Implementing PCIe Advanced Error Reporting

Gene Saghi

Vice President

Astek Corporation

Topics

- PCIe Advanced Error Classification
- Advanced Error Reporting Capability Structure
- Error Header Logging
- Error Detection
- Error Pollution
- Advisory Non-Fatal Errors
- Error Messages
- Error Handling Flowchart
- Verification

Perspective

- The material in this presentation is derived from endpoint implementations, but much of it applies to other implementations
- The focus is primarily on the transaction layer aspects of advanced error reporting (AER)
-  PCIe base specification 1.0a, together with the errata on AER
  - ✓ Are difficult to follow and can mislead you
  - ✓ Do not include the requirements on error pollution and advisory non-fatal errors
- Get a copy of PCIe base specification 1.1 and use that!

Advanced Error Classification

- Correctable errors
  - ✓ PCIe protocol allows recovery without loss of data
  - ✓ Examples:
    - Receiver error (incorrect running disparity, 8b/10b decode error, etc.)
    - Replay timer timeout
- Uncorrectable errors
  - ✓ Two subclasses:
    - Non-Fatal
      - Transaction is unreliable
      - Requester/Receiver logic or system management software has the opportunity to recover without disturbing other ongoing transactions
      - Link still fully functional
    - Fatal
      - Link and related hardware are unreliable
      - Reset of components on the link generally required

Advanced Error Classification (Cont.)

- Advisory non-fatal errors
  - ✓ Specific non-fatal errors where detecting agent is generally not in best position to determine if error is recoverable or not
  - ✓ Detecting agent sends correctable error message rather than uncorrectable error message
  - ✓ Software decides if the error is recoverable
  - ✓ Example:
    - Completer sends completion with UR/CA status
    - Ultimate receiver of a poisoned TLP

Advanced Error Classification (Cont.)

- Error pollution
  - ✓ Errors have the potential of being reported at more than one level
  - ✓ To allow root cause to be more easily determined, multiply-reported errors should be avoided
  - ✓ Error pollution establishes an error precedence for the reporting of errors

PCIe AER Extended Capability Structure

	Byte Offset
PCIe Enhanced Capability Header	00h
Uncorrectable Error Status Register	04h
Uncorrectable Error Mask Register	08h
Uncorrectable Error Severity Register	0Ch
Correctable Error Status Register	10h
Correctable Error Mask Register	14h
Advanced Error Capabilities and Control Register	18h
Header Log Register	1Ch
Root Error Command	2Ch
Root Error Status	30h
Error Source Identification Register / Correctable Error Identification Register	34h

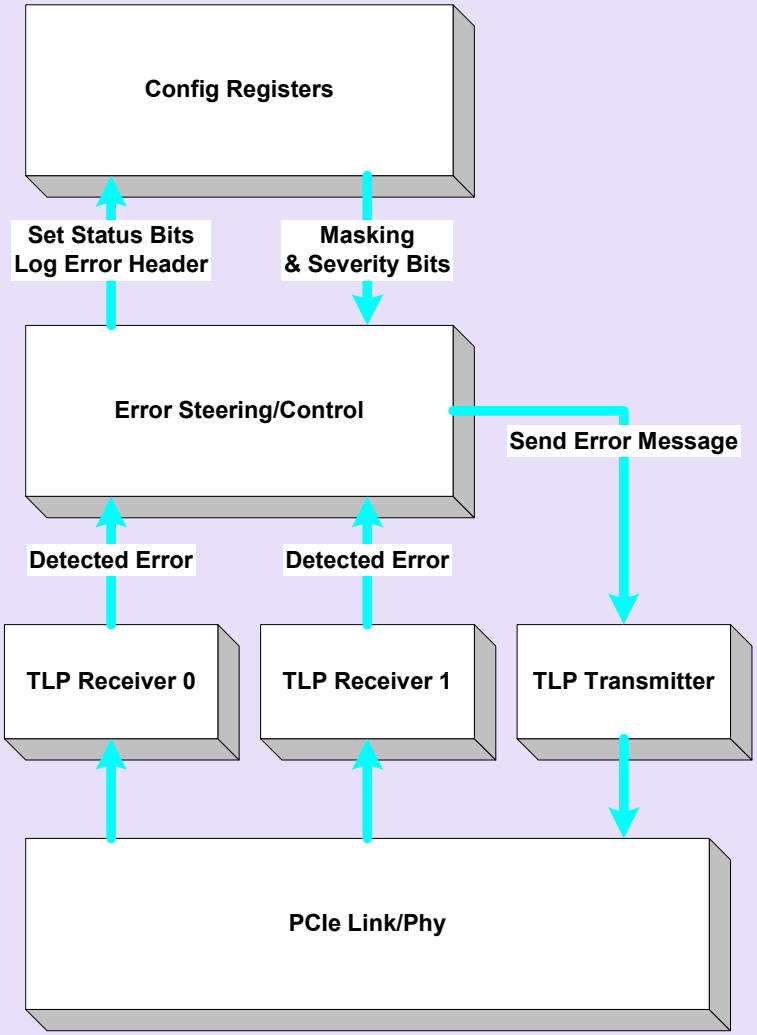
AER Capability Structure (Cont.)

- Uncorrectable errors can be individually masked
- Uncorrectable errors can be individually mapped to fatal or non-fatal
- Correctable errors can be individually masked, but there is no severity mapping
- Advisory non-fatal errors are reported as correctable only when the associated severity bit is set to non-fatal

Error Header Logging

- Error Header Logging
  - ✓ Headers logged only on first error detected
  - ✓ Headers logged only when the error is a result of receiving a TLP
  - ✓ The header log registers are valid only when an error bit is set in the uncorrectable error status register
  - ✓ When logging a header, the first error pointer must also be set
  - ✓ When the error status bit corresponding to the first error pointer is cleared, the first error pointer and the error header are no longer valid

Example AER Block Diagram



Error Detection - Malformed TLP

- Malformed TLP uncorrectable error
  - ✓ TLP length cannot exceed Max Payload Size
  - ✓ TLP length and TD fields must match observed TLP size
  - ✓ TLP Address/Length can not cross a 4-KB boundary (optional check)
  - ✓ I/O TLP rule violation (optional check)
    - Type Class, Attribute, & Last DW BE must all be 0
    - Length must be 1
  - ✓ Many others, see Table 6-2 in Section 6.2.7 for a list (19 of them!) of specification references

Malformed TLP (Cont.)

-  Don't forget to consider the case of a missing EOP symbol
  - ✓ Link/Phy core may not detect a missing EOP for some time
  - ✓ For most TLPs, just count down the packet length and stop moving data when that reaches 0.
  - ✓ For completions, the byte count can be used to insure too much data is not passed on.
-  Receivers must not update flow control info. when a malformed TLP is received and when it is unclear as to which buffer to release

Error Detection - ECRC Check Failure

- ECRC support is optional
- ECRC support is reported in the AER Capabilities and Control register
- ECRC generation and checking are enabled in the AER Capabilities and Control register
- If ECRC checking is enabled
  - ✓ Check ECRC on all incoming TLPs w/ digest field
  - ✓ Error reported as advisory non-fatal or as fatal (depending on severity bit)

Error Detection – Unsupported Request

- Unsupported request (UR) uncorrectable error
  - ✓ Least clearly defined of all the errors because much depends on the user implementation
  - ✓ Examples of unsupported requests
    - Unsupported request type – by design or configuration
    - Unsupported message code other than vendor-defined type 1, which is not treated as an error
    - When in D1, D2, or $D3_{hot}$ power states, any received memory or I/O TLP is treated as an unsupported request
    - An unsupported non-posted request causes the receiver to generate a completion with unsupported request status
    - Many others, see Table 6-2 in Section 6.2.7 for a list (14 of them!) of specification references

Unsupported Request (Cont.)

-  Search PCIe 1.1 spec for phrase "unsupported request" and make sure you have covered all of the bases

Error Detection – Poisoned TLP

- Poisoned TLP uncorrectable error
  - ✓ A transmitter that knows it is sending a TLP with bad data, poisons the TLP by setting the EP bit in the header
  - ✓ The receiver of a poisoned configuration write TLP must discard the TLP and return a completion with status UR
  - ✓ The receiver of a poisoned memory, I/O, or message
    - That addresses a control register or control structure must handle it as an unsupported request
    - Otherwise, it may optionally use the data
  - ✓ Support for TLP poisoning is optional
  - ✓ If AER is supported, support for correctly handling poisoned TLPs is not optional

Error Detection - Completion Timeout

- Completion timeout uncorrectable error
  - ✓ Devices that issue split requests must support a completion timeout mechanism
    - Minimum timeout is 50 usec, but recommendation is that minimum be no less than 10 msec
    - Maximum timeout is 50 msec
  - ✓  PCIe 2.0 drafts make the timeout value variable and include configuration register bits for controlling the timeout range

Error Detection – Unexpected Completion

- Unexpected completion uncorrectable error
  - ✓ A receiver that receives a completion that does not match one of its outstanding requests
    - Discards the completion
    - Reports the error as an unexpected completion

Error Detection – Completer Abort

- Completer abort uncorrectable error
  - ✓ Support is optional
  - ✓ A completion with status completer abort (CA) is issued in response to a request that requires a completion if
    - The request violates the device's programming model
    - The device is permanently unable to process the request due to a device-specific error
  - ✓  Any requester that receives a completion with status UR or CA must
    - Free resources and buffer space associated with the request
    - Treat the completion as the final completion for that request

Error Detection

- Receiver overflow
  - ✓ Optional – TLP is received that consumes more than the advertised amount of flow control credit
  - ✓ See section 2.6.1.2
- Flow control protocol error
  - ✓ Optional – one of many flow control protocol rules has been violated
  - ✓ See section 2.6.1

Error Detection – Link/Phy

- Physical layer detected errors
  - ✓ Receiver error
- Link layer detected errors
  - ✓ Bad TLP
  - ✓ Bad DLLP
  - ✓ Replay timeout
  - ✓ Replay Num rollover
  - ✓ Data link layer protocol error
  - ✓ Surprise down

Error Pollution

- Error priority is:
  - ✓ (1) Receiver overflow
  - ✓ (2) Flow control protocol error
  - ✓ (3) ECRC check failed
  - ✓ (4) Malformed TLP
  - ✓ (5) Unsupported request, completer abort, unexpected completion
  - ✓ (6) Poisoned TLP received
- Only one of the above errors shall be reported with respect to the reception of a single TLP

Error Pollution (Cont.)

-  Support for error pollution implies careful attention to timing of errors detected
  - ✓ Consider a configuration write packet received with TC != 0 and with an ECRC error
    - The malformed packet error will likely be seen first
    - To satisfy the error pollution rules, only the ECRC error should be reported
  - ✓ My recommended approach is to report an error only after all TLP checks have been performed

Advisory Non-Fatal Errors

- Advisory non-fatal (ANF) errors (see 6.2.3.2.4)
  - ✓ For following detected unmasked, non-fatal errors
    - Set appropriate status bit in uncorrectable error status register
    - Send correctable error message (if enabled) and set ANF error status bit in correctable error status register
    - Set first error pointer and log error header per usual rules
  - ✓ Completer must treat an unsupported request or completer abort as an ANF error

ANF Errors (Cont.)

- ✓ Error cases that must be treated as ANF errors are
  - Completer that returns a completion with status UR or CA
  - An intermediate receiver of a TLP with a non-fatal error (e.g., poisoned TLP, TLP with bad ECRC)
  - Ultimate receiver of a poisoned TLP
  - Requester of a completion that times out may repeat the request multiple times
    - Each timeout must be handled as an ANF error
    - Final timeout handled as a non-fatal error if requester stops trying to issue the request
  - Receiver of an unexpected completion

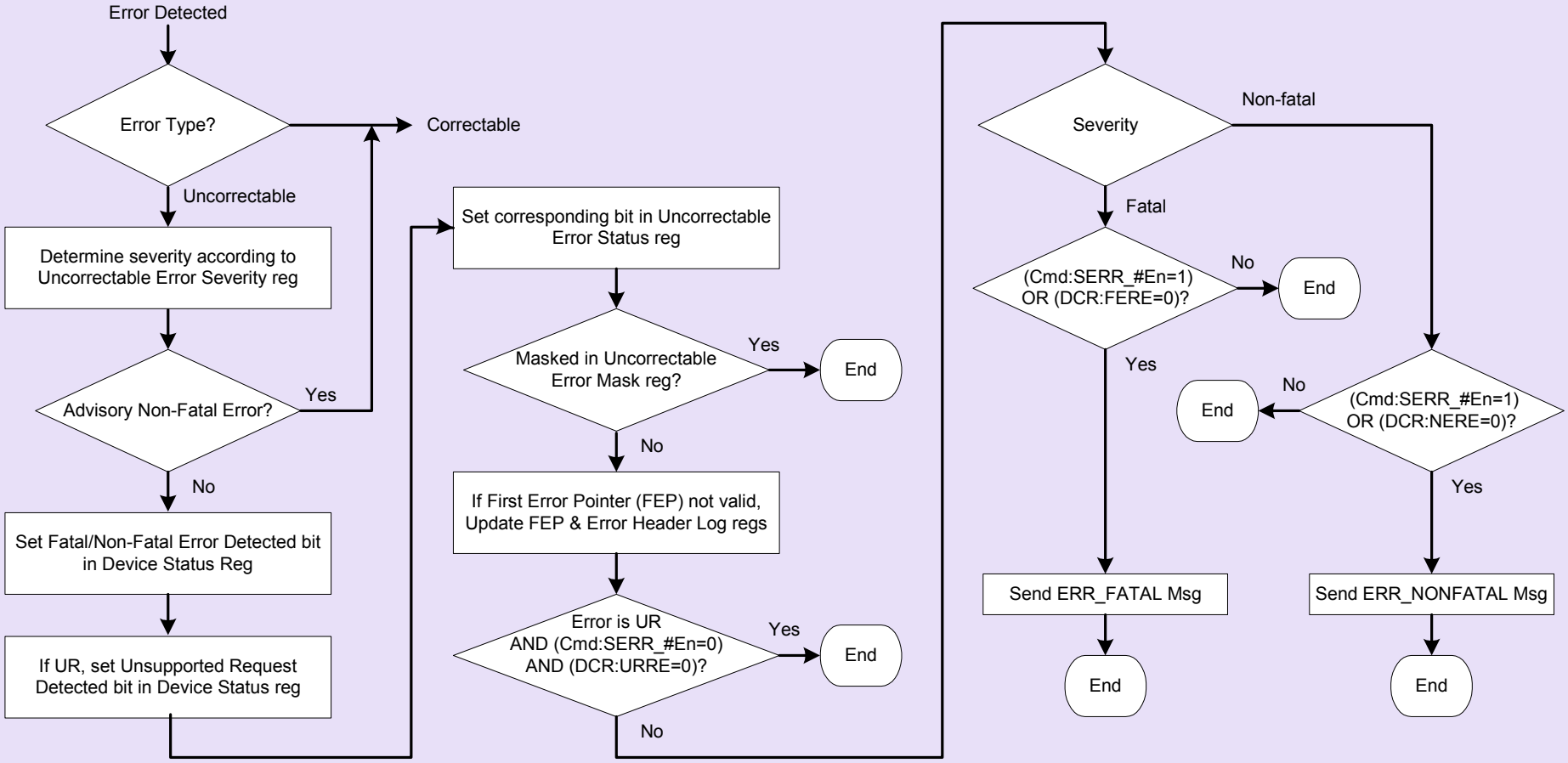
Error Messages

- Endpoints, if enabled to do so, send error messages when errors are detected
- Three types of messages
  - ✓ Correctable
  - ✓ Non-fatal
  - ✓ Fatal
- Multiple errors at same level may be merged
- At least one error message must be sent for detected errors at each severity level

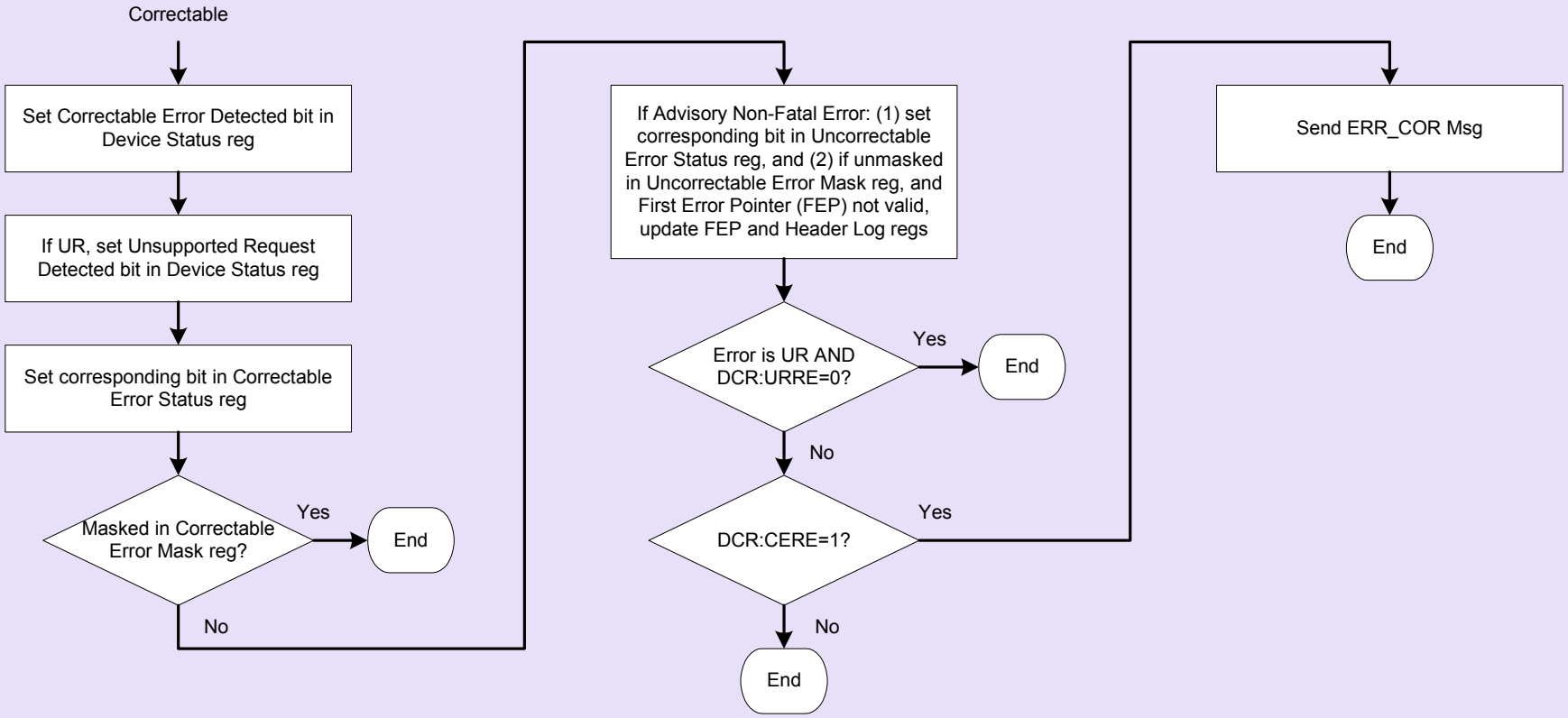
Error Messages (Cont.)

-  Error messages can be enabled/disabled in several different places
  - ✓ SERR# Enable bit in PCI Command Register enables fatal and non-fatal error messages
  - ✓ PCIe Device Control register has enables for fatal, non-fatal, and correctable error messages
  - ✓ Either method can be used to enable fatal and non-fatal messages
  - ✓ Remember that multifunction devices maintain these registers on a per function basis

Uncorrectable Error Flowchart



Correctable Error Flowchart



Logging Errors

-  Some errors are not function-specific
  - ✓ All physical layer errors
  - ✓ All data link layers
  - ✓ ECRC failure
  - ✓ Unsupported request caused by no function claiming a TLP
  - ✓ Flow control protocol error
  - ✓ Malformed TLP
- Multi-function devices register and log such errors in all functions
- Multi-function devices must send
  - ✓ One error message per mapped severity level when one or more functions are enabled to send the error message
  - ✓ No error message only when none of the functions is enabled to send the error message

Logging Errors (Cont.)

-  Make sure you are considering the correct function number
  - ✓ For received configuration packets, the function number is provided in the TLP
  - ✓ For memory and I/O packets, the function number is determined from the address provided in the TLP
  - ✓ For completion packets the function number is part of the requester ID
-  Clearing an AER status bit does not clear any associated PCI status bit
-  Review PCI status register in PCIe spec to determine when to set and clear those bits

Verification

-  Verify, verify, verify!
  - ✓ Error conditions are difficult to test with random simulations
  - ✓ Write good directed simulations and check
    - Correct setting and clearing of error status bits
      - When masked, unmasked, and re-mapped
      - Check that error pollution rules are followed
      - Check advisory non-fatal error reporting
    - Correct logging of error headers
    - Correct setting of the first error pointer
    - Enabling and disabling of error messages
    - Special multifunction error reporting cases

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

For more information please go to
www.pcisig.com

Implementing PCIe Advanced Error Reporting

Gene Saghi

Vice President

Astek Corporation