Architecture Tutorial
Applied Computing Conference 2000
May 23, 2000

Alan Goodrum
Chairman, PCI-X Workgroup
Staff Fellow, Compaq Computer Corporation
Purpose of this tutorial

This tutorial is

- Introduction to PCI-X
  - Key features
  - Key benefits
- Aimed primarily at digital designers
- Limited by time

This tutorial is NOT

- Detailed study of every protocol feature
- Detailed study of electrical requirements
- Detailed study of bridge requirements
- Detailed study of new configuration registers
Agenda

- Key Features
- Card and System Interoperability
- Protocol
- Software Aspects
- Electricals
- Performance
- Summary
Key PCI-X Features
I/O Bandwidth vs. Time

Bandwidth (MB/s)


Ethernet
Internet Backbone
OC 192
T3
SCSI

10 Gbit/s
1 Gbit/s
T3

Today 2000

6MHz 16-bit
66/133MHz 32/64-bit
33MHz 32/64-bit
10MHz 32-bit
66MHz 32/64-bit

10 MHz
32-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

66/133MHz
32/64-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ


Bandwidth (MB/s)

1,000
10,000
100
10
1
32/64-bit
32-bit

66/133MHz
32/64-bit

33MHz
32/64-bit

66MHz
32/64-bit

10MHz
32-bit

6MHz
16-bit

ISA

EISA

PCI

PCIX

COMPAQ
Key PCI-X Features

- PCI-X Systems
  - 32- or 64-bit,
  - 3.3 Volt I/O
  - Trade off speed for slots
    - 1 slot @ 133 MHz; 2 slots @ 100 MHz; 4 slots @ 66 MHz

- PCI-X Devices
  - 32- or 64-bit
  - 66, 133MHz
  - 3.3Volt I/O or Universal

- Bus runs in PCI-X or conventional mode
  (similar to 33/66 MHz modes in PCI 2.2)

- PCIXCAP mode pin similar to M66EN

- Integrates well with emerging switched fabric protocols like InfiniBand
Key PCI-X Features - cont’d

- Attribute phase for each transaction
  - Byte count
  - Initiator ID
  - Handling instructions
  - Tag
- More intelligent use of wait states
  - Only target initial wait states supported
- Standard block size movements (I/O Cache line)
  - Fixed transaction disconnection points on 128 byte boundaries
- Split Transactions replace Delayed Transactions
- Makes multi-threaded operation practical
- Electrical design for PCI-X easier than conv 66 MHz
Key PCI-X Features - cont’d

- Relaxed Transaction Ordering
  - Optional function for removing unnecessary blocking cases
- Support for non-cache-coherent transaction
- New configuration registers accessed via Capabilities data structure
  - No S/W initialization required
  - Default values always functional
- Improved error handling
  - Allows cards an increased range of options for handling data parity errors
Building on the Foundation of PCI 2.1 and 2.2

- Compatibility mode for 33MHz PCI 2.2 (3.3v)
  - PCI-X systems can accept standard PCI cards
  - PCI-X cards can work in current PCI systems
- Requires no Device Driver or OS modification
- New features designed for easy migration
- Similarity to 2.1/2.2 protects development infrastructure
- Required support for Message-Signaled Interrupts and PCI Power Management (D0 & D3)
- Designed to support PCI Hot-Plug
  (new Hot-Plug System Driver)
## PCI-X System Flexibility -- Speed vs. Slot Tradeoff

<table>
<thead>
<tr>
<th>Bus Width</th>
<th>Bus Width</th>
<th>Bus Bandwidth</th>
<th>PCI Slots</th>
<th>PCI-X Slots</th>
</tr>
</thead>
<tbody>
<tr>
<td>32-bit</td>
<td>33 MHz</td>
<td>133 MB/s</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>64-bit</td>
<td>66 MHz</td>
<td>533 MB/s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>64-bit</td>
<td>100 MHz</td>
<td>800 MB/s</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>64-bit</td>
<td>133 MHz</td>
<td>1066 MB/s</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

- The most common implementation of PCI today is 33 MHz per bus segment.
- PCI-X doubles the number of slots per bus segment at 66 MHz.
- PCI-X at 100 MHz provides enterprise-class I/O bandwidth.
- PCI-X at 133 MHz is the first interconnect to exceed 1 Gbyte/s.
PCI-X System Flexibility -- Hierarchical Structure

PCI-X bridges allow PCI-X to link up to 256 separate bus segments.

Smarter protocol makes high-performance PCI-X bridges practical
Card and System Interoperability
The Million-Dollar Question

What is PCI’s I/O voltage migration plan?

a. All 3.3V PCI cards must be 5V tolerant.
b. Universal slots accept both 5V and 3.3 V keyed cards.
c. PCI slots have been mostly 3.3V I/O for years.
d. Universal cards plug into both 5V and 3.3V keyed slots.

The correct answer is....
Hardware Compatibility -- PCI 5v to 3.3v I/O Migration Story

Universal PCI adapter cards are keyed for both 5v and 3.3v slots
Adapter Card Selection

- Cost sensitive 32-bit Cards
- PCI-X cards will work in current PCI systems just like 66MHz conventional cards
- Conventional Speeds & Bandwidth
  - 33MHz: 133MB/sec
  - 66MHz: 266MB/sec (optional)
- PCI-X Speeds & Bandwidth
  - 66MHz: 266MB/sec
  - 133MHz: 533MB/sec (optional)
- 3.3v or Universal

- High performance 64-bit cards
- PCI-X cards will work in current PCI systems just like 66MHz conventional cards
- Conventional Speeds & Bandwidth
  - 33MHz: 266MB/sec
  - 66MHz: 533MB/sec (optional)
- PCI-X Speeds & Bandwidth
  - 66MHz: 533MB/sec
  - 133MHz: 1066MB/sec (optional)
- 3.3v or Universal
PCI-X System Configuration

- PCI-X slots accept both PCI & PCI-X adapters
- For best performance group cards by speed and type (PCI-X or PCI) per bus segment:
  - Conventional 33MHz cards
  - Conventional 66MHz card
  - PCI-X 66 MHz cards
  - PCI-X 133 MHz card
# Interoperability Matrix

<table>
<thead>
<tr>
<th></th>
<th>conventional PCI cards</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>33 MHz (5V I/O)</td>
<td>33 MHz (3.3V I/O or Universal)</td>
<td>66 MHz (3.3V I/O or Universal)</td>
<td>66 MHz (3.3V I/O or Universal)</td>
</tr>
<tr>
<td><strong>conventional system</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33 MHz</td>
<td>33 (5V I/O)</td>
<td>33 (5V I/O)</td>
<td>33 (5V I/O)</td>
<td>33 (5V I/O)</td>
<td>33 (5V I/O)</td>
</tr>
<tr>
<td>66 MHz</td>
<td></td>
<td>33</td>
<td>66</td>
<td>a) 33(^1)</td>
<td>a) 33(^1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>b) 66</td>
<td>b) 66</td>
</tr>
<tr>
<td><strong>PCI-X system</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>66 MHz</td>
<td></td>
<td>33</td>
<td>33</td>
<td>66</td>
<td>66</td>
</tr>
<tr>
<td>100 MHz</td>
<td></td>
<td>33</td>
<td>a) 33(^1)</td>
<td>66</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>b) 66</td>
<td></td>
<td></td>
</tr>
<tr>
<td>133 MHz</td>
<td></td>
<td>33</td>
<td>a) 33(^1)</td>
<td>66</td>
<td>133</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>b) 66</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend**

- xx Conventional PCI system or expansion card operating in conventional mode.
- xx = nominal clock frequency in MHz.

- xx PCI-X system and expansion card operating in PCI-X mode.
- xx = nominal clock frequency in MHz.

- xx Most popular cases.

**Note:**
1. PCI-X system and devices must support conventional 33MHz timing, and may optionally support conventional 66MHz timing.
System Initialization & Interoperability

- **Device and Expansion Card Requirement**
  - PCI-X Expansion card identifies PCI-X capability via **PCIXCAP** pin
    - (open = PCI-X 133, pulldown = PCI-X 66, GND = conv. PCI)
  - PCI-X devices enter PCI-X mode when **RST#** deasserts with **TRDY#**, **STOP#**, or **DEVSEL#** asserted on an idle bus

- **System Requirements (M66EN & PCIXCAP)**
  - The system is required to determine the proper operating mode for the bus and to apply the appropriate PCI-X initialization pattern to the bus before the rising edge of **RST#**

- **Mode and Frequency Initialization Sequence**
  - Bus mode (Conventional PCI or PCI-X)
    - If PCI-X “133”, “100”, “66” modes
    - If conventional PCI “66”, “33” modes
Reset & Initialization Sequence

- Conventional vs. PCI-X mode selected at rising edge or RST#
- PCI-X bus freq range encoded at rising edge of RST#

![Diagram showing the reset and initialization sequence with timing notations for PCI_CLK, RST#, FRAME#, IRDY#, TRDY#, STOP#, DEVSEL#, and PCI-X_initialization pattern decode.](image)

PCI-X initialization pattern decode

- D: transparent latch
- En: PCI-X_mode_en

-t_{rlcx}, t_{rst_clk (ref)}, t_{rslt (ref)}, t_{prsu}, t_{prh}
PCI-X Protocol
PCI 2.2 vs. PCI-X Comparison

- Write transaction, same device select timing, same wait states, 6 data phases
- Conventional PCI bus requires 9 clocks, PCI-X bus requires 10 clocks
Transaction Phases

Address Phase

Attribute Phase

Target Response Phase

Turn Around

Data Transfer

Bus Transaction

Initiator
Termination

COMPAQ
Behind the Curtain
How can the bus be faster but the timing easier?

- Register-to-register design allows maximum flight time
Behind the Curtain

PCI Clock, 33 MHz
- 30 ns period
- 7 ns setup time

PCI Clock, 66 MHz
- 15 ns period
- 3 ns setup time

PCI-X registered protocol allocates a full clock period for logic decision
- @ 66MHz - 15ns
- @ 133MHz - 7.5ns
## PCI-X Terms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allowable Disconnect Boundary (ADB)</td>
<td>The initiator and target are permitted to disconnect byte-count transactions only on naturally aligned 128-byte boundaries.</td>
</tr>
<tr>
<td>Requester</td>
<td>Initiator that first introduces a transaction into the PCI-X domain.</td>
</tr>
<tr>
<td>Completer</td>
<td>The device addressed by a transaction (other than a Split Completion).</td>
</tr>
<tr>
<td>Sequence</td>
<td>One or more transactions associated with carrying out a single logical transfer by a requester.</td>
</tr>
<tr>
<td>Attributes</td>
<td>Byte count, requester or completer ID, bus number, sequence number, and other transaction handling instructions.</td>
</tr>
<tr>
<td>Split Transaction</td>
<td>A single logical transfer containing a initial transaction (the Split Request) that the target (the completer) terminates with Split Response, followed by one or more transactions (the Split Completions) initiated by the completer to send the read data (if a read) or a completion message back to the requester.</td>
</tr>
</tbody>
</table>
PCI-X Peer Transaction Flow
PCI-X Features

- AD bus specifies starting byte address (including AD[2:0])
- Byte Enable bus is reserved (driven high) for all transactions except Memory Write
- Wait states not allowed, except target initial wait states. Always pairs for memory write and Split Completion
- PCI-X “DWORD transaction” like convention PCI “single data phase” transaction (Config, I/O, Special Cycle).

**DEVSEL#** decode speed:

<table>
<thead>
<tr>
<th>Decode Speed</th>
<th>PCI-X</th>
<th>Conventional PCI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 clock after address</td>
<td>Not Supported</td>
<td>Fast</td>
</tr>
<tr>
<td>2 clocks after address</td>
<td>Decode A</td>
<td>Medium</td>
</tr>
<tr>
<td>3 clocks after address</td>
<td>Decode B</td>
<td>Slow</td>
</tr>
<tr>
<td>4 clocks after address</td>
<td>Decode C</td>
<td>SUB</td>
</tr>
<tr>
<td>6 clocks after address</td>
<td>SUB</td>
<td>N/A</td>
</tr>
</tbody>
</table>
# PCI-X Burst and DWORD Transactions

<table>
<thead>
<tr>
<th>Burst Transactions</th>
<th>DWORD Transactions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Commands:</strong></td>
<td><strong>Commands:</strong></td>
</tr>
<tr>
<td>• Memory Read Block</td>
<td>• Interrupt Acknowledge</td>
</tr>
<tr>
<td>• Memory Write Block</td>
<td>• Special Cycle</td>
</tr>
<tr>
<td>• Memory Write</td>
<td>• I/O Read</td>
</tr>
<tr>
<td>• Alias to Memory Read Block</td>
<td>• I/O Write</td>
</tr>
<tr>
<td>• Alias to Memory Write Block</td>
<td>• Configuration Read</td>
</tr>
<tr>
<td>• Split Completion</td>
<td>• Configuration Write</td>
</tr>
<tr>
<td>• Memory Read DWORD</td>
<td>• Memory Read DWORD</td>
</tr>
<tr>
<td><strong>64- or 32-bit data transfers.</strong></td>
<td><strong>32-bit data transfers only</strong></td>
</tr>
<tr>
<td>Starting address specified on AD bus down to a byte address (includes all AD bus).</td>
<td>Starting address specified on AD bus down to a byte address (includes all AD bus), except for configuration transactions, which are DWORD aligned (AD[1:0] indicate configuration transaction type).</td>
</tr>
<tr>
<td>Supports one or more data phases and always in address order.</td>
<td>Supports only single data phase.</td>
</tr>
<tr>
<td>During the data phases the C/BE# bus is reserved and driven high by the initiator for all transactions except Memory Write. The C/BE# bus contains valid byte enables for Memory Write transactions. Any byte enable pattern is permitted (between the starting and ending address, inclusive), including no byte enables asserted.</td>
<td>During the attribute phase the Requester Attributes contains valid byte enables. Any byte enable pattern is permitted, including no byte enables asserted.</td>
</tr>
<tr>
<td>During the data phase the C/BE# bus is reserved and driven high by the initiator.</td>
<td></td>
</tr>
</tbody>
</table>
## PCI-X Command Encoding

<table>
<thead>
<tr>
<th>C/BE[3:0]# or C/BE[7:4]#</th>
<th>Conventional PCI Command (reference)</th>
<th>PCI-X Command</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000b</td>
<td>Interrupt Acknowledge</td>
<td>Interrupt Acknowledge</td>
<td>DWORD</td>
</tr>
<tr>
<td>0001b</td>
<td>Special Cycles</td>
<td>Special Cycles</td>
<td>DWORD</td>
</tr>
<tr>
<td>0010b</td>
<td>I/O Read</td>
<td>I/O Read</td>
<td>DWORD</td>
</tr>
<tr>
<td>0011b</td>
<td>I/O Write</td>
<td>I/O Write</td>
<td>DWORD</td>
</tr>
<tr>
<td>0100b</td>
<td>Reserved</td>
<td>Reserved</td>
<td>na</td>
</tr>
<tr>
<td>0101b</td>
<td>Reserved</td>
<td>Reserved</td>
<td>na</td>
</tr>
<tr>
<td>0110b</td>
<td>Memory Read</td>
<td>Memory Read DWORD</td>
<td>DWORD</td>
</tr>
<tr>
<td>0111b</td>
<td>Memory Write</td>
<td>Memory Write</td>
<td>Burst</td>
</tr>
<tr>
<td>1000b</td>
<td>Reserved</td>
<td>Alias to Memory Read Block</td>
<td>Burst</td>
</tr>
<tr>
<td>1001b</td>
<td>Reserved</td>
<td>Alias to Memory Write Block</td>
<td>Burst</td>
</tr>
<tr>
<td>1010b</td>
<td>Configuration Read</td>
<td>Configuration Read</td>
<td>DWORD</td>
</tr>
<tr>
<td>1011b</td>
<td>Configuration Write</td>
<td>Configuration Write</td>
<td>DWORD</td>
</tr>
<tr>
<td>1100b</td>
<td>Memory Read Multiple</td>
<td>Split Completion</td>
<td>Burst</td>
</tr>
<tr>
<td>1101b</td>
<td>Dual Address Cycle</td>
<td>Dual Address Cycle</td>
<td>na</td>
</tr>
<tr>
<td>1110b</td>
<td>Memory Read Line</td>
<td>Memory Read Block</td>
<td>Burst</td>
</tr>
<tr>
<td>1111b</td>
<td>Memory Write and Invalidate</td>
<td>Memory Write Block</td>
<td>Burst</td>
</tr>
</tbody>
</table>
DWORD Transactions

DWORD Write Transaction with No Wait States.

Notice the initiator continues driving the bus, and IRDY# remains asserted in clock 7, even though this is one clock past the single clock in which data was transferred (clock 6). In PCI-X the initiator requires two clocks to respond to the assertion of TRDY#.

DWORD Read Transaction with two Target initial Wait States

Notice BE# bus reserved and driven high, because byte enables are in the requester attributes.
PCI-X Protocol--
Attributes and Split Transactions
# Transaction Attributes

## Requester Attributes for Burst Transactions (most memory read/write)

<table>
<thead>
<tr>
<th>35</th>
<th>32</th>
<th>31</th>
<th>30</th>
<th>29</th>
<th>28</th>
<th>24</th>
<th>23</th>
<th>16</th>
<th>15</th>
<th>11</th>
<th>10</th>
<th>08</th>
<th>07</th>
<th>00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upper Byte Count</td>
<td>R</td>
<td>N</td>
<td>S</td>
<td>RO</td>
<td>Tag</td>
<td>Requester Bus Number</td>
<td>Requester Device Number</td>
<td>Requester Function Number</td>
<td>Lower Byte Count</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C/BE[3-0]#</td>
<td>AD[31:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Requester Attributes for DWORD Transactions
(Memory Read DWORD, I/O read/write, Special Cycle, Int Ack)

<table>
<thead>
<tr>
<th>35</th>
<th>32</th>
<th>31</th>
<th>30</th>
<th>29</th>
<th>28</th>
<th>24</th>
<th>23</th>
<th>16</th>
<th>15</th>
<th>11</th>
<th>10</th>
<th>08</th>
<th>07</th>
<th>00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte Enables</td>
<td>R</td>
<td>N</td>
<td>S</td>
<td>RO</td>
<td>Tag</td>
<td>Requester Bus Number</td>
<td>Requester Device Number</td>
<td>Requester Function Number</td>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C/BE[3-0]#</td>
<td>AD[31:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

RO -- Relax ordering
NS -- No Snoop
# Transaction Attributes

## Split Completion Address

<table>
<thead>
<tr>
<th>BUS CMD</th>
<th>R</th>
<th>R</th>
<th>RO</th>
<th>Tag</th>
<th>Requester Bus Number</th>
<th>Requester Device Number</th>
<th>Requester Function Number</th>
<th>R</th>
<th>Lower Address [6:0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>C/BE[3:0]#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AD[31:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

RO -- Relax ordering,

## Completer Attributes

<table>
<thead>
<tr>
<th>Upper Byte Count</th>
<th>B</th>
<th>C</th>
<th>M</th>
<th>S</th>
<th>C</th>
<th>M</th>
<th>R</th>
<th>Completer Bus Number</th>
<th>Completer Device Number</th>
<th>Completer Function Number</th>
<th>Lower Byte Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>C/BE[3:0]#</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AD[31:0]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SCM -- Split Completion Message  
SCE -- Split Completion Error  
BCM -- Byte Count Modified
Split Transactions

PCI-X Bus

Requester A

Initiator

Target

Split Transaction Requester

Completer B

Target

Split Transaction Completer

Initiator

 REQ GNT

Requester Attributes
Address, Memory Read

Split Response

(Requester's Attribute)

Split Completion
Immediate Response

Completer Attributes
DATA

Immediate Response

34
PCI-X Protocol--
Configuration Transactions
Config Address & Attributes

PCI 2.2 Type 1 to Type 0 Configuration Address (ref)

<table>
<thead>
<tr>
<th>Type 1</th>
<th>Type 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUS CMD</td>
<td>BUS CMD</td>
</tr>
<tr>
<td>C/BE[3-0]#</td>
<td>C/BE[3-0]#</td>
</tr>
<tr>
<td>BUS Number</td>
<td>Bus Number</td>
</tr>
<tr>
<td>Dev Number</td>
<td>Dev Number</td>
</tr>
<tr>
<td>Func Number</td>
<td>Func Number</td>
</tr>
<tr>
<td>Register Number</td>
<td>Register Number</td>
</tr>
<tr>
<td>0 1</td>
<td>0 0</td>
</tr>
</tbody>
</table>

PCI-X Config Type 1 to Type 0 Configuration Address

<table>
<thead>
<tr>
<th>Type 1</th>
<th>Type 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUS CMD</td>
<td>BUS CMD</td>
</tr>
<tr>
<td>C/BE[3-0]#</td>
<td>C/BE[3-0]#</td>
</tr>
<tr>
<td>BUS Number</td>
<td>Bus Number</td>
</tr>
<tr>
<td>Dev Number</td>
<td>Dev Number</td>
</tr>
<tr>
<td>Func Number</td>
<td>Func Number</td>
</tr>
<tr>
<td>Register Number</td>
<td>Register Number</td>
</tr>
<tr>
<td>0 1</td>
<td>0 0</td>
</tr>
</tbody>
</table>

Configuration Attributes

<table>
<thead>
<tr>
<th>Byte Enables</th>
<th>R</th>
<th>R</th>
<th>R</th>
<th>Tag</th>
<th>Requester Bus Number</th>
<th>Requester Device Number</th>
<th>Requester Function Number</th>
<th>Secondary Bus Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>C/BE[3::0]#</td>
<td>AD[31:0]</td>
<td>AD[31:0]</td>
<td>AD[31:0]</td>
<td>AD[31::0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Configuration Transactions

4 clocks of valid address before asserting FRAME#
PCI-X Protocol--
PCI-X Bridges
PCI-X Bridge Design
“This is not your father’s bridge”

- **Split Transactions make consistently high-performance bridges possible**
  - No speculative prefetch penalty
  - Forwards read completion data similar to posted memory writes
- Converts between PCI-X and conventional PCI, if necessary
- Frequency and mode of secondary bus independent of primary bus
- Transaction ordering rules similar to conventional PCI
- Performance-tuning registers to enable adjustment of request rate to match completion rate
PCI-X Bridge Transaction Flow

Start Here

Requester (e.g. Adapter Card)

Sequence Requestor
I/O
CFG
Int. Ack.
MemRead

Error Termination (Sequence Ends)

Split Completion
Completion Data
Read
SCM
Write Ack
Error Status

Immediate Response
Completion Data

Requestor Interface

Split Completion
Initiator I/O
CFG
Int. Ack.
MemRead

Error Response
Transaction Rescheduled

Split Response
I/O
CFG
Int. Ack.
MemRead

Immediate Response
Posted Memory Write

Forwarded Completion

Split Completion
Exception Message Initiator

Immediate Response Completion Data Accepted

Error Response
Transaction Rescheduled

Forwarded Completion

Split Completion
Initiator I/O
CFG
Int. Ack.
MemRead

Immediate Response
Completion Data

Requestor Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

Split Response
I/O
CFG
Int. Ack.
MemRead

Immediate Response
Posted Memory Write

Forwarded Sequence
I/O
CFG
Int. Ack.
MemRead

Forwarded Sequence
Posted Memory Write

Error Response
Transaction Rescheduled

Forwarded Completion

Requestor Interface

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

PCI-X Bridge Transaction Flow

Target Interface

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

Initiator Interface

Target Interface

Primary Bus

Secondary Bus

Initiator Interface

Target Interface

Initiator Interface

Target Interface
PCI-X Bridge Design--Summary

- Great system performance
- More complex than conventional PCI bridges
- See the PCI-X spec for full details
PCI-X Protocol--
Parity Generation and Checking
Burst Write Parity Operation

Burst Write or Split Completion Transaction Parity Operation
Burst Read Parity Operation

Burst Read Transaction (Immediate Completion) Parity Operation
DWORD Read Parity Operation with Decode A and No Initial Wait States
DWORD Read Parity Operation

DWORD Read Parity Operation with Decode B and No Initial Wait States
PCI-X Protocol--
Exception Handling

- All PCI-X devices must provide one of the following levels of support for data parity error recovery:
  - Notify device driver of problem. Driver attempts recover or resets device or system.
  - Assert SERR#.
- Split Transaction Exception rules
  - Parity errors on Split Completion
  - Master-Abort and Target-Abort messages
- Status registers in configuration space can only be cleared by the system-specific software after logging the exceptions
PCI-X Protocol--
Arbitration Rules

- Relevant clock for **GNT#** one clock earlier than conventional PCI (registered bus)
  - In general, **GNT#** asserted two clocks prior to start of transaction
  - Initiator permitted to start transaction one clock after **GNT#** deasserted
  - If the arbiter deasserts **GNT#** to one device, it cannot assert **GNT#** to another device until the next clock.

- Fair opportunities for all devices to execute configuration transactions

- In PCI Hot-Plug systems arbiter must coordinate with the Hot-Plug Controller

- The default Latency Timer value for initiators in PCI-X mode is 31
PCI Hot Plug Support

- **Hardware Impact**
  - Hot-Plug Controller provides to the Hot-Plug System Driver the means to check the `PCIXCAP` pin to identify PCI-X adapters
  - The Hot-Plug Controller drives the PCI-X initialization pattern on the bus with the proper timing prior to the rising edge of `RST#` for that slot
  - The Hot-Plug Controller coordinates with the arbiter for bus ownership during hot insertion
  - PCI-X devices ignore `TRDY#`, `STOP#` and `DEVSEL#` on an idle bus

- **Software Impact**
  - The Hot-Plug System Driver must read the inserted card’s `M66EN` and `PCIXCAP` pin to ensure that inserted adapter supports bus frequency and operating mode of the bus
Software Aspects of PCI-X
Software Compatibility

- No OS or driver change required
  - New config registers default to functional values
  - Optional performance tuning registers
  - Other config registers unchanged
  - No device programming model changes required
- Optional improved error handling
  - Enables smart device and new driver to recover from PERR# event
- Updated Hot-Plug System Driver
Configuration Space

- PCI-X devices use the standard PCI config header
- New PCI-X registers use the Capability List
- The PCI-X list item includes
  - An 8-bit PCI-X Capability ID (standard reg)
  - An 8-bit pointer to next list item (standard reg)
  - A 32-bit PCI-X Command—controls various modes and features of the PCI-X device
  - A 24-bit PCI-X Status—identifies the capabilities and current operating mode of the device
- PCI-X bridges have different registers
Software Summary--“It Just Works”

Up to 8 times I/O throughput improvement without changing your OS
PCI-X Electrical Design
A Word to the Wise

- 33 MHz PCI spec was forgiving
- High-freq designs are more complex
  - PCI 66, PCI-X 66, PCI-X 133
  - Use high-freq design techniques and principles
    - Doc # tc000301tb
  - Simulate system topology
  - Must follow PCI-X spec precisely
- If you don’t understand *every word* of Chap 9, find someone who does.
Signal Quality Guidelines for PCI-X

- Report of the PCI-X Electrical Subgroup to be available mid year 2000
- Includes discussion of:
  - I/O Buffer Design
  - Overshoot
  - Ringback
  - Settling time
  - Inter Symbol Interference
  - Secondary Effects
    - Ground Bounce
    - Cross Talk
    - DC Offset
  - Summary results of almost 2 million SPICE simulations
## Timing Budget

### Setup Time Budget

<table>
<thead>
<tr>
<th>Parameter</th>
<th>133 MHz PCI-X</th>
<th>100 MHz PCI-X</th>
<th>66 MHz PCI-X</th>
<th>66 MHz Conventional PCI (ref)</th>
<th>33 MHz Conventional PCI (ref)</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{val}$ (max)</td>
<td>3.8</td>
<td>3.8</td>
<td>3.8</td>
<td>6</td>
<td>11</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{prop}$ (max)</td>
<td>2.0</td>
<td>4.5</td>
<td>9.0</td>
<td>5</td>
<td>10</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{skew}$ (max)</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{su}$ (max)</td>
<td>1.2</td>
<td>1.2</td>
<td>1.7</td>
<td>3</td>
<td>7</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{cyc}$</td>
<td>7.5</td>
<td>10</td>
<td>15</td>
<td>15</td>
<td>30</td>
<td>ns</td>
</tr>
</tbody>
</table>

### Hold Time Budget

<table>
<thead>
<tr>
<th>Parameter</th>
<th>PCI-X</th>
<th>66 MHz Conventional PCI (ref)</th>
<th>33 MHz Conventional PCI (ref)</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{val}$ (min)</td>
<td>0.7</td>
<td>2</td>
<td>2</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{prop}$ (min)</td>
<td>0.3</td>
<td>0</td>
<td>0</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{skew}$ (max)</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>ns</td>
</tr>
<tr>
<td>$T_{h}$ (min)</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>ns</td>
</tr>
</tbody>
</table>
PCI-X V/I Curves vs. PCI 2.2 V/I Curves

PCI-X Pull-Up Output Buffer V/I Curves

PCI-X Pull-Down Output Buffer V/I Curves
PCI-X Mechanical Requirements
PCI-X Mechanical Requirements

- Same card and slot mechanical requirements as conventional PCI (3.3V I/O)
- New labeling requirement (ECR in progress)
  - Systems must identify slot capability (method not specified)
  - Cards must be marked

![Card Dimensions Diagram]

**Add-in Card**
- **Standard Length**: Add-in Card
- **Short Length**: Add-in Card (Variable Height)
- **Low Profile**: Add-in Card

![Add-in Card Dimensions Diagram]
PCI-X Performance
PCI-X Demonstration Using Compaq ProLiant 8500 Server

- Cable-less, tool-less design
- Serviceable in minutes
- Reduce training & downtime
- Improved product availability

Key Customer Benefit - Easy configuration, installation, upgrade, and service.
Compaq & Intel Industry Standard 8-Way Architecture

2 Way Interleaved SDRAM Memory

Even Memory Port

Odd Memory Port

Bus 1 Cache
Coherency
Accelerator

Bus 2 Cache
Coherency
Accelerator

100 MHz AGTL+ bus 1

100 MHz AGTL+ bus 2

ProFusion

PCI-X vs. conventional PCI protocol comparison

With only I/O subsystem upgrade

COMPAQ
PCI-X Advanced Protocol Gives You More Usable Bandwidth at Any Speed

With “real hardware” PCI-X is over 60% faster than conventional PCI (your mileage may vary)

PCI 64bit/33MHz
140MByte/s

2 PCI-X adapters pulling data
230MByte/s

6 conventional PCI adapters pulling data

COMPAQ
PCI-X vs. PCI: 4K-Byte Read Performance

66MHz PCI-X up to 33% faster than 66MHz PCI 2.2

Note: Assumes Ideal memory controller with 32-byte CPU cache line and ideal 64-bit PCI adapters
Prognostications for PCI-X

- Rapid deployment in Server market beginning 2H2000
  - Server vendors on Workgroup: Compaq, Dell, HP, IBM, Intel
  - Peripheral vendors on Workgroup
    - Disk: Adaptec, Mylex (IBM), LSI
    - NIC: 3Com, Intel
  - Si and IP vendors on Workgroup: InSilicon (Phoenix), Intel, LSI, ServerWorks

- Migration into workstation, desktop, and embedded markets
  - Replaces conv PCI as devices move to 3.3V-only in next 2 years

- Continue to dominate local I/O applications even after InfiniBand begins solving new problems in distributed computing applications in 2001 and 2002

- Extend the life of PCI 5-10 years
Summary

PCI-X compatibility

- Fully interoperable with conventional PCI
- Easy design migration from conventional PCI
  - Your conv PCI experience prepares you for PCI-X
- Easier electrical design than 66 MHz conventional PCI
- No OS or driver changes required (only Hot-Plug System Driver)

PCI-X performance

- Over 1 Gbyte/s
- Byte counts, Split Transactions make whole system work smarter
- PCI-X adapter cards are good citizens: No more bus hogs
- Efficient P2P bridges for hierarchy of buses
- Increases system flexibility: 4-slots per bus at 66MHz
- Integrates well with distributed I/O standards like InfiniBand
Take away

PCI-X is THE logical next step for your I/O designs

Hardware and Software Compatibility
Better Performance
Need I say more?
For More Information

* PCI SIG  [www.pcisig.com](http://www.pcisig.com)
  - PCI-X 1.0 specification
  - PCI-X compliance checklist
    [http://www.pcisig.com/tech/docs.html](http://www.pcisig.com/tech/docs.html)
  - PCI-X spec errata
    [http://www.pcisig.com/tech/ecn_ecr.html](http://www.pcisig.com/tech/ecn_ecr.html)
  - Technical support
    techsupp@pcisig.com
  - General information
    pci-x@pcisig.com
For More Information

Compaq

- PCI-X enablement site (sample core, docs, Golden Master Program, links to other vendors):
  www.compaq.com/PCI-X

- Information
  PCI-X@compaq.com

- Electrical Design Considerations for PCI-X and 66-MHz PCI Cards,
  http://www.compaq.com/support/techpubs/whitepapers/WhitePapers_Industry_Technology.html,
  Doc # tc000301tb
Backup
# PCI-X Comparison

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>PCI</th>
<th>AGP 1.0</th>
<th>AGP 2.0</th>
<th>PCI-X</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCI slot compatibility</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>100 MHz Bus speed</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>133 MHz Bus speed</td>
<td>No</td>
<td>66 MHz DDR</td>
<td>66 MHz DDR</td>
<td>Yes</td>
</tr>
<tr>
<td>266 MHz Bus speed</td>
<td>No</td>
<td>No</td>
<td>66 MHz QDR *</td>
<td>No</td>
</tr>
<tr>
<td>Data Bus Width</td>
<td>32/64</td>
<td>32</td>
<td>32</td>
<td>32/64</td>
</tr>
<tr>
<td>Address Bus Width</td>
<td>32/64</td>
<td>32/36/64</td>
<td>32/47/64</td>
<td>64</td>
</tr>
<tr>
<td>Max Bus Bandwidth(MB/sec)</td>
<td>533</td>
<td>533</td>
<td>1064</td>
<td>1064</td>
</tr>
<tr>
<td>Multiple slots</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Hierarchical bus topology</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Split Transactions</td>
<td>Open</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Transaction Byte count</td>
<td>No</td>
<td>32 (256 **)</td>
<td>64 (256**)</td>
<td>4K</td>
</tr>
<tr>
<td>Non-coherent Transactions</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>No/Relax Ordering Rules</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Device &amp; Bus # ID ***</td>
<td>No</td>
<td>No, N/A</td>
<td>No, N/A</td>
<td>Yes</td>
</tr>
</tbody>
</table>

* Quad data rate (4X)

** Special long read command

*** Allows system to be tuned more precisely for optimal performance
Background--
The need for a faster PCI bus
Changing Systems Architectures

Faster CPU requires more I/O bandwidth

Emergence of distributed systems architectures, SAN and STAN

Separating CPU-plexes and I/O by fast thin bus segments

Current PCI Based Systems Need a Refresh
I/O Bandwidth vs. Time

- **1986**: ISA 6MHz 16-bit
- **1988**: EISA 10MHz 32-bit
- **1990**: 33MHz 32/64-bit
- **1992**: 66MHz 32/64-bit
- **1994**: OC 192
- **1996**: T3
- **1998**: 1 Gbit/s
- **2000**: 10 Gbit/s
- **2002**: Internet Backbone
PCI-X Objectives

- Address customer requirements for greater I/O performance
- Evolutionary I/O upgrade
  - Investment protection
  - Leverage PCI prevalence—“The most successful interconnect”
  - Maintain compatibility with installed base
- More Slots @ 66MHz
- Ease of design
- Strong industry support
PCI-X Burst Write with Target Wait States
PCI-X Burst Read with Target Wait States

Initiator's View of the PCI Bus

Target's View of the PCI Bus
Split Transactions

- Bus efficiency of Read almost as good as Write
- Split Transaction components
  - Step 1. Requester requests bus and arbiter grants bus
  - Step 2. Requester initiates transaction
  - Step 3. Targeted (completer) communicates intent with new target termination, Split Response
  - Step 4. Completer executes transaction internally
  - Step 5. Completer requests bus and arbiter grants bus
  - Step 6. Completer initiates Split Completion
- Split Completion routed back to requester across bridges using requester’s bus number and device number
Logic Block Diagram for Bypassing Source Sampling

- **PCI LOGIC**
- **Logic Gates**
- **I/O Buffer**
- **IOB1**

* -- All flip-flops are assumed rising edge triggered
Device Internal Timing Example

* -- All Flip Flops are assumed to be rising edge triggered
PCI-X vs PCI: 8x512-Byte Read Simulation

66MHz PCI-X up to 19% faster than 66MHz PCI 2.2

Note: Assumes ideal memory controller with 32-byte CPU cache line and ideal 64-bit PCI adapters.