MIPI D-PHY

Reference Design

FPGA-RD-02040-1.7

September 2020

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 and with all faults, and all risk associated with such information is entirely with Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. 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. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. 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.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
DCS	Display Command Set
EoT	End of Transmission
HS	High-Speed
LP	Low Power
LSE	Lattice Synthesis Engine
MIPI	Mobile Industry Processor Interface

1. Introduction

The Mobile Industry Processor Interface (MIPI) has become a specification standard for interfacing components in consumer mobile devices. A very popular MIPI bus, which provides high-speed connectivity, is called the D-PHY. The MIPI D-PHY specification provides a physical layer definition, which is typically used for camera and display interfacing. The MIPI D-PHY Reference Design allows moderate to advanced FPGA users the capability to receive and transmit data with respect to the MIPI D-PHY specification. Furthermore, the D-PHY Reference Design is the foundation for higher layer protocol designs such as MIPI CSI2 and MIPI DSI.

This reference design supports the following Lattice devices:

- MachXO2
- MachXO3
- LatticeECP3
- ECP5

2. Key Features

- Interfaces to MIPI CSI2, MIPI DSI, RX, and TX devices
- Supports Unidirectional HS (High Speed) 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

3. MIPI D-PHY Operational Overview

The MIPI D-PHY is a bus which incorporates one clock lane and from one to four data lanes. The clock and data lanes can switch between two 1.2 V LVCMOS signals or one differential SLVS200 pair. Operating in differential mode is referred to as HS mode. In HS mode, video data is delivered over a differential pair. For example, video data being sent from an image sensor is sent in HS mode. The clock is center aligned with the data in HS mode.

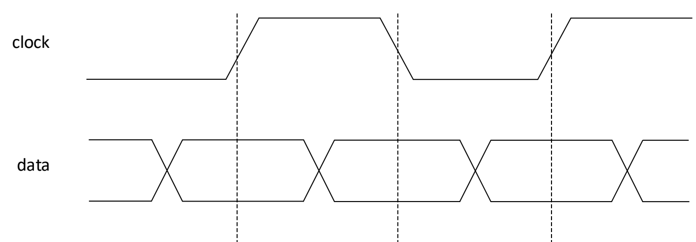


Figure 3.1. D-PHY Clock to Data Relationship for HS Mode

Depending on the application, the HS mode may be utilized at all times or the D-PHY can switch from HS differential lanes to single ended. When the D-PHY is sending single ended data, this is called LP mode. In Camera and Display applications, LP is entered during the blanking period to reduce power. Additionally, in Display applications, LP mode is used for configuration of the screen.

4. Functional Description

The MIPI D-PHY Reference Design is provided in two Lattice Diamond® projects. Each project instantiates the receiving or the transmitting D-PHY modules individually. For the purpose of this document the RX Reference Design refers to the design, which receives HS data and the TX Reference Design refers to the design, which transmits HS data. In both designs, LP data can be transferred bi-directionally on any data or clock lane. HS data is de-serialized/serialized to and from single data rate byte packets respectively. LP data is provided as a two-bit interface for individual extraction and control of the P and N wire pair. An external resistor network is needed on the interface pins of the FPGA. This resistor network is different for RX and TX modules. Both modules support bidirectional LP communication and unidirectional HS communication.

5. Receiving Interface

The D-PHY RX Reference Design gives users the ability to receive HS data on one clock lane and up to four data lanes. Each clock and data lane uses a total of four I/O. Two I/O pins are used to receive the HS data with the LVDS25 I/O type, which configures them as differential. The LVDS25 I/O pair must be used in order to handle the 200 mV common mode voltage. The other two I/O provide 50 Ω single ended termination by driving each LVCMOS12 signal to 0 V while in HS mode. Additionally, the LVCMOS12 signals are configured to transmit and receive bidirectional CMOS data during LP mode.

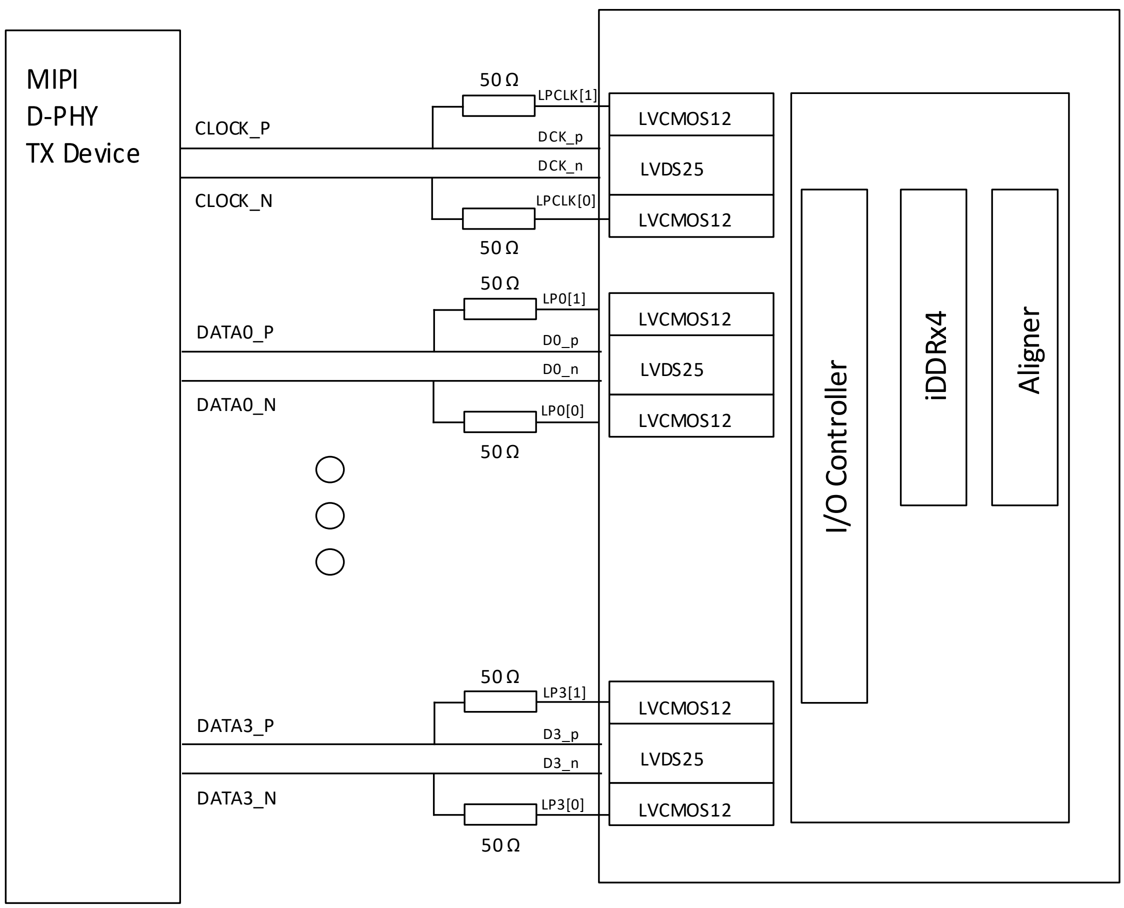


Figure 5.1. Unidirectional Receive HS Mode and Bidirectional LP Mode Interface Implementation

In some applications, LP mode is not needed. If this is the case only two I/O are needed and a single 100 Ω parallel termination resistor can be used. If the Lattice FPGA being used has built in 100 Ω termination, this can be used in this case as well. See Figure 5.2 for this simplified interface.

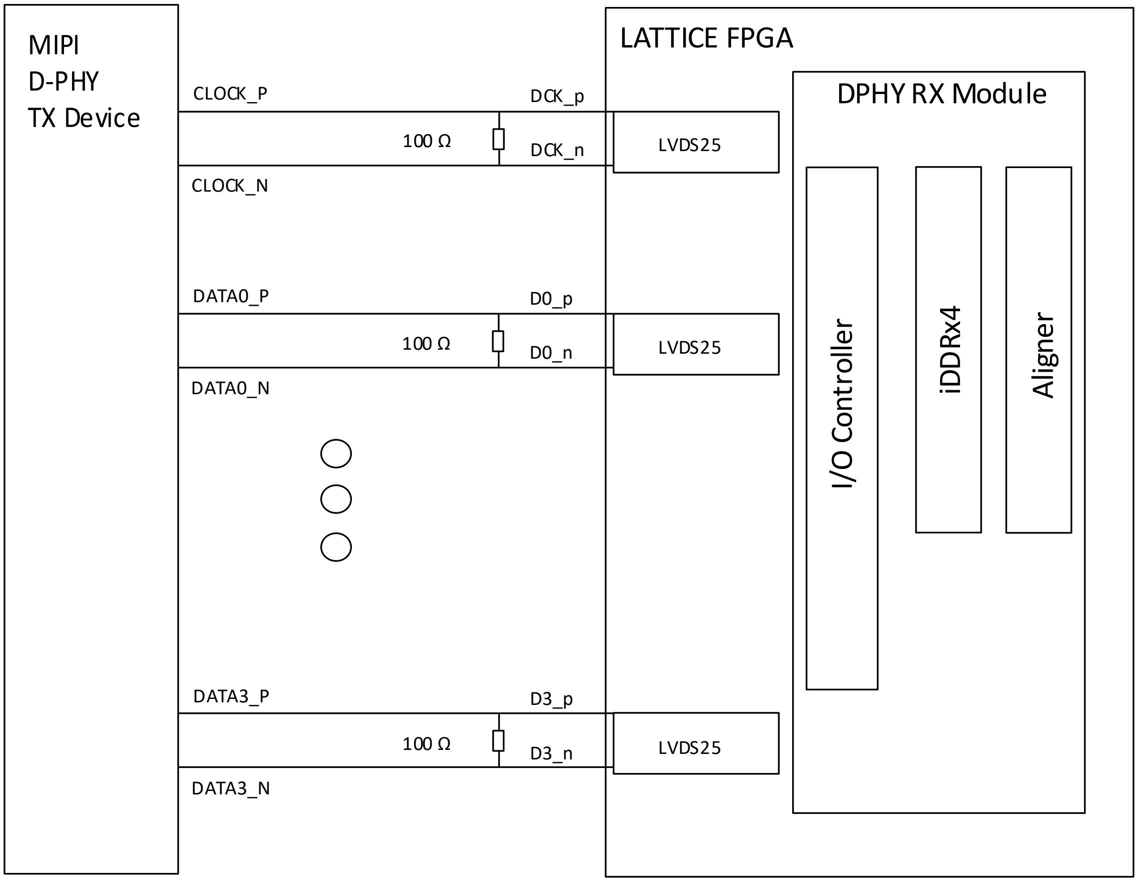


Figure 5.2. Unidirectional Receive HS Mode and Bidirectional LP Mode Interface Implementation

Within the D-PHY RX module, HS data is de-serialized using iDDR gearbox primitives. The MachXO2™ design uses the iddrx4 gearbox primitive, which derives a divide-by-4 clock and 8-bit byte data directly. LatticeECP3™ FPGA's use the iddrx2 gearbox primitives with additional down conversion logic to derive this same divide-by-4 clock and 8- bit data bus. The number of HS data lanes used can be controlled by `define compiler directives. Options are 1, 2, 3 or 4 data lanes. The HS clock lane is available at all times.

The MIPI byte clock only runs while in HS mode. The MIPI clock is not continuous unless the user places the transmitting device in "free-running" clock mode. This reference design can run with or without a "free-running" clock. However, it is important to understand and consider design accommodations that need to take place in order to continue the processing of data when the clock lane is in LP mode.

If a PLL is expected to be used in a particular RX D-PHY design, a continually running clock needs to be obtained in some way since PLL in the FPGA will take as much as milliseconds to lock. Obtaining a continually running clock can be achieved in multiple ways. One common practice is to place the transmitting device in free-running clock mode. This causes the clock lane to never enter LP mode and remain continually operating in HS mode. For both CSI-2 and DSI, there is a requirement in the specification that all transmitting and receiving devices must have the capability of operating with a continually running clock. Therefore, running the clock lane in LP mode is a user preference. If it is desired to have the clock lane enter LP mode, then the second common practice is to use a secondary clock source that is either on the same clock domain or very close to the same frequency as the MIPI clock. The data on the MIPI clock is then converted to the secondary clock domain using a crossing clock domain FIFO. The third common practice is to use a clock mux with a secondary clock very close to the HS clock and switch between clocks during LP and HS mode transitions in attempts to keep the PLL locked.

After the data is de-serialized the 8-bit data is byte and lane aligned so that MIPI byte data is available on each byte clock cycle. This alignment is done based on the recognition of the MIPI HS_Sync sequence, which is transmitted on all data lanes one clock cycle before the packet header. hs_en is used to reset the alignment module. When hs_en is *low* the word alignment is reset; when *high* the word aligner looks for the next HS_Sync sequence seen. The sync signal will initially be *low*. The sync signal will go high when the HS_Sync sequence is detected and the byte data at the output of the aligner are properly aligned.

The aligner module consists of two subsidiary modules. The first module byte aligns the 8-bit data from the de-serializer. The second module aligns each of the data lanes to each other. In some cases, lane alignment or lane and word alignment is not needed. `define compiler directives allow the user to turn on and off the word and lane alignment features.

HS termination is controlled by the term_en signal of the IO_Controller module. Although there is no direct contention detection mechanism with this design, enabling the termination can be done in a number of ways. One way is to use the HS clock to observe the LP to HS data transition on one of the data lines since the clock lane will enter HS mode sooner and exit HS mode later than the data lines.

Another option is to initializing the LP signals as inputs at startup and watch for the LP to HS on the individual clock and data lanes. Once the sequence is detected term_en can be set 'low' by the user enabling the termination. Once enabled, the HS data can be observed for any conditions desired to leave HS mode. Condition examples would be EoT (End of Transmission Packets), observation of the end of a packet with no consecutive bursting packet, bad ECC or checksum, timeout if no packet header or HS-Sync sequence is seen after a certain amount of time, etc.

The IO_Controller module also controls LP signals. Each data lane has a lp*_dir signal which controls the direction of the LP data between the transmitting device and the FPGA. `define compiler directives allow users to turn on/off LP I/O for each clock and data lane individually. This can be handy if the user only needs LP mode for one or two MIPI D-PHY data lanes. The LP signals are defined as two bit busses. Although there is no difference between signal 1 and 0 of a bus, signal 1 is typically connected to the P wire side and 0 to the N wire side. This is simply to keep consistency with the LP transition identification scheme.

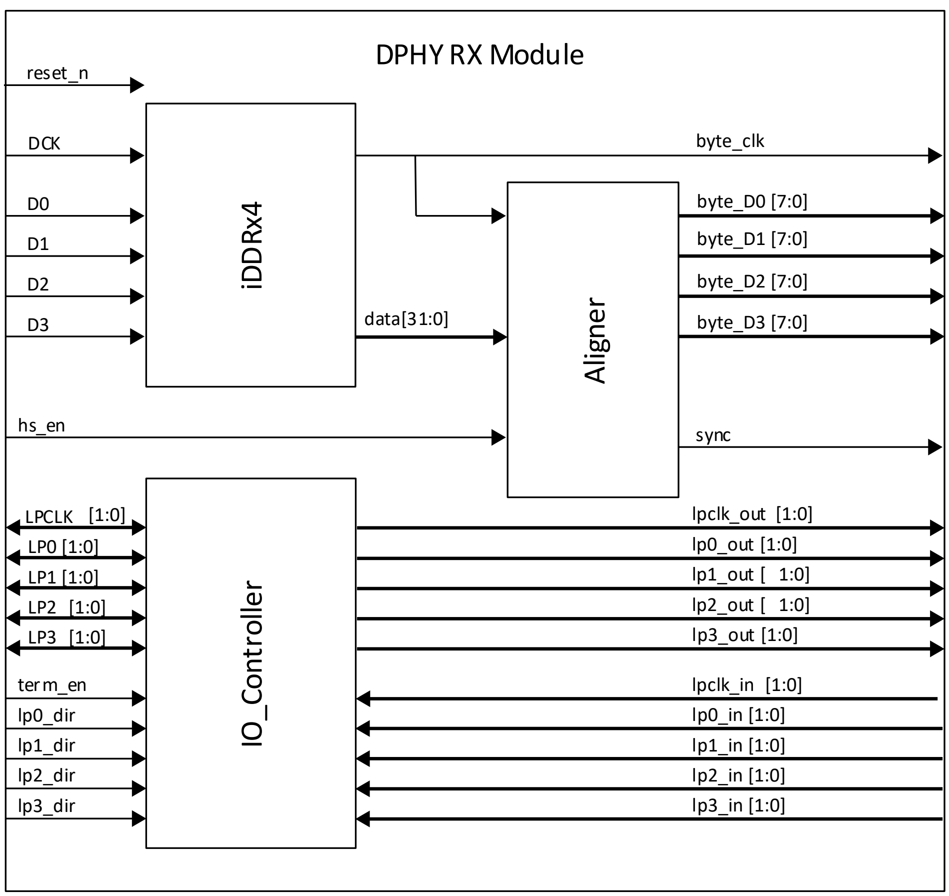


Figure 5.3. D-PHY RX Module Functional Block Diagram

Table 5.1. D-PHY RX Compiler Directives

Signal	Description
`define HS_3	Generates I/O for four HS data lanes
`define HS_2	Generates I/O for four HS data lanes Overridden if HS_3 is defined
`define HS_1	Generates I/O for four HS data lanes Overridden if HS_3 or HS_2 is defined
`define HS_0	Generates I/O for four HS data lanes Overridden if HS_3, HS_2, or HS_1 is defined
`define LP_CLK	Generates I/O for LP mode on clock lane
`define LP_0	Generates I/O for LP mode on data lane 0
`define LP_1	Generates I/O for LP mode on data lane 1
`define LP_2	Generates I/O for LP mode on data lane 2
`define LP_3	Generates I/O for LP mode on data lane 3
`define HS_3	Generates I/O for four HS data lanes
`define HS_2	Generates I/O for four HS data lanes Overridden if HS_3 is defined
`define HS_1	Generates I/O for four HS data lanes Overridden if HS_3 or HS_2 is defined
`define HS_0	Generates I/O for four HS data lanes

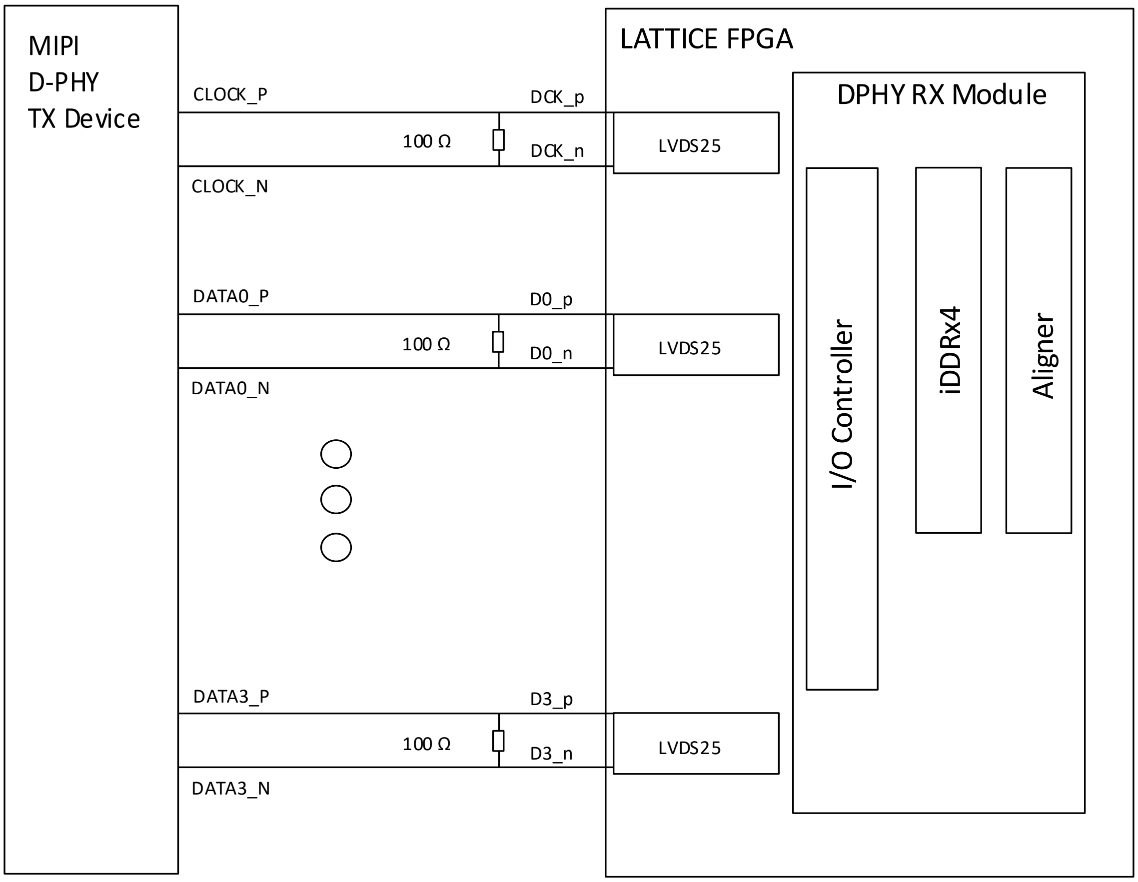
Table 5.2. D-PHY RX Module I/O List

Signal	Direction	Description
reset_n	Input	Resets module (Active *low*)
DCK	Input	HS Clock
D0	Input	HS Data lane 0
D1	Input	HS Data lane 1
D2	Input	HS Data lane 2
D3	Input	HS Data lane 3
hs_en	Input	Initializes word aligner to align on next HS-Sync Sequence
byte_clk	Output	Byte Clock = DCK/4
byte_D0 [7:0]	Output	Byte data, data lane 0
byte_D1 [7:0]	Output	Byte data, data lane 1
byte_D2 [7:0]	Output	Byte data, data lane 2
byte_D3 [7:0]	Output	Byte data, data lane 3
sync	Output	Active *high* when byte data is aligned
LPCLK [1:0]	Bidirectional	LP clock lane; LPCLK[1] = P wire, LPCLK[0] = N wire
LP0 [1:0]	Bidirectional	LP data lane 0; LP0[1] = P wire, LP0[0] = N wire
LP1 [1:0]	Bidirectional	LP data lane 1; LP1[1] = P wire, LP1[0] = N wire
LP2 [1:0]	Bidirectional	LP data lane 2; LP2[1] = P wire, LP2[0] = N wire
LP3 [1:0]	Bidirectional	LP data lane 3; LP3[1] = P wire, LP3[0] = N wire
term_en	Input	Enables termination by setting LP signals at outputs and 'low' Overrides lp_dir control signal
lpclk_dir	Input	Controls the direction of LP data 0 – LP data receive 1 – LP data transmit
lp0_dir	Input	Controls the direction of LP data 0 – LP data receive 1 – LP data transmit

Signal	Direction	Description
lp1_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lp2_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lp3_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lpclk_out [1:0]	Output	LP data receive Available when lp_dir = *0* and term_en = *0*
lp0_out [1:0]	Output	LP data receive Available when lp_dir = *0* and term_en = *0*
lp1_out [1:0]	Output	LP data receive Available when lp_dir = *0* and term_en = *0*
lp2_out [1:0]	Output	LP data receive Available when lp_dir = *0* and term_en = *0*
lp3_out [1:0]	Output	LP data receive Available when lp_dir = *0* and term_en = *0*
lpclk_in [1:0]	Input	LP data transmit Available when lp_dir = *1* and term_en = *0*
lp0_in [1:0]	Input	LP data transmit Available when lp_dir = *1* and term_en = *0*
lp1_in [1:0]	Input	LP data transmit Available when lp_dir = *1* and term_en = *0*
lp2_in [1:0]	Input	LP data transmit Available when lp_dir = *1* and term_en = *0*
lp3_in [1:0]	Input	LP data transmit Available when lp_dir = *1* and term_en = *0*

6. Transmitting Interface

The transmitting interface gives users the ability to utilize one clock lane and up to four data lanes. Each lane uses a total of four I/O. Two I/O pins are used to transmit the HS data with the LVDS25E I/O type, which configures the output pins as differential pairs. The other two I/O are used to provide a voltage dividing circuit while in HS mode and to transmit or receive 1.2 V CMOS data in LP mode. See Figure 6.1. Note that in some applications LP mode is not needed. If this is the case only two FPGA I/O are needed instead of four. The 50 Ω resistors can be replaced with 70 Ω resistors tied to the ground, in place of the 1.2 V of the LVCMOS I/O connections. See Figure 6.2.



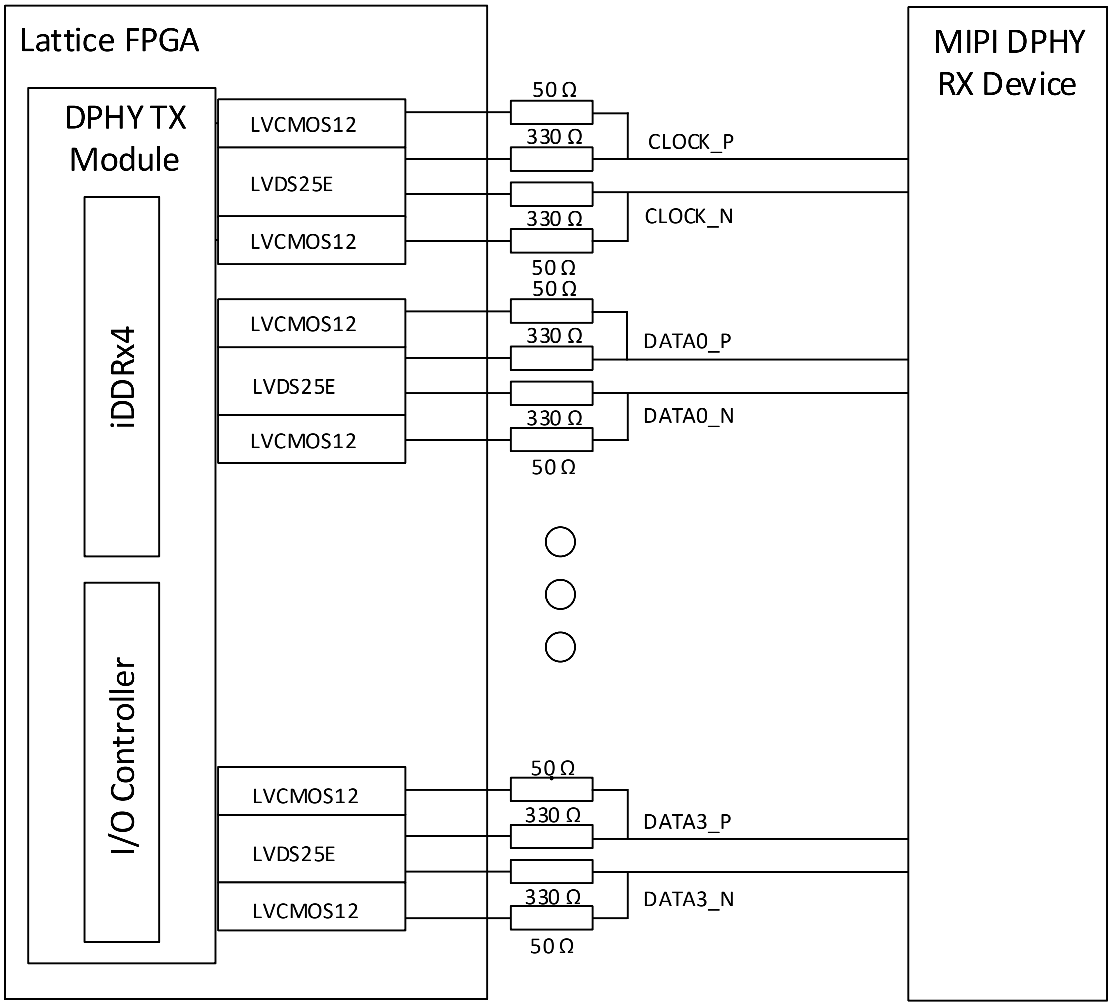


Figure 6.1. Unidirectional Transmit HS Mode and Bidirectional LP Mode Interface Implementation

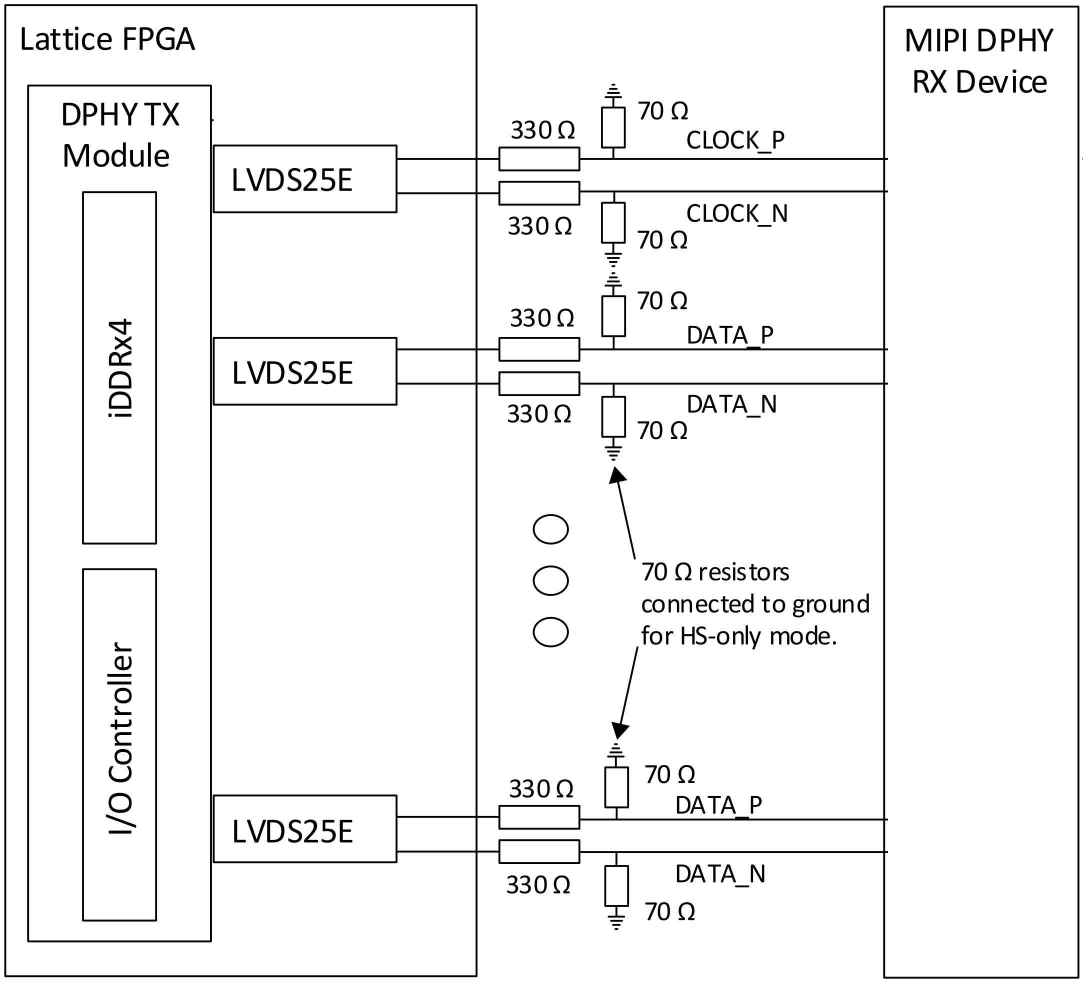


Figure 6.2. Unidirectional Transmit HS Mode Only Implementation

Within the D-PHY TX module HS data is serialized using oddr gearbox primitives. The MachXO2 design uses the oddrx4 gearbox primitive. LatticeECP3 FPGA's use the oddrx2 gearbox primitives with additional up conversion logic to serialize the byte data. MIPI D-PHY data is center aligned and therefore a PLL with a 0 and 90 degree phase shifting is used for the HS data and clock respectively.

The IO_Controller module controls the HS and LP data traffic. The hs_clk_en and hs_data_en signals place the clock and data lanes in HS mode when active 'high'. While in HS mode the I/O controller sets the CMOS signals low to create a voltage divider network on the emulated LVDS output signals to achieve a 200 mV common mode volt- age. When hs_clk_en or hs_data_en is 'low', the LVDSE I/O is set to high impedance so it does not interfere with LP data transmissions. There is an hs_clk_en control signal and an hs_data_en signal, because the MIPI specification defines the clock lane going in to and out of HS mode before and after the data lanes. The lp_dir signal controls the LP mode direction. The lp*_dir control signal is overridden when hs_*_en='1'. The IO_Controller module also controls LP data traffic while in LP mode. lp*_dir signals control the direction of LP data being transmitted or received.

Compiler directives control the I/O available for the HS data lanes as well as which lanes will have LP control I/O. The LP signals are defined as two bit busses. Although there is no difference between signal 1 and 0 of a bus, signal 1 is typically connected to the P wire side and 0 to the N wire side. This is simply to keep consistency with the LP transition identification scheme.

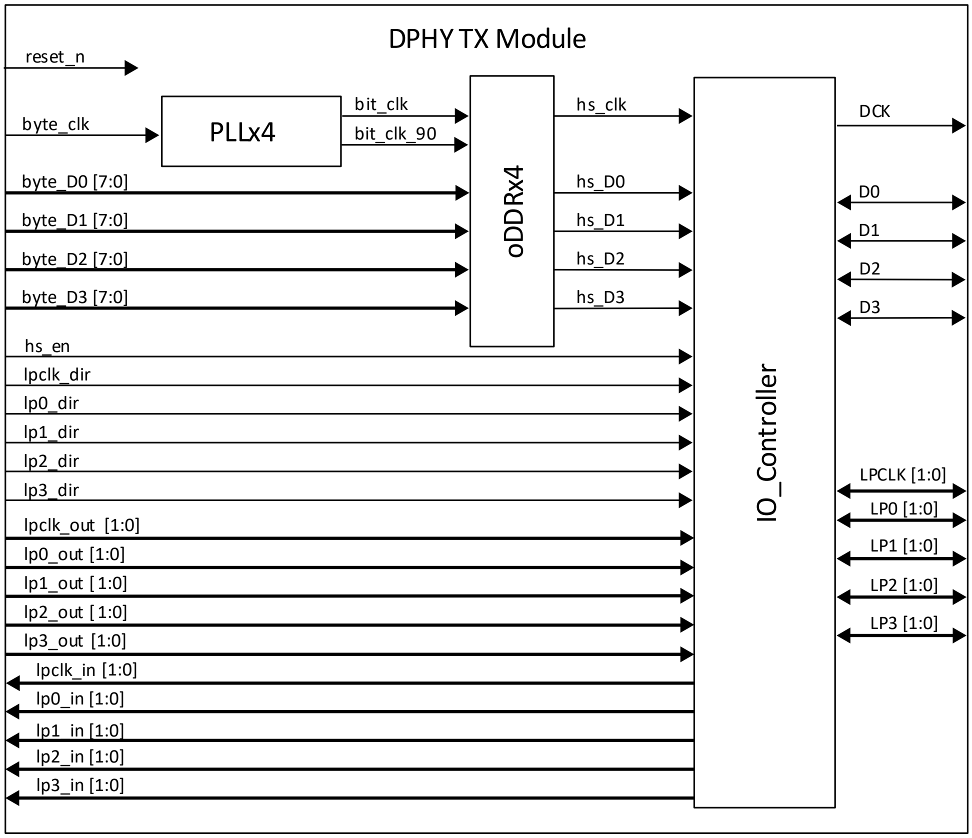


Figure 6.3. D-PHY TX Module Functional Block Diagram

Table 6.1. D-PHY TX Compiler Directives

Signal	Description
`define HS_3	Generates I/O for four HS data lanes
`define HS_2	Generates I/O for four HS data lanes Overridden if HS_3 is defined
`define HS_1	Generates I/O for four HS data lanes Overridden if HS_3 or HS_2 is defined
`define HS_0	Generates I/O for four HS data lanes Overridden if HS_3, HS_2, or HS_1 is defined
`define LP_CLK	Generates I/O for LP mode on clock lane
`define LP_0	Generates I/O for LP mode on data lane 0
`define LP_1	Generates I/O for LP mode on data lane 1
`define LP_2	Generates I/O for LP mode on data lane 2
`define LP_3	Generates I/O for LP mode on data lane 3
`define HS_3	Generates I/O for four HS data lanes
`define HS_2	Generates I/O for four HS data lanes Overridden if HS_3 is defined
`define HS_1	Generates I/O for four HS data lanes Overridden if HS_3 or HS_2 is defined
`define HS_0	Generates I/O for four HS data lanes

Table 6.2. D-PHY TX Module I/O List

Signal	Direction	Description
reset_n	Input	Resets module (Active 'low')
DCK	Output	HS Clock
D0	Output	HS Data lane 0
D1	Output	HS Data lane 1
D2	Output	HS Data lane 2
D3	Output	HS Data lane 3
byte_clk	Input	Byte Clock = DCK/4
byte_D0 [7:0]	Input	Byte data, data lane 0
byte_D1 [7:0]	Input	Byte data, data lane 1
byte_D2 [7:0]	Input	Byte data, data lane 2
byte_D3 [7:0]	Input	Byte data, data lane 3
LPCLK [1:0]	Bidirectional	LP clock lane; LPCLK[1] = P wire, LPCLK[0] = N wire
LP0 [1:0]	Bidirectional	LP data lane 0; LP0[1] = P wire, LP0[0] = N wire
LP1 [1:0]	Bidirectional	LP data lane 1; LP1[1] = P wire, LP1[0] = N wire
LP2 [1:0]	Bidirectional	LP data lane 2; LP2[1] = P wire, LP2[0] = N wire
LP3 [1:0]	Bidirectional	LP data lane 3; LP3[1] = P wire, LP3[0] = N wire
hs_clk_en	Input	Enable HS clock on output, Sets LPCLK signals 'low' Overrides lpclk_dir control signal
hs_data_en	Input	Enable HS clock on output, Sets LP0, LP1, LP2 and LP3 signals 'low' Overrides lp0_dir - lp3_dir control signals
lpclk_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lp0_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lp1_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lp2_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lp3_dir	Input	Controls the direction of LP data *0* – LP data receive *1* – LP data transmit
lpclk_out [1:0]	Input	LP data receive Available when lp_dir = *0* and hs_en = *0*
lp0_out [1:0]	Input	LP data receive Available when lp_dir = *0* and hs_en = *0*
lp1_out [1:0]	Input	LP data receive Available when lp_dir = *0* and hs_en = *0*
lp2_out [1:0]	Input	LP data receive Available when lp_dir = *0* and hs_en = *0*
lp3_out [1:0]	Input	LP data receive Available when lp_dir = *0* and hs_en = *0*
lpclk_in [1:0]	Output	LP data transmit Available when lp_dir = *1* and hs_en = *0*
lp0_in [1:0]	Output	LP data transmit Available when lp_dir = *1* and hs_en = *0*

Signal	Direction	Description
lp1_in [1:0]	Output	P data transmit Available when lp_dir = *1* and hs_en = *0*
lp2_in [1:0]	Output	LP data transmit Available when lp_dir = *1* and hs_en = *0*
lp3_in [1:0]	Output	LP data transmit Available when lp_dir = *1* and hs_en = *0*

7. Packaged Design

Reference Designs are available for MachXO2, MachXO3L, LatticeECP3 and ECP5™ programmable devices. The *rd1182_mipi_dphy_reference_design* package contains designs for all of these devices. The *source* folder contains Verilog source code. Much of the code for LatticeECP3, ECP5, MachXO2 and MachXO3L is shared and utilizes the exact same Verilog source code files. Source code that is device specific is contained within a subfolder labelled *ecp3*, *ecp5*, *xo2* or *xo3l*. A verilog testbench is provided in the *testbench* folder for RX and TX D- PHY designs. The *ecp3*, *ecp5*, *xo2*, and *xo3l* folders contain the RX and TX project subfolders for the LatticeECP3, ECP5, MachXO2 and MachXO3L designs respectively. Each RX and TX design contains a Lattice Diamond project directory labeled *dphy_rx* and *dphy_tx*. Simulation project directory labeled *simulation* contains VO files for simulation. Note that it is recommended that you access the simulation through Lattice Diamond software. This is described further in the Simulation section of this document.

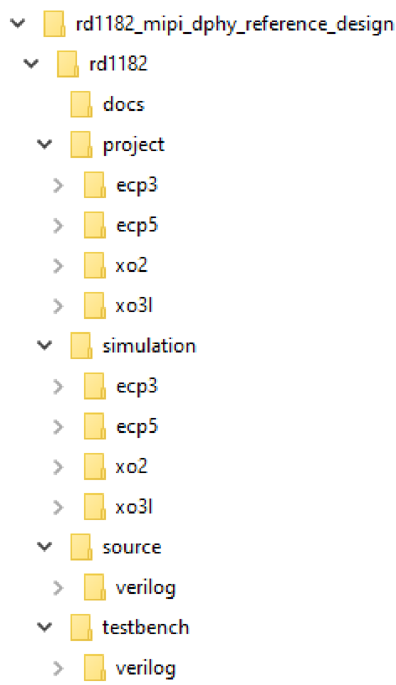


Figure 7.1. MIPI D-PHY Reference Design File Directory Structure

Table 7.1. MIPI D-PHY Reference Design File Directory Summary

Folder	Summary
rd1182_mipi_dphy_reference_design	Main file directory containing reference design
docs	Contains RD documents
project	Contains LatticeECP3, ECP5, MachXO2, and MachXO3L projects
simulation	Contains do files and VO files
source	Contains Verilog source code
testbench	Contains Verilog testbench

8. Simulation

A simulation project and testbench is available for RX and TX devices. The simulation environment can be accessed by double clicking on the Simulation.spf script file in Lattice Diamond from the file list. Aldec ActiveHDL then opens after clicking OK to the pop-up windows. Compile the project and initialize the simulation. Add signals to the waveform viewer that are desired to be viewer and run the simulation.

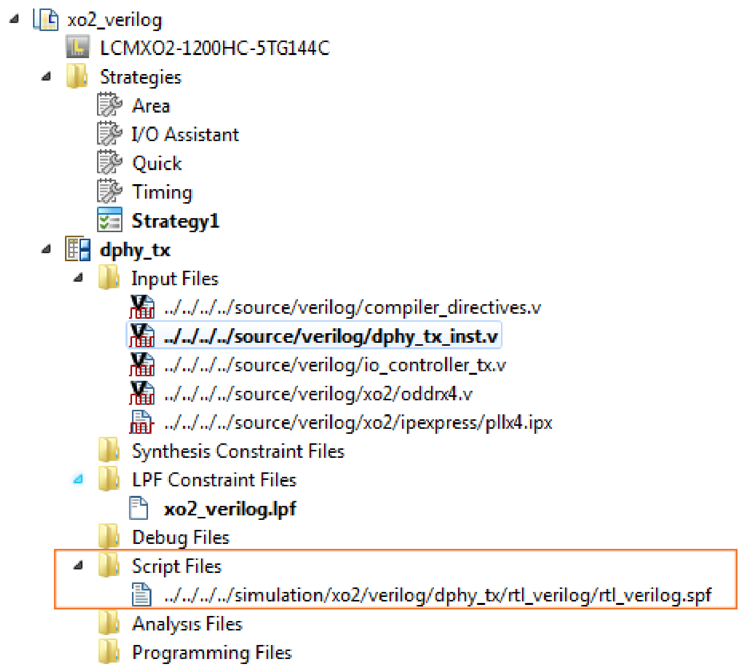
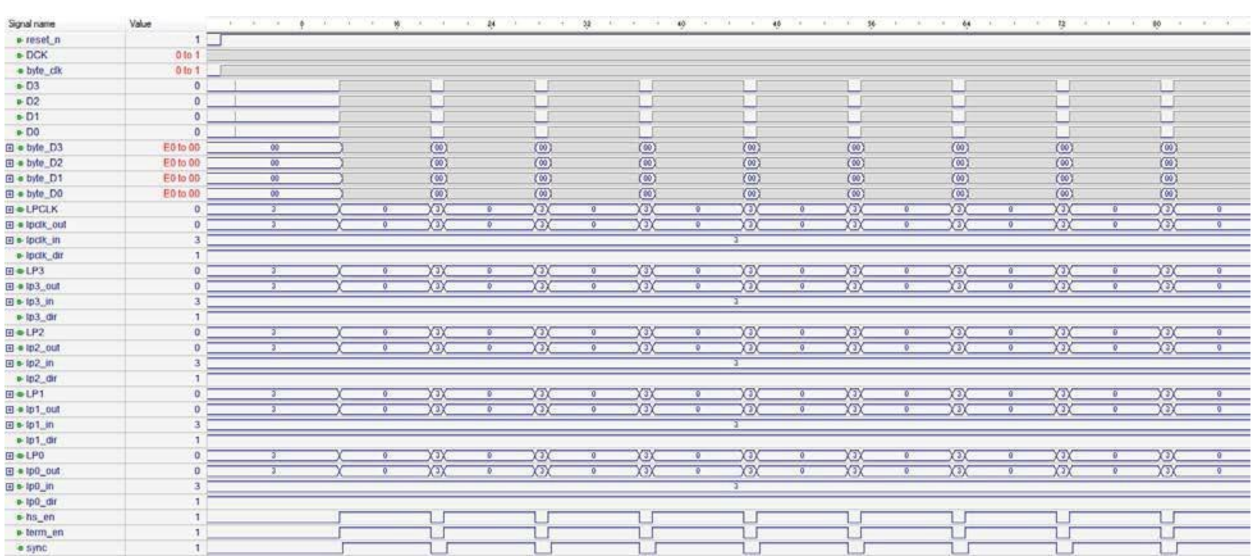


Figure 8.1. Simulation Wizard Script File Access

The simulation testbench sends data typical of what would be seen on a MIPI Data transmission. The testbench is configured for four HS and LP data lanes and an HS and LP clock lane. For the RX D-PHY Reference Design simulation, serial data is generated in the testbench. It is then decoded by the reference design. For the TX D-PHY Reference Design, byte data is generated by the testbench, which is then serialized by the reference design.



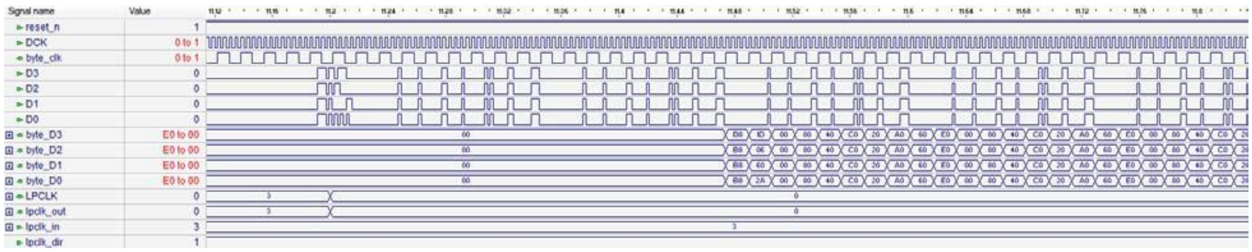


Figure 8.2. RX Simulation

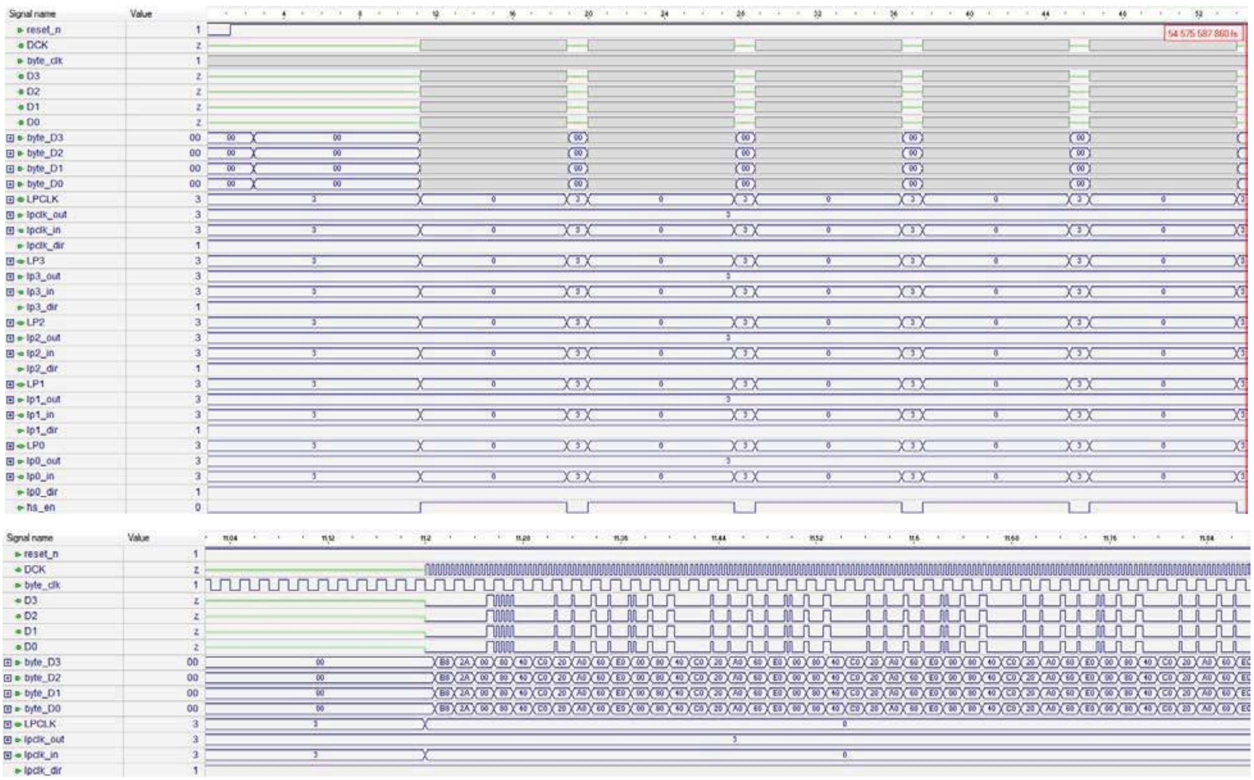


Figure 8.3. TX Simulation

9. IBIS Simulation

IBIS Functional Simulation is used to ensure proper differential and common mode voltage levels for RX and TX resistor network setups.

The RX resistor network is simulated to ensure proper functionality in various LP and HS modes. The resistor network for the HS mode is simulated at 375 MHz using a CSI2 TX IBIS model to drive the waveform.

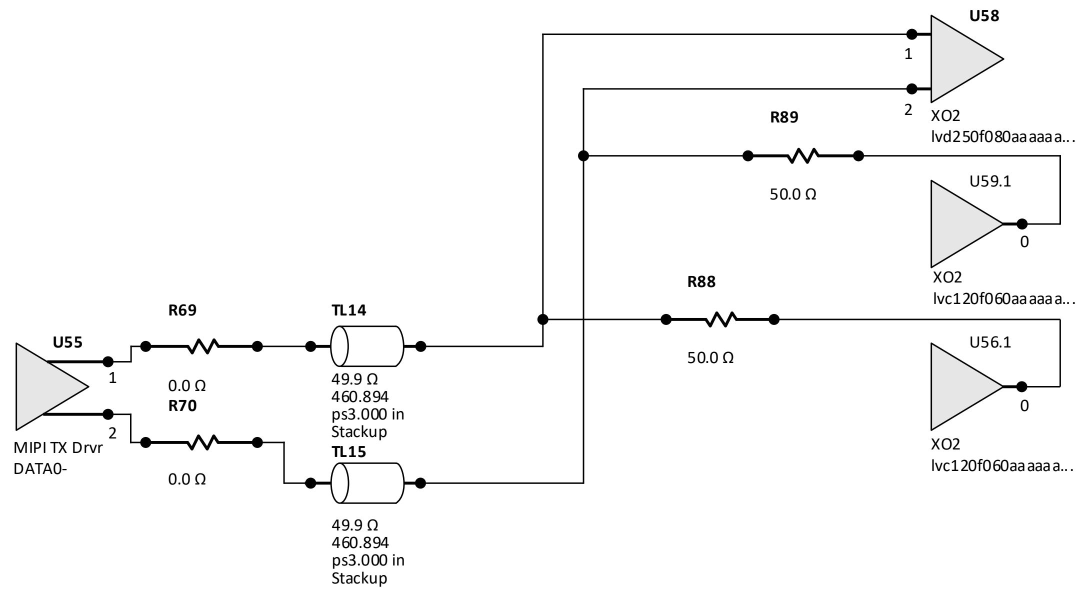


Figure 9.1. IBIS RX, HS-Mode Simulation Circuit

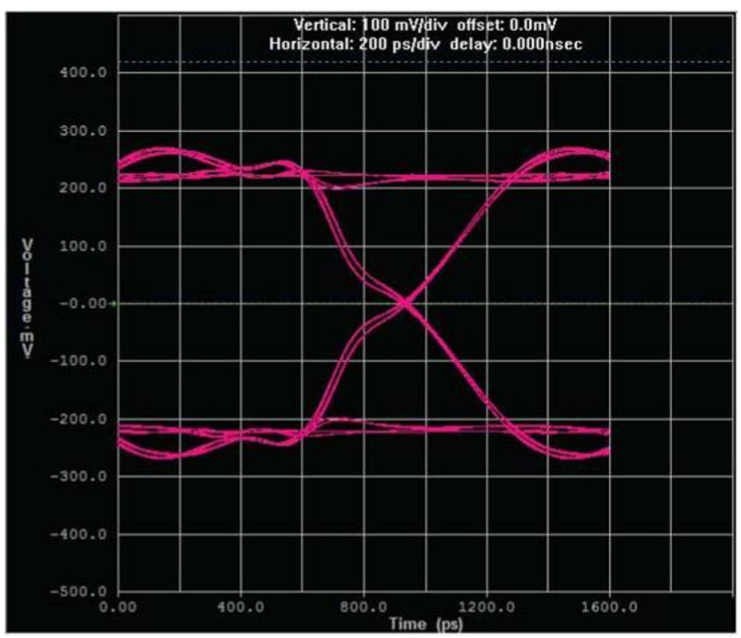


Figure 9.2. IBIS RX, HS-Mode Simulation Eye Waveform

The TX resistor network is simulated to ensure proper functionality in various LP and HS modes. Most importantly, HS mode is simulated based on the recommended TX test setup in the MIPI D-PHY specification under typical conditions.

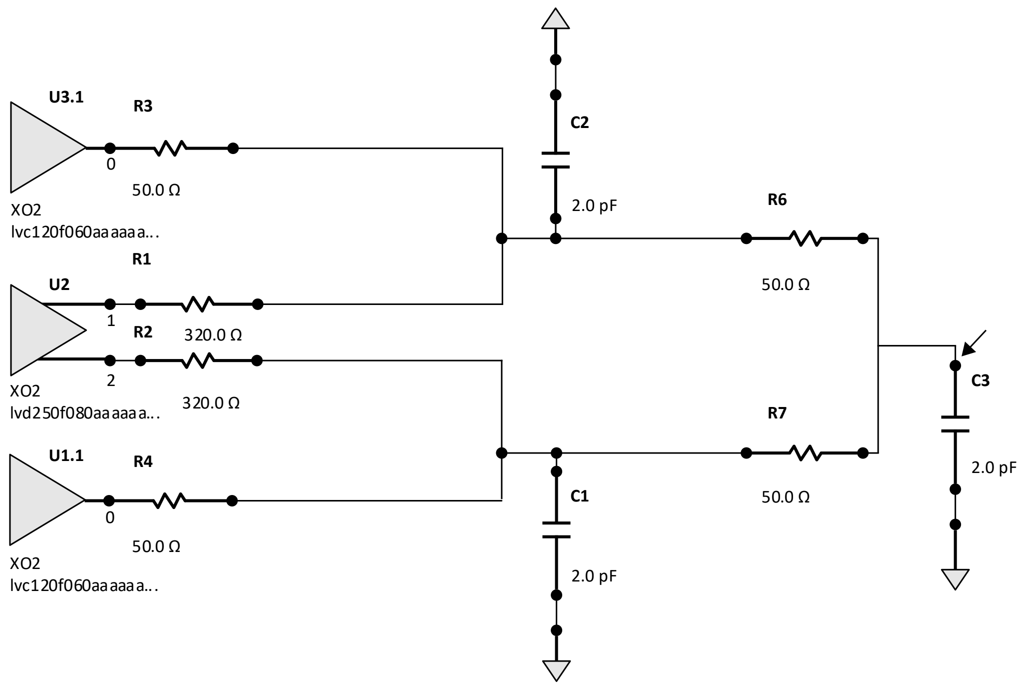


Figure 9.3. IBIS TX, HS-Mode Simulation Circuit

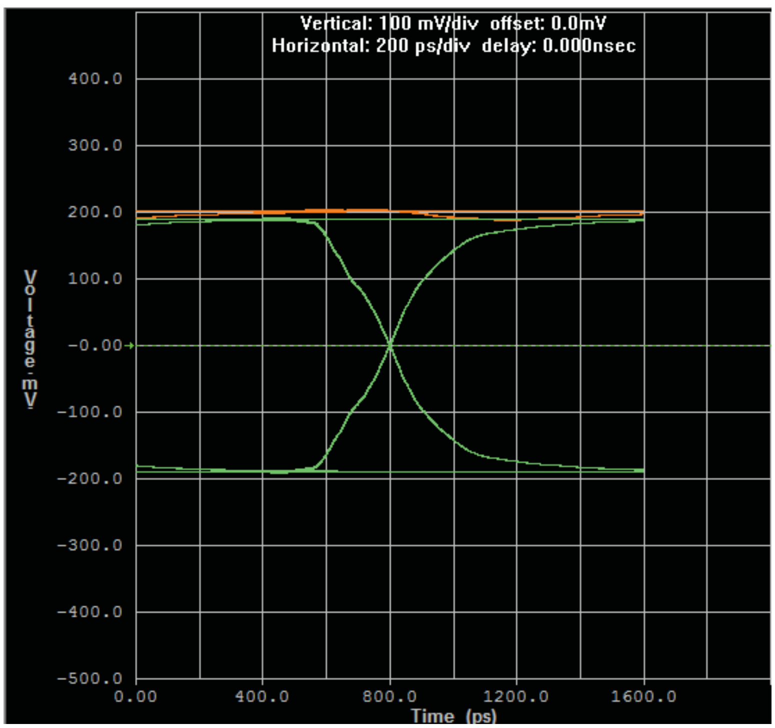


Figure 9.4. IBIS TX, HS-Mode Simulation Eye Waveform

Tested with MachXO2 Drivers at 375 MHz under typical conditions. The waveform in green represents the simulated differential voltage. The waveform in orange represents the simulated common mode voltage.

10. Hardware Analysis

Hardware Analysis is performed on the RX and TX D-PHY Reference Design using a Snapdragon S4 Plus APQ8060A Development Platform and a MachXO2-4000 adapter board. The RX and TX Reference Designs are connected back to back. MIPI DSI data is received from the APQ8060 to the MachXO2 device. The RX Reference Design de-serialized, word aligned, and lane aligned the two MIPI D-PHY data lanes. The aligned data is then sent to the TX Reference Design, which serializes and sends the data to output pins. This data is then transmitted from the MachXO2 to the Wintek DSI display that comes with the development platform. This test verified both LP and HS communication. The LP data is received and transmitted on one clock lane and two data lanes. Also, data lane 0 configured the panel with MIPI DCS (Display Command Set) commands. The HS data is received and transmitted on two data lanes and one clock lane.

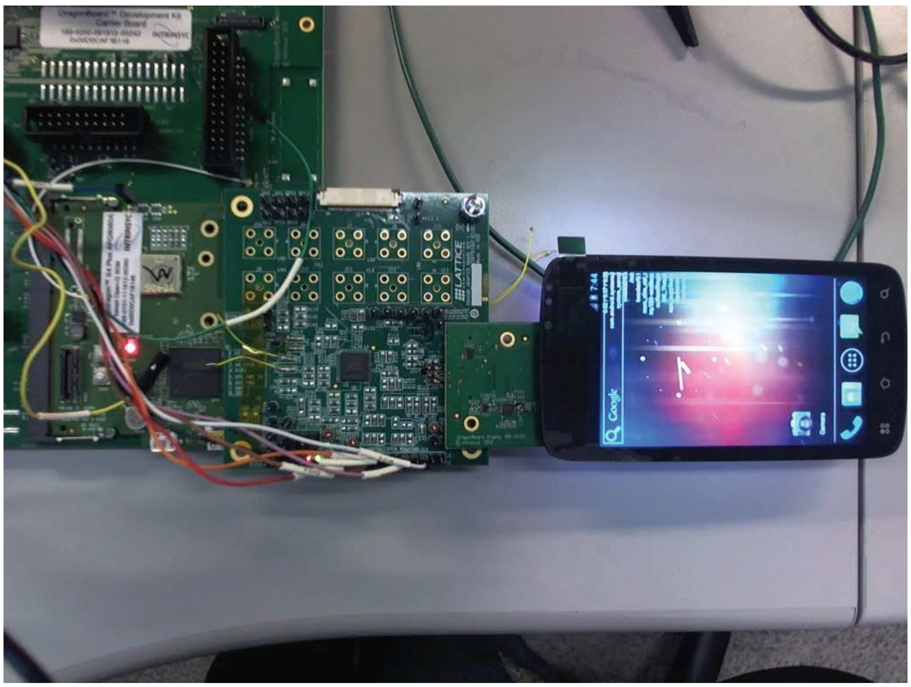
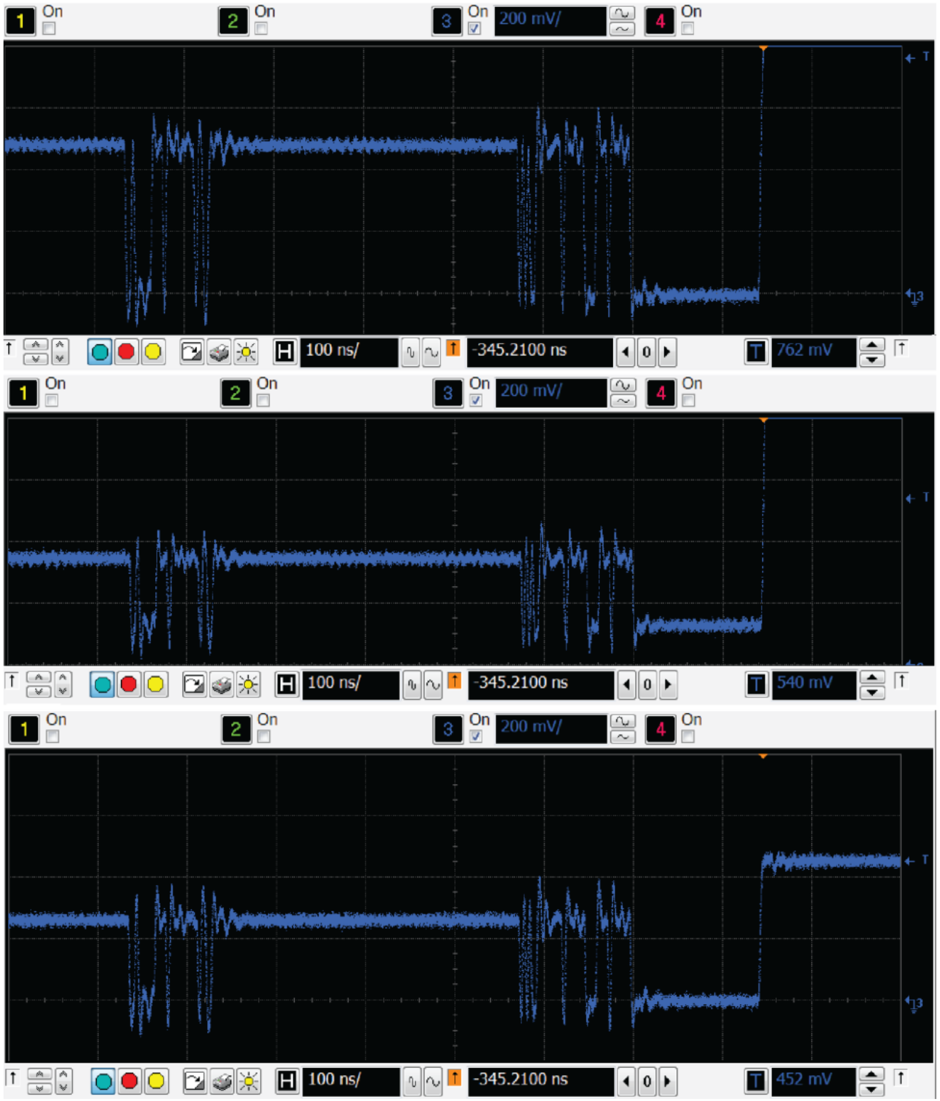


Figure 10.1. IBIS TX, HS-Mode Simulation Eye Waveform

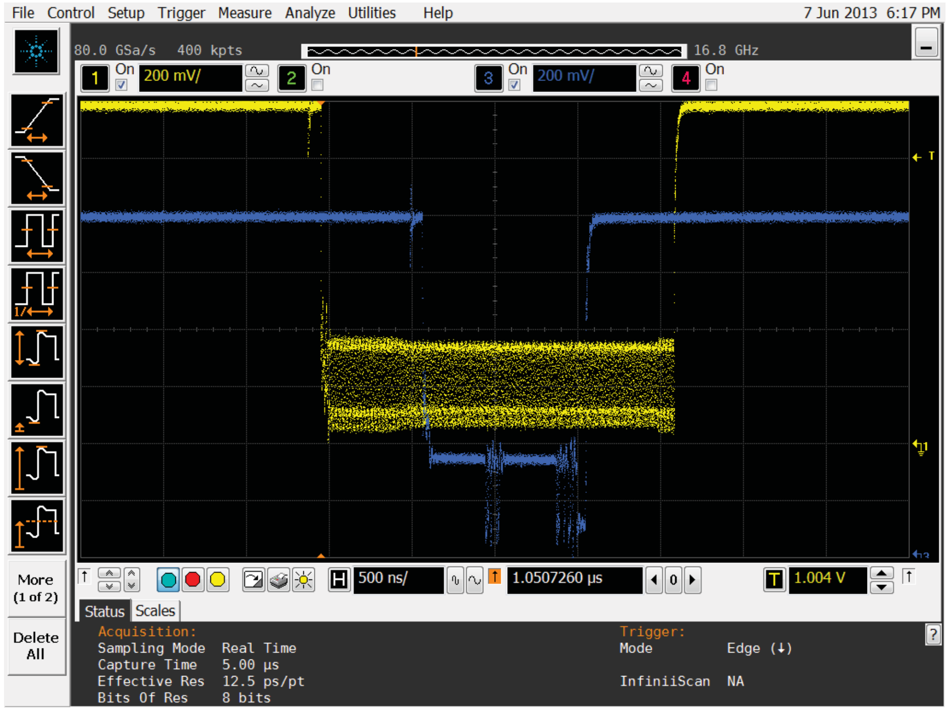
Receiving termination options are tested by viewing data transmissions with and without 50 Ω and 100 Ω terminations. When 50 Ω terminations released by setting CMOS signals to inputs, LP data could be captured at 1.2 V.



Top - No Termination. Middle - Internal 100 Ω Parallel Termination.
Bottom - 50 Ω Single Ended Termination. Captured with 500 MHz, single ended probe.

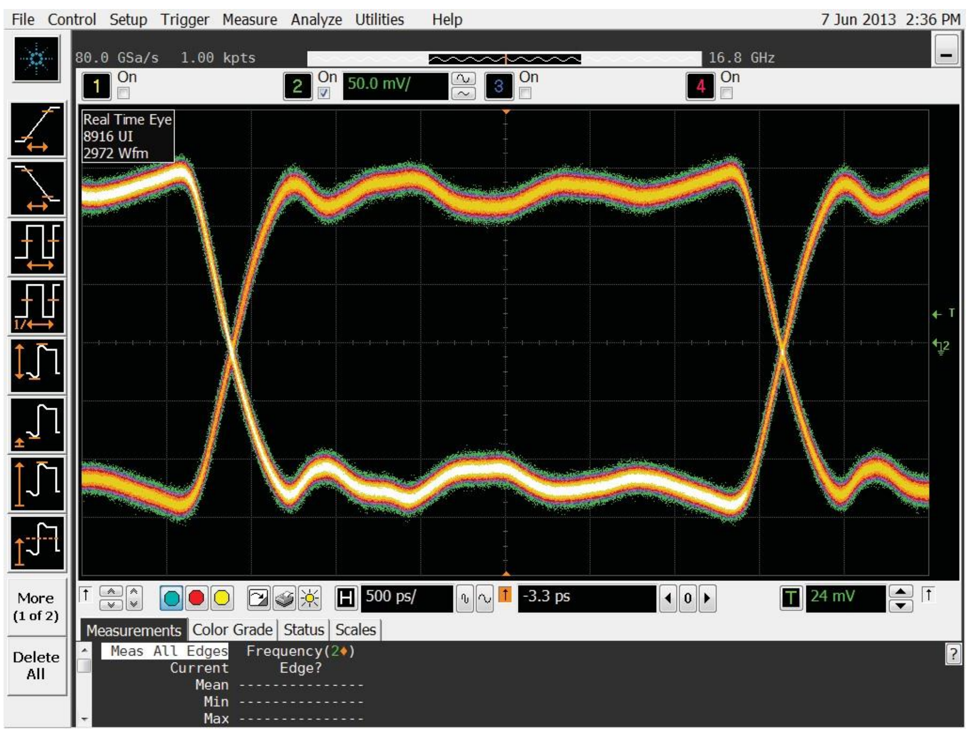
Figure 10.2. Receiving HS Data with No Termination, Internal 100 Ω Parallel Termination and 50 Ω Single Ended Termination

The transmission interface is tested to ensure HS and LP data could be delivered at the appropriate voltages. HS Data is tested to ensure a 200 mV common mode voltage at a +–100 mV differential voltage.

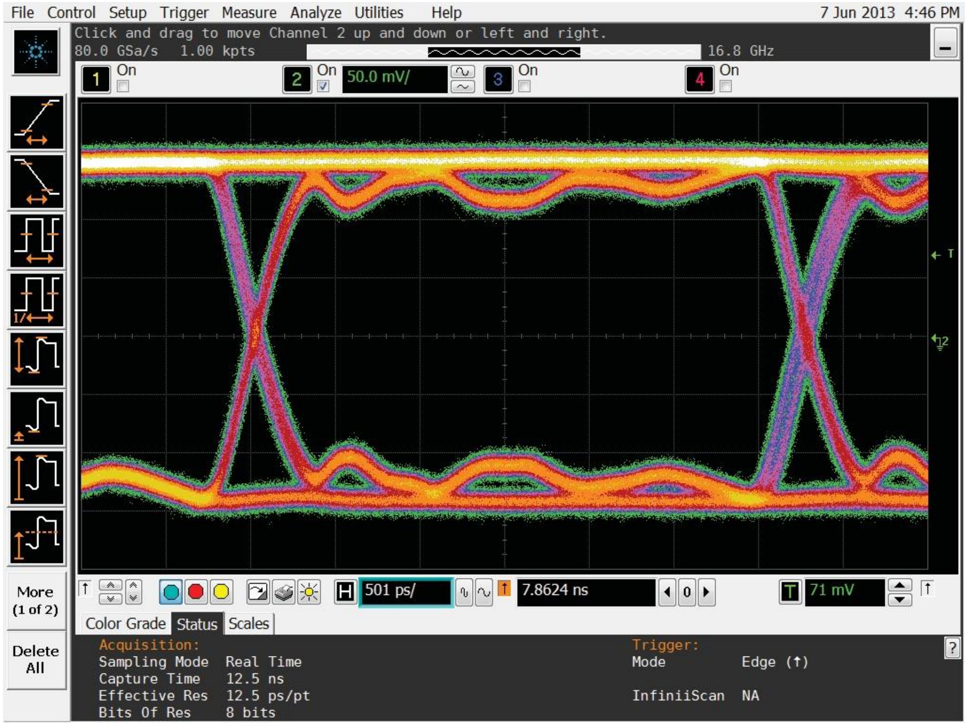


Captured with 500 MHz, single ended probe.
HS Clock and Data at ~320 Mbps Transmission Clock eye diagram

Figure 10.3. Clock and Data Transmission Waveforms from Lattice MachXO2 Device Showing HS and LP Modes



Captured with 16 GHz, Differential probe. Clock at ~160 MHz.



Captured with 16 GHz, Differential probe. Data at ~320 Mbps.

Figure 10.4. Clock and Data Transmission Eye Diagrams from Lattice MachXO2 Device in HS Mode

11. Device Pinout and Bank Voltage Requirements

Choosing a proper pinout to interface with another D-PHY device is essential to meet functional and timing requirements.

The following are rules for choosing a proper pinout on MachXO2 devices:

- Bank 2 should be used for HS inputs (DCK, D0, D1, D2, D3) with the RX D-PHY Reference Design since these pins utilize iDDRx4 gearbox primitives
- Bank 0 should be used for HS outputs (DCK, D0, D1, D2, D3) with the TX D-PHY Reference Design since these pins utilize oDDRx4 gearbox primitives
- The VCCIO voltage for banks 0 and 2 should be 2.5 V
- The HS input clock (DCK) for the RX D-PHY Reference Design should use an edge clock on bank 2
- The HS data signals (D0, D1, D2, D3) for the RX and TX D-PHY Reference Designs should only use A/B I/O pairs
- LP signals (LPCLK, LP0, LP1, LP2, LP3) for RX and TX D-PHY Reference Designs can use any other bank
- The VCCIO voltage for the bank containing LP signals (LPCLK, LP0, LP1, LP2, LP3) should be 1.2 V
- When in doubt, run the pinout through Lattice Diamond software and check for errors

With the rules mentioned, a recommend pinout is provided for the most common packages used for this Reference Design. For the MachXO2 the cs132bga is the most common package. The pinouts chosen below are pin compatible with MachXO2-1200, MachXO2-2000 and MachXO2-4000 devices.

Table 11.1. Recommended RX Pinout and Package

Signal	MachXO2-1200/2000/4000 csBGA132 Package		MachXO3L 2100C caBGA256 Package		LatticeECP3-150EA fpBGA1156		ECP5 LFE5U-45F-8MG285C	
DCK_p	Bank 2	N6	Bank 2	T7	Bank 2	U26	Bank 3	H4
DCK_n		P6		R8		U27		G4
D0_p		M11		P8		L26		M3
D0_n		P12		T8		M25		L3
D1_p		P8		R7		L32		K3
D1_n		M8		P7		L31		K1
D2_p		P2		T5		L34		K2
D2_n		N2		R6		L33		L1
D3_p		N3		P4		K29		M2
D3_n		P4		T4		K30		N2
LPCLK [1]	Bank 1	J13	Bank 1	P16	Bank 7	P5	Bank 2	A2
LPCLK [0]		K12		N15		N8		A3
LP0 [1]		K13		L15		N1		B7
LP0 [0]		K14		M14		N10		C7
LP1 [1]		L14		M16		N2		B2
LP1 [0]		M13		M15		M10		D6
LP2 [1]		M12		P15		N3		D1
LP2 [0]		M14		N14		N5		C1
LP3 [1]		N13		R16		N4		D2
LP3 [0]		N14		N16		M5		B1

Table 11.2. Recommended TX Pinout and Package

Signal	MachXO2-1200/2000/4000 csBGA132 Package		MachXO3L 2100C caBGA256 Package		LatticeECP3-150EA fpBGA1156		ECP5 LFE5U-85F-6MG285CES	
DCK_p	Bank 0	A7	Bank 0	A4	Bank 6	AJ4	Bank 6	L15
DCK_n		B7		C5		AK4		L16
D0_p		B5		D10		AP5		K17
D0_n		C6		E10		AP6		L18
D1_p		A2		B11		AL4		K16
D1_n		B3		A12		AM4		K15
D2_p		A10		C8		AL5		J18
D2_n		C11		A8		AM5		K18
D3_p		C12		D6		AJ5		J17
D3_n		A12		E7		AJ6		H18
LPCLK [1]	Bank 1	E12	Bank 1	C16	Bank 7	M2	Bank 7	B18
LPCLK [0]		E14		E14		M7		B17
LP0 [1]		E13		N14		M3		A16
LP0 [0]		F12		R16		M9		C16
LP1 [1]		F13		P16		M4		A12
LP1 [0]		F14		N15		N9		C13
LP2 [1]		G12		N16		L4		C12
LP2 [0]		G14		P15		K5		B12
LP3 [1]		G13		M15		L5		C17
LP3 [0]		H12		C16		K6		A17

The I/O timing analysis shows the setup and hold time window for the HS data paths. Setup and hold timing is based on HS clock (DCK) and HS data (D0, D1, D2, and D3) I/O.

Table 11.3. RX I/O Timing

Device Family	Speed Grade - 4		Speed Grade - 5		Speed Grade - 6	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
MachXO2	198	344	219	287	233	287

Device Family	Speed Grade - 5		Speed Grade - 6	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
MachXO3L	219	287	233	287

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
LatticeECP3	471	471	403	403	321	321

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
ECP5	471	471	403	403	321	321

Table 11.4. RX Design Timing

Device Family	Speed Grade - 4	Speed Grade - 5	Speed Grade - 6
	byte_clk (MHz)	byte_clk (MHz)	byte_clk (MHz)
MachXO2 (LSE)	93.374	101.286	112.08
MachXO2 (Syn)	93.519	102.564	105.208

Device Family	Speed Grade - 5	Speed Grade - 6
	byte_clk (MHz)	byte_clk (MHz)
MachXO3L (LSE)	71.291	82.912
MachXO3L (Syn)	89.526	101.554

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 7	
	byte_clk (MHz)	div2clk (MHz)	byte_clk (MHz)	div2clk (MHz)	byte_clk (MHz)	div2clk (MHz)
LatticeECP3	153.516	282.008	164.826	305.998	182.315	339.905

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	byte_clk (MHz)	div2clk (MHz)	byte_clk (MHz)	div2clk (MHz)	byte_clk (MHz)	div2clk (MHz)
ECP5 (LSE)	81.679	206.398	104.264	286.123	122.56	341.413
ECP5 (Syn)	75.890	195.88	89.59	225.93	117.028	250.62

Table 11.5. TX I/O Timing

Device Family	Speed Grade - 4		Speed Grade - 5		Speed Grade - 6	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
MachXO2	710	710	570	570	455	455

Device Family	Speed Grade - 5		Speed Grade - 6	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
MachXO3L	570	570	455	455

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
LatticeECP3	431	431	370	370	285	285

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
ECP5 (Syn)	676	676	560	560	442	442

Table 11.6. TX Design Timing

Device Family	Speed Grade - 4		Speed Grade - 5		Speed Grade - 6	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
MachXO2	710	710	570	570	455	455

Device Family	Speed Grade - 5		Speed Grade - 6	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
MachXO3L	570	570	455	455

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
LatticeECP3	431	431	370	370	285	285

Device Family	Speed Grade - 6		Speed Grade - 7		Speed Grade - 8	
	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)	Setup (ps)	Hold (ps)
ECP5 (Syn)	676	676	560	560	442	442

Device Family	Speed Grade - 4			Speed Grade - 5			Speed Grade - 6		
	byte_clk (MHz)	bit_clk (MHz)	bit_clk_90	byte_clk (MHz)	bit_clk (MHz)	bit_clk_90	byte_clk (MHz)	bit_clk (MHz)	bit_clk_90
MachXO2 (LSE)	345.662	262.605	262.605	388	315.060	315.060	436.681	378.215	378.215
MachXO2 (Syn)	345.662	262.605	262.605	388	315.060	315.060	436.681	378.215	378.215

Device Family	Speed Grade - 5		Speed Grade - 5		Speed Grade - 6	
	bit_clk (MHz)	bit_clk_90	bit_clk (MHz)	bit_clk_90	bit_clk (MHz)	bit_clk_90
MachXO3L (LSE)	262.605	262.605	317.259	317.259	400.0	400.0
MachXO3L (Syn)	262.60	262.60	317.259	317.259	400.0	400.0

Device Family	Speed Grade -6				Speed Grade -7			
	byte_clk (MHz)	div2clk (MHz)	bit_clk (MHz)	bit_clk_90	byte_clk (MHz)	div2clk (MHz)	bit_clk (MHz)	bit_clk_90
LatticeECP3	176.678	183.419	375	375	208.333	188.608	420	420

Device Family	Speed Grade -8			
	byte_clk (MHz)	div2clk (MHz)	bit_clk (MHz)	bit_clk_90
LatticeECP3	230.309	197.511	500	500

Device Family	Speed Grade - 6			Speed Grade - 7			Speed Grade - 8		
	byte_clk (MHz)	bit_clk_ 90	clkdiv4 (MHz)	byte_clk (MHz)	bit_clk_	clkdiv4 (MHz)	byte_clk (MHz)	bit_clk_	clkdiv4 (MHz)
ECP5 (LSE)	220.99	312.50	189.21	250.627	350.877	208.247	253.485	400	246.853
ECP5 (Syn)	235.10	312.0		252.972	350.877		274.424	400	

12. Resource Utilization

The resource utilization tables below represent the device usage in various configurations of the D-PHY Reference Design. Resource utilization is performed on the reference design in configurations of 1, 2 and 4 data lanes. For each of these configurations LP mode on the data lanes used is turned on. In addition, HS and LP clock signals were available for each configuration. For the RX D-PHY Reference Design, word alignment and lane alignment is turned on except where stated otherwise. The alignment module consumes the majority of the resources in the design. If alignment is unneeded, the number of registers and LUTs used are significantly reduced.

Table 12.1. RX Resource Utilization

Device Family	Synthesis Engine	Configuration	Register	LUT	EBR	PLL	Gearbox	Clock Divider
MachXO2[1]	LSE	1 Data Lanes (LP+HS)	71	94	0	0	1	1
		2 Data Lanes (LP+HS)	353	476	0	0	2	1
		4 Data Lanes (LP+HS)	697	962	0	0	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	272	348	0	0	4	1
		4 Data Lanes (LP+HS), No Alignment	4	18	0	0	4	1
	Synplify Pro	1 Data Lanes (LP+HS)	71	68	0	0	1	1
		2 Data Lanes (LP+HS)	465	422	0	0	2	1
		4 Data Lanes (LP+HS)	617	557	0	0	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	272	254	0	0	4	1
		4 Data Lanes (LP+HS), No Alignment	4	18	0	0	4	1
MachXO3L[2]	LSE	1 Data Lanes (LP+HS)	71	97	0	0	1	1
		2 Data Lanes (LP+HS)	353	491	0	0	2	1
		4 Data Lanes (LP+HS)	697	964	0	0	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	272	346	0	0	4	1
		4 Data Lanes (LP+HS), No Alignment	4	18	0	0	4	1
	Synplify Pro	1 Data Lanes (LP+HS)	71	68	0	0	1	1
		2 Data Lanes (LP+HS)	465	422	0	0	2	1
		4 Data Lanes (LP+HS)	617	557	0	0	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	272	254	0	0	4	1
		4 Data Lanes (LP+HS), No Alignment	4	18	0	0	4	1
LatticeECP3[3]	—	1 Data Lanes (LP+HS)	111	116	2	0	1	1
		2 Data Lanes (LP+HS)	337	307	4	0	2	1
		4 Data Lanes (LP+HS)	625	555	6	0	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	312	294	2	0	4	1
		4 Data Lanes (LP+HS), No Alignment	44	65	2	0	4	1

Device Family	Synthesis Engine	Configuration	Register	LUT	EBR	PLL	Gearbox	Clock Divider
ECP5 [4]	LSE	1 Data Lanes (LP+HS)	144	196	1	1	1	1
		2 Data Lanes (LP+HS)	490	535	3	1	2	1
		4 Data Lanes (LP+HS)	742	672	5	1	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	345	446	1	1	4	1
		4 Data Lanes (LP+HS), No Alignment	77	114	1	1	4	1
	Synplify Pro	1 Data Lanes (LP+HS)	143	166	1	1	1	1
		2 Data Lanes (LP+HS)	489	463	3	1	2	1
		4 Data Lanes (LP+HS)	741	667	5	1	4	1
		4 Data Lanes (LP+HS), Word Alignment Only	361	385	1	1	4	1
		4 Data Lanes (LP+HS), No Alignment	76	109	1	1	4	1

Notes:

1. Performance and utilization characteristics are generated using LCMXO2 -1200HC-4MG132C with Lattice Diamond 3.3 design software.
   When using this design in a different device, density, speed, or grade, performance and utilization may vary.
2. Performance and utilization characteristics are generated using LCMXO3l -2100C-5BG256C with Lattice Diamond 3.3 design software.
   When using this design in a different device, density, speed, or grade, performance and utilization may vary.
3. Performance and utilization characteristics are generated using LFE3-70EA -8FN484C with Lattice Diamond 3.3 design software.
   When using this design in a different device, density, speed, or grade, performance and utilization may vary.
4. Performance and utilization characteristics are generated using LFE5U-45F -8MG285C with Lattice Diamond 3.3 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.

Table 12.2. TX Resource Utilization

Device Family	Synthesis Engine	Configuration	Register	LUT	EBR	PLL	Gearbox	Clock Divider
MachXO2[1]	LSE	1 Data Lanes (LP+HS)	4	9	0	1	2	1
		2 Data Lanes (LP+HS)	4	12	0	1	3	1
		4 Data Lanes (LP+HS)	4	18	0	1	5	1
	Synplify Pro	1 Data Lanes (LP+HS)	4	9	0	1	2	1
		2 Data Lanes (LP+HS)	4	12	0	1	3	1
		4 Data Lanes (LP+HS)	4	18	0	1	5	1
MachXO3L[2]	LSE	1 Data Lanes (LP+HS)	4	9	0	1	2	1
		2 Data Lanes (LP+HS)	4	12	0	1	3	1
		4 Data Lanes (LP+HS)	4	18	0	1	5	1
	Synplify Pro	1 Data Lanes (LP+HS)	4	9	0	1	2	1
		2 Data Lanes (LP+HS)	4	12	0	1	3	1
		4 Data Lanes (LP+HS)	4	18	0	1	5	1
LatticeECP3[3]	—	1 Data Lanes (LP+HS)	46	57	1	1	2	1
		2 Data Lanes (LP+HS)	46	60	1	1	3	1
		4 Data Lanes (LP+HS)	46	66	1	1	5	1
ECP5[4]	LSE	1 Data Lanes (LP+HS)	87	113	1	1	2	1
		2 Data Lanes (LP+HS)	87	116	1	1	3	1
		4 Data Lanes (LP+HS)	87	122	1	1	5	1
	Synplify Pro	1 Data Lanes (LP+HS)	86	105	1	1	2	1
		2 Data Lanes (LP+HS)	86	108	1	1	3	1
		4 Data Lanes (LP+HS)	86	114	1	1	5	1

Notes:

1. Performance and utilization characteristics are generated using LCMXO2 -1200HC-4MG132C with Lattice Diamond 3.3 design software.
   When using this design in a different device, density, speed, or grade, performance and utilization may vary.
2. Performance and utilization characteristics are generated using LCMXO3l -2100C-5BG256C with Lattice Diamond 3.3 design software.
   When using this design in a different device, density, speed, or grade, performance and utilization may vary.
3. Performance and utilization characteristics are generated using LFE3-17EA-8FN484C with Lattice Diamond 3.3 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.
4. Performance and utilization characteristics are generated using LFE5U-85F-6MG285CES with Lattice Diamond 3.3 design software. When using this design in a different device, density, speed, or grade, performance and utilization may vary.

13. References

- MIPI Alliance Specification for Camera Serial Interface 2 (CSI2) V1.01
- MIPI Alliance Specification for D-PHY V1.1

Technical Support Assistance

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

Revision History

Revision 1.7, September 2020

Section	Change Summary
All	• Changed document title from *MIPI D-PHY Interface IP* to *MIPI DPHY*. • Updated document template.
Disclaimers	Added this section.
Acronyms in This Document	Added this section.
Introduction	Added supported devices.
Packaged Design	Updated folder names.
—	• Changed *ohm* to *Ω* in paragraphs and figures. • Other minor changes in formatting and style.

Revision 1.6, May 2018

Section	Change Summary
All	Changed document number from RD1182 to FPGA-RD-02040.
Device Pinout and Bank Voltage Requirements	Updated values in Table 8 and Table 10.
Technical Support Assistance	Updated information.

Revision 1.5, January 2015

Section	Change Summary
All	Added support for MachXO3L and ECP5.
Packaged Design	General update.
Simulation	Updated Figure 9, Simulation Wizard Script File Access
Device Pinout and Bank Voltage Requirements	General update. Updated the following tables to add support for MachXO3L and ECP5: • Table 8, RX I/O Timing • Table 9, RX Design Timing • Table 10, TX I/O Timing • Table 11, TX Design Timing.
Resource Utilization	Updated the following tables to add support for MachXO3L and ECP5: • Table 12, RX Resource Utilization • Table 13, TX Resource Utilization.

Revision 1.4, March 2014

Section	Change Summary
Transmitting Interface	• Updated introduction. • Updated Figure 6, Unidirectional Transmit HS Mode Only Implementation. Indicated 70 Ω resistors.

Revision 1.3, January 2014

Section	Change Summary
Transmitting Interface	• Updated Figure 5, Unidirectional Transmit HS Mode and Bidirectional LP Mode Interface Implementation. Changed the 320 Ω resistor to 330 Ω. • Updated Figure 6, Unidirectional Transmit HS Mode Only Implementation. Changed the 320 Ω resistor to 330 Ω.

Revision 1.2, August 2013

Section	Change Summary
Receiving Interface	Updated Figure 2, Unidirectional Receive HS Mode and Bidirectional LP Mode Interface Implementation.

Revision 1.1, August 2013

Section	Change Summary
	Updated Table 10. TX I/O Timing.

Revision 1.0, July 2013

Section	Change Summary
All	Initial release.