



# Congestion/Quality of Service

Peter Onufryk  
IDT

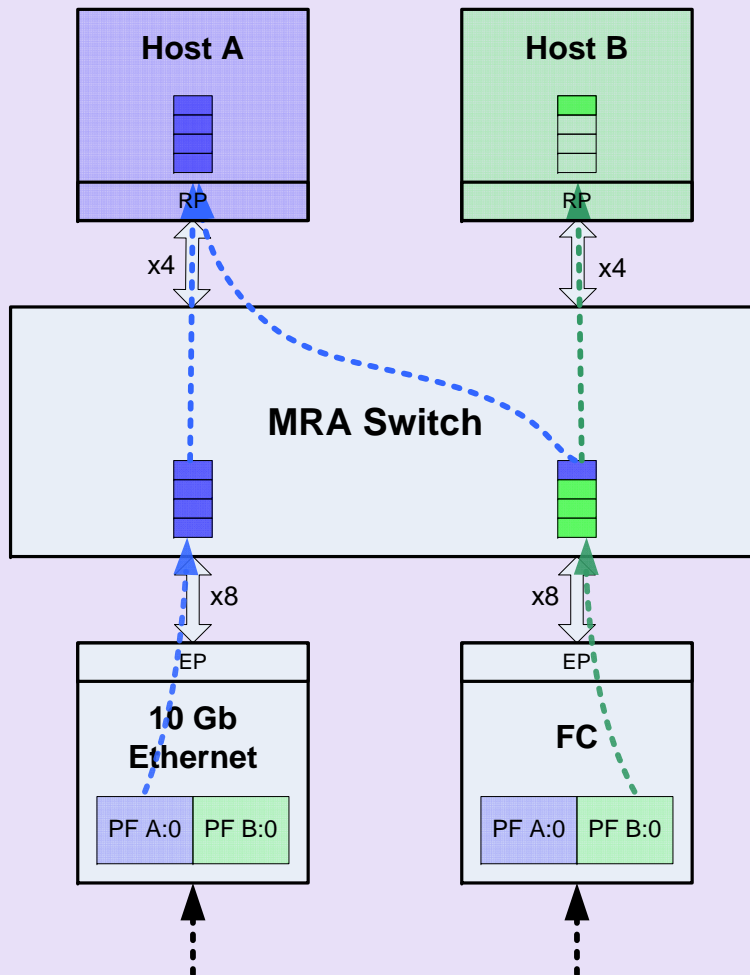
# Disclaimer

- NOTE: The information in this presentation refers to a specification still in the development process. This presentation reflects the current thinking of the workgroup, but all material is subject to change before the specification is released.

# Sources of Congestion

- Hardware fault or software error
  - ✓ A fault in hardware or software error in configuration of a device in the fabric
- Static rate mismatch
  - ✓ A mismatch in the capacity of the path from a device injecting traffic into the fabric (e.g., a x8 Endpoint Device) and the ultimate destination (e.g., a x1 Root Complex)
  - ✓ Congestion due to a static rate mismatch would occur even if the fabric were otherwise idle
- Traffic merging
  - ✓ Traffic merging of multiple flows, none of which individually suffer from a static rate mismatch, causing the capacity of an element in the fabric to be exceeded

# Multi-Root Congestion



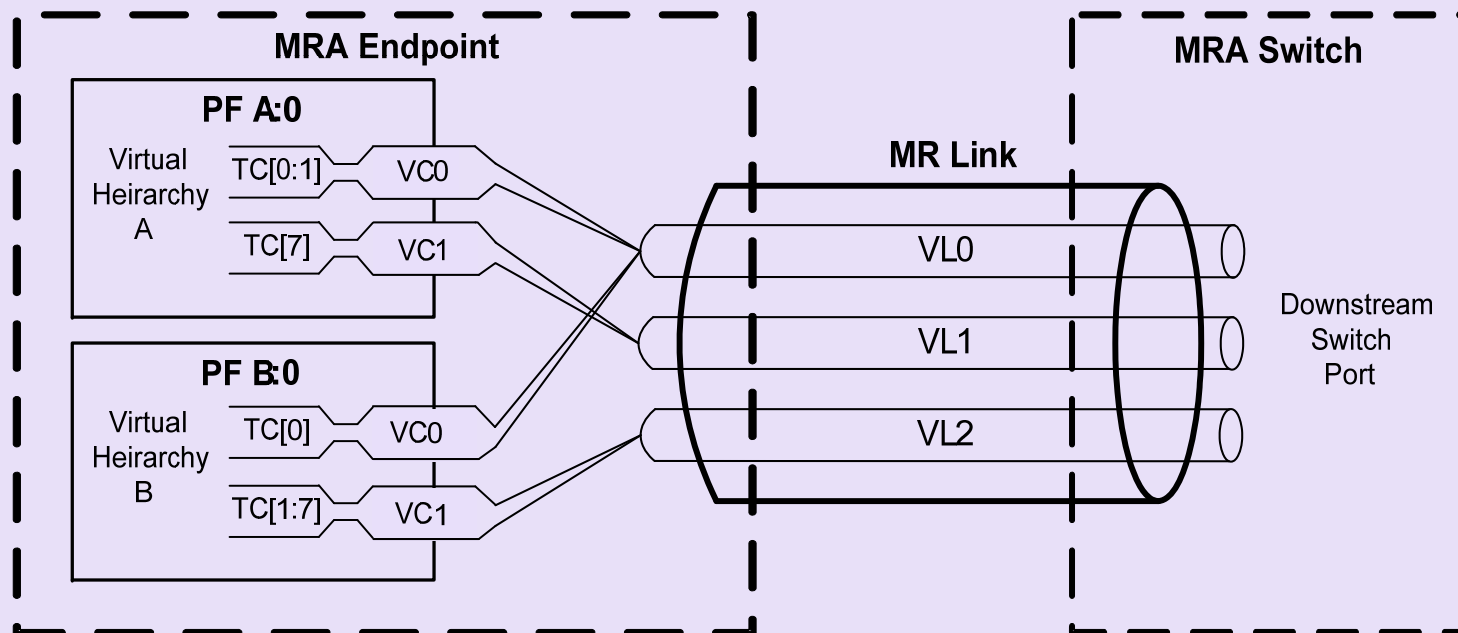
- SR and MR systems have the same sources of congestion
- In MR systems it is desirable to isolate congestion in one VH from affecting performance of another VH
- Static rate mismatches more likely in MR topologies due to endpoint sharing

# Congestion Management Goals

- Address congestion due to interaction of VHs introduced by MR
  - ✓ It is not a goal to address congestion within a VH that would have been present in an equivalent SR system
- Preserve SR Virtual Channel (VC) semantics within a VH
- Allow systems to be constructed where a fault in one VH does not bring down another VH
- Allow systems to be constructed that provide forward progress guarantees within a VH when congestion exists on unrelated VH(s)
- Support a range of implementation options
  - ✓ From low cost/simple implementations to ones that support complete isolation of VHs

# Virtual Links

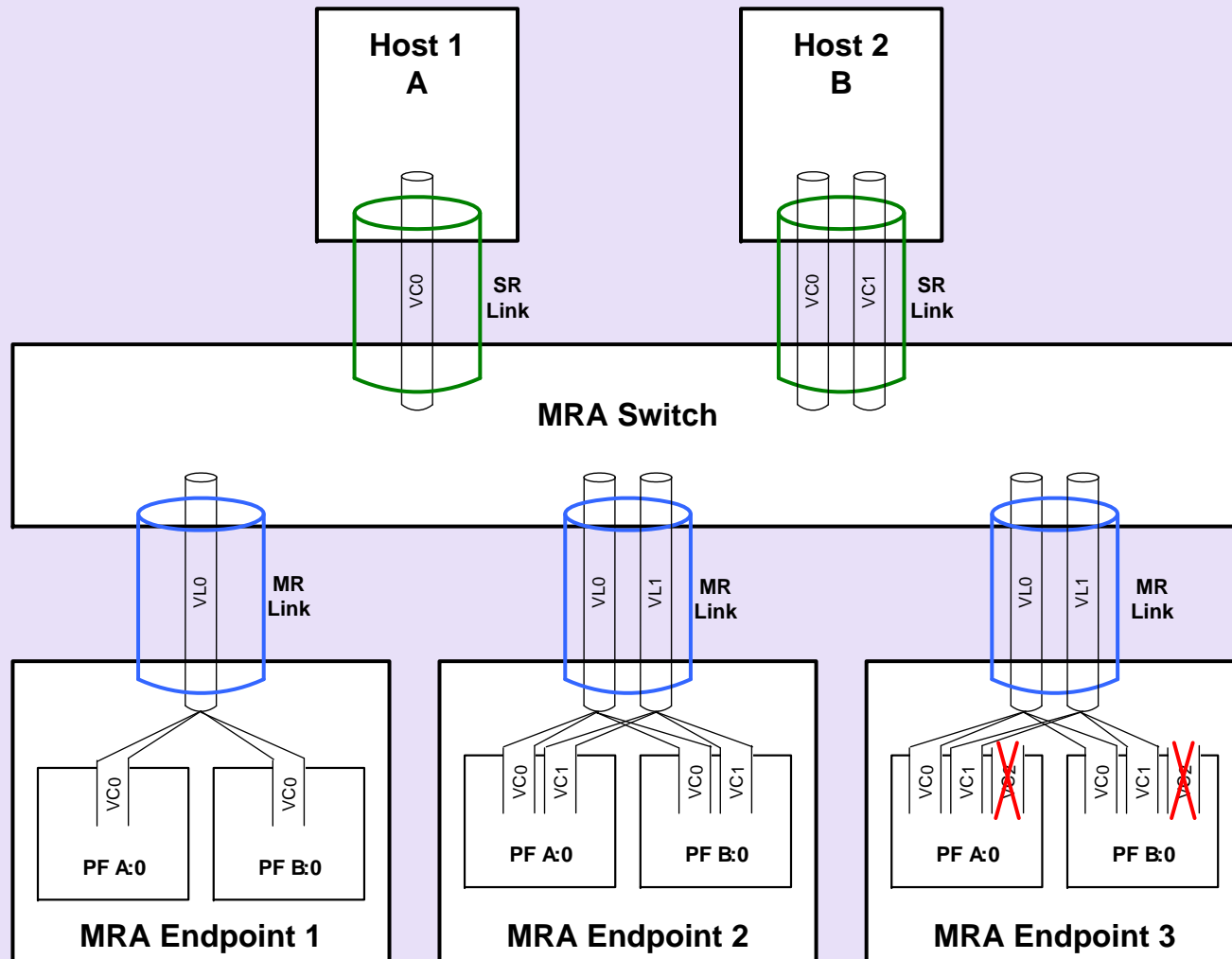
- A device may implement 1 to 8 Virtual Links (VLs)
  - ✓ VLs are the MR equivalent of VCs in PCIe® Base
  - ✓ VLs are associated with a link and are not end-to-end



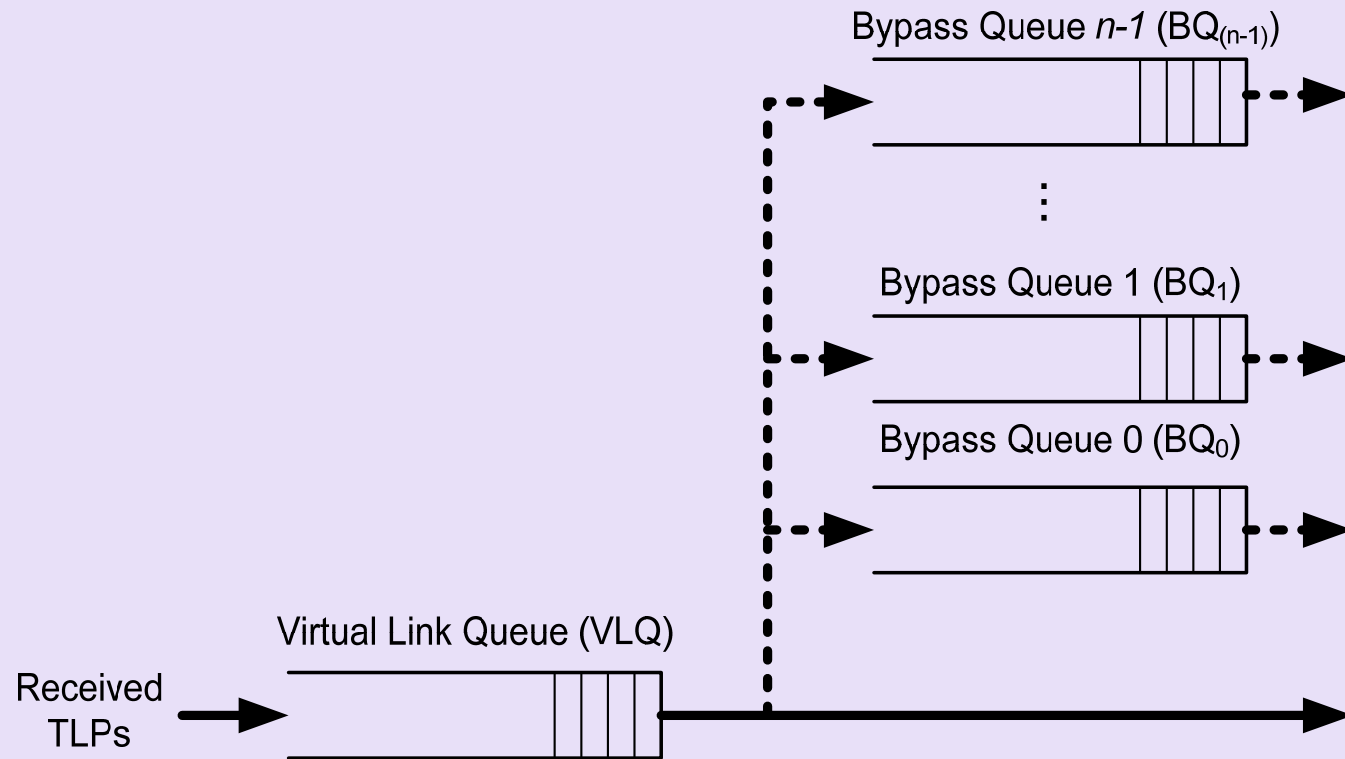
# (VH,VC) to VL Mapping

- Only one VC from a VH may be mapped to a VL
  - ✓ **Allowed:** (VH A, VC0) → VL0 , (VH A, VC1) → VL1
  - ✓ **Not Allowed:** (VH A, VC0) → VL0 , (VH A, VC1) → VL0
- Support for VLs beyond VL0 is optional
  - ✓ Since only one VC from a VH may be mapped to a VL, this implies that VHs associated with such a component only implement VC0
- MRA components that implement multiple VLs must implement a (VH,VC) to VL mapping capability
  - ✓ If only implement VL0, then can only support a single VC
    - (VHx, VC0) → VL0
  - ✓ VC ID to VL map
    - For a switch located in the VS Bridge Table
    - For an endpoint located in a Function Table Entry

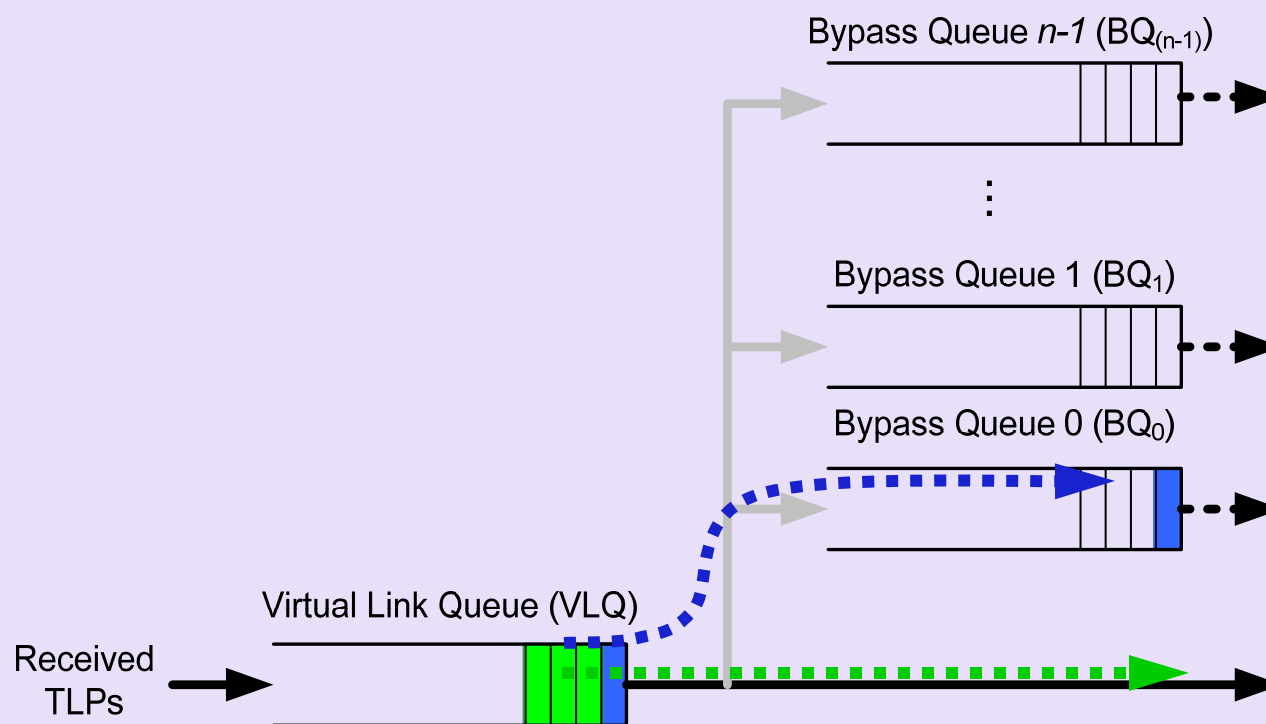
# VC to VL Mapping Example



# Model of Queuing at VL Receiver



# Bypass Example



# Buffer Accounting

- Virtual Link Buffer (VLB) is the total amount of buffering (i.e., credits) available to a VL for the VLQ and any BQs
  - ✓ There is a VLB value associated with each type  
(P, NP, Cpl) x (Header, Data)
- VL and VH credits are tracked per type
  - ✓ VL credits correspond to size of VLB
  - ✓ VH credits correspond to maximum allowed size of a BQ
- A TLP in the VLQ or a BQ consumes both VL and VH credits
  - ✓ A TLP moved from the VLQ to a BQ does not result in a release of buffer credits
- Both VL and VH credits are released when a TLP is processed and removed from the input buffer



# Flow Control Information Tracked by a Transmitter

- A transmitter tracks the following quantities for each enabled VL and VH
  - ✓ CREDITS\_CONSUMED
    - Computed in the same manner as in PCIe Base
  - ✓ CREDIT\_LIMIT
    - Configured by MR PCIM during MR link initialization
- Transmitter gating function rules for VLs and VHs are each the same as in PCIe Base
- A TLP may be transmitted if the transmitter gating function passes for both the VL and VH



# Flow Control Information Tracked by a Receiver

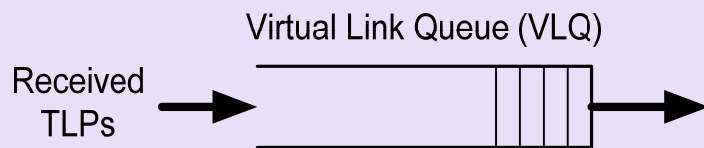
- A receiver tracks the following quantities for each enabled VL and VH
  - ✓ CREDITS\_ALLOCATED
    - Initial value communicated from Receiver to Transmitter using MR flow control initialization protocol
      - For VLs
        - Corresponds to the size of the VLB at the Receiver
      - For VHs
        - Corresponds to the maximum BQ size at the Receiver
        - Typically the same for all VHs but this is not required
    - VL and VH values are incremented as buffer space becomes available due to processing of received TLPs
  - ✓ CREDITS\_RECEIVED (*optional*)
    - Computed in the same manner as in PCIe Base

# Implementation Options

- MR specification outlines
  - ✓ Model of queuing at receiver
  - ✓ FC information tracked by transmitter and receiver
  - ✓ Transmitter gating function
- Specification does not dictate an input buffering implementation
  - ✓ People are free to innovate



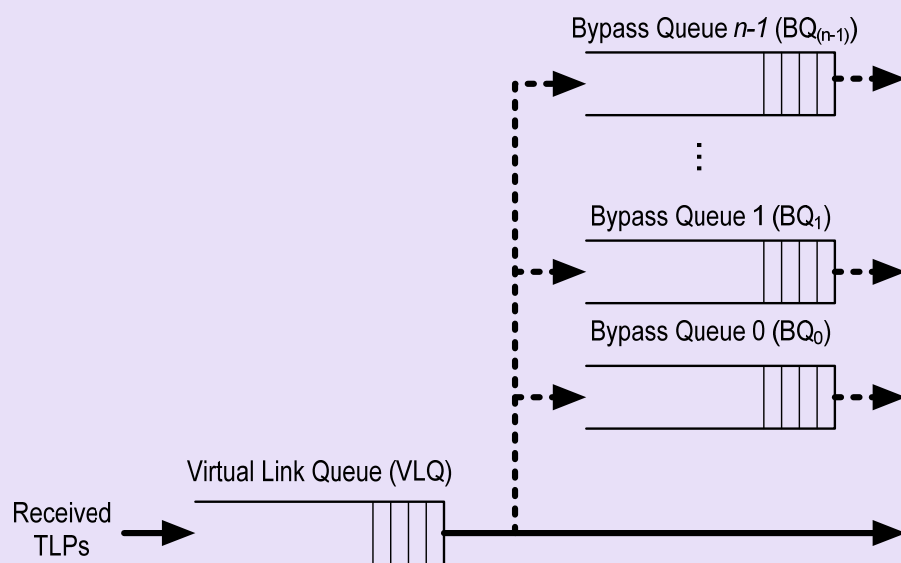
# Example Simple FIFO Implementation



Per VL and Type Structure

- Implement a single queue
  - ✓ VLB = VLQ
- Advertised credits
  - ✓ VL credits corresponds to size of VLQ
  - ✓ Logically advertise infinite VH credits
- Looks like a VC in base

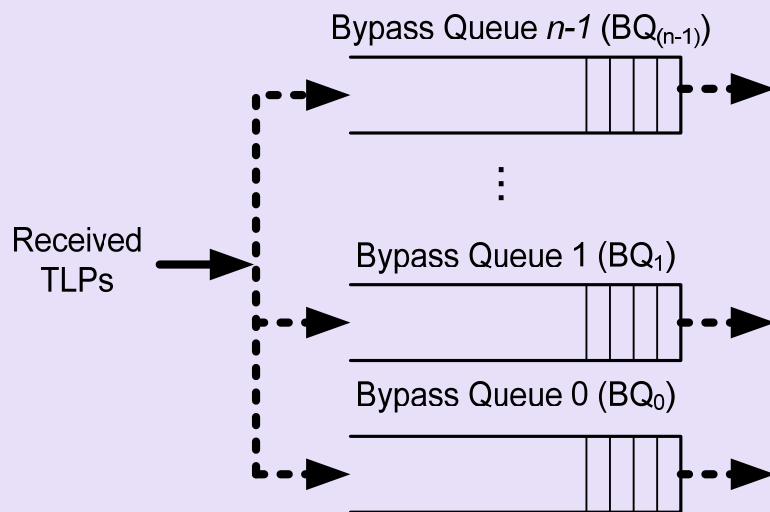
# Example Intermediate Implementation



Per VL and Type Structure

- Linked-list based queuing structure at receiver
  - ✓  $VLB = VLQ + \text{SUM}(BQ)$
- Advertised credits
  - ✓ VL credits corresponds to total memory available for all queues (i.e., VLB)
  - ✓ VH credits corresponds to maximum size of a BQ

# Example Per VH Queuing Implementation



Per VL and Type Structure

- Implement queue for each VH
  - ✓ Queues need not be of equal size
  - ✓  $VLB = \text{SUM}(BQ)$
- Advertised credits
  - ✓ Advertise infinite VL credits
  - ✓ VH credits corresponds to maximum queue size

# MR Flow Control Packets

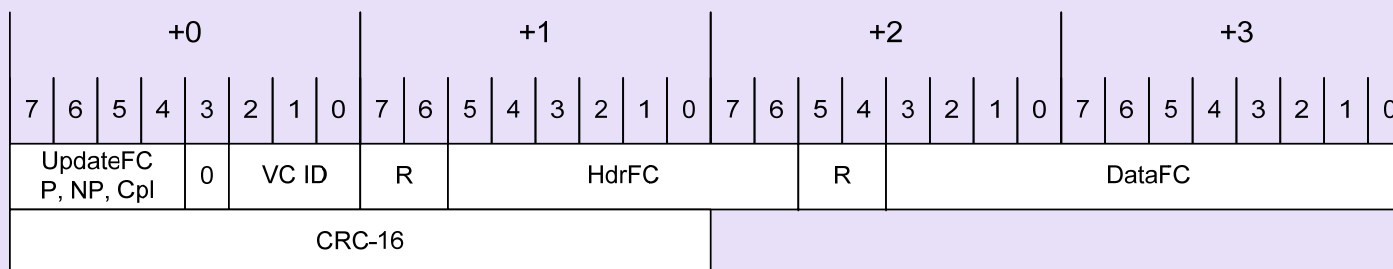
- Length of FCPs are the same as in PCIe Base
- FCPs are protected by a CRC-16
- The unit of flow control credits is the same as in PCIe Base
  - ✓ Data credit
    - 4 DWORDS
  - ✓ Header credit
    - Maximum sized PCIe Base header + MR TLP prefix + TLP digest
- A goal is to limit the amount of link bandwidth consumed by MR FCPs
  - ✓ Reducing FC update traffic associated with inactive flows
    - After an MR UpdateFC has been sent TBD [proposed 4] times with the same value, subsequent MR UpdateFC FCPs must be scheduled for transmission once every TBD [proposed 100 ms] (-0%/+50%)
  - ✓ Track only VH credits on the wire

# MR Flow Control Packets

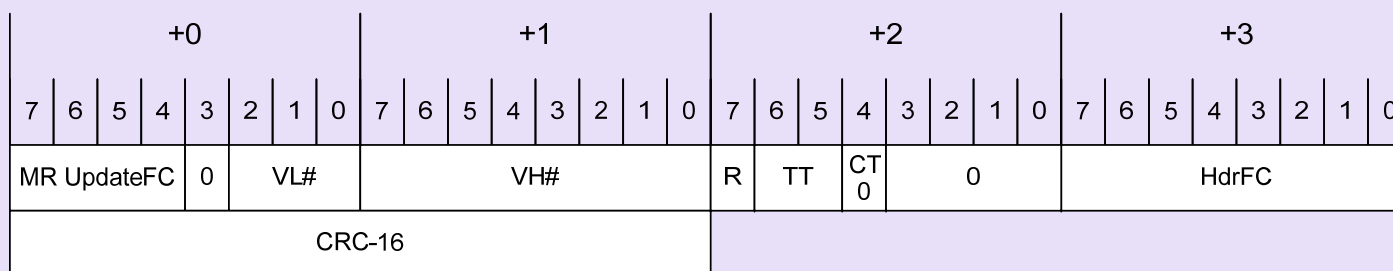
- Track only VH credits on the wire
  - ✓ UpdateFC contains both VL and VH IDs
  - ✓ VH credit update determined directly from FCP
  - ✓ VL credit update determined implicitly from FCP using VH transmitter state
  - ✓ If both VL and VH credits are not infinite, then must track VH credits
    - Advertise maximum VH credits
      - 2047 credits for data payload
      - 127 credits for header
- Transmitter Behavior
$$\text{CREDITS\_RECEIVED} = (\text{UpdateFC} - \text{CREDIT\_LIMIT}_{\text{VH}}) \bmod 2^{\text{[Field Size]}}$$
$$\text{CREDIT\_LIMIT}_{\text{VH}} = \text{UpdateFC}$$
$$\text{CREDIT\_LIMIT}_{\text{VL}} = (\text{CREDIT\_LIMIT}_{\text{VL}} + \text{CREDITS\_RECEIVED}) \bmod 2^{\text{[Field Size]}}$$
- UpdateFC Errors
  - ✓ A lost UpdateFC may cause a delay in the return of VH credits
  - ✓ Delay postpones when VH and VL credits are available to transmitter
  - ✓ Credits are never lost or counted twice



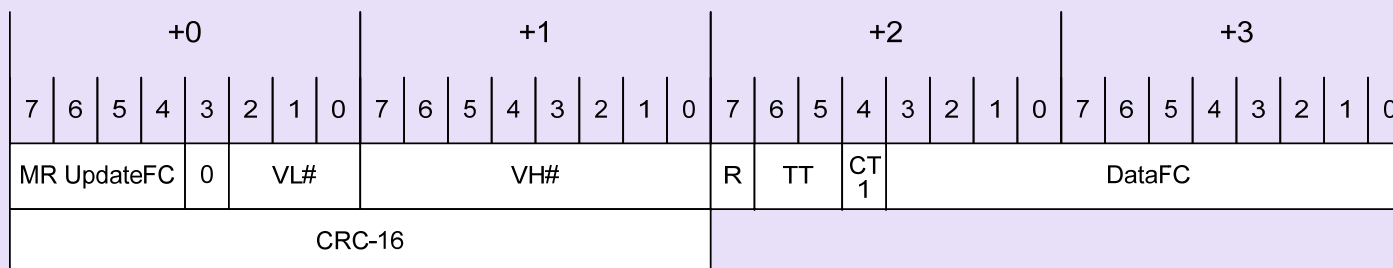
# MR UpdateFC DLLP Format



**PCIe Base UpdateFC DLLP Format**



**MR UpdateFC VH Header DLLP Format**

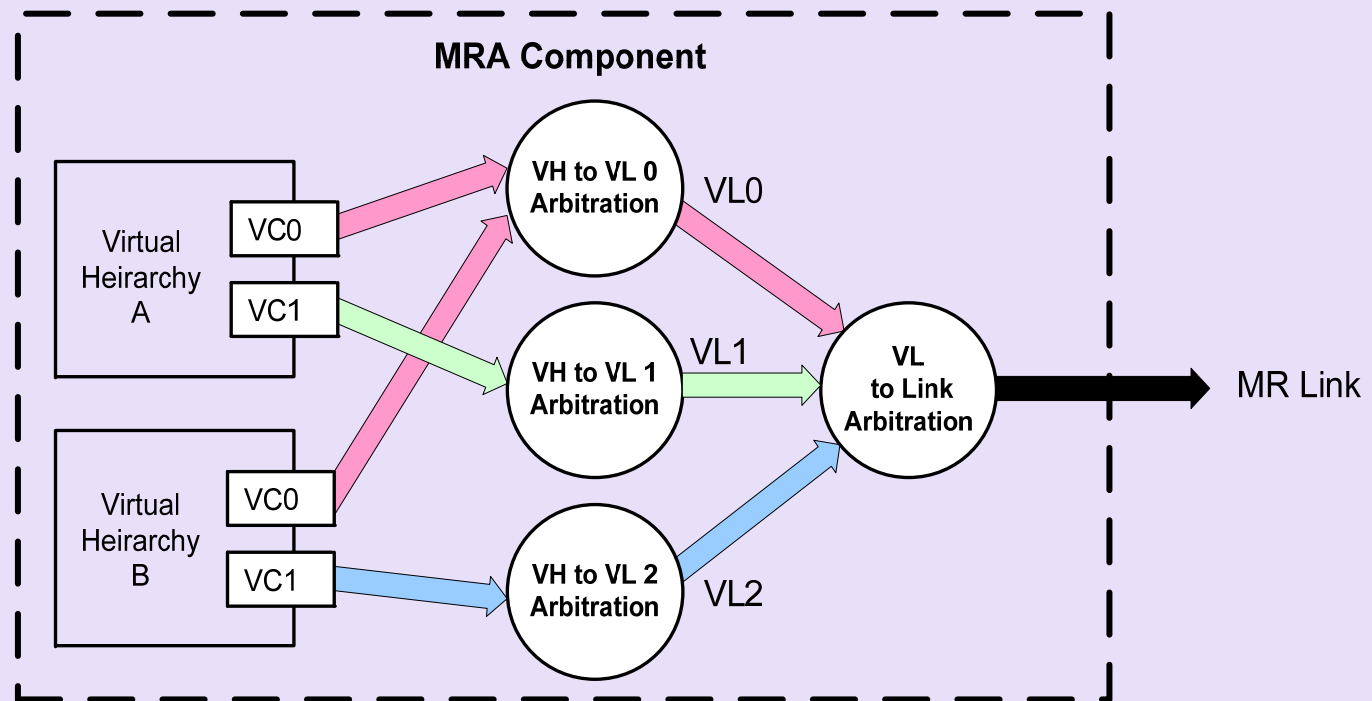


**MR UpdateFC VH Data DLLP Format**

# FCP Information Tracking

- MR flow control information is tracked by receivers and transmitters for all enabled VLs and VHs
- VH reset
  - ✓ A receiver must still send update FCPs and return credits even if a VH is reset and is flushing packets
- Enabling and disabling of VCs within a VH
  - ✓ A receiver must still send update FCPs and return credits even if a VC within a VH is disabled
  - ✓ Enabling a VC does not result in initiation of the PCIe Base flow control initialization protocol

# MR Arbitration Model



# VL to Link Arbitration

- VL to Link arbitration is associated with a port
  - ✓ Guarantees forward progress on all enabled data flow
  - ✓ Allows differentiated service characteristics for data flows within an MR topology
  - ✓ Provides ability to tune bandwidth and end-to-end latency between components in an MR topology
- Software configurability of VL to Link arbitration is optional
  - ✓ If not implemented then must implement a hardwired fixed arbitration scheme (e.g., Round Robin) that guarantees forward progress



# Software Configurability of VL to Link Arbitration

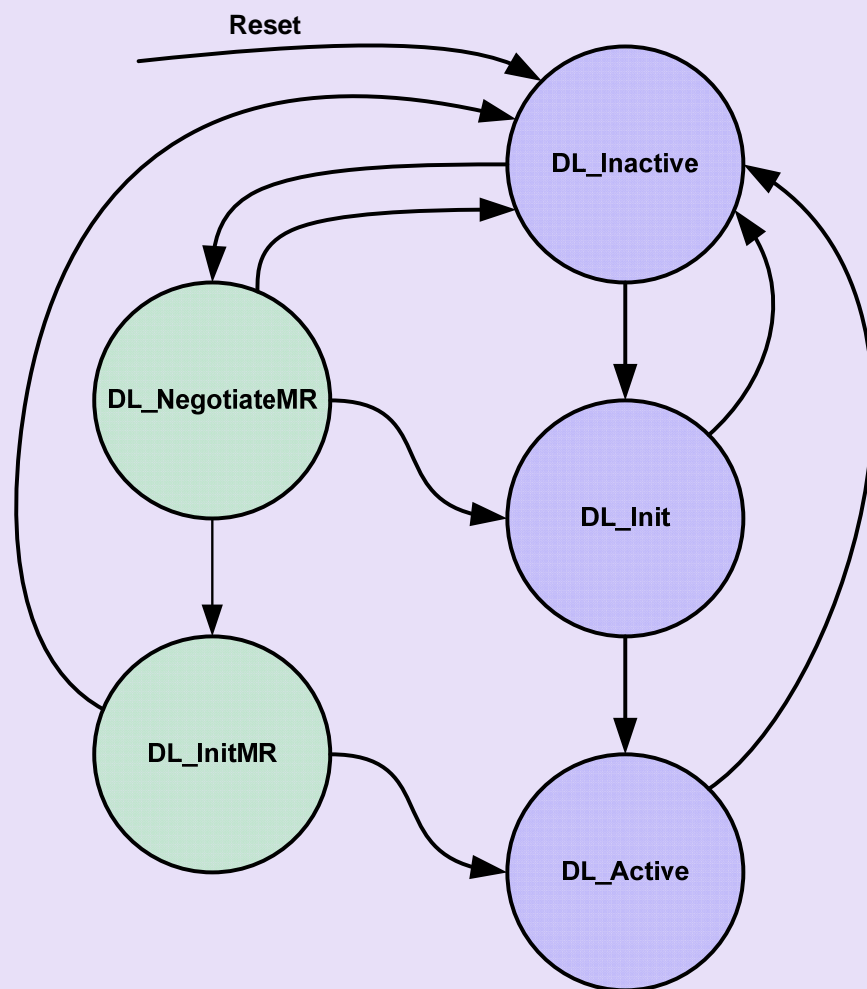
- VLs may be partitioned into a low priority group and a high priority group
  - ✓ Bit vector is used to assign group membership
  - ✓ More flexible than PCIe base which assigns membership sequentially based on Low Priority Extended VC Count
- High priority group
  - ✓ Strict priority arbitration based on VL number (VL0 lowest, VL7 highest)
- Low priority group
  - ✓ Software configurable arbitration mechanism
    - Hardwired fixed (e.g., round robin)
    - Weighted Round Robin (WRR) with 32, 64, 128 or 256 phases
    - Time-based WRR with 128 phases
    - Vendor defined arbitration



# VH to VL Arbitration

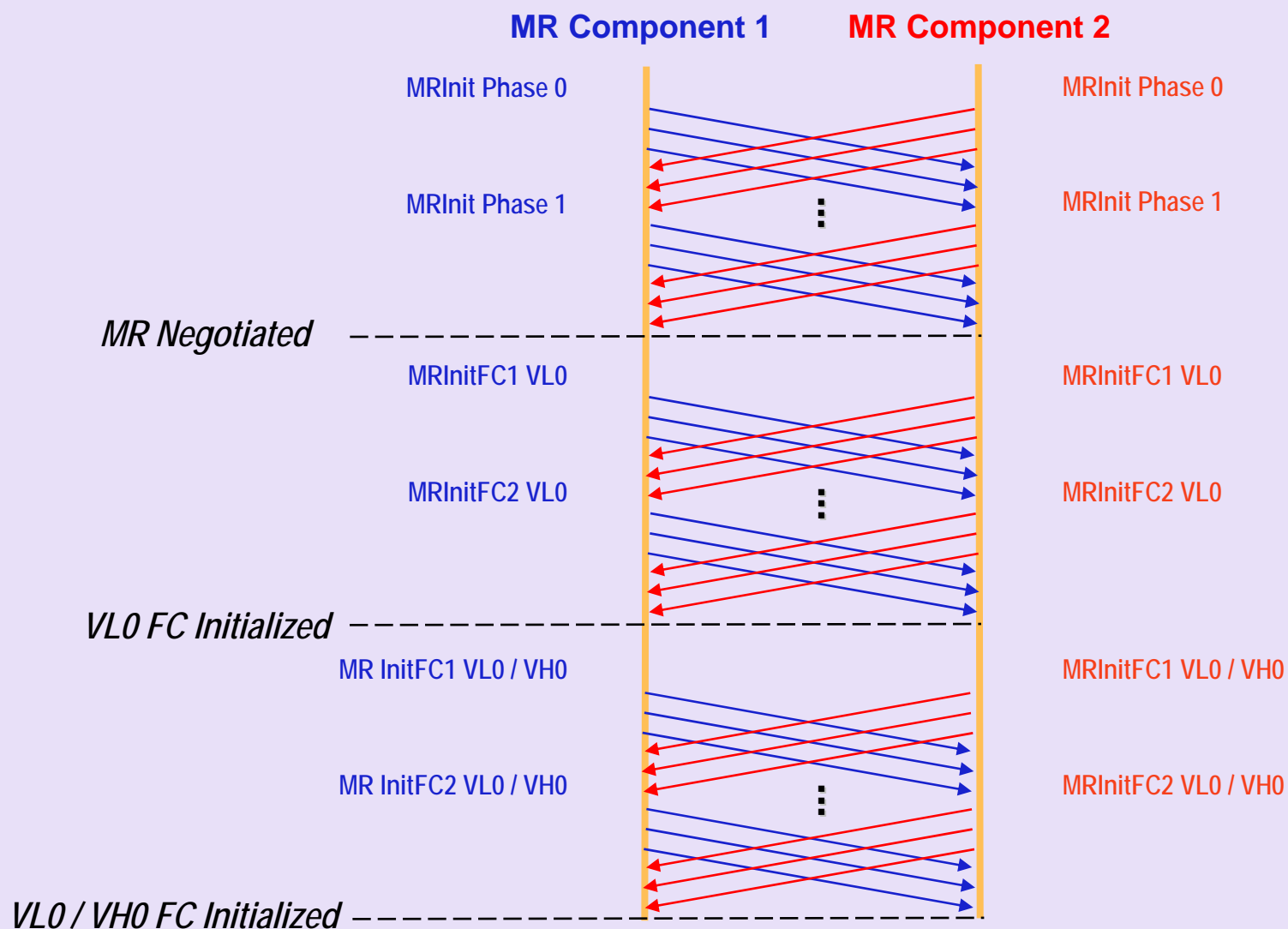
- VH to VL arbitration is associated with each VL supported by a port
- Not software configurable
  - ✓ An implementation must support a hardwired fixed arbitration scheme that guarantees forward progress

# MR Link Initialization Protocol and Flow Control Initialization

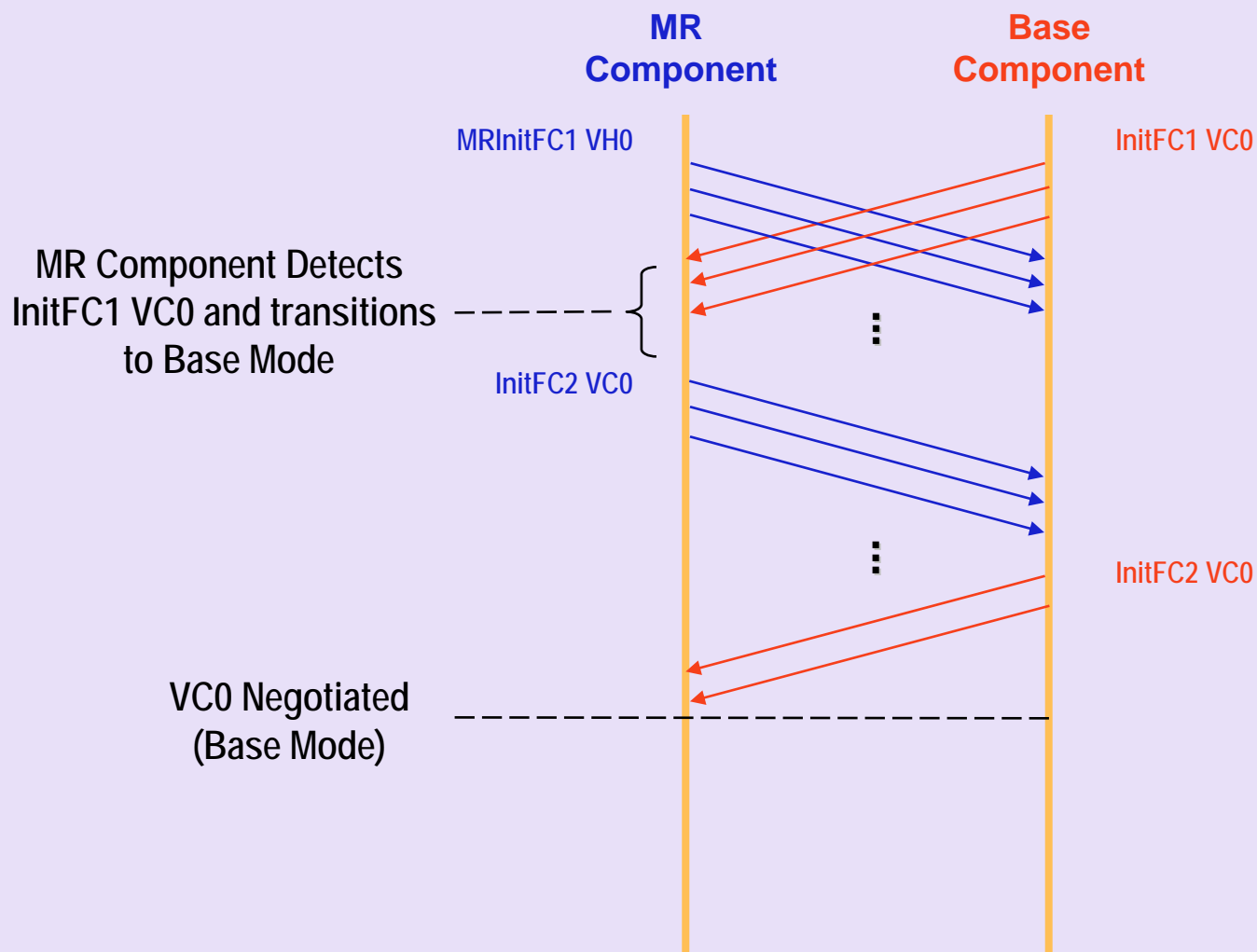


- Link initialization protocol used to determine if link will use MR protocol or PCIe Base protocol
- MR links autonomously negotiate default virtual link (VL0) and default virtual hierarchy (VH0)
  - ✓ Similar to VC0 in PCIe Base

# MR to MR Initialization



# MR to Base Initialization





# Information Exchanged during MR Link Initialization

	+0								+1								+2								+3										
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			
Byte 0	Type 0000 0001b								Phase	Vendor Specific Init				Device / Port Type				Auth	Version 001b				0	MaxVL				MaxVH							
Byte 4	16-bit CRC																																		

## MRInit DLLP Format

<b>Phase</b>	<b>MR link initialization phase (i.e., Phase 0 or Phase 1)</b>
<b>Vendor Specific Init</b>	<b>Vendor specific software value transmitted by switches and captured by Switches and Devices</b>
<b>Device / Port Type</b>	<b>Indicates the device/port type of the sender. Same encoding as that of the Device / Port Type field in the PCI Express® Capabilities Register defined in PCIe Base.</b>
<b>Authorized</b>	<b>For switches, this bit is set to indicate that the sender is an Authorized port on an MR Capable switch</b>
<b>Version</b>	<b>Must be 001b in this version of the specification</b>
<b>MaxVL</b>	<b>Maximum number of Virtual Links supported by sender</b>
<b>MaxVH</b>	<b>Maximum number of Virtual Hierarchies supported by the sender.</b>

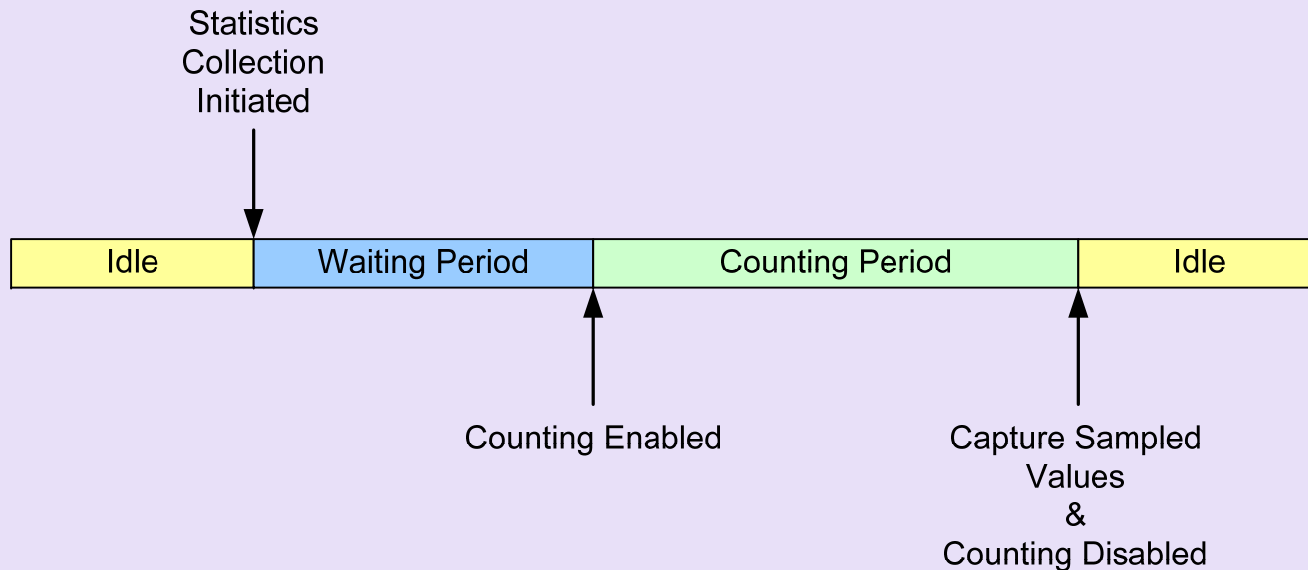
# Performance Monitoring

- MR congestion may result in traffic associated with one VH affecting the performance of unrelated VHs
- Performance monitoring capabilities are optional capabilities that may be used to diagnose, plan, and tune the performance of a multi-root system
  - ✓ Allows device vendor independent software to monitor performance and manage congestion
- Goals and objectives
  - ✓ It is a goal to standardize MR performance monitoring capabilities
  - ✓ It is not a goal to monitor or count errors
  - ✓ It is not a goal to monitor performance within a VH (i.e., SR performance)
  - ✓ It is a goal to provide an extensible framework for vendor specific performance monitoring capabilities and future enhancements

# Statistics Collection

- Captured statistics
  - ✓ Counted values
    - Number of occurrences of a selected event over a sampling period
    - 64-bit max counter (may implement fewer bits)
  - ✓ Sampled values
    - Snapshot of system state
- Initiated by writing to a configuration space register

# Statistics Collection



- Waiting period
  - ✓ Range 0 to 65 ms in units of microseconds (16-bit time value)
- Sampling period
  - ✓ Range 0 to 16 s in units of microseconds (24-bit time value)

# Standardized Counted Values

- Counted events
  - ✓ Number of transmitted TLPs
  - ✓ Number of received TLPs
  - ✓ Number of transmitted TLP DWORDS (from and including STP to END)
  - ✓ Number of received TLP DWORDS (from and including STP to END)
  - ✓ Number of transmitted Idle characters
  - ✓ Number of received Idle characters
- Optional filtering based on VH, VL, and TLP type (P, NP, CP)
  - ✓ Number of symbol times a TLP is blocked from transmission due to
    - Lack of VL flow control credits
    - Lack of VH flow control credits
    - Lack of VL or VH flow control credits
  - ✓ Number of transmitted DLLPs of any type
  - ✓ Number of received DLLPs of any type

# Standardized Sampled Values

- Sampled values are optional
  - ✓ The number of available transmit credits of a particular type\* associated with a VL computed as
$$(\text{CREDIT\_LIMIT} - \text{CREDITS\_CONSUMED}) \bmod 2^{\text{Field Size}}$$
  - ✓ The number of available transmit credits of a particular type\* associated with a VH computed as
$$(\text{CREDIT\_LIMIT} - \text{CREDITS\_CONSUMED}) \bmod 2^{\text{Field Size}}$$
  - ✓ The number of available receive credits of a particular type\* associated with a VL computed as
$$(\text{CREDITS\_ALLOCATED} - \text{CREDITS\_RECEIVED}) \bmod 2^{\text{Field Size}}$$
  - ✓ The number of available receive credits of a particular type\* associated with a VH computed as
$$(\text{CREDITS\_ALLOCATED} - \text{CREDITS\_RECEIVED}) \bmod 2^{\text{Field Size}}$$

\*Type corresponds to (P, NP, Cpl) x (Header, Data)



# Summary

- Virtual Links
  - ✓ Used to manage independent flows with different QoS characteristics
  - ✓ Similar to VCs in PCIe Base
- Bypass Queues
  - ✓ Used to reduce/eliminate effects of congestion within a VL between VHs
  - ✓ Wide variety of implementation options
    - From a single queue to full VH isolation
- Logical extension of PCIe Base
  - ✓ Similar transmitter and receiver flow control behavior
  - ✓ Similar FCP format
  - ✓ Similar arbitration algorithms
- Standardizing optional performance monitoring capability
  - ✓ Allows device vendor independent software to monitor performance and manage congestion



Thank you for attending the  
PCI-SIG Developers Conference  
2007.

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