PCI Express® Basics

Joe Winkles
Senior Staff Engineer
MindShare, Inc.
PCI Express Introduction

- PCI Express architecture is a high performance, IO interconnect for peripherals in computing/communication platforms
- Evolved from PCI and PCI-X® architectures
  - Yet PCI Express architecture is significantly different from its predecessors PCI and PCI-X
- PCI Express is a serial point-to-point interconnect between two devices
- Implements packet based protocol for information transfer
- Scalable performance based on number of signal Lanes implemented on the PCI Express interconnect
PCI Express Terminology

PCI Express Device A

PCI Express Device B

Signal

Wire

Link

Lane
**Assumes 2.5 Gbits/s signaling in each direction**
- 80% BW utilized due to 8b/10b encoding overhead
- Aggregate bandwidth implies simultaneous traffic in both directions
- Peak bandwidth is higher than any bus available

**PCle® 2.0 will support PHYs offering 5 Gbits/s signaling - doubling above bandwidth numbers**

<table>
<thead>
<tr>
<th>Link Width</th>
<th>x1</th>
<th>x2</th>
<th>x4</th>
<th>x8</th>
<th>x12</th>
<th>x16</th>
<th>x32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregate BW (GBytes/s)</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>16</td>
</tr>
</tbody>
</table>
PCI Express Features

- Point-to-point connection
- Serial bus means fewer pins
- Scaleable: x1, x2, x4, x8, x12, x16, x32
- Dual Simplex connection
- 2.5Gbits/s transfer/direction/s
- Packet based transaction protocol
Differential Signaling

- Electrical characteristics of PCI Express signal
  - Differential signaling
    - Transmitter Differential Peak voltage = 0.4 - 0.6 V
    - Transmitter Common mode voltage = 0 - 3.6 V

- Two devices at opposite ends of a Link may support different DC common mode voltages
Example PCI Express Topology

Legend
- PCI Express Device Downstream Port
- PCI Express Device Upstream Port
Example PCI Express Topology – Root & Switch
Example Low Cost PCI Express System

Processor

Root Complex

PCI Express

GFX

Serial ATA

USB 2.0

LPC

HDD

IO Controller Hub (ICH)

GB Ethernet

Add-In

Add-In

Add-In

PCI Express

Link

DDR SDRAM

IEEE 1394

PCI

Slots

SIO

COM1

COM2

PCI Express

Add-In

Add-In

Add-In

Among other things, this diagram shows how different components of a computer system, such as the processor, root complex, and various I/O controllers, are connected through PCI Express links. The diagram also highlights the integration of various peripherals including hard drives, USB ports, and Ethernet connectivity.
Example PCI Express Server System
Request are translated to one of four transaction types by the Transaction Layer:

1. **Memory Read or Memory Write.** Used to transfer data from or to a memory mapped location
   - The protocol also supports a *locked memory read* transaction variant.

2. **I/O Read or I/O Write.** Used to transfer data from or to an I/O location
   - These transactions are restricted to supporting legacy endpoint devices.

3. **Configuration Read or Configuration Write.** Used to discover device capabilities, program features, and check status in the 4KB PCI Express configuration space.

4. **Messages.** Handled like posted writes. Used for event signaling and general purpose messaging.
## PCI Express TLP Types

<table>
<thead>
<tr>
<th>Description</th>
<th>Abbreviated Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Read Request</td>
<td>MRd</td>
</tr>
<tr>
<td>Memory Read Request – Locked Access</td>
<td>MRdLk</td>
</tr>
<tr>
<td>Memory Write Request</td>
<td>MWr</td>
</tr>
<tr>
<td>IO Read Request</td>
<td>IORd</td>
</tr>
<tr>
<td>IO Write Request</td>
<td>IOWr</td>
</tr>
<tr>
<td>Configuration Read Request Type 0 and Type 1</td>
<td>CfgRd0, CfgRd1</td>
</tr>
<tr>
<td>Configuration Write Request Type 0 and Type 1</td>
<td>CfgWr0, CfgWr1</td>
</tr>
<tr>
<td>Message Request without Data Payload</td>
<td>Msg</td>
</tr>
<tr>
<td>Message Request with Data Payload</td>
<td>MsgD</td>
</tr>
<tr>
<td>Completion without Data (used for IO, configuration write completions and read completion with error completion status)</td>
<td>Cpl</td>
</tr>
<tr>
<td>Completion with Data (used for memory, IO and configuration read completions)</td>
<td>CplID</td>
</tr>
<tr>
<td>Completion for Locked Memory Read without Data (used for error status)</td>
<td>CplLk</td>
</tr>
<tr>
<td>Completion for Locked Memory Read with Data</td>
<td>CplDLk</td>
</tr>
</tbody>
</table>
Each request or completion header is tagged as to its type, and each of the packet types is routed based on one of three schemes:
- Address Routing
- ID Routing
- Implicit Routing

Memory and IO requests use address routing.
Completions and Configuration cycles use ID routing.
Message requests have selectable routing based on a 3-bit code in the message routing sub-field of the header type field.
Programmed I/O Transaction

**Requester:**
- Step 1: Root Complex (requester) initiates Memory Read Request (MRd)
- Step 4: Root Complex receives CplD

**Completer:**
- Step 2: Endpoint (completer) receives MRd
- Step 3: Endpoint returns Completion with data (CplD)
**Completer:**
- Step 2: Root Complex (completer) receives MRd
- Step 3: Root Complex returns Completion with data (CplD)

**Requester:**
- Step 1: Endpoint (requester) initiates Memory Read Request (MRd)
- Step 4: Endpoint receives CplD
Peer-to-Peer Transaction

Requester:
- Step 1: Endpoint (requester) initiates Memory Read Request (MRd)
- Step 4: Endpoint receives CplD

Completer:
- Step 2: Endpoint (completer) receives MRd
- Step 3: Endpoint returns Completion with data (CplD)
TLP Origin and Destination

PCI Express Device A

- Device Core
- PCI Express Core
- Logic Interface
- Transaction Layer
- Data Link Layer
- Physical Layer

PCI Express Device B

- Device Core
- PCI Express Core
- Logic Interface
- Transaction Layer
- Data Link Layer
- Physical Layer

Link

TLP Transmitted

TLP Received

TLP Origin and Destination
Information in core section of TLP comes from Software Layer / Device Core

Bit transmit direction

Created by Transaction Layer

Appended by Data Link Layer

Appended by Physical Layer
DLLP Origin and Destination
**DLLP Structure**

- ACK / NAK Packets
- Flow Control Packets
- Power Management Packets
- Vendor Defined Packets
Ordered-Set Origin and Destination

PCI Express Device A

- Device Core
- PCI Express Core
- Logic Interface
- Transaction Layer
- Data Link Layer
- Physical Layer

PCI Express Device B

- Device Core
- PCI Express Core
- Logic Interface
- Transaction Layer
- Data Link Layer
- Physical Layer

Ordered-Set Origin and Destination

Ordered-Set Transmitted

Link

Ordered-Set Received
Ordered-Set Structure

- Training Sequence One (TS1)
  - 16 character set: 1 COM, 15 TS1 data characters
- Training Sequence Two (TS2)
  - 16 character set: 1 COM, 15 TS2 data characters
- SKIP
  - 4 character set: 1 COM followed by 3 SKP identifiers
- Electrical Idle (IDLE)
  - 4 characters: 1 COM followed by 3 IDL identifiers
- Fast Training Sequence (FTS)
  - 4 characters: 1 COM followed by 3 FTS identifiers
Quality of Service (QoS) policy through Virtual Channel and Traffic Class tags

<table>
<thead>
<tr>
<th>TC[7:0]</th>
<th>VC0</th>
<th>vc0</th>
<th>TC[7:3]</th>
<th>VC1</th>
<th>VC1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arbitration</td>
<td></td>
<td></td>
<td>Transaction</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Port Arbitration and VC Arbitration

TC[2:0] to VC0
TC[7:3] to VC1

VC0
VC1

VC0
VC1

Port Arbitration and VC Arbitration

TC[2:0] to VC0
TC[7:3] to VC1
Credit-based *flow control* is point-to-point based, not end-to-end.

Receiver sends Flow Control Packets (FCP) which are a type of DLLP (Data Link Layer Packet) to provide the transmitter with credits so that it can transmit packets to the receiver.
ACK/NAK Protocol Overview

Transmit Device A
From Transaction Layer
Rx
Tx

Data Link Layer
Replay Buffer
Mux
De-mux
DLLP
TLP
Sequence
LCRC

Receiver Device B
To Transaction Layer
Rx
Tx

Data Link Layer
Error Check
De-mux
Mux
DLLP
TLP
Sequence
LCRC

Link

TLP
Sequence
TLP
LCRC

Protocol Overview
ACK/NAK Protocol: Point-to-Point

ACK returned for good reception of Request or Completion
NAK returned for error reception of Request or Completion
PCI Express supports three interrupt reporting mechanisms:

1. **Message Signaled Interrupts (MSI)**
   - Legacy endpoints are required to support MSI (or MSI-X) with 32- or 64-bit MSI capability register implementation
   - Native PCI Express endpoints are required to support MSI with 64-bit MSI capability register implementation

2. **Message Signaled Interrupts - X (MSI-X)**
   - Legacy and native endpoints are required to support MSI-X (or MSI) and implement the associated MSI-X capability register

3. **INTx Emulation.**
   - Native and Legacy endpoints are required to support Legacy INTx Emulation
   - PCI Express defines in-band messages which emulate the four physical interrupt signals (INTA-INTD) routed between PCI devices and the system interrupt controller
   - Forwarding support required by switches
PCI Express
Error Handling

- All PCI Express devices are required to support some combination of:
  - Existing software written for generic PCI error handling, and which takes advantage of the fact that PCI Express has mapped many of its error conditions to existing PCI error handling mechanisms.
  - Additional PCI Express-specific reporting mechanisms
- Errors are classified as **correctable** and **uncorrectable**.
- **Uncorrectable** errors are further divided into:
  - Fatal uncorrectable errors
  - Non-fatal uncorrectable errors.
Errors classified as correctable, degrade system performance, but recovery can occur with no loss of information

- Hardware is responsible for recovery from a correctable error and no software intervention is required.

- Even though hardware handles the correction, logging the frequency of correctable errors may be useful if software is monitoring link operations.

- An example of a correctable error is the detection of a link CRC (LCRC) error when a TLP is sent, resulting in a Data Link Layer retry event.
Errors classified as uncorrectable impair the functionality of the interface and there is no specification mechanism to correct these errors.

The two subgroups are fatal and non-fatal.

1. **Fatal Uncorrectable Errors**: Errors which render the link unreliable
   - First-level strategy for recovery may involve a link reset by the system
   - Handling of fatal errors is platform-specific

2. **Non-Fatal Uncorrectable Errors**: Uncorrectable errors associated with a particular transaction, while the link itself is reliable
   - Software may limit recovery strategy to the device(s) involved
   - Transactions between other devices are not affected
Baseline Error Reporting

- Enabling/disabling error reporting
- Providing error status
- Providing error status for Link Training
- Initiating Link Re-training

- Registers provide control and status for
  - Correctable errors
  - Non-fatal uncorrectable errors
  - Fatal uncorrectable errors
  - Unsupported request errors
Advanced Error Reporting

- Finer granularity in defining error type
- Ability to define severity of uncorrectable errors
  - Either send ERR_FATAL or ERR_NONFATAL message for a given error
- Support for error logging of error type and TLP header related to error
- Ability to mask reporting of errors
- Enable/disable root reporting of errors
- Identify source of errors
PCI Express Configuration Space

PCI-compliant register area can be accessed in PCI manner or via PCI Express Enhanced Configuration Mechanism.

PCI Express Capability Structure must be implemented within this area.

Optional PCI Express Extended Capability register sets are implemented within this area:
- Advanced Error Reporting Capability
- Virtual Channel Capability
- Device Serial Number Capability
- Power Budgeting Capability.

First extended capability register set must be located at offset 100h.
Thank you for attending the PCI-SIG Developers Conference 2006.

For more information please go to www.pcisig.com
PCI Express Basics

Joe Winkles
Senior Staff Engineer
MindShare, Inc.