



# PCI Express® Basics

**Richard Solomon**  
**LSI Corporation**



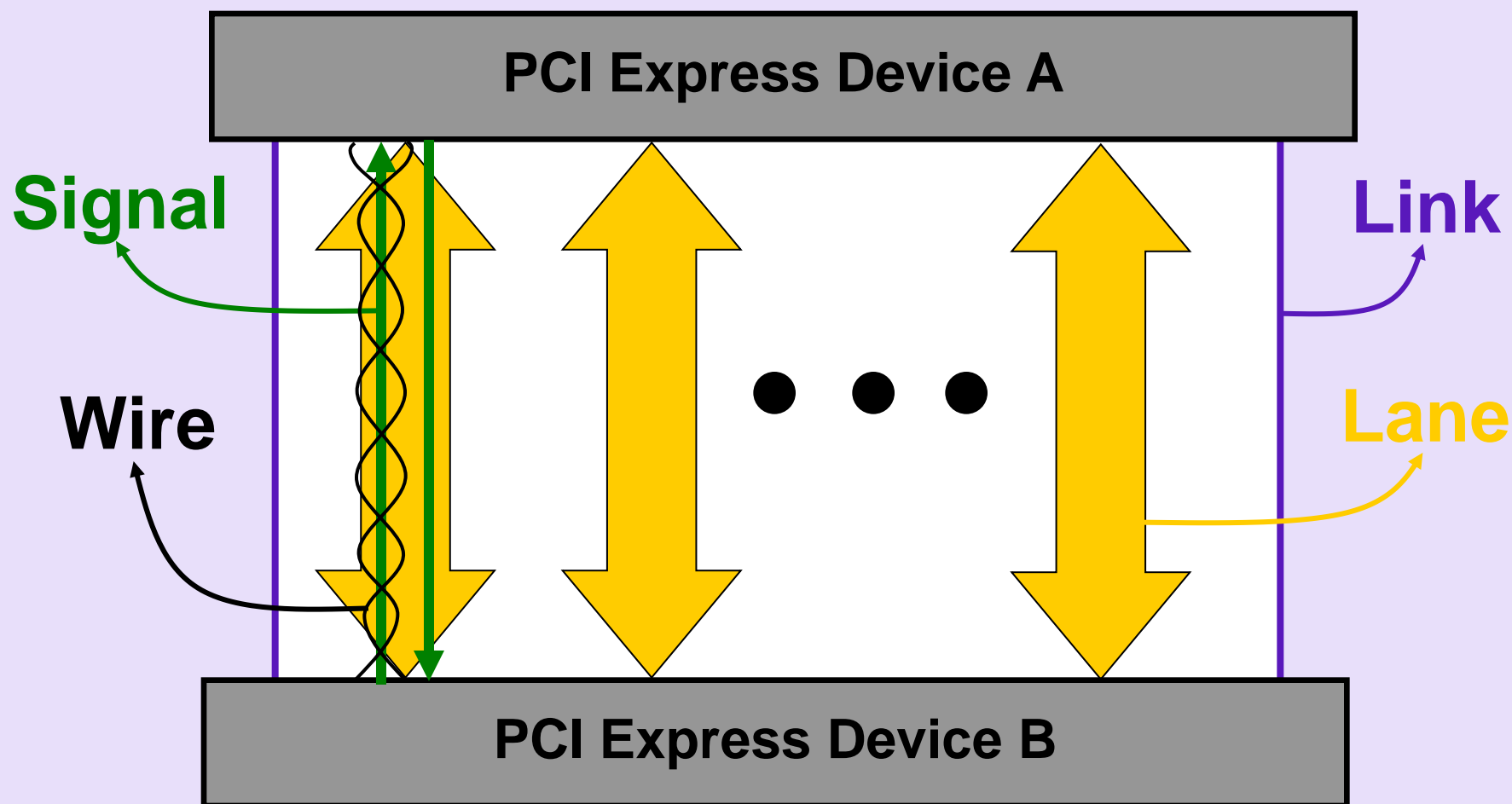
# Acknowledgements

- I would like to acknowledge the contributions of Ravi Budruk, 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 Throughput

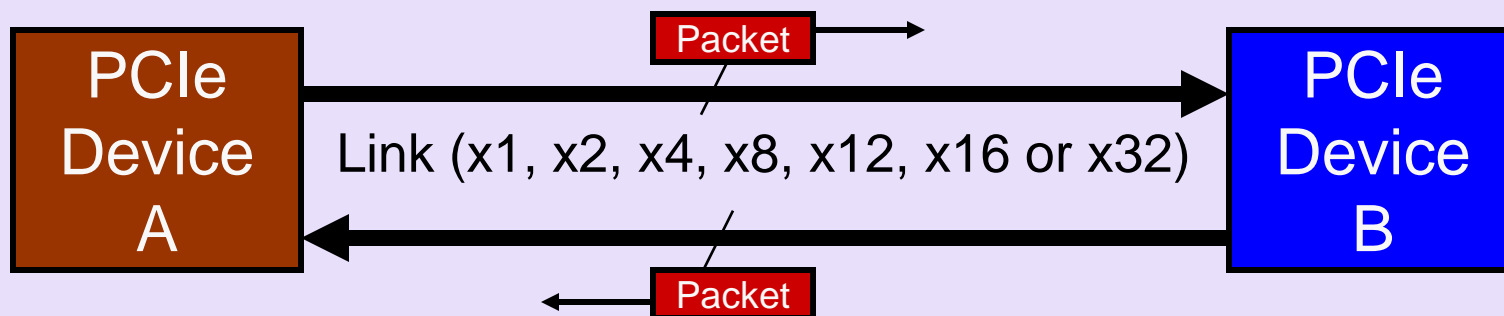
|                           | Link Width |    |    |    |     |     |     |
|---------------------------|------------|----|----|----|-----|-----|-----|
|                           | x1         | x2 | x4 | x8 | x12 | x16 | x32 |
| <b>PCIe 1.x BW (GB/s)</b> | 0.5        | 1  | 2  | 4  | 6   | 8   | 16  |
| <b>PCIe 2.x BW (GB/s)</b> | 1          | 2  | 4  | 8  | 12  | 16  | 32  |
| <b>PCIe 3.0 BW (GB/s)</b> | 2          | 4  | 8  | 16 | 24  | 32  | 64  |

## Derivation of these numbers:

- 2.5 GT/s (PCIe 1.x), 5.0 GT/s (PCIe 2.x), or 8GT/s (PCIe 3.0) signaling in each direction
- 20% overhead due to 8b/10b encoding in 1.x and 2.x
- “Aggregate bandwidth”, implying traffic in both directions

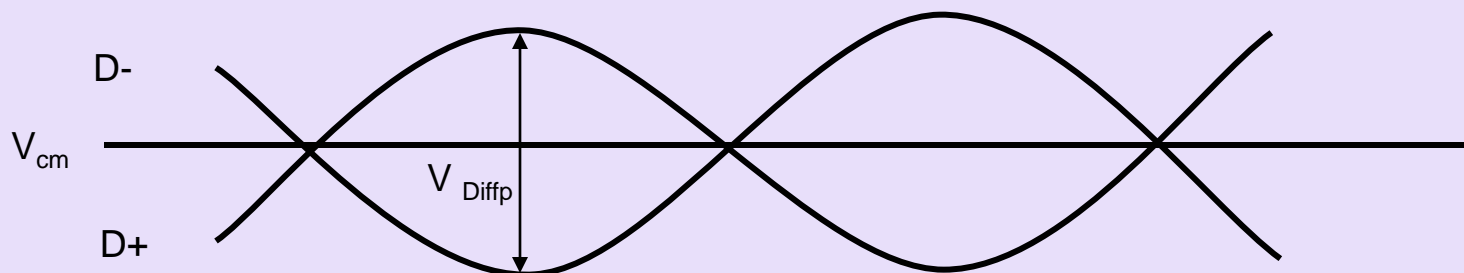
# PCI Express Features

- Point-to-point connection
- Serial bus means fewer pins
- Scalable: x1, x2, x4, x8, x12, x16, x32
- Dual Simplex connection
- 2.5, 5.0 and 8.0 GT/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

# Additional Features

- Switches used to interconnect 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

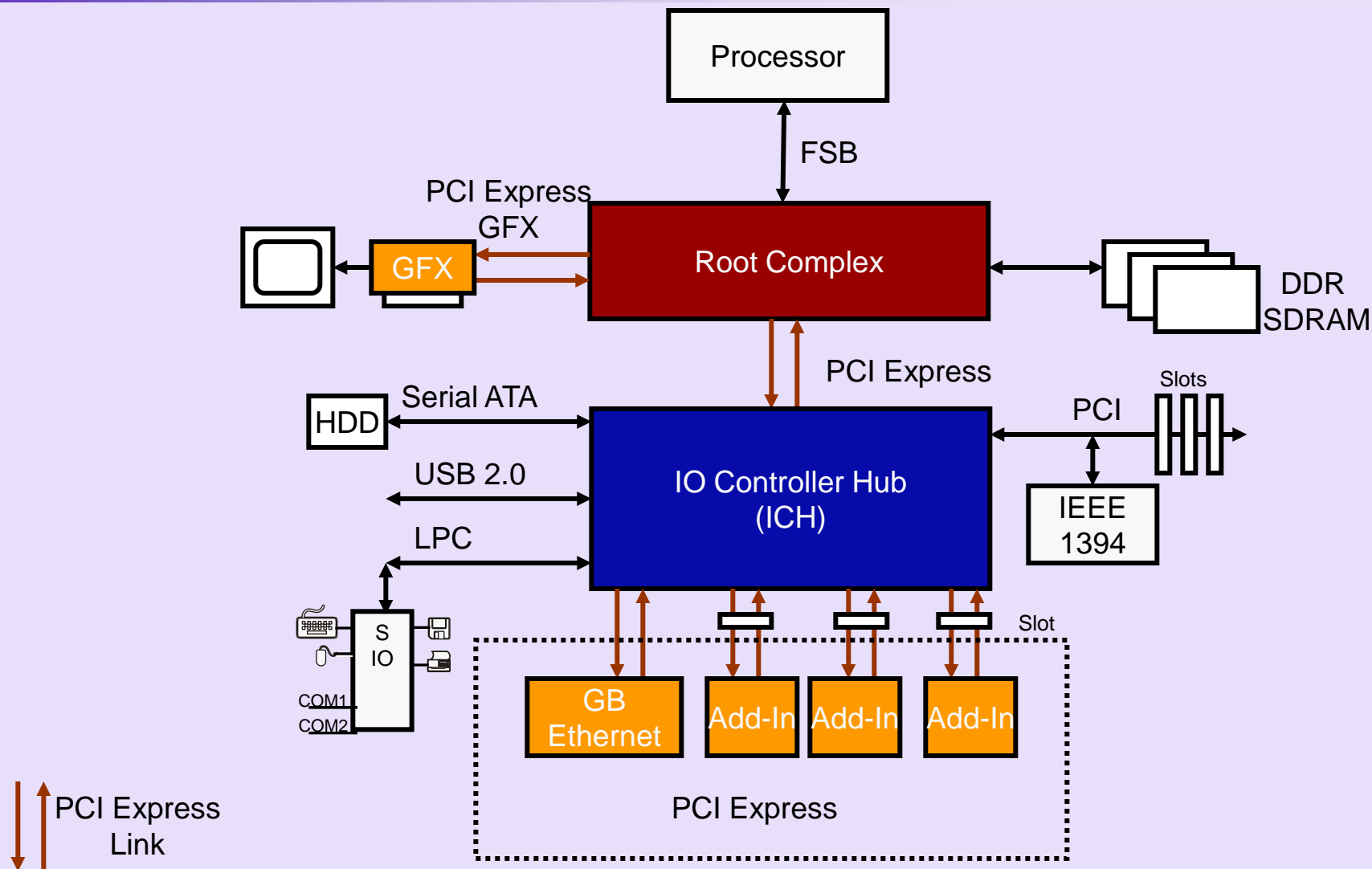
- Flow Control
  - ✓ No retry as in PCI
- MSI style interrupt handling
  - ✓ Also supports legacy PCI interrupt handling in-band
- Advanced power management
  - ✓ Active State PM
  - ✓ PCI compatible PM

# 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



# PCI Express System



# Transaction Types, Address Spaces

- 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

| Description   | Abbreviated 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           |

# Three Methods For Packet Routing

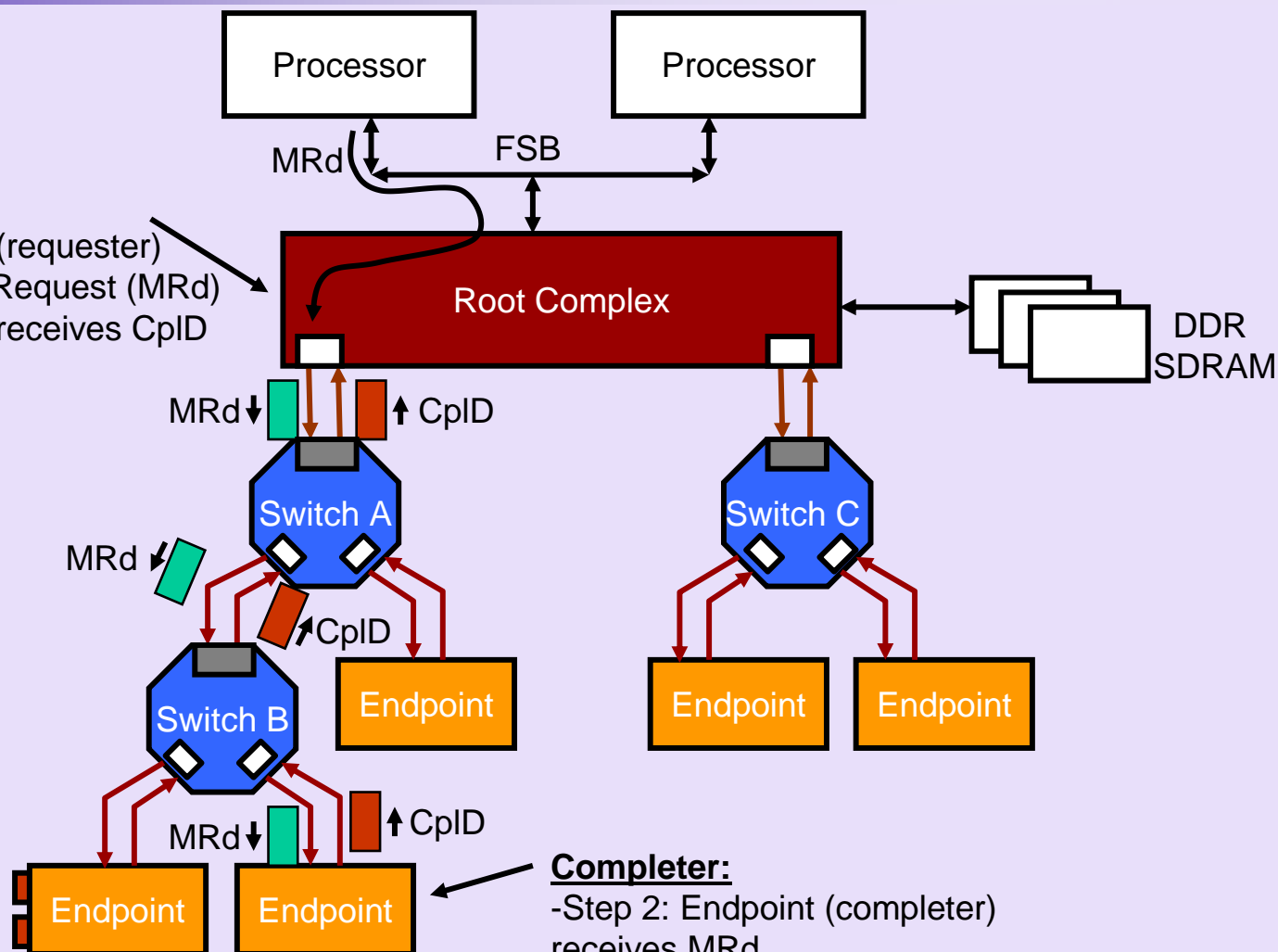
- Each request or completion header is tagged as to its *type*, and each of the packet types is routed based on one of three schemes:
  - ✓ Address Routing
  - ✓ ID Routing
  - ✓ Implicit Routing
- Memory and IO requests use address routing.
- Completions and Configuration cycles use ID routing.
- Message 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



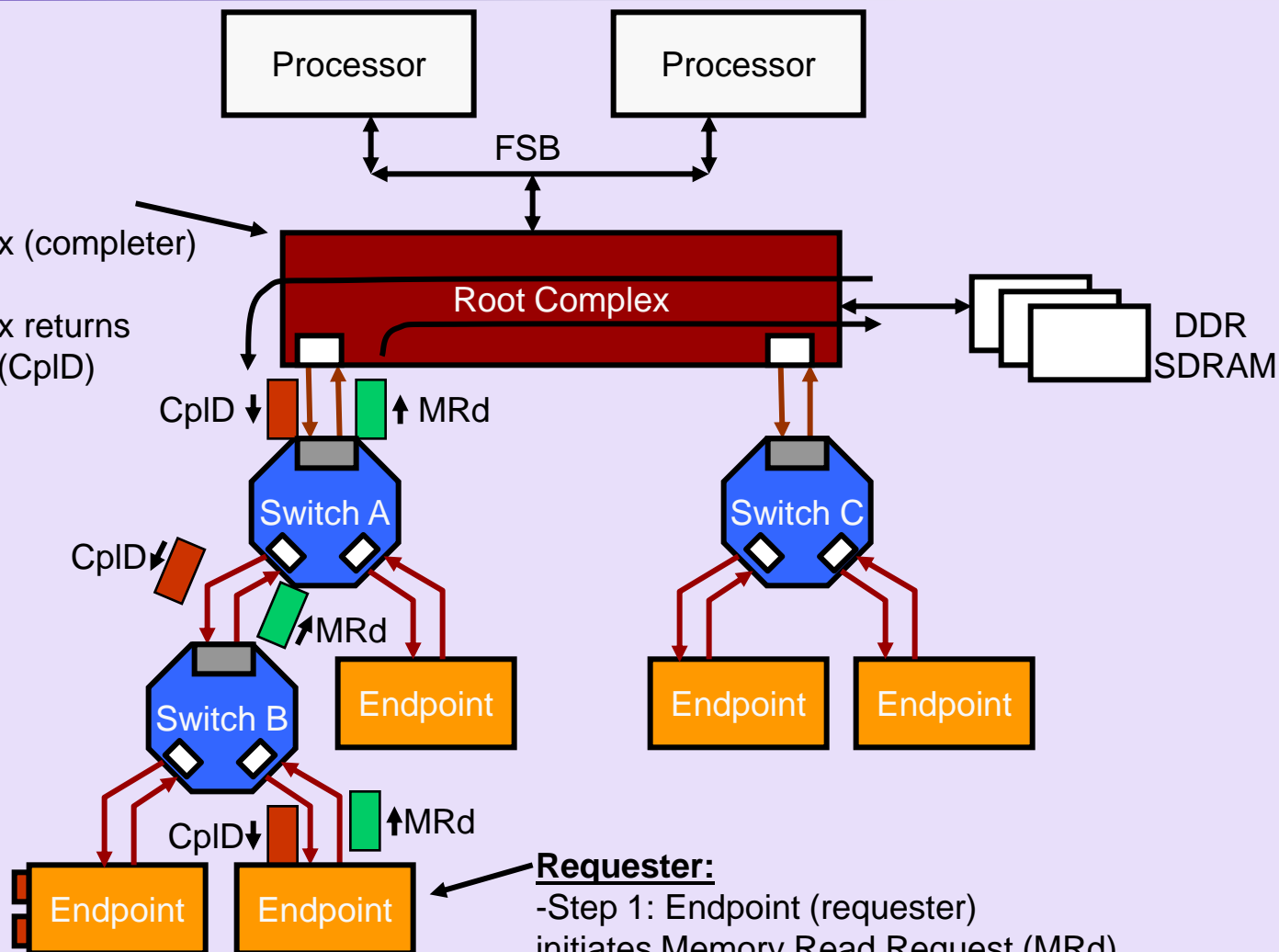
## Completer:

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

# DMA Transaction

## Completer:

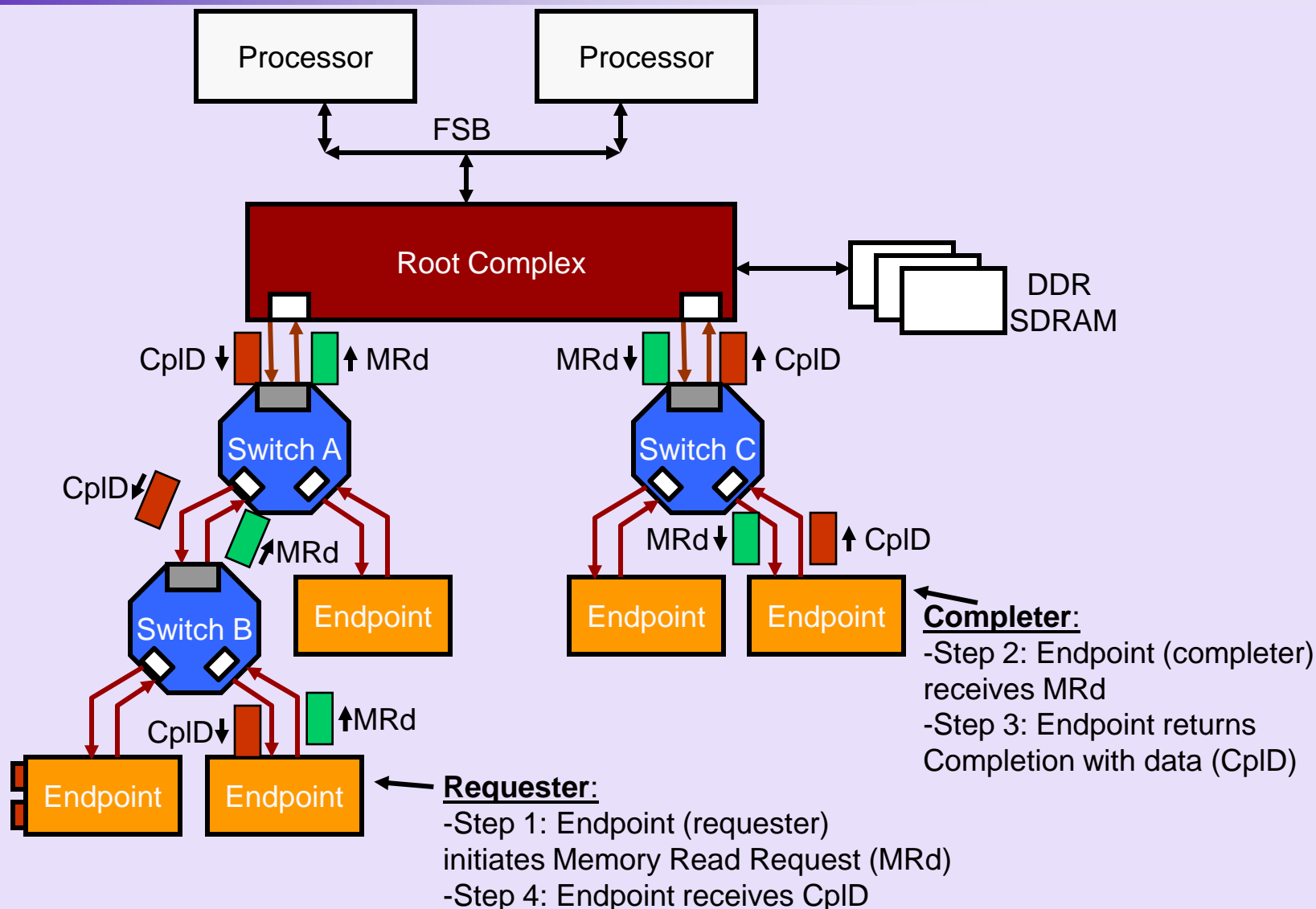
- Step 2: Root Complex (completer) receives MRd
- Step 3: Root Complex returns Completion with data (CpID)



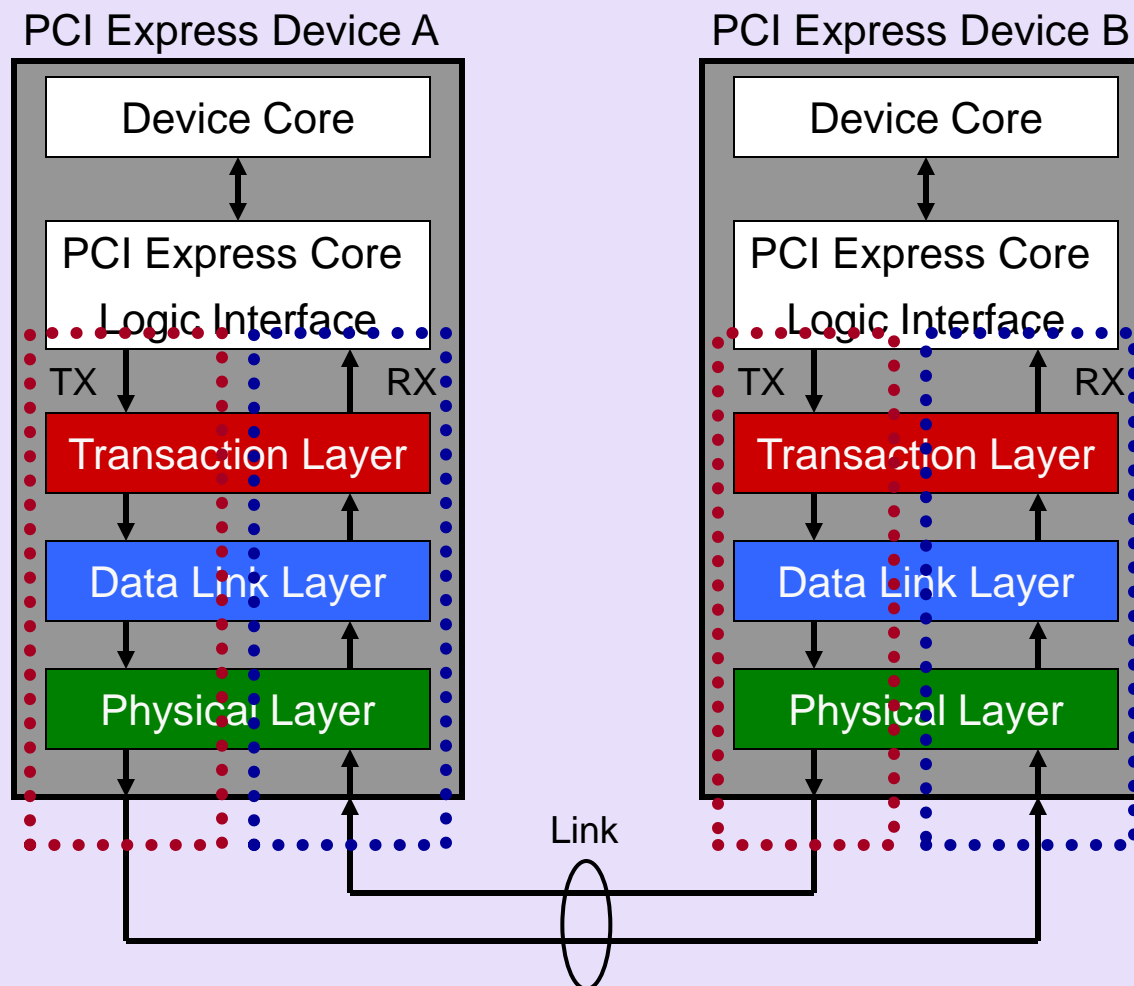
## Requester:

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

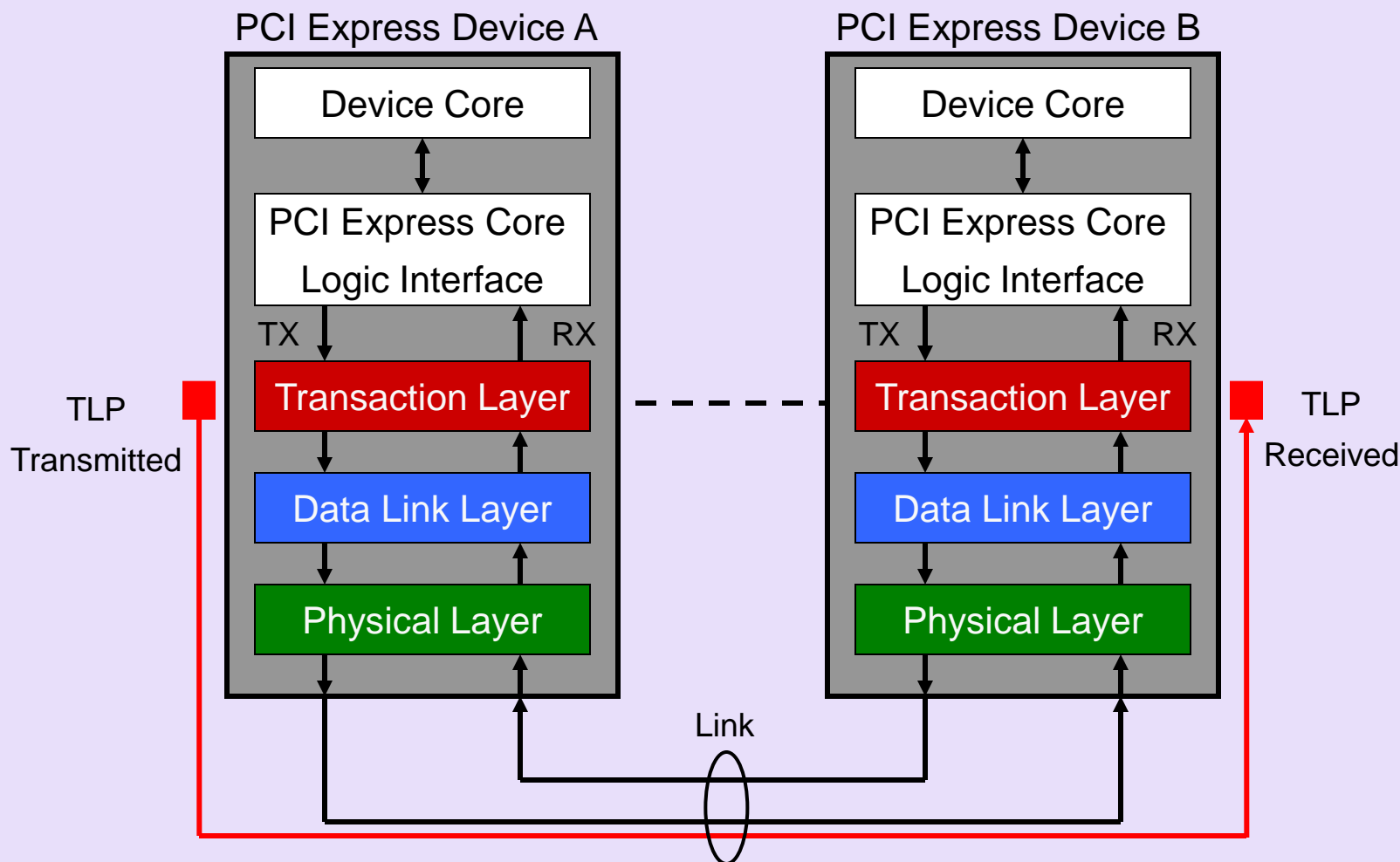
# Peer-to-Peer Transaction



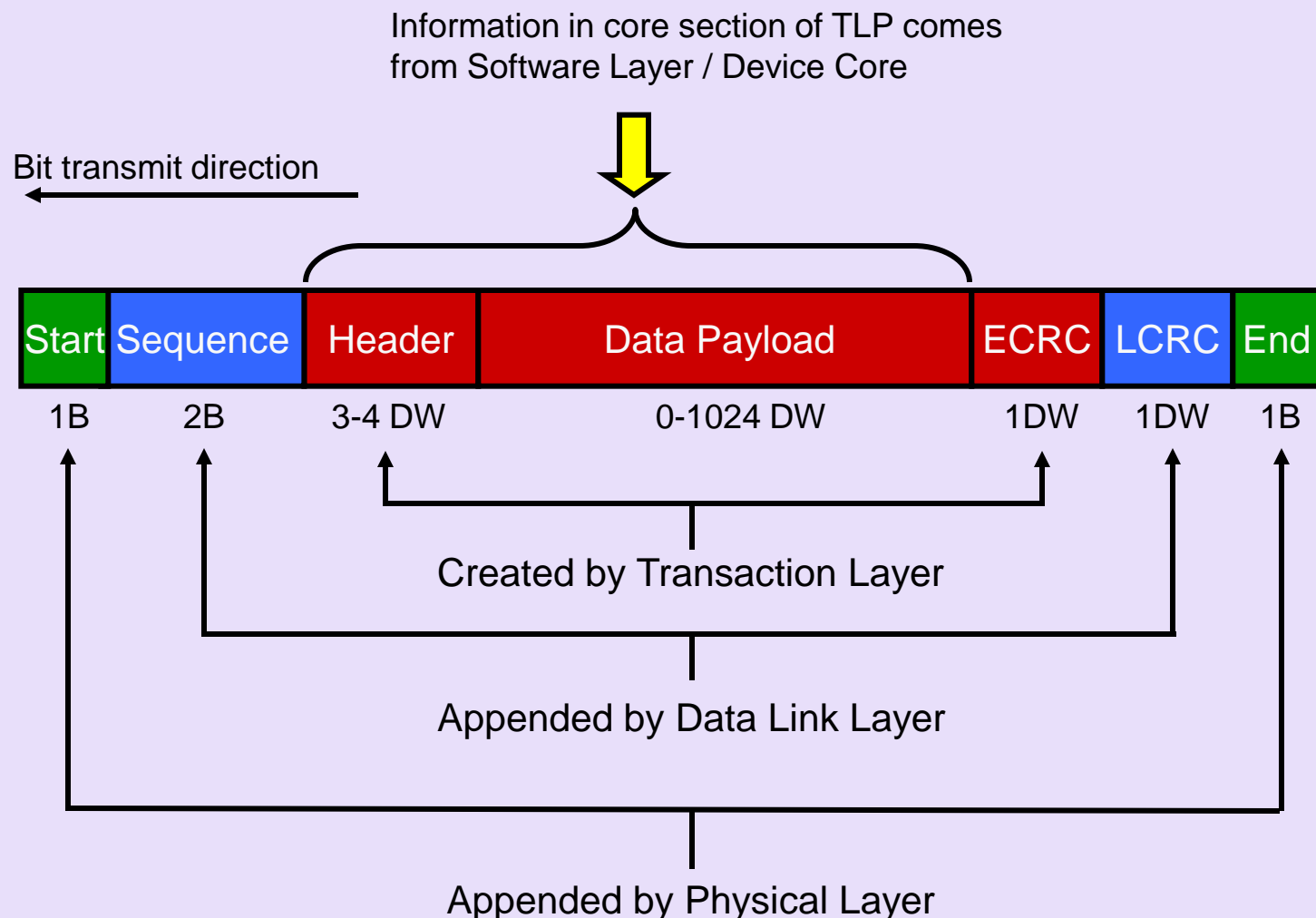
# PCI Express Device Layers



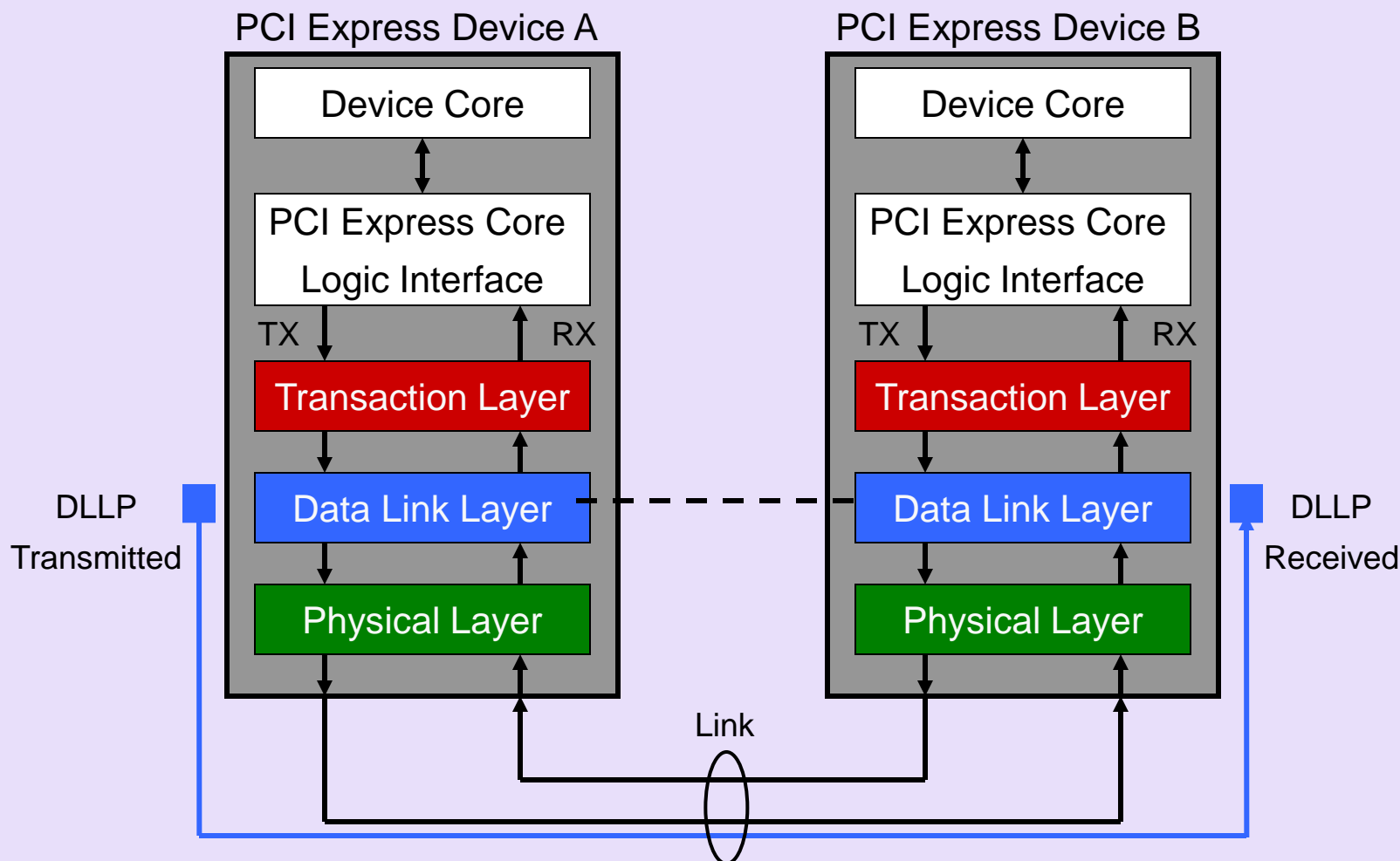
# TLP Origin and Destination



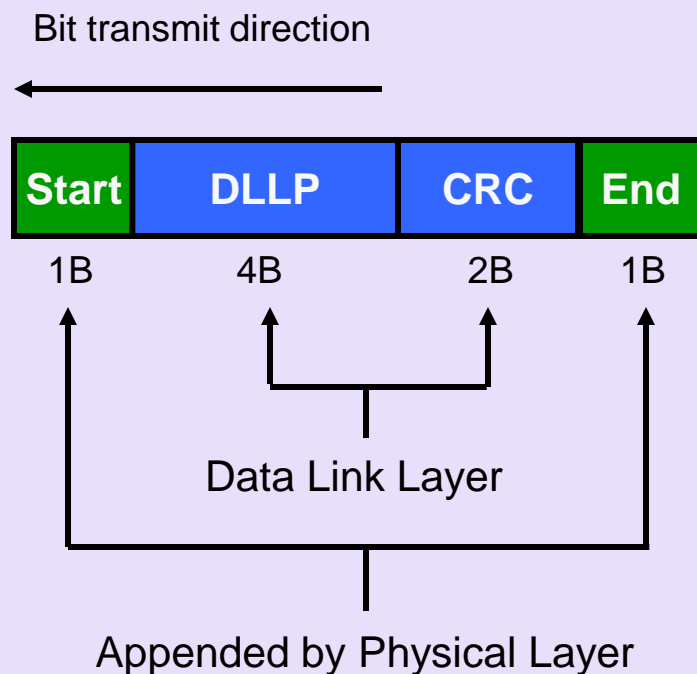
# TLP Structure



# DLLP Origin and Destination



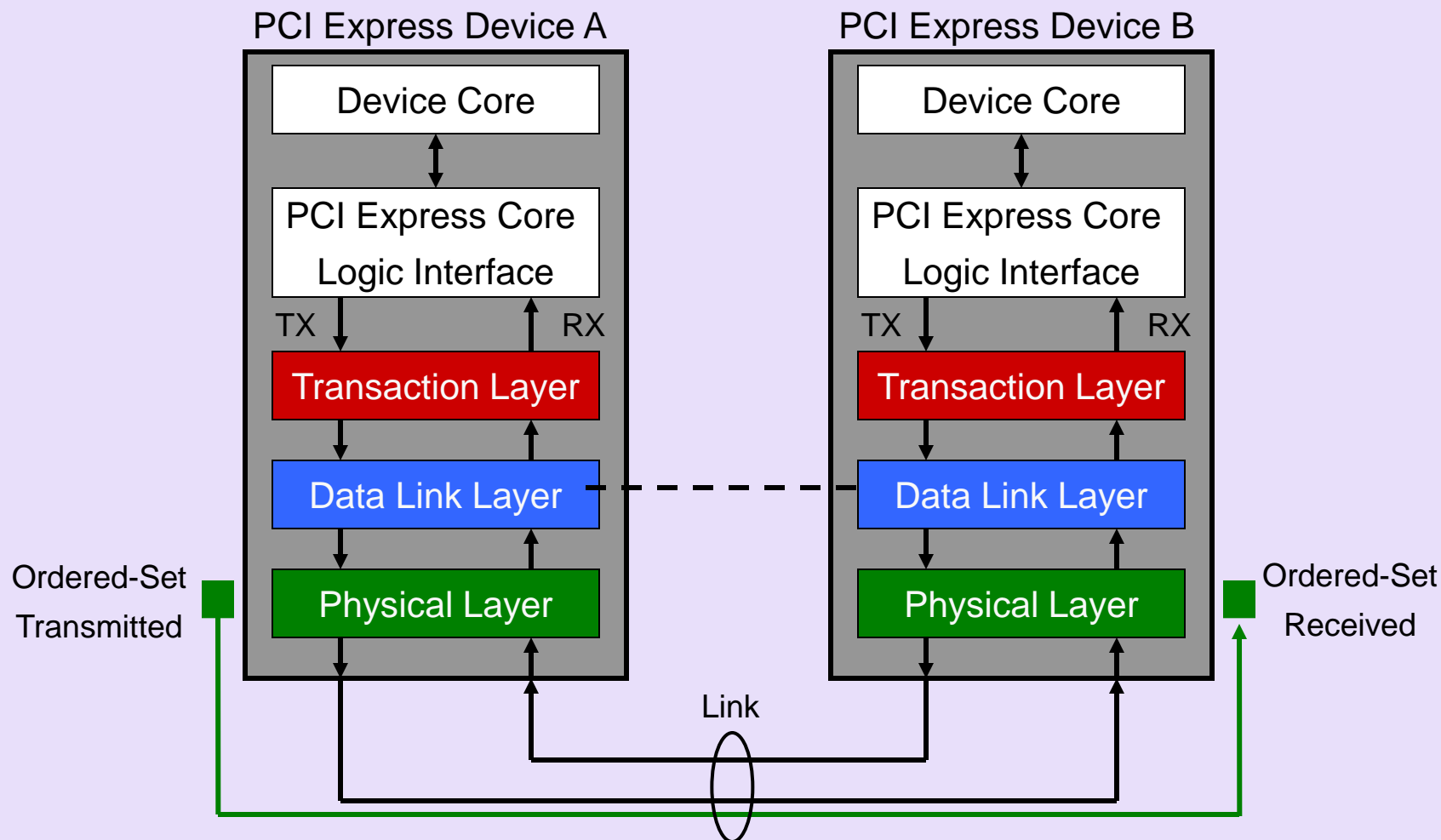
# DLLP Structure



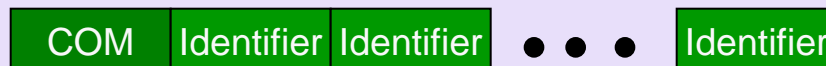
- ACK / NAK Packets
- Flow Control Packets
- Power Management Packets
- Vendor Defined Packets



# Ordered-Set Origin and Destination

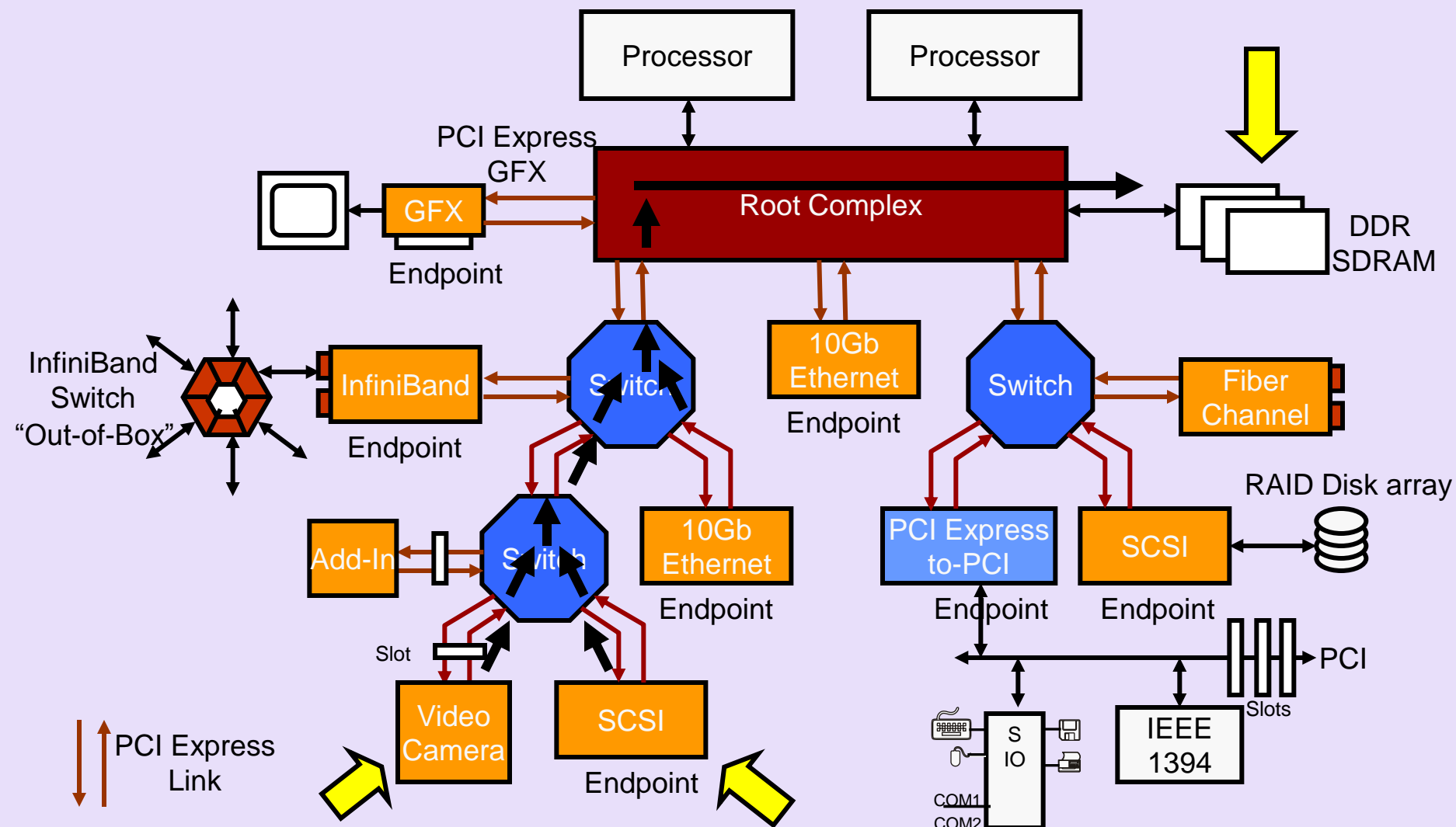


# Ordered-Set Structure



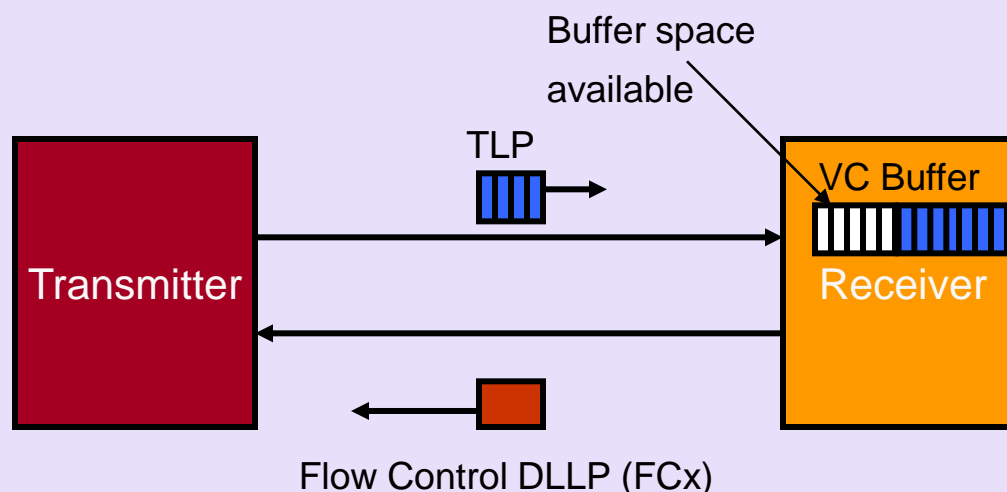
- Training Sequence One (TS1)
  - ✓ 16 character set: 1 COM, 15 TS1 data characters
- Training Sequence Two (TS2)
  - ✓ 16 character set: 1 COM, 15 TS2 data characters
- SKIP
  - ✓ 4 character set: 1 COM followed by 3 SKP identifiers
- Fast Training Sequence (FTS)
  - ✓ 4 characters: 1 COM followed by 3 FTS identifiers
- Electrical Idle (IDLE)
  - ✓ 4 characters: 1 COM followed by 3 IDL identifiers
- Electrical Idle Exit (EIEOS) (new to 2.0 spec)
  - ✓ 16 characters

# Quality of Service



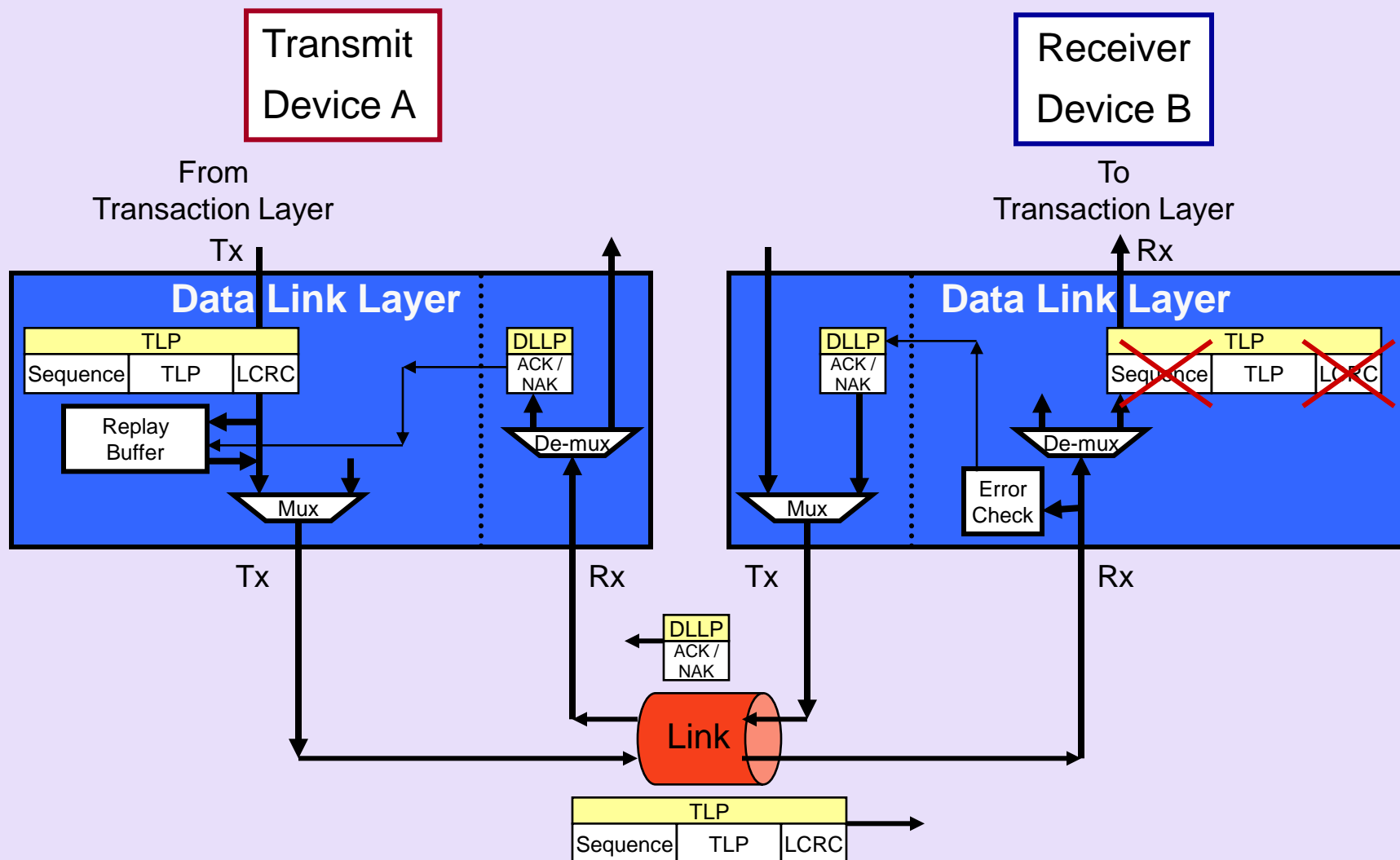
# PCI Express Flow Control

- Credit-based *flow control* is point-to-point based, not end-to-end

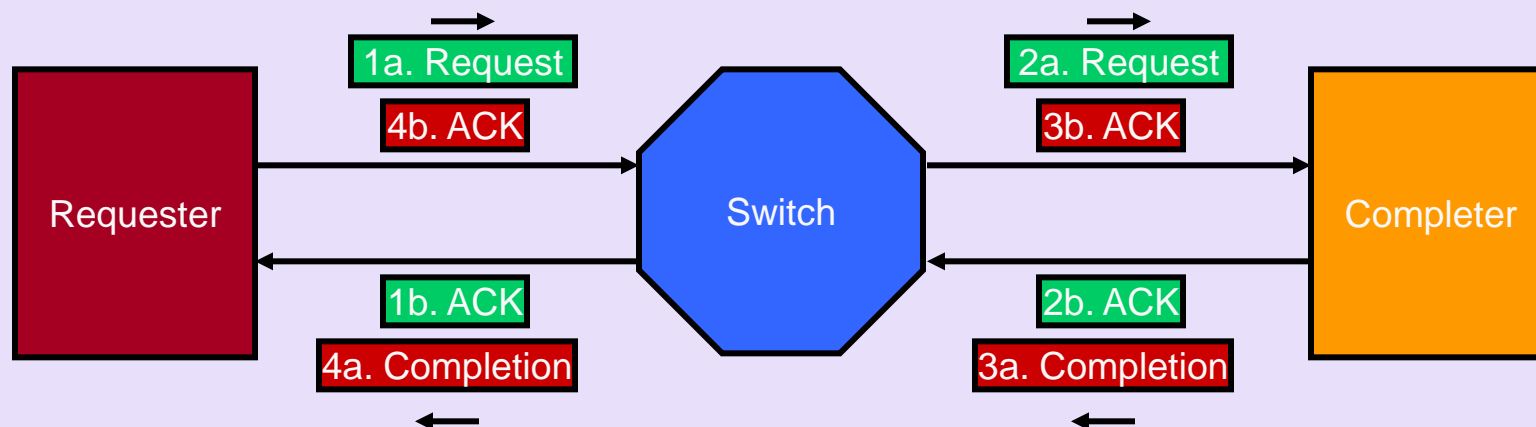


Receiver sends Flow Control Packets (FCP) which are a type of DLLP (Data Link Layer Packet) to provide the transmitter with credits so that it can transmit packets to the receiver

# ACK/NAK Protocol Overview



# ACK/NAK Protocol: Point-to-Point

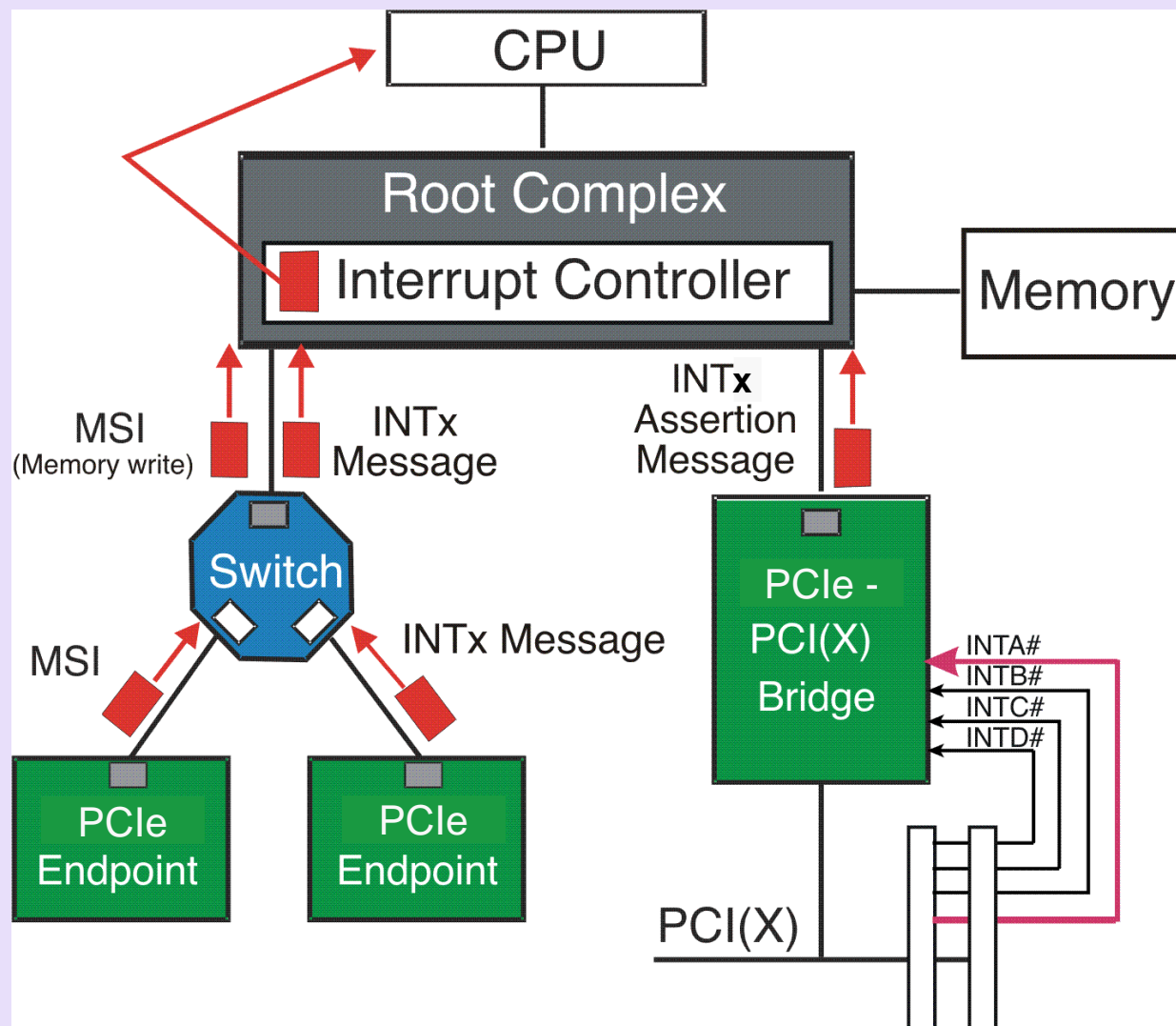


ACK returned for good reception of Request or Completion  
NAK returned for error reception of Request or Completion

# Interrupt Model: Three Methods

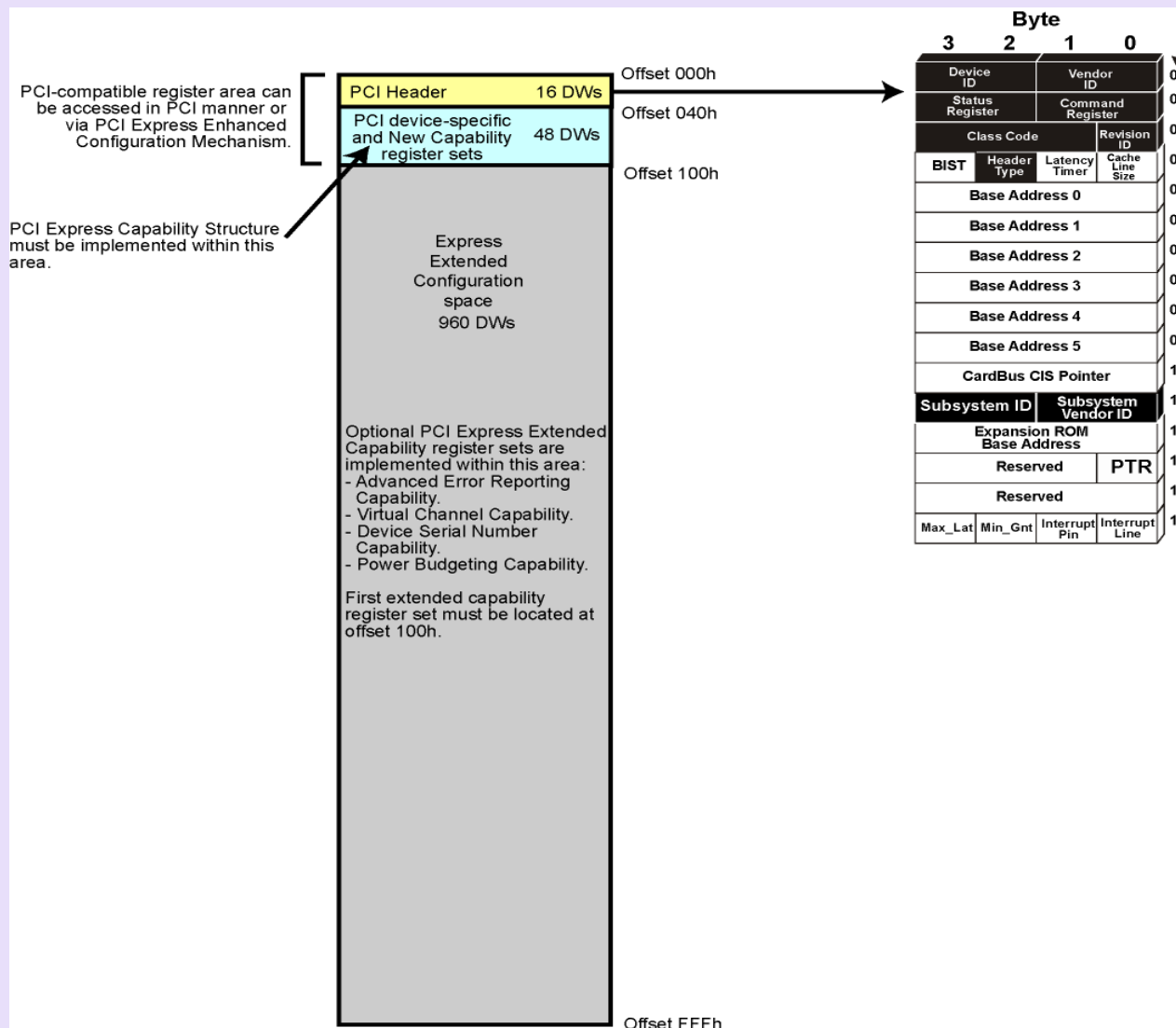
- PCI Express supports three interrupt reporting mechanisms:
  1. **Message Signaled Interrupts (MSI)**
    - Legacy endpoints are required to support MSI (or MSI-X) with 32- or 64-bit MSI capability register implementation
    - Native PCI Express endpoints are required to support MSI with 64-bit MSI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

# Native and Legacy Interrupts





# PCI Express Configuration Space



Thank you for attending the  
PCI-SIG Developers Conference 2010.

For more information please go to  
[www.pcisig.com](http://www.pcisig.com)