CSI-2/DSI D-PHY Rx IP

User Guide

FPGA-IPUG-02081-2.1

January 2024

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
AXI	Advance eXtensible Interface
CIL	Control and Interface Logic
CRC	Cyclic Redundancy Check
CSI-2	Camera Serial Interface-2
DSI	Digital Serial Interface
ECC	Error Correction Code
EoTP	End of Transmission Packet
FIFO	First In First Out
FPGA	Field-Programmable Gate Array
HS	High-Speed
LMMI	Lattice Memory Mapped Interface
LP	Low-Power
LUT	Look-up Table
MIPI	Mobile Industry Processor Interface
PLL	Phase-Locked Loop
SoT	Start of Transmit
SoTp	Start-of-Transmission Pattern

1. Introduction

1.1. Overview of the IP

The Lattice Semiconductor CSI-2/DSI D-PHY Receiver IP Core converts DSI or CSI-2 data to 8-bit, 16-bit, 32-bit, or 64-bit data for Lattice FPGA devices built on the Lattice Nexus™ platform as indicated in the dark gray boxes in Figure 1.1.

The CSI-2/DSI D-PHY Receiver IP Core is intended for use in applications that require a D-PHY receiver in the FPGA logic. D-PHY Rx IP includes in both the high-speed and low-power modules. The payload data (image data) uses the high-speed mode whereas the control and status information are sent through low-power mode. The number of D-PHY data lanes for data transmission is configurable and supported 1, 2, 3, or 4 data lanes.

The IP design is implemented in Verilog HDL language.

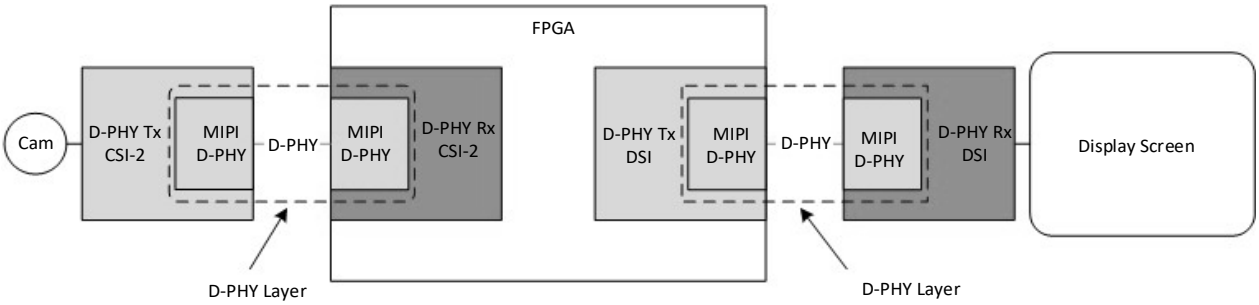

Figure 1.1. D-PHY Rx IP

1.2. Quick Facts

Table 1.1 presents a summary of the CSI-2/DSI D-PHY Rx IP Core.

Table 1.1. CSI-2/DSI D-PHY Rx IP Core Quick Facts

IP Requirements	Supported FPGA Family	Lattice Avant™, MachXO5™-NX, CrossLink™-NX, CertusPro™-NX, Certus™-NX
Resource Utilization	Targeted Devices	LAV-AT-E70, LFMXO5-25, LFMXO5-55T, LFMXO5-100T, LIFCL-40, LIFCL-33, LIFCL-17, LFCPNX-100, LFCPNX-50, LFD2NX-40, LFD2NX-17
	Supported User Interfaces	LMMI/LINTR/AXI4-Stream Interface
	Resources	See Table A.1 and Table A.2.
Design Tool Support	Lattice Implementation	IP Core v1.0.x – Lattice Radiant™ software 2.0 IP Core v1.1.x – Lattice Radiant software 2.1 IP Core v1.2.x – Lattice Radiant software 2.2 or later IP Core v1.5.x – Lattice Radiant software 2022.1 IP Core v1.6.x for Nexus – Lattice Radiant software 2023.1 IP Core v1.6.x for Avant – Lattice Radiant software 2023.2
	Synthesis	Lattice Synthesis Engine (LSE)
		Synopsys® Synplify Pro® for Lattice
	Simulation	For a list of supported simulators, see the Lattice Radiant software user guide.

1.3. Features

- Compliant with MIPI D-PHY v1.2, MIPI DSI v1.1, and MIPI CSI-2 v1.2 specifications.
- Selection between Hard Rx D-PHY or Soft Rx D-PHY implementation. Hard Rx D-PHY is only available for CrossLink-NX devices.
- Supports MIPI DSI and MIPI CSI-2 interfaces up to 6 Gb/s for Soft D-PHY and up to 10 Gb/s for Hard D-PHY.
- Supports 1, 2, 3, or 4 data lanes and one clock lane.
- Supports continuous and non-continuous MIPI D-PHY clock.
- Supports all MIPI DSI Video Mode of operations.
  - Non-Burst Mode with Sync Pulses
  - Non-Burst Mode with Sync Events
  - Burst Mode
- Optional packet parsing or parallel data translation only.
- Supports optional periodic deskew detection.
- Supports all MIPI DSI compatible video formats.
- Supports all MIPI CSI-2 compatible video formats.

1.3.1. Hard CSI-2/DSI D-PHY Rx IP Core Features

- Maximum rate is up to 2500 Mbps per lane
- Supports 8x or 16x gearing
- Option to use internal or external clock source
- Option to use hardened Control and Interface Logic (CIL) or Fabric Logic
- Hard D-PHY is supported only on CrossLink-NX devices

1.3.2. Soft CSI-2/DSI D-PHY Rx IP Core Features

- Maximum rate of up to 1500 Mbps per lane for Lattice Avant, CrossLink-NX, Certus-NX, and CertusPro-NX devices
- Supports 8x gearing only
- External clock source

1.4. Licensing and Ordering Information

An IP specific license string is required to enable full use of the CSI-2/DSI D-PHY Receiver IP in a complete, top-level design.

The IP can be fully evaluated through functional simulation and implementation (synthesis, map, place and route) without an IP license string. This IP supports Lattice's IP hardware evaluation capabilities. You can create versions of the IP to operate in hardware for a limited time (approximately four hours) without requiring an IP license string. A license string is required to enable timing simulation and to generate a bitstream file that does not include the hardware evaluation timeout limitation.

For more information about pricing and availability of the Lattice Semiconductor CSI-2/DSI D-PHY Receiver IP, contact your local Lattice Sales Office.

1.4.1. Ordering Part Number

Table 1.2. Ordering Part Number

Device Family	Part Number	
	Single Machine Annual	Multi-Site Perpetual
CrossLink-NX	D-PHY-RX-CNX-US	D-PHY-RX-CNX-UT
Certus-NX	D-PHY-RX-CTNX-US	D-PHY-RX-CTNX-UT
CertusPro-NX	D-PHY-RX-CPNX-US	D-PHY-RX-CPNX-UT
Lattice Avant-E	DPHY-RX-AVE-US	DPHY-RX-AVE-UT
Bundled	MIPI-BNDL-US	MIPI-BNDL-UT

1.5. IP Validation Summary

Table 1.3 shows the validation status for the CSI-2/DSI D-PHY Receiver IP core. The ✓ mark indicates whether the IP has been validated for Simulation, Timing, or with Hardware.

Table 1.3. IP Validation Level

Device Family	IP Version	Validation Level		
		Simulation	Timing	Hardware
Lattice Nexus ™	1.6.0	✓	✓	—
Lattice Avant™	1.6.0	✓	✓	—

1.6. Minimum Device Requirements

Refer to Table A.2 for the minimum required resource to instantiate this IP.

1.7. Naming Conventions

1.7.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL.

1.7.2. Signal Names

- _*n* are active low (asserted when value is logic 0)
- _*i* are input signals
- _*o* are output signals

2. Functional Description

2.1. IP Architecture Overview

Figure 2.1 and Figure 2.2 show the architecture blocks and data flow in the CSI-2/DSI D-PHY Receiver IP.

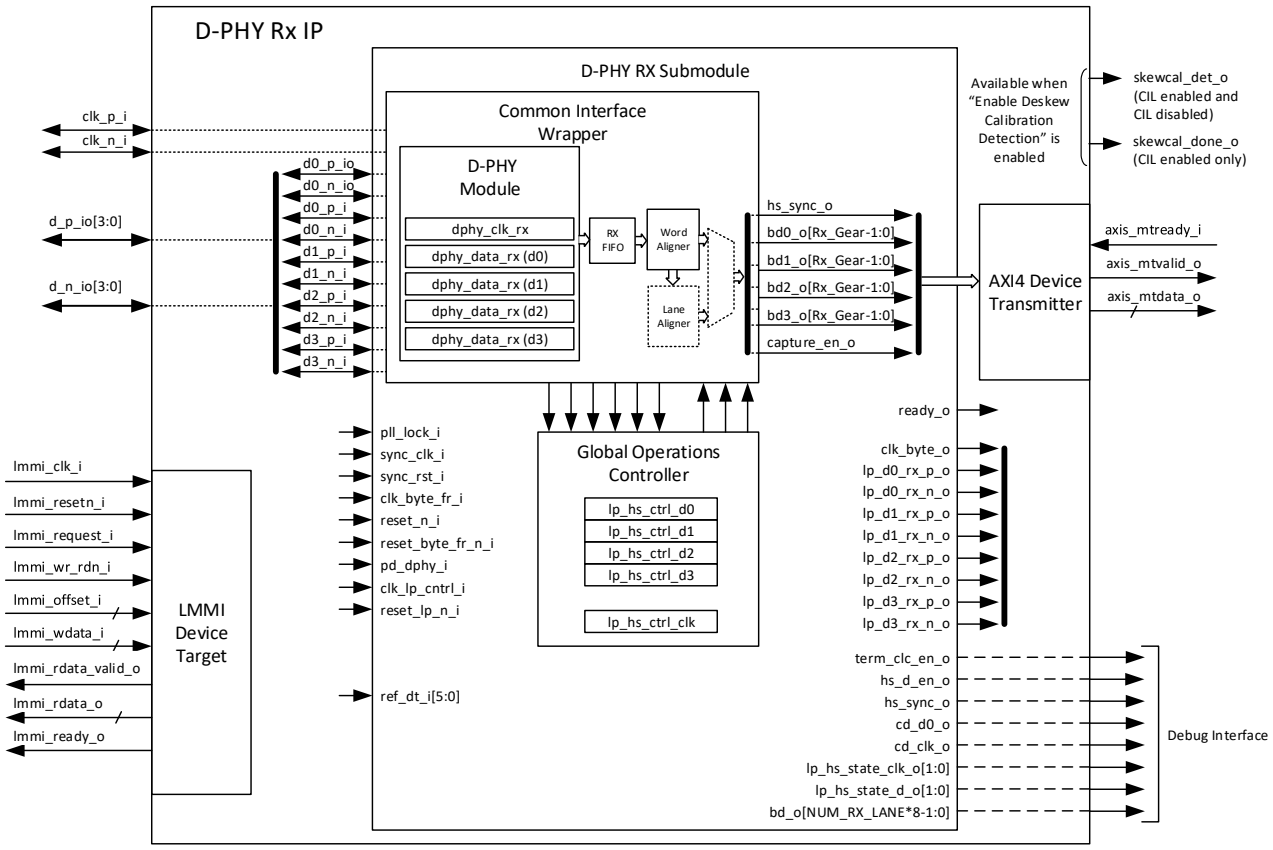

Figure 2.1. D-PHY Rx IP Block Diagram with AXI4-Stream Enabled, LMMI Enabled, and Packet Parser OFF

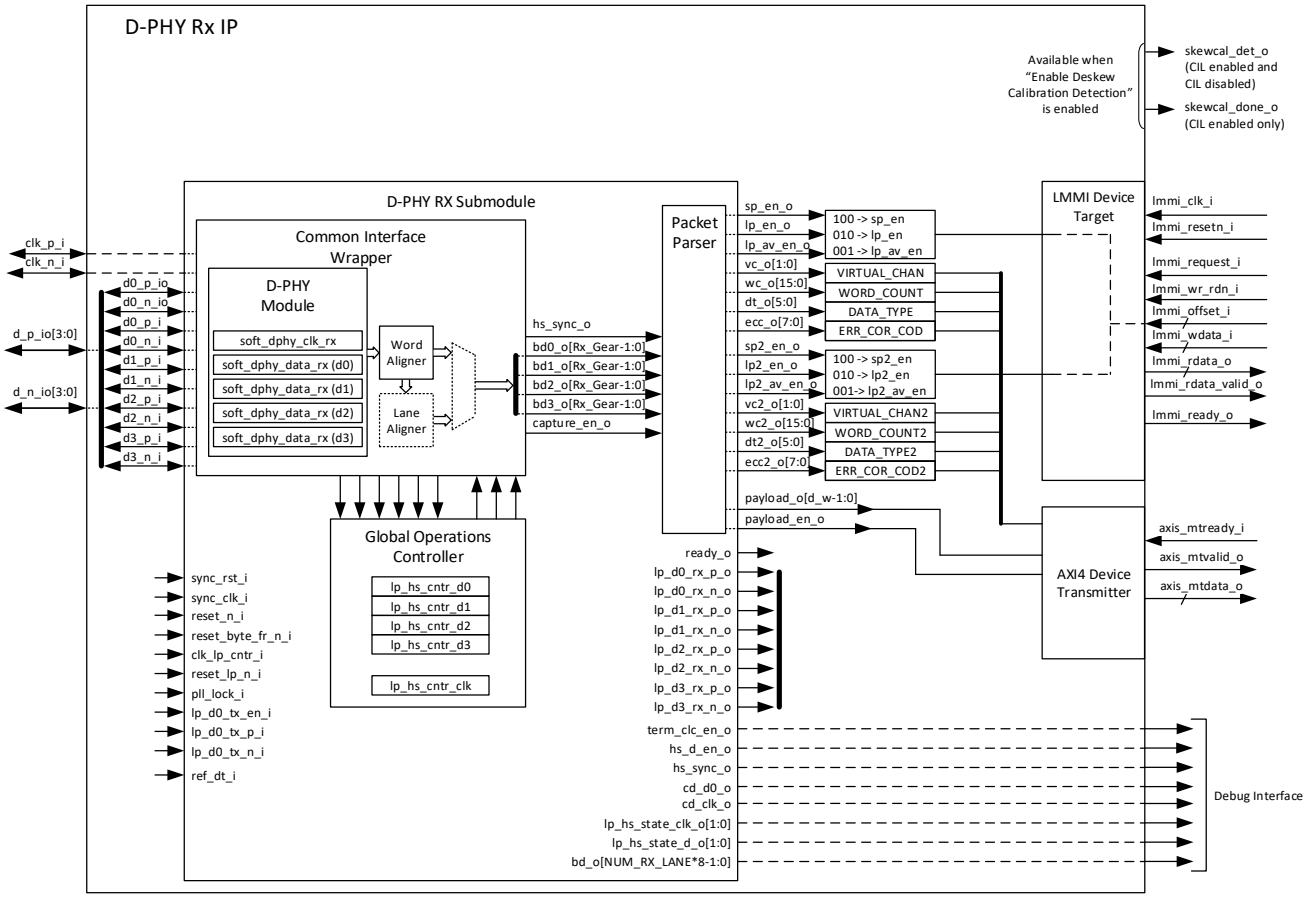

Figure 2.2. D-PHY Rx IP Block Diagram with AXI4-Stream Enabled, and LMMI Disabled

The CSI-2/DSI D-PHY Receiver IP includes the following layers:

- Common Interface Wrapper
- Global Operations Controller
- Packet Parser (optional)
- AXI4 Device Transmitter (optional)
- LMMI Device Target (optional)

The D-PHY Rx IP block diagram configured with the AXI4-Stream OFF and the Packet Parser OFF is shown in Figure 2.10.

The D-PHY Rx IP block diagram configured with the AXI4-Stream ON and the Packet Parser OFF is shown in Figure 2.11.

The D-PHY Rx IP block diagram configured with the AXI4-Stream OFF and the Packet Parser ON is shown in Figure 2.13.

The D-PHY Rx IP block diagram configured with the AXI4-Stream ON and the Packet Parser ON is shown in Figure 2.14.

2.2. User Interfaces

Table 2.1 lists the available user interface and protocols used on the CSI-2/DSI D-PHY Receiver IP.

Table 2.1. User Interfaces and Supported Protocols

User Interface	Supported Protocols	Description
Control	LMMI	Configures the control registers of the Hard D-PHY Rx IP.
Device Transmitter	AXI4	Transmits the payload data (byte data or packet data with virtual channel, data type, and word count).

2.2.1. LMMI Device Target Interface

The LMMI Device Target module (Lattice Memory Mapped Interface) is used for configuring the control registers of the Hard D-PHY Rx IP.

For more information on LMMI, refer to Lattice Memory Mapped Interface and Lattice Interrupt Interface User Guide (FPGA-UG-02039).

2.2.2. AXI4-Stream Device Transmitter Interface

AXI4-Stream Device Transmitter provides an interface for transmitting payload data (byte data or packet data with virtual channel, data type, and word count). Figure 2.3 shows data format when AXI4-Stream is enabled.

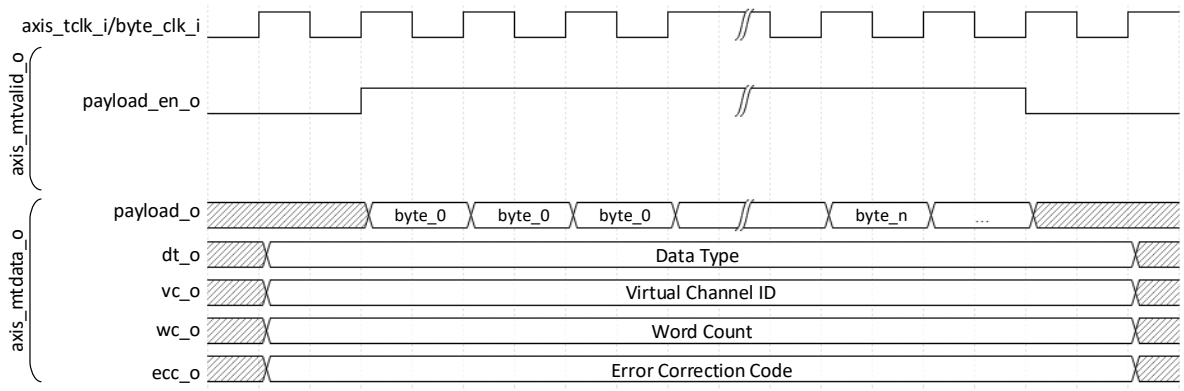

Figure 2.3. AXI4-Stream is ON and Packet Parser is ON Data Format

If the AXI4-Stream device is not enabled and the D-PHY Rx IP is configured with Packet Parser (see Figure 2.13 and Figure 2.14), the following internal signals turn to top level input signals as shown in Figure 2.15:

- payload_en_o
- payload_o[NUM_Rx_LANES*Rx_Gear-1:0]
- dt_o[5:0]
- vc_o[1:0]
- wc_o[15:0]
- ecc_o[7:0]
- dt2_o[5:0]
- vc2_o[1:0]
- wc2_o[15:0]
- ecc2_o[7:0]

If the AXI4-Stream device is not enabled and the D-PHY Rx IP is configured without Packet Parser (see Figure 2.11 and Figure 2.12), the following internal signals turn to top level input signals:

- capture_en_o
- bd0_o[Rx_Gear-1:0]
- bd1_o[Rx_Gear-1:0]

- bd2_o[Rx_Gear-1:0]
- bd3_o[Rx_Gear-1:0]

2.3. D-PHY Rx Common Interface Wrapper

The Common Interface Wrapper module instantiates either the hard or the soft D-PHY module to receive MIPI D-PHY data from all enabled data lanes.

When soft D-PHY is used, this module also includes the Word Aligner module and optionally a Lane Aligner to ensure the correct alignment of bytes among the D-PHY data lanes. See Word Aligner and Optional Lane Aligner for more details regarding these modules.

2.4. D-PHY Module

The D-PHY module instantiates either the hardened D-PHY module or a soft logic equivalent using the FPGA fabric and generic blocks.

2.4.1. Hard D-PHY

Hardened D-PHYs are available in CrossLink-NX devices. Each of these hard blocks contains a Control Interface Logic (CIL) that detects the lane transitions.

The skew calibration detection feature is available for Hard D-PHY configuration. The skew_cal_det_o asserts if it detects the all 1's sync pattern and clears when DP/DN lines are back to LP11 state. An additional signal skew_cal_done_o is available when CIL is enabled. You can optionally enable this feature for data rates of 1.5 Gbps and below.

The hardened PHY module divides the high-speed D-PHY clock to create the output byte clocks clk_byte_o and clk_byte_hs_o. The clk_byte_hs_o toggles when the D-PHY clock lane is active, and can be used as the input clock clk_byte_fr_i when the source D-PHY clock is continuously running. The clk_byte_o, is active only after the deserializer within the hard PHY block detects the Start-of-Transmission pattern (SoTp) and stops when the data lanes go out of high-speed mode. This clock is used as strobe to latch the parallel byte data from the hardened D-PHY block. Because of the dependence on the SoTp detection, the clk_byte_o is out of phase with the clk_byte_hs_o.

The hardened D-PHY also makes use of the input sync_clk_i to clock various control logic. The hardened D-PHY requires a certain period between high-speed transactions to be able to properly handle the internal signals crossing these clock domains. This period is described by the following equations:

For continuous clock mode:

$$T_{HS-EXIT} + T_{LPX} \geq 8T_{clk\_byte} + 3T_{sync\_clk}$$

For non-continuous clock mode:

$$T_{CLK-POST} + T_{LPX} \geq 8T_{clk\_byte} + 3T_{sync\_clk}$$

2.4.2. Soft D-PHY

When the soft PHY is used, the clk_byte_o is the same as the clk_byte_hs_o. Both are the outputs of a clock divider with the D-PHY clock as its input. Refer to the High-Speed IO Interface document of each device for the Soft MIPI D-PHY architecture implementation details.

2.5. RX_FIFO

As mentioned in the D-PHY Module section, the data words from the D-PHY module are synchronous with the output clock ssclk_byte_hs_o. The rest of the data path, including the soft aligner modules, are clocked by the input clk_byte_fr_i. An rx_fifo module is used to cross the data between these two clock domains. When the hardened Control Interface Logic (CIL) is enabled, the RX_FIFO Module is not used. Instead, a single 1024-deep dual-clock FIFO is used.

In the case of the soft PHY implementation, the RX_FIFO is instantiated before the D-PHY module and the Word Aligner module. Data being buffered includes the hs-zero bytes before the SoTp, the actual packets, the trail bits, and the data lane value right before the D-PHY lanes transition to LP-11.

The following subsections describe the various implementation types.

2.5.1. RX_FIFO OFF

If the clk_byte_fr_i has the same frequency and is synchronous with the clock strobe clk_byte_o, the RX_FIFO can be removed. In the actual generated design, there is still a fixed 4-deep single clock FIFO implemented as LUTs.

This setting is recommended when the D-PHY module is using Soft D-PHY implementation with the D-PHY clock running continuously in high-speed mode and the clk_byte_fr_i is driven by the clk_byte_hs_o.

2.5.2. RX_FIFO_TYPE = SINGLE

The RX_FIFO Single type is primarily intended as an elastic FIFO to buffer the frequency or phase difference between the stoppable byte clock domain (clk_byte_o) and the continuous byte clock domain.

The MIPI D-PHY protocol does not allow data throttling when high-speed transfer is already on-going. Full or empty FIFO condition does not halt the data stream. Configure the depth of the FIFO and the packet delay to ensure it does not overflow or underflow.

When the two byte clocks have slightly different frequency (such as using two different oscillators with ppm tolerance), it is recommended to increase the buffered data before reading out from the FIFO using the packet delay parameter. If the continuous clock is slightly slower than the strobe clk_byte_o and the low-power period between high-speed transfers is enough to absorb the time difference, the delay value can be set to 1; the depth depends on the amount of possible accumulated data because of the clock difference.

If the two byteclocks are asynchronous with each other, set Clock Mode to DC.

When the frequency of both clocks are exactly the same but are out-of-phase, when the input clock clk_byte_fr_i is driven by the clk_byte_hs_o from the Hard D-PHY module in continuous clock mode, set the delay value to 1 and depth to 16, which is the minimum depth for the FIFO control signals crossing clock boundaries.

In all cases of the Single Type RX_FIFO, the read from the FIFO continues until the empty flag asserts. The interval between high-speed transactions (low-power blanking) must be long enough to ensure the FIFO is already empty before the next one is written, otherwise, the FIFO assumes it is still part of the previous data stream. This causes the word aligner to miss the SoTp of the second data stream and interpret the packets erroneously.

Figure 2.4 illustrates the contents of the FIFO when a high-speed blanking DSI stream is being buffered using the RX_FIFO SINGLE.

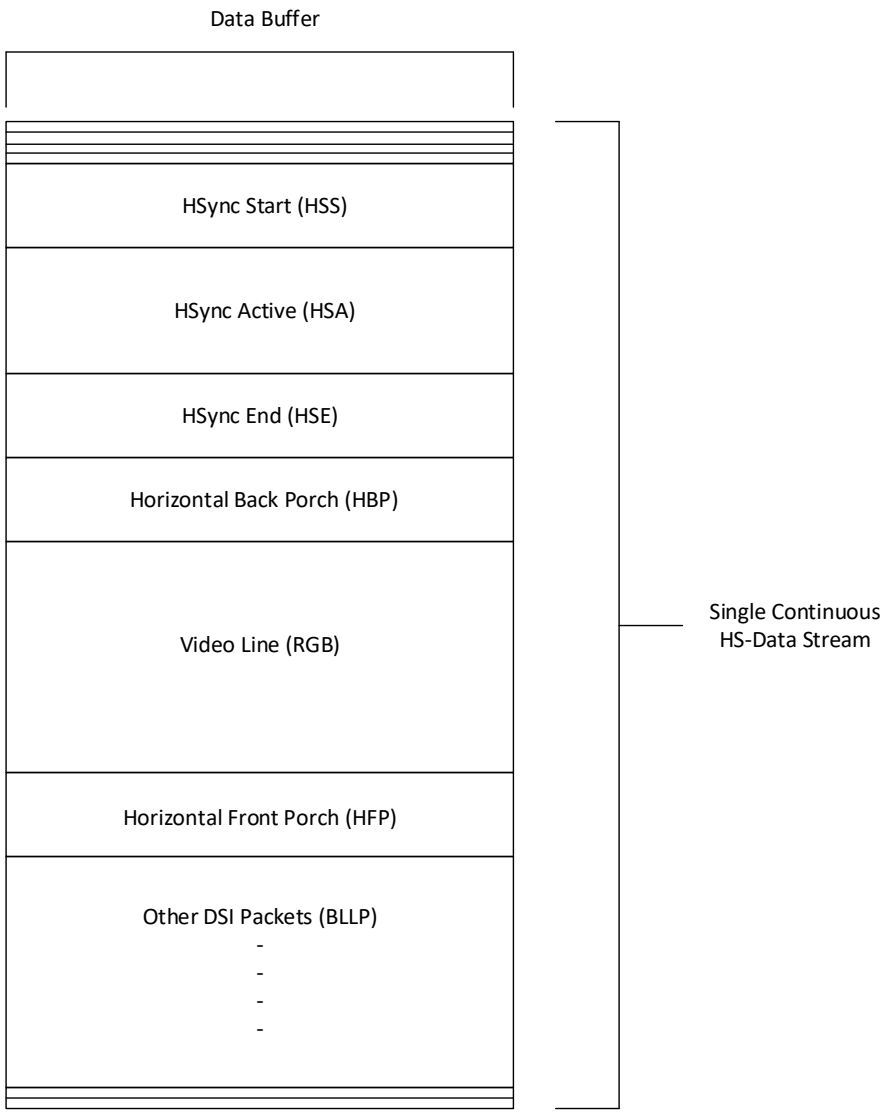

Figure 2.4. RX_FIFO SINGLE

2.5.3. RX_FIFO_TYPE = PINGPONG

This FIFO instantiates two dual_clock FIFOs that alternately stores data of every high-speed transaction. Each data buffer can hold the largest data within a high-speed transaction, including the hs-zero, SoTp, and trail bits.

Similar with the single type, this implementation also has a parameterized delay before reading out from the buffer to maintain intervals between packets. If packet delay = 0, read starts once the empty signal deasserts and the other one is not busy with read. If packet delay is non-zero, read from that buffer starts once the delay value is met and the other one is not busy with read. This implementation does not track the number of entries within the buffers; read stops once the empty flag asserts. Each buffer is reset after the read operation.

This type is more suitable for high-speed transfers with short intervals because data is written alternately between the two buffers. This is recommended for DSI with low-power blanking.

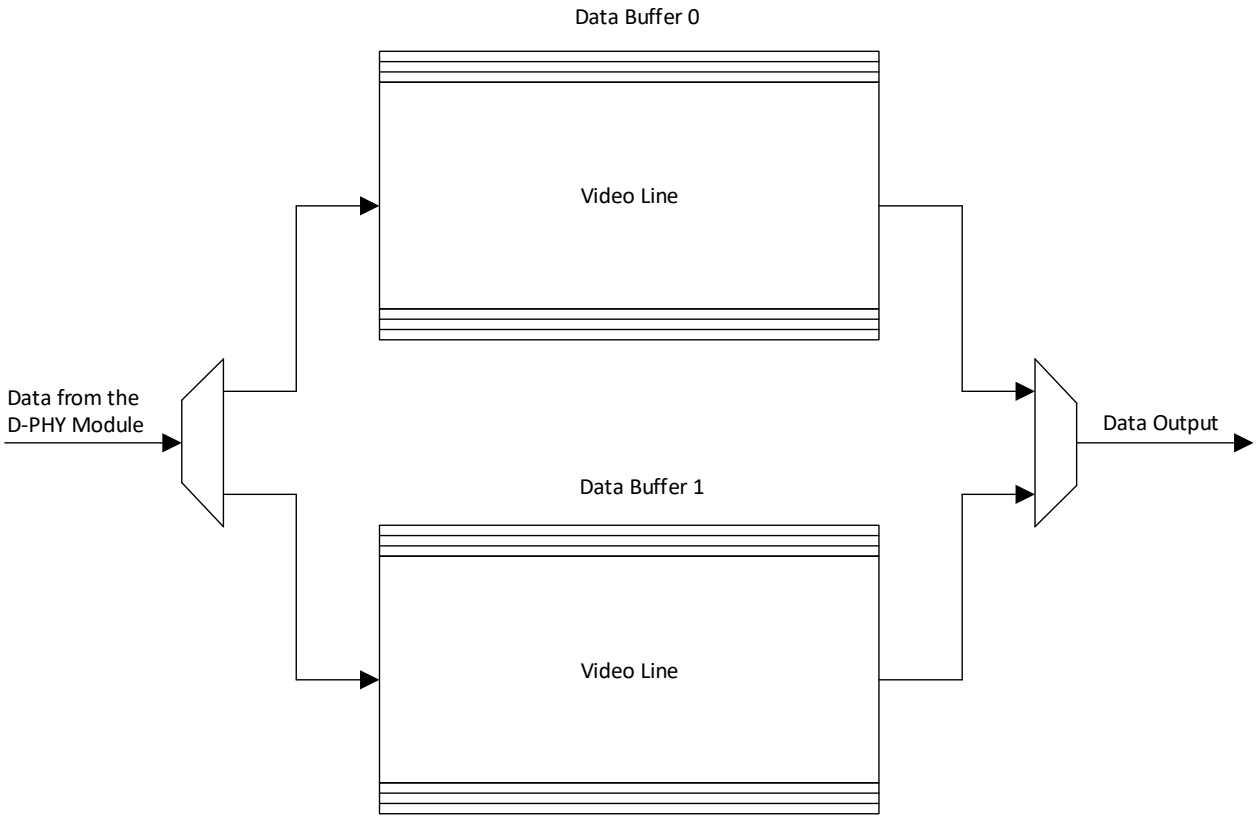

Figure 2.5. RX_FIFO PINGPONG

2.5.4. RX_FIFO_TYPE = Queue

This FIFO instantiates one dual-clock FIFO. This FIFO also acts as a circular buffer that holds data from multiple high-speed transactions.

Unlike the other two types, this FIFO does not have a delay counter. Instead, HS data is buffered completely and a counter tracks the number of rows written during the high-speed transaction. This count is stored in an entry queue.

When there is a valid entry in the entry queue, read from the data buffer is triggered. The number of read cycles from the data buffer corresponds to the entry read from the entry queue. This enables the FIFO controller to distinguish the boundaries between successive HS transactions. This setting introduces significant latency on the first video line, but also enables the IP to support short intervals between HS transactions.

This implementation is suitable for CSI-2, where the packet intervals are not critical, but the intervals between successive high-speed transactions are short.

Figure 2.6 illustrates a sample entry within a 4-deep entry queue for a CSI-2 sequence.

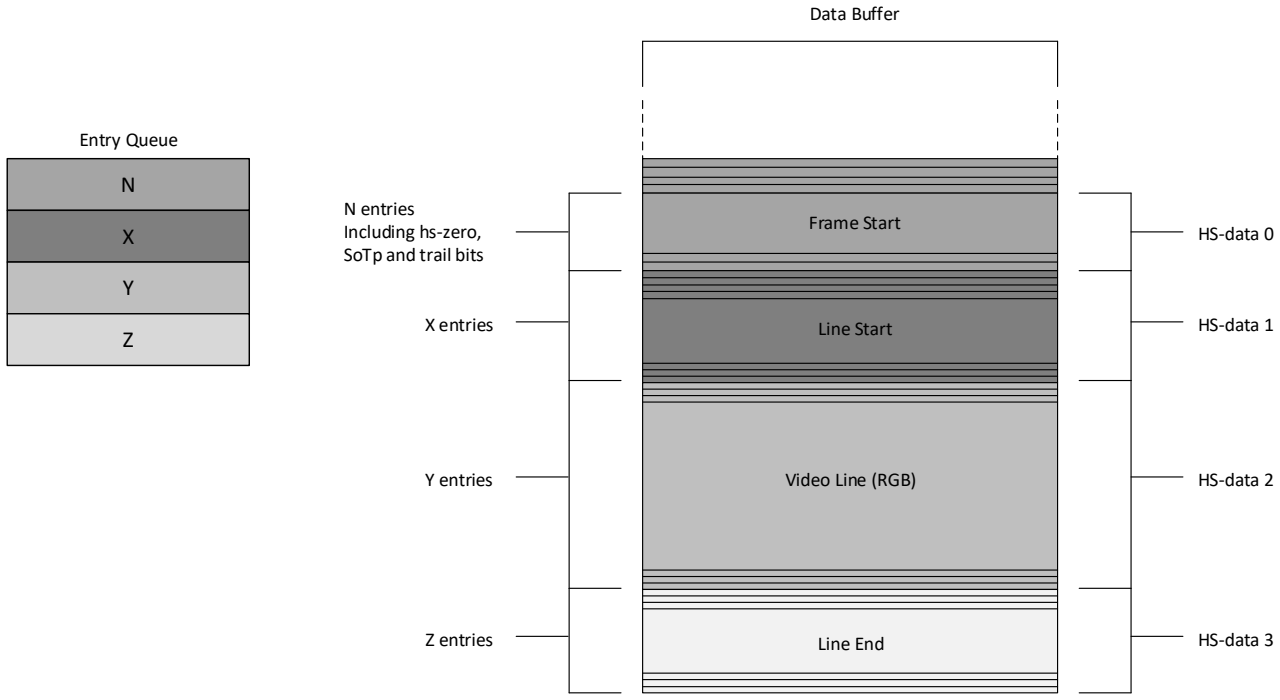

Figure 2.6. RX_FIFO QUEUE

2.6. Global Operations Controller

This block detects the low power state transitions of the clock and data lanes. This controller also controls the resistor termination when switching between low power and high speed states. The Global Operations Controller only supports low power to high speed sequence (LP-11 -> LP-01 -> LP-00 -> HS0 -> HS0/1 -> LP11). Figure 2.7 shows the LP-to-HS transition flow diagram for data lanes.

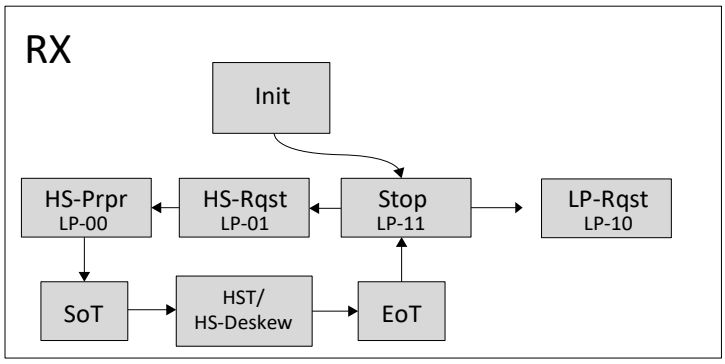

Figure 2.7. MIPI D-PHY Rx LP to HS Transition Flow Diagram on Data Lanes

When the MIPI D-PHY clock is continuous, the HS termination enable of clock lane is tied to VCC. When the MIPI D-PHY clock is non-continuous, the HS termination enable of clock lane becomes active right after proper LP to HS transition is observed. This function requires a reference clock input. Figure 2.8 shows the required LP to HS transition on clock lane per MIPI D-PHY Specification version 2.1.

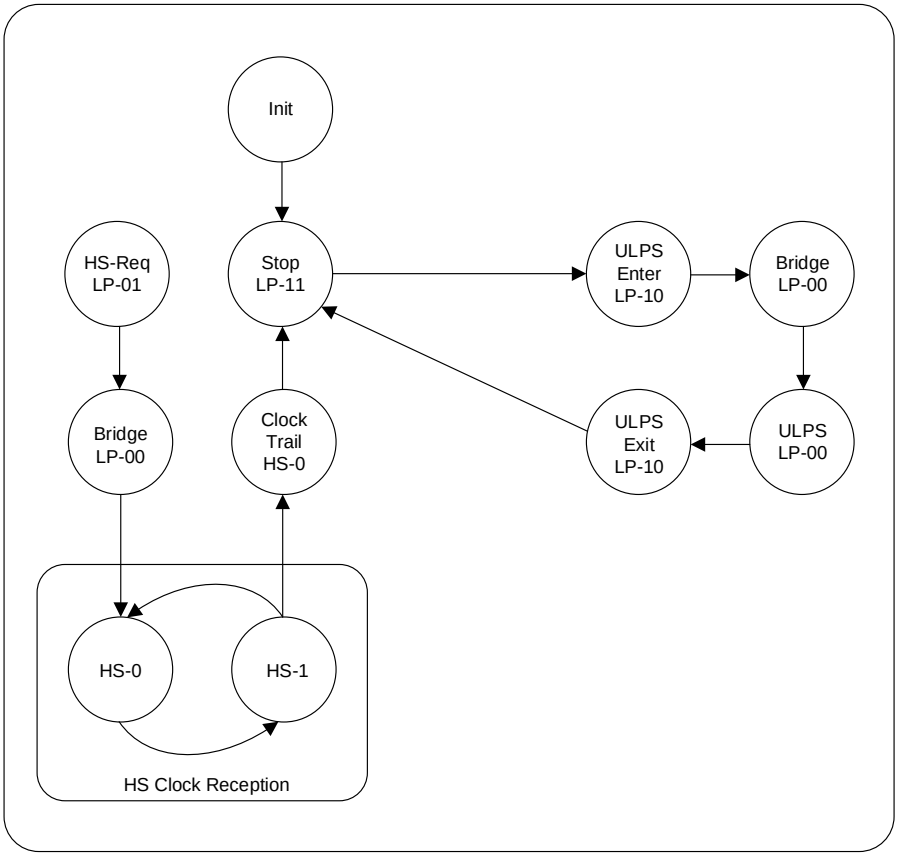

Figure 2.8. MIPI D-PHY Rx LP to HS transition on Clock Lane

During normal operation, a data lane is either in control or in high-speed (HS) mode.

In low-power (control) mode, the D-PHY transmitter performs the high-speed entry sequence which consists of driving LP11 -> LP01 -> LP00 on the data lanes as shown in Figure 2.9. Upon successful detection of this sequence, the Global Operations Controller enables the differential resistor termination to receive the high-speed data. A free-running byte clock is used during HS mode.

Once enabled, HS receiver termination continues to receive the data until it encounters the LP11 state on the lanes, which is also known as the Stop State. The Stop State brings back the data lane from high-speed mode to low-power mode.

To handle the transition effects in the data lanes when going from low power to high-speed mode, the D-PHY protocol requires the receiver to neglect the data lanes for a certain time interval, tHS_SETTLE. In the IP, you can set this time interval through the Data Settle Cycle attribute in the GUI. If LMMI interface is available, you can configure this interval by writing to the NOCIL_DSETTLE LSB and MSB registers. Alternatively, if LMMI is disabled but the Configurable Data Settle Count is checked, an input port rxcsr_datsettlecyc_i is available for the same function.

For Hard D-PHY with CIL bypassed, the counter for the data settle, and the FSM that detects the low power to high-speed transition of the data lanes, are in the clk_byte_fr_i domain.

For Soft D-PHY, the FSM is also in the clk_byte_fr_i domain, but the tHS_SETTLE timer is in the clk_byte_o domain.

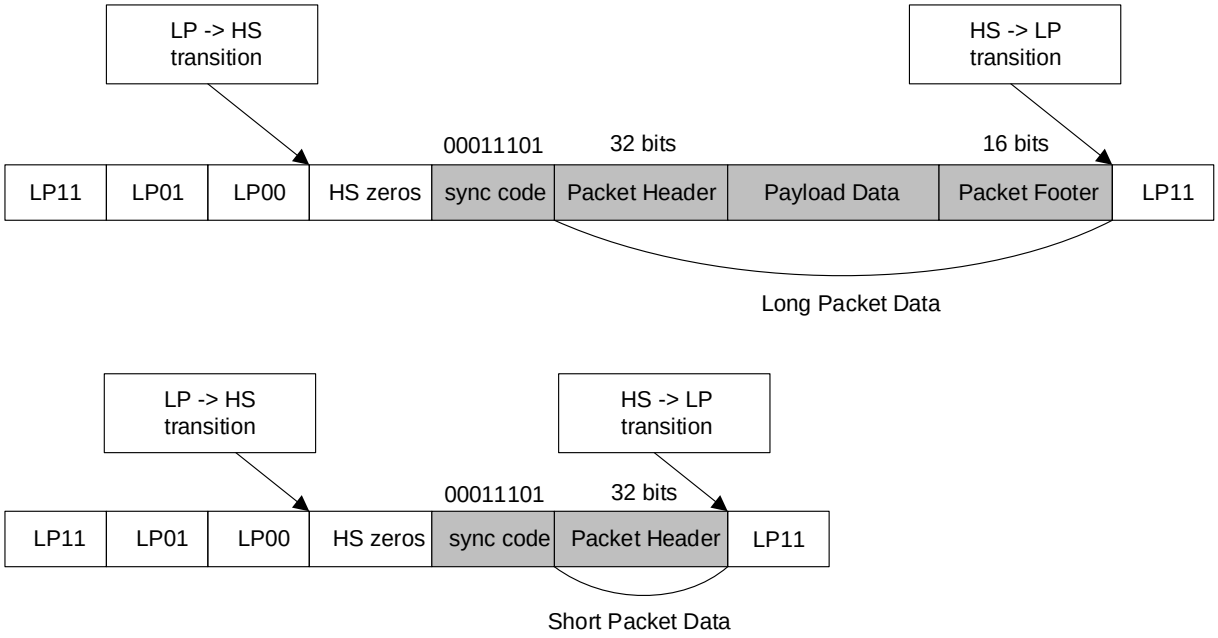

Figure 2.9. High-Speed Entry Sequence and Payload Data Transmission Cycle on Data Lanes

2.7. D-PHY Rx IP without Packet Parser

When D-PHY Rx IP is configured without the packet parser, as shown in Figure 2.10 and Figure 2.11, the output is the received bytes from the D-PHY Rx IP starting from the reception of the Start of Transmit (SoT) code in all the active data lanes until the detection of LP-11, signifying the end of high-speed transmission. The interfacing logic obtains and decodes the valid data packets from the trail. This configuration is useful for bridging D-PHY packets without going to the protocol level.

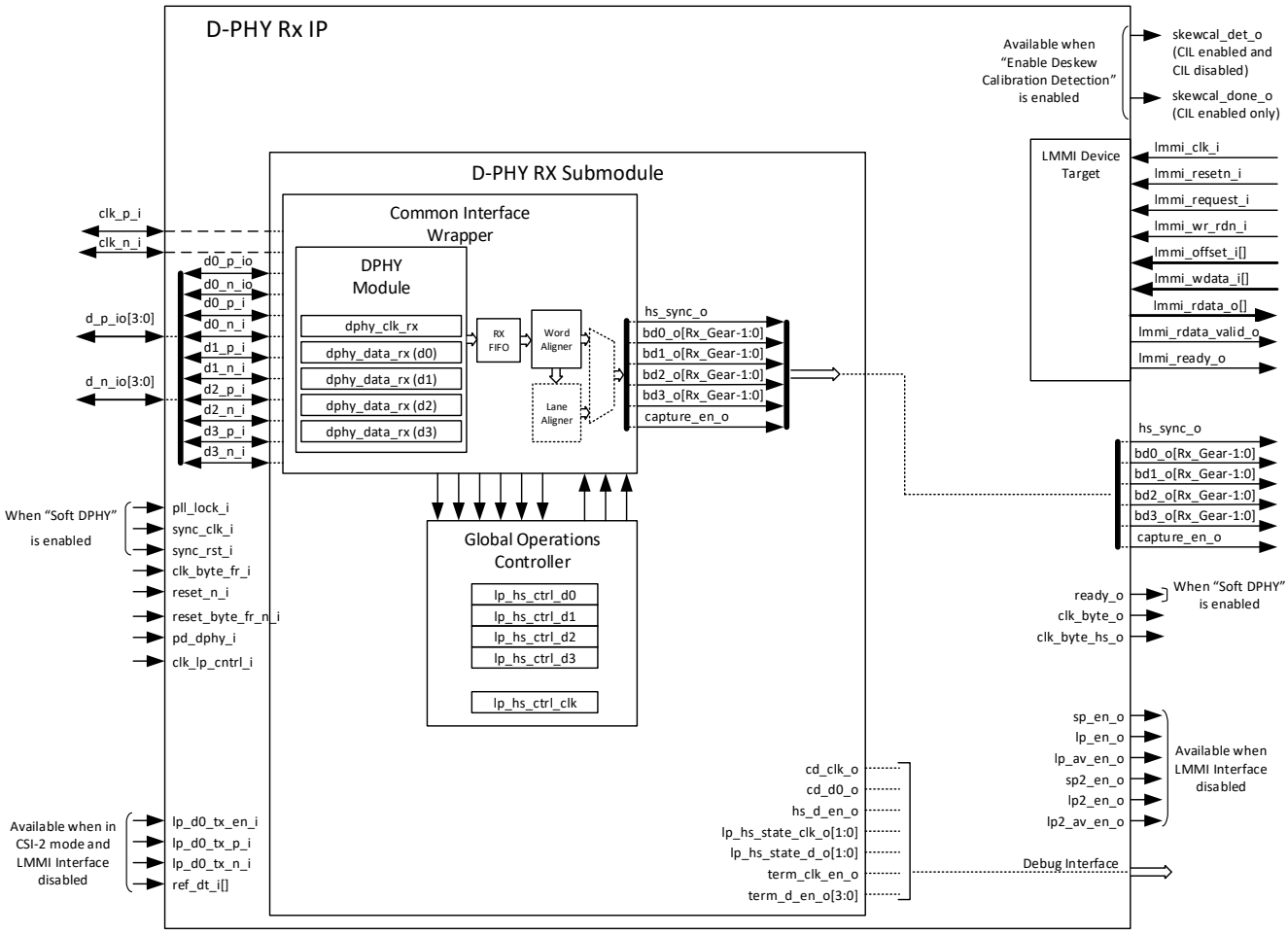

Figure 2.10. D-PHY Rx IP Configuration with AXI4-Stream Disabled and Packet Parser Disabled

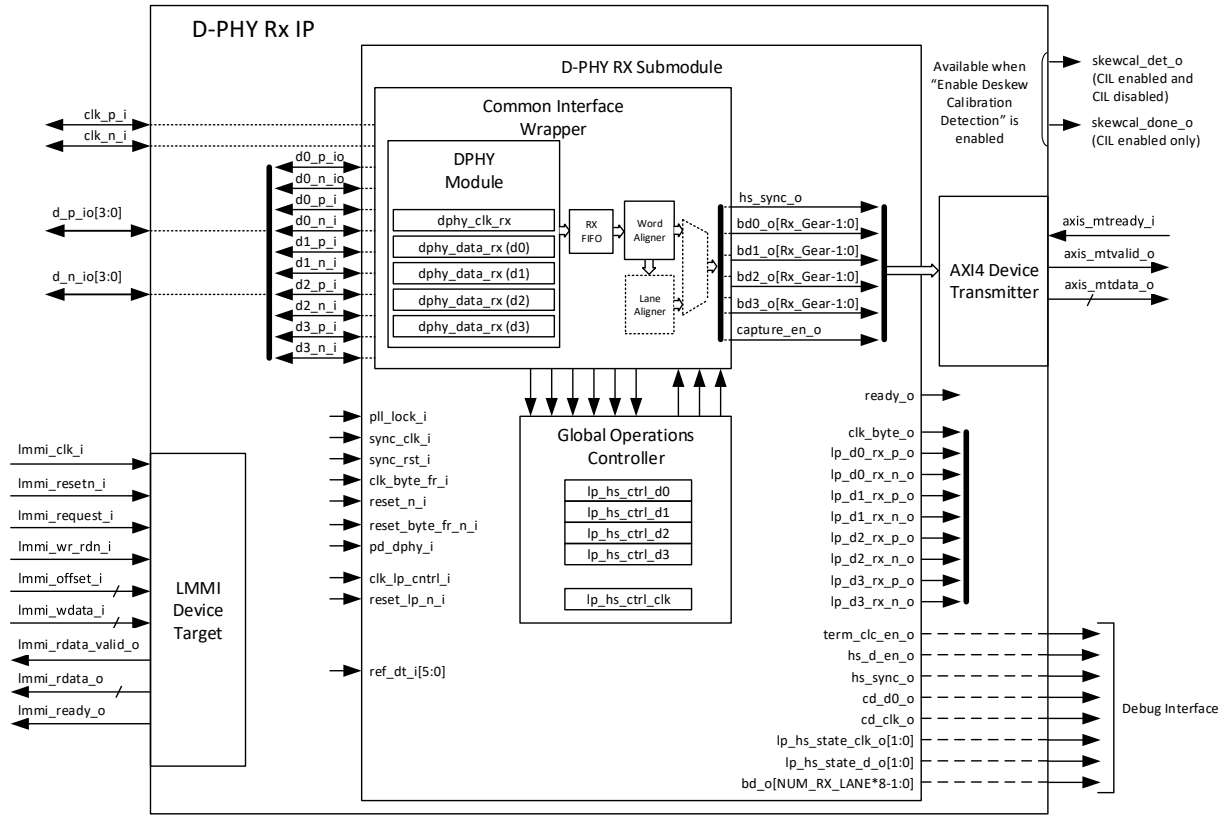

Figure 2.11. D-PHY Rx IP Configuration with AXI4-Stream Enabled and Packet Parser Disabled

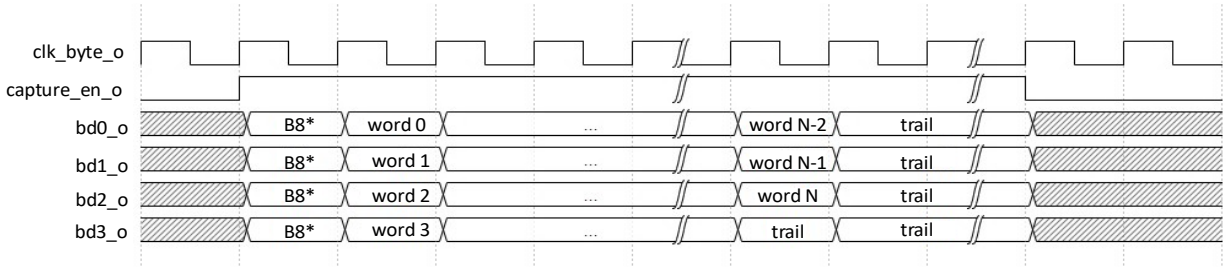

Figure 2.12. D-PHY Rx IP Output Timing Diagram without Packet Parser

2.8. D-PHY Rx IP with Packet Parser

When D-PHY Rx IP is configured with the DSI/CSI-2 packet parser included, as shown in Figure 2.13 and Figure 2.14, the parser checks the incoming data for a valid data type and the corresponding packet fields.

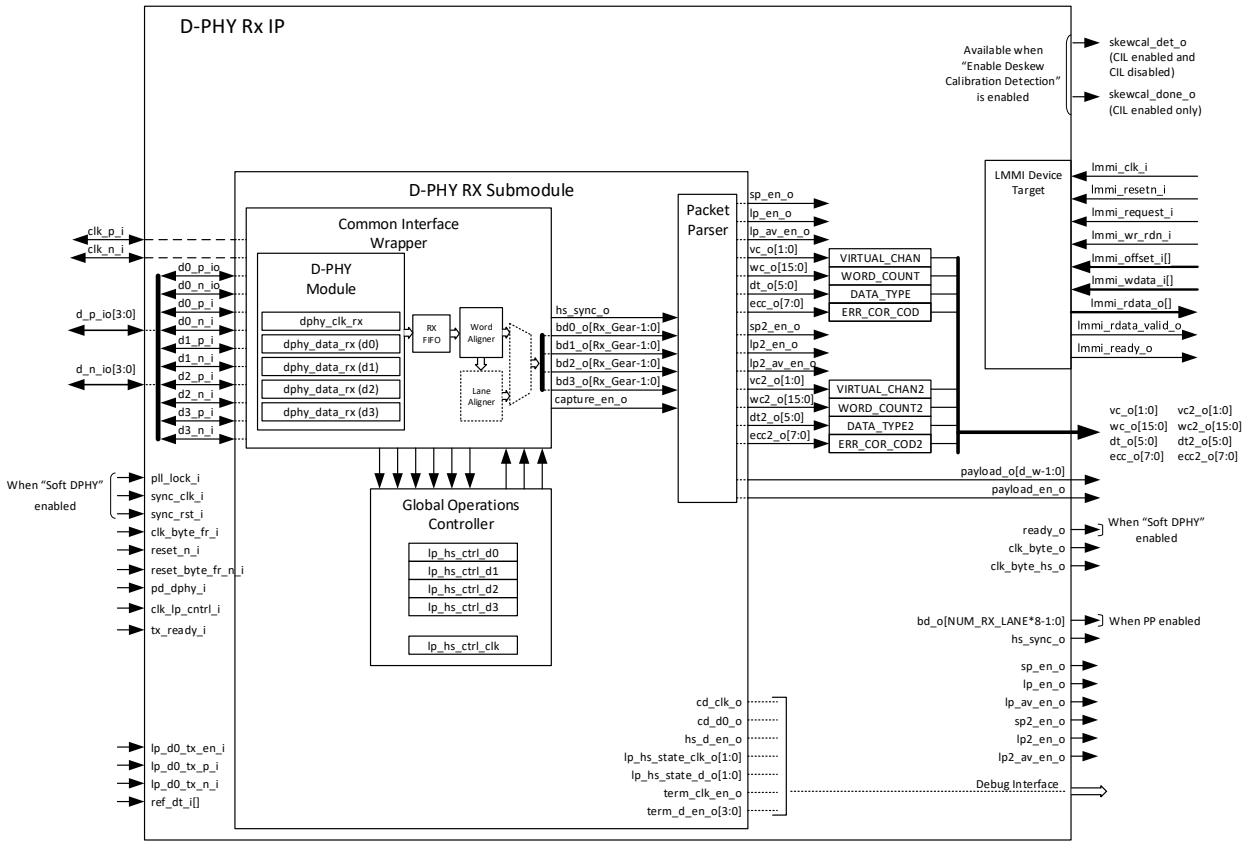

Figure 2.13. D-PHY Rx IP Configuration with AXI4-Stream Disabled and Packet Parser Enabled

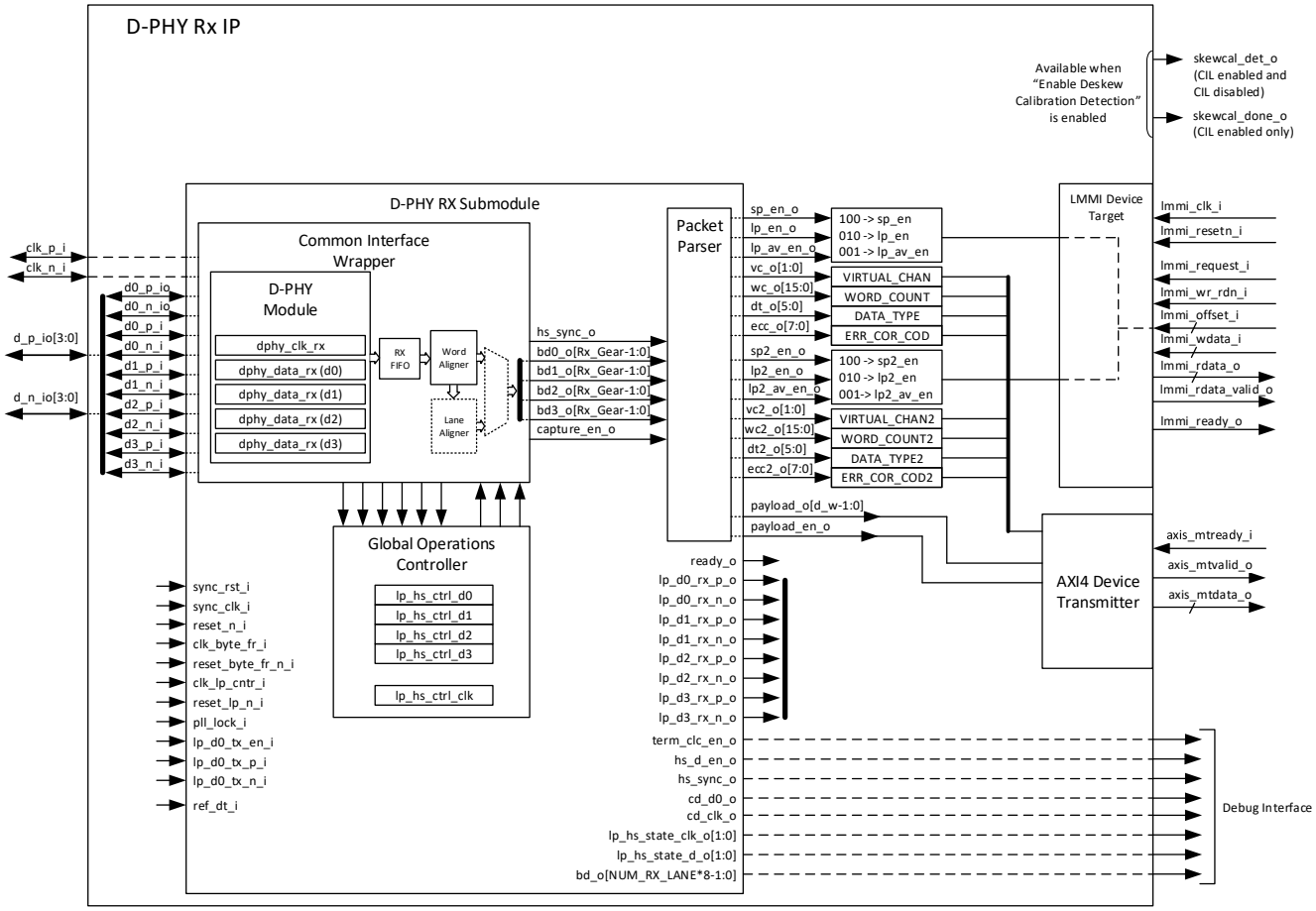

Figure 2.14. D-PHY Rx IP Configuration with AXI4-Stream Enabled and Packet Parser Enabled

2.9. Packet Parser

The Packet Parser module parses the data bytes from D-PHY Common Interface Wrapper, and detects short and long packets defined by MIPI DSI or MIPI CSI-2. This block extracts video data and other control parameters from the packets. There are no signals from the external logic to this block to control the flow of output data. The interfacing logic must provide ample buffering to ensure the continuous flow of data from this submodule is transferred correctly. The output-timing diagram of D-PHY Rx IP interface with the packet parser enabled is shown in Figure 2.15.

The lp_en_o or sp_en_o signal asserts when a valid data type is received. These signals also indicate the valid data type, virtual channel ID, wordcount, and ECC fields. The lp_av_en_o only asserts with the lp_en_o, if the long packet received is the same as the input reference data type ref_dt_i. This is to differentiate active video packets from other long packets, such as null or blanking. Consequently, this signal does not assert on any video data type other than the defined ref_dt_i value. The payload_en_o signal indicates that the data in the payload_o bus contain the valid payload bytes. The width of the payload, data_width, is the number of gear bits multiplied by the number of data lanes. Upper data bytes for the last payload data must be ignored if the wordcount is not a multiple of data_width/8. The interfacing module extracts the correct payload bytes based on the valid output wordcount wc_o.

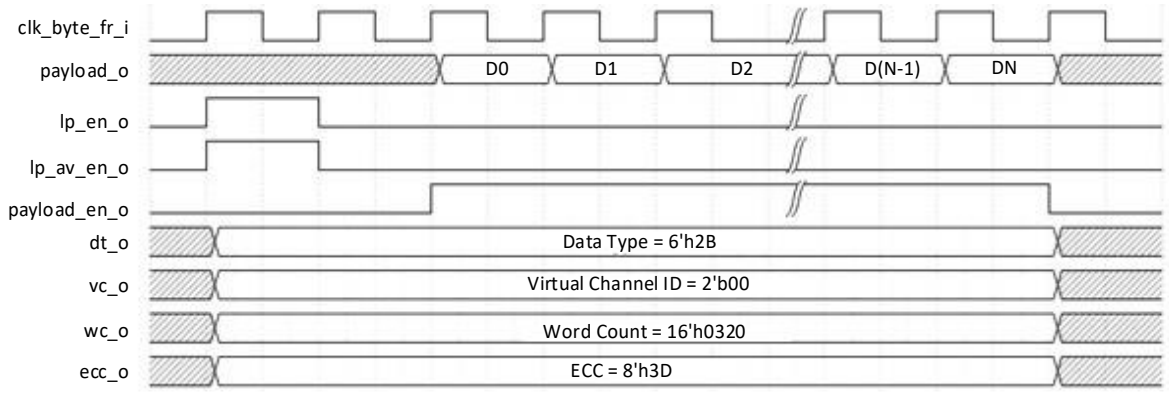

Figure 2.15. D-PHY Rx IP Output Timing Diagram with Packet Parser

When the input data bus going to the packet parser is greater than 32, the Second Set of Packet is valid. In this configuration, two packet headers may simultaneously be decoded within the same byte clock cycle.

The packet parser input and output timing diagram with valid Second Set of Packet information is shown in Figure 2.16.

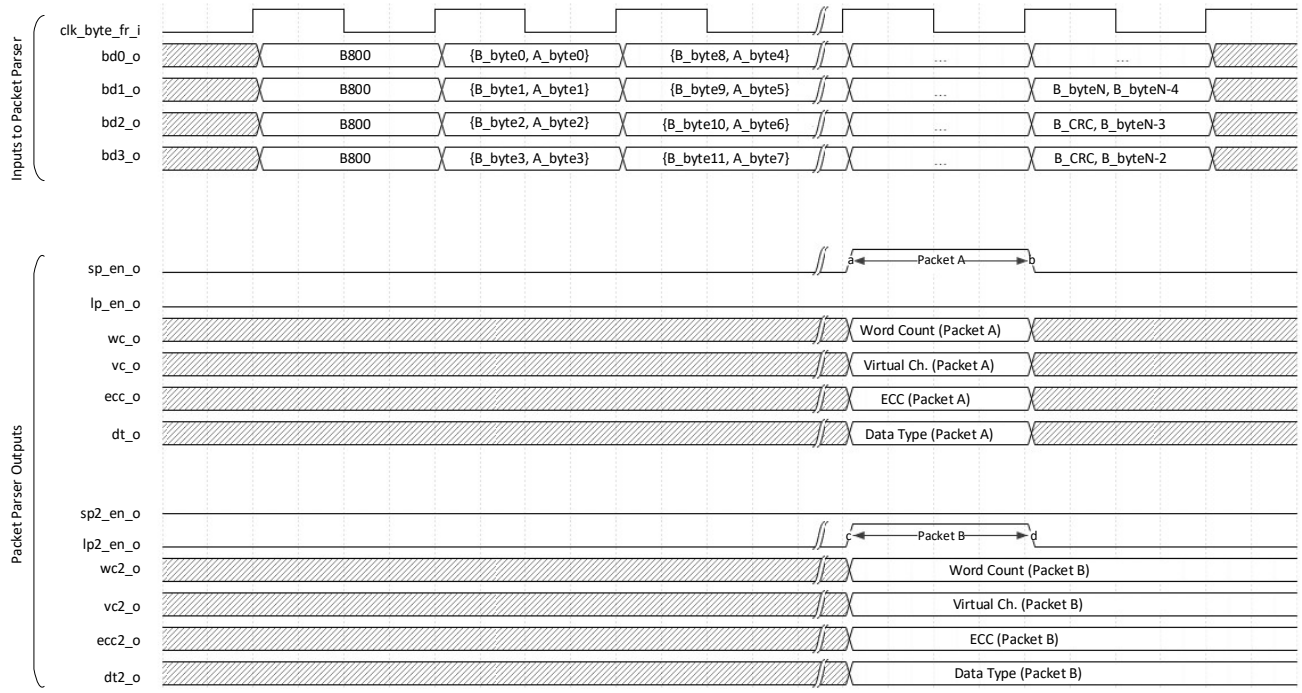

Figure 2.16. Output Timing Diagram with Valid Second Set of Packet Information

2.10. Word Aligner and Optional Lane Aligner

Before the payload data of every HS burst on each lane, the transmitting D-PHY inserts a sync sequence Start-of-Transmit (SoT) pattern. For hardened PHY, the hard deserializer checks for the sync pattern on each lane. For soft PHY, a Word Aligner module detects the pattern in the deserialized data and establishes the correct byte boundary on the high-speed payload data.

The Word Aligner logic detects the SoT pattern from each lane and ensures the parallel data are word (byte) aligned. The design assumes that input data lanes are driven at the same time, and skew between data lanes are less than 1 UI. However, because of deserialization, data buffering, and handling of clock domain crossing signals, the detection of the aligned data from all the lanes may not happen at the same cycle. An optional lane aligner module may be instantiated to rectify this issue, see Figure 2.17.

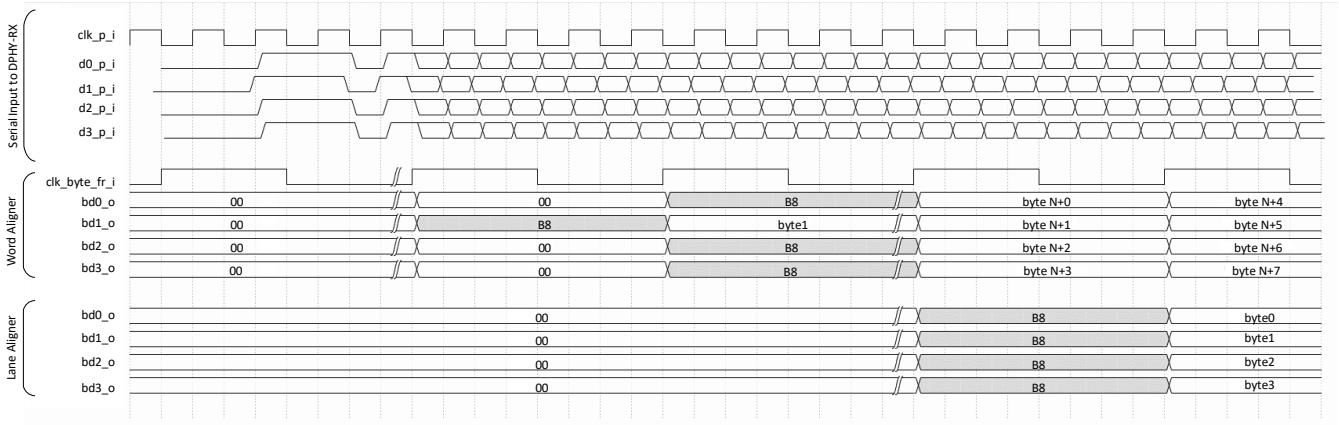

Figure 2.17. Lane Data Alignment

2.11. Lane/Gear Dynamic Reconfiguration with CRC Checking Function

Starting from v1.6.0, the IP with hard D-PHY with CIL enabled can be reconfigured at run time to support different lane and/or gear configurations by programming certain registers and CRC checking function. The LMMI interface must be enabled to access the Hard D-PHY registers.

To enable these functions, check the following attributes in the Module/IP Blocks Wizard in the Lattice Radiant software:

- **Enable Hard D-PHY with CIL Bypass** unchecked.
- **Enable Packet Parser and CRC Check**.
- **Enable LMMI Interface** – required for Hard D-PHY. Internal Hard D-PHY registers are accessible only through LMMI.
- **Enable Lane/Gear Dynamic Reconfiguration** – required to dynamically change the number of active lanes and the width of the deserialized data (gear). Set the parameters of Number of RX Lanes and RX Gear to the maximum value at compile time to dynamically reconfigure all possible combinations.

To reconfigure the dynamic registers, follow these steps:

1. Stop the D-PHY Transmitter from sending packets.
2. Configure the IP registers.
   - For Hard D-PHY with CIL_enabled:
     - Write to register 0x09[2:1] to change between Gear 8 and Gear 16.
     - Write to register 0x0A[2:1] to change the number of active lanes.
     - Write to register 0x0A[3], 0x0B[3:0], and 0x0C[0] to change the tCLK-SETTLE counter value.
     - Write to register 0x0F and 0x10 to change the tHS-SETTLE counter value.
3. Configure the D-PHY Transmitter.
4. Assert all the resets to the IP *except* the LMMI reset:
   - reset_byte_fr_n_i
   - reset_n_i
   - pd_dphy_i
   - sync_rst_i
5. Trigger the D-PHY Transmitter to send normal transactions.

If you only require dynamic reconfiguration of lane/gear without the CRC check function, do not enable the CRC Check and Lane/Gear Dynamic Reconfiguration attributes. You can enable Hard D-PHY with CIL Bypass unchecked and the LMMI Interface, and dynamically write to the corresponding Hard D-PHY registers.

3. IP Parameter Description

The configurable attributes of the D-PHY Rx IP are shown in the following tables. You can configure the IP by setting the attributes accordingly in the IP Catalog's Module/IP wizard of the Lattice Radiant software.

Wherever applicable, default values are in bold.

3.1. General

Table 3.1. General Attributes

Attribute	Selectable Values	Description
Receiver		
Rx Interface Type	DSI, **CSI-2**	D-PHY Rx interface type.
D-PHY RX IP	**Hard D-PHY**, Soft D-PHY	Implementation of the PHY layer of the D-PHY Rx. For Certus-NX devices, only "Soft D-PHY" is available.
Number of RX Lanes	1, 2, 3, **4**	Number of D-PHY Rx high-speed ports. 3-lane configuration is only available when the Packet Parser is disabled.
RX Gear	**8**, 16	This is the width of the deserialized data stream per data lane. *RX Gear = 16* is available only on *D-PHY RX IP = 'Hard D-PHY'*.
Enable Deskew Calibration Detection	checked, **unchecked**	This entry is available when data rate is less than 1.5 Gbps. This is automatically set to enabled (and grayed out) at data rate over 1.5 Gbps as this feature is required for data rates over 1.5 Gbps.
Clock		
RX Line Rate per Lane (Mbps)[1]	160–2500, **1000**	Maximum bandwidth for *RX Gear = 16*,
	80–1500, **1000**	Maximum bandwidth for *RX Gear = 8*.
D-PHY Clock Frequency (MHz)	40–1250, **500**	Operating frequency of the PHY layer.
Byte Clock Frequency (MHz)	10–187.5, **125**	Operating frequency of the byte_clock_o of the receiving end. Grayed out – non-configurable.
D-PHY Clock Mode	**Continuous**, Non-continuous	Determines the clock mode of the PHY layer. The D-PHY protocol has an option for the clock lane to go to low-power in between high-speed transfers to reduce power consumption. If the mode selected is "Non-continuous", the control logic that detects the LP-to-HS and HS-to-LP sequences of the clock lane is enabled. Otherwise, the IP assumes the D-PHY clock lane is always in high-speed mode.
Sync Clock Frequency (MHz)	24–200, **60**	Operating frequency of the components interfaced with the fabric.
Module Architecture		
CIL Bypass	**checked**, unchecked	When using *D-PHY RX IP = 'Hard D-PHY'*, this option bypasses the built-in Control Interface Logic (CIL) of the Hard D-PHY. CIL is the hardened block that controls the clock and data lane state transitions. If the CIL is bypassed, soft logic is used.
Enable Lane Aligner Module	checked, **unchecked**	Enables the line alignment feature. This feature is available when *D-PHY RX IP = "Soft D-PHY"*.
Enable Packet Parser	**checked**, unchecked	Enables or disables the packet parser function. When the packet parser is enabled, the IP reads through the contents of the packet header and converts them into CSI-2 or DSI related bus/signal outputs. When this is disabled, the deserialized data are sent out without analyzing the contents.
Enable AXI4-Stream Interface	checked, **unchecked**	Enables the AXI4-Stream bus.

Attribute	Selectable Values	Description
Enable LMMI Interface	checked, **unchecked**	Enables the LMMI bus.
Enable Miscellaneous Status Signals	checked, **unchecked**	Enables or disables the miscellaneous status signals. When enabled, select internal signals such as high-speed termination enables are available to the top level IP wrapper. This may be used for debugging purposes.
Enable CRC Check	checked, **unchecked**	Available for selection when Enable Packet Parser is checked, Enable AXI4-Stream Interface is unchecked, Rx Interface Type = CSI-2, D-PHY RX IP = “Hard D-PHY”, and CIL Bypass is unchecked. When this option is checked, the IP core computes the CRC for the payload and checks against the packet footer CRC. CRC error flag with valid is provided as output status signals.
Enable Lane/Gear Dynamic Reconfiguration	checked, **unchecked**	Available for selection when Enable Packet Parser is checked, Enable AXI4-Stream Interface is unchecked, Rx Interface Type = CSI-2, D-PHY RX IP = “Hard D-PHY”, CIL Bypass is unchecked and Enable CRC Check is checked. When this option is checked, the CRC checking function performs based on the dynamically configured lane and gear. Do not need check this option if you do not need CRC checking function. You can dynamically reconfigure the lane and/or gear setting of the Hard D-PHY with CIL Bypass unchecked. You can enable the LMMI interface and dynamically write to the corresponding registers for lane and/or gear change.
Timing Parameter		
Customize Data Settle Cycle	checked, **unchecked**	Enables customization of the data settle parameter, when *CIL Bypass* is checked. Check this option and re-calculate the Data Settle Cycle value based on the guidance.
Data Settle Cycle	1–27, **14**	Value of the data settle parameter when *CIL Bypass* is checked. This parameter corresponds to the THS-SETTLE timing parameter as defined in the MIPI D-PHY specification. When this Customize Data Settle Cycle parameter is checked, you can determine the value of Data Settle Cycle (in the unit of byte clock cycle) based on the equation of ceil((85 ns+6*UI) / TCLK_BYTE) , where 1 UI is equal to half of the period of D-PHY clock and TCLK_BYTE is equal to the period of byte clock (clk_byte_o for Soft D-PHY, clk_byte_fr_i for Hard D-PHY with CIL bypassed). For Soft D-PHY mode, the calculated value has to be offset down by 3 clk_byte_o cycles for clock domain crossing. Example: D-PHY Rx IP = Soft D-PHY D-PHY Clock Frequency (MHz) = 220 MHz Frequency of clk_byte_o = 220/4 = 55 MHz TCLK_BYTE = 1/55 = 18182 ps 1 UI = (1/220)/2 = 2272.5 ps 85 ns+6*UI = 85000 + 6*(2272.5) = 85000 + 13635 = 98635 ps In this example, Data Settle Cycle = ceil(98635/18182) = 6. Taking the 3 clock cycles internal delay into account, the selected value must be set to 6–3 = 3.
Customize CIL Data Settle	checked, **unchecked**	Enables customization of the data settle parameter, when *CIL Bypass* is

Attribute	Selectable Values	Description
		unchecked.
CIL Data Settle Cycle	4–10, **6**	Value of the data settle parameter when *CIL Bypass* is unchecked.
Customize CIL Clock Settle	checked, **unchecked**	Enables customization of the clock settle parameter, when *CIL Bypass* is unchecked.
CIL Clock Settle Cycle	4–19, **9**	Value of the clock settle parameter when *CIL Bypass* is unchecked.
Configurable Data Settle Count	checked, **unchecked**	When checked, an input bus rxcsr_datasettlecyc_i is available. You can adjust the tHS-SETTLE timer value without generating a new bitstream. This attribute is available starting from IP version 1.6.0.

Notes:

1. The maximum data rate depends on the gear, family, package, and speed grade of the device. Check the device data sheet for more information.
2. In IP version 1.6.0 onwards, a GUI tab for modifying the delay cell within the soft PHY is available.

3.2. RX_FIFO Settings

Table 3.2. RX FIFO Settings Attributes

Attribute	Selectable Values	Description
RX_FIFO		
RX_FIFO Enable	**checked**, unchecked	Enables or disables the RX FIFO function. When enabled, the FIFO after the PHY may be customized. Otherwise, a 4-register deep buffer is used. When using *D-PHY RX IP = Hard-D-PHY*, this option cannot be disabled. This FIFO buffers all high-speed transactions on the D-PHY data lanes, including the trail bits. For Soft D-PHY, this also includes the hs-zero bits.
Type	**PINGPONG**, QUEUE, SINGLE	Type of the FIFO implemented. This feature is available when *RX_FIFO Enable* is checked.
Implementation	**EBR**,LUT	Selects the memory implementation of the FIFO. This feature is available when *RX_FIFO Enable* is checked.
Depth	16, 32, 64, **128**, 256, 512, 1024, 2048, 4096	Selects the memory depth of the FIFO. This feature is available when *RX_FIFO Enable* is checked.
Number of Queue Entries	2, **4**, 8	Determines the amount of Queue Entries available for the RX_FIFO. This option is editable when *Type = "QUEUE"* and *RX_FIFO* is checked.
Configurable FIFO Read Delay	checked, **unchecked**	When this is checked, an input bus rxcsr_rxfifo_pktdly_i is available to adjust the delay before the contents of the RX_FIFO is read.
Default FIFO Read Delay	0–16384, **8**	Determines the amount of delay of the RX_FIFO. This value is valid when *Type = "PINGPONG" or Type = "SINGLE"*, and *RX_FIFO* is checked. When this parameter is set to 0, the entire hs-transaction is buffered before FIFO is read, including the trail bits. For Soft D-PHY, this includes the hs-zero bits. For non-zero values, this sets the number of cycles of clk_byte_fr_i before reading from the RX_FIFO starting from the deassertion of the fifo_empty signal.
Counter Width	1–14, **4**	When *Type = "QUEUE"*, this sets the width of the counter that tracks the number of write cycles per high-speed transaction, including trail bits (for Hard D-PHY) and hs-zero bits (for Soft D-PHY). For *Type = "PINGPONG"* or "SINGLE", this is the width of the packet delay counter. This value is editable when *Type = "QUEUE"* and *RX_FIFO* is checked. If the Type = "Queue", set to minimum of the ceiling value of log2(TOTAL_HS_BITS/(NUM_RX_LANE*NUM_RX_GEAR)). For example: Number of RX lane = 4

Attribute	Selectable Values	Description
		RX Gear = 8 Total HS bits including HS-ZERO and HS-TRAIL = 24000 The ceiling value is ceil(log2(24000/(4*8))) = 10
Clock Mode	SC,**DC**	Determines if the FIFO is implemented in Single Clock and Dual Clock. This value is editable when *Type = "SINGLE"* and *RX_FIFO* is checked. The clock mode is always *"DC" w*hen *D-PHY RX IP = "Hard D-PHY"*.
Misc Signals	checked, **unchecked**	When checked, this shows the fifo_empty and fifo_full status signals at the top level module.

3.3. Soft PHY Settings

Table 3.3. Soft PHY Settings Attributes

Attribute	Selectable Values	Description
Soft PHY		
Delay Mode	**Edge Clock Centered**, User Defined	"Edge Clock Centered" uses a predetermined static delay value of the delay cells. The value is used to center align the D-PHY data with respect to the D-PHY clock edges. "User Defined" allows for customization of the delay cell settings.
Coarse Delay	**0 ns** 0.8 ns 1.6 ns	This attribute is editable only when the Delay Mode is "User Defined". Selects between 0, 0.8, or 1.6 ns coarse delay.
Fine Delay	0-127	This attribute is editable only when the Delay Mode is "User Defined". Each step value adds 12.5 ps delay on the data lanes.

4. Signal Description

This section describes the D-PHY Rx IP ports.

4.1. D-PHY Rx Interface

Table 4.1. D-PHY Rx Port Description

Port Name	Direction	Mode/Configuration	Description
D-PHY Rx			
tx_rdy_i	In	—	Indicates that the module at the receiving end is ready to receive data from the CSI-2/DSI D-PHY Rx IP module.
ref_dt_i[5:0]	In	Available when LMMI is disabled and Enable Packet Parser is enabled.	Reference Data Type.
rxcsr_datsettlecyc_i [7:0]	In	Available when Configurable Data Settle Count is checked and LMMI is disabled	This controls the tHS-SETTLE protocol timing parameter. Check the t-HSZERO parameter of the D-PHY transmitter to ensure the tHS-SETTLE setting can properly detect the Start-of-Transmit pattern.
rxcsr_rxfifo_pktdly_i[15:0]	In	Available when Configurable RX_FIFO Read Delay is checked and LMMI is disabled	This input controls the delay before the contents of the RX_FIFO is read out. This is applicable for pingpong or single RX_FIFO types.
rxcsr_dropnull_i	In	Available when LMMI is disabled and Enable Packet Parser is enabled.	This signal is tied to 0 when it is not exposed. Drive this signal when it is exposed: • 1'b0 – Null and Blanking packets trigger an assertion of lp_en. Payload is also be transmitted out. The output signal lp_av_en stays low. • 1'b1 – Null and Blanking packets are ignored by the IP.
rxcsr_vcx_on_i	In	Available when CSI-2 mode is enabled, LMMI is disabled, and Enable Packet Parser is enabled.	This signal is tied to 0 when it is not exposed. Drive this signal when it is exposed: • 1'b0 – no extended virtual channel ID; uses 24-bit Hamming code. • 1'b1 – packet header ECC byte[7:6] is used as extended virtual channel ID; uses 26-bit Hamming code.
lp_d0_tx_en_i	—	Available in DSI Mode and CIL Bypass is checked	Active high. Enables low-power transmit back to the DSI host. Data lane 0 goes out of differential mode and switches to low-power signaling.
lp_d0_tx_p_i	—	Available in DSI Mode and CIL Bypass is checked	This is the transmit value for the low-power data lane 0 p-channel.
lp_d0_tx_n_i	—	Available in DSI Mode and CIL Bypass is checked	This is the transmit value for the low-power data lane 0 n-channel.
sync_rst_i	—	—	Active high, synchronized reset.
sync_clk_i	In	—	Low speed or oscillator clock.
ready_o	Out	—	Indicates the state of gddr_sync.
pll_lock_i	In	—	PLL lock indicator, if a PLL is used to generate a free-running byte clock. Set this to 1 if a PLL is not used. This is also used to reset rx_fifo and is connected to the start_i pin of gddr_sync.
reset_n_i	In	—	Active low asynchronous system reset.

Port Name	Direction	Mode/Configuration	Description
clk_lp_ctrl_i	In	—	Clocks the logic that detects the Rx D-PHY clock lane LP <-> HS transitions. The frequency of this clock is at least twice the low-power clock frequency of the Rx D-PHY clock. The clock period of clk_lp_ctrl_i is at most a half of TLPX to properly sample the LP to HS clock lane transitions. Do not drive this clock if the Rx clock mode is HS_ONLY.
clk_byte_fr_i	In	—	Continuously running byte clock. This is div8 (in Gear 16) or div4 (in Gear 8) of the input D-PHY clock. This also clocks the logic that detects the Rx D-PHY data lane transitions (lp_hs_ctrl_d0-3 modules). This is used by the word_align, lane_align, and capture_control modules. Payload output is also in this clock domain.
reset_lp_n_i	In	—	Low asserted reset for the nets in the clk_lp_hs_ctrl clock domain. The signal driving this port must be synchronized to the clk_lp_hs_ctrl.
reset_byte_fr_n_i	In	—	Low asserted reset for the nets in the clk_byte_fr clock domain. The signal driving this port must be synchronized to the clk_byte_fr.
pd_dphy_i	In	--	This powers down the hardened D-PHY block. This can be used as an asynchronous high asserted hard reset of the hardened D-PHY block. This signal must not be tied to 1'b0. Assert this signal in the beginning, and then de-assert after the VCC is stabilized. Alternatively, drive this signal with the AND of the global reset with the PLL lock signal.
clk_p_io, clk_n_io	In/Out	—	MIPI D-PHY clock lane (positive and negative signals).
d_p_io[*BUS_WIDTH* – 1:0], d_n_io[*BUS_WIDTH* – 1:0]	In/Out	—	MIPI D-PHY data lanes.
lp_p_rx_o[*BUS_WIDTH* – 1:0], lp_n_rx_o[*BUS_WIDTH* – 1:0]	Out	—	Low-power data lanes.
bd_o[31:0]	Out	Available when Enable Packet Parser is checked	Valid only for Rx gear 8. This is the D-PHY byte data.
clk_byte_o	Out	—	Byte clock generated from the D-PHY module based on the input D-PHY clock lane. This clock latches the internal parallel byte data from dphy_rx_wrap. This is div4 or div8 of the D-PHY clock lane frequency. This is only active when the data lanes are in high-speed mode.
clk_byte_hs_o	Out	—	Generated byte clock from the D-PHY module based on the input D-PHY clock lane, active only when the clock lanes are in high-speed mode. This clock is the same as the clk_byte_o when the D-PHY implementation is Soft D-PHY.
hs_sync_o	Out	—	This indicates the successful detection of the synchronization code ‘B8 in the data lanes. This signal asserts from the start of synchronization pattern ‘B8 up to the last data captured before detecting LP-11 state of any lane (for Soft D-PHY) or data lane 0 (for Hard D-PHY).
payload_en_o	Out	Available when Enable Packet Parser is checked	This signifies the arrival of valid payload data without the CRC.
payload_o[dw-1:0]	Out	Available when Enable Packet Parser is checked	This is the payload of a long packet, excluding the CRC bits, arranged the way it is received in the data lanes. The width of this bus is the number of gear bits multiplied by the number of lanes (RX_GEAR*NUMBER_OF_LANE).
payload_bytevld_o[7:0]	Out	Available when Enable	Each bit corresponds to a valid byte in the payload_o.

Port Name	Direction	Mode/Configuration	Description
		Packet Parser is checked	[0] – payload_o [7:0] is valid [1] – payload_o [15:8] is valid [2] – payload_o [23:16] is valid [3] – payload_o [31:24] is valid [4] – payload_o [39:32] is valid [5] – payload_o [47:40] is valid [6] – payload_o [55:48] is valid [7] – payload_o [63:56] is valid
payload_crc_o[15:0]	Out	Available when Enable Packet Parser is checked	Long packet footer CRC bytes.
payload_crcvld_o	Out	Available when Enable Packet Parser is checked	Indicates the payload_crc_o is valid when asserted.
crc_error_o	Out	Available when Enable CRC Check is checked	When asserted, it indicates the packet footer CRC bytes do not match the computed payload CRC bytes.
crc_check_o	Out	Available when Enable CRC Check is checked	Indicates the crc_error_o is valid when asserted.
ecc_1bit_error_o	Out	Available when Enable Packet Parser is checked. Unavailable for DSI mode.	Indicates that one bit error has been detected and corrected in the packet header when asserted.
ecc_2bit_error_o	Out	Available when Enable Packet Parser is checked. Unavailable for DSI mode.	Indicates that 2 or more errors have been detected in the packet header when asserted. Long packet payload is dropped.
ecc_byte_error_o	Out	Available when Enable Packet Parser is checked. Unavailable for DSI mode.	Indicates that a flipped bit has been detected in the packet header ECC byte when asserted. Long packet payload is propagated.
ecc_check_o	Out	Available when Enable Packet Parser is checked. Unavailable for DSI mode.	Indicates that ecc_1bit_error_o, ecc_2bit_error_o, and ecc_byte_error_o signals are valid when asserted.
dt_o[5:0]	Out	Available when Enable Packet Parser is checked	CSI-2 or DSI 6-bit data type field.
vc_o[1:0]	Out	Available when Enable Packet Parser is checked	2-bit virtual channel ID of the packet.
wc_o[15:0]	Out	Available when Enable Packet Parser is checked	16-bit Word Count field. This denotes the number of bytes in the payload of a long packet. In a short packet, this contains a 2-byte data. Enabled when packet formatter is valid.
ecc_o[7:0]	Out	Available when Enable Packet Parser is checked	This is the received 8-bit error correction code (ECC). The submodule does not detect or correct any ECC errors.
dt2_o[5:0]	Out	Available when Enable Packet Parser is checked	Same as dt_o but only if NUM_RX_LANE* RX_GEAR == 64.
vc2_o[1:0]	Out	Available when Enable Packet Parser is checked	Same as vc_o but only if NUM_RX_LANE* RX_GEAR == 64.
wc2_o[15:0]	Out	Available when Enable	Same as wc_o but only if NUM_RX_LANE* RX_GEAR == 64.

Port Name	Direction	Mode/Configuration	Description
		Packet Parser is checked	
ecc2_o[7:0]	Out	Available when Enable Packet Parser is checked	Same as ecc_o but only if NUM_RX_LANE* RX_GEAR == 64.
bd0_o[rx_gear-1:0]	Out	Available when Enable Packet Parser is unchecked	Parallel data from lane 0. This is 8-bit wide if the design is configured as Gear 8, or 16-bit wide if gearing is 16.
bd1_o[rx_gear-1:0]	Out	Available when Enable Packet Parser is unchecked	Parallel data from lane 1. This is 8-bit wide if the design is configured as Gear 8, or 16-bit wide if gearing is 16.
bd2_o[rx_gear-1:0]	Out	Available when Enable Packet Parser is unchecked	Parallel data from lane 2. This is 8-bit wide if the design is configured as Gear 8, or 16-bit wide if gearing is 16.
bd3_o[rx_gear-1:0]	Out	Available when Enable Packet Parser is unchecked	Parallel data from lane 3. This is 8-bit wide if the design is configured as Gear 8, or 16-bit wide if gearing is 16.
capture_en_o	Out	Available when Enable Packet Parser is unchecked	This functions as a valid signal for bd0_o to bd3_o. This is enabled from the start of synchronization pattern ‘B8 up to the last data captured before detecting LP-11 state of any lane (for Soft D-PHY) or data lane 0 (for Hard D-PHY). This is the same as hs_sync_o.
skewcal_det_o	Out	Available when Enable Deskew Calibration Detection is enabled and CIL Bypass is checked or unchecked	Indicates that skew calibration burst is detected.
skewcal_done_o	Out	Available when Enable Deskew Calibration Detection is enabled and CIL Bypass is unchecked	Indicates that skew calibration of RX is done.
dphy_cfg_num_lanes[1:0]	Out	Available when Enable Packet Parser is checked	Number of active lanes that are configured for the IP: 2’b00: 1 lane 2’b01: 2 lanes 3’b10: 3 lanes 4’b11: 4 lanes
dphy_rxdatawidth_hs[1:0]	Out	Available when Enable Packet Parser is checked	Number of gears that are configured for the IP: 2’b00: Gear 8 2’b01: Gear 16 3’b10: Reserved 4’b11: Reserved

***Note:**

- BUS_WIDTH – Number of D-PHY Lanes, 1 to 4 (available in the user interface)

4.2. LMMI Device Target Interface

Table 4.2. LMMI Device Target Interface Signals Description

Port Name	Direction	Mode/Configuration	Description
LMMI Device Target Interface			
lmmi_clk_i	In	Available when Enable LMMI Interface is checked	LMMI Interface clock.
lmmi_resetn_i	In	Available when Enable LMMI Interface is checked	Active low signal to reset the configuration registers.
lmmi_wdata_i[LDW[1] – 1:0]	In	Available when Enable LMMI Interface is checked	Start transaction.
lmmi_wr_rdn_i	In	Available when Enable LMMI Interface is checked	Write = HIGH, Read = LOW.
lmmi_offset_i[LOW[2] – 1:0]	In	Available when Enable LMMI Interface is checked	Register offset, starting at offset 0.
lmmi_request_i	In	Available when Enable LMMI Interface is checked	Write data.
lmmi_ready_o	Out	Available when Enable LMMI Interface is checked	Read data.
lmmi_rdata_o[LDW[1] – 1:0]	Out	Available when Enable LMMI Interface is checked	Read transaction is complete and lmmi_rdata_o[] contains valid data.
lmmi_rdata_valid_o	Out	Available when Enable LMMI Interface is checked	Ready to start a new transaction.

Notes:

1. LDW – LMMI Data Width
   - If CIL_BYPASS is unchecked, then LDW = 4
   - Otherwise LDW = 8
2. LOW – LMMI Offset Width
   - If CIL_BYPASS is unchecked, then LOW = 5
   - Otherwise LOW = 7

4.3. AXI4-Stream Device Transmitter Interface

Table 4.3. AXI4-Stream Device Transmitter Interface Signals Description

Port Name	Direction	Mode/Configuration	Description
AXI4-Stream Device Transmitter Interface			
axis_stready_i	In	Available when Enable AXI4-Stream Interface is checked	Indicates that the module at the receiving end is ready to receive data from the CSI-2/DSI D-PHY RX IP module.
axis_mtvalid_o	Out	Available when Enable AXI4-Stream Interface is checked	AXI4-Stream indicates that data to be transmitted is valid.
axis_mtdata_o[N-1:0]*	Out	Available when Enable AXI4-Stream Interface is checked	Payload data transmitting channel (byte data or packet data with virtual channel, data type, and word count).

***Notes:**

- N depends on configuration:
  - If Parser = ”OFF”, then N = NUM_Rx_LANES × RX_GEAR
  - If Parser = ”ON” AND NUM_Rx_LANES × RX_GEAR = 64, then N = NUM_Rx_LANES × Rx_GEAR + 64
  - If Parser = ”ON” AND NUM_Rx_LANES × RX_GEAR != 64, then N = NUM_Rx_LANES × Rx_GEAR + 32

4.4. RX FIFO Status Interface

Table 4.4. RX FIFO Status Interface Signals Description

Port Name	Direction	Mode/Configuration	Description
RX FIFO Status Signals (available when Misc Signals of RX_FIFO is checked)			
rxdatsyncfr_state_o[1:0]	Out	—	State Machine for reading data from FIFO. SINGLE Mode: 2’b00 – IDLE State 2’b01 – Read data from buffer instance 0 QUEUE Mode: 2’b00 – IDLE State 2’b01 – Read data from buffer instance 0 2’b11 – Read data done PINGPONG Mode: 2’b00 – IDLE State 2’b01 – Read data from buffer instance 0 2’b10 – Read data from buffer instance 1 2’b11 – Read data done
rxemptyfr0_o	Out	—	FIFO empty flag
rxfullfr0_o	Out	—	FIFO full of instance 0
rxfullfr1_o	Out	Only applicable in PINGPONG Mode	FIFO full of instance 1
rxque_curstate_o[1:0]	Out	Not applicable in SINGLE Mode	State Machine of RX Queue: 2’b00 – IDLE State 2’b01 – Pop entry from queue 2’b10 – Wait for read data from buffer is done 2’b11 – One delay cycle before Idle
rxque_empty_o	Out	Not applicable in SINGLE Mode	RX Queue empty flag
rxque_full_o	Out	Not applicable in SINGLE Mode	RX Queue full flag
fifo_dly_err	Out	Not applicable in QUEUE Mode	An error flag that indicates a write happened when there is still an outstanding transfer in the RX FIFO. This flag is cleared when a new HS transfer happens.
fifo_undflw_err	Out	—	An error flag that indicates a read happened while the FIFO is empty. This happens if the TX clock is faster than RX clock and there is not enough data in the FIFO. This flag is cleared when a new HS transfer happens. Increase the FIFO delay setting to give time for data to accumulate in the buffer.
fifo_ovflw_err	Out	—	An error flag that indicates a write happens while the FIFO is full. This happens if the TX cannot flush out the FIFO fast enough. This flag is cleared when a new HS transfer happens. Decrease the delay setting, increase the FIFO depth or both.

4.5. Miscellaneous Status Interface

Table 4.5. Miscellaneous Status Interface Signals Description

Port Name	Direction	Mode/Configuration	Description
Miscellaneous Status Signals (available when Enable Miscellaneous Status is checked)			
term_clk_en_o	Out	—	Active-high enable signal for the line termination of the D-PHY clock lane. This is asserted on detection of transition from LP-11 to LP-01 of the clock lane, and de-asserted upon detection of LP-11 after a high-speed mode.
term_d_en_o[*NUM_LANES*-1:0]	Out	Depends on the number of lanes chosen	Active-high enable signal for the line termination of the D-PHY clock lane. This is asserted on detection of transition from LP-11 to LP-01 of the lanes, and de-asserted upon detection of LP-11 after a high-speed mode.
hs_d_en_o	Out	—	Active-high high-speed mode enable signal for data lane d0. For Hard D-PHY IP, this signal is also used for HS mode enable for other data lanes.
cd_d0_o	Out	—	Contention detection indicator on lane 0.
cd_clk_o	Out	—	Contention detection indicator on clock lane.
lp_hs_state_clk_o[1:0]	Out	—	2-bit state encoding of the D-PHY clock controller: 2'b00 – Idle State 2'b01 – LP11 State 2'b10 – LP01 State 2'b11 – HS State
lp_hs_state_d_o[1:0]	Out	—	2-bit state encoding of the D-PHY data lane 0 controller: 2'b00 – Idle State 2'b01 – LP11 State 2'b10 – LP01 State 2'b11 – HS State

5. Register Description

All registers listed in Table 5.1, Table 5.2, and Table 5.3 are accessible through LMMI.

5.1. Register Address Map

Table 5.1. General Configuration Registers

Offset (6 bits)	Access Type	Register Name	Description
Available in Hard D-PHY			
0x00-0x1E	R/W	Hard D-PHY Registers	LMMI accessible Hard D-PHY control registers
Packet Parser Registers			
0x0A	RW	LANE_SETTING_ADDR	Number of Lane Select
0x1F	R	VC_DT_ADDR	Virtual channel and Data type register address
0x20	R	WCL_ADDR	Word count low register address
0x21	R	WCH_ADDR	Word count high register address
0x22	R	ECC_ADDR	ECC register address
0x27	RW	REFDT_ADDR	Reference data type
0x35	RW	CONTROL_ADDR	Parser Controls
Available for *Gear x16* with *NUM_RX_LANE x4*			
0x23	R	VC_DT_ADDR	Virtual channel 2 and Data type 2 register address
0x24	R	WC2L_ADDR	Word count 2 low register address
0x25	R	WC2H_ADDR	Word count 2 high register address
0x26	R	ECC2_ADDR	ECC 2 register address
Error Control and Status Registers			
0x28	W1C	ERROR_STATUS_ADDR	ECC and CRC error status address
0x29	RW	ERROR_STATUS_EN_ADDR	ECC and CRC error status enable address
0x30	R	CRC_BYTE_LOW_ADDR	Received payload CRC LSB address
0x31	R	CRC_BYTE_HIGH_ADDR	Received payload CRC MSB address
0x32	W1C	ERROR_CTRL_ADDR	Hard D-PHY control error address
0x33	W1C	ERROR_HS_SOT_ADDR	Hard D-PHY Start-of-Transmit error address
0x34	W1C	ERROR_HS_SOT_SYNC_ADDR	Hard D-PHY Start-of-Transmit synchronization error address
0x36	RW	NOCIL_RXFIFO_DEL	RX_FIFO read delay for Soft D-PHY or CIL_bypassed Hard D-PHY
0x37	RW	NOCIL_DSETTLE_LSB	Data Settle LSB register for Soft D-PHY or CIL_bypassed Hard D-PHY
0x38	RW	NOCIL_DSETTLE_MSB	Data Settle MSB register for Soft D-PHY or CIL_bypassed Hard D-PHY

5.2. General Configuration Registers

Table 5.2. Configuration Registers (MIPI Programmable Bits)

ADDR	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
0x1F	Header Packet Virtual Channel ID [1:0]		Header Packet Data Type [5:0]					
0x20	Header Packet Byte 1 (Word Count LSB) [7:0]							
0x21	Header Packet Byte 2 (Word Count MSB) [7:0]							
0x22	Header Packet Byte 3 (ECC Byte) [7:0]							
0x23	Header Packet 2 Virtual Channel ID [1:0]		Header Packet 2 Data Type [5:0]					
0x24	Header Packet 2 Byte 1 (Word Count LSB) [7:0]							
0x25	Header Packet 2 Byte 2 (Word Count MSB) [7:0]							
0x26	Header Packet 2 Byte 3 (ECC Byte) [7:0]							
0x27	unused		Reference Data Type. If the received packet data type is the same as the reference data type, the lp_av_en_i is asserted.					
0x28	unused				2-bit ECC error – more than 2 flipped bits encountered in the received packet header. Packet is dropped.	1-bit ECC error – a flipped bit is encountered in the received packet header and is corrected. Packet transmission continues.	ECC byte error – a flipped bit is encountered in the ECC byte. Packet transmission continues.	CRC error – a CRC error is encountered in the payload bits.
0x29	unused				2-bit ECC error enable – if this bit is 1, 2-bit ECC error is reported. Otherwise, bit[3] of the ERROR_STATUS register stays at 0.	1-bit ECC error enable – if this bit is 1, 1-bit ECC error is reported. Otherwise, bit[2] of the ERROR_STATUS register stays at 0.	ECC byte error enable – if this bit is 1, 1-bit ECC error is reported. Otherwise, bit[1] of the ERROR_STATUS register stays at 0.	CRC error enable – if this bit is 1, CRC error is reported. Otherwise, bit[0] of the ERROR_STATUS register stays at 0.
0x30	Payload CRC [7:0]							
0x31	Payload CRC [15:8]							
0x32	unused				This bit asserts when an incorrect state sequence is detected in Hard D-PHY data lane 3.	This bit asserts when an incorrect state sequence is detected in Hard D-PHY data lane 2.	This bit will assert when an incorrect state sequence is detected in Hard D-PHY data lane 1.	This bit asserts when an incorrect state sequence is detected in Hard D-PHY data lane 0.
0x33	unused				This bit asserts when a 1-bit Start-of-Transmit error is detected in	This bit asserts when a 1-bit Start-of-Transmit error is detected in Hard D-PHY	This bit asserts when a 1-bit Start-of-Transmit error is detected in Hard D-PHY	This bit asserts when a 1-bit Start-of-Transmit error is detected in Hard D-PHY

ADDR	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
					Hard D-PHY data lane 3.	data lane 2.	data lane 1.	data lane 0.
0x34	unused				This bit asserts when a Start-of-Transmit error is detected in Hard D-PHY data lane 3. Entire HS packet is dropped.	This bit asserts when a Start-of-Transmit error is detected in Hard D-PHY data lane 2. Packet is dropped.	This bit asserts when a Start-of-Transmit error is detected in Hard D-PHY data lane 1. Packet is dropped.	This bit asserts when a Start-of-Transmit error is detected in Hard D-PHY data lane 0. Packet is dropped.
0x35	unused						0- extended virtual channel disabled. 1- extended virtual channel enabled.	0 – blank and null packets are transmitted. 1- blank and null packets are dropped.
0x36	RX FIFO Read Delay [7:0] – This input controls the number of byte clocks (clk_byte_fr_i) before the contents of the RX_FIFO is read out. This is applicable for pingpong or single RX_FIFO types.							
0x37	Datasettlecyc[7:0] – LSB of the data settle cycle register used when the IP is configured as Soft D-PHY or CIL_bypassed Hard D-PHY. This is the number of clk_byte_o that the IP ignores for the HS data lanes after transitioning from LP mode (tHS-SETTLE protocol timing parameter). Check the t-HSZERO parameter of the D-PHY transmitter to ensure the tHS-SETTLE setting detects the Start-of-Transmit pattern.							
0x38	Datasettlecyc[15:8] – MSB of the data settle cycle register used when the IP is configured as Soft D-PHY or CIL_bypassed Hard D-PHY. This is the number of clk_byte_o that the IP ignores for the HS data lanes after transitioning from LP mode (tHS-SETTLE protocol timing parameter). Check the t-HSZERO parameter of the D-PHY transmitter to ensure the tHS-SETTLE setting detects the Start-of-Transmit pattern.							

5.3. Hard D-PHY Configuration Registers/Bits for Hard MIPI D-PHY IP

Table 5.3. Configuration Registers (MIPI Programmable Bits)

ADDR [5:0]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
0x00	HSEL RX High-Speed Select. [0] – Less than <= 1.5 Gbps [1] – Higher than 1.5 Gbps	AUTO_PD_EN Powers down inactive lanes. [0] – Lanes are kept powered up and at LP11. [1] – Lanes powered down.	PRIMARY_SECONDARY Selects the PHY IP configuration. [0] – Secondary [1] – Primary	DSI_CSI Selects the PHY IP application. [0] – CSI2 [1] – DSI
0x01	RXCDRP[1:0] LP-CD threshold voltage. Default is 2'b01. Min – 200 mV, Max – 450 mV		00*	
0x02	EN_CIL Enables or disables CIL. [0] – CIL bypassed [1] – CIL enabled	RXLPRP[2:0] Adjusts the threshold voltage and hysteresis of LP-RX, default setting is 2'b001.		
0x03	010*			DESKEW_EN Enables Deskew feature affects ERRSYNC/NOSYNC. [0] – Deskew disabled

ADDR [5:0]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
				[1] – Deskew enabled
0x09	Lane0_sel[0] (See MSB below at 0x0A)	RxDataWidthHS[1:0] High-Speed Receive Data Width Select. 2'b00 – 1/8 the HS bit rate 2'b01 – 1/16 the HS bit rate 2'b10 – 1/32 the HS bit rate		0*
0x0A	uc_PRG_RXHS_SETTLE[0] (See MSB below at 0x0B)	cfg_num_lanes[1:0] Sets the number of active lanes. Value from 0 to 3.		Lane0_sel[1] This determines which lane acts as data lane 0 in HS Operation mode. Value from 0 to 3.
0x0B	uc_PRG_RXHS_SETTLE[4:1] Bits used to program T_HS_SETTLE. For *clock* pin. *T_HS_SETTLE* = (*uc_PRG_RXHS_SETTLE* + 1) × (Tperiod of sync_clk_i)			
0x0C	000*			uc_PRG_RXHS_SETTLE[5] (See LSB above at 0x0B)
0x0F	u_PRG_RXHS_SETTLE[1:0] (See MSB below at 0x10)		00*	
0x10	u_PRG_RXHS_SETTLE[5:2] Bits used to program T_HS_SETTLE. For *data* pins. *T_HS_SETTLE* = (*uc_PRG_RXHS_SETTLE* + 1) × (Tperiod of sync_clk_i)			
0x14	TEST_ENBL[1:0] (See MSB below at 0x15)		00*	
0x15	TEST_ENBL[5:2] Six-bit signal that enables the testing modes.			
0x16	TEST_PATTERN[3:0]			
0x17	TEST_PATTERN[7:4]			
0x18	TEST_PATTERN[11:8]			
0x19	TEST_PATTERN[15:12]			
0x1A	TEST_PATTERN[19:16]			
0x1B	TEST_PATTERN[23:20]			
0x1C	TEST_PATTERN[27:24]			
0x1D	TEST_PATTERN[31:28] TEST_PATTERN[31:0] is the programmable pattern used by the BIST pattern generator and pattern matcher.			
0x1E	000*			cont_clk_mode Enables the clock lane of the secondary device to maintain HS reception state during continuous clock mode operation. [0] – Disabled [1] – Enabled

***Note**: These bits must be set to the indicated value when writing to this register. Otherwise, the IP may malfunction.

6. Designing with the IP

This section provides information on how to generate the IP Core using the Lattice Radiant software and how to run simulation and synthesis. For more details on the Lattice Radiant software, refer to the Lattice Radiant Software User Guide.

6.1. Generating and Instantiating the IP

You can use the Lattice Radiant software to generate IP modules and integrate them into the device's architecture. To generate the D-PHY Rx IP in the Lattice Radiant software, follow these steps:

1. Create a new Lattice Radiant software project or open an existing project.
2. In the **IP Catalog** tab, double-click **CSI-2/DSI D-PHY Receiver** under **IP, Audio_Video_and_Image_Processing** category. The **Module/IP Block Wizard** opens as shown in Figure 6.1. Enter values in the **Component name** and the **Create in** fields and click **Next**.

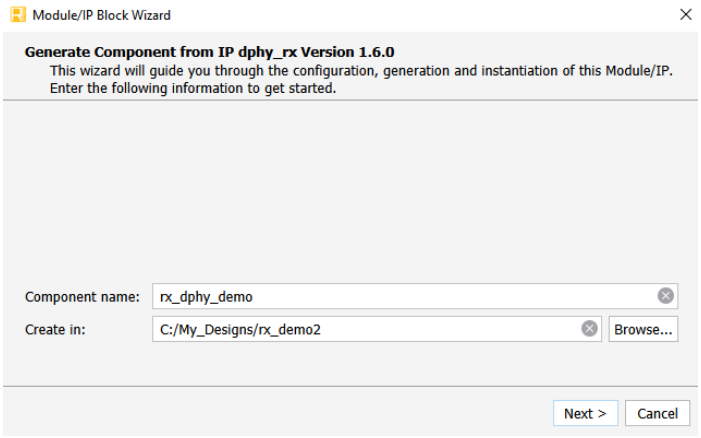

Figure 6.1. Module/IP Block Wizard

3. In the next **Module/IP Block Wizard** window, customize the selected CSI-2/DSI D-PHY Receiver IP using drop-down lists and check boxes. Figure 6.2 shows an example configuration of the CSI-2/DSI D-PHY Receiver IP. For details on the configuration options, refer to the IP Parameter Description section.

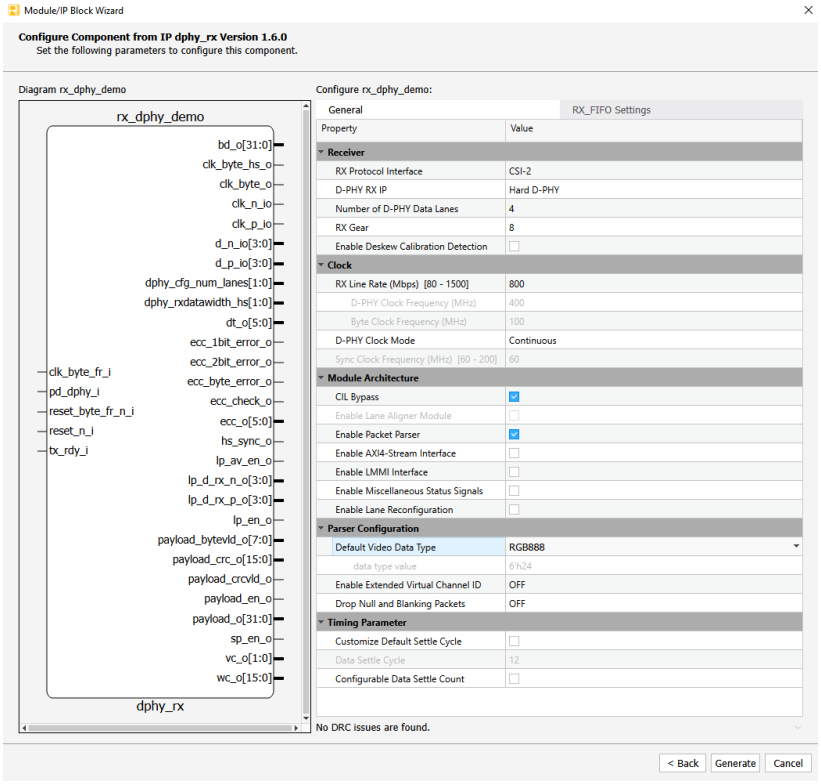

Figure 6.2. IP Configuration

4. Click **Generate**. The **Check Generated Result** dialog box opens, showing design block messages and results as shown in Figure 6.3.

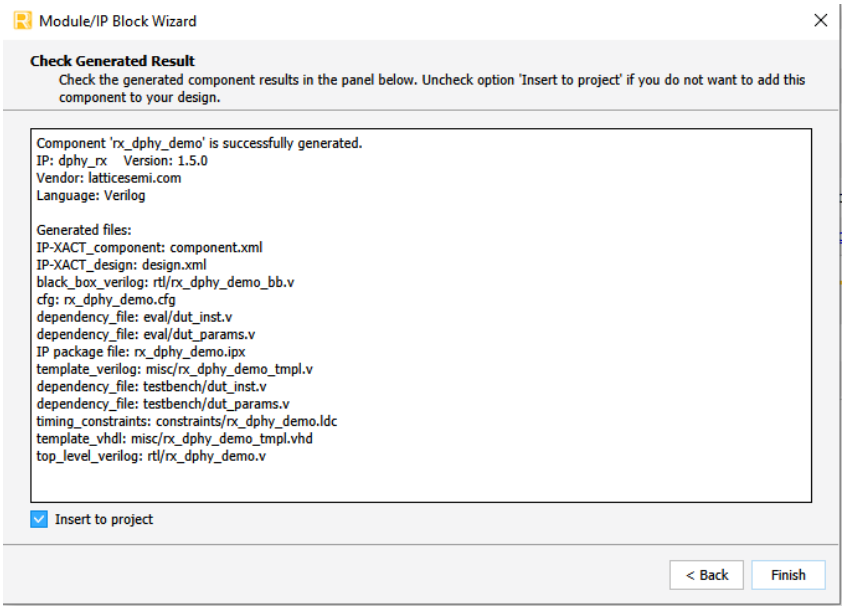

Figure 6.3. Check Generated Result

5. Click **Finish**. All the generated files are placed under the directory paths in the **Create in** and the **Component name** fields shown in Figure 6.1.

6.1.1. Generated Files and File Structure

The generated CSI-2/DSI D-PHY Receiver module package includes the closed-box (<Component name>_bb.v) and instance templates (<Component name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Component name>.v) that can be used as an instantiation template for the module is also provided. You may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in Table 6.1.

Table 6.1. Generated File List

Attribute	Description
<Component name>.ipx	This file contains the information on the files associated to the generated IP.
<Component name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/<Component name>.v	This file provides an example RTL top file that instantiates the module.
rtl/<Component name>_bb.v	This file provides the synthesis closed-box.
misc/<Component name>_tmpl.v misc /<Component name>_tmpl.vhd	These files provide instance templates for the module.

6.2. Design Implementation

Completing your design includes additional steps to specify analog properties, pin assignments, and timing and physical constraints. You can add and edit the constraints using the Device Constraint Editor or by manually creating a PDC File.

Post-Synthesis constraint files (.pdc) contain both timing and non-timing constraint .pdc source files for storing logical timing/physical constraints. Constraints that are added using the Device Constraint Editor are saved to the active .pdc file. The active post-synthesis design constraint file is then used as input for post-synthesis processes.

Refer to the relevant sections in the Lattice Radiant Software User Guide for more information on how to create or edit constraints and how to use the Device Constraint Editor.

6.3. Timing Constraints

Refer to the generated constraints file *<ip_instance_path>/constraints/<instance_name>.ldc* for details on how to constrain the IP. For the sample .pdc, refer to the file *<ip_instance_path>/eval/post-syn_constraints.pdc*.

Refer to Lattice Radiant Timing Constraints Methodology for details on how to constrain your design.

6.4. Physical Constraints

For Soft D-PHY mode, define the MIPI pins (clk_p_i, d_p_io_*) with IO_TYPE of MIPI_DPHY in the .pdc file. An example is shown in Figure 6.4.

```
284 ldc_set_port -iobuf {IO_TYPE=MIPI_DPHY} [get_ports clk_p_i]
285 ldc_set_port -iobuf {IO_TYPE=MIPI_DPHY} [get_ports {d_p_io[*]}]
```

Figure 6.4. Defining MIPI DPHY Port Pins Using MIPI_DPHY IO_TYPE

6.5. Specifying the Strategy

The Lattice Radiant software provides two predefined strategies: Area and Timing. The software also enables you to create customized strategies. For details on how to create a new strategy, refer to the Strategies section of the Lattice Radiant Software user guide.

6.6. Running Functional Simulation

You can run functional simulation after the IP is generated.

To run functional simulation, follow these steps:

1. Click the  button located on the **Toolbar** to initiate the **Simulation Wizard** shown in Figure 6.5.

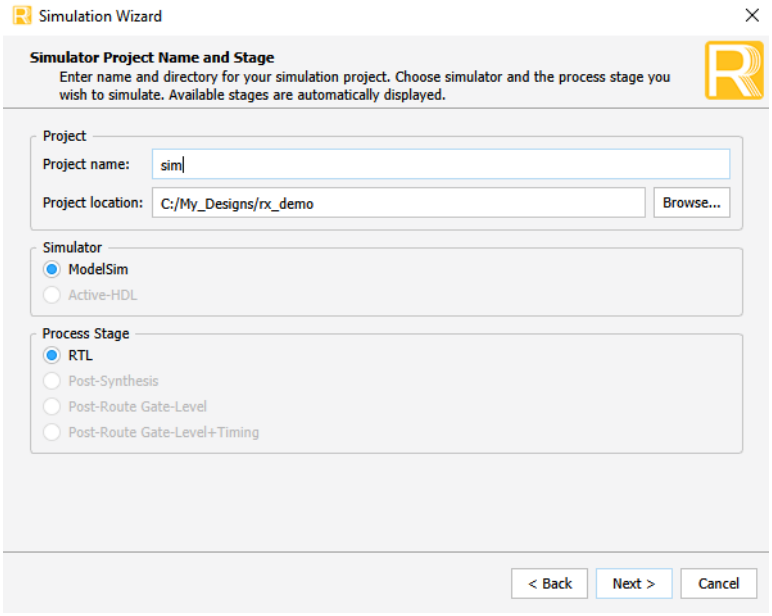

Figure 6.5. Simulation Wizard

2. Click **Next** to open the **Add and Reorder Source** window as shown in Figure 6.6.

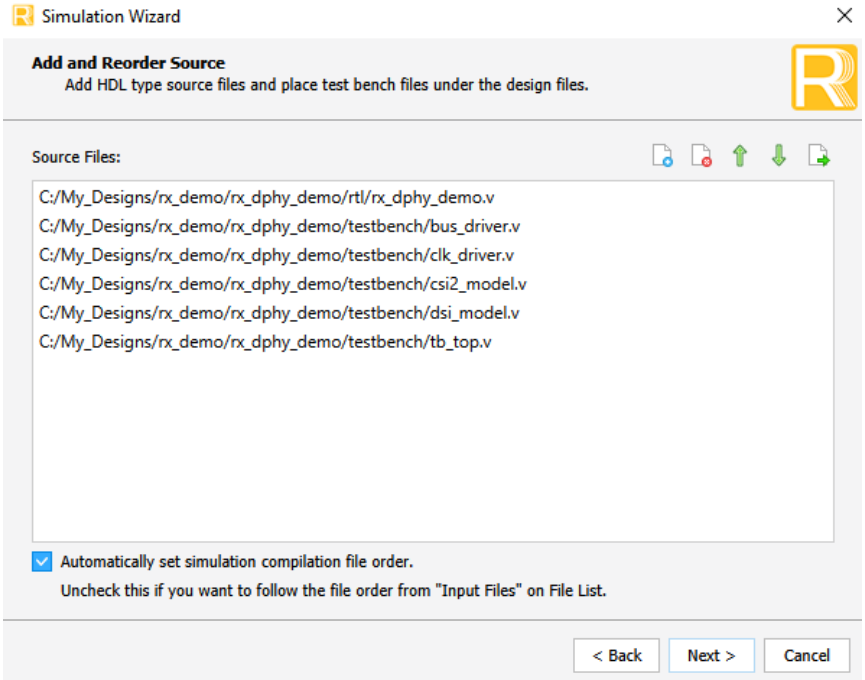

Figure 6.6. Add and Reorder Source

3. Click **Next**. The **Summary** window opens.
4. Set **Run Simulation** to 0 to ensure the simulation runs completely. Click **Finish** to run the simulation.

The waveform in Figure 6.7 shows an example simulation result.

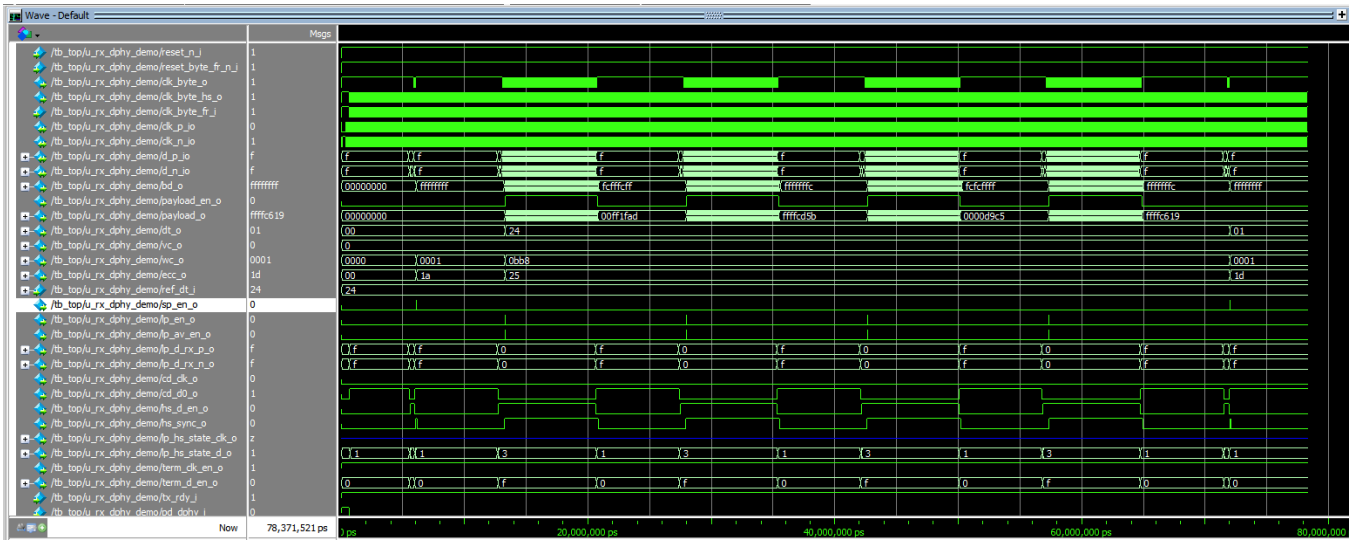

Figure 6.7. Simulation Waveform

6.6.1. Simulation Results

When the simulation is complete, the output in the Transcript window is shown in Figure 6.8.

```
# Start of data comparing
# End of data comparing
# ***PAYLOAD DATA PASS***
# **SIMULATION PASSED**
# Test end
# ** Note: $stop    : C:/My_Designs/rx_demo/rx_dphy_demo/testbench/tb_top.v(1039)
#    Time: 78371521 ps  Iteration: 0  Region: /tb_top/genblk6
# Break in Task test_end at C:/My_Designs/rx_demo/rx_dphy_demo/testbench/tb_top.v line 1039

VSIM 7>
```

Figure 6.8. Passing Simulation Log

If your simulation failed, ensure that the reset signals and clock signals are set up as described in the Functional Description section. You can also enable Miscellaneous status signals and FIFO Misc signals to debug the functional simulation.

7. Debugging

This section lists the possible issues and the suggested troubleshooting steps.

7.1. Debug Methods

7.1.1. Geared Down D-PHY Clock (clk_byte_hs_o) Does Not Toggle

In non-continuous clock mode, clk_byte_hs_o toggles only during high-speed transmission on the MIPI D-PHY data lanes. In continuous clock mode, this output clock toggles continuously.

When this clock does not toggle as expected, check to ensure the upstream source (camera sensors, application processors, and so on) is alive. Verify that the upstream device configuration matches with the IP parameters settings (for example, clock mode, bit rates). Set up all the input clocks and reset signals going into the IP module as recommended.

7.1.2. Corrupted Output Data Packet

If you suspect corruption on the output data packets, follow these guidelines:

- Set the Data Settle Cycle per the range and equation in Table 3.1.
- Check the packet headers against the transmitted packet.
  - Example: Check if the word count in the packet header matches the number of clock cycle of payload_en_o = high.
- Configure the RX FIFO as recommended. Ensure the FIFO is full and FIFO is empty flags are not asserted during the high-speed transaction.

7.2. Debug Tools

You can use the tool described in the subsection to debug the CSI-2/DSI D-PHY Receiver IP design issues.

7.2.1. Reveal Analyzer

The Reveal Analyzer continuously monitors signals within the FPGA for specific conditions that range from simple to complex conditions. When the trigger condition occurs, the Reveal Analyzer saves signal values preceding, during, and following the event for analysis, including a waveform presentation. The data can be saved in the following format:

- Value change dump file (.vcd) that can be used with tools such as ModelSim™.
- ASCII tabular format that can be used with tools such as Microsoft® Excel.

Before running the Reveal Analyzer, use the Reveal Inserter to add Reveal modules to your design. In these modules, specify the signals to monitor, define the trigger conditions, and set other preferred options. The Reveal Analyzer supports multiple logic analyzer cores using hard/soft JTAG interface. You can have up to 15 modules, typically one for each clock region of interest. When the modules are set up, regenerate the bitstream data file to program the FPGA.

During debug cycles, this tool uses a divide and conquer method to narrow down the problem areas into many small functional blocks to control and monitor the status of each block.

Refer to Reveal User Guide for Radiant Software for details on how to use the Reveal Analyzer.

Consider setting up triggers with the following signals:

- Start-of-Transmission pattern (SoTp) on the bd_o output pins.
- Transitions on the enable signals, such as payload_en_o, sp_en_o, lp_en_o, lp_en_av_o.
- Transitions on the Miscellaneous status flags and FIFO Misc flags.

8. Design Considerations

8.1. Design Considerations in Continuous Clock Mode

- Ensure the clock mode, bit rate, and number of lanes matches between the IP and the upstream devices.
- Synchronize the synchronized reset signals to the respective clock domains. Refer to the User Interfaces section.
- Ensure the Data Settle Cycle is within MIPI D-PHY specifications and IP range. Refer to the Data Settle Cycle attribute in Table 3.1.
- Drive clk_byte_fr_i clock pin from the clk_byte_hs_o clock pin. Refer to the User Interfaces section.
- Set the RX FIFO to OFF. Refer to the RX_FIFO OFF section.

8.2. Design Considerations in Non-Continuous Clock Mode

- Ensure the clock mode, bit rate, and number of lanes matches between the IP and the upstream devices.
- Synchronize the synchronized reset signals to the respective clock domains. Refer to the User Interfaces section.
- Ensure the Data Settle Cycle is within MIPI D-PHY specifications and IP range. Refer to the Data Settle Cycle attribute in Table 3.1.
- If possible, generate a clock either from a PLL or OSC to drive clk_byte_fr_i pin that matches the frequency of the clk_byte_hs clock. Refer to the User Interfaces section.
- If the clk_byte_fr and clk_byte_hs clock frequencies match, generate the minimum depth of a single RX FIFO with the smallest number of packet delay. Refer to the RX_FIFO_TYPE = SINGLE section.
- Determine the Low Power Blanking period from the upstream device and choose a suitable RX FIFO with the appropriate parameters. Refer to the RX_FIFO section.

8.3. Limitations

- Escape Mode, Ultra Low Power State (ULPS), and Bus Turnaround sequences are not yet supported.

Appendix A. Resource Utilization

Table A.1 and Table A.2 show the maximum frequency and resource utilization for a certain IP configuration.

Table A.1. Device and Tool Tested

—	Value
Software Version	Lattice Radiant 2023.1.1 production build
Device Used	LIFCL-40-9BG400C
Performance Grade	9_High-Performance_1.0V
Synthesis Tool	Synplify Pro (R) U-2023.03LR-1, Build 098R, May 29 2023

Table A.2. Resource Utilization[1]

Lane (Gear)	RX Interface Type	IP Type	Bit Rate (Lane)	Parser	AXI Bus	LMMI Bus	Registers	LUT[2]	EBR	High-Speed I/O Resources
4(8)	CSI-2	Hard D-PHY[4]	1000 Mbps	EN	EN	DIS	566	428	1	1 x Hard D-PHY
4(8)	CSI-2	Soft D-PHY	1000 Mbps	EN	EN	DIS	797	1081	5	4 x IDDRX4, 1 x ECLKDIV, 1 x ECLKSYNC
4(16)[3]	CSI-2	Hard D-PHY[4]	2500 Mbps	EN	EN	DIS	817	805	2	1 x Hard D-PHY
4(8)[3]	CSI-2	Soft D-PHY	1500 Mbps	EN	EN	DIS	797	1188	5	4 x IDDRX4, 1 x ECLKDIV, 1 x ECLKSYNC
4(16)	DSI	Hard D-PHY[4]	2000 Mbps	EN	EN	EN	978	1313	2	1 x Hard D-PHY
4(16)	DSI	Hard D-PHY[5]	2000 Mbps	EN	EN	EN	827	1337	4	1 x Hard D-PHY
4(8)	DSI	Hard D-PHY[4]	1200 Mbps	EN	EN	DIS	619	728	1	1 x Hard D-PHY
2 (8)	DSI	Hard D-PHY[4]	800 Mbps	EN	DIS	DIS	374	476	1	1 x Hard D-PHY
4(16)[3]	CSI-2	Hard D-PHY[4]	2500 Mbps	EN	EN	EN	838	837	2	1 x Hard D-PHY
4(8)[3]	CSI-2	Hard D-PHY[4]	1500 Mbps	EN	DIS	EN	519	578	1	1 x Hard D-PHY
2(8)	CSI-2	Hard D-PHY[4]	800 Mbps	EN	DIS	DIS	365	428	1	1 x Hard D-PHY

Notes:

1. All other settings are default.
2. The *distributed RAM* utilization is accounted for in the total LUT4s utilization. The actual LUT4 utilization is distribution among *logic*, *distributed RAM*, and *ripple logic*.
3. Data Settle settings change in these configurations to match the target bit rate per lane.
4. Hard D-PHY – CIL Bypassed
5. Hard D-PHY – CIL Enabled

For more information regarding a specific configuration, generate the IP, run synthesis and MAP, and check the MAP reports for resource utilization.

References

- CrossLink-NX web page
- Certus-NX web page
- CertusPro-NX web page
- MachXO5-NX web page
- Avant-E web page
- Avant-G web page
- Avant-X web page
- Lattice Radiant Software web page
- Lattice Insights for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 2.1, January 2024

Section	Change Summary
All	Renamed document from *CSI-2/DSI D-PHY Rx IP Core - Lattice Radiant Software* to *CSI-2/DSI D-PHY Rx IP*.
Disclaimers	Updated disclaimers.
Introduction	• Reworked section contents. • Changed *LAV-AT-500E* to *LAV-AT-E70* in Table 1.1. • Reworked section 7 *Ordering Part Number* and renamed to subsection 1.4 Licensing and Ordering Information. • Reworked subsection 6.4 *IP Evaluation* and renamed to subsection 1.5 IP Validation Summary. • Added subsection 1.6 Minimum Device Requirements. • Reworked subsection 1.5 *Conventions* and renamed to subsection 1.7 Naming Conventions.
Functional Description	• Reworked section 2 *Functional Description* and renamed to subsection 2.1 IP Architecture Overview. • Added subsection 2.2 User Interfaces. • Reworked subsection 2.10 *LMMI Device* and renamed to subsection 2.2.1 LMMI Device Target Interface. • Reworked subsection 2.9 *AXI4-Stream Device Transmitter* and renamed to subsection 2.2.2 AXI4-Stream Device Transmitter Interface. • Reworked subsection 2.2.1 *Hard D-PHY* and moved to subsection 2.4.1 Hard D-PHY. • Reworked subsection 2.3 *RX_FIFO* and moved to subsection 2.5 RX_FIFO. • Reworked subsection 2.4 *Global Operations Controller* and moved to subsection 2.6 Global Operations Controller. • Reworked subsection 2.5 *D-PHY Rx IP without Packet Parser* and moved to subsection 2.7 D-PHY Rx IP without Packet Parser. • Reworked subsection 2.7 *Packet Parser* and moved to subsection 2.9 Packet Parser. • Reworked subsection 2.8 *Word Aligner and Optional Lane Aligner* and moved to subsection 2.10 Word Aligner and Optional Lane Aligner. • Added subsection 2.11 Lane/Gear Dynamic Reconfiguration with CRC Checking Function.
IP Parameter Description	Reworked section 4 *Attribute Summary* and renamed to section 3 IP Parameter Description.
Signal Description	Reworked section 3 *Signal Description* and moved to section 4 Signal Description.
Register Description	Reworked section 5 *Internal Registers* and renamed to section 5 Register Description.
Designing with the IP	• Reworked section 6 *IP Generation, Simulation, and Validation* and renamed to section 6 Designing with the IP. • Reworked subsection 6.1 *Generating the IP* and renamed to subsection 6.1 Generating and Instantiating the IP. • Added subsection 6.2 Design Implementation. • Reworked subsection 6.3 *Constraining the IP* and renamed to *subsection* 6.3 Timing Constraints and 6.4 Physical Constraints. • Added subsection 6.5 Specifying the Strategy. • Reworked subsection 6.2 *Running Functional Simulation* and moved to subsection 6.6 Running Functional Simulation.
Debugging	Added this section.
Design Considerations	Added this section.
Resource Utilization	Updated for the latest software version.
References	Reworked section contents.

Revision 2.0, August 2023

Section	Change Summary
Inclusive Language	Newly added this section.
All	• Changed all instances of *Device Master* to *Device Transmitter*. • Changed all instances of *Device Slave* to *Device Target*.
Introduction	Table 1.1. CSI-2/DSI D-PHY Rx IP Core Quick Facts: • newly added *MachXO5-NX* to Supported FPGA Families; • newly added *LIFCL-33*, *LFCPNX-50*, *LFMXO5-25*, *LFMXO5-55T*, and *LFMXO5-100T* to Targeted Devices; • newly added *IP Core v1.5.x - Lattice Radiant software 2022.1* to Lattice Implementation.
Signal Description	Changed *AXI4 Master Stream* to *AXI4 Stream Transmitter* in Table 3.1. D-PHY Rx Port Description.

Functional Description	• In the D-PHY Module section:
	  • divided the contents of this section into Hard D-PHY and Soft D-PHY;
	  • updated the descriptions of hard D-PHY and soft D-PHY.
	• Changed *Init Slave* to *Init* in Figure 2.6. MIPI D-PHY Rx LP to HS Transition Flow Diagram on Data Lanes and Figure 2.7. MIPI D-PHY Rx LP to HS transition on Clock Lane.

Attribute Summary	Table 4.1. Attributes Table: • newly added the following attributes under soft PHY: *Delay Mode*, *Coarse Delay*, and *Fine Delay*; • newly added the following table note: *In the IP version 1.6.0 onwards, a GUI tab for modifying the delay cell within the soft PHY is available*.

Internal Registers	• Updated the Offset column of Table 5.1. General Configuration Registers.
	• In Table 5.2. Configuration Registers (MIPI Programmable Bits):
	  • changed *MASTER_SLAVE* to *PRIMARY_SECONDARY*;
	  • changed *Slave* to *Secondary*;
	  • changed *Master* to *Primary*;
	  • changed *slave clock lane* to *clock lane of the secondary device*.

Ordering Part Number	Removed the following OPNs: D-PHY-TX-CTNX-US, D-PHY-TX-CNX-U, D-PHY-TX-CNX-UT, D-PHY-TX-CNX-US, D-PHY-TX-CTNX-U, D-PHY-TX-CTNX-UT, D-PHY-TX-CPNX-U, D-PHY-TX-CPNX-UT, D-PHY-TX-CPNX-US, DPHY-TX-AVE-U, DPHY-TX-AVE-UT, DPHY-TX-AVE-US.
References	Newly added links to Lattice Radiant Software Web Page, CertusPro-NX Devices Web Page, Certus-NX Devices Web Page, CrossLink-NX Devices Web Page, MachXO5-NX Devices Web Page, Lattice Avant-E Devices Web Page, Lattice Insight for Training Series and Learning Plans.

Revision 1.9, December 2022

Section	Change Summary
Introduction	• Added Lattice Avant to Supported FPGA Family and added LAV-AT-500E to Targeted Devices in Table 1.1. CSI-2/DSI D-PHY Rx IP Core Quick Facts. • Added bullet *Hard D-PHY is supported on Crosslink-NX devices only* in Section 1.3. • Added bullets *Maximum rate of up to 1500 Mbps per lane for Crosslink-NX, Certus-NX, and CertusPro-NX devices* and *Maximum rate of up to 1800 Mbps per lane for Avant devices* in Section 1.4.
Functional Description	Updated below figures to remove *reset_byte_n_i*: • Figure 2.1. D-PHY Rx IP Block Diagram with AXI4-Stream Enabled, LMMI Enabled and Packet Parser OFF • Figure 2.2. D-PHY Rx IP Block Diagram with AXI4 Stream Enabled and LMMI Disabled • Figure 2.9. D-PHY Rx IP Configuration with AXI4-Stream Disabled and Packet Parser Disabled • Figure 2.11. D-PHY Rx IP Output Timing Diagram without Packet Parser • Figure 2.12. D-PHY Rx IP Configuration with AXI4-Stream Disabled and Packet Parser Enabled

Section	Change Summary
	• Figure 2.13. D-PHY Rx IP Configuration with AXI4-Stream Enabled and Packet Parser Enabled
Signal Description	• Added ports fifo_dly_err, fifo_undflw_err, and fifo_ovflw_err and deleted reset_byte_n_i in Table 3.1. D-PHY Rx Port Description.
Attribute Summary	• Added foot note *The maximum data rate depends on the gear, as well as the family, package and speed grade of the device. Please check the device data sheet for more information* to RX Line Rate per Lane (Mbps) in Table 4.1. Attributes Table.
IP Generation, Simulation and Validation	• Changed the title of section 6 from *Core Generation, Simulation and Validation* to *IP Generation, Simulation and Validation*. • Changed the title of section 6.1 from *Generation and Synthesis* to *Generating the IP*. • Changed the title of section 6.2 from *Functional Simulation* to *Running Functional Simulation*. • Added section 6.3. • Changed the title from Core Validation to IP Evaluation and updated description in section 6.4. • Deleted Licensing and Evaluation section.
Ordering Part Number	Added Avant-E OPNs.

Revision 1.8, September 2022

Section	Change Summary
Functional Description	Updated vc_o for Virtual Channel ID and dt_o for Data Type in Figure 2.17 in the AXI4-Stream Device Master section.

Revision 1.7, July 2022

Section	Change Summary
Functional Description	Updated the AXI4-Stream Device Master section. • Removed the payload_en_o signal from the axis_mtdata_ o group in In Figure 2.17. • Reordered the internal signals list.

Revision 1.6, September 2021

Section	Change Summary
Functional Description	Updated D-PHY Module section to add information on skew calibration.
Signal Description	Updated Table 3.1 to add rows for RX FIFO Status Signals, updated description for pd_dphy_i. lp_hs_state_clk_o[1:0] and lp_hs_state_d_o[1:0], and updated mode/configuration for skewcal_done_o.
Attribute Summary	Updated Table 4.1 to increase supported packet delay count and modify attribute descriptions for RX_FIFO.
Internal Registers	Updated Table 5.1 to move 0x39 to Always Available row.

Revision 1.5, June 2021

Section	Change Summary
Introduction	Updated section content, including Table 1.1 to include CertusPro-NX support.
Attribute Summary	Updated Table 4.1.
Licensing and Evaluation	Updated content in Licensing the IP and removed Hardware Evaluation section.
Ordering Part Number	Updated part number to include CertusPro-NX.

Revision 1.4, February 2021

Section	Change Summary
Introduction	• Removed ADC IP Core Native Interface from Table 1.1. • Updated MIPI D-PHY and the MIPI CSI-2 specifications to v1.2 in the Features section.

Revision 1.3, November 2020

Section	Change Summary
Introduction	• Updated Table 1.1. CSI-2/DSI D-PHY Rx IP Core Quick Facts. • Updated Lattice Implementation. • Updated reference to the Lattice Radiant Software User Guide. • Added support for optional periodic deskew detection to the Features section.
Functional Description	• Added contents to the RX_FIFO section. • Added skewcal_det_o and skewcal_done_o output ports to Figure 2.1, Figure 2.2, Figure 2.9, Figure 2.10, and Figure 2.12. • Updated Figure 2.6. • Updated heading to Word Aligner and Optional Lane Aligner. • Removed Figure 2.18.
Signal Description	• Added skewcal_det_o and skewcal_done_o output ports to Table 3.1. D-PHY Rx Port Description.
Attribute Summary	• Added Enable Deskew Calibration Detection to Table 4.1. Attributes Table.
Core Generation, Simulation, and Validation	• Updated reference to the Lattice Radiant Software User Guide • Updated Figure 6.1. Configure Block of D-PHY Rx. • Updated Figure 6.2. Check Generating Result.
References	Updated reference to the Lattice Radiant Software User Guide

Revision 1.2, August 2020

Section	Change Summary
Acronyms in This Document	Updated content.
Introduction	• Updated Table 1.1. • Updated content of Hard CSI-2/DSI D-PHY Rx IP Core Features and Soft CSI-2/DSI D-PHY Rx IP Core Features section.
Functional Description	• Updated content of D-PHY Rx Common Interface Wrapper, RX_FIFO, Global Operations Controller, Word Aligner and Optional Line Aligner, AXI4-Stream Device Master, and LMMI Device section. • Updated the following figures: • Figure 2.1 • Figure 2.2 • Figure 2.9 • Figure 2.10 • Figure 2.12 • Figure 2.13 • Figure 2.14 • Figure 2.15 • Figure 2.16
Signal Description	Updated Table 3.1.
Attribute Summary	Updated Table 4.1.
Internal Registers	Updated Table 5.2.
Core Generation, Simulation, and Validation	Updated step 1 in Functional Simulation section.
Ordering Part Number	Updated section content.

Section	Change Summary
Appendix A. Resource Utilization	Updated Table A.2.

Revision 1.1, February 2020

Section	Change Summary
Attribute Summary	Updated Table 4.1. Attributes Table.
Port Description	Updated Table 3.1. D-PHY Rx Port Description.
Functional Description	Added RX_FIFO section.

Revision 1.0, December 2019

Section	Change Summary
All	Initial release.

www.latticesemi.com