



Optimizing PCIe[®] Port Performance

Ilya Granovsky

Engineering and Technology Services

IBM



Agenda

- Motivation
- Performance Parameters Overview
- Data Link Layer Parameters
- Transaction Layer Parameters
- Data Path Architecture Aspects
- Summary

Motivation

- Support system requirements
- Achieve optimal resources utilization for target performance
 - ✓ Silicon size
 - ✓ Power
 - ✓ Development and verification costs

Agenda

- Motivation
- Performance Parameters Overview
- Data Link Layer Parameters
- Transaction Layer Parameters
- Data Path Architecture Aspects
- Summary

- Bandwidth

$$\text{BW} = \frac{\text{[total transferred data size]}}{\text{[transfer time]}} \quad \text{[bytes/sec]}$$

- ✓ Measured for each path (Tx/Rx, reads/writes)
- ✓ Should indicate sustainable values (not peak performance)
- ✓ Assumes appropriate system configuration, optimal payload size and alignment, no errors

Other Parameters

- Latency
- Cost/Power
 - ✓ Total Power Consumption (Logic Size)
 - ✓ Power Efficiency
 - ✓ Power Management Support
 - ✓ System Resources Utilization
- QoS
 - ✓ Bandwidth Allocation
 - ✓ Congestion Avoidance
- Error Protection
 - ✓ Link layer mechanisms
 - ✓ Transaction layer mechanisms

PCIe[®] Bandwidth

Theoretical bandwidth

- x16 – 4 GByte/sec per direction
- x8 – 2 GByte/sec per direction
- x4 – 1 GByte/sec per direction
- x1 – 0.25 GByte/sec per direction

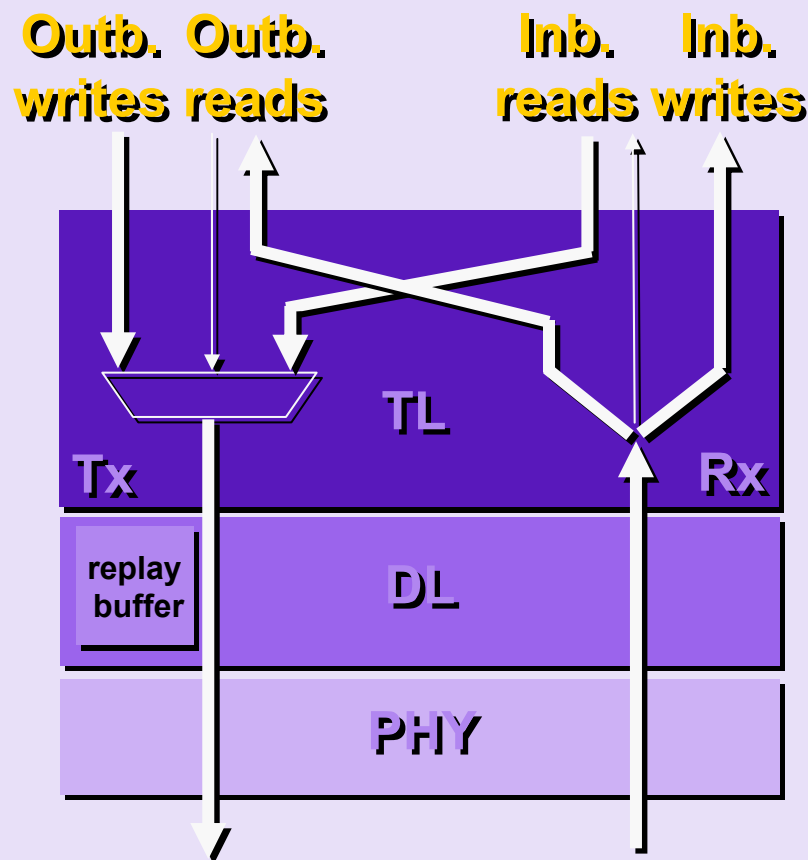
Degradation Factors

- Packet overhead
 - ✓ TLP header
 - ✓ ECRC
 - ✓ Link header and framing
- Link Management
 - ✓ FC Updates
 - ✓ ACK/NAL DLLPs
- System Efficiency
- Congestion

System Performance

■ Specify your system performance criteria

- ✓ inbound reads
- ✓ inbound writes
- ✓ outbound reads
- ✓ outbound writes
- ✓ receive latency
- ✓ transmit latency



Agenda

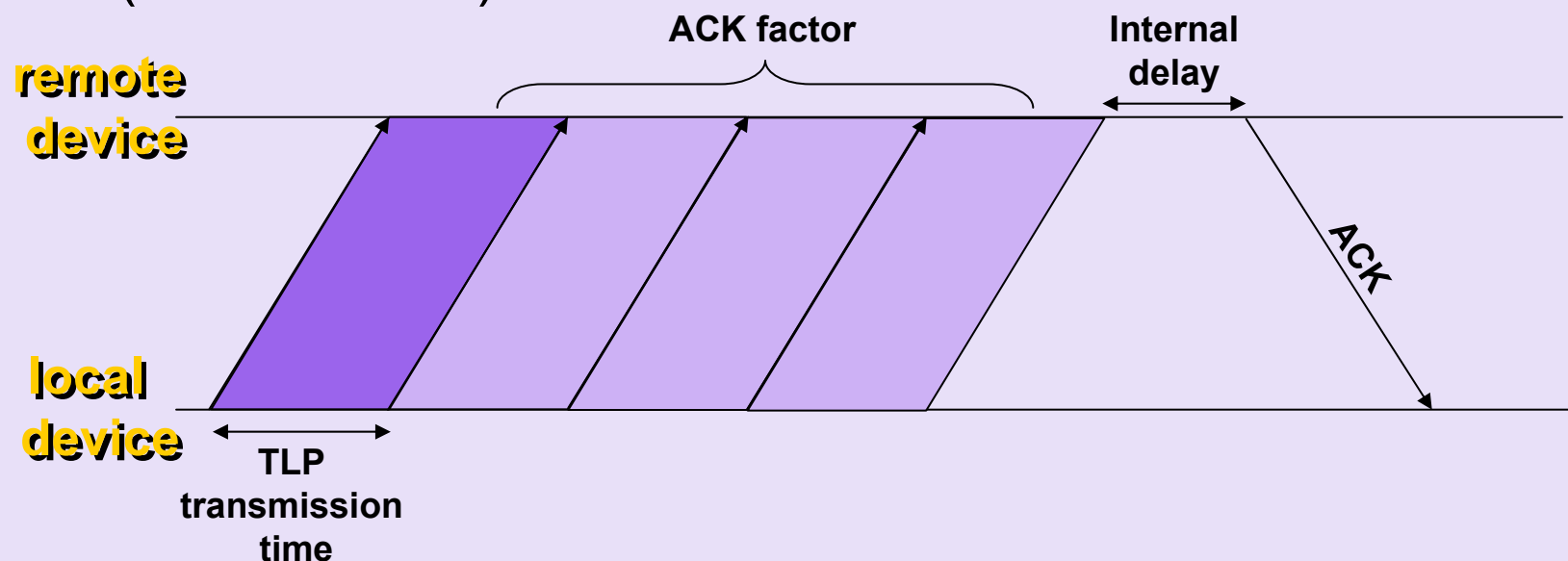
- Motivation
- Performance Parameters Overview
- Data Link Layer Parameters
- Transaction Layer Parameters
- Data Path Architecture Aspects
- Summary

Data Link Layer Parameters

- Replay Buffer
- ACK DLLPs coalescing
- Flow Control Update Protocol

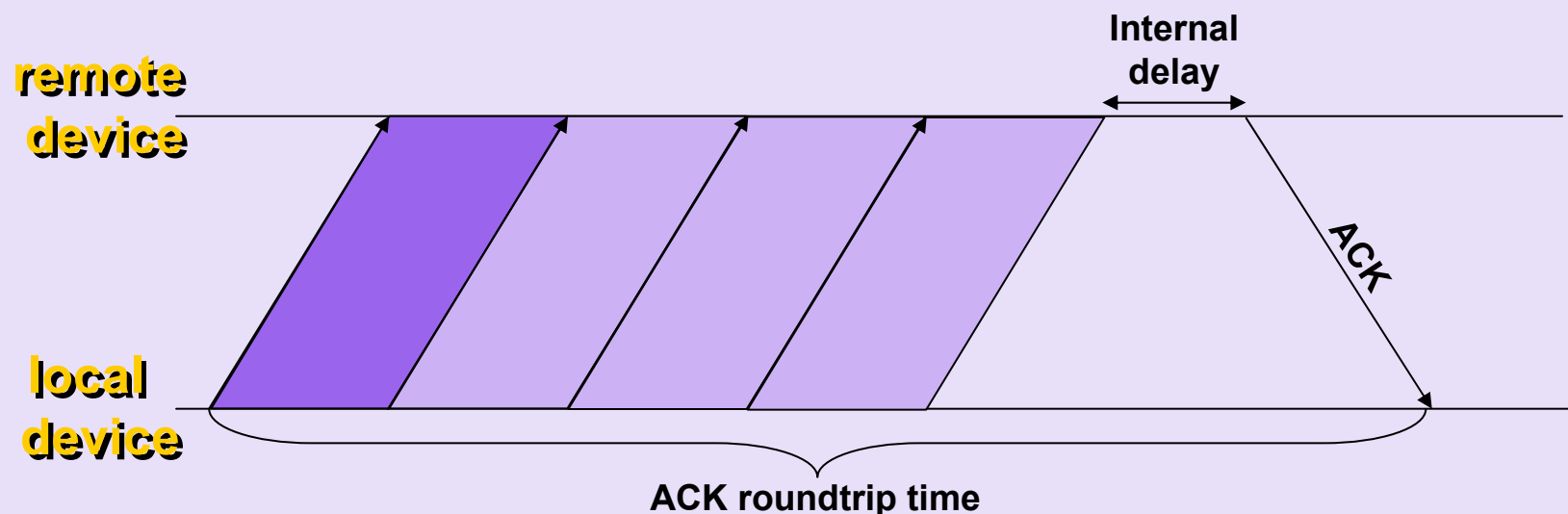
Replay Buffer

- Impacts transmit bandwidth
- Stores outbound TLPs until acknowledged by receiver to allow retransmission in case of LCRC error
- Usually ACK DLLPs are combined for number of TLPs (“ACK factor”)



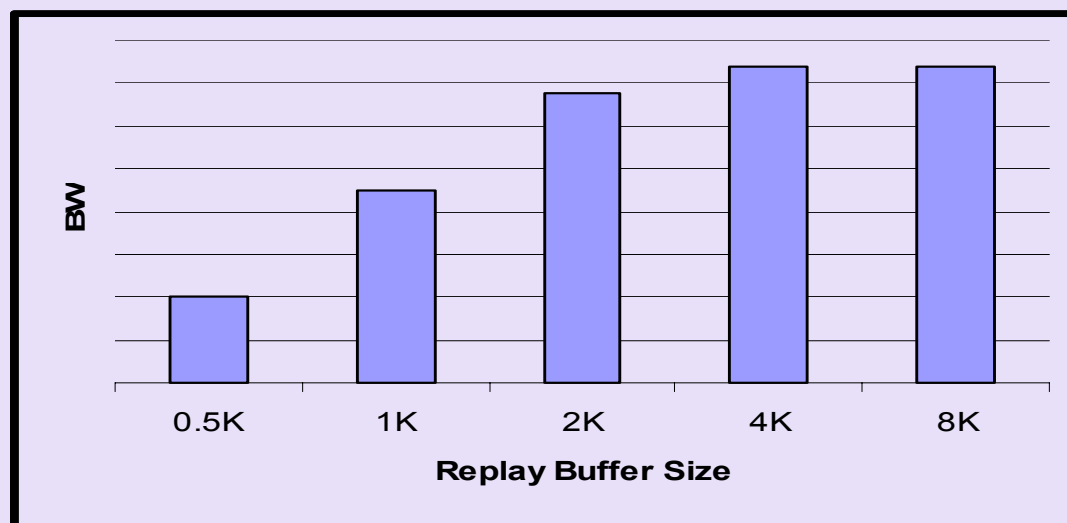
Replay Buffer – cont.

- Replay buffer should be large enough to compensate for ACK roundtrip time
 - ✓ (TLP transmission time) X ACK factor
 - ✓ TLP Transmission delay
 - ✓ Internal processing delay in receiver
 - ✓ ACK transmission delay



Replay Buffer – cont.

- ACK factor – allows ACK DLLPs coalescing
 - ✓ Pro - reduces link management overhead
 - ✓ Con - requires larger replay buffer
- Simulation results - outbound writes bandwidth vs. replay buffer size, x8 link, packet size of 512B



Flow Control Update Policy

- FC management controls receive flow but impacts transmit overhead
- With every TLP
 - ✓ Con - increases link management overhead
 - ✓ Pro - requires smaller receive buffers
 - ✓ Pro - improves buffering utilization
- Once in time period
 - ✓ Pro - reduces link management overhead
 - ✓ Con - requires larger receive buffers

Agenda

- Motivation
- Performance Parameters Overview
- Data Link Layer Parameters
- Transaction Layer Parameters
- Data Path Architecture Aspects
- Summary

Transaction Layer Parameters

- Max Payload Size
- Max Read Size
- Receive buffering
- Reads performance optimization

Max Payload Size

Large Payloads (1KB - 4 KB) - pros

- Better Link Utilization
 - ✓ Lower TL overhead (4 header DWORDs for 1024 data DWORDs)
 - ✓ Less frequent FC updates
- Less Header Credits
 - ✓ Smaller header buffers
 - ✓ Lower headers processing overhead

Max Payload Size

Large Payloads (1KB - 4 KB) - cons

- Requires Large Buffers
 - ✓ TL data buffers
 - ✓ Replay buffers
- Needs PCIe native software
- Has to be supported by all the devices in the system

Max Payload Size – cont.

Small Payloads (128 bytes - 256 bytes) - pros

- Require Smaller Buffers
 - ✓ Simplifies data buffer management if implemented in 128 byte units (no data credits management needed)
- Supported by all devices
- Compatible with existing PCI software

Max Payload Size – cont.

Small Payloads (128 bytes - 256 bytes) - cons

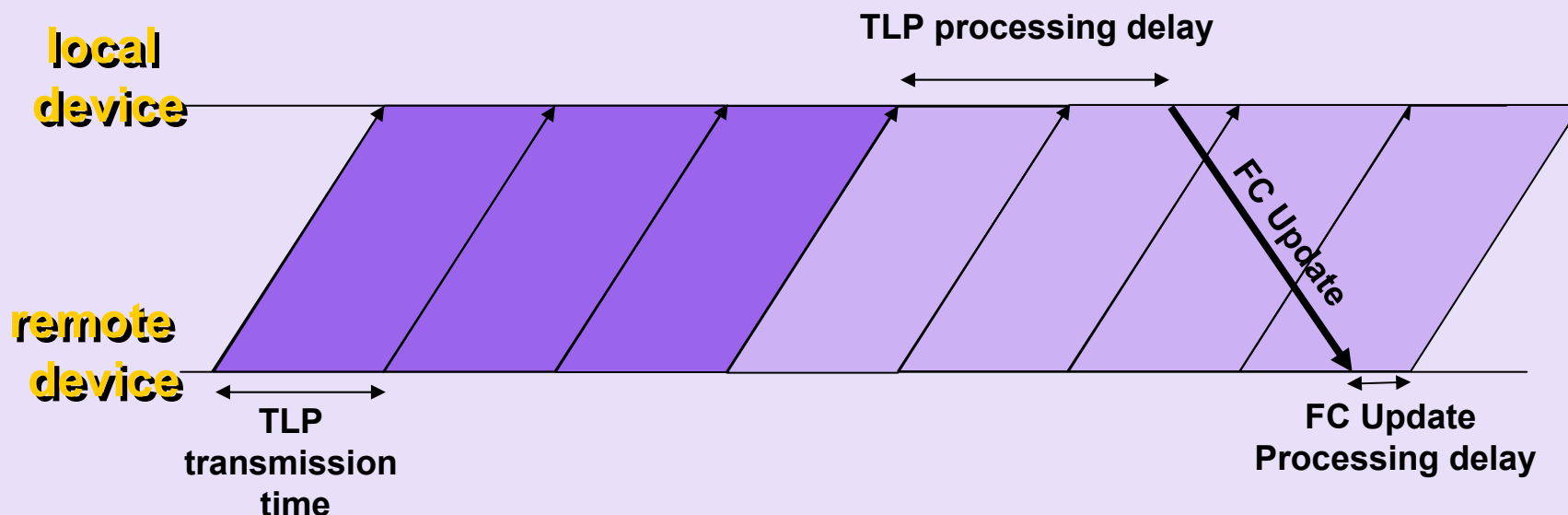
- Lower Link Utilization
 - ✓ High per-packet overhead (4 header DWORDs for 32 data DWORDs)
 - ✓ Frequent FC updates

Max Payload Size – System Limitations

- Default payload size of 128B – legacy software is unable to modify max payload parameter programmed through new configuration space capability.
- All devices in the system should be designed for the same Max payload
- Max payload size of 512B seems to be the sweet spot between link overhead and buffers size

Receive Buffers

- Receiver should advertise amount of credits sufficient to support continuous data flow
- Consider FC update roundtrip from the remote transmitter point



Receive Buffers – cont.

Example

- x8 link, 128B max payload size => TLP transmit time of 72ns
- FC Update is issued once in 4 TLPs
- TLP processing delay is 60ns
- Transmit/receive latency is 90ns
- FC Update processing delay is 20ns
- => roundtrip time is:

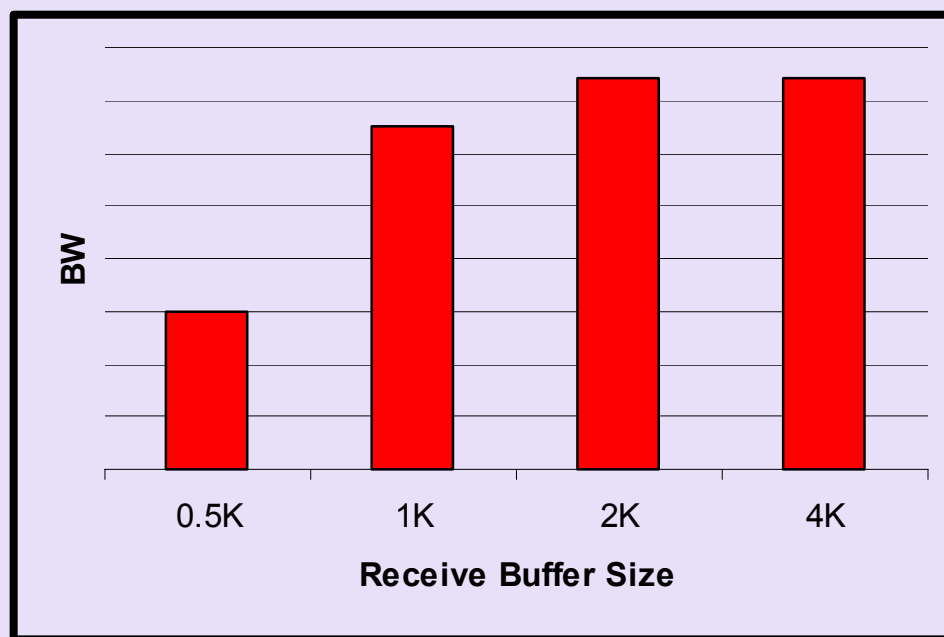
$$90ns + 4 \times 72ns + 60ns + 90ns + 20ns = 548ns$$

- => Rx data buffer should be:

$$(548ns/72ns) \times 128B \approx \mathbf{1KB}$$

Receive Buffers – cont.

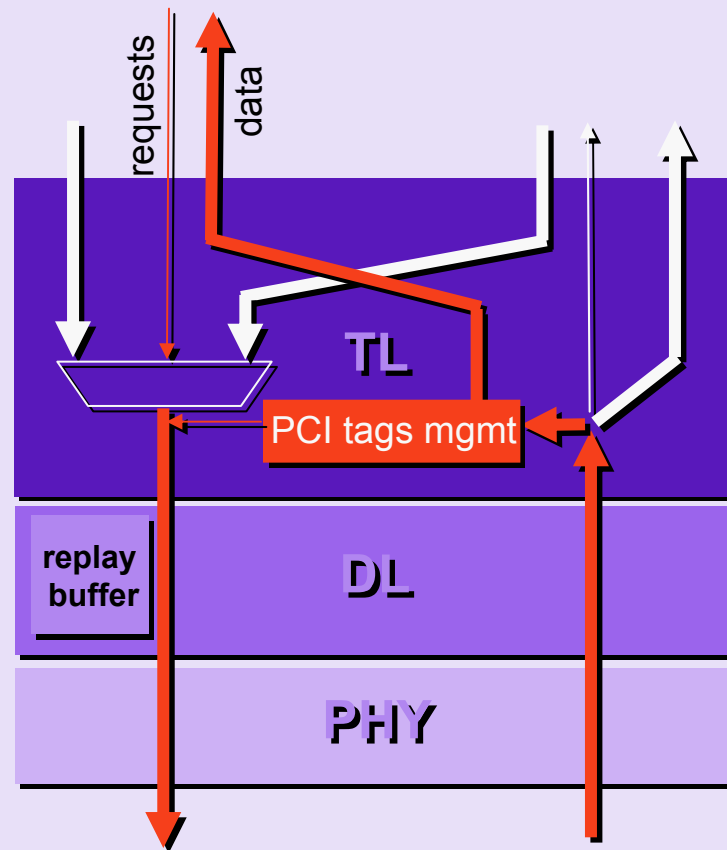
- Simulation results - inbound writes bandwidth vs. Rx posted data buffer size, x8 link, fixed TLP payload size of 512B



Optimizing Reads Performance

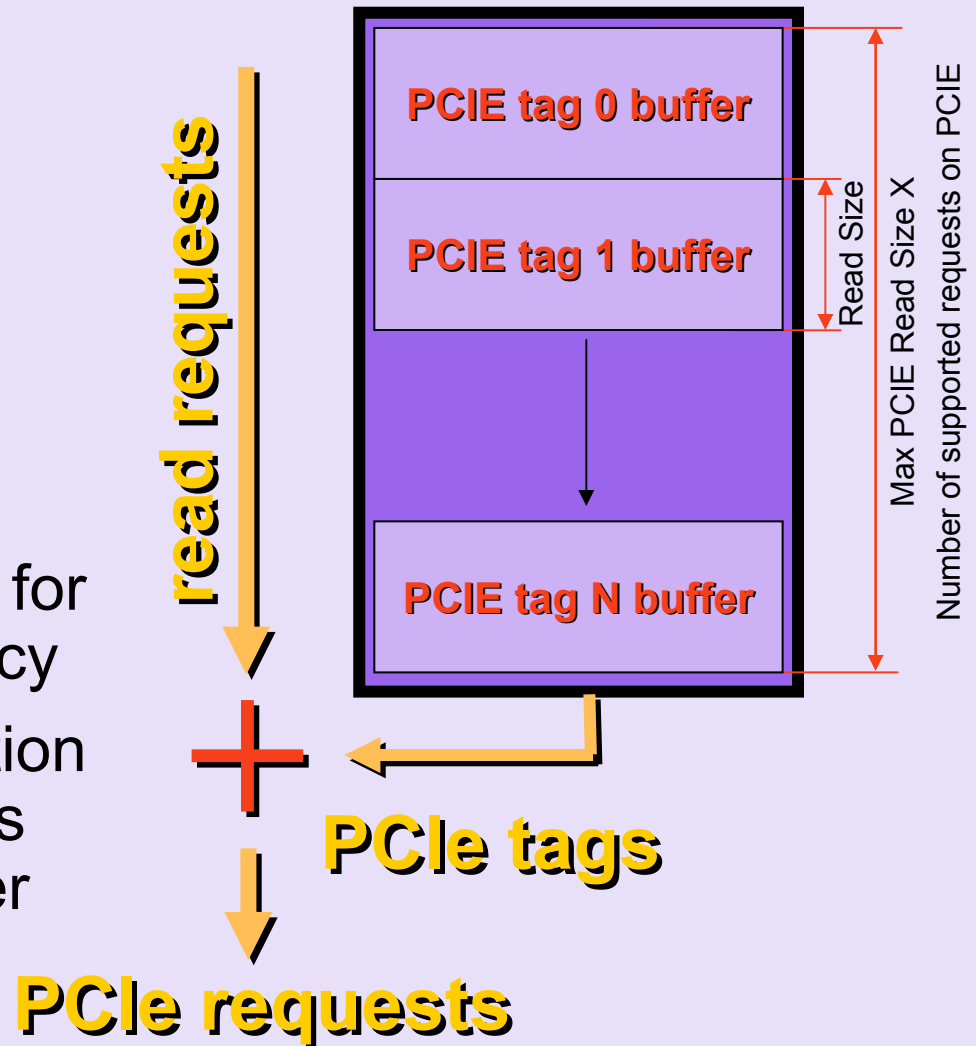
- Outbound Read Parameters
 - ✓ Outbound read requests queue depth
 - ✓ Max Read Size
 - ✓ Number of PCI tags allocated to reads (number of read transactions outstanding on PCIe)
 - ✓ Max Payload Size

Outbound reads



Optimizing Reads Performance – cont.

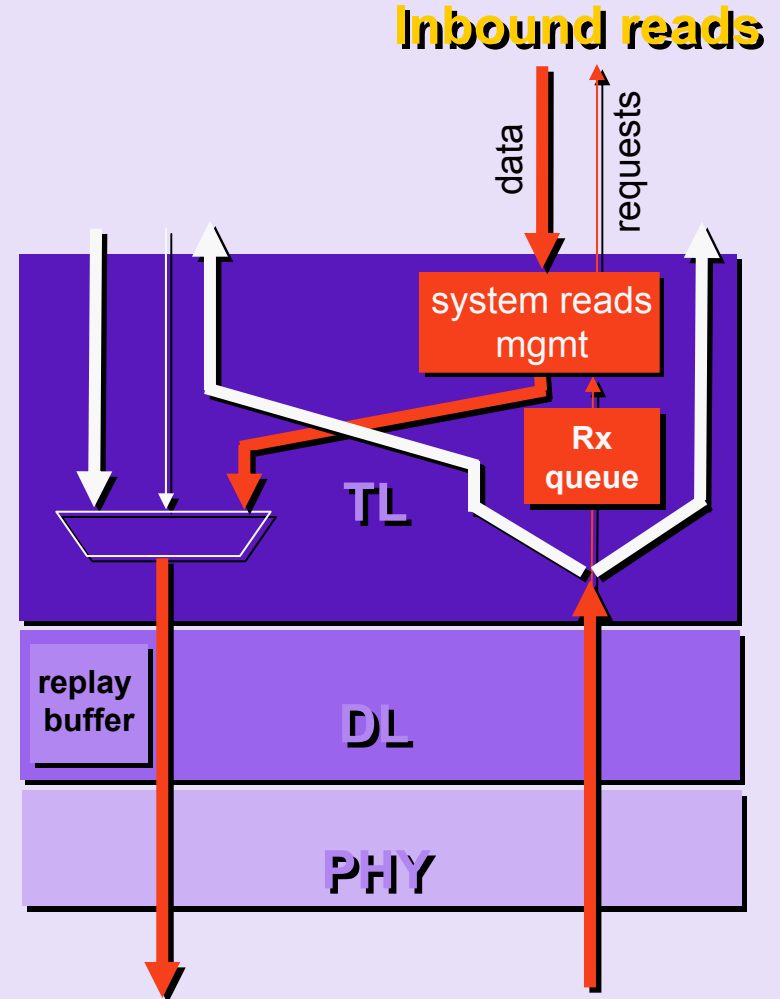
- Read data buffer
 - ✓ Read data buffer is pre-allocated by requestor when forwarding read request to PCIe
 - ✓ Read data buffer should compensate for read roundtrip latency
 - ✓ Simple implementation – associate PCI tags with fixed data buffer segments.



Optimizing Reads Performance – cont.

■ Inbound Read Parameters

- ✓ Inbound read requests queue size – NP header credits
- ✓ Max read size supported on system bus
- ✓ System bus read latency
- ✓ Number of reads outstanding on system bus



Optimizing Reads Performance - example

- Target Inbound Reads Bandwidth – 3.2 Gbyte/sec
- System bus latency (round trip) – 1000 ns
- Need outstanding reads of 3200 bytes total to sustain max bandwidth
 - ✓ 2 reads of 2K
 - ✓ **4 reads of 1K**
 - ✓ 8 reads of 512 bytes
 - ✓ 16 reads of 256 bytes
- Depends on system bus architecture
 - ✓ Request sizes supported
 - ✓ Requests processing latency
 - ✓ Pipelining capabilities

Optimizing Reads Performance - summary

- Consider read transaction roundtrip time for evaluating read request size and number of outstanding reads
- Make sure that read requests queue is not limiting reads performance (there is always read requests in the queue)
- Use large read requests on PCIe (4K) to reduce request processing overhead
- Consider read transactions timeout implementation (see Completion Timeout ECN)

Agenda

- Motivation
- Performance Parameters Overview
- Data Link Layer Parameters
- Transaction Layer Parameters
- Data Path Architecture Aspects
- Summary

Store and Forward Buffering

- Requires larger buffers – at least $2X[\text{max_packet_size}]$
 - ✓ Additional buffering to cover for FC update latency
- Increases latency
 - ✓ Significant latency hit for large payloads
 - ✓ Latency impact is reduced in case of wide link
 - 128 bytes TLP, x16 link = 36ns
- Simplifies error handling implementation – all the errors are handled before packet is stored

Cut-through Buffering

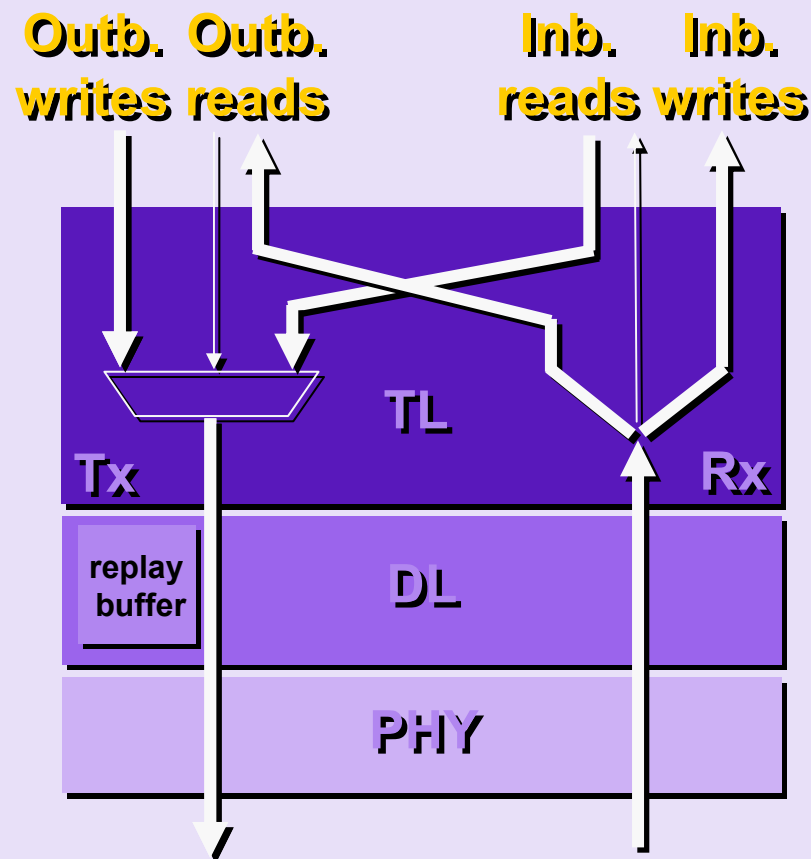
- Lower latency
- Smaller buffers
 - ✓ If FC update latency is large, buffer size advantage becomes negligible
- Complex implementation
 - ✓ Should support errors forwarding through the buffers
 - ✓ Complex credits management

Virtual Channels

- Multiple VC systems require separate logical buffers to allow queue bypass
 - ✓ Header buffers usually require separate physical arrays to allow low-latency arbitration, data buffers may share same physical array.
- Larger buffers required to support max peak bandwidth for different VCs
 - ✓ Some VCs can be defined as low-performance and consume minimal buffering

Ordering Rules

- Consider ordering rules enforcement implementation aspects in case of mixed traffic:
 - ✓ Outbound read requests/completions push posted writes
 - ✓ Inbound read requests/completions push posted writes
- Congestion on one path will impact others



Ordering Rules – cont.

- Consider queues bypassing implementation whenever possible
 - ✓ Posted requests with Relaxed Ordering attribute set can bypass other posted requests
 - ✓ Non-posted requests can bypass other non-posted requests
 - ✓ I/O or Configuration write completions can bypass posted requests
 - ✓ Completions with Relaxed Ordering attribute set can bypass posted requests
 - ✓ Completions are allowed to bypass completions associated with other requests

Agenda

- Motivation
- Performance Parameters Overview
- Data Link Layer Parameters
- Transaction Layer Parameters
- Data Path Architecture Aspects
- Summary

Summary

- Accurate system specification is essential for optimal design
 - ✓ Target bandwidth
 - ✓ Latencies estimation
- Improve bandwidth by reducing link overhead
 - ✓ Large payloads – 512B is the sweet spot
 - ✓ Large read requests
 - ✓ Larger replay buffers
 - ✓ Efficient credits management

Summary

- Estimate posted data buffer sizes based on FC update roundtrip
- Architecture reads support (tags, data buffers) based on read transactions roundtrip
- Store-and-forward buffers are effective for small payloads (128B) but significantly increase latency for large packets.

Thank you for attending the
PCI-SIG Developers Conference 2006.

For more information please go to
www.pcisig.com



Optimizing PCIe Port Performance

Ilya Granovsky

Engineering and Technology Services

IBM

