EXIMIUS
Excellence Everyday

IP User Guide

LPIF Adapter IP

Revision: 0.5

14 December 2021

Table of Contents

Table of Figures

Table of Tables

Revision History

Revision	Date	Author	Summary of Changes
0.1	11/4/21	Art Arizpe	Initial draft
0.5	12/14/21	Art Arizpe	Clean up.

Table 1 Revision History

1. Overview

The LPIF Adapter IP provides chiplet designers drop-in IP for PCIe or CXL interconnection of system-in-package solutions utilizing standard SOC, FPGA, and Accelerator silicon.

- Maximizes re-use/interoperability with the chiplet Ecosystem
- Reduces design/validation time and cost of new chiplet designs

LPIF is an industry standard, open-source interface for PCIe and CXL Logical PHYs.

1.1 Description

LPIF to AIB adaptor forwards LPIF signals across AIB. It is optimized for AIB 2.0 and is backwards compatible with AIB 1.x in both symmetrical and asymmetrical use cases (AIB2.0+AIB2.0, AIB1.0+AIB1.0, and AIB2.0+AIB1.0). It supports channel packing in Gen2 to Gen2 systems using only four AIB channels to replace sixteen Flex Bus lanes. AIB 2.0 Gen1 and AIB 1.x does not support channel packing resulting in one-to-one mapping of Flex Bus lanes to AIB channels (x16 requires 16 channels).

Using this IP, maximum throughput for CXL is increased to 512Gbps using AIB from 489.3Gbps using standard Flex Bus Gen5 x16 link. This is accomplished by transmitting the equivalent of a Protocol ID flit header and of sync headers in parallel with each flit over AIB. Maximum usable data throughput is 496.5Gbps for this IP vs. 474.5Gbps for Flex Bus. The difference between maximum throughput and maximum data throughput is due to CXL's inclusion of a two-byte CRC with each 64-bytes of data. The Link Layer is responsible for constructing each 66B CXL flit prior to sending over LPIF. Industry standard LPIF IPs interface to each other using 64B data widths meaning it takes two LPIF clock cycles to transfer an entire flit. Hence the degraded maximum data throughput is a limitation of the Link Layer's LPIF implementation and not of the AIB link or this Adaptor. This IP can be used with LPIF IPs that implement non-standard 66B data widths, or Link Layers that drop the CRC bytes, to realize maximum data throughput rates of 512Gbps, but such use cases are outside the scope of this document. CRC is required for Flex Bus but are not needed for AIB due to differences in the physical layer. Both dice must support CXL without CRC to do so.

Latency on the TX path (Adaptor's LPIF interface to last bit leaves the die) is two LPIF clock cycles for 1GHz LPIF clock rate. Latency on the RX path (first bit enters the die to LPIF data on adaptor's interface) varies based on implementation. As LPIF data is split across multiple AIB Lanes, this IP requires a Channel Alignment IP be used on the receive path to adjust for Lane-to-Lane skew within the AIB HIPs and EMIB interconnect. Careful design of the full die-to-die path focused on minimizing Lane-to-Lane skew will minimize latency on the RX path. Please refer to the AIB Channel Alignment HAS for specific details of incurred latency penalty for skewed channels.

Bifurcation of a 512Gbps link to support multiple endpoints at lower bandwidths is not supported by this IP directly. To bifurcate a link, multiple adapters must be implemented to support each bandwidth option. Supporting a single 512Gbps link or two 256Gbps links requires three instances of this IP to be implemented. The LPIF and AIB sides of each instance must be passed through MUX/DEMUX to accomplish bifurcation. As this is an architectural choice, it's outside the scope of this HAS. However, an implementation guide for bifurcated links may be published later to assist IP users with these architectures.

AIB 2.0 is the intended version of AIB for this IP. AIB 1.0 can also be used but there are some drawbacks to doing so. Most notable of these is the 4x increase in AIB channels required to maintain a given data rate vs. AIB 2.0. Supporting AIB 1.0 requires the IP to be parameterized meaning that a user may generate synthesis and simulation sources for use with either AIB version, but the generated sources cannot be used with another AIB version or gen than it was generated for. The IP also supports generation for multiple clock rates as outlined in the Landing Zone section. Only matched rate clocks may be used within a single die meaning both LPIF and AIB sides of the IP must be clocked at the same rate and phase. Mixed die use cases where one die is using AIB 1.0 and the other is using AIB 2.0 requires that the AIB 2.0 die operates in Gen1 mode for interoperability and will incur the same x4 channel penalty as the AIB 1.0 die. Using this IP in an implementation with AIB 2.0 that supports both Gen1 and Gen2 for use with multiple companion die (i.e. an FPGA for bring up and later an ASIC) requires that this IP is connected to the AIB IP using configurable MUX/DEMUX to select how many AIB channels are used. Similar MUX/DEMUX is required on the interface

between the AIB IP and Channel Alignment IP. This mixed mode application is beyond this document's scope.

1.2 LPIF to AIB Adapter Example

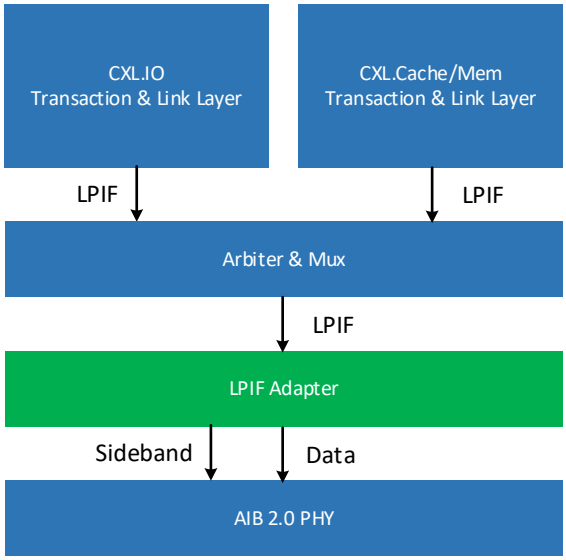

Figure 1 LPIF to AIB Example

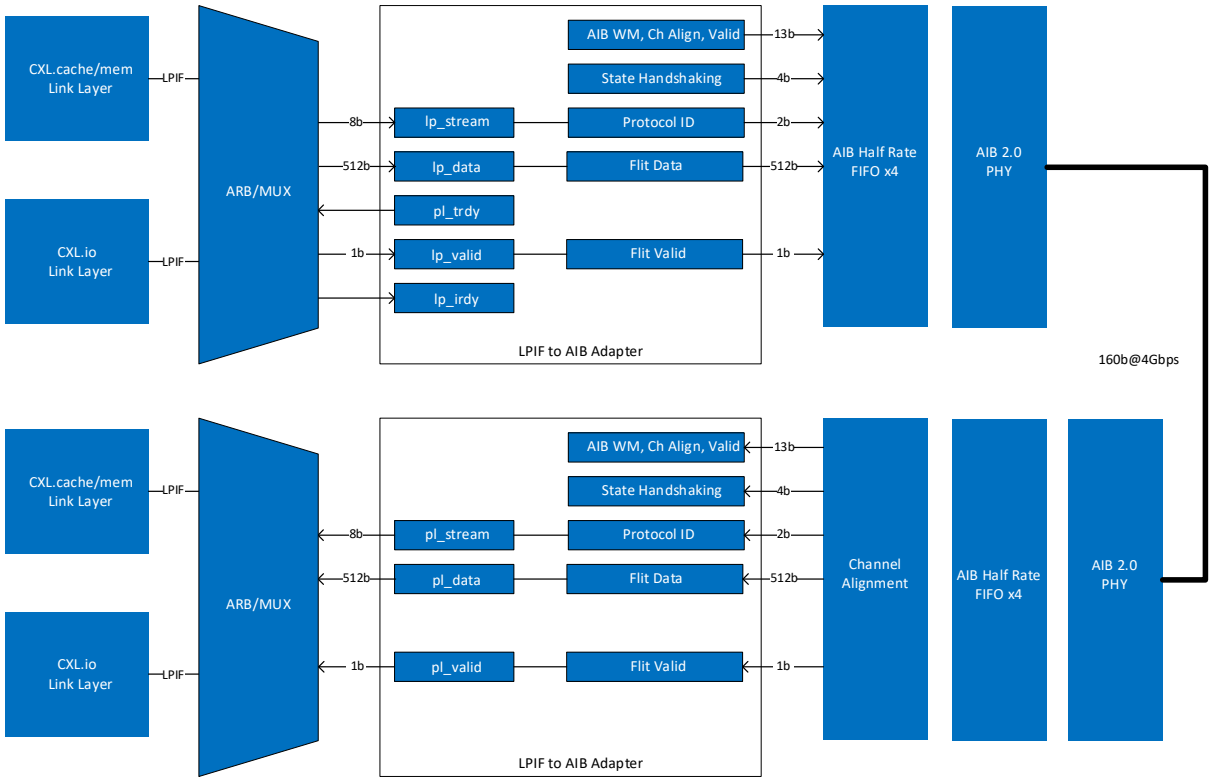

Figure 2 TX + RX Path (Gen2 Half-Rate Quad-Channel)

1.3 Use Cases

Several use cases are provided to illustrate how this IP is intended to be used. It is important to note that in none of these cases is this IP used to pass PCIe protocol over AIB, only CXL is supported. All intended use cases rely on CLX.io protocol to fully replace PCIe protocol functionality.

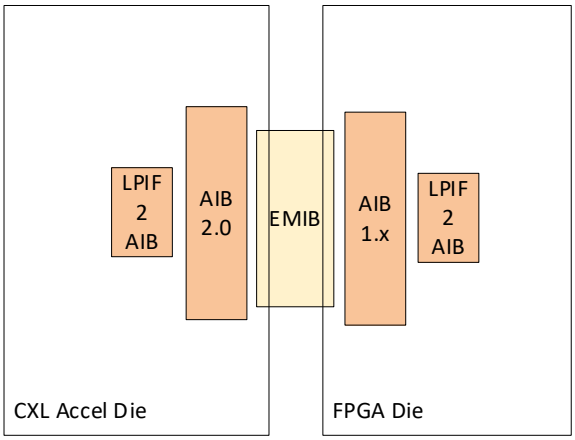

Figure 3 AIB 2.0 to AIB 1.x Use Case

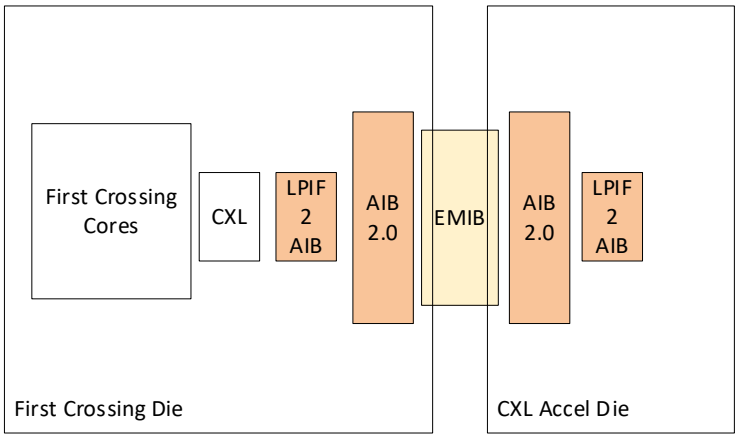

Figure 4 AIB 2.0 to AIB 2.0 Use Case

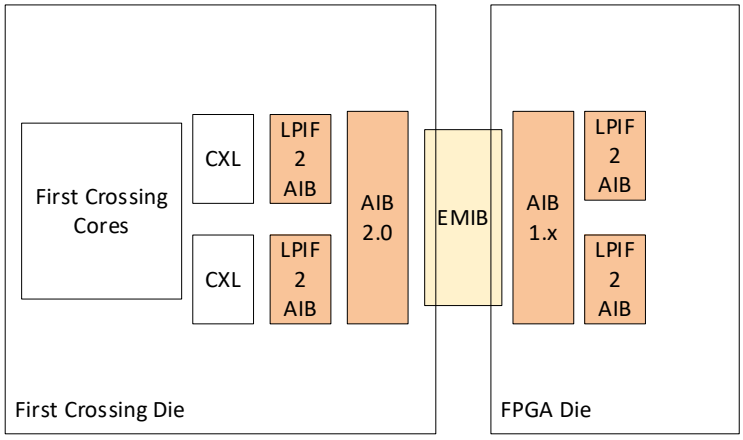

Figure 5 Dual Link Use Case

			Die B						
			AIB2.0/Gen2			AIB2.0/Gen1		AIB1.0	
			4:1	2:1	1:1	2:1	1:1	2:1	1:1
Die A	AIB2.0/Gen2	4:1	Supported	Supported	Supported	Not Possible	Not Possible	Not Possible	Not Possible
		2:1	Supported	POR	POR	Not Possible	Not Possible	Not Possible	Not Possible
		1:1	Supported	POR	Supported	Not Possible	Not Possible	Not Possible	Not Possible
	AIB2.0/Gen1	2:1	Not Possible	Not Possible	Not Possible	Not Advised (possible)	Not Advised (possible)	Supported	Supported
		1:1	Not Possible	Not Possible	Not Possible	Not Advised (possible)	Not Advised (possible)	POR	Supported
	AIB1.0	2:1	Not Possible	Not Possible	Not Possible	Supported	POR	Supported	Supported
		1:1	Not Possible	Not Possible	Not Possible	Supported	Supported	Supported	Supported

Figure 6 Supported Die-to-Die Modes

1.4 Asymmetric Gearboxing

1.4.1 Overview

In order to support the AIB modes and data rates as described above, the IP provides asymmetric gearboxing where the transmit gearbox of one chiplet and the receive gearbox of the remote chiplet are necessarily different. In this scenario, the overall bandwidth of the interface must be preserved, but the individual frequency and widths of each side may vary.

As a simple example, the transmitting side may be using a Gen 2.0 PHY running at full rate, which nominally has a bandwidth of 160 Gbps and requires the m_wr_clk to be running at 2GHz. The receive side may also use a Gen 2.0 PHY, but wants to limit the m_rd_clk to 1GHz for implementation reasons. This would require the receive side to operate in Half Rate mode with a double width data bus, which would provide the same bandwidth as the transmit side.

In Asymmetric Gearboxing, the LPIF data will be concatenated or split depending on if the channel is being upsized or downsized, respectively. Table 2 Assymetric Gearboxing shows what rates correspond to upsizing or downsizing. Note that Full to Full, Half to Half and Quarter to Quarter are symmetric, and are not included in this table and are generally handled fully by the standard AIB Adapter and PHY.

Transmit Rate	Receive Rate	Acronym	Terminology	Notes
Full	Half	F2H	Upsizing	Two beats of TX are concatenated into one beat of RX
Full	Quarter	F2Q	Upsizing	Four beats of TX are concatenated into one beat of RX
Half	Full	H2F	Downsizing	One beat of TX are split into two beats of RX
Half	Quarter	H2Q	Upsizing	Two beats of TX are concatenated into one beat of RX
Quarter	Full	Q2F	Downsizing	One beat of TX are split into four beats of RX
Quarter	Half	Q2H	Downsizing	One beat of TX are split into two beats of RX

Table 2 Assymetric Gearboxing

1.4.2 Implementation

To support Asymmetric Gearboxing, the data path must be quantized in a way that either side of the asymmetric interface can understand.

To provide the maximum flexibility, the IP uses a Replicated Struct that fits inside the Full Rate mode. By designing this way, any combination of Full, Half or Quarter on the transmit (TX) can interact with any combination of Full, Half or Quarter on the receive (RX).

There are potential simplifications that could be made at the cost of reduced interoperability, for example by eliminating support for Full TX and Full RX, the IP could use a larger Replicated Struct and not require duplicate locations for Strobe and Markers.

The Replicated Struct is the minimum quanta of data to be sent or received. This should be replicated two times for Half Rate and four times for Quarter Rate. **Figure 8 Assymetric Gearboxing Full to Quarter** shows the replicated struct for the Full Rate TX to Quarter Rate RX.

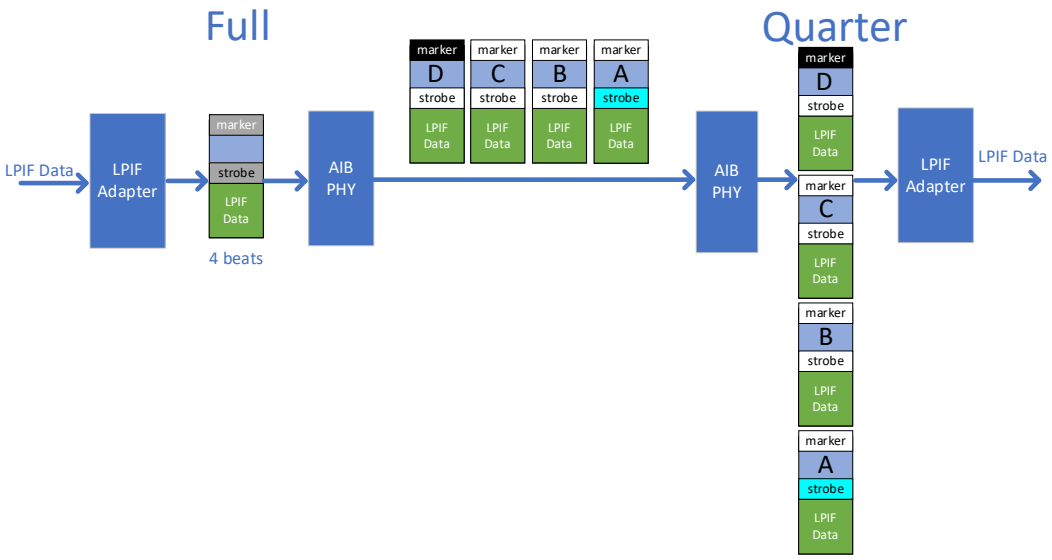

Figure 7 Assymetric Gearboxing Full to Quarter

1.4.3 LPIF Data to AIB Data replicated struct example

This example shows the data mapping for and AIB Gen2, TX, operating at Full Rate, using 4 channels, each with 80 data bits. The LPIF data width is 256 bits.

Channel 0 contains LPIF protocol control information. All channels contain marker, strobe, and Flit Data and Flit Data Valid bits.

AIB Channel	Data	Number of bits
0	AIB Marker	1
	AIB Channel Alignment Strobe	1
	Valid	1
	LPIF State	4

	LPIF Protocol ID	2
	Flit Start Byte	5
	Flit Data	66
	Total data	**80**
1	AIB Marker	1
	AIB Channel Alignment Strobe	1
	Flit Data	78
	Total data	**80**
2	AIB Marker	1
	AIB Channel Alignment Strobe	1
	Flit Data	78
	Total data	**80**
3	AIB Marker	1
	AIB Channel Alignment Strobe	1
	Flit Data	34
	Flit Data Valid	32
	Rsvd	12
	Total data	**80**

Table 3 Replicated Struct Example

2. Features

- Complies with CXL 2.0, AIB 2.0, and LPIF 1.0 specifications
- Provides a single link layer port
- Supports x16, x8, x4 configurations for AIB 2.0
- Supports x2, x1 configurations for AIB 1.0
- Link Layer interface runs @1GHz in ASIC, 500 MHz in FPGA
- Virtual Link State Machine
- Support for optional configuration port for Link Layer
- Support for AXI4-lite configuration port
- Configurable mapping of Stream ID bits to Protocol ID encoding
- AIB Channel count: 1 to 16
- Lanes per AIB Channel: 1x – 4x Lanes / Channel
- AIB Interface Modes, Full, Half, and Quarter Rate
- AIB 2.0/Gen2, AIB2.0/Gen1, AIB1.0

2.1 Clock Rate vs Data Width (CXL 32GT x16)

LPIF Clock	LPIF Data Width	AIB FIFO Clock	AIB FIFO Data Width (Gen 2, 4 Lanes per Channel)
0.5GHz	8 B/Lane (128B)	0.5GHz	256 bits (out of 320) x4 Ch
1.0GHz	4 B/Lane (64B)	1.0GHz	128 bits (out of 160) x4 Ch
2.0GHz*	2 B/Lane (32B)	2.0GHz	64 bits (out of 80) x4 Ch

** 2.0 GHz Clock is only supported in a leading edge process node*

Table 4 Gen 2, 4 Lanes per channel

LPIF Clock	LPIF Data Width	AIB FIFO Clock	AIB FIFO Data Width (Gen 1, 1 Lane per Channel)
0.5GHz	8 B/Lane (128B)	0.5GHz	64 bits (out of 80) x16 Ch
1.0GHz	4 B/Lane (64B)	1.0GHz	32 bits (out of 40) x16 Ch

Table 5 Gen 1, 1 Lane per channel

2.2 CXL Specification Variances for AIB

- Idle cycles on Link to PHY data path
  - Instead of inserting NULL Flits as a Flex Bus Logical PHY does, the Adapter will forward LPIF Data Byte Enable bus across AIB
- Flit Headers
  - Instead of inserting a 16b Protocol ID Header to each Flit as a Flex Bus Logical PHY does, the Adapter uses a 2b bus indicating Flit destination (CXL.io, CXL.mem/cache, ARB/MUX, or Logical PHY)
  - Logical PHY Flit destination is only used for NULL Flits and will not be supported
  - EDS Token or Implied EDS Token Protocol ID types per destination are consolidated

3. Block Diagram

The module contains the following sub-blocks.

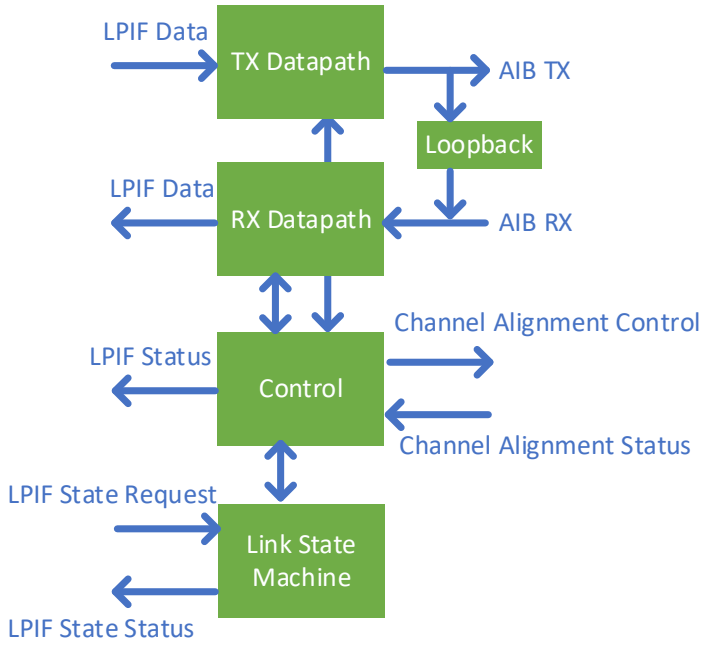

Figure 8 LPIF Adapter

3.1 TX Datapath

This takes LPIF data from the Link Layer, creates the appropriate replicated struct for each channel, and interfaces to the AIB PHY. This also provides the gearbox markers, channel alignment strobes, and state handshaking information.

3.2 RX Datapath

This receives data from the Channel Alignment IP, extracts fields from the replicated struct, and forwards the LPIF data to the Link Layer.

3.3 Control

This uses the current link state and controls both the TX and RX datapaths. It also controls and receives status from the Channel Alignment IP. This also contains the CSRs.

3.4 Link State Machine

- Replaces LTSSM
- Prior to exiting RESET, AIB Channel must be locked at both die
- State Transition handshaking across die (except LinkError and Retrain)
- Retrain state is only for Link/PHY handshaking, no retraining is performed
- Active state does not support L0s
- L1 state does not support substates

3.4.1 Sideband path

- Allows upstream and downstream LPIF Adapters to handshake state transitions without decoding in-band packets
- There are nine mutually exclusive states that must be communicated between Adapters
- Four bits of AIB channel 0 are used to pass state information in each direction
- Including all state encoding bits with each FIFO clock minimizes transition latency

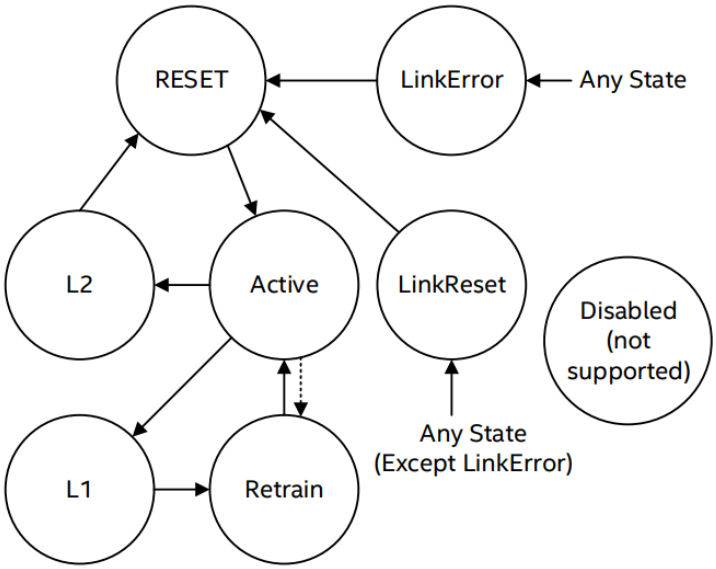

Figure 9 Virtual LSM

3.4.2 Link States

The LPIF State Status Machine is specified in the LPIF Specification, Section 5.2. The specific entry and exit conditions are documented in the LPIF Specification.

The following is a description of the states as they apply to an AIB PHY.

3.4.3 Reset State

The Reset State can be entered on de-assertion of the LPIF interface reset signal or from states as described in the LPIF Specification. In this state the physical layer is permitted to begin its initialization process. For the LPIF Adapter, the AIB PHY initialization complete is indicated by signals ms_tx_transfer_en, ms_rx_transfer_en, sl_tx_transfer_en, and sl_rx_transfer_en. In addition, the Channel Alignment IP indicates align_done. See 6.1 Initialization bring-up sequence for a description.

3.4.4 Active State

The LPIF Adapter does not support the L0s state.

3.4.5 L1 State

The LPIF Adapter supports the L1 power management state, however no power or clock gating is implemented. L1 does not support substates.

3.4.6 L2 State

The LPIF Adapter supports the L2 power management state, however no power or clock gating is implemented.

3.4.7 Retrain State

The LPIF Adapter enters the retrain state when directed by the Link Layer, due to internal retrain requests, or as requested by the remote physical layer. There is no AIB PHY retraining such as recalibration performed.

4. Clocks and Resets

Please refer to **Table 5 Parameters** for a definition of the parameters.

Signal Name	Width	Direction	Type	Description
lclk	1	in	LPIF Clock	Same as LPIF Specification
reset	1	in	reset	Reset signal for this IP. Reset is handled by system level reset controller. AIB and Channel Alignment reset sequencing is TBD

Table 6 Clocks and Resets

The LPIF to AIB Adaptor has three clock domains. LPIF TX and RX operates in the lclk domain. AIB TX interface operates in the AIB TX clock domain. AIB Channel Alignment interface operates in the com_clk clock domain. These clocks must be matched in frequency and phase.

4.1 Clock Gating

When in L1 and L2 power states the LPIF to AIB Adaptor does not employ clock gating.

4.2 PLLs

Clocks must be generated external of this IP. Use of PLL or FPGA equivalent is required. Clock rates are based on highest frequency that will close timing in customer design.

5. Interfaces

Please refer to **Table 5 Parameters** for a definition of the parameters.

5.1 LPIF Interface

The LPIF Interface side of the IP connects to standard LPIF Link Layer IP or an Arbiter/MUX block in the case of multiple Link Layers. Only signals relevant to CXL are implemented on this interface. Legacy PCIe Data Link Layers cannot interface with this IP as implemented.

Signal Name	Width	Direction	Type	Description
pl_trdy	1	out	LPIF Data	Same as LPIF Die to Die Whitepaper

Signal Name	Width	Direction	Type	Description
pl_data	Variable	out	LPIF Data	Same as LPIF Specification
pl_valid	Variable	out	LPIF Data	Same as LPIF Specification
pl_stream	8	out	LPIF Data	Same as LPIF Specification
pl_error	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_trainerror	1	out	LPIF Status	Same as LPIF Specification
pl_cerror	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_stallreq	1	out	LPIF Status	Same as LPIF Specification
pl_tmstmp	1	out	LPIF Status	Same as LPIF Specification
pl_tmstmp_stream	8	out	LPIF Status	Same as LPIF Specification
pl_phyinl1	1	out	LPIF Status	Same as LPIF Specification
pl_phyinl2	1	out	LPIF Status	Same as LPIF Specification
lp_irdy	1	in	LPIF Data	Same as LPIF Specification
lp_data	Variable	in	LPIF Data	Same as LPIF Specification
lp_valid	Variable	in	LPIF Data	Same as LPIF Specification
lp_stream	8	in	LPIF Data	Same as LPIF Specification
lp_stallack	1	in	LPIF Status	Same as LPIF Specification
lp_state_req	4	in	LPIF Status	Same as LPIF Specification
pl_state_sts	4	out	LPIF Status	Same as LPIF Specification
lp_tmstmp	1	in	LPIF Status	Same as LPIF Specification
lp_linkerror	1	in	LPIF Status	Same as LPIF Specification
pl_quiesce	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
lp_flushed_all	1	in	LPIF Status	Same as LPIF Die to Die Whitepaper
lp_rcvd_crc_err	1	in	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_lnk_cfg	3	out	LPIF Status	Same as LPIF Specification
pl_lnk_up	1	out	LPIF Status	Same as LPIF Specification
pl_rxframe_errmask	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_portmode	1	out	LPIF Status	Same as LPIF Specification
pl_portmode_val	1	out	LPIF Status	Same as LPIF Specification
pl_speedmode	3	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_clr_lnkeqreq	3	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_set_lnkeqreq	3	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_inband_pres	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_ptm_rx_delay	8	out	LPIF Status	Same as LPIF Die to Die Whitepaper

Signal Name	Width	Direction	Type	Description
pl_setlabs	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_setbms	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_surprise_lnk_down	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_protocol	3	out	LPIF Status	Tied to CXL.io + CXL.mem/cache (100b)
pl_protocol_vld	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
pl_err_pipestg	1	out	LPIF Status	Same as LPIF Die to Die Whitepaper
lp_wake_req	1	in	LPIF Status	Same as LPIF Specification
pl_wake_ack	1	out	LPIF Status	Same as LPIF Specification
lp_force_detect	1	in	LPIF Status	Same as LPIF Specification
pl_phyinrecenter	1	out	LPIF Status	Same as LPIF Specification
pl_exit_cg_req	1	out	LPIF CLK Gating	Same as LPIF Specification
lp_exit_cg_ack	1	in	LPIF CLK Gating	Same as LPIF Specification
pl_cfg	Variable	out	LPIF Config	Same as LPIF Specification
pl_cfg_vld	1	out	LPIF Config	Same as LPIF Specification
lp_cfg	Variable	in	LPIF Config	Same as LPIF Specification
lp_cfg_vld	1	in	LPIF Config	Same as LPIF Specification
lp_crc	Variable	in	LPIF Data	Same as LPIF Specification
lp_crc_valid	Variable	in	LPIF Data	Same as LPIF Specification
pl_crc	Variable	out	LPIF Data	Same as LPIF Specification
pl_crc_valid	Variable	out	LPIF Data	Same as LPIF Specification
lp_device_present	1	in	LPIF Status	Same as LPIF Specification
pl_clk_req	1	out	LPIF Status	Same as LPIF Specification
lp_clk_ack	1	in	LPIF Status	Same as LPIF Specification
lp_pri	2	in	LPIF Status	Same as LPIF Specification

Table 7 LPIF Interface

5.2 AIB Interface

The AIB Interface side of the IP connects to both AIB IP and Channel Alignment IP blocks. TX data is forwarded over AIB through the FIFO data path. RX data is received using Channel Alignment.

This IP controls the AIB and Channel Alignment blocks' relevant clock, reset, and configuration signals. The status signals are sampled for indicating when it is ready to begin state changes within the IP.

Signal Name	Width	Direction	Type	Description
data_in_f	Variable	out	AIB TX	AIB TX FIFO Data, Connection spans all channels
ns_mac_rdy	1	out	AIB Status	Only connected at channel 0
fs_mac_rdy	1	in	AIB Status	Only connected at channel 0
ns_adapter_rstn	Variable	out	AIB Status	One bit per channel
sl_rx_transfer_en	Variable	in	AIB Status	One bit per channel
ms_tx_transfer_en	Variable	in	AIB Status	One bit per channel
ms_rx_transfer_en	Variable	in	AIB Status	One bit per channel
sl_tx_transfer_en	Variable	in	AIB Status	One bit per channel
m_rxfifo_align_done	Variable	in	AIB Status	One bit per channel
wa_error	Variable	in	AIB Status	Word Alignment Error, One bit per channel
wa_error_cnt	Variable	in	AIB Status	Word Alignment Error Count, Four bits per channel
dual_mode_select	1	TBD	AIB Config	TBD
m_gen2_mode	1	in	AIB Config	TBD
i_conf_done	1	in	AIB Config	TBD
por	Variable	in	AIB Config	One bit per channel
dout	Variable	in	CA Data	AIB RX Data Aligned across channels
align_done	1	in	CA Status	RX Path is ready
align_err	1	in	CA Status	RX data is corrupt
fifo_full	1	in	CA Status	TBD
fifo_pfull	1	in	CA Status	TBD
fifo_empty	1	in	CA Status	TBD
fifo_pempty	1	in	CA Status	TBD
align_fly	1	out	CA Config	Always High
stb_wd_sel	8	out	CA Config	TBD (Word location of Channel Alignment bit)
stb_bit_sel	40	out	CA Config	TBD (Bit location of Channel Alignment bit)
stb_intv	8	out	CA Config	TBD
fifo_full_val	6	out	CA Config	TBD
fifo_pfull_val	6	out	CA Config	TBD
fifo_empty_val	3	out	CA Config	TBD
fifo_pempty_cal	3	out	CA Config	TBD
rden_dly	3	out	CA Config	TBD

Signal Name	Width	Direction	Type	Description
tx_online	1	out	CA Config	Controls CA operation
rx_online	1	out	CA Config	Controls CA operation
delay_x_value	16	in	CA Config	Controls online timing
delay_y_value	16	in	CA Config	Controls online timing
delay_z_value	16	in	CA Config	Controls online timing
lpbk_en	1	in	Misc	Controls loopback mode

Table 8 AIB Interface

Control signals are pseudo-static and must be stable before reset deassertion. They cannot change during normal operation. They can change when reset is asserted.

6. Verification

The module has a stand-alone testbench. The testbench contains the LPIF Adapters, Channel Alignment modules, AIB PHYs, and the CXL VIP modules.

6.1 Initialization bring-up sequence

The initialization and bring-up sequence involve generating the TX_ONLINE and RX_ONLINE signals from system level signals allowing for an automated Online Synchronization SM to perform the sequencing of the various Online bits in the system.

The underlying scheme is to take some key signals from the PHY (namely the *_rx_transfer_en) and use them to start several timers in the system including in the LPIF Adapter. These timers are configured to ensure remote side events have occurred, namely Word Marker Recovery by the PHY and Channel Alignment.

The counters are configured using fields called X, Y and Z that are:

- X = Delay after *_rx_transfer_en until Word Marking is guaranteed to be synchronized.
- Y = Delay after sending Strobe that Channel Alignment is guaranteed.
- Z = Modest delay to ensure receive is ready before transmit occurs.

Each counter is an 16 bit counter. Systems can be configured with values ranging from 0 to 255, with 0 meaning to skip the counter.

Note that the system generally triggers off of *_rx_transfer_en, which is obviously referring to the receive side. If the remote receive side is a leader, we should use ms_rx_transfer_en and if it is a follower, we should use sl_rx_transfer_en. This can be hardcoded or dynamically adjusted via dual_mode_select. Note that dual_mode_select is local to the chiplet, so the transmit side signal dual_mode_select describes the inverse of the remote receive side (i.e. TX Leader means the RX is necessarily Follower)

6.1.1 TX Initialization

The TX initialization needs to detect when all the AIB PHYs are initialized.

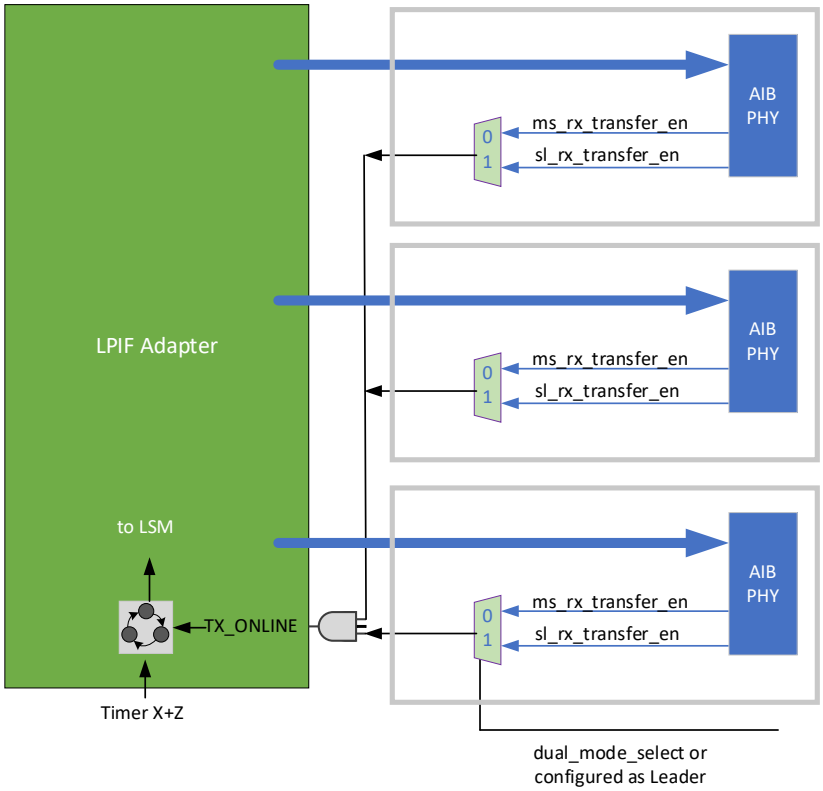

Figure 10 TX Initialization

6.1.2 RX Initialization

The RX initialization needs to detect when all the AIB PHYs are initialized and when the Channel Alignment IP is aligned.

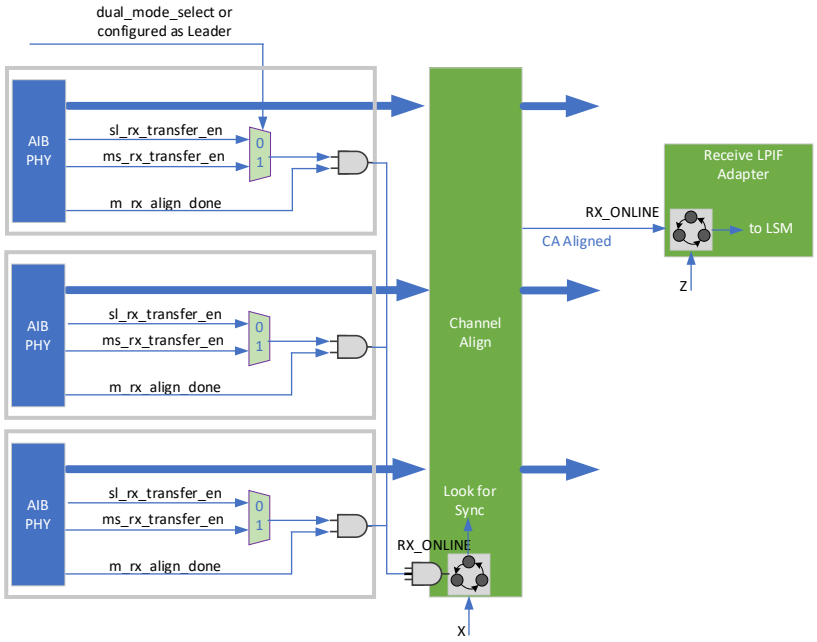

Figure 11 RX Initialization

6.2 Tested Configurations

Item	Target
Number of AIB channels	1x – 16x
Flex Bus Lanes per AIB Channel	1x-4x Lanes / Channel
AIB Interface Modes	Full, Half, and Quarter - AIB 2.0/Gen2, AIB2.0/Gen1 Full and Half - AIB 1.0
LPIF Data Widths in bytes	32, 64, 128

Table 9 Tested Configurations

7. Implementation

The ASIC Fmax is 1 GHz in a 16nm or better process, with a 15% clock period setup margin.

The FPGA Fmax is 500 MHz in an Agilex 10nm device, speed grade -2, and a 15% clock period setup margin.

The validation device is a Stratix 10 with a target FPGA Fmax of 500 MHz, speed grade -2.

The goal is to have a common RTL code base for the ASIC and the FPGA. Any differences should be minimized or handled in a parameterizable manner. For example, if the ASIC and the FPGA need different number of pipeline stages to close timing, this should be done with parameters.

All of the module data inputs and outputs will be flopped, possibly in a several stage pipeline.

8. Configuration

8.1 Compile Time Parameters

Parameter	Supported Values	Details
AIB_VERSION	1, 2	Version of AIB IP connected to LPIF to AIB Adaptor
AIB_GENERATION	1, 2	Generation of AIB used in system
AIB_LANES	1,2,4,8,16	Number of AIB Lanes to use for LPIF connection. Values 1,2,8,16 are only valid when AIB_GENERATION is set to 1
AIB_CLOCK_RATE	500,1000,2000	Clock rate in MHz of AIB FIFO interface. Clock rate and AIB_LANES, AIB_GENERATION, and AIB_VERSION are used together for calculating FIFO width
LPIF_CLOCK_RATE	500,1000,2000	Clock rate in MHz of LPIF interface.
LPIF_DATA_WIDTH	32,64,128	Width in bytes of LPIF data buses. A width of 128 is for FPGAs.
LPIF_PIPELINE_STAGES	0,1,2,3,4,5,6	Number of pipelining stages in LPIF path. See Section 9 of the LPIF Specification for impact of pipelining. This will reflect the inserting of pipeline stages between the link layer and LPIF Adapter because of floorplan and timing requirements.
MEM_CACHE_STREAM_ID	001b,010b,100b	Stream ID value associated with CXL.mem/cache Link Layer. Value must be unique among all *_STREAM_ID parameters.
IO_STREAM_ID	001b,010b,100b	Stream ID value associated with CXL.mem/cache Link Layer. Value must be unique among all *_STREAM_ID parameters.

ARB_MUX_STREAM_ID	001b,010b,100b	Stream ID value associated with CXL.mem/cache Link Layer. Value must be unique among all *_STREAM_ID parameters.

Table 10 Parameters

8.2 Supported configurations

The following table shows the compile time parameter values for the supported configurations.

CXL Lanes	LPIF_CLOCK_RATE (GHz)	LPIF_DATA_WIDTH (bytes)	AIB_CLOCK_RATE (GHz)	AIB_VERSION	AIB_GENERATION	AIB_LANES	AIB_BITS_PER_LANE
x16	0.5	128	0.5	2	2	4	320
x16	1.0	64	1.0	2	2	4	160
x16	2.0	32	2.0	2	2	4	80
x8	0.5	128	0.5	2	2	2	320
x8	1.0	64	1.0	2	2	2	160
x8	2.0	32	2.0	2	2	2	80
x4	0.5	128	0.5	2	2	1	320
x4	1.0	64	1.0	2	2	1	160
x4	2.0	32	2.0	2	2	1	80
x16	0.5	128	0.5	2	1	16	80
x16	1.0	64	1.0	2	1	16	40
x8	0.5	128	0.5	2	1	8	80
x8	1.0	64	1.0	2	1	8	40
x4	0.5	128	0.5	2	1	4	80
x4	1.0	64	1.0	2	1	4	40
x2	0.5	128	0.5	2	1	2	80
x2	1.0	64	1.0	2	1	2	40
x1	0.5	128	0.5	2	1	1	80
x1	1.0	64	1.0	2	1	1	40

Table 11 Support configurations

9. References

AIB specification: https://github.com/intel/aib-phy-hardware/blob/master/docs/AIB_Intel_Specification%201_2%20.pdf

CXL Specification version 1.1

Logical PHY Interface Specification, version 1.0

LPIF AIB HAS 0.5

Channel Alignment FS 0.5