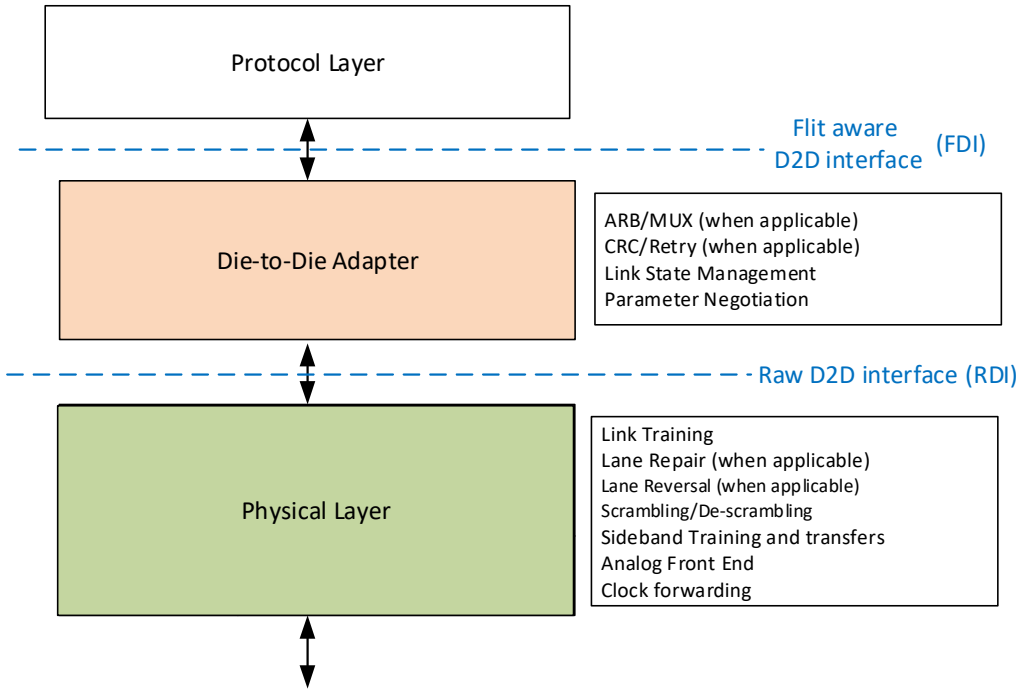
UCIe Protocol

Agenda for Protocol Part

- Protocol Layer
- Die-to-Die (D2D) Adapter
- Logical Physical Layer

UCIe Protocol Layer



- Protocols supported
  - PCIe
  - CXL
  - Streaming: Vendor defined protocol
- Multiple Flit formats are permitted as a transport mechanism
- Raw formats are supported where D2D Adapter CRC/Retry is bypassed – supported for all protocols

PCIe non-Flit mode and CXL 68B Flit Mode

- Supported Formats
  - Raw
    - Retry/CRC if applicable is the responsibility of the Protocol Layer
  - 68B Flit Formats
    - Retry/CRC is the responsibility of the D2D Adapter
    - Framing follows CXL.io 68B Flit packing/unpacking rules
    - Ack/Nak, PM DLLPs are not used
      - LCRC, DLLP CRC driven to 0b by the Transmitter; ignored by the Receiver
      - Sequence number, Frame CRC, Frame parity within the STP token are driven to 0 by Transmitter
    - Protocol Layer drives the 64B per Flit of Protocol information on FDI.
    - D2D Adapter adds the Flit header and CRC bytes (additional 4B to make a 68B Flit) and performs the necessary byte shifting
    - For CXL protocol, the Arb/Mux functionality is in the D2D Adapter
- No 8b/10b or 128b/130b encodings are needed for D2D transfers

PCIe Flit mode

- Supported Formats
  - Raw
    - Retry/CRC/FEC, if applicable, is the responsibility of the Protocol Layer
  - Standard 256B Flit
    - Retry/CRC is the responsibility of the D2D Adapter
    - Framing follows PCIe 256B Flit packing/unpacking rules
    - PM, Link Management DLLPs are not used
    - Flow Control DLLPs are sent separately over FDI, and the D2D Adapter inserts them into the Flit (don't go through the Retry buffer in the D2D Adapter)

CXL 256B Flit Mode

- Supported Formats
  - Raw
    - Retry/CRC/FEC, if applicable, is the responsibility of the Protocol Layer
  - Standard 256B Flit and Latency Optimized Flits
    - Retry/CRC is the responsibility of the D2D Adapter
    - Framing follows CXL 256B Flit Mode packing/unpacking rules
    - PM, Link Management DLLPs are not used
    - For CXL.io, Flow Control DLLPs are sent separately over FDI, and the D2D Adapter inserts them into the Flit (don't go through the Retry buffer in the D2D Adapter)
- Arb/Mux functionality in the D2D Adapter

Streaming Protocol

- UCIe 1.0 only supports Streaming Protocol in Raw format

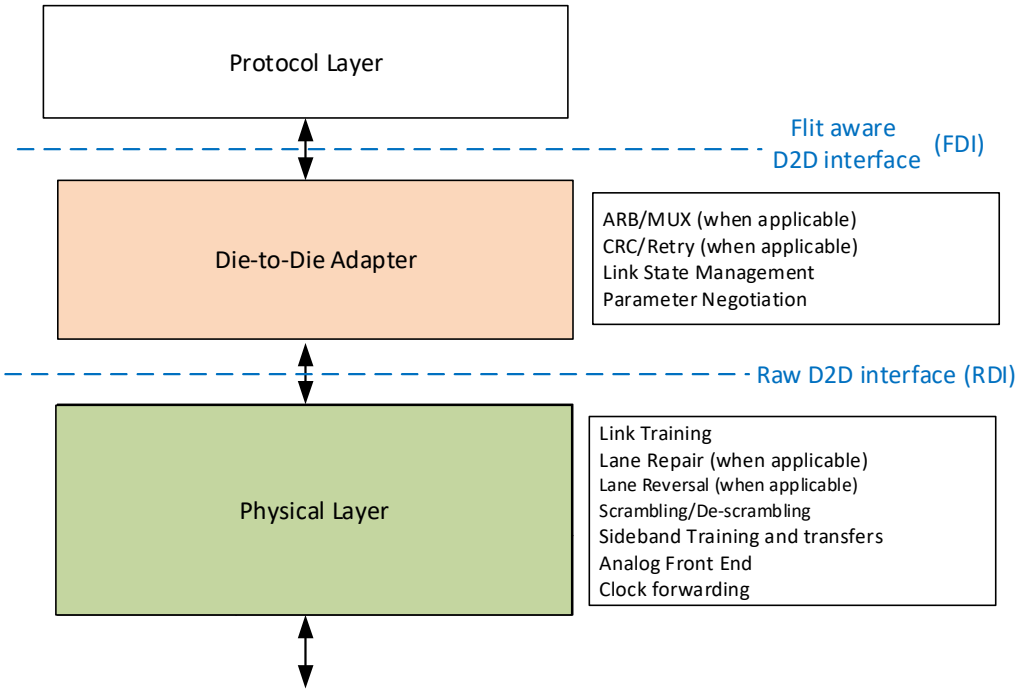
Die-to-Die Adapter

Overview: D2D Adapter

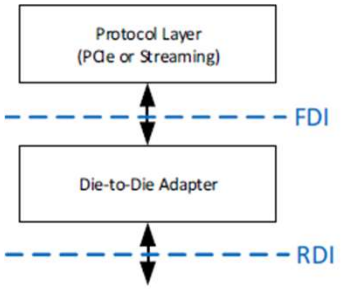
- Functionality Partitioning
- Example Configurations
- Flit Format details
- Protocol and Flit Format Matrix
- State Machine Hierarchy
- Parameter Exchanges
- PM flow example

Functionality Partitioning

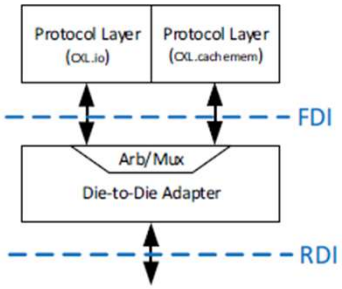
- For CXL, ARB/MUX functionality is absorbed in the D2D Adapter
- Lightweight CRC computation
- Flit Retry mechanism leveraged from PCIe Flit Mode
- Link State Management for Reset/Active/PM/Error flows
- D2D specific or Protocol specific parameters negotiated with remote Link partner



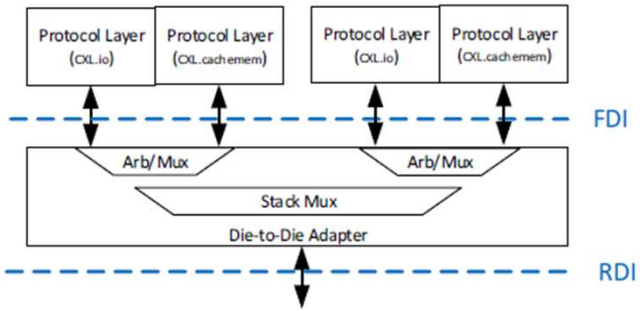
Example Configurations



(a) Single Protocol

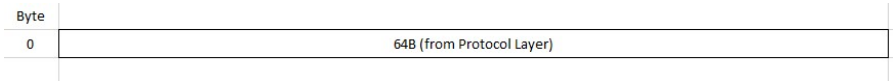


(b) Single CXL stack



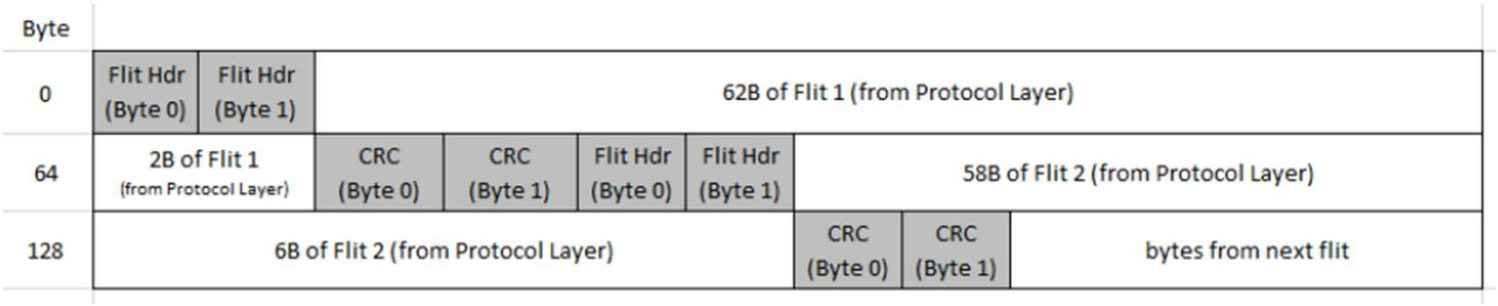
(c) Two CXL stacks multiplexed inside the adapter

Raw Format (Format 1)



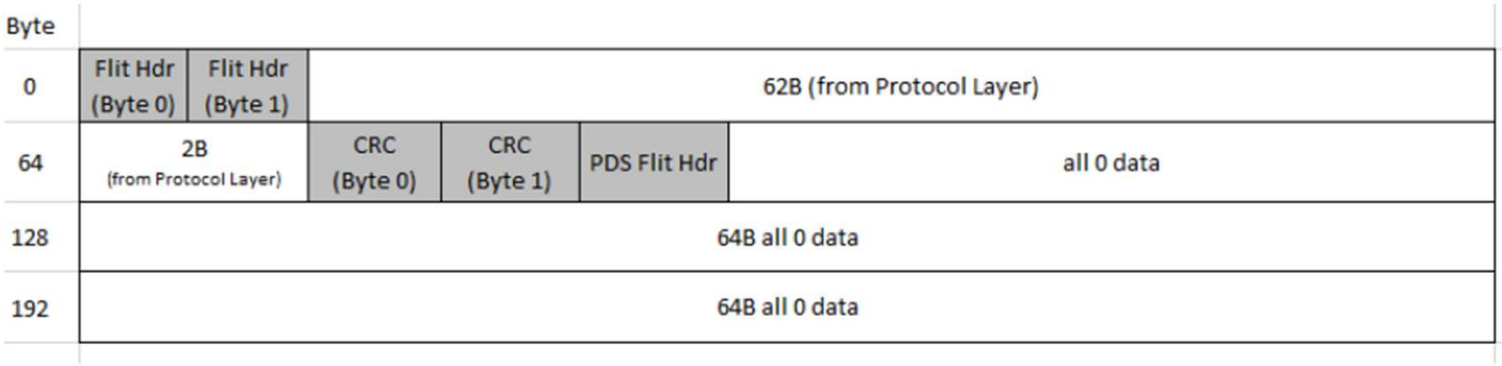
- All examples show 64B data width on FDI
- For Raw format, all bytes are populated by the Protocol Layer, and D2D Adapter data path is bypassed
- D2D Adapter simply forwards all the bytes to RDI without any modifications or additions
- Raw format is permitted for all Protocols
  - Mandatory for D2D Adapter to support this if supporting Streaming Protocol

68B Flit Format (Format 2)



- Used for PCIe non-Flit Mode and CXL.io 68B Flit Mode protocols
- Protocol Layer presents 64B of the Flit on FDI
- D2D Adapter adds 2B Flit Header, 2B CRC and performs the required barrel shifting before transmitting over RDI
  - CRC covers the 2B Flit header as well as the 64B of Protocol information
- Flit header carries protocol identifier, stack identifier, sequence number, Ack/Nak completion, pause of data stream indication

68B Flit Format (Format 2) contd.



Byte						
0	Flit Hdr (Byte 0)	Flit Hdr (Byte 1)	62B (from Protocol Layer)			
64	2B (from Protocol Layer)		CRC (Byte 0)	CRC (Byte 1)	PDS Flit Hdr	all 0 data
128	64B all 0 data					
192	64B all 0 data					

- Since 68B is not a multiple of the number of Lanes, a pause of data stream indication is needed to make sure the Receiver understands when the Transmitter has stopped sending new Flits
- Denoted by a special Flit header followed by all 0 data (such that at least 256B of data is always transferred in this example)

Standard 256B End Header Flit Format (Format 3)

Byte									
0	Flit Chunk 0 64B (from Protocol Layer)								
64	Flit Chunk 1 64B (from Protocol Layer)								
128	Flit Chunk 2 64B (from Protocol Layer)								
192	Flit Chunk 3 44B (from Protocol Layer)	Flit Hdr (Byte 0)	Flit Hdr (Byte 1)	DLP Bytes 2:5	10B Reserved	CRC0 (Byte 0)	CRC0 (Byte 1)	CRC1 (Byte 0)	CRC1 (Byte 1)

- Used for PCIe Flit Mode protocol
- Protocol Layer sends the Flit over FDI, drives 0b on reserved bits and the bits filled in by the D2D Adapter (Flit Marker populated by Protocol Layer in DLP Bytes 2:5)
  - DLLPs sent/received over separate signals on FDI
- D2D Adapter fills in DLLP into DLP Bytes 2:5 if Flit Marker is not present

Standard 256B Start Header Flit Format (Format 4)

CXL.cachemem

Byte								
0	Flit Hdr (Byte 0)	Flit Hdr (Byte 1)	62B of Flit Chunk 0 (from Protocol Layer)					
64	Flit Chunk 1 64B (from Protocol Layer)							
128	Flit Chunk 2 64B (from Protocol Layer)							
192	50B of Flit Chunk 3 (from Protocol Layer)			10B Reserved	CRC0 (Byte 0)	CRC0 (Byte 1)	CRC1 (Byte 0)	CRC1 (Byte 1)

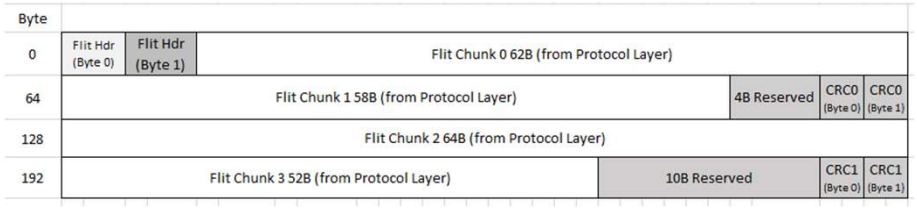
CXL.io

Byte								
0	Flit Hdr (Byte 0)	Flit Hdr (Byte 1)	Flit Chunk 0 62B (from Protocol Layer)					
64	Flit Chunk 1 64B (from Protocol Layer)							
128	Flit Chunk 2 64B (from Protocol Layer)							
192	Flit Chunk 3 46B (from Protocol Layer)			DLP Bytes 2:5	10B Reserved	CRC0 (Byte 0)	CRC0 (Byte 1)	CRC1 (Byte 0)	CRC1 (Byte 1)

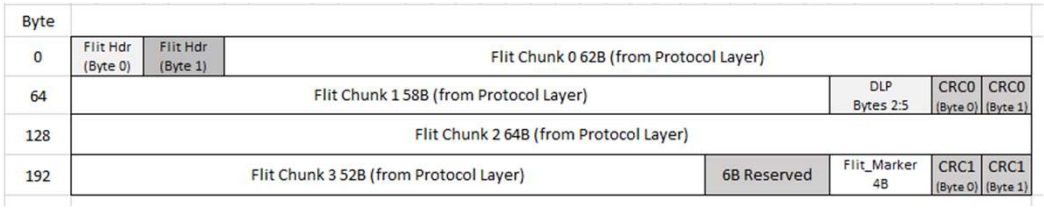
- Used for CXL 256B Flit Mode protocol
- Protocol Layer sends the Flit over FDI, drives 0b on reserved bits and the bits filled in by the D2D Adapter
  - For CXL.io, Flit Marker populated by Protocol Layer in DLP Bytes 2:5. DLLPs sent/received over separate signals on FDI. D2D Adapter fills in DLLP into DLP Bytes 2:5 if Flit Marker is not present
  - For CXL.cachemem, there are no DLP bytes in the Flit, the corresponding bytes of the Flit are used for Payload instead
- Follows the Framing rules for the Standard Flit in CXL Specification

Latency Optimized 256B without Optional bytes (Format 5)

CXL.cachemem



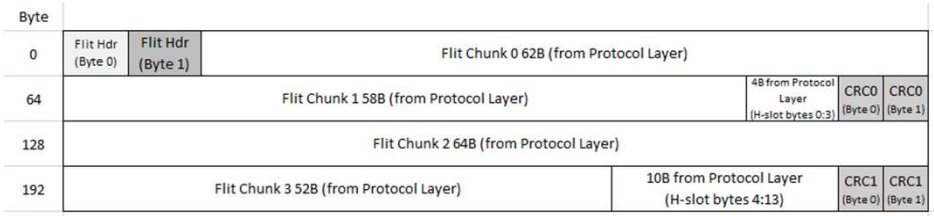
CXL.io



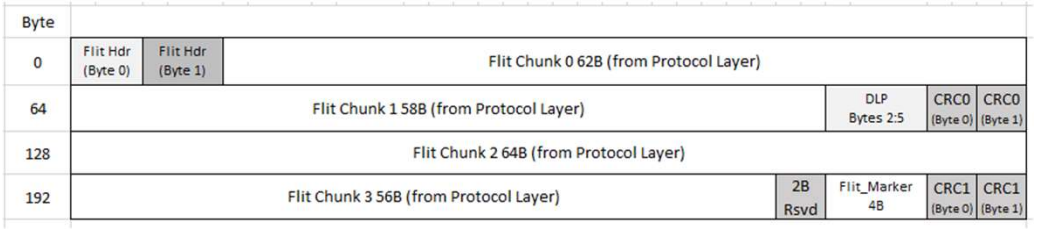
- Used for CXL 256B Flit Mode protocol
- Protocol Layer sends the Flit over FDI, drives 0b on the bits filled in by the D2D Adapter
  - For CXL.io, Flit Marker populated by Protocol Layer in DLP Bytes 2:5. DLLPs sent/received over separate signals on FDI
  - For CXL.cachemem, there are no DLP bytes in the Flit, the corresponding bytes of the Flit are reserved
- Follows the Framing rules for Latency Optimized Flit in CXL Specification

Latency Optimized 256B with Optional bytes (Format 6)

CXL.cachemem



CXL.io



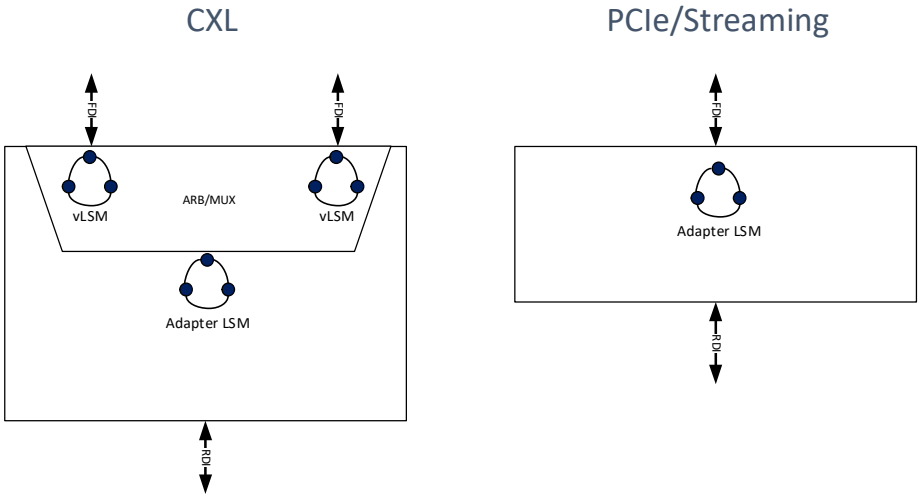
- Used for CXL 256B Flit Mode protocol
- Protocol Layer sends the Flit over FDI, drives 0b on the bits filled in by the D2D Adapter
  - For CXL.io, Flit Marker populated by Protocol Layer in DLP Bytes 2:5. DLLPs sent/received over separate signals on FDI
  - For CXL.cachemem, there are no DLP bytes in the Flit, the corresponding bytes of the Flit are used for payload
- Follows the Framing rules for Latency Optimized Flit in CXL Specification with additional bytes from Protocol Layer for added efficiency
  - CXL.io gets an extra DWord of TLP information
  - CXL.cachemem gets an extra 14 Bytes of payload

Protocol and Flit Format matrix

Format Number	Flit Format Name	PCIe Non-Flit Mode	PCIe Flit Mode	CXL 68B Flit Mode	CXL 256B Flit Mode	Streaming*
1	Raw	Optional	Optional	Optional	Optional	Mandatory
2	68B	Mandatory	N/A	Mandatory	N/A	N/A
3	Standard 256B End Header	N/A	Mandatory	N/A	N/A	N/A
4	Standard 256B Start Header	N/A	N/A	N/A	Mandatory	N/A
5	Latency Optimized 256B without optional bytes	N/A	N/A	N/A	Optional	N/A
6	Latency Optimized 256B with optional bytes	N/A	N/A	N/A	Strongly Recommended	N/A

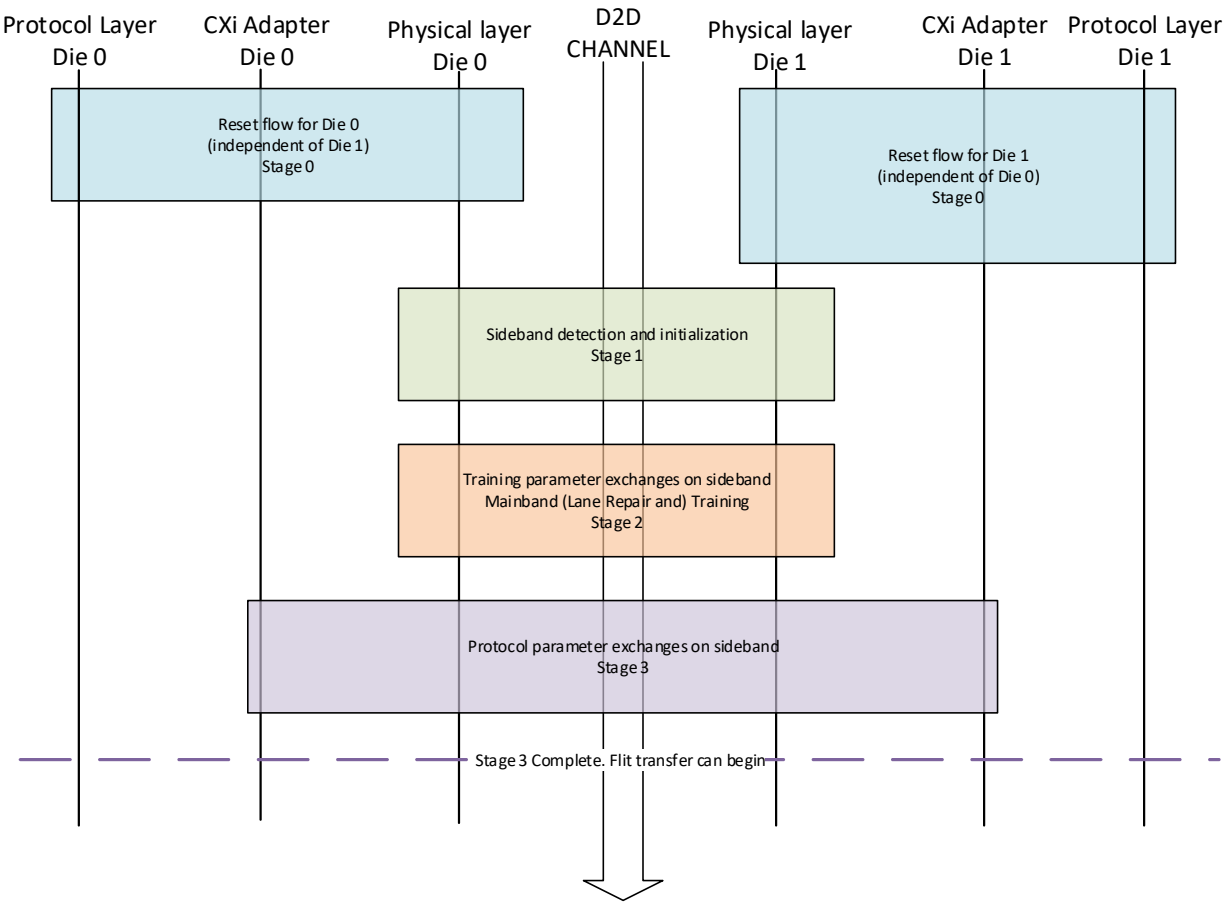
*Streaming column is for D2D Adapter. Protocol Layer interop is vendor specific

State Machine Hierarchy

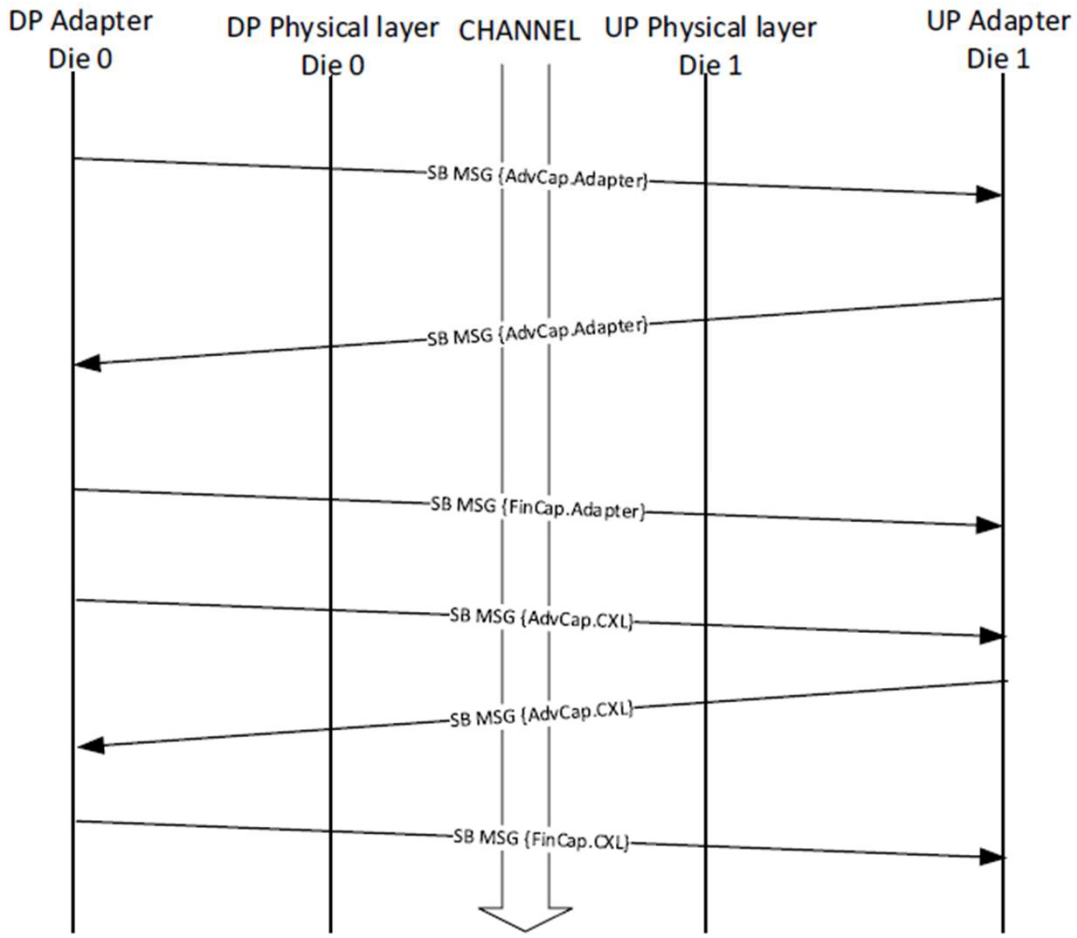


- For CXL, vLSM is exposed on FDI
- For PCIe/Streaming, D2D Adapter LSM is exposed on FDI
- vLSM handshakes with remote Link partner use ALMPs and follow CXL 256B Flit Mode rules and format
- D2D Adapter LSM handshakes with remote Link partner use sideband Link

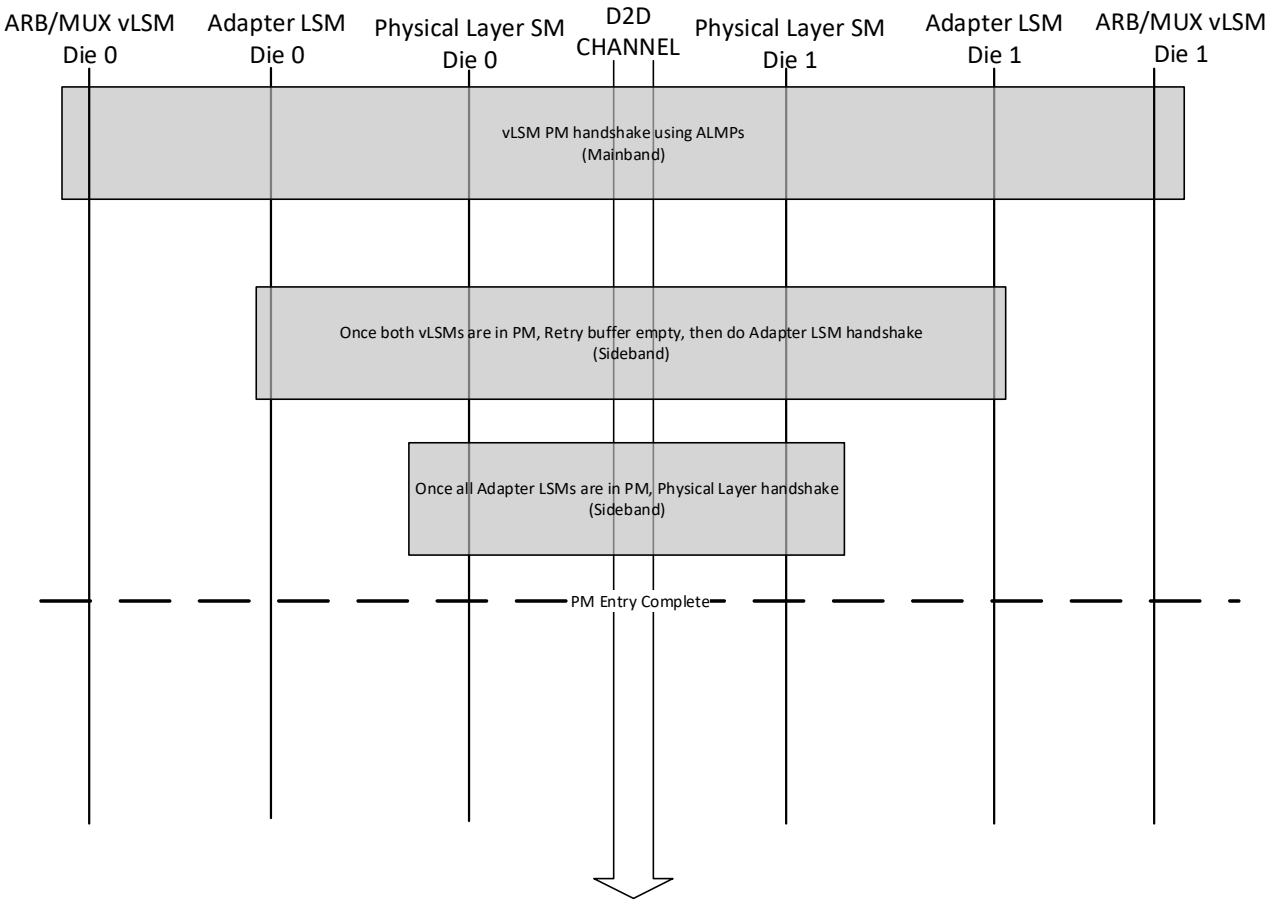
Link Initialization flow



Parameter Exchanges (CXL)



Example PM Entry flow for CXL

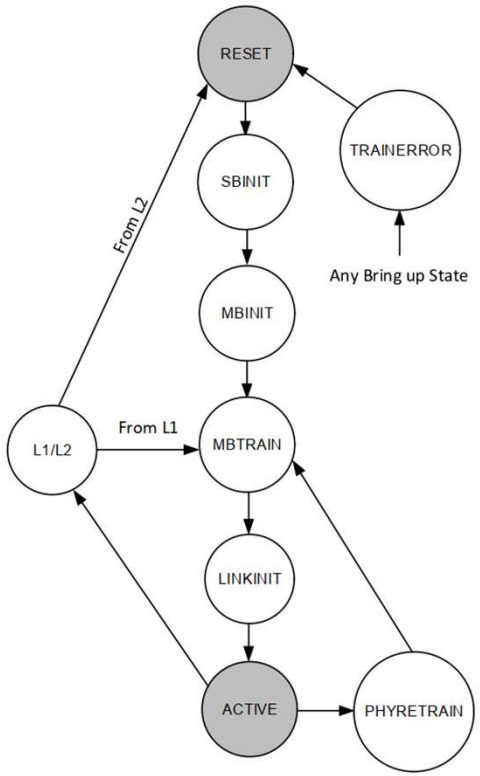


Logical Physical Layer (logPHY)

LOGPHY Functions

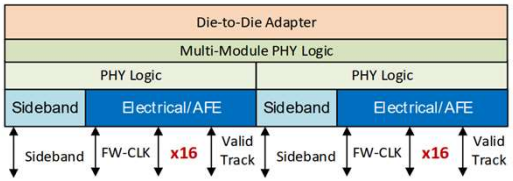
- Byte to Lane mapping for data transmission over Lanes
  - Separate Valid Lane for indicating data transfer
- Interconnect redundancy remapping for Advanced Package configurations
- Width degradation support for Standard Package Configurations
- Scrambling and training pattern generation
- Lane reversal
- Link initialization, training and power management states
- Transmitting and receiving sideband messages

Link Training State Machine



- Sideband (SB) initialization
  - Initialization and repair (advanced package) of SB
- Mainband (MB) initialization
  - Parameter exchange, Clock, Valid & MB repair & MB reversal
- Mainband training
  - Data training, at speed repair/degrade and speed degrade
- LINKINIT
  - Exchange RDI Link management messages
- PHYRETRAIN: Retrain based on Link errors
  - Allow repair or degrade
- L1/L2: Lower power state
- TRAINERROR: For uncorrectable internal errors and Link down conditions
- Active: Transactions are sent and received

Multi-Module support



- One, two or four module per D2D Adapter are allowed
  - Both Advanced and Standard Package
- Standard package example configurations shown here