINT1}$**	168
V_{CC}	158	V_{SS}	32	V_{SS}	136	**$\overline{XINT2}$**	167
V_{CC}	165	V_{SS}	43	V_{SS}	142	**$\overline{XINT3}$**	166
V_{CC}	171	V_{SS}	47	V_{SS}	148	**$\overline{XINT4}$**	163
V_{CC}	173	V_{SS}	48	V_{SS}	152	**$\overline{XINT5}$**	162
V_{CC}	176	V_{SS}	53	V_{SS}	155	**$\overline{XINT6}$**	161
V_{CC}	179	V_{SS}	58	V_{SS}	156	**$\overline{XINT7}$**	160
V_{CC}	185	V_{SS}	65	V_{SS}	157	—	—

Table 11. 80960Hx PQ4 Pinout—Pin Number Order (Sheet 1 of 2)

PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name
1	V_{CC}	**31**	V_{CC}	**61**	D28	**91**	BSTALL
2	V_{SS}	**32**	V_{SS}	**62**	D29	**92**	V_{CC}
3	V_{SS}	**33**	V_{CC}	**63**	D30	**93**	V_{SS}
4	V_{CC}	**34**	D12	**64**	D31	**94**	V_{SS}
5	$\overline{FAIL}$	**35**	D13	**65**	V_{SS}	**95**	V_{CC}
6	$\overline{ONCE}$	**36**	D14	**66**	V_{CC}	**96**	D/$\overline{C}$
7	V_{SS}	**37**	D15	**67**	$\overline{BTERM}$	**97**	$\overline{SUP}$
8	V_{SS}	**38**	V_{CC}	**68**	$\overline{READY}$	**98**	V_{SS}
9	V_{CC}	**39**	D16	**69**	HOLD	**99**	$\overline{LOCK}$
10	$\overline{BOFF}$	**40**	D17	**70**	V_{SS}	**100**	BREQ
11	V_{CC}	**41**	D18	**71**	V_{CC}	**101**	V_{CC}
12	D0	**42**	D19	**72**	HOLDA	**102**	V_{CC}
13	D1	**43**	V_{SS}	**73**	V_{SS}	**103**	V_{SS}
14	D2	**44**	V_{CC}	**74**	V_{CC}	**104**	A31
15	D3	**45**	D20	**75**	V_{SS}	**105**	A30
16	V_{SS}	**46**	V_{CC}	**76**	V_{CC}	**106**	A29
17	V_{CC}	**47**	V_{SS}	**77**	$\overline{ADS}$	**107**	A28
18	V_{SS}	**48**	V_{SS}	**78**	$\overline{BE3}$	**108**	V_{SS}
19	V_{CC}	**49**	V_{CC}	**79**	$\overline{BE2}$	**109**	V_{CC}
20	D4	**50**	D21	**80**	V_{SS}	**110**	A27
21	D5	**51**	D22	**81**	V_{CC}	**111**	A26
22	D6	**52**	D23	**82**	$\overline{BE1}$	**112**	A25
23	D7	**53**	V_{SS}	**83**	$\overline{BE0}$	**113**	A24
24	V_{SS}	**54**	D24	**84**	$\overline{BLAST}$	**114**	V_{SS}
25	V_{CC}	**55**	D25	**85**	$\overline{DEN}$	**115**	V_{CC}
26	D8	**56**	D26	**86**	V_{SS}	**116**	V_{SS}
27	D9	**57**	D27	**87**	V_{CC}	**117**	V_{CC}
28	D10	**58**	V_{SS}	**88**	W/$\overline{R}$	**118**	A23
29	D11	**59**	V_{CC}	**89**	DT/$\overline{R}$	**119**	A22
30	V_{SS}	**60**	V_{CC}	**90**	$\overline{WAIT}$	**120**	A21

Table 11. 80960Hx PQ4 Pinout—Pin Number Order (Sheet 2 of 2)

PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name	PQ4 Pin	Signal Name
121	A20	**143**	V_{CC}	**165**	V_{CC}	**187**	V_{CC}
122	V_{SS}	**144**	A7	**166**	$\overline{XINT3}$	**188**	TDO
123	V_{CC}	**145**	A6	**167**	$\overline{XINT2}$	**189**	$\overline{PCHK}$
124	A19	**146**	A5	**168**	$\overline{XINT1}$	**190**	V_{SS}
125	A18	**147**	A4	**169**	$\overline{XINT0}$	**191**	TDI
126	A17	**148**	V_{SS}	**170**	V_{SS}	**192**	TMS
127	A16	**149**	V_{CC}	**171**	V_{CC}	**193**	$\overline{TRST}$
128	V_{CC}	**150**	A3	**172**	V_{SS}	**194**	TCK
129	V_{SS}	**151**	A2	**173**	V_{CC}	**195**	V_{SS}
130	V_{SS}	**152**	V_{SS}	**174**	$\overline{RESET}$	**196**	V_{CC}
131	V_{CC}	**153**	V_{CC}	**175**	CLKIN	**197**	VCC5
132	A15	**154**	V_{CC}	**176**	V_{CC}	**198**	V_{SS}
133	A14	**155**	V_{SS}	**177**	VCCPLL	**199**	V_{CC}
134	A13	**156**	V_{SS}	**178**	V_{SS}	**200**	V_{SS}
135	A12	**157**	V_{SS}	**179**	V_{CC}	**201**	V_{CC}
136	V_{SS}	**158**	V_{CC}	**180**	CT3	**202**	DP3
137	V_{CC}	**159**	$\overline{NMI}$	**181**	CT2	**203**	DP2
138	A11	**160**	$\overline{XINT7}$	**182**	CT1	**204**	V_{CC}
139	A10	**161**	$\overline{XINT6}$	**183**	CT0	**205**	V_{SS}
140	A9	**162**	$\overline{XINT5}$	**184**	V_{SS}	**206**	DP0
141	A8	**163**	$\overline{XINT4}$	**185**	V_{CC}	**207**	DP1
142	V_{SS}	**164**	V_{SS}	**186**	V_{SS}	**208**	STEST

3.3 Package Thermal Specifications

The 80960Hx is specified for operation when T_C (case temperature) is within the range of 0 °C to 85 °C. T_C may be measured in any environment to determine whether the 80960Hx is within the specified operating range. Measure the case temperature at the center of the top surface, opposite the pins. Refer to Figure 5.

T_A (ambient temperature) is calculated from θ_{CA} (thermal resistance from case to ambient) using Equation 1:

Equation 1. Calculation of Ambient Temperature (T_A)

$$T_A = T_C - (P \cdot \theta_{CA})$$

Table 12 shows the maximum T_A allowable (without exceeding T_C) at various airflows and operating frequencies (f_{CLKIN}).

Note that T_A is greatly improved by attaching fins or a heatsink to the package. P (maximum power consumption) is calculated by using the typical I_{CC} as tabulated in Section 4.6, “DC Specifications” on page 40 and V_{CC} of 3.3 V.

Figure 5. Measuring 80960Hx PGA Case Temperature

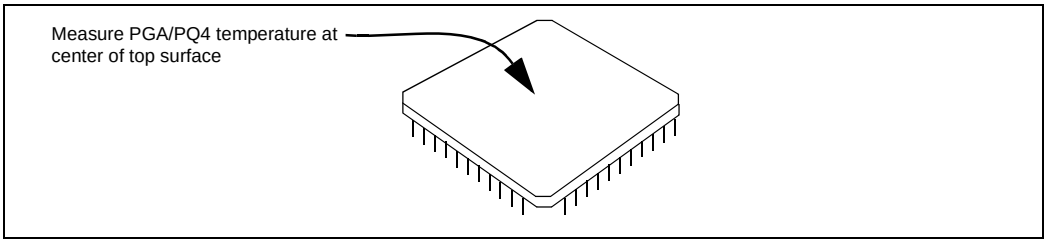


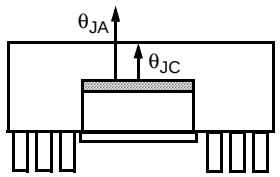
Table 12. Maximum T_A at Various Airflows in °C (PGA Package Only)

			Airflow-ft/min (m/sec)					
		f_{CLKIN} (MHz)	0 (0)	200 (1.01)	400 (2.03)	600 (3.04)	800 (4.06)	1000 (5.07)
Core 1X Bus Clock	T_A with Heatsink†	25 33 40	69 63 59	74 70 67	78 75 73	79 77 75	80 79 77	80 79 77
	T_A without Heatsink	25 33 40	64 56 50	67 62 56	71 67 63	74 70 67	75 72 69	76 74 71
Core 2X Bus Clock	T_A with Heatsink†	16 25 33 40	68 58 49 41	73 66 60 55	77 73 69 65	79 75 71 68	80 77 74 72	80 77 74 72
	T_A without Heatsink	16 25 33 40	62 49 38 27	66 56 46 38	71 62 55 48	73 66 60 55	75 68 63 58	76 71 66 62
Core 3X Bus Clock	T_A with Heatsink†	20 25	53 45	63 58	71 67	73 70	76 73	76 73
	T_A without Heatsink	20 25	43 33	51 42	58 51	63 58	66 61	68 64

† *0.285” high unidirectional heatsink (Al alloy 6061, 50 mil fin width, 150 mil center-to-center fin spacing).

Table 13. 80960Hx 168-Pin PGA Package Thermal Characteristics

Thermal Resistance — °C/Watt						
Parameter	Airflow — ft./min (m/sec)					
	0 (0)	200 (1.01)	400 (2.03)	600 (3.07)	800 (4.06)	1000 (5.07)
θ Junction-to-Case (Case measured as shown in Figure 5.)	1.5	1.5	1.5	1.5	1.5	1.5
θ Case-to-Ambient (No Heatsink)	17	14	11	9	8	7
θ Case-to-Ambient (With Heatsink)[3]	13	9	6	5	4	4



NOTES:
1. This table applies to 80960Hx PGA plugged into socket or soldered directly to board.
2. $\theta_{JA} = \theta_{JC} + \theta_{CA}$
3. 0.285” high unidirectional heatsink (Al alloy 6061, 50 mil fin width, 150 mil center-to-center fin spacing).

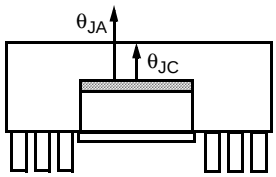
Table 14. Maximum T_A at Various Airflows in °C (PQ4 Package Only)

			Airflow-ft/min (m/sec)					
		f_{CLKIN} (MHz)	0 (0)	200 (1.01)	400 (2.03)	600 (3.04)	800 (4.06)	1000 (5.07)
Core 1X Bus Clock	T_A with Heatsink†	25 33 40	71 67 63	76 74 71	79 77 75	79 77 75	80 79 77	80 79 77
	T_A without Heatsink	25 33 40	70 65 61	73 68 65	75 72 69	75 72 69	76 74 71	76 74 71
Core 2X Bus Clock	T_A with Heatsink†	16 25 33 40	71 62 55 48	76 71 66 62	79 75 71 68	79 75 71 68	80 77 74 72	80 77 74 72
	T_A without Heatsink	16 25 33 40	69 60 52 42	72 64 57 51	75 68 63 58	75 68 63 58	76 71 66 62	76 71 66 62
Core 3X Bus Clock	T_A with Heatsink†	20 25	58 51	68 64	73 70	73 70	76 73	76 73
	T_A without Heatsink	20 25	56 48	61 55	66 61	66 61	68 64	68 64

† 0.285” high unidirectional heatsink (Al alloy 6061, 50 mil fin width, 150 mil center-to-center fin spacing).

Table 15. 80960Hx 208-Pin PQ4 Package Thermal Characteristics

Thermal Resistance — °C/Watt						
Parameter	Airflow — ft./min (m/sec)					
	0 (0)	200 (1.01)	400 (2.03)	600 (3.07)	800 (4.06)	1000 (5.07)
θ Junction-to-Case (Case measured as shown in Figure 5.)	1	1	1	1	1	1
θ Case-to-Ambient (No Heatsink)	12	10	8	8	7	7
θ Case-to-Ambient (With Heatsink)[3]	11	7	5	5	4	4



NOTES:
1. This table applies to 80960Hx PQ4 plugged into socket or soldered directly to board.
2. $\theta_{JA} = \theta_{JC} + \theta_{CA}$
3. 0.285” high unidirectional heatsink (Al alloy 6061, 50 mil fin width, 150 mil center-to-center fin spacing).

3.4 Heat Sink Adhesives

Intel recommends silicone-based adhesives to attach heat sinks to the PGA package. There is no particular recommendation concerning the PQ4 package.

3.5 PowerQuad4 Plastic Package

The 80960Hx family is available in an improved version of the common 208-lead SQFP plastic package called the PowerQuad4* (PQ4). The PQ4 package dimensions and lead pitch are identical to the SQFP package and the former PQ2 package, so the PQ4 fits into the same board footprint. The advantage of the PQ4 package is the superior thermal conductivity that allows the plastic version of the 80960Hx to operate with the same 0 °C to 85 °C temperature specifications as the more expensive ceramic PGA package.

The PQ4 package integrates a copper heat sink within the package to dissipate heat effectively. See Table 14 and Table 15 for more information.

3.6 Stepping Register Information

The memory-mapped register at FF008710H contains the 80960Hx Device ID. The ID is identical to the ID obtained from a JTAG Query. Figure 6 defines the current 80960Hx Device IDs. The value for device identification is compliant with the IEEE 1149.1 specification and Intel standards. Table 16 describes the fields of the device ID.

Figure 6. 80960Hx Device Identification Register

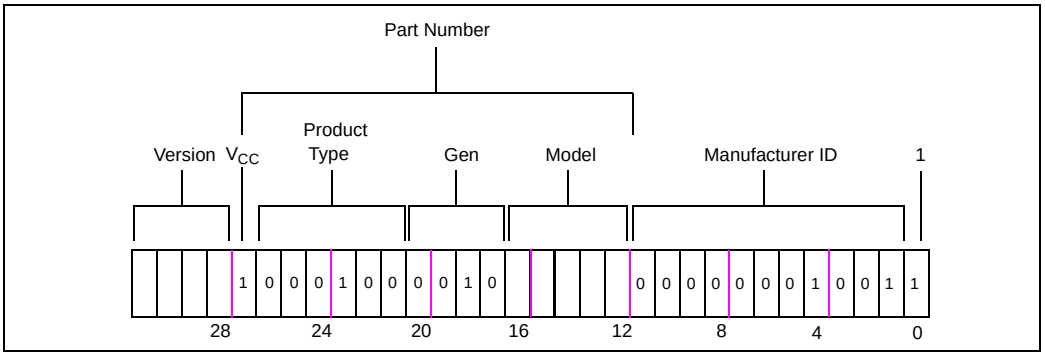


Table 16. Fields of 80960Hx Device ID

Field	Value	Definition
Version	See Table 18.	Indicates major stepping changes.
V_{CC}	1 = 3.3 V device	Indicates that a device is 3.3 V.
Product Type	00 0100 (Indicates i960 CPU)	Designates type of product.
Generation Type	0010 = H-series	Indicates the generation (or series) the product belongs to.
Model	See Table 17.	Indicates member within a series and specific model information.
Manufacturer ID	000 0000 1001 (Indicates Intel)	Manufacturer ID assigned by IEEE.

Table 17. 80960Hx Device ID Model Types

Device	Version	V_{CC}	Product	Gen.	Model	Manufacturer ID	'1'
80960HA	See Table 18.	1	000100	0010	00000	00000001001	1
80960HD		1	000100	0010	00001	00000001001	1
80960HT		1	000100	0010	00010	00000001001	1

Table 18. Device ID Version Numbers for Different Steppings

Stepping	Version
A0	0000
A1	0001
A2	0001
B0, B2	0010

NOTE: This data sheet applies to the B2 stepping.

3.7 Sources for Accessories

The following is a list of suggested sources for 80960Hx accessories. This is neither an endorsement nor a warranty of the performance of any of the listed products and/or companies.

Sockets

- 3M Textool Test and Interconnection Products
  6801 River Place Blvd. MS 130-3N-29
  Austin, TX 78726-9000
  (800) 328-0411 FAX: (800) 932-9373
- Concept Mfg, Inc. (Decoupling Sockets)
  400 Walnut St. Suite 609
  Redwood City, CA 94063
  (415) 365-1162 FAX: (415) 365-1164

Heatsinks/Fins

- Thermalloy, Inc.
  2021 West Valley View Lane
  Dallas, TX 75234-8993
  (972) 243-4321 FAX: (972) 241-4656
- Wakefield Engineering, Inc.
  60 Audubon Road
  Wakefield, MA 01880
  (617) 245-5900 FAX: (617) 246-0874
- Aavid Thermal Technologies, Inc.
  One Kool Path
  Laconia, NH 03247-0400
  (603) 523-3400

4.0 Electrical Specifications

4.1 Absolute Maximum Ratings

Table 19. Absolute Maximum Ratings

Parameter	Maximum Rating
Storage Temperature	–65 ºC to +150 ºC
Case Temperature Under Bias	–65 ºC to +110 ºC
Supply Voltage with respect to V_{SS}	–0.5 V to + 4.6 V
Voltage on VCC5 with respect to V_{SS}	–0.5 V to + 6.5 V
Voltage on Other Pins with respect to V_{SS}	–0.5 V to VCC5 + 0.5 V

Warning: Stressing the device beyond the “Absolute Maximum Ratings” may cause permanent damage. These are stress ratings only. Operation beyond the “Operating Conditions” is not recommended and extended exposure beyond the “Operating Conditions” may affect device reliability.

4.2 Operating Conditions

Table 20. Operating Conditions

Symbol	Parameter	Min	Max	Units
V_{CC}	Supply Voltage	3.15	3.45	V
VCC5	Input Protection Bias	3.15	5.5	V
f_{CLKIN} 1xcore	Input Clock Frequency - 1x Core (80960HA)	16	40	MHz
f_{CLKIN} 2xcore	Input Clock Frequency - 2x Core (80960HD)	16	40	MHz
f_{CLKIN} 3xcore	Input Clock Frequency - 3x Core (80960HT)	16	25	MHz
T_C	Case Temp Under Bias (PGA and PQ4 Packages)	0	85	^oC

4.3 Recommended Connections

Power and ground connections must be made to multiple V_{CC} and V_{SS} (GND) pins. Every 80960Hx-based circuit board should include power (V_{CC}) and ground (V_{SS}) planes for power distribution. Every V_{CC} pin must be connected to the power plane; every V_{SS} pin must be connected to the ground plane. Pins identified as “NC” —no connect pins—**must not** be connected in the system.

Liberal decoupling capacitance should be placed near the 80960Hx. The processor may cause transient power surges when its output buffers transition, particularly when connected to large capacitive loads.

Low inductance capacitors and interconnects are recommended for best high-frequency electrical performance. Inductance may be reduced by shortening the board traces between the processor and decoupling capacitors as much as possible. Capacitors specifically designed for PGA packages offer the lowest possible inductance.

For reliable operation, always connect unused inputs to an appropriate signal level. In particular, any unused interrupt ($\overline{\text{XINT7:0}}$, $\overline{\text{NMI}}$) input should be connected to V_{CC} through a pull-up resistor, as should $\overline{\text{BTERM}}$ when not used. Pull-up resistors should be in the range of 20 KΩ for each pin tied high. When $\overline{\text{READY}}$ or HOLD are not used, the unused input should be connected to ground. **N.C. pins must always remain unconnected.**

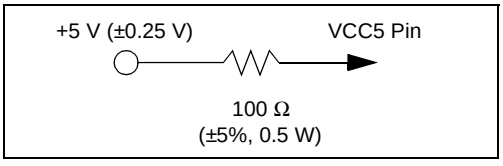
4.4 VCC5 Pin Requirements (V_{DIFF})

In mixed-voltage systems that drive 80960Hx processor inputs in excess of 3.3 V, the VCC5 pin must be connected to the system’s 5 V supply. To limit current flow into the VCC5 pin, there is a limit to the voltage differential between the VCC5 pin and the other V_{CC} pins. The voltage differential between the 80960Hx VCC5 pin and its 3.3 V V_{CC} pins should never exceed 2.25 V. This limit applies to power-up, power-down, and steady-state operation. Table 21 outlines this requirement.

Meeting this requirement ensures proper operation and ensures that the current draw into the VCC5 pin does not exceed the I_{CC5} specification.

When the voltage difference requirements cannot be met due to system design limitations, an alternate solution may be employed. As shown in Figure 7, a minimum of 100 Ω series resistor may be used to limit the current into the VCC5 pin. This resistor ensures that current drawn by the VCC5 pin does not exceed the maximum rating for this pin.

Figure 7. VCC5 Current-Limiting Resistor



This resistor is not necessary in systems that may ensure the V_{DIFF} specification.

In 3.3 V-only systems and systems that drive 80960Hx pins from 3.3 V logic, connect the VCC5 pin directly to the 3.3 V V_{CC} plane.

Table 21. V_{DIFF} Specification for Dual Power Supply Requirements (3.3 V, 5 V)

Sym	Parameter	Min	Max	Units	Notes
V_{DIFF}	VCC5-V_{CC} Difference		2.25	V	VCC5 input should not exceed V_{CC} by more than 2.25 V during power-up and power-down, or during steady-state operation.

4.5 VCCPLL Pin Requirements

When the voltage on the VCCPLL power supply pin exceeds the V_{CC} pin voltage by 0.5 V at any time, including the power up and power down sequences, excessive currents may permanently damage on-chip electrostatic discharge (ESD) protection diodes. The damage may accumulate over multiple episodes.

Pragmatically, this problem only occurs when the VCCPLL and V_{CC} pins are driven by separate power supplies or voltage regulators. Applications that use one power supply for VCCPLL and V_{CC} are not typically at risk. Verify that your application does not allow the VCCPLL voltage to exceed V_{CC} by 0.5 V.

The VCCPL low-pass filter recommended in the Developer's Manual does not promote this problem.

4.6 D.C. Specifications

Table 22. 80960Hx D.C. Characteristics (Sheet 1 of 2)

Per the conditions described in Section 4.3, "Recommended Connections" on page 38.

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{IL}	Input Low Voltage	– 0.3		+0.8	V	
V_{IH}	Input High Voltage	2.0		VCC5 + 0.3	V	
V_{OL}	Output Low Voltage All outputs except $\overline{FAIL}$			0.4 0.2	V	I_{OL} = 3 mA I_{OL} = 100 µA
V_{OL}	Output Low Voltage $\overline{FAIL}$ pin			0.4	V	I_{OL} = 5 mA
V_{OH}	Output High Voltage	2.4 V_{CC} – 0.2			V V	I_{OH} = –3 mA I_{OH} = –100 µA
I_{LI}	Input Leakage Current Non-Test Inputs TDI, TMS, $\overline{TRST}$ and $\overline{ONCE}$	-1		1 -110	µA µA	$0 \le V_{IN} \le V_{CC}$ V_{IN} = 0 V
I_{LO}	Output Leakage Current Non-Test Outputs TDO pin			1 5	µA µA	$0.45 \le V_{OUT} \le V_{CC}$ $0.45 \le V_{OUT} \le V_{CC}$
I_{CC} Active (Power Supply)	80960HA 25 33 40 80960HD 32 50 66 80 80960HT 60 75			579 765 927 631 985 1300 1578 1165 1455	mA	4, 5
I_{CC} Active (Thermal)	80960HA 25 33 40 80960HD 32 50 66 80 80960HT 60 75		392 518 628 413 645 851 1034 752 938		mA	4, 6
I_{CC} Test (Reset Mode)	80960HA 25 33 40 80960HD 32 50 66 80 80960HT 60 75			330 436 528 382 595 785 955 702 878	mA	7, 8
I_{CC} Test (ONCE mode)				25	mA	7

Table 22. 80960Hx D.C. Characteristics (Sheet 2 of 2)

Per the conditions described in Section 4.3, "Recommended Connections" on page 38.

Symbol	Parameter	Min	Typ	Max	Units	Notes
I_{CC5} Current on the VCC5 Pin	80960HA 80960HD 80960HT			200 200 200	µA	9
C_{IN}	Input Capacitance for: PQ4 PGA			 12 12	 pF pF	F_C = 1 MHz[10]
C_{OUT}	Output Capacitance of each output pin			12	pF	F_C = 1 MHz[3, 10]
$C_{I/O}$	I/O Pin Capacitance			12	pF	F_C = 1 MHz[10]
R_{PU}	Internal Pull-Up Resistance for ONCE, TMS, TDI and TRST	30	65	100	kΩ	

NOTES:

1. I_{CC} Maximum is measured at worst case frequency, V_{CC}, and temperature, with device operating and outputs loaded to the test conditions described in Section 4.7.1, "AC Test Conditions" on page 45.
2. I_{CC} Typical is not tested.
3. Output Capacitance is the capacitive load of a floating output.
4. Measured with device operating and outputs loaded to the test conditions in Figure 8, "AC Test Load" on page 45. Input signals rise to V_{CC} and fall to V_{SS}.
5. I_{CC} Active (Power Supply) value is provided for selecting your system's power supply. It is measured using one of the worst case instruction mixes with V_{CC} = 3.45 V. This parameter is characterized but not tested.
6. I_{CC} Active (Thermal) value is provided for your system's thermal management. Typical I_{CC} is measured with V_{CC} = 3.3 V and temperature = 25°C. This parameter is characterized but not tested.
7. I_{CC} Test (Power modes) refers to the I_{CC} values that are tested when the 80960HA/HD/HT is in Reset mode or ONCE mode with V_{CC} = 3.45 V.
8. Worst case is V_{CC} = 3.45 V, 0 °C.
9. I_{CC5} is tested at V_{CC} = 3.0 V, VCC5 = 5.25 V.
10. Pin capacitance is characterized, but not tested.

4.7 A.C. Specifications

Table 23. 80960Hx A.C. Characteristics (Sheet 1 of 2)

Per conditions in Section 4.2, "Operating Conditions" on page 37 and Section 4.7.1, "AC Test Conditions" on page 45.

Symbol	Parameter	Min	Max	Units	Notes
Input Clock[1, 7]					
T_F	CLKIN Frequency 80960HA 80960HD 80960HT	16 16 16	40 40 25	MHz MHz MHz	
T	CLKIN Period 80960HA 80960HD 80960HT	 25 25 40	 62.5 62.5 62.5	 ns ns ns	
T_{CS}	CLKIN Period Stability	-250	+250	ps	11
T_{CH}	CLKIN High Time	8		ns	11
T_{CL}	CLKIN Low Time 80960HA 80960HD 80960HT	8 8 8		ns ns ns	11
T_{CR}	CLKIN Rise Time	0	4	ns	11
T_{CF}	CLKIN Fall Time	0	4	ns	11
Synchronous Outputs[1, 2, 3, 6]					
T_{OV1}, T_{OH1}	Output Valid Delay and Output Hold for all outputs except DT/$\overline{R}$, $\overline{BLAST}$ and BREQ for 3.3 V and 5 V inputs and I/Os.	1.5	9.5	ns	
T_{OV2}, T_{OH2}	Output Valid Delay and Output Hold for DT/$\overline{R}$ 80960HA 80960HD 80960HT	 T/2 + 1.5 3T/4 + 1.5 5T/6 + 1.5	 T/2 + 9.5 3T/4 + 9.5 5T/6 + 9.5	 ns ns ns	
T_{OV3}, T_{OH3}	Output Valid Delay and Output Hold for $\overline{BLAST}$	1.5	9	ns	
T_{OV4}, T_{OH4}	Output Valid Delay and Output Hold for BREQ	0.5	9	ns	
T_{OV5}, T_{OH5}	Output Valid Delay and Output Hold for A3:2	1.5	8.5		
T_{OF}	Output Float for all outputs	1.5	9	ns	11
Synchronous Inputs[1, 7, 8, 9]					
T_{IS1}	Input Setup for all inputs except $\overline{READY}$, $\overline{BTERM}$, HOLD, and $\overline{BOFF}$	2.5		ns	
T_{IH1}	Input Hold for all inputs except $\overline{READY}$, $\overline{BTERM}$, HOLD, and $\overline{BOFF}$	2.5		ns	

NOTE: See Table 24, "AC Characteristics Notes" on page 44 for all notes related to AC specifications.

Table 23. 80960Hx A.C. Characteristics (Sheet 2 of 2)

Per conditions in Section 4.2, "Operating Conditions" on page 37 and Section 4.7.1, "AC Test Conditions" on page 45.

Symbol	Parameter	Min	Max	Units	Notes
T_{IS2}	Input Setup for $\overline{READY}$, $\overline{BTERM}$, HOLD, and $\overline{BOFF}$	6		ns	
T_{IH2}	Input Hold for $\overline{READY}$, $\overline{BTERM}$, HOLD, and $\overline{BOFF}$	2.5		ns	
Relative Output Timings[1, 2, 3, 6, 10]					
T_{AVSH1}	A31:2 Valid to $\overline{ADS}$ Rising	T – 5	T + 5	ns	10
T_{AVSH2}	$\overline{BE3:0}$, W/$\overline{R}$, $\overline{SUP}$, D/$\overline{C}$ Valid to $\overline{ADS}$ Rising	T – 5	T + 5	ns	10
T_{AVEL1}	A31:2 Valid to $\overline{DEN}$ Falling	T – 5	T + 5	ns	10
T_{AVEL2}	$\overline{BE3:0}$, W/$\overline{R}$, $\overline{SUP}$ Valid to $\overline{DEN}$ Falling	T – 5	T + 5	ns	10
T_{NLQV}	$\overline{WAIT}$ Falling to Output Data Valid	-5	5	ns	10
T_{DVNH}	Output Data Valid to $\overline{WAIT}$ Rising	-5 + N*T	5 + N*T	ns	4, 10
T_{NLNH}	$\overline{WAIT}$ Falling to $\overline{WAIT}$ Rising	-4 + N*T	4 + N*T	ns	4, 10
T_{NHQX}	Output Data Hold after $\overline{WAIT}$ Rising	-5 + (N+1)*T	5 + (N+1)*T	ns	5, 10
T_{EHTV}	DT/$\overline{R}$ Hold after $\overline{DEN}$ High	T/2 – 5	Infinite	ns	10
T_{TVEL}	DT/$\overline{R}$ Valid to $\overline{DEN}$ Falling 80960HA 80960HD 80960HT	 T/2 – 4 T/4 – 4 T/6 – 4		 ns ns ns	10
Relative Input Timings[1, 7, 10]					
T_{IS7}	$\overline{XINT7:0}$, $\overline{NMI}$ Input Setup	6		ns	9
T_{IH7}	$\overline{XINT7:0}$, $\overline{NMI}$ Input Hold	2.5		ns	9
T_{IS8}	$\overline{RESET}$ Input Setup	3		ns	8
T_{IH8}	$\overline{RESET}$ Input Hold	T/4 + 1		ns	8

NOTE: See Table 24, "AC Characteristics Notes" on page 44 for all notes related to AC specifications.

Table 24. A.C. Characteristics Notes

NOTES:
1. See Section 4.8, “AC Timing Waveforms” on page 46 for waveforms and definitions.
2. See Figure 25, “Output Delay or Hold vs. Load Capacitance” on page 52 for capacitive derating information for output delays and hold times.
3. See Figure 22, “Rise and Fall Time Derating at 85 °C and Minimum VCC” on page 51 for capacitive derating information for rise and fall times.
4. Where N is the number of N_{RAD}, N_{RDD}, N_{WAD} or N_{WDD} wait states that are programmed in the Bus Controller Region Table. WAIT never goes active when there are no wait states in an access.
5. N = Number of wait states inserted with READY.
6. These specifications are ensured by the processor.
7. These specifications must be met by the system for proper operation of the processor.
8. RESET is an asynchronous input that has no required setup and hold time for proper operation. However, to ensure the device exits the reset mode synchronized to a particular clock edge, the rising edge of RESET must meet setup and hold times to the rising edge of the CLKIN.
9. The interrupt pins are synchronized internally by the 80960Hx. They have no required setup or hold times for proper operation. These pins are sampled by the interrupt controller every clock and must be active for at least two consecutive CLKIN rising edges when asserting them asynchronously. To ensure recognition at a particular clock edge, the setup and hold times shown must be met.
10. Relative Output timings are not tested.
11. Not tested.
12. The processor minimizes changes to the bus signals when transitioning from a bus cycle to an idle bus for the following signals: A31:4, SUP, CT3:0, D/C, LOCK, W/R, BE3:0.

Table 25. 80960Hx Boundary Scan Test Signal Timings

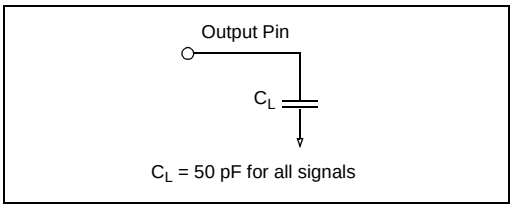
Symbol	Parameter	Min	Max	Units	Notes
T_{BSF}	TCK Frequency	0	8	MHz	
T_{BSC}	TCK Period	125	Infinite	ns	
T_{BSCH}	TCK High Time	40		ns	Measured at 1.5 V†
T_{BSCL}	TCK Low Time	40		ns	Measured at 1.5 V†
T_{BSCR}	TCK Rise Time		8	ns	0.8 V to 2.0 V†
T_{BSCF}	TCK Fall Time		8	ns	2.0 V to 0.8 V†
T_{BSIS1}	Input Setup to TCK — TDI, TMS	8		ns	
T_{BSIH1}	Input Hold from TCK — TDI, TMS	10		ns	
T_{BSOV1}	TDO Valid Delay	3	30	ns	
T_{BSOF1}	TDO Float Delay		36	ns	†
T_{BSOV2}	All Outputs (Non-Test) Valid Delay	3	30	ns	Relative to TCK
T_{BSOF2}	All Outputs (Non-Test) Float Delay		36	ns	Relative to TCK†
T_{BSIS2}	Input Setup to TCK - All Inputs (Non-Test)	8		ns	
T_{BSIH2}	Input Hold from TCK - All Inputs (Non-Test)	10		ns	

† Not tested.

4.7.1 A.C. Test Conditions

A.C. values are derived using the 50 pF load shown in Figure 8. Figure 25, “Output Delay or Hold vs. Load Capacitance” on page 52, shows how timings vary with load capacitance. Input waveforms (except for CLKIN) are assumed to have a rise and fall time of ≤ 2 ns from 0.8 V to 2.0 V.

Figure 8. A.C. Test Load



4.8 A.C. Timing Waveforms

Figure 9. CLKIN Waveform

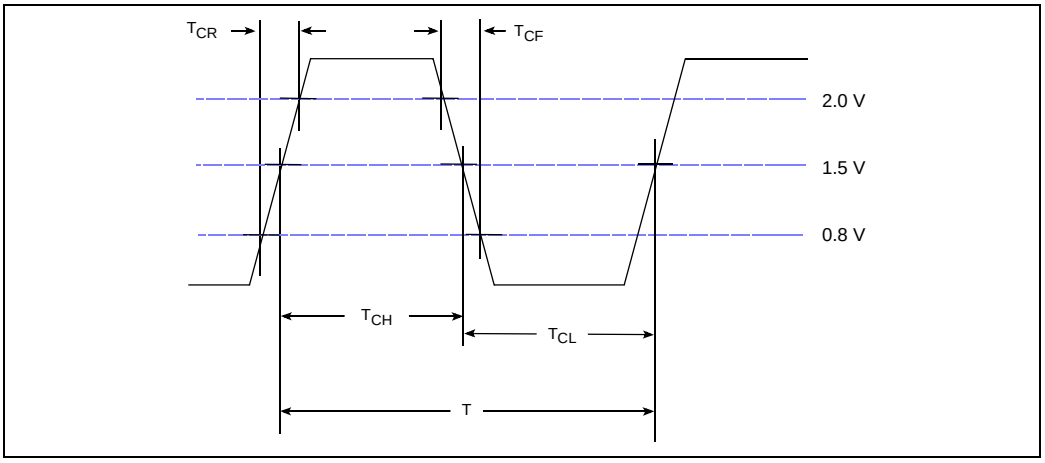


Figure 10. Output Delay Waveform

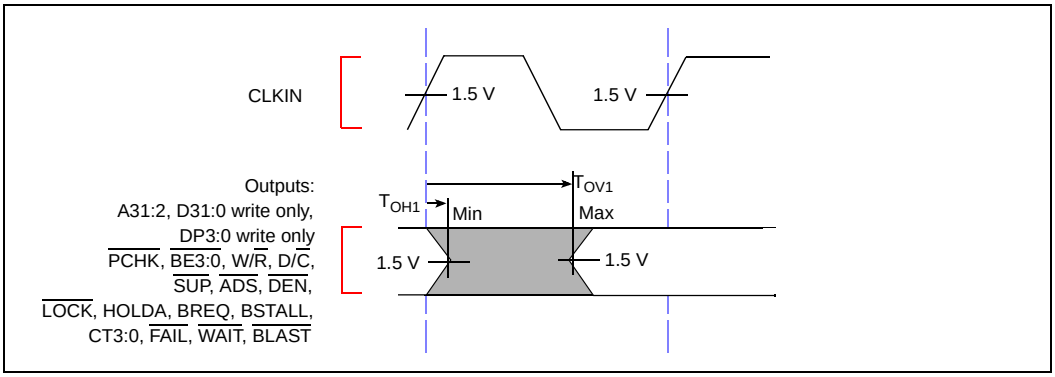


Figure 11. Output Delay Waveform

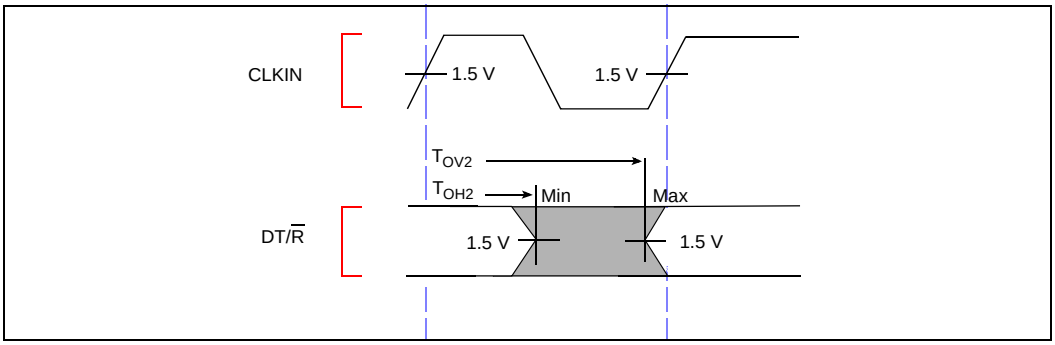


Figure 12. Output Float Waveform

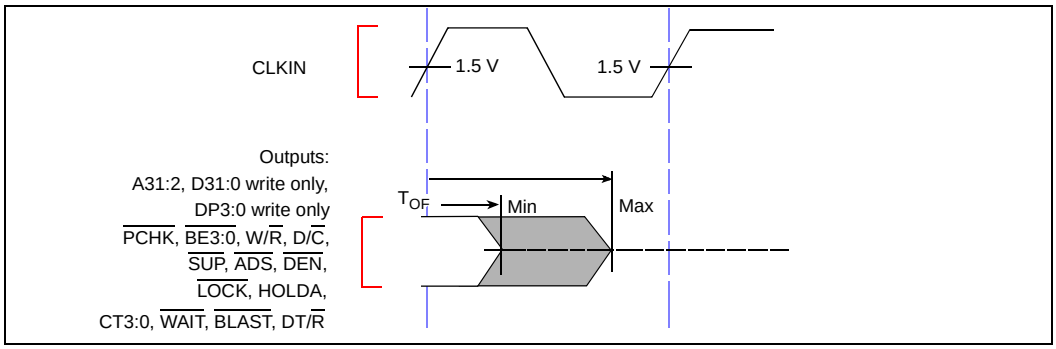


Figure 13. Input Setup and Hold Waveform

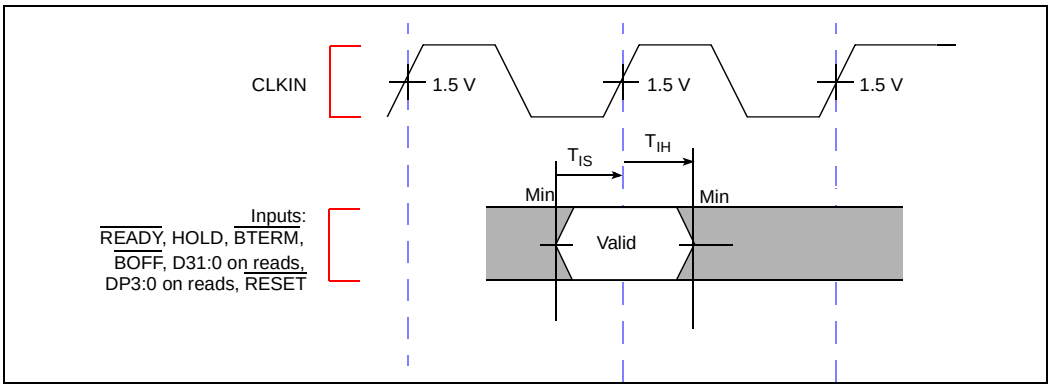
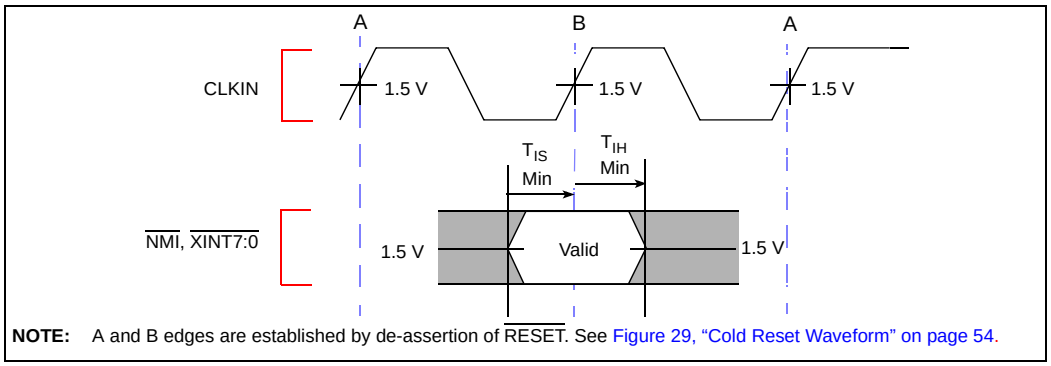
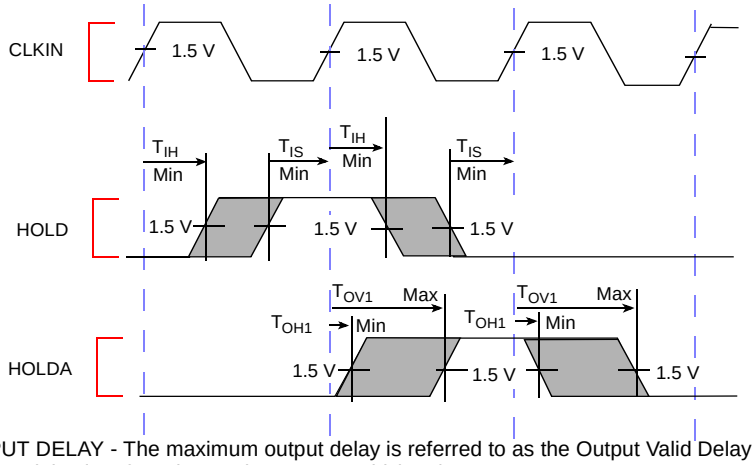


Figure 14. NMI, XINT7:0 Input Setup and Hold Waveform



NOTE: A and B edges are established by de-assertion of RESET. See Figure 29, “Cold Reset Waveform” on page 54.

Figure 15. Hold Acknowledge Timings



T_{OV} T_{OH} — OUTPUT DELAY - The maximum output delay is referred to as the Output Valid Delay (T_{OV}). The minimum output delay is referred to as the Output Hold (T_{OH}).

T_{IS} T_{IH} — INPUT SETUP AND HOLD - The input setup and hold requirements specify the sampling window during which synchronous inputs must be stable for correct processor operation.

Figure 16. Bus Backoff ($\overline{BOFF}$) Timings

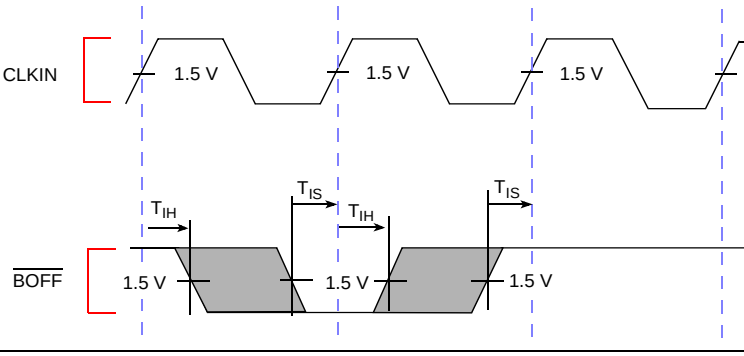


Figure 17. TCK Waveform

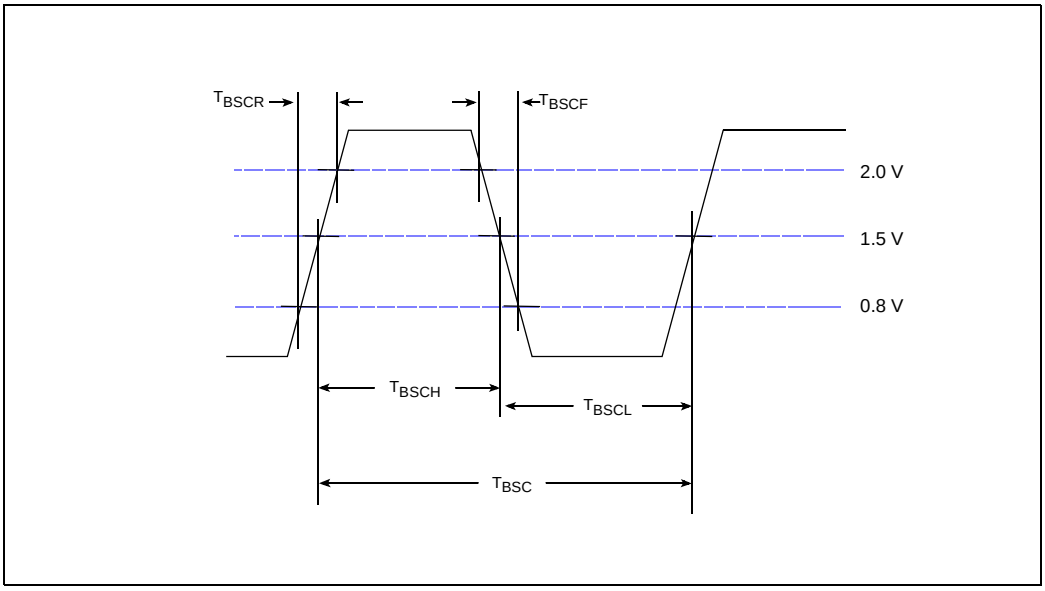


Figure 18. Input Setup and Hold Waveforms for T_{BSIS1} and T_{BSIH1}

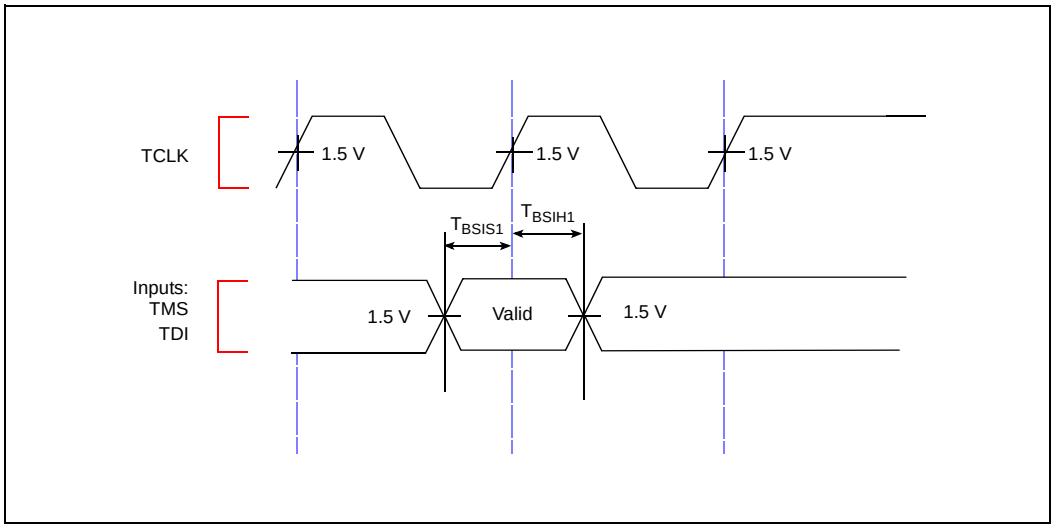


Figure 19. Output Delay and Output Float for T_{BSOV1} and T_{BSOF1}

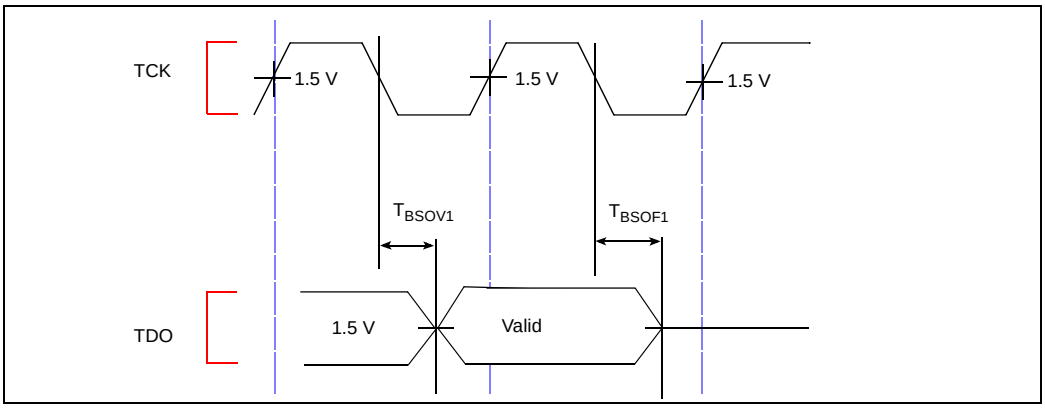


Figure 20. Output Delay and Output Float Waveform for T_{BSOV2} and T_{BSOF2}

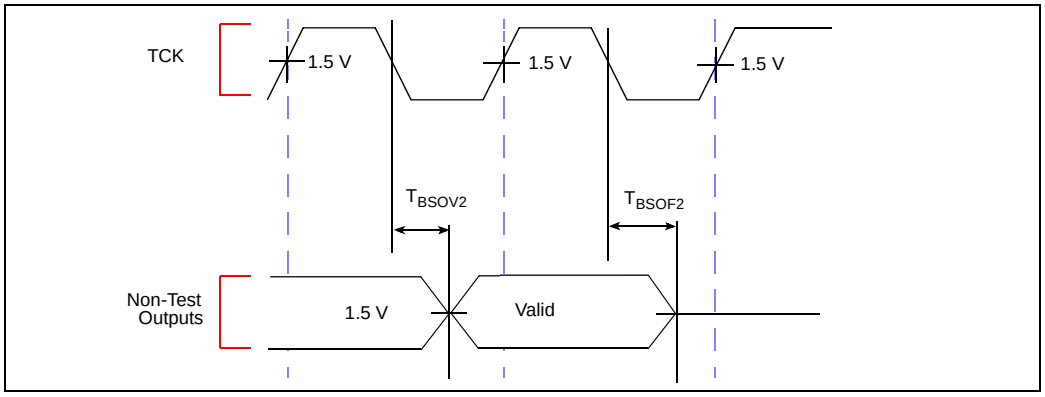


Figure 21. Input Setup and Hold Waveform for T_{BSIS2} and T_{BSIH2}

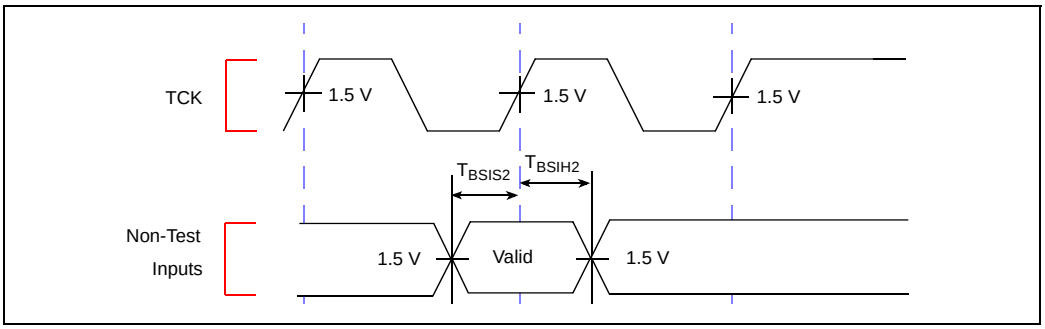


Figure 22. Rise and Fall Time Derating at 85 °C and Minimum V_{CC}

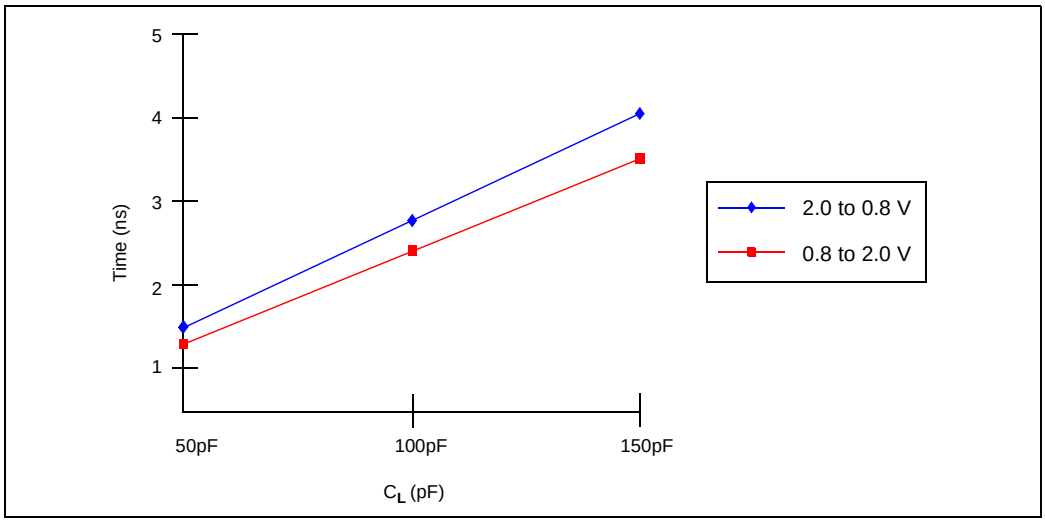


Figure 23. I_{CC} Active (Power Supply) vs. Frequency

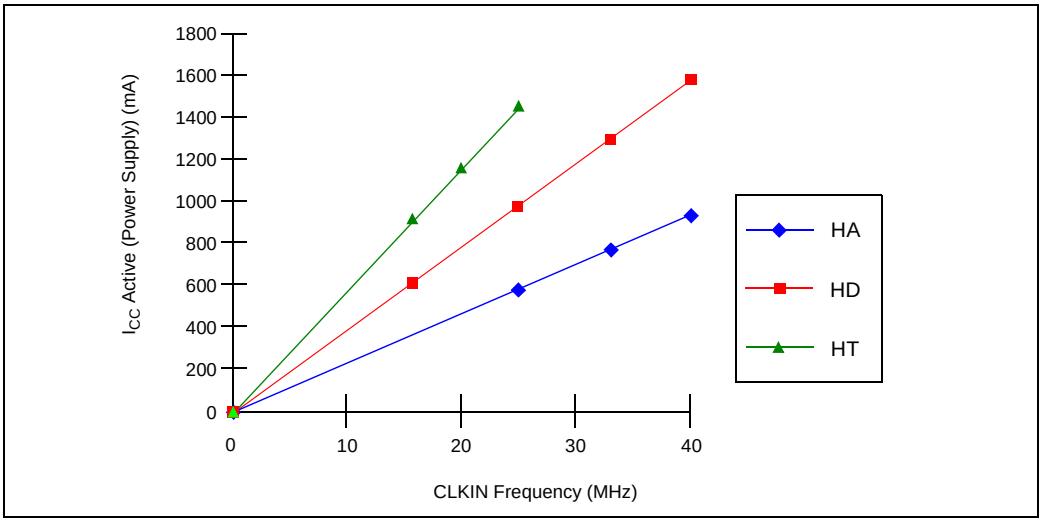


Figure 24. I_{CC} Active (Thermal) vs. Frequency

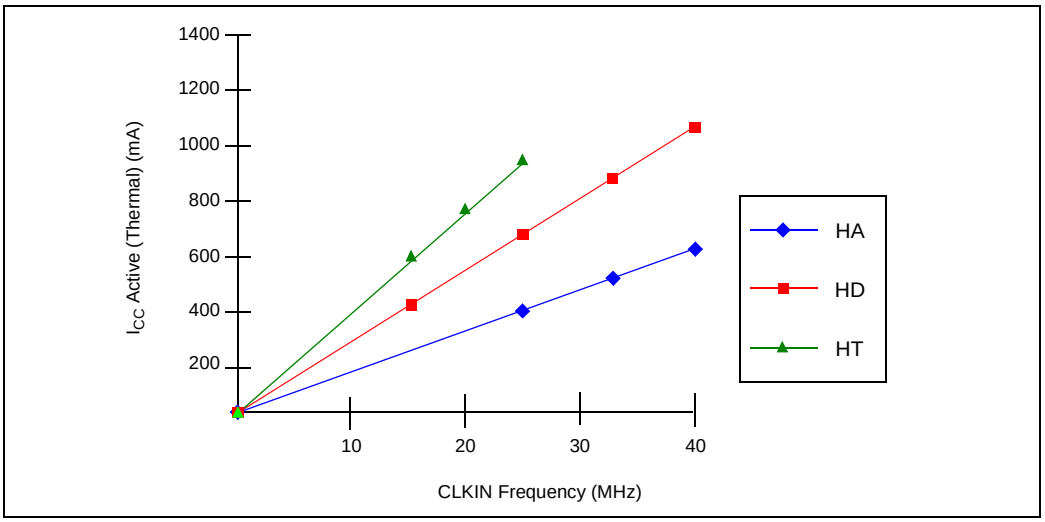


Figure 25. Output Delay or Hold vs. Load Capacitance

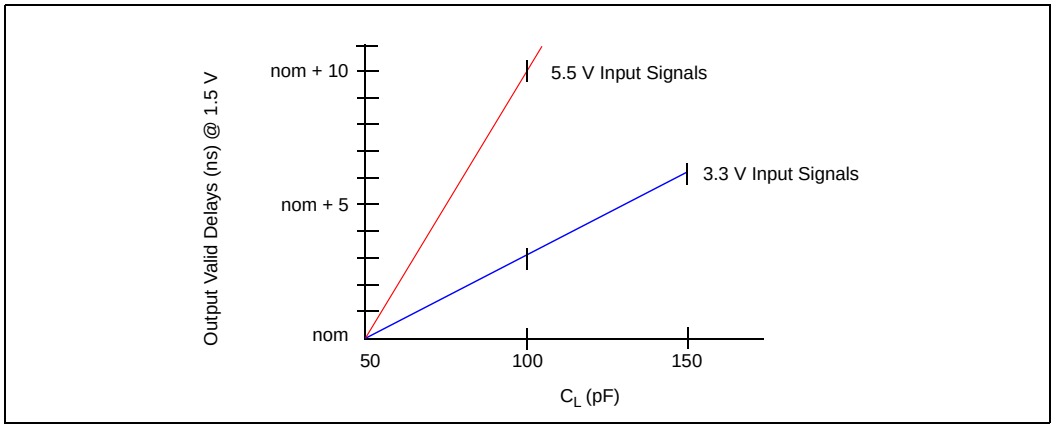


Figure 26. Output Delay vs. Temperature

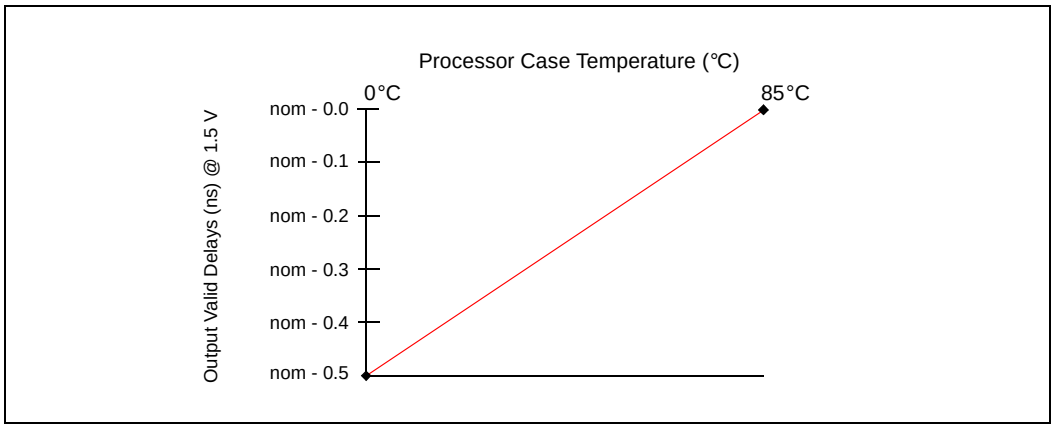


Figure 27. Output Hold Times vs. Temperature

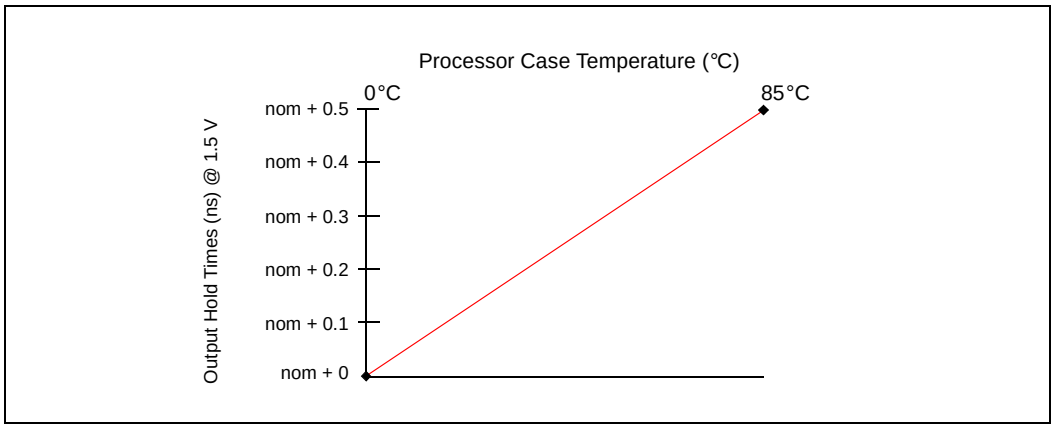
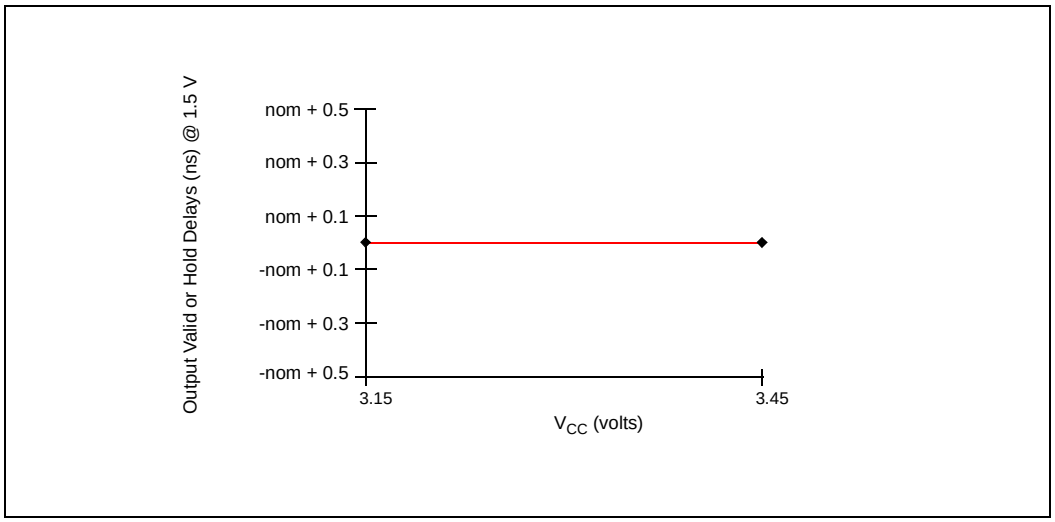


Figure 28. Output Delay vs. V_{CC}



5.0 Bus Waveforms

Figure 29. Cold Reset Waveform

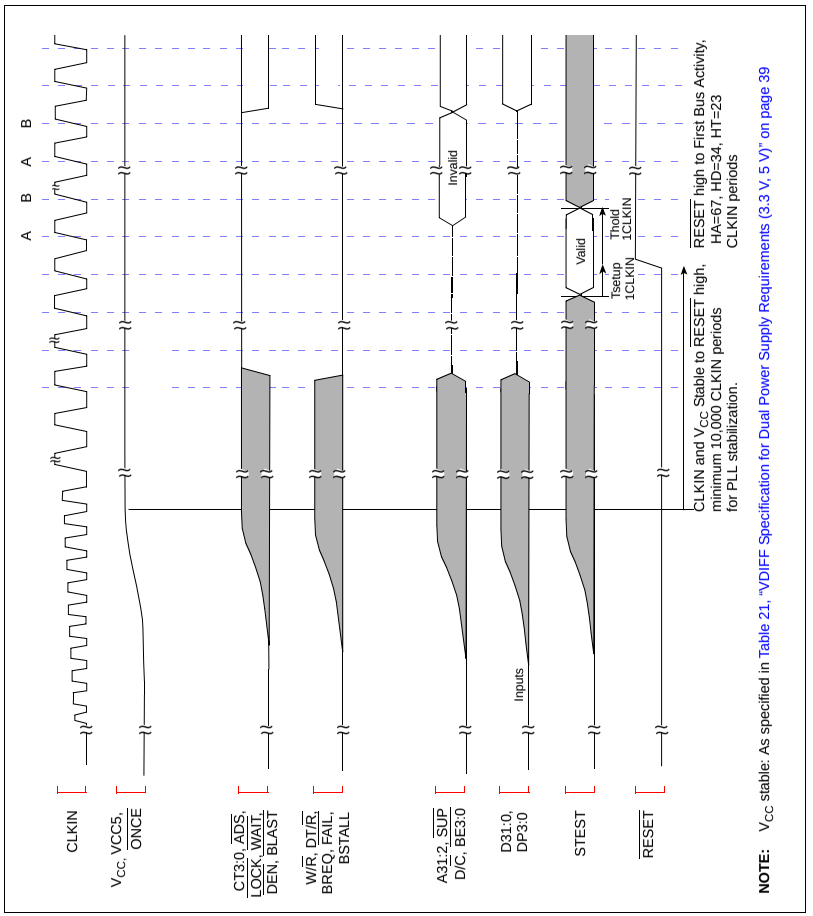


Figure 30. Warm Reset Waveform

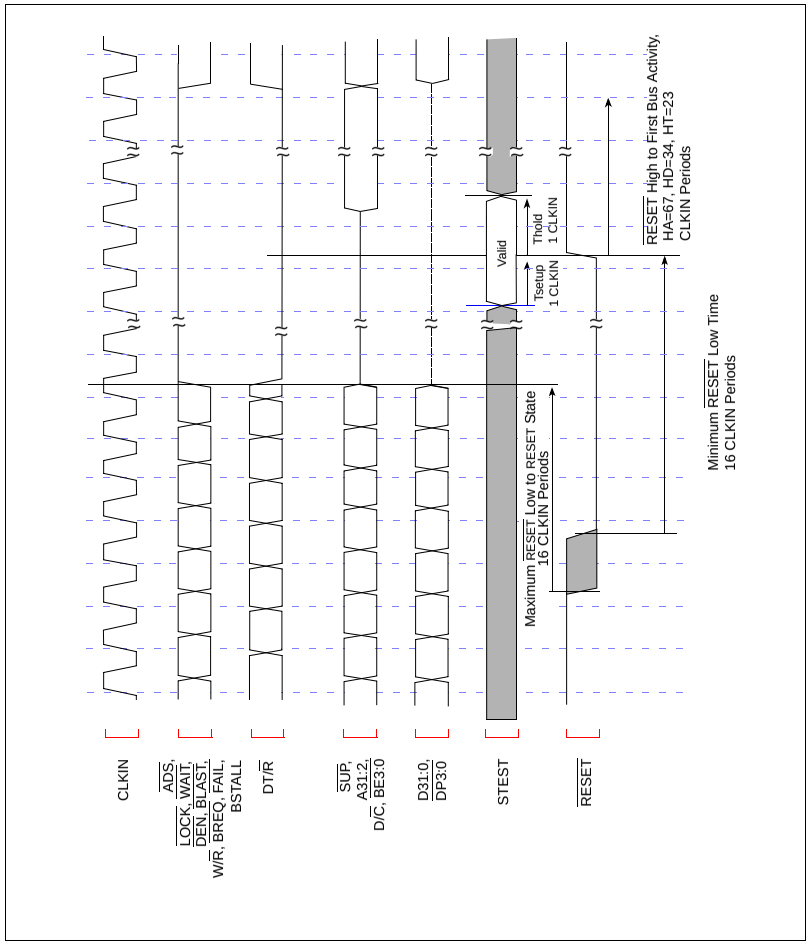


Figure 31. Entering $\overline{\text{ONCE}}$ Mode

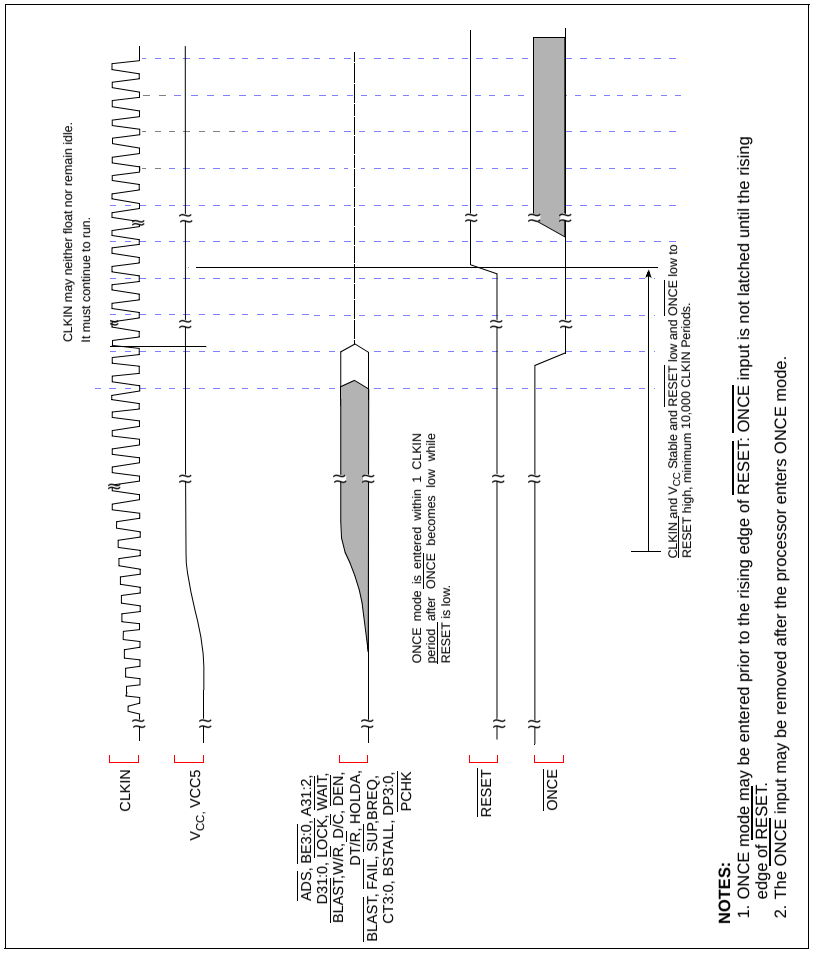


Figure 32. Non-Burst, Non-Pipelined Requests without Wait States

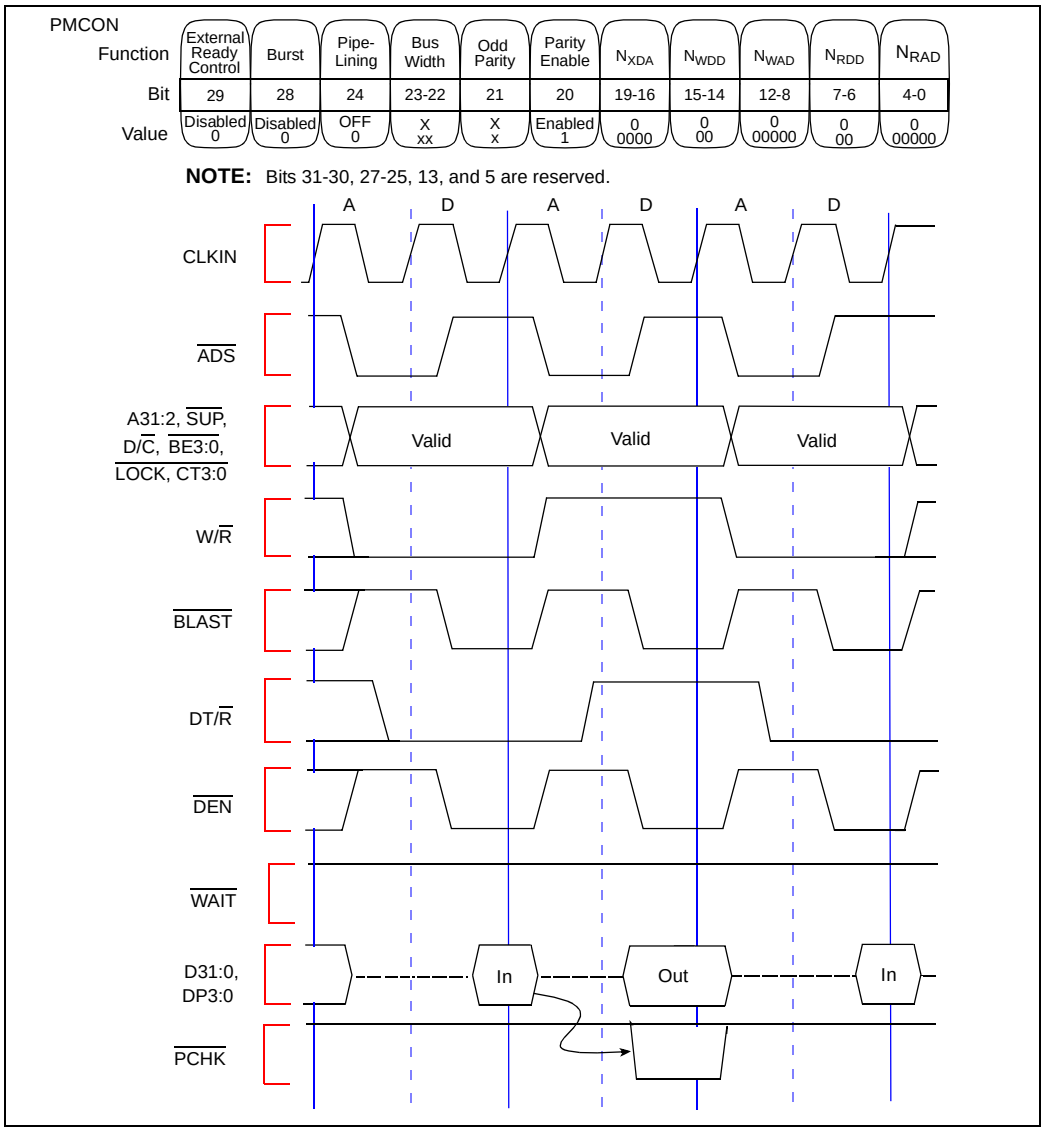


Figure 33. Non-Burst, Non-Pipelined Read Request with Wait States

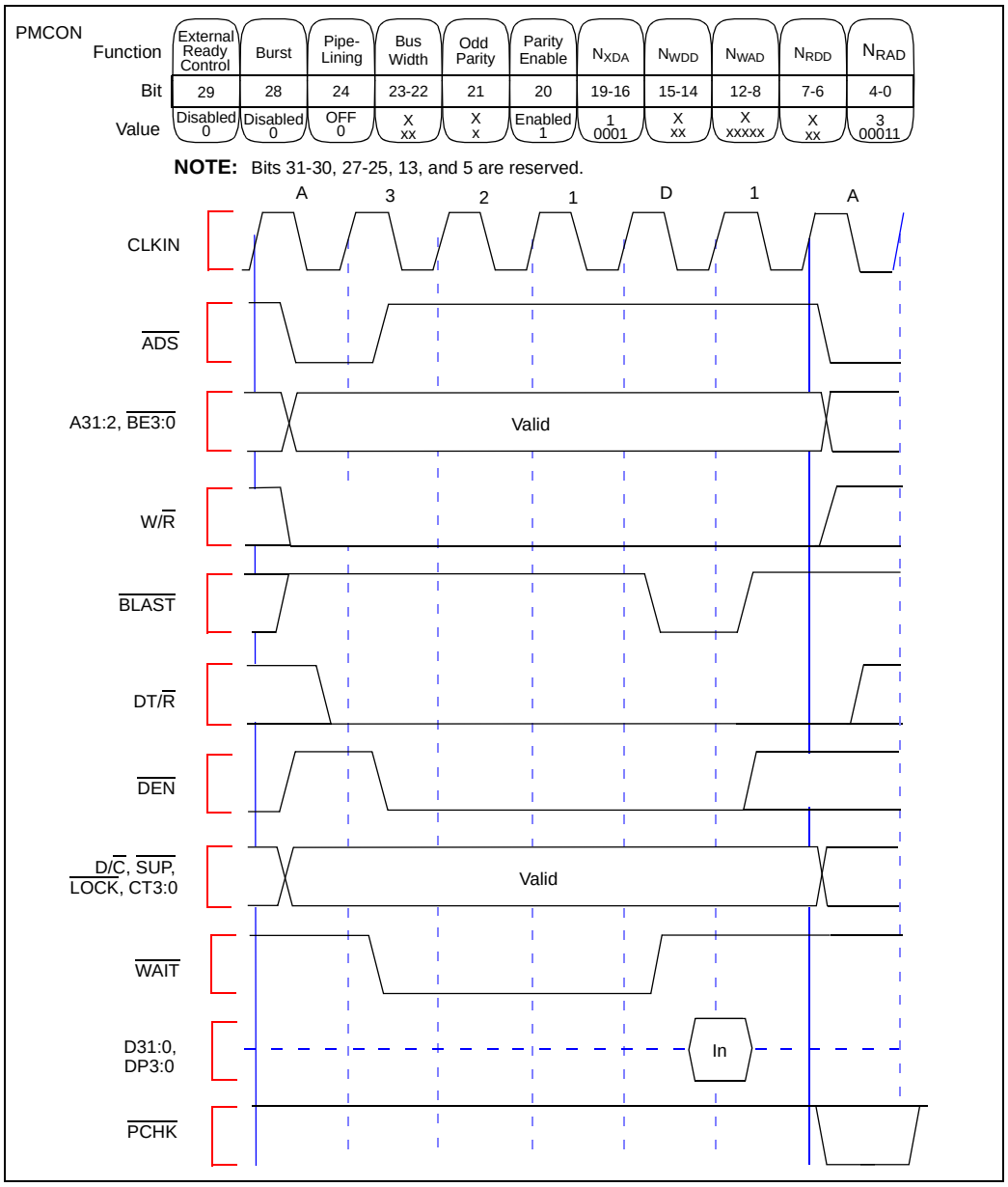


Figure 34. Non-Burst, Non-Pipelined Write Request with Wait States

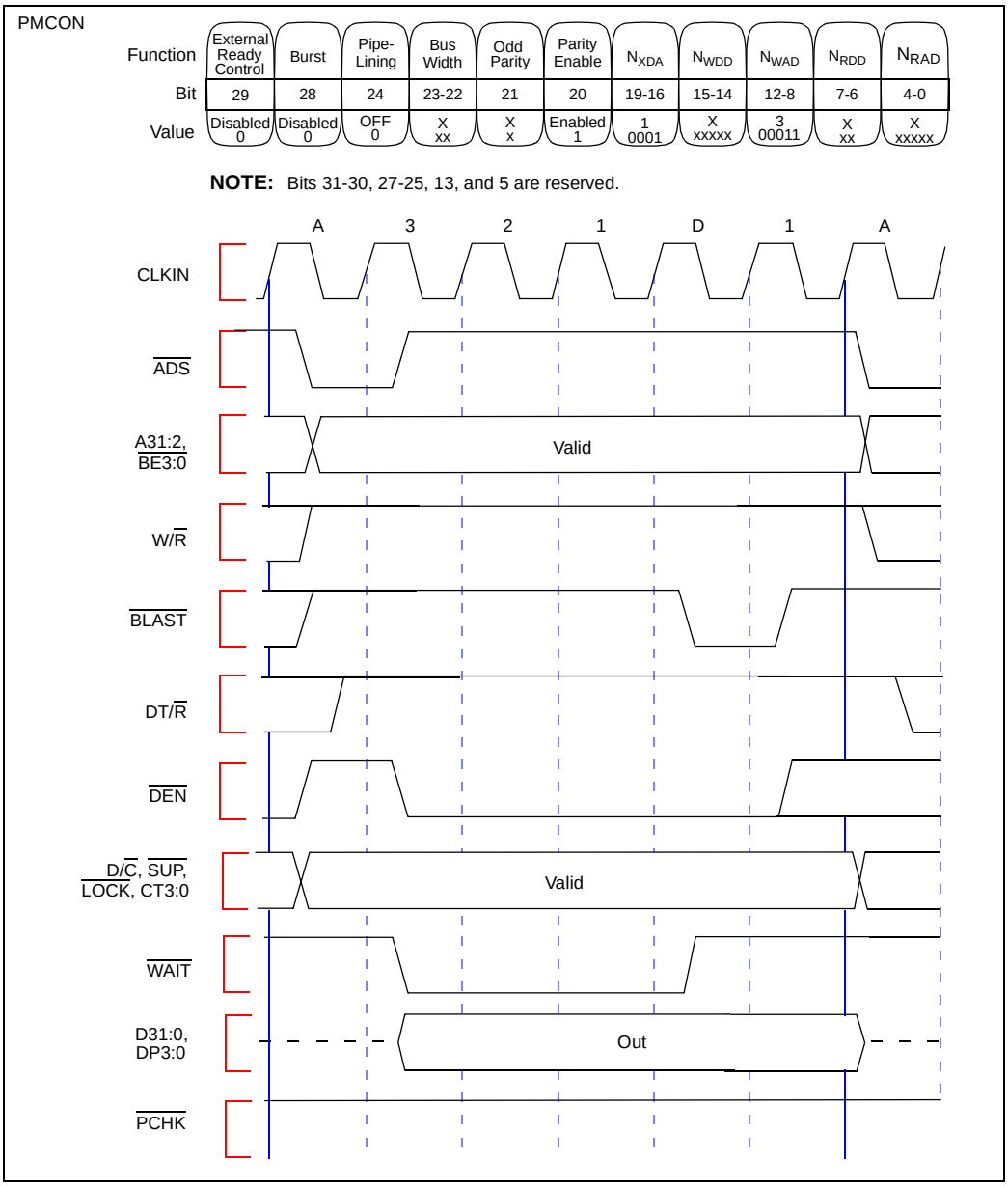


Figure 35. Burst, Non-Pipelined Read Request without Wait States, 32-Bit Bus

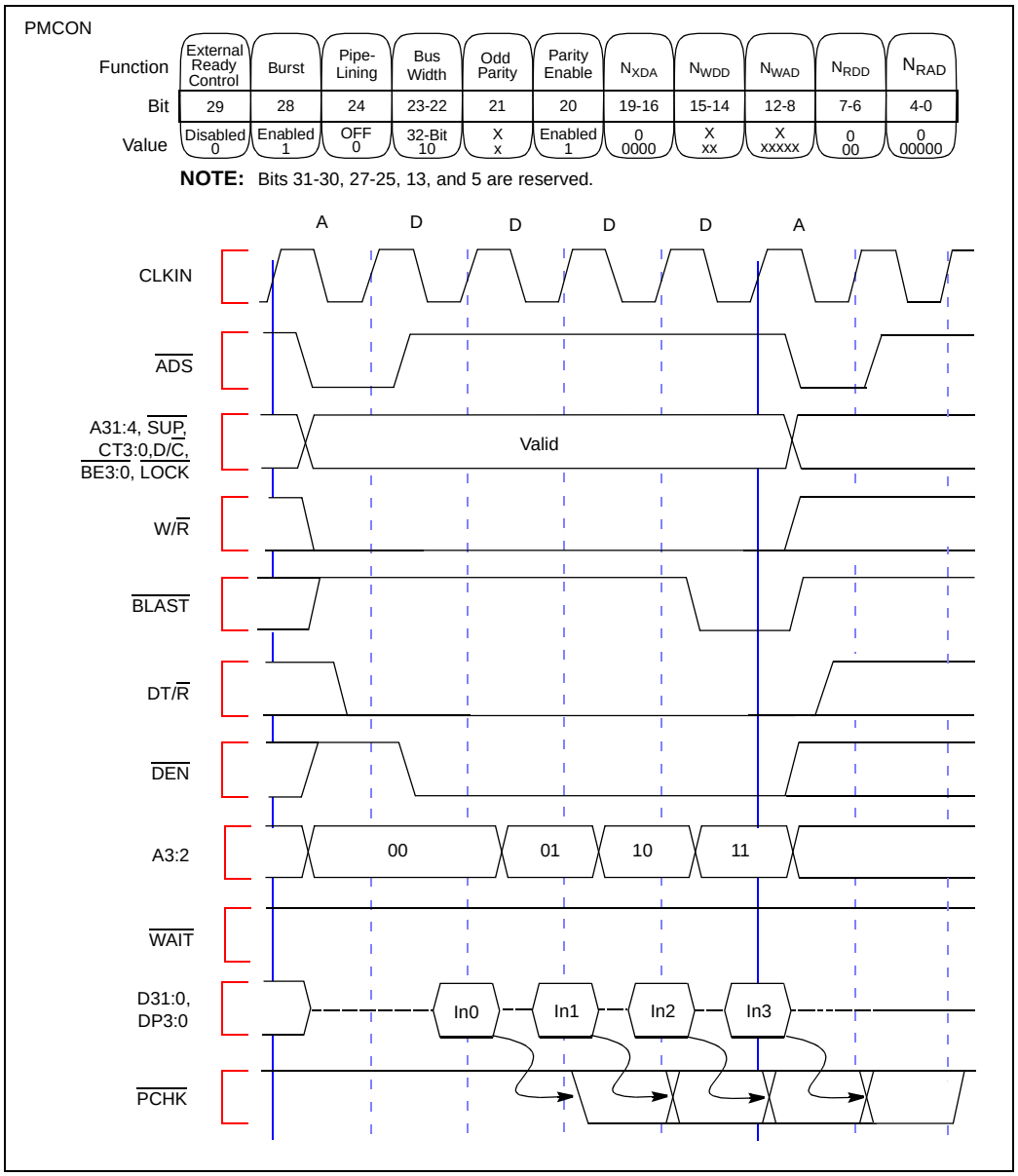


Figure 36. Burst, Non-Pipelined Read Request with Wait States, 32-Bit Bus

PMCON

Function	External Ready Control	Burst	Pipe-Lining	Bus Width	Odd Parity	Parity Enable	N_{XDA}	N_{WDD}	N_{WAD}	N_{RDD}	N_{RAD}
Bit	29	28	24	23-22	21	20	19-16	15-14	12-8	7-6	4-0
Value	Disabled 0	Enabled 1	OFF 0	32-Bit 10	X x	Enabled 1	1 0001	X xx	X xxxxx	1 01	2 00010

NOTE: Bits 31-30, 27-25, 13, and 5 are reserved.

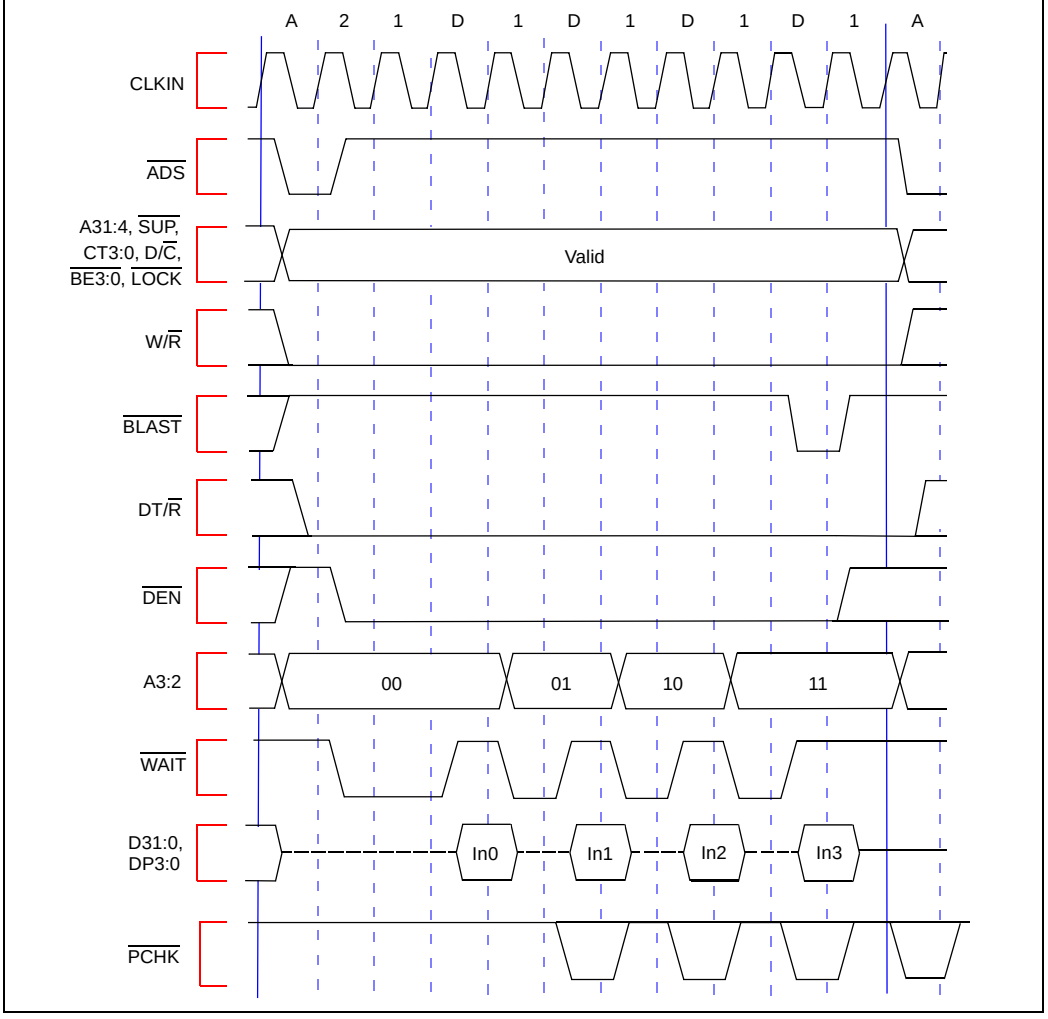


Figure 37. Burst, Non-Pipelined Write Request without Wait States, 32-Bit Bus

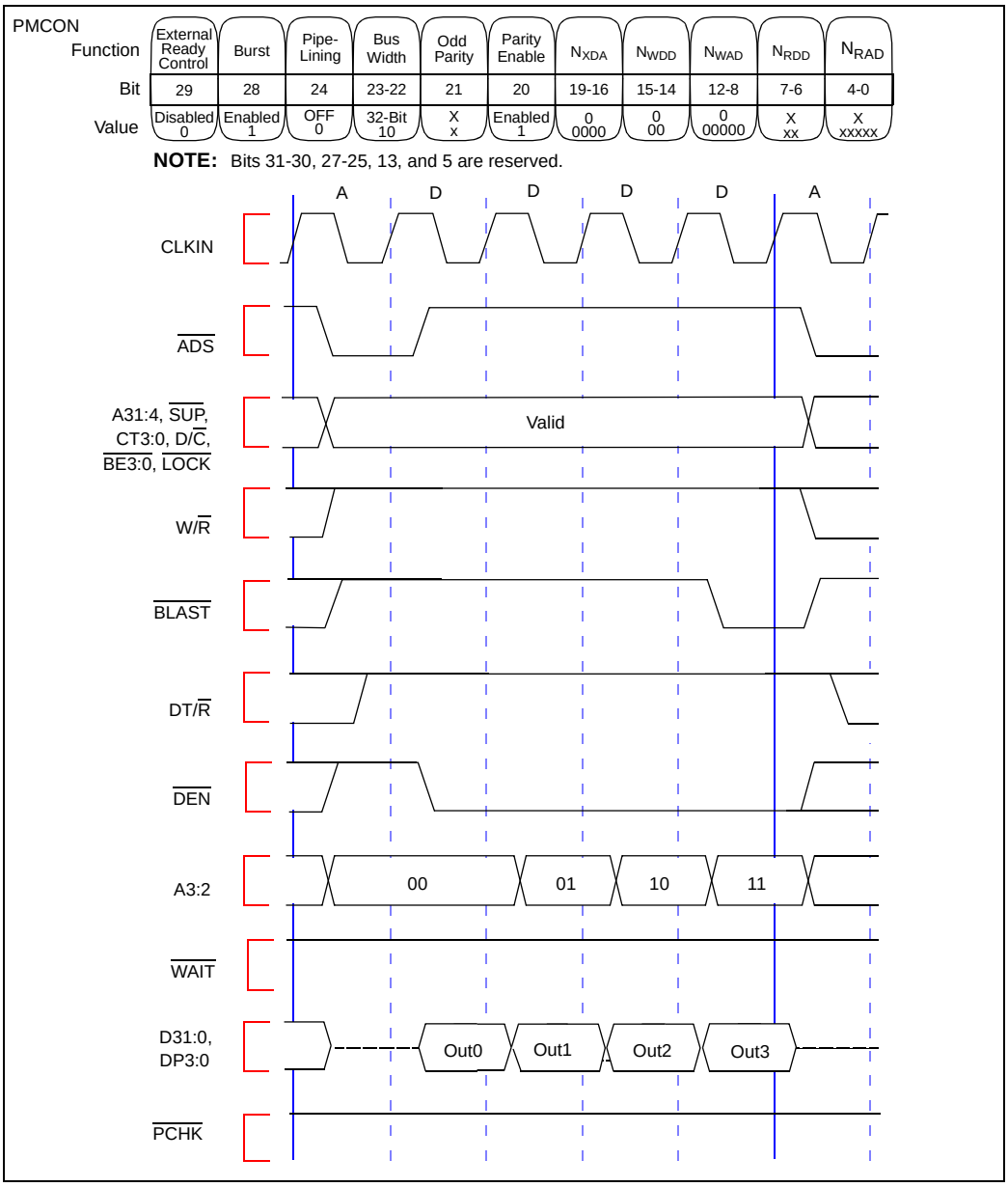


Figure 38. Burst, Non-Pipelined Write Request with Wait States, 32-Bit Bus

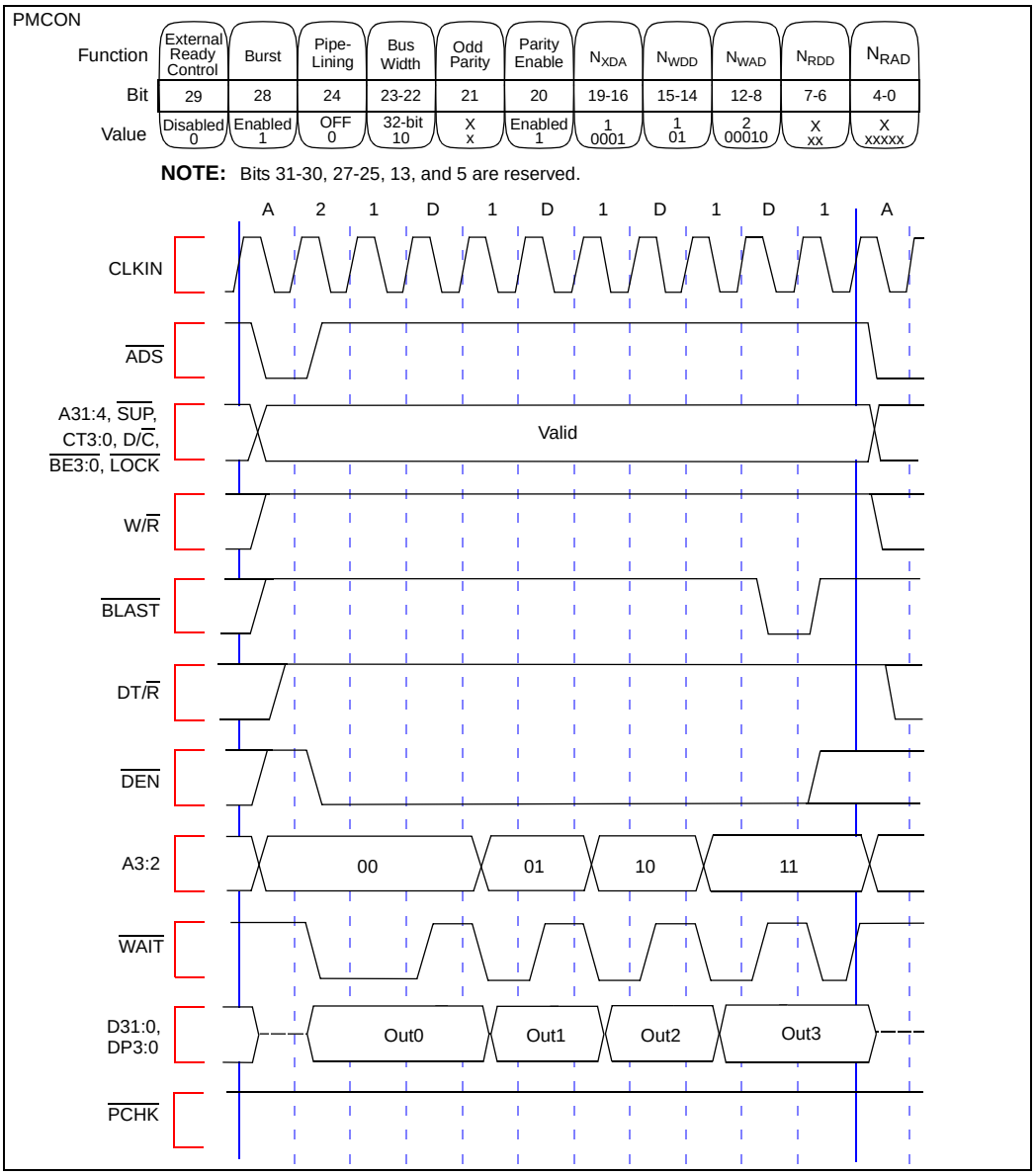


Figure 39. Burst, Non-Pipelined Read Request with Wait States, 16-Bit Bus

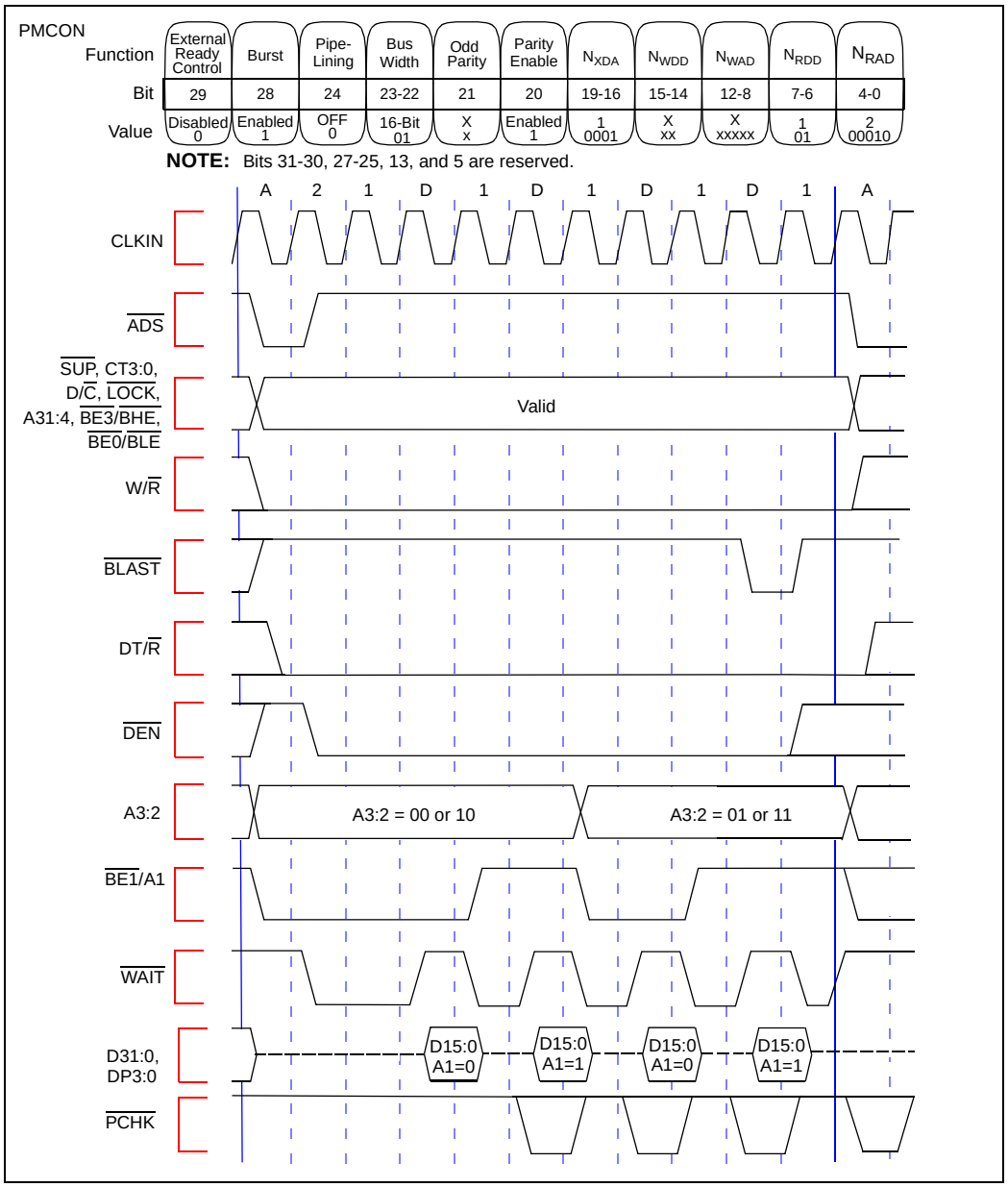


Figure 40. Burst, Non-Pipelined Read Request with Wait States, 8-Bit Bus

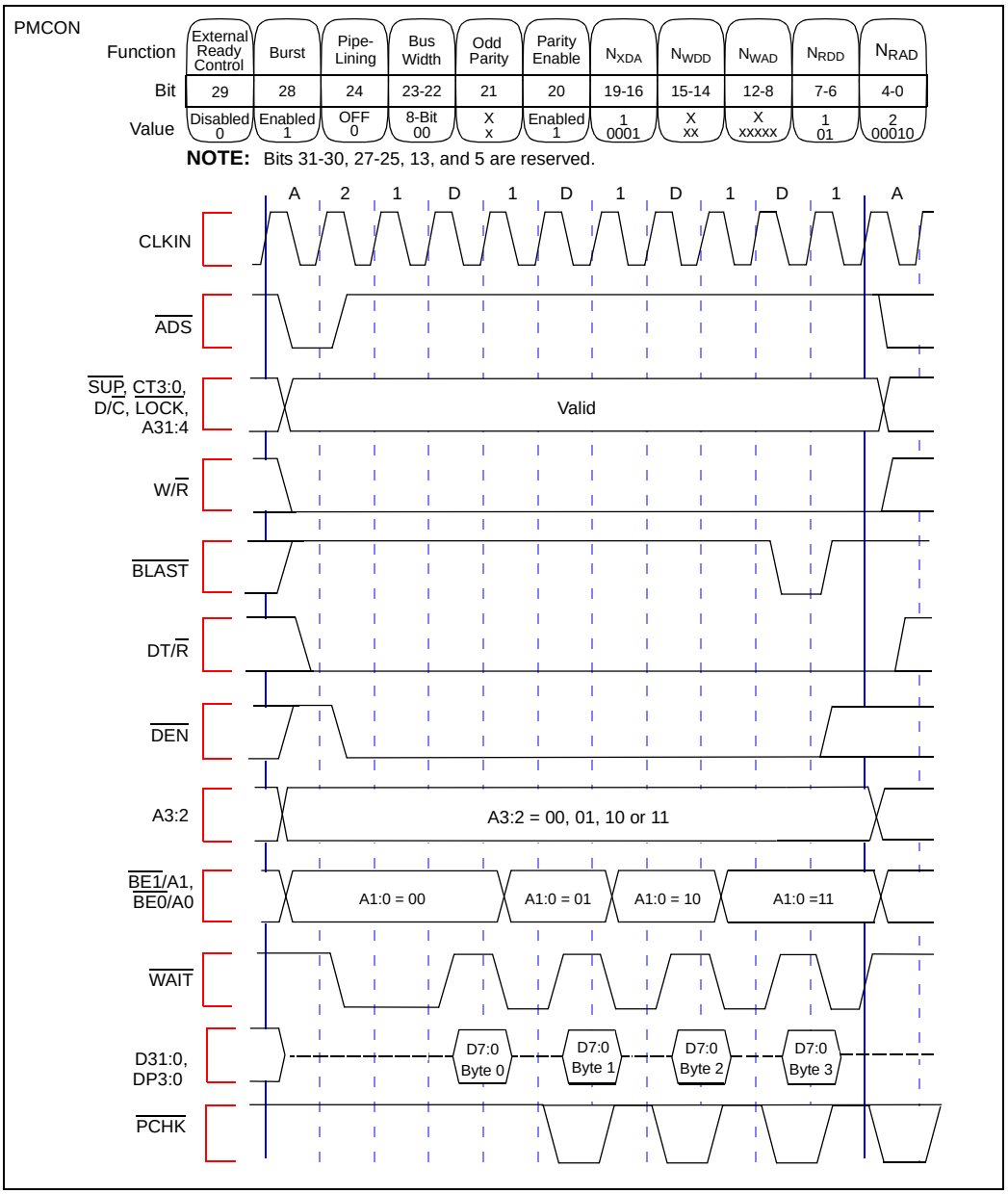
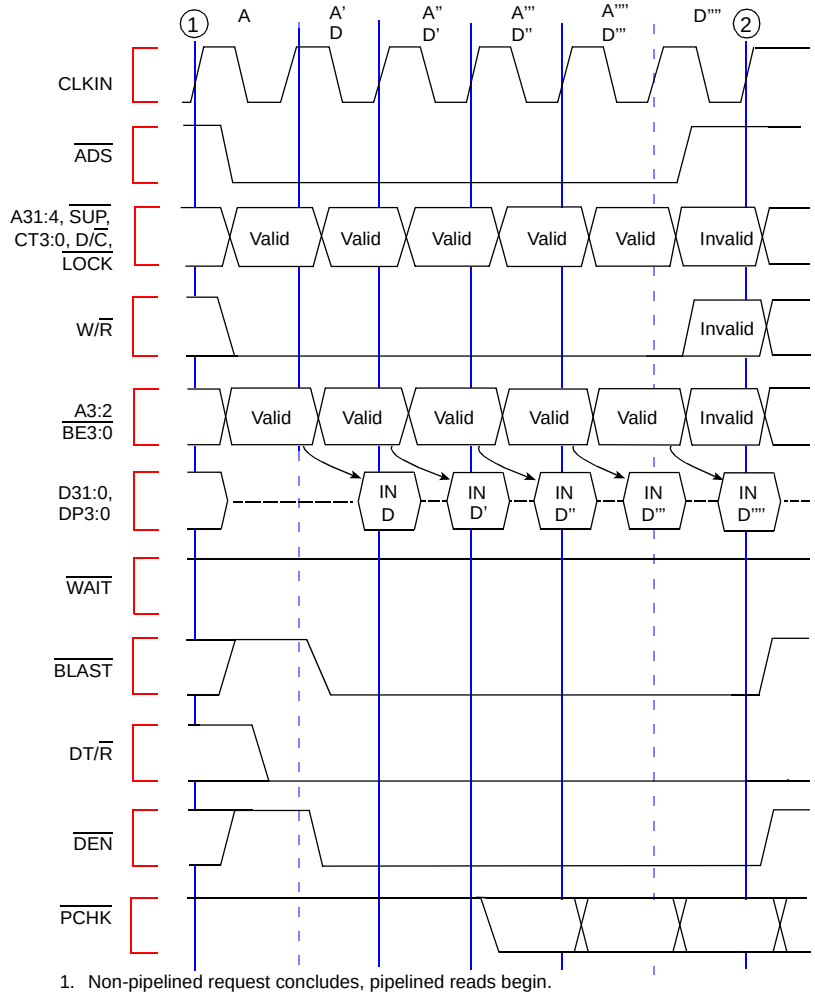


Figure 41. Non-Burst, Pipelined Read Request without Wait States, 32-Bit Bus

PMCON Function	External Ready Control	Burst	Pipe-Lining	Bus Width	Odd Parity	Parity Enable	N_{XDA}	N_{WDD}	N_{WAD}	N_{RDD}	N_{RAD}
Bit	29	28	24	23-22	21	20	19-16	15-14	12-8	7-6	4-0
Value	X x	Disabled 0	ON 1	32-Bit 10	X x	Enabled 1	X xxxx	X xx	X xxxxx	X xx	0 00000

NOTE: Bits 31-30, 27-25, 13, and 5 are reserved.



1. Non-pipelined request concludes, pipelined reads begin.
2. Pipelined reads conclude, non-pipelined requests begin.

Figure 42. Non-Burst, Pipelined Read Request with Wait States, 32-Bit Bus

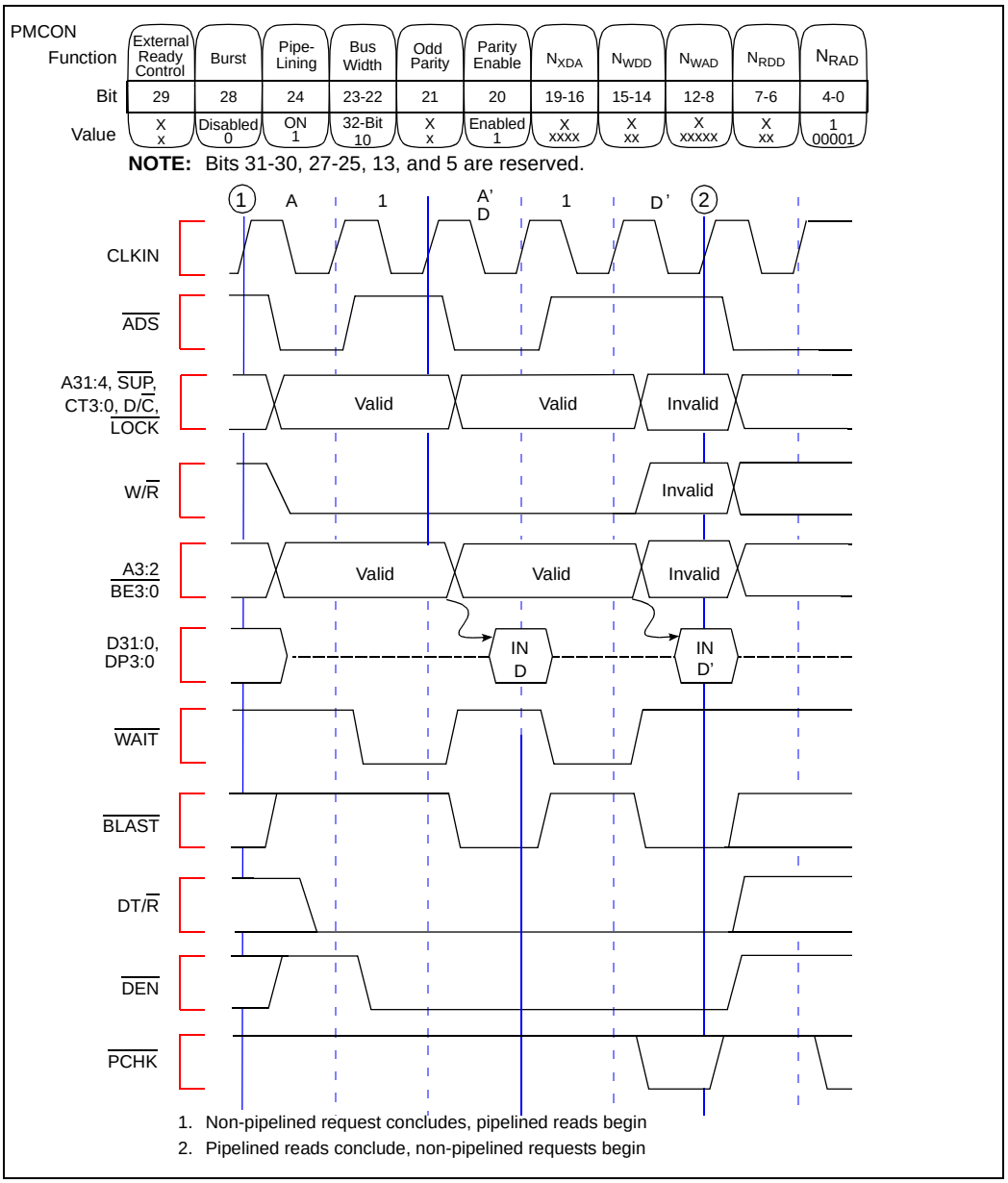
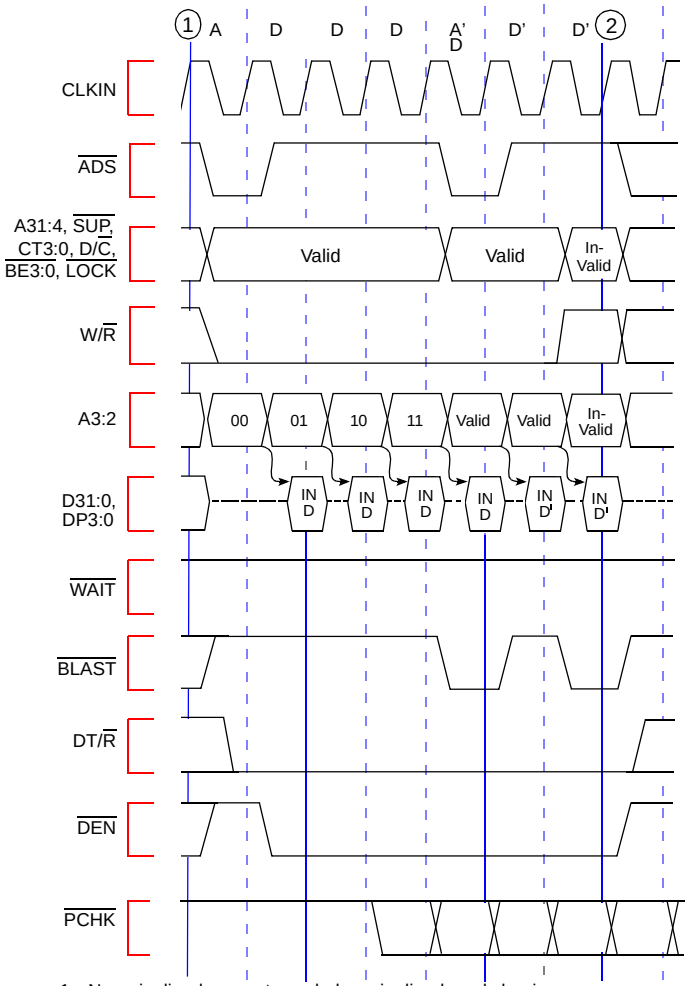


Figure 43. Burst, Pipelined Read Request without Wait States, 32-Bit Bus

PMCON Function	External Ready Control	Burst	Pipe-Lining	Bus Width	Odd Parity	Parity Enable	N_{XDA}	N_{WDD}	N_{WAD}	N_{RDD}	N_{RAD}
Bit	29	28	24	23-22	21	20	19-16	15-14	12-8	7-6	4-0
Value	X x	Enabled 1	ON 1	32-Bit 10	X x	Enabled 1	X xxxx	X xx	X xxxxx	0 00	0 00000

NOTE: Bits 31-30, 27-25, 13, and 5 are reserved.



1. Non-pipelined request concludes, pipelined reads begin
2. Pipelined reads conclude, non-pipelined requests begin

Figure 44. Burst, Pipelined Read Request with Wait States, 32-Bit Bus

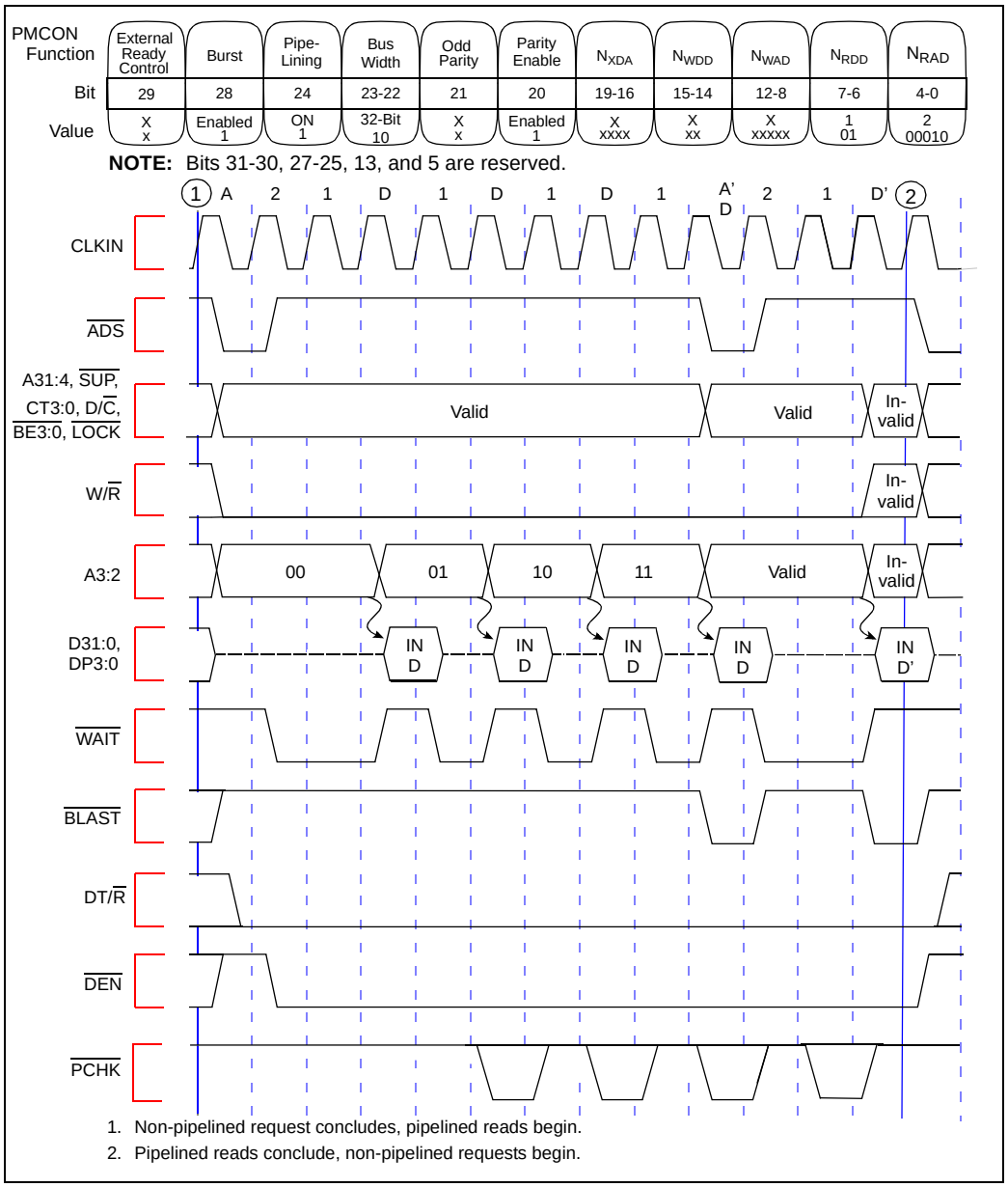


Figure 45. Burst, Pipelined Read Request with Wait States, 8-Bit Bus

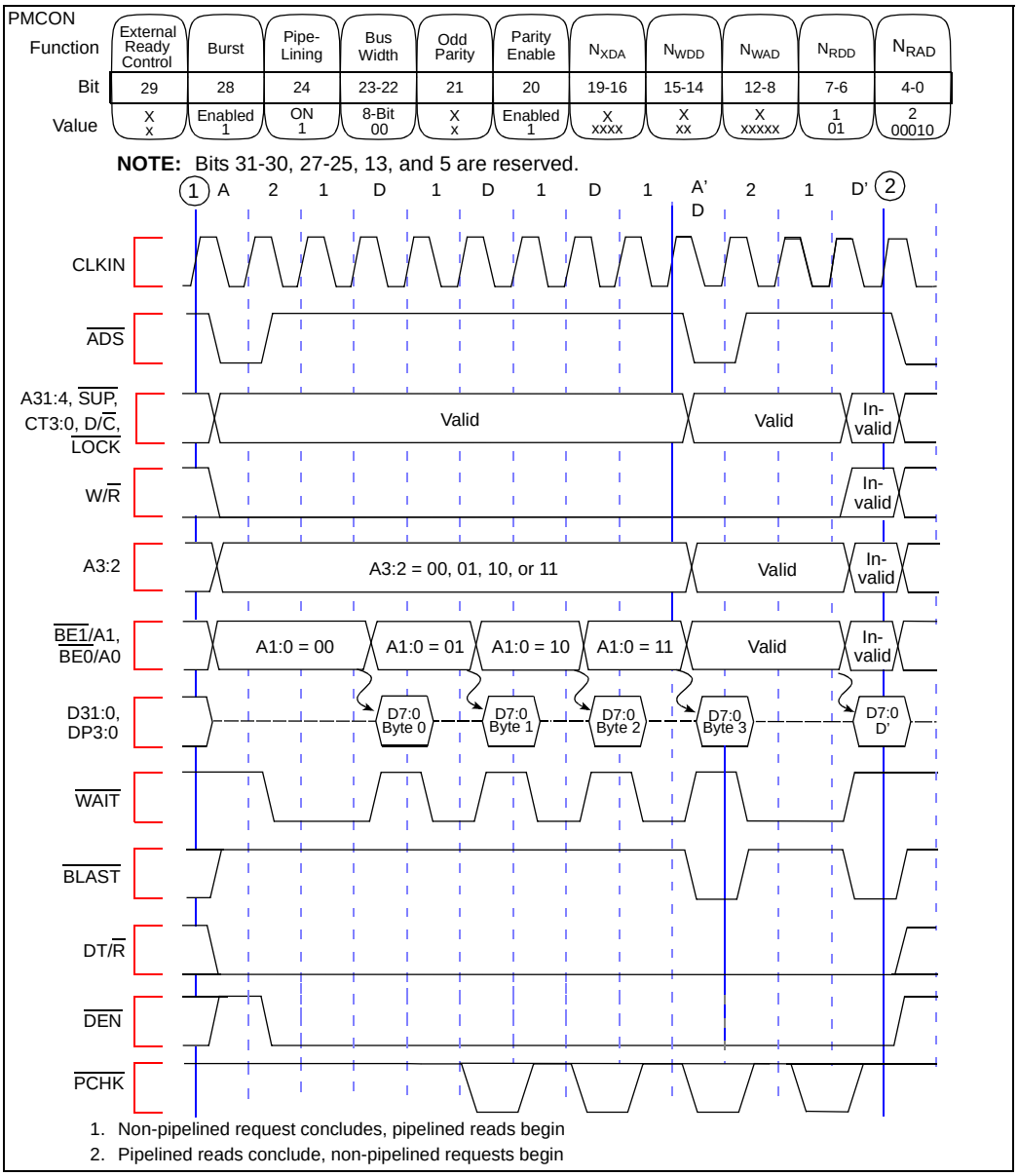
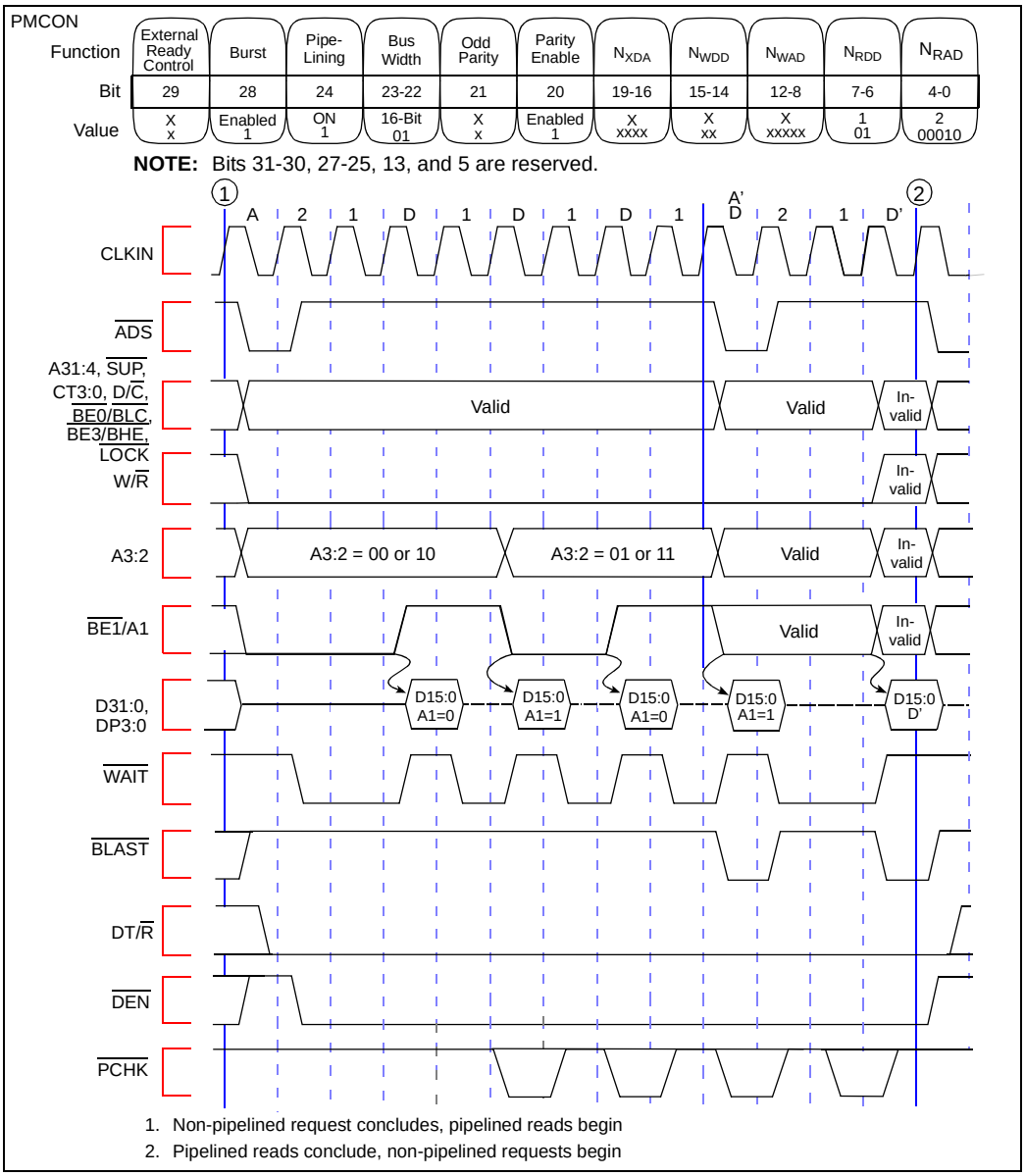
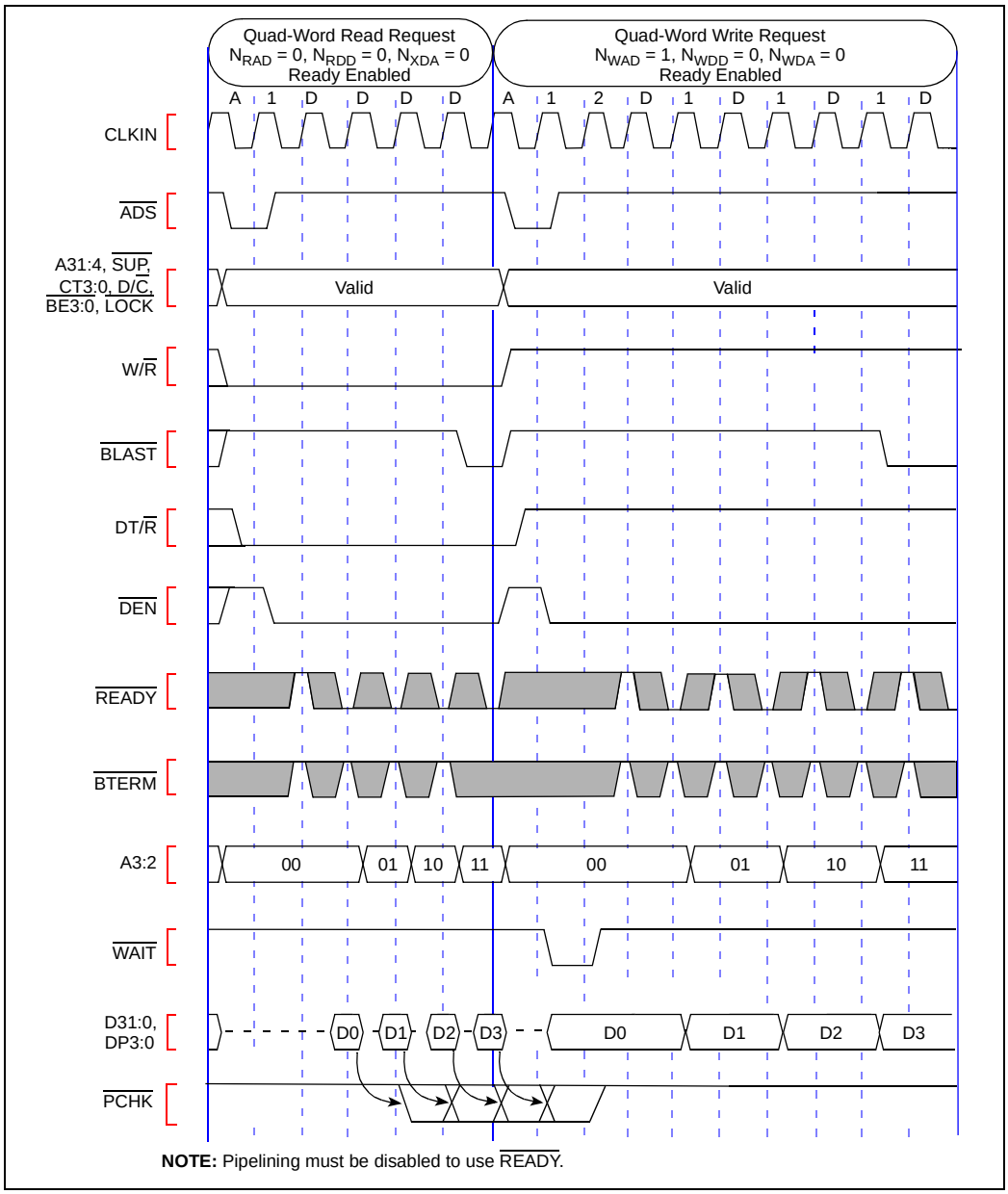


Figure 46. Burst, Pipelined Read Request with Wait States, 16-Bit Bus



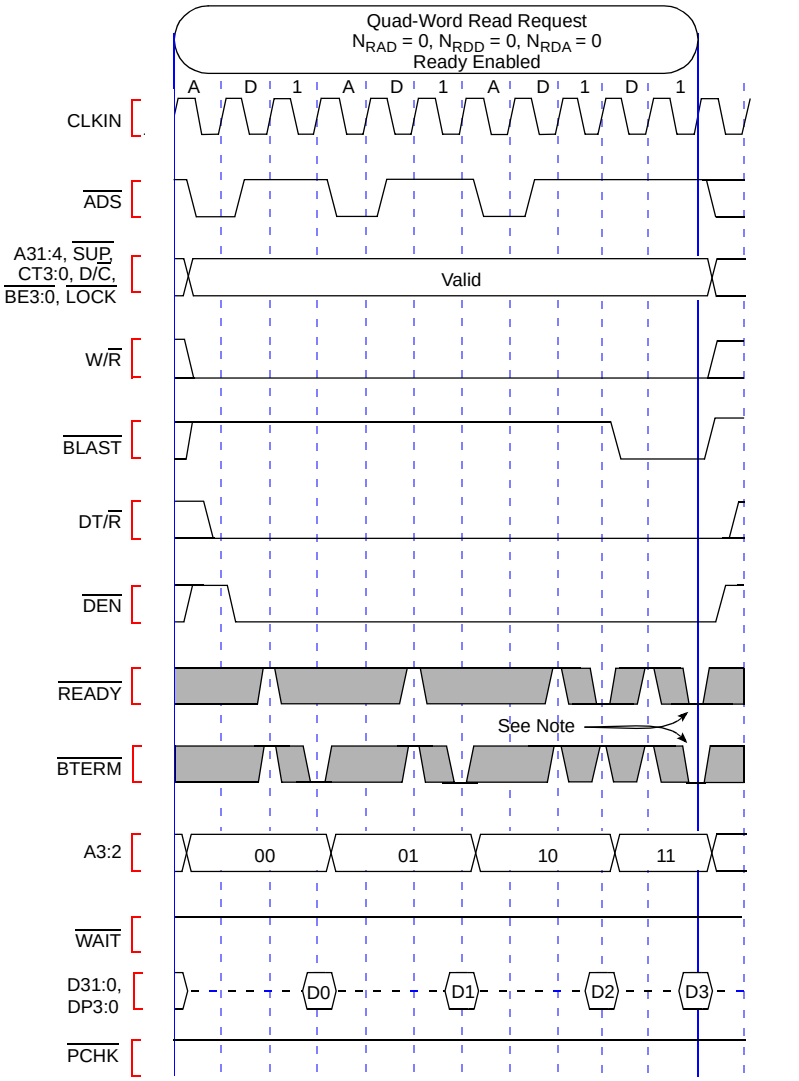
1. Non-pipelined request concludes, pipelined reads begin
2. Pipelined reads conclude, non-pipelined requests begin

Figure 47. Using External $\overline{\text{READY}}$



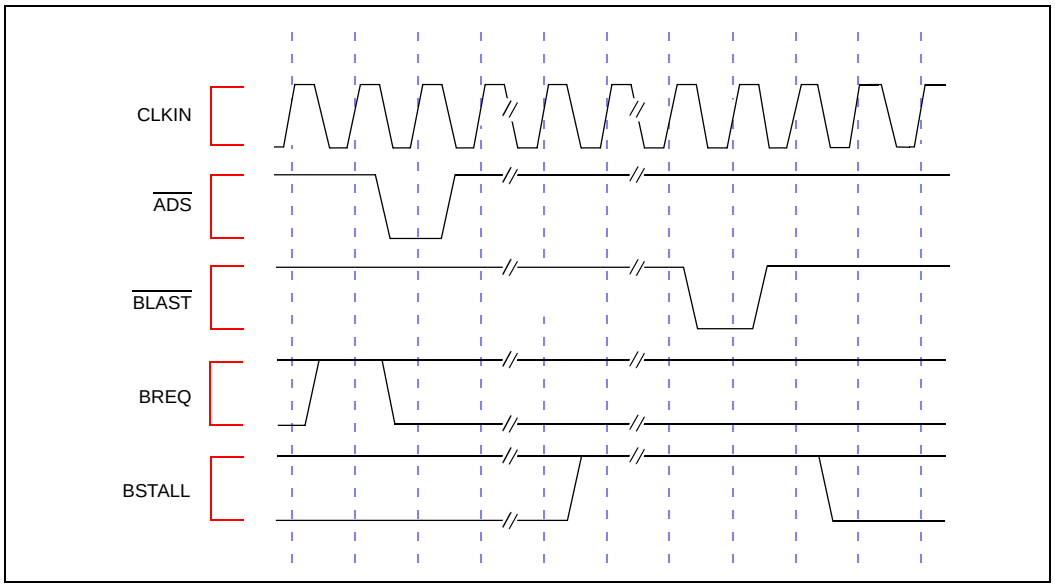
NOTE: Pipelining must be disabled to use $\overline{\text{READY}}$.

Figure 48. Terminating a Burst with $\overline{\text{BTERM}}$



Note: $\overline{\text{READY}}$ adds memory access time to data transfers, whether or not the bus access is a burst access. $\overline{\text{BTERM}}$ interrupts a bus access, whether or not the bus access has more data transfers pending. Either the $\overline{\text{READY}}$ signal or the $\overline{\text{BTERM}}$ signal terminates a bus access when the signal is asserted during the last (or only) data transfer of the bus access.

Figure 49. BREQ and BSTALL Operation



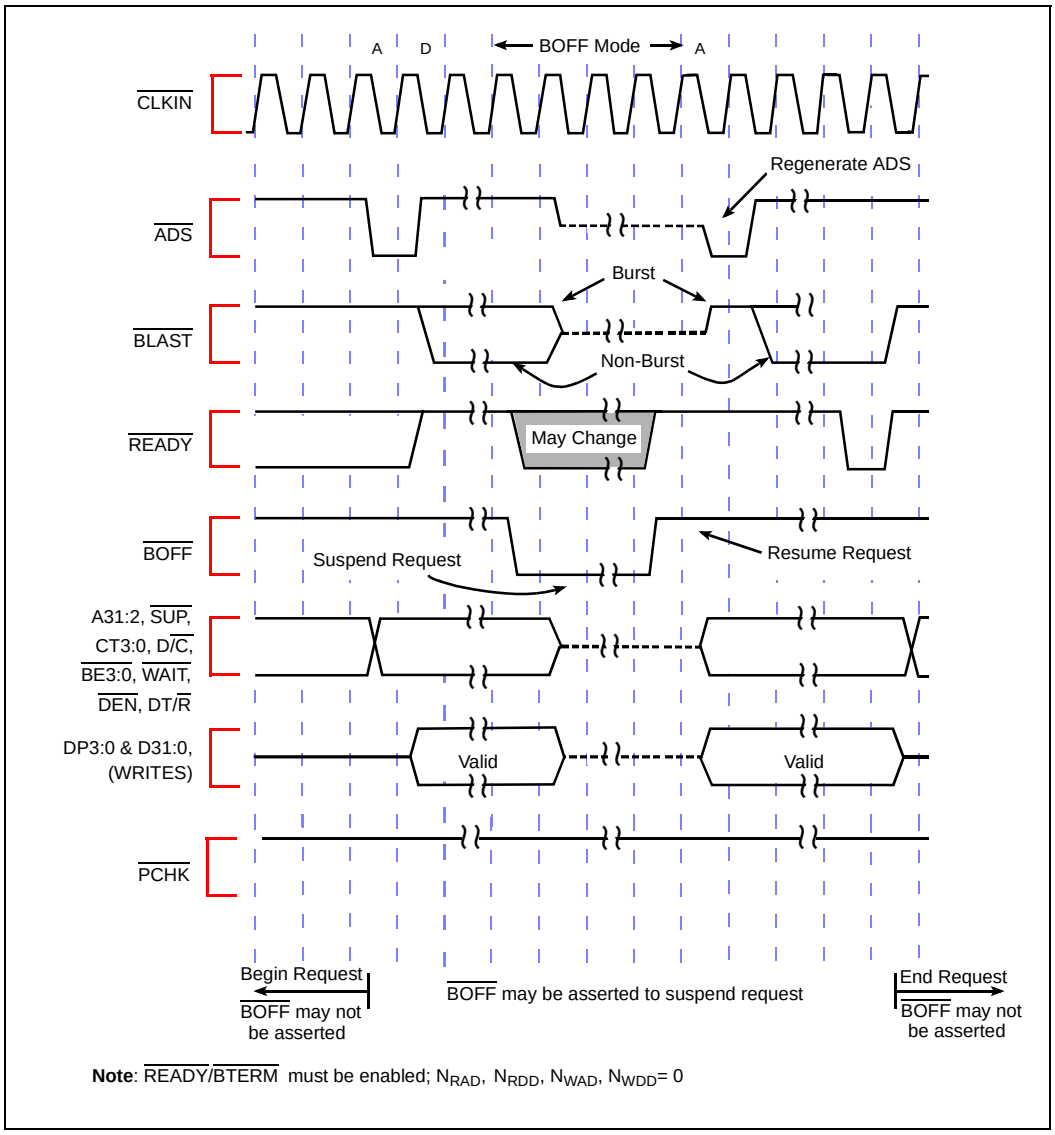
The processor may stall (BSTALL asserted) even with an empty bus queue (BREQ deasserted). Depending on the instruction stream and memory wait states, the two signals may be separated by several CLKIN cycles.

Bus arbitration logic that logically ‘ANDs’ BSTALL and BREQ will not correctly grant the bus to the processor in all stall cases, potentially degrading processor performance.

Do not logically ‘AND’ BSTALL and BREQ together in arbitration logic. Instead, the simplest bus arbitration should logically “OR” BSTALL and BREQ to determine the processor’s bus ownership requirements.

More sophisticated arbitration should recognize the priority nature of these two signals. Using a traffic light analogy, BREQ is a ‘yellow light’ warning of a possible processor stall and BSTALL is a ‘red light’ indicating a stall in progress.

Figure 50. $\overline{BOFF}$ Functional Timing. $\overline{BOFF}$ occurs during a burst or non-burst data cycle.



Note: READY/BTERM must be enabled; N_{RAD}, N_{RDD}, N_{WAD}, N_{WDD}= 0

Figure 51. HOLD Functional Timing

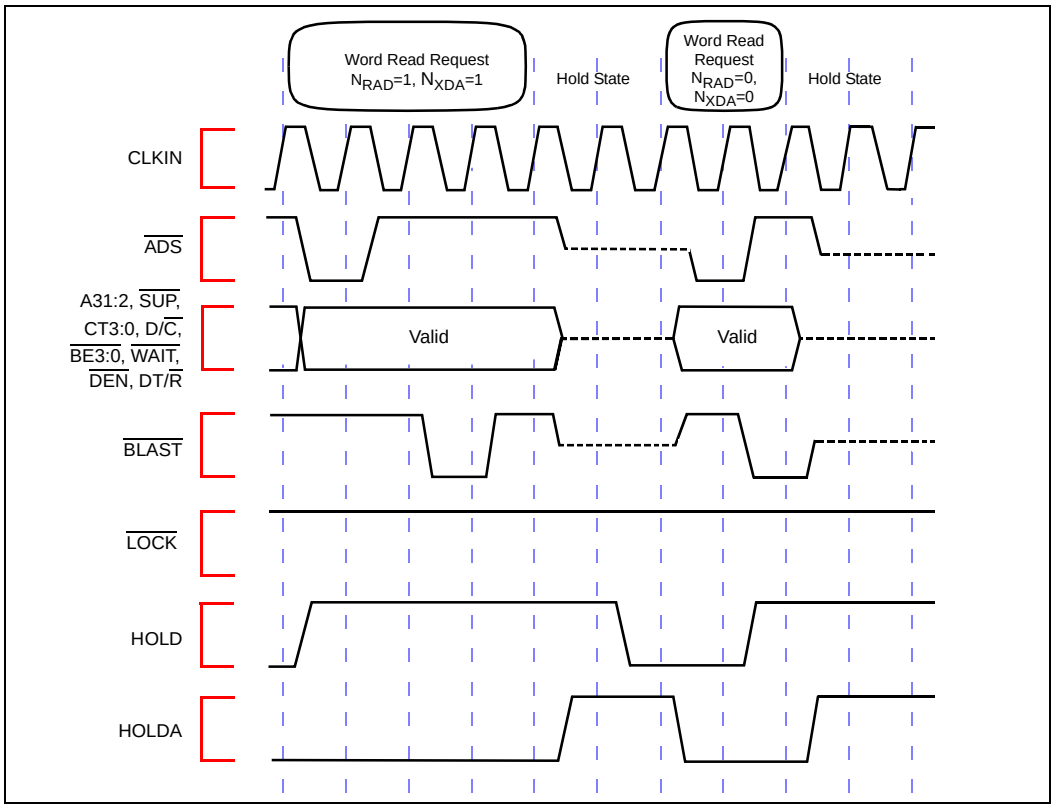


Figure 52. $\overline{\text{LOCK}}$ Delays HOLDA Timing

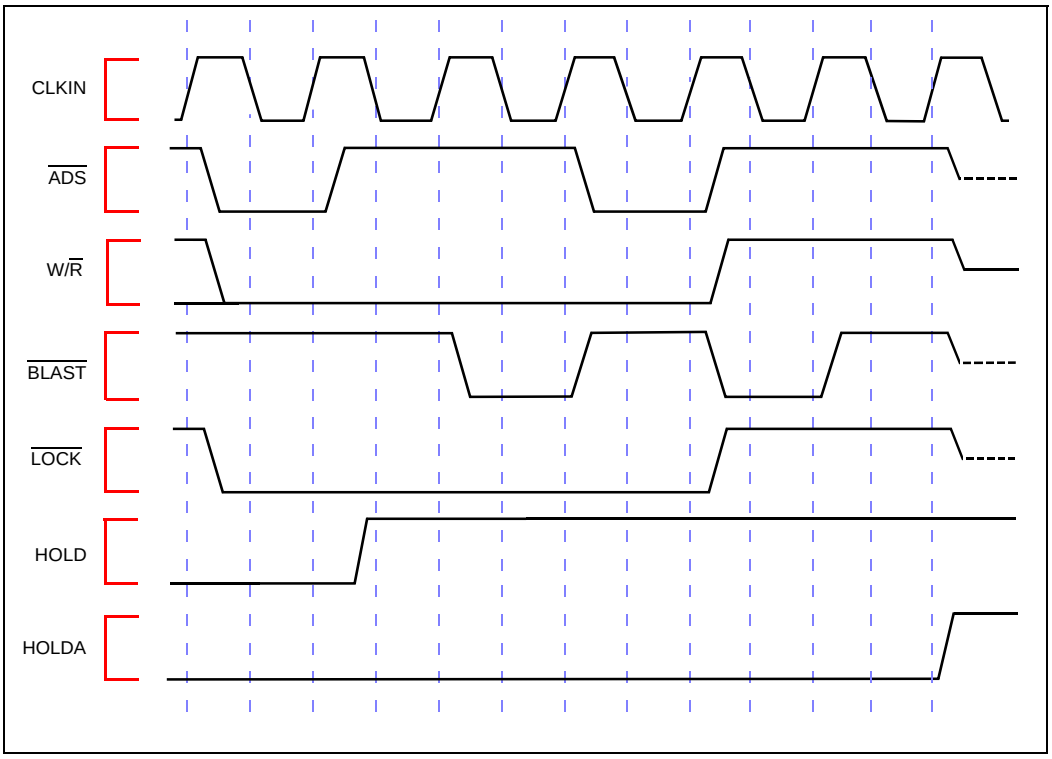


Figure 53. $\overline{\text{FAIL}}$ Functional Timing

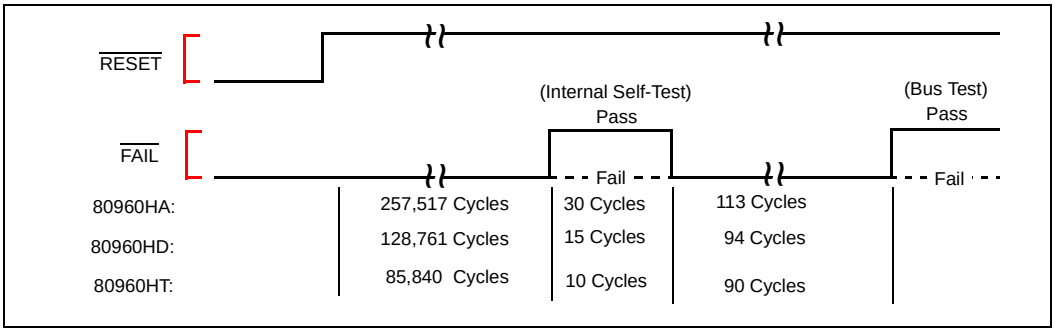
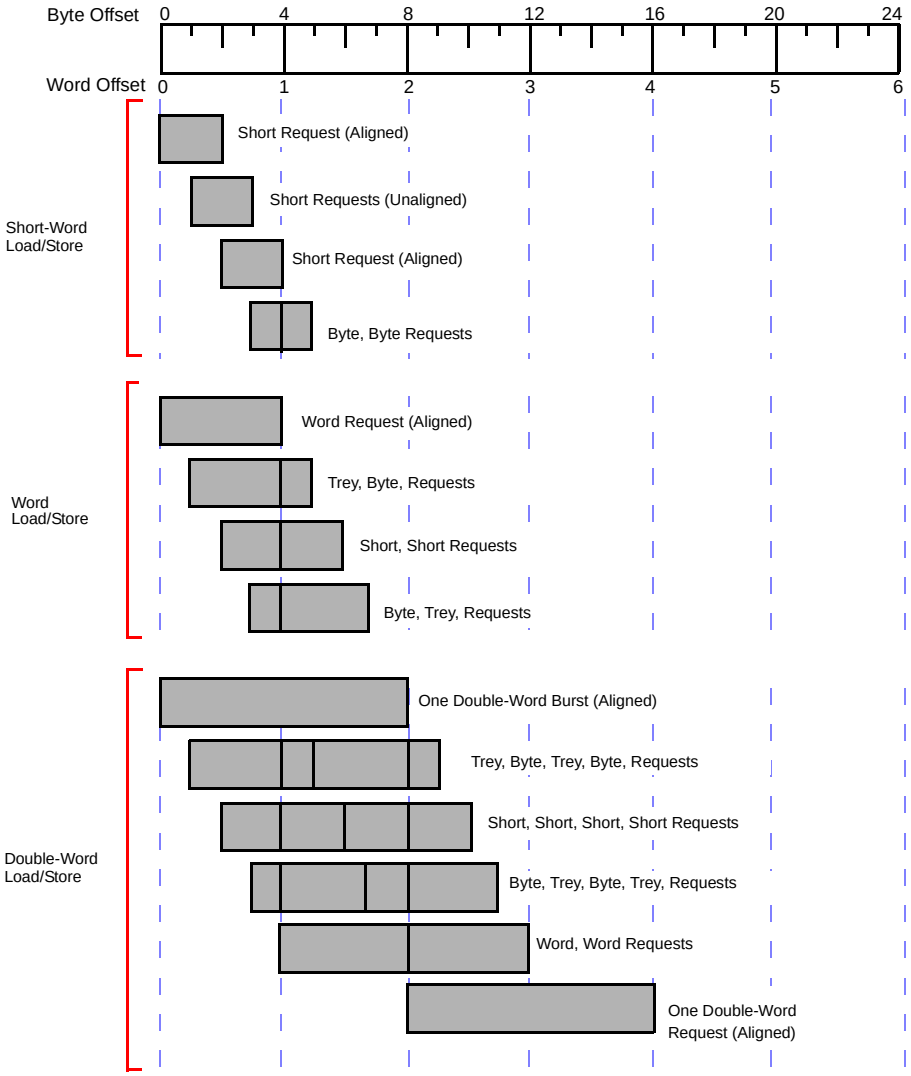


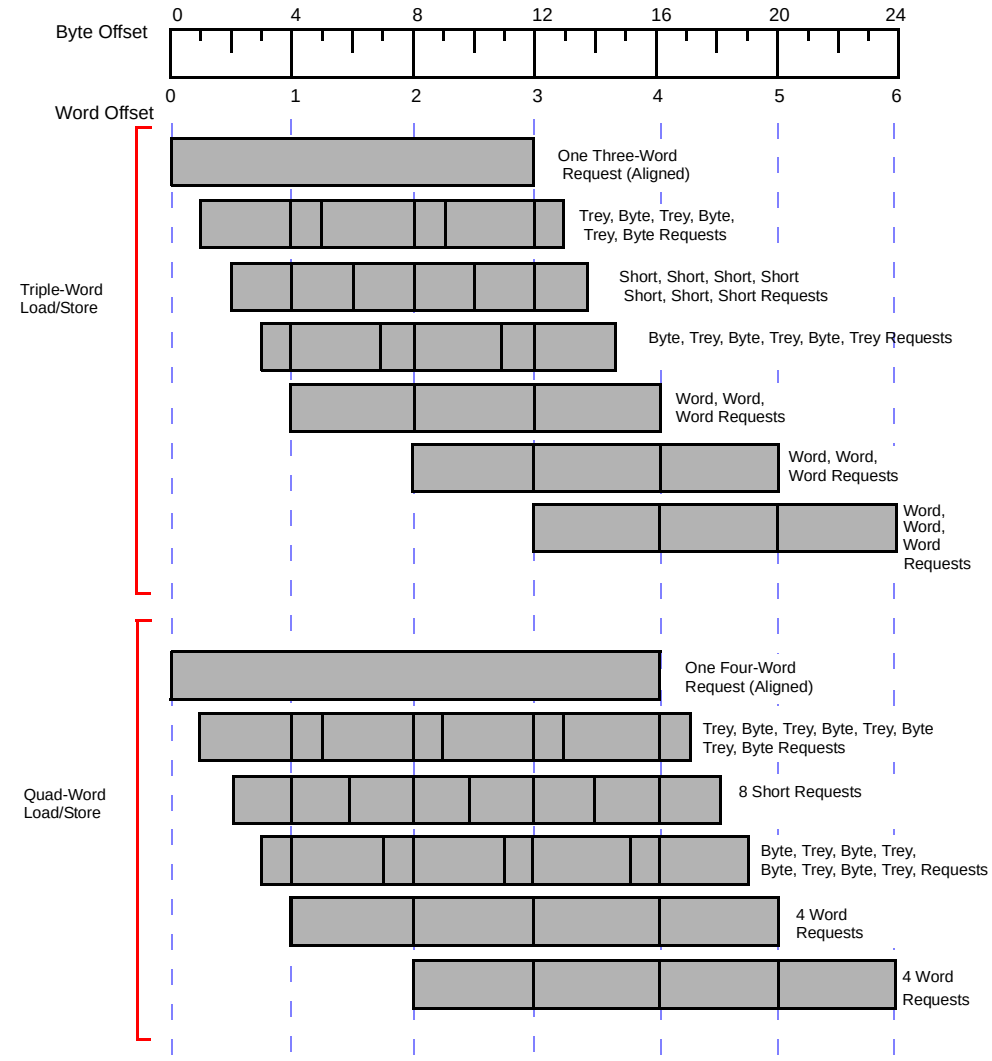
Figure 54. A Summary of Aligned and Unaligned Transfers for 32-Bit Regions



NOTES:

1. All requests that are less than a word in size and are cacheable will be promoted to a word to be cached. This causes adjacent requests to occur for full words to the same address.

Figure 55. A Summary of Aligned and Unaligned Transfers for 32-Bit Regions (Continued)



NOTES:

1. All requests that are less than a word in size and are cacheable will be promoted to a word to be cached. This causes adjacent requests to occur for full words to the same address.

Figure 56. A Summary of Aligned and Unaligned Transfers for 16-Bit Bus

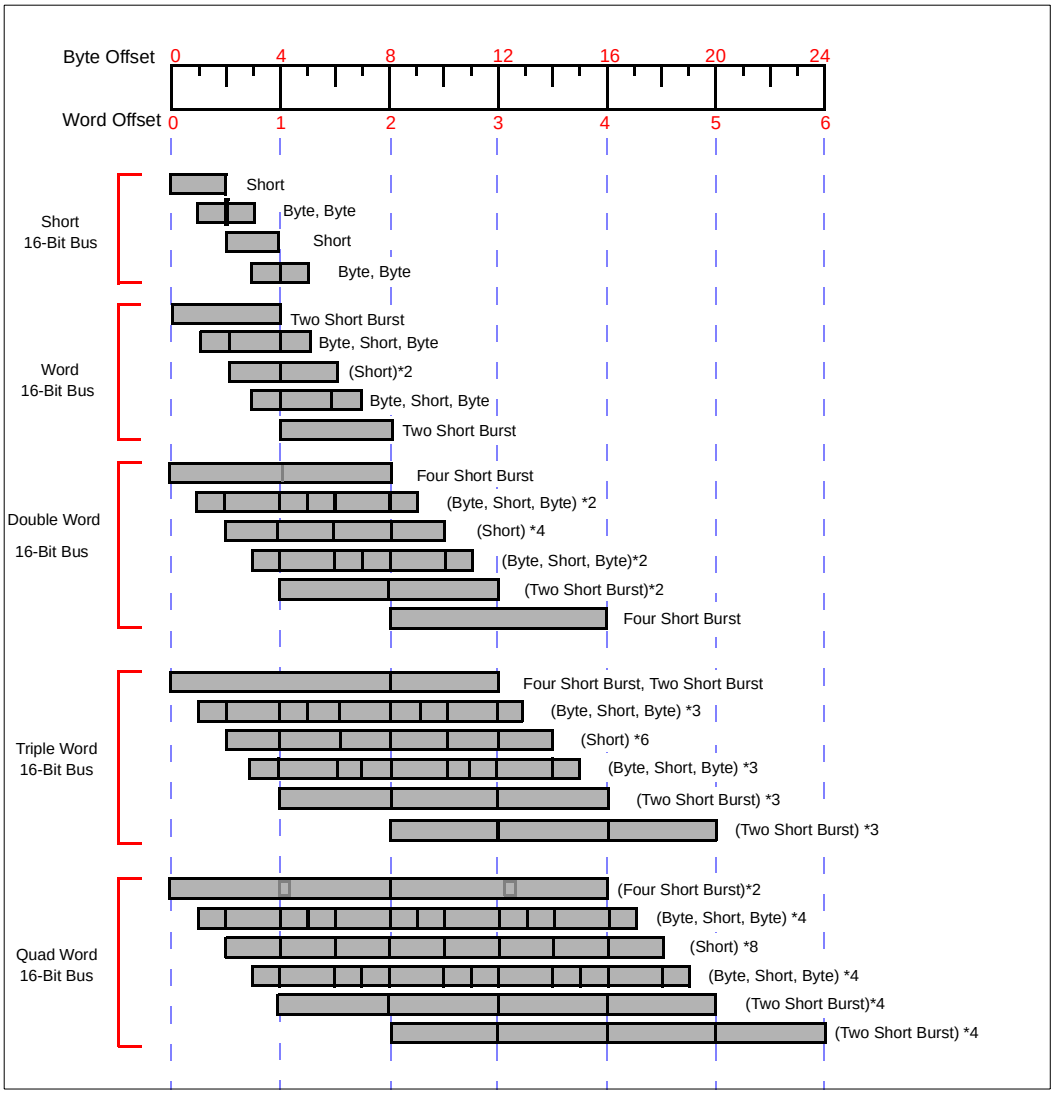


Figure 57. A Summary of Aligned and Unaligned Transfers for 8-Bit Bus

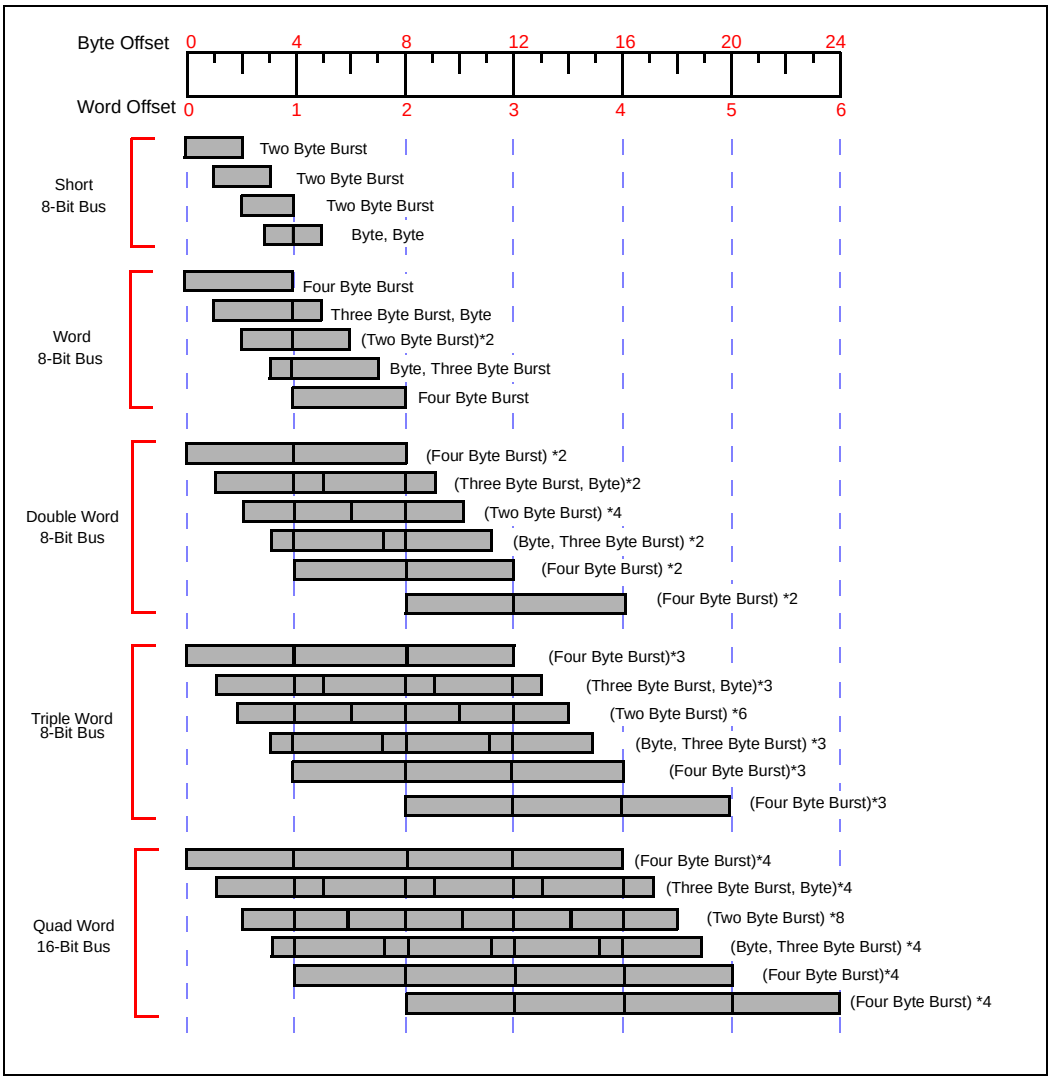


Figure 58. Idle Bus Operation

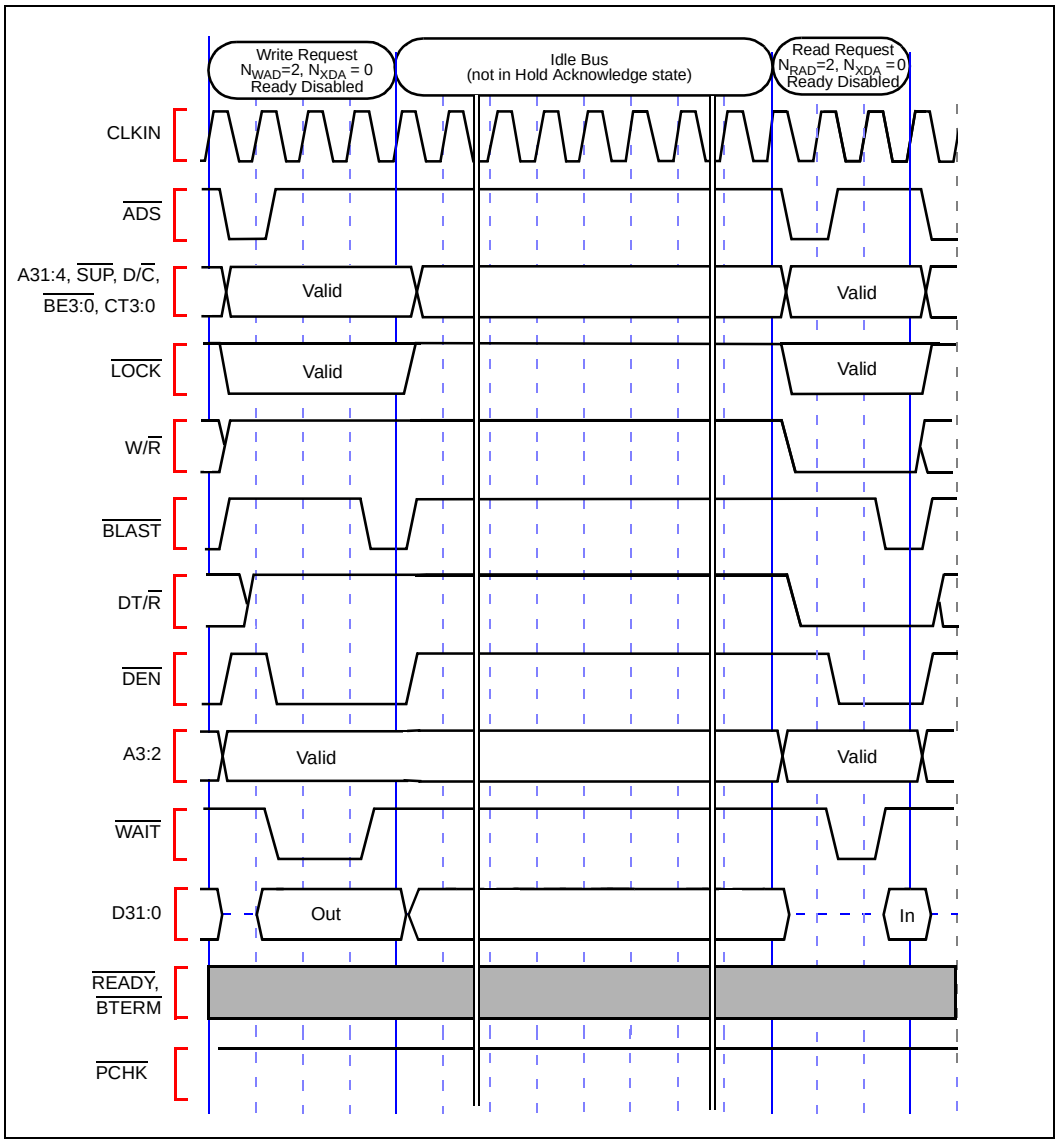
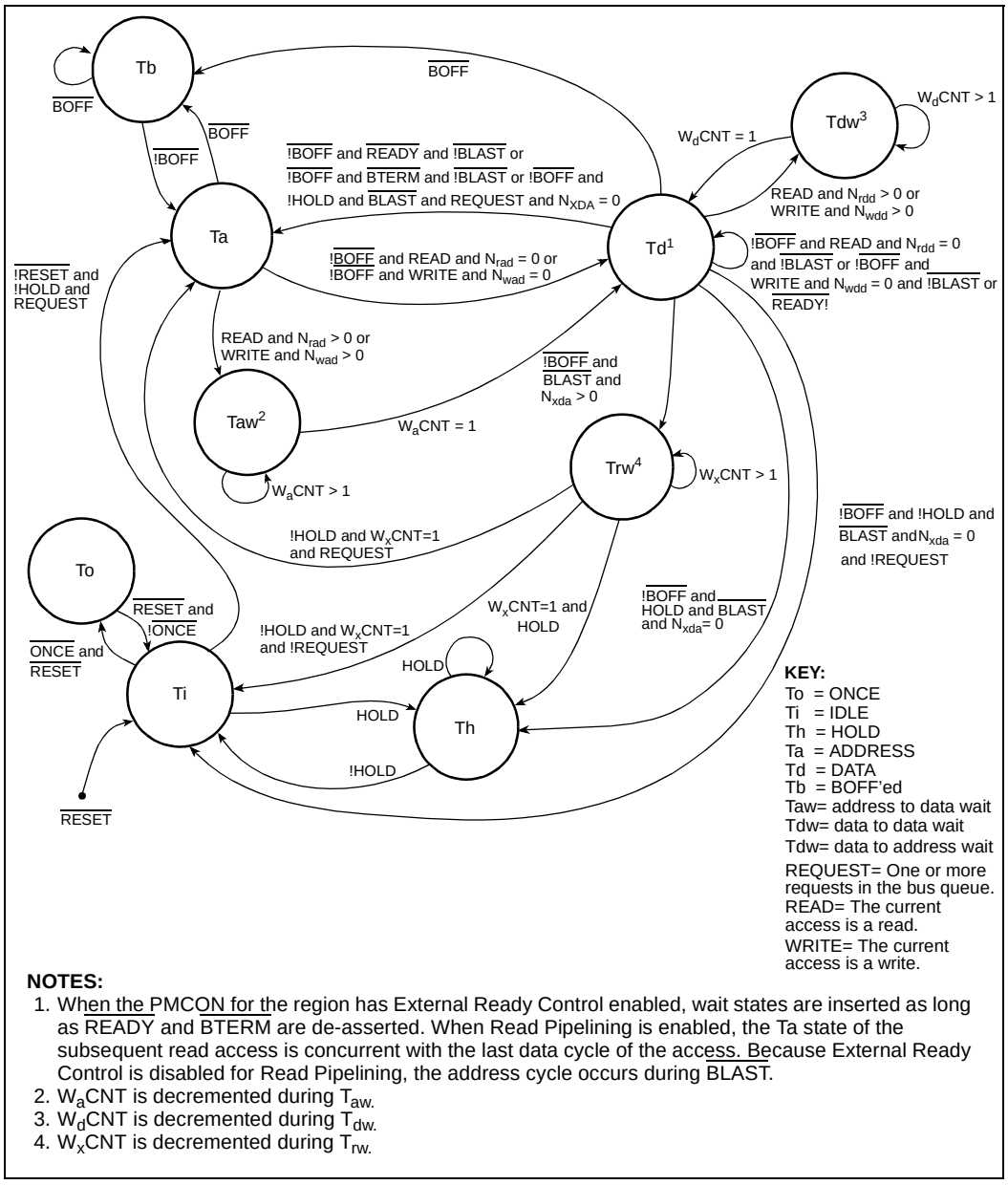


Figure 59. Bus States



NOTES:

1. When the PMCON for the region has External Ready Control enabled, wait states are inserted as long as READY and BTERM are de-asserted. When Read Pipelining is enabled, the Ta state of the subsequent read access is concurrent with the last data cycle of the access. Because External Ready Control is disabled for Read Pipelining, the address cycle occurs during BLAST.
2. W_aCNT is decremented during T_{aw}.
3. W_dCNT is decremented during T_{dw}.
4. W_xCNT is decremented during T_{rw}.

5.1 80960Hx Boundary Scan Chain

Table 26. 80960Hx Boundary Scan Chain (Sheet 1 of 4)

#	Boundary Scan Cell	Cell Type	Comment
	DP3	Bidirectional	
	DP2	Bidirectional	
	DP0	Bidirectional	
	DP1	Bidirectional	
	STEST	Input	
	FAILBAR	Output	
	Enable for FAILBAR, BSTALL and BREQ	Control	
	ONCEBAR	Input	
	BOFFBAR	Input	
	D0	Bidirectional	
	D1	Bidirectional	
	D2	Bidirectional	
	D3	Bidirectional	
	D4	Bidirectional	
	D5	Bidirectional	
	D6	Bidirectional	
	D7	Bidirectional	
	Enable for DP(3:0) and D(31:0)	Control	
	D8	Bidirectional	
	D9	Bidirectional	
	D10	Bidirectional	
	D11	Bidirectional	
	D12	Bidirectional	
	D13	Bidirectional	
	D14	Bidirectional	
	D15	Bidirectional	
	D16	Bidirectional	
	D17	Bidirectional	
	D18	Bidirectional	
	D19	Bidirectional	
	D20	Bidirectional	

NOTES:
1. Cell#1 connects to TDO and cell #112 connects to TDI.
2. All outputs are tri-state.
3. In output and bidirectional signals, a logical 1 on the enable signal enables the output. A logical 0 tri-states the output.

Table 26. 80960Hx Boundary Scan Chain (Sheet 2 of 4)

#	Boundary Scan Cell	Cell Type	Comment
	D21	Bidirectional	
	D22	Bidirectional	
	D23	Bidirectional	
	D24	Bidirectional	
	D25	Bidirectional	
	D26	Bidirectional	
	D27	Bidirectional	
	D28	Bidirectional	
	D29	Bidirectional	
	D30	Bidirectional	
	D31	Bidirectional	
	BTERMBAR	Input	
	RDYBAR	Input	Appears as READYBAR in BSDL file.
	HOLD	Input	
	HOLDA	Output	
	Enable for HOLDA control	Control	
	ADSBAR	Output	
	BE3BAR	Output	Appears as BEBAR(3:0) in BSDL file.
	BE2BAR	Output	
	BE1BAR	Output	
	BE0BAR	Output	
	BLASTBAR	Output	
	DENBAR	Output	
	WRRDBAR	Output	Appears as WRBAR in BSDL file.
	DTRBAR	Output	
	Enable for DTRBAR	Control	
	WAITBAR	Output	
	BSTALL	Output	
	DATACODBAR	Output	Appears as DCBAR in BSDL file.
	USERSUPBAR	Output	Appears as SUPBAR in BSDL file.
	Enable for ADSBAR, BEBAR, BLASTBAR, DENBAR, WRRDBAR, WAITBAR, DCBAR, SUPBAR and LOCKBAR,	Control	

NOTES:
1. Cell#1 connects to TDO and cell #112 connects to TDI.
2. All outputs are tri-state.
3. In output and bidirectional signals, a logical 1 on the enable signal enables the output. A logical 0 tri-states the output.

Table 26. 80960Hx Boundary Scan Chain (Sheet 3 of 4)

#	Boundary Scan Cell	Cell Type	Comment
	LOCKBAR	Output	
	BREQ	Output	
	A31	Output	
	A30	Output	
	A29	Output	
	A28	Output	
	A27	Output	
	A26	Output	
	A25	Output	
	A24	Output	
	A23	Output	
	A22	Output	
	A21	Output	
	A20	Output	
	A19	Output	
	A18	Output	
	A17	Output	
	A16	Output	
	Enable for A(31:0) and CT(3:0)	Control	
	A15	Output	
	A14	Output	
	A13	Output	
	A12	Output	
	A11	Output	
	A10	Output	
	A9	Output	
	A8	Output	
	A7	Output	
	A6	Output	
	A5	Output	
	A4	Output	
	A3	Output	
	A2	Output	
	NMIBAR	Input	

NOTES:
1. Cell#1 connects to TDO and cell #112 connects to TDI.
2. All outputs are tri-state.
3. In output and bidirectional signals, a logical 1 on the enable signal enables the output. A logical 0 tri-states the output.

Table 26. 80960Hx Boundary Scan Chain (Sheet 4 of 4)

#	Boundary Scan Cell	Cell Type	Comment
	XINT7BAR	Input	Appears as XINTBAR(7:0) in BSDL file.
	XINT6BAR	Input	
	XINT5BAR	Input	
	XINT4BAR	Input	
	XINT3BAR	Input	
	XINT2BAR	Input	
	XINT1BAR	Input	
	XINT0BAR	Input	
	RESETBAR	Input	
	CLKIN	Input	
	CT3	Output	Appears as CT(3:0) in BSDL file.
	CT2	Output	
	CT1	Output	
	CT0	Output	
	PCHK	Output	Appears as PCHKBAR in BSDL file.
	PCHK enable	Control	

NOTES:
1. Cell#1 connects to TDO and cell #112 connects to TDI.
2. All outputs are tri-state.
3. In output and bidirectional signals, a logical 1 on the enable signal enables the output. A logical 0 tri-states the output.

5.2 Boundary Scan Description Language Example

The Boundary-Scan Description Language (BSDL) for PGA Package Example, as shown in Example 1, meets the de-facto standard means of describing essential features of ANSI/IEEE 1149.1-1993 compliant devices.

The Boundary-Scan Description Language (BSDL) for PQ2 Package Example is shown in Example 2 on page 96.

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 1 of 8)

```
-- Copyright Intel Corp. 1995

- - ************************************************************************

- - Intel Corporation makes no warranty for the use of its products and assumes no
responsibility for any errors which may appear in this document nor does it make
a commitment to update the information contained herein.

- - ************************************************************************

- - Boundary-Scan Description Language (BSDL Version 0.0) is a de-facto standard
means of describing essential features of ANSI/IEEE 1149.1-1990 compliant
devices. This language is under consideration by the IEEE for formal inclusion
within a supplement to the 1149.1-1990 standard. The generation of the supplement
entails an extensive IEEE review and a formal acceptance balloting procedure
which may change the resultant form of the language. Be aware that this process
may extend well into 1993, and at this time the IEEE does not endorse or hold an
opinion on the language.

- - ************************************************************************

--

-- i960(R) Processor BSDL Model
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 2 of 8)

```
-- Project code HA
-- File **NOT** verified electrically
-- ------------------------------------------------
-- Rev 0.7     18 Dec  1995  Updated for A-1 stepping.
-- Rev 0.6     08 Dec  1994
-- Rev 0.5     21 Nov  1994
-- Rev 0.4     31 Oct  1994
-- Rev 0.3     26 July 1994
-- Rev 0.2     22 June 1994
-- Rev 0.1     16 Mar  1994
-- Rev 0.0     30 Aug  1993
entity Ha_Processor is
   generic(PHYSICAL_PIN_MAP : string:= “PGA”);

   port (A         : out     bit_vector(2 to 31);
         ADSBAR    : out     bit;
         BEBAR     : out     bit_vector(0 to 3);
         BLASTBAR  : out     bit;
         BOFFBAR   : in      bit;
         BREQ      : out     bit;
         BSTALL    : out     bit;
         BTERMBAR  : in      bit;
         CT        : out     bit_vector(0 to 3);
         CLKIN     : in      bit;
         D         : inout   bit_vector(0 to 31);
         DENBAR    : out     bit;
         DP        : inout   bit_vector(0 to 3);
         DTRBAR    : out     bit;
         DCBAR     : out     bit;
         FAILBAR   : out     bit;
         HOLD      : in      bit;
         HOLDA     : out     bit;
         LOCKBAR   : out     bit;
         NMIBAR    : in      bit;
         ONCEBAR   : in      bit;
         PCHKBAR   : out     bit;
         READYBAR  : in      bit;
         RESETBAR  : in      bit;
         STEST     : in      bit;
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 3 of 8)

```
         SUPBAR    : out     bit;
         TCK       : in      bit;
         TDI       : in      bit;
         TDO       : out     bit;
         TMS       : in      bit;
         TRST      : in      bit;
         WAITBAR   : out     bit;
         WRBAR     : out     bit;
         XINTBAR   : in      bit_vector(0 to 7);
         FIVEVREF  : linkage bit;
         VCCPLL    : linkage bit;
         VOLTDET   : out     bit;
         VCC1      : linkage bit_vector(0 to 23);
         VCC2      : linkage bit_vector(0 to 20);
         VSS1      : linkage bit_vector(0 to 25);
         VSS2      : linkage bit_vector(0 to 22);
         NC        : linkage bit_vector(0 to 4)

   );

use STD_1149_1_1990.all;
use i960ha_a.all;


attribute PIN_MAP of Ha_Processor : entity is PHYSICAL_PIN_MAP;
constant PGA:PIN_MAP_STRING :=
           “A               : (D16, D17, E16, E17, F17, G16, G17, H17, J17,”&
           “                   K17, L17, L16, M17, N17, N16, P17, Q17, P16,”&
           “                   P15, Q16, R17, R16, Q15, S17, R15, S16, Q14, ”&
           “            R14, Q13, S15),
           “ADSBAR    : R06,”&
           “BEBAR     : (R09, S07, S06, S05),”&
           “BLASTBAR  : S08,”&
           “BOFFBAR   : B01,”&
           “BREQ      : R13,”&
           “BSTALL    : R12,”&
           “BTERMBAR  : R04,”&
           “CT        : (A11, A12, A13, A14),”&
           “CLKIN     : C13,”&
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 4 of 8)

```
            “D               : (E03, C02, D02, C01, E02, D01, F02, E01, F01,”&
            “                    G01, H02, H01, J01, K01, L02, L01, M01, N01,”&
            “                    N02, P01, P02, Q01, P03, Q02, R01, S01, Q03,”&
            “             R02, Q04, S02, Q05, R03),”&
            “DENBAR    : S09,”&
            “DP        : (A03, B03, A04, B04),”&
            “DTRBAR    : S11,”&
            “DCBAR     : S13,”&
            “FAILBAR   : A02,”&
            “HOLD      : R05,”&
            “HOLDA     : S04,”&
            “LOCKBAR   : S14,”&
            “NMIBAR    : D15,”&
            “ONCEBAR   : C03,”&
            “PCHKBAR   : B08,”&
            “READYBAR  : S03,”&
            “RESETBAR  : A16,”&
            “STEST     : B02,”&
            “SUPBAR    : Q12,”&
            “TCK       : B05,”&
            “TDI       : A07,”&
            “TDO       : A08,”&
            “TMS       : B06,”&
            “TRST      : A06,”&
            “WAITBAR   : S12,”&
            “WRBAR     : S10,”&
            “ XINTBAR   : (B15, A15, A17, B16, C15, B17, C16, C17),”&
            “FIVEVREF  : C05,”&
            “VOLTDET   : A05,”&
            “VCCPLL    : B10,”&
            “ VCC1      : (M02, K02, J02, G02, N03, F03, C06, B07, B09, B11,”&
            “               B12, C14, E15, F16, H16, J16, K16, M16, N15, Q06,”&
            “               R07, R08, R10, R11),”&
            “ VSS1      : (G03, H03, J03, K03, L03, M03, C07, C08, C09, C10,”&
            “               C11, C12, Q07, Q08, Q09, Q10, Q11, F15, G15, H15,”&
            “               J15, K15, L15, M15, A01, C04),”&
            “NC        : (A09, A10, B13, B14, D03)”;
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 5 of 8)

```
  attribute Tap_Scan_In    of  TDI   : signal is true;
  attribute Tap_Scan_Mode  of  TMS   : signal is true;
  attribute Tap_Scan_Out   of  TDO   : signal is true;
  attribute Tap_Scan_Reset of  TRST  : signal is true;
  attribute Tap_Scan_Clock of  TCK   : signal is (66.0e6, BOTH);

  attribute Instruction_Length of Ha_Processor: entity is 4;

  attribute Instruction_Opcode of Ha_Processor: entity is

        "BYPASS     (1111)," &
        "EXTEST     (0000)," &
        "SAMPLE     (0001)," &
        "IDCODE     (0010)," &
        "RUBIST     (0111)," &
        "CLAMP      (0100)," &
        "HIGHZ      (1000)," &
        "Reserved   (1011, 1100)";

  attribute Instruction_Capture of Ha_Processor: entity is "0001";

  attribute Instruction_Private of Ha_Processor: entity is "Reserved" ;

  attribute Idcode_Register of Ha_Processor: entity is
        "0010"              &  --version,
        "1000100001000000"  &  --part number
        "00000001001"       &  --manufacturers identity
        "1";                   --required by the standard
  attribute Register_Access of Ha_Processor: entity is
        "Runbist[32]     (RUBIST)," &
        "Bypass          (CLAMP, HIGHZ)";

{************************************************************************}
{  The first cell, cell 0, is closest to TDO                             }
{  BC_1:Control, Output3  CBSC_1:Bidir  BC_4: Input, Clock               }
{************************************************************************}
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 6 of 8)

```
attribute Boundary_Cells of Ha_Processor: entity is “BC_4, BC_1, CBSC_1”;
attribute Boundary_Length of Ha_Processor: entity is 112;
attribute Boundary_Register of Ha_Processor: entity is

   “0  (CBSC_1,  DP(3),       bidir,   X,   17, 1,  Z),”   &
   “1  (CBSC_1,  DP(2),       bidir,   X,   17, 1,  Z),”   &
   “2  (CBSC_1,  DP(0),       bidir,   X,   17, 1,  Z),”   &
   “3  (CBSC_1,  DP(1),       bidir,   X,   17, 1,  Z),”   &
   “4  (BC_4,    STEST,       input,   X),”   &
   “5  (BC_1,    FAILBAR,     output3, X,   6,  1,  Z),”   &
   “6  (BC_1,    *,           control, 1),”   &
   “7  (BC_4,    ONCEBAR,     input,   X),”   &
   “8  (BC_4,    BOFFBAR,     input,   X),”   &
   “9  (CBSC_1,  D(0),        bidir,   X,  17,  1,  Z),”   &
   “10 (CBSC_1,  D(1),        bidir,   X,  17,  1,  Z),”   &
   “11 (CBSC_1,  D(2),        bidir,   X,  17,  1,  Z),”   &
   “12 (CBSC_1,  D(3),        bidir,   X,  17,  1,  Z),”   &
   “13 (CBSC_1,  D(4),        bidir,   X,  17,  1,  Z),”   &
   “14 (CBSC_1,  D(5),        bidir,   X,  17,  1,  Z),”   &
   “15 (CBSC_1,  D(6),        bidir,   X,  17,  1,  Z),”   &
   “16 (CBSC_1,  D(7),        bidir,   X,  17,  1,  Z),”   &
   “17 (BC_1,    *,           control, 1),”   &
   “18 (CBSC_1,  D(8),        bidir,   X,  17,  1,  Z),”   &
   “19 (CBSC_1,  D(9),        bidir,   X,  17,  1,  Z),”   &
   “20 (CBSC_1,  D(10),       bidir,   X,  17,  1,  Z),”   &
   “21 (CBSC_1,  D(11),       bidir,   X,  17,  1,  Z),”   &
   “22 (CBSC_1,  D(12),       bidir,   X,  17,  1,  Z),”   &
   “23 (CBSC_1,  D(13),       bidir,   X,  17,  1,  Z),”   &
   “24 (CBSC_1,  D(14),       bidir,   X,  17,  1,  Z),”   &
   “25 (CBSC_1,  D(15),       bidir,   X,  17,  1,  Z),”   &
   “26 (CBSC_1,  D(16),       bidir,   X,  17,  1,  Z),”   &
   “27 (CBSC_1,  D(17),       bidir,   X,  17,  1,  Z),”   &
   “28 (CBSC_1,  D(18),       bidir,   X,  17,  1,  Z),”   &
   “29 (CBSC_1,  D(19),       bidir,   X,  17,  1,  Z),”   &
   “30 (CBSC_1,  D(20),       bidir,   X,  17,  1,  Z),”   &
   “31 (CBSC_1,  D(21),       bidir,   X,  17,  1,  Z),”   &
   “32 (CBSC_1,  D(22),       bidir,   X,  17,  1,  Z),”   &
   “33 (CBSC_1,  D(23),       bidir,   X,  17,  1,  Z),”   &
   “34 (CBSC_1,  D(24),       bidir,   X,  17, 1,  Z),”   &
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 7 of 8)

```
      “35 (CBSC_1,  D(25),       bidir,   X,  17, 1,  Z),”   &
      “36 (CBSC_1,  D(26),       bidir,   X,  17, 1,  Z),”   &
      “37 (CBSC_1,  D(27),       bidir,   X,  17, 1,  Z),”   &
      “38 (CBSC_1,  D(28),       bidir,   X,  17, 1,  Z),”   &
      “39 (CBSC_1,  D(29),       bidir,   X,  17, 1,  Z),”   &
      “40 (CBSC_1,  D(30),       bidir,   X,  17, 1,  Z),”   &
      “41 (CBSC_1,  D(31),       bidir,   X,  17, 1,  Z),”   &
      “42 (BC_4,    BTERMBAR,    input,   X),”   &
      “43 (BC_4,    READYBAR,    input,   X),”   &
      “44 (BC_4,    HOLD,        input,   X),”   &
      “45 (BC_1,    HOLDA,       output3, X,  46, 1,  Z),”   &
      “46 (BC_1,    *,           control, 1),”   &
      “47 (BC_1,    ADSBAR,      output3, X,  61, 1,  Z),”   &
      “48 (BC_1,    BEBAR(3),    output3, X,  61, 1,  Z),”   &
      “49 (BC_1,    BEBAR(2),    output3, X,  61, 1,  Z),”   &
      “50 (BC_1,    BEBAR(1),    output3, X,  61, 1,  Z),”   &
      “51 (BC_1,    BEBAR(0),    output3, X,  61, 1,  Z),”   &
      “52 (BC_1,    BLASTBAR,    output3, X,  61, 1,  Z),”   &
      “53 (BC_1,    DENBAR,      output3, X,  61, 1,  Z),”   &
      “54 (BC_1,    WRBAR,       output3, X,  61, 1,  Z),”   &
      “55 (BC_1,    DTRBAR,      output3, X,  56, 1,  Z),”   &
      “56 (BC_1,    *,           control, 1),”   &
      “57 (BC_1,    WAITBAR,     output3, X,  61, 1,  Z),”   &
      “58 (BC_1,    BSTALL,      output3, X,  6,  1,  Z),”   &
      “59 (BC_1,    DCBAR,       output3, X,  61, 1,  Z),”   &
      “60 (BC_1,    SUPBAR,      output3, X,  61, 1,  Z),”   &
      “61 (BC_1,    *,           control, 1),”   &
      “62 (BC_1,    LOCKBAR,     output3, X,  61, 1,  Z),”   &
      “63 (BC_1,    BREQ,        output3, X,  6,  1,  Z),”   &
      “64 (BC_1,    A(31),       output3, X,  80, 1,  Z),”   &
      “65 (BC_1,    A(30),       output3, X,  80, 1,  Z),”   &
      “66 (BC_1,    A(29),       output3, X,  80, 1,  Z),”   &
      “67 (BC_1,    A(28),       output3, X,  80, 1,  Z),”   &
      “68 (BC_1,    A(27),       output3, X,  80, 1,  Z),”   &
      “69 (BC_1,    A(26),       output3, X,  80, 1,  Z),”   &
      “70 (BC_1,    A(25),       output3, X,  80, 1,  Z),”   &
      “71 (BC_1,    A(24),       output3, X,  80, 1,  Z),”   &
      “72 (BC_1,  A(23),       output3,  X,  80, 1,  Z),”   &
      “73 (BC_1,  A(22),       output3,  X,  80, 1,  Z),”   &
```

Example 1. Boundary-Scan Description Language (BSDL) for PGA Package Example (Sheet 8 of 8)

```
      "74 (BC_1,  A(21),       output3,  X,  80, 1,  Z)," &
      "75 (BC_1,  A(20),       output3,  X,  80, 1,  Z)," &
      "76 (BC_1,  A(19),       output3,  X,  80, 1,  Z)," &
      "77 (BC_1,  A(18),       output3,  X,  80, 1,  Z)," &
      "78 (BC_1,  A(17),       output3,  X,  80, 1,  Z)," &
      "79 (BC_1,  A(16),       output3,  X,  80, 1,  Z)," &
      "80 (BC_1,  *,           control,  1)," &
      "81 (BC_1,  A(15),       output3,  X,  80, 1,  Z)," &
      "82 (BC_1,  A(14),       output3,  X,  80, 1,  Z)," &
      "83 (BC_1,  A(13),       output3,  X,  80, 1,  Z)," &
      "84 (BC_1,  A(12),       output3,  X,  80, 1,  Z)," &
      "85 (BC_1,  A(11),       output3,  X,  80, 1,  Z)," &
      "86 (BC_1,  A(10),       output3,  X,  80, 1,  Z)," &
      "87 (BC_1,  A(9),        output3,  X,  80, 1,  Z)," &
      "88 (BC_1,  A(8),        output3,  X,  80, 1,  Z)," &
      "89 (BC_1,  A(7),        output3,  X,  80, 1,  Z)," &
      "90 (BC_1,  A(6),        output3,  X,  80, 1,  Z)," &
      "91 (BC_1,  A(5),        output3,  X,  80, 1,  Z)," &
      "92 (BC_1,  A(4),        output3,  X,  80, 1,  Z)," &
      "93 (BC_1,  A(3),        output3,  X,  80, 1,  Z)," &
      "94 (BC_1,  A(2),        output3,  X,  80, 1,  Z)," &
      "95 (BC_4,  NMIBAR,      input,    X)," &
      "96 (BC_4,  XINTBAR(7),  input,    X)," &
      "97 (BC_4,  XINTBAR(6),  input,    X)," &
      "98 (BC_4,  XINTBAR(5),  input,    X)," &
      "99 (BC_4,  XINTBAR(4),  input,    X)," &
      "100(BC_4,  XINTBAR(3),  input,    X)," &
      "101(BC_4,  XINTBAR(2),  input,    X)," &
      "102(BC_4,  XINTBAR(1),  input,    X)," &
      "103(BC_4,  XINTBAR(0),  input,    X)," &
      "104(BC_4,  RESETBAR,    input,    X)," &
      "105(BC_4,  CLKIN,       input,    X)," &
      "106(BC_1,  CT(3),       output3,  X,  80, 1,  Z)," &
      "107(BC_1,  CT(2),       output3,  X,   80, 1, Z)," &
      "108(BC_1,  CT(1),       output3,  X,   80, 1, Z)," &
      "109(BC_1,  CT(0),       output3,  X,   80, 1, Z)," &
      "110(BC_1,  PCHKBAR,     output3,  X,  111, 1, Z)," &
      "111(BC_1,  *,           control,  1)";
end Ha_Processor;
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 1 of 8)

```
-- Copyright Intel Corporation 1995, 1996

-- *************************************************************************

-- Intel Corporation makes no warranty for the use of its products and assumes no
responsibility for any errors which may appear in this document nor does it make
a commitment to update the information contained herein.

-- *************************************************************************

-- Boundary-Scan Description Language (BSDL Version 0.0) is a de-facto

-- standard means of describing essential features of ANSI/IEEE 1149.1-1990
compliant devices. This language is under consideration by the IEEE for formal
inclusion within a supplement to the 1149.1-1990 standard. The generation of the
supplement entails an extensive IEEE review and a formal acceptance balloting
procedure which may change the resultant form of the language. Be aware that this
process may extend well into 1993, and at this time the IEEE does not endorse or
hold an opinion on the language.

-- i960(R) Processor BSDL Model
-- Project code HA
-- File **NOT** verified electrically
-- ----------------------------------------------
-- Rev 0.8      4 Apr  1996  Changed for PQ2 Package
-- Rev 0.7     18 Dec  1995  Updated for A-1 stepping.
-- Rev 0.6     08 Dec  1994
-- Rev 0.5     21 Nov  1994
-- Rev 0.4     31 Oct  1994
-- Rev 0.3     26 July 1994
-- Rev 0.2     22 June 1994
-- Rev 0.1     16 Mar  1994
-- Rev 0.0     30 Aug  1993
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 2 of 8)

```
entity Ha_Processor is
   generic(PHYSICAL_PIN_MAP : string:= “PQ2”);
   port (A          : out     bit_vector(2 to 31);
         ADSBAR     : out     bit;
         BEBAR      : out     bit_vector(0 to 3);
         BLASTBAR   : out     bit;
         BOFFBAR    : in      bit;
         BREQ       : out     bit;
         BSTALL     : out     bit;
         BTERMBAR   : in      bit;
         CT         : out     bit_vector(0 to 3);
         CLKIN      : in      bit;
         D          : inout   bit_vector(0 to 31);
         DENBAR     : out     bit;
         DP         : inout   bit_vector(0 to 3);
         DTRBAR     : out     bit;
         DCBAR      : out     bit;
         FAILBAR    : out     bit;
         HOLD       : in      bit;
         HOLDA      : out     bit;
         LOCKBAR    : out     bit;
         NMIBAR     : in      bit;
         ONCEBAR    : in      bit;
         PCHKBAR    : out     bit;
         READYBAR   : in      bit;
         RESETBAR   : in      bit;
         STEST      : in      bit;
         SUPBAR     : out     bit;
         TCK        : in      bit;
         TDI        : in      bit;
         TDO        : out     bit;
         TMS        : in      bit;
         TRST       : in      bit;
         WAITBAR    : out     bit;
         WRBAR      : out     bit;
         XINTBAR    : in      bit_vector(0 to 7);
         FIVEVREF   : linkage bit;
         VCCPLL     : linkage bit;
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 3 of 8)

```
        VCC1      : linkage bit_vector(0 to 23);
        VCC2      : linkage bit_vector(0 to 23);
        VSS1      : linkage bit_vector(0 to 23);
        VSS2      : linkage bit_vector(0 to 23)

);

use STD_1149_1_1990.all;
use i960ha_a.all;


attribute PIN_MAP of Ha_Processor : entity is PHYSICAL_PIN_MAP;
 constant PQ2:PIN_MAP_STRING :=

      “A          : (151, 150, 147, 146, 145, 144, 141, 140, 139, 138,”&
      “               135, 134, 133, 132, 127, 126, 125, 124, 121, 120,”&
      “               119, 118, 113, 112, 111, 110, 107, 106, 105, 104),”&
      “ADSBAR     : 77,”&
      “BEBAR      : (83, 82, 79, 78),”&
      “BLASTBAR   : 84,”&
      “BOFFBAR    : 10,”&
      “BREQ       : 100,”&
      “BSTALL     : 91,”&
      “BTERMBAR   : 67,”&
      “CT         : (183, 182, 181, 180),”&
      “CLKIN      : 175,”&
      “D          : (12, 13, 14, 15, 20, 21, 22, 23, 26, 27, 28, 29,”&
      “              34, 35, 36, 37, 39, 40, 41, 42, 45, 50, 51, 52,”&
      “              54, 55, 56, 57, 61, 62, 63, 64),”&
      “DENBAR     : 85,”&
      “DP         : (206, 207, 203, 202),”&
      “DTRBAR     : 89,”&
      “DCBAR      : 96,”&
      “FAILBAR    : 5,”&
      “HOLD       : 69,”&
      “HOLDA      : 72,”&
      “LOCKBAR    : 99,”&
      “NMIBAR     : 159,”&
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 4 of 8)

```
        "ONCEBAR   : 6,"&
        "PCHKBAR   : 189,"&
        "READYBAR  : 68,"&
        "RESETBAR  : 174,"&
        "STEST     : 208,"&
        "SUPBAR    : 97,"&
        "TCK       : 194,"&
        "TDI       : 191,"&
        "TDO       : 188,"&
        "TMS       : 192,"&
        "TRST      : 193,"&
        "WAITBAR   : 90,"&
        "WRBAR     : 88,"&
        "XINTBAR   : (169, 168, 167, 166, 163, 162, 161, 160),"&
        "FIVEVREF  : 197,"&
        "VCCPLL    : 177,"&
        "VCC1      : (1, 4, 9, 11, 17, 19, 25, 31, 33, 38, 44, 46,"&
        "             49, 59, 60, 66, 71, 74, 76, 81, 87, 92, 95, 101),"&
        "VCC2      : (102, 109, 115, 117, 123, 128, 131, 137, 143, 149,"&
        "             153, 154, 158, 165, 171, 173, 176, 179, 185, 187,"&
        "             196, 199, 201, 204),"&
        "VSS1      : (2, 3, 7, 8, 16, 18, 24, 30, 32, 43, 47, 48,"&
        "             53, 58, 65, 70, 73, 75, 80, 86, 93, 94, 98, 103),"&
        "VSS2      : (108, 114, 116, 122, 129, 130, 136, 142, 148, 152,"&
        "             155, 156, 157, 164, 170, 172, 178, 184, 186, 190,"&
        "             195, 198, 200, 205)";


  attribute Tap_Scan_In    of  TDI   : signal is true;
  attribute Tap_Scan_Mode  of  TMS   : signal is true;
  attribute Tap_Scan_Out   of  TDO   : signal is true;
  attribute Tap_Scan_Reset of  TRST  : signal is true;
  attribute Tap_Scan_Clock of  TCK   : signal is (66.0e6, BOTH);

  attribute Instruction_Length of Ha_Processor: entity is 4;

  attribute Instruction_Opcode of Ha_Processor: entity is
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 5 of 8)

```
        “BYPASS     (1111),” &
        “EXTEST     (0000),” &
        “SAMPLE     (0001),” &
        “IDCODE     (0010),” &
        “RUBIST     (0111),” &
        “CLAMP      (0100),” &
        “HIGHZ      (1000),” &
        “Reserved   (1011, 1100)”;

   attribute Instruction_Capture of Ha_Processor: entity is “0001”;

   attribute Instruction_Private of Ha_Processor: entity is “Reserved” ;

   attribute Idcode_Register of Ha_Processor: entity is
        “0001” &  version,
        “1000100001000000”     &  part number
        “00000001001”&  manufacturers identity
        “1”;    required by the standard

   attribute Register_Access of Ha_Processor: entity is
        “Runbist[32]     (RUBIST),” &
        “Bypass          (CLAMP, HIGHZ)”;
*****************************************************************************
{  The first cell, cell 0, is closest to TDO                          }
{  BC_1:Control, Output3  CBSC_1:Bidir  BC_4: Input, Clock            }
*****************************************************************************

   attribute Boundary_Cells of Ha_Processor: entity is “BC_4, BC_1, CBSC_1”;
   attribute Boundary_Length of Ha_Processor: entity is 112;
   attribute Boundary_Register of Ha_Processor: entity is

        “0  (CBSC_1,  DP(3),       bidir,   X,   17, 1,  Z),”   &
        “1  (CBSC_1,  DP(2),       bidir,   X,   17, 1,  Z),”   &
        “2  (CBSC_1,  DP(0),       bidir,   X,   17, 1,  Z),”   &
        “3  (CBSC_1,  DP(1),       bidir,   X,   17, 1,  Z),”   &
        “4  (BC_4,  STEST,         input,   X),”   &
        “5  (BC_1,  FAILBAR,       output3, X,   6,  1,  Z),”   &
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 6 of 8)

```
"6  (BC_1,    *,          control, 1),"   &
"7  (BC_4,    ONCEBAR,    input,   X),"   &
"8  (BC_4,    BOFFBAR,    input,   X),"   &
"9  (CBSC_1,  D(0),       bidir,   X,  17,  1,  Z),"   &
"10 (CBSC_1,  D(1),       bidir,   X,  17,  1,  Z),"   &
"11 (CBSC_1,  D(2),       bidir,   X,  17,  1,  Z),"   &
"12 (CBSC_1,  D(3),       bidir,   X,  17,  1,  Z),"   &
"13 (CBSC_1,  D(4),       bidir,   X,  17,  1,  Z),"   &
"14 (CBSC_1,  D(5),       bidir,   X,  17,  1,  Z),"   &
"15 (CBSC_1,  D(6),       bidir,   X,  17,  1,  Z),"   &
"16 (CBSC_1,  D(7),       bidir,   X,  17,  1,  Z),"   &
"17 (BC_1,    *,          control, 1),"   &
"18 (CBSC_1,  D(8),       bidir,   X,  17,  1,  Z),"   &
"19 (CBSC_1,  D(9),       bidir,   X,  17,  1,  Z),"   &
"20 (CBSC_1,  D(10),      bidir,   X,  17,  1,  Z),"   &
"21 (CBSC_1,  D(11),      bidir,   X,  17,  1,  Z),"   &
"22 (CBSC_1,  D(12),      bidir,   X,  17,  1,  Z),"   &
"23 (CBSC_1,  D(13),      bidir,   X,  17,  1,  Z),"   &
"24 (CBSC_1,  D(14),      bidir,   X,  17,  1,  Z),"   &
"25 (CBSC_1,  D(15),      bidir,   X,  17,  1,  Z),"   &
"26 (CBSC_1,  D(16),      bidir,   X,  17,  1,  Z),"   &
"27 (CBSC_1,  D(17),      bidir,   X,  17,  1,  Z),"   &
"28 (CBSC_1,  D(18),      bidir,   X,  17,  1,  Z),"   &
"29 (CBSC_1,  D(19),      bidir,   X,  17,  1,  Z),"   &
"30 (CBSC_1,  D(20),      bidir,   X,  17,  1,  Z),"   &
"31 (CBSC_1,  D(21),      bidir,   X,  17,  1,  Z),"   &
"32 (CBSC_1,  D(22),      bidir,   X,  17,  1,  Z),"   &
"33 (CBSC_1,  D(23),      bidir,   X,  17,  1,  Z),"   &
"34 (CBSC_1,  D(24),      bidir,   X,  17,  1,  Z),"   &
"35 (CBSC_1,  D(25),      bidir,   X,  17,  1,  Z),"   &
"36 (CBSC_1,  D(26),      bidir,   X,  17,  1,  Z),"   &
"37 (CBSC_1,  D(27),      bidir,   X,  17,  1,  Z),"   &
"38 (CBSC_1,  D(28),      bidir,   X,  17,  1,  Z),"   &
"39 (CBSC_1,  D(29),      bidir,   X,  17,  1,  Z),"   &
"40 (CBSC_1,  D(30),      bidir,   X,  17,  1,  Z),"   &
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 7 of 8)

```
        "41 (CBSC_1,  D(31),     bidir,    X,  17, 1,  Z)," &
        "42 (BC_4,  BTERMBAR,    input,    X)," &
        "43 (BC_4,  READYBAR,    input,    X)," &
        "44 (BC_4,  HOLD,        input,    X)," &
        "45 (BC_1,  HOLDA,       output3,  X,  46, 1,  Z)," &
        "46 (BC_1,  *,           control,  1)," &
        "47 (BC_1,  ADSBAR,      output3,  X,  61, 1,  Z)," &
        "48 (BC_1,  BEBAR(3),    output3,  X,  61, 1,  Z)," &
        "49 (BC_1,  BEBAR(2),    output3,  X,  61, 1,  Z)," &
        "50 (BC_1,  BEBAR(1),    output3,  X,  61, 1,  Z)," &
        "51 (BC_1,  BEBAR(0),    output3,  X,  61, 1,  Z)," &
        "52 (BC_1,  BLASTBAR,    output3,  X,  61, 1,  Z)," &
        "53 (BC_1,  DENBAR,      output3,  X,  61, 1,  Z)," &
        "54 (BC_1,  WRBAR,       output3,  X,  61, 1,  Z)," &
        "55 (BC_1,  DTRBAR,      output3,  X,  56, 1,  Z)," &
        "56 (BC_1,  *,           control,  1)," &
        "57 (BC_1,  WAITBAR,     output3,  X,  61, 1,  Z)," &
        "58 (BC_1,  BSTALL,      output3,  X,  6,  1,  Z)," &
        "59 (BC_1,  DCBAR,       output3,  X,  61, 1,  Z)," &
        "60 (BC_1,  SUPBAR,      output3,  X,  61, 1,  Z)," &
        "61 (BC_1,  *,           control,  1)," &
        "62 (BC_1,  LOCKBAR,     output3,  X,  61, 1,  Z)," &
        "63 (BC_1,  BREQ,        output3,  X,  6,  1,  Z)," &
        "64 (BC_1,  A(31),       output3,  X,  80, 1,  Z)," &
        "65 (BC_1,  A(30),       output3,  X,  80, 1,  Z)," &
        "66 (BC_1,  A(29),       output3,  X,  80, 1,  Z)," &
        "67 (BC_1,  A(28),       output3,  X,  80, 1,  Z)," &
        "68 (BC_1,  A(27),       output3,  X,  80, 1,  Z)," &
        "69 (BC_1,  A(26),       output3,  X,  80, 1,  Z)," &
        "70 (BC_1,  A(25),       output3,  X,  80, 1,  Z)," &
        "71 (BC_1,  A(24),       output3,  X,  80, 1,  Z)," &
        "72 (BC_1,  A(23),       output3,  X,  80, 1,  Z)," &
        "73 (BC_1,  A(22),       output3,  X,  80, 1,  Z)," &
        "74 (BC_1,  A(21),       output3,  X,  80, 1,  Z)," &
        "75 (BC_1,  A(20),       output3,  X,  80, 1,  Z)," &
```

Example 2. Boundary-Scan Description Language (BSDL) for PQ2 Package Example (Sheet 8 of 8)

```
          "76 (BC_1,  A(19),       output3,  X,  80, 1,  Z)," &
          "77 (BC_1,  A(18),       output3,  X,  80, 1,  Z)," &
          "78 (BC_1,  A(17),       output3,  X,  80, 1,  Z)," &
          "79 (BC_1,  A(16),       output3,  X,  80, 1,  Z)," &
          "80 (BC_1,  *,           control,  1)," &
          "81 (BC_1,  A(15),       output3,  X,  80, 1,  Z)," &
          "82 (BC_1,  A(14),       output3,  X,  80, 1,  Z)," &
          "83 (BC_1,  A(13),       output3,  X,  80, 1,  Z)," &
          "84 (BC_1,  A(12),       output3,  X,  80, 1,  Z)," &
          "85 (BC_1,  A(11),       output3,  X,  80, 1,  Z)," &
          "86 (BC_1,  A(10),       output3,  X,  80, 1,  Z)," &
          "87 (BC_1,  A(9),        output3,  X,  80, 1,  Z)," &
          "88 (BC_1,  A(8),        output3,  X,  80, 1,  Z)," &
          "89 (BC_1,  A(7),        output3,  X,  80, 1,  Z)," &
          "90 (BC_1,  A(6),        output3,  X,  80, 1,  Z)," &
          "91 (BC_1,  A(5),        output3,  X,  80, 1,  Z)," &
          "92 (BC_1,  A(4),        output3,  X,  80, 1,  Z)," &
          "93 (BC_1,  A(3),        output3,  X,  80, 1,  Z)," &
          "94 (BC_1,  A(2),        output3,  X,  80, 1,  Z)," &
          "95 (BC_4,  NMIBAR,      input,    X)," &
          "96 (BC_4,  XINTBAR(7),  input,    X)," &
          "97 (BC_4,  XINTBAR(6),  input,    X)," &
          "98 (BC_4,  XINTBAR(5),  input,    X)," &
          "99 (BC_4,  XINTBAR(4),  input,    X)," &
          "100(BC_4,  XINTBAR(3),  input,    X)," &
          "101(BC_4,  XINTBAR(2),  input,    X)," &
          "102(BC_4,  XINTBAR(1),  input,    X)," &
          "103(BC_4,  XINTBAR(0),  input,    X)," &
          "104(BC_4,  RESETBAR,    input,    X)," &
          "105(BC_4,  CLKIN,       input,    X)," &
          "106(BC_1,  CT(3),       output3,  X,  80, 1,  Z)," &
          "107(BC_1,  CT(2),       output3,  X,  80, 1,  Z)," &
          "108(BC_1,  CT(1),       output3,  X,  80, 1,  Z)," &
          "109(BC_1,  CT(0),       output3,  X,  80, 1,  Z)," &
          "110(BC_1,  PCHKBAR,     output3,  X,  111,1,  Z)," &
          "111(BC_1,  *,           control,  1)";
end Ha_Processor;
```

This page intentionally left blank.