2.0 Protocol Layer

Universal Chiplet Interconnect express (UCIe) maps PCIe and CXL, as well as any Streaming protocol. Throughout the UCIe Specification, Protocol-related features are kept separate from Flit Formats and packetization. This is because UCIe provides different transport mechanisms that are not necessarily tied to protocol features (e.g., PCIe non-Flit mode packets are transported using CXL.io 68B Flit Format). Protocol features include the definitions of Transaction Layer and higher layers, as well as Link Layer features not related to Flit packing/Retry (e.g., Flow Control negotiations etc.).

The following terminology is used throughout this specification to identify Protocol-level features:

- PCIe Flit mode: To reference Flit mode-related Protocol features defined in *PCIe Base Specification*
- PCIe non-Flit mode: To reference non-Flit mode-related Protocol features defined in *PCIe Base Specification*
- CXL 68B Flit mode: To reference 68B Flit mode-related Protocol features defined in *CXL Specification*
- CXL 256B Flit mode: To reference 256B Flit mode-related Protocol features defined in *CXL Specification*

The following protocol mappings are supported over the UCIe mainband:

- PCIe Flit mode
- CXL 68B Flit mode, CXL 256B Flit Mode: If CXL is negotiated, each of CXL.io, CXL.cache, and CXL.mem protocols are negotiated independently.
- Streaming protocol: This offers generic modes for a user defined protocol to be transmitted using UCIe.
- Management Transport protocol: This allows transport of manageability packets.

Note: RCD/RCH/eRCD/eRCH are not supported. PCIe non-Flit Mode is supported using CXL.io 68B Flit Format as the transport mechanism.

The Protocol Layer requirements for interoperability are as follows:

- A Protocol Layer must support PCIe non-Flit mode if it is advertising the 68B Flit Mode parameter from Table 3-1.
- If a Protocol Layer supports CXL 256B Flit Mode, it must support PCIe Flit Mode and 68B Flit Mode as defined in *CXL Specification* for CXL.io protocol.
- A Protocol Layer advertising CXL is permitted to only support CXL 68B Flit Mode without supporting CXL 256B Flit Mode or PCIe Flit Mode

IMPLEMENTATION NOTE

Table 2-1 summarizes the mapping of the above rules from a specification version to a protocol mode.

Table 2-1. Specification to protocol mode requirements

Native Specification Supported[a]	PCIe Non-Flit Mode	CXL 68B Flit Mode	CXL 256B Flit Mode	PCIe Flit Mode
PCIe	Mandatory	N/A	N/A	Optional
CXL 2.0	Mandatory (for CXL.io)	Mandatory	N/A	N/A
CXL 3.0	Mandatory (for CXL.io)	Mandatory	Mandatory	Mandatory (for CXL.io)

a. The same table applies to derivative version numbers for the specifications.

The Die-to-Die (D2D) Adapter negotiates the protocol with the remote Link partner and communicates it to the Protocol Layer(s). For each protocol, UCIe supports multiple modes of operation (that must be negotiated with the remote Link partner depending on the advertised capabilities, Physical Layer Status as well as usage models). These modes have different Flit Formats and are defined to enable different trade-offs around efficiency, bandwidth and interoperability. The spectrum of supported protocols, advertised modes and Flit Formats must be determined at SoC integration time or during the Die-specific reset bring up flow. The Die-to-Die Adapter uses this information to negotiate the operational mode as a part of Link Training and informs the Protocol Layer over the Flit-aware Die-to-Die Interface (FDI). See Section 3.2 for parameter exchange rules in the Adapter.

The subsequent sections provide an overview of the different modes from the Protocol Layer's perspective, hence they cover the supported formats of operation as subsections per protocol. The Protocol Layer is responsible for transmitting data over FDI in accordance with the negotiated mode and Flit Format. The illustrations of the Flit Formats in this chapter show an example configuration of a 64B data path in the Protocol Layer mapped to a 64-Lane module of Advanced Package configuration on the Physical Link of UCIe. Certain Flit Formats have dedicated bit positions filled in by the Adapter, and details associated with these are illustrated separately in Chapter 3.0. For other Link widths, see the Byte to Lane mappings defined in Section 4.1.1. Figure 2-1 shows the legend for color-coding convention used when showing bytes within a Flit in the Flit Format examples in the UCIe Specification.

Figure 2-1. Color-coding Convention in Flit Format Byte Map Figures

Color Shading	Description
	Some bits populated by the Protocol Layer, some bits populated by the Adapter.
	All bits populated by Adapter.
	All bits populated by the Protocol Layer.

2.1 PCIe

UCIe supports the Flit Mode defined in *PCIe Base Specification*. See *PCIe Base Specification* for the protocol definition. UCIe supports the non-Flit Mode using the CXL.io 68B Flit Formats as the transport mechanism. There are five UCIe operating formats supported for PCIe, and these are defined in the subsections that follow.

2.1.1 Raw Format

This format is optional. All bytes are populated by the Protocol Layer. The intended usage is for UCIe Retimers transporting PCIe protocol. An example usage of this format is where a CPU and an I/O Device are in different Rack/chassis and connected through a UCIe Retimer using Off-Package Interconnect as shown in Figure 1-2. Retry, CRC and FEC (if applicable) are taken care of by the Protocol Layer when using Raw Format. It is strongly recommended for the UCIe Retimers to check and count errors using either the parity bits of the 6B FEC or the Flit Mode 8B CRC defined in *PCIe Base Specification* for this mode to help characterize the Off Package Interconnect (to characterize or debug the Link that is the dominant source of errors). See Section 3.3.1 as well.

2.1.2 Standard 256B End Header Flit Format

This format is mandatory when PCIe Flit Mode protocol is supported. It is the standard Flit Format defined in *PCIe Base Specification* for Flit Mode and the main motivation of supporting this Flit Format is to enable interoperability with vendors that only support the standard PCIe Flit Formats. The Protocol Layer must follow the Flit Formats for Flit transfer on FDI, driving 0 on the fields reserved for Die-to-Die Adapter. The PM and Link Management DLLPs are not used over UCIe. The other DLLPs (that are applicable for PCIe Flit Mode) and Flit Status definitions follow the same rules including packing as defined in *PCIe Base Specification*. It is strongly recommended for implementations to optimize out any 8b/10b, 128b/130b, and non-Flit Mode related CRC/Retry or framing logic from the Protocol Layer in order to obtain area and power efficient designs for UCIe applications. Portions of the DLP bytes must be driven by the Protocol Layer for Flit_Marker assignment; see Section 3.3.3 for details of the Flit Format.

2.1.3 68B Flit Format

This mode is mandatory when PCIe protocol or CXL protocol is supported. The transport mechanism for this is the same as CXL.io 68B Flit Formats. See Section 2.3.2 for the CXL.io DLLP rules that apply for Non-Flit Mode for PCIe as well. It is strongly recommended for implementations to optimize out any 8b/10b, 128b/130b and non-Flit Mode related CRC/Retry logic from the Protocol Layer in order to obtain area and power efficient designs for UCIe applications. To keep the framing rules consistent, Protocol Layer for PCIe non-Flit mode must still drive the LCRC bytes with a fixed value of 0, and the Receiver must ignore these bytes and never send any Ack or Nak DLLPs. Framing tokens are applied as defined for CXL.io 68B Flit Mode operation in *CXL Specification*. It is recommended for the transmitter to drive the sequence number, DLLP CRC, Frame CRC and Frame parity in STP to 0; the receiver must ignore these fields. Given that UCIe Adapter provides reliable Flit transport, framing errors, if detected by the Protocol Layer, are likely due to uncorrectable internal errors and it is permitted to treat them as such.

2.1.4 Standard 256B Start Header Flit Format

This is an optional format for PCIe Flit Mode, supported if Standard Start Header for PCIe protocol Capability is supported. The Protocol Layer must follow the Flit Formats for Flit transfer on FDI, driving 0 on the fields reserved for Die-to-Die Adapter. The PM and Link Management DLLPs are not used over UCIe. The other DLLPs (that are applicable for PCIe Flit Mode) and Flit Status definitions follow the same rules including packing as defined in *PCIe Base Specification*. It is strongly recommended for implementations to optimize out any 8b/10b, 128b/130b and non-Flit Mode related CRC/Retry or framing logic from the Protocol Layer in order to obtain area and power efficient designs for UCIe applications. Portions of the DLP bytes must be driven by the Protocol Layer for Flit_Marker assignment; see Section 3.3.3 for details of the Flit Format.

2.1.5 Latency-Optimized 256B with Optional Bytes Flit Format

This is an optional format for PCIe Flit Mode, supported if Latency-Optimized Flit with Optional Bytes for PCIe protocol capability is supported. It is the Latency-Optimized Flit with Optional Bytes Flit Format for PCIe, as defined in Section 3.3.4. The Protocol Layer must follow the Flit Formats for Flit transfer on FDI, driving 0 on the fields reserved for Die-to-Die Adapter. The PM and Link Management DLLPs are not used over UCIe. The other DLLPs (that are applicable for PCIe Flit Mode) and Flit Status definitions follow the same rules including packing as defined in *PCIe Base Specification*. It is strongly recommended for implementations to optimize out any 8b/10b, 128b/130b and non-Flit Mode related CRC/Retry or framing logic from the Protocol Layer in order to obtain area and power efficient designs for UCIe applications. Portions of the DLP bytes must be driven by the Protocol Layer for Flit_Marker assignment; see Section 3.3.4 for details of the Flit Format.

2.2 CXL 256B Flit Mode

See *CXL Specification* for details on the protocol layer messages and slot formats for "CXL 256B Flit Mode". There are four possible operational formats for this protocol mode (there are two formats in Section 2.2.3), defined in the subsections that follow. The light orange bytes are inserted by the Adapter (see Figure 2-1). In cases where these are shown as part of the main data path (e.g., in the Standard 256B Flit Format), the Protocol Layer must drive 0 on them on the Transmitter, and ignore them on the Receiver.

2.2.1 Raw Format

This format is optional. All bytes are populated by the Protocol Layer. The intended usage is for UCIe Retimers transporting CXL 256B Flit Mode protocol. An example usage of this format is where a CPU and an I/O Device are in different Rack/chassis and connected through a UCIe Retimer using Off-Package Interconnect. Retry, CRC and FEC are taken care of by the Protocol Layer. It is strongly recommended for the UCIe Retimers to check and count errors using either the parity bits of the 6B FEC or the Flit Mode 8B CRC or 6B CRC; depending on which Flit Format was enabled. This helps to characterize and debug the Off-Package Interconnect which is the dominant source of errors. For CXL.cachemem, Viral or poison containment (if applicable) must be handled within the Protocol Layer for this format. See Section 3.3.1 as well.

2.2.2 Latency-Optimized 256B Flit Formats

The support for this format is strongly recommended for "CXL 256B Flit Mode" over UCIe. Two Flit Formats are defined, which provide two independent operating points. These formats are derived from the Latency-Optimized Flits defined in *CXL Specification*. The only difference for the second Flit Format is that it gives higher Flit packing efficiency by providing Protocol Layer with extra bytes. For CXL.io this results in extra 4B of TLP information, and for CXL.cachemem it results in an extra 14B H-slot that can be packed in the Flit. This slot is ordered between Slots 7 and 8. It is included in both Groups B and C, similar to Slot 7. See *CXL Specification* for reference of the packing rules. Support for the first or second format is negotiated at the time of Link bring up. See Section 3.3.4 for the details for the Flit Formats.

The Latency-Optimized formats enable the Protocol Layer to consume the Flit at 128B boundary, reducing the accumulation latency significantly. When this format is negotiated, the Protocol Layer must follow this Flit Format for Flit transfer on FDI, driving 0 on the fields reserved for Die-to-Die Adapter.

The Ack, Nak, PM, and Link Management DLLPs are not used over UCIe for CXL.io for any of the 256B Flit Modes. The other DLLPs and Flit_Marker definitions follow the same rules as defined in *CXL Specification*. Portions of the DLP bytes must be driven by the Protocol Layer for Flit_Marker assignment; see Section 3.3.3 for details on how DLP bytes are driven.

For CXL.cachemem for this mode, FDI provides an **lp_corrupt_crc** signal to help optimize for latency while guaranteeing Viral containment. See Chapter 10.0 for details of interface rules for Viral containment.

2.2.3 Standard 256B Start Header Flit Format

This format is mandatory when "CXL 256B Flit Mode" protocol is supported. It is the Standard 256B Flit Format defined in *CXL Specification* for 256B Flit Mode and the main motivation of supporting this Flit Format is to enable interoperability with vendors that only support the Standard 256B Flit Formats. The Protocol Layer must follow the Flit Formats for Flit transfer on FDI, driving 0 on the fields reserved for Die-to-Die Adapter. The Ack, Nak, PM, and Link Management DLLPs are not used over UCIe for CXL.io. The other DLLPs and Flit Status definitions follow the same rules and packing as defined in *CXL Specification*. Portions of the DLP bytes must be driven by the Protocol Layer for Flit_Marker assignment; see Section 3.3.3 for details of the Flit Formats and on how DLP bytes are driven.

For CXL.cachemem in this format, FDI provides an **lp_corrupt_crc** signal to help optimize for latency while guaranteeing Viral containment. See Section 10.2 for details of interface rules for Viral containment.

See Section 3.3.3 for details about this Flit Format.

2.3 CXL 68B Flit Mode

The *CXL Specification* provides details on the protocol layer messages and slot formats for CXL 68B Flit Mode. There are two operational formats possible for this protocol, and these are defined in the subsections that follow. The light orange bytes are inserted by the Adapter (see Figure 2-1).

2.3.1 Raw Format

This format is optional. All bytes are populated by the Protocol Layer. The intended usage is for UCIe Retimers transporting "CXL 68B Flit Mode" protocol. An example usage of this format is where a CPU and an I/O Device are in different Rack/chassis and connected through a UCIe Retimer using an Off-Package Interconnect. Retry and CRC are taken care of by the Protocol Layer. See Section 3.3.1 as well.

2.3.2 68B Flit Format

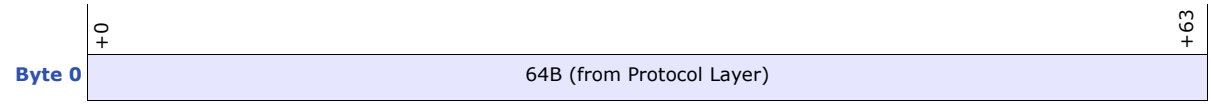
This format is mandatory when CXL 68B Flit Mode protocol is negotiated. This follows the corresponding 68B Flit Format defined in *CXL Specification* and the main motivation of supporting this Flit Format is to enable interoperability with vendors that only support the baseline CXL formats. The Protocol Layer presents 64B of the Flit (excluding the Protocol ID and CRC) on FDI (shown in Figure 2-2), and the Die-to-Die Adapter inserts a 2B Flit Header and 2B CRC and performs the byte shifting required to arrange the Flits in the format shown in Figure 3-11.

The Ack, Nak, and PM DLLPs are not used for CXL.io in this mode. Credit updates and other remaining DLLPs for CXL.io are transmitted in the Flits as defined in *CXL Specification*. For CXL.io, the Transmitter must not implement Retry in the Protocol Layer (because Retry is handled in the Adapter). To keep the framing rules consistent, Protocol Layer for CXL.io must still drive the LCRC bytes with a fixed value of 0, and the Receiver must ignore these bytes and never send any Ack or Nak DLLPs. Framing tokens are applied as defined for CXL.io 68B Flit Mode operation. It is recommended for the transmitter to drive the sequence number, DLLP CRC, Frame CRC and Frame parity in STP to 0. The receiver must ignore these fields. Given that UCIe Adapter provides reliable Flit

transport, framing errors, if detected by the Protocol Layer are likely due to uncorrectable internal errors and it is permitted to treat them as such.

For CXL.cachemem, the "Ak" field defined by *CXL Specification* in the Flit is reserved, and the Retry Flits are not used (because Retry is handled in the Adapter). Link Initialization begins with sending the INIT.Param Flit without waiting for any received Flits. Viral containment (if applicable) must be handled within the Protocol Layer for the 68B Flit Mode. *CXL Specification* introduced Error Isolation as a way to reduce the blast radius of downstream component fatal errors compared to CXL Viral Handling and provide a scalable way to handle device failures across a network of switches shared between multiple Hosts. Specifically, Viral relies on a complete host reset to recover whereas Error Isolation may recover by resetting the virtual hierarchy below the root port. Because CXL-defined Retry Flits (which carry the viral notification for 68B Flits in CXL) are not used in 68B Flit mode in UCIe, it is recommended for implementations to rely on error isolation at the CXL Root Port for fatal errors on CXL.cachemem downstream components in 68B Flit mode (similar to Downstream Port Containment for CXL.io).

Figure 2-2. 68B Flit Format on FDI[a]



a. See Figure 2-1 for color mapping.

2.4 Streaming Protocol

This is the default protocol that must be advertised if none of the PCIe or CXL protocols are going to be advertised and negotiated with the remote Link partner. If Streaming Flit Format capability is not supported, then the operational formats that can be used are either Raw Format or vendor defined extensions. Streaming Flit Format capability is supported if any of 68B Flit Format for Streaming Protocol, Standard 256B End Header Flit Format for Streaming Protocol, Standard 256B Start Header Flit Format for Streaming Protocol, Latency-Optimized 256B Flit Format without Optional Bytes for Streaming Protocol or Latency-Optimized 256B Flit Format with Optional Bytes for Streaming Protocol bits are set in the UCIe Link Capability register.

2.4.1 Raw Format

This is mandatory for Streaming protocol support in Adapter implementations. Protocol Layer interoperability is vendor defined. All bytes are populated by the Protocol Layer. See Section 3.3.1 as well.

2.4.2 68B Flit Format

This format is only applicable if Streaming Flit Format capability is supported. It is an optional format that permits implementations to utilize the 68B Flit Format from the Adapter for Streaming protocols. See Section 3.3.2 for details of the Flit Format.

The Protocol Layer presents 64B per Flit on FDI, and the Die-to-Die Adapter inserts a 2B Flit Header and 2B CRC and performs the byte shifting required to arrange the Flits in the format shown in Figure 3-11. On the receive data path, the Adapter strips out the Flit Header and CRC bytes to only present the 64B per Flit to the Protocol Layer on FDI.

2.4.3 Standard 256B Flit Formats

This format is only applicable if Streaming Flit Format capability is supported. Implementations are permitted to utilize the Standard 256B Start Header Flit Format or Standard 256B End Header Flit

Format from the Adapter for Streaming protocols. See Section 3.3.3 for details of the Flit Format and to see which of the reserved fields in the Flit Header are driven by the Protocol Layer. The Protocol Layer presents 256B per Flit on FDI, driving 0b on the bits reserved for the Adapter. The Adapter fills in the applicable Flit Header and CRC bytes. On the Rx datapath, the Adapter forwards the Flit received from the Link as it is, and the Protocol Layer must ignore the bits reserved for the Adapter (for example the CRC bits).

2.4.4 Latency-Optimized 256B Flit Formats

This format is applicable only when Streaming Flit Format capability is supported. Implementations are permitted to utilize the Latency-Optimized 256B with Optional Bytes Flit Format or Latency-Optimized 256B without Optional Bytes Flit Format for Streaming protocols. See Section 3.3.4 for details of the Flit Format and to see which of the reserved fields in the Flit Header are driven by the Protocol Layer. The Protocol Layer presents 256B per Flit on FDI, driving 0b on the bits reserved for the Adapter. The Adapter fills in the applicable Flit Header and CRC bytes. On the Rx datapath, the Adapter forwards the Flit received from the Link as is, and the Protocol Layer must ignore the bits reserved for the Adapter (e.g., the CRC bits).

2.5 Management Transport Protocol

This protocol is used to carry management network packets over the mainband. The format for these packets is shown in Section 8.2.2.2. The 68B Flit Format is not permitted for this protocol. Raw mode and any of the 256B Flit Formats are permitted for this protocol. When using the 256B Flit Formats, the Protocol Layer presents 256B per Flit on the FDI, driving 0 on the bits that are reserved for the Adapter. The Adapter fills in the applicable Flit Header and CRC bytes. On the Rx data path, the Adapter forwards the Flit received from the Link as is, and the Management Port Gateway must ignore the bits reserved for the Adapter (e.g., the CRC bits).

See Section 8.2.5.2.3 for details of mapping the Management Transport Packets (MTPs) over Management Flits.

§ §