Performance Metrics

PEX 8608/8609: 8-Lane, 8-Port Gen 2 PCIe Switch

Introduction

This white paper discusses measures of performance for the PEX 8608/8609, including throughput and latency. It also provides guidelines for programming on-chip registers, to boost performance beyond that provided by the general-purpose default values.

This white paper also differentiates *PCI Express Base r1.1* (Gen 1) rates, 2.5 Giga-Transfers/second (GT/s), from *PCI Express Base r2.0* (Gen 2) rates, 5.0 GT/s, by including “Gen 1” or “Gen 2”, as appropriate, after the xWidths (*for example*, “x4 Gen 1” indicates x4 at 2.5 GT/s, and “x4 Gen 2” indicates x4 at 5.0 GT/s).

Throughput

Throughput measures the amount of Payload bytes transferred per unit time. PCI Express has various possible throughput values, depending upon the Link width, Payload size, traffic distribution, and Transaction Layer Packet (TLP) overhead, all of which are under software control. To comprehend PCI Express throughput, a basic understanding of the underlying PCI Express fundamentals is necessary.

Shared Wire

Bytes are transmitted across PCI Express wires during each symbol time, regardless of traffic load. The bytes are classified into three wire traffic types:

- Transaction Layer Packets (TLPs) that carry Payloads
- Data Link Layer Packets (DLLPs)
- Physical Layer (PHY) Ordered-Sets

Electrical idles (including PADs) are not counted as traffic. To measure throughput and understand how the Link performs, count all three wire traffic types while tracking the amount of time that elapses. PHY SKIP Ordered-Sets occur irregularly and can mostly be ignored. A fully used Link requires 99% TLPs and DLLPs in each direction. The ratio of TLPs to DLLPs depends upon the application.

Unidirectional Throughput

Ideal PCI Express throughput in the unidirectional bandwidth case is illustrated in Figure 1 and defined in Table 1. Figures 4-7 include the best case unidirectional throughput for the link width and speed with a curve labeled **Calc 0 DLLP/TLP**. Payload size is in bytes (B) for each of these figures.

Figure 1: Ideal PCI Express Throughput in Unidirectional Bandwidth Case

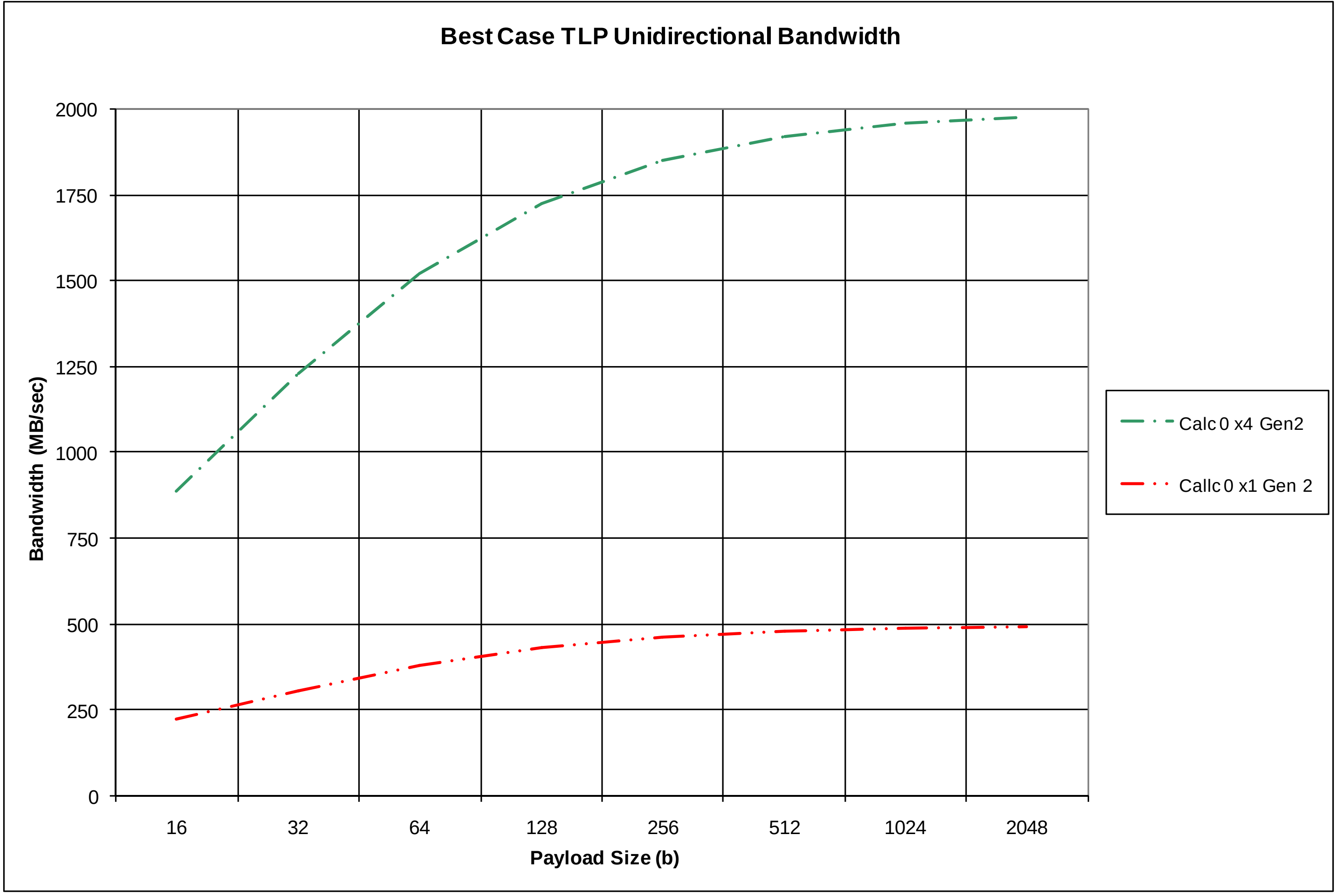


Table 1 illustrates the way in which the maximum throughput increases with larger Payload sizes, and with wider and faster Links. The *PCI Express Base r2.0* allows a default Maximum Payload Size (MPS) of 128 bytes. The PEX 8608/8609, however, supports an MPS of up to 2,048 bytes on x4 port widths, to achieve a bandwidth higher than obtainable with the minimum MPS value. The PEX 8608/8609 MPS is restricted to 512 bytes when the programmed port width is X1.

Unidirectional PCI Express throughput has maximal TLPs on the wire going in one direction. The other direction of the bidirectional Link is mostly unused. DLLPs that share the wire (as per the *PCI Express Base r2.0*) are typically transmitted in response to a TLP; therefore, DLLPs travel in the opposite direction of TLPs. Thus, for unidirectional traffic, DLLP traffic does not interfere with TLP bandwidth.

It is useful to make a clarification regarding PCI Express Memory Read (MRd) Requests and their corresponding Completions with Data (CPLD). The Length field in the MRd Request can be up to 4 KB. The MRd TLP, however, is only 12 to 20 bytes in length. The Completion for the MRd carries the data. Typically, a Root Complex transmits multiple, partial Completions with 64-byte Payload size (endpoint devices must transmit Completions of at least 128-byte granularity). As a result, even with large Read sizes, the bandwidth expected for the Read is limited by the size of the Completions' data Payload.

For example, if a series of MRd Requests are sent in the upstream direction each with a large Read size, and the Completer sends only Completions with 64-byte Payloads, the maximum bandwidth expected would be close to the unidirectional 64-byte Payload data points in Figure 1. (Refer also to the Read Completion Throughput section)

Table 1: Ideal Unidirectional Throughput Numbers (in gigabytes per second)

Payload (Bytes)	Calc 0 x4 Gen 2	Calc 0 X1 Gen 2
16	0.886	0.221
32	1.227	0.307
64	1.519	0.380
128	1.724	0.431
256	1.849	0.462
512	1.918	0.480
1,024	1.955	0.489
2,048	1.974	0.493

Ideal PCI Express Throughput

This section discusses how to calculate ideal PCI Express throughput, as explained in the Unidirectional Throughput section.

The PEX 8608/8609 signaling operates at 2.5 GT/s/Lane (Gen 1) and 5.0 GT/s/Lane (Gen 2). The PEX 8608/8609 allows Lanes to be grouped into x1, and x4 widths. This bandwidth is de-rated, according to the factors described within this section.

PCI Express protocol has built-in 8b/10b encoding, which immediately removes 20% of the throughput:

8b/10b_encoding_hit = 8/10 = 0.8

TLPs include overhead as part of the PCI Express protocol. Each TLP includes a Header of 12 to 16 bytes (16 bytes only for 64-bit addressing; otherwise, TLPs all include 12-byte Headers). TLPs can also have an optional End-to-end Cyclic Redundancy Check (ECRC) of 4 bytes. All TLPs require a Data Link Layer (DLL) and PHY framing symbol overhead of 8 bytes. The total TLP overhead is as follows:

TLP_overhead_min = 12 + 8 = 20 bytes
TLP_overhead_max = 16 + 4 + 8 = 28 bytes

Figure 2: TLP Packet Structure with 20 Bytes of Overhead and 32 Bytes of Data Payload

STP	SEQ	SEQ	HDR
HDR	HDR	HDR	HDR
HDR	HDR	HDR	ADDR
ADDR	ADDR	ADDR	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	DATA
DATA	DATA	DATA	CRC
CRC	CRC	CRC	END

The *PCI Express Base r2.0* requires that DLLPs and SKIP Ordered-Sets share the same wire as TLPs, allowing these other traffic sources to reduce TLP throughput. The best case with the least impact (reduction to TLP bandwidth corresponding to unidirectional traffic scenarios) can be calculated.

To cover lossy Link behavior, the *PCI Express Base r2.0* requires all enabled types of UpdateFC to be transmitted at least once every 30 μs. However, typically UpdateFC FCPs are scheduled for transmission more frequently than this requirement. ACK DLLPs are transmitted to indicate that the receiver has received one or more TLPs reliably. The device that transmitted a series of TLPs will then deallocate those TLPs from its Retry buffer. The *PCI Express Base r2.0* specifies ACK transmission latency limits based on link speeds and payload sizes. It also specifies an AckFactor that represents the number of maximum size TLPs which can be received before an ACK is sent. The frequency of ACK transmissions is chosen to balance the link bandwidth efficiency and Retry buffer size. Each DLLP contains 8 bytes.

Figure 3 illustrates the DLLP Packet structure. A DLLP derating factor can be modeled simply by the number of UpdateFC or ACK DLLP bytes transmitted per each TLP. This factor, representing the decrease in TLP throughput due to DLLP traffic, is then added to the TLP overhead that was described earlier

DLLP_derating = 8 bytes * num_dllp_per_TLP

Figure 3: DLLP Packet Structure (8 Bytes)

SDP	DLLP	DLLP	DLLP
DLLP	CRC	CRC	END

A SKIP Ordered-Set can be modeled as occurring once per 1,180 symbol times. The *PCI Express Base r2.0* provides a range of 1,180 to 1,538 symbol times, and the value used by the PEX 8608/8609 is once every 1,180 symbol times. A SKIP Ordered-Set requires 4 symbol times to transmit. Therefore, throughput is decreased by:

SKIP_derating = (1,180 / 1,184)

The absolute Link Rate would be the number of lanes times the transmission frequency in GT/sec:

link_rate = num_lanes * transmission_frequency

Placing together all the overhead and throughput de-rating, the ideal PCI Express unidirectional bandwidth can be calculated for any Payload, as follows:

Ideal_pcie_unidirectional_bandwidth = link_rate * 8b/10b_encoding_hit * skip_derating
* (payload_size / (payload_size + tlp_overhead + DLLP_derating))

The above formula, using tlp_overhead_min (for the tlp_overhead variable), zero DLLP per TLP, and the appropriate Link rate, was used to calculate the values for the ideal curves illustrated earlier, in Figure 1.

Bidirectional PCI Express Throughput

Although unidirectional flows have virtually no DLLP traffic flowing in the same direction as the TLP, to model bidirectional traffic, DLLPs require prominent consideration. Three different calculated DLLP rates provide a useful reference – 0, 1, and 2 DLLPs per TLP (DLLP/TLP).

The worst case, 2 DLLP/TLP, applies wherein every TLP causes one ACK and one UpdateFC DLLP. The ACK acknowledges the TLP arrived and the UpdateFC provides additional credits, allowing additional TLPs of the same type to be transmitted.

Note: Worst case is approximate. There can be an additional UpdateFC time every 30 us.

The best case, 0 DLLP/TLP, is the unidirectional traffic case, because no DLLPs travel in the same direction as the TLP flow. Table 2 summarizes the amount of Link bandwidth used by the Payload, for various DLLP policies and Payload sizes when using TLPs with the minimum overhead of 20 bytes (32-bit addressing and no ECRC). The values calculated used the following formula:

(data payload bytes / (data payload bytes + overhead bytes + DLLP bytes)) * SKIP_derating * 100%

Table 2. TLP Data Payload Percent of Link Bandwidth Used for Different DLLP Rates[1]

Payload (Bytes)	0 DLLP/TLP (Ideal) (Percent)	1 DLLP/TLP (Percent)	2 DLLP/TLP (Percent)
0	0.00	0.00	0.00
4	16.61	12.46	9.97
8	28.47	22.15	18.12
16	44.29	36.24	30.67
32	61.33	53.15	46.90
64	75.93	69.33	63.78
128	86.19	81.77	77.79
256	92.44	89.84	87.38
512	95.92	94.49	93.12
1,024	97.75	97.01	96.28
2,048	98.70	98.32	97.94

It is expected that in any traffic pattern, a maximally and optimally used Links' throughput will operate somewhere in the range between 0 and 2 DLLP/TLP. Because DLLP and TLP counts are easily measured with standard PCI Express logic analyzers, understanding the DLLP-to-TLP ratio aids in understanding PCI Express Link behavior. If the DLLP Count is more than 2x the TLP Count, the Link is probably underused.

The exact ratio of DLLPs to TLPs depends upon a variety of factors, which to some extent remain outside the *PCI Express Base r2.0* guidelines. Figures 4-9 illustrate the measured PEX 8608/8609 bidirectional throughput, with default register values for their listed Link widths. This bidirectional performance was measured using simulation CAE tools. The theoretical link partner whose TLP traffic is mixed with ACK and UpdateFC DLLPs was a custom bus functional model who's ACK and credit update policies emulate those of a realistic link partner connected to a PEX 8608/8609 port. Tables 3-8 were used to provide the numbers that are plotted in Figures 4-9. As a reference, each figure/graph contains ideal calculated throughput curves for three DLLP policies – 0, 1, or 2 DLLP/TLP. The measured bidirectional curves are the performance points of the PEX 8608/8609 when transmitting sustained back-to-back TLPs of the same size on all ports simultaneously. All ports of the PEX 8608/8609 are also receiving sustained back-to-back TLPs of the same size on all ports simultaneously from the simulated bus functional model.

As illustrated in the following graphs and tables, the PEX 8608/8609 unidirectional cases track along the ideal 0 DLLP/TLP curve, for all values, 16 to 2,048 bytes. The bidirectional measurement is a good indicator of

[1] *The SKIP_derating factor used was (1,180 / 1,184).*

efficient bandwidth management, because the switch must process ACKs, UpdateFCs, and TLP traffic, in both directions.

With default credit and DLLP policy values, the PEX 8608/8609 is able to run bidirectional traffic at better (in some cases, significantly better) than 1 DLLP/TLP rates for Payload sizes of 16 to 512 bytes, and maintain better than 2 DLLP/TLP throughput up to 1024 bytes for x4 Gen 2. For larger Payload sizes, adding additional Payload credits, by tuning the default register values, could possibly increase throughput. Adjusting the credit values, and the factors/impact of doing so, are discussed in more detail in the Ingress Resources section.

By default, for all Link widths, the PEX 8608/8609 operates at better than 1 DLLP/TLP for Payload sizes of 32 to 256 bytes. For larger Payload sizes with some port configurations, the default register values could possibly be fine-tuned, to allow for improved throughput. The DLLP Policies section discusses tuning and consideration factors in further detail.

Figure 4: Measured PEX 8608/8609 Bidirectional Throughput TLP x4 Gen 2 Payload Bandwidth

The following bi-directional performance was measured using back-to-back TLP traffic with simultaneous peer-to-peer ingress source to nearest neighbor egress destination. All ports were operating simultaneously with the TLP traffic stream of each destination port routed entirely from a single source port.

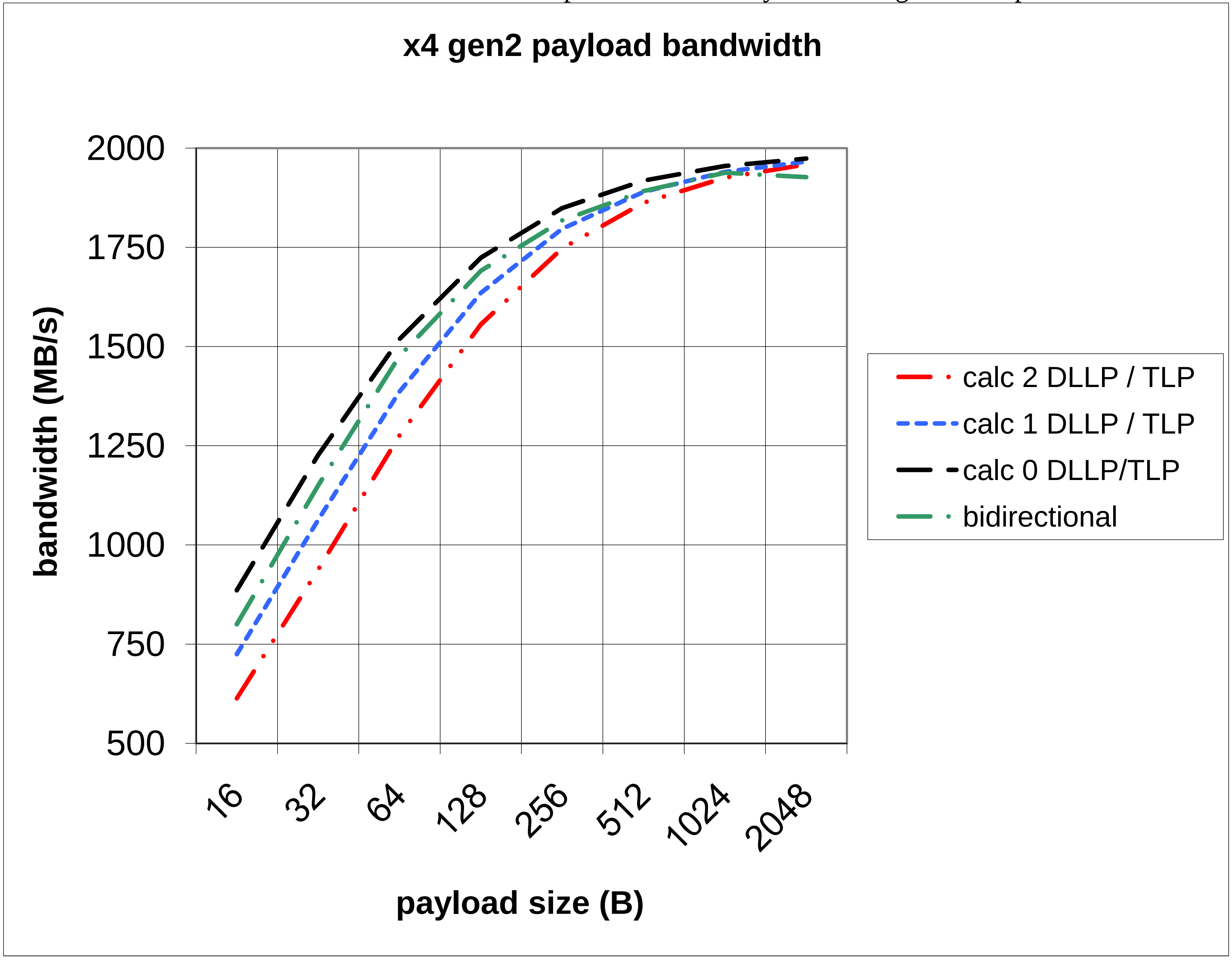


Table 3: x4 Gen 2 Throughput Numbers (in Gigabytes per Second)

Payload (Bytes)	x4 Gen 2 Calculated Throughput for Unidirectional and Bidirectional with Calc 0 DLLP/TLP	x4 Gen 2 Calculated Throughput for Bidirectional with Calc 1 DLLP/TLP	x4 Gen 2 Calculated Throughput for Bidirectional with Calc 2 DLLP/TLP	x4 Gen 2 PEX 8608/8609 Throughput for Bidirectional with Default DLLP/TLP Settings/Behavior
16	0.886	0.725	0.613	0.795
32	1.227	1.063	0.938	1.150
64	1.519	1.387	1.276	1.476
128	1.724	1.635	1.556	1.691
256	1.849	1.797	1.748	1.818
512	1.918	1.890	1.862	1.892
1,024	1.955	1.940	1.926	1.938
2,048	1.974	1.966	1.959	1.927

Figure 5: Measured PEX 8608/8609 Bidirectional Throughput TLP x4 Gen 1 Payload Bandwidth

The following bi-directional performance was measured using back-to-back TLP traffic with simultaneous peer-to-peer ingress source to nearest neighbor egress destination. All ports were operating simultaneously with the TLP traffic stream of each destination port routed entirely from a single source port.

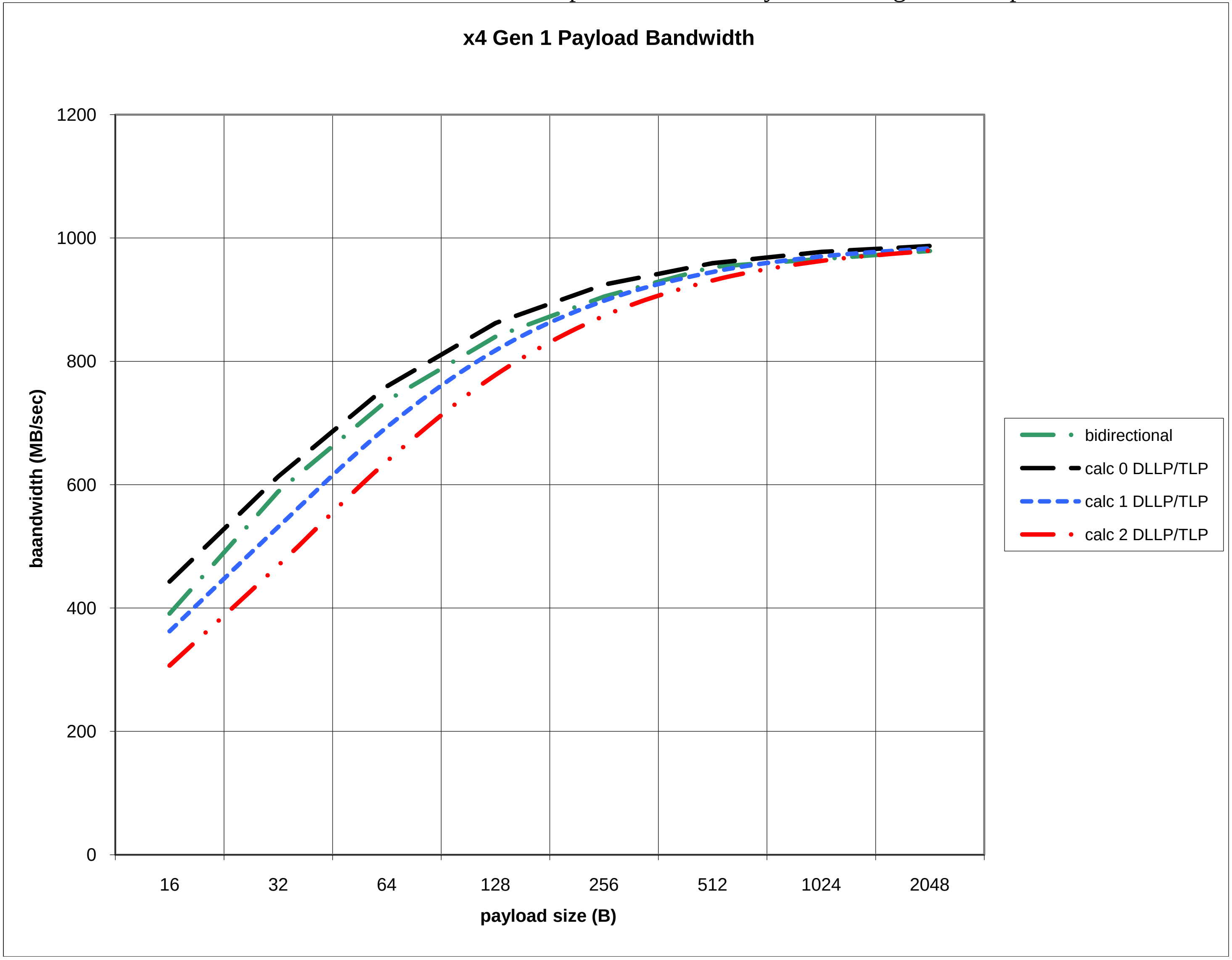


Table 4: x4 Gen1 Throughput Numbers (in Gigabytes per Second)

Payload (Bytes)	x4 Gen1 Calculated Throughput for Unidirectional and Bidirectional with Calc 0 DLLP/TLP	x4 Gen 1 Calculated Throughput for Bidirectional with Calc 1 DLLP/TLP	x4 Gen 1 Calculated Throughput for Bidirectional with Calc 2 DLLP/TLP	x4 Gen 1 PEX 8608/8609 Throughput for Bidirectional with Default DLLP/TLP Settings/Behavior
16	0.443	0.362	0.307	.391
32	0.613	0.532	0.469	.589
64	0.759	0.693	0.638	.736
128	0.862	0.818	0.778	.840
256	0.924	0.898	0.874	.905
512	0.959	0.945	0.931	.953
1,024	0.978	0.970	0.963	.966
2,048	0.987	0.983	0.979	.979

Figure 6: Measured PEX 8608/8609 Bidirectional Throughput TLP x1 Gen 2 Payload Bandwidth

The following bi-directional performance was measured using back-to-back TLP traffic with simultaneous peer-to-peer ingress source to nearest neighbor egress destination. All ports were operating simultaneously with the TLP traffic stream of each destination port routed entirely from a single source port.

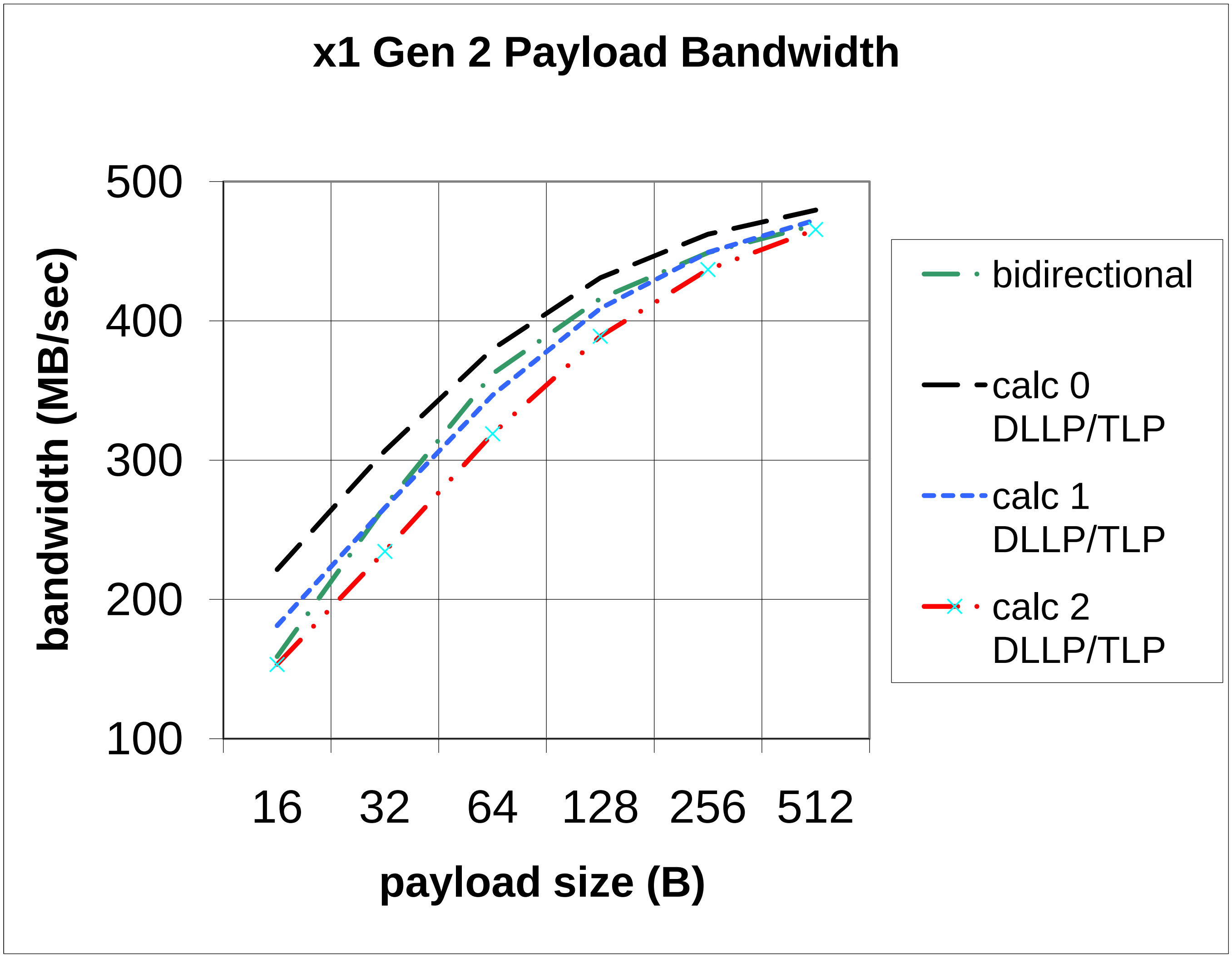


Table 5: x1 Gen2 Throughput Numbers (in Gigabytes per Second)

Payload (Bytes)	x1 Gen2 Calculated Throughput for Unidirectional and Bidirectional with Calc 0 DLLP/TLP	x1 Gen2 Calculated Throughput for Bidirectional with Calc 1 DLLP/TLP	x1 Gen2 Calculated Throughput for Bidirectional with Calc 2 DLLP/TLP	x1 Gen2 PEX 8608/8609 Throughput for Bidirectional with Default DLLP/TLP Settings/Behavior
16	0.221	0.181	0.153	0.159
32	0.307	0.266	0.234	0.267
64	0.380	0.347	0.319	0.362
128	0.431	0.409	0.389	0.416
256	0.462	0.449	0.437	0.449
512	0.480	0.472	0.466	0.469

Figure 7: Measured PEX 8608/8609 Bidirectional Throughput TLP x1 Gen 1 Payload Bandwidth

The following bi-directional performance was measured using back-to-back TLP traffic with simultaneous peer-to-peer ingress source to nearest neighbor egress destination. All ports were operating simultaneously with the TLP traffic stream of each destination port routed entirely from a single source port.

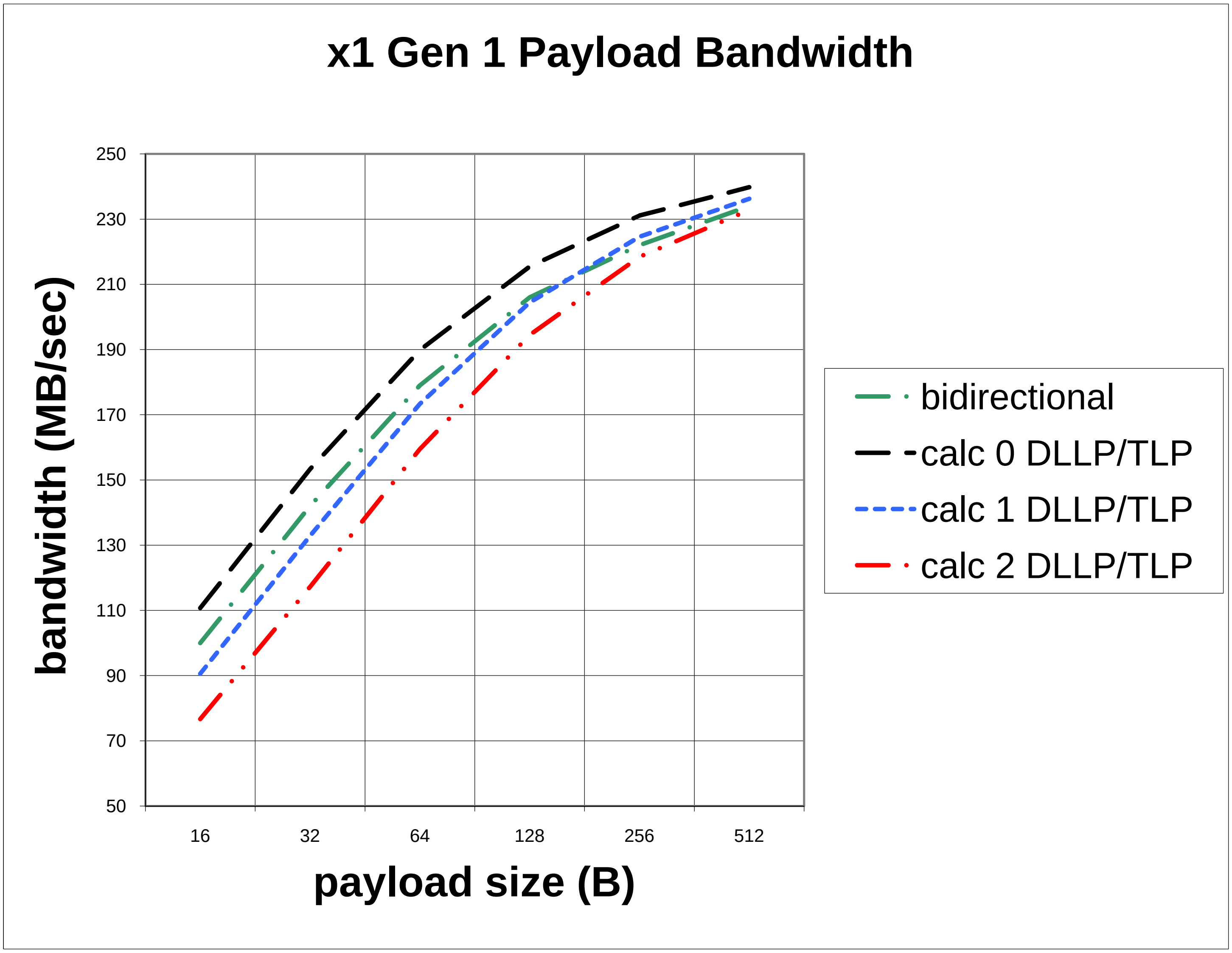


Table 6: x1 Gen 1 Throughput Numbers (in Gigabytes per Second)

Payload (Bytes)	x1 Gen 1 Calculated Throughput for Unidirectional and Bidirectional with Calc 0 DLLP/TLP	x1 Gen 1 Calculated Throughput for Bidirectional with Calc 1 DLLP/TLP	x1 Gen 1 Calculated Throughput for Bidirectional with Calc 2 DLLP/TLP	x1 Gen 1 PEX 8608/8609 Throughput for Bidirectional with Default DLLP/TLP Settings/Behavior
16	0.111	0.91	0.77	0.100
32	0.153	0.133	0.117	0.142
64	0.190	0.173	0.159	0.179
128	0.215	0.204	0.194	0.206
256	0.231	0.225	0.218	0.222
512	0.240	0.236	0.233	0.234

Read Completion Throughput

Read Completion throughput is illustrated in Figures 8-9, using the values listed in Tables 7-8. For the calculated curves Calc 0 and Calc 2, note the following:

- Read Completion Payload size matches the Read Request size
- Read throughput does not include time to forward the Read Request

Figure 8: Measured PEX 8608/8609 Read Completion Throughput x4 Gen 2 Bandwidth

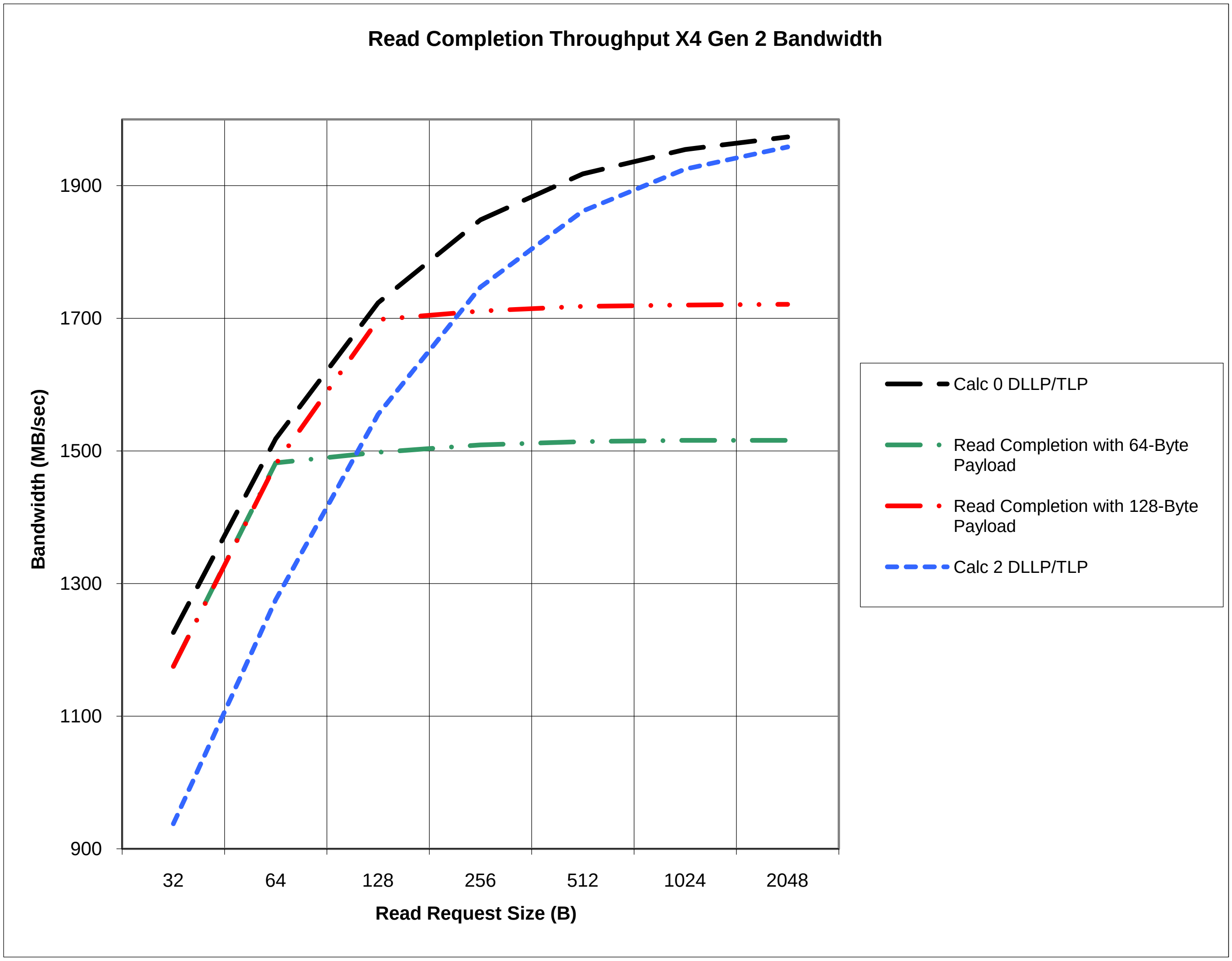


Table 7: Measured x4 Gen 2 Read Completion Throughput (in Gigabytes per Second)

Read Request Transfer Size (Bytes)	X4 Gen 2 Calculated Throughput for Calc 0 DLLP/TLP	X4 Gen 2 Calculated Throughput for Calc 2 DLLP/TLP	X4 Gen 2 PEX 8608/8609 Read Throughput When Receiving Completions Containing 64-Byte Payloads with Default Settings/Behavior	X4 Gen 2 PEX 8608/8609 Read Throughput When Receiving Completions Containing 128-Byte Payloads with Default Settings/Behavior
16	0.886	0.613	–	–
32	1.227	0.938	1.175	1.175
64	1.519	1.276	1.482	1.480
128	1.724	1.556	1.498	1.698
256	1.849	1.748	1.509	1.711
512	1.918	1.862	1.514	1.718
1,024	1.955	1.926	1.516	1.720
2,048	1.974	1.959	1.516	1.721

Figure 9: Measured PEX 8608/8609 Read Completion Throughput x1 Gen 2 Bandwidth

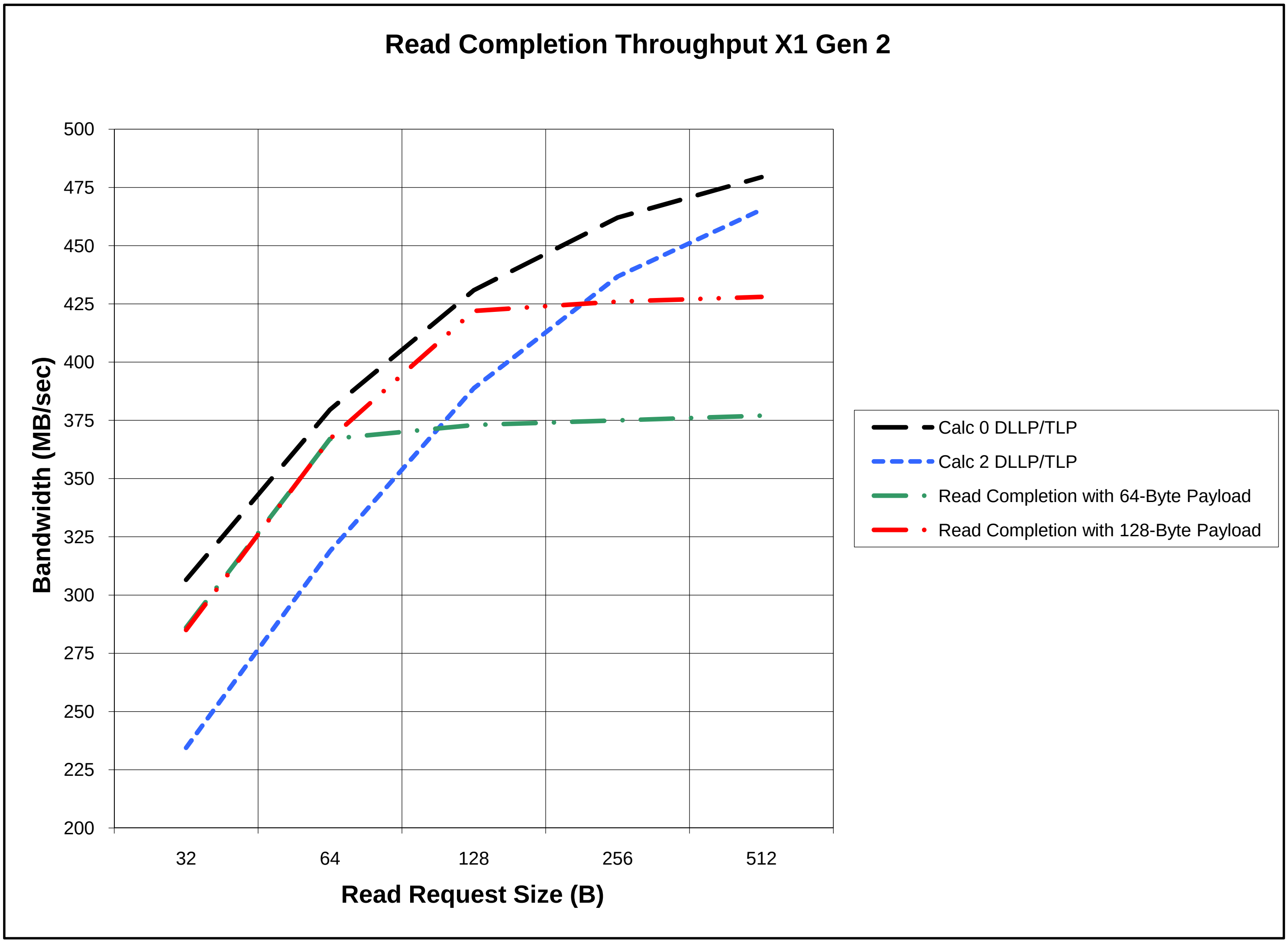


Table 8: Measured x1 Gen 2 Read Completion Throughput (in Gigabytes per Second)

Read Request Transfer Size (Bytes)	X1 Gen 2 Calculated Throughput for Calc 0 DLLP/TLP	X1 Gen 2 Calculated Throughput for Calc 2 DLLP/TLP	X1 Gen 2 PEX 8608/8609 Read Throughput When Receiving Completions Containing 64-Byte Payloads with Default Settings/Behavior	X1 Gen 2 PEX 8608/8609 Read Throughput When Receiving Completions Containing 128-Byte Payloads with Default Settings/Behavior
16	0.221	0.153	–	–
32	0.307	0.234	0.286	0.285
64	0.380	0.319	0.367	0.367
128	0.431	0.389	0.373	0.422
256	0.462	0.437	0.375	0.426
512	0.479	0.465	0.377	0.428

DLLP Policies

As previously discussed, DLLP rates can vary from 0 to 2 or more DLLPs/TLP. The PEX 8608/8609 allows programming to affect the DLLP rate. Figures 4 through 9 illustrate that an increase in DLLPs reduces the total TLP throughput. Therefore, for designs that require high performance, it would be beneficial to minimize DLLP rates. Transmitting fewer DLLPs, however, can result in credit starvation or Replay buffer filling, which can have a detrimental effect on TLP bandwidth. Care must be taken when changing the default PEX 8608/8609 DLLP transmission rate.

Typically, TLPs have higher transmission priority on the wire than DLLPs. The PEX 8608/8609, however, allows DLLPs to have higher priority under certain conditions, meaning that DLLPs can transmit before starting a new TLP. The decision to transmit a DLLP ahead of a TLP is referred to as *DLLP policy*.

The PEX 8608/8609 can be programmed to alter its default DLLP policies, to emphasize improved TLP throughput, faster acknowledgement, more credit, or simplest behavior. The PEX 8608/8609 default policies were designed to achieve optimal performance for most applications. Programmable choices for a DLLP policy, however, allow for further optimization.

ACK DLLP Policy

An *ACK DLLP* is a response indicating to the TLP Transmitter that the Receiver received a “good” copy of the TLP, meaning that it acknowledged the receipt of the TLP. The simplest policy is to send 1 ACK for every received TLP, resulting in a 1 DLLP/TLP rate for ACK alone. What an ACK means to the TLP Transmitter is that the TLP Transmitter can remove any stored copy of that TLP, because it is unnecessary to resend the TLP. ACK DLLPs can be combined, so that one ACK DLLP can serve to acknowledge multiple TLPs. This collapsing of ACKs is the basis of the ACK DLLP policy choices. Less-frequent, more-collapsed ACKs have the least impact on TLP transmit bandwidth, meaning that less-frequent ACKs result in less than 1 DLLP/TLP.

The PEX 8608/8609 ACK policy consists of two parts – a Timer and TLP Counter. The default ACK Timer policy/value varies according to the negotiated link width, operating link speed, and maximum packet size, as recommended in the *PCI Express Base r1.1* or *PCI Express Base r2.0*, respectively. The numbers in Table 9 define some of the possible default values, in symbol times. Note that these have been adjusted to be different than the *PCI Express Base* values to compensate for hardware specific latencies.

Table 9: Sample PEX8608/8609 ACK Latency Timer Values

Maximum Payload Size	X1 Gen 2 (Symbol Times)	X4 Gen 2 (Symbol Times)
128 bytes	288	124
256 bytes	466	169
512 bytes	610	205

The ACK Transmission Latency Timer loads the appropriate value when a TLP is received and known to be good, meaning a few clocks after the END framing symbol is received. The Timer counts down each symbol time (every 4 ns (*PCI Express Base r1.1*) or 2 ns (*PCI Express Base r2.0*)). When the Timer reaches 0, an ACK DLLP takes higher priority over new TLPs (*that is*, an ACK DLLP is transmitted before a new TLP is started). The ACK DLLP transmitted acknowledges all TLPs, up to the most recently arrived good TLP.

The TLP Counter counts down on each TLP arrival until it reaches zero and then schedules a high-priority ACK DLLP. The default initialization value for the TLP Counter is 16, meaning a high-priority ACK is scheduled upon the arrival of 16 TLPs. The **Ingress Control Shadow** register *ACK TLP Counter Timeout* field (Port 0, and also NT Port Virtual Interface if Port 0 is the NT Port, offset 664h[10:9]) controls the ACK TLP Counter. The default value of 00b allows 16 TLPs before a high-priority ACK. A value of 01b allows 8 TLPs per ACK, a value of 10b allows 4 TLPs per ACK, and a value of 11b disables the Counter.

Either the Timer or the TLP Counter mechanism can cause a high-priority ACK DLLP to be scheduled, and the first one to do so, re-initializes both mechanisms to their stating parameters. For example, the time for 16 TLPs can be less than the ACK Timer above, in which case an ACK is sent earlier. The TLP Counter is useful for any system with a large programmed MPS (resulting in a large timer value), that is capable of sending short TLPs, *such as* 12-byte Memory Reads. Rather than require the Transmitter to save possibly 100+ small TLPs, it need only save 16, plus whatever else arrives in the round-trip time.

If there is no TLP traffic being transmitted (*that is*, the Transmit Link is idle), an ACK DLLP can be transmitted immediately, before the Timer expires. This is an opportunistic low-priority ACK because it does not contend with a TLP in transmission. When an opportunistic low-priority ACK is transmitted, both the Latency Timer and TLP Counter re-initialize, waiting for a new TLP to arrive to begin counting again.

The PEX 8608/8609 allows a programmable override of the default Ack_Latency_Timer value, on a per Port basis, by programming the **ACK Transmission Latency Limit** register *ACK Transmission Latency Limit* field (offset 1F8h[11:0]). The value in this register is loaded when a new TLP arrives and a high-priority ACK DLLP is attempted when the Timer reaches 0. For fastest ACK response, this Timer can be programmed to 000h, resulting in one DLLP ACK transmitted immediately per each TLP received. For less impact on Transmit TLP bandwidth, a larger value can be programmed, resulting in less-frequent ACKs.

In general, a slower ACK response does not impact the Receive TLP stream, and aids the TLP Transmit stream. Every PCI Express device contains storage (Retry buffer) for storing TLPs while waiting for ACKs. The amount of Retry buffer storage a device contains is vendor-dependent. The number of TLPs the PEX 8608/8609 can store depends upon the type and size of TLPs received (Refer to the Ingress Resources section). The PEX 8608/8609 holds TLPs in the Retry buffer while waiting for an ACK. At some point, if the Retry buffer storage fills, then no new TLPs can be sent until a new received ACK frees up space. In this case, the ACK can become a performance bottleneck.

UpdateFC DLLP Policy

An *UpdateFC DLLP* is transmitted in response to a received TLP. The UpdateFC DLLP replenishes the connected device with additional credit, to allow the Transmitter to transmit more TLPs of that type. Each TLP that arrives consumes credit, and eventually, a stream of TLPs consumes all the available credit, unless an UpdateFC DLLP provides additional credit. However, if the connected device has sufficient credit to transmit more TLPs, it is not necessary to transmit UpdateFC DLLPs to it. The UpdateFC policy determines how and when to transmit an UpdateFC DLLP.

There are two parts to the UpdateFC policy – frequency of transmitting the updates and credit amount. This section discusses only the frequency. Refer to the Ingress Resources section for details regarding credit amounts.

If the PEX 8608/8609 is not transmitting TLPs (*that is*, the Transmit Link is idle), and credit to replenish the credit used becomes available, the PEX 8608/8609 immediately transmits an UpdateFC DLLP to the connected device. This is an opportunistic, low-priority UpdateFC DLLP.

However, if the PEX 8608/8609 is busy transmitting TLPs to the connected device, the switch does not transmit an UpdateFC DLLP until a programmed threshold is crossed. The PEX 8608/8609 provides four threshold options – 100%, 75% (default), 50%, and 25%. Whenever the remaining credit drops below the programmed threshold, an UpdateFC DLLP is given high priority, meaning that the UpdateFC DLLP is transmitted before a new TLP is started. There is a separate threshold for Header and Payload credits, for each TLP type – Posted, Non-Posted, and Completion – for each Port, and each VC located in the **Ingress Credit Handler (INCH) Threshold** registers (Ports 0-1, 4-9, , and also NT Port Virtual Interface if Port 0 is the NT Port, offsets A00h through A5Ch).

The example of UpdateFC options (provided in Table 10) chart how, for the various options, an UpdateFC is triggered. This example is for a traffic stream of six back-to-back, 256-byte Posted TLPs, using an x4 Port, where the maximum Posted Header credit is 25 and the maximum Posted Payload credit is 128. A 256byte Payload requires 16 credits (1 credit per 16 bytes). Therefore, each TLP in this case consumes 1 Header and 16 Payload credits.

Once a high-priority UpdateFC is triggered, if there are sufficient on-chip resources to do so, the running credit deficit is fully restored. For most non-congested applications, it is likely that ample chip resources will exist, to fully restore credit with every UpdateFC. However, if resources are running low, only a portion of the running credit is restored. If the threshold for transmitting an UpdateFC remains crossed, then, as more resources become available, a subsequent DLLP is transmitted until the deficit is satisfied.

Selecting the 100% policy results in a high-priority UpdateFC for every TLP received. By itself, this policy results in 1 DLLP/TLP, without factoring in the ACK policy. The 75% policy triggers 1 DLLP for every 2 TLPs for this traffic load, which results in 0.5 DLLP/TLP without the ACK. The 50% policy results in 0.25 DLLP/TLP, and the 25% policy results in 0.16 DLLP/TLP.

Table 10: Example UpdateFC Options

TLP Received	Running Credit Header, Payload Consumed/Total	25% Remains Triggers when 6 Header or 32 Payload Credits Remain	50% Remains Triggers when 12 Header or 64 Payload Credits Remain	75% Remains Triggers when 18 Header or 96 Payload Credits Remain	Less than 100% Remains Update ASAP
TLP0	24/25, 112/128	–	–	–	UpdateFC
TLP1	23/25, 96/128	–	–	UpdateFC	UpdateFC
TLP2	22/25, 80/128	–	–	UpdateFC	UpdateFC
TLP3	21/25, 64/128	–	UpdateFC	UpdateFC	UpdateFC
TLP4	20/25, 48/128	–	UpdateFC	UpdateFC	UpdateFC
TLP5	19/25, 32/128	UpdateFC	UpdateFC	UpdateFC	UpdateFC

Unidirectional DLLP Policies

For unidirectional ingress traffic, the PEX 8608/8609 DLLP policies allow the most-frequent DLLPs, because DLLPs do not interfere with any egress TLPs. (DLLPs flow in the opposite direction of TLPs)

The PEX 8608/8609 can transmit a DLLP ACK almost immediately upon receiving and verifying a TLP. A faster ACK results in fast Transmitter de-allocation of the TLP, and can therefore allow a shallow TLP Replay buffer. The default values can be overwritten, to increase or decrease the ACK DLLP rate. For unidirectional traffic, a small number, *such as* 1, is recommended.

The number programmed into the **ACK Transmission Latency Limit** register *ACK Transmission Latency Limit* field (offset 1F8h[11:0]) sets the ACK Transmission Latency Timer, to count the number of symbol times after receiving a TLP, before transmitting an ACK.

Similar to the ACK programmability, the PEX 8608/8609 can immediately transmit an UpdateFC after receiving only the TLP Header. By transmitting an UpdateFC earlier, the total credit advertised can be minimized. For large Payloads (*such as* 1,024 and 2,048 bytes), reserve PEX 8608/8609 resources only as necessary. By programming fewer credits and having a fast UpdateFC policy, the system does not run out of credits and the PEX 8608/8609 does not waste buffer space on reservations that do not arrive. The following are the recommended settings for a unidirectional port:

- Set the UpdateFC policy for unidirectional ingress traffic to 100%
- Set the port’s initial advertised credits to be sufficient to accept 3 to 4 TLPs of the targeted payload size. This may free up more initial credits that could be re-assigned to other bidirectional ports.

Ingress Resources

The PEX 8608/8609 manages ingress credit on an even and odd port basis. There are separate credit handler engines for even ports and odd ports. An internal hardware processing partition of even ports or odd ports is referred to as a “Half station.” For each Half station, there are two central resources of on-chip ingress credit RAM – Header and Payload. The total Half station Header RAM size is 256 Header credits and the total Half station Payload RAM size is 2048 Payload credits. A Header credit reserves 4 dwords of RAM storage regardless of a 3DW or 4DW actual header size. By design, a PEX 8608/8609 Half station reserves 12 Header credits and 32 Payload credits out of the RAM totals for special hardware-specific storage needs.

The STRAP_PORTCFG*x* balls configure the number of Ports, and the width of each Port. At initialization, the PEX 8608/8609 optimally assigns the credits, based upon the selected Port configuration. This assignment has been carefully designed so that all ports of a given configuration can sustain back-to-back bidirectional traffic without experiencing backpressure due to starvation of ingress credits. Note that the maximum allowable payload size may be limited on some smaller port width configurations in order to achieve this continuous bandwidth.

Header RAM stores TLP Headers. This means that every Header credit advertised reserves one Header RAM location, and so every TLP received by the PEX 8608/8609 uses up one Header RAM location. After subtracting the reserved operational Half station Header RAM locations there remains a net total of 244 header RAM entries available.

Payload RAM stores TLP Payload. A Payload credit unit is 16 bytes. Of the 2016 user-configurable entries, 4 credits must be allocated for each Posted and Completion Header credit advertised. These credits are used internally for linking Posted and Completion TLP Payload to their respective Header.

Every Port receives and transmits the following three traffic types (packets):

- Posted (P)
- Non-Posted (NP)
- Completions (Cpl)

Each Port may be assigned to one or two Virtual Channels (VC0 or VCX):
Each traffic type (P, NP, and Cpl) and Virtual Channel (VC0, VCX) for each Port needs credit. The **INCH Threshold** register bit fields (refer to Table 14) allocate the credit to be reserved (and advertised) for each Port, VC, and traffic type. The credits allocated for each of the three traffic types and two possible VCs for each Port, remain dedicated to that Port, VC, and traffic type.

As TLPs arrive, they are stored in the PEX 8608/8609, until an ACK is received from the final destination. Before the ACK is received, each TLP stored consumes credit, and continues to occupy RAM until released. The RAM/credit is released only after the Receiver has acknowledged to the Sender the arrival of the TLP without any errors, per ACK/NAK policy. The total number of TLPs stored, but not yet forwarded to and acknowledged by, the receiving PCI Express device, depends upon congestion and the ACK policy of that PCI Express device.

There are trade-offs between the number of credits allocated for a particular traffic type and Port combination, perhaps more for one system configuration than another. To alleviate these trade-offs, the PEX 8608/8609 contains an innovative Dynamic Buffering design that allows a programmable-sized portion of the RAM to store any of the three traffic-type TLPs from any Half station Port. The credits that remain after allocating credits for each of the three traffic types and VC for each Port, become part of a *Dynamic buffer*. The Dynamic buffer is essentially a common pool of Half station credits, and is discussed in detail in the Dynamic Buffering section and Ingress Credit Handler Threshold Registers.

The PEX 8608/8609 default credit allocation values, which create a variable sized Dynamic buffer for each of the possible Port configurations, are optimal for most applications. They do not normally need to be re-programmed. Detailed tables of the default initial credit allocation for all three TLP types, and an explanation of the common credit pool, are addressed in the following sections.

Initial Credit Allocation

The PEX 8608/8609 default credit allocation values depend upon the strapped Port width, not the negotiated Port width. The initial credit values that the Initialization Flow Control (InitFC) DLLP advertises on a per-Port basis, which the PEX 8608/8609 transmits after Linkup, are listed in Table 11 (in credits) and Table 12 (in bytes).

The amount of credit that a Port initially advertises is controlled by the **INCH Threshold** register field settings (Ports 0-1, 4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, offsets A00h-AECh. (Refer to the Ingress Credit Handler Threshold Registers section in this document). The default value of these registers changes, depending upon the STRAP_PORTCFG*x* ball values. Because of the way the hardware links the default **INCH Threshold** register credit values into actual reserved RAM storage, in some cases one or two additional initial Payload credits will be allocated out of the Half station common pool. If this occurs, the Port will advertise this additional initial credit in its InitFC DLLPs.

Note that the default initial credits for VCX are all infinite. If a second VC is actually enabled, the user must calculate and program the desired optimal VCX initial credit values. These values should be programmed into the **INCH Threshold** register fields via an EEPROM.

Table 11: Initial Port Credit Allocation in credit units (*where* 1 Header credit means storage is available for one Header of any size, and 1 Payload Credit = 16 bytes)

Configured Port Width	Posted Header/Payload	Non-Posted Header/Payload	Completion Header/Payload
Virtual Channel X			
x4	infinite/infinite	infinite/infinite	infinite/infinite
x1	infinite/infinite	infinite/infinite	infinite/infinite
Virtual Channel 0			
x4	26/256	26/infinite	26/224
x1	7/64	7/infinite	5/64

Table 12: Initial Port Credit Allocation for Header/Payload in byte units

Configured Port Width	Posted Header/Payload	Non-Posted Header/Payload	Completion Header/Payload
Virtual Channel X			
x4	infinite/infinite	infinite/infinite	infinite/infinite
x1	infinite/infinite	infinite/infinite	infinite/infinite
Virtual Channel 0			
x4	416/4096	416/infinite	416/3584
x1	112/1024	112/ infinite	80/64

Dynamic Buffering

The PEX 8608/8609 default credit values are optimal for most applications, to maintain back-to-back TLP traffic indefinitely, without running out of credit. After any of the initial credit (storage space) is used, more resources are automatically made available from the Half station Dynamic buffer, to maintain the initial credit allotment. These additional resources taken from the Dynamic buffer are not reserved ahead of time. Therefore, they can be used for either Virtual Channel or any of the three TLP types – Posted, Non-Posted, or Completion (P, NP, or Cpl, respectively). Because the TLP type is not pre-specified, these extra resources are termed a *common credit pool*.

When credit is actually replenished depends upon the UpdateFC DLLP policy (refer to UpdateFC DLLP Policy), which is controlled by setting the **INCH Threshold** register *FC Update High-Priority Threshold* fields (Ports 0-1, 4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, offsets A00h-AECh, [19:18, 17:16], as appropriate). These thresholds are relative to the initial credits allocated to a Port. Common Pool credits that are allocated to a Port before a high-priority UpdateFC DLLP is sent can be de-allocated if the Port’s initial credits are restored before the UpdateFC DLLP is sent.

The common credit pool for Header and Payload credit is as follows:

- **Common Header pool –** What remains in the Header RAM space after subtracting the advertised Header credits of each Half station Port for the three TLP types, plus one or two Virtual Channels.

Figure 10 illustrates the way in which the initial allocation of the PEX 8608/8609 Header RAM is partitioned (by default) for a Half Station configured as a single X4 width port with only VC0 in use.

- **Common Payload pool (Common Payload/Completion pool) –** What remains after subtracting the following from the Payload RAM:
  - 4 credits for each Posted and Completion Header Credit advertised, for each Half station Port.
  - Posted (Payload) credits advertised for each Virtual Channel and Half station Port.
  - Completion credits advertised for each Virtual Channel and Half station Port.

Figure 11 illustrates the way in which the PEX 8608/8609 Payload (and Completion) RAM is partitioned (by default) for a device configuration of one Half Station with a single X4 width port.

Figure 10: One X4 Port Half Station Default Configuration of Header RAM (256 Total Credits)

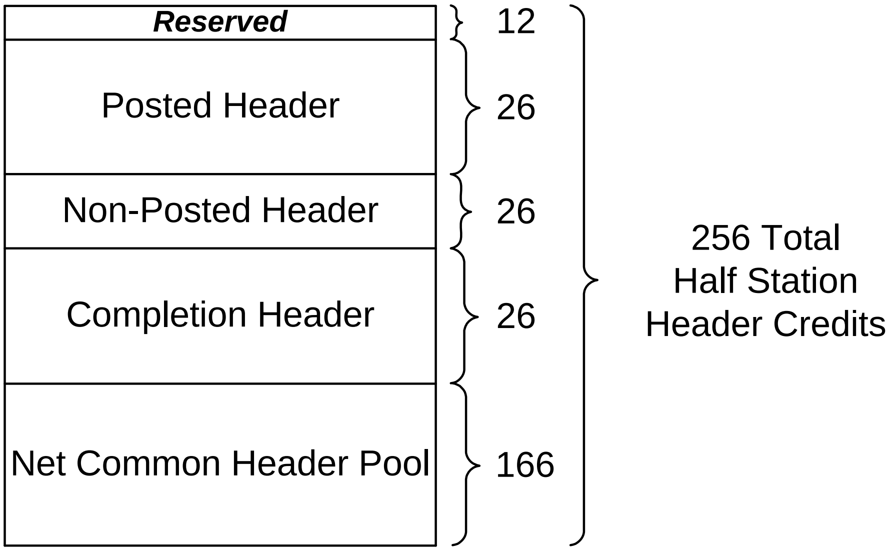
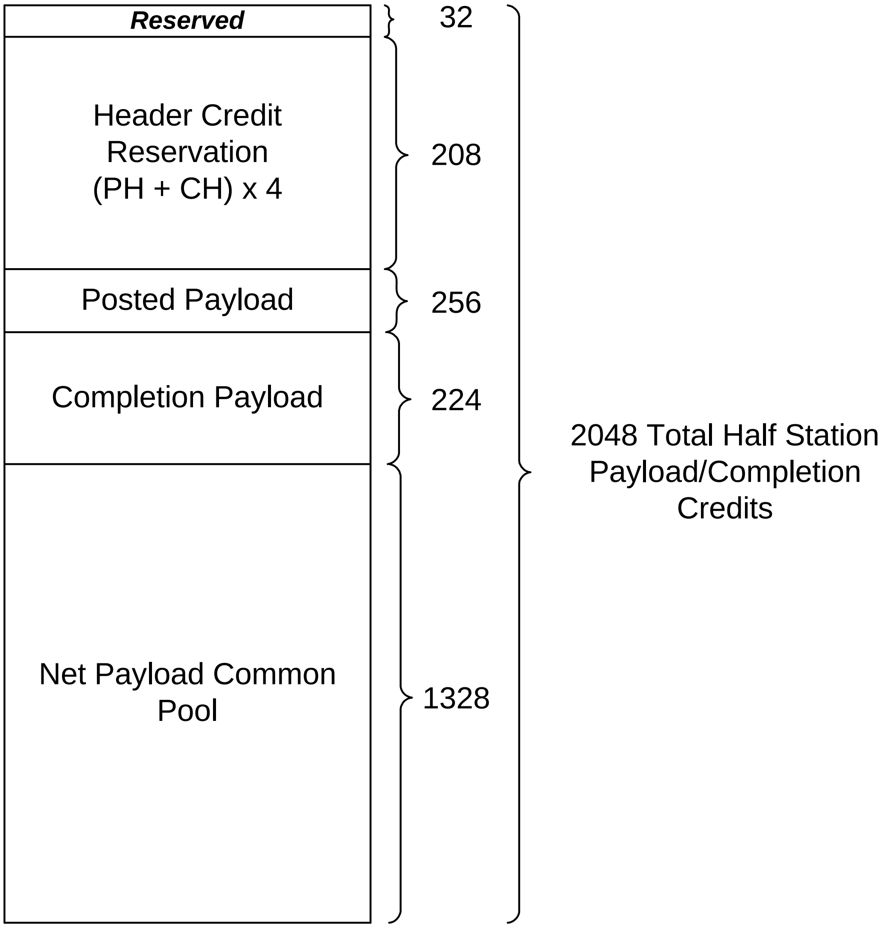


Figure 11: Single X4 Port Half Station Payload/Completion RAM Default Configuration of 2048 Credits


Reserved
32
Header Credit Reservation (PH + CH) x 4
208
Posted Payload
256
Completion Payload
224
Net Payload Common Pool
1328
2048 Total Half Station Payload/Completion Credits

A larger common pool allows the most flexibility for handling any possible instantaneous traffic stream, without back-pressuring ingress traffic. The PEX 8608/8609's initial credit allocation default settings leave sufficient on-chip RAM to accommodate numerous large TLPs in the common pool, after default values for the initial credits are subtracted. Table 13 below summarizes the common pool default allotment for several PEX8608/8609 Half Station Port configurations.

Table 13: PEX 8608/8609 Half Station Port Configuration Common Pool Default Allotments

Half station Port Configuration	Common Pool Header Credits	Common Pool Payload Credits
x1x1x1x1	168	1312
x4	166	1328

Notes:

1. *Actual RAM usage for TLP and Header storage and linking is variable.*
2. The subtraction of 4 credits for each Posted and Completion Header credit advertised, for each Port, is a simplification – the Common Pool credit values can be larger.

Ingress Credit Handler Threshold Registers (Offsets A00h-AECh)

For each Port, there are six sets of **INCH Threshold** registers (Ports 0-1, 4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, offsets A00h-AECh) – VC0 Posted, VC0 Non-Posted, VC0 Completion, VCX Posted, VCX Non-Posted, and VCX Completion. Table 14 lists the lower 16 bits for each register. (To view the complete register set, refer to Ingress Credit Handler Threshold section of the PEX8608/8609 Data Book).

The Non-Posted credits for Payload are cleared to 000h, which equates to infinite credits. Because Non-Posted TLPs only have a 1-DWord Payload, they will never be longer than 5 DWords. Because the Header RAM is 5 DWords wide, only one Non-Posted Header credit is necessary to store a Non-Posted TLP.

Table 14: INCH Threshold Registers (Offsets A00h-AECh); Payload and header Credit Fields

Ports	Register Offset	Type	Payload		Header	
			Bit(s)	Description	Bit(s)	Description
Port 0, VC0	A00h	Posted	8:3	Payload Credit	15:9	Header Credit
	A04h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A08h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 0, VCX	A0Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	A10h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A14h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 1, VC0	A18h	Posted	8:3	Payload Credit	15:9	Header Credit
	A1Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A20h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 1, VCX	A24h	Posted	8:3	Payload Credit	15:9	Header Credit
	A28h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A2Ch	Completion	8:3	Payload Credit	15:9	Header Credit
Port 4, VC0	A60h	Posted	8:3	Payload Credit	15:9	Header Credit
	A64h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A68h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 4, VCX	A6Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	A70h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A74h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 5 VC0	A78h	Posted	8:3	Payload Credit	15:9	Header Credit
	A7Ch	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A80h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 5 VCX	A84h	Posted	8:3	Payload Credit	15:9	Header Credit
	A88h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A8Ch	Completion	8:3	Payload Credit	15:9	Header Credit
Port 6 VC0	A90h	Posted	8:3	Payload Credit	15:9	Header Credit
	A94h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	A98h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 6 VCX	A9Ch	Posted	8:3	Payload Credit	15:9	Header Credit
	AA0h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AA4h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 7 VC0	AA8h	Posted	8:3	Payload Credit	15:9	Header Credit
	AACh	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AB0h	Completion	8:3	Payload Credit	15:9	Header Credit

Ports	Register Offset	Type	Payload		Header	
			Bit(s)	Description	Bit(s)	Description
Port 7 VCX	AB4h	Posted	8:3	Payload Credit	15:9	Header Credit
	AB8h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	ABCh	Completion	8:3	Payload Credit	15:9	Header Credit
Port 8 VC0	AC0h	Posted	8:3	Payload Credit	15:9	Header Credit
	AC4h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AC8h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 8 VCX	ACCh	Posted	8:3	Payload Credit	15:9	Header Credit
	AD0h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AD4h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 9 VC0	AD8h	Posted	8:3	Payload Credit	15:9	Header Credit
	ADCh	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AE0h	Completion	8:3	Payload Credit	15:9	Header Credit
Port 9 VCX	AE4h	Posted	8:3	Payload Credit	15:9	Header Credit
	AE8h	Non-Posted	8:0	000h (infinite)	15:9	Header Credit
	AECh	Completion	8:3	Payload Credit	15:9	Header Credit

Adjusting Initial Credit Values (Ingress Resources)

The default Initial Credit values listed in Tables 11 and 12 can be changed by the user. However, to do so, the values must be changed before the initial credit advertisement by writing the registers through serial EEPROM or the I^{2}C bus. It is also possible to use software to program the Credit registers over the Link; however, if the Link is up, credit cannot be removed, and values can only be increased. The Credit registers are sticky – a Hot Reset preserves any programmed values, and thereby allows any of the available programming methods to program credits at any time, even after the Link is up, if a Hot Reset is issued afterward to rerun the InitFC sequence.

When changing any credit value, follow the rules outlined in this section; otherwise, the credit can be incorrectly issued and data can be lost.

Credit is partitioned/programmed on Half-station basis. All even port numbers belong to Half-station 0 and all odd port numbers belong to Half-station 1. Per the *PCI Express Base r2.0*, the minimum initial credit must be sufficient to meet the credit requirements of the MPS. To meet this requirement with a 2,048-byte MPS, the minimum credit value assigned to a Port, for both Posted and Completion TLPs, must be 128 credits each (one credit represents 16 bytes of storage). Note that the maximum packet size for programmed x1 port widths is fixed at 512 bytes.

Additionally, because each TLP may not optimally fill each location in the internal RAM, the Header credit affects Payload credit used to store the Payloads. Therefore, for every Posted or Completion Header credit reserved, 4 credits from the Payload/Completion RAM must also be held in reserve.

The following abbreviations are used in the rules outlined in this section:

- *PH* is the total Posted Header credits advertised or that can be stored
- *NPH* is the total Non-Posted Header credits advertised or that can be stored
- *CH* is the total Completion Header credits advertised or that can be stored
- *MPS* is the Maximum Payload Size
- *Hmax* is the maximum number of Header credits that can be assigned, per Half-station
- *Pmax* is the maximum number of Payload and Completion credits

The total credit advertised, per Half-station, must follow these rules:

1. Sum of all Header credits $\leq$ Hmax = 244.
   Sum of all Header credits = sum (all Ports PH + NPH + CH).
2. Payload and Completion credit must be sufficient for 1 MPS, for each Port.

   Note: *Non-Posted Payload credit is infinite and Read-Only.*

3. Sum of all Payload and Completion credits assigned, per Half-station,
   is $\leq$ Pmax = 2,016 $\leq$ (Advertised Posted Payload + Advertised Completion Payload
   + (4 x (sum of all Ports PH + CH)))

Programming Example – To satisfy these rules, the following example is presented, using the default values of a Half-station strapped to have two x4 Ports.

1. From Initial Port Credit Allocation Table 11, the sum of all Header credits = 2 x (26 + 26 + 26) = 156, which is ≤ Hmax = 244.
2. For a 2,048-byte MPS, 128 credits must be allocated/advertised for both a Posted and Completion Payload, for each Port. Therefore, for the two Ports in the example, this uses 2 x (256 + 224) credits, or 1,000 Payload/Completion credits.
3. For this case, the Payload/Completion credits used = 1,000 credits +
   4 credits x (2 Ports x (26 + 26)/Port) = 1,416 credits which is ≤ Pmax (2,016).

For this example:

- Common Header Pool = (244 - 156) = 88 Header Credits
- Common Payload/Completion Pool = (2,016 - 1,416) = 600 Payload/Completion Credits

Credit Allocation When Common Pool Is Consumed

A Half-station's common credit pool is consumed by any Port's Ingress queue for that Half-station, on a first-come, first-served basis. If the Half-station's common pool is completely consumed (and therefore each Half-station Port's credit), the PEX 8608/8609 Half-station is in a congested state. In the congested state, the PEX 8608/8609 Half-station is unable to provide additional credit, to any of its Ports, until credit is released. Credit is released only after the Receiver acknowledges the packet with an ACK. In this state, the **INCH Threshold Port x VC0/VCX Posted** register *Congested Port Weight* field (Ports 0-1.4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, [22:20]) provides a method for users to weight each Port's request for more credit, from the Half-station's internal credit allocation logic.

In the congested state, the Half-station's internal credit allocation logic decides which Port will receive the next available credit, by evaluating the following:

- Each Port's *Congested Port Weight* field setting
- Number of Common Pool credits that Port has already consumed
- History of which Ports have recently received credit

By default, the *Congested Port Weight* field for each Port is cleared to 000b. The default value is called an *effective rate setting*. For the default case, if a Half-station is configured into four Ports (x1,x1,x1,x1) each Port receives credit updates based upon on the Port's negotiated width. The x1 Ports would each receive 25% of the credit updates as they become available.

Table 15 defines the *Congested Port Weight* field values. Requests are weighted, based upon the Port's effective Link width, relative to the effective Link widths of the other Half-station's ports. Settings can reduce or increase a Port's effective rate, down to x1 or up to x4, respectively. (*That is*, regardless of the actual Link width, a Link can only be reduced to an x1 effective rate, or increased to an x4 effective rate.) The effective Link Width Request weight is calculated, by multiplying the Port's negotiated Link width (not strapped width) times the *Congested Port Weight* field setting.

The Half station's internal credit allocation logic decides how to allocate the Common Pool credits, as they become available in a congested scenario. Regardless of the Common Pool credit availability, each Port maintains ownership of the ingress credits that were initially allocated to it. In a congested state, those dedicated credits are replenished to their assigned Port, once an ACK is received from the final destination.

The effective rate setting applies to both the Common Header and Common Payload/Completion pools, for the selected Half-station and Port. Although the effective rate setting is in the **INCH Threshold Port x VC0 VCX Posted** register, the value applies to credit updates for all three possible traffic types (Posted, Non-Posted, and Completion).

Table 15: INCH Threshold Port x VC0 VCX Posted Register *Congested Port Weight* Field (Ports 0-1, 4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, [22:20]) Values

***Congested Port Weight* Setting (Bits [22:20])**	**Description**
000b = eff_rate	Request is weighted, based upon the Port's Link width relative to the effective Link widths of the other Stations' Ports.
010b = 4x eff_rate	Increases the weight of a request by 4 (*for example*, an x1 Link Width Request weight increases to an x4 Link Width Request weight).
100b = 0	Port receives no credit out of the common pool, until a decongested state is reached.
110b = eff_rate/4	Decreases the weight of a request by 4 (*for example*, an x4 Link Width Request weight decreases to an x1 Link Width Request weight).

INCH Port Pool Registers (Offsets 940h, 944h, 948h)

The **INCH Port Pool Setting** registers (Ports 0-1, 4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, offsets 940h, 944h, 948h) are registers whose original intent was to provide another level of reservation for Common Pool credits. These registers are essentially redundant to what is accomplished by changing the values of the **INCH Threshold** registers (Ports 0-1, 4-9, and also NT Port Virtual Interface if Port 0 is the NT Port, offsets A00h-A2Ch, A60h-AECh).

Consider the INCH Port Pool register to be *reserved* and only change the credit settings, using the INCH Threshold registers. Do not change the INCH Port Pool register from its default values, unless directed otherwise by PLX Technical Support.

The **INCH Port Pool** register initial values are provided in sets of two for each Port – Payload pool and Header pool. Table 16 lists the bit ranges for each Port.

The initial values of the **INCH Port Pool** registers are all cleared to 0h, which means that by default, there is no additional level of reservation. Additionally, the **INCH Threshold** registers default values evenly allocate all available credit, across all enabled Ports.

Table 16: INCH Port Pool Setting for Ports 0-1, 4-9 (Offsets 940h, 944h, 948h)

Ports	Offset	Payload Pool Bit(s)	Header Pool Bit(s)
0	940h	2:0	6:4
4	944h	2:0	6:4
5	944h	10:8	14:12
6	944h	18:16	22:20
7	944h	26:24	30:28
8	948h	2:0	6:4
9	948h	10:8	14:12

Wait for ACK – Avoiding Congestion

Once a TLP arrives, it remains on the PEX 8608/8609 until it is no longer required. The TLP can quickly egress the PEX 8608/8609. However, until an ACK is received, indicating that the TLP was correctly received, each TLP must remain stored on the PEX 8608/8609 and be ready to be re-sent multiple times. While stored on the PEX 8608/8609, the TLP continues to use Half-station common pool credit resources.

The *PCI Express Base r2.0* recommends sending an ACK within the approximate time it takes to send 1.5 to 3 MPS TLPs. It does not, however, suggest that smaller TLPs obtain faster ACKs. This paper describes the way in which the PEX 8608/8609 sends an ACK. However, the PEX 8608/8609, has no way of knowing its Link partner’s ACK policy.

To minimize the amount of TLPs stored on the PEX 8608/8609 while waiting for an ACK, follow these guidelines:

- Avoid traffic patterns where a great deal of back-to-back TLP bytes travel from a wide Link to a single narrow Link, because the narrow Link can only forward TLPs at a fraction of the ingress rate. For example, if a 4-KB MRd is requested to the Host through a PEX8608/8609 X1 Downstream port, and the PEX8616 Upstream Port is x4, the PEX8608/8609 Upstream Port will eventually transmit 4-KB Read Completion data to the X1 Downstream Requester, four times faster than the Requester can receive the data. If the Requester repeats many of these MRd Requests, large amounts of Read Completion data that require storage on the PEX8608/8609 will quickly accumulate.

- If there are many small TLPs, determine whether the PEX 8608/8609’s ACK response time can be reduced, as per the PCI Express Base r2.0 guidelines.

- Evenly space the TLP pattern, rather than use a burst of many back-to-back TLPs followed by a long stall, to even the distribution and accommodate a fixed ACK Transmission Latency Timer.

Latency

Latency is the length of time it takes to proceed from one event to another. Latency can be measured in several different ways, but perhaps the most common measurement for a PCI Express switch is *Start TLP-to-Start TLP (STP-to-STP) latency*. Figure 12 and Table 17 illustrate an STP-to-STP latency measurement. When the Egress Start TLP symbol is transmitted out of a switch before the Ingress Port End symbol arrives, the transfer is termed *Cut-Thru*. If there is no egress Port queue established, the PEX 8608/8609 always cuts the packet through. The PEX 8608/8609 has the same latency, regardless of whether the traffic is upstream or peer-to-peer.

As expected with the PEX 8608/8609 Cut-Thru architecture, STP-to-STP latency is basically constant for all Payload sizes. A faster ingress link can receive the Header for decode faster, with a slightly lower latency. There will generally be a constant latency for any ingress width to the same egress width or any ingress width to a smaller egress width, operating at the same link speed. This is indicated by the shaded-green entries in Table 17.

For cases in which the egress Port has a higher bandwidth than the ingress Port, then a fraction of the packet must be buffered (to prevent under run in the middle of the TLP) before the TLP can be forwarded to the egress link.

The fractional buffering required is given by the following formula:

F = (E-I) / E

Where

- *F* is the fraction sum
- *E* is egress bandwidth
- *I* is ingress bandwidth

Figure 12: Start TLP-to-Start TLP Latency Measurement

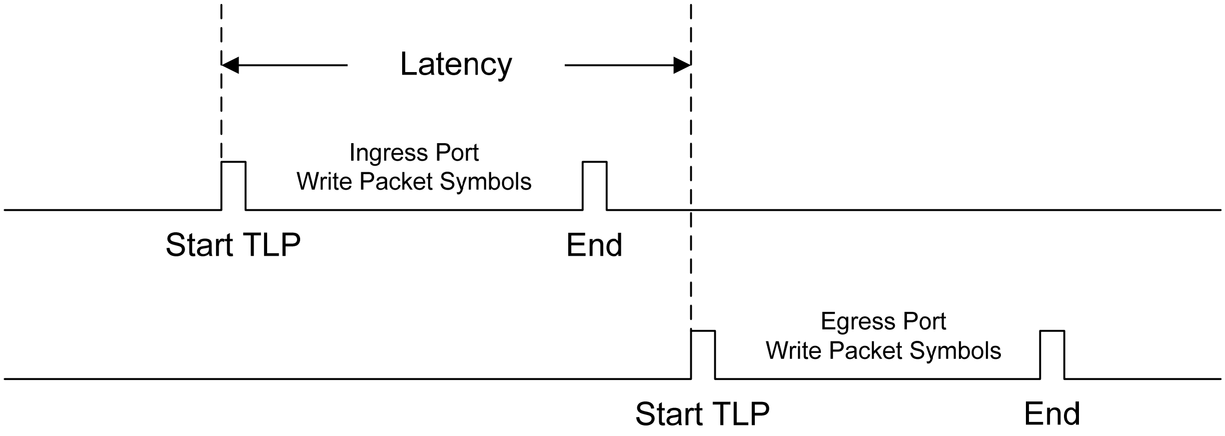


Table 17: Sample STP-to-STP Latency

Latency for TLP with a Data Payload of 4/64/256 Bytes (in ns)			
	From Ingress		
	x4 Gen 2	x2 Gen2	x1 Gen 2
To Egress			
`x4 Gen 2	148/154/138/	164/214/314	184/310/614
x2 Gen 2	150/150/166	174/180/180	200/296/488
x1 Gen 2	180/180/180	188/188/188	180/180/188

Host-Centric Latency

Host-centric traffic flows only to or from the Host. Host-centric latency depends upon the number of active streams. If there is only one active stream, or if the total Host bandwidth is greater than or equal to the sum of all traffic streams, the traffic is well-balanced and the latency measurements provided in Table 17 apply.
If there is more traffic than an upstream Host can sink, congestion occurs when all the TLPs concurrently attempt to use the limited Host bandwidth. The latencies in that case depend upon the level of traffic congestion. In this case, Host bandwidth is at 100%, but the total downstream bandwidth is more than the Host bandwidth, and latencies continue to increase until the congestion eases.

Another case of increased latency is if the Host serially sends large amounts of Read Completion data to one downstream Port and then other downstream Ports. *For example*, if the upstream Port is x4 and the four downstream Ports are all x1, it appears that there should not be a latency build-up, because the bandwidth matches. However, if the Host cannot interleave the destinations, one destination must wait until the Host completes transmitting traffic to the other destinations, before it can receive any Read Completion data. In this case, the round-trip Read latency can significantly increase.

For example, suppose that one downstream Port transmits 16, 4-KB MRd Requests upstream. Those Read Requests represent 64 KB of data. If the upstream Port is x4 and the downstream Port is x1, the Read Completions back up into the PEX 8608/8609, perhaps all the way to the Root Complex. Suppose another downstream Port transmits only one, 1-KB MRd Request upstream, shortly after the 16, 4KB MRds were received by the Root Complex. For many Root Complexes, this one, 1-KB Read Request from the second device must wait for the 16, 4-KB MRd Requests from the first device to complete before being serviced. The PEX 8608/8609 buffer is approximately 10 KB; therefore, the second downstream device must wait for 54 KB (64KB-10KB) of Completion data to transmit across an x1 Link before it begins to receive its Read Completions. On an x1 Link, 54 KB takes about 202 μs, which significantly increases the second device’s latency. The PEX 8608/8609 contains Read Pacing logic that prevents this type of latency increase that occurs when multiple devices concurrently read data from the Root Complex. (Refer to the PEX 8608/8609 Data Book for detailed information on the Read Pacing feature).

Peer-to-Peer Latency

If there is no egress Port queue established (such as multiple streams to the same destination port), peer-to-peer latencies match the best-case values listed in Table 17. The PEX 8608/8609 has the same latency, regardless of whether the traffic is Host-centric or peer-to-peer. Latency is constant in the non-congested case, no matter what Source Port or Destination Port, if the Source Port has the same or greater bandwidth than the Destination Port.

The discussion for Host-centric traffic applies to all Ports for peer-to-peer traffic. It is recommended however, that peer-to-peer application should rely on system-specific methods for balancing traffic flow.

Other Latency Measurements

In addition to STP-to-STP latency, there are other latencies to consider. Table 18 lists various best-case latencies for several Link widths and speeds. Transmitted DLLPs can be required to wait for a TLP. DLLP policies can prevent sending a DLLP for a time period longer than the best case.

Table 18: Miscellaneous Best Case Link Latencies (in ns)

Latency[2]	X4 Gen 2	X2 Gen 2	X1 Gen 2
STP in to UpdateFC SDP out[3]	132	144	180
TLP’s END in to ACK SDP out	72	72	72
UpdateFC SDP in to STP out	116	120	140

[2] *Gen 1 latency values are expected to be the same as, or very close to, Gen 2 latency values.*

[3] *SDP is “Start DLLP” symbol, STP is “Start TLP” symbol, and END is “End of TLP” symbol..*

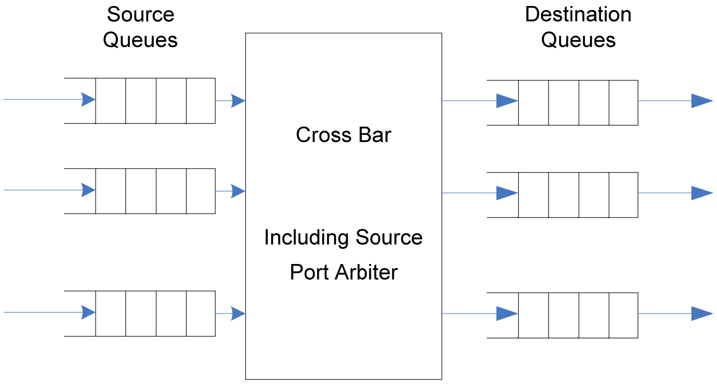
Queuing Options

On-chip queuing does not exist in balanced bandwidth scenarios, where the total ingress bandwidth is less than or equal to the egress bandwidth. In the common case, where the total ingress bandwidth is greater than the egress bandwidth, queues develop on the PEX 8608/8609. The PEX 8608/8609 provides two alternatives to where to locate such queuing (refer to Figure 13):

- **Destination queue** – Associated with a single Destination Port. All the TLPs in a Destination queue will egress out the same Port.
- **Source queue** – Associated with a single ingress Port. All the TLPs in a Source queue come from the same Port.

Each queue is discussed in the sections that follow.

Figure 13: On-Chip Queuing



Destination Queuing

Note: For the queuing examples provided in this section, "Port 1" indicates "first Port," not the Port physically identified as Port 1.

The default behavior is for all queues to develop at the Destination Port. If TLPs are arriving from four sources to a common Destination Port, the TLPs are scheduled according to First-In, First-Out (FIFO). The crossbar can forward a TLP every core clock(4ns), to each Destination queue; therefore, it is unlikely that a Source queue can develop or last very long.

A Destination queue develops whenever the *ingress rate* – the sum of all ingress Ports targeting a Destination Port – exceeds the egress rate. A Destination queue might also develop in a credit-starved situation, where there is no credit available to forward TLPs.

For example, if TLPs arriving from three sources all go to a common Destination Port, the TLPs are scheduled, based upon the order in which they arrive at the Destination queue FIFO[4]. If all three flows are equally active, the TLPs naturally interleave as 1,2,3,1,2,3. If two of the Ports, however, have a head start before the third Port turns on, the output can be 1,2,1,2,1,2,1,2,1,2,3,1,2,3. In this case, all the new Port (3rd Port) TLPs must wait for the earlier 1st, 2nd Port traffic to be transferred before the 3rd Port TLPs can be transferred. Therefore, the latency for 3rd Port traffic to travel through the PEX 8608/8609 can widely vary, based upon the traffic passing through the switch.

[4] *Conventional PCI Strong Ordering rules can override the FIFO. Conventional PCI requires Posted TLPs to be able to pass Non-Posted and Completion TLPs, to avoid deadlock.*

Source Queuing

Note: For the queuing examples provided in this section, "Port 1" indicates "first Port," not the Port physically identified as Port 1.

Source queuing can be enabled for applications that require deterministic bounded latency for a few Ports, while the latency for other Ports is not as important.

Source queuing is enabled by greatly reducing the Destination queue depth. When the Destination queue reaches the newly reduced maximum depth, any subsequent TLPs targeting that Port are not forwarded, but are queued up in a per-Source Port-based queue. The Source Port queue does not forward TLPs until the Destination queue drops to a programmed threshold, upon which TLP forwarding is re-enabled.

Note: A Source Port queue that cannot forward to a Destination queue blocks all subsequent TLPs arriving on that same Source Port, although the target Port is a different destination.

Note: Source Queuing and Read Pacing should not be enabled at the same time. The two features are incompatible and doing so may result in fatal errors.

The **Port Egress TLP Threshold** register (offset F10h) controls the minimum and maximum queue depths. Table 19 summarizes the register bit settings. The Port Lower TLP Count is the number of TLPs to which the Destination queue must reach after becoming saturated, before re-enabling TLP forwarding. The Port Upper TLP Count is the number of TLPs that can be queued in the Destination queue.

In the Destination queue example provided in the previous Destination Queuing section, the early arriving Port 1,2, TLPs stalled Port 3's TLP for an indeterminate length of time. By programming, with source queuing enabled and a destination Port Lower TLP Count Set to 1 and Port Upper TLP Count Set to 2 (TLPs), the worst case is that Port 3 must wait for two TLPs (1,2) before getting its first turn. With these settings, the example TLP output would be 1,2,3,1,2,3,1,2,3. The *turn to be forwarded* refers to a Source Port arbitration wait, described in the next section.

For the PEX 8608/8609, to avoid unnecessary idles on the destination Link, program a Port Lower TLP Count of 1, and a Port Upper TLP Count of 2.

Table 19: Port Egress TLP Threshold Register Port Lower and Upper TLP Counts (Offset F10h)

Bit(s)	Name	Description
10:0	**Port Lower TLP Count**	When Source Scheduling is disabled due to the Port Upper TLP Count (threshold) being exceeded, Source Scheduling is re-enabled when the Port TLP Count goes below the Port Lower TLP Count (this threshold). Because the default setting of this field is 7FFh (2,047), which is greater than the maximum amount of TLPs that can be queued in the PEX 8608/8609, the Source Scheduler is disabled, by default.
26:16	**Port Upper TLP Count**	When the Port TLP Count is greater than or equal to this value, the Source Scheduler disables TLP Scheduling to this egress Port. Because the default setting of this field is 7FFh (2,047), which is greater than the maximum amount of TLPs that can be queued in the PEX 8608/8609, the Source Scheduler is disabled, by default.

Note: Register bits not identified in this table are **reserved**.

Port Arbitration

In the crossbar that connects the Source queues to the Destination queues, there is a Port Arbiter for each Destination Port. The Port Arbiter ensures that each Source Port receives a deterministic bandwidth connecting to a Destination Port. Every Port has two default fixed Round Robin Port Arbiters, one for each Virtual Channel.

In addition to the default fixed Round Robin Port Arbiter, there is one Weighted Round-Robin (WRR) Port arbitration hardware resource that may enabled by system software. The Device-Specific WRR arbitration is also Round-Robin, but with programmable weighting for a particular Port or Ports. For the PEX8608/8609, the Upstream Port number will always be between 0 and 1. Therefore, the Upstream Port will always use the Weighted Round-Robin arbiter resource by default. An NT Port cannot make use of the WRR resource in a PEX8608/8609.

System software discovers the *Port Arbitration Capability* as reflected in the **VC0 Resource Capability** register (Ports 0, 1, offset 158h, bits [1:0]). If the system software wishes to make use of an advertised WRR arbitration with 32 phases capability for the Upstream Port, it programs the *Port Arbitration Select* code to 1h in the **VCO Resource Control** register (Upstream Port 0, 1, offset 15Ch, bits [19:17]).

The WRR Source Port Arbiter has a 32-phase Port Arbitration Table, as outlined in the *PCI Express Base r2.0*, and documented in the **Port Arbitration Table x to x Phase** registers (Upstream Port 0, 1, offsets 1A8h through 1B4h). (Refer to the *PCI Express Base r2.0* for further details.)

Once one or more phase registers are written, the software writes the *Load Port Arbitration Table* bit in the **VCO Resource Control** register (Ports 0, 1, offset 15Ch, [16]). When written, the register values are transferred to the WRR arbitration logic, and take effect immediately.

Port arbitration makes decisions on a per-TLP basis. A Port with more short TLPs will appear to receive less bandwidth, compared to a Port with fewer long TLPs, if both Ports have the same weight and both target a congested Port.

Port Bandwidth Allocation

For applications that need to allocate a fixed bandwidth to each Port, the PEX 8608/8609 can help enforce the relative bandwidth ratio between Ports in a congested scenario.

By combining source queuing, Port arbitration, and initial credit, as well as some knowledge of average Payload size, many combinations of Port bandwidth allocation are possible.