Errata for the PCI Express® Base Specification Revision 2.1

April 17, 2010

REVISION	REVISION HISTORY	DATE
	Initial release.	04/17/2010

PCI-SIG® disclaims all warranties and liability for the use of this document and the information contained herein and assumes no responsibility for any errors that may appear in this document, nor does PCI-SIG make a commitment to update the information contained herein.

Contact the PCI-SIG office to obtain the latest revision of this specification.

Questions regarding the PCI Express Base Specification or membership in PCI-SIG may be forwarded to:

Membership Services
www.pcisig.com
E-mail: administration@pcisig.com
Phone: 503-619-0569
Fax: 503-644-6708

Technical Support
techsupp@pcisig.com

DISCLAIMER

This PCI Express Base Specification Errata document is provided "as is" with no warranties whatsoever, including any warranty of merchantability, noninfringement, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. PCI-SIG disclaims all liability for infringement of proprietary rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

Contents

A1 Rules for Use of Data Poisoning 5
A2 Memory, I/O, and Configuration Request Rules 6
A3 Vendor_Defined Messages 6
A4 Error Message Forwarding and PCI Mapping for Bridge – Rules 7
A5 Configuration.Linkwidth.Start 7
A6 Capability Version 8
A7 Uncorrectable Error Severity Register 8
A8 Rules for Transmitters and Receivers 9
A9 Transaction Descriptor – Transaction ID Field 9
A10 VC Enable 10
A11 Enable Clock Power Management 11
A12 Request Handling Rules 12
A13 Prefetchable BARs 12
B1 Latency Tolerance Reporting(LTR) Mechanism 14
B2 Mapping Between PCI 3.0 and PCI Express Registers 14
B3 IDO Completion Enable 15
B4 Physical Layer Services 15
B5 Detect.Active 15
B6 Terms and Acronyms 16
B7 Address Based Routing Rules and Memory, I/O, and Configuration Request Rules 22
B8 First/Last DW Byte Enables Rules 23
B9 Memory, I/O, and Configuration Request Rules 23
B10 End-End TLP Prefix Processing 23
B11 ST Modes of Operation 23
B12 Latency Tolerance Mechanism 24
B13 ID Based Routing Rules 24
B14 ST Modes 24
B15 Latency Tolerance Mechanism 25
B16 Device Capabilities 2 Register 26
B17 ACS Control Register 26
B18 Latency Tolerance Reporting Capability 27
B19 TPH Requester Capability Structure 27
B20 TPH Requester Capability Register—ST Table Location 27
B21 TPH Requester Capability Register—ST Table Size 28
B22 TPH Requester Control Register—ST Model Select 28
B23 TPH Requester Control Register—TPH Requester Enable 29
B24 TPH ST Table 29
B25 TPH ST Table—ST Upper and ST Lower 29
B26 Common Packet Header Fields 30
B27 Compliance Pattern 30
B28 Modified Compliance Pattern 31

B29 Polling.Compliance 32
B30 OBFF ECN Update [affects PCI Express CEM Specification, Revision 2.0] 33
B31 MC_Base Address Register 33
B32 D3 State 34
B33 Trusted Configuration Space 34
B34 Latency Tolerance Reporting Mechanism 34
B35 Latency Tolerance Reporting Mechanism 35
B36 Additional References to Tables 35
B37 Capitalizing “Reserved” 36
B38 L0 36
B39 The Set of Lanes in Polling.Active 37

A1 Rules for Use of Data Poisoning

In Section 2.7.2.2, page 147, lines 18-38, make the following edits:

- ~~Receipt of a Poisoned TLP is a reported error associated with the Receiving Function (see Section 6.2)~~
- ~~A Poisoned Configuration Write Request must be discarded by the Completer, and a Completion with a Completion Status of UR is returned (see Section 2.2.9).~~
  - ~~A Switch must route a Poisoned Configuration Write Request in the same way it would route a non-Poisoned Request, unless the Request addresses the Configuration Space of the Switch itself, in which case the Switch is the Completer for the Request and must follow the above rule.~~
- ~~A Poisoned I/O or Memory Write Request, or a Message with data (except for vendor-defined Messages), that addresses a control register or control structure in the Completer must be handled as an Unsupported Request (UR) by the Completer (see Section 2.2.9).~~
  - ~~A Switch must route a Poisoned I/O or Memory Write Request or Message with data in the same way it would route the same Request if it were not poisoned, unless the Request addresses a control register or control structure of the Switch itself, in which case the Switch is the Completer for the Request and must follow the above rule.~~
- ~~Unless there's a higher precedence error, a Completer must handle a Poisoned AtomicOp Request as a Poisoned TLP Received error, and must also return a Completion with a Completion Status of Unsupported Request (UR). See Sections 6.2.3.2.3, 6.2.3.2.4, and 6.2.5. The value of the target location must remain unchanged.~~
  - ~~A Switch must route a Poisoned AtomicOp Request the same way it would route the same Request if it were not poisoned, unless the Request targets a Memory Space location in the Switch itself, in which case the Switch is the Completer for the Request and must follow the above rule.~~
- The following Requests with Poisoned data must not modify the value of the target location:
  - Configuration Write Request
  - Any of the following that target a control register or control structure in the Completer: I/O Write Request, Memory Write Request, or non-vendor-defined Message with data
  - AtomicOp Request

  Unless there is a higher precedence error, a Completer must handle these Requests as a Poisoned TLP Received error[footnote] and the Completer must also return a Completion with a Completion Status of Unsupported Request (UR) if the Request is Non-Posted. See Section 6.2.3.2.3, Section 6.2.3.2.4, and Section 6.2.5.

  A Switch must route the Request the same way it would route the same Request if it were not Poisoned, unless the Request targets a location in the Switch itself, in which case the Switch is the Completer for the Request and must follow the above rules.

[footnote] Due to ambiguous language in earlier versions of this specification, a component is permitted to handle this error as an Unsupported Request, but this is strongly discouraged

A2 Memory, I/O, and Configuration Request Rules

In Section 2.2.7, page 76, line 23, add the following text:

- ❑ Requests must not specify an Address/Length combination which causes a Memory Space access to cross a 4-KB boundary.
  - Receivers may optionally check for violations of this rule. If a Receiver implementing this check determines that a TLP violates this rule, the TLP is a Malformed TLP
    - ♦ If checked, this is a reported error associated with the Receiving Port (see Section 6.2)
  - For AtomicOp Requests, the mandatory Completer check for natural alignment of the Address (see above) already guarantees that the access will not cross a 4-KB boundary so a separate 4-KB boundary check is not necessary.
  - Note that if a 4-KB boundary check is performed for AtomicOp CAS Requests, this check must comprehend that the TLP Length value is based on the size of two operands, whereas the access to Memory Space is based on the size of one operand

A3 Vendor_Defined Messages

In Section 2.2.8.6, page 92, lines 10 and 12, make the following edits:

~~Receivers~~ Completers silently discard Vendor_Defined Type 1 Messages which they are not designed to receive – this is not an error condition.

~~Receivers~~ Completers handle the receipt of an unsupported Vendor_Defined Type 0 Message as an Unsupported Request, and the error is reported according to Section 6.2.

In Section 2.9.1, page 150, line 2, make the following edits:

- ❑ The Port must terminate any PME Turn_Off handshake Requests targeting the Port in such a way that the Port is considered to have acknowledged the PME_Turn_Off request (see the Implementation Note in Section 5.3.3.2.1).
- ❑ The Port must handle Vendor Defined Message Requests as described in Section 2.2.8.6 (e.g., silently discard Vendor Defined Type 1 Messages Requests that it is not designed to receive) since the DL_Down prevents the Request from reaching its targeted Function.
- ❑ for all other Posted Requests, discarding the Requests
  - This is a reported error associated with the Function for the (virtual) Bridge associated with the Port (see Section 6.2), and must be reported as an Unsupported Request. For Root Ports, the reporting of this error is optional.
  - For a Posted Request already being processed by the Transaction Layer, the Port is permitted not to report ~~it~~ the error.

A4 Error Message Forwarding and PCI Mapping for Bridge – Rules

In Section 6.2.8.1, page 385, Line 8, make the following edits:

…

- ❑ Poisoned TLPs are forwarded according to the same rules as non-Poisoned TLPs

…

- When forwarding a Poisoned Completion Downstream:
  - ♦ Set the Detected Parity Error bit in the Status register
  - ♦ Set the Master Data Parity Error bit in the ~~Secondary~~ Status register if the Parity Error Response ~~Enable~~ bit in the ~~Bridge Control~~ Command register is set

A5 Configuration.Linkwidth.Start

In Section 4.3.6.3.1.1, page 222, Line 11, make the following edit:

- ❑ In the optional case where a crosslink is supported, the next state is Disable after all Lanes that ~~detected a Receiver during Detect~~ are transmitting TS1 Ordered Sets receive two consecutive TS1 Ordered Sets with the Disable Link bit asserted.

In Section 4.3.6.3.1.1, page 223, Line 1, make the following edit:

- ❑ Next state is Loopback after all Lanes that ~~detected a Receiver during Detect~~are transmitting TS1 Ordered Sets, that are also receiving Ordered Sets, receive the Loopback bit asserted in two consecutive TS1 Ordered Sets.

In Section 4.3.6.3.1.2, page 225, Lines 5, 8, and 10, make the following edits:

- ❑ Next state is Disable after any Lanes that ~~detected a Receiver during Detect~~are transmitting TS1 Ordered Sets, receive two consecutive TS1 Ordered Sets with the Disable Link bit asserted.
  - Note: In the optional case where a crosslink is supported, the next state is Disable only after all Lanes that ~~detected a Receiver during Detect~~are transmitting TS1 Ordered Sets, that are also receiving TS1 Ordered Sets, receive the Disable Link bit asserted in two consecutive TS1 Ordered Sets.
- ❑ Next state is Loopback after all Lanes that ~~detected a Receiver during Detect~~are transmitting TS1 Ordered Sets, that are also receiving TS1 Ordered Sets, receive the Loopback bit asserted in two consecutive TS1 Ordered Sets.

A6 Capability Version

In Section 7.10.1, page 556, Table 7-29, Bits 19:16, make the following edit:

Bit Location	Register Description	Attributes
19:16	**Capability Version** – This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. ~~Must be 2h for this version of the specification.~~For this version of the specification, this field must be 2h if End-End TLP Prefix Supported (see Section 7.8.15) is Set and must be 1h otherwise.	RO

In Section 7.10.12, page 571, line 6 add the following text:

The TLP Prefix Log Registers beyond the number supported by the Function are hardwired to zero. For example, if a Functions, Max End-End TLP Prefixes field contains 10b (indicating 2 DW of buffering) then the third and fourth TLP Prefix Log Registers are hardwired to zero. If the AER Capability Version field is less than 2h, the entire TLP Prefix Log register is not present.

A7 Uncorrectable Error Severity Register

In Section 7.10.4, page 560, line 5, make the following edit:

The Uncorrectable Error Severity register controls whether an individual error is reported as a Nonfatal or Fatal error. An error is reported as fatal when the corresponding error bit in the severity register is Set. If the bit is Clear, the corresponding error is considered non-fatal. Refer to Section 6.2 for further details. Register fields for bits not implemented by the Function are hardwired to ~~the specified default value~~ an implementation specific value.

A8 Rules for Transmitters and Receivers

In Section 4.2.7.1, page 259, line 9, make the following edit:

- Any and all time spent in ~~any lower power Link state (L0s, L1, L2) or in any other~~ state when the Transmitter is electrically idle ~~(such as Detect and Recovery.Speed)~~ does not count in ~~(and may optionally reset)~~ the 1180 to 1538 Symbol Time interval used to schedule the transmission of SKP ordered sets.

- ~~During all lower power Link states any counter(s) or other mechanisms used to schedule SKP Ordered Sets must be reset.~~ It is recommended that any counter(s) or other mechanisms used to schedule SKP Ordered Sets be reset any time when the Transmitter is electrically idle.

In Section 4.2.7.2, page 259, line 21, add the following text:

- Receivers shall be tolerant to receive and process SKP Ordered Sets at an average interval between 1180 to 1538 Symbol Times.

  - Note: Since Transmitters in electrical idle are not required to reset their mechanism for time-based scheduling of SKP Ordered Sets, Receivers shall be tolerant to receive the first time-scheduled SKP Ordered Set following electrical idle in less than the average time interval between SKP Ordered Sets.

A9 Transaction Descriptor – Transaction ID Field

In Section 2.2.6.2, page 71, lines 3 and 10, make the following edits:

- Tag[7:0] is ~~a~~an 8-bit field generated by each Requester, and it must be unique for all outstanding Requests that require a Completion for that Requester

  - Receivers/Completers must handle correctly 8-bit Tag values regardless of the setting of their Extended Tag Field Enable bit. Refer to the *PCI Express to PCI/PCI-X Bridge Specification* for details on the bridge handling of extended tags.

  - If the Extended Tag Field Enable bit (see Section 7.8.4) is clear, the maximum number of outstanding Requests per Function shall be limited to 32, and only the lower 5 bits of the Tag field are used with the remaining upper 3 bits required to be 000b

  - If the Extended Tag Field Enable bit (see Section 7.8.4) is set, the maximum is increased to 256, and the entire Tag field is used

  - The initial value of the Extended Tag Field Enable bit (see Section 7.8.4) is implementation specific

  - Receiver/Completer behavior is undefined if multiple uncompleted Requests are issued non-unique Tag values

A10 VC Enable

In Section 6.18, page 461, lines 11, 21, and 25, make the following edits:

When directed to a non-D0 state by a Write to the PMCSR register, if a device's ~~had~~ most recently ~~previously~~ transmitted LTR message (since the last DL_Down to DL_Up transition) reported one or both latency fields with ~~the~~ any Requirement bit set, then it must send a new LTR Message with both of the Requirement bits clear prior to transitioning to the non-D0 state.

When the LTR Mechanism Enable bit is cleared, if a device's most recently sent LTR Message (since the last DL_DOWN to DL_Up transition) reported latency tolerance values with any Requirement bit set, then it must send a new LTR Message with all the Requirement bits clear.

An LTR Message from a device reflects the tolerable latency from the perspective of the device, for which the platform must consider the service latency itself, plus the delay added by the use of Clock Power Management (CLKREQ#), if applicable. The service latency itself is defined as follows:

- If the device issues a Read Request, latency is measured as delay from transmission of the END symbol in the Request TLP to the receipt of the STP symbol in the first Completion TLP for that Request.
- If the device issues one or more Write Requests such that it cannot issue another Write Request due to Flow Control backpressure, the latency is measured from the transmission of the END symbol of the TLP that exhausts the FC credits to the receipt of the SDP symbol of the DLLP returning more credits for that Request type.

A11 Enable Clock Power Management

In Section 7.8.7, page 521, Table 7-16, Bit 8, make the following edit:

Bit Location	Register Description	Attributes
8	**Enable Clock Power Management** – Applicable only for Upstream Ports and with form factors that support a "Clock Request" (CLKREQ#) mechanism, this bit operates as follows: 0b Clock power management is disabled and device must hold CLKREQ# signal low. 1b When this bit is Set, the device is permitted to use CLKREQ# signal to power manage Link clock according to protocol defined in appropriate form factor specification. For a non-ARI multi-Function device, power-management-configuration software must only Set this bit if all Functions of the multi-Function device indicate a 1b in the Clock Power Management bit of the Link Capabilities register. The component is permitted to use the CLKREQ# signal to power manage Link clock only if this bit is Set for all Functions. For ARI Devices, Clock Power Management is enabled solely by the setting in Function 0. The settings in the other Functions always return whatever value software programmed for each, but otherwise are ignored by the component. Downstream Ports and components that do not support Clock Power Management (as indicated by a 0b value in the Clock Power Management bit of the Link Capabilities register) must hardwire this bit to 0b. Default value of this bit is 0b, unless specified otherwise by the form factor specification.	RW

A12 Request Handling Rules

In Section 2.3.1, page 107, lines 15-19, reformat the text in as shown in red:

- For Configuration Requests only, following reset it is possible for a device to terminate the request but indicate that it is temporarily unable to process the Request, but will be able to process the Request in the future – in this case, the Configuration Request Retry Status (CRS) Completion Status is used (see Section 6.6). Valid reset conditions after which a device is permitted to return CRS are:
  - ♦ Cold, Warm, and Hot Link Resets
  - ♦ FLRs
  - ♦ A reset initiated in response to a $D3_{hot}$ to $D0_{uninitialized}$ device state transition.

  A device Function is explicitly not permitted to return CRS following a software-initiated reset (other than an FLR) of the device, e.g., by the device's software driver writing to a device-specific reset bit. Additionally, a device Function is not permitted to return CRS after having previously returned a Successful Completion without an intervening valid reset (i.e., FLR or Conventional Reset) condition.

A13 Prefetchable BARs

In Section 1.3.2.2, page 41, line 20, make the following edit:

- ❑ A PCI Express Endpoint requesting memory resources through a BAR must set the BAR's Prefetchable bit unless the range contains locations with read side-effects or locations in which the Function does not tolerate write merging. See Section 7.5.2.1 for additional guidance on having the Prefetchable bit Set.

In Section 2.2.5, page 67, line 16, make the following edit:

IMPLEMENTATION NOTE

Read Request with TPH to Non-Prefetchable Space

Memory Read Requests with the TH bit Set and that target Non-Prefetchable Memory Space should only be issued when it can be guaranteed that completion of such reads will not create undesirable side effects. See Section 7.5.2.1 for consideration of certain BARs that may have the Prefetchable bit Set even though they map some locations with read side-effects.

In Section 7.5.2.1, page 488, after line 3, insert the following implementation note:

IMPLEMENTATION NOTE

Additional Guidance on the Prefetchable Bit in Memory Space BARs

PCI Express adapters with Memory Space BARs that request a large amount of non-prefetchable Memory Space (e.g., over 64 MB) may cause shortages of that Space on certain scalable platforms, since many platforms support a total of only 1 GB or less of non-prefetchable Memory Space. This may limit the number of such adapters that can be supported on those platforms. For this reason, it is especially encouraged for BARs requesting large amounts of Memory Space to have their Prefetchable bit Set, since prefetchable Memory Space is more bountiful on most scalable platforms.

While a Memory Space BAR is required have its Prefetchable bit Set if none of the locations within its range have read side effects and all of the locations tolerate write merging, there are system configurations where having the Prefetchable bit Set will still allow correct operation even if those conditions are not met. For those cases, it may make sense for the adapter to have the Prefetchable bit Set in certain candidate BARs so that the system can map those BARs into prefetchable Memory Space in order to avoid non-prefetchable Memory Space shortages.

On PCI Express systems that meet the criteria enumerated below, setting the Prefetchable bit in a candidate BAR will still permit correct operation even if the BAR's range includes some locations that have read side-effects or cannot tolerate write merging. This is primarily due to the fact that PCI Express Memory Reads always contain an explicit length, and PCI Express Switches never prefetch or do byte merging. Generally only 64-bit BARs are good candidates, since only Legacy Endpoints are permitted to set the Prefetchable bit in 32-bit BARs, and most scalable platforms map all 32-bit Memory BARs into non-prefetchable Memory Space regardless of the Prefetchable bit value.

Here are criteria that are sufficient to guarantee correctness for a given candidate BAR:

- The entire path from the host to the adapter is over PCI Express.
- No conventional PCI or PCI-X devices do peer-to-peer reads to the range mapped by the BAR.
- The PCI Express Host Bridge does no byte merging. (This is believed to be true on most platforms.)
- Any locations with read side-effects are never the target of Memory Reads with the TPH bit Set. See Section 2.2.5.

The above criteria are a simplified set that are sufficient to guarantee correctness. Other less restrictive but more complex criteria may also guarantee correctness, but are outside the scope of this specification.

B1 Latency Tolerance Reporting(LTR) Mechanism

In Section 6.18, page 460, line 20, make the following edits:

When enabling the LTR mechanism in a hierarchy, devices closest to the Root Port must be enabled first~~, then moving downwards towards the leaf Endpoints~~.

If an LTR Message is received at a ~~Root~~ Downstream Port that does not support LTR or if LTR is not enabled, the Message must be treated as an Unsupported Request.

B2 Mapping Between PCI 3.0 and PCI Express Registers

In Section 7.5.1.1, page 481, line 2, add the following text:

Table 7-3 establishes the mapping between PCI 3.0 and PCI Express for the PCI 3.0 Configuration Space Command register. For PCI Express to PCI/PCI-X Bridges, refer to the *PCI Express to PCI/PCI-X Bridge Specification* for additional requirements for some bits in this register.

In Section 7.5.1.2, page 483, line 2, add the following text:

Table 7-4 establishes the mapping between PCI 3.0 and PCI Express for PCI 3.0 Configuration Space Status register. For PCI Express to PCI/PCI-X Bridges, refer to the *PCI Express to PCI/PCI-X Bridge Specification* for additional requirements for some bits and fields in this register.

In Section 7.5.3.3, page 489, line 13, add the following text:

This register does not apply to PCI Express. It must be read-only and hardwired to 00h. For PCI Express to PCI/PCI-X Bridges, refer to the *PCI Express to PCI/PCI-X Bridge Specification* for additional requirements for this register.

In Section 7.5.3.4, page 489, line 15, add the following text:

Table 7-5 establishes the mapping between PCI 3.0 and PCI Express for PCI 3.0 Configuration Space Secondary Status register. For PCI Express to PCI/PCI-X Bridges, refer to the *PCI Express to PCI/PCI-X Bridge Specification* for additional requirements for some bits and fields in this register.

In Section 7.5.3.6, page 491, line 2, add the following text:

Table 7-6 establishes the mapping between PCI 3.0 and PCI Express for the PCI 3.0 Configuration Space Bridge Control register. For PCI Express to PCI/PCI-X Bridges, refer to the *PCI Express to PCI/PCI-X Bridge Specification* for additional requirements for some bits in this register.

B3 IDO Completion Enable

In Section 7.8.16, page 546, Table 7-25, Bit 9, edit as follows:

Bit Location	Register Description	Attributes
9	**IDO Completion Enable** – If this bit is Set, the Function is permitted to set the ID-Based Ordering (IDO) bit (Attribute[2]) of Completions it returns (see Section 2.2.6.3 and Section 2.4). Endpoints, including RC Integrated Endpoints, and Root Ports are permitted to implement this capability. A Function is permitted to hardwire this bit to 0b if it never sets the IDO attribute in ~~Requests~~Completions. Default value of this bit is 0b.	RW

B4 Physical Layer Services

In Section 1.5.4.3, page 49, line 11, edit as follows:

Interface initialization, maintenance control, and status tracking:

- Reset/Hot-Plug control/status
- Interconnect power management
- Width and Lane mapping negotiation
- ~~Polarity reversal~~Lane polarity inversion

B5 Detect.Active

In Section 4.2.6.1.2, page 214, line 20, make the following edit:

- The Transmitter performs a Receiver Detection sequence on all unconfigured Lanes that can form one or more Links (see Section ~~4.3.1.8~~4.3.5.7 for more information).

B6 Terms and Acronyms

On page 30 make the following edit:

invariant — A field of a TLP header or TLP Prefix that contains a value that cannot legally be modified as the TLP flows through the PCI Express fabric.

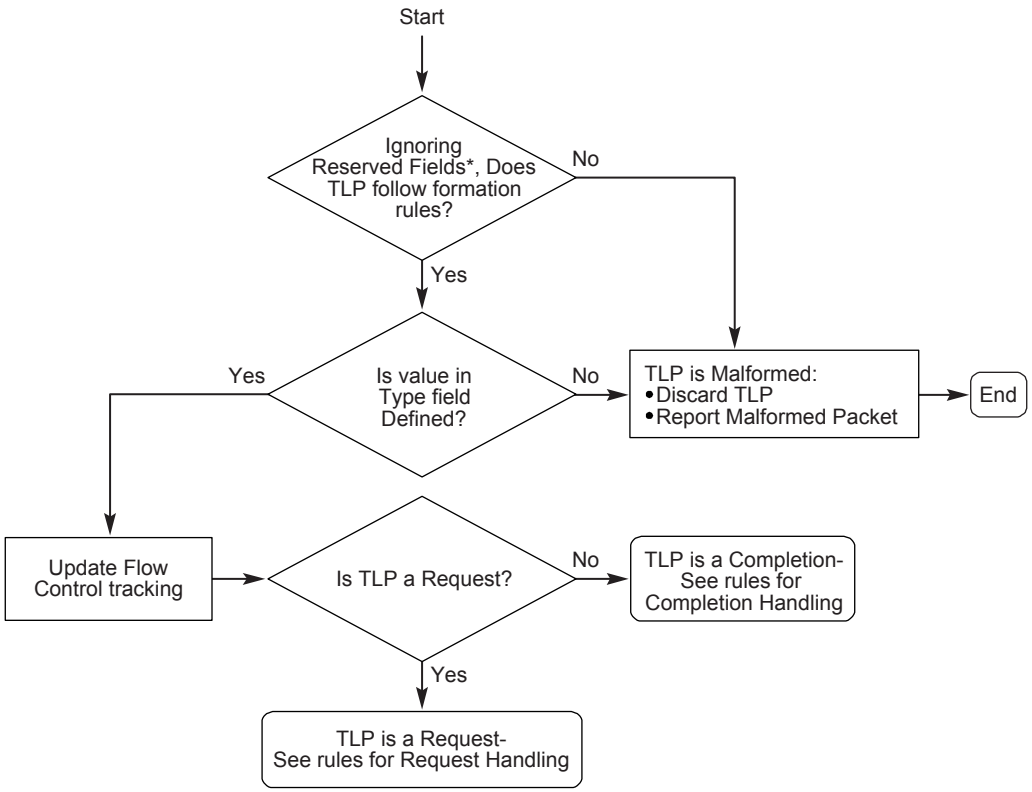
On page 31 make the following edits:

Packet — A fundamental unit of information transfer consisting of an optional TLP Prefix, followed by a header ~~that~~and, in some cases, ~~is~~ followed by a data payload.

On page 33 make the following edit:

variant — A field of a TLP header or TLP Prefix that contains a value that is subject to possible modification according to the rules of this specification as the TLP flows through the PCI Express fabric.

In Section 2.3, page 102, replace Figure 2-30: Flowchart for Handling of Received TLPs with the following:



*TLP fields which are marked Reserved are not checked at the Receiver

OM13771A

In Section 2.6.1.2, page 140, line 21, make the following edit:

TLP Overhead	Represents the additional TLP components which consume Link bandwidth (TLP Prefix, header, LCRC, framing Symbols) and is treated here as a constant value of 28 Symbols

In Section 3.5.2.1, page 172, line 2, make the following edit:

TLP Overhead	represents the additional TLP components which consume Link bandwidth (TLP Prefix, header, LCRC, framing Symbols) and is treated here as a constant value of 28 Symbols

In Section 3.5.3.1, page 182, line 32, make the following edit:

TLP Overhead	represents the additional TLP components which consume Link bandwidth (TLP Prefix, header, LCRC, framing Symbols) and is treated here as a constant value of 28 Symbols.

In Section 2.7.1, page 142, lines 21 and 26, make the following edits:

A 32-bit ECRC is calculated for the ~~entire~~ TLP (End-End TLP Prefixes, header, and data payload) using the following algorithm and appended to the end of the TLP (see Figure 2-3):

- The ECRC value is calculated using the following algorithm (see Figure 2-37).
- The polynomial used has coefficients expressed as 04C1 1DB7h
- The seed value (initial value for ECRC storage registers) is FFFF FFFFh
- All invariant fields of the TLP's End-End TLP Prefixes (if present), header, and the entire data payload (if present) are included in the ECRC calculation, all bits in variant fields must be set to 1b for ECRC calculations.

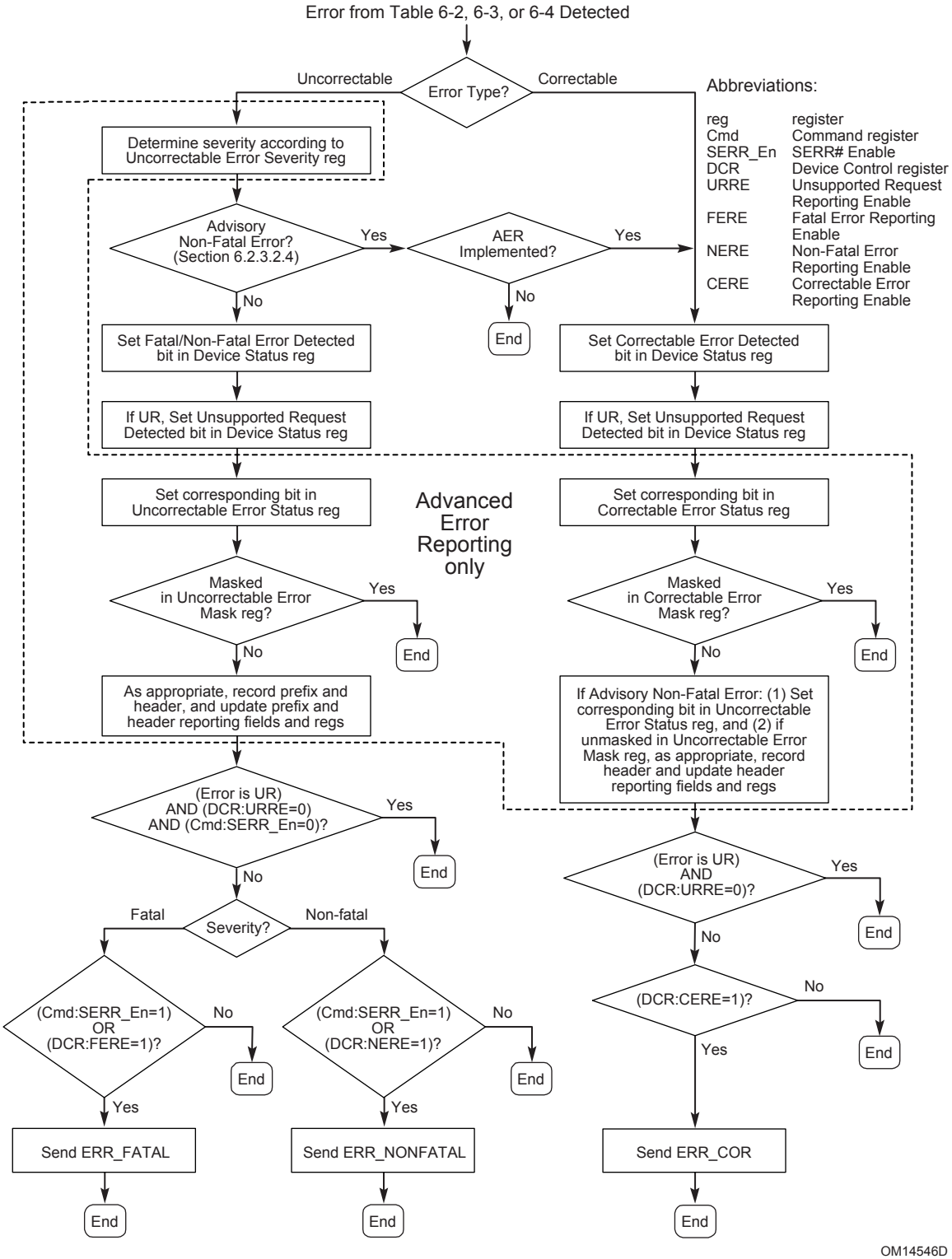
In Section 6.2.4.2, page 374, line 36, make the following edit:

Since an implementation only has the ability to record a finite number of headers, it is important that software services the First Error Pointer ~~and Header Log~~, Header Log, and TLP Prefix Log registers in a timely manner, to limit the risk of missing this information for subsequent errors.

In Section 6.2.4.3, page 375, line 25, make the following edits:

If the "corresponding" uncorrectable error bit in the Uncorrectable Error Mask register is clear and the error is one that requires header logging, then the prefix and header ~~is~~are recorded, subject to the availability of resources.

In Section 6.2.5 , page 377, replace Figure 6-2: Flowchart Showing Sequence of Device Error Signaling and Logging Operations with the following:



In Section 6.2.7, page 379, make the following edits to Table 6-2:

Error Name	Error Type (Default Severity)	Detecting Agent Action	References
Corrected Internal Error	Correctable (masked by default)	*Component:* Send ERR_COR to Root Complex.	Section 6.2.9
Uncorrectable Internal Error	Uncorrectable (Fatal and masked by default)	*Component:* Send ERR_FATAL to Root Complex. Optionally, log the prefix/header of the first TLP associated with the error.	Section 6.2.9
Header Log Overflow	Correctable (masked by default)	*Component:* Send ERR_COR to Root Complex.	Section 6.2.4.2

In Section 6.2.7, page 381, make the following edits to Table 6-5:

Error Name	Error Type (Default Severity)	Detecting Agent Action	References
Poisoned TLP Received	Uncorrectable (Non-Fatal)	*Receiver:* Send ERR_NONFATAL to Root Complex or ERR_COR for the Advisory Non-Fatal Error cases described in Sections 6.2.3.2.4.2 and 6.2.3.2.4.3. Log the prefix/header of the Poisoned TLP.	Section 2.7.2.2
ECRC Check Failed		*Receiver (if ECRC checking is supported):* Send ERR_NONFATAL to Root Complex or ERR_COR for the Advisory Non-Fatal Error case described in Section 6.2.3.2.4.2. Log the prefix/header of the TLP that encountered the ECRC error.	Section 2.7.1

Error Name	Error Type (Default Severity)	Detecting Agent Action	References
Unsupported Request (UR)		*Request Receiver:* Send ERR_NONFATAL to Root Complex or ERR_COR for the Advisory Non-Fatal Error case described in Section 6.2.3.2.4.1. Log the prefix/header of the TLP that caused the error.	Section 2.2.8.6, Section 2.3.1, Section 2.3.2, Section 2.7.2.2., Section 2.9.1, Section 5.3.1, Section 6.2.3.1, Section 6.2.6, Section 6.2.8.1, Section 6.5.7, Section 7.3.1, Section 7.3.3, Section 7.5.1.1, Section 7.5.1.2
Completion Timeout		*Requester:* Send ERR_NONFATAL to Root Complex or ERR_COR for the Advisory Non-Fatal Error case described in Section 6.2.3.2.4.4.	Section 2.8
Completer Abort		*Completer:* Send ERR_NONFATAL to Root Complex or ERR_COR for the Advisory Non-Fatal Error case described in Section 6.2.3.2.4.1. Log the prefix/header of the Request that encountered the error.	Section 2.3.1
Unexpected Completion		*Receiver:* Send ERR_COR to Root Complex. This is an Advisory Non-Fatal Error case described in Section 6.2.3.2.4.5. Log the prefix/header of the Completion that encountered the error.	Section 2.3.2
ACS Violation		*Receiver (if checking):* Send ERR_NONFATAL to Root Complex. Log the prefix/header of the Request TLP that encountered the error.	
MC Blocked TLP		*Receiver (if checking):* Send ERR_NONFATAL to Root Complex. Log the prefix/header of the Request TLP that encountered the error.	Section 6.14.4

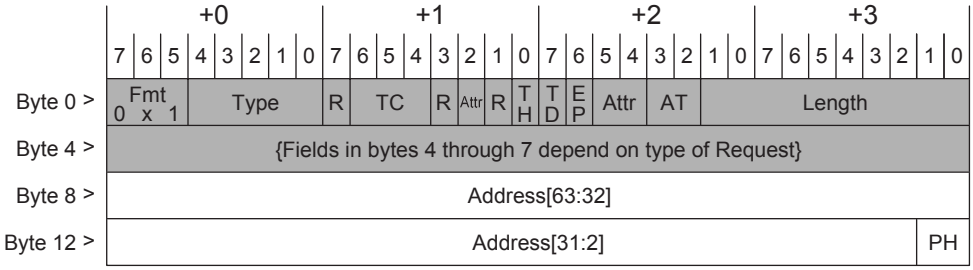
Error Name	Error Type (Default Severity)	Detecting Agent Action	References
AtomicOp Egress Blocked		*Egress Port:* Send ERR_COR to Root Complex. This is an Advisory Non-Fatal Error case described in Section 6.2.3.2.4.1. Log the prefix/header of the AtomicOp Request that encountered the error.	Section.6.15.2
Receiver Overflow	Uncorrectable (Fatal)	*Receiver (if checking):* Send ERR_FATAL to Root Complex.	Section 2.6.1.2
Flow Control Protocol Error		*Receiver (if checking):* Send ERR_FATAL to Root Complex.	Section 2.6.1
Malformed TLP		*Receiver:* Send ERR_FATAL to Root Complex. Log the prefix/header of the TLP that encountered the error.	Section 2.2.2, Section 2.2.3, Section 2.2.5, Section 2.2.7, Section 2.2.8.1, Section 2.2.8.2, Section 2.2.8.3, Section 2.2.8.4, Section 2.2.8.5, Section 2.2.9, Section 2.2.10, Section 2.2.10.1, Section 2.2.10.2, Section 2.3, Section 2.3.1, Section 2.3.1.1, Section 2.3.2, Section 2.5, Section 2.5.3, Section 2.6.1, Section 2.6.1.2, Section 6.2.4.4, Section 6.3.2
TLP Prefix Blocked		*Egress Port:* Send ERR_NONFATAL to Root Complex or ERR_COR for the Advisory Non-Fatal Error case described in Section 6.2.3.2.4.1. Log the prefix/header of the TLP that encountered the error.	Section 2.2.10.2

In Section 6.12.4, page 435, line 17, make the following edits:

~~The~~ AER prefix/header logging and the Prefix Log/Header Log ~~register~~ registers may be used to determine the prefix/header of the offending Request.

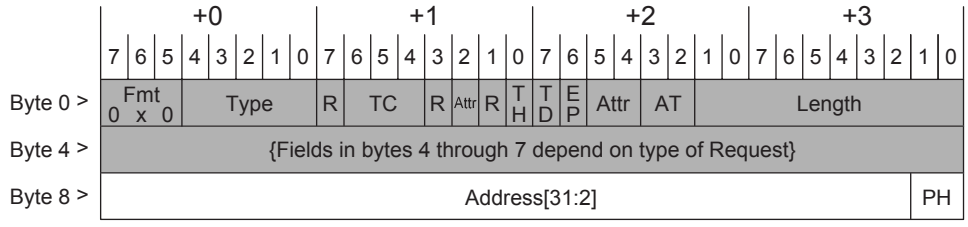
B7 Address Based Routing Rules and Memory, I/O, and Configuration Request Rules

In Section 2.2.4.1, page 64, replace Figure 2-7: 64-bit Address Routing with the following:



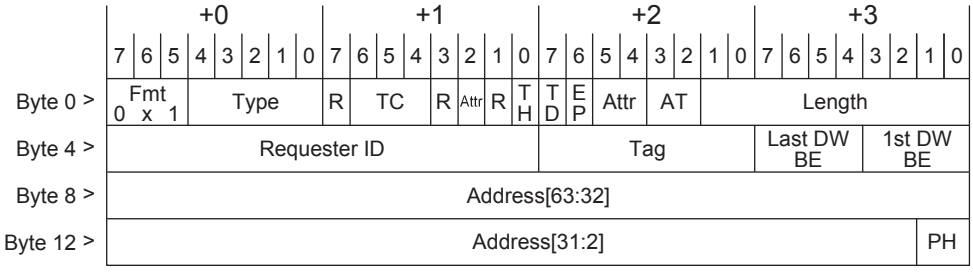
OM14544C

In Section 2.2.4.1, page 64, replace Figure 2-8: 32-bit Address Routing with the following:



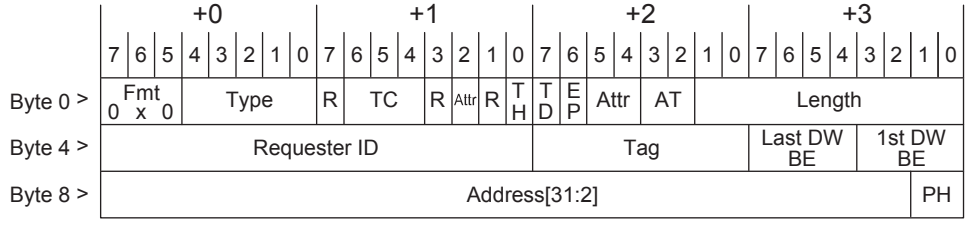
OM14543C

In Section 2.2.7, page 77, replace Figure 2-15: Request Header Format for 64-bit Addressing of Memory with the following:



OM13764C

In Section 2.2.7, page 77, replace Figure 2-16: Request Header Format for 32-bit Addressing of Memory with the following:



OM13763C

B8 First/Last DW Byte Enables Rules

In Section 2.2.5, page 67, line 4, make the following edits:

For Memory Read Requests that have the TH bit Set, the Byte Enable fields are repurposed to carry the ST[7:0] field (Refer to Section 2.2.7.1 for details), and values for the Byte Enables are implied as defined below. ~~Such Requests~~ The TH bit must only be ~~issued~~ set in Memory Read Requests when it is acceptable to complete ~~the~~ those Requests as if all bytes for the requested ~~payload~~ data were enabled.

- ❑ For Memory ~~Reads~~ Read Requests that have the TH bit Set, the following values are implied for the Byte Enables.

B9 Memory, I/O, and Configuration Request Rules

In Section 2.2.7, page 75, line 21, make the following edits:

- Last DW BE[3:0] and 1st DW BE[3:0]. For Memory Read Requests and AtomicOp Requests with the TH bit Set, the byte location for the Last DW BE[3:0] and 1st DW BE [3:0] fields in the header are repurposed to carry ST[7:0] field. ~~and the values for the DW BE fields are implied to be reserved. Otherwise, the DW BE fields are reserved.~~ For Memory Read Requests with the TH bit Clear, see Section 2.2.5 for First/Last DW Byte Enable Rules. For AtomicOp Requests with TH bit Set, the values for the DW BE fields are implied to be reserved. For AtomicOp Requests with TH bit clear, the DW BE fields are reserved.

B10 End-End TLP Prefix Processing

In Section 2.2.10.2, page 100, line 15, make the following edit:

- ❑ For routing elements, the End-End TLP Prefix ~~Egress~~ Blocking bit in each Egress Port determines whether TLPs containing End-End TLP Prefixes can be transmitted via that Egress Port (see Section 7.8.16).

B11 ST Modes of Operation

In Section 6.17.3, page 459, line 3, make the following edits:

A Function that is capable of generating TPH Requests is required to support the No ST Mode of operation~~, and support~~. Support for other ST Modes~~modes~~ of ~~operations~~operation is optional. Only one ST Mode~~mode~~ of operation can be selected at a time by programming ST Mode Select.

B12 Latency Tolerance Mechanism

In Section 6.18, page 464, line 9, make the following edits:

Time D for a Read transaction is the time between the transmission of the END symbol in the Request TLP to the receipt of the STP symbol in the Completion TLP for that Request. Time D for a Write transaction is the time between the transmission of the END symbol of the TLP that exhausts the FC credit to the receipt of the SDP symbol of the DLLP returning more credits for that Request type.

B13 ID Based Routing Rules

In Section 2.2.4.2, page 65, line 13, make the following edit:

This specification defines Vendor_Defined Messages that are ID Routed (Section 2.2.8.6). Other specifications are permitted to define additional ID Routed Messages.

B14 ST Modes

In Section 6.17.3, page 458, make the following edits to Table 6-12:

ST Mode Select [2:0]	ST Mode Name	Description
000	No ST Mode	The Function must use a value of all zeroes for all Steering Tags.
001	Interrupt Vector Mode	Each Steering Tag is selected by an MSI/MSI-X interrupt vector number. The Function is required to use the Steering Tag value from an ST Table entry that can be indexed by a valid MSI/MSI-X interrupt vector number.
010	Device Specific Mode	It is recommended for the Function to use a Steering Tag value from an ST Table entry, but it is not required.
All other encodings	~~All other encodings~~Reserved	Reserved for future use.

B15 Latency Tolerance Mechanism

In Section 6.18, page 462, lines 13-39, make the following edits:

Switches must collect the Messages from Downstream Ports that have the LTR mechanism enabled and transmit a "conglomerated" Message Upstream according to the following rules:

- If a Switch supports the LTR feature, it must support the feature on its Upstream Port and all Downstream Ports.
- A Switch Upstream Port is permitted to transmit LTR Messages only when its LTR Mechanism Enable bit is Set or shortly after software clears its LTR Mechanism Enable bit as described earlier in this section.
- ~~The acceptable latency values for the Message sent Upstream by the Switch must reflect the lowest values received from any Downstream Port.~~
  - ~~When any Downstream Port reports a field with the Requirement bit set to 1 and a LatencyValue of all 0's (regardless of the LatencyScale value), the Message sent Upstream must report a LatencyScale of 000b and a LatencyValue of all 0's.~~
- The acceptable latency values for the Message sent Upstream by the Switch must be calculated as follows:
  - If none of the Downstream Ports receive an LTR Message containing a requirement for a certain type of traffic (snoop/no-snoop), ~~the~~ then any LTR Message sent by the switch must not set the Requirement bit corresponding to that type of traffic.
  - Define LTRdnport[N] as the value reported in the LTR Message received at Downstream Port N, with these adjustments if applicable:
    - LTRdnport[N] is effectively infinite if the Requirement bit is clear or if a Not Permitted LatencyScale value is used
    - LTRdnport[N] must be 0 if the Requirement bit is 1 and the LatencyValue field is all 0's regardless of the LatencyScale value
  - Define LTRdnportMin as the minimum value of LTRdnport[N] across all Downstream Ports
  - Define Lswitch as all latency induced by the Switch
    - If Lswitch dynamically changes based on the Switch's operational mode, the Switch must not allow Lswitch to exceed 20% of LTRdnportMin, unless Lswitch for the Switch's lowest latency mode is greater, in which case the lowest latency state must be used
  - Calculate the value to transmit upstream, LTRconglomerated, as LTRdnportMin – Lswitch, unless this value is less than 0 in which case LTRconglomerated is 0
  - If LTRconglomerated is 0, both the LatencyValue and LatencyScale fields must be all 0's in the conglomerated LTR message
  - ~~Any additional latency induced by the Switch must be accounted for in the conglomerated Message. A Switch must ensure that its Link and internal power management and other internal operation shall not cause its conglomerated latency to be reduced by more than 20% of the lowest received latency.~~

- ~~If any Downstream Port reports a field(s) for which the Requirement bit is clear, or uses a Not Permitted LatencyScale value, that Port must not be considered when determining the corresponding field(s) reported in the Message sent Upstream.~~
  - ~~Valid, permitted values for other fields must still be considered.~~
- When a Switch Downstream Port goes to DL_Down status, the previous latencies recorded for that Port must be treated as invalid~~.~~. A new LTR message must be ~~and the latencies to be transmitted Upstream updated and a new conglomerated Message~~ transmitted Upstream if the conglomerated latencies are changed as a result.

B16 Device Capabilities 2 Register

In Section 7.8.15, page 542, make the following edits to Table 7-24:

Bit Location	Register Description	Attributes
13:12	**TPH Completer Supported** – Value indicates Completer support for TPH or Extended TPH. Applicable only to Root Ports and Endpoints.~~; must be 00b for other Function types~~. For all other Functions, this field is Reserved. Defined Encodings are: 00b TPH and Extended TPH Completer not supported. 01b TPH Completer supported; Extended TPH Completer not supported. 10b Reserved. 11b Both TPH and Extended TPH Completer supported. See Section 6.17 for details.	RO

B17 ACS Control Register

In Section 7.16.3, page 610, make the following edits to Table 7-67:

Bit Location	Register Description	Attributes
6	**ACS Direct Translated P2P Enable (T)** – When Set, overrides the ACS P2P Request Redirect and ACS P2P Egress Control mechanisms with peer-to-peer Memory Requests whose Address Translation (AT) field indicates a Translated address (see Section 6.12.3). This bit is ignored if ACS Translation Blocking Enable (B) is 1b. Default value of this bit is 0b. Must be hardwired to 0b if the ACS Direct Translated P2P functionality is not implemented.	RW

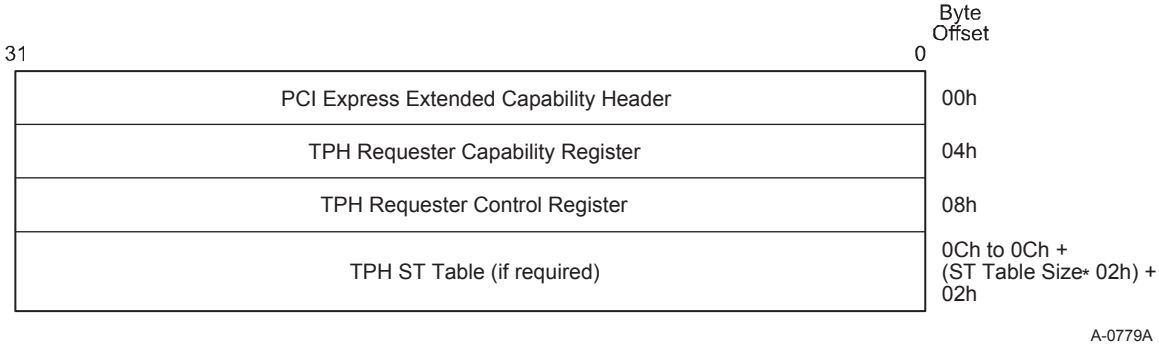
B18 Latency Tolerance Reporting Capability

In Section 7.25, page 651, line 3, make the following edits:

The PCI Express Latency Tolerance Reporting (LTR) Capability is an optional Extended Capability that allows software to provide platform latency information to components with Upstream Ports (Endpoints and Switches), and is required for Switch Upstream Ports and Endpoints if the ~~component~~ Function supports the LTR mechanism. It is not applicable to Root Ports, Bridges, or ~~Downstream Ports in a~~ Switch Downstream Ports.

B19 TPH Requester Capability Structure

In Section 7.26, page 653, replace Figure 7-126: TPH Extended Capability Structure with the following:



B20 TPH Requester Capability Register—ST Table Location

In Section 7.26.2, page 655, make the following edits to Table 7-109:

Bit Location	Register Description	Attributes
10:9	**ST Table Location** – Value indicates if and where the ST Table is located. Defined Encodings are: 00 – ST Table is not present. 01 – ST Table is located in the TPH Requester Capability structure. 10 – ST Table is located in the MSI-X Table ~~structure~~. 11 – Reserved A Function that only supports the No ST Mode of operation must have a value of 00b in this field. A Function may report a value of 10b only if it implements an MSI-X Capability structure.	RO

B21 TPH Requester Capability Register—ST Table Size

In Section 7.26.2, page 655, make the following edits to Table 7-109:

Bit Location	Register Description	Attributes
26:16	**ST Table Size** – Value indicates the maximum number of ST Table entries the Function may use. Software reads this field to determine the ST Table Size N, which is encoded as N-1. For example, a returned value of ~~"00000000011"~~00000000011b indicates a table size of ~~4~~ four entries. There is an upper limit of 64 entries when the ST Table is located in the TPH Requester Capability structure. There is an upper limit of ~~2 K~~2048 entries when the ST Table is located in the MSI-X Table. This field is only applicable for Functions that implement an ST Table as indicated by the ST Table Location field. Otherwise, the value in this field is undefined.	RO

B22 TPH Requester Control Register—ST Model Select

In Section 7.26.3, page 656, make the following edits to Table 7-110:

Bit Location	Register Description	Attributes
2:0	**ST Mode Select** – selects the ST ~~mode~~Mode of operation. Defined encodings are: 000b – No ST Mode 001b – Interrupt Vector Mode 010b – Device Specific Mode 0thers – reserved for future use Functions that support only the No ST Mode of operation must hardwire this field to 000b. Function operation is undefined if software enables a mode of operation that does not correspond to a mode supported by the Function. The default value of this field is 000b. See Section ~~6.17.2~~6.17.3 for details on ST modes of operation.	RW

B23 TPH Requester Control Register—TPH Requester Enable

In Section 7.26.3, page 656, make the following edits to Table 7-110:

Bit Location	Register Description	Attributes
9:8	**TPH Requester Enable** – Controls the ability to issue Request TLPs using either TPH or Extended TPH. Defineddefined encodings are: 00b – Function operating as a Requester is not permitted to issue Requests with TPH or Extended TPH. 01b – Function operating as a Requester is permitted to issue Requests with TPH and is not permitted to issue Requests with Extended TPH. 10b – Reserved. 11b – Function operating as a Requester is permitted to issue Requests with TPH and Extended TPH. The default value of this field is 00b.	RW

B24 TPH ST Table

In Section 7.26.4, page 657, lines 1-5, make the following edits:

The ST Table must be implemented in the TPH Requester Capability Structurestructure if the value of the ST Table Location field is 01b. For all other values, the ST Entry Registersregisters must not be implemented. Each implemented ST Entry is 16 bits. The number of ST Entry Registersregisters implemented must be equal to the number of ST Table entries supported by the Function, which is the value of the ST Table Size field plus one.

B25 TPH ST Table—ST Upper and ST Lower

In Section 7.26.4, page 657, make the following edits to Table 7-111:

Bit Location	Register Description	Attributes
7:0	**ST Lower** – If the Function implements a TPH Requester Capability structure, and the ST Table Location indicates a value of 01b, then this This field contains the lower 8 bits of a Steering Tag. Default value of this field is 0h.	RW
15:8	**ST Upper** – If the Function implements a TPH Requester Capability structure, and the ST Table Location indicates a value of 01b, and the Function's Extended TPH Requester Supported bit is Set, then this field contains the upper 8 bits of a Steering Tag. Otherwise, this field is RsvdP. Default value of this field is 00h.	RW

B26 Common Packet Header Fields

In Section 2.2.1, page 57, make the following edits to Table 2.2:

Table 2-2: Fmt[21:0] Field Values

Fmt[21:0]	Corresponding TLP Format
000b	3 DW header, no data
001b	4 DW header, no data
010b	3 DW header, with data
011b	4 DW header, with data
100b	TLP Prefix
	All encodings not shown above are reserved (see Section 2.3).

In Section 2.2.1, page 58, make the following edit to the title of Table 2.3:

Table 2-3: Fmt[21:0] and Type[4:0] Field Encodings

B27 Compliance Pattern

In Section 4.2.8, page 260, lines 3-9, make the following edits:

During Polling, the Polling.Compliance substate must be entered from Polling.Active based on the conditions described in Section 4.2.6.2.1. The compliance pattern consists of the sequence of 8b/10b Symbols K28.5, D21.5, K28.5, and D10.2 repeating. The ~~compliance~~Compliance sequence is as follows:

Symbol	K28.5	D21.5	K28.5	D10.2
Current Disparity	Negative	Positive	Positive	Negative
Pattern	0011111010	1010101010	1100000101	0101010101

The first Compliance sequence Symbol must have negative disparity. It is permitted to create a disparity error to align the running disparity to the negative disparity of the first Compliance sequence Symbol.

For any given device that has multiple Lanes, every eighth Lane is delayed by a total of four Symbols. A two Symbol delay occurs at both the beginning and end of the four Symbol Compliance Pattern sequence. Note: A x1 device, or a xN device operating a Link in x1 mode, is permitted to include the Delay ~~s~~Symbols with the Compliance Pattern.

This delay sequence on every eighth Lane is then:

Symbol:	D	D	K28.5	D21.5	K28.5	D10.2	D	D

Where D is a K28.5 Symbol. The first D Symbol has negative disparity to align the delay disparity with the disparity of the Compliance sequence.

In Section 4.2.1.3, page 190, lines 3-10, make the following edits:

A Transmitter is permitted to pick any disparity, unless otherwise required, when first transmitting differential data after being in an Electrical Idle state. The Transmitter must then follow proper 8b/10b encoding rules until the next Electrical Idle state is entered.

The initial disparity for a Receiver that detects an exit from Electrical Idle is set to the disparity of the first Symbol used to obtain Symbol lock. Disparity may also be-reinitialized if Symbol lock is lost and regained during the transmission of differential information due to an implementation specific number of errors. All following received Symbols after the initial disparity is set must be ~~in the~~ found in the proper column corresponding to the current running disparity.

B28 Modified Compliance Pattern

In Section 4.2.9, page 261, line 5, make the following edits:

The Modified Compliance Pattern consists of the same basic Compliance Pattern sequence (see Section 4.2.8) with one change. Two identical error status Symbols followed by two K28.5 are appended to the basic Compliance sequence of 8b/10b Symbols (K28.5, D21.5, K28.5, and D10.2) to form the Modified Compliance Sequence of (K28.5, D21.5, K28.5, D10.2, error status Symbol, error status Symbol, K28.5, K28.5). The first Modified Compliance Sequence Symbol must have negative disparity. It is permitted to create a disparity error to align the running disparity to the negative disparity of the first Modified Compliance Sequence Symbol. For any given device that has multiple Lanes, every eighth Lane is moved by a total of eight Symbols. Four Symbols of K28.5 occurs at the beginning and another four Symbols of K28.7 occurs at the end of the eight Symbol Modified Compliance Pattern sequence. The first D Symbol has negative disparity to align the delay disparity with the disparity of the Modified Compliance Sequence. After the 16 Symbols are sent, the delay Symbols are advanced to the next Lane, until the delay Symbols have been sent on all eight lanes. Then the delay Symbols cycle back to Lane 0, and the process is repeated. It is permitted to advance the delay sequence across all eight lanes, regardless of the number of lanes detected or supported. Note: A x1 device, or a xN device operating a Link in x1 mode, in permitted to include the Delay symbols with the Modified Compliance Pattern.

B29 Polling.Compliance

In Section 4.2.6.2.2, page 220, lines 30-38, reformat the text as shown in red:

- Else (i.e., entry to Polling.Compliance is not due to the Compliance Receive bit assertion with Loopback bit deassertion in the received consecutive TS Ordered Sets and not due to the Enter Compliance bit and the Enter Modified Compliance bit in the Link Control 2 register set to 1b)

  (a) Transmitter sends out the compliance pattern on all Lanes that detected a Receiver during Detect at the data rate and de-emphasis level determined above.

  (b) Next state is Polling.Active if any of the following three conditions are true:

  1. Electrical Idle exit has been detected at the Receiver of any Lane that detected a Receiver during Detect, and the Enter Compliance bit in the Link Control 2 register is 0b.
  2. The Enter Compliance bit in the Link Control 2 register has changed to 0b (from 1b) since entering Polling.Compliance.
  3. The component is a Downstream component, the Enter Compliance bit in the Link Control 2 register is set to 1b and an EIOS has been detected on any Lane. The Enter Compliance bit is reset to 0b when this condition is true.

  If the Transmitter is transmitting at a data rate other than 2.5 GT/s, or the Enter Compliance bit in the Link Control 2 register was set to 1b during entry to Polling.Compliance, the Transmitter sends eight consecutive EIOSs and enters Electrical Idle prior to transitioning to Polling.Active. During the period of Electrical Idle, the data rate is changed to 2.5 GT/s and stabilized. The period of Electrical Idle is greater than 1 ms but must not exceed 2 ms.

  Note: Sending multiple EIOSs provides enough robustness such that the other component detects at least one EIOS and exits Polling.Compliance substate when the configuration register mechanism was used for entry.

B30 OBFF ECN Update [affects PCI Express CEM Specification, Revision 2.0]

On page 13 of the ECN, make the following edits (in red) to the Power Sequencing and Reset Signal Timings table:

Table 2-4: Power Sequencing and Reset Signal Timings

Symbol	Parameter	Min	Max	Units	Notes	Figure
T_{PVPERL}	Power stable to PERST# inactive	100		ms	1	Figure 2-10
$T_{PERST-CLK}$	REFCLK stable before PERST# inactive	100		µs	2	Figure 2-10
T_{PERST}	PERST# active time	100		µs		Figure 2-11
T_{FAIL}	Power level invalid to PERST# active		500	ns	3	Figure 2-13
T_{WKRF}	WAKE# rise – fall time		100	ns	4	Figure 2-14
$T_{WAKE-TX-MINMAXPULSE}$	Minimum/Maximum WAKE# pulse width; applies to both active-inactive-active and inactive-active-inactive cases	300	500	ns	5	
$T_{WAKE-FALL-FALLCPU-ACTIVE}$	Time between two falling WAKE# edges when signaling CPU Active	700	1000	ns	5	

Notes:

1. Any supplied power is stable when it meets the requirements specified for that power supply.
2. A supplied reference clock is stable when it meets the requirements specified for the reference clock. The PERST# signal is asserted and de-asserted asynchronously with respect to the supplied reference clock.
3. The PERST# signal must be asserted within T_{FAIL} of any supplied power going out of specification.
4. Measured from WAKE# assertion/de-assertion to valid input level at the system PM controller. Since WAKE# is an open-drain signal, the rise time is dependent on the total capacitance on the platform and the system board pull-up resistor. It is the responsibility of the system designer to meet the rise time specification.
5. Refers to timing requirement for indicating an active window.

B31 MC_Base Address Register

In Section 7.21.4, page 636, make the following edits:

7.21.4 ~~Multicast Base~~MC_Base Address Register (Offset 08h)

Table 7-88: ~~Multicast Base~~MC_Base Address Register

B32 D3 State

In Section 5.3.1.4, page 321, line 5, make the following edit:

D3 support is required, (both the $D3_{cold}$ and the $D3_{hot}$ states). Functions supporting PME generation from D3 must support it for both $D3_{cold}$ and the $D3_{hot}$ states.

Functional context is required to be maintained by Functions in the $D3_{hot}$ state if the No_Soft_Reset field in the PMCSR is Set. In this case, software is not required to re-initialize the Function after a transition from $D3_{hot}$ to D0 (the Function will be in the $D0_{~~initialized~~active}$ state). If the No_Soft_Reset bit is Clear, functional context is not required to be maintained by the Function in the $D3_{hot}$ state. As a result, in this case software is required to fully re-initialize the Function after a transition to D0 as the Function will be in the $D0_{uninitialized}$ state.

The Function will be reset if the Link state has transitioned to the L2/L3 Ready state regardless of the value of the No_Soft_Reset bit.

B33 Trusted Configuration Space

In Section 2.2.1, page 58, footnote 3, make the following edit:

[3] Deprecated TLP Types: previously used for Trusted Configuration Space (TCS), which is no longer supported by this specification. If a Receiver does not implement TCS, the Receiver must treat such Requests as Malformed Packets

B34 Latency Tolerance Reporting Mechanism

In Section 6.18, page 463, line 12, make the following edits:

When the latency requirement is updated during a series of Requests, it is required that the updated latency figure be comprehended by the RC prior to servicing ~~no later than the larger of either (a) waiting as long as the previously indicated latency or (b) following the servicing of~~ a subsequent Request. In all cases the updated latency value must take effect within a time period equal to or less than the previously reported latency requirement. It is permitted for the RC to comprehend the updated latency figure earlier than this limit.

B35 Latency Tolerance Reporting Mechanism

In Section 6.18, page 464, lines 6-24, make the following edits:

Time C is the period during which the transition from L1 ASPM to L0 takes place. ~~This value will not exceed the maximum L1 exit latency reported by the Endpoint.~~

Time D for a Read transaction is the time between the transmission of the END symbol in the Request TLP to the receipt of the STP symbol in the Completion TLP. Time D for a Write transaction is the time between the transmission of the END symbol of the TLP that exhausts the FC credits to the receipt of the SDP symbol in the DLLP returning more credits. This value will not exceed the latency in effect.

Time E is the period where the data path from the Endpoint to system memory is open, and data transactions are not subject to the leadoff latency.

The LTR latency semantic reflects the tolerable latency seen by the device as measured by one or both of the following:

> Case 1: the device may or may not support Clock PM, but has not deasserted its CLKREQ# signal – The latency observed by the device is represented in Figure 6-16 as the sum of times C and D.
>
> Case 2: the device supports Clock PM and has deasserted CLKREQ#- The latency observed by the device is represented as the sum of times A, C, and D.

To effectively use the LTR mechanism in conjunction with Clock PM, the device will know or be able to measure times B and C, so that it knows when to assert CLKREQ#. The actual values of Time A, Time C, and Time D may vary dynamically, and it is the responsibility of the platform to ensure the sum will not exceed the latency.

B36 Additional References to Tables

In Section 2.6.1.2, page 140, line 27, make the following edit:

UpdateFactor	Represents the number of maximum size TLPs sent during the time between UpdateFC receptions, and is used to balance Link bandwidth efficiency and receive buffer sizes – the value varies according to Max_Payload_Size and Link width, and is included in Table 2-38 and Table 2-39.

In Section 3.5.2.1, page 171, line 20, make the following edit:

- If the Transmit Retry Buffer contains TLPs for which no Ack or Nak DLLP has been received, and (as indicated by REPLAY_TIMER) no Ack or Nak DLLP has been received for a period exceeding the time indicated in Table 3-4 for 2.5 GT/s mode operation and Table 3-5 for 5.0 GT/s mode operation, the Transmitter initiates a replay.

In Section 3.5.2.1, page 172, line 8, make the following edit:

AckFactor	represents the number of maximum size TLPs which can be received before an Ack is sent, and is used to balance Link bandwidth efficiency and retry buffer size – the value varies according to Max_Payload_Size and Link width, and is ~~included~~defined in Table 3-6 and Table 3-7.

In Section 3.5.3.1, page 183, line 5, make the following edit:

AckFactor	represents the number of maximum size TLPs which can be received before an Ack is sent, and is used to balance Link bandwidth efficiency and retry buffer size – the value varies according to Max_Payload_Size and Link width, and is defined in Table 3-6 and Table 3-7.

B37 Capitalizing "Reserved"

Excluding page 101 footnote 19, page 118 line 15, page 415 line 28, and page 665 line 24:

Change all occurrences of "reserved" to "Reserved".

B38 L0

In Section 4.2.6.5, page 246, lines 10 and 26, make the following edits:

- Note: "if directed" is defined as both ends of the Link having agreed to enter L1 immediately after the condition of both the receipt and transmission of the EIOS(s) is met. A transition to L1 can be initiated by PCI-PM (see Section 5~~4~~.3.2.1), or by ASPM (see Section 5.4.1.2.1).
- Note: "if directed" is defined as both ends of the Link having agreed to enter L2 immediately after the condition of both the receipt and transmission of the EIOS(s) is met (see Section 5~~4~~.3.2.3 for more details).

B39 The Set of Lanes in Polling.Active

In Section 4.2.6.2.1, page 217, line 1, make the following edits:

…

ii At least a predetermined ~~number~~set of Lanes that detected a Receiver during Detect have detected an exit from Electrical Idle at least once since entering Polling.Active.

- Note: This may prevent~~s~~ one or more bad Receivers or Transmitters from holding up a valid Link from being configured, and allow~~s~~ for additional training in Polling.Configuration. The exact ~~number~~set of predetermined Lanes is implementation specific. Note that up to the 1.1 specification this predetermined ~~number~~set was ~~equal to~~ the total ~~number~~set of Lanes that detected a Receiver.
- Note: Any Lane that receives eight consecutive TS1 or TS2 Ordered Sets should have detected an exit from Electrical Idle at least once since entering Polling.Active.

- Else Polling.Compliance if either (a) or (b) is true:

  (a) ~~less than~~not all Lanes from the predetermined ~~number~~set of Lanes from (ii) above have detected an exit from Electrical Idle since entering Polling.Active.

  (b) any Lane that detected a Receiver during Detect received eight consecutive TS1 Ordered Sets (or their complement) with the Lane and Link numbers set to PAD (K23.7), the Compliance Receive bit (bit 4 of Symbol 5) is 1b, and the Loopback bit (bit 2 of Symbol 5) is 0b.

…

In Section 4.2.6.2.2, page 220, line 21, make the following edits:

…

(b) Next state is Polling.Active if any of the following three conditions are true:

1. Electrical Idle exit has been detected at the Receiver of any Lane that detected a Receiver during Detect, and the Enter Compliance bit in the Link Control 2 register is 0b.

   Note: the "any Lane" condition supports the usage model described in the "Compliance Load Board Usage to Generate Compliance Patterns" Implementation Note earlier in this section.

2. The Enter Compliance bit in the Link Control 2 register has changed to 0b (from 1b) since entering Polling.Compliance.
3. The component is a Downstream component, the Enter Compliance bit in the Link Control 2 register is set to 1b and an EIOS has been detected on any Lane. The Enter Compliance bit is reset to 0b when this condition is true.

…