Intel® IXP45X and Intel® IXP46X Product Line of Network Processors

Datasheet

Product Features

For a complete list of product features, see "Product Features" on page 9. This document describes in full the features of the silicon. Some of these features require enabling software supplied by Intel. Please refer to the *Intel® IXP400 Software Programmer's Guide* for information on which features are enabled at this time.

These features do *not* require enabling software

- Intel XScale® Core — Up to 667 MHz
- PCI v. 2.2 33/66 MHz (Host/Option)
- USB 1.1 Device Controller
- USB 2.0 Host Controller
- DDRI SDRAM Interface
- Master/Target Capable Expansion bus
- Two UARTs
- Internal Bus Performance Monitoring Unit
- 16 GPIO
- Four Internal Timers
- Synchronous Serial Protocol (SSP) Port
- I^2C Interface
- Spread Spectrum clocking for Reduced EMI
- Packaging
  - —544-Pin PBGA
  - —Commercial/Extended Temperature
  - —Lead-Free Support

These features require enabling software. For information on which features are enabled at this time, see the *Intel® IXP400 Software Programmer's Guide*.

- Cryptography Unit (Random Number Generator and Exponentiation Unit)
- Encryption/Authentication (AES/ AES-CCM/3DES/DES/SHA-1/SHA-256/ SHA-384/SHA-512/MD-5)
- Two High-Speed, Serial Interfaces
- Three Network Processor Engines
- Up to three MII Interfaces
- Up to six SMII Interfaces
- Up to one UTOPIA Level 2 Interface
- IEEE-1588 Hardware Assist

Typical Applications

- Small-to-Medium Business Router
- Industrial Controllers
- Modular Router
- Access Points (802.11a/b/g)
- Network-Attached Storage
- Wired/Wireless RFID Readers
- VoIP Integrated Access Device (IAD)
- Video IP Telephones
- Security Gateway/Router
- Network Printers
- Control Plane
- Mini-DSLAM

Document Number: 306261-002
May 2005

Contents

Figures

Tables

Revision History

Date	Revision	Description
May 2005	002	Added support for Intel® IXP455 Network Processor including Table 1 on page 14, Figure 3 on page 18, Table 27 on page 89, and Table 82 on page 163. Section 4.0, "Package Information" on page 40: added "Package Markings" and "Part Numbers" sections. Table 11 on page 48: enhanced description of DDRI_CB[7:0]. Table 49 on page 128: added $T_{SLEW\ RATE}$ information.
March 2005	001	Initial release of document.

1.0 Product Features

1.1 Product Line Features

Note: This document discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. A subset of these features is supported by certain processors in the IXP45X/IXP46X product line, such as the Intel® IXP460 or Intel® IXP455 network processors. For details on feature support listed by processor, see Table 1 on page 14.

Some of the features described in this document require software delivered by Intel. Some features may not be enabled with current software releases. The features which require software are identified within this document. Please refer to the *Intel® IXP400 Software Programmer's Guide* for information on which features are enabled at this time.

- Intel XScale® Core (compliant with Intel® StrongARM* architecture)
  - — High-performance processor based on Intel XScale® Microarchitecture
  - — Seven/eight-stage Intel® Super-Pipelined RISC Technology
  - — Memory Management Unit (MMU)
    - 32-entry, data memory management unit
    - 32-entry, instruction memory management unit (MMU)
    - 32-KByte, 32-way, set associative instruction cache
    - 32-KByte, 32-way, set associative data cache
    - 2-KByte, two-way, set associative mini-data cache
    - 128-entry, branch target buffer
    - Eight-entry write buffer
    - Four-entry fill and pend buffers
  - — Clock speeds:
    - 266 MHz
    - 400 MHz
    - 533 MHz
    - 667 MHz (Not supported on Intel® IXP455 Network Processor)
  - — Intel® StrongARM* Version 5TE Compliant
  - — Intel® Media Processing Technology
    Multiply-accumulate coprocessor
  - — Debug unit
    Accessible through JTAG port
- PCI interface
  - — 32-bit interface
  - — Selectable clock
    - 33-MHz clock output produced by GPIO15
    - 1- to 66-MHz clock input
  - — *PCI Local Bus Specification*, Revision 2.2 compatible

  - — PCI arbiter supporting up to four external PCI devices (four REQ/GNT pairs)
  - — Host/option capable
  - — Master/target capable
  - — Two DMA channels
- USB 1.1 device controller
  - — Full-speed capable
  - — Embedded transceiver
  - — 16 endpoints
- USB 2.0 host controller
  - — Low-speed and full-speed capable
  - — Embedded transceiver
  - — EHCI Compliant
  - — Separate interface from USB 1.1 device controller
- DDRI-266 SDRAM interface
  - — Internally multi-ported Memory Controller Unit (Three Internal Ports)
  - — 32-bit data
  - — 13-bit address
  - — 133.32 MHz (which is 4 * OSC_IN input pin)
  - — Supports 128/256/512/1,024-Mbit technologies
  - — Unbuffered DDRI SDRAM support only
  - — Up to eight open pages simultaneously maintained
  - — Support for 32 Mbyte, minimum; 1 Gbyte, maximum
  - — User-enabled, single-bit error correction/multi-bit error detection ECC support (ECC not supported on Intel® IXP455 Network Processor)
- Expansion interface
  - — Master/Target interface
  - — 25-bit address
  - — 32-bit data
  - — Eight programmable outbound chip selects
  - — One inbound chip select
  - — Four request/grant pairs
  - — Outbound transfers (IXP45X/IXP46X network processors are the master to external target devices)
  - — Inbound transfers (IXP45X/IXP46X network processors are a target to external masters)
  - — Bus tri-state for sideband transfers (External masters accesses to external target device)
  - — Outbound transfer support

- Supports Intel/Motorola* microprocessors
- Multiplexed-style bus cycles
- Simplex-style bus cycles
- DSP support for Texas Instruments* DSPs supporting HPI*-8 bus cycles
- DSP support for Texas Instruments DSPs supporting HPI-16 bus cycles
- Synchronous flash support
- Flow through ZBT SRAM burst support
- Up to 80-MHz operation at 40 pF load
- Supports even/odd-parity generation and checking in all extended modes and in some legacy modes (Intel and Motorola style bus cycles)

— Inbound transfer support
- Single transfer or burst support

- Cryptography Unit
  — Exponentiation Unit (EAU)
  — Random Number Generator (RNG)
  — Secure Hash Algorithm (SHA)
- Two UART Interfaces
  — 1,200 Baud to 921 Kbaud
  — 16550 compliant
  — 64-byte Tx and Rx FIFOs
  — CTS and RTS modem-control signals
- Synchronous Serial Port Interface
  — Master Mode Only
  — Motorola's Serial Peripheral Interface (SPI)
  — National's Microwire*
  — Texas Instruments' synchronous serial protocol (SSP)
- I^2C interface
  — Multi-master capable
  — Slave capable
  — Fast-mode support 400 Kbps
  — Slow-mode support 100 Kbps
- Internal bus performance monitoring unit (IBPMU)
  — Seven 27-bit event counters
  — Monitoring of internal-bus occurrences and duration events
- 16 GPIOs
- Four internal timers
  — Watchdog Timer
  — General-Purpose Timer

  - — Two one-shot timers
- Packaging
  - — 544-pin PBGA
  - — Commercial temperature (0° to 70° C)
  - — Extended temperature (-40° to 85° C)
  - — Lead Free Support

The remaining features described in the product line features list require software in order for these features to be functional. To determine if the feature is enabled, see the *Intel® IXP400 Software Programmer's Guide*.

- Three network processor engines (NPEs)**Note 1**
  Used to off load typical Layer-2 networking functions such as:
  - — Ethernet filtering
  - — ATM SARing
  - — HDLC
  - — Layer-2 switching
  - — Security acceleration (AES/DES3/SHA/MD-5)
- Configurable Network Interface, configurable in the following manner: **Notes 1, 3**
  - — Three MII/SMII/SS-SMII interfaces
  - — Two MII/SMII/SS-SMII interface + 1 UTOPIA Level 2 interface
  - — One MII/SMII/SS-SMII interface + 1 UTOPIA Level 2 interface + four SMII/SS-SMII interfaces
  - — Two MII/SMII/SS-SMII interfaces + four SMII/SS-SMII interfaces
- MII/SMII/SS-SMII interfaces are: **Note 1**
  - — 802.3 MII interfaces that additionally support SMII/SS-SMII interfaces
  - — Single MDIO interface to control the MII/SMII/SS-SMII interfaces
- UTOPIA Level 2 Interface is: **Note 1**
  - — Eight-bit interface
  - — Up to 33-MHz clock speed
  - — Five transmit and five receive address lines
- Encryption/Authentication **Note 1**
  - — DES
  - — DES3
  - — AES 128-bit and 256-bit
  - — Single-pass AES-CCM
  - — SHA-1, SHA-256, SHA-384, SHA-512
  - — MD-5
- Two high-speed, serial interfaces **Note 1**

  - — Six-wire
  - — Supports speeds up to 8.192 MHz
  - — Supports connection to T1/E1 framers
  - — Supports connection to CODEC/SLICs
  - — Eight HDLC channels
  - — Clock source provided from an external source or internal HSS clock divider
- IEEE 1588 Hardware Assistance **Notes 2, 3**
  - — Time master support
  - — Time target support

Notes:

1. This feature requires Intel supplied software. To determine if this feature is enabled by a particular software release, see the *Intel® IXP400 Software Programmer's Guide*.
2. Although this feature has direct access from the Intel XScale® Core, this feature monitors the activity of the MII interfaces which requires Intel-supplied software to operate.
3. Four SMII/SS-SMII interfaces and IEEE 1588 hardware assistance are not available for the Intel® IXP455 Network Processor.

1.2 Model-Specific Features

Table 1. Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Features

Feature	Intel® IXP465	Intel® IXP460	Intel® IXP455
Processor Speed (MHz)	266 / 400 / 533 / 667	266 / 400 / 533 / 667	266 / 400 / 533
GPIO	X	X	X
UART 0/1	X	X	X
HSS 0 (NPE-A)†	X		X
HSS 1 (NPE-A)†	X		X
UTOPIA 2/ MII / SMII (NPE A)†	X		X
MII / SMII / 4-Port SMII (NPE B)†	X		X††
MII / SMII (NPE C)†	X	X	X
USB 1.1 Device Controller	X	X	X
USB 2.0 Host Controller	X	X	X
PCI	32-bit, up to 66-MHz	32-bit, up to 66-MHz	32-bit, up to 66-MHz
Expansion Bus	32-bit or 16-bit, 80-MHz, Host Support, Parity Support	32-bit or 16-bit, 80-MHz, Host Support, Parity Support	32-bit or 16-bit, 80-MHz, Host Support, Parity Support
DDRI-266 SDRAM	32-bit, 133-MHz clock with ECC	32-bit, 133-MHz clock with ECC	32-bit, 133-MHz clock without ECC
AES / AES-CCM/ DES / DES3 †	X		X
Cryptography Unit	X		X
Multi-Channel HDLC †	8		8
SHA / MD-5 †	X		X
IEEE1588 Hardware Assistance	X	X	
I^2C	X	X	X
SSP	X	X	X
Commercial Temperature	X	X	X
Extended Temperature	X	X	X

† These features require Intel-supplied software in order to be operational. To determine if the feature is enabled, see the *Intel® IXP400 Software Programmer's Guide*.

†† 4-Port SMII is not supported on the IXP455 network processor.

2.0 About This Document

This datasheet contains a functional overview of the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors, as well as mechanical data (package signal locations and simulated thermal characteristics), targeted electrical specifications, and some bus functional wave forms for the device.

Detailed functional descriptions — other than parametric performance — are published in the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Other related documents are shown in Table 2.

Table 2. Related Documents

Document Title	Document #
Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual	306262
Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Hardware Design Guidelines	305261
Intel® IXP4XX Product Line of Network Processors Specification Update	306428
Intel® IXP400 Software Programmer's Guide	252539
Intel XScale® Core Developer's Manual	273473
Intel XScale® Microarchitecture Technical Summary	—
PCI Local Bus Specification, Revision 2.2	N/A
Universal Serial Bus Specification, Revision 1.1	N/A
DDR SDRAM Specification	N/A

3.0 Functional Overview

The Intel® IXP45X and Intel® IXP46X Product Line of Network Processors are compliant with the Intel® StrongARM* Version 5TE instruction-set architecture (ISA). The IXP45X/IXP46X network processors are designed with Intel, 0.18-micron semiconductor process technology. This process technology — along with the compactness of the Intel® StrongARM* RISC ISA, the ability to simultaneously process data with up to three integrated network processing engines (NPEs), and numerous dedicated-function peripheral interfaces — enables the IXP45X/IXP46X network processors to operate over a wide range of low-cost networking applications with industry-leading performance.

As indicated in Figure 1, Figure 2, and Figure 3, the IXP45X/IXP46X network processors combine many features with the Intel XScale® Core to create a highly integrated processor applicable to LAN/WAN-based networking applications in addition to other embedded networking applications.

This section briefly describes the main features of the product. For detailed functional descriptions, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Figure 1. Intel® IXP465 Network Processor Block Diagram

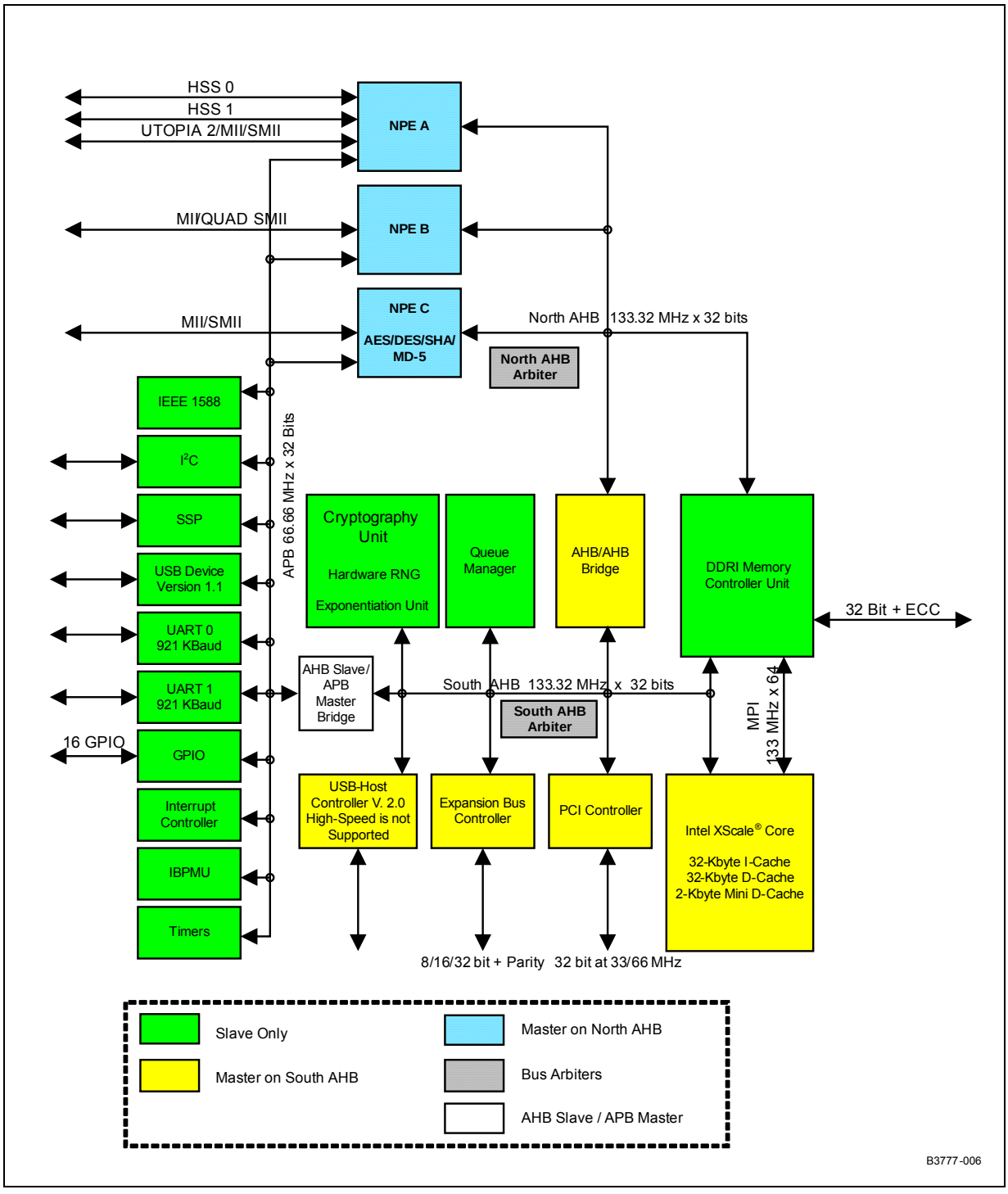


Figure 2. Intel® IXP460 Network Processor Block Diagram

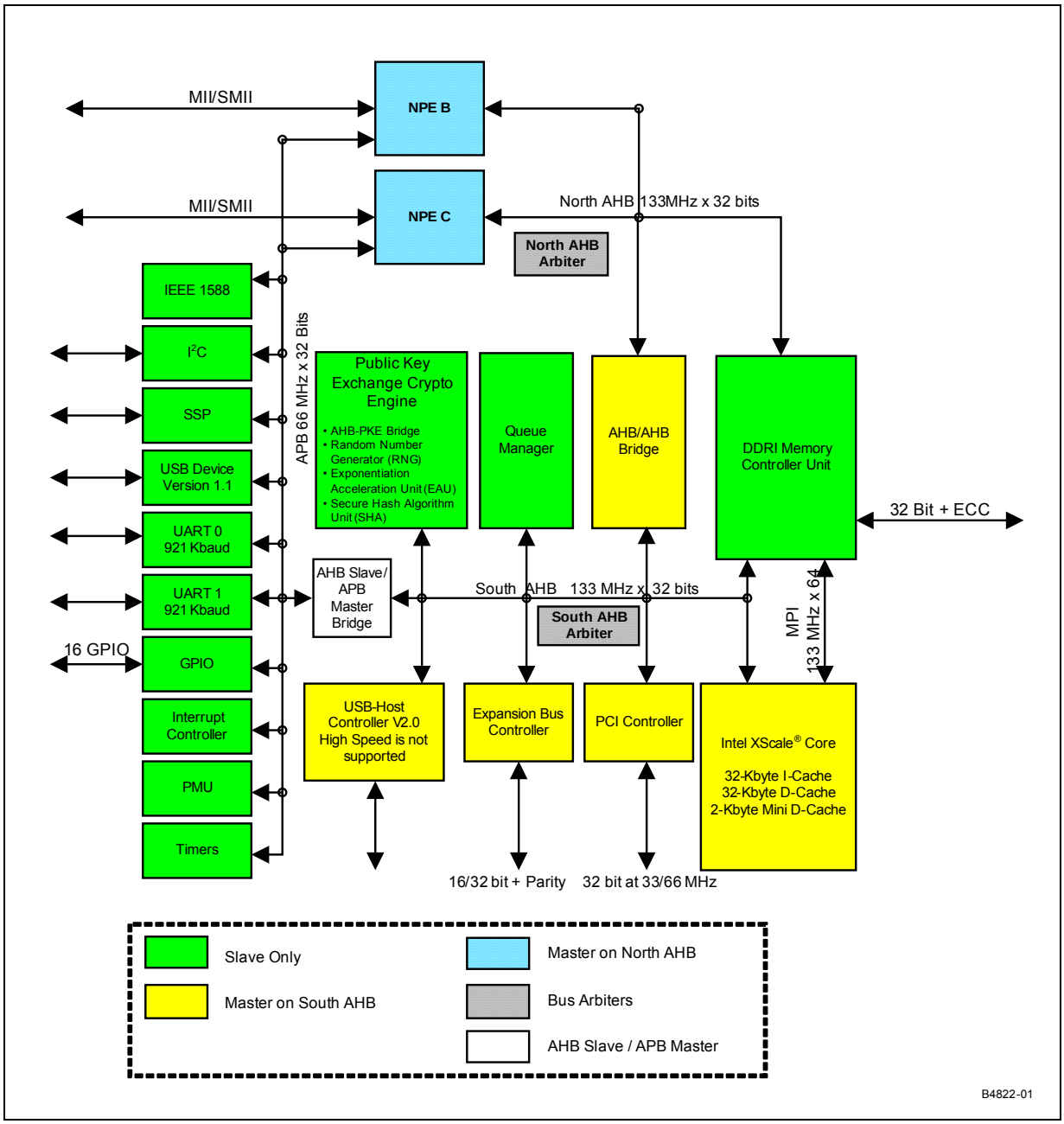
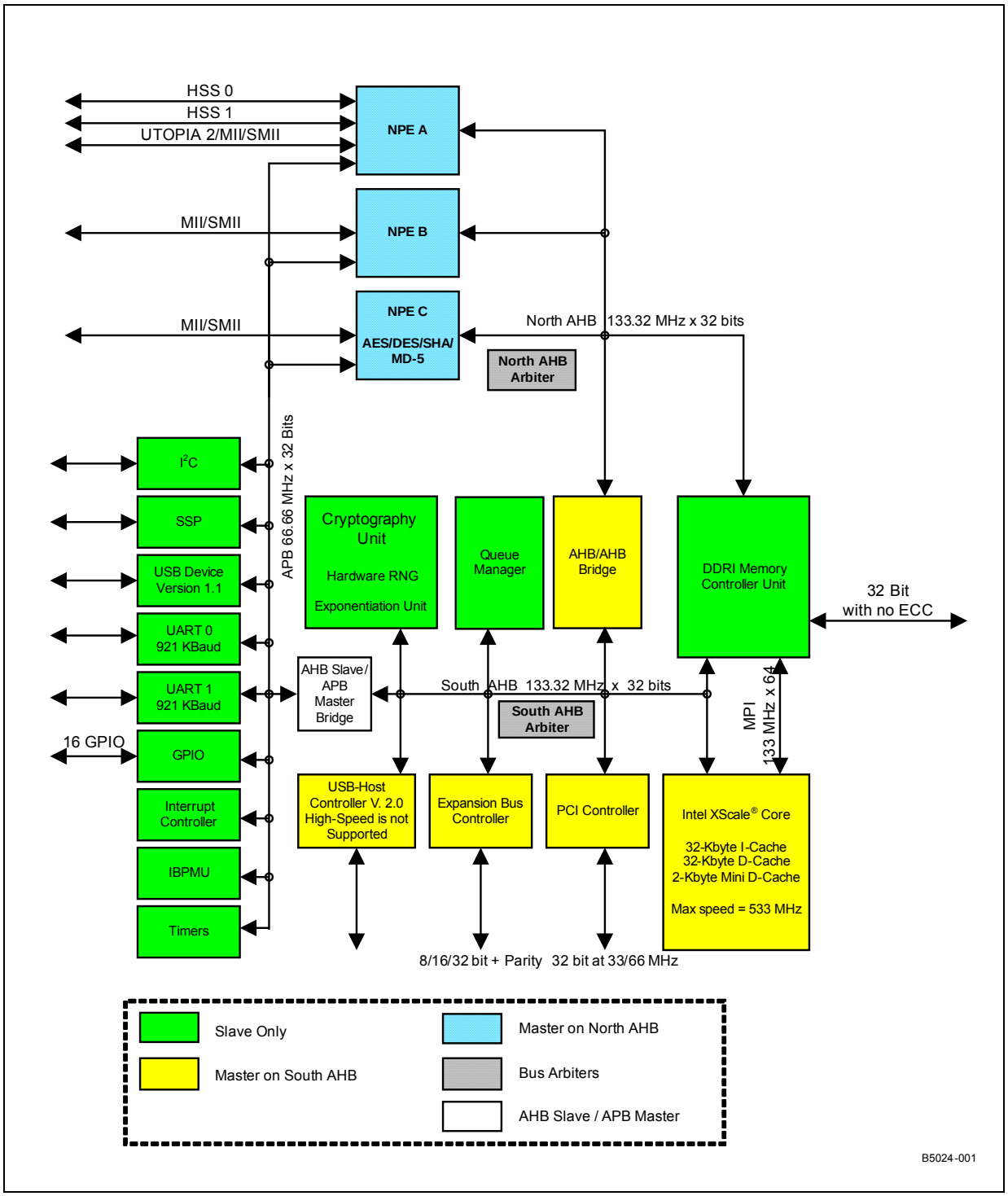


Figure 3. Intel® IXP455 Network Processor Block Diagram



3.1 Key Functional Units

The following sections briefly describe the functional units and their interaction in the system. For more detailed information, refer to the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Unless otherwise specified, the functional descriptions apply to all of the IXP45X/IXP46X network processors. For specific information on supported interfaces, refer to Table 1 on page 14. For model-specific block diagrams, see Figure 1 on page 16, Figure 2 on page 17, and Figure 3 on page 18.

3.1.1 Network Processor Engines (NPEs)

The network processor engines (NPEs) are dedicated-function processors containing hardware coprocessors integrated into the IXP45X/IXP46X network processors. The NPEs are used to off load processing function required by the Intel XScale core.

These NPEs are high-performance, hardware-multi-threaded processors with additional local-hardware-assist functionality used to off load highly processor-intensive functions such as MII (MAC), CRC checking/generation, AAL 2 segmentation and re-assembly, AES, AES-CCM, DES, DES3, SHA-1/256/384/512, MD5, etc.

All instruction code for the NPEs are stored locally and is accessed using a dedicated instruction memory bus. Likewise, a separate dedicated data memory bus allows accesses to local code store as well as DDR SDRAM via the AHB bus.

These NPEs support processing of the dedicated peripherals that can include:

- One UTOPIA Level 2 (Universal Test and Operation PHY Interface for ATM) interface
- Two High-Speed Serial (HSS) interfaces
- Up to three Media-Independent Interface (MII), up to six Serial Media Independent Interfaces (SMII)/Source-Synchronous Serial Media Independent Interfaces (SS-SMII), or some combination of each.

Table 3 specifies the possible combination of interfaces for the NPEs contained on the IXP45X/IXP46X network processors. These configurations are determined by the factory programmed fuse settings or by software that configures the part during boot-up.

Table 3. Network Processor Functions (Sheet 1 of 2)

Device	UTOPIA	HSS	MII /SMII A	MII / 4 SMII B	MII / SMII C	AES / DES / DES3	HDLC	SHA MD-5
Configuration 0 (default)	X	X		MII	MII	X	8	X
Configuration 1	X	X		SMII	MII	X	8	X
Configuration 2	X	X		SMII	SMII	X	8	X
Configuration 3	X	X		4 SMII	MII	X	8	X
Configuration 4	X	X		4 SMII	SMII	X	8	X
Configuration 5		X	MII	MII	MII	X	8	X
Configuration 6		X	MII	SMII	MII	X	8	X

Table 3. Network Processor Functions (Sheet 2 of 2)

Device	UTOPIA	HSS	MII /SMII A	MII / 4 SMII B	MII / SMII C	AES / DES / DES3	HDLC	SHA MD-5
Configuration 7		X	MII	SMII	SMII	X	8	X
Configuration 8		X	MII	4 SMII	MII	X	8	X
Configuration 9		X	MII	4 SMII	SMII	X	8	X
Configuration 10		X	SMII	4 SMII	SMII	X	8	X
Configuration 11		X	SMII	SMII	SMII	X	8	X
Configuration 12		X	SMII	SMII	MII	X	8	X
Configuration 13		X	SMII	4 SMII	MII	X	8	X

The NPE core is a hardware-multi-threaded processor engine that is used to accelerate functions that are difficult to achieve high performance in a standard RISC processor. Each NPE core is a 133.32-MHz (which is 4 * OSC_IN input pin) processor core that has self-contained instruction memory and self-contained data memory that operate in parallel. Each NPE core has 4 K words of instruction memory and 4 K words of data memory.

In addition to having separate instruction/data memory and local-code store, the NPE core supports hardware multi-threading with support for multiple contexts. The support of hardware multi-threading creates an efficient processor engine with minimal processor stalls due to the ability of the processor core to switch contexts in a single clock cycle, based on a prioritized/preemptive basis. The prioritized/preemptive nature of the context switching allows time-critical applications to be implemented in a low-latency fashion — which is required when processing multi-media applications.

The NPE core also connects to several hardware-based coprocessors that are used to implement functions that are difficult for a processor to implement. These functions include:

- HSS Serialization/ De-serialization
- DES/DES3/AES
- MD-5
- Learning/filtering content addressable memory
- UTOPIA Level 2 Framing
- CRC checking/generation
- SHA-1/256/384/512
- HDLC bit stuffing/de-stuffing
- Media Access Controller functionality

These coprocessors are implemented in hardware, enabling the coprocessors and the NPE processor core to operate in parallel.

With the addition of the new switching coprocessor (SWCP) and the Ethernet coprocessors, functions like a four-port, Layer-2 switch can be easily implemented using all Intel-based silicon. Also, by using NPEs to implement switching functions, value added features like VLAN or IP switching can be easily upgraded using existing silicon. Therefore, speeding up the end customer's time to market while keeping product costs the same.

The combined forces of the hardware multi-threading, local-code store, independent instruction memory, independent data memory, and parallel processing — contained on the NPE — allows the Intel XScale core to be utilized for application purposes. The multi-processing capability of the peripheral interface functions allows unparalleled performance to be achieved by the application running on the Intel XScale core.

3.1.2 Internal Bus

The internal bus architecture of the IXP45X/IXP46X network processors are designed to allow parallel processing to occur and to isolate bus utilization, based on particular traffic patterns. The bus is segmented into four major buses:

- North Advanced, High-Performance Bus (AHB)
- South AHB
- Memory Port Interface
- Advanced Peripheral Bus (APB)

3.1.2.1 North AHB

The North AHB is a 133.32-MHz (which is 4 * OSC_IN input pin), 32-bit bus that can be mastered by the NPE A, NPE B, or NPE C. The targets of the North AHB can be the DDRI SDRAM or the AHB/AHB bridge. The AHB/AHB bridge allows the NPEs to access the peripherals and internal targets on the South AHB.

Data transfers by the NPEs on the North AHB to the South AHB are targeted predominately to the queue manager. Transfers to the AHB/AHB bridge may be “posted” — when writing — or “split” — when reading.

When a transaction is “posted,” a master on the North AHB requests a write to a peripheral on the South AHB. If the AHB/AHB Bridge has a free FIFO location, the write request will be transferred from the master on the North AHB to the AHB/AHB bridge. The AHB/AHB bridge will complete the write on the South AHB, when it can obtain access to the peripheral on the South AHB. The North AHB is released to complete another transaction.

When a transaction is “split,” a master on the North AHB requests a read of a peripheral on the South AHB. If the AHB/AHB bridge has a free FIFO location, the read request will be transferred from the master on the North AHB to the AHB/AHB bridge. The AHB/AHB bridge will complete the read on the South AHB, when it can obtain access to the peripheral on the South AHB.

Once the AHB/AHB bridge has obtained the read information from the peripheral on the South AHB, the AHB/AHB bridge notifies the arbiter, on the North AHB, that the AHB/AHB bridge has the data for the master that requested the “split” transfer. The master on the North AHB — that requested the split transfer — will arbitrate for the North AHB and transfer the read data from the AHB/AHB bridge. The North AHB is released to complete another transaction while the North AHB master — that requested the “split” transfer — waits for the data to arrive.

These “posting” and “splitting” transfers allow control of the North AHB to be given to another master on the North AHB — enabling the North AHB to achieve maximum efficiency. Transfers to the AHB/AHB bridge are considered to be small and infrequent, relative to the traffic passed between the NPEs and the DDRI SDRAM on the North AHB.

When multiple masters arbitrate for the North AHB, the masters are awarded access to the bus in a round-robin fashion. Each transaction can be no longer than an eight-word burst. This implementation promotes fairness within the system.

3.1.2.2 South AHB

The South AHB is a 133.32-MHz (which is 4 * OSC_IN input pin), 32-bit bus that can be mastered by the Intel XScale core, PCI controller, Expansion Bus Interface, USB Host Controller, and the AHB/AHB bridge. The targets of the South AHB Bus can be the DDRI SDRAM, PCI Controller, Queue Manager, Expansion Bus, or the AHB/APB bridge. As a special case, the Intel XScale® Core is the only master which can access the Cryptography Unit (target).

Accesses across the APB/AHB bridge allows interfacing to peripherals attached to the APB. The Expansion bus and PCI controller can be configured to support split transfers.

Arbitration on the South AHB are round-robin. Each transaction can be no longer than an eight-word burst. This implementation promotes fairness within the system.

3.1.2.3 Memory Port Interface

The Memory Port Interface (MPI) is a 128-bit bus that provides the Intel XScale core a dedicated interface to the DDRI SDRAM. The Memory Port Interface operates at 133.32 MHz (which is 4 * OSC_IN input pin).

The Memory Port Interface stores memory transactions from the Intel XScale core which have not been processed by the Memory Controller. The Memory Port Interface supports eight core processor read transactions up to 32 bytes each. That total equals the maximum number of outstanding transaction the Core Processor Bus Controller can support. (That includes core DCU [4 - load requests to unique cache lines], IFU [2 - prefetch], IMM [1 - tablewalk], DMM [1 - tablewalk].)

The Memory Port Interface also supports eight core-processor-posted write transactions up to 16 bytes each.

Arbitration on the Memory Port Interface is not required due to no contention with other masters. Arbitration will exist in the DDRI memory controller between all of the main internal busses.

3.1.2.4 APB Bus

The APB Bus is a 66.66-MHz (which is 2* OSC_IN input pin), 32-bit bus that can be mastered by the AHB/APB bridge only. The targets of the APB bus can be:

- USB 1.1 device controller
- The internal bus performance monitoring unit (IBPMU)
- GPIO
- IEEE 1588 Hardware Assist
- I^2C
- UARTs
- All NPEs
- Interrupt controller
- Timers
- Serial Peripheral Port Interface

The APB interface is also used as an alternate-path interface to the NPEs and is used for NPE code download and configuration.

No arbitration is required due to a single master implementation.

3.1.3 MII/SMII Interfaces

The IXP45X/IXP46X network processors can be configured to support up to three MII, up to six SMII/SS-SMII industry-standard, or some combination thereof, media-independent interface (MII) interfaces. These interfaces are integrated into the IXP45X/IXP46X network processors with separate media-access controllers and in many cases independent network processing engines. (See Table 3 for allowable combinations.)

The independent NPEs and MACs allow parallel processing of data traffic on the MII interfaces and off loading of processing required by the Intel XScale core. The IXP45X/IXP46X network processors are compliant with the IEEE 802.3 specification.

In addition to the MII interfaces, the IXP45X/IXP46X network processors include a single management data interface that is used to configure and control PHY devices that are connected to the MII interfaces. The IXP45X/IXP46X network processors provide support for serial media independent interface (SMII).

3.1.4 UTOPIA Level 2

The integrated UTOPIA Level 2 interface works with a network-processing engine core for several of the IXP45X/IXP46X network processors. The pins of the UTOPIA Level 2 interface are multiplexed with one of the MII/SMII interfaces. (See Table 3 for details.)

The UTOPIA Level 2 interface supports a single- or a multiple-physical-interface configuration with cell-level or octet-level handshaking. The network processing engine handles segmentation and reassembly of ATM cells, CRC checking/generation, and transfer of data to/from memory. This allows parallel processing of data traffic on the UTOPIA Level 2 interface, off-loading these processing tasks from the Intel XScale core.

The IXP45X/IXP46X network processors are compliant with the ATM Forum, *UTOPIA Level 2 Specification*, Revision 1.0.

3.1.5 USB 1.1 Device Interface

The integrated USB 1.1 device interface supports full-speed operation and 16 endpoints and includes an integrated transceiver.

There are:

- Six isochronous endpoints (three input and three output)
- One control endpoints
- Three interrupt endpoints
- Six bulk endpoints (three input and three output)

3.1.6 USB 2.0 Host Interface

USB Host functionality is implemented on the IXP45X/IXP46X network processors. The function being performed is defined by the USB 2.0 specification, maintained by usb.org and the interface is (largely) EHCI compliant, as defined by Intel.

Not all features defined by the 2.0 specification are supported for this implementation. The following is a partial list of supported features:

- Host function
- Low-speed interface
- Full-speed interface
- EHCI register interface

The following is a partial list of features *not* supported:

- Device function
- OTG function
- High-speed interface

3.1.7 PCI Controller

The IXP45X/IXP46X network processors' PCI controller is compatible with the *PCI Local Bus Specification*, Rev. 2.2. The PCI interface is 32-bit compatible bus and capable of operating as either a host or an option (i.e. not the Host). This PCI implementation supports 3.3 V I/O only.

3.1.8 DDRI SDRAM Controller

The IXP45X/IXP46X network processors integrate a high-performance, multi-ported Memory Controller Unit (MCU) to provide a direct interface between the IXP45X/IXP46X network processors and their local memory subsystem. The MCU supports:

- DDRI 266 SDRAM
- 128/256/512-Mbit, 1-Gbit DDRI SDRAM technology support
- Only unbuffered DRAM support (No registered DRAM support)
- Dedicated port for Intel XScale core to DDR SDRAM
- Between 32 Mbyte and 1 Gbyte of 32-bit DDR SDRAM for low-cost solutions
- Single-bit error correction, multi-bit detection support (ECC)
- 32-, 40-bit wide Memory Interfaces (non-ECC and ECC support)

The DDRI SDRAM interface provides a direct connection to a high-bandwidth and reliable memory subsystem. The DDRI SDRAM interface is a 32-bit-wide data path.

An 8-bit Error Correction Code (ECC) across each 32-bit word improves system reliability. It is important to note that ECC is also referred to as CB in many DIMM specifications. The pins on IXP45X/IXP46X network processors are called DDRI_CB[7:0]. The controller supports the 8 bits due to the fact that internally it is a 32- or 64-bit controller. However, this implementation of the controller only supports 32 bits.

Note: The IXP455 network processor does not support ECC functionality.

The ECC circuitry was designed to operate always on a 64-bit word and when operating in 32-bit mode, the upper 32 bits are driven to zeros internally. To summarize the impact to the customer, the full 8 bits of ECC must be stored and read from a memory array in order for the ECC logic to work. An 8-bit-wide memory must be used when implementing ECC.

The memory controller only corrects single bit ECC errors on read cycles. The ECC is stored into the DDRI SDRAM array along with the data and is checked when the data is read. If the code is incorrect, the MCU corrects the data (if possible) before reaching the initiator of the read. ECC error scrubbing must be done with software. User-defined fault correction software is responsible for scrubbing the memory array and handling double-bit errors.

In order to limit double-bit errors from occurring, periodically reading the entire usable memory array will allow the hardware unit within the memory controller to correct any single-bit, ECC errors that may have occurred prior to these errors becoming double-bit ECC errors. Using this method is system-dependent.

It is important to note as well, that when sub-word writes (byte writes or half-word writes) to a 32-bit memory with ECC enabled, the memory controller will implement read-modify writes. Implementing read-modify writes is important to understand when understanding performance implications when writing software.

To understand a read-modify write, understanding that a byte to be written falls within a 32-bit word which is addressed on a word-aligned boundary. When a byte write is requested, the memory controller will read the 32-bit word which encompasses the byte that is to be written. The memory controller will then modify the specified byte, calculate a new ECC, and then write the entire 32-bit word back into the memory location it was read from.

The value written back into the memory location will contain the 32-bit word with the modified byte and the new ECC value.

The MCU supports two banks of DDR SDRAM. The MCU has support for unbuffered DDRI 266 only.

Table 4 illustrates the supported DDR SDRAM configurations for the IXP45X/IXP46X network processors. The 128/256/512-Mbit, 1-Gbit DDRI SDRAM devices comprise four internal leaves. The MCU controls the leaf selects within 128/256/512-Mbit, 1-Gbit DDRI SDRAM by toggling DDRI_BA[0] and DDRI_BA[1].

The two DDR SDRAM chip enables (DDRI_CS[1:0]#) support a DDRI SDRAM memory subsystem consisting of two banks. The base address for the two contiguous banks are programmed in the DDR SDRAM Base Register (SDBR) and must be aligned to a 32-Mbyte boundary. The size of each DDR SDRAM bank is programmed with the DDR SDRAM boundary registers (SBR0 and SBR1).

Table 4. Supported DDRI Memory Configurations (Sheet 1 of 2)

DDRI SDRAM Technology	DDRI SDRAM Arrangement	# Banks	Address Size		Leaf Select		Total Memory Size[1]	Page Size[2]
			Row	Col	DDRI_BA[1]	DDRI_BA[0]		
128 Mbit	16M x 8	1	12	10	I_AD[26]	I_AD[25]	64 Mbyte	4K
		2					128 Mbyte	4K
	8M x 16	1	12	9	I_AD[25]	I_AD[24]	32 Mbyte	2K
		2					64 Mbyte	2K
256 Mbit	32M x 8	1	13	10	I_AD[27]	I_AD[26]	128 Mbyte	4K
		2					256 Mbyte	4K
	16M x 16	1	13	9	I_AD[26]	I_AD[25]	64 Mbyte	2K
		2					128 Mbyte	2K

NOTES:
1. Table indicates 32-bit-wide memory subsystem sizes
2. Table indicates 32-bit-wide memory page sizes

Table 4. Supported DDRI Memory Configurations (Sheet 2 of 2)

DDRI SDRAM Technology	DDRI SDRAM Arrangement	# Banks	Address Size		Leaf Select		Total Memory Size[1]	Page Size[2]
			Row	Col	DDRI_BA[1]	DDRI_BA[0]		
512 Mbit	64M x 8	1	13	11	I_AD[28]	I_AD[27]	256 Mbyte	8K
		2					512 Mbyte	8K
	32M x 16	1	13	10	I_AD[27]	I_AD[26]	128 Mbyte	4K
		2					256 Mbyte	4K
1 Gbit	128M x 8	1	14	11	I_AD[29]	I_AD[28]	512 Mbyte	8K
		2					1 Gbyte	8K
	64M x 16	1	14	10	I_AD[28]	I_AD[27]	256 Mbyte	4K
		2					512 Mbyte	4K

NOTES:
1. Table indicates 32-bit-wide memory subsystem sizes
2. Table indicates 32-bit-wide memory page sizes

The memory controller is a 32-bit only interface. If a x16 memory chip is used, a minimum of two memory chips would be required to facilitate the 32-bit interface required by the IXP45X/IXP46X network processors. If ECC is required, additional memories would need to be added. For more information on DDRI SDRAM support and configuration see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

The memory controller internally interfaces to the North AHB, South AHB, and Memory Port Interface with independent interfaces. This architecture allows DDRI SDRAM transfers to be interleaved and pipelined to achieve maximum possible efficiency.

The maximum burst size supported to the DDRI SDRAM interface is eight 32-bit words. This burst size allows the best efficiency/fairness performance between peripheral accesses from the North AHB, the South AHB, and the MPI.

The programming priority of the MCU is for the Memory Port Interface to have the highest priority and two AHB ports will have the next highest priority. For more information on MCU arbitration support and configuration see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

One item to be aware of is that when ECC is being used, the memory chip chosen to support the ECC must match that of the technology chosen on the interface. Therefore, if x8 in a given configuration technology is chosen then the ECC memory chip must be the same. If a x16 configuration is chosen then a x16 chip must be used for the ECC chip.

3.1.9 Expansion Interface

The expansion interface allows easy and — in most cases — glue-less connection to peripheral devices. It also provides input information for device configuration after reset.

Some of the peripheral device types are SRAM, flash, ATM control interfaces, and DSPs used for voice applications. (Some voice configurations can be supported by the HSS interfaces and the Intel XScale core, implementing voice-compression algorithms.)

The expansion interface functions in two modes of operation:

- Legacy (16-bit, data mode)
- Enhanced (32-bit, data mode)

3.1.9.1 Expansion Bus Legacy Mode of Operation

In the legacy mode of operation, the expansion interface is a 16-bit interface that allows an address range of 512 bytes to 16 Mbytes, using 24 address lines for each of the eight independent chip selects.

Accesses to the expansion bus interface is completed in five phases. Each of the five phases can be lengthened or shortened by setting various configuration registers on a per-chip-select basis. This feature allows the IXP45X/IXP46X network processors to connect to a wide variety of peripheral devices with varying speeds.

The expansion interface supports Intel or Motorola* microprocessor-style bus cycles. The bus cycles can be configured to be multiplexed address/data cycles or separate address/data cycles for each of the eight chip-selects.

Additionally, Chip Selects 4 through 7 can be configured to support Texas Instruments* HPI-8 or HPI-16 style accesses for DSPs.

The expansion interface is an asynchronous interface to externally connected chips. However, a clock must be supplied to expansion interface of the IXP45X/IXP46X network processors for the interface to operate. This clock can be driven from GPIO 15 or an external source. The maximum clock rate that the expansion interface can accept in legacy mode of operation is 66 MHz. If GPIO 15 is used as the clock source, the Expansion Bus interface can only be clocked at a maximum of 33.32 MHz. GPIO 15's maximum clock rate is 33.32 MHz.

By providing this legacy mode of operation, code developed for previous generations of this platform becomes easily portable.

3.1.9.2 Expansion Bus Enhanced Mode of Operation

In the enhanced mode of operation, the expansion interface is a 32-bit interface that allows an address range of 512 bytes to 32 Mbytes per chip select on IXP45X/IXP46X network processors, using 25 address lines for each of the eight independent chip selects.

Additionally, in enhanced mode, the interface supports shared access to the bus with external masters. This shared access is achieved with four request/grant pins and an integrated arbiter. Not only can external devices access each other, but they can also access the IXP45X/IXP46X network processors' internal registers (including the DDRI SDRAM interface).

The advantage to this feature is that shared memory access can be achieved by using the DDRI SDRAM interface attached to IXP45X/IXP46X network processors. This lowers the system's overall bill of materials.

Enhanced mode also supports synchronous transfers at speeds of up to 80 MHz with a 40-pF load. In addition to fully synchronous support, the enhanced mode also supports burst transfers of up to eight-word lengths. The synchronous bus support is compatible to Zero Bus Turnaround (ZBT) SRAM cycles for inbound/outbound transactions for both read/write transactions.

Additionally, the outbound read transactions can support the Intel StrataFlash® K3 synchronous-burst support.

Byte-wide parity is an optional configuration of this interface in all modes of operation except:

- Intel StrataFlash® K3 synchronous-burst mode
- HPI mode

At the de-assertion of reset, the 25-bit address bus is used to capture configuration information from the levels that are applied to the pins at this time. External pull-up/pull-down resistors are used to tie the signals to particular logic levels. (For additional details, see "Package Information" on page 40.) If a signal is required to be placed into a pull-up state during this initialization period, the IXP45X/IXP46X network processors contain internal weak pull-ups. Depending upon the system design, pull-down resistors may be the only thing required.

3.1.10 High-Speed, Serial Interfaces

The high-speed, serial interfaces (HSS) are six-signal interfaces that support serial transfer speeds from 512 KHz to 8.192 MHz, for some models of the IXP45X/IXP46X network processors. (For processor-specific speeds, see Table 3 on page 19.)

Each interface allows direct connection of up to four T1/E1 framers and CODEC/SLICs to the IXP45X/IXP46X network processors. The high-speed, serial interfaces are capable of supporting various protocols, based on the implementation of the code developed for the network processor engine core.

For a list of supported protocols, see the *Intel® IXP400 Software Programmer's Guide*.

3.1.11 UARTs

The UART interfaces are a 16550-compliant UART with the exception of transmit and receive buffers. Transmit and receive buffers are 64 bytes-deep versus the 16 bytes required by the 16550 UART specification.

The interfaces can be configured to support speeds from 1,200 Baud to 921 Kbaud. The interfaces support configurations of:

- Five, six, seven, or eight data-bit transfers
- One or two stop bits
- Even, odd, or no parity

The request-to-send (RTS_N) and clear-to-send (CTS_N) modem control signals also are available with the interface for hardware flow control.

3.1.12 GPIO

There are 16 GPIO pins supported by the IXP45X/IXP46X network processors. GPIO pins 0 through 13 can be configured to be general-purpose input or general-purpose output. Additionally, GPIO pins 0 through 12 can be configured to be an interrupt input.

GPIO Pin 14 can be configured similar to GPIO Pin 13 or as a clock output. The output-clock configuration can be set at various speeds, up to 33 MHz, with various duty cycles. GPIO Pin 14 is configured as an input, upon reset.

GPIO Pin 15 can be configured the same as GPIO Pin 13 or as a clock output. The output-clock configuration can be set at various speeds, up to 33 MHz, with various duty cycles. GPIO Pin 15 is configured as an output, upon reset. GPIO Pin 15 can be used to clock the expansion interface, after reset.

Several other GPIO pins can serve as an alternate function, as outlined in Table 5.

Table 5. GPIO Alternate Function Table

GPIO Pin Number	Alternate Function
0	External USB 1.1 Device Clock
1	External USB 2.0 Host Clock
2	NPE A External Condition 0
3	NPE A External Condition 1
4	NPE B External Condition 4
5	NPE B External Condition 5
6	NPE C External Condition 2
7	Auxiliary IEEE1588 Master Snapshot
8	Auxiliary IEEE1588 Slave Snapshot
9:13	(Reserved)
14	Output Clock 14
15	Output Clock 15

3.1.13 Internal Bus Performance Monitoring Unit (IBPMU)

The IXP45X/IXP46X network processors contain a performance monitoring unit that may be used to capture predefined events within the system outside of the Intel XScale core. These features aid in measuring and monitoring various system parameters that contribute to the overall performance of the processor.

The Performance Monitoring (PMON) facility provided comprises:

- Eight Programmable Event Counters (PECx)
- Previous Master/Slave Register
- Event Selection Multiplexor

The programmable event counters are 27 bits wide. Each counter may be programmed to observe one event from a defined set of events. An event consists of a set of parameters which define a start condition and a stop condition.

The monitored events are selected by programming the Event Select Registers (ESR).

3.1.14 Interrupt Controller

The IXP45X/IXP46X network processors implement up to 64 interrupt sources to allow an extension of the Intel XScale core's FIQ and IRQ interrupt sources. These sources can originate from some external GPIO pins, internal peripheral interfaces, or internal logic.

The interrupt controller can configure each interrupt source as an FIQ, IRQ, or disabled. The interrupt sources tied to Interrupt 0 to 7 can be prioritized. The remaining interrupts are prioritized in ascending order. For example, Interrupt 8 has a higher priority than 9, 9 has a higher priority than 10, and 30 has a higher priority that 31.

An additional level of priority can be set for interrupts 32 through 64. This priority setting gives any interrupt between 32 through 64 priority over interrupts 0 through 31.

3.1.15 Timers

The IXP45X/IXP46X network processors contain four internal timers operating at 66.667 MHz (which is 2* OSC_IN input pin) to allow task scheduling and prevent software lock-ups. The device has four 32-bit counters:

- Watch-Dog Timer
- Timestamp Timer
- Two general-purpose Timers

The Timestamp Timer and the two general-purpose timers have the optional ability to use a pre-scaled clock. A programmable pre-scaler can be used to divide the input clock by a 16-bit value. The input clock can be either the APB clock (66.66 MHz) or a 20-ns version of the APB clock (50 MHz). By default all timers use the APB clock.

The 16-bit pre-scale value ranges from divide by 2 to 65,536 and results in a new clock enable available for the timers that ranges from 33.33 MHz down to 1,017.26 Hz.

The Timestamp Timer also contains a 32-bit compare register that allows an interrupt to be created at times other than time 0.

3.1.16 IEEE 1588 Hardware Assistance

In a distributed control system containing multiple clocks, individual clocks tend to drift apart. Some kind of correction mechanism is necessary to synchronize the individual clocks to maintain global time, which is accurate to some clock resolution. The IEEE 1588 standard for a precision clock synchronization protocol for networked measurement and control systems can be used for this purpose. The IEEE 1588 standard defines several messages that can be used to exchange timing information.

The IXP45X/IXP46X network processors implement the IEEE 1588 hardware-assist logic on three of the MII interfaces. Using the hardware assist logic along with software running on the Intel XScale core, a full source or sink capable IEEE-1588 compliant network node can be implemented.

Note: The IXP455 network processor does not support IEEE 1588 hardware-assist.

3.1.17 Synchronous Serial Port Interface

The IXP45X/IXP46X network processors have a dedicated Synchronous Serial Port (SSP) interface. The SSP interface is a full-duplex synchronous serial interface. It can connect to a variety of external analog-to-digital (A/D) converters, audio and telecom CODECs, and many other devices which use serial protocols for transferring data.

It supports National’s Microwire*, Texas Instruments’* synchronous serial protocol (SSP), and Motorola's* serial peripheral interface (SPI*) protocol.

The SSP operates in master mode (the attached peripheral functions as a slave), and supports serial bit rates from 7.2 Kbps to 1.8432 Mbps using the on-chip, 3.6864-MHz clock, and bit rates from 65.10 Kbps to 16.67 Mbps using a maximum off-chip, 33.33 MHz clock. Serial data formats may range from 4 to 16 bits in length. Two on-chip register blocks function as independent FIFOs for data, one for each direction. The FIFOs are 16 entries deep x 16 bits wide. Each 32-bit word from the system fills one entry in a FIFO using the lower half 16-bits of a 32-bit word.

3.1.18 I²C Interface

The I²C Bus Interface Unit allows the IXP45X/IXP46X network processors to serve as a master and slave device residing on the I²C bus. The I²C bus is a two-pin serial bus. SDA is the data pin for input and output functions and SCL is the clock pin for reference and control of the I²C bus.

The I²C bus allows the IXP45X/IXP46X network processors to interface to other I²C peripherals and micro-controllers for system management functions. The serial bus requires a minimum of hardware for an economical system to relay status and reliability information on the IXP45X/ IXP46X network processors subsystem to an external device.

The I²C Bus Interface Unit is a peripheral device that resides on the IXP45X/IXP46X network processors' APB. Data is transmitted to and received from the I²C bus via a buffered interface. Control and status information is relayed through a set of memory-mapped registers. Refer to the I²C Bus Specification for complete details on I²C bus operation.

The I²C supports:

- Multi-master capabilities
- Slave capabilities

The I²C unit supports both fast-mode operation — at 400 Kbps — and standard mode — at 100 Kbps. Fast mode logic levels, formats, capacitive loading and protocols function the same in both modes. The I²C unit does not support I²C 10-bit addressing or CBUS.

3.1.19 AES/DES/SHA/MD-5

The IXP45X/IXP46X network processors implement on-chip hardware acceleration for underlying security and authentication algorithms.

The encryption/decryption algorithms supported are AES, single pass AES-CCM, DES, and triple DES. These algorithms are commonly found when implementing IPSEC, VPN, WEP, WEP2, WPA, and WPA2.

The authentication algorithms supported are MD-5, SHA-1, SHA-256, SHA-384, and SHA-512. Inclusion of SHA-384 and SHA-512 allows 256-bit key authentication to pair up with 256-bit AES support.

3.1.20 Cryptography Unit

The Cryptography Unit implements three major functions:

- Exponentiation Unit (EAU)
- Random Number Generator (RNG)
- Secure Hash Algorithm (SHA function for the RNG)

The EAU supports various large number arithmetic operations. These operations include modular exponentiation, modular reduction, multiply, add and subtract. These operations are controlled through a set of memory mapped registers. Parameters for and results of the operations are written in little-endian ordering into a RAM (contained within the EAU) which the EAU state machine accesses and also uses for temporary registers. The arithmetic operations supported by the EAU are used by software executing in the host processor to build larger cryptographic functions such as signing and verification procedures. Since the EAU executes only one operation at a time, the host processor must serialize the required operations to the EAU.

The EAU begins operating after the host processor has moved data into the EAU RAM and loads the EAU's command register with an appropriate command. After executing the command, the EAU appropriately sets its status bits and waits idle until it receives another command from the host processor.

The RNG unit provides a digital, random-number generation capability. It uses a LFSR (Linear Feedback Shift Register) to generate a sequence of pseudo-random bits. These sequences are shifted into a FIFO of 32-bit words, which may be read sequentially from the random number register. A new word is generated every 32 clocks and the RNG will buffer 16 of these words at a time.

The output of the RNG should be passed through the SHA engine for added randomness. The host processor (Intel XScale core) is responsible for implementing this SHA-based, random-number generation. The LFSR also allows one entropy source. The entropy source is fed in from a PN sequence generator which has a period of 2^42 - 1. The coefficients for the PN sequence is chosen such that it produces the maximal sequence length. The coefficients are not mentioned for security reasons. The coefficients for the 128-stage LSFR are similarly not mentioned here for security reasons.

3.1.21 Queue Manager

The Queue Manager provides a means for maintaining coherency for data handling between various processors cores contained on the IXP45X/IXP46X network processors (NPE to NPE, NPE to Intel XScale core, etc.). It maintains the queues as circular buffers in an embedded 8-Kbyte SRAM. The Queue Manager also implements the status flags and pointers required for each queue.

The Queue Manager manages 64 independent queues. Each queue is configurable for buffer and entry size. Additionally status flags are maintained for each queue.

The Queue Manager interfaces include an Advanced High-performance Bus (AHB) interface to the NPEs and Intel XScale core (or any other AHB bus master), a Flag Bus interface, an event bus (to the NPE condition select logic), and two interrupts to the Intel XScale core.

The AHB interface is used for configuration of the Queue Manager and provides access to queues, queue status, and SRAM. Individual queue status for queues 0-31 is communicated to the NPEs via the flag bus. Combined queue status for queues 32-63 are communicated to the NPEs via the event bus. The two interrupts, one for queues 0-31 and one for queues 32-63, provide status interrupts to the Intel XScale core.

3.2 Intel XScale® Core

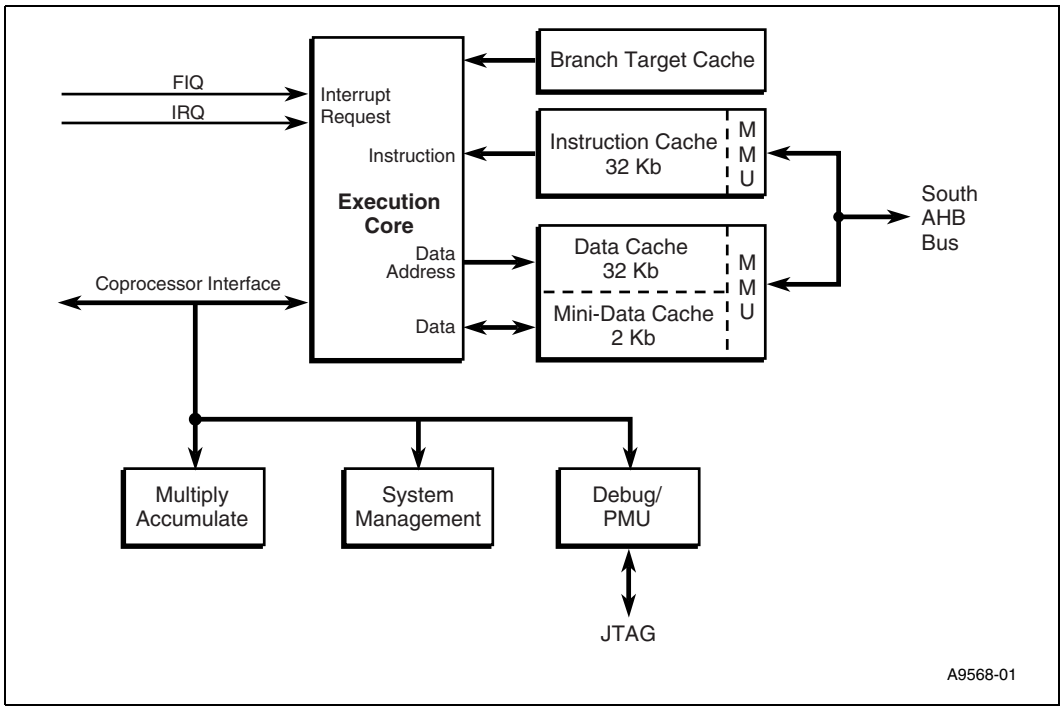
The Intel XScale technology is compliant with the Intel® StrongARM* Version 5TE instruction-set architecture (ISA). The Intel XScale core, shown in Figure 4, is designed with Intel, 0.18-micron production semiconductor process technology. This process technology — with the compactness of the Intel® StrongARM* RISC ISA — enables the Intel XScale core to operate over a wide speed and power range, producing industry-leading mW/MIPS performance.

Intel XScale core features include:

- Seven/eight-stage super-pipeline promotes high-speed, efficient core performance
- 128-entry branch target buffer keeps pipeline filled with statistically correct branch choices
- 32-entry instruction memory-management unit for logical-to-physical address translation, access permissions, and Instruction-Cache (I-cache) attributes
- 32-entry data-memory management unit for logical-to-physical address translation, access permissions, Data-Cache (D-Cache) attributes
- 32-Kbyte instruction cache can hold entire programs, preventing core stalls caused by multi-cycle memory accesses
- 32-Kbyte data cache reduces core stalls caused by multi-cycle memory accesses
- 2-Kbyte mini-data cache for frequently changing data streams avoids “thrashing” of the D-cache
- Four-entry, fill-and-pend buffers to promote core efficiency by allowing “hit-under-miss” operation with data caches
- Eight-entry write buffer allows the core to continue execution while data is written to memory
- Multiple-accumulate coprocessor that can do two simultaneous, 16-bit, SIMD multiplies with 40-bit accumulation for efficient, high-quality media and signal processing
- Performance monitoring unit (PMU) furnishing two 32-bit event counters and one 32-bit cycle counter for analysis of hit rates, etc.

  This PMU is for the Intel XScale core only. An additional PMU is supplied for monitoring of internal bus performance.
- JTAG debug unit that uses hardware break points and 256-entry trace history buffer (for flow-change messages) to debug programs

Figure 4. Intel XScale® Core Block Diagram



3.2.1 Super Pipeline

The super pipeline is composed of integer, multiply-accumulate (MAC), and memory pipes.

The integer pipe has seven stages:

- Branch Target Buffer (BTB)/Fetch 1
- Fetch 2
- Decode
- Register File/Shift
- ALU Execute
- State Execute
- Integer Writeback

The memory pipe has eight stages:

- The first five stages of the Integer pipe (BTB/Fetch 1 through ALU Execute)
  . . . then finishes with the following memory stages
- Data Cache 1
- Data Cache 2
- Data Cache Writeback

The MAC pipe has six to nine stages:

- The first four stages of the Integer pipe (BTB/Fetch 1 through Register File/ Shift) . . . then finishes with the following MAC stages
- MAC 1
- MAC 2
- MAC 3
- MAC 4
- Data Cache Writeback

The MAC pipe supports a data-dependent early terminate where stages MAC 2, MAC 3, and/or MAC 4 are bypassed.

Deep pipes promote high instruction execution rates only when a means exists to successfully predict the outcome of branch instructions. The branch target buffer provides such a means.

3.2.2 Branch Target Buffer

Each entry of the 128-entry Branch Target Buffer (BTB) contains the address of a branch instruction, the target address associated with the branch instruction, and a previous history of the branch being taken or not taken. The history is recorded as one of four states:

- Strongly taken
- Weakly taken
- Weakly not taken
- Strongly not taken

The BTB can be enabled or disabled via Coprocessor 15, Register 1.

When the address of the branch instruction hits in the BTB and its history is strongly or weakly taken, the instruction at the branch target address is fetched. When its history is strongly or weakly not-taken, the next sequential instruction is fetched. In either case the history is updated.

Data associated with a branch instruction enters the BTB the first time the branch is taken. This data enters the BTB in a slot with a history of strongly not-taken (overwriting previous data when present).

Successfully predicted branches avoid any branch-latency penalties in the super pipeline. Unsuccessfully predicted branches result in a four-to-five-cycle, branch-latency penalty in the super pipeline.

3.2.3 Instruction Memory Management Unit

For instruction pre-fetches, the Instruction Memory Management Unit (IMMU) controls logical-to-physical address translation, memory access permissions, memory-domain identifications, and attributes (governing operation of the instruction cache).

The IMMU contains a 32-entry, fully associative instruction-translation, look-aside buffer (ITLB) that has a round-robin replacement policy. ITLB entries zero through 30 can be locked.

When an instruction pre-fetch misses in the ITLB, the IMMU invokes an automatic table-walk mechanism that fetches an associated descriptor from memory and loads it into the ITLB. The descriptor contains information for logical-to-physical address translation, memory-access permissions, memory-domain identifications, and attributes governing operation of the I-cache.

The IMMU then continues the instruction pre-fetch by using the address translation just entered into the ITLB. When an instruction pre-fetch hits in the ITLB, the IMMU continues the pre-fetch using the address translation already resident in the ITLB.

Access permissions for each of up to 16 memory domains can be programmed. When an instruction pre-fetch is attempted to an area of memory in violation of access permissions, the attempt is aborted and a pre-fetch abort is sent to the core for exception processing. The IMMU and DMMU can be enabled or disabled together.

3.2.4 Data Memory Management Unit

For data fetches, the Data Memory Management Unit (DMMU) controls logical-to-physical address translation, memory-access permissions, memory-domain identifications, and attributes (governing operation of the data cache or mini-data cache and write buffer). The DMMU contains a 32-entry, fully associative data-translation, look-aside buffer (DTLB) that has a round-robin replacement policy. DTLB entries 0 through 30 can be locked.

When a data fetch misses in the DTLB, the DMMU invokes an automatic table-walk mechanism that fetches an associated descriptor from memory and loads it into the DTLB. The descriptor contains information for logical-to-physical address translation, memory-access permissions, memory-domain identifications, and attributes (governing operation of the D-cache or mini-data cache and write buffer).

The DMMU continues the data fetch by using the address translation just entered into the DTLB. When a data fetch hits in the DTLB, the DMMU continues the fetch using the address translation already resident in the DTLB.

Access permissions for each of up to 16 memory domains can be programmed. When a data fetch is attempted to an area of memory in violation of access permissions, the attempt is aborted and a data abort is sent to the core for exception processing.

The IMMU and DMMU can be enabled or disabled together.

3.2.5 Instruction Cache

The Instruction Cache (I-Cache) can contain high-use, multiple-code segments or entire programs, allowing the core access to instructions at core frequencies. This prevents core stalls caused by multi-cycle accesses to external memory.

The 32-Kbyte I-cache is 32-set/32-way associative, where each set contains 32 ways and each way contains a tag address, a cache line of instructions (eight 32-bit words and one parity bit per word), and a line-valid bit. For each of the 32 sets, 0 through 28 ways can be locked. Unlocked ways are replaceable via a round-robin policy.

The I-cache can be enabled or disabled. Attribute bits within the descriptors — contained in the ITLB of the IMMU — provide some control over an enabled I-cache.

When a needed line (eight 32-bit words) is not present in the I-cache, the line is fetched (critical word first) from memory via a two-level, deep-fetch queue. The fetch queue allows the next instruction to be accessed from the I-cache, but only when its data operands do not depend on the execution results of the instruction being fetched via the queue.

3.2.6 Data Cache

The Data Cache (D-Cache) can contain high-use data such as lookup tables and filter coefficients, allowing the core access to data at core frequencies. This prevents core stalls caused by multi-cycle accesses to external memory.

The 32-Kbyte D-cache is 32-set/32-way associative, where each set contains 32 ways and each way contains a tag address, a cache line (32 bytes with one parity bit per byte) of data, two dirty bits (one for each of two eight-byte groupings in a line), and one valid bit. For each of the 32 sets, zero through 28 ways can be locked, unlocked, or used as local SRAM. Unlocked ways are replaceable via a round-robin policy.

The D-cache (together with the mini-data cache) can be enabled or disabled. Attribute bits within the descriptors, contained in the DTLB of the DMMU, provide significant control over an enabled D-cache. These bits specify cache operating modes such as read and write allocate, write-back, write-through, and D-cache versus mini-data cache targeting.

The D-cache (and mini-data cache) work with the load buffer and pend buffer to provide “hit-under-miss” capability that allows the core to access other data in the cache after a “miss” is encountered. The D-cache (and mini-data cache) works in conjunction with the write buffer for data that is to be stored to memory.

3.2.7 Mini-Data Cache

The mini-data cache can contain frequently changing data streams such as MPEG video, allowing the core access to data streams at core frequencies. This prevents core stalls caused by multi-cycle accesses to external memory. The mini-data cache relieves the D-cache of data “thrashing” caused by frequently changing data streams.

The 2-Kbyte, mini-data cache is 32-set/two-way associative, where each set contains two ways and each way contains a tag address, a cache line (32 bytes with one parity bit per byte) of data, two dirty bits (one for each of two eight-byte groupings in a line), and a valid bit. The mini-data cache uses a round-robin replacement policy, and cannot be locked.

The mini-data cache (together with the D-cache) can be enabled or disabled. Attribute bits contained within a coprocessor register specify operating modes write and/or read allocate, write-back, and write-through.

The mini-data cache (and D-cache) work with the load buffer and pend buffer to provide “hit-under-miss” capability that allows the core to access other data in the cache after a “miss” is encountered. The mini-data cache (and D-cache) works in conjunction with the write buffer for data that is to be stored to memory.

3.2.8 Fill Buffer and Pend Buffer

The four-entry fill buffer (FB) works with the core to hold non-cacheable loads until the bus controller can act on them. The FB and the four-entry pend buffer (PB) work with the D-cache and mini-data cache to provide “hit-under-miss” capability, allowing the core to seek other data in the caches while “miss” data is being fetched from memory.

The FB can contain up to four unique “miss” addresses (logical), allowing four “misses” before the core is stalled. The PB holds up to four addresses (logical) for additional “misses” to those addresses that are already in the FB. A coprocessor register can specify draining of the fill and pend (write) buffers.

3.2.9 Write Buffer

The write buffer (WB) holds data for storage to memory until the bus controller can act on it. The WB is eight entries deep, where each entry holds 16 bytes. The WB is constantly enabled and accepts data from the core, D-cache, or mini-data cache.

Coprocessor 15, Register 1 specifies whether WB coalescing is enabled or disabled. When coalescing is disabled, stores to memory occur in program order — regardless of the attribute bits within the descriptors located in the DTLB.

When coalescing is enabled, the attribute bits within the descriptors located in the DTLB are examined to determine when coalescing is enabled for the destination region of memory. When coalescing is enabled in both CP15, R1 and the DTLB, data entering the WB can coalesce with any of the eight entries (16 bytes) and be stored to the destination memory region, but possibly out of program order.

Stores to a memory region specified to be non-cacheable and non-bufferable by the attribute bits within the descriptors located in the DTLB causes the core to stall until the store completes. A coprocessor register can specify draining of the write buffer.

3.2.10 Multiply-Accumulate Coprocessor

For efficient processing of high-quality, media-and-signal-processing algorithms, the Multiply-Accumulate Coprocessor (CP0) provides 40-bit accumulation of 16 x 16, dual-16 x 16 (SIMD), and 32 x 32 signed multiplies. Special MAR and MRA instructions are implemented to move the 40-bit accumulator to two core-general registers (MAR) and move two core-general registers to the 40-bit accumulator (MRA). The 40-bit accumulator can be stored or loaded to or from D-cache, mini-data cache, or memory using two STC or LDC instructions.

The 16 x 16 signed multiply-accumulates (MIAxy) multiply either the high/high, low/low, high/low, or low/high 16 bits of a 32-bit core general register (multiplier) and another 32-bit core general register (multiplicand) to produce a full, 32-bit product that is sign-extended to 40 bits and added to the 40-bit accumulator.

Dual-signed, 16 x 16 (SIMD) multiply-accumulates (MIAPH) multiply the high/high and low/low 16-bits of a packed 32-bit, core-general register (multiplier) and another packed 32-bit, core-general register (multiplicand) to produce two 16-bits products that are both sign-extended to 40 bits and added to the 40-bit accumulator.

The 32 x 32 signed multiply-accumulates (MIA) multiply a 32-bit, core-general register (multiplier) and another 32-bit, core-general register (multiplicand) to produce a 64-bit product where the 40 LSBs are added to the 40-bit accumulator. The 16 x 32 versions of the 32 x 32 multiply-accumulate instructions complete in a single cycle.

3.2.11 Performance Monitoring Unit

The performance monitoring unit (PMU) contains two 32-bit, event counters and one 32-bit, clock counter. The event counters can be programmed to monitor I-cache hit rate, data caches hit rate, ITLB hit rate, DTLB hit rate, pipeline stalls, BTB prediction hit rate, and instruction execution count.

3.2.12 Debug Unit

The debug unit is accessed through the JTAG port. The industry-standard, IEEE 1149.1 JTAG port consists of a test access port (TAP) controller, boundary-scan register, instruction and data registers, and dedicated signals TDI, TDO, TCK, TMS, and TRST#.

The debug unit — when used with debugger application code running on a host system outside of the Intel XScale core — allows a program, running on the Intel XScale core, to be debugged. It allows the debugger application code or a debug exception to stop program execution and redirect execution to a debug-handling routine.

Debug exceptions are instruction breakpoint, data breakpoint, software breakpoint, external debug breakpoint, exception vector trap, and trace buffer full breakpoint. Once execution has stopped, the debugger application code can examine or modify the core's state, coprocessor state, or memory. The debugger application code can then restart program execution.

The debug unit has two hardware-instruction, break point registers; two hardware, data-breakpoint registers; and a hardware, data-breakpoint control register. The second data-breakpoint register can be alternatively used as a mask register for the first data-breakpoint register.

A 256-entry trace buffer provides the ability to capture control flow messages or addresses. A JTAG instruction (LDIC) can be used to download a debug handler via the JTAG port to the mini-instruction cache (the I-cache has a 2-Kbyte, mini-instruction cache, like the mini-data cache, that is used only to hold a debug handler).

4.0 Package Information

This section contains information on the following topics:

- "Package Description" which includes "Package Drawings", "Package Markings", and "Part Numbers"
- "Functional Signal Definitions" on page 46
- "Signal-Pin Descriptions" on page 89
- "Package Thermal Specifications" on page 114

4.1 Package Description

The IXP45X/IXP46X network processors are built using a 544-ball, plastic ball grid array (PBGA) package with a drop-in heat spreader (H).

For all extended temperature products and the 667-MHz speed option of the commercial temperature product, a 10-mm-high, thermal-adhesive-based heat sink will be required. The heat sink does not force the addition of any surface area to the board design.

4.1.1 Package Drawings

The package is shown in Figure 5 and Figure 6.

Figure 5. 544-Pin Lead PBGA Package — First of Two Drawings

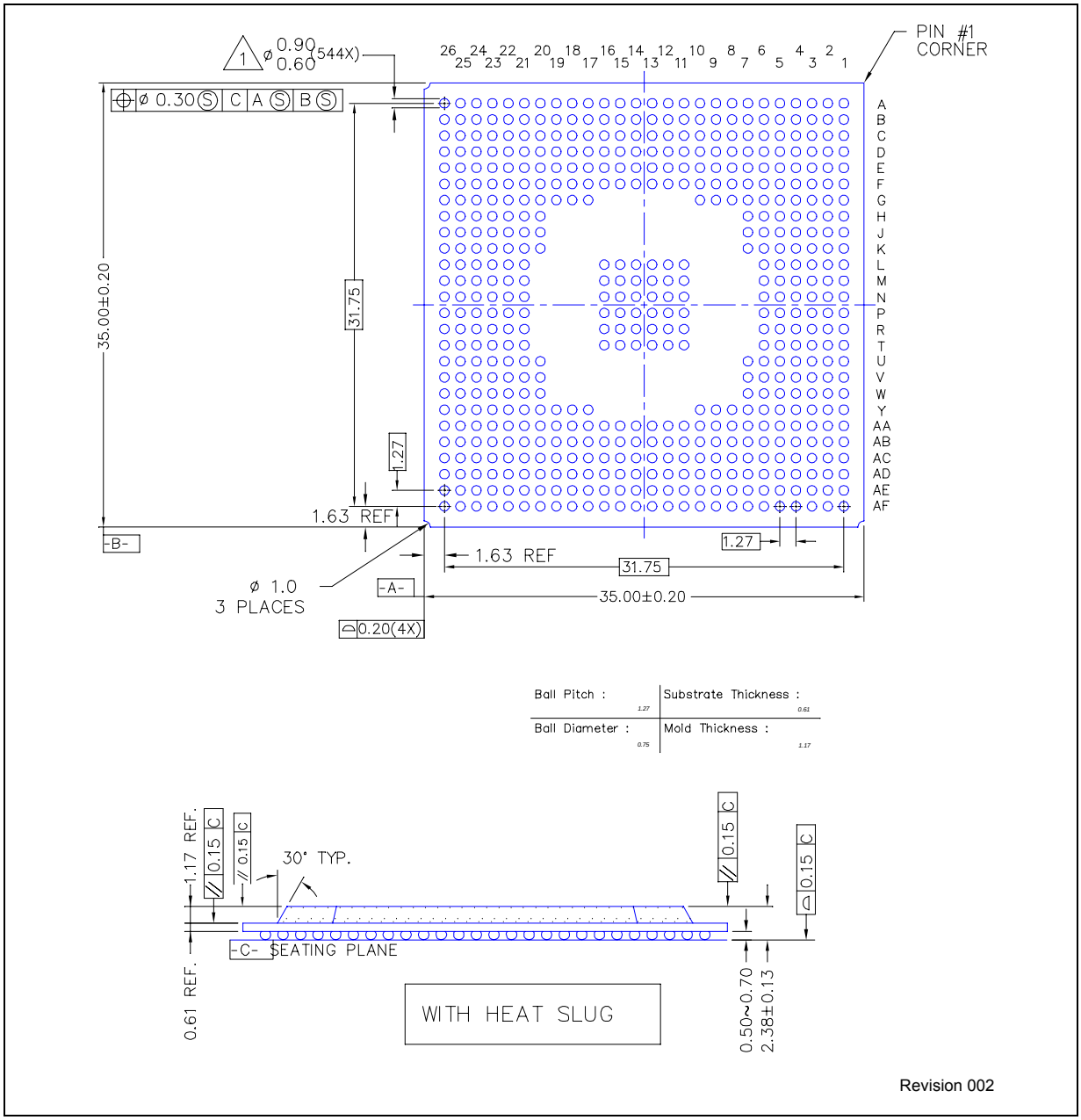
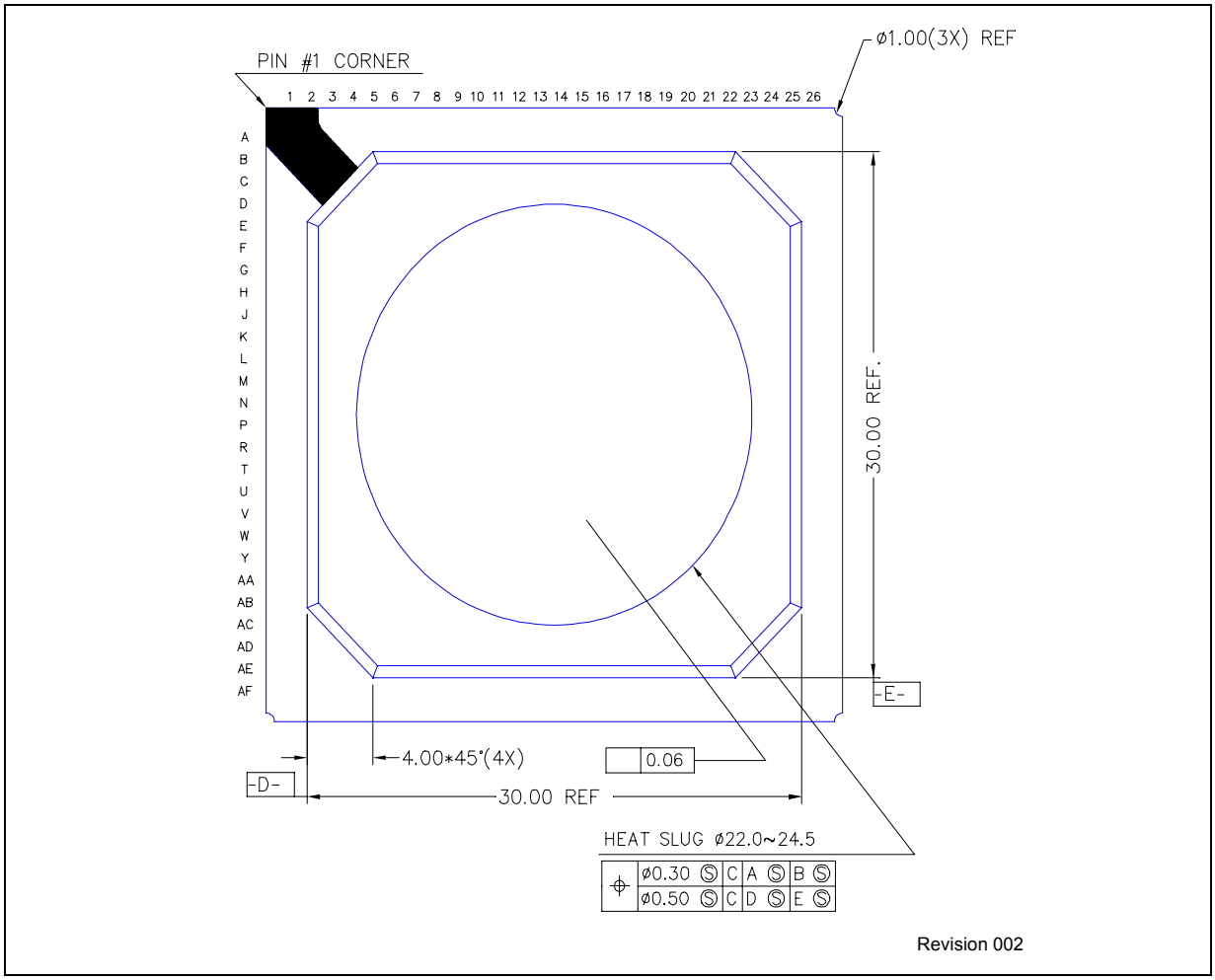
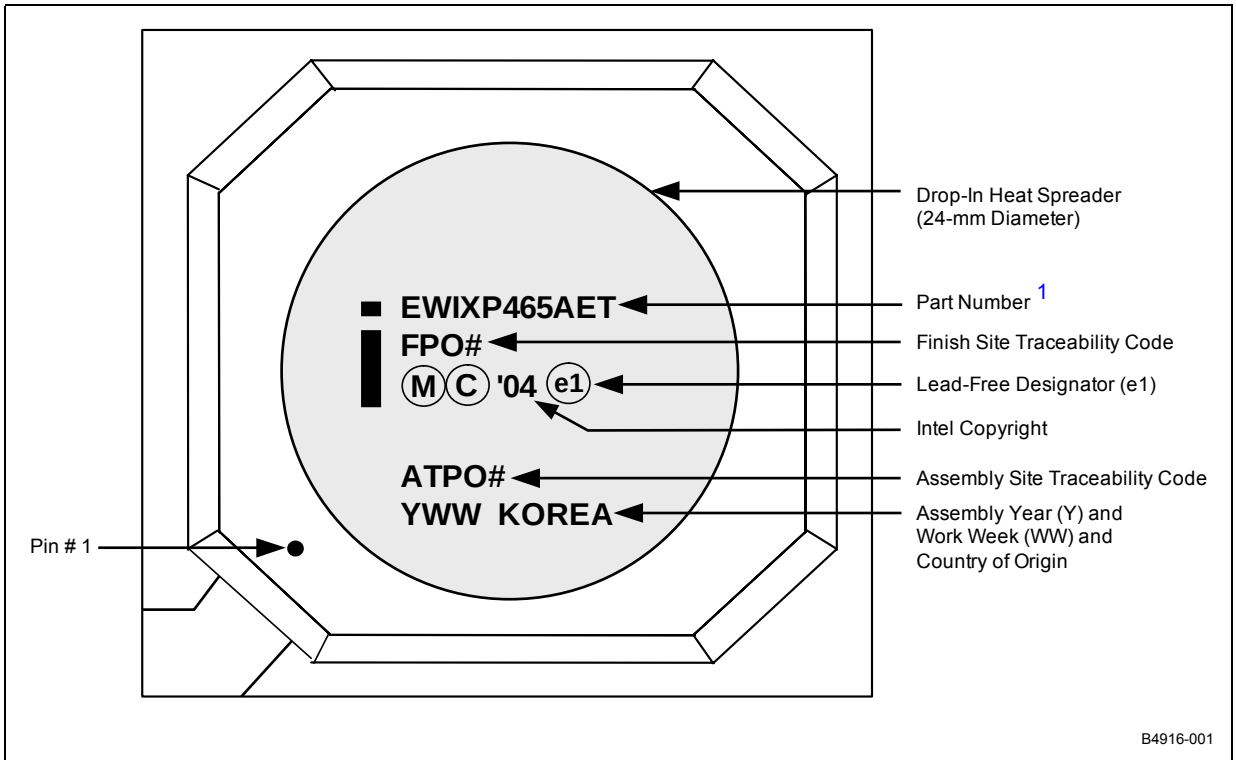


Figure 6. 544-Pin Lead PBGA Package — Second of Two Drawings



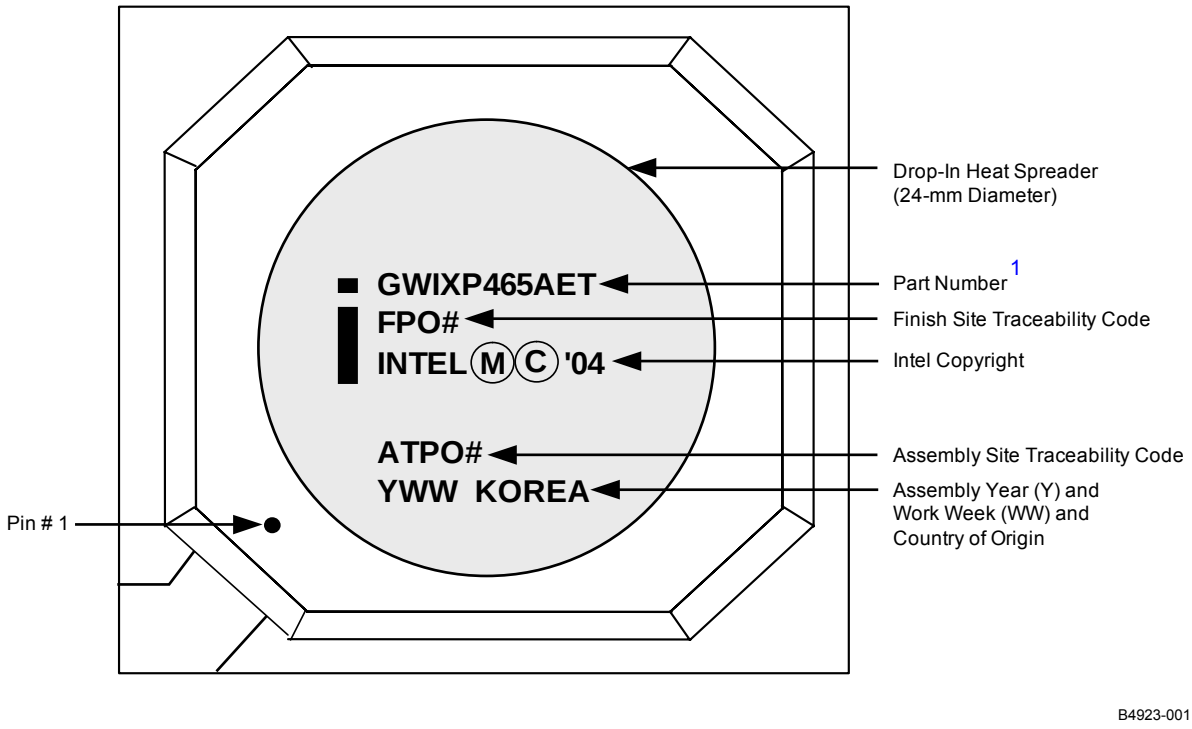
4.1.2 Package Markings

Figure 7. Package Markings: Intel® IXP45X and Intel® IXP46X Product Line of Network Processors— Extended and Commercial Temperature, Lead-Free / Compliant with Standard for Restriction on the Use of Hazardous Substances (RoHS)



Notes:
1. **Part Number** field — For the different part numbers of Intel® IXP45X and Intel® IXP46X Product Line of Network Processors, see Section 4.1.3.
2. Package ball counts — Intel® IXP45X and Intel® IXP46X Product Line of Network Processors have a ball count of 544.
3. Drawing is not to scale. Marking content is an example.

Figure 8. Package Markings: Intel® IXP45X and Intel® IXP46X Product Line of Network Processors — Commercial and Extended Temperature, Lead-Based



Notes:
1. **Part Number** field — For the different part numbers of Intel® IXP45X and Intel® IXP46X Product Line of Network Processors, see Section 4.1.3.
2. Package ball counts — Intel® IXP45X and Intel® IXP46X Product Line of Network Processors have a ball count of 544.
3. Drawing is not to scale. Marking content is an example.

4.1.3 Part Numbers

The tables in this section list the part numbers for the IXP46X product line of network processors (Table 6 and Table 7) and the IXP45X product line of network processors (Table 8 and Table 9).

Table 6. Intel® IXP46X Product Line Part Numbers: Lead (pb) Packaging (Sheet 1 of 2)

Device	Stepping	Speed (MHz)	Part Number	Temperature Offering
IXP465	A0	667	GWIXP465AE	Commercial
IXP465	A0	533	GWIXP465AD	Commercial
IXP465	A0	400	GWIXP465AC	Commercial
IXP465	A0	266	GWIXP465AB	Commercial
IXP460	A0	667	GWIXP460AE	Commercial
IXP460	A0	533	GWIXP460AD	Commercial
IXP460	A0	400	GWIXP460AC	Commercial

Table 6. Intel® IXP46X Product Line Part Numbers: Lead (pb) Packaging (Sheet 2 of 2)

Device	Stepping	Speed (MHz)	Part Number	Temperature Offering
IXP460	A0	266	GWIXP460AB	Commercial
IXP465	A0	667	GWIXP465AET	Extended
IXP465	A0	533	GWIXP465ADT	Extended
IXP465	A0	400	GWIXP465ACT	Extended
IXP465	A0	266	GWIXP465ABT	Extended
IXP460	A0	667	GWIXP460AET	Extended
IXP460	A0	533	GWIXP460ADT	Extended
IXP460	A0	400	GWIXP460ACT	Extended
IXP460	A0	266	GWIXP460ABT	Extended

Table 7. Intel® IXP46X Product Line Part Numbers: Lead Free (pb-free) Packaging

Device	Stepping	Speed (MHz)	Part Number	Temperature Offering
IXP465	A0	667	EWIXP465AE	Commercial
IXP465	A0	533	EWIXP465AD	Commercial
IXP465	A0	400	EWIXP465AC	Commercial
IXP465	A0	266	EWIXP465AB	Commercial
IXP460	A0	667	EWIXP460AE	Commercial
IXP460	A0	533	EWIXP460AD	Commercial
IXP460	A0	400	EWIXP460AC	Commercial
IXP460	A0	266	EWIXP460AB	Commercial
IXP465	A0	667	EWIXP465AET	Extended
IXP465	A0	533	EWIXP465ADT	Extended
IXP465	A0	400	EWIXP465ACT	Extended
IXP465	A0	266	EWIXP465ABT	Extended
IXP460	A0	667	EWIXP460AET	Extended
IXP460	A0	533	EWIXP460ADT	Extended
IXP460	A0	400	EWIXP460ACT	Extended
IXP460	A0	266	EWIXP460ABT	Extended

Table 8. Intel® IXP45X Product Line Part Numbers: Lead (pb) Packaging (Sheet 1 of 2)

Device	Stepping	Speed (MHz)	Part Number	Temperature Offering
IXP455	A0	533	GWIXP455AD	Commercial
IXP455	A0	400	GWIXP455AC	Commercial
IXP455	A0	266	GWIXP455AB	Commercial

Table 8. Intel® IXP45X Product Line Part Numbers: Lead (pb) Packaging (Sheet 2 of 2)

Device	Stepping	Speed (MHz)	Part Number	Temperature Offering
IXP455	A0	533	GWIXP455ADT	Extended
IXP455	A0	400	GWIXP455ACT	Extended
IXP455	A0	266	GWIXP455ABT	Extended

Table 9. Intel® IXP45X Product Line Part Numbers: Lead Free (pb-free) Packaging

Device	Stepping	Speed (MHz)	Part Number	Temperature Offering
IXP455	A0	533	EWIXP455AD	Commercial
IXP455	A0	400	EWIXP455AC	Commercial
IXP455	A0	266	EWIXP455AB	Commercial
IXP455	A0	533	EWIXP455ADT	Extended
IXP455	A0	400	EWIXP455ACT	Extended
IXP455	A0	266	EWIXP455ABT	Extended

4.2 Functional Signal Definitions

The signal definition tables list pull-up and pull-down resistor recommendations when the particular *enabled* interface is not being used in the application. These external resistor requirements are only needed if the particular model of IXP45X/IXP46X network processors has the particular interface *enabled* and the interface is not required in the application.

Warning: **None** of the IXP45X/IXP46X network processors' I/O pins are 5-V tolerant.

Disabled features within the IXP45X/IXP46X network processors do not require external resistors, as the processor will have internal pull-up or pull-down resistors enabled as part of the *disabled* interface. To determine which interfaces are not enabled within the IXP45X/IXP46X network processors, see Table 1 on page 14.

Table 10 presents the legend for interpreting the **Type** field in the other tables in this section of the document.

Table 10. Signal Type Definitions (Sheet 1 of 2)

Symbol	Description
I	Input pin only
O	Output pin only
I/O	Pin can be either an input or output
OD	Open Drain pin
PWR	Power pin
GND	Ground pin
1	Driven to Vcc

Table 10. Signal Type Definitions (Sheet 2 of 2)

Symbol	Description
0	Driven to Vss
X	Driven to unknown state
ID	Input is disabled
H	Pulled up to Vcc
L	Pulled to Vss
PD	Pull-up Disabled
Z	Output Disabled
VO	A valid output level is driven, allowed states -- 1, 0, H
VB	Valid level on the signal, allowed states - 1, 0, H, Z
VI	Need to drive a valid input level, allowed states - 1, 0, H, Z
VOD	Valid Open Drain output, allowed states are 0 or Z
PE	Pull-up Enabled, equivalent to H
TRI	Output Only/Tristatable
ePU	External 10K ohm Pull-Up is required on the board
N/C	No Connect
-	Pin must be connected as described

This section includes the following tables:

Table 11. DDR SDRAM Interface (Sheet 1 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
DDRI_CK[2:0]	Z	0	VO	VO	O	DDR SDRAM Clock Out — Provide the positive differential clocks to the external SDRAM memory subsystem.
DDRI_CK_N[2:0]	Z	1	VO	VO	O	DDR SDRAM Clock Out — Provide the negative differential clocks to the external SDRAM memory subsystem.
DDRI_CS_N[1:0]	Z	Z	VO	VO	O	Chip Select — Must be asserted for all transactions to the DDR SDRAM device. One per bank.
DDRI_RAS_N	Z	Z	VO	VO	O	Row Address Strobe — Indicates that the current address on DDRI_MA[13:0] is the row.
DDRI_CAS_N	Z	Z	VO	VO	O	Column Address Strobe — Indicates that the current address on DDRI_MA[13:0] is the column.
DDRI_WE_N	Z	Z	VO	VO	O	Write Strobe — Defines whether or not the current operation by the DDR SDRAM is to be a read or a write.
DDRI_DM[4:0]	Z	Z	VO	VO	O	Data Bus Mask — Controls the DDR SDRAM data input buffers. Asserting DDRI_WE_N causes the data on DDRI_DQ[31:0] and DDRI_CB[7:0] to be written into the DDR SDRAM devices. DDRI_DM[4:0] controls this operation on a per byte basis. DDRI_DM[3:0] are intended to correspond to each byte of a word of data. DDRI_DM[4] is intended to be utilized for the ECC byte of data.
DDRI_BA[1:0]	Z	Z	VO	VO	O	DDR SDRAM Bank Selects — Controls which of the internal DDR SDRAM banks to read or write. DDRI_BA[1:0] are used for all technology types supported.
DDRI_MA[13:0]	Z	Z	VO	VO	O	Address bits 13 through 0 — Indicates the row or column to access depending on the state of DDRI_RAS_N and DDRI_CAS_N.
DDRI_DQ[31:0]	Z	VB	VB	VB	I/O	Data Bus — 32-bit wide data bus.
DDRI_CB[7:0]	Z	VB	VB	VB	I/O	ECC Bus — Eight-bit error correction code which accompanies the data on DDRI_DQ[31:0]. When ECC is disabled and not being used in a system design, these signals are not required for any connection.
DDRI_DQS[4:0]	Z	VB	VB	VB	I/O	Data Strobes Differential — Strobes that accompany the data to be read or written from the DDR SDRAM devices. Data is sampled on the negative and positive edges of these strobes. DDRI_DQS[3:0] are intended to correspond to each byte of a word of data. DDRI_DQS4] is intended to be utilized for the ECC byte of data.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 11. DDR SDRAM Interface (Sheet 2 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
DDRI_CKE[1:0]	Z	b'00	VO	VO	O	Clock enables — One clock after DDRI_CKE[1:0] is de-asserted, data is latched on DQ[31:0] and DDRI_CB[7:0]. Burst counters within DDR SDRAM device are not incremented. De-asserting this signal places the DDR SDRAM in self-refresh mode. For normal operation, DDRI_CKE[1:0] must be asserted.
DDRI_RCVENOUT_N	Z	1	VO	VO	O	RECEIVE ENABLE OUT must be connected to DDRI_RCVENIN_N signal of the IXP45X/ IXP46X network processors and the propagation delay of the trace length must be matched to the clock trace plus the average DQ Traces.
DDRI_RCVENIN_N	Z	VI	VI	VI	I	RECEIVE ENABLE IN provides delay information for enabling the input receivers and must be connected to the DDRI_RCVENOUT_N signal of the IXP45X/IXP46X network processors.
DDRI_RCOMP	Tied off to a resistor	Tied off to a resistor	Tied off to a resistor	Tied off to a resistor	O	20 Ohm 1% tolerance Resistor connected to ground used for process/temperature adjustments.
DDRI_VREF	VCCM/ 2	VCCM/2	VCCM/2	VCCM/2	I	DDR SDRAM Voltage Reference — is used to supply the reference voltage to the differential inputs of the memory controller pins.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 12. PCI Controller (Sheet 1 of 5)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
PCI_AD[31:0]	Z	Z	VB	VB	I/O	PCI Address/Data bus used to transfer address and bidirectional data to and from multiple PCI devices. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_CBE_N[3:0]	Z	Z	VB	VB	I/O	PCI Command/Byte Enables is used as a command word during PCI address cycles and as byte enables for data cycles. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_PAR	Z	Z	VB	VB	I/O	PCI Parity used to check parity across the 32 bits of PCI_AD and the four bits of PCI_CBE_N. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_FRAME_N	Z	Z	VB	VB	I/O	PCI Cycle Frame used to signify the beginning and duration of a transaction. The signal will be inactive prior to or during the final data phase of a given transaction. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 12. PCI Controller (Sheet 2 of 5)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
PCI_TRDY_N	Z	Z	VB	VB	I/O	PCI Target Ready informs that the target of the PCI bus is ready to complete the current data phase of a given transaction. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_IRDY_N	Z	Z	VB	VB	I/O	PCI Initiator Ready informs the PCI bus that the initiator is ready to complete the transaction. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_STOP_N	Z	Z	VB	VB	I/O	PCI Stop indicates that the current target is requesting the current initiator to stop the current transaction. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_PERR_N	Z	Z	VB	VB	I/O	PCI Parity Error asserted when a PCI parity error is detected — between the PCI_PAR and associated information on the PCI_AD bus and PCI_CBE_N — during all PCI transactions, except for Special Cycles. The agent receiving data will drive this signal. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 12. PCI Controller (Sheet 3 of 5)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
PCI_SERR_N	Z	Z	VB	VB	I/OD	PCI System Error asserted when a parity error occurs on special cycles or any other error that will cause the PCI bus not to function properly. This signal can function as an input or an open drain output. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_DEVSEL_N	Z	Z	VB	VB	I/O	PCI Device Select: • When used as an output, PCI_DEVSEL_N indicates that device has decoded that address as the target of the requested transaction. • When used as an input, PCI_DEVSEL_N indicates if any device on the PCI bus exists with the given address. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_IDSEL	Z	Z	VI	VI	I	PCI Initialization Device Select is a chip select during configuration reads and writes. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_REQ_N[3:1]	Z	Z	VI	VI	I	PCI arbitration request: Used by the internal PCI arbiter to allow an agent to request the PCI bus. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 12. PCI Controller (Sheet 4 of 5)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
PCI_REQ_N[0]	Z	Z	VI	VI / VO	I/O	PCI arbitration request: • When configured as an input (PCI arbiter enabled), the internal PCI arbiter will allow an agent to request the PCI bus. • When configured as an output (PCI arbiter disabled), the pin will be used to request access to the PCI bus from an external arbiter. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_GNT_N[3:1]	Z	Z	VO	VO	O	PCI arbitration grant: Generated by the internal PCI arbiter to allow an agent to claim control of the PCI bus.
PCI_GNT_N[0]	Z	Z	VO	VI / VO	I/O	PCI arbitration grant: • When configured as an output (PCI arbiter enabled), the internal PCI arbiter to allow an agent to claim control of the PCI bus. • When configured as an input (PCI arbiter disabled), the pin will be used to claim access of the PCI bus from an external arbiter. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 12. PCI Controller (Sheet 5 of 5)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
PCI_INTA_N	Z	Z	Z	VOD	O/D	PCI interrupt: Used to request an interrupt. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the PCI soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
PCI_CLKIN	Z	VI	VI	VI	I	PCI Clock: Clock provides timing for all transactions on PCI. All PCI signals — except INTA#, INTB#, INTC#, and INTD# — are sampled on the rising edge of CLK and timing parameters are defined with respect to this edge. The PCI clock rate can operate at up to 66 MHz. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 13. High-Speed, Serial Interface 0 (Sheet 1 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
HSS_TXFRAME0	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) transmit frame signal can be configured as an input or an output to allow an external source become synchronized with the transmitted data. Often known as a Frame Sync signal. Configured as an input upon reset. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
HSS_TXDATA0	Z	Z	VOD	VOD	OD	Transmit data out. Open Drain output. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor to V_{CCP}. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
HSS_TXCLK0	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) transmit clock signal can be configured as an input or an output. The clock can be a frequency ranging from 512 KHz to 8.192 MHz. Used to clock out the transmitted data. Configured as an input upon reset. Frame sync and data can be selected to be generated on the rising or falling edge of the transmit clock.
HSS_RXFRAME0	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) receive frame signal can be configured as an input or an output to allow an external source to become synchronized with the received data. Often known as a Frame Sync signal. Configured as an input upon reset. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 13. High-Speed, Serial Interface 0 (Sheet 2 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
HSS_RXDATA0	Z	VI	VI	VI	I	Receive data input. Can be sampled on the rising or falling edge of the receive clock. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
HSS_RXCLK0	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) receive clock signal can be configured as an input or an output. The clock can be from 512 KHz to 8.192 MHz. Used to sample the received data. Configured as an input upon reset. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 14. High-Speed, Serial Interface 1 (Sheet 1 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
HSS_TXFRAME1	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) transmit frame signal can be configured as an input or an output to allow an external source to be synchronized with the transmitted data. Often known as a Frame Sync signal. Configured as an input upon reset. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
HSS_TXDATA1	Z	Z	VOD	VOD	OD	Transmit data out. Open Drain output. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor to V_{CCP}. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
HSS_TXCLK1	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) transmit clock signal can be configured as an input or an output. The clock can be a frequency ranging from 512 KHz to 8.192 MHz. Used to clock out the transmitted data. Configured as an input upon reset. Frame sync and Data can be selected to be generated on the rising or falling edge of the transmit clock. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.
HSS_RXFRAME1	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) receive frame signal can be configured as an input or an output to allow an external source to be synchronized with the received data. Often known as a Frame Sync signal. Configured as an input upon reset. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 14. High-Speed, Serial Interface 1 (Sheet 2 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
HSS_RXDATA1	Z	VI	VI	VI	I	Receive data input. Can be sampled on the rising or falling edge of the receive clock. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the HSS soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
HSS_RXCLK1	Z	Z	VB	VB	I/O	The High-Speed Serial (HSS) receive clock signal can be configured as an input or an output. The clock can be from 512 KHz to 8.192 MHz. Used to sample the received data. Configured as an input upon reset. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 1 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_OP_CLK / ETHA_TXCLK	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA Transmit clock input. Also known as UTP_TX_CLK. This signal is used to synchronize all UTOPIA transmit outputs to the rising edge of the UTP_OP_CLK. **MII Mode of Operation:** Externally supplied transmit clock. • 25 MHz for 100 Mbps operation • 2.5 MHz for 10 Mbps **SMII Mode of Operation:** Not Used. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.
UTP_OP_FCO	Z	Z	Z	VO	TRI	UTOPIA flow control output signal. Also known as the TXENB_N signal. Used to inform the selected PHY that data is being transmitted to the PHY. Placing the PHY's address on the UTP_OP_ADDR — and bringing UTP_OP_FCO to logic 1, during the current clock — followed by the UTP_OP_FCO going to a logic 0, on the next clock cycle, selects which PHY is active in MPHY mode. In SPHY configurations, UTP_OP_FCO is used to inform the PHY that the processor is ready to send data. This signal must be tied to Vcc with an external 10-KΩ resistor.
UTP_OP_SOC	Z	Z	Z	VO	TRI	Start of Cell. Also known as TX_SOC. Active high signal is asserted when UTP_OP_DATA contains the first valid byte of a transmitted cell. This signal must be tied to Vss with an external 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 2 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_OP_DATA[3:0] / ETHA_TXDATA[3:0]	Z	Z	Z	VO	TRI	**UTOPIA Mode of Operation:** UTOPIA output data. Also known as UTP_TX_DATA. Used to send data from the processor to an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Transmit data bus to PHY, asserted synchronously with respect to ETHA_TXCLK. This MAC interface does not contain hardware hashing capabilities local to the interface. In this mode of operation the pins represented by this interface are ETHA_TXDATA3:0]. **SMII mode of operation:** Not used.
UTP_OP_DATA[4] / ETHA_TXEN	Z	Z	Z	VO	TRI	**UTOPIA Mode of Operation:** UTOPIA output data. Also known as UTP_TX_DATA. Used to send data from the processor to an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Indicates that the PHY is being presented with nibbles on the MII interface. Asserted synchronously, with respect to ETHA_TXCLK, at the first nibble of the preamble, and remains asserted until all the nibbles of a frame are presented. This MAC does not contains hardware hashing capabilities local to the interface. **SMII mode of operation:** Not used.
UTP_OP_DATA[6:5]	Z	Z	Z	VO	TRI	**UTOPIA Mode of Operation:** UTOPIA output data. Also known as UTP_TX_DATA. Used to send data from the processor to an ATM UTOPIA Level 2-compliant PHY.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 3 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_OP_DATA[7] / SMII_TXDATA[4]	Z	Z	Z	VO	TRI	**UTOPIA Mode of Operation:** UTOPIA output data. Also known as UTP_TX_DATA. Used to send data from the processor to an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Not used. **SMII mode of operation:** Output data for SMII interface number four. The data on this signal is transmitted synchronously with respect to the rising edge of SMII_CLK when operating as an SMII interface and synchronously with respect to the rising edge of SMII_TXCLK when operating as a Source Synchronous SMII interface
UTP_OP_ADDR[4:0]	Z	Z	Z	VO	O	Transmit PHY address bus. Used by the processor when operating in MPHY mode to poll and select a single PHY at any given time. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
UTP_OP_FCI	Z	VI	VI	VI	I	UTOPIA Output data flow control input: Also known as the TXFULL/CLAV signal. Used to inform the processor of the ability of each polled PHY to receive a complete cell. For cell-level flow control in an MPHY environment, TxClav is an active high tri-stateable signal from the MPHY to ATM layer. The UTP_OP_FCI, which is connected to multiple MPHY devices, will see logic high generated by the PHY, one clock after the given PHY address is asserted — when a full cell can be received by the PHY. The UTP_OP_FCI will see a logic low generated by the PHY one clock cycle, after the PHY address is asserted — if a full cell cannot be received by the PHY. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 4 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_IP_CLK / ETHA_RXCLK	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA Receive clock input. Also known as UTP_RX_CLK. This signal is used to synchronize all UTOPIA-received inputs to the rising edge of the UTP_IP_CLK. **MII Mode of Operation:** Externally supplied receive clock. • 25 MHz for 100 Mbps operation • 2.5 MHz for 10 Mbps This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** Not used. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.
UTP_IP_FCI	Z	VI	VI	VI	I	UTOPIA Input Data flow control input signal. Also known as RXEMPTY/CLAV. Used to inform the processor of the ability of each polled PHY to send a complete cell. For cell-level flow control in an MPHY environment, RxClav is an active high tri-stateable signal from the MPHY to ATM layer. The UTP_IP_FCI, which is connected to multiple MPHY devices, will see logic high generated by the PHY, one clock after the given PHY address is asserted, when a full cell can be received by the PHY. The UTP_IP_FCI will see a logic low generated by the PHY, one clock cycle after the PHY address is asserted if a full cell cannot be received by the PHY. In SPHY mode, this signal is used to indicate to the processor that the PHY has an octet or cell available to be transferred to the processor. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 5 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_IP_SOC	Z	VI	VI	VI	I	Start of Cell. RX_SOC Active-high signal that is asserted when UTP_IP_DATA contains the first valid byte of a transmitted cell. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
UTP_IP_DATA[3:0] / ETHA_RXDATA[3:0]	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA input data. Also known as RX_DATA. Used by to the processor to receive data from an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Receive data bus from the PHY, asserted synchronously with respect to ETHA_RXCLK. **SMII mode of operation:** Not used. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 6 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_IP_DATA[4] / ETHA_RXDV	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA input data. Also known as RX_DATA. Used by to the processor to receive data from an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Receive data valid, used to inform the MII interface that the Ethernet PHY is sending data. This MAC does not contains hardware hashing capabilities local to the interface. **SMII mode of operation:** Not used. When this interface/signal is enabled and is not being used in a system design, the interface/ signal should be pulled high with a 10-KΩresistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 7 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_IP_DATA[5] / ETHA_COL	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA input data. Also known as RX_DATA. Used by the processor to receive data from an ATM UTOPIA Level 2-compliant PHY. • When NPE A is configured in UTOPIA mode of operation and the signal is not being used, it should be pulled high through a 10-KΩ resistor. **MII Mode of Operation:** Asserted by the PHY when a collision is detected by the PHY. • When NPE A is configured in MII mode of operation and the signal is not being used, it should be pulled low through a 10-KΩ resistor. **SMII Mode of Operation:** Not used. • When NPE A is configured in SMII mode of operation, this signal must be pulled high through a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/ or the NPE-A Ethernet soft fuse (refer to the Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*) and is not being used in a system design, this interface/ signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 8 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_IP_DATA[6] / ETHA_CRS	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA input data. Also known as RX_DATA. Used by to the processor to receive data from an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Asserted by the PHY when the transmit medium or receive medium are active. De-asserted when both the transmit and receive medium are idle. Remains asserted throughout the duration of collision condition. PHY asserts CRS asynchronously and de-asserts synchronously with respect to ETHA_RXCLK. **SMII mode of operation:** Not used. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
UTP_IP_DATA[7] / SMII_RXDATA[4]	Z	VI	VI	VI	I	**UTOPIA Mode of Operation:** UTOPIA input data. Also known as RX_DATA. Used by to the processor to receive data from an ATM UTOPIA Level 2-compliant PHY. **MII Mode of Operation:** Not Used. **SMII mode of operation:** Input data for SMII interface number four. The data on this signal is received synchronously with respect to the rising edge of SMII_CLK when operating as an SMII interface and synchronously with respect to the rising edge of SMII_RXCLK when operating as a Source Synchronous SMII interface. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the UTOPIA and/or the NPE-A Ethernet soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 15. UTOPIA Level 2/MII_A/ SMII[4] Interface (Sheet 9 of 9)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
UTP_IP_ADDR[4:0]	Z	Z	Z	VO	O	Receive PHY address bus. Used by the processor when operating in MPHY mode to poll and select a single PHY at any one given time.
UTP_IP_FCO	Z	Z	Z	VO	TRI	UTOPIA Input Data Flow Control Output signal: Also known as the RX_ENB_N. In SPHY configurations, UTP_IP_FCO is used to inform the PHY that the processor is ready to accept data. In MPHY configurations, UTP_IP_FCO is used to select which PHY will drive the UTP_RX_DATA and UTP_RX_SOC signals. The PHY is selected by placing the PHY's address on the UTP_IP_ADDR and bringing UTP_OP_FCO to logic 1 during the current clock, followed by the UTP_OP_FCO going to a logic 0 on the next clock cycle. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† For information on selecting the desired interface, see the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*.

Table 16. MII/SMII Interfaces (Sheet 1 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHB_TXCLK / SMII_CLK	Z	VI	VI	VI	I	**MII Mode of Operation:** Externally supplied transmit clock. • 25 MHz for 100 Mbps operation • 2.5 MHz for 10 Mbps This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** 125-MHz input clock used as the reference clock when operating in SMII or Source Synchronous SMII mode of operation. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.
ETHB_TXDATA[3:0] / SMII_TXDATA[0] / SMII_TXDATA[1] / SMII_TXDATA[2] / SMII_TXDATA[3]	Z	0	VO	VO	O	**MII Mode of Operation:** Transmit data bus to PHY, asserted synchronously with respect to ETHB_TXCLK. This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** Each SMII_TXDATA line is an interface to a separate physical port. ETHB_TXDATA[3] is multiplexed with SMII_TXDATA[3], ETHB_TXDATA[2] is multiplexed with SMII_TXDATA[2], ETHB_TXDATA[1] is multiplexed with SMII_TXDATA[1], ETHB_TXDATA[0] is multiplexed with SMII_TXDATA[0] The data on these signal are transmitted synchronously with respect to the rising edge of SMII_CLK when operating as an SMII interface and synchronously with respect to the rising edge of SMII_TXCLK when operating as a Source Synchronous SMII interface

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 2 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHB_TXEN / SMII_TXCLK	Z	0	VO	VO	O	**MII Mode of Operation:** Indicates that the PHY is being presented with nibbles on the MII interface. Asserted synchronously, with respect to ETHB_TXCLK, at the first nibble of the preamble and remains asserted until all the nibbles of a frame are presented. This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** 125-MHz clock that is used to send data to a physical interface when operating in a Source Synchronous SMII mode of operation.
ETHB_RXCLK / SMII_RXCLK	Z	VI	VO	VO	I	**MII Mode of Operation:** Externally supplied receive clock. • 25 MHz for 100 Mbps operation • 2.5 MHz for 10 Mbps This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** 125-MHz clock that is used to sample data being received from a physical interface when operating in a Source Synchronous SMII mode of operation. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 3 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHB_RXDATA[3:0] / SMII_RXDATA[0] / SMII_RXDATA[1] / SMII_RXDATA[2] / SMII_RXDATA[3]	Z	VI	VI	VI	I	**MII Mode of Operation:** Receive data bus from PHY, data sampled synchronously with respect to ETHB_RXCLK. This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** Each SMII_RXDATA line is a separate physical port ETHB_RXDATA[3] is multiplexed with SMII_RXDATA[3], ETHB_RXDATA[2] is multiplexed with SMII_RXDATA[2], ETHB_RXDATA[1] is multiplexed with SMII_RXDATA[1], ETHB_RXDATA[0] is multiplexed with SMII_RXDATA[0] The data on these signal are received synchronously with respect to the rising edge of SMII_CLK when operating as an SMII interface and synchronously with respect to the rising edge of SMII_RXCLK when operating as a Source Synchronous SMII interface When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the NPE-B Ethernet 0 and/or the NPE Ethernet 1-3 soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection. One special configuration exists for the board designer. When NPE B is configured in SMII mode of operation and a subset of the four SMII ports are utilized (i.e. All four are enabled but only two are being connected). The unused inputs must be tied high with a 10-KΩ resistor.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 4 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHB_RXDV / SMII_RXSYNC	Z	VI	VI	VI	I	**MII Mode of Operation:** Receive data valid, used to inform the MII interface that the Ethernet PHY is sending data. This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** In Source Synchronous mode of operation, this signal is an input from a synchronous pulse created once every 10 SMII_RXCLK reference clocks to signal the start of the next 10 bits of data to be received. SMII_RXCLK Reference clock operates at 125MHz. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the NPE-B Ethernet 0 and/or the NPE Ethernet 1-3 soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
ETHB_COL	Z	VI	VI	VI	I	**MII Mode of Operation:** Asserted by the PHY when a collision is detected by the PHY. This MAC interface does not contain hardware hashing capabilities local to the interface. • When NPE B is configured in MII mode of operation and the signal is not being used, it should be pulled low through a 10-KΩ resistor. **SMII Mode of Operation:** Not used. • When NPE B is configured in SMII mode of operation, this signal must be pulled high with a 10-KΩ resistor. When this interface is disabled via the NPE-B Ethernet 0 and/or the NPE Ethernet 1-3 soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 5 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHB_CRS/ SMII_SYNC/ SMII_TXSYNC	Z	Z	Z	VI / VO	I/O	**MII Mode of Operation:** Asserted by the PHY when the transmit medium or receive medium is active. De-asserted when both the transmit and receive medium are idle. Remains asserted throughout the duration of a collision condition. PHY asserts CRS asynchronously and de-asserts synchronously, with respect to ETHB_RXCLK. This MAC interface does not contain hardware hashing capabilities local to the interface. **SMII Mode of Operation:** In SMII Mode of Operation, this signal is an output that creates a synchronous pulse once every 10 SMII_CLK reference clocks to signal the start of the next 10 bits of data to be transmitted/ received. SMII_CLK Reference clock operates at 125MHz. In Source Synchronous mode of operation, a synchronous pulse output created once every 10 SMII_TXCLK clocks to signal the start of the next 10 bits of data to be transmitted. SMII_TXCLK operates at 125MHz. When this interface/signal is enabled and is not being used in a system design, the interface/ signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the NPE-B Ethernet 0 and/or the NPE Ethernet 1-3 soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection. In MII mode of operation, this signal is a valid input. In SMII mode of operation this signal is a valid output.
ETH_MDIO	Z	Z	Z	VB	I/O	Management data output. Provides the write data to both PHY devices connected to each MII interface. An external pull-up resistor of 1.5K ohm is required on ETH_MDIO to properly quantify the external PHYs used in the system. For specific implementation, see the IEEE 802.3 specification. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
ETH_MDC	Z	Z	VI	VI / VO	I/O	Management data clock. Management data interface clock is used to clock the MDIO signal as an output and sample the MDIO as an input. The ETH_MDC is an input on power up and can be configured to be an output through an Intel API as documented in the *Intel® IXP400 Software Programmer's Guide*.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 6 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHC_TXCLK	Z	VI	VI	VI	I	Externally supplied transmit clock. • 25 MHz for 100 Mbps operation • 2.5 MHz for 10 Mbps This MAC contains hardware hashing capabilities local to the interface. This signal should be pulled high through a 10-KΩ resistor when being utilized in SMII mode of operation. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor.
ETHC_TXDATA[3:1]	Z	0	VO	VO	O	**MII Mode of Operation:** Transmit data bus to PHY, asserted synchronously with respect to ETHC_TXCLK. This MAC contains hardware hashing capabilities local to the interface. **SMII Mode of Operation:** Not used in SMII mode of operation.
ETHC_TXDATA[0] / SMII_TXDATA[5]	Z	0	VO	VO	O	**MII Mode of Operation:** Transmit data bus to PHY, asserted synchronously with respect to ETHC_TXCLK. This MAC contains hardware hashing capabilities local to the interface. **SMII Mode of Operation:** The data on this signal is transmitted synchronously with respect to the rising edge of SMII_CLK when operating as an SMII interface and synchronously with respect to the rising edge of SMII_TXCLK when operating as a Source Synchronous SMII interface
ETHC_TXEN	Z	0	VO	VO	O	Indicates that the PHY is being presented with nibbles on the MII interface. Asserted synchronously, with respect to ETHC_TXCLK, at the first nibble of the preamble, and remains asserted until all the nibbles of a frame are presented. This MAC contains hardware hashing capabilities local to the interface.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 7 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHC_RXCLK	Z	VI	VI	VI	I	Externally supplied receive clock. • 25 MHz for 100 Mbps operation • 2.5 MHz for 10 Mbps This MAC contains hardware hashing capabilities local to the interface. Should be pulled high through a 10-KΩ resistor when not being utilized in the system or when in SMII mode of operation.
ETHC_RXDATA[3:1]	Z	VI	VI	VI	I	Receive data bus from PHY, data sampled synchronously, with respect to ETHC_RXCLK. This MAC contains hardware hashing capabilities local to the interface. • Not used when operating in SMII mode of operation. Should be pulled high through a 10-KΩ resistor when not being utilized in the system or when in SMII mode of operation.
ETHC_RXDATA[0] / SMII_RXDATA[5]	Z	VI	VI	VI	I	**MII Mode of Operation:** Receive data bus from PHY, data sampled synchronously, with respect to ETHC_RXCLK. This MAC contains hardware hashing capabilities local to the interface **SMII Mode of Operation:** The data on this signal is received synchronously with respect to the rising edge of SMII_CLK when operating as an SMII interface and synchronously with respect to the rising edge of SMII_RXCLK when operating as a Source Synchronous SMII interface Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
ETHC_RXDV	Z	VI	VI	VI	I	Receive data valid, used to inform the MII interface that the Ethernet PHY is sending data. This MAC contains hardware hashing capabilities local to the interface. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 16. MII/SMII Interfaces (Sheet 8 of 8)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
ETHC_COL	Z	VI	VI	VI	I	**MII Mode of Operation:** Asserted by the PHY when a collision is detected by the PHY. This MAC contains hardware hashing capabilities local to the interface. • When NPE C is configured in MII mode of operation and the signal is not being used, it should be pulled low through a 10-KΩ resistor. **SMII Mode of Operation:** Not used. • When NPE C is configured in SMII mode of operation, this signal must be pulled high through a 10-KΩ resistor. When this interface is disabled via the NPE-C Ethernet soft fuse (refer to the Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual*) and is not being used a system desing, this interface/signal is not required for any connection.
ETHC_CRS	Z	VI	VI	VI	I	Asserted by the PHY when the transmit medium or receive medium are active. De-asserted when both the transmit and receive medium are idle. Remains asserted throughout the duration of collision condition. PHY asserts CRS asynchronously and de-asserts synchronously with respect to ETHC_RXCLK. This MAC contains hardware hashing capabilities local to the interface. Should be pulled high through a 10-KΩ resistor when not being utilized in the system or when in SMII mode of operation.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for information on how to select the interface desired

Table 17. Expansion Bus Interface (Sheet 1 of 4)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
EX_CLK	Z	VI	VI	VI	I	Input clock signal used to sample all expansion interface inputs and clock all expansion interface outputs.
EX_ALE	H	H	VO	VO / Z	TRI	Expansion bus Address-latch enable used for multiplexed address/data bus accesses, as an Advance pin for Intel synchronous modes of operation/ZBT SRAM mode of operation, and LD_N for ZBT SRAM. EX_ALE is always used by outbound transfers.
EX_ADDR[24:0]	H	H	VB	VB	I/O	Expansion bus address used as an output for data accesses over the expansion bus when executing outbound transactions and used as an input for data accesses over the expansion bus when executing inbound transactions. Also, used as an input during reset to capture device configuration. These signals have a weak pull-up resistor attached internally. Based on the desired configuration, various address signals must be tied low in order for the device to operate in the desired mode. A 10K ohm pull down resistor is required to override these pull-up resistors. These pull-ups are disabled when PLL_LOCK is asserted and the IXP45X/IXP46X network processors drive the signal based upon grant. EX_ADDR is driven by IXP45X/IXP46X network processors except when grant is asserted to an external master or during reset. ***Very Important Note: See Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual for additional details on address strapping.***
EX_WR_N	H	H	VB	VB	I/O	Expansion bus write enable signal is used as an Intel-mode write strobe / Motorola-mode data strobe (EXP_MOT_DS_N) / TI*-mode data strobe (TI_HDS1_N) / ZBT SRAM mode read/ write_n(ZBT_RD_WR_N) for outbound transactions. This signal is an output for outbound transactions. Expansion bus write enable signal is used as a write enable signal to the IXP45X/IXP46X network processors for inbound transaction support. This signal is an input for inbound transactions. EX_WR_N is driven by IXP45X/IXP46X network processors unless grant is asserted to an external master
EX_RD_N	H	H	VB	VB	I/O	Expansion bus read enable signal is used as an Intel-mode read strobe / Motorola-mode read-not-write (EXPB_MOT_RNW) / TI mode read-not-write (TI_HR_W_N) / ZBT SRAM mode output enable (ZBT_OE_N) for outbound transactions. This signal is an output for outbound transactions. Expansion bus read enable signal is used as a read enable signal to the IXP45X/IXP46X network processors for inbound transaction support. This signal is an input for inbound transactions. EX_RD_N is driven by IXP45X/IXP46X network processors unless grant is asserted to an external master.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 17. Expansion Bus Interface (Sheet 2 of 4)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
EX_CS_N[7:0]	H	H	VB	VB	I/O	Used to drive chip selects for outbound transactions for the expansion bus. • Chip selects 0 through 7 can be configured to support Intel/Intel Synchronous/Motorola/ZBT SRAM bus cycles. • Chip selects 4 through 7 can be configured to support TI HPI bus cycles. • These signal are also sampled by the arbiter to determine when to arbitrate. Driving the signals from an external interface has no effect on the operation of anything but the arbiter. • External board pull-ups are required on EX_CS_N to ensure this signal remains deasserted (especially in a multi-master environment). Additionally, the system designer is responsible for ensuring that all the tri-stated signals do not become indeterminate. If they become indeterminate, excessive power consumption will occur in the PAD input buffers.
EX_DATA[31:0]	H	H	VB	VB	I/O	Expansion bus, bidirectional data
EX_BE_N[3:0]	H	H	VB	VB	I/O	Expansion bus Byte enables. EX_BE_N is used to select the particular bytes that will be written or read when executing outbound transfers. When executing inbound transfers, EX_BE_N will be used to select sub-word writes. Only 32 bit reads of the expansion bus is supported when operating on inbound transfers. EX_BE_N is driven by the IXP45X/IXP46X network processors unless grant is asserted to an external master.
EX_IOWAIT_N	Z	VI	VI	VI	I	Data ready/acknowledge from expansion bus devices. Expansion bus access is halted when an external device sets EX_IOWAIT_N to logic 0 and resume from the halted location once the external device sets EX_IOWAIT_N to logic 1. This signal affects accesses that use EX_CS_N[7:0] when the chip select is configured in Intel and Motorola modes of operation. During idle cycles, the board is responsible for ensuring that EX_IOWAIT_N is pulled-up. Additionally, EX_IOWAIT_N must always be pulled high during Micron ZBT, Intel Synchronous Mode, and HPI cycles Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
EX_RDY_N[3:0]	Z	VI	VI	VI	I	HPI interface ready signals. Can be configured to be active high or active low. These signals are used to halt accesses using Chip Selects 7 through 4 when the chip selects are configured to operate in HPI mode. There is one RDY signal per chip select. This signal only affects accesses that use EX_CS_N[7:4]. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 17. Expansion Bus Interface (Sheet 3 of 4)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
EX_PARITY[3:0]	H	H	VB	VB	I/O	Byte wide parity protection on the EX_DATA[31:0]
EX_REQ_N[3:1]	Z	H	VI/H*	VI/H*	I	Signals used by external masters to gain access to the bus. An external master asserts this signal to the internal arbiter to request access to use the expansion bus signals. * When configured in external arbiter mode of operation, an internal pull-up will be enabled, thus the pins will be driven to VCC Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
EX_REQ_GNT_N	Z	VI	VI	VI	I	When the IXP45X/IXP46X network processors are functioning as the Expansion bus arbiter, this signal will serve as the request input from an external master. If there is an external arbiter used for expansion bus accesses, this signal will serve as the expansion bus grant input from the external arbiter. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
EX_GNT_N[3:1]	Z	b'111	VO	VO	O	Signals used by the arbiter to inform external masters that the master is now granted access to use the bus. In response to an EX_REQ_N being asserted by an external master, the arbiter will output the corresponding EX_GNT_N signal to inform the external master that the expansion bus is clear for that master to utilize.
EX_GNT_REQ_N	Z	1	VO	VO	O	When the IXP45X/IXP46X network processors are functioning as the Expansion bus arbiter, this signal will serve as the grant output for an external master asserting the corresponding request. If there is an external arbiter used for expansion bus accesses, this signal will serve as the expansion bus request output signal to the external arbiter.
EX_SLAVE_CS_N	Z	VI	VI	VI	I	The expansion bus chip select input is used to determine when an external master is attempting to access the IXP45X/IXP46X network processors' expansion bus interface and internal memory map of the processors. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 17. Expansion Bus Interface (Sheet 4 of 4)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
EX_BURST	Z	VI	VI	VI	I	For inbound transfers, this signal is used to signify that a burst operation is being requested to occur. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
EX_WAIT_N	H	H	H	VO/H	TRI	Expansion bus IXP45X/IXP46X network processors wait. EX_WAIT_N is driven by the processors when EX_SLAVE_CS_N is asserted. After the de-assertion of EX_SLAVE_CS_N, the IXP45X/ IXP46X network processors will stop driving this signal. A pull-up in the PAD IO is enabled all the time to prevent this bus from floating or transitioning to VSS. This signal is used to hold off an external master when the expansion interface cannot be accessed immediately. Most commonly seen when a read access of the interface is occurring and the data has not been returned from the internal peripheral unit to the expansion interface.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 18. UART Interfaces (Sheet 1 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
RXDATA0	Z	VI	VI	VI	I	UART serial data input to UART Pins. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
TXDATA0	Z	V0	VO	VO	O	UART serial data output. The TXD signal is set to the MARKING (logic 1) state upon a reset operation.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 18. UART Interfaces (Sheet 2 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
CTS0_N	H	VI/H	VI/H	VI/H	I	UART CLEAR-TO-SEND input to UART Pins. When logic 0, this pin indicates that the modem or data set connected to the UART interface of the processor is ready to exchange data. The signal is a modem status input whose condition can be tested by the processor. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
RTS0_N	Z	V0	VO	VO	O	UART REQUEST-TO-SEND output: When logic 0, this informs the modem or the data set connected to the UART interface of the processor that the UART is ready to exchange data. A reset sets the request to send signal to logic 1. LOOP-mode operation holds this signal in its inactive state (logic 1)
RXDATA1	Z	VI	VI	VI	I	UART serial data input. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
TXDATA1	Z	VO	VO	VO	O	UART serial data output. The TXD signal is set to the MARKING (logic 1) state upon a Reset operation.
CTS1_N	H	VI/H	VI/H	VI/H	I	UART CLEAR-TO-SEND input to UART pins. When logic 0, this pin indicates that the modem or data set connected to the UART interface of the processor is ready to exchange data. The signal is a modem status input whose condition can be tested by the processor. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
RTS1_N	Z	V0	VO	VO	O	UART REQUEST-TO-SEND output: When logic 0, this informs the modem or the data set connected to the UART interface of the processor that the UART is ready to exchange data. A reset sets the request to send signal to logic 1. LOOP-mode operation holds this signal in its inactive state (logic 1).

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 19. Serial Peripheral Port Interface

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
SSP_SCLK	Z	0	VO	VO	O	SSP_SCLK is the serial bit clock used to control the timing of a transfer. SSP_SCLK can be generated internally (Master mode) as defined by a control register bit internal to the IXP45X/IXP46X network processors.
SSP_SFRM	Z	1	VO	VO	O	SSP_SFRM is the serial frame indicator that indicates the beginning and the end of a serialized data word. The SSP_SFRM can be generated internally (Master mode) or taken from an external source (Slave mode) as defined by a control register bit internal to the IXP45X/IXP46X network processors. This signal may be active low or active high depending upon the mode of operation. Please refer to the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual* for additional details.
SSP_TXD	Z	0	VO	VO	O	SSP_TXD is the Transmit data (serial data out) serialized data line. Sample length is a function of the selected serial data sample size.
SSP_RXD	Z	VI	VI	VI	I	SSP_RXD is the Receive data (serial data in) serialized data line. Sample length is a function of the selected serial data sample size. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
SSP_EXTCLK	Z	VI	VI	VI	I	SSP_EXTCLK is an external clock which can be selected to replace the internal 3.6864 MHz clock. The SSP_EXTCLK input is selected by setting various internal registers to appropriate values. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 20. I²C Interface

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
I2C_SDA	Z	Z	VOD	VOD	I/O/OD	The receive and transmit data/address line used to communicate between various master and slave I²C interfaces. A pull up resistor is required on this interface. Please refer to the I²C specification.
I2C_SCL	Z	Z	VOD	VOD	I/O/OD	The master and slave clock line used to communicate between various master and slave I²C interfaces. A pull up resistor is required on this interface. Please refer to the I²C specification.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 21. USB Host/Device Interfaces (Sheet 1 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
USB_DPOS	Z	Z	VB	VB	I/O	Positive signal of the differential USB receiver/driver for the USB device interface. **NOTE:** This pin requires an 18Ω external series resistor. This resistor is located after the pin, but before the pull-down resistor. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled low with a 10-KΩ resistor. When this interface is disabled via the USB Device soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
USB_DNEG	Z	Z	VB	VB	I/O	Negative signal of the differential USB receiver/driver for the USB device interface. **NOTE:** This pin requires an 18Ω external series resistor. This resistor is located after the pin, but before the pull-down resistor. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled low with a 10-KΩ resistor. When this interface is disabled via the USB Device soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
USB_HPOS	Z	Z	VB	VB	I/O	Positive signal of the differential USB receiver/driver for the USB host interface. **NOTE:** This pin requires an 20Ω external series resistor. This resistor is located after the pin, but before the pull-down resistor. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled low with a 10-KΩ resistor. When this interface is disabled via the USB Device soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Hardware Design Guidelines* for additional board design details.

Table 21. USB Host/Device Interfaces (Sheet 2 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
USB_HNEG	Z	Z	VB	VB	I/O	Negative signal of the differential USB receiver/driver for the USB host interface. **NOTE:** This pin requires an 20Ω external series resistor. This resistor is located after the pin, but before the pull-down resistor. When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled low with a 10-KΩ resistor. When this interface is disabled via the USB Device soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.
USB_HPEN	Z	Z	VO	VO	O	Enable to the external VBUS power source
USB_HPWR	Z	Z	VI	VI	I	External VBUS power is in over current condition When this interface/signal is enabled and is not being used in a system design, the interface/signal should be pulled high with a 10-KΩ resistor. When this interface is disabled via the USB Device soft fuse (refer to Expansion Bus Controller chapter of the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Developer's Manual)* and is not being used in a system design, this interface/signal is not required for any connection.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

†† Please refer to the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Hardware Design Guidelines* for additional board design details.

Table 22. Oscillator Interface

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
OSC_IN	n/a	VI	VI	VI	I	33.33-MHz, sinusoidal input signal. Can be driven by an oscillator.
OSC_OUT	n/a	VO	VO	VO	O	33.33-MHz, sinusoidal output signal. Left disconnected when being driven by an oscillator.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 23. GPIO Interface

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
GPIO[12:0]	Z	Z	VI	VB	I/O	General purpose Input/Output pins. May be configured as an input or an output. As an input, each signal may be configured a processor interrupt. Default after reset is to be configured as inputs. Some GPIO may serve as an optional alternate function. Refer to Section 3.1.12, "GPIO" on page 28, for additional details on alternate function mapping. Should be pulled high using a 10-KΩ resistor when not being utilized in the system.
GPIO[13]	Z	Z	VI	VB	I/O	General purpose input/output pins. May be configured as an input or an output. Default after reset is to be configured as inputs. Some GPIO may serve as an optional alternate function. Refer to Section 3.1.12, "GPIO" on page 28, for additional details on alternate function mapping. Should be pulled high using a 10-KΩ resistor when not being utilized in the system.
GPIO[14]	Z	Z	VI	VB	I/O	Can be configured the same as GPIO Pin 13 or as a clock output. Configuration as an output clock can be set at various speeds of up to 33 MHz with various duty cycles. Configured as an input, upon reset. Some GPIO may serve as an optional alternate function. Refer to Section 3.1.12, "GPIO" on page 28, for additional details on alternate function mapping. Should be pulled high through a 10-KΩ resistor when not being utilized in the system.
GPIO[15]	Z	clkout / VO	VO	VB	I/O	Can be configured the same as GPIO Pin 13 or as a clock output. Configuration as an output clock can be set at various speeds of up to 33 MHz with various duty cycles. Configured as an output, upon reset. Can be used to clock the expansion interface, after reset. Some GPIO may serve as an optional alternate function. Refer to Section 3.1.12, "GPIO" on page 28, for additional details on alternate function mapping. Should be pulled high through a 10-KΩ resistor when not being utilized in the system. The interface should be set to an input in the not used configuration.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 24. JTAG Interface

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
JTG_TMS	H	VI / H	VI/H	VI/H	I	Test mode select for the IEEE 1149.1 JTAG interface.
JTG_TDI	H	VI /H	VI/H	VI/H	I	Input data for the IEEE 1149.1 JTAG interface.
JTG_TDO	Z	VO/Z	VO / Z	VO / Z	TRI	Output data for the IEEE 1149.1 JTAG interface.
JTG_TRST_N	H	VI/H	VI	VI	I	Used to reset the IEEE 1149.1 JTAG interface. **Important:** The JTG_TRST_N signal must be asserted (driven low) during power-up, otherwise the TAP controller will not be initialized properly and the processor may be locked. When the JTAG interface is not being used, the signal must be pulled low using a 10-KΩ resistor.
JTG_TCK	Z	VI	VI	VI	I	Used as the clock for the IEEE 1149.1 JTAG interface.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 25. System Interface (Sheet 1 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
BYPASS_CLK	Z	VI	VI	VI	I	Used for test purposes only. Must be pulled high using a 10-KΩ resistor for normal operation.
SCANTESTMODE_N	VI/H	VI/H	VI / H	VI / H	I	Used for test purposes only. Must be pulled high using a 10-KΩ resistor for normal operation.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 25. System Interface (Sheet 2 of 2)

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
RESET_IN_N	VI/H	VI/H	VI / H	VI / H	I	Used as a reset input to the device after power up conditions have been met. Power up conditions include the power supplies reaching a safe stable condition and the PLL achieving a locked state and the PWRON_RESET_N coming to an active state prior to the RESET_IN_N coming to an active state.
PWRON_RESET_N	VI/H	VI/H	V I/ H	VI / H	I	Signal used at power up to reset all internal logic to a known state after the PLL has achieved a locked state. The PWRON_RESET_N signal is a 3.3-V signal.
HIGHZ_N	VI / H	VI / H	VI / H	VI / H	I	Used for test purposes only. Must be pulled high using a 10-KΩ resistor for normal operation.
PLL_LOCK	Z	VO	VO	VO	O	Signal used to inform external reset logic that the internal PLL has achieved a locked state. PLL_LOCK will also be de-asserted during a watchdog timeout.
RCOMP_REF	Tied off to a resistor	Tied off to a resistor	Tied off to a resistor	Tied off to a resistor	O	Signal used to control PCI and SMII drive strength characteristics. Drive strength is varied on PCI and SMII signals depending upon temperature. Pin requires a 34-Ω +/- 1% tolerance resistor to ground. (Refer to Figure 14 on page 119.)
SPARE1	n/a	n/a	n/a	n/a	n/a	No Connection is to be made to this signal
SPARE2	n/a	n/a	n/a	n/a	n/a	No Connection is to be made to this signal

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

Table 26. Power Interface

Name	Power on Reset†	Reset†	Normal After Reset Until Software Enables	Normal After Software Enables	Type†	Description
VCC	N/A	N/A	N/A	N/A	I	1.3-V power supply input pins used for the internal logic.
VCCP	N/A	N/A	N/A	N/A	I	3.3-V power supply input pins used for the peripheral (I/O) logic.
VCCM	N/A	N/A	N/A	N/A	I	2.5-V power supply input pins used for the DDR memory interface
VSS	N/A	N/A	N/A	N/A	I	Ground power supply input pins used for both the 3.3-V, 2.5-V, and the 1.3-V power supplies.
OSC_VCCP	N/A	N/A	N/A	N/A	I	3.3-V power supply input pins used for the peripheral (I/O) logic of the analog oscillator circuitry. Require special power filtering circuitry. Refer to Figure 12 on page 118.
OSC_VSSP	N/A	N/A	N/A	N/A	I	Ground input pins used for the peripheral (I/O) logic of the analog oscillator circuitry. Used in conjunction with the OSC_VCCP pins. Requires special power filtering circuitry. Refer to Figure 12 on page 118.
OSC_VCC	N/A	N/A	N/A	N/A	I	1.3-V power supply input pins used for the internal logic of the analog oscillator circuitry. Requires special power filtering circuitry. Refer to Figure 13 on page 118.
OSC_VSS	N/A	N/A	N/A	N/A	I	Ground power supply input pins used for the internal logic of the analog oscillator circuitry. Used in conjunction with the OSC_VCC pins. Requires special power filtering circuitry. Refer to Figure 13 on page 118.
VCCPLL1	N/A	N/A	N/A	N/A	I	1.3-V power supply input pins used for the internal logic of the analog phase lock-loop circuitry. Requires special power filtering circuitry. Refer to Figure 9 on page 116.
VCCPLL2	N/A	N/A	N/A	N/A	I	1.3-V power supply input pins used for the internal logic of the analog phase lock-loop circuitry. Requires special power filtering circuitry. Refer to Figure 10 on page 117.
VCCPLL3	N/A	N/A	N/A	N/A	I	1.3-V power supply input pins used for the internal logic of the analog phase lock-loop circuitry. Requires special power filtering circuitry. Refer to Figure 11 on page 117.

NOTE: This table discusses all features supported on the Intel® IXP45X and Intel® IXP46X Product Line of Network Processors. For details on feature support listed by processor, see Table 1 on page 14.

† For a legend of the **Type** codes, see Table 10 on page 46.

4.3 Signal-Pin Descriptions

When designing with a multifunction processor, sometimes a board design may be built to allow a group of products to be produced from a single board design. When this occurs, some features of a given processor may not be used. The *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Hardware Design Guidelines* gives the system designer a guide to determine how the signals must be conditioned and how the part behaves under given configurations.

Table 27. Processors' Ball Map Assignments (Sheet 1 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
A1	VSS			X	X	X
A2	VSS			X	X	X
A3	DDRI_CB[0]			X	X	
A4	DDRI_CK_N[1]			X	X	X
A5	DDRI_DM[3]			X	X	X
A6	DDRI_DQ[30]			X	X	X
A7	DDRI_DQ[26]			X	X	X
A8	DDRI_DQ[25]			X	X	X
A9	DDRI_DQ[22]			X	X	X
A10	DDRI_DQ[18]			X	X	X
A11	DDRI_MA[4]			X	X	X
A12	VSS			X	X	X
A13	DDRI_MA[0]			X	X	X
A14	DDRI_MA[8]			X	X	X
A15	DDRI_BA[1]			X	X	X
A16	VCCM			X	X	X
A17	DDRI_CS_N[1]			X	X	X
A18	DDRI_CAS_N			X	X	X
A19	DDRI_DM[1]			X	X	X
A20	DDRI_DQ[13]			X	X	X
A21	DDRI_DQ[11]			X	X	X
A22	DDRI_DM[0]			X	X	X
A23	DDRI_DQ[6]			X	X	X
A24	VCCM			X	X	X
A25	VSS			X	X	X
A26	VSS			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 2 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
B1	VSS			X	X	X
B2	VSS			X	X	X
B3	DDRI_CB[2]			X	X	
B4	DDRI_CB[1]			X	X	
B5	DDRI_CK[1]			X	X	X
B6	DDRI_DQS[3]			X	X	X
B7	DDRI_DQ[31]			X	X	X
B8	DDRI_DQ[24]			X	X	X
B9	DDRI_DQ[21]			X	X	X
B10	DDRI_DQ[23]			X	X	X
B11	DDRI_DQ[17]			X	X	X
B12	DDRI_MA[5]			X	X	X
B13	DDRI_MA[7]			X	X	X
B14	DDRI_MA[9]			X	X	X
B15	DDRI_MA[11]			X	X	X
B16	DDRI_CS_N[0]			X	X	X
B17	DDRI_RAS_N			X	X	X
B18	DDRI_VREF			X	X	X
B19	DDRI_DQS[1]			X	X	X
B20	DDRI_DQ[14]			X	X	X
B21	DDRI_DQS[0]			X	X	X
B22	DDRI_DQ[5]			X	X	X
B23	DDRI_DQ[7]			X	X	X
B24	VCCP			X	X	X
B25	USB_HPEN			X	X	X
B26	VSS			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 3 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
C1	PCI_AD[30]			X	X	X
C2	PCI_GNT_N[0]			X	X	X
C3	DDRI_CB[3]			X	X	
C4	DDRI_CB[5]			X	X	
C5	DDRI_CB[7]			X	X	
C6	DDRI_DQS[4]			X	X	X
C7	DDRI_DQ[28]			X	X	X
C8	DDRI_DQ[27]			X	X	X
C9	DDRI_DM[2]			X	X	X
C10	VCCM			X	X	X
C11	DDRI_DQ[16]			X	X	X
C12	VSS			X	X	X
C13	DDRI_MA[1]			X	X	X
C14	VCCM			X	X	X
C15	DDRI_BA[0]			X	X	X
C16	DDRI_CKE[0]			X	X	X
C17	DDRI_CK[0]			X	X	X
C18	DDRI_RCVENOUT_N			X	X	X
C19	DDRI_DQ[15]			X	X	X
C20	DDRI_DQ[10]			X	X	X
C21	DDRI_DQ[4]			X	X	X
C22	DDRI_DQ[3]			X	X	X
C23	DDRI_DQ[2]			X	X	X
C24	VSS			X	X	X
C25	EX_CS_N[2]			X	X	X
C26	EX_CS_N[5]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 4 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
D1	VCCP			X	X	X
D2	PCI_REQ_N[1]			X	X	X
D3	PCI_GNT_N[1]			X	X	X
D4	VSS			X	X	X
D5	DDRI_CB[4]			X	X	
D6	DDRI_CB[6]			X	X	
D7	DDRI_DM[4]			X	X	X
D8	DDRI_DQ[29]			X	X	X
D9	VSS			X	X	X
D10	DDRI_DQS[2]			X	X	X
D11	DDRI_DQ[19]			X	X	X
D12	DDRI_MA[3]			X	X	X
D13	DDRI_MA[6]			X	X	X
D14	DDRI_MA[10]			X	X	X
D15	DDRI_MA[13]			X	X	X
D16	DDRI_CKE[1]			X	X	X
D17	DDRI_CK_N[0]			X	X	X
D18	DDRI_DQ[12]			X	X	X
D19	DDRI_DQ[9]			X	X	X
D20	DDRI_DQ[8]			X	X	X
D21	VCCM			X	X	X
D22	DDRI_DQ[1]			X	X	X
D23	DDRI_DQ[0]			X	X	X
D24	EX_CS_N[1]			X	X	X
D25	EX_CS_N[4]			X	X	X
D26	VCCP			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 5 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
E1	PCI_AD[26]			X	X	X
E2	PCI_AD[28]			X	X	X
E3	PCI_REQ_N[0]			X	X	X
E4	PCI_GNT_N[2]			X	X	X
E5	VSS			X	X	X
E6	VSS			X	X	X
E7	VCCM			X	X	X
E8	VCCM			X	X	X
E9	VCCM			X	X	X
E10	VSS			X	X	X
E11	DDRI_DQ[20]			X	X	X
E12	VCCM			X	X	X
E13	DDRI_MA[2]			X	X	X
E14	VSS			X	X	X
E15	DDRI_MA[12]			X	X	X
E16	DDRI_WE_N			X	X	X
E17	DDRI_RCVENIN_N			X	X	X
E18	VSS			X	X	X
E19	VSS			X	X	X
E20	VCCM			X	X	X
E21	VSS			X	X	X
E22	VSS			X	X	X
E23	USB_HPWR			X	X	X
E24	EX_CS_N[3]			X	X	X
E25	EX_GNT_N[1]			X	X	X
E26	EX_REQ_N[2]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 6 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
F1	PCI_AD[21]			X	X	X
F2	PCI_CBE_N[3]			X	X	X
F3	VCCP			X	X	X
F4	PCI_AD[31]			X	X	X
F5	PCI_GNT_N[3]			X	X	X
F6	VSS			X	X	X
F7	DDRI_CK[2]			X	X	X
F8	VSS			X	X	X
F9	VCCM			X	X	X
F10	VCC			X	X	X
F11	VCC			X	X	X
F12	VCC			X	X	X
F13	VCC			X	X	X
F14	VCC			X	X	X
F15	VCC			X	X	X
F16	VCC			X	X	X
F17	VSS			X	X	X
F18	VSS			X	X	X
F19	VSS			X	X	X
F20	VSS			X	X	X
F21	SPARE1			X	X	X
F22	VSS			X	X	X
F23	EX_CS_N[0]			X	X	X
F24	EX_GNT_REQ_N			X	X	X
F25	EX_REQ_N[1]			X	X	X
F26	VSS			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 7 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
G1	PCI_AD[18]			X	X	X
G2	PCI_AD[20]			X	X	X
G3	PCI_AD[22]			X	X	X
G4	PCI_AD[25]			X	X	X
G5	PCI_REQ_N[3]			X	X	X
G6	PCI_INTA_N			X	X	X
G7	RCOMP_REF			X	X	X
G8	DDRI_CK_N[2]			X	X	X
G9	VCC			X	X	X
G10	VCC			X	X	X
G11						
G12						
G13						
G14						
G15						
G16						
G17	VCC			X	X	X
G18	VCC			X	X	X
G19	VCC			X	X	X
G20	DDRI_RCOMP			X	X	X
G21	USB_DPOS			X	X	X
G22	USB_HPOS			X	X	X
G23	EX_SLAVE_CS_N			X	X	X
G24	EX_REQ_GNT_N			X	X	X
G25	EX_ADDR[22]			X	X	X
G26	EX_ADDR[18]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 8 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
H1	VCCP			X	X	X
H2	PCI_AD[17]			X	X	X
H3	VSS			X	X	X
H4	PCI_AD[24]			X	X	X
H5	PCI_AD[29]			X	X	X
H6	PCI_REQ_N[2]			X	X	X
H7	PCI_SERR_N			X	X	X
H8						
H9						
H10						
H11						
H12						
H13						
H14						
H15						
H16						
H17						
H18						
H19						
H20	USB_DNEG			X	X	X
H21	USB_HNEG			X	X	X
H22	EX_CS_N[7]			X	X	X
H23	EX_GNT_N[3]			X	X	X
H24	EX_ADDR[23]			X	X	X
H25	EX_ADDR[21]			X	X	X
H26	EX_ADDR[17]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 9 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
J1	PCI_CLKIN			X	X	X
J2	PCI_FRAME_N			X	X	X
J3	PCI_AD[16]			X	X	X
J4	PCI_AD[19]			X	X	X
J5	PCI_AD[23]			X	X	X
J6	PCI_AD[27]			X	X	X
J7	VCC			X	X	X
J8						
J9						
J10						
J11						
J12						
J13						
J14						
J15						
J16						
J17						
J18						
J19						
J20	VCC			X	X	X
J21	EX_CS_N[6]			X	X	X
J22	VSS			X	X	X
J23	EX_ADDR[24]			X	X	X
J24	EX_ADDR[20]			X	X	X
J25	EX_ADDR[5]			X	X	X
J26	VCCP			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 10 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
K1	VSS			X	X	X
K2	PCI_DEVSEL_N			X	X	X
K3	PCI_CBE_N[2]			X	X	X
K4	PCI_STOP_N			X	X	X
K5	VSS			X	X	X
K6	PCI_IDSEL			X	X	X
K7	VCC			X	X	X
K8						
K9						
K10						
K11						
K12						
K13						
K14						
K15						
K16						
K17						
K18						
K19						
K20	VCC			X	X	X
K21	EX_GNT_N[2]			X	X	X
K22	EX_REQ_N[3]			X	X	X
K23	EX_ADDR[19]			X	X	X
K24	EX_ADDR[4]			X	X	X
K25	EX_RD_N			X	X	X
K26	EX_CLK			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 11 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
L1	PCI_AD[11]			X	X	X
L2	PCI_CBE_N[1]			X	X	X
L3	PCI_PERR_N			X	X	X
L4	PCI_PAR			X	X	X
L5	PCI_IRDY_N			X	X	X
L6	VCC			X	X	X
L7						
L8						
L9						
L10						
L11	VSS			X	X	X
L12	VSS			X	X	X
L13	VSS			X	X	X
L14	VSS			X	X	X
L15	VSS			X	X	X
L16	VSS			X	X	X
L17						
L18						
L19						
L20						
L21	VCC			X	X	X
L22	VCCP			X	X	X
L23	EX_ALE			X	X	X
L24	EX_BE_N[0]			X	X	X
L25	EX_BE_N[2]			X	X	X
L26	EX_BE_N[3]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 12 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
M1	PCI_CBE_N[0]			X	X	X
M2	PCI_AD[12]			X	X	X
M3	PCI_AD[14]			X	X	X
M4	PCI_AD[13]			X	X	X
M5	PCI_AD[15]			X	X	X
M6	VCC			X	X	X
M7						
M8						
M9						
M10						
M11	VSS			X	X	X
M12	VSS			X	X	X
M13	VSS			X	X	X
M14	VSS			X	X	X
M15	VSS			X	X	X
M16	VSS			X	X	X
M17						
M18						
M19						
M20						
M21	VCC			X	X	X
M22	EX_WR_N			X	X	X
M23	EX_BE_N[1]			X	X	X
M24	VSS			X	X	X
M25	EX_IOWAIT_N			X	X	X
M26	EX_RDY_N[0]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 13 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
N1	PCI_AD[6]			X	X	X
N2	PCI_AD[4]			X	X	X
N3	VCCP			X	X	X
N4	PCI_AD[10]			X	X	X
N5	PCI_AD[9]			X	X	X
N6	VCC			X	X	X
N7						
N8						
N9						
N10						
N11	VSS			X	X	X
N12	VSS			X	X	X
N13	VSS			X	X	X
N14	VSS			X	X	X
N15	VSS			X	X	X
N16	VSS			X	X	X
N17						
N18						
N19						
N20						
N21	VCC			X	X	X
N22	EX_ADDR[3]			X	X	X
N23	EX_ADDR[2]			X	X	X
N24	EX_RDY_N[1]			X	X	X
N25	EX_RDY_N[2]			X	X	X
N26	EX_RDY_N[3]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 14 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
P1	VSS			X	X	X
P2	PCI_TRDY_N			X	X	X
P3	PCI_AD[2]			X	X	X
P4	PCI_AD[8]			X	X	X
P5	PCI_AD[0]			X	X	X
P6	VCC			X	X	X
P7						
P8						
P9						
P10						
P11	VSS			X	X	X
P12	VSS			X	X	X
P13	VSS			X	X	X
P14	VSS			X	X	X
P15	VSS			X	X	X
P16	VSS			X	X	X
P17						
P18						
P19						
P20						
P21	VCC			X	X	X
P22	EX_DATA[23]			X	X	X
P23	EX_PARITY[1]			X	X	X
P24	EX_PARITY[2]			X	X	X
P25	EX_BURST			X	X	X
P26	EX_WAIT_N			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 15 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
R1	PCI_AD[7]			X	X	X
R2	PCI_AD[5]			X	X	X
R3	PCI_AD[3]			X	X	X
R4	PCI_AD[1]			X	X	X
R5	HSS_TXFRAME0			X		X
R6	VCC			X	X	X
R7						
R8						
R9						
R10						
R11	VSS			X	X	X
R12	VSS			X	X	X
R13	VSS			X	X	X
R14	VSS			X	X	X
R15	VSS			X	X	X
R16	VSS			X	X	X
R17						
R18						
R19						
R20						
R21	VCC			X	X	X
R22	VCC			X	X	X
R23	EX_DATA[21]			X	X	X
R24	EX_DATA[22]			X	X	X
R25	EX_DATA[15]			X	X	X
R26	VCCP			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 16 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
T1	HSS_TXDATA0			X		X
T2	HSS_TXCLK0			X		X
T3	HSS_RXFRAME0			X		X
T4	HSS_RXDATA0			X		X
T5	HSS_RXCLK0			X		X
T6	ETHC_TXEN			X	X	X
T7						
T8						
T9						
T10						
T11	VSS			X	X	X
T12	VSS			X	X	X
T13	VSS			X	X	X
T14	VSS			X	X	X
T15	VSS			X	X	X
T16	VSS			X	X	X
T17						
T18						
T19						
T20						
T21	VCC			X	X	X
T22	EX_DATA[17]			X	X	X
T23	EX_DATA[11]			X	X	X
T24	VSS			X	X	X
T25	EX_DATA[13]			X	X	X
T26	EX_DATA[14]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 17 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
U1	HSS_TXFRAME1			X		X
U2	HSS_TXDATA1			X		X
U3	VCCP			X	X	X
U4	HSS_TXCLK1			X		X
U5	VSS			X	X	X
U6	ETHC_RXDV			X	X	X
U7	VCC			X	X	X
U8						
U9						
U10						
U11						
U12						
U13						
U14						
U15						
U16						
U17						
U18						
U19						
U20	VCC			X	X	X
U21	EX_DATA[28]			X	X	X
U22	EX_DATA[30]			X	X	X
U23	EX_DATA[16]			X	X	X
U24	EX_DATA[18]			X	X	X
U25	EX_DATA[12]			X	X	X
U26	EX_DATA[20]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 18 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
V1	HSS_RXFRAME1			X		X
V2	HSS_RXDATA1			X		X
V3	HSS_RXCLK1			X		X
V4	ETHC_TXDATA[1]			X	X	X
V5	ETHC_RXDATA[3]			X	X	X
V6	VCC			X	X	X
V7	VCC			X	X	X
V8						
V9						
V10						
V11						
V12						
V13						
V14						
V15						
V16						
V17						
V18						
V19						
V20	VCC			X	X	X
V21	EX_DATA[25]			X	X	X
V22	VSS			X	X	X
V23	EX_DATA[6]			X	X	X
V24	EX_DATA[8]			X	X	X
V25	EX_DATA[10]			X	X	X
V26	EX_DATA[19]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 19 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
W1	VCCP			X	X	X
W2	ETHC_TXDATA[3]			X	X	X
W3	VSS			X	X	X
W4	ETHC_RXDATA[2]			X	X	X
W5	ETHC_CRS			X	X	X
W6	ETHB_COL			X	X	X
W7	ETHB_TXDATA[0]	SMII_TXDATA[0]		X	X	X
W8						
W9						
W10						
W11						
W12						
W13						
W14						
W15						
W16						
W17						
W18						
W19						
W20	EX_ADDR[12]			X	X	X
W21	EX_ADDR[6]			X	X	X
W22	EX_DATA[2]			X	X	X
W23	VCCP			X	X	X
W24	EX_DATA[29]			X	X	X
W25	EX_DATA[31]			X	X	X
W26	EX_DATA[9]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 20 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
Y1	ETHC_TXDATA[2]			X	X	X
Y2	ETHC_TXDATA[0]	SMII_TXDATA[5]		X	X	X
Y3	ETHC_RXDATA[1]			X	X	X
Y4	ETHC_COL			X	X	X
Y5	ETHB_TXEN	SMII_TXCLK		X	X	X
Y6	ETHB_TXDATA[1]	SMII_TXDATA[1]		X	X Config. 1 only	X Config. 1 only
Y7	VCCP			X	X	X
Y8	ETHB_RXDATA[3]	SMII_RXDATA[3]		X	X Config. 1 only	X Config. 1 only
Y9	VCC			X	X	X
Y10	VCC			X	X	X
Y11						
Y12						
Y13						
Y14						
Y15						
Y16						
Y17	VCC			X	X	X
Y18	VCC			X	X	X
Y19	EX_ADDR[13]			X	X	X
Y20	EX_ADDR[11]			X	X	X
Y21	EX_ADDR[10]			X	X	X
Y22	EX_ADDR[7]			X	X	X
Y23	EX_DATA[1]			X	X	X
Y24	EX_DATA[4]			X	X	X
Y25	EX_DATA[5]			X	X	X
Y26	EX_DATA[7]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 21 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
AA1	ETHC_TXCLK			X	X	X
AA2	ETHC_RXDATA[0]	SMII_RXDATA[5]		X	X	X
AA3	VSS			X	X	X
AA4	ETHB_TXDATA[3]	SMII_TXDATA[3]		X	X Config. 1 only	X Config. 1 only
AA5	VSS			X	X	X
AA6	SPARE2					
AA7	ETHB_RXDV	SMII_RXSYNC		X	X	X
AA8	ETHB_RXDATA[1]	SMII_RXDATA[1]		X	X Config. 1 only	X Config. 1 only
AA9	UTP_OP_DATA[4]	ETHA_TXEN		X		X
AA10	UTP_IP_DATA[3]	ETHA_RXDATA[3]		X		X
AA11	VCC			X	X	X
AA12	VCC			X	X	X
AA13	VCC			X	X	X
AA14	VCC			X	X	X
AA15	VCC			X	X	X
AA16	VCC			X	X	X
AA17	GPIO[7]			X	X	X
AA18	GPIO[1]			X	X	X
AA19	SSP_RXD			X	X	X
AA20	EX_ADDR[14]			X	X	X
AA21	VSS			X	X	X
AA22	EX_ADDR[0]			X	X	X
AA23	EX_PARITY[3]			X	X	X
AA24	EX_DATA[26]			X	X	X
AA25	EX_DATA[27]			X	X	X
AA26	VSS			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 22 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
AB1	ETHC_RXCLK			X	X	X
AB2	ETH_MDIO			X	X	X
AB3	ETHB_TXDATA[2]	SMII_TXDATA[2]		X	X Config. 1 only	X Config. 1 only
AB4	VSS			X	X	X
AB5	VSS			X	X	X
AB6	VSS			X	X	X
AB7	ETHB_RXDATA[2]	SMII_RXDATA[2]		X	X Config. 1 only	X Config. 1 only
AB8	UTP_OP_DATA[5]			X		X
AB9	VSS			X	X	X
AB10	UTP_OP_FCI			X		X
AB11	VCCPLL2			X	X	X
AB12	VCCPLL3			X	X	X
AB13	VCC			X	X	X
AB14	PLL_LOCK			X	X	X
AB15	VCCP			X	X	X
AB16	RXDATA0			X	X	X
AB17	GPIO[12]			X	X	X
AB18	GPIO[8]			X	X	X
AB19	VCCP			X	X	X
AB20	I2C_SDA			X	X	X
AB21	VSS			X	X	X
AB22	VCCP			X	X	X
AB23	EX_ADDR[1]			X	X	X
AB24	EX_PARITY[0]			X	X	X
AB25	EX_DATA[24]			X	X	X
AB26	EX_DATA[3]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 23 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
AC1	ETH_MDC			X	X	X
AC2	ETHB_CRS	SMII_SYNC	SMII_TXSYNC	X	X	X
AC3	VCCP			X	X	X
AC4	VSS			X	X	X
AC5	VCCP			X	X	X
AC6	UTP_OP_DATA[7]	SMII_TXDATA[4]		X		X
AC7	UTP_OP_DATA[3]	ETHA_TXDATA[3]		X		X
AC8	UTP_IP_DATA[5]	ETHA_COL		X		X
AC9	UTP_OP_FCO			X		X
AC10	VCCP			X	X	X
AC11	UTP_IP_FCI			X		X
AC12	VCCPLL1			X	X	X
AC13	UTP_IP_ADDR[4]			X		X
AC14	RESET_IN_N			X	X	X
AC15	JTG_TMS			X	X	X
AC16	JTG_TCK			X	X	X
AC17	CTS0_N			X	X	X
AC18	GPIO[14]			X	X	X
AC19	GPIO[9]			X	X	X
AC20	GPIO[4]			X	X	X
AC21	SSP_TXD			X	X	X
AC22	EX_ADDR[15]			X	X	X
AC23	VSS			X	X	X
AC24	EX_ADDR[9]			X	X	X
AC25	EX_DATA[0]			X	X	X
AC26	VCCP			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 24 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
AD1	ETHB_TXCLK	SMII_CLK		X	X	X
AD2	ETHB_RXCLK	SMII_RXCLK		X	X	X
AD3	ETHB_RXDATA[0]	SMII_RXDATA[0]		X	X	X
AD4	UTP_IP_DATA[7]	SMII_RXDATA[4]		X		X
AD5	VCCP			X	X	X
AD6	UTP_IP_DATA[6]	ETHA_CRS		X		X
AD7	VSS			X	X	X
AD8	UTP_OP_SOC			X		X
AD9	UTP_OP_ADDR[3]			X		X
AD10	UTP_IP_SOC			X		X
AD11	OSC_VCCP			X	X	X
AD12	OSC_VSS			X	X	X
AD13	OSC_VCC			X	X	X
AD14	UTP_IP_ADDR[0]			X		X
AD15	VSS			X	X	X
AD16	JTG_TDI			X	X	X
AD17	TXDATA1			X	X	X
AD18	TXDATA0			X	X	X
AD19	VSS			X	X	X
AD20	GPIO[13]			X	X	X
AD21	GPIO[5]			X	X	X
AD22	GPIO[0]			X	X	X
AD23	SSP_SFRM			X	X	X
AD24	I2C_SCL			X	X	X
AD25	EX_ADDR[16]			X	X	X
AD26	EX_ADDR[8]			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 25 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
AE1	VSS			X	X	X
AE2	UTP_OP_DATA[6]			X		X
AE3	UTP_OP_DATA[2]	ETHA_TXDATA[2]		X		X
AE4	UTP_OP_DATA[1]	ETHA_TXDATA[1]		X		X
AE5	UTP_IP_DATA[4]	ETHA_RXDV		X		X
AE6	UTP_IP_DATA[1]	ETHA_RXDATA[1]		X		X
AE7	VCCP			X	X	X
AE8	UTP_OP_ADDR[4]			X		X
AE9	UTP_OP_ADDR[1]			X		X
AE10	UTP_IP_FCO			X		X
AE11	OSC_VSSP			X	X	X
AE12	OSC_VSSP			X	X	X
AE13	UTP_IP_ADDR[3]			X		X
AE14	UTP_IP_ADDR[1]			X		X
AE15	VCCP			X	X	X
AE16	SCANTESTMODE_N			X	X	X
AE17	JTG_TDO			X	X	X
AE18	RXDATA1			X	X	X
AE19	RTS1_N			X	X	X
AE20	RTS0_N			X	X	X
AE21	GPIO[15]			X	X	X
AE22	GPIO[6]			X	X	X
AE23	GPIO[2]			X	X	X
AE24	SSP_SCLK			X	X	X
AE25	SSP_EXTCLK			X	X	X
AE26	VSS			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

Table 27. Processors' Ball Map Assignments (Sheet 26 of 26)

Ball	Signal Name			Processor Number		
	Configuration 1	Configuration 2	Configuration 3	Intel® IXP465	Intel® IXP460	Intel® IXP455
AF1	VSS			X	X	X
AF2	VSS			X	X	X
AF3	UTP_OP_DATA[0]	ETHA_TXDATA[0]		X		X
AF4	UTP_IP_DATA[2]	ETHA_RXDATA[2]		X		X
AF5	VSS			X	X	X
AF6	UTP_IP_DATA[0]	ETHA_RXDATA[0]		X		X
AF7	UTP_OP_CLK	ETHA_TXCLK		X		X
AF8	UTP_OP_ADDR[2]			X		X
AF9	UTP_OP_ADDR[0]			X		X
AF10	UTP_IP_CLK	ETHA_RXCLK		X		X
AF11	OSC_IN			X	X	X
AF12	VCCP			X	X	X
AF13	OSC_OUT			X	X	X
AF14	UTP_IP_ADDR[2]			X		X
AF15	BYPASS_CLK			X	X	X
AF16	PWRON_RESET_N			X	X	X
AF17	HIGHZ_N			X	X	X
AF18	JTG_TRST_N			X	X	X
AF19	VCCP			X	X	X
AF20	CTS1_N			X	X	X
AF21	VSS			X	X	X
AF22	GPIO[11]			X	X	X
AF23	GPIO[10]			X	X	X
AF24	GPIO[3]			X	X	X
AF25	VSS			X	X	X
AF26	VSS			X	X	X

NOTE: Interfaces not being utilized at a system level may require external pull-up or pull-down resistors. For specific details and requirements, see Section 4.2, "Functional Signal Definitions" on page 46.

NOTE: Configuration 1,2 and 3 are set by the Expansion bus configuration when Reset is deasserted.

Blank field indicates no physical ball on package.

4.4 Package Thermal Specifications

The thermal parameters defined in Table 28 and Table 29 are based on simulated results of packages assembled on standard multi-layer, 2s2p, 1.0-oz copper layer boards in a natural convection environment. The maximum case temperature is based on the maximum junction temperature and defined by the relationship:

$$T_{case}\ max = T_{jmax} - (\Psi_{JT} \times Power)$$

Where Ψ_{JT} is the junction-to-package top thermal characterization parameter. If the case temperature exceeds the specified T_{case} max, thermal enhancements such as heat sinks or forced air will be required. In the tables below, Θ_{JA} is the package junction-to-air thermal resistance.

Table 28. 2.8-Watt Maximum Power Dissipation

Package Type	Estimated Power (TPD)	Θ_{JA}	Ψ_{JT}	T_{case} Max. †
35mm HSBGA	2.8W	12.5°C/W	1.4 °C/W	116 °C
† This is a not to exceed maximum allowable case temperature.				

Table 29. 3.3-Watt Maximum Power Dissipation

Package Type	Estimated Power (TPD)	Θ_{JA}	Ψ_{JT}	T_{case} Max. †
35mm HSBGA	3.3W	12.5°C/W	1.4 °C/W	115 °C
† This is a not to exceed maximum allowable case temperature.				

5.0 Electrical Specifications

5.1 Absolute Maximum Ratings

Parameter	Maximum Rating
Ambient Air Temperature (Extended)	-40º C to 85º C
Ambient Air Temperature (Commercial)	0º C to 70º C
Supply Voltage Core	-0.3 V to 2.1 V
Supply Voltage I/O	-0.3 V to 3.6 V
Supply Voltage DDR	-0.3V to 2.75V
Supply Voltage Oscillator (V_{CCOSC})	-0.3 V to 2.1 V
Supply Voltage Oscillator (V_{CCOSCP})	-0.3 V to 3.6 V
Supply Voltage PLL (V_{CCPLL1})	-0.3 V to 2.1 V
Supply Voltage PLL (V_{CCPLL2})	-0.3 V to 2.1 V
Supply Voltage PLL (V_{CCPLL3})	-0.3 V to 2.1 V
Voltage On Any I/O Ball	-0.3 V to 3.6V
Storage Temperature	-55º C to 125º C

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.

5.2 V_{CCPLL1}, V_{CCPLL2}, V_{CCPLL3}, V_{CCOSCP}, V_{CCOSC} Pin Requirements

To reduce voltage-supply noise on the analog sections of the IXP45X/IXP46X network processors, the phase-lock loop circuits (V_{CCPLL1}, V_{CCPLL2}, V_{CCPLL3}) and oscillator circuit (V_{CCOSCP}, V_{CCOSC}) require isolated voltage supplies.

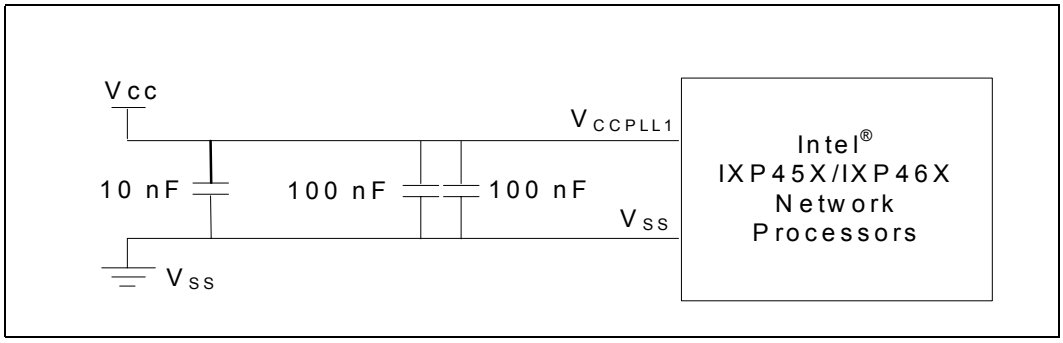
The filter circuits for each supply are shown in the following sections.

5.2.1 V_{CCPLL1} Requirement

A parallel combination of a 10-nF capacitor (for bypass) and a 200-nF capacitor (for a first-order filter with a cut-off frequency below 30 MHz) must be connected to the V_{CCPLL1} pin of the IXP45X/IXP46X network processors.

The ground of both capacitors should be connected to the nearest V_{SS} supply pin. Both capacitors should be located less than 0.5 inch away from the V_{CCPLL1} pin and the associated V_{SS} pin. In order to achieve the 200-nF capacitance, a parallel combination of two 100-nF capacitors may be used as long as the capacitors are placed directly beside each other.

Figure 9. V_{CCPLL1} Power Filtering Diagram

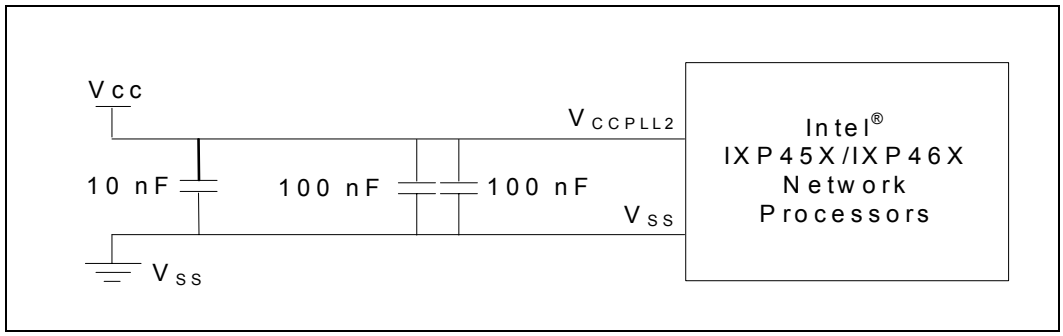


5.2.2 V_{CCPLL2} Requirement

A parallel combination of a 10-nF capacitor (for bypass) and a 200-nF capacitor (for a first-order filter with a cut-off frequency below 30 MHz) must be connected to the V_{CCPLL2} pin of the IXP45X/IXP46X network processors.

The ground of both capacitors should be connected to the nearest V_{SS} supply pin. Both capacitors should be located less than 0.5 inch away from the V_{CCPLL2} pin and the associated V_{SS} pin. In order to achieve the 200-nF capacitance, a parallel combination of two 100-nF capacitors may be used as long as the capacitors are placed directly beside each other.

Figure 10. V_{CCPLL2} Power Filtering Diagram

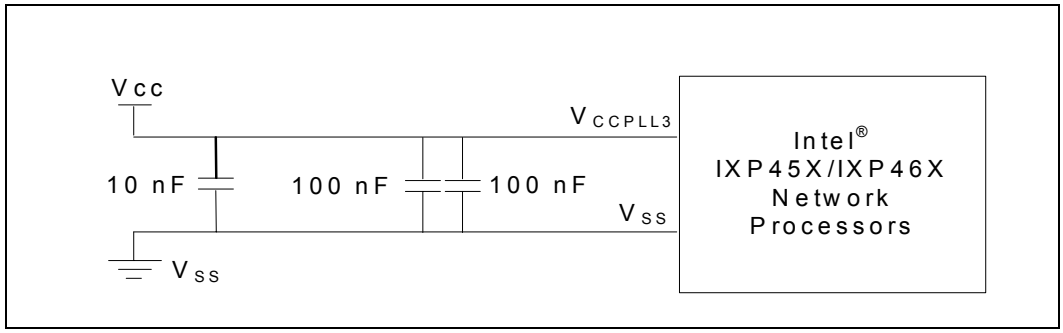


5.2.3 V_{CCPLL3} Requirement

A parallel combination of a 10-nF capacitor (for bypass) and a 200-nF capacitor (for a first-order filter with a cut-off frequency below 30 MHz) must be connected to the V_{CCPLL3} pin of the IXP45X/IXP46X network processors.

The ground of both capacitors should be connected to the nearest V_{SS} supply pin. Both capacitors should be located less than 0.5 inch away from the V_{CCPLL3} pin and the associated V_{SS} pin. In order to achieve the 200-nF capacitance, a parallel combination of two 100-nF capacitors may be used as long as the capacitors are placed directly beside each other.

Figure 11. V_{CCPLL3} Power Filtering Diagram



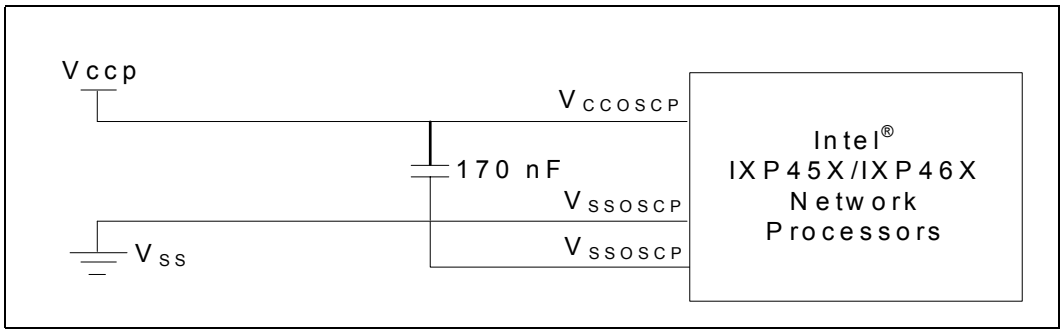
5.2.4 V_{CCOSCP} Requirement

A single, 170-nF capacitor must be connected between the V_{CCP_OSCP} pin and V_{SSP_OSCP} pin of the IXP45X/IXP46X network processors. This capacitor value provides both bypass and filtering.

When 170 nF is an inconvenient size, capacitor values between 150 nF to 200 nF can be used with little adverse effects, assuming that the effective series resistance of the 200-nF capacitor is under 50 mΩ

In order to achieve a 200-nF capacitance, a parallel combination of two 100-nF capacitors may be used as long as the capacitors are placed directly beside each other. V_{SSP_OSCP} is made up with two pins, AD10 and AF10. Ensure that both pins are connected as shown in Figure 12.

Figure 12. V_{CCOSCP} Power Filtering Diagram



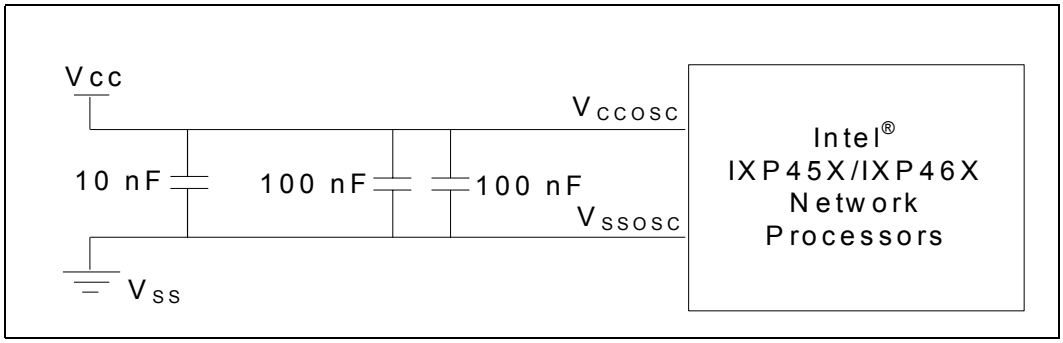
5.2.5 V_{CCOSC} Requirement

A parallel combination of a 10-nF capacitor (for bypass) and a 200-nF capacitor (for a first-order filter with a cut-off frequency below 33 MHz) must be connected to both of the V_{CCOSC} pins of the IXP45X/IXP46X network processors.

The grounds of both capacitors should be connected to the V_{SSOSC} supply pin. Both capacitors should be located less than 0.5 inch away from the V_{CCOSC} pin and the associated V_{SSOSC} pin.

In order to achieve a 200-nF capacitance, a parallel combination of two 100-nF capacitors may be used as long as the capacitors are placed directly beside each other.

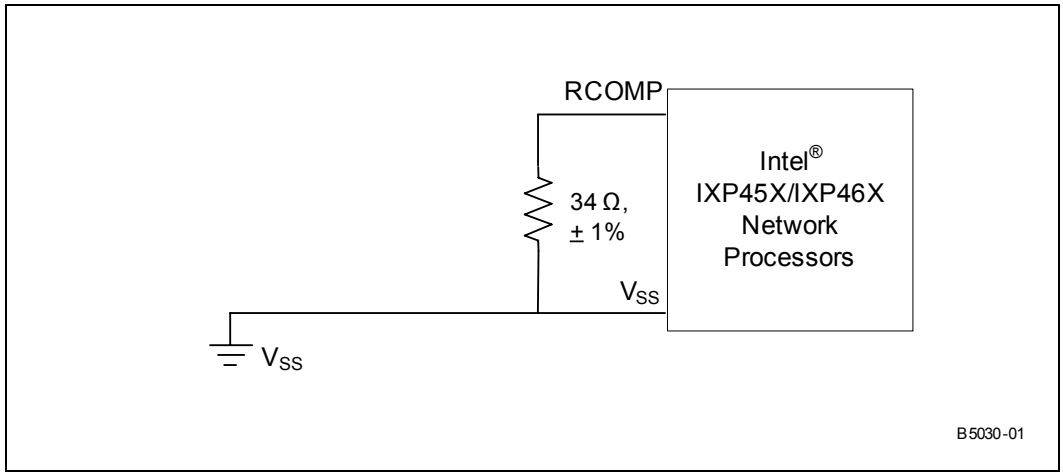
Figure 13. V_{CCOSC} Power Filtering Diagram



5.3 RCOMP Pin Requirements

Figure 14 shows the requirements for the RCOMP pin.

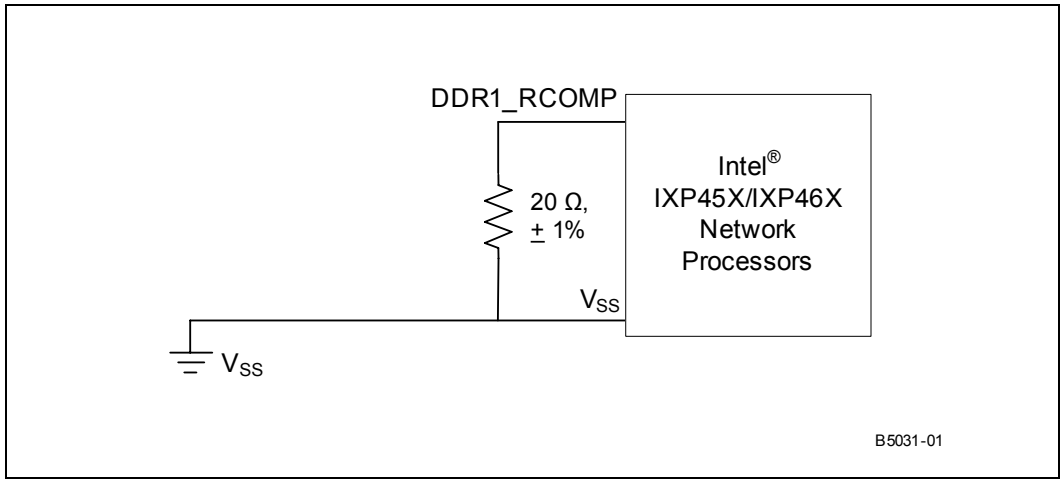
Figure 14. RCOMP Pin External Resistor Requirements



5.4 DDRI_RCOMP Pin Requirements

Figure 15 shows the requirements for the DDRI_RCOMP pin.

Figure 15. DDRI_RCOMP Pin External Resistor Requirements



5.5 DC Specifications

5.5.1 Operating Conditions

Table 30. Operating Conditions

Symbol	Parameter	Min.	Typ.	Max.	Units	Notes
V_{CCP}	Voltage supplied to the I/O, with the exception of the DDRI SDRAM Interface.	3.135	3.3	3.465	V	
V_{CC}	Voltage supplied to the internal logic. (266, 400 and 533MHz)	1.235	1.3	1.365	V	
	Voltage supplied to the internal logic. (667MHz)	1.33	1.4	1.47	V	
V_{CCM}	Voltage supplied to the DDRI SDRAM Interface.	2.300	2.5	2.700	V	
V_{CCOSC}	Voltage supplied to the internal oscillator logic.	1.235	1.3	1.365	V	
V_{CCOSCP}	Voltage supplied to the oscillator I/O.	3.135	3.3	3.465	V	
V_{CCPLL1}	Voltage supplied to the analog phase-lock loop.	1.235	1.3	1.365	V	
V_{CCPLL2}	Voltage supplied to the analog phase-lock loop.	1.235	1.3	1.365	V	

5.5.2 PCI DC Parameters

Table 31. PCI DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		0.5 V_{CCP}			V	3, 4
V_{IL}	Input-low voltage				0.3 V_{CCP}	V	3
V_{OH}	Output-high voltage	I_{OUT} = -500 µA	0.9 V_{CCP}			V	3
V_{OL}	Output-low voltage	I_{OUT} = 1500 µA			0.1 V_{CCP}	V	3
I_{IL}	Input-leakage current	0 < V_{IN} < V_{CCP}	-10		10	µA	1, 3
C_{IN}	Input-pin capacitance			5		pF	2, 3
C_{OUT}	I/O or output pin capacitance			5		pF	2,3
C_{IDSEL}	IDSEL-pin capacitance			5		pF	2,3
L_{PIN}	Pin inductance			20		nH	2,3

NOTES:
1. Input leakage currents include hi-Z output leakage for all bidirectional buffers with tri-state outputs.
2. These values are typical values seen by the manufacturing process and are not tested.
3. For additional information, see the *PCI Local Bus Specification*, Revision 2.2.
4. Please consult the *Intel® IXP4XX Product Line of Network Processors Specification Update* for the VIH specification.

5.5.3 USB 1.1 DC Parameters

Table 32. USB 1.1 DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	1
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = -6.1 * V_{OH}mA	2.8			V	
V_{OL}	Output-low voltage	IOUT = 6.1 * V_{OH}mA			0.3	V	
I_{IL}	Input-leakage current	0 < V_{IN} < V_{CCP}	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	2

NOTES:
1. Please consult the product specification update for the VIH specification.
2. These values are typical values seen by the manufacturing process and are not tested.

5.5.4 UTOPIA Level 2 DC Parameters

Table 33. UTOPIA Level 2 DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = -8 mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 8 mA			0.5	V	
I_{OH}	Output current at high voltage	V_{OH} > 2.4 V	-8			mA	
I_{OL}	Output current at low voltage	V_{OL} < 0.5 V	8			mA	
I_{IL}	Input-leakage current	0 < V_{IN} < V_{CCP}	-10		10	µA	1
C_{IN}	Input-pin capacitance			5		pF	2
C_{OUT}	I/O or output pin capacitance			5		pF	2

NOTES:
1. Input leakage currents include hi-Z output leakage for all bidirectional buffers with tri-state outputs.
2. These values are typical values seen by the manufacturing process and are not tested.

5.5.5 MII/SMII DC Parameters

Table 34. MII/SMII DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OHMII}	Output-high voltage	I_{OUT} = - 6 mA	2.4			V	
V_{OLMII}	Output-low voltage	I_{OUT} = 6 mA			0.4	V	
V_{OHSMII}	Output-high voltage	I_{OUT} = -10 mA	2.4			V	
V_{OLSMII}	Output-low voltage	I_{OUT} = 10mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.6 MDI DC Parameters

Table 35. MDI DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = - 6 mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 6 mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1
C_{INMDIO}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.7 DDRI SDRAM Bus DC Parameters

Table 36. DDRI SDRAM Bus DC Parameters (Sheet 1 of 2)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{DDRI_VREF}	I/O Reference voltage		$0.49*V_{CCM}$		$0.51*V_{CCM}$	V	
V_{IH}	Input-high voltage		V_{DDRI_VREF}+ 0.15		V_{CCM}+0.3	V	1
V_{IL}	Input-low voltage		-0.3		V_{DDRI_VREF}- 0.15	V	2
V_{OH}	Output-high voltage	I_{OUT} = -15mA	1.95			V	

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.
2. Only 2.5V DDRI SDRAM is supported

Table 36. DDRI SDRAM Bus DC Parameters (Sheet 2 of 2)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{OL}	Output-low voltage	I_{OUT} = 15mA			0.35	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCM}$	-10		10	µA	
C_{IO}	I/O-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.
2. Only 2.5V DDRI SDRAM is supported

5.5.8 Expansion Bus DC Parameters

Table 37. Expansion Bus DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OHDRV0}	Output-high voltage	I_{OUT} = -8 mA	2.4			V	1, 2
V_{OLDRV0}	Output-low voltage	I_{OUT} = 8 mA			0.4	V	1, 2
V_{OHDRV1}	Output-high voltage	I_{OUT} = -14 mA	2.4			V	1, 3
V_{OLDRV1}	Output-low voltage	I_{OUT} = 14mA			0.4	V	1, 3
V_{OHDRV2}	Output-high voltage	I_{OUT} = -20 mA	2.4			V	1, 4
V_{OLDRV2}	Output-low voltage	I_{OUT} = 20 mA			0.4	V	1, 4
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	2

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.
2. The values represented with this voltage parameter would typically be used in a system in which the expansion bus interfaces a single load of 6pF placed less than 2 inches away from the IXP45X/IXP46X network processors. This drive strength setting should be used to avoid ringing when minimal loading is attached. Please use IBIS models and simulation tools to guarantee the design.
3. The values represented with this voltage parameter would typically be used in a system in which the expansion bus interfaces four loads of 6pF each. All components are placed no further than 4 inches away from the IXP45X/IXP46X network processors. This drive strength setting should be used to avoid ringing when medium loading is attached. Please use IBIS models and simulation tools to guarantee the design.
4. The values represented with this voltage parameter would typically be used in a system in which the expansion bus interfaces eight loads of 6pF and all components are placed less than 6 inches from the IXP45X/IXP46X network processors. Another use case of this drive strength would typically be four loads of 6pF operating at 80MHz. This drive strength setting should be used to avoid ringing when maximum loading or frequency is utilized. Please use IBIS models and simulation tools to guarantee the design.

5.5.9 High-Speed, Serial Interface 0 DC Parameters

Table 38. High-Speed, Serial Interface 0 DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = - 6mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 6mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.10 High-Speed, Serial Interface 1 DC Parameters

Table 39. High-Speed, Serial Interface 1 DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = -6mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 6mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.11 UART DC Parameters

Table 40. UART DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = - 4mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 4mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.
2. This interface has been designed assuming a single load which can be between 5pF to 25pF.

5.5.12 Serial Peripheral Interface DC parameters

Table 41. Serial Peripheral Interface DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = - 6mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 6mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.13 I^2C Interface DC Parameters

Table 42. I^2C Interface DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = n/a	n/a	n/a	n/a	V	2
V_{OL}	Output-low voltage	I_{OUT} = 4mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.
2. Voltage output high for this interface is not applicable due to it being an open drain I/O.

5.5.14 GPIO DC Parameters

Table 43. GPIO DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage for GPIO 0 to GPIO 13	I_{OUT} = -16 mA	2.4			V	
V_{OL}	Output-low voltage for GPIO 0 to GPIO 13	I_{OUT} = 16 mA			0.4	V	
V_{OH}	Output-high voltage for GPIO 14 and GPIO 15	I_{OUT} = -4 mA	2.4			V	
V_{OL}	Output-low voltage for GPIO 14 and GPIO 15	I_{OUT} = 4 mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.15 JTAG DC Parameters

Table 44. JTAG DC Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = -4 mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 4 mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.

5.5.16 Reset DC Parameters

Table 45. PWRON_RESET _N and RESET_IN_N Parameters

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	

5.5.17 All Remaining I/O DC Parameters

Table 46. All Remaining I/O DC Parameters (JTAG, PLL_LOCK)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
V_{IH}	Input-high voltage		2.0			V	
V_{IL}	Input-low voltage				0.8	V	
V_{OH}	Output-high voltage	I_{OUT} = - 4mA	2.4			V	
V_{OL}	Output-low voltage	I_{OUT} = 4mA			0.4	V	
I_{IL}	Input-leakage current	$0 < V_{IN} < V_{CCP}$	-10		10	µA	
C_{IN}	Input-pin capacitance			5		pF	1

NOTES:
1. These values are typical values seen by the manufacturing process and are not tested.
2. These parameters are only applicable to signal other than power and ground signals.

5.6 AC Specifications

5.6.1 Clock Signal Timings

5.6.1.1 Processors' Clock Timings

Table 47. Devices' Clock Timings

Symbol	Parameter	Min.	Nom.	Max.	Units	Notes
V_{IH}	Input-high voltage	2.0			V	
V_{IL}	Input-low voltage			0.8	V	
$T_{FREQUENCY}$	Clock frequency for IXP45X/IXP46X network processors oscillator.	33.33	33.33	33.33	MHz	1, 3
$\triangle_{FREQUENCY}$	Clock tolerance over -40° C to 85° C.	-50		50	ppm	
C_{IN}	Pin capacitance of IXP45X/IXP46X network processors inputs.		5		pF	
T_{DC}	Duty cycle	35	50	65	%	2

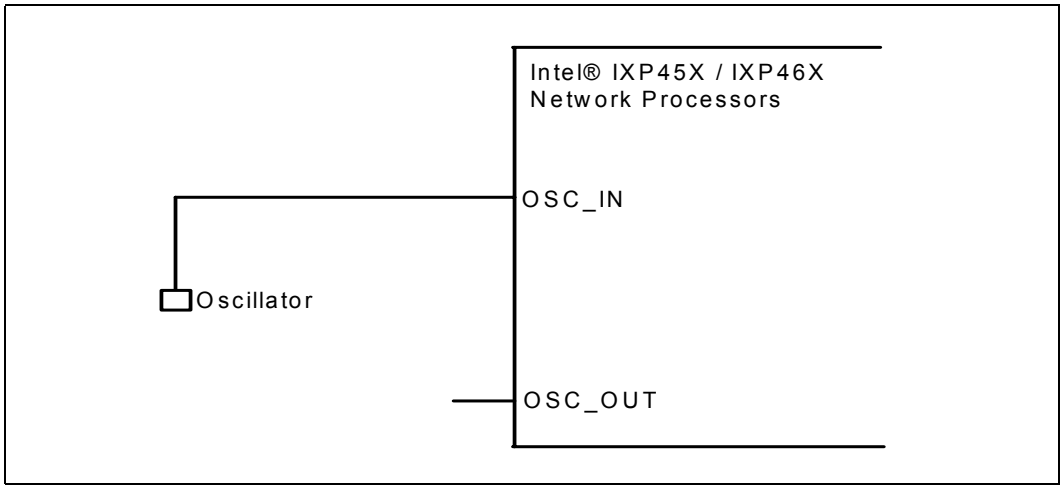
NOTES:
1. This value is oscillator input. Use as an oscillator input, tie to the crystal input pin and leave the crystal output pin disconnected.
2. This parameter applies when driving the clock input with an oscillator.
3. Where the IXP45X/IXP46X network processors are configured with an input reference-clock, the slew rate should never be faster than 2.5 V/nS to ensure proper PLL operation. To properly guarantee PLL operation at the slower slew rate, the Vih and Vil levels need to be met at the 33.33MHz frequency.

Table 48. Processors' Clock Timings Spread Spectrum Parameters

Spread-Spectrum Conditions	Min	Max	Notes
Frequency deviation from 33.33 MHz as a percentage	-2.0%	+0.0%	Characterized and guaranteed by design, but not tested. Do not over-clock the PLL input. The A.C. timings will not be guaranteed if the device exceeds 33.33 MHz.
Modulation Frequency		50 KHz	Characterized and guaranteed by design, but not tested

NOTES:
1. It is important to note that when using spread spectrum clocking, other clocks in the system will change frequency over a specific range. This change in other clocks can present some system level limitations. Please refer to the application note titled *Spread Spectrum Clocking to Reduce EMI Application Note*, when designing a product that utilizes spread spectrum clocking.

Figure 16. Typical Connection to an Oscillator



5.6.1.2 PCI Clock Timings

Table 49. PCI Clock Timings

Symbol	Parameter	33 MHZ		66 MHZ		Units	Notes
		Min.	Max.	Min.	Max.		
$T_{PERIODPCICLK}$	Clock period for PCI Clock	30		15		ns	
$T_{CLKHIGH}$	PCI Clock high time	11		6		ns	
T_{CLKLOW}	PCI Clock low time	11		6		ns	
$T_{SLEW\ RATE}$	Slew Rate requirements for PCI Clock	1	4	1.5	4	V/ns	

5.6.1.3 MII/SMII Clock Timings

Table 50. MII/SMII Clock Timings

Symbol	Parameter	Min.	Nom.	Max.	Units	Notes
$T_{period100Mbit}$	Clock period for SMII_CLK reference clock when operating in SMII or Source Synchronous mode of operation		8	8	ns	
$T_{period100Mbit}$	Clock period for SMII_TXCLK and SMII_RXCLK clock when operating in Source Synchronous SMII mode of operation		8	8	ns	
$T_{period100Mbit}$	Clock period for Tx and Rx Ethernet clocks		40	40	ns	
$T_{period10Mbit}$	Clock period for Tx and Rx Ethernet clocks		400	400	ns	
T_{duty}	Duty cycle for Tx and Rx Ethernet clocks	35	50	65	%	
$T_{rise/fall}$	Rise and fall time requirements for Tx and Rx Ethernet clocks			2	ns	

5.6.1.4 UTOPIA Level 2 Clock Timings

Table 51. UTOPIA Level 2 Clock Timings

Symbol	Parameter	Min.	Nom.	Max.	Units	Notes
T_{period}	Clock period for Tx and Rx UTOPIA Level 2 clocks			30.303	ns	
T_{duty}	Duty cycle for Tx and Rx UTOPIA Level 2 clocks	40	50	60	%	
$T_{rise/fall}$	Rise and fall time requirements for Tx and Rx UTOPIA Level 2 clocks			2	ns	

5.6.1.5 Expansion Bus Clock Timings

Table 52. Expansion Bus Clock Timings

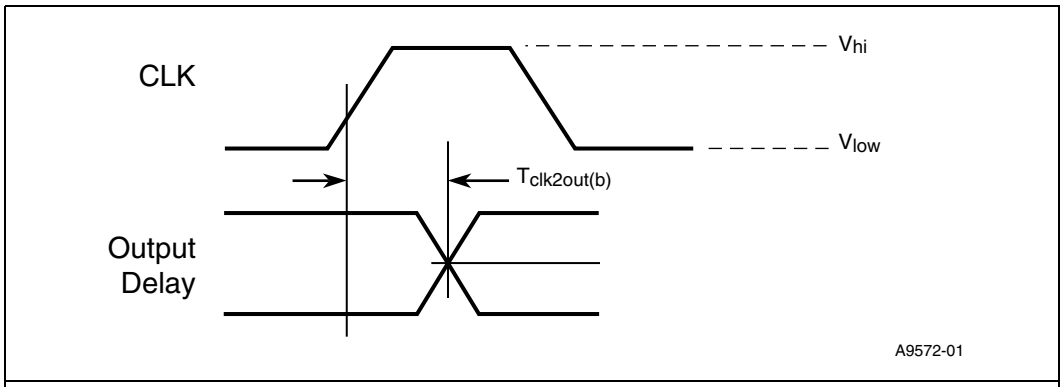
Symbol	Parameter	Min.	Nom.	Max.	Units	Notes
T_{period}	Clock period for expansion bus clock			12.5	ns	
T_{duty}	Duty cycle for expansion bus clock	40	50	60	%	
$T_{rise/fall}$	Rise and fall time requirements for expansion bus clock			2	ns	

5.6.2 Bus Signal Timings

The AC timing waveforms are shown in the following sections.

5.6.2.1 PCI

Figure 17. PCI Output Timing



NOTE: V_{HI} = 0.6 V_{CC} and V_{LOW} = 0.2 V_{CC}.

Figure 18. PCI Input Timing

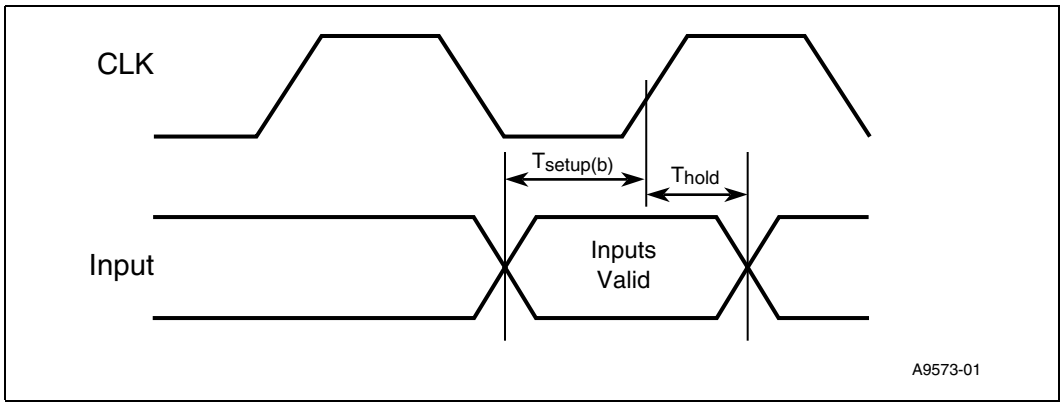


Table 53. PCI Bus Signal Timings

Symbol	Parameter	33 MHz		66 MHz		Units	Notes
		Min.	Max.	Min.	Max.		
$T_{clk2outb}$	Clock to output for all bused signals. This is the PCI_AD[31:0], PCI_CBE_N [3:0], PCI_PAR, PCI_FRAME_N, PCI_IRDY_N, PCI_TRDY_N, PCI_STOP_N, PCI_DEVSEL_N, PCI_PERR_N, PCI_SERR_N	2	11	1	6	ns	1, 2, 5, 7, 8
$T_{clk2out}$	Clock to output for all point-to-point signals. This is the PCI_GNT_N and PCI_REQ_N(0) only.	2	12	1	6	ns	1, 2, 5, 7, 8
T_{setupb}	Input setup time for all bused signals. This is the PCI_AD[31:0], PCI_CBE_N [3:0], PCI_PAR, PCI_FRAME_N, PCI_IRDY_N, PCI_TRDY_N, PCI_STOP_N, PCI_DEVSEL_N, PCI_PERR_N, PCI_SERR_N	7		3		ns	4, 6, 7, 8
T_{setup}	Input setup time for all point-to-point signals. This is the PCI_REQ_N and PCI_GNT_N(0) only.	10, 12		5		ns	3, 4, 7, 8
T_{hold}	Input hold time from clock.	0		0		ns	4, 7, 8
$T_{rst-off}$	Reset active-to-output float delay		40		40	ns	5, 6, 7, 8

NOTES:
1. See the timing measurement conditions.
2. Parts compliant to the 3.3 V signaling environment.
3. REQ# and GNT# are point-to-point signals and have different output valid delay and input setup times than do bused signals. GNT# has a setup of 10 ns for 33 MHz and 5 ns for 66 MHz; REQ# has a setup of 12 ns for 33 MHz and 5 ns for 66 MHz.
4. RST# is asserted and de-asserted asynchronously with respect to CLK.
5. All PCI outputs must be asynchronously driven to a tri-state value when RST# is active.
6. Setup time applies only when the device is not driving the pin. Devices cannot drive and receive signals at the same time.
7. Timing was tested with a 70-pF capacitor to ground.
8. For additional information, see the *PCI Local Bus Specification*, Revision 2.2.

5.6.2.2 USB 1.1 Interface

For timing parameters, see the USB 1.1 specification. The USB 1.1 interface for the IXP45X/ IXP46X network processors supports both a device or function controller only and a host only controller. The IXP45X/IXP46X network processors USB 1.1 device interface cannot be line-powered.

To assure proper operation with the IXP45X/IXP46X network processors USB interfaces, please consult the *Intel® IXP45X and Intel® IXP46X Product Line of Network Processors Hardware Design Guidelines* and the *Intel® IXP4XX Product Line of Network Processors Specification Update.*

5.6.2.3 UTOPIA Level 2

Figure 19. UTOPIA Level 2 Input Timings

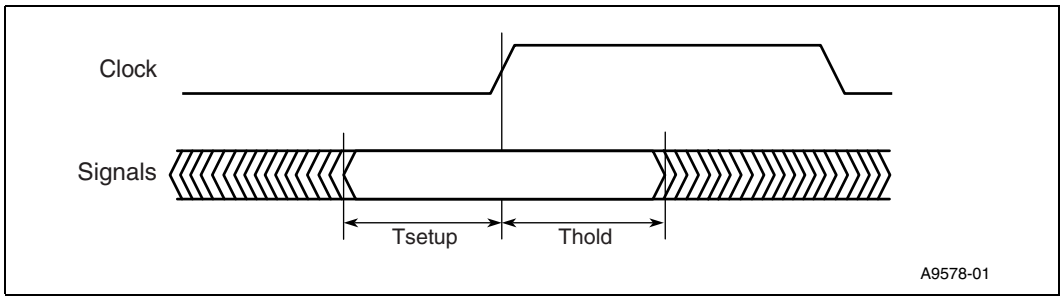


Table 54. UTOPIA Level 2 Input Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{setup}	Input setup prior to rising edge of clock. Inputs included in this timing are UTP_IP_DATA[7:0], UTP_IP_SOC, AND UTP_IP_FCI, and UTP_OP_FCI.	8		ns	
T_{hold}	Input hold time after the rising edge of the clock. Inputs included in this timing are UTP_IP_DATA[7:0], UTP_IP_SOC, and UTP_IP_FCI, and UTP_OP_FCI.	1		ns	

Figure 20. UTOPIA Level 2 Output Timings

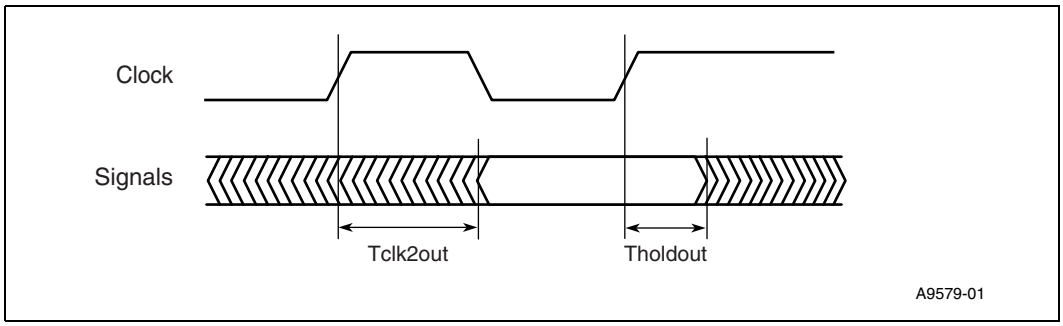


Table 55. UTOPIA Level 2 Output Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
$T_{clk2out}$	Rising edge of clock to signal output. Outputs included in this timing are UTP_OP_SOC, UTP_OP_FCO, UTP_IP_FCO, UTP_OP_DATA[7:0], UTP_IP_ADDR[4:0] and UTP_OP_ADDR[4:0].		17	ns	1
$T_{holdout}$	Signal output hold time after the rising edge of the clock. Outputs included in this timing are UTP_OP_SOC, UTP_OP_FCO, UTP_IP_FCO, UTP_OP_DATA[7:0], UTP_IP_ADDR[4:0] and UTP_OP_ADDR[4:0].	1		ns	1

NOTES:
1. Timing was designed for a system load between 5pF and 25pF

5.6.2.4 MII/SMII

Figure 21. SMII Output Timings

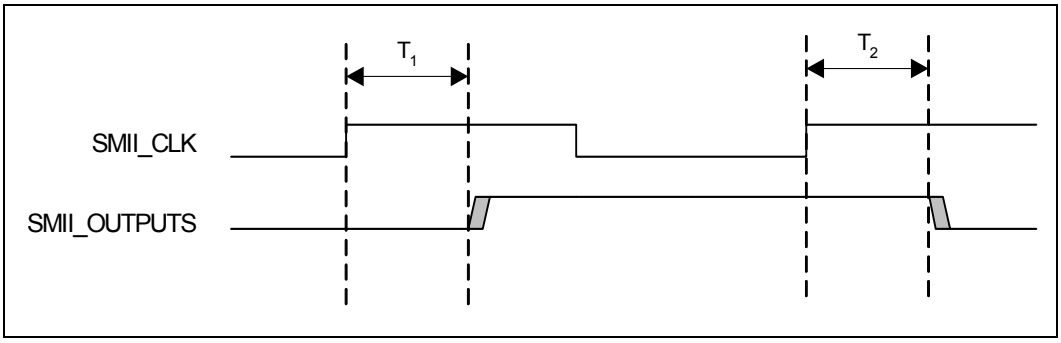


Table 56. SMII Output Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_1	Clock to output delay for SMII_TXD[4:0] and SMII_SYNC with respect to rising edge of SMII_CLK	1.5	5	ns	1
T_2	SMII_TXD[4:0] and SMII_SYNC hold time after SMII_CLK.	1.5		ns	1

NOTES:
1. Timing was designed for a system load between 5pF and 15pF

Figure 22. SMII Input Timings

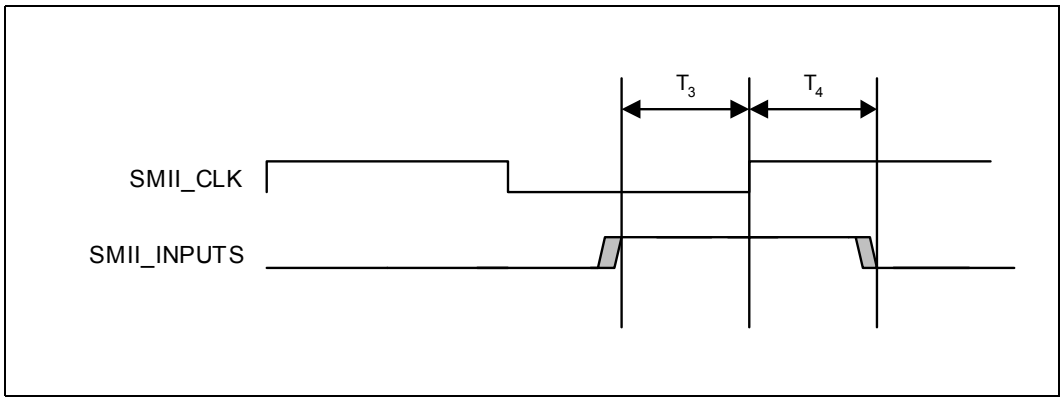


Table 57. SMII Input Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_3	SMII_RXD setup time prior to rising edge of SMII_CLK	1.5		ns	
T_4	SMII_RXD hold time after the rising edge of SMII_CLK	1		ns	

Figure 23. Source Synchronous SMII Output Timings

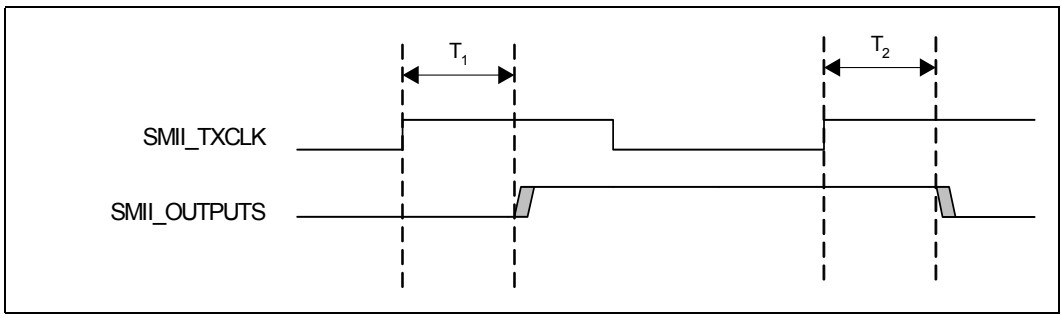


Table 58. Source Synchronous SMII Output Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_1	Clock to output delay for SMII_TXD[4:0] and SMII_TXSYNC with respect to rising edge of SMII_TXCLK	1.5	5	ns	1
T_2	SMII_TXD[4:0] and SMII_TXSYNC hold time after SMII_TXCLK.	1.5		ns	1

NOTES:
1. Timing was designed for a system load between 5pF and 15pF

Figure 24. Source Synchronous SMII Input Timings

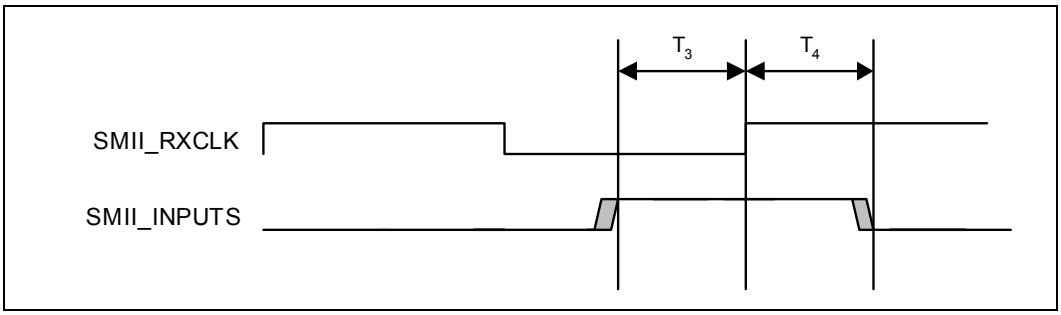


Table 59. Source Synchronous SMII Input Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_3	SMII_RXD and SMII_RXSYNC setup time prior to rising edge of SMII_RXCLK	1.5		ns	1
T_4	SMII_RXD and SMII_RXSYNC hold time after the rising edge of SMII_CLK	1		ns	1

NOTES:
1. Timing was designed for a system load between 5pF and 15pF

Figure 25. MII Output Timings

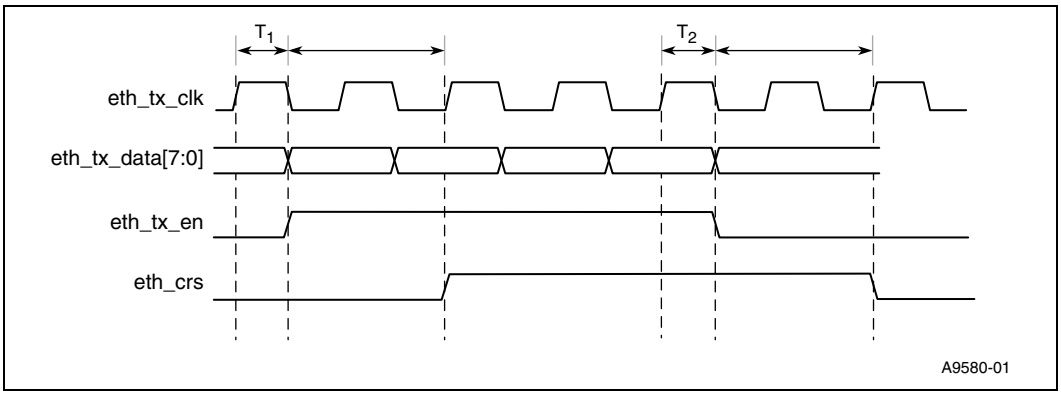


Table 60. MII Output Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_1	Clock to output delay for ETH_TXDATA and ETH_TXEN.		12.5	ns	1, 2
T_2	ETH_TXDATA and ETH_TXEN hold time after ETH_TXCLK.	1.5		ns	2

NOTES:
1. These values satisfy t the MII specification requirement of 0 ns to 25 ns clock to output delay.
2. Timing was designed for a system load between 5 pF and 15 pF.

Figure 26. MII Input Timings

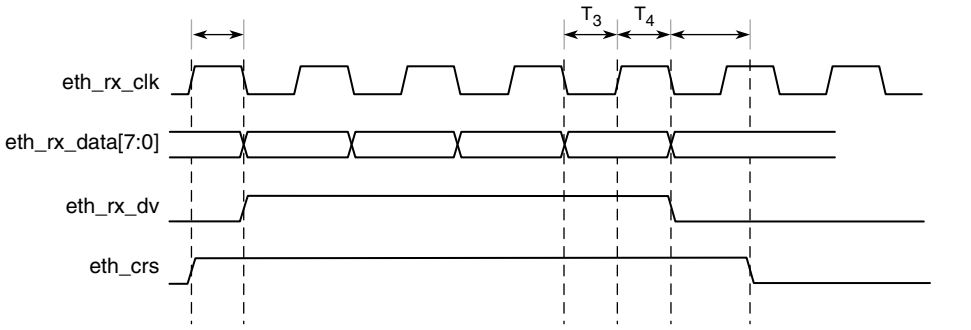


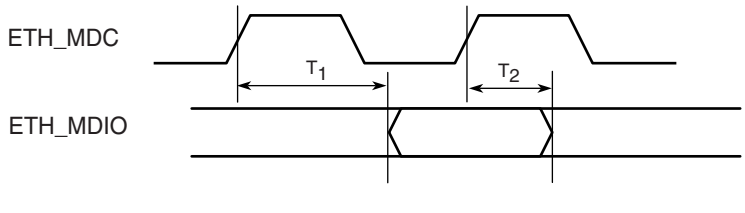
Table 61. MII Input Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_3	ETH_RXDATA and ETH_RXDV setup time prior to rising edge of ETH_RXCLK	5.5		ns	1
T_4	ETH_RXDATA and ETH_RXDV hold time after the rising edge of ETH_RXCLK	0		ns	1, 2

NOTES:
1. These values satisfying the 10-ns setup and hold time requirements necessary for the MII specification.
2. The T4 input hold timing parameter is not 100% tested and is guaranteed by design.

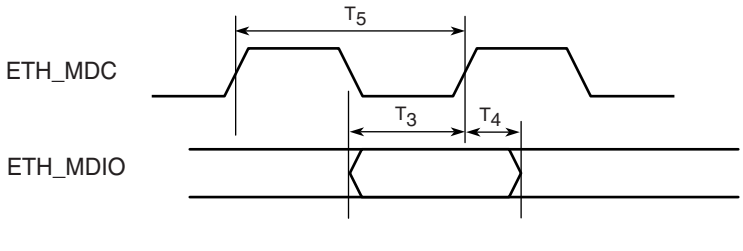
5.6.2.5 MDIO

Figure 27. MDIO Output Timings



NOTE: Processor is sourcing MDIO.

Figure 28. MDIO Input Timings



NOTE: PHY is sourcing MDIO.

Table 62. MDIO Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T1	ETH_MDIO, clock to output timing with respect to rising edge of ETH_MDC clock		ETH_MDC/2 + 15 ns	ns	
T2	ETH_MDIO output hold timing after the rising edge of ETH_MDC clock	10		ns	
T3	ETH_MDIO input setup prior to rising edge of ETH_MDC clock	3		ns	
T4	ETH_MDIO hold time after the rising edge of ETH_MDC clock	1		ns	
T5	ETH_MDC clock period	125	500	ns	1

NOTE:
1. Timing was designed for a system load between 5pF and 20pF

5.6.2.6 DDRI SDRAM Bus

Figure 29. DDRI SDRAM Write Timings

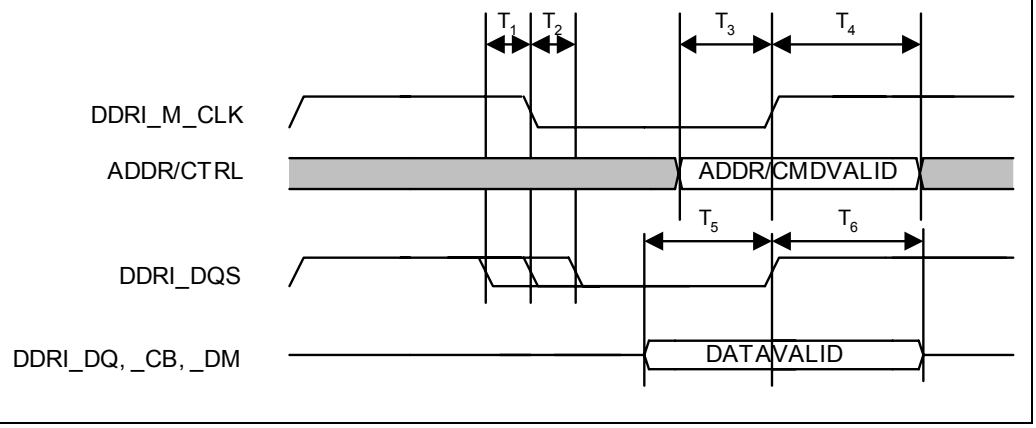


Table 63. DDRI SDRAM Write Timings Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_1	Output valid for DDRI_DQS prior to each edge of DDRI_M_CLK.		1.4	ns	1
T_2	DDRI_DQS output hold time after each edge of the DDRI_M_CLK.		1.0	ns	1
T_3	Output valid for ADDR/CTRL prior to the rising edge of DDRI_M_CLK. Address and control signals consist of DDRI_RAS_N, DDRI_CAS_N, DDRI_CS_N, DDRI_WE_N, DDRI_BA, DDRI_MA, and DDRI_CKE.	1.5		ns	1
T_4	ADDR/CTRL output hold time after the rising edge of the DDRI_M_CLK. Address and control signals consist of DDRI_RAS_N, DDRI_CAS_N, DDRI_CS_N, DDRI_WE_N, DDRI_BA, DDRI_MA, and DDRI_CKE.	1.5		ns	1
T_5	Output valid for DDRI_DQ, DDRI_CB, and DDRI_DM prior to each edge of DDRI_DQS.	1.0		ns	
T_6	DDRI_DQ, DDRI_CB, and DDRI_DM output hold time after each edge of the DDRI_DQS.	1.0		ns	

NOTES:
1. DDRI_M_CLK is representative of all DDRI_CK and DDRI_CK_N signals. The rising edge of DDRI_M_CLK represents the crossover point of the respective DDRI_CK and DDRI_CK_N signals. The skew between the separate DDR clocks have been compensated in the timings which have been described. The period to period clock jitter on each DDRI_M_CLK pair is spec'ed at +/-100ps.

Figure 30. DDRI SDRAM Read Timings (2.0 CAS Latency)

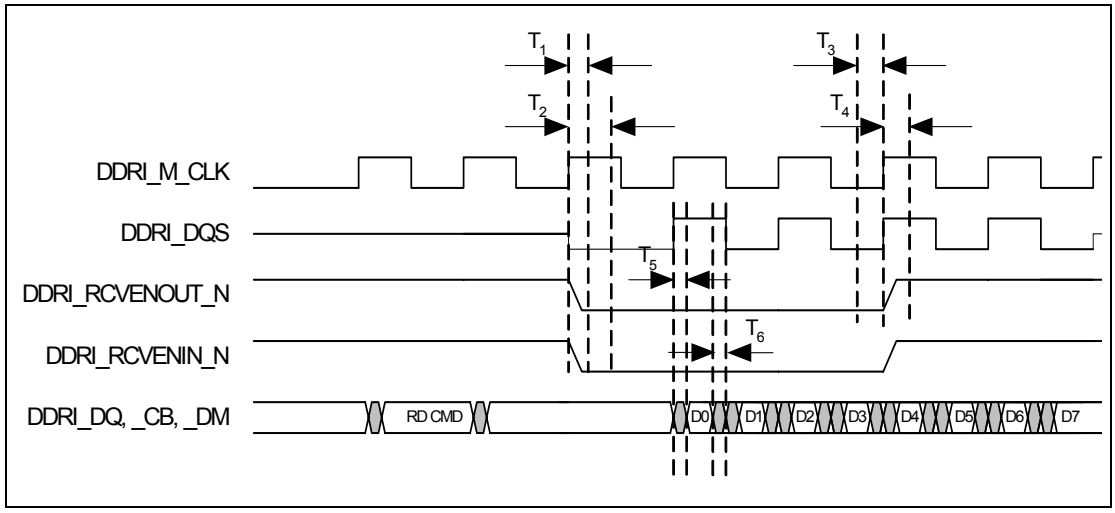


Figure 31. DDRI SDRAM Read Timings (2.5 CAS Latency)

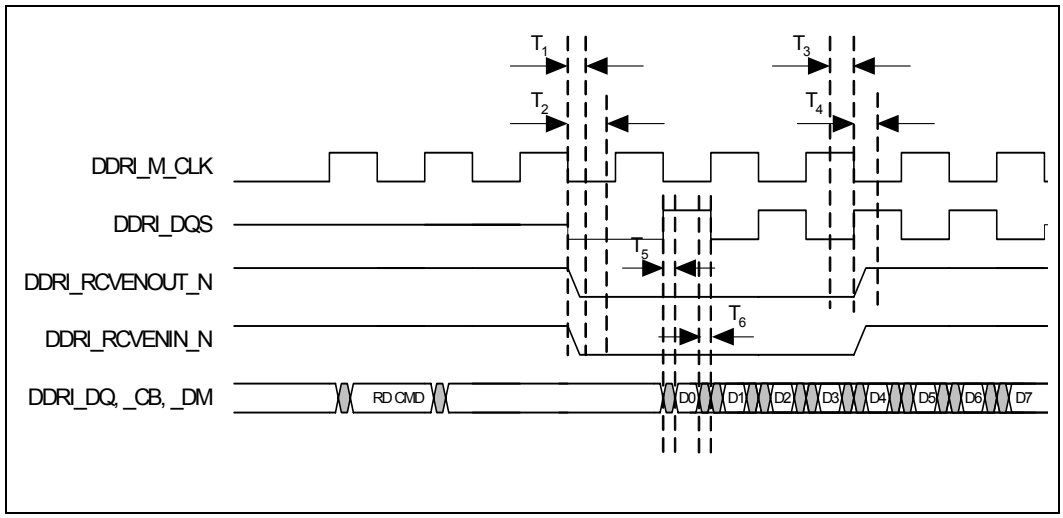


Table 64. DDRI SDRAM Read Timing Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_1	DDRI_RCVENOUT_N minimum output valid time after DDRI_M_CLK	0.9		ns	1
T_2	DDRI_RCVENOUT_N maximum output valid time after DDRI_M_CLK		2.7	ns	1
T_3	DDRI_RCVENIN_N input valid time before DDRI_DQS	3.6		ns	
T_4	DDRI_RCVENIN_N hold time from DDRI_DQS valid	-0.1		ns	
T_5	Maximum delay for Data valid after any edge of DDRI_DQS. Both of these signal are inputs from the memory during read operations.		0.75	ns	
T_6	Maximum guaranteed time before data begins to transition to the next valid data prior to any DDRI_DQS clock edge. Both of these signal are inputs from the memory during read operations. This time in conjunction with timing parameter T_5 specify the window for which the DDRI data signals can operate with the memory controller on the IXP45X/IXP46X network processors.		1.0	ns	

NOTES:
1. Designed to JEDEC specification, it is recommended that IBIS models be used to verify signal integrity on individual designs

5.6.2.7 Expansion Bus

5.6.2.7.1 Expansion Bus Synchronous Operation

Figure 32. Expansion Bus Synchronous Timing

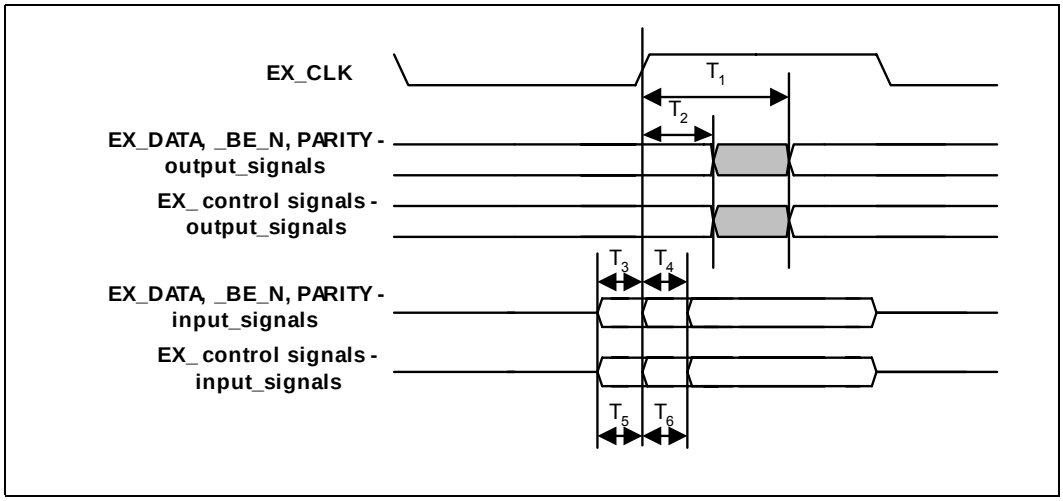


Table 65. Expansion Bus Synchronous Operation Timing Values

Symbol	Parameter	Low Drive		Med Drive		Hi Drive		Units	Notes
		Min.	Max.	Min.	Max.	Min.	Max.		
T_1	Valid rising edge of EX_CLK to valid signal on the output.		10		8.5		6.5	ns	1, 2, 3, 4
T_2	Valid signal hold time after the rising edge of EX_CLK	1		1		1		ns	1, 2, 3, 4
T_3	Valid data signal on an input prior to the rising edge of EX_CLK	2		2		2		ns	1, 2, 3, 4
T_4	Required hold time of a data input after the rising edge of EX_CLK	0.5		0.5		0.5		ns	1, 2, 3, 4
T_5	Valid control/arbiter signal on an input prior to the rising edge of EX_CLK	3.5		3.5		3.5		ns	1, 2, 3, 4
T_6	Required hold time of a control/arbiter input after the rising edge of EX_CLK	0.5		0.5		0.5		ns	1, 2, 3, 4

NOTES:
1. Timing was designed for a system load between 5pF and 60pF for low drive setting at typically no more than a 33MHz clock
2. Timing was designed for a system load between 5pF and 50pF for medium drive setting at typically no more than a 66MHz clock
3. Timing was designed for a system load between 5pF and 40pF for high drive setting at typically no more than a 80MHz clock
4. Drive settings do not apply to EX_CS_N signals and are expected to be point to point. The timing on this signal was designed for a system load between 5pF and 10pF
5. EX_control_signals output signals consist of EX_ALE, EX_ADDR, EX_CS_N, EX_GNT_REQ_N, EX_GNT_N, EX_RD_N, EX_WR_N, EX_WAIT_N
6. EX_control_signals input signals consist of EX_ADDR, EX_CS_N, EX_SLAVE_CS_N, EX_REQ_GNT_N, EX_REQ_N, EX_BURST, EX_RD_N, EX_WR_N

5.6.2.7.2 Expansion Bus Asynchronous Operation

Figure 33. Intel Multiplexed Mode

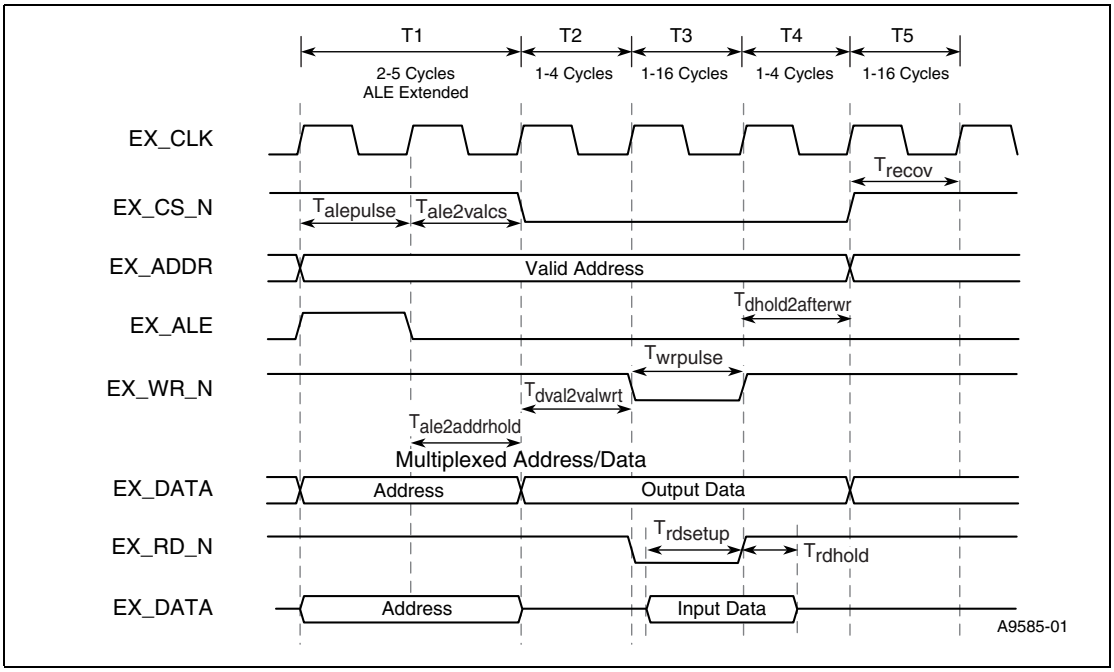


Table 66. Intel Multiplexed Mode Values (Sheet 1 of 2)

Symbol	Parameter	Min.	Max.	Units	Notes
Talepulse	Pulse width of ALE (ADDR is valid at the rising edge of ALE)	1	4	Cycles	1, 7
Tale2addrhold	Valid address hold time after from falling edge of ALE	1	1	Cycles	1, 2, 7
Tdval2valwrt	Write data valid prior to WR_N falling edge	1	4	Cycles	3, 7
Twrpulse	Pulse width of the WR_N	1	16	Cycles	4, 7
Tdholdafterwr	Valid data after the rising edge of WR_N	1	4	Cycles	5, 7
Tale2valcs	Valid chip select after the falling edge of ALE	1	4	Cycles	7

NOTES:
1. The EX_ALE signal is extended form T to 4T nnsec based on the programming of the T1 timing parameter. The parameter Tale2addrhold is fixed at T.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Table 66. Intel Multiplexed Mode Values (Sheet 2 of 2)

Symbol	Parameter	Min.	Max.	Units	Notes
Trdsetup	Data valid required before the rising edge of RD_N	5.3	14.7	ns	
Trdhold	Data hold required after the rising edge of RD_N	2		ns	
Trecov	Time needed between successive accesses on expansion interface.	1	16	Cycles	6

NOTES:
1. The EX_ALE signal is extended form T to 4T nnsec based on the programming of the T1 timing parameter. The parameter Tale2addrhold is fixed at T.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 34. Intel Simplex Mode

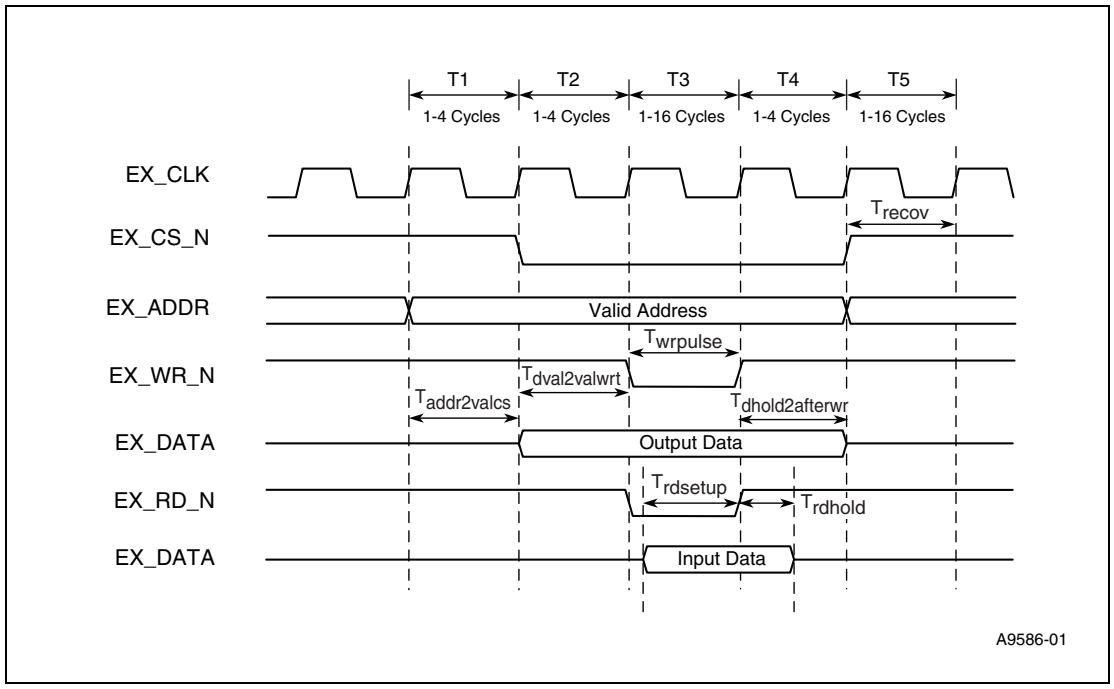


Table 67. Intel Simplex Mode Values

Symbol	Parameter	Min.	Max.	Units	Notes
$T_{addr2valcs}$	Valid address to valid chip select	1	4	Cycles	1, 2, 7
$T_{dval2valwrt}$	Write data valid prior to EXPB_IO_WRITE_N falling edge	1	4	Cycles	3, 7
$T_{wrpulse}$	Pulse width of the EXP_IO_WRITE_N	1	16	Cycles	4, 7
$T_{dholdafterwr}$	Valid data after the rising edge of EXPB_IO_WRITE_N	1	4	Cycles	5, 7
$T_{rdsetup}$	Data valid required before the rising edge of EXP_IO_READ_N	5.3	14.7	ns	
T_{rdhold}	Data hold required after the rising edge of EXP_IO_READ_N	2		ns	
T_{recov}	Time required between successive accesses on the expansion interface.	1	16	Cycles	6

NOTES:
1. EX_ALE is not valid in simplex mode of operation.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 35. Motorola* Multiplexed Mode

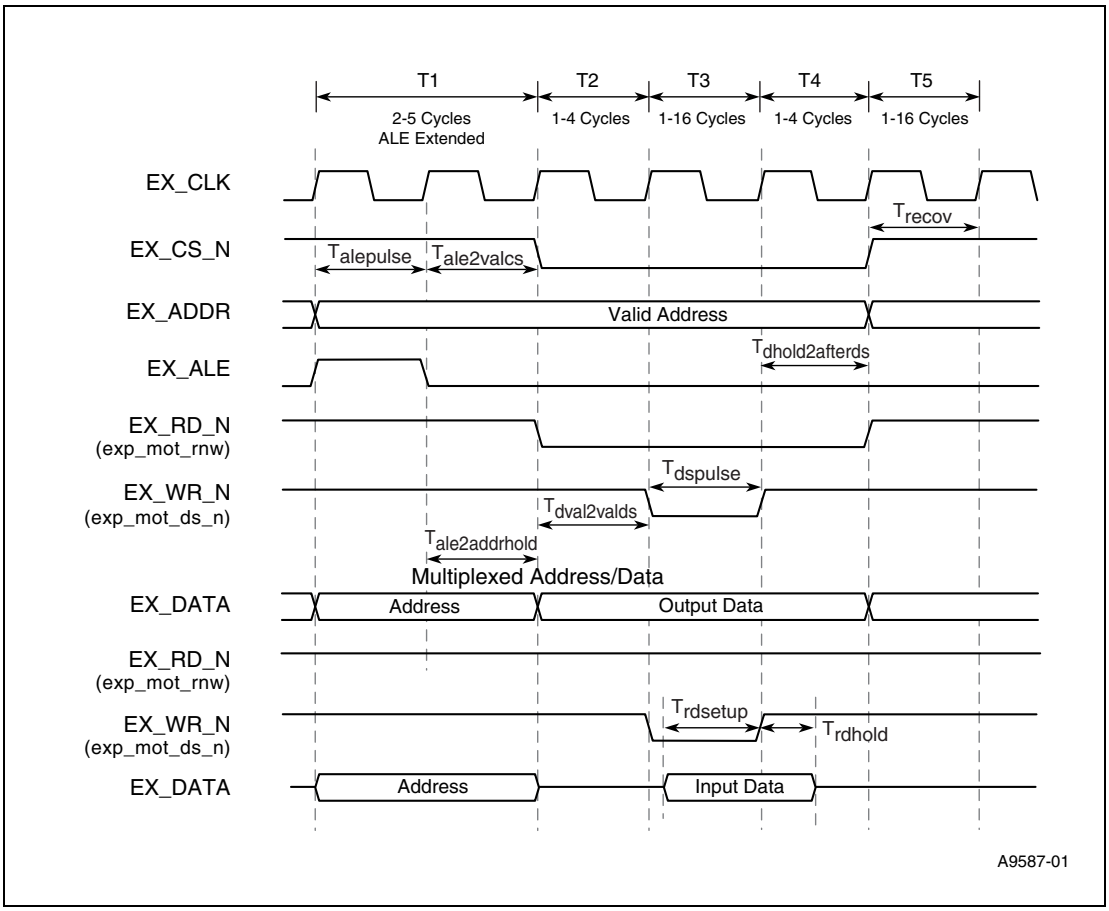


Table 68. Motorola* Multiplexed Mode Values (Sheet 1 of 2)

Symbol	Parameter	Min.	Max.	Units	Notes
$T_{alepulse}$	Pulse width of ALE (ADDR is valid at the rising edge of ALE)	1	4	Cycles	1, 7
$T_{ale2addrhold}$	Valid address hold time after from falling edge of ALE	1	1	Cycles	1, 2, 7
$T_{dval2valds}$	Write data valid prior to EXP_MOT_DS_N falling edge	1	4	Cycles	3, 7

NOTES:
1. The EX_ALE signal is extended form T to 4T nnsec, based on the programming of the T1 timing parameter. The parameter Tale2addrhold is fixed at T.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Table 68. Motorola* Multiplexed Mode Values (Sheet 2 of 2)

Symbol	Parameter	Min.	Max.	Units	Notes
$T_{dspulse}$	Pulse width of the EXP_MOT_DS_N	1	16	Cycles	4, 7
$T_{dholdafterds}$	Valid data after the rising edge of EXP_MOT_DS_N	1	4	Cycles	5, 7
$T_{ale2valcs}$	Valid chip select after the falling edge of ALE	1	4	Cycles	7
$T_{rdsetup}$	Data valid required before the rising edge of EXP_MOT_DS_N	5.3	14.7	ns	
T_{rdhold}	Data hold required after the rising edge of EXP_MOT_DS_N	2		ns	
T_{recov}	Time needed between successive accesses on expansion interface.	1	16	Cycles	6

NOTES:
1. The EX_ALE signal is extended form T to 4T nnsec, based on the programming of the T1 timing parameter. The parameter Tale2addrhold is fixed at T.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 36. Motorola* Simplex Mode

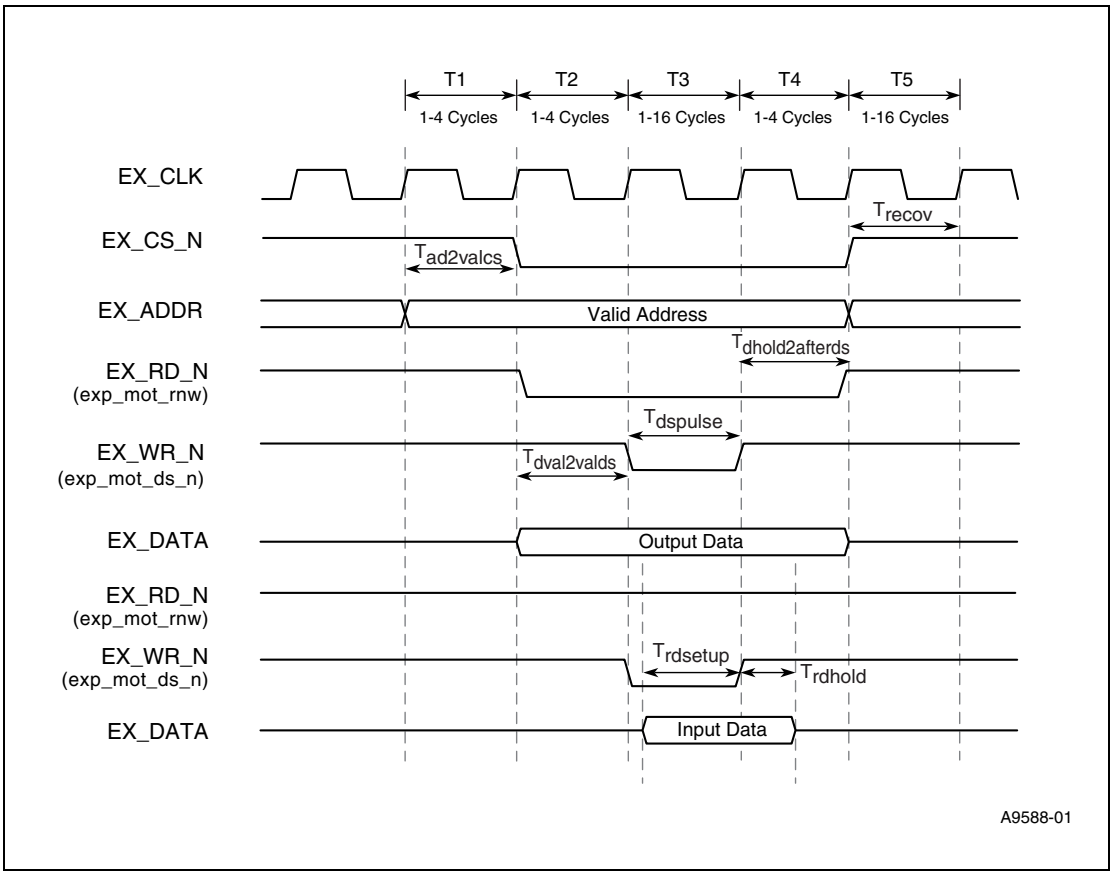


Table 69. Motorola* Simplex Mode Values (Sheet 1 of 2)

Symbol	Parameter	Min.	Max.	Units	Notes
$T_{ad2valcs}$	Valid address to valid chip select	1	4	Cycles	1, 2, 7
$T_{dval2valds}$	Write data valid prior to EXP_MOT_DS_N falling edge	1	4	Cycles	3, 7
$T_{dspulse}$	Pulse width of the EXP_MOT_DS_N	1	16	Cycles	4, 7
$T_{dholdafterds}$	Valid data after the rising edge of EXP_MOT_DS_N	1	4	Cycles	5, 7

NOTES:
1. EX_ALE is not valid in simplex mode of operation.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Table 69. Motorola* Simplex Mode Values (Sheet 2 of 2)

Symbol	Parameter	Min.	Max.	Units	Notes
$T_{rdsetup}$	Data valid required before the rising edge of EXP_MOT_DS_N	5.3	14.7	ns	
T_{rdhold}	Data hold required after the rising edge of EXP_MOT_DS_N	2		ns	
T_{recov}	Time required between successive accesses on the expansion interface.	1	16	Cycles	6

NOTES:
1. EX_ALE is not valid in simplex mode of operation.
2. Setting the address phase parameter (T1) will adjust the duration that the address appears to the external device.
3. Setting the data setup phase parameter (T2) will adjust the duration that the data appears prior to a data strobe (read or write) to an external device.
4. Setting the data strobe phase parameter (T3) will adjust the duration that the data strobe appears (read or write) to an external device. Data will be available during this time as well.
5. Setting the data hold strobe phase parameter (T4) will adjust the duration that the chip selects, address, and data (during a write) will be held.
6. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the expansion interface.
7. T is the period of the clock measured in ns.
8. Clock to output delay for all signals will be a maximum of 15 ns for devices requiring operation in synchronous mode.
9. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 37. HPI*–8 Mode Write Accesses

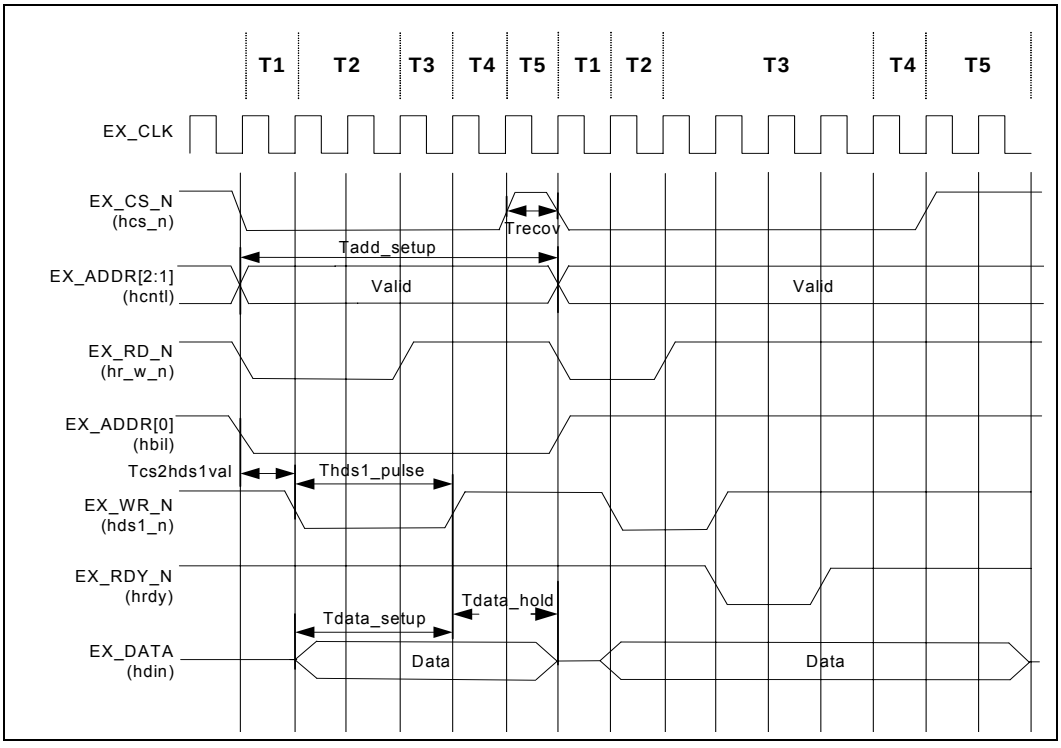


Table 70. HPI* Timing Symbol Description

State	Description	Min	Max	Unit	Notes
T1	Address Timing	3	4	Cycles	1, 5, 6
T2	Setup/Chip Select Timing	3	4	Cycles	2, 6
T3	Strobe Timing	2	16	Cycles	3, 5, 6
T4	Hold Timing	3	4	Cycles	6
T5	Recovery Phase	2	17	Cycles	6

Table 71. HPI*–8 Mode Write Accesses Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{add_setup}	Valid time that address is asserted on the line. The address is asserted at the same time as chip select.	11	45	Cycles	1, 5, 6
$T_{cs2hds1val}$	Delay from chip select being active and the HDS1 data strobe being active.	3	4	Cycles	5, 6
T_{hds1_pulse}	Pulse width of the HDS1 data strobe	4	5	Cycles	2, 4, 5
T_{data_setup}	Data valid prior to the rising edge of the HDS1 data strobe.	4	5	Cycles	3, 5, 6
T_{data_hold}	Data valid after the rising edge of the HDS1 data strobe.	4	36	Cycles	3, 6
T_{recov}	Time required between successive accesses on the expansion interface.	2	17	Cycles	4, 6

NOTES:
1. The address phase parameter (T1) must be set to a minimum value of 2. This value allows three T clocks for the address phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/ IXP46X network processors have had sufficient time to recognize the HRDY and hold the address phase for at least one clock pulse after the HRDY is de-active.
2. The data setup phase parameter (T2) must be set to a minimum value of 2. This value allows three T clocks for setup phase.
3. The data strobe phase parameter (T3) must be set to a minimum value of 1. This value allows two T clocks for the data phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/ IXP46X network processors have had sufficient time to recognize the HRDY and hold the data setup phase for at least one clock pulse after the HRDY is de-active
4. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the Expansion Bus interface.
5. HRDY can be asserted by the DSP at any point in the access. The interface will not leave states T1 or T3 until HRDY is de-active
6. One cycle is the period of the Expansion Bus clock.
7. Timing was designed for a system load between 5 pF and 60 pF for high drive setting.

Table 72. Setup/Hold Timing Values in Asynchronous Mode of Operation

Parameter	Min.	Max.	Units	Notes
Output Valid after rising edge of EX_CLK		10	ns	1
Output Hold after rising edge of EX_CLK	0		ns	1
Input Setup prior to rising edge of EX_CLK	3.5		ns	1
Input Hold required after rising edge of EX_CLK	0.5		ns	1

NOTES:
1. The Setup and Hold Timing values are for all modes.

Table 73. HPI*-16 Multiplexed Write Accesses Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{add_setup}	Valid time that address is asserted on the line. The address is asserted at the same time as chip select.	11	45	Cycles	1, 5, 6
$T_{cs2hds1val}$	Delay from chip select being active and the HDS1 data strobe being active.	3	4	Cycles	5, 6
T_{hds1_pulse}	Pulse width of the HDS1 data strobe	4	5	Cycles	2, 4, 5
T_{data_setup}	Data valid prior to the rising edge of the HDS1 data strobe.	4	5	Cycles	3, 5, 6
T_{data_hold}	Data valid after the rising edge of the HDS1 data strobe.	4	36	Cycles	3, 6
T_{recov}	Time required between successive accesses on the expansion interface.	2	17	Cycles	4, 6

NOTES:
1. The address phase parameter (T1) must be set to a minimum value of 2. This value allows three T clocks for the address phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/IXP46X network processors have had sufficient time to recognize the HRDY and hold the address phase for at least one clock pulse after the HRDY is de-active.
2. The data setup phase parameter (T2) must be set to a minimum value of 2. This value allows three T clocks for setup phase.
3. The data strobe phase parameter (T3) must be set to a minimum value of 1. This value allows two T clocks for the data phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/IXP46X network processors have had sufficient time to recognize the HRDY and hold the data setup phase for at least one clock pulse after the HRDY is de-active
4. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the Expansion Bus interface.
5. HRDY can be asserted by the DSP at any point in the access. The interface will not leave states T1 or T3 until HRDY is de-active
6. One cycle is the period of the Expansion Bus clock.
7. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 38. HPI*-16 Multiplexed Write Mode

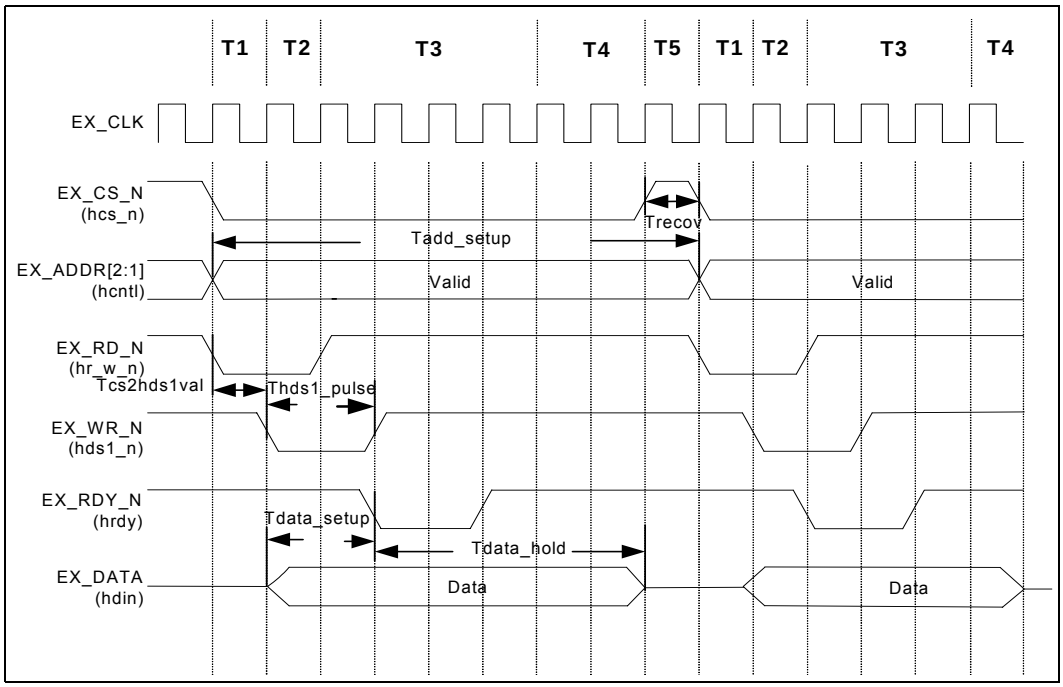


Table 74. HPI*-16 Multiplexed Read Accesses Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{add_setup}	Valid time that address is asserted on the line. The address is asserted at the same time as chip select.	11	45	Cycles	1, 5, 6
$T_{cs2hds1val}$	Delay from chip select being active and the HDS1 data strobe being active.	3	4	Cycles	5, 6
T_{hds1_pulse}	Pulse width of the HDS1 data strobe	4	5	Cycles	2, 4, 5
T_{data_setup}	Data is valid from the time from of the falling edge of HDS1_N to when the data is read.	4	5	Cycles	3, 5, 6
T_{recov}	Time required between successive accesses on the expansion interface.	2	17	cycles	4, 6

NOTES:
1. The address phase parameter (T1) must be set to a minimum value of 2. This value allows three T clocks for the address phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/ IXP46X network processors have had sufficient time to recognize the HRDY and hold the address phase for at least one clock pulse after the HRDY is de-active.
2. The data setup phase parameter (T2) must be set to a minimum value of 2. This value allows three T clocks for setup phase.
3. The data strobe phase parameter (T3) must be set to a minimum value of 1. This value allows two T clocks for the data phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/ IXP46X network processors have had sufficient time to recognize the HRDY and hold the data setup phase for at least one clock pulse after the HRDY is de-active
4. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the Expansion Bus interface.
5. HRDY can be asserted by the DSP at any point in the access. The interface will not leave states T1 or T3 until HRDY is de-active
6. One cycle is the period of the Expansion Bus clock.
7. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 39. HPI*-16 Multiplexed Read Mode

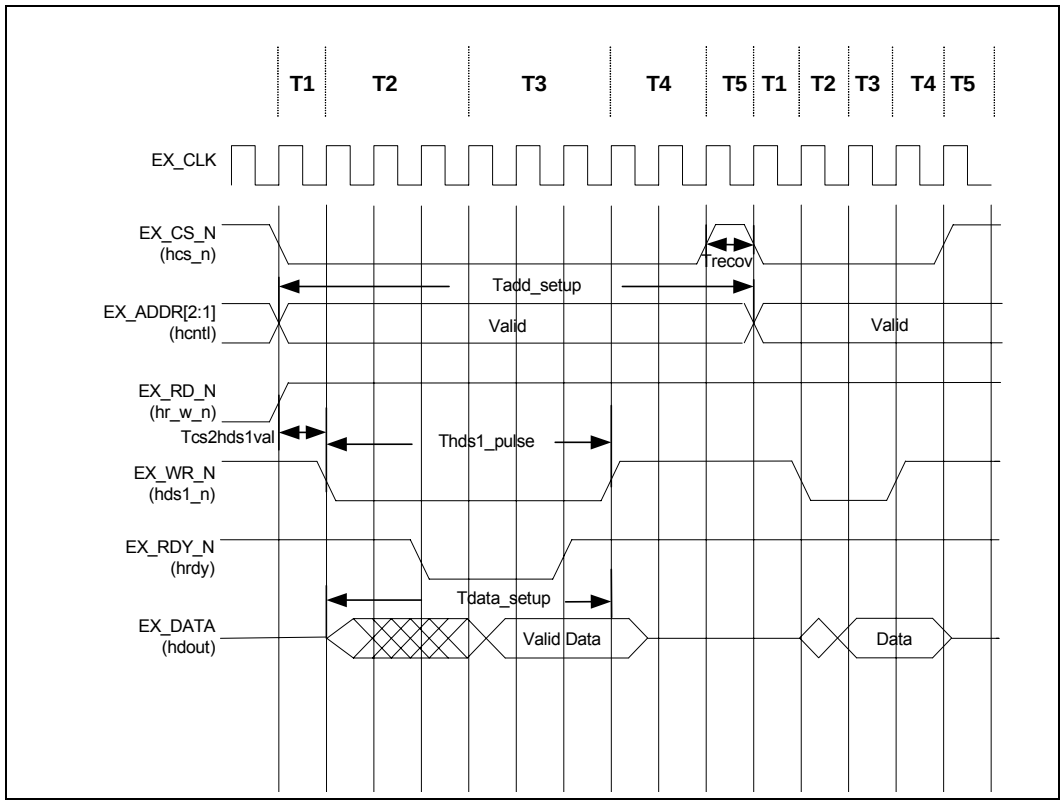


Table 75. HPI-16 Non-Multiplexed Read Accesses Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{add_setup}	Valid time that address is asserted on the line. The address is asserted at the same time as chip select.	11	45	Cycles	1, 5, 6
$T_{cs2hds1val}$	Delay from chip select being active and the HDS1 data strobe being active.	3	4	Cycles	5, 6
T_{hds1_pulse}	Pulse width of the HDS1 data strobe	4	5	Cycles	2, 4, 5
T_{data_setup}	Data is valid from the time from of the falling edge of HDS1_N to when the data is read.	4	5	Cycles	3, 5, 6
T_{recov}	Time required between successive accesses on the expansion interface.	2	17	Cycles	4, 6

NOTES:
1. The address phase parameter (T1) must be set to a minimum value of 2. This value allows three T clocks for the address phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/IXP46X network processors have had sufficient time to recognize the HRDY and hold the address phase for at least one clock pulse after the HRDY is de-active.
2. The data setup phase parameter (T2) must be set to a minimum value of 2. This value allows three T clocks for setup phase.
3. The data strobe phase parameter (T3) must be set to a minimum value of 1. This value allows two T clocks for the data phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/IXP46X network processors have had sufficient time to recognize the HRDY and hold the data setup phase for at least one clock pulse after the HRDY is de-active
4. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the Expansion Bus interface.
5. HRDY can be asserted by the DSP at any point in the access. The interface will not leave states T1 or T3 until HRDY is de-active
6. One cycle is the period of the Expansion Bus clock.
7. Timing was designed for a system load between 5pF and 60pF for high drive setting

Figure 40. HPI*-16 Non-Multiplexed Read Mode

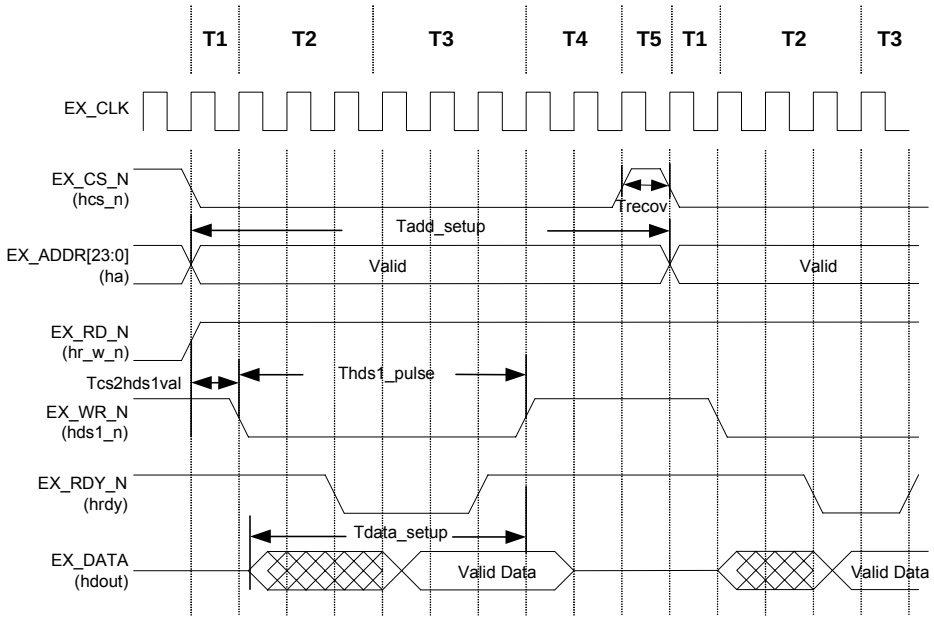
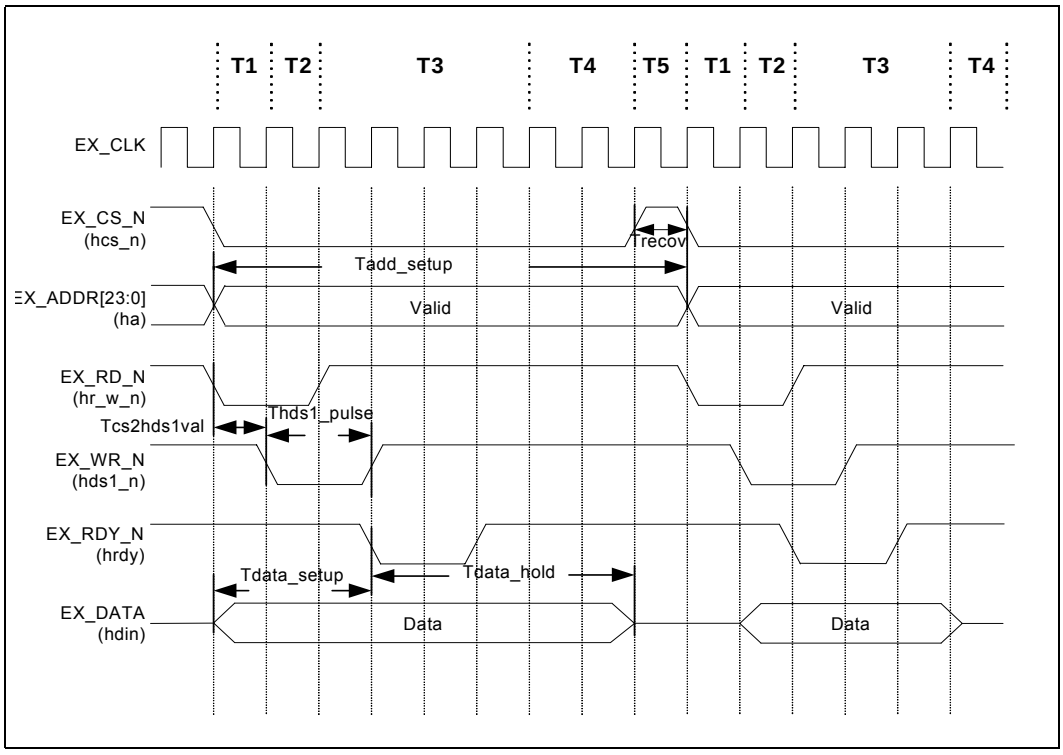


Table 76. HPI-16 Non-Multiplexed Write Accesses Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{add_setup}	Valid time that address is asserted on the line. The address is asserted at the same time as chip select.	11	45	Cycles	1, 5, 6
$T_{cs2hds1val}$	Delay from chip select being active and the HDS1 data strobe being active.	3	4	Cycles	5, 6
T_{hds1_pulse}	Pulse width of the HDS1 data strobe	4	5	Cycles	2, 4, 5
T_{data_setup}	Data valid prior to the rising edge of the HDS1 data strobe.	4	5	Cycles	3, 5, 6
T_{data_hold}	Data valid after the rising edge of the HDS1 data strobe.	4	36	Cycles	3, 6
T_{recov}	Time required between successive accesses on the expansion interface.	2	17	Cycles	4, 6

NOTES:
1. The address phase parameter (T1) must be set to a minimum value of 2. This value allows three T clocks for the address phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/IXP46X network processors have had sufficient time to recognize the HRDY and hold the address phase for at least one clock pulse after the HRDY is de-active.
2. The data setup phase parameter (T2) must be set to a minimum value of 2. This value allows three T clocks for setup phase.
3. The data strobe phase parameter (T3) must be set to a minimum value of 1. This value allows two T clocks for the data phase. This setting is required to ensure that in the event of an HRDY, the IXP45X/IXP46X network processors have had sufficient time to recognize the HRDY and hold the data setup phase for at least one clock pulse after the HRDY is de-active
4. Setting the recovery phase parameter (T5) will adjust the duration between successive accesses on the Expansion Bus interface.
5. HRDY can be asserted by the DSP at any point in the access. The interface will not leave states T1 or T3 until HRDY is de-active
6. One cycle is the period of the Expansion Bus clock.
7. Timing was designed for a system load between 5 pF and 60 pF for high drive setting

Figure 41. HPI*-16 Non-Multiplexed Write Mode



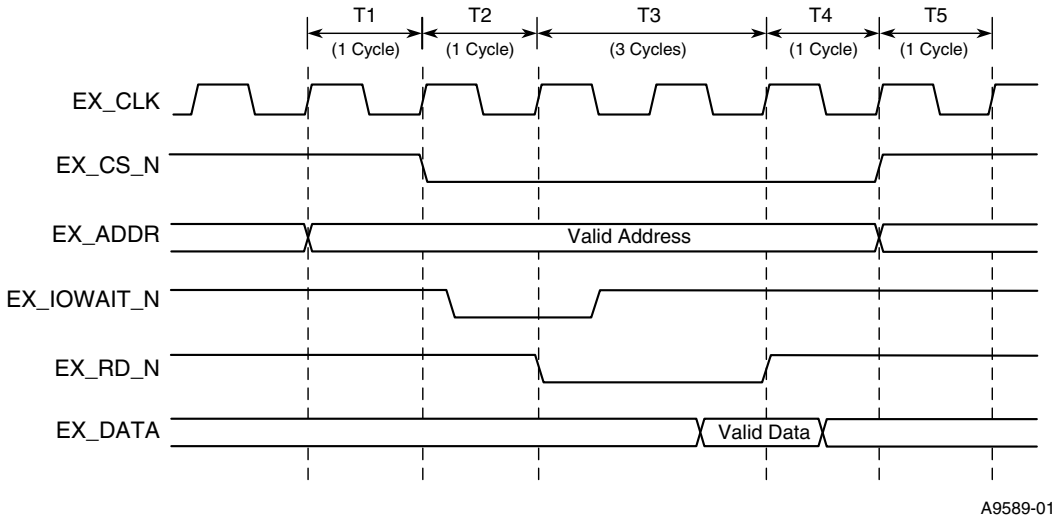
5.6.2.7.3 EX_IOWAIT_N

The EX_IOWAIT_N signal is available to be shared by devices attached to chip selects 0 through 7, when configured in Intel or Motorola modes of operation. The shared device will assert EX_IOWAIT_N in the T2 phase or a read or write transaction. During idle cycles, the board is responsible for ensuring that EX_IOWAIT_N is pulled-up. The Expansion bus controller will always ignore EX_IOWAIT_N for Synchronous Intel mode writes.

When an external device asserts EX_IOWAIT_N before the first cycle of a Strobe phase of a read or write transaction, the Expansion bus controller will hold in the Strobe phase until the EX_IOWAIT_N signal returns to an inactive state. Since there is a synchronizer cell on EX_IOWAIT_N, the external device must assert EX_IOWAIT_N three cycles before the deassertion of EX_WR_N/EX_RD_N. This implies that the value programmed in the T2 and T3 phase cannot both be equal to zero. After EX_IOWAIT_N is deasserted the Expansion bus controller will transition to the T4 - Hold state after the T3 counter reaches zero.

The EX_IOWAIT_N signal only affects the interface during the strobe phase of a read/write transfer. If Chip Selects 4 through 7 are configured in HPI mode of operation, each chip select will have a corresponding HRDY signal called EX_RDY. The polarity of the ready signal is programmable. Chip Select 4 corresponds to EX_RDY signal 0 and Chip Select 7 corresponds to EX_RDY signal 3.

Figure 42. Expansion Bus I/O Wait



NOTE: Notice that the access is an Intel-style simplex read access. The data strobe phase is set to a value to last three clock cycles. The data is returned from the peripheral device prior to the three clocks and the peripheral device de-asserts EX_IOWAIT_N. The data strobe phase terminates after two clocks even though the strobe phase was configured to pulse for three clocks.

5.6.2.8 Serial Peripheral Port Interface Timing

Figure 43. Serial Peripheral Interface Timing

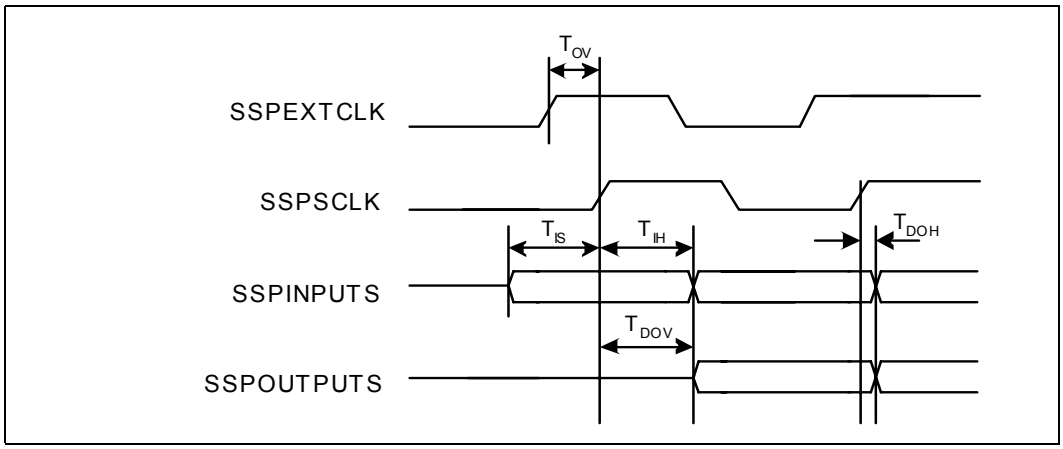


Table 77. Serial Peripheral Port Interface Timing Values

Symbol	Parameter	Min.	Max.	Units	Notes
T_{PER_INTCLK}	Minimum and maximum clock periods which can be produced by the SSP_SCLK when the clock is being generated from the internal 3.7033MHz clock.	.0072	1.8432	MHz	
T_{PER_EXTCLK}	Minimum and maximum clock period which can be produced by the SSP_SCLK when the clock is being generated from the externally supplied maximum clock rate of 33 MHz clock (SSP_EXTCLK).	.06445	16.5	MHz	
T_{IS}	Input Setup time for data prior to the valid edge of SSP_SCLK. These signals include SSP_SRXD.	15		ns	
T_{IH}	Input hold time for data after the to the valid edge of SSP_SCLK. These signals include SSP_SRXD.	0		ns	
T_{DOV}	SSP_SCLK clock to output valid delay from output signals. These signals include SSP_STXD and SSP_SFRM.	1	6	ns	
T_{DOH}	Output data hold valid from valid edge of SSP_SCLK. These signals include SSP_STXD and SSP_SFRM.	1		ns	
T_{OV}	Output Valid Delay from SSP_EXTCLK to SSP_SCLK in external clock mode	2	15	ns	

1. Timing was designed for a system load between 5pF and 40pF
2. Clock jitter on the SSPSCLK is designed to be an average of the specified clock frequency. The SSPSCLK jitter specification is unspecified.

5.6.2.9 I²C Interface Timing

Figure 44. I²C Interface Timing

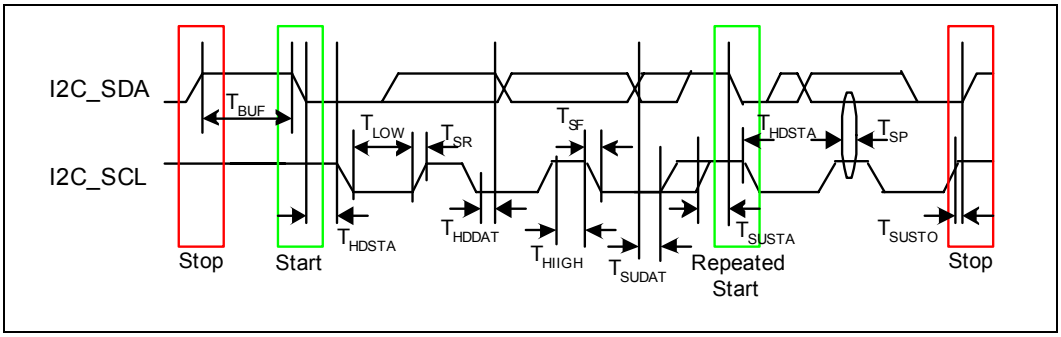


Table 78. I²C Interface Timing Values

Symbol	Parameter	Min.	Max.	Min.	Max.	Units	Notes
FSCL	SCL Clock Frequency	0	100	0	400	KHz	
TBUF	Bus Free Time Between STOP and START Condition	4.7		1.3		µs	
THDSTA	Hold Time (repeated) START Condition	4		0.6		µs	2
TLOW	SCL Clock Low Time	4.7		1.3		µs	1
THIGH	SCL Clock High Time	4		0.6		µs	1
TSUSTA	Setup Time for a Repeated START Condition	4.7		0.6		µs	
THDDAT	Data Hold Time	0	3.45	0	0.9	µs	
TSUDAT	Data Setup Time	250		100		ns	
TSR	SCL and SDA Rise Time		1000	20+0.1Cb	300	ns	
TSF	SCL and SDA Fall Time		300	20+0.1Cb	300	ns	
TSUSTO	Setup Time for STOP Condition	4		0.6		µs	

NOTES:
1. Not tested
2. After this period, the first clock pulse is generated

5.6.2.10 High-Speed, Serial Interfaces

Figure 45. High-Speed, Serial Timings

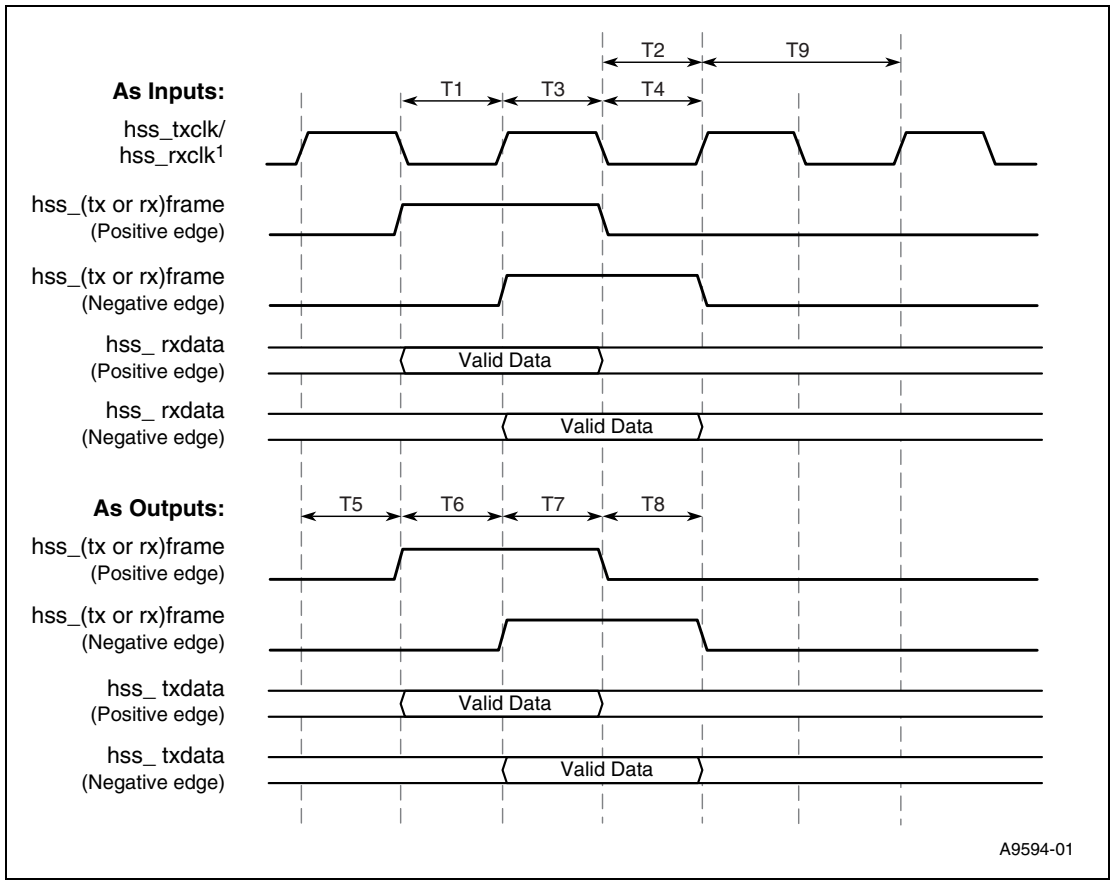


Table 79. High-Speed, Serial Timing Values

Symbol	Parameter	Min.	Max.	Units	Notes
T1	Setup time of HSS_TXFRAME, HSS_RXFRAME, and HSS_RXDATA prior to the rising edge of clock	5		ns	1, 2, 3
T2	Hold time of HSS_TXFRAME, HSS_RXFRAME, and HSS_RXDATA after the rising edge of clock	0		ns	1, 2, 3
T3	Setup time of HSS_TXFRAME, HSS_RXFRAME, and HSS_RXDATA prior to the falling edge of clock	5		ns	1, 2, 3
T4	Hold time of HSS_TXFRAME, HSS_RXFRAME, and HSS_RXDATA after the falling edge of clock	0		ns	1, 2, 3
T5	Rising edge of clock to output delay for HSS_TXFRAME, HSS_RXFRAME, and HSS_TXDATA		15	ns	1, 4
T6	Falling edge of clock to output delay for HSS_TXFRAME, HSS_RXFRAME, and HSS_TXDATA		15	ns	1, 3, 4
T7	Output Hold Delay after rising edge of final clock for HSS_TXFRAME, HSS_RXFRAME, and HSS_TXDATA	0		ns	1, 3, 4
T8	Output Hold Delay after falling edge of final clock for HSS_TXFRAME, HSS_RXFRAME, and HSS_TXDATA	0		ns	1, 3, 4
T9	HSS_TXCLK period and HSS_RXCLK period	1/8.192 MHz	1/512 KHz	ns	5

NOTES:
1. HSS_TXCLK and HSS_RXCLK may be coming from external independent sources or being driven by the IXP45X/IXP46X network processors. The signals are shown to be synchronous for illustrative purposes and are not required to be synchronous.
2. Applicable when the HSS_RXFRAME and HSS_TXFRAME signals are being driven by an external source as inputs into the IXP45X/IXP46X network processors. Always applicable to HSS_RXDATA.
3. The HSS_RXFRAME and HSS_TXFRAME can be configured to accept data on the rising or falling edge of the given reference clock. HSS_RXFRAME and HSS_RXDATA signals are synchronous to HSS_RXCLK and HSS_TXFRAME and HSS_TXDATA signals are synchronous to the HSS_TXCLK.
4. Applicable when the HSS_RXFRAME and HSS_TXFRAME signals are being driven by the IXP45X/IXP46X network processors to an external source. Always applicable to HSS_TXDATA.
5. The HSS_TXCLK can be configured to be driven by an external source or be driven by the IXP45X/IXP46X network processors. The slowest clock speed that can be accepted or driven is 512 KHz. The maximum clock speed that can be accepted or driven is 8.192 MHz. The clock duty cycle accepted will be 50/50 + 20%.
6. Timing was designed for a system load between 5 pF and 30 pF for high drive setting

5.6.2.11 JTAG

Figure 46. Boundary-Scan General Timings

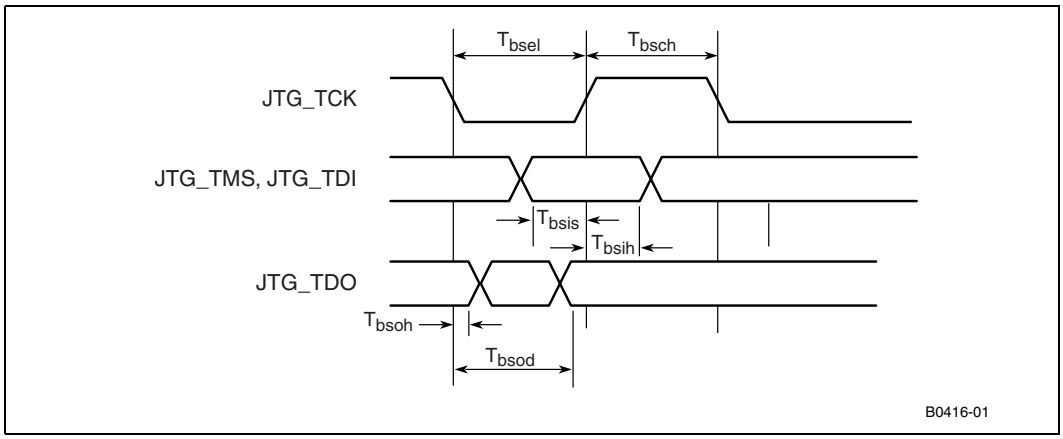


Figure 47. Boundary-Scan Reset Timings

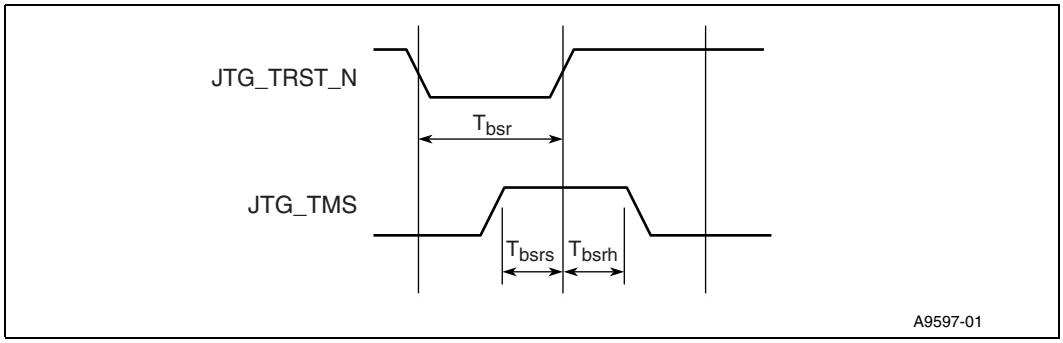


Table 80. Boundary-Scan Interface Timings Values

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Units	Notes
T_{bscl}	JTG_TCK low time		50			ns	2
T_{bsch}	JTG_TCK high time		50			ns	2
T_{bsis}	JTG_TDI, JTG_TMS setup time to rising edge of JTG_TCK		10			ns	
T_{bsih}	JTG_TDI, JTG_TMS hold time from rising edge of JTG_TCK		10			ns	
T_{bsoh}	JTG_TDO hold time after falling edge of JTG_TCK		1.5			ns	1
T_{bsod}	JTG_TDO clock to output from falling edge of JTG_TCK				40	ns	1
T_{bsr}	JTG_TRST_N reset period		30			ns	
T_{bsrs}	JTG_TMS setup time to rising edge of JTG_TRST_N		10			ns	
T_{bsrh}	JTG_TMS hold time from rising edge of JTG_TRST_N		10			ns	

NOTES:
1. Tests completed with a 40-pF load to ground on JTAG_TDO.
2. JTG_TCK may be stopped indefinitely in either the low or high phase.

5.6.3 Reset Timings

Figure 48. Reset Timings

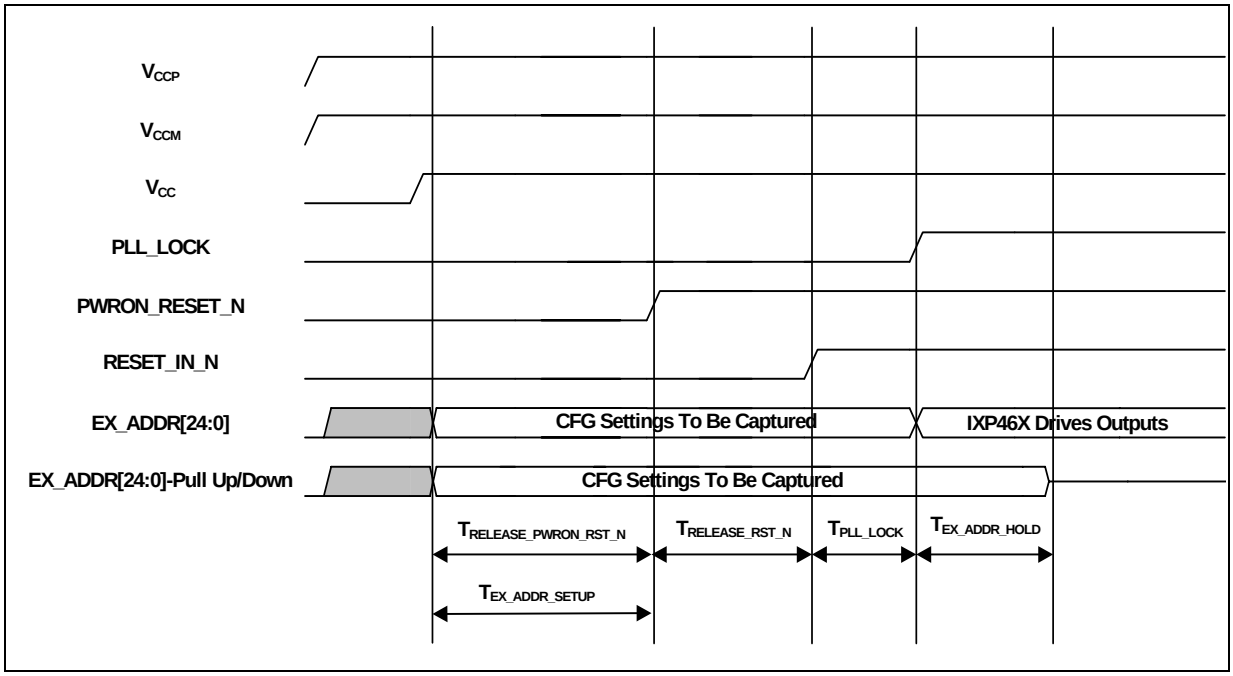


Table 81. Reset Timings Table Parameters

Symbol	Parameter	Min.	Typ.	Max.	Units	Note
$T_{RELEASE\_PWON\_RST\_N}$	Minimum time required to hold the PWON_RST_N at logic 0 state after stable power has been applied to the IXP45X/ IXP46X network processors.	500			ms	1
$T_{RELEASE\_RESET\_IN\_N}$	Minimum time required to hold the RESET_IN_N at logic 0 state after PWON_RST_N has been released to a logic 1 state. The RESET_IN_N signal must be held low when the PWON_RST_N signal is held low.	10			ns	
T_{PLL_LOCK}	Maximum time for PLL_LOCK signal to drive to logic 1 after RESET_IN_N is driven to logic 1 state. The boot sequence does not occur until this period is complete.			10	µs	
$T_{EX\_ADDR\_SETUP}$	Minimum time for the EX_ADDR signals to drive the inputs prior to RESET_IN_N being driven to logic 1 state. This is used for sampling configuration information.	50			ns	2
$T_{EX\_ADDR\_HOLD}$	Minimum/maximum time for the EX_ADDR signals to drive the inputs prior to PLL_LOCK being driven to logic 1 state. This is used for sampling configuration information.	0		20	ns	2
T_{WARM_RESET}	Minimum time required to drive RESET_IN_N signal to logic 0 in order to cause a reset after the IXP45X/IXP46X network processors have been in normal operation. The power must remain stable and the PWON_RST_N signal must remain stable.	500			ns	

NOTES:
1. $T_{RELEASE\_PWRON\_RST\_N}$ is the time required for the internal oscillator to reach stability. When an external oscillator is being used the 500-ms delay is not required.
2. The expansion bus address is captured as a derivative of the RESET_IN_N signal going high. When a programmable-logic device is used to drive the EX_ADDR signals instead of pull-downs, the signals must be active until PLL_LOCK is active.

5.7 Power Sequence

The 3.3-V I/O voltage (V_{CCP}) and the 2.5-V I/O voltage (V_{CCM}) must be powered up at least 1 µs before the core voltage (V_{CC}). The IXP45X/IXP46X network processors core voltage (V_{CC}) must never become stable prior to the 3.3-V I/O voltage (V_{CCP}) or the 2.5-V I/O voltage (V_{CCM}).

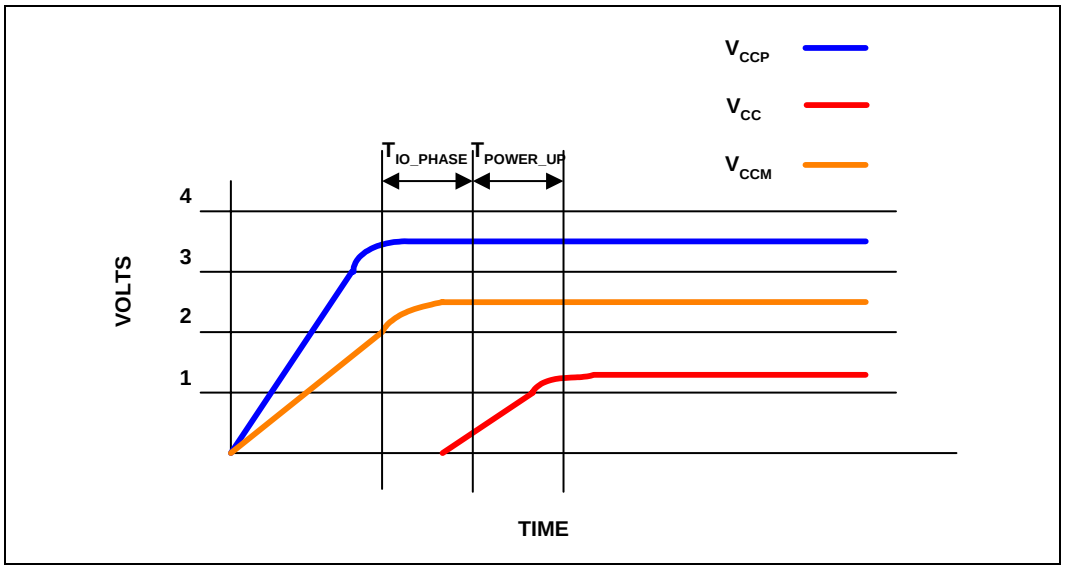
Sequencing between V_{CCP} and V_{CCM} can occur in any order with respect to one another. T_{IO_PHASE} can be:

- V_{CCP} prior to V_{CCM}
- V_{CCM} prior to V_{CCP}
- V_{CCP} simultaneously to V_{CCM}

The V_{CCOSC}, V_{CCPLL1}, V_{CCPLL2}, and V_{CCPLL3} voltages follow the V_{CC} power-up pattern. The V_{CCOSCP} follows the V_{CCP} power-up pattern.

The value for T_{POWER_UP} must be at least 1 µs before the later of V_{CCP} and V_{CCM}. The T_{POWER_UP} timing parameter is measured between the later of the I/O power rails (V_{CCP} at 3.3 V or V_{CCM} at 2.5 V) and V_{CC} at 1.3 V.

Figure 49. Power-up Sequence Timing



5.8 Power Dissipation

The Intel® IXP45X and Intel® IXP46X Product Line of Network Processors were tested assuming a typical worst case networking application under a tester environment. The following power assessments in Table 82 assume this typical worst case networking application using the interface activity factors listed in Table 83. The actual power may vary if interface activity factors are different from Table 83. If applications do not require use of certain peripherals or if interfaces operate at lower activity factors, then the power required by the part may be significantly less than the numbers stated in Table 82.

Table 82. Power Dissipation Values

Part Type	Power Rail	Icc (mA)	Power Per Rail (mW)†	Maximum Power Dissipation (Watts)
Intel® IXP45X and Intel® IXP46X Product Line of Network Processors— 266 MHz	3.3 V	88	305	2.8
	2.5 V	255	669	
	1.3 V	1335	1822	
Intel® IXP45X and Intel® IXP46X Product Line of Network Processors — 400 MHz	3.3 V	88	305	3.0
	2.5 V	255	669	
	1.3 V	1485	2027	
Intel® IXP45X and Intel® IXP46X Product Line of Network Processors — 533 MHz	3.3 V	88	305	3.2
	2.5 V	255	669	
	1.3 V	1630	2225	
Intel® IXP46X Product Line — 667 MHz	3.3 V	88	305	3.6
	2.5 V	255	669	
	1.4 V	1785	2624	
† Power in mW is calculated using Maximum Vcc specification for each power rail.				

Activity factor is directly proportional to the overall power consumption where each application will have a different activity factor and different power conclusion. Table 83 illustrates the activity factor of each interface on the tester during the typical worst case networking application.

Table 83. Power Dissipation Test Conditions

Interface	DDR (data/addr)	PCI (addr/cntl)	EXP (data/cntl)	Ethernet (data)	UTOPIA (data)
Activity Factor	15% / 6%	16% / 10%	5% / 3.7%	20%	17%

NOTES:
1. All output clocks toggling at their specified rate.
2. Tester did not include termination resistors on any interface for power analysis.
3. Tester measures power at 85 degrees F Ambient.
4. Current measurements are average and not peak.
5. Intel XScale® Core processor tested running DSP software.

5.9 Ordering Information

For ordering information, please contact your local Intel sales representative.