MIPI D-PHY Module

IP Version: v1.9.0

User Guide

FPGA-IPUG-02061-1.9

December 2025

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 REDUNDANCIES, 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
CIL	Control Interface Logic
CSI	Camera Serial Interface
DSI	Display Serial Interface
DUT	Device Under Test
EBR	Embedded Block RAM
FPGA	Field-Programmable Gate Array
GDDR	Graphics Double Data Rate
GSR	Global Set Reset
HDL	Hardware Description Language
HS	High Speed
I/O	Input/Output
IP	Intellectual Property
LMMI	Lattice Memory Mapped Interface
LP	Low Power
LSE	Lattice Synthesis Engine
LUT	Look-Up Table
PDC	Physical Design Constraint
PHY	Physical Layer
PLL	Phase-locked Loop
PPI	PHY-Protocol Interface
RAM	Random Access Memory
RX	Receiver
TX	Transmitter
UI	Unit Interval
ULPS	Ultra Low Power State

1. Introduction

1.1. Overview

MIPI D-PHY bus is a physical serial data communication layer on which the protocols like CSI-2 (Camera Serial Interface 2), DSI (Display Serial Interface) runs. The MIPI D-PHY bus physically connects the camera sensor to the application processor (for CSI-2) and application processor to the display device (for DSI) as shown in the Figure 1.1.

The Lattice Semiconductor MIPI D-PHY Module IP incorporates one clock lane and configurable number of data transmission lanes. The MIPI D-PHY Module IP supports 1, 2, and 4 data lanes.

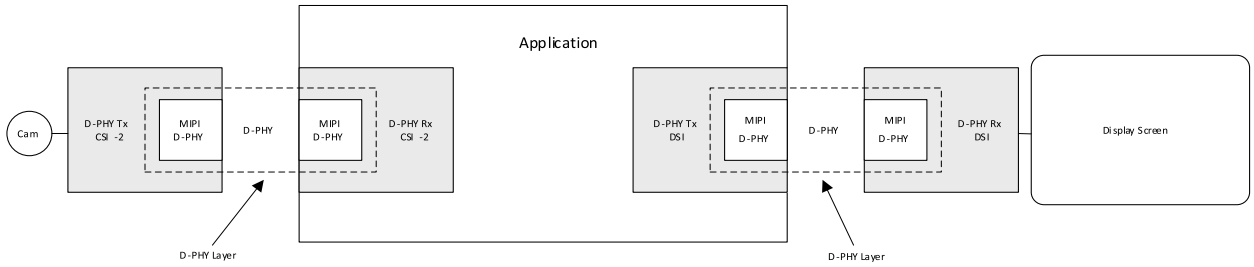

Figure 1.1. MIPI D-PHY Module IP

Every data lane of the transmitter and receiver consists of two wires (differential pair or two single-ended): data_p_io and data_n_io. The clock lane consists of clk_p_io and clk_n_io (differential pair or two single-ended). Data transmission occurs on these paired wires connecting receiver and transmitter communicating modules.

1.2. Quick Facts

Table 1.1 presents a summary of the MIPI D-PHY Module IP.

Table 1.1. Summary of the MIPI D-PHY Module IP

IP Requirements	Supported Devices	MachXO5™-NX, CrossLink™-NX, CertusPro™-NX, Certus™-NX
	IP Changes[1]	Refer to the MIPI D-PHY Module Release Notes (FPGA-RN-02105).
Resource Utilization	Resources	See the Resource Utilization section
Design Tool Support	Lattice Implementation	IP Core v1.9.0 – Lattice Radiant™ Software 2025.2
	Synthesis	Lattice Synthesis Engine (LSE)
		Synopsys® Synplify Pro® for Lattice
	Simulation	For a list of supported simulators, see the Lattice Radiant software user guide.

Note:

1. In some instances, the IP may be updated without changes to the user guide. This user guide may reflect an earlier IP version but remains fully compatible with the later IP version. Refer to the IP Release Notes for the latest updates.

1.3. Features

General features for both Hard and Soft MIPI D-PHY Module IP:

- Interfaces to MIPI CSI-2/DSI, RX and TX devices
- Supports unidirectional HS (high-speed) operation mode
- Supports bidirectional LP (low power) operation modes
- Deserializes and serializes HS data into byte data packets
- Provides methods for contention detection and termination switching
- Supports 1, 2, 4 data lanes and one clock lane
- Configurable RX and TX

- External reference clock source

Hard MIPI D-PHY:

- Maximum rate is up to 2500 Mbps per lane
- Supported gearing: 8x, 16x
- Configurable via LMMI PLL settings
- Bypass CIL mode
- Continues/non-continuous D-PHY clock mode
- Reference clock frequency to MIPI PLL 24 MHz to 200 MHz
- Interface clock frequency 40 MHz to 1250 MHz

Soft MIPI D-PHY:

- Supported rate is up to 1500 Mbps per lane
- Supported gearing: 8x
- *MIPI_DPHY* I/O type
- Interface clock frequency 40 MHz to 750 MHz

1.4. Licensing and Ordering Information

The MIPI D-PHY Module IP is provided at no additional cost with the Lattice Radiant software.

1.5. IP Validation Summary

Table 1.2 shows the validation status for the MIPI D-PHY Module IP core. The ✓ mark indicates whether the IP has been validated for Simulation, Timing, or with Hardware.

Table 1.2. IP Validation Level

Device Family	IP Version	Validation Level		
		Simulation	Timing	Hardware
Lattice Nexus™	1.9.0	✓	✓	—

1.6. Minimum Device Requirements

Refer to Table A.1 for the minimum required resource to instantiate this IP.

1.7. Conventions

1.7.1. Nomenclature

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

1.7.2. Signal Names

Signal names that end with:

- *_n* are active low
- *_i* are input signals
- *_o* are output signals
- *_io* are bi-directional input/output signals

2. Functional Description

2.1. Overview

Depending on the configuration set from Attribute Table (see Table 3.1), the internal resources of MIPI D-PHY Module IP allow you to configure it either as a hard MIPI D-PHY TX, a hard MIPI D-PHY RX, a soft MIPI D-PHY TX, or a soft MIPI D-PHY RX (see Figure 2.1).

Regardless of the implementation, the MIPI D-PHY Module IP provides the same external interface.

The generated MIPI D-PHY TX/RX Module wrapper is used to make a connection between the hard D-PHY Module and higher protocol layers.

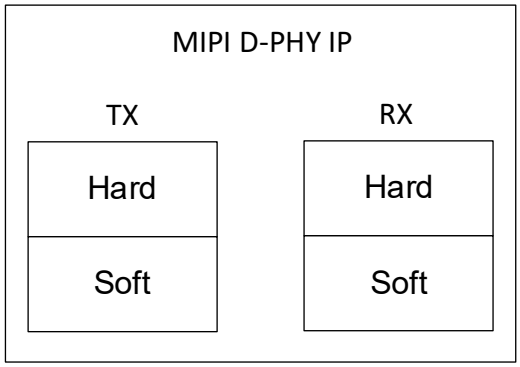

Figure 2.1. MIPI D-PHY Module

Each lane of MIPI D-PHY Module includes both the HS and LP modes.

The HS modules operate in a differential manner. The modules utilize the low voltage swing of the payload data signals to transfer the information. The module contains an on-die termination between Dp and Dn. The low power module, an unterminated module, operate in single-ended manner.

For transmitting the payload data (image data), all type of MIPI D-PHY Modules use the high-speed modules. For transmitting the control and status information, all type of MIPI D-PHY IPs use the low power modules utilizing low frequency signals.

The bandwidth can be increased by increasing the number of data lanes. By increasing the number of lanes, the same quantity of data can be transmitted on multiple lanes in lesser time. MIPI D-PHY Module uses forward source synchronous clock, which is used by all the data lanes of MIPI D-PHY receiver for capturing the high-speed data signals.

The Hard MIPI D-PHY represents Universal Lane Module, which incorporates D-PHY Lane I/O Buffers Module on the lane side and Control Interface Logic (CIL) as shown in Figure 2.2.

D-PHY Lane I/O Buffers Module includes high-speed differential transmitter (HS-TX), high-speed differential receiver (HS-RX), low power single-ended transmitter (LP-TX), low power single-ended receiver (LP-RX), and low power contention detector (LP-CD).

Control Interface Logic controls the operation of D-PHY Lane I/O Buffers Module and provides the PHY-Protocol Interface (PPI) logic, which includes set of signals to cover the functionality of the lane and interface to the protocol level side. The CIL functions depend on the lane type and lane side.

The HS-TX and HS-RX within a single Lane Module must not be enabled simultaneously during normal operation.

The LP-CD function is required for bi-directional operation. The LP-CD function is enabled to detect contention situations while the LP-TX is driving Low-Power states. The LP-CD checks for contention before driving a new state on the line.

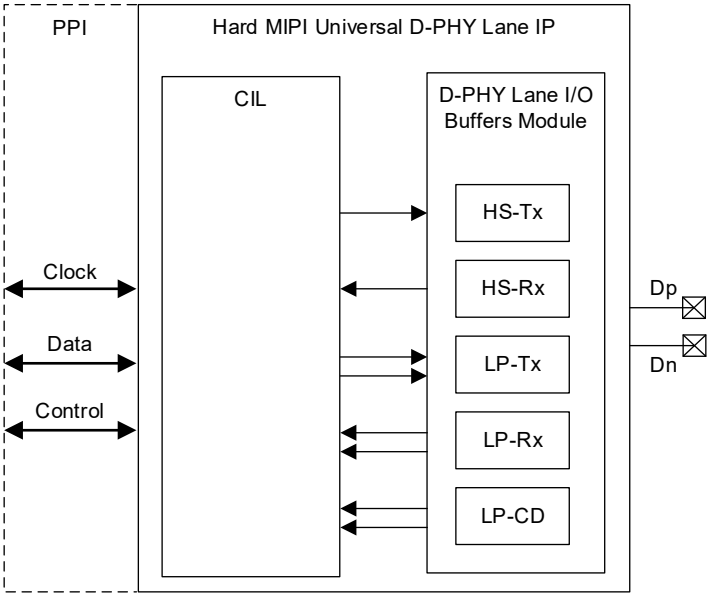

Figure 2.2. Hard MIPI D-PHY IP Universal Lane

2.1.1. Operation Mode

The Global Operation Module controls HS request path and timing using attributes in Table 3.1.

This module controls the timing entering HS and coming from HS entering to LP. Figure 2.3 shows the LP-to-HS transition flow diagram for data lanes.

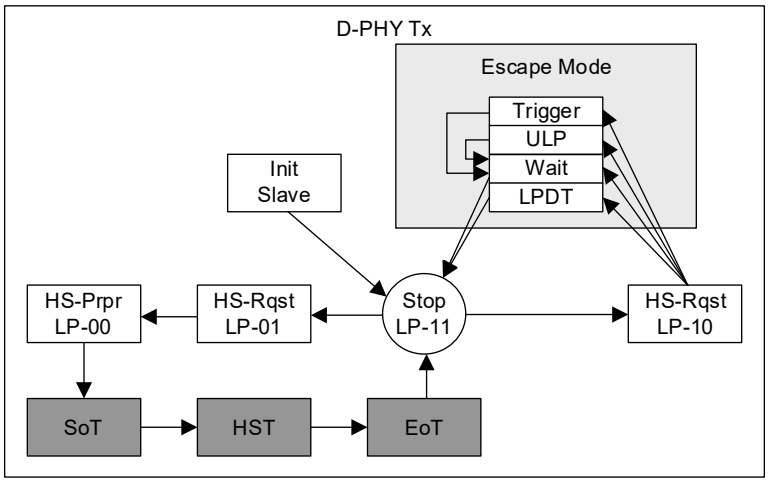

Figure 2.3. MIPI D-PHY TX LP to HS Transition Flow Diagram on Data Lanes

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

For sending payload data (the image data), the transmitter drives a sequence on data lanes to enter the receiver from the low power mode to high-speed mode.

As part of the initialization of D-PHY, initially all the lanes are held at LP11 state for a specified time. This LP11 state is also known as stop state. After this, for sending the image data, the transmitter drives a sequence on the receiver to enter the receiver lanes from low power mode to high-speed mode. The high-speed entry sequence, see Figure 2.4, consists of driving LP11->LP01->LP00 (LP->HS transition) on the differential lane. On successful reception of this sequence, the high-speed receiver module enables the termination to receive the high-speed differential data.

After LP-to-HS transition, the transmitter sends HS Zeros (V(Dn)>V(Dp)) for a specified amount of time to make sure that the receiver is enabled properly before any payload data is transmitted.

Before the payload data of every HS burst on each lane, the transmitting D-PHY inserts a sync sequence (00011101). This sync sequence is used by the data lanes of the receiving D-PHY to establish synchronization with the high-speed payload data.

The LP11 state brings back the data lane from high-speed mode to low power mode.

After every HS burst, the data lanes go to LP11 state. A single HS burst represents the image data corresponding to one of the horizontal lines of an image and the LP11 state in-between the HS bursts represents the blanking periods.

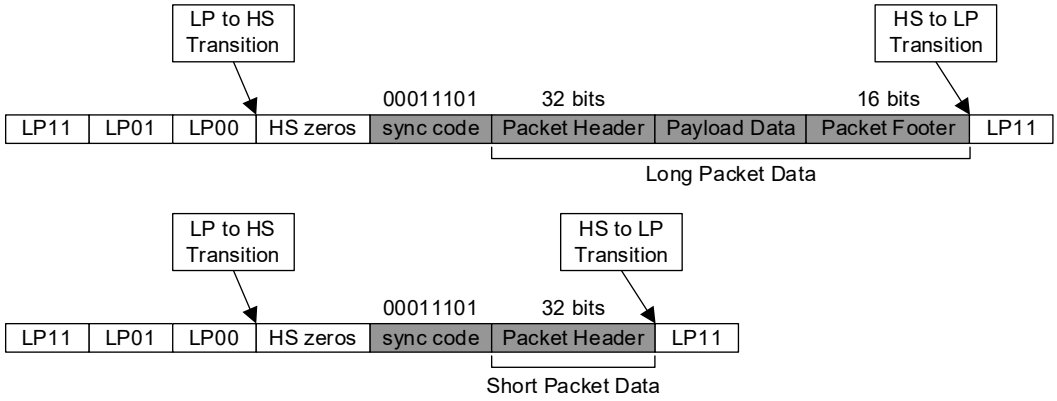

Figure 2.4. High-Speed Entry Sequence and Payload Data Transmission Cycle on Data Lanes

2.1.2. LMMI Interface

The LMMI (Lattice Memory Mapped Interface) Secondary Module is used for configuring the control registers of the Hard MIPI D-PHY Module. For details on the configuration, refer to the Hard MIPI D-PHY Configuration Registers in Table 5.1.

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

3. IP Parameter Description

MIPI D-PHY Module IP configuration attributes summary is shown in Table 3.1. All attributes can be configured from the General tab of the Lattice Radiant software user interface.

Wherever applicable, default values are in bold.

Table 3.1. Attributes Table

Attribute	Selectable Values	Value Entry Format	Description	Dependency on Other Attributes
Interface Type	Transmit, **Receive**	Pull-down Menu	Selects interface type as Receive or Transmit.	—
MIPI Interface Application	**CSI2**, DSI	Pull-down Menu	Protocol or application to be interface with MIPI D-PHY.	—
D-PHY Module Type	**Hard MIPI D-PHY**, Soft MIPI D-PHY	Pull-down Menu	Selects D-PHY module implementation.	Hard MIPI D-PHY is available only on CrossLink-NX devices (LIFCL-17 and LIFCL-40).
HARD_IP DPHY Mode	CIL, NO CIL, **CIL Bypassed**	Pull-down Menu	Selects Hard D-PHY module implementation.	*D-PHY Module Type* = Hard MIPI D-PHY
D-PHY PLL Mode	Internal, **External**	Pull-down Menu	Selects to use Lattice PLL Soft IP or external PLL.	*Interface Type* = Transmit
D-PHY Clock Mode	**Continuous**, Non-continuous	Pull-down Menu	D-PHY clock mode to be used on Hard D-PHY implementation.	*D-PHY Module Type* = Hard MIPI D-PHY
Interface Clock Frequency (MHz)	40–750, **160**	Real Number Field	MIPI interface input clock frequency.	*D-PHY Module Type* = Soft MIPI D-PHY For CSFBGA121, FFG672 and LFG672 packages: 40–750 MHz Other available packages: 40 – 625 MHz
	40–1250, **160**			*D-PHY Module Type* = Hard MIPI D-PHY
Interface Data Rate (Mbps)	80–1500, **320**	Calculated	Data rate to be used by MIPI D-PHY interface. Display for information only.	*D-PHY Module Type* = Soft MIPI D-PHY For CSFBGA121, FFG672 and LFG672 packages: 80–1500 Mbps Other available packages: 80–1250 Mbps
	80–2500, **320**			*D-PHY Module Type* = Hard MIPI D-PHY
Gearing Ratio	For Soft MIPI D-PHY – **8** For Hard MIPI D-PHY – **8**, 16	Pull-down Menu	Selects gearing ratio.	*Interface Type* = Receive
	For Soft – **8** For Hard – **8**, 16			*Interface Type* = Transmit
Bus Width	**1**, 2, 4	Integer Field	Total number of data lanes.	—
Reference Clock Frequency (MHz)	**24**–200	Integer Field	PLL input clock frequency.	*Interface Type* = Transmit, or *D-PHY Module Type* = Hard MIPI and *HARD_IP DPHY Mode* = CIL

4. Signal Description

This section describes the MIPI D-PHY Module ports. A list of Input and Output signals along with their descriptions is shown in Table 4.1.

Table 4.1. MIPI D-PHY Module Port Descriptions

Port	Type	Mode/Configuration	Description
Clock and Reset			
pll_clkop_i	Input	Available when *Interface Type* is Transmit and *D-PHY PLL Mode* is External.	Input clock from external PLL.
pll_clkos_i	Input	Available when *Interface Type* is Transmit and *D-PHY PLL Mode* is External.	90-degree shifted input clock from external PLL.
clk_byte_o	Output	—	Byte clock.
sync_clk_i	Input	—	GDDR SYNC reference clock.
sync_rst_i	Input	—	GDDR SYNC reset.
LMMI Interface			
lmmi_clk_i	Input	Available when *D-PHY Module type* is Hard MIPI D-PHY.	LMMI interface clock. When *D-PHY Module type* is Hard MIPI D-PHY, this signal is limited to a maximum operating frequency of 60 MHz.
lmmi_resetn_i	Input	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Active low signal to reset the configuration registers.
lmmi_request_i	Input	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Start transaction.
lmmi_wr_rdn_i	Input	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Write = HIGH, Read = LOW.
lmmi_offset_i[4:0]	Input	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Register offset, starting at offset 0.
lmmi_wdata_i[3:0]	Input	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Write data.
lmmi_rdata_o[3:0]	Output	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Read data.
lmmi_rdata_valid_o	Output	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Read transaction is complete and lmmi_rdata_o contains valid data.
lmmi_ready_o	Output	Available when *D-PHY Module type* is Hard MIPI D-PHY.	Secondary is ready to start a new transaction. Secondary can insert wait states by holding this signal low.
MIPI D-PHY High-Speed TX			
hs_tx_en_i	Input	Available when *Interface Type* is Transmit.	High-speed transmit mode enable.
hs_tx_data_i[DW[1]-1:0]	Input	Available when *Interface Type* is Transmit.	High-speed transmit data.
hs_tx_data_en_i	Input	Available when *Interface Type* is Transmit and *Mode Type* is Hard MIPI D-PHY.	High-speed transmit data enable.
hs_tx_clk_en_i	Input	Available when *Interface Type* is Transmit and *D-PHY Module type* is Soft MIPI D-PHY.	High-speed transmit mode clock enable.
txclk_hsgate_i	Input	Available when *Interface type* is Transmit, *D-PHY Module Type* is Hard MIPI D-PHY, and *Hard D-PHY Mode* is CIL Bypassed.	High-speed transmitter clock gate signal.

Port	Type	Mode/Configuration	Description
hs_tx_cil_ready_o[NUM_of_LANES – 1:0]	Output	Available when *Interface Type* is Transmit, *D-PHY Module Type* is Hard MIPI D-PHY, and *D-PHY Mode* is set to CIL.	This active high signal indicates that TxDataHS is accepted by the corresponding lane to be serially transmitted.
MIPI D-PHY High-Speed RX			
hs_rx_en_i	Input	Available when *Interface Type* is in Receive and *D-PHY Module Type* as Soft MIPI D-PHY.	High-speed receive mode enable.
hs_rx_clk_en_i	Input	Available when *Interface Type* is Receive, *D-PHY Module type* is Hard MIPI D-PHY, and *Hard D-PHY Mode* is CIL Bypassed.	Active high enable signal for the line termination of the D-PHY clock lane.
hs_rx_data_en_i	Input	Available when *Interface Type* is Receive, *D-PHY Module type* is Hard MIPI D-PHY, and *Hard D-PHY Mode* is CIL Bypassed.	Active high enable signal for the line termination of the D-PHY data lane.
hs_data_des_en_i	Input	Available when *Interface Type* is Receive, *D-PHY Module type* is Hard MIPI D-PHY, and *Hard D-PHY Mode* is CIL Bypassed.	Active high signal to override the deserializer token detector and enable deserializer byte data.
hs_rx_data_o[DW[1]-1:0]	Output	Available when *Interface Type* is Receive.	High-speed receive data. The data is gated by *clk_byte_o* clock.
hs_rx_data_sync_o[BUS_WIDTH – 1:0]	Output	Available when *Interface Type* is Receive and *Module Type* is Hard MIP.	Produces a pulse when de-serializer detects sync char(B8). On negedge of the pulse the bus *hs_rx_data_o* contains a valid data.
data_lane_ss_o[NUM_of_LANES – 1:0]	Output	Available when *Interface Type* is Receive and *D-PHY Mode* = CIL Bypassed.	This active high signal indicates that the corresponding lane is currently in Stop state.
MIPI D-PHY Low Power TX			
lp_tx_clk_p_i	Input	Available when *Interface Type* is Transmit.	Low power transmit positive lane clock.
lp_tx_clk_n_i	Input	Available when *Interface Type* is Transmit.	Low power transmit negative lane clock.
lp_tx_data_p_i[BUS_WIDTH – 1:0]	Input	Available when *Interface Type* is Transmit.	Low power transmit positive data lane.
lp_tx_data_n_i[BUS_WIDTH – 1:0]	Input	Available when *Interface Type* is Transmit.	Low power transmit negative data lane.
lp_tx_data_en_i	Input	Available when *Interface Type* is Transmit and *D-PHY Mode* = CIL Bypassed.	Low power transmit data enable.
lp_tx_en_i	Input	Available when *Interface Type* is Transmit.	Low power transmit enable.
MIPI D-PHY Low Power RX			
lp_rx_en_i	Input	Available when *Interface Type* is Receive.	Low power receive mode enable.
lp_rx_clk_p_o	Output	Available when *Interface Type* is Receive.	Low power receive positive lane clock.
lp_rx_clk_n_o	Output	Available when *Interface Type* is Receive.	Low power receive positive lane clock.
lp_rx_data_p_o[BUS_WIDTH – 1:0]	Output	Available when *Interface Type* is Receive.	Low power receive data positive.
lp_rx_data_n_o[BUS_WIDTH – 1:0]	Output	Available when *Interface Type* is Receive.	Low power receive data negative.

Port	Type	Mode/Configuration	Description
MIPI D-PHY			
clk_p_io, clk_n_io	Input/Output	—	MIPI D-PHY clock lane.
data_p_io[NUM_of_LANES – 1:0], data_n_io[NUM_of_LANES – 1:0]	Input/Output	—	MIPI D-PHY data lanes.
Misc			
pll_lock_i	Input	Available when *Interface Type* is Transmit and *D-PHY PLL Mode* is External.	Lock signal from external PLL.
pd_dphy_i	Input	—	Power down. For Hard MIPI D-PHY, this signal must only be deasserted to low after the VCC is fully ramped up.
ready_o	Output	—	Ready from D-PHY or from PLL.
usrstdby_i	Input	Available when *Interface Type* is Transmit and *D-PHY Module type* is Hard MIPI D-PHY.	Active high puts the hard D-PHY block to standby mode.

Notes:

1. DW = BUS_WIDTH × GEAR
2. BUS_WIDTH – Number of D-PHY Lanes, 1, 2, 4 (available on user interface)
3. GEAR – Number of bits to be transferred

5. Register Description

All configuration registers for Hard MIPI D-PHY (programmable bits) in Table 5.1 are controlled through LMMI bus.

Table 5.1. Configuration Registers (MIPI Programmable Bits)

Offset	Register Name	Access Type	Description
0x00	HSEL, AUTO_PD_EN, PRIMARY_SECONDARY, DSI_CSI	RW	For universal control pins
0x01	RXCDRP	RW	For BIST control pin
0x02	EN_CIL, RXLPRP	RW	For universal and BIST control pin
0x03	PLLCLKBYPASS, LOCK_BYP, DESKEW_EN	RW	For universal control pins
0x04-0x08	CN, CM and CO	RW	For PLL dividers
0x09-0x0A	LANE0_SEL, RxDataWidthHS, TxDataWidthHS, CFG_NUM_LANES	RW	For universal control pins
0x0C-0x15	uc_PRG_HS_ZERO, uc_PRG_HS_PREPARE, uc_PRG_HS_TRAIL, u_PRG_HS_ZERO, u_PRG_HS_PREPARE, u_PRG_HS_TRAIL, TEST_ENBL	RW	For Universal PHY Protocol Interface (PPI) clock and data lane and BIST control pins
0x16-0x1D	TEST_PATTERN	RW	For BIST pattern generator and matcher
0x1E	CONT_CLK_MODE	RW	For universal control pin

The behavior of registers to write and read access is defined by the access type, which is defined in Table 5.2.

Table 5.2. Access Type Definition

Access Type	Behavior on Read Access	Behavior on Write Access
RO	Returns register value	Ignores write access.
WO	Returns 0	Updates register value.
RW	Returns register value	Updates register value.
RSVD	Returns 0	Ignores write access.

5.1. HSEL, AUTO_PD_EN, PRIMARY_SECONDARY, and DSI_CSI

Table 5.3. HSEL, AUTO_PD_EN, PRIMARY_SECONDARY, and DSI_CSI at 0x00 Offset Address

Field	Name	Description	Access	Width	Reset
[3]	HSEL	RX high speed select 1'b0 – For 1.5 Gbps operation and below 1'b1 – For higher than 1.5 Gbps operation	RW	1	1'b0
[2]	AUTO_PD_EN	Powers down inactive lanes 1'b0 – Inactive lanes powered up and at LP11 1'b1 – Inactive lanes are powered down	RW	1	1'b0
[1]	PRIMARY_SECONDARY	Selects PHY IP configuration 1'b0 – Secondary configuration 1'b1 – Primary configuration	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
[0]	DSI_CSI	Selects PHY IP application 1'b0 – CSI2 configuration 1'b1 – DSI configuration	RW	1	1'b0

5.2. RXCDRP

Table 5.4. RXCDRP at 0x01 Offset Address

Field	Name	Description	Access	Width	Reset
[3:2]	RXCDRP	LP-CD threshold voltage with minimum and maximum value of 200 mV and 450mV respectively. Default is 2'b01.	RW	2	2'b01
[1:0]	—	Reserved. When writing to this register, keep bit to 0. Otherwise, the IP may malfunction.	RW	2	2'b00

5.3. EN_CIL and RXLPRP

Table 5.5. EN_CIL and RXLPRP at 0x02 Offset Address

Field	Name	Description	Access	Width	Reset
[3]	EN_CIL	Enables or disables CIL 1'b0 – CIL bypassed 1'b1 – CIL enabled	RW	1	1'b0
[2:0]	RXLPRP	Adjust the threshold voltage and hysteresis of LP-RX.	RW	3	3'b001

5.4. PLLCLKBYPASS, LOCK_BYP, and DESKEW_EN

Table 5.6. PLLCLKBYPASS, LOCK_BYP, and DESKEW_EN at 0x03 Offset Address

Field	Name	Description	Access	Width	Reset
[3]	TST[0]	When writing to this register, keep this bit to 1'b1. Otherwise, the IP may malfunction.	RW	1	1'b1
[2]	PLLCLKBYPASS	Internal PLL bypass control 1'b0 – Internal PLL enabled 1'b1 – Internal PLL bypassed	RW	1	1'b1
[1]	LOCK_BYP	When clock lane exits from ULPS, this input determines if the PLL LOCK signal is used to gate the TxWordClkHS. 1'b0 – PLL LOCK signal gates TxWordClkHS clock 1'b1 – PLL LOCK signal does not gate TxWordClkHS clock	RW	1	1'b0
[0]	DESKEW_EN	Enables de-skew feature which modifies ERRSYNC/NOSYNC behavior. 1'b0 – De-skew feature disabled 1'b1 – De-skew feature enabled	RW	1	1'b0

5.5. CN, CM, and CO

Table 5.7. CN PLL Divider at 0x04 Offset Address

Field	Name	Description	Access	Width	Reset
[3]	CN[0]	The N parameter of the internal PLL in the equation, Output = M/(N×O). See Table 5.13 for values.	RW	1	1'b0
[2:0]	TST[3:1]	When writing to this register, keep bit to 3'b100. Otherwise, the IP may malfunction.	RW	3	3'b100

Table 5.8. CN PLL Divider at 0x05 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	CN[4:1]	The N parameter of the internal PLL in the equation, Output = M/(N×O). See Table 5.13 for values.	RW	4	4'b0000

Table 5.9. CM PLL Divider at 0x06 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	CM[3:0]	The M parameter of the internal PLL in the equation, Output = M/(N×O). See Table 5.14 for values.	RW	4	4'b0000

Table 5.10. CM PLL Divider at 0x07 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	CM[7:4]	The M parameter of the internal PLL in the equation, Output = M/(N×O). See Table 5.14 for values.	RW	4	4'b0000

Table 5.11. CO PLL Divider at 0x08 Offset Address

Field	Name	Description	Access	Width	Reset
[3]	TxDataWidthHS[0]	LSB High-Speed Transmit Byte Clock. See Table 5.15 for MSB.	RW	1	1'b0
[2:0]	CO[2:0]	The O parameter of the internal PLL in the equation, Output = M/(N×O). See Table 5.12 for values.	RW	3	3'b000

5.5.1. Divider Values for PLL Programming

Table 5.12 shows output divider and its actual values.

Table 5.12. CO Table of Values

Control O Value	Actual O Value
000	1
001	2
010	4
011	8
111	16

Table 5.13 shows input divider and its actual values.

Table 5.13. CN Table of Values

Control N Value	Actual N Value	Control N Value	Actual N Value
11111	1	11010	17
00000	2	11101	18
10000	3	11110	19
11000	4	01111	20
11100	5	10111	21
01110	6	11011	22
00111	7	01101	23
10011	8	10110	24
01001	9	01011	25
00100	10	00101	26
00010	11	10010	27
10001	12	11001	28
01000	13	01100	29
10100	14	00110	30
01010	15	00011	31
10101	16	00001	32

Table 5.14 shows feedback divider and its actual values.

Table 5.14. CM Table of Values

Control M Value	Actual M Value	Control M Value	Actual M Value	Control M Value	Actual M Value	Control M Value	Actual M Value
111X0000	16	10001100	76	00001000	136	01000100	196
111X0001	17	10001101	77	00001001	137	01000101	197
111X0010	18	10001110	78	00001010	138	01000110	198
111X0011	19	10001111	79	00001011	139	01000111	199
111X0100	20	10010000	80	00001100	140	01001000	200
111X0101	21	10010001	81	00001101	141	01001001	201
111X0110	22	10010010	82	00001110	142	01001010	202
111X0111	23	10010011	83	00001111	143	01001011	203
111X1000	24	10010100	84	00010000	144	01001100	204
111X1001	25	10010101	85	00010001	145	01001101	205
111X1010	26	10010110	86	00010010	146	01001110	206
111X1011	27	10010111	87	00010011	147	01001111	207
111X1100	28	10011000	88	00010100	148	01010000	208
111X1101	29	10011001	89	00010101	149	01010001	209
111X1110	30	10011010	90	00010110	150	01010010	210
111X1111	31	10011011	91	00010111	151	01010011	211
11000000	32	10011100	92	00011000	152	01010100	212
11000001	33	10011101	93	00011001	153	01010101	213
11000010	34	10011110	94	00011010	154	01010110	214
11000011	35	10011111	95	00011011	155	01010111	215
11000100	36	10100000	96	00011100	156	01011000	216
11000101	37	10100001	97	00011101	157	01011001	217
11000110	38	10100010	98	00011110	158	01011010	218
11000111	39	10100011	99	00011111	159	01011011	219

Control M Value	Actual M Value	Control M Value	Actual M Value	Control M Value	Actual M Value	Control M Value	Actual M Value
11001000	40	10100100	100	00100000	160	01011100	220
11001001	41	10100101	101	00100001	161	01011101	221
11001010	42	10100110	102	00100010	162	01011110	222
11001011	43	10100111	103	00100011	163	01011111	223
11001100	44	10101000	104	00100100	164	01100000	224
11001101	45	10101001	105	00100101	165	01100001	225
11001110	46	10101010	106	00100110	166	01100010	226
11001111	47	10101011	107	00100111	167	01100011	227
11010000	48	10101100	108	00101000	168	01100100	228
11010001	49	10101101	109	00101001	169	01100101	229
11010010	50	10101110	110	00101010	170	01100110	230
11010011	51	10101111	111	00101011	171	01100111	231
11010100	52	10110000	112	00101100	172	01101000	232
11010101	53	10110001	113	00101101	173	01101001	233
11010110	54	10110010	114	00101110	174	01101010	234
11010111	55	10110011	115	00101111	175	01101011	235
11011000	56	10110100	116	00110000	176	01101100	236
11011001	57	10110101	117	00110001	177	01101101	237
11011010	58	10110110	118	00110010	178	01101110	238
11011011	59	10110111	119	00110011	179	01101111	239
11011100	60	10111000	120	00110100	180	01110000	240
11011101	61	10111001	121	00110101	181	01110001	241
11011110	62	10111010	122	00110110	182	01110010	242
11011111	63	10111011	123	00110111	183	01110011	243
10000000	64	10111100	124	00111000	184	01110100	244
10000001	65	10111101	125	00111001	185	01110101	245
10000010	66	10111110	126	00111010	186	01110110	246
10000011	67	10111111	127	00111011	187	01110111	247
10000100	68	00000000	128	00111100	188	01111000	248
10000101	69	00000001	129	00111101	189	01111001	249
10000110	70	00000010	130	00111110	190	01111010	250
10000111	71	00000011	131	00111111	191	01111011	251
10001000	72	00000100	132	01000000	192	01111100	252
10001001	73	00000101	133	01000001	193	01111101	253
10001010	74	00000110	134	01000010	194	01111110	254
10001011	75	00000111	135	01000011	195	01111111	255

5.6. LANE0_SEL, RxDataWidthHS, TxDataWidthHS, and CFG_NUM_LANES

Table 5.15. LANE0_SEL, RxDataWidthHS, and TxDataWidthHS at 0x09 Offset Address

Field	Name	Description	Access	Width	Reset
[3]	LANE0_SEL[0]	LSB of LANE0_SEL. Determines which lane acts as data lane 0 in high-speed operation mode. See Table 5.15 for MSB.	RW	1	1'b0

Field	Name	Description	Access	Width	Reset
[2:1]	RxDataWidthHS	High speed receive data width select. 2'b00 – 1/8 of the high speed bit rate 2'b01 – 1/16 of the high speed bit rate 2'b10 – 1/32 of the high speed bit rate	RW	2	2'b00
[0]	TxDataWidthHS[1]	MSB high speed transmit byte clock. 2'b00 – 1/8 of the high speed bit rate 2'b01 – 1/16 of the high speed bit rate 2'b10 – 1/32 of the high speed bit rate	RW	1	1'b0

Table 5.16. LANE0_SEL and CFG_NUM_LANES at 0x0A Offset Address

Field	Name	Description	Access	Width	Reset
[3]	—	Reserved. When writing to this register, keep bit to 0. Otherwise, the IP may malfunction.	RW	1	1'b0
[2:1]	CFG_NUM_LANES	Specifies the number of active lanes. 2'b00 – 1 active lane 2'b01 – 2 active lanes 2'b10 – 3 active lanes 2'b11 – 4 active lanes	RW	2	2'b00
[0]	LANE0_SEL[1]	MSB of LANE0_SEL. Determines which lane acts as data lane 0 in high-speed operation mode. 2'b00 – Lane 0 acts as data lane 0 2'b01 – Lane 1 acts as data lane 0 2'b10 – Lane 2 acts as data lane 0 2'b11 – Lane 3 acts as data lane 0	RW	1	1'b0

5.7. Universal PPI Clock and Date Lane Control Pins

Table 5.17. uc_PRG_HS_ZERO and uc_PRG_HS_PREPARE at 0x0C Offset Address

Field	Name	Description	Access	Width	Reset
[3:2]	uc_PRG_HS_ZERO[1:0]	Programs T_CLK_ZERO time in the beginning of high-speed transmission mode for clock lane pins. T_CLK_ZERO = (uc_PRG_HS_ZERO + 4) × (ByteClk Period)	RW	2	2'b00
[1]	uc_PRG_HS_PREPARE[0]	T_CLK_PREPARE time in the beginning of high-speed transmission mode for clock lane pins. 1'b0 – Tperiod of sync_clk_i 1'b1 – 1.5 × Tperiod of sync_clk_i	RW	1	1'b0
[0]	—	Reserved. When writing to this register, keep bit to 0. Otherwise, the IP may malfunction.	RW	1	1'b0

Table 5.18. uc_PRG_HS_ZERO at 0x0D Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	uc_PRG_HS_ZERO[5:2]	Programs T_CLK_ZERO time in the beginning of high-speed transmission mode for clock lane pins. T_CLK_ZERO = (uc_PRG_HS_ZERO + 4) × (ByteClk Period)	RW	4	4'b0000

Table 5.19. uc_PRG_HS_TRAIL and uc_PRG_HS_ZERO at 0x0E Offset Address

Field	Name	Description	Access	Width	Reset
[3:1]	uc_PRG_HS_TRAIL[2:0]	Programs T_CLK_TRAIL time in the end of high-speed transmission mode for clock lane pins. T_HS_TRAIL = (uc_PRG_HS_TRAIL) × (ByteClk Period)	RW	3	3'b000
[0]	uc_PRG_HS_ZERO[6]	Programs T_CLK_ZERO time in the beginning of high-speed transmission mode for clock lane pins. T_CLK_ZERO = (uc_PRG_HS_ZERO + 4) × (ByteClk Period)	RW	1	1'b0

Table 5.20. uc_PRG_HS_TRAIL at 0x0F Offset Address

Field	Name	Description	Access	Width	Reset
[3:2]	—	Reserved. When writing to this register, keep bit to 0. Otherwise, the IP may malfunction.	RW	2	2'b00
[1:0]	uc_PRG_HS_TRAIL[4:3]	Programs T_CLK_TRAIL time in the end of high-speed transmission mode for clock lane pins. T_HS_TRAIL = (uc_PRG_HS_TRAIL) × (ByteClk Period)	RW	2	2'b00

Table 5.21. u_PRG_HS_ZERO and u_PRG_HS_PREPARE at 0x11 Offset Address

Field	Name	Description	Access	Width	Reset
[3:2]	u_PRG_HS_ZERO[1:0]	Programs T_HS_ZERO time in the beginning of high-speed transmission mode for data lane pins. See Table 5.22 for MSB.	RW	2	2'b00
[1:0]	u_PRG_HS_PREPARE[1:0]	T_HS_PREPARE time in the beginning of high-speed transmission mode for data lane pins. 2'b00 – Tperiod of sync_clk_i 2'b01 – 1.5 × Tperiod of sync_clk_i 2'b10 – 2 × Tperiod of sync_clk_i 2'b11 – 2.5 × Tperiod of sync_clk_i	RW	2	2'b00

Table 5.22. u_PRG_HS_ZERO at 0x12 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	u_PRG_HS_ZERO[5:2]	Programs T_HS_ZERO time in the beginning of high-speed transmission mode for data lane pins. T_HS_ZERO = (u_PRG_HS_ZERO + 5 + 2M) × (ByteClk Period), where M is: 0 – Single interface 1 – Double interface 2 – Quad interface	RW	4	4'b0000

Table 5.23. u_PRG_HS_TRAIL at 0x13 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	u_PRG_HS_TRAIL[3:0]	Programs T_HS_TRAIL time in the end of high-speed transmission mode for data lane pins. T_HS_TRAIL = (u_PRG_HS_TRAIL) × (ByteClk Period)	RW	4	4'b0000

Table 5.24. TEST_ENBL and u_PRG_HS_TRAIL at 0x14 Offset Address

Field	Name	Description	Access	Width	Reset
[3:2]	TEST_ENBL[1:0]	Six-bit signal that enables the testing modes. See Table 5.25 for MSB.	RW	2	2'b00
[1:0]	u_PRG_HS_TRAIL[5:4]	Programs T_HS_TRAIL time in the end of high-speed transmission mode for data lane pins. T_HS_TRAIL = (u_PRG_HS_TRAIL) × (ByteClk Period)	RW	2	2'b00

Table 5.25. TEST_ENBL at 0x15 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_ENBL[5:2]	Six-bit signal that enables the testing modes.	RW	4	4'b0000

5.8. TEST_PATTERN

Table 5.26. TEST_PATTERN at 0x16 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[3:0]	Used by the BIST pattern generator and matcher.	RW	4	4'b0000

Table 5.27. TEST_PATTERN at 0x17 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[7:4]	Used by the BIST pattern generator and matcher.	RW	4	4'b0000

Table 5.28. TEST_PATTERN at 0x18 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[11:8]	Used by the BIST pattern generator and matcher.	RW	4	4'b0000

Table 5.29. TEST_PATTERN at 0x19 Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[15:12]	Used by the BIST pattern generator and matcher.	RW	4	4'b0000

Table 5.30. TEST_PATTERN at 0x1A Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[19:16]	Used by the BIST pattern generator and matcher.	RW	4	4'b0000

Table 5.31. TEST_PATTERN at 0x1B Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[23:20]	Used by the BIST pattern generator and matcher.	RW	4	4'b0010

Table 5.32. TEST_PATTERN at 0x1C Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[27:24]	Used by the BIST pattern generator and matcher.	RW	4	4'b0000

Table 5.33. TEST_PATTERN at 0x1D Offset Address

Field	Name	Description	Access	Width	Reset
[3:0]	TEST_PATTERN[31:28]	Used by the BIST pattern generator and matcher.	RW	4	4'b1000

5.9. CONT_CLK_MODE

Table 5.34. CONT_CLK_MODE at 0x1E Offset Address

Field	Name	Description	Access	Width	Reset
[3:2]	—	Reserved.	RW	2	2'b00
[1]	—	Reserved. When writing to this register, keep bit to 0. Otherwise, the IP may malfunction.	RW	1	1'b0
[0]	CONT_CLK_MODE	Controls the secondary clock lane to maintain HS reception state during continuous clock mode generation. 1'b0 – Secondary clock lane disabled 1'b1 – Secondary clock lane enabled	RW	1	1'b1

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.

Note: The screenshots provided are for reference only. Details may vary depending on the version of the IP or software being used. If there have been no significant changes to the GUI, a screenshot may reflect an earlier version of the IP.

6.1. Generating and Instantiating the IP

You can use the Lattice Radiant software to generate IP modules and integrate the modules into the device architecture.

To generate the MIPI D-PHY Module 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 **MIPI D-PHY Module** under **Module, Architecture_Modules, IO** 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**.

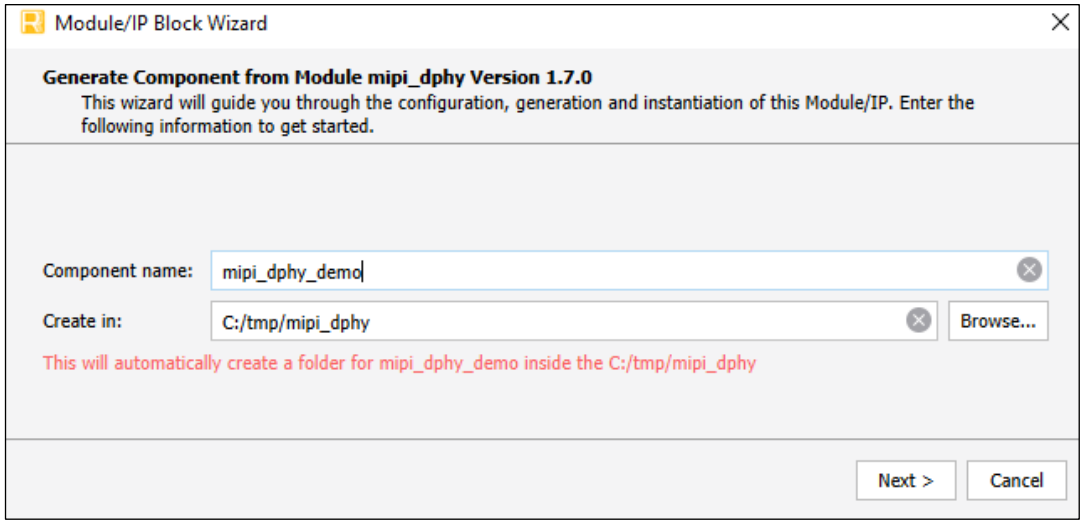

Figure 6.1. Module/IP Block Wizard

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

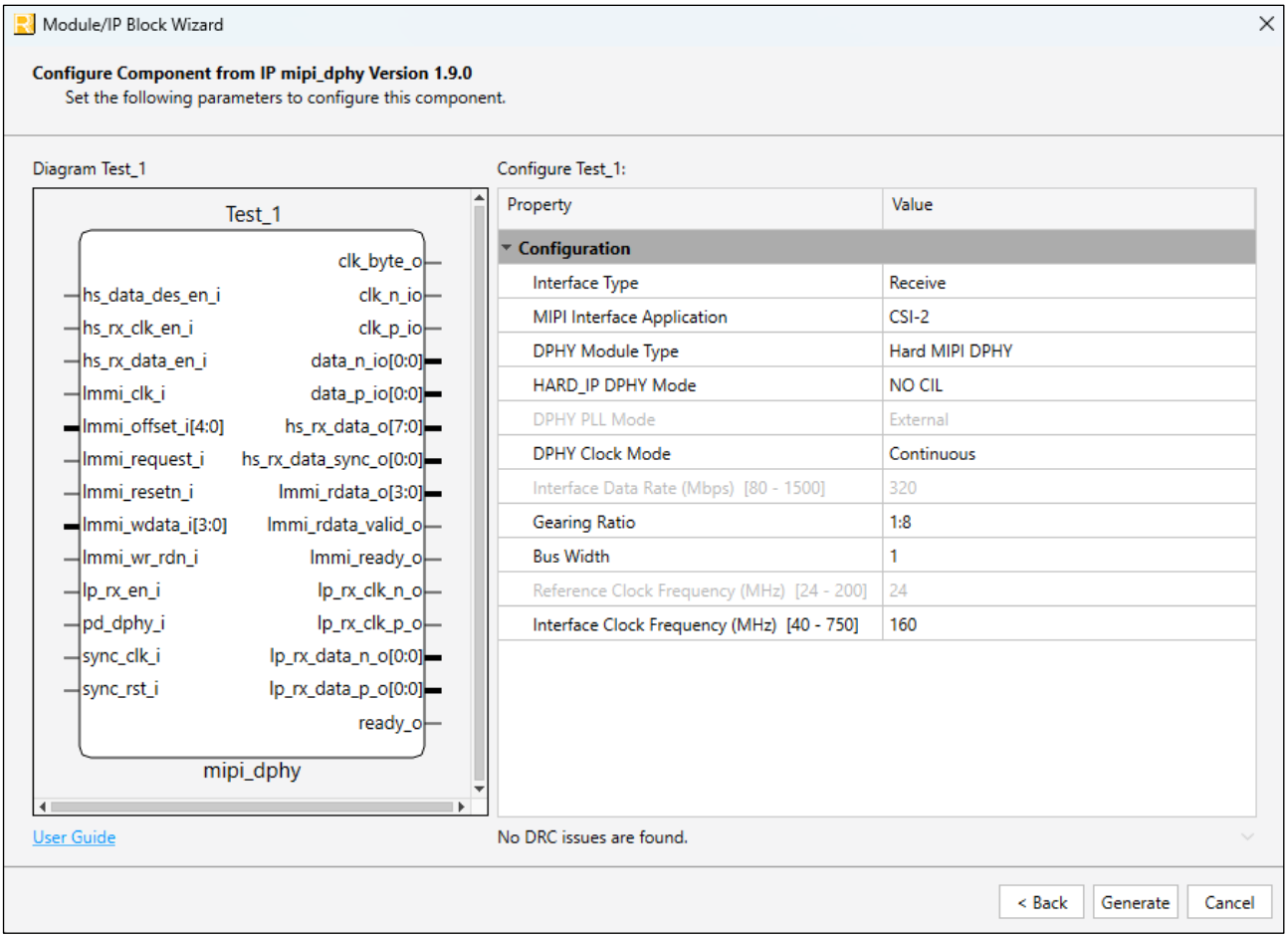

Figure 6.2. IP Configuration

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

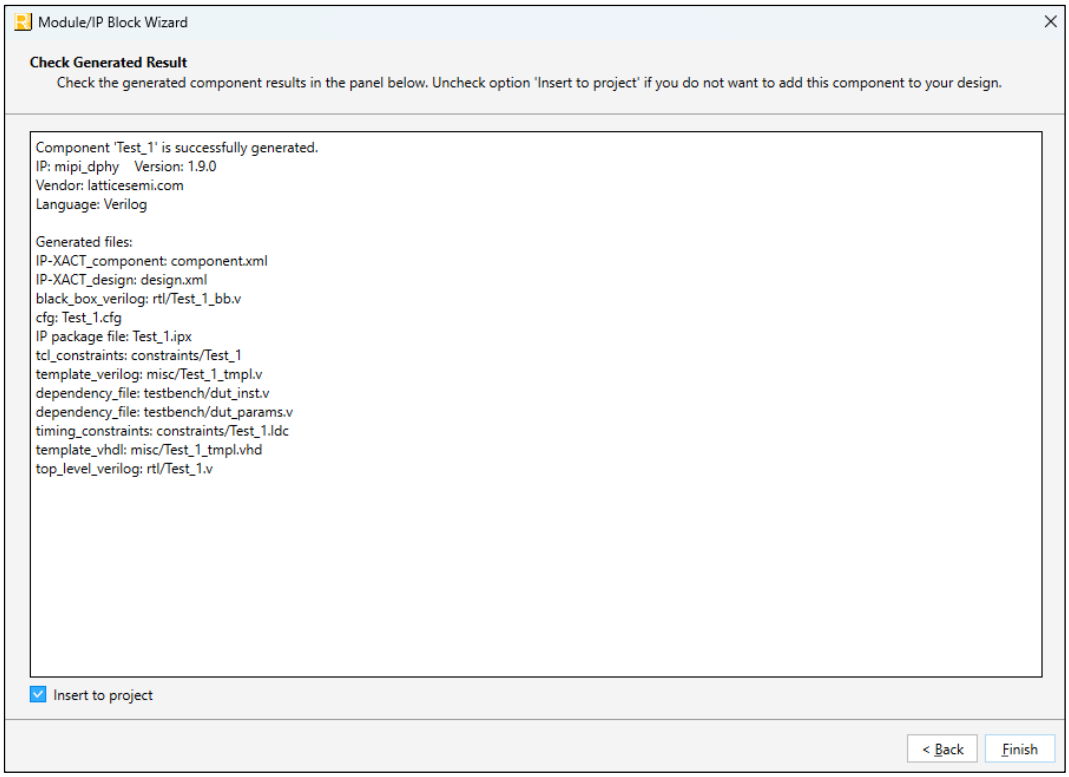

Figure 6.3. Check Generated Result

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 MIPI D-PHY 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

The timing constraints are based on the clock frequency used. The timing constraints for the IP are defined in the relevant constraint files. The following example shows the IP timing constraints generated for the MIPI D-PHY Module IP.

```
create_clock -name {sync_clk_i} -period 13.33 [get_ports sync_clk_i]
create_clock -name {clk_p_io} -period 1.33333333333333 [get_pins -hierarchical u_mipi_clk.IOA_inst/O]
set_false_path -to [get_pins -hierarchical u_eclkdiv.ECLKDIV_inst/DIVRST]
```

Figure 6.4. Timing Constraint File (.pdc) for the MIPI D-PHY IP using Soft D-PHY for Receiver

Generate input clocks accordingly. Note the byte clock frequencies are auto generated as a generated clock, depending on the gear and MIPI D-PHY interface frequency. For timing closure, add the following timing constraints in the .pdc file.

- For Soft MIPI D-PHY RX mode, define the sync clock frequency and MIPI D-PHY interface frequency. For example:
  - create_clock -name {sync_clk_i} -period 13.33 [get_ports sync_clk_i]
  - create_clock -name {clk_p_io} -period 0.67 [get_pins -hierarchical u_mipi_clk.IOA_inst/O]
- For Soft MIPI D-PHY TX mode with external PLL, define the sync clock frequency and clkop and clkos frequencies. For example:
  - create_clock -name {sync_clk_i} -period 13.33 [get_ports sync_clk_i]
  - create_clock -name {pll_clkop_i} -period 25 [get_ports pll_clkop_i]
  - create_clock -name {pll_clkos_i} -period 25 [get_ports pll_clkos_i]
- For Soft MIPI D-PHY TX mode with internal PLL, define the sync clock frequency. The internal MIPI Interface Clock are auto generated as a generate clock. For example:
  - create_clock -name {sync_clk_i} -period 13.33 [get_ports sync_clk_i]

Define timing exceptions as follows:

- For Soft D-PHY mode, set false path to the DIVRST pin of the ECLKDIV block. For example:
  - set_false_path -to [get_pins -hierarchical u_eclkdiv.ECLKDIV_inst/DIVRST]

Refer to Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059) for details on how to constrain your design.

6.4. Physical Constraints

For Soft MIPI D-PHY, both the data and clock lanes need to be located on the HPIO banks with 1.2 V VCCIO. Refer to sysI/O User Guide for Nexus Platform (FPGA-TN-02067) for more information.

For Hard MIPI D-PHY pin placement, refer to Appendix A of CrossLink-NX Hardened D-PHY Usage Guide (FPGA-TN-02081).

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.

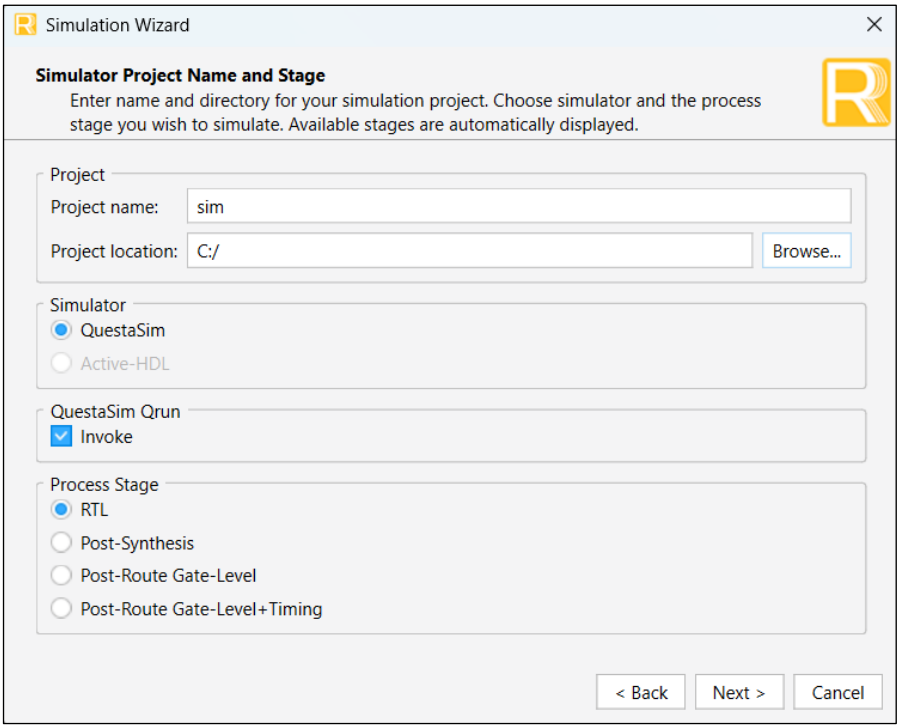

Figure 6.5. Simulation Wizard

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

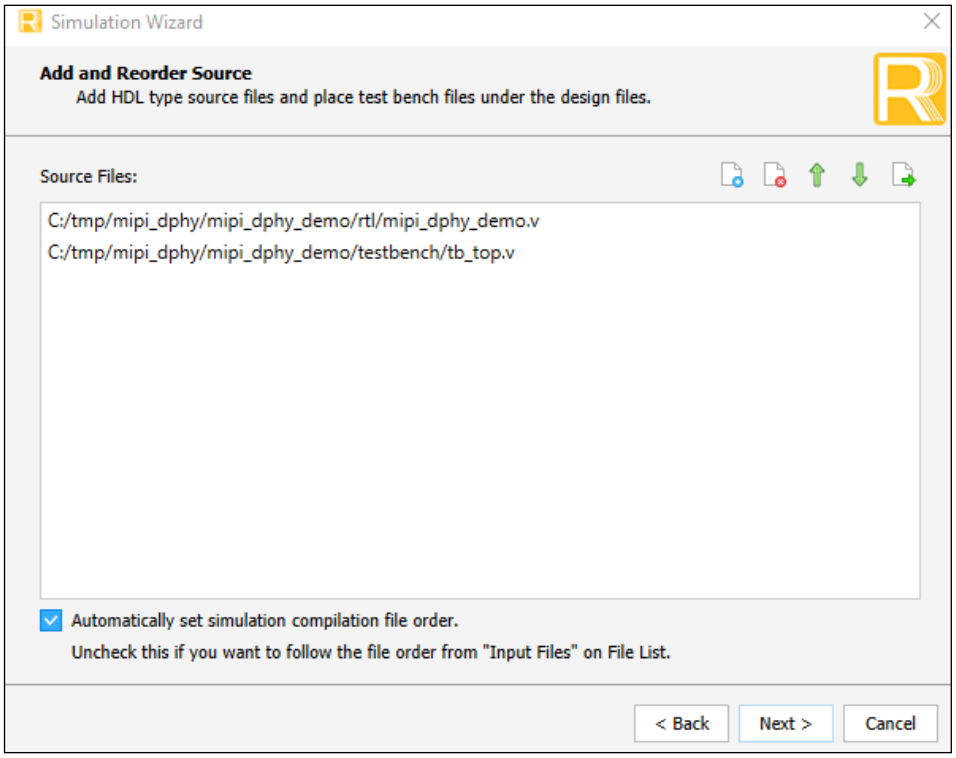

Figure 6.6. Add and Reorder Source Window

3. Click **Next**. The Parse HDL files for simulation window opens. Set Simulation Top Module to tb_top as show in Figure 6.7.

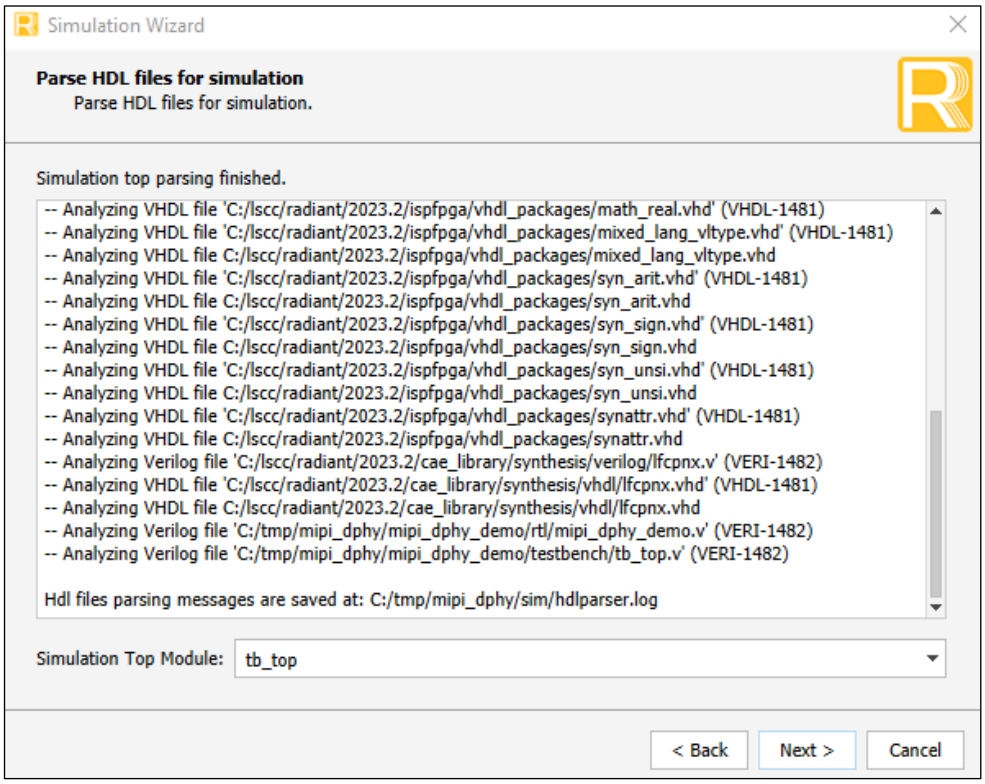

Figure 6.7. Parse HDL Files for Simulation Window

4. Click **Next**. The **Summary** window is shown. Set **Default Run** to 0 ns to complete the run as shown in Figure 6.8.

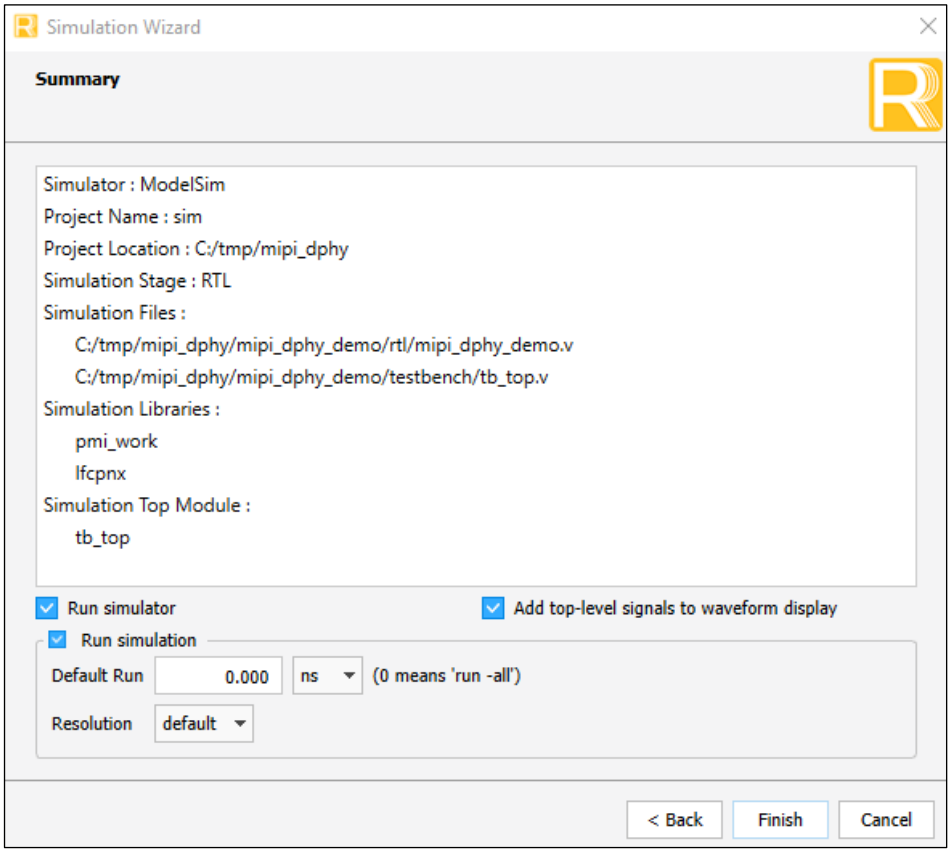

Figure 6.8. Simulation Wizard Summary Window

5. Click **Finish** to run the simulation.

The waveform in Figure 6.9 shows an example simulation result.

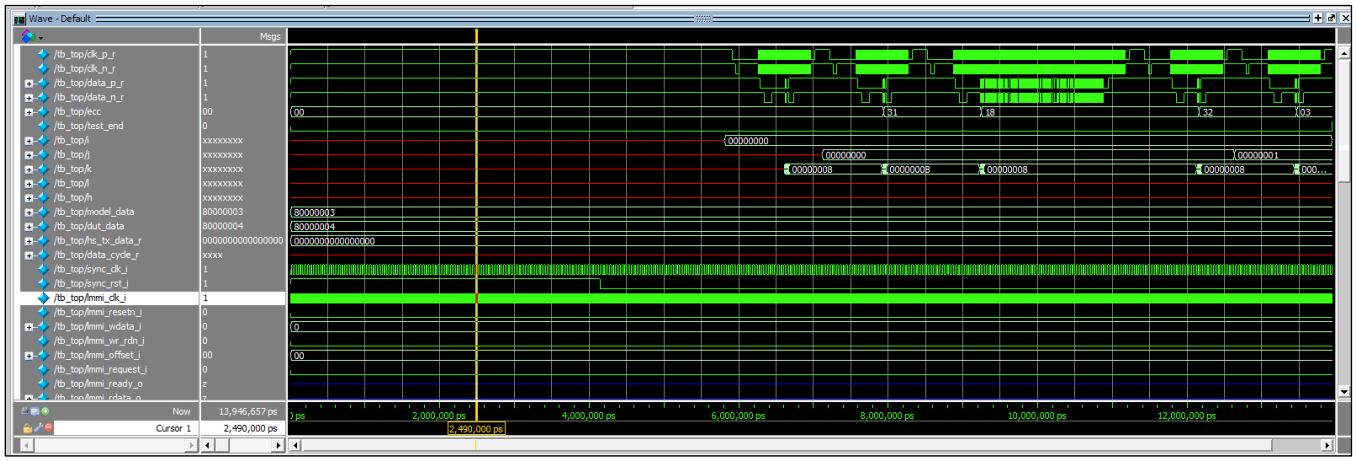

Figure 6.9. Simulation Waveform

6.6.1. Simulation Results

To display all the IP pins in the simulation, use the following TCL command:

```
add wave /tb_top/<instance_name>/*
```

For example:

```
add wave /tb_top/u_mipi_dphy_demo/*
```

When the simulation is complete successfully, the Transcript console shows “SIMULATION PASSED” as shown in Figure 6.10.

```
# ***Test Start***
# ***IP Reset***
# ***Finish Reset ***
# Wait for Ready
# IP Ready
# ***SIMULATION PASSED***
# ** Note: $finish    : C:/tmp/mipi_dphy/mipi_dphy_demo/testbench/tb_top.v(1229)
#    Time: 13946657 ps  Iteration: 0  Instance: /tb_top
# 1
# Break in Task data_check_t at C:/tmp/mipi_dphy/mipi_dphy_demo/testbench/tb_top.v line 1229

VSIM 2>
```

Figure 6.10. Simulation Log

6.6.2. Testbench Control for Hard MIPI D-PHY Simulation

Hard MIPI D-PHY uses timing parameters to ensure proper operation. The testbench that is provided with the IP uses frequency values that are within the valid ranges of the timing parameters.

For Receive Hard MIPI D-PHY CIL Bypassed Mode, testbench parameters in the following table can be set to a value within the specified range for proper operation.

Table 6.2. Configurable Hard MIPI D-PHY Timing Parameters on IP Testbench

Timing Parameter	Minimum (ns)	Maximum (ns)	Description
T_CLK_PREPARE	38	95	Time that Hard MIPI D-PHY drives the clock lane LP-00 line state immediately before the HS-0 line state starts HS transmission.
T_HS_PREPARE	40 ns + 4 × UI	85 ns + 6 × UI	Time that Hard MIPI D-PHY drives the data lane LP-00 line state immediately before the HS-0 line state starts HS transmission.
T_HS_TRAIL	8 × UI	60 ns + 4 × UI	Time that Hard MIPI D-PHY drives the flipped differential state after last payload data bit of a HS transmission burst.

Note:

1. UI corresponds to half of the D-PHY clock period.

7. Debugging

This section lists possible issues and suggested troubleshooting steps that you can follow.

7.1. Debug Methods

7.1.1. No output from the module

If there is no output signal from the module, use the following guidelines:

- Check if the PLL is locked.
- For Hard D-PHY, ensure pd_dphy_i pin is de-asserted only after the VCC is fully ramped up.
- Check the presence of GSR using Netlist Analyzer. Remove GSR if the GSR is mistakenly synthesized as one of the reset signals.

7.2. Debug Tools

You can use various tools to debug MIPI D-PHY Module 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 QuestaSim®.
- 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 to problem areas into many small functional blocks to control and monitor the status of each block.

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

8. Running Functional SimulationDesign Considerations

8.1. Design Considerations for Receiver

- For Hard MIPI D-PHY, ensure that pd_dphy_i pin is not hard tied to low. VCC must be completely ramped up before the pin deassertion.
  - An example of implementation is to use the inversion of a PLL lock signal to drive the pin.
- Ensure to follow the MIPI D-PHY Specification V1.2 on transitioning from low power to high-speed mode:
  - Check for LP11 to LP01 change on the data.
  - Detect LP01 to LP00 state.
  - Wait for $T_{HS\text{-}PREPARE}$ of the transmitter to complete, then assert hs_rx_en_i.
  - Note the hs_rx_en_i must be asserted before the sync bit is sent to avoid data corruption. For example, during the $T_{HS\text{-}ZERO}$ period.
- To achieve optimal sampling of hs_rx_data_o on the D-PHY controller, use a negative-edge triggered flip-flop with clock_byte_o as clock.

8.2. Design Considerations for Transmitter

- For Hard MIPI D-PHY, ensure that pd_dphy_i pin is not hard tied to low. VCC must be completely ramped up before the pin de-assertion.
  - An example of implementation is to use the inversion of a PLL lock signal to drive the pin.
- For external PLL, ensure that pll_clkos_i pin is 90-degree shifted from the pll_clkop_i signal.
- Ensure to follow the MIPI D-PHY Specification V1.2 on transitioning from low power to high-speed mode.

Appendix A. Resource Utilization

Table A.1 shows the resource utilization for MIPI D-PHY Module IP version 1.7.0 using LFCPNX-100-9LFG672I device with Lattice Synthesis Engine of the Lattice Radiant software 2023.2. Default configuration is used, and some attributes are changed from the default value to show the effect on the resource utilization.

Table A.1. Resource Utilization[1]

Interface Type	MIPI Interface Application	Bus Width	Interface clock Frequency (MHz)	Clk Fmax (MHz)[2]	Registers	LUTs[3]	EBRs	DSPs
Receive	CSI-2	1	40	200	10	33	0	0
Receive	CSI-2	4	750	200	10	33	0	0
Transmit[4]	CSI-2	1	40	200	10	33	0	0
Transmit[4]	CSI-2	4	750	200	10	33	0	0
Receive	DSI	1	40	200	10	33	0	0
Receive	DSI	4	750	200	10	33	0	0
Transmit[4]	DSI	1	40	200	10	36	0	0
Transmit[4]	DSI	4	750	200	10	36	0	0

Notes:

1. This table lists only the resource utilization for Soft D-PHY mode, Gear 8.
2. Fmax is generated when the FPGA design contains only the MIPI D-PHY module, and sync clock is 75 MHz. Fmax measures the maximum clk_byte_o frequencies for a given configuration.
3. The distributed RAM utilization is accounted for in the total LUT4 utilization. The actual LUT4 utilization is distribution among logic, distributed RAM, and ripple logic.
4. Calculated based on External PLL mode.

Appendix B. Limitations

When MIPI_DPHY soft IP is configured with the following selections in the user interface, simulation may fail when Reference Clock Frequency is set to higher than 60 MHz:

- Interface Type = Receive
- D-PHY Module Type = Hard MIPI D-PHY
- Hard_IP D-PHY Mode = CIL

References

- MIPI D-PHY Module Release Notes (FPGA-RN-02105)
- Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059)
- Lattice Memory Mapped Interface and Lattice Interrupt Interface User Guide (FPGA-UG-02039)
- sysI/O User Guide for Nexus Platform (FPGA-TN-02067)
- CrossLink-NX Hardened D-PHY Usage Guide (FPGA-TN-02081)
- Reveal User Guide for Radiant Software
- CrossLink-NX web page
- Certus-NX web page
- CertusPro-NX web page
- MachXO5-NX 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

Note: In some instances, the IP may be updated without changes to the user guide. The user guide may reflect an earlier IP version but remains fully compatible with the later IP version. Refer to the IP Release Notes for the latest updates.

Revision 1.9, IP v1.9.0, December 2025

Section	Change Summary
All	• Added a note on IP version in Quick Facts and *Revision History* sections. • Performed minor formatting and editorial edits.
Acronyms in This Document	Updated list of acronyms.

Introduction	• Removed bus lane support for 3 data lanes in Overview and Features sections.
	• Updated Table 1.1. Summary of the MIPI D-PHY Module IP as follows:
	  • Renamed *Supported FPGA Family* to *Supported Devices*.
	  • Removed the *Targeted Devices* row.
	  • Added the *IP Changes* row.
	  • Added IP version.
	  • Removed earlier IP versions.
	• Updated the Licensing and Ordering Information section.
	• Updated IP version in Table 1.2. IP Validation Level.

IP Parameter Description	Updated dependency on other attributes for the *D-PHY Module Type* and *Reference Clock Frequency (MHz)* attributes in Table 3.1. Attributes Table.
Signal Description	Updated Table 4.1. MIPI D-PHY Module Port Descriptions as follows: • Updated lmmi_clk_i, hs_tx_data_en_i, hs_rx_data_sync_o[BUS_WIDTH – 1:0], and lp_tx_data_en_i signals. • Updated the note on BUS_WIDTH.

Designing with the IP	• Added a note on IP version in GUI in the Designing with the IP section.
	• Updated the following figures:
	  • Figure 6.2. IP Configuration
	  • Figure 6.3. Check Generated Result
	  • Figure 6.5. Simulation Wizard

Debugging	• Changed ModelSim to QuestaSim in the Reveal Analyzer section. • Removed the *ModelSim* section.
Design Considerations	Added guidelines on optimal sampling of hs_rx_data_o on the D-PHY controller in the Design Considerations for Receiver section.
References	Updated references.

Revision 1.8, March 2024

Section	Change Summary
All	Renamed document from *MIPI D-PHY Module - Lattice Radiant Software* to *MIPI D-PHY Module*.
Disclaimers	Updated disclaimers.
Inclusive Language	Added the inclusive language boilerplate.
Acronyms in This Document	Added definition for GSR.

Introduction	• Updated subsection 1.1 Overview.
	• Reworked subsection 1.2 *Features* and moved to subsection 1.3 Features.
	• Added the following subsections:
	  • 1.2 Quick Facts
	  • 1.4 Licensing and Ordering Information
	  • 1.5 IP Validation Summary
	  • 1.6 Minimum Device Requirements
	• Moved subsection 1.3 *Conventions* to subsection 1.7 Conventions.

Functional Description	Updated subsection 2.1 Overview.

Section	Change Summary
IP Parameter Description	Reworked subsection 2.3 *Attribute Summary* and renamed to section 3 IP Parameter Description.
Signal Description	Reworked subsection 2.2 *Signal Description* and moved to section 4 Signal Description.
Register Description	Reworked subsection 2.4 *Register Description* and moved to section 5 Register Description.

Designing with the IP	• Renamed section 3. *IP Generation, Simulation, and Validation* and moved to section 6 Designing with the IP.
	• Reworked subsection 3.1 *Generating the IP* and renamed to subsection 6.1 Generating and Instantiating the IP.
	• Added the following subsections:
	  • 6.2 Design Implementation
	  • 6.3 Timing Constraints
	  • 6.4 Physical Constraints
	  • 6.5 Specifying the Strategy
	• Reworked subsection 3.2 *Running Functional Simulation* and moved to subsection 6.6 Running Functional Simulation.

Debugging	Added this section.
Design Considerations	Added this section.
Appendix A. Resource Utilization	Updated resource utilization per the latest software version.
Appendix B. Limitations	Updated this section.
References	Updated references.

Revision 1.7, November 2022

Section	Change Summary
Functional Description	Added *txclk_hsgate_i* port information in Table 2.1. MIPI D-PHY Module Port Descriptions.
IP Generation, Simulation, and Validation	• Updated the heading of Section 3 from “IP Generation and Evaluation” to “IP Generation, Simulation, and Validation”. • Removed “Licensing the IP” section. • Updated the heading of Section 3.1 from “Generation and Synthesis” to “Generating the IP”. • Updated Figure 3.1. Module/IP Block Wizard, Figure 3.2. Configure Block of MIPI D-PHY Module, and Figure 3.3. Check Generated Result. • Updated the heading of Section 3.3 from “Hardware Evaluation” to “IP Evaluation”.

Revision 1.6, November 2021

Section	Change Summary
Functional Description	• Updated Table 2.1. MIPI D-PHY Module Port Descriptions. Added hs_rx_clk_en_i, hs_rx_data_en_i, and hs_data_des_en_i ports. • Updated Table 2.2. Attributes Table. Added packages under Dependency on Other Attributes for Interface Clock Frequency (MHz) and Interface Data Rate (Mbps). • Added the Testbench Control for Hard MIPI D-PHY Simulation section.
IP Generation and Evaluation	Added the Required Post Synthesis Constraints section.
Appendix A. Resource Utilization	Updated device, including Table A.2. Resource Utilization.

Revision 1.5, June 2021

Section	Change Summary
Introduction	• Removed Quick Facts section. • Updated Figure 1.1.
Functional Description	Updated the Register Description section.
Appendix A. Resource Utilization	Updated content, including Table A.1.

Section	Change Summary
References	Updated this section.

Revision 1.4, March 2021

Section	Change Summary
Functional Description	Updated Table 2.2 to add CSFBGA121 package information for Interface Clock Frequency and Interface Data Rate attribute.

Revision 1.3, December 2020

Section	Change Summary
Introduction	Added Table 1.1.
Functional Description	• Rename sync_clk to sync_clk_i, pll_clockop_i to pll_clkop_i and pll_clockos_i to pll_clkos_i. • Added hs_tx_clk_en_i, lp_tx_en_i and pll_lock_i ports. • Added Attribute Description table in Attribute Summary.
IP Generation and Evaluation	Updated this section.
Appendix A. Resource Utilization	Added resource utilization.
References	Updated this section.

Revision 1.2, June 2020

Section	Change Summary
All	• Added Certus-NX support. • Added Lattice Radiant Software 2.1 support.

Functional Description	• Added pll_clockop_i and pll_clockos_i ports in Table 2.1.
	• Updated Table 2.2.
	  • Added D-PHY Module Type dependency.
	  • Removed attributes.

IP Generation and Evaluation	• Updated Figure 3.1. • Updated procedure steps in Running Functional Simulation.
Appendix A. Limitations	Removed limitation.

Revision 1.1, February 2020

Section	Change Summary
Functional Description	• Added pll_clockop_i and pll_clockos_i ports in Table 2.1. MIPI D-PHY Module Port Description. • Added D-PHY PLL Mode attribute in Table 2.2. Attributes Summary.
IP Generation and Evaluation	Updated Figure 3.1. Configure Block if MIPI D-PHY.
Appendix A. Limitations	Changed MIXEL to Hard_IP.

Revision 1.0, December 2019

Section	Change Summary
All	Changed document status from Preliminary to final.
Acronyms in This Document	Added this section.
Functional Description	• Revised port names to lp_rx_data_p_o[BUS_WIDTH – 1:0] and lp_rx_data_n_o[BUS_WIDTH – 1:0] in Table 2.1. MIPI D-PHY Module Port Descriptions. • Revised 0x30 offset information and added footnote in Table 2.3. Configuration Registers (MIPI Programmable Bits).
IP Generation and Evaluation	• Updated source file in Running Functional Simulation procedure to tb_top.v.

Section	Change Summary
	• Updated Figure 3.5. Adding and Reordering Source.
Appendix A. Limitations	Added this section.
References	Removed reference to JEDEC website.
All	Minor editorial changes

www.latticesemi.com