Gen-Z Link-level Reliability (LLR)

April 2019

This presentation covers link-level reliability. With the exception of the P2P * OpClasses, LLR is an optional capability that is applied based on solution-specific needs or operating environments.

Disclaimer

This 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. Gen-Z Consortium disclaims all liability for infringement of proprietary rights, relating to use of information in this document. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

Gen-Z is a trademark or registered trademark of the Gen-Z Consortium.

All other product names are trademarks, registered trademarks, or servicemarks of their respective owners.

All material is subject to change at any time at the discretion of the Gen-Z Consortium

http://genzconsortium.org/

Link-level Reliability (LLR)

- LLR provides Reliable Delivery of end-to-end packets between interface pairs
  - Reliable Delivery is defined as uncorrupted, exactly-once, in-order end-to-end packet delivery.
- LLR is intended for solutions that require Reliable Delivery between interface pairs
  - For example, component interfaces that support P2P-Core, P2P-Coherency, or P2P-Vendor-defined OpClasses are required to support LLR
    - Reliable Delivery simplifies implementation and improves end-to-end performance
  - For example, LLR reduces end-to-end packet retransmission due to transient errors on "noisy" paths
    - Errors that trigger link resynchronization, ECRC errors, etc.
- LLR does not modify the end-to-end packet format
  - Each interface maintains an implicit sequence number associated with each transmitted end-to-end packet
  - Each interface maintains a copy of the end-to-end packet until the associated acknowledged

GEN Z

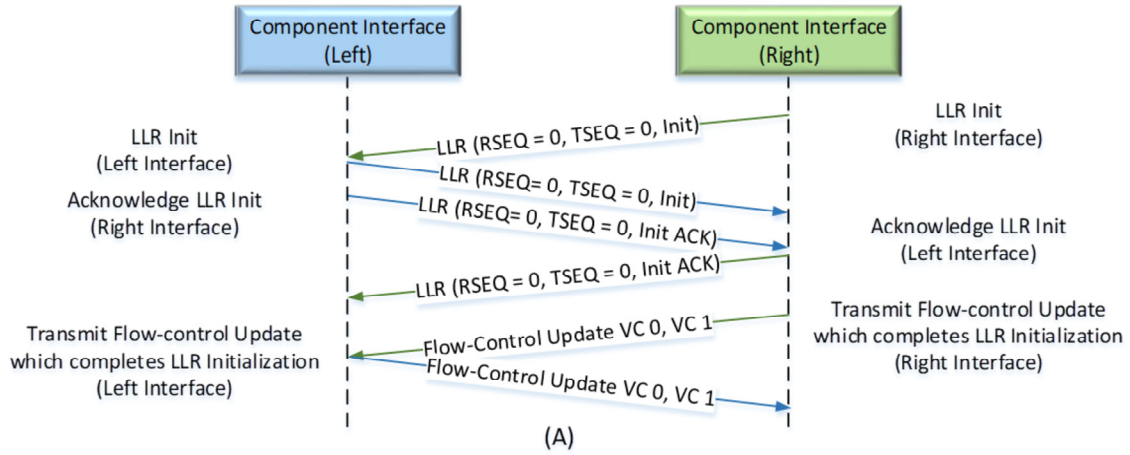
LLR is used on a link-hop basis. It provides reliable delivery of end-to-end packets between a pair of interfaces connected by a link.

LLR is mandatory for P2P-Core, P2P-Coherency, and P2P-Vendor-defined OpClass solutions, and optional for all other OpClasses. LLR enables memory operations to be pipelined across a link in the order posted; this improves performance and simplifying implementations.

LLR should be used in solutions where transient errors are more common, and the performance impact of end-to-end retransmission is application visible. For example, a link with a BER greater than 10e-12 or in a longer distance cable solution (e.g., 50-100 m)

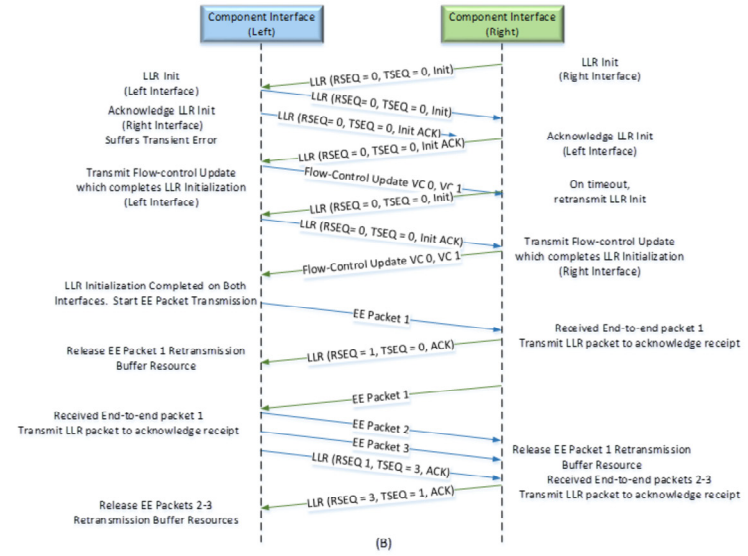
Unlike other technologies, Gen-Z LLR does not require end-to-end packet modifications or additional link layer protocol bytes which degrade performance. Instead, each Gen-Z interface maintains an implicit sequence number associated with each end-to-end packet and a retransmission queue. Once an end-to-end packet is acknowledged, retransmission resources are released.

LLR Initialization with Normal Exchange



This slide illustrates LLR initialization. LLR uses a 3-way exchange to communicate the implicit transmit and receive sequence numbers. A LLR Init is transmitted every 8192 UI until it is acknowledged by a LLR Init ACK. Upon receipt of a LLR Init ACK, the interface transmits explicit flow-control packets that complete LLR initialization and enable end-to-end packets to be transmitted Periodic transmission based on UI enables the protocol to operate across multiple signaling rates and link widths. The larger UI enables the protocol to operate across relatively long cable lengths.

LLR Initialization with Transient Error

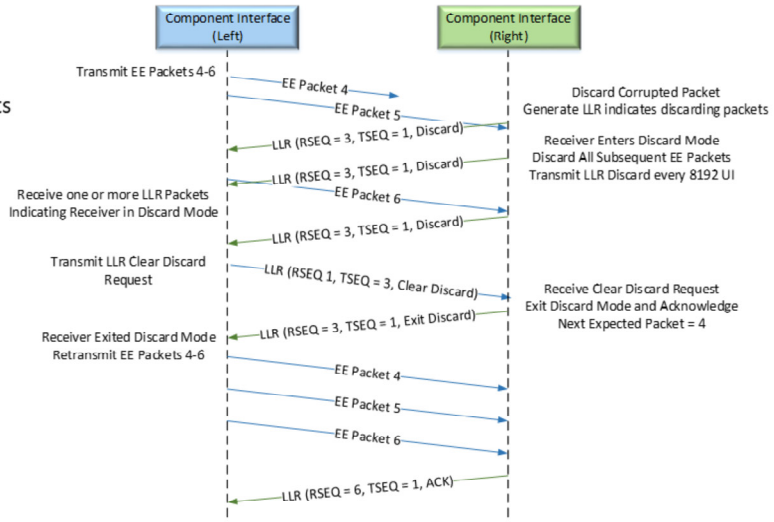


GEN Z

This slide illustrates LLR initialization where the LLR Init ACK transmitted by the left interface is lost due to a transient error.

LLR Error Recovery

- Upon detecting a packet transient error
  - Orange discards packet
  - Orange discards all subsequent EE packets
  - Orange transmits LLR Discard
    - Indicates present RSEQ and RSEQ
  - Purple receives LLR Discard and stops EE
  - Purple transmits LLR Clear Discard
  - Orange receives LLR Clear Discard and transmits LLR Exit Discard
  - Purple resumes transmitting EE packets post last acknowledged EE packet
  - Orange LLR acknowledges EE packets
- LLR Discard transmitted every 8192 UI
  - If exceed 256 LLR Discard without a LLR Clear Discard, then retrain physical



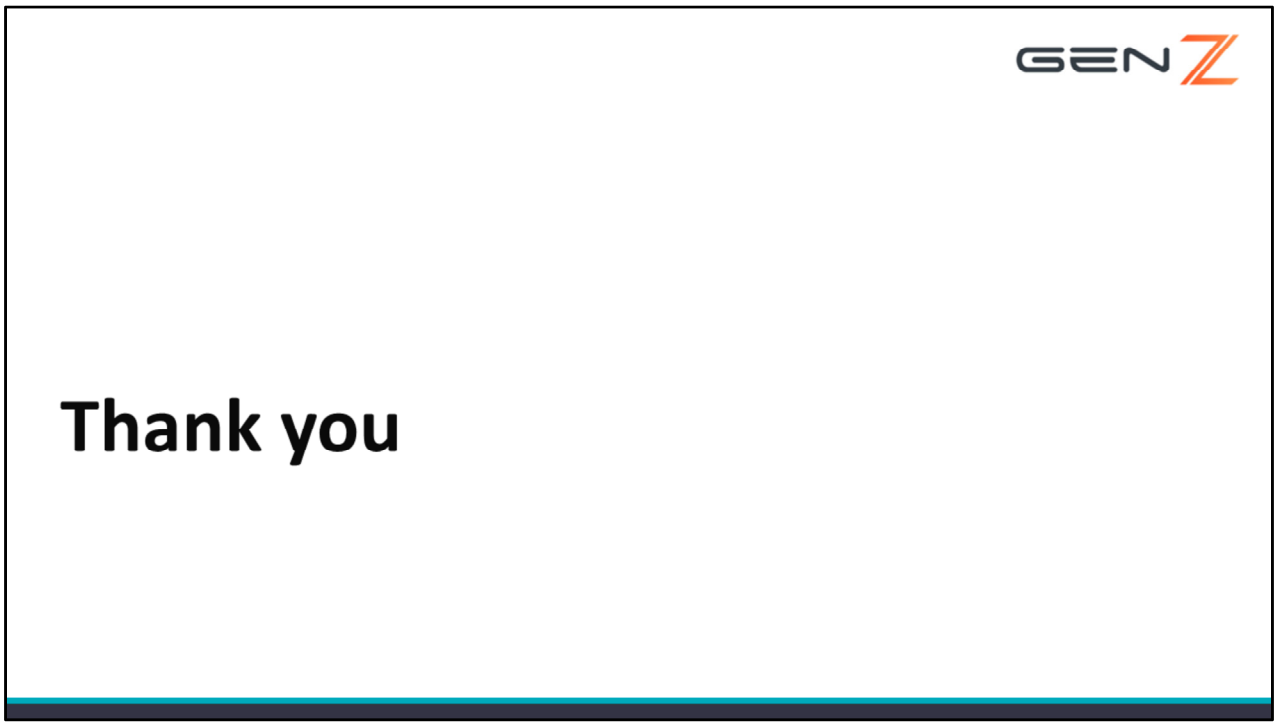
This slide illustrates the steps taken upon detecting a transient error. Once a transient error is detected, the receiving interface silently discards all new end-to-end packets, and transmits a LLR Discard to inform the transmitter of the error. Upon receipt of the LLR Discard, the transmitter determines the last successfully received packet and releases all resources associated with acknowledged packets. It then transmits a LLR Clear Discard to indicate it is ready to start retransmission. Upon receipt of a LLR Exit Discard, the transmitter retransmits all outstanding end-to-end packets and resumes normal operation.

Additional LLR Requirements

- Each interface shall maintain an interface-local LLR Timer (LLRT)
  - LLRT period shall be 2^{20} UI
- Each interface shall transmit a LLR ACK packet when the LLRT expires
  - If an interface fails to receive at least one LLR ACK for 4 consecutive LLRT periods, then retrain physical
- If an EE packet is unacknowledged for 3 consecutive LLRT periods, then initiate the LLR retransmission
- RSEQ indicates the number of EE packets received on any VC
  - Interface monotonically increments RSEQ modulo 2^{16} for each successfully received EE packet
- TSEQ indicates the number of EE packets transmitted on any VC
  - Interface monotonically increments TSEQ modulo 2^{16} for each transmitted EE packet
  - Interface suspends EE packet transmission if the number of unacknowledged packets equals $(2^{16} - 1)$
- LLR shall be triggered on any error that triggers link resynchronization
- LLR may be triggered on any ECRC error (default / required behavior, selectively disable if both agree)

GEN Z

This slide covers additional LLR requirements, e.g., each interface maintains a LLR Timer to ensure that a LLR ACK packet is transmitted at least once every 2e20 UI.



This concludes this presentation. Thank you.