PCI Express® 1.1 Link, Transaction and Configuration Protocols

Milpitas, CA Dec 5, 2005

Mike Jackson

Sr. Staff Engineer, MindShare, Inc.

Agenda

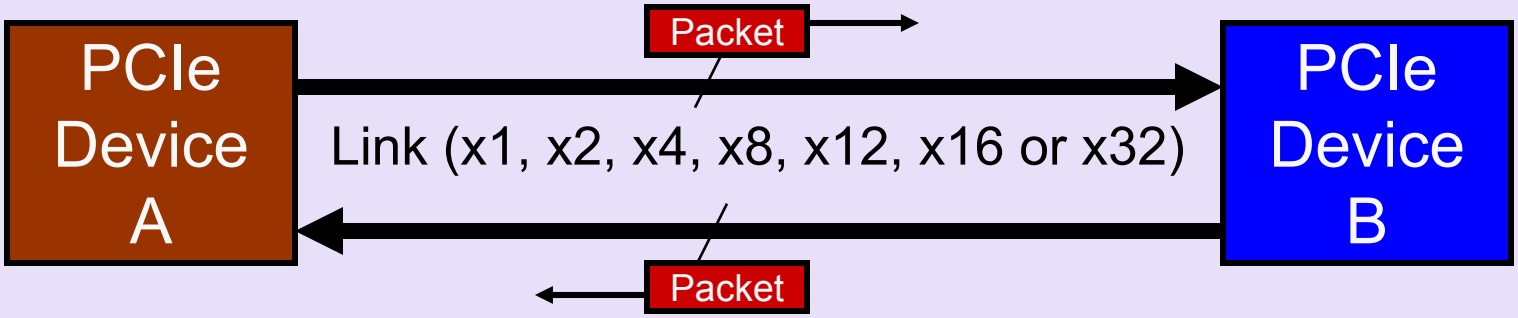
- PCIe™ Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

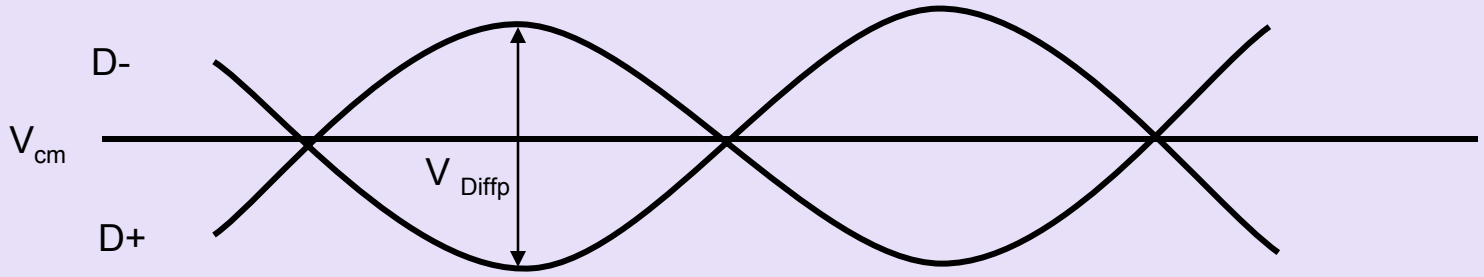
PCI Express Way - Features

- Point-to-point connection
- Serial bus means fewer pins
- Scaleable: x1, x2, x4, x8, x12, x16, x32
- Gen 1 2.5Gbits/s transfer/direction/s
- Gen 1 Bandwidth: 0.5, 1, 2, 4, 6, 8, 16 GByte/s respectively
- Dual Simplex connection
- Packet based transaction protocol



Additional Features

- 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

Additional Features

- Switches connect multiple devices
- Packet-based protocol
- Bandwidth and clocking
- Same memory, IO and configuration address space as PCI
  - ✓ Similar transaction types as PCI with additional message transaction
- PCI Express Transactions include:
  - ✓ memory read/write, memory read lock, IO read/write, configuration read/write, message requests
- Split transaction model for non-posted

Additional Features

- Data Integrity and Error Handling
  - ✓ RAS capable (Reliable, Available, Serviceable)
  - ✓ Data integrity at: 1) Link level, 2) end-to-end
- Virtual channels (VCs) and traffic classes (TCs) to support differentiated traffic or Quality of Service (QoS)
  - ✓ The ability to define levels of performance for packets of different TCs
  - ✓ 8 TC's and 8 VC's available

Additional Features

- Flow Control
  - ✓ No retry or disconnect as in PCI
- Interrupt Mechanism
  - ✓ Legacy PCI interrupt style mechanism
  - ✓ MSI
  - ✓ MSI-X
- Advanced power management
  - ✓ Active State PM (L0, L0s, L1 Active)
  - ✓ PCI compatible PM (L1 and L2/L3)

Additional Features

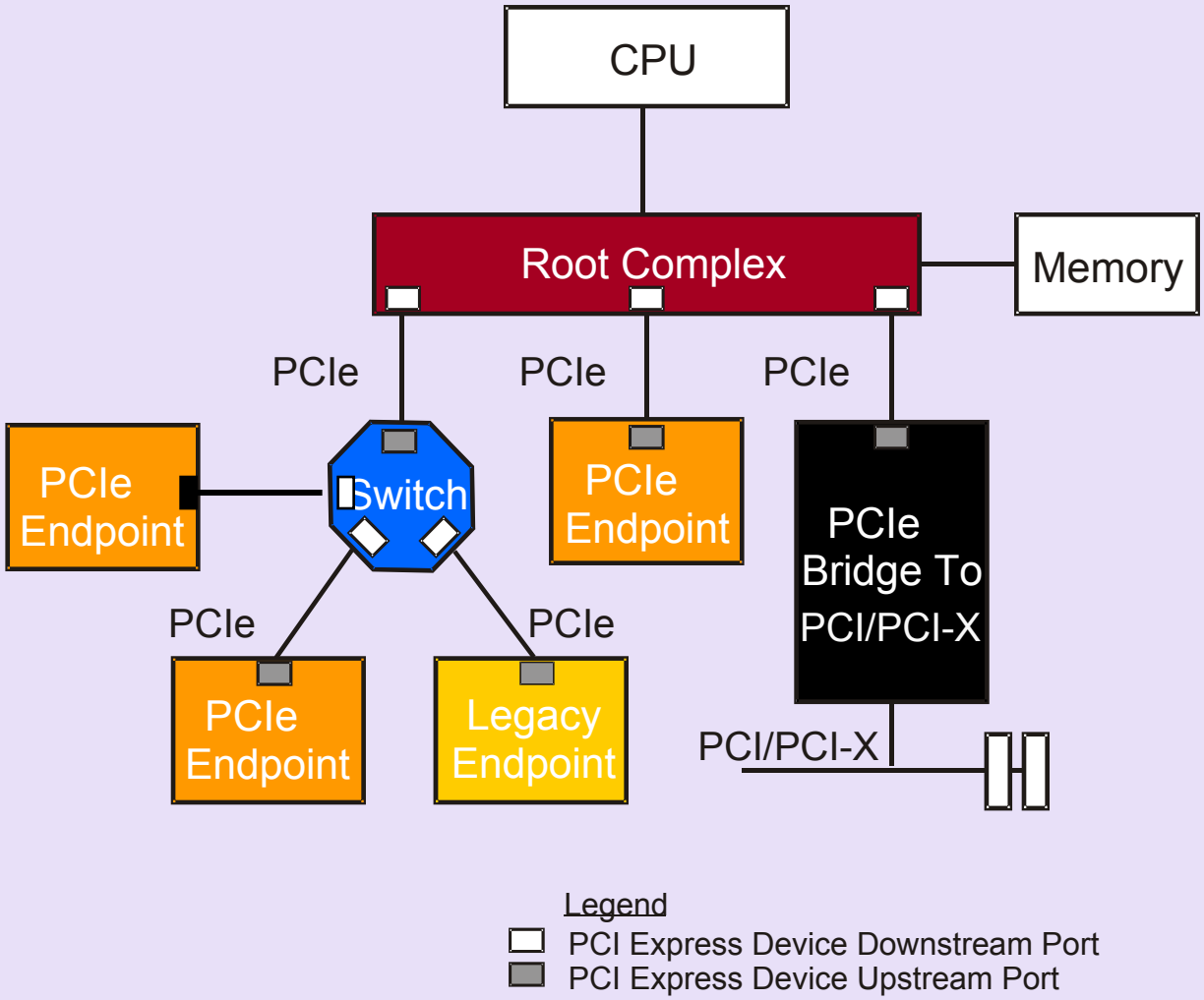
- Hot Plug and Hot Swap support
  - ✓ Native
  - ✓ No sideband signals
- PCI compatible software model
  - ✓ PCI configuration and enumeration software can be used to enumerate PCI Express hardware
  - ✓ PCI Express system will boot existing OS
  - ✓ PCI Express supports existing device drivers
  - ✓ New additional configuration address space requires OS and driver update

Additional Features

- Mechanical Form Factors
  - ✓ PCI-like peripheral card and connector
  - ✓ Mini PCI Express form factor
  - ✓ ExpressCard* form factor
  - ✓ ExpressModule™ form factor
  - ✓ AdvancedTCA and CompactPCI Express form factors for backplane implementation
- Future Form Factors
  - ✓ Cable. Spec in rev 0.3 as of July 2004

* Other names and brands may be claimed as the property of others.

PCI Express Topology



Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

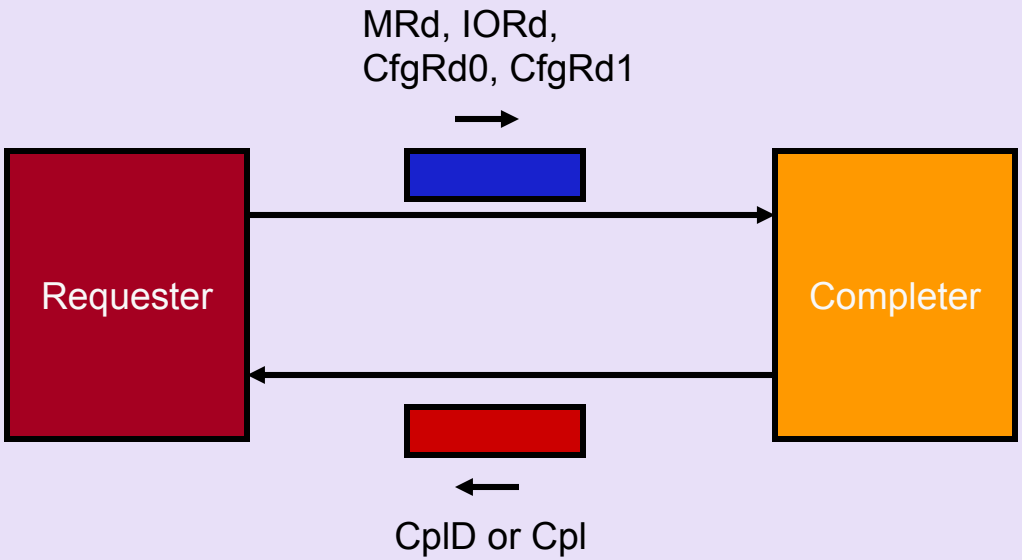
PCI Express Transaction Types

Transaction Type	Non-Posted or Posted
Memory Read	Non-Posted
Memory Write	Posted
Memory Read Lock	Non-Posted
IO Read	Non-Posted
IO Write	Non-Posted
Configuration Read (Type 0 and Type 1)	Non-Posted
Configuration Write (Type 0 and Type 1)	Non-Posted
Message with Data and without Data	Posted

PCI Express TLP Types

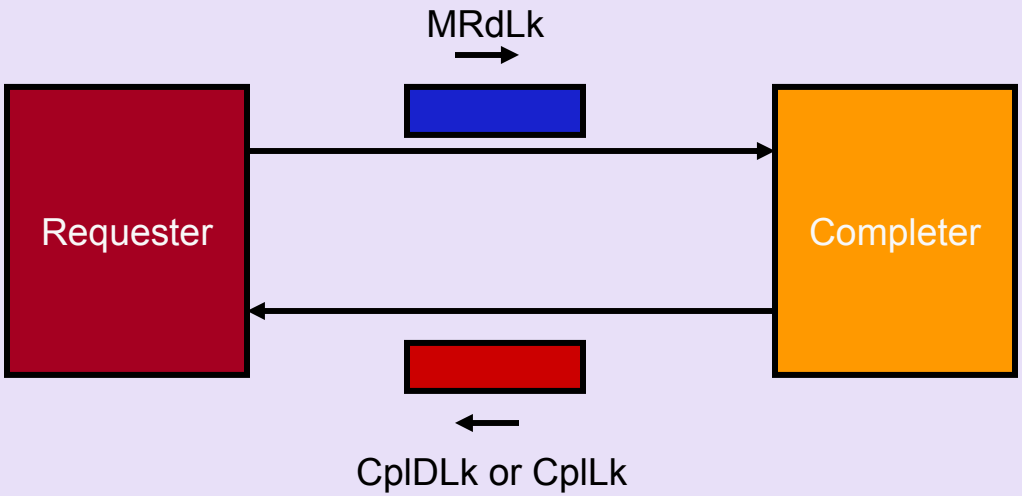
Description	Abbreviated TLP Name
Memory Read Request	MRd
Memory Read Request – Locked Access	MRdLk
Memory Write Request	MWr
IO Read Request	IORd
IO Write Request	IOWr
Configuration Read Request Type 0 and Type 1	CfgRd0, CfgRd1
Configuration Write Request Type 0 and Type 1	CfgWr0, CfgWr1
Message Request without Data Payload	Msg
Message Request with Data Payload	MsgD
Completion without Data (used for IO, configuration write completions and read completion with error completion status)	Cpl
Completion with Data (used for memory, IO and configuration read completions)	CplD
Completion for Locked Memory Read without Data (used for error status)	CplLk
Completion for Locked Memory Read with Data	CplDLk

Transaction Protocol – Non-Posted Transactions



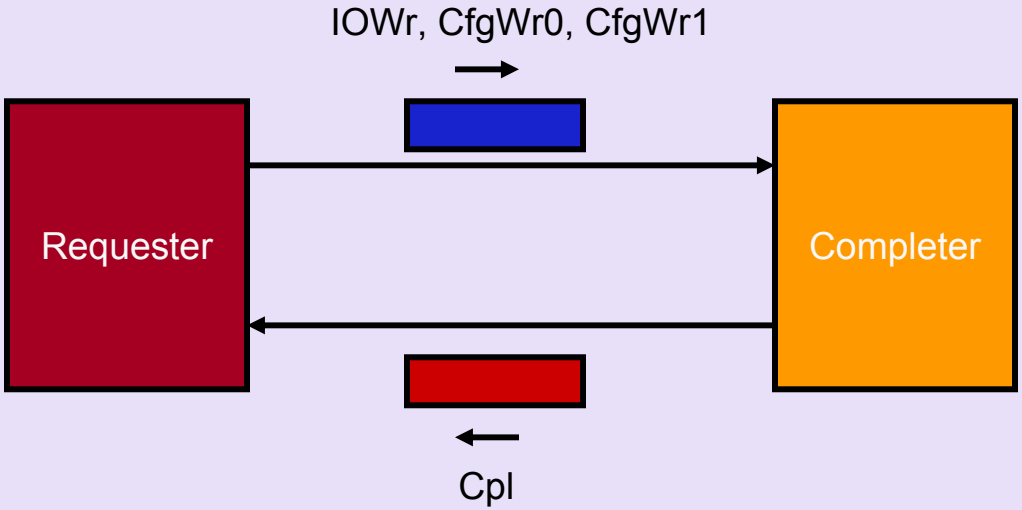
Legend:
MRd = Memory Read Request
IORd = IO Read Request
CfgRd0 = Type 0 Configuration Read Request
CfgRd1 = Type 1 Configuration Read Request
CplD = Completion with data for normal completion of MRd, IORd, CfgRd0, CfgRd1
Cpl = Completion without data for error completion of MRd, IORd, CfgRd0, CfgRd1

Transaction Protocol – Non-Posted Locked Transactions



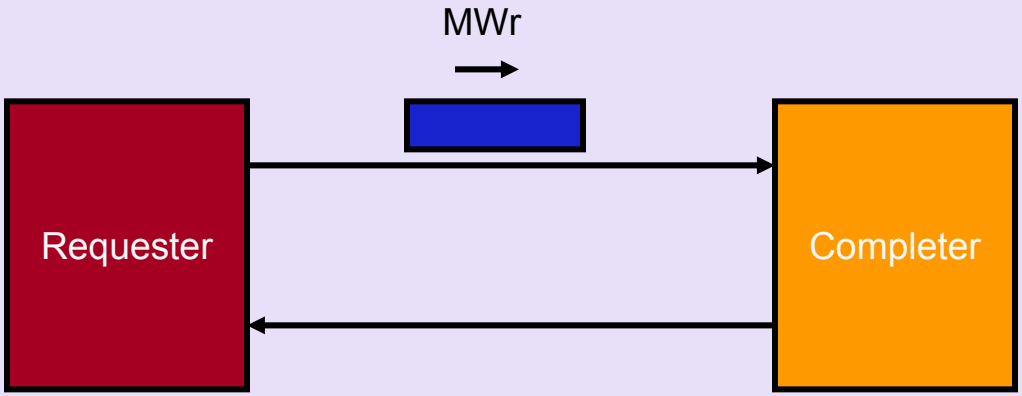
Legend:
MRdLk = Memory Read Lock Request
CplDLk = Locked normal Completion with data for normal completion of MRdLk
CplLk = Locked error Completion without data for error completion of MRdLk

Transaction Protocol – Non-Posted Write Transactions



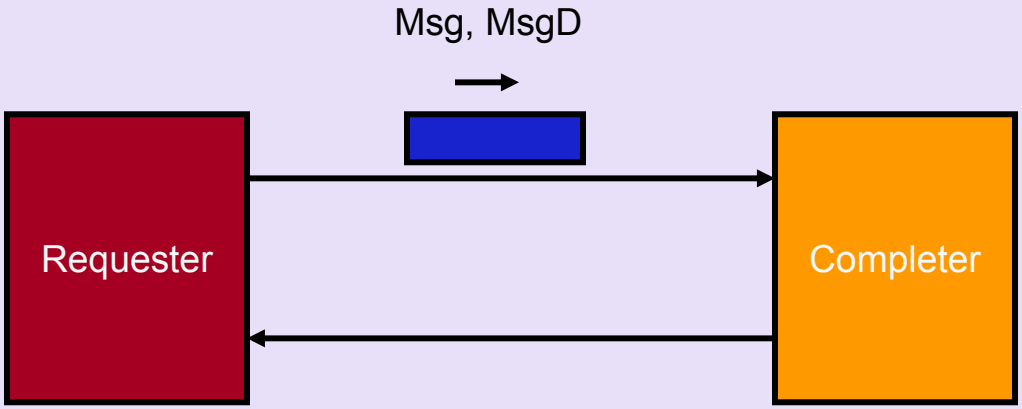
Legend:
IOWr = IO Write Request
CfgWr0 = Type 0 Configuration Write Request
CfgWr1 = Type 1 Configuration Write Request
Cpl = Completion without data for normal or error completion of IOWr, CfgWr0, CfgWr1

Transaction Protocol – Posted Memory Write Transactions



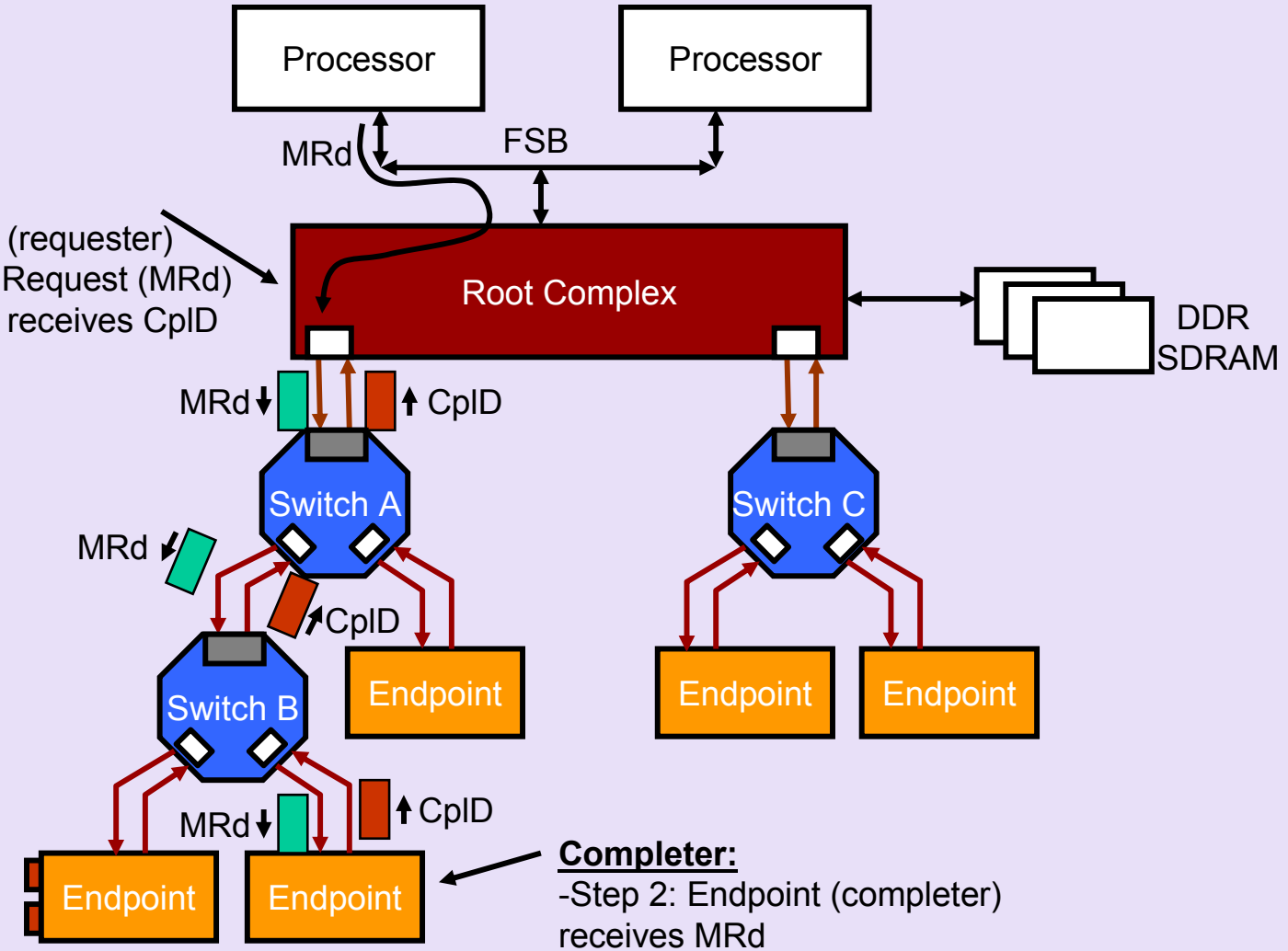
Legend:
MWr = Memory Write Request. No completions for this transaction

Transaction Protocol – Posted Message Transactions



Legend:
Msg = Message Request without data
MsgD = Message Request with data

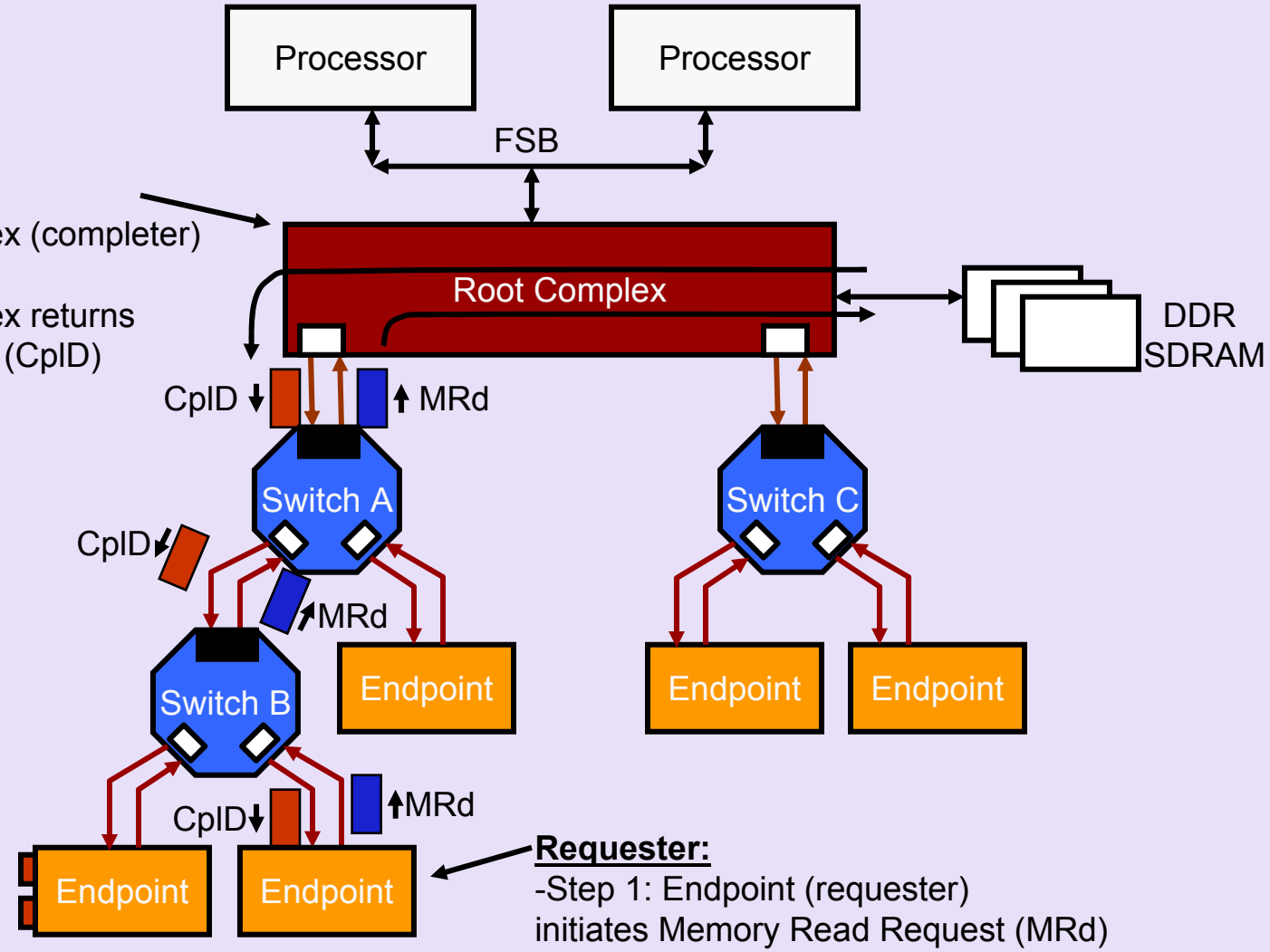
Non-Posted Memory Read



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)

Non-Posted Memory Read

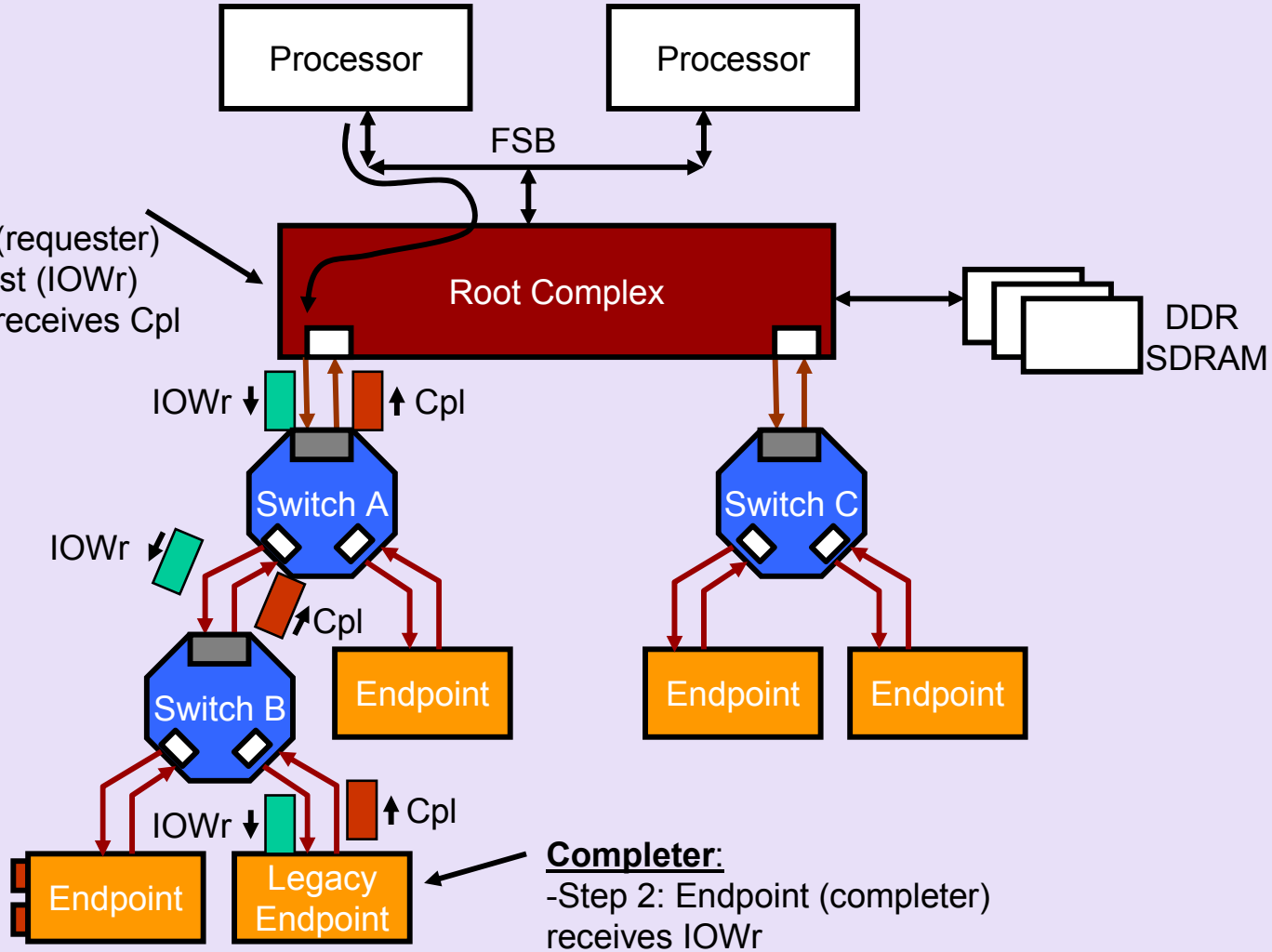


Completer:
-Step 2: Root Complex (completer) receives MRd
-Step 3: Root Complex returns Completion with data (CplD)

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

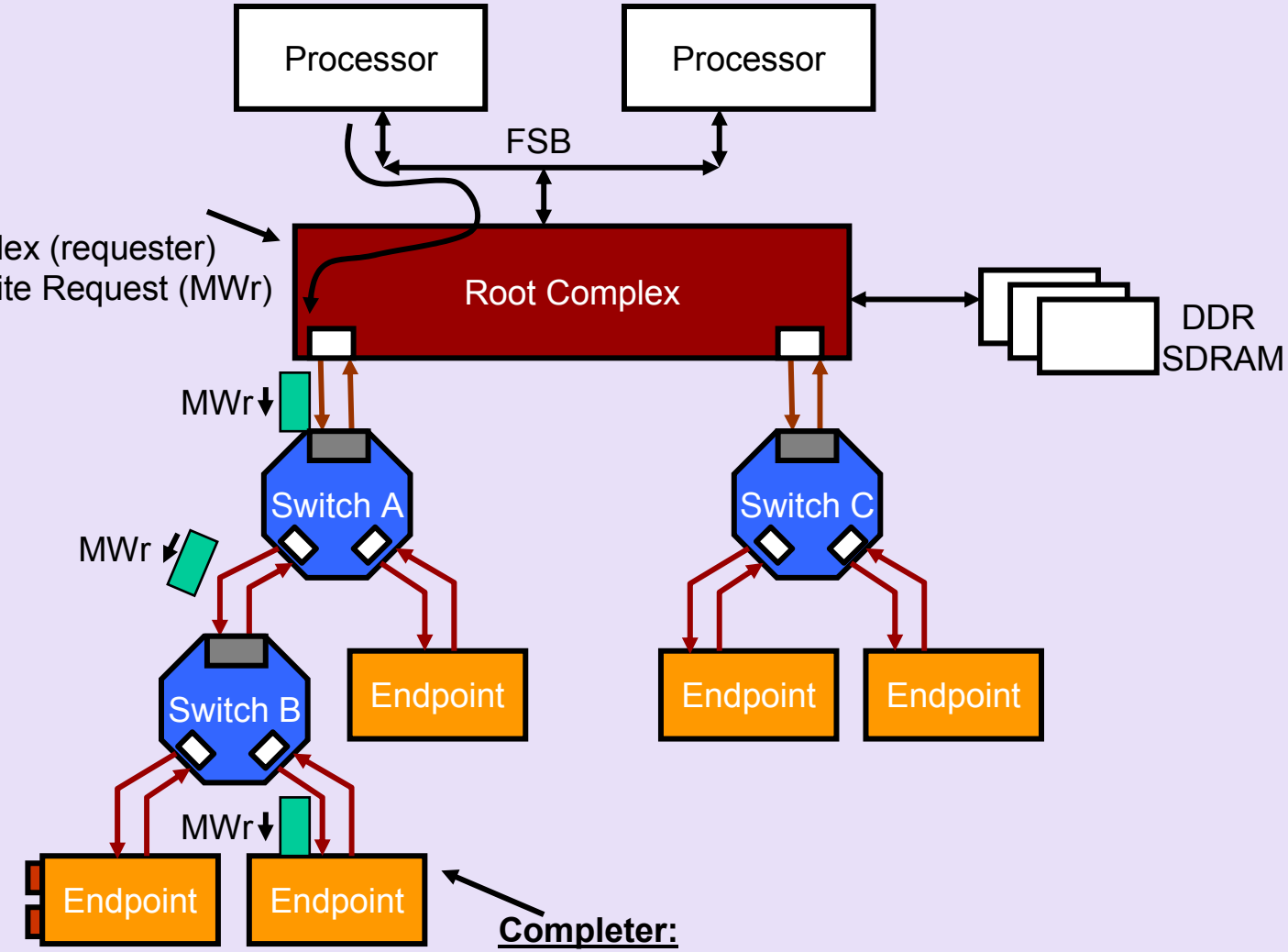
Non-Posted IO Write

Requester:
-Step 1: Root Complex (requester) initiates IO Write Request (IOWr)
-Step 4: Root Complex receives Cpl



Completer:
-Step 2: Endpoint (completer) receives IOWr
-Step 3: Endpoint returns Completion without data (Cpl)

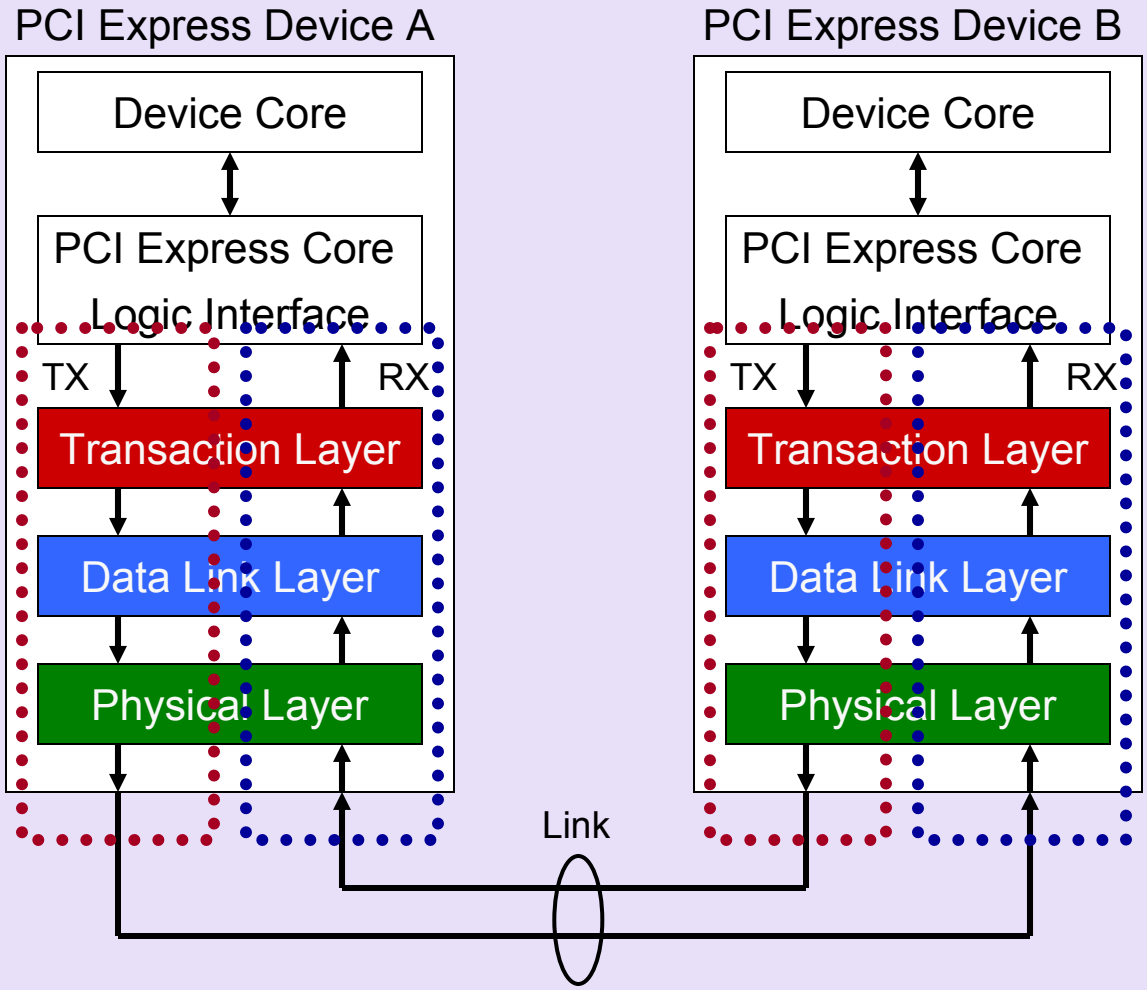
Posted Memory Write



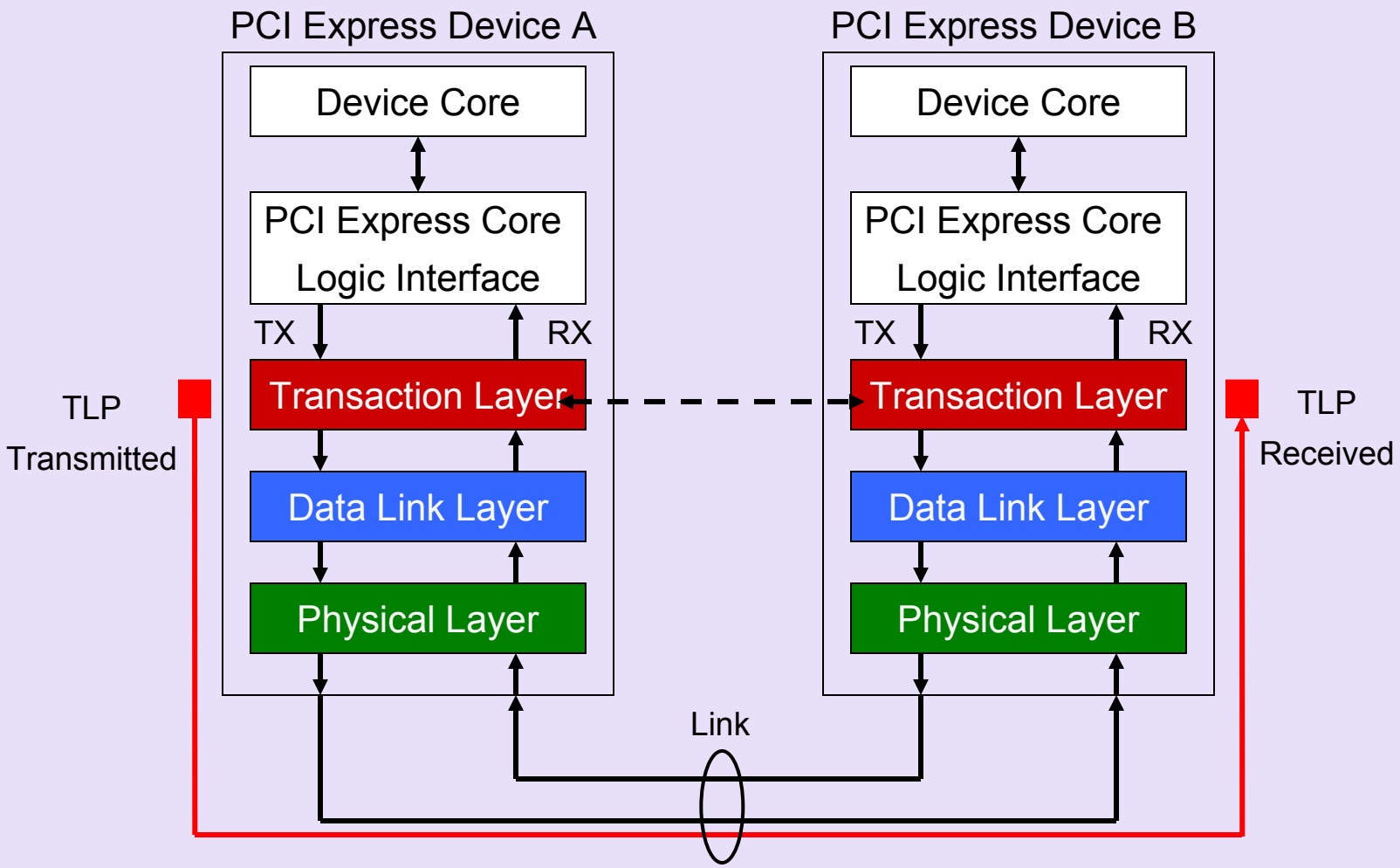
Requester:
-Step 1: Root Complex (requester) initiates Memory Write Request (MWr)

Completer:
- Step 2: Endpoint (completer) receives MWr

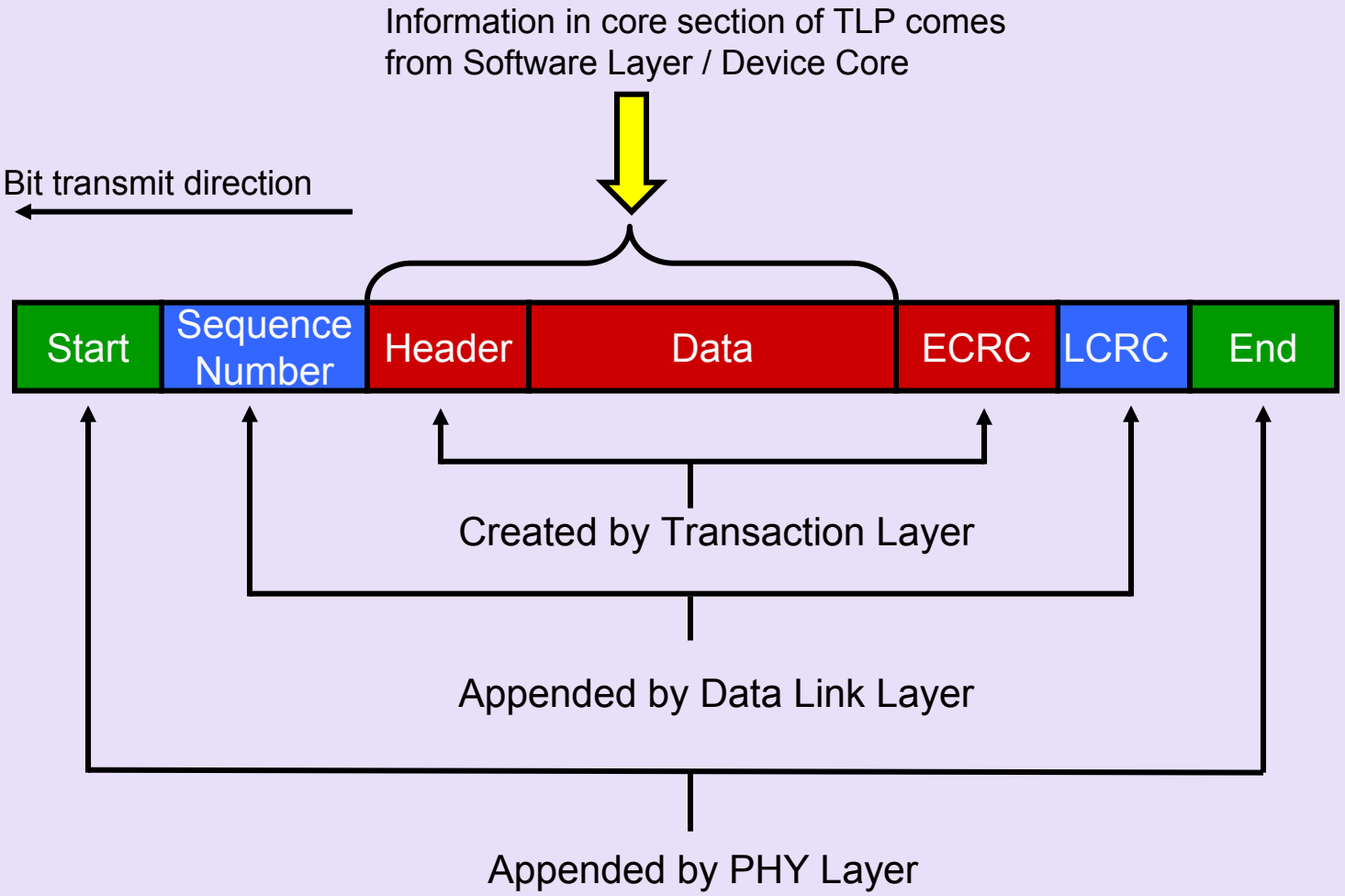
PCI Express Device Layers



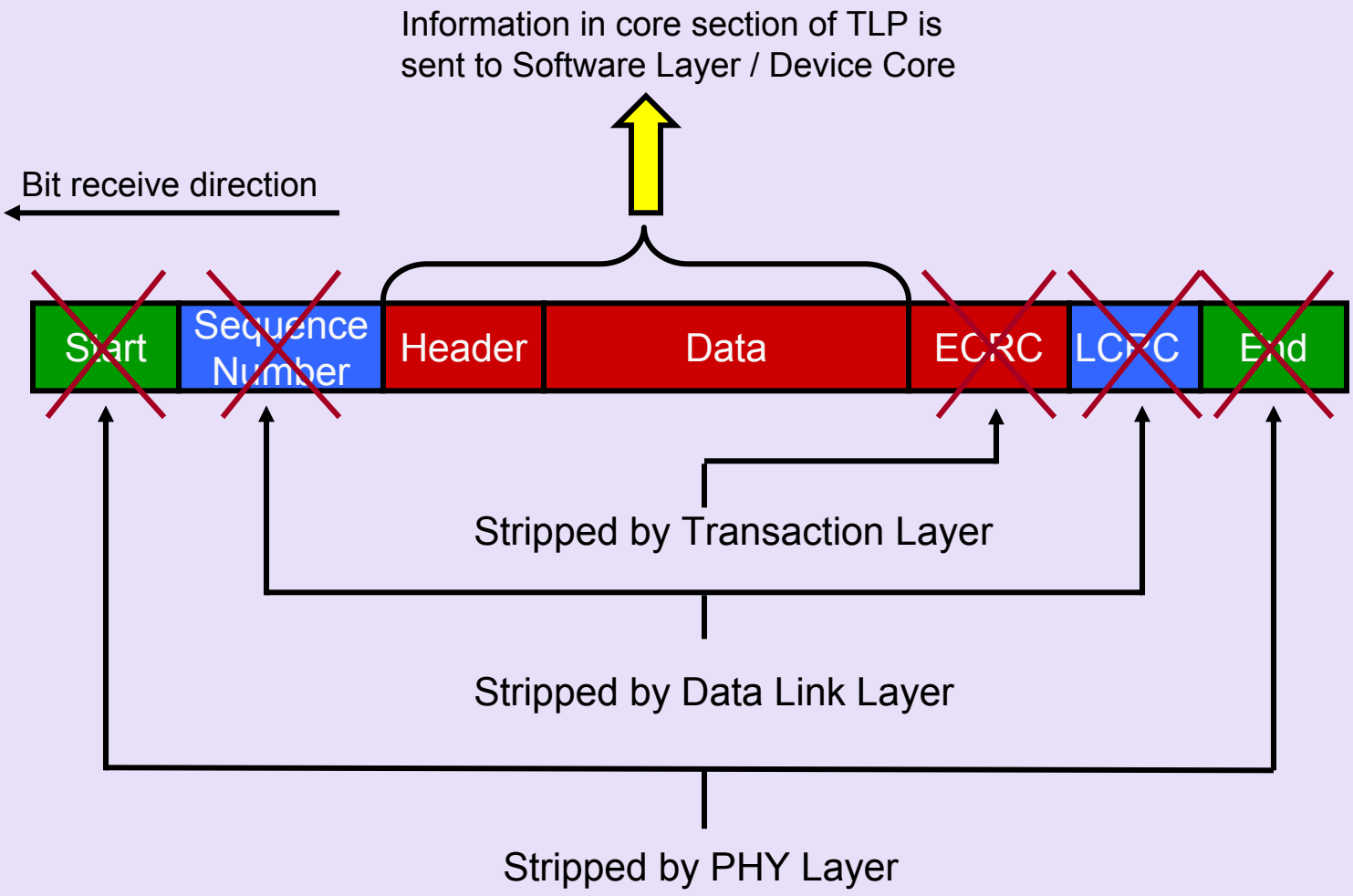
TLP Origin and Destination



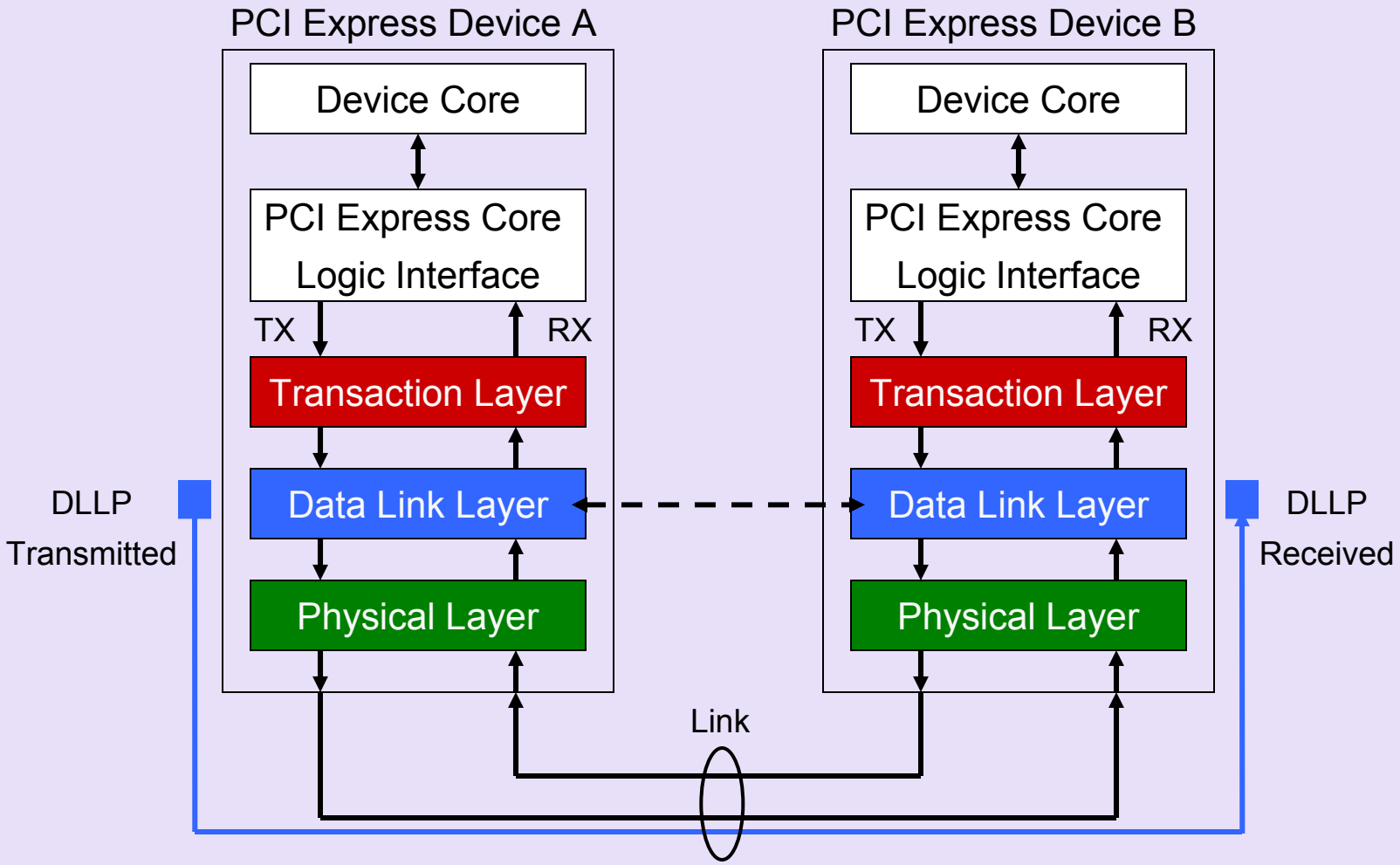
TLP Assembly



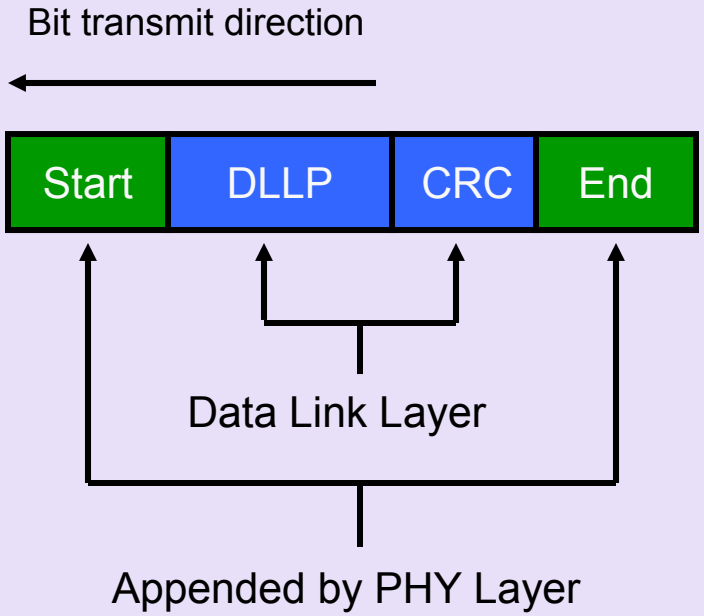
TLP Disassembly



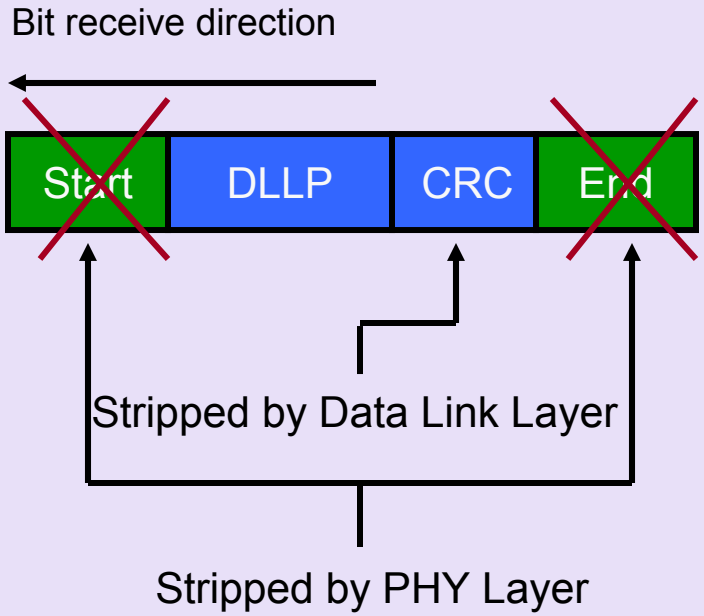
DLLP Origin and Destination



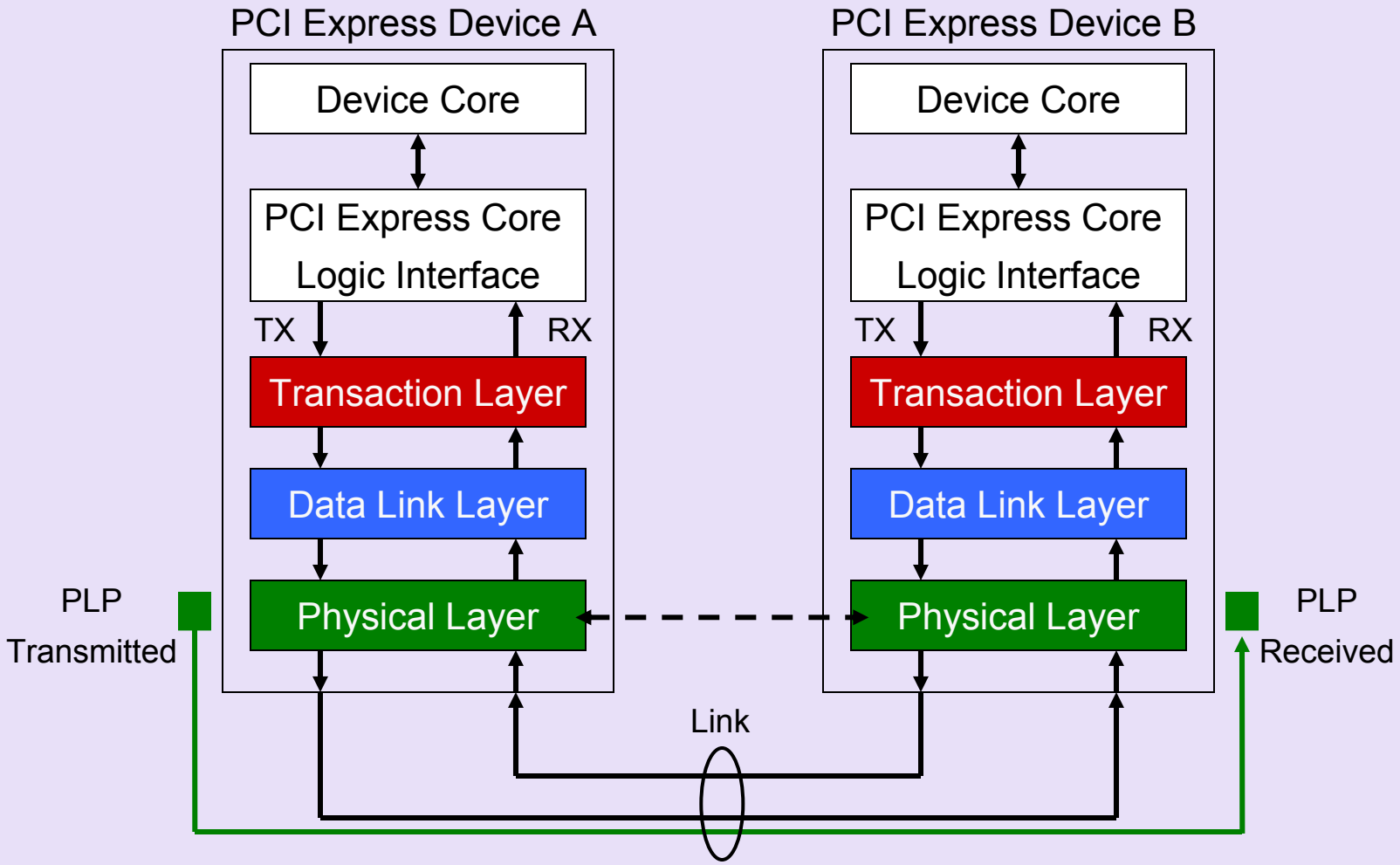
DLLP Assembly



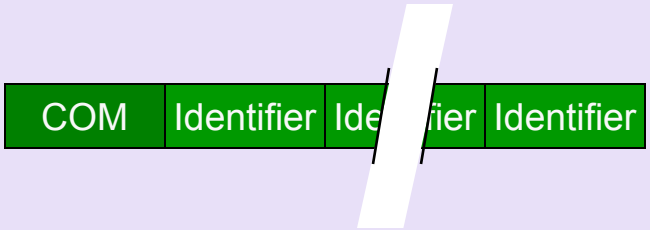
DLLP Disassembly



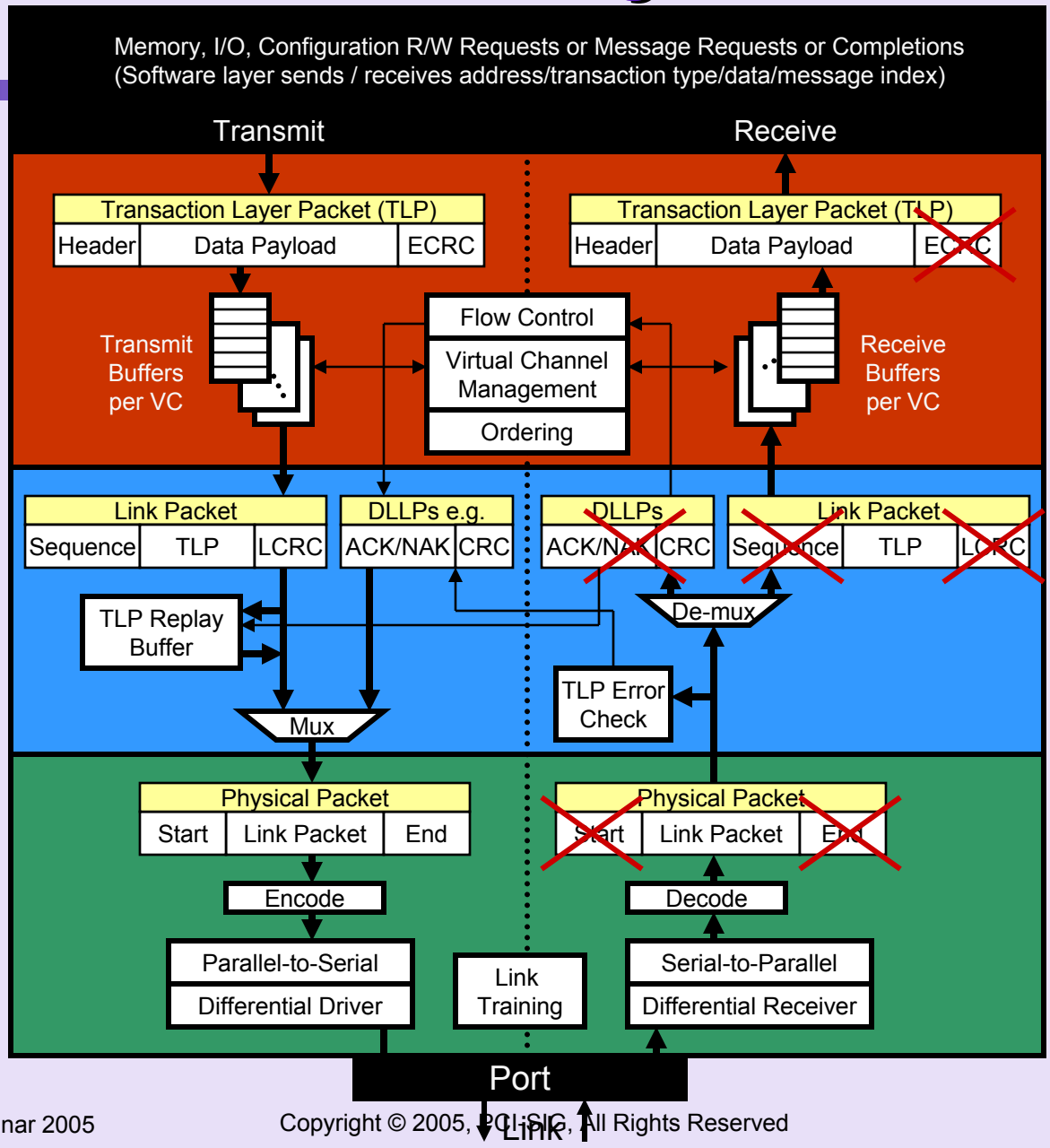
PLP Origin and Destination



PLP or Ordered-Set Structure



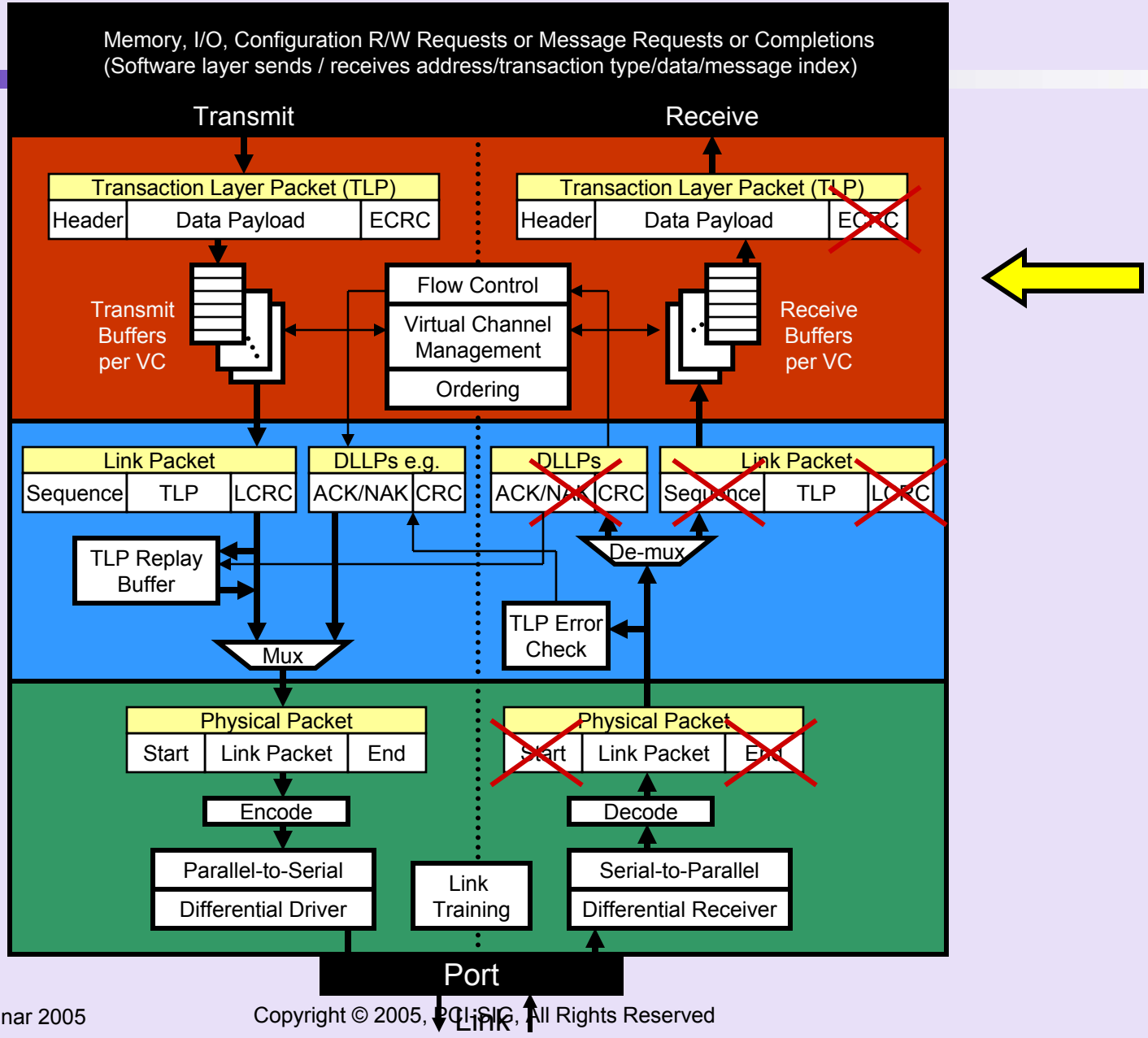
Device with Layer Details



Agenda

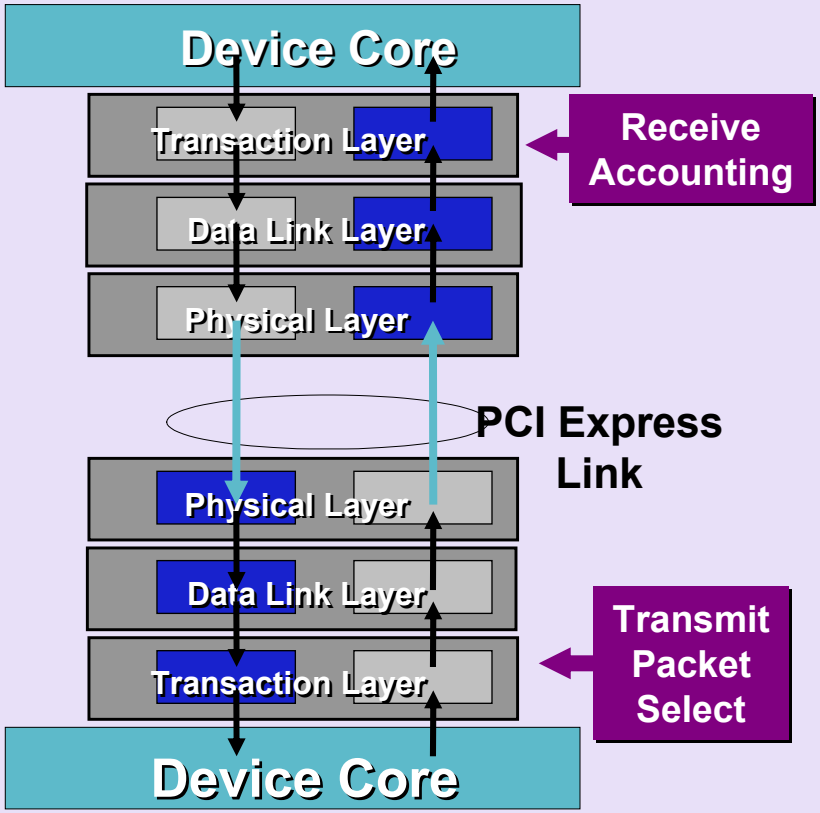
- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Transaction Layer



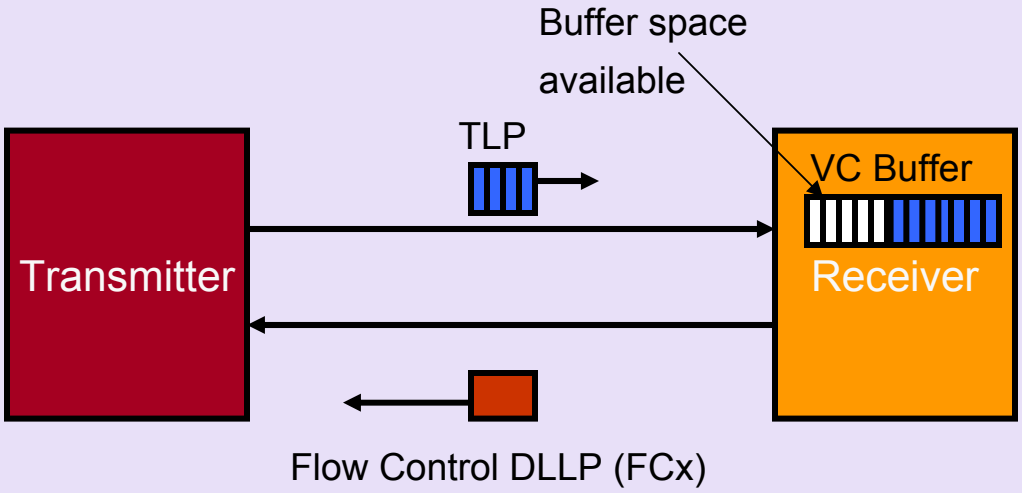
Credit-Based Flow Control

- Transaction to Transaction Layer across one Link
- Prevents overflow of receiver buffers
- Enables compliance with ordering rules
- Credit-based scheme
  - ✓ Transmitter throttles according to its supply of credits
  - ✓ Requesters can not use FC to throttle completions



Flow control and ordering rules enable deadlock-free operations

Credit-Based Flow Control

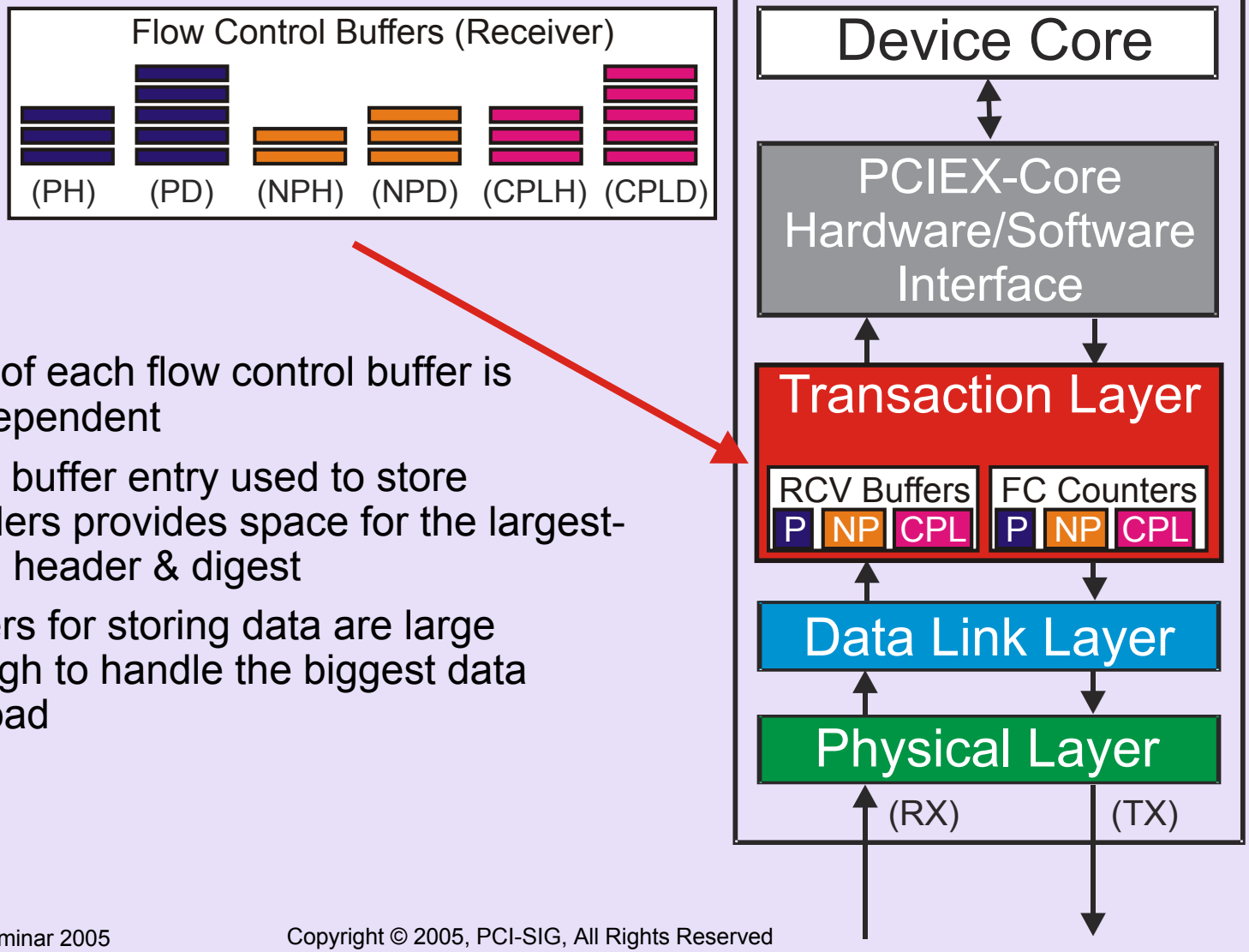


Receiver sends Flow Control Packets (FCP) which are a type of DLLP (Data Link Layer Packet) to provide the transmitter with credits so that it can transmit packets to the receiver

Credit-Based Flow Control

- Handled by the Transaction Layer in cooperation with the Data Link Layer
  - ✓ DLLPs used to exchange FC information
- FC information covers separately with 6 queues:
  - ✓ Posted and Non-Posted Request Queues, Completion Queues
  - ✓ Separate Header vs. Data Queues
- FC is orthogonal to the data integrity mechanisms
- FC Ack does not imply Request completion

Flow Controlled Buffers

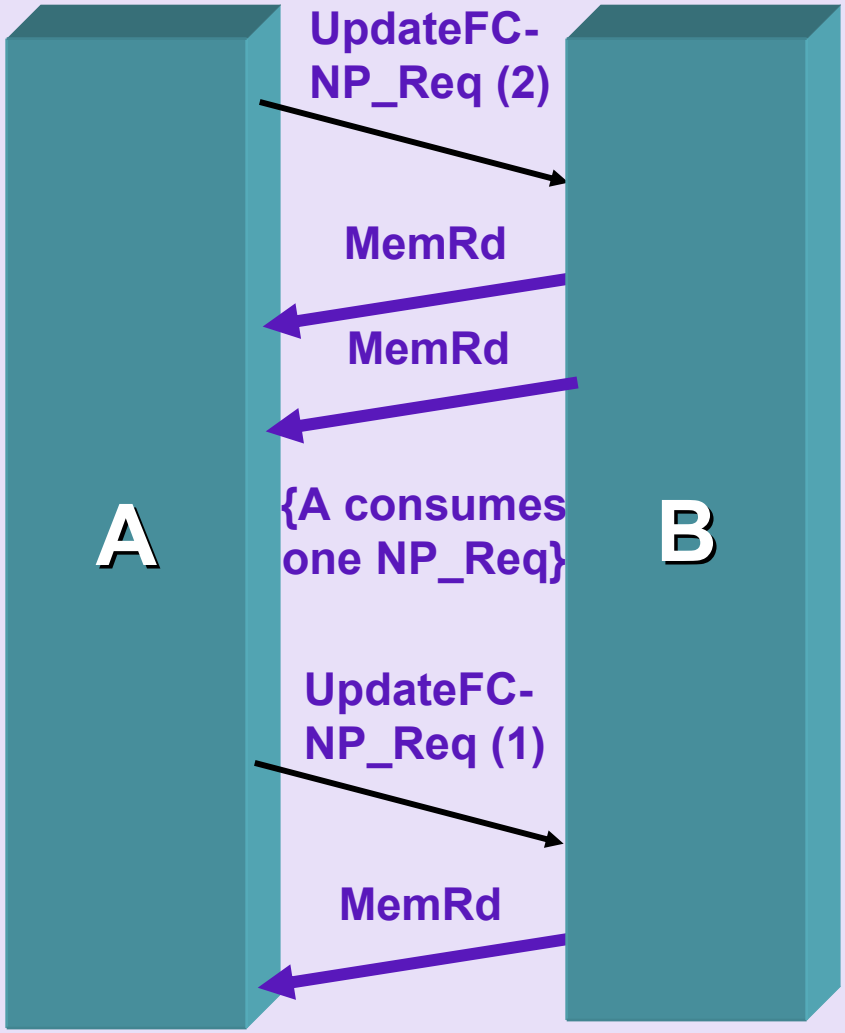


The size of each flow control buffer is design dependent

- ✓ Each buffer entry used to store Headers provides space for the largest-sized header & digest
- ✓ Buffers for storing data are large enough to handle the biggest data payload

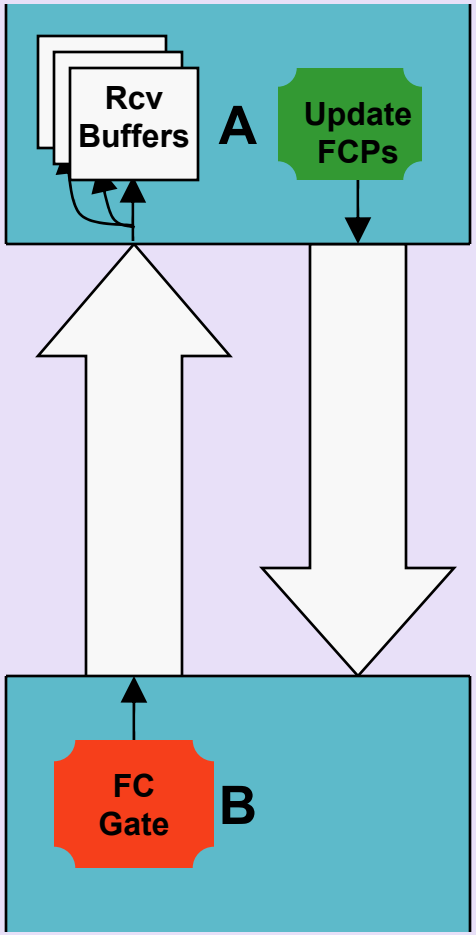
Flow Control Example

1. A - advertises buffer space for two Non-Posted Requests
2. B - sends two Memory Read Requests
3. A - consumes one of the Non-Posted Requests
4. A - advertises the released buffer space to B
5. B - sends another Memory Read Request



Receive Buffer Sizing and Flow Control Credit Return Policy

- Receive Buffer sizing and Flow Control (FC) credit return policy have a significant effect on performance
  - ✓ Much inbound traffic → Receive Buffer optimization critical
- Tradeoff: Bandwidth for Update Flow Control Packets (UpdateFCPs) vs. Receive Buffer Size
  - ✓ UpdateFCPs affect *outbound* traffic
- Link Width & Max/Typical Payload size must be considered



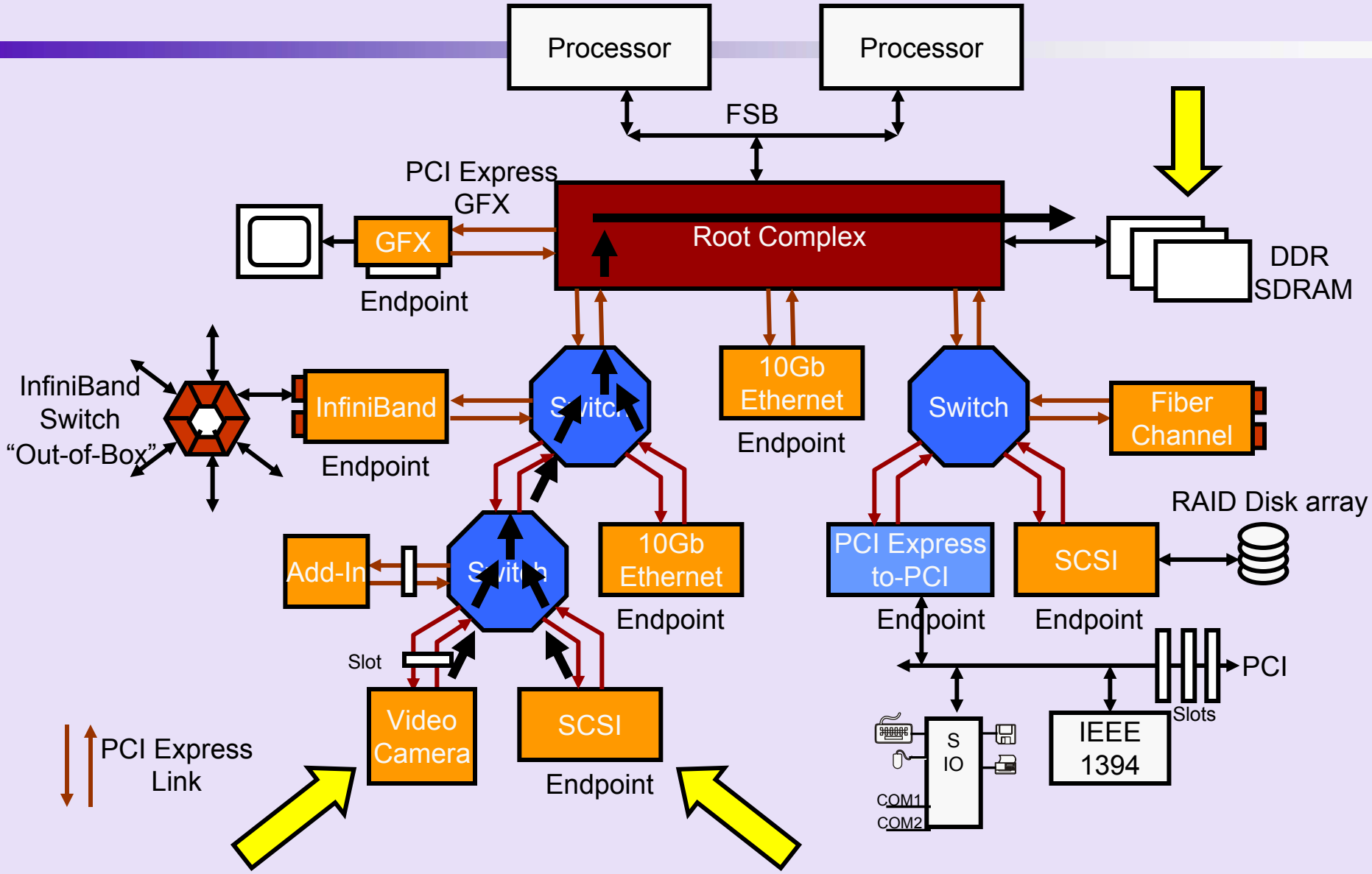
Related Efficiency Considerations

- Flow Control data credit granularity also has an effect on utilization
  - ✓Payloads of multiples of 4DW are optimal
- For best service, 128B address alignment highly desirable
- For B/W efficiency reasons, requests of 128B or larger are recommended
  - ✓Local buffering/caching may be desirable

Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Quality of Service



Virtual Channels

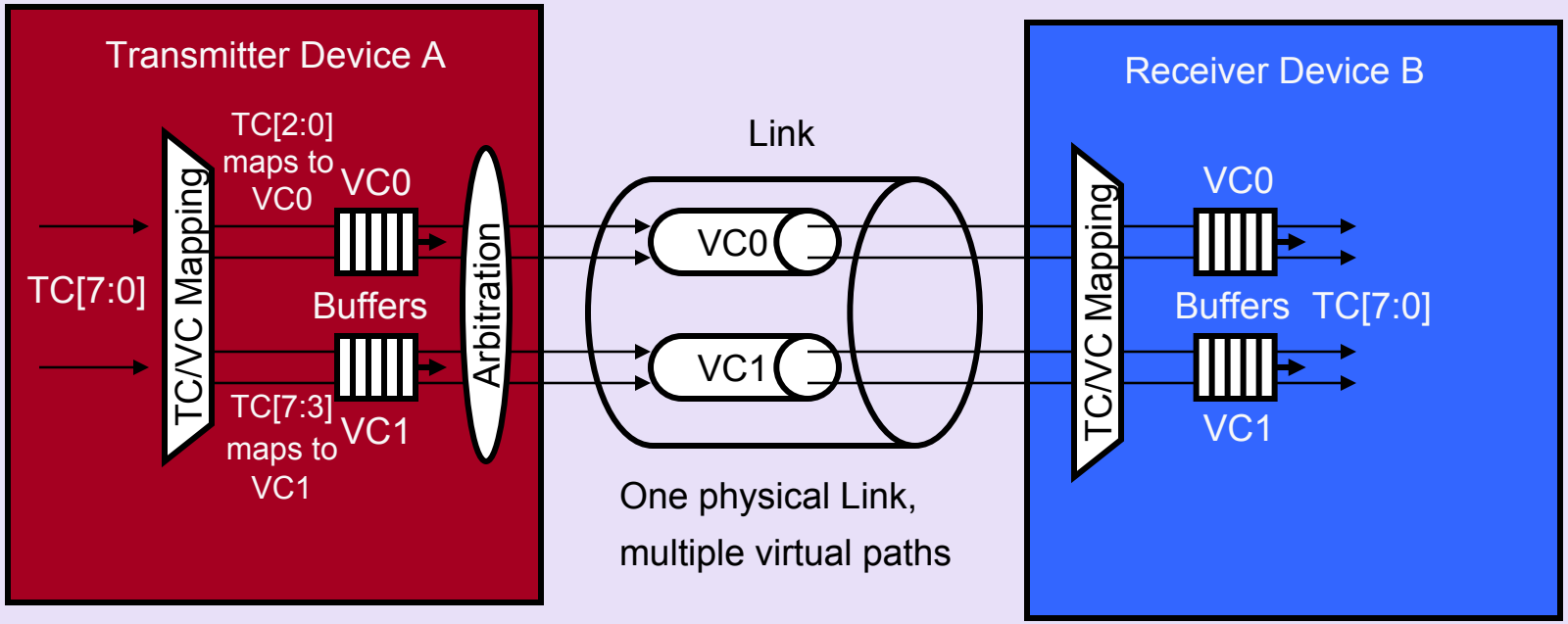
- Virtual Channels = mechanism for decoupling independent traffic
  - ✓ Fundamentally new capability for PCI
- Virtual Channels = support for Quality of Service
- Root Complex & Switch support key to enabling platform-level Traffic Class differentiated servicing
- Virtual Channels provide basis for interconnect Isochronous support
  - ✓ Good: Support key features needed for Isoc – be “Isoc Ready”
  - ✓ Best: Support Isoc capability

Support Recommendations:

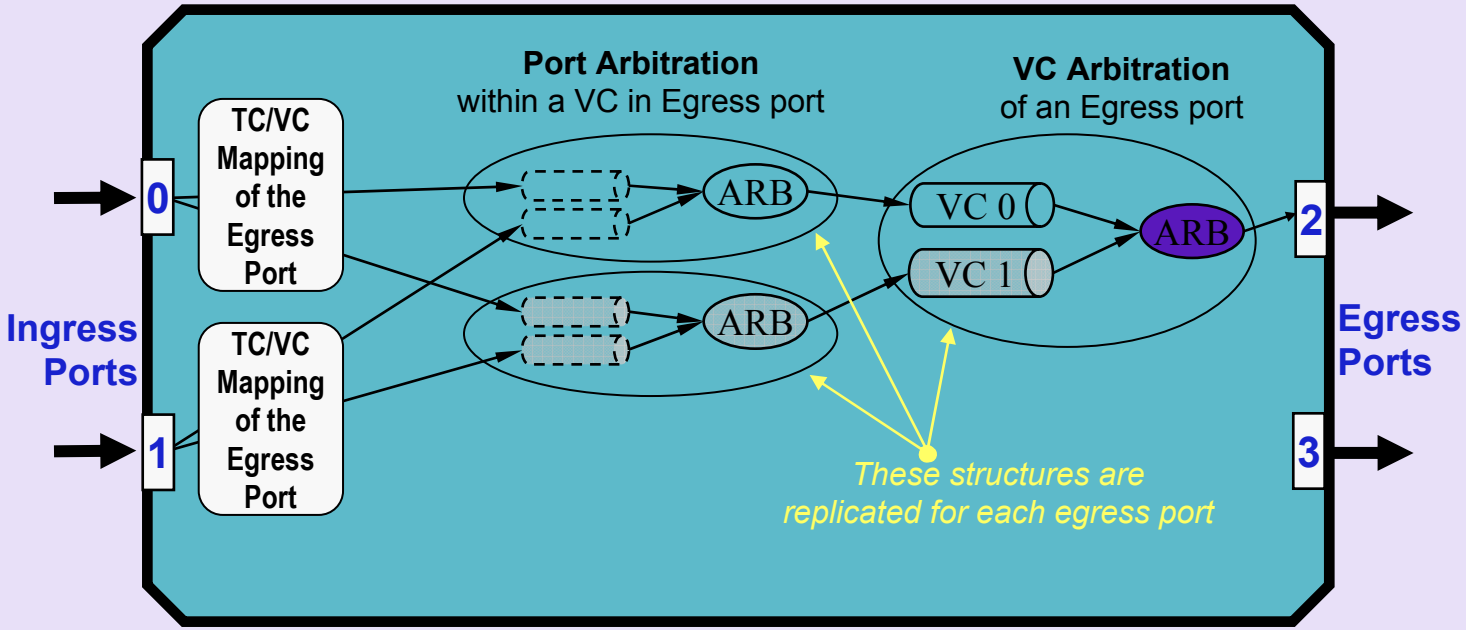
- Root Complex & Switch:
  - ✓ Minimum of 2 VCs
  - ✓ VC arbitration such as strict priority
  - ✓ Time-based WRR port arbitration for VCs other than VC0
  - ✓ Meet Isoch latency guidelines
- Endpoint:
  - ✓ Minimum of 2 VCs
  - ✓ Size buffers to meet latency guidelines
  - ✓ Collect data up to Max_Payload_Size
  - ✓ Naturally align data payloads
  - ✓ Proper use of “No Snoop”

Quality of Service

- Quality of Service (QoS) policy through Virtual Channel and Traffic Class tags



Switch Arbitration Model

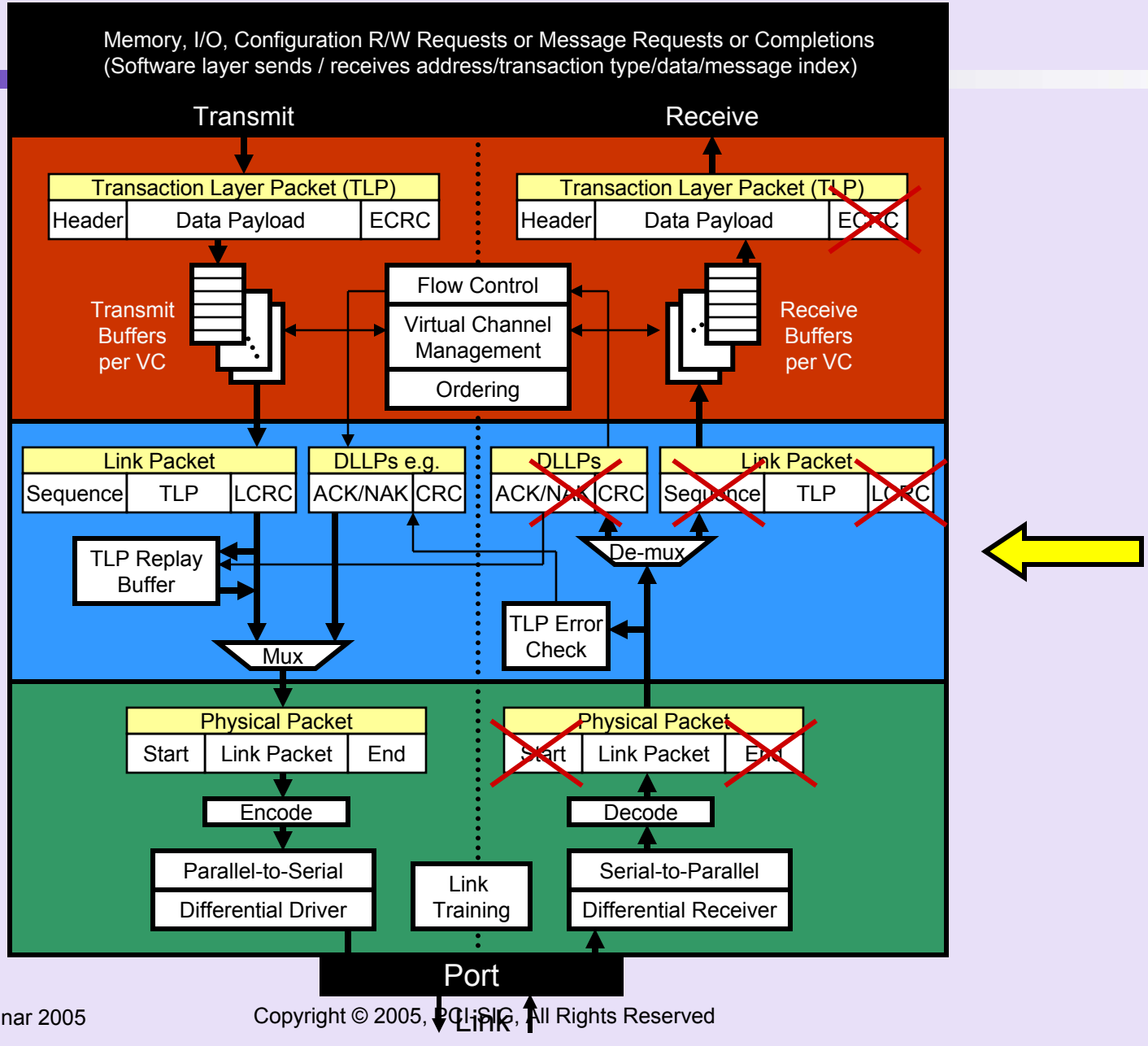


- Port Arbitration:
  - ✓ Traffic targeting same VC/Egress Port
  - ✓ Fixed Round-Robin (RR), programmable Weighted RR, programmable Time-based WRR
- VC Arbitration:
  - ✓ Traffic from different VC competing for the Link
  - ✓ Fixed priority, RR, programmable WRR

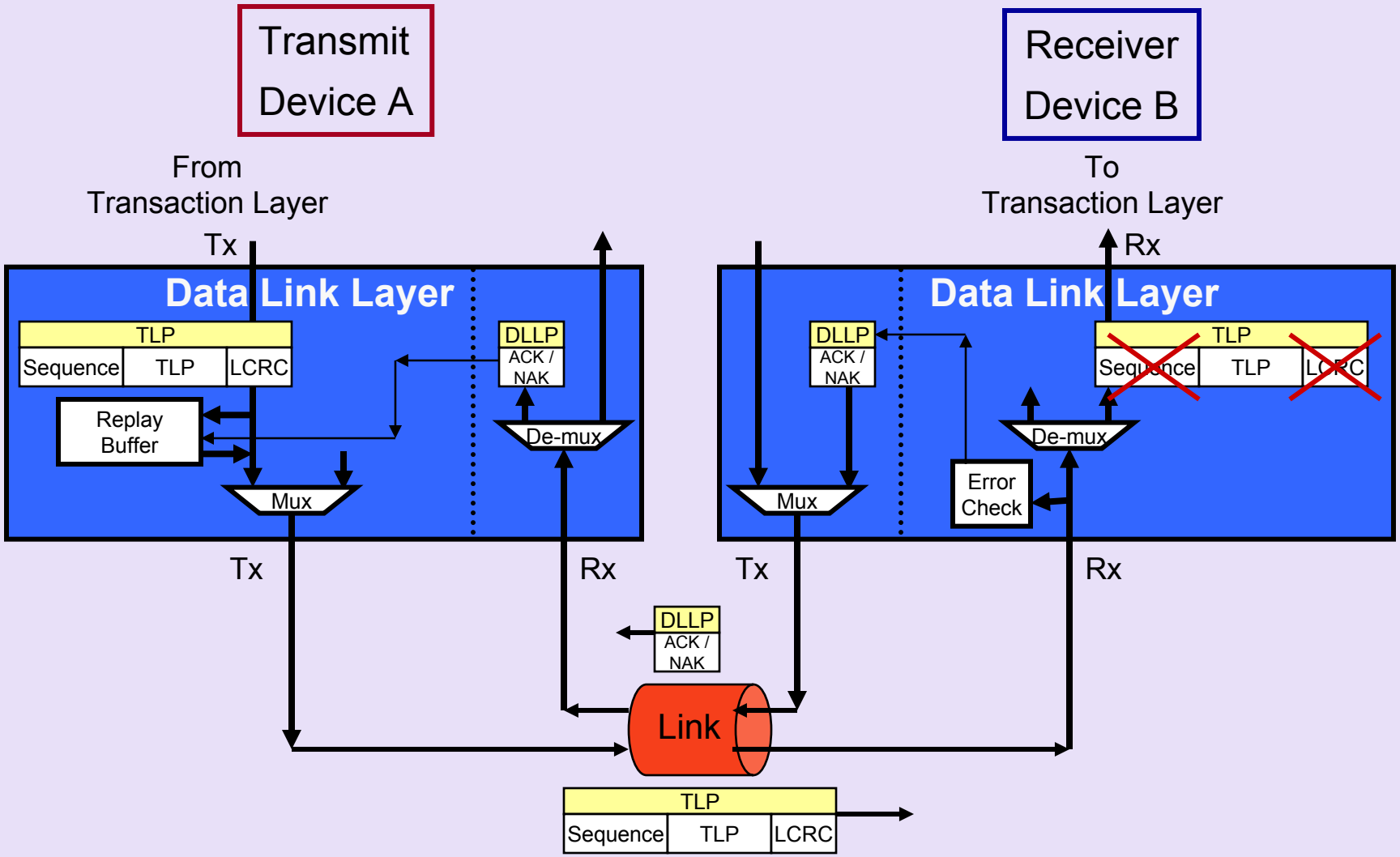
Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Data Link Layer



ACK/NAK Protocol Overview



Data Integrity Support

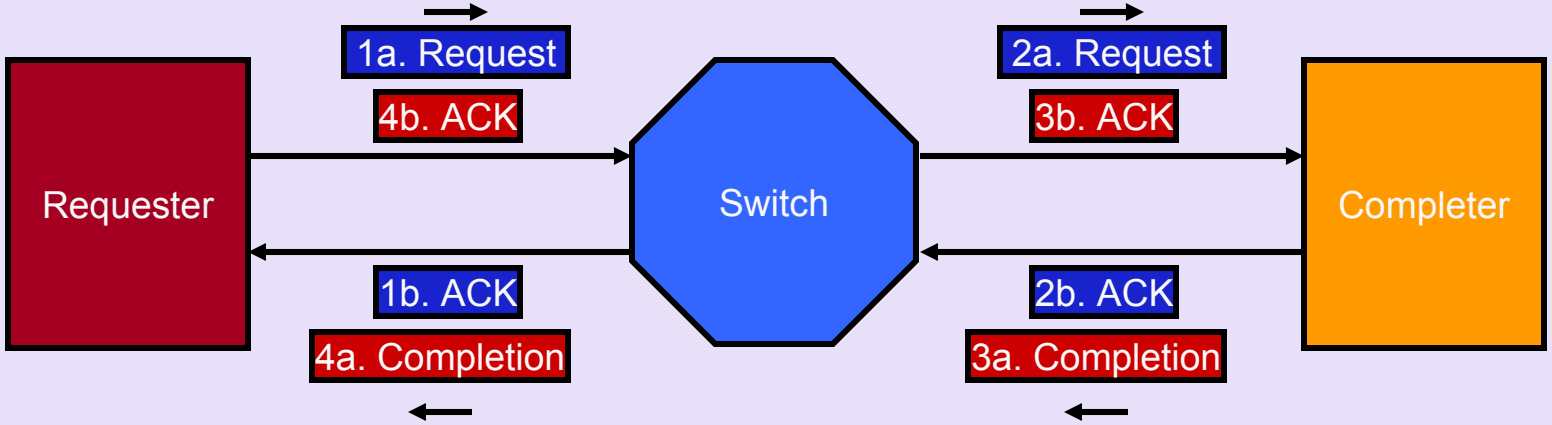
- Requirements for Robust Data Integrity
- Data Link Layer Mechanisms (Link/local):
  - ✓ TLPs protected using 32bit CRC
  - ✓ DLLPs protected using 16bit CRC
  - ✓ TLP error recovery through Data Link-level retry
  - ✓ Supplemental coverage through 8b/10b
  - ✓ Loss of packets detected using Sequence Numbers
- Transaction Layer Mechanisms (End-to-End):
  - ✓ Optional coverage using 32bit CRC
  - ✓ Data Poisoning capability

Robust Data Integrity Allows for Signaling Frequency Headroom

Link Data Integrity for TLPs

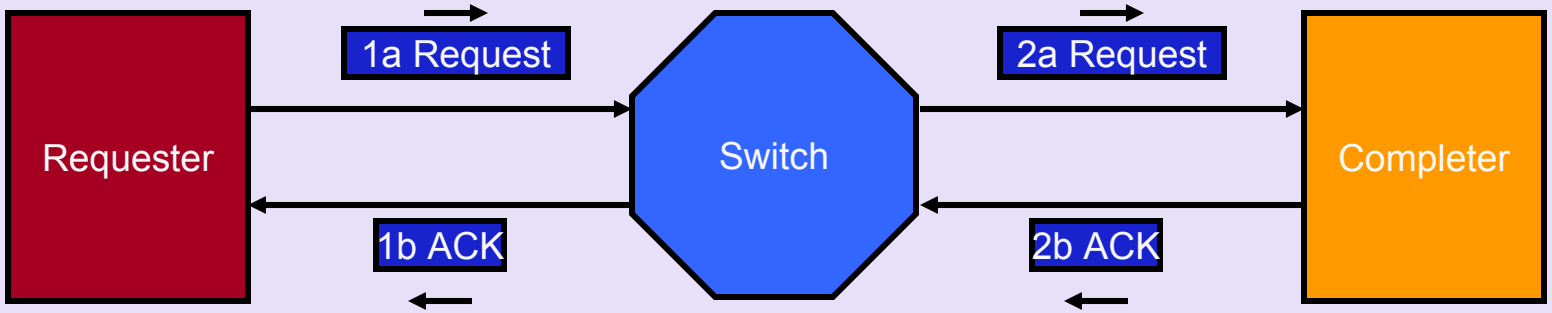
- Covers integrity of Link between two directly attached PCI Express devices across one Link
- Transmit side :
  - ✓ Applies 32bit CRC and Sequence # to Transaction Layer Packets
  - ✓ Buffers TLPs to allow retransmission
- Receive side:
  - ✓ Validates received TLPs by:
    - Checking the CRC code
    - Checking the Packet Sequence Number
    - Checking Phy Layer status for 8b/10b errors and framing errors
  - ✓ In the case of error:
    - Affected packet and following packets are discarded
    - NACK DLLP is sent to Transmitter to request retransmission
- Transmitter time-out causes re-transmission if TLP completely lost

Non-Posted Transactions



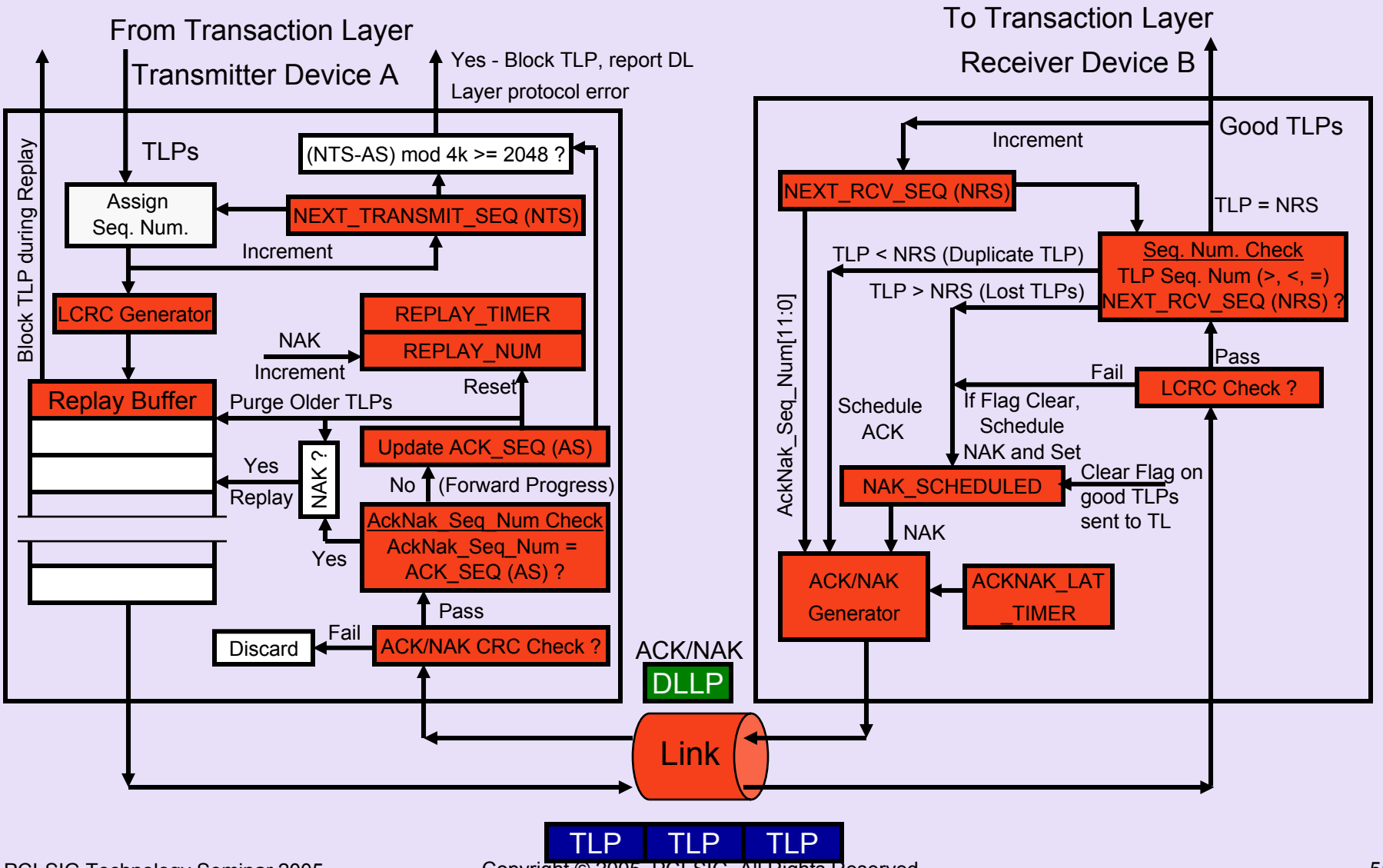
ACK returned for good reception of Request or Completion
NAK returned for error reception of Request or Completion

Posted Transactions



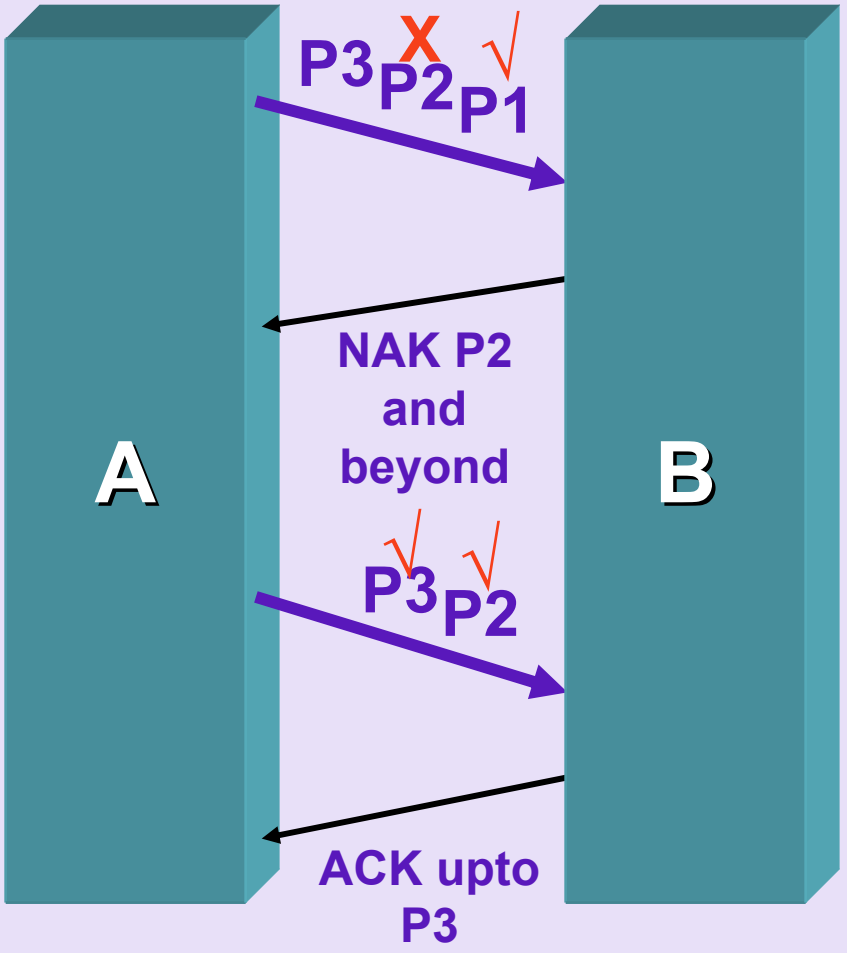
ACK returned for good reception of Request or Completion
NAK returned for error reception of Request or Completion

Elements of ACK/NAK Protocol



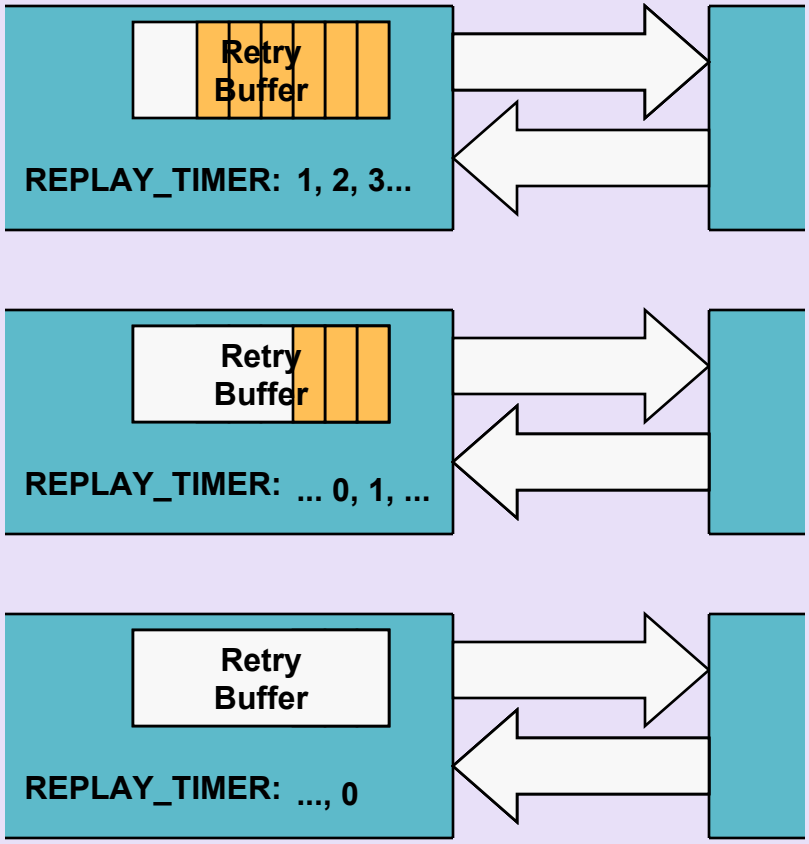
Link Data Integrity – Retry Example

1. Three TLPs sent from A to B
2. Packet 2 corrupted
3. B detects corruption and issues Nak DLLP
4. A resends Packet 2 and following Packet
5. B acknowledges successful receipt of Packets



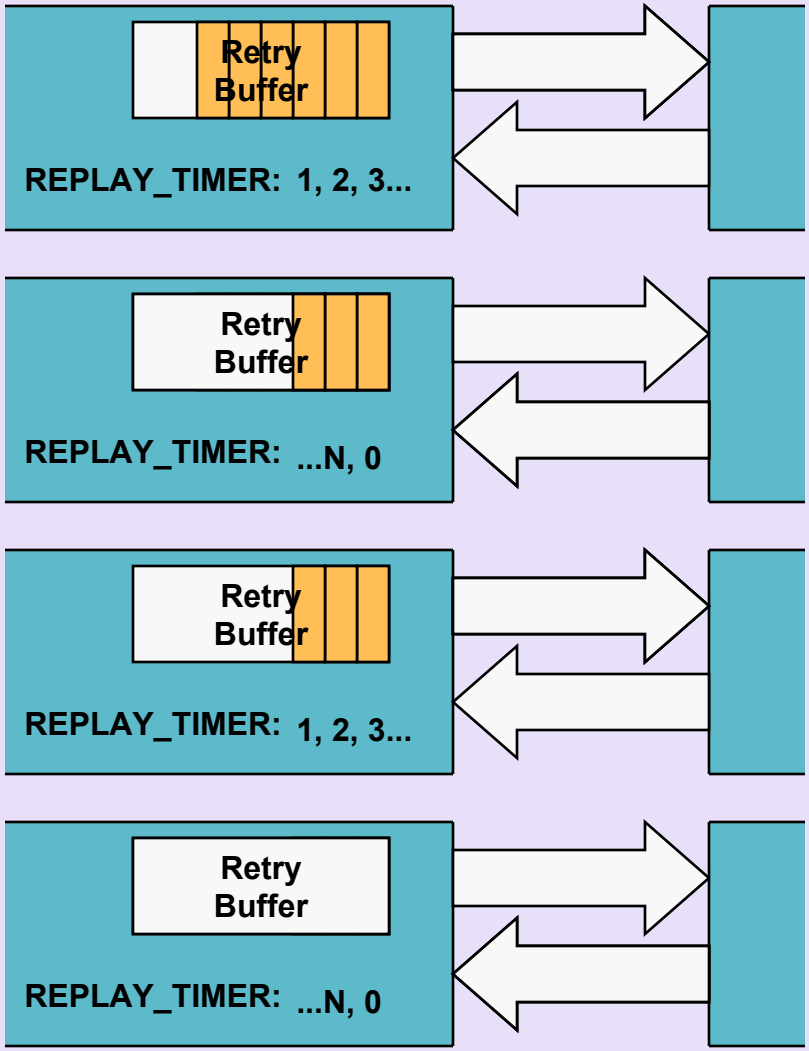
Example: REPLAY_TIMER Operation – Case 1

- Six TLPs are transmitted
  - ✓ REPLAY_TIMER started
- An Ack is received that acknowledges three of the transmitted TLPs
  - ✓ REPLAY_TIMER reset and restarted, because there are still outstanding unacknowledged TLPs
- ... eventually, the remaining TLPs are Ack'd
  - ✓ REPLAY_TIMER resets and holds – no outstanding unacknowledged TLPs



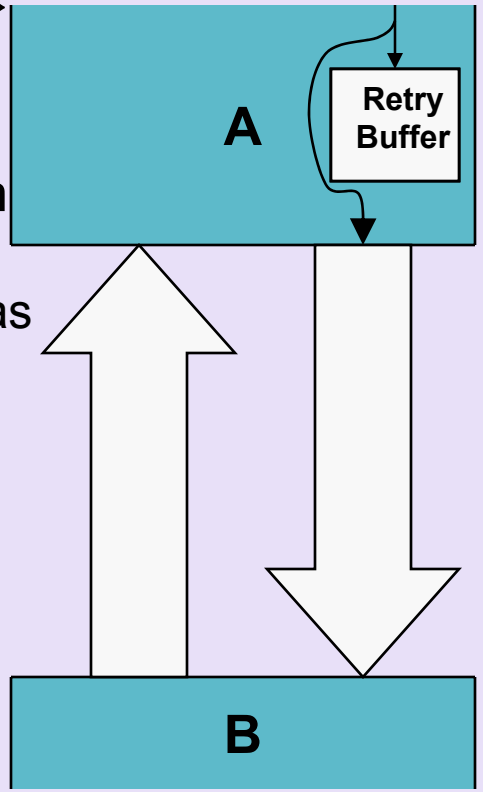
Example: REPLAY_TIMER Operation – Case 2

- Six TLPs are transmitted
  - ✓ REPLAY_TIMER started
- *Nak* acknowledges three of the transmitted TLPs...
  - ✓ REPLAY_TIMER resets and holds
- ... causes retransmission of other three
  - ✓ REPLAY_TIMER restarted
- ... eventually, the remaining TLPs are Ack'd
  - ✓ REPLAY_TIMER resets and holds – no outstanding unacknowledged TLPs



Retry Buffer Sizing

- Transmitter out of Retry Buffer space → Stop transmitting
  - ✓ All transmitted TLPs must be kept in Retry Buffer until Ack'd by the other component on the Link
    - Note that this includes Completions as well as Requests
  - ✓ Much outbound traffic → Retry Buffer optimization critical
- Ack policy and L0s also affect optimal Retry Buffer sizing



- To download a whitepaper on retry buffer size optimization, go to: www.mindshare.com/knowledge/?section=PCI%20Express|TM

Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Interrupts

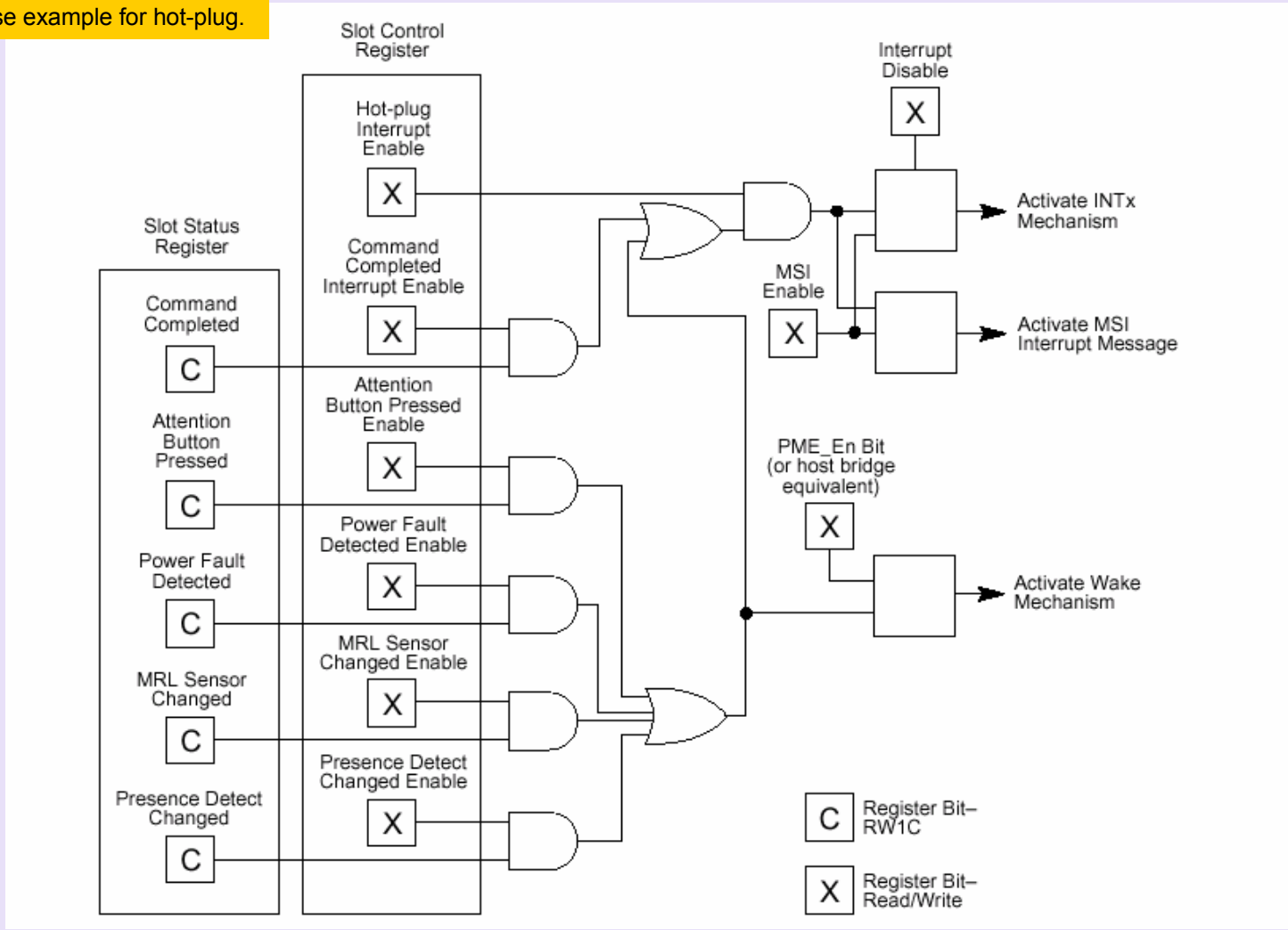
- PCI Express supports three interrupt mechanisms:
  - ✓ **INTx: level triggered**
  - ✓ **MSI: edge triggered**
  - ✓ **MSI-X: edge triggered; newly added via ECN**
- INTx, MSI, & MSI-X are mutually exclusive
  - ✓ **Enabling MSI or MSI-X disables INTx**
  - ✓ **SW is prohibited from enabling MSI & MSI-X concurrently**
  - ✓ **MSI/MSI-X are not controlled by (INTx) interrupt disable bit**
  - ✓ **MSI/MSI-X messages are memory write requests and can be disabled by clearing the BME (Bus Master Enable) bit**
  - ✓ **MSI/MSI-X/INTx interrupts can only be signaled in D0 state**

Interrupts

- Level-triggered interrupts can result in interrupt storms
  - ✓ **Especially high risk in virtual wire scenarios if de-assert messages not sent correctly**
  - ✓ **Be sure to:**
    - De-assert INTx in low-power states
    - De-assert INTx when source becomes masked
    - De-assert INTx when interrupts become disabled
  - ✓ **Asserts / De-asserts must be sent in pairs**
- Switches must synthesize de-asserts as necessary
  - ✓ **For example, if attached device is surprise removed**

Interrupts: Design Example

NOTE: Example intended to illustrate device interrupt logic. Do not use example for hot-plug.

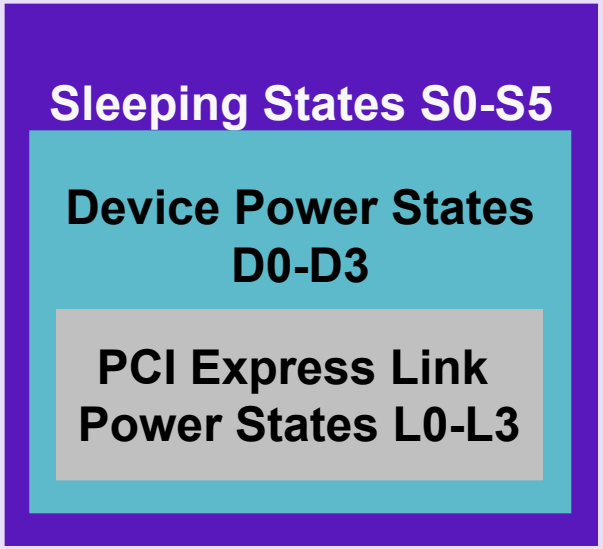


Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

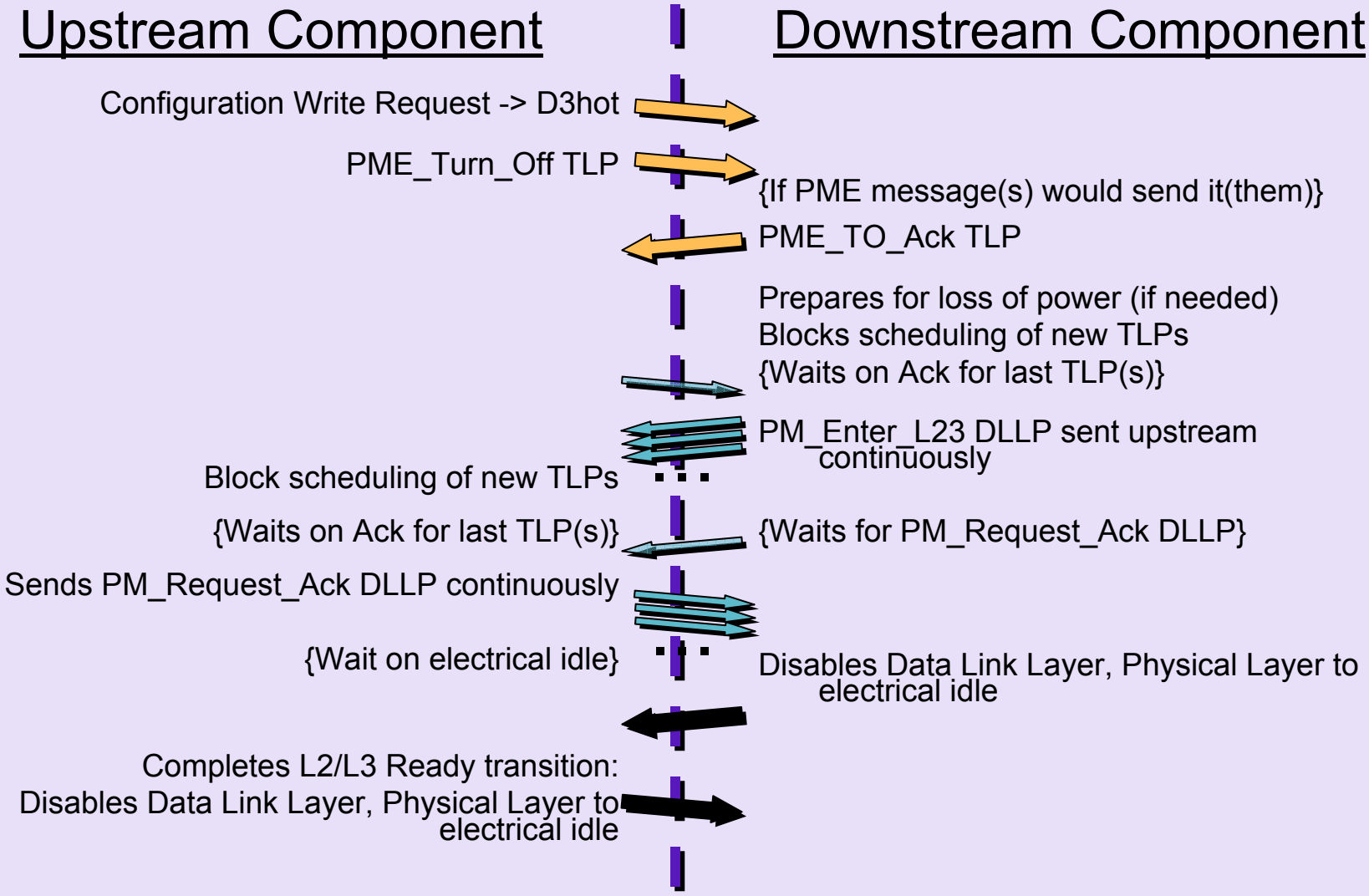
PCI Express Power Management

- Builds on PCI Power Management (PM)
  - ✓ Compatible with existing PCI PM software stacks
- System Level, Device Level and Link Level PM States
- Enhanced PM capabilities
  - ✓ Aggressive power reduction through Active State PM (L0s, L1)
  - ✓ Improved PME using in-band messaging
  - ✓ Improved definition and SW control of Vaux

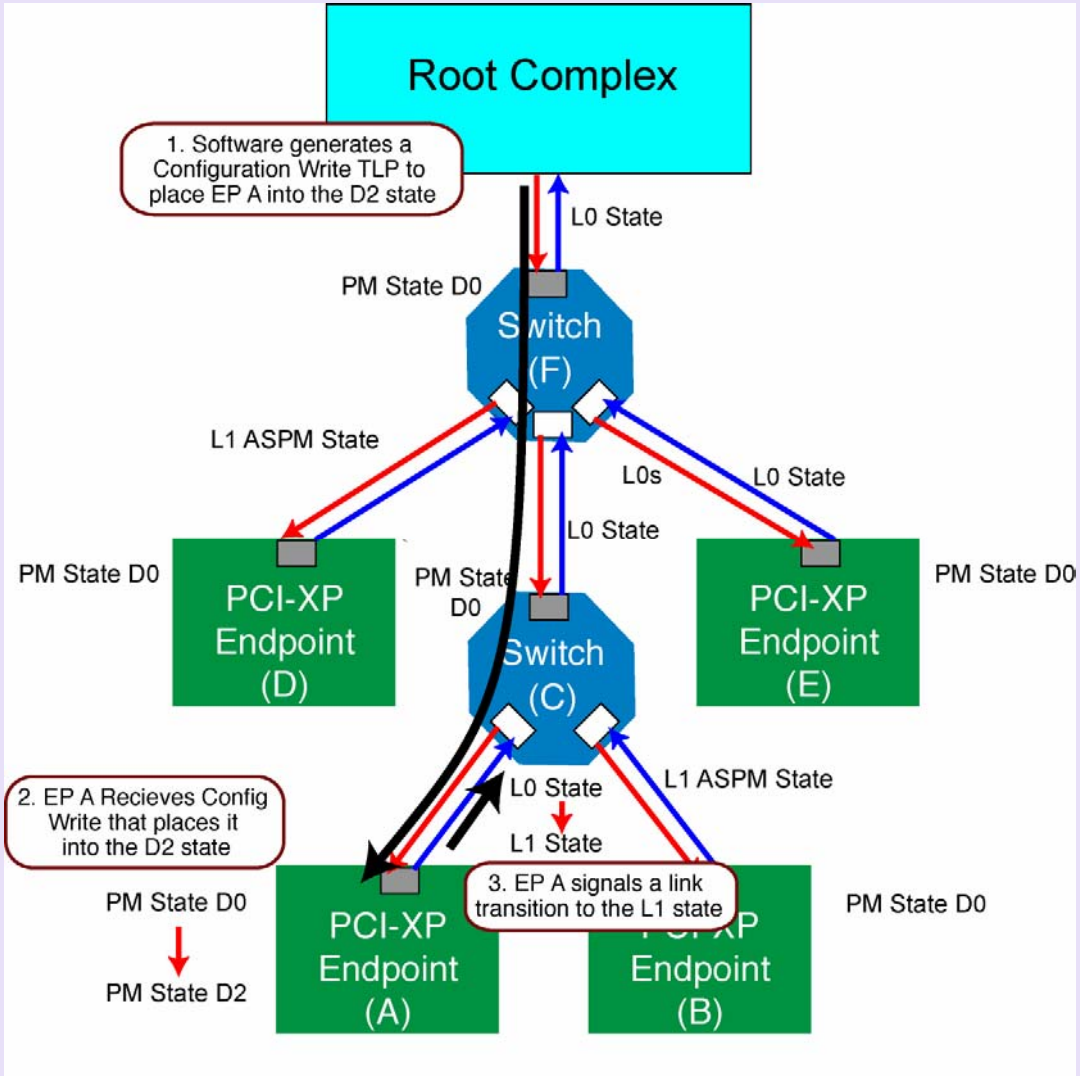


PCI Express Advances Platform PM While Preserving Software Investment

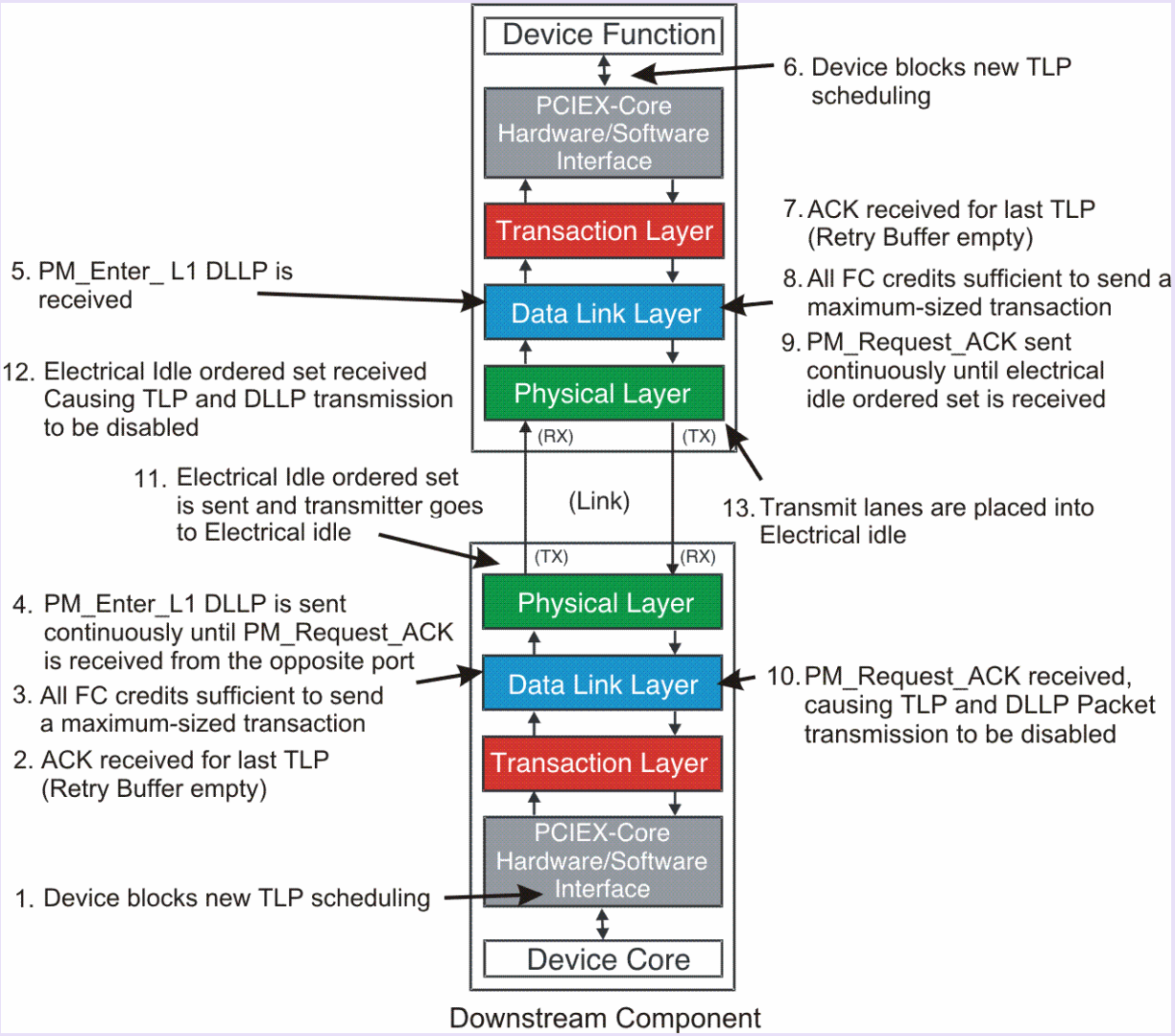
Example: D0 to D3hot L2/L3 Ladder Diagram



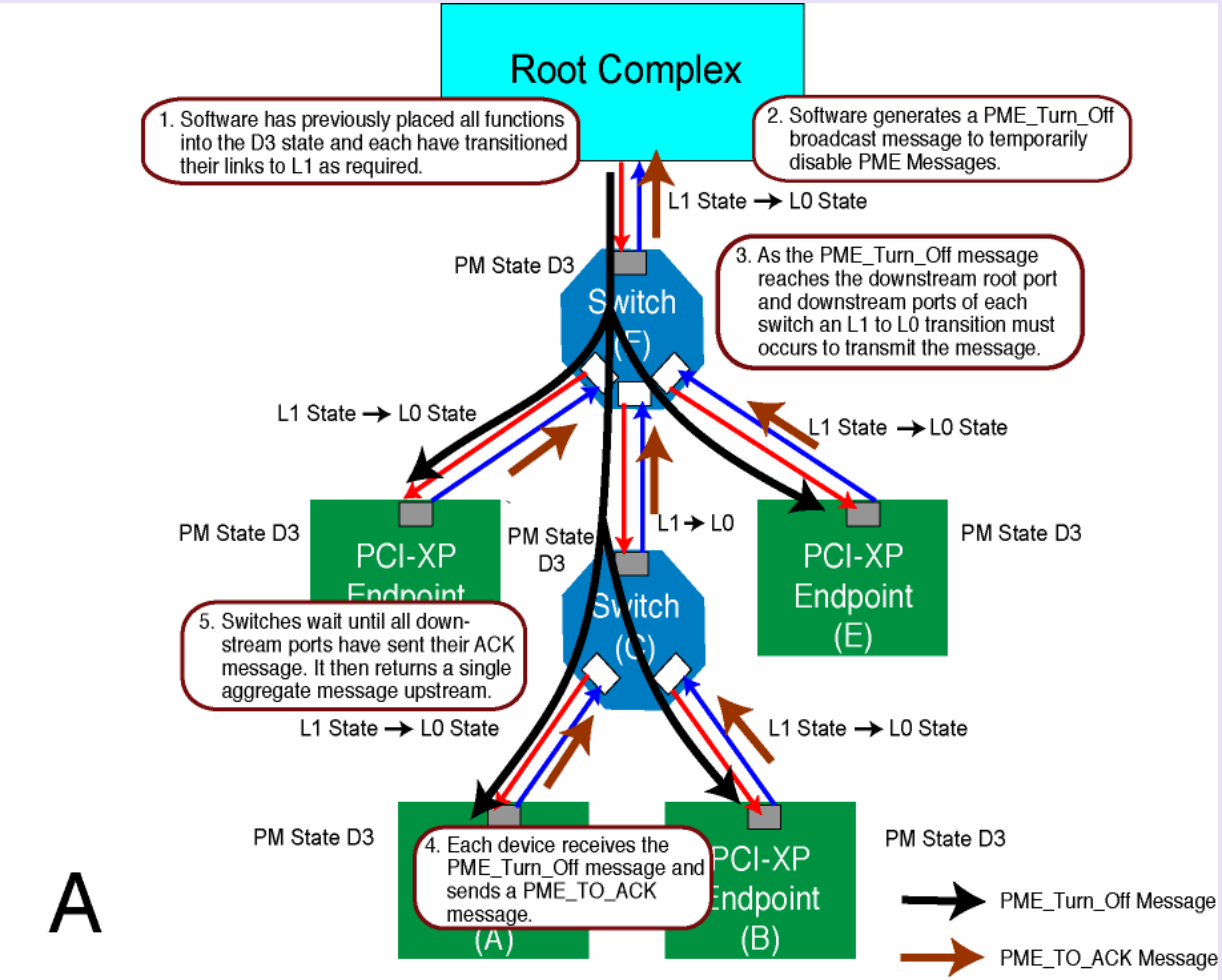
Software Puts Device/Link in D2/L1



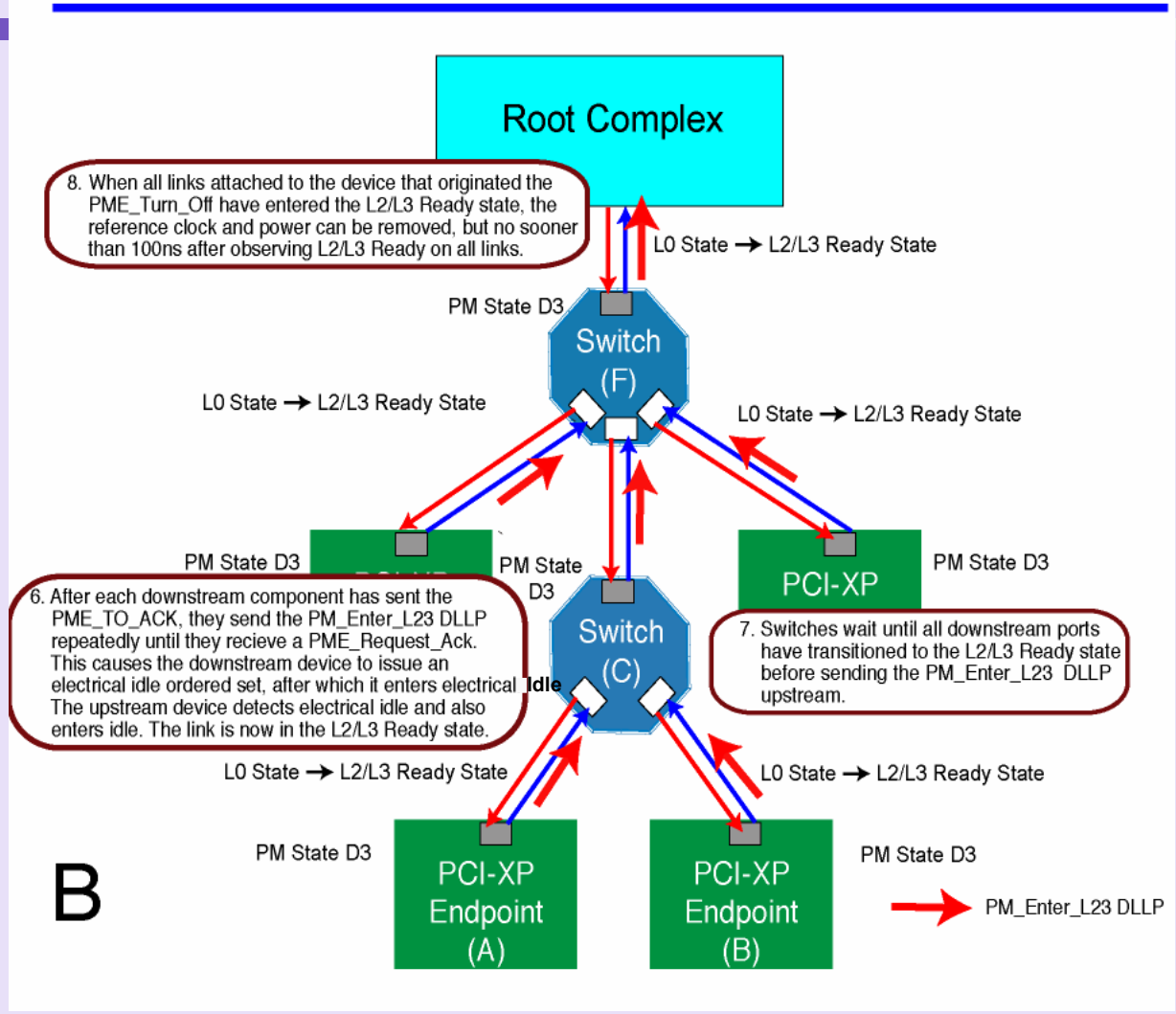
Transitioning Link From L0 to L1



Negotiation to Enter L2/L3 Ready



Negotiation to Enter L2/L3 Ready



Power Management

- Devices cannot source memory on IO requests when in a non-D0 state
  - ✓ **Remember to de-assert any pending INTx interrupts when transitioning out of D0 to a low-power state**
- PME_Turn_Off message can be received at any time
  - ✓ **Not just in non-D0 states**
  - ✓ **Simply indicates to device that power and clocks are going to be removed**
    - For example, can be received on system shutdown
- PME sequence is a two-stage process
  - ✓ **Two-stage process:**
    - Wakeup is separated from request for service
  - ✓ **Wakeup is responsible for getting clocks and power**
  - ✓ **Service is requested through PME message**

Support for Power Management Event (PME)

- PCI Express separates Wakeup mechanism from PME semantic
  - ✓ Wakeup using WAKE# or Beacon
  - ✓ PME Message sent following wakeup
- PME mechanism compatible with existing PME handling software
- PCI Express extends PME mechanism by including ID of requesting agent
  - ✓ Results in reduced overhead in PME processing
- Switches route PME messages from any downstream port to their upstream port
- Works with PCI Express-to-PCI bridges

Active State Power Management

- Software controlled power management (PCI-PM) provides mechanisms for intelligent PM
  - ✓ Important in conventional PCI, important in PCI Express
  - ✓ PCI Express is PCI-PM compatible
- Active State Power Management (ASPM) provides *additional* benefit
  - ✓ Low latency – minimum impact on performance
  - ✓ Finer granularity of control – more opportunity for power savings compared to software controlled PM
- Serial signaling technology consumes power when not sending data
  - ✓ “idle” = logical idle, not electrical idle
  - ✓ PCI Express uses ASPM to reduce power in logical idle
- ASPM is important to minimize power consumption with minimum performance impact

Power Management - Summary

- PCI Express Power Management is
  - ✓ Compatible with PCI-PM
  - ✓ Further improved with Active State Power Management
- Use PCI Express Power Management to reduce power consumption without impacting performance
- More PM information is available:
  - ✓ PM focused DevCon presentation
  - ✓ Online

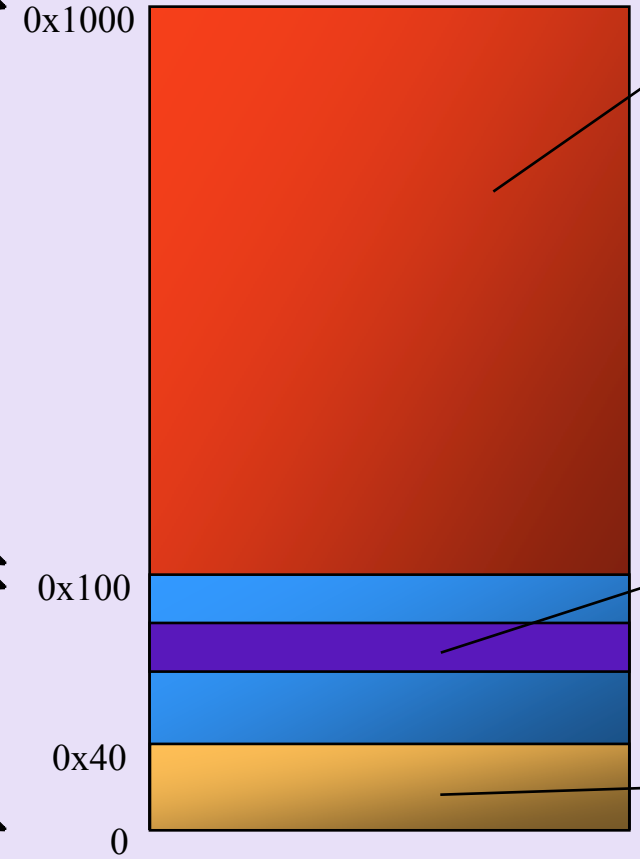
Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Configuration Space

PCI Express Extended Configuration Space ***(Not available on legacy OS)***

PCI-Compatible Configuration Space (Available on PC-compatible platforms with legacy OS via CF8/CFC)



Extended configuration space for PCI Express parameters capabilities (First extended capability begins at offset 0x100)

PCI Express Capability Structure

PCI 2.x Compatible Configuration Header

Configuration Space

- PCI-Compatible and Extended Config Space

- Do not rely on Extended Config space to be available in legacy environments
  - ✓ **Extended Configuration Space access may not be available in legacy OS scenarios**
  - ✓ **If access to Extended Configuration Space elements is really needed, design for aliasing elements through a BAR or PCI-Compatible Configuration Space region**

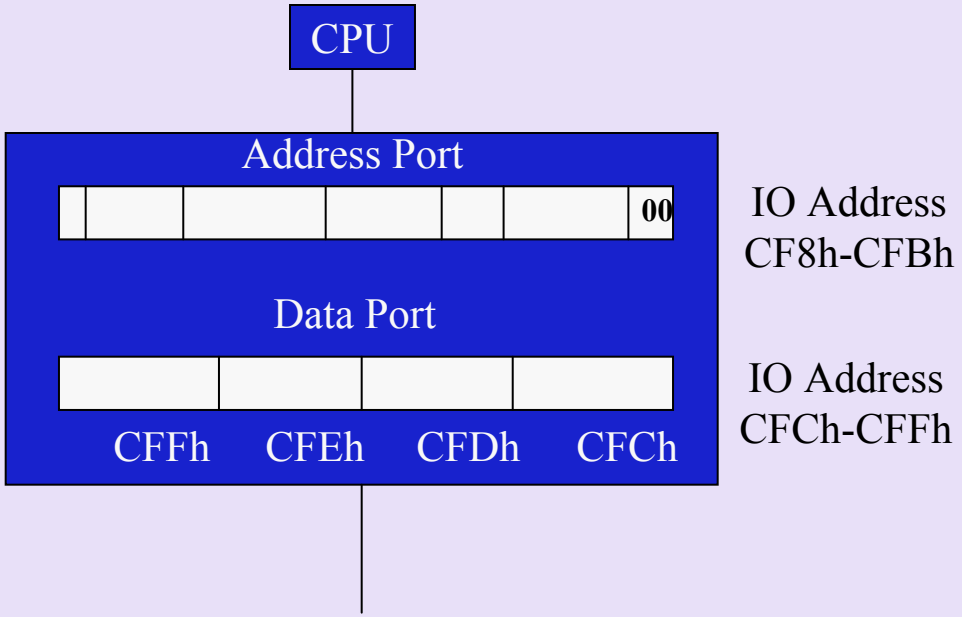
- Note that registers critical to device functionality are all located in PCI-Compatible Configuration Space
  - ✓ **PCI Express Capability Structure located below 256 bytes**

Configuration Space

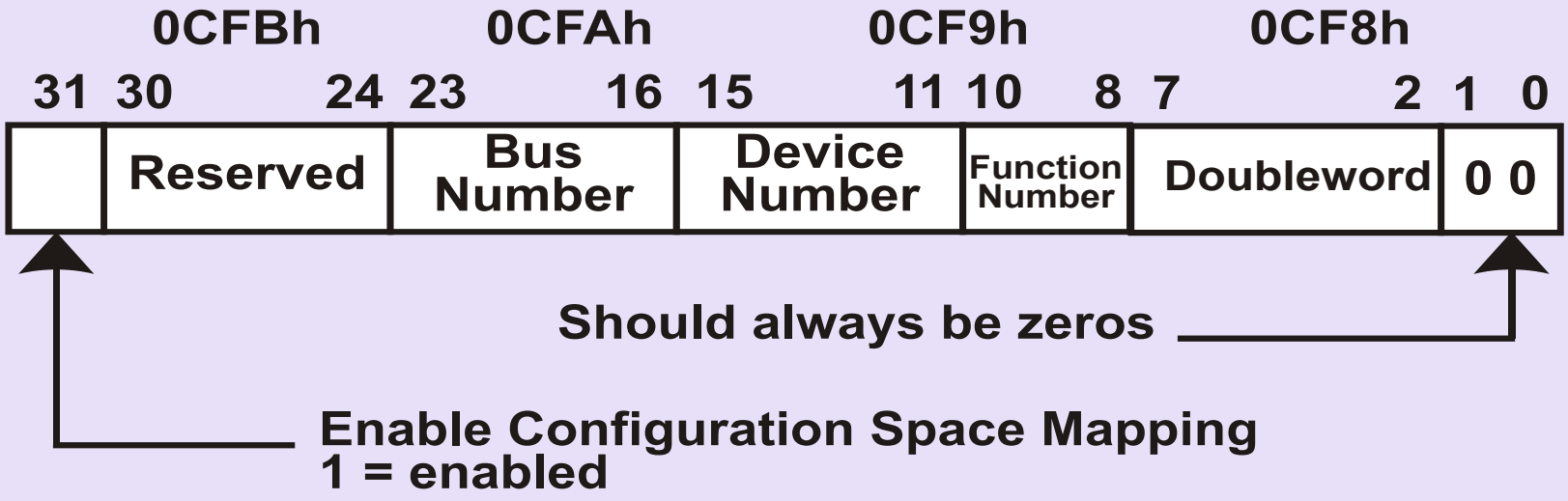
- Spec-defined capabilities are for system software use
  - ✓ **Device-specific software should rely on OS services for access to these registers**
  - ✓ **Writes to spec-defined capabilities are not recommended for software other than system software (firmware / OS)**
    - Improper use can cause upgrade problems during OS migration; OS has permission to trap writes to header
- Recommend that device-specific registers be located in BAR regions

PCI Compatible Configuration Mechanism

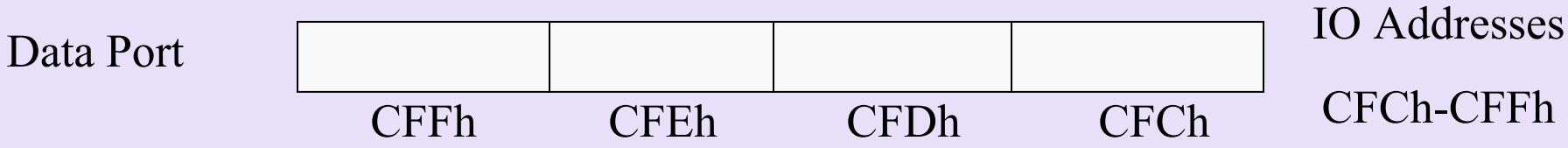
- Configuration Mechanism #1
  - ✓ I/O reads and writes from the processor are converted to configuration reads and writes by the host bridge
  - ✓ I/O addresses 0CF8 and 0CFC are used



Configuration Address Port @ CF8h



Configuration Data Register @ CFCh



Enhanced PCI Express Configuration Mechanism

- 28 bit address encoded on system address bus as shown
- Bits A[63:28] are platform specific
  - ✓ Must be communicated to root complex and OS by BIOS in an implementation-specific method
  - ✓ Considered the configuration “Base Address”

Memory Address	Configuration Space
A[20+(n*-1):20]	Bus Number [7:0]
A[19:15]	Device [4:0]
A[14:12]	Function [2:0]
A[11:8]	Extended Register [3:0]
A[7:0]	Register [7:0]

***n is between 1 and 8**

Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

New With 1.1

- Hot Plug
  - ✓ Hot-Plug messages not supported
  - ✓ Implementation simplified for Endpoints
  - ✓ Most changes in Sections 2.2.8.7 & 6.7
- Error Reporting
  - ✓ Improves ability of system software to handle several cases of Completer-detected non-fatal errors via “Advisory Non-Fatal Error Reporting” facility
  - ✓ Key benefit: UR reporting need not be disabled during device enumeration
  - ✓ See DevCon Presentation “PCIe Error Reporting ECN” for more detail
  - ✓ Most changes in Section 6.2, 7.8.3 & 7.8.4

New With 1.1 - Continued

- PME_Turn_Off
  - ✓ All Endpoints must respond to the PME_Turn_Off handshake request in all Device power states
  - ✓ Changes in Sections 5.2 and 5.3
- Reset Limit Adjustment
  - ✓ Time limit from end of Fundamental Reset to entry to LTSSM Detect state changed from 80ms to 20ms
  - ✓ Changes in Section 6.6
- Go to: www.mindshare.com/knowledge/ to download a whitepaper that documents Spec 1.1 changes

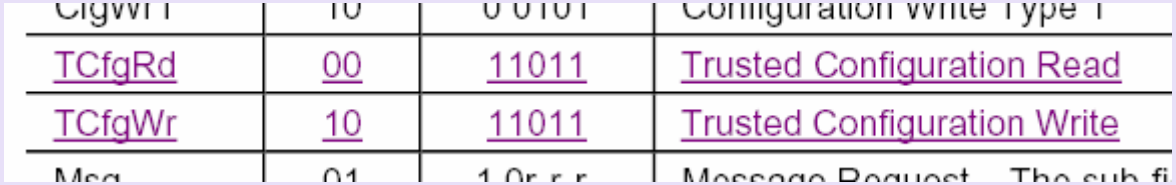
Rev. 1.1 Base Spec Updates/ECRs

PCIe Rev 1.1 Errata – Class Code Conflict

- Class Code conflict for RC Event Collector
- If you implement RC Event Collector,
  - ✓ Consider flexible mechanisms for setting class code
  - ✓ Wait for errata correcting conflict
- Resolution details still WIP –
  Look for errata to be published

Trusted Configuration – New Optional Capability

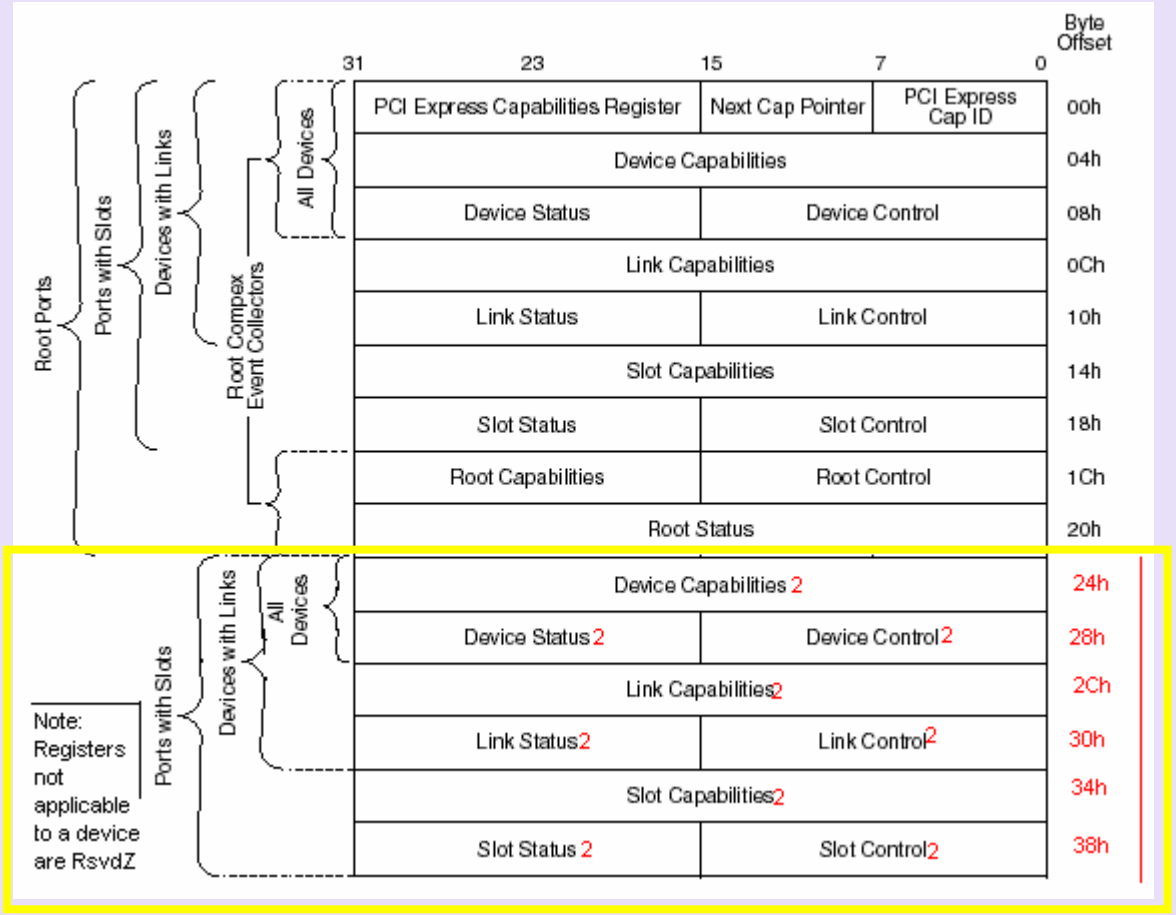
- *Status: Draft ECN – these changes will work into the next spec, which is 2.0*
- Optional new address space - Trusted Configuration Space (TCS)
  - ✓ Software in Trusted Software Environment may issue Trusted Configuration Requests (new TLPs – see below)
  - ✓ TCS access provided through Trusted Configuration Access Mechanism (TCAM)
- Devices "know" TCS requests are "trustworthy"
- Software "knows" device is "trustworthy"
- See DevCon presentation "PCIe Trusted Config Space & Link Speed Controls"



CfgWr1	10	0 0101	Configuration Write Type 1
TCfgRd	00	11011	Trusted Configuration Read
TCfgWr	10	11011	Trusted Configuration Write
Msg	01	1 0r r r	Message Request – The sub-fi

PCIe Capability Structure Expansion

- *Status: ECR*
- Adds space for new capabilities
  - ✓ Completion Timeout – see next foil
- Implementation required only as needed
- Adds clear requirements for unimplemented register space



Completion Timeout Control

- *Status: ECR*
- Adds capability to disable Completion Timeout
  - ✓ "Required" for all devices implementing ECN
  - ✓ In future, simply required
- Adds *optional* capability for system firmware/software to set Completion Timeout time value
  - ✓ Devices indicate supported ranges from the four bins defined: 50us to 10ms, 10ms to 250ms, 250ms to 4s, 4s to 64s
  - ✓ Two selectable ranges for each bin
    - Example: in 10ms to 250ms range – 16ms to 55ms, 65ms to 210ms
  - ✓ Ranges are separated by invalid value ranges to ensure non-overlapping timeouts

Bandwidth Change Notification Mechanism

- *Status: ECR*

- Adds mechanism to signal link has changed width or speed
  - ✓ Uses interrupt & status bit
  - ✓ Covers BW change due to hardware autonomous action or software direction

Agenda

- PCIe Features
- Protocol Overview
- Flow Control, Buffering
- Virtual Channels
- Link Data Integrity & Retry Buffer
- Interrupts
- Power Management
- Configuration Space
- Review: What's New with 1.1
- Updates to PCIe™ Revision 1.1 Base Spec
  - ✓ Errata
  - ✓ New capabilities – ECNs in progress
- Summary / Call to Action

Summary

- PCI Express advances overall platform capabilities while preserving PCI architecture and software investments
- Layered approach and scaleable features provide a foundation for technology stability
- New capabilities enable important emerging applications

Summary

- **Legacy software compatibility and transition strategy to PCI Express-aware software environment must be considered at all times**
- **MSI/MSI-X have major advantages over INTx, but correct interrupt handling to avoid loss of edge-triggered interrupts is not intuitive**
- **Implement MSI-X instead of MSI when multiple vectors are merited**
- **Firmware plays key role in enabling new PCI Express features**
- **Base all new components on Rev. 1.1**
  - ✓ Many improvements over the 1.0a
- **Comprehend errata and additions**
  - ✓ Errata comprehension required for compliance
  - ✓ ECN implementation permitted but not required
- **Track Gen 2 development work to be ready for future speed increase**
  - ✓ 1.1 Components not directly affected

Call to Action

- Comprehend PCI Express technology in your product roadmaps
  - ✓ Invest in PCI Express building blocks and infrastructure
- Engage with PCI-SIG Serial Technical Communications Workgroup
  - ✓ Evangelize PCI Express across computing and communications industries
  - ✓ Take advantage of technical enabling
  - ✓ Participate in compliance and interoperability forums

Thank you for attending the PCI-SIG Technology Seminar 2005.

For more information please go to

www.pcisig.com

and

www.mindshare.com

PCI Express® 1.1 Link, Transaction and Configuration Protocols

Mike Jackson

Sr. Staff Architect, MindShare, Inc.