OpenLDI/FPD-Link/LVDS Transmitter IP

IP Version: v1.8.0

User Guide

FPGA-IPUG-02117-1.6

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

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
AXI	Advance eXtensible Interface
CSR	Control and Status Registers
FPD	Flat Panel Display
LVDS	Low Voltage Differential Signaling
LDI	LVDS Display Interface
OpenLDI	Open LVDS Display Interface
RTL	Register Transfer Language
UVSI	Unified Video Streaming Interface

1. Introduction

The increasing demand for better display technology makes bridging applications popular. FPD-Link is a common application interface. Similar to Channel Link and Camera Link, FPD-Link also uses LVDS interface for the physical layer.

The LVDS standard is commonly used in high-speed differential interface among consumer device, industrial control, medical, and automotive applications. It offers low voltage, low power, and improved signal integrity, which are advantages over single-ended technology.

The 7:1 LVDS interface is a popular standard for source asynchronous interfaces consisting of multiple data bits and clocks. Typically, one channel of 7:1 LVDS interface consists of five LVDS pairs (one clock and four data) depending on the data type it supports.

This document describes the use of the FPD-Link Transmitter IP and Lattice FPGA technology for LVDS interface applications. The design, which can be applied in multiple configurations, is implemented in Verilog HDL. It can be targeted to CrossLink™-NX, Certus™-NX, CertusPro™-NX, MachXO5™-NX, Lattice Avant™, and Certus-N2 FPGA devices and implemented using the Lattice Radiant™ software Place and Route tool integrated with the Synplify Pro® synthesis tool.

1.1. Overview of the IP

The Lattice Semiconductor FPD-Link Transmitter IP translates parallel video streams to a Low Voltage Differential Signaling (LVDS) interface for a Flat Panel Display Link (FPD-Link) connection to a display. The IP converts pixel data into the standard OpenLDI serial video interface domain. The input interface for the design consists of the RGB control signals, pixel clock, and up to two pixels per clock. The output interface consists of a data bus, vertical and horizontal sync flags, a data enable, a clock in the OpenLDI (FPD-Link) interface format, and optional debug signals.

1.2. Quick Facts

Table 1.1. Summary of the FPD-Link Transmitter IP

IP Requirements	Supported Devices	CrossLink-NX, Certus-NX (except LFD2NX-35, LFD2NX-65), CertusPro-NX, MachXO5-NX (except LFMXO5-35, LFMXO5-35T, LFMXO5-65, LFMXO5-65T), Lattice Avant, Certus-N2
	IP Changes[1]	Refer to the OpenLDI/FPD-Link/LVDS Transmitter IP Release Notes (FPGA-RN-02012).
Resource Utilization	Supported User Interface	Parallel to Native LVDS Interface, AXI4-Stream Interface, Unified Video Streaming Interface, AXI4-Lite Interface (for register access)
	Resources	Refer to Appendix A. Resource Utilization.
Design Tool Support	Lattice Implementation	IP Core v1.8.0 – Lattice Radiant Software 2025.2
	Synthesis	Synopsys® Synplify Pro for Lattice, Lattice Synthesis Engine
	Simulation	Refer to the Lattice Radiant Software User Guide for the list of supported simulators.

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. IP Support Summary

Table 1.2. FPD-Link Transmitter IP Support Readiness

Device Family	Data Type	Number of Input Pixels per Clock	Video Data Interface	Number of Tx Channels	Data Mapping Format	Tx Line Rate per Lane (Mbps)	Sync Clock Frequency (MHz)	Register Interface	Miscellaneous Signals	Test Mode	Radiant Timing Model	Hardware Validated
Avant	RGB888	1, 2	Native, Legacy, UVSI	1, 2	JEIDA, VESA	1050	150	Off, AXI4-Lite	Off, On	Off, On	Preliminary	No
	RGB666	1, 2	Native, Legacy, UVSI	1, 2	JEIDA, VESA	1050	150	Off, AXI4-Lite	Off, On	Off, On	Preliminary	No
CertusPro-NX	RGB888	1, 2	Native, Legacy, UVSI	1, 2	JEIDA, VESA	945	10, 95, 135	Off, AXI4-Lite	Off, On	Off, On	Final	No
	RGB666	1, 2	Native, Legacy, UVSI	1, 2	JEIDA, VESA	945	10, 95, 135	Off, AXI4-Lite	Off, On	Off, On	Final	No
CrossLink-NX	RGB888	1, 2	Native, Legacy, UVSI	1, 2	JEIDA, VESA	945	10, 95, 135	Off, AXI4-Lite	Off, On	Off, On	Final	No
	RGB666	1, 2	Native, Legacy, UVSI	1, 2	JEIDA, VESA	945	10, 95, 135	Off, AXI4-Lite	Off, On	Off, On	Final	No

1.4. Features

Key features of the FPD-Link Transmitter IP include:

- Compliant with Open LVDS Display Interface (OpenLDI) v0.95 specifications.
- Transmits in OpenLDI unbalanced operating mode format.
- Supports RGB888 and RGB666 video formats.
- Supports transmitting in Dual Channel Flat Panel Display Link Protocol (7:1 LVDS).
- Supports three to four LVDS data lanes per channel.
- Supports one or two input pixels per clock.
- Supports interfacing up to 7.560 Gb/s for Nexus devices and 8.4 Gb/s for Avant devices.
- Supports AXI4-Stream Receiver (Legacy) interface.
- Supports Unified Video Streaming interface in compliance with the AXI4-Stream protocol.
- Supports AXI4-Lite interface for register access.
- Supports dynamic reconfiguration in Native Pixel and Unified Video Streaming interfaces.

1.5. Licensing and Ordering Information

The FPD-Link Transmitter IP is provided at no additional cost with the Lattice Radiant software.

1.6. Hardware Support

Hardware support is available in a future release.

1.7. Minimum Device Requirements

Refer to Appendix A for the minimum device requirements to instantiate the IP.

1.8. Conventions

1.8.1. Nomenclature

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

1.8.2. Signal Names

Signal names that end with:

- *_n* are active low (asserted when value is logic 0)
- *_i* are input signals
- *_o* are output signals

1.8.3. Data Ordering and Data Types

- The most significant bit within the pixel data is the highest index.
- Pixel data order before distribution to LVDS lanes is {Red[MSB:0], Green[MSB:0], Blue[MSB:0]}. One or two pixels may be sent for distribution to LVDS lanes in one pixel clock cycle, depending on the number of Tx channels setting. If there are multiple pixels per clock cycle, the pixel in the lower bits is the first pixel received. For instance, the pixel order for two pixels per clock is {pixel1, pixel0}, where pixel0 is received first and pixel1 is received last.
- Pixel data is transmitted over LVDS lanes according to OpenLDI 18-bit and 24-bit unbalanced operating mode formats.

1.8.4. Attribute Names

Attribute names in this document are formatted in title case and italicized (*Attribute Name*).

2. Functional Description

2.1. IP Architecture Overview

Figure 2.1 shows the top-level functional block diagram of the FPD-Link Transmitter IP core. The dashed lines in the figure indicate optional components or signals, which may or may not be available in the IP depending on the attribute and/or device selected.

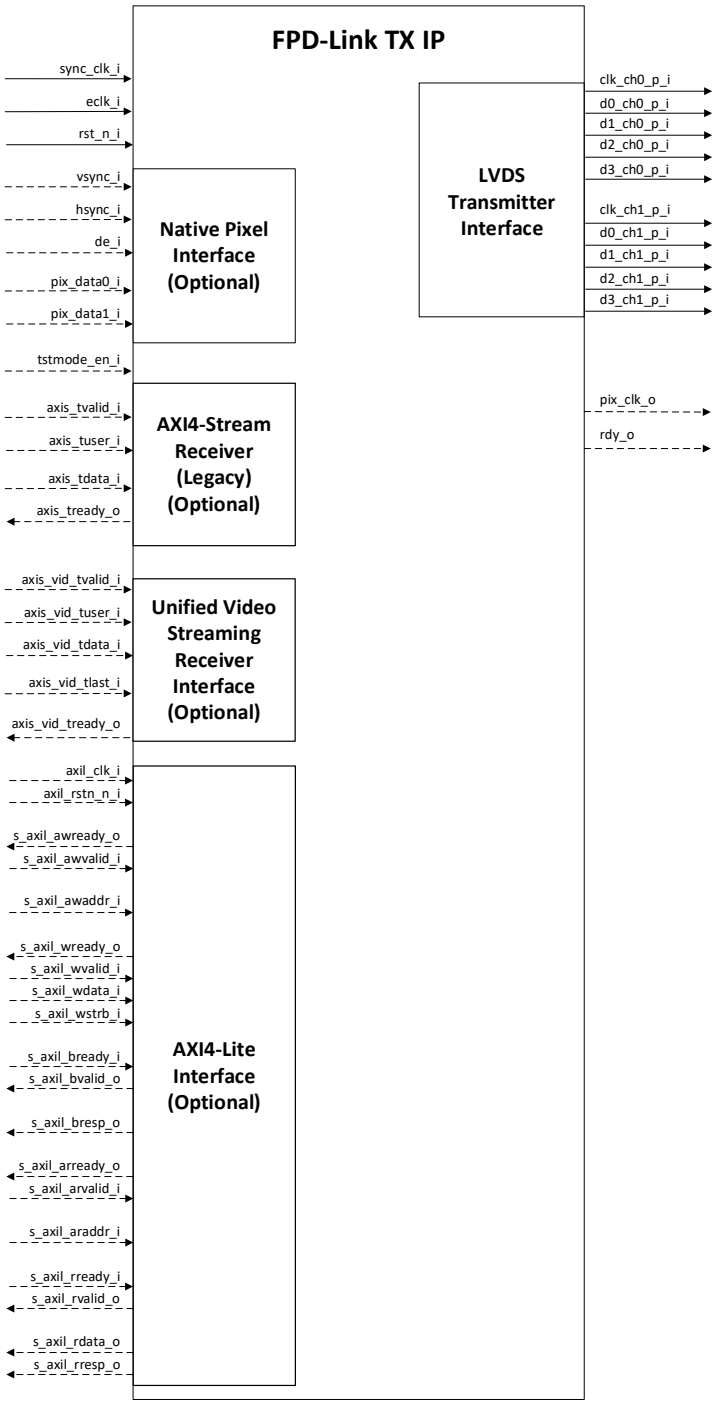

Figure 2.1. FPD-Link Transmitter IP Block Diagram

Figure 2.2 shows the functional block diagram of the FPD-Link Transmitter IP core.

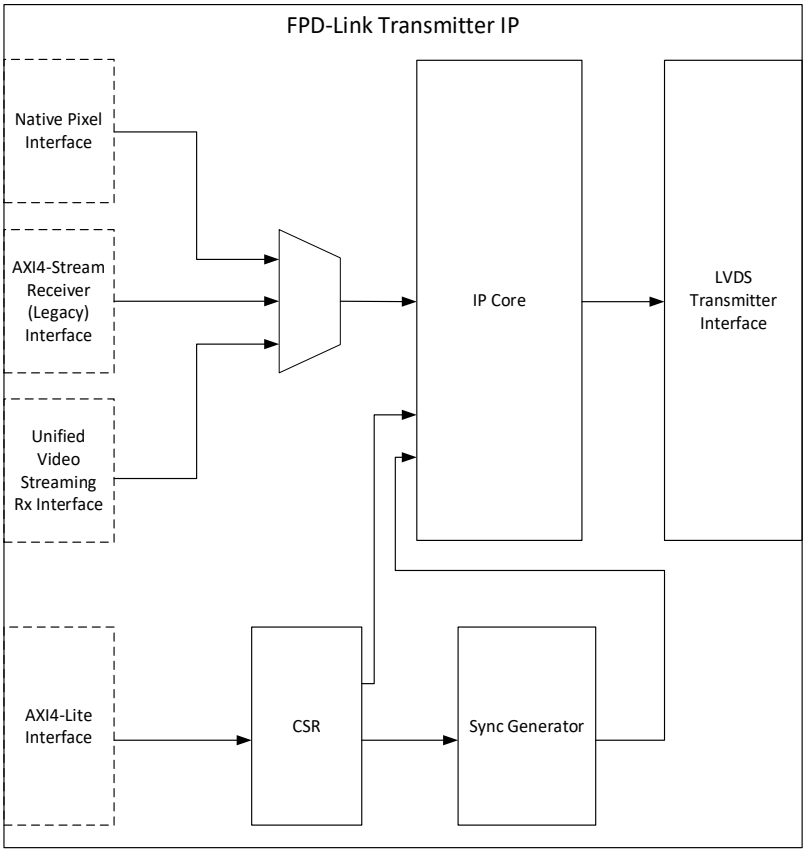

Figure 2.2. Functional Block Diagram

The FPD-Link Transmitter IP includes the following layers:

- IP core
- LVDS Transmitter Interface
- Native Pixel Interface (optional)
- AXI4-Stream Receiver (Legacy) Interface (optional)
- Unified Video Streaming Rx Interface (optional)
- AXI4-Lite Interface (optional)
- Control and status registers (CSR) module
- Sync Generator module

2.2. Clocking

Figure 2.3 shows the clock domains in the FPD-Link Tx IP.

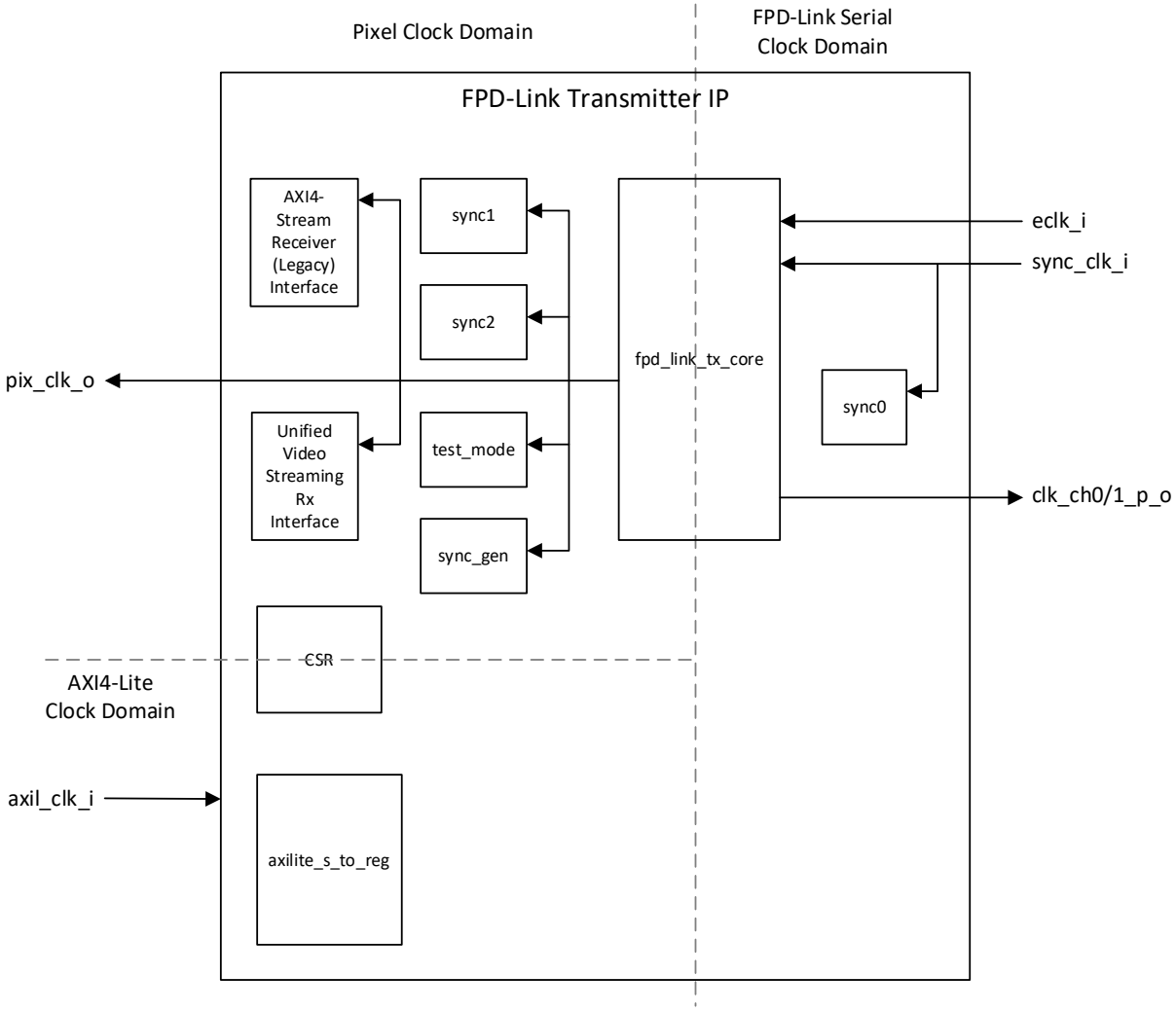

Figure 2.3. FPD-Link Tx IP Clock Domain Block Diagram

2.2.1. Clocking Overview

The general formula for computing the required clocks of the IP:

Tx Line Rate (Total) $= Tx\ Line\ Rate\ (per\ Lane) \times No.\ of\ Tx\ Lanes\ \times No.\ of\ Tx\ Channels$

pixclk (Pixel Clock) $= \frac{Tx\ Line\ Rate\ (per\ Lane)}{Tx\ Gear}$

Tx LVDS Output Clock $= pixclk \times \frac{Tx\ Gear}{7}$

Tx LVDS ECLK $= pixclk \times \frac{Tx\ Gear}{2}$

Number of Pixels per Pixel Clock $= \frac{Tx\ Gear}{7} \times No.\ of\ Tx\ Channels$

2.3. Reset

The FPD-Link Transmitter IP has two reset signals. The first is used as the system reset and the second is used in the AXI4-Lite interface and CSR module. Figure 2.4 shows the reset domain in the FPD-Link Tx IP.

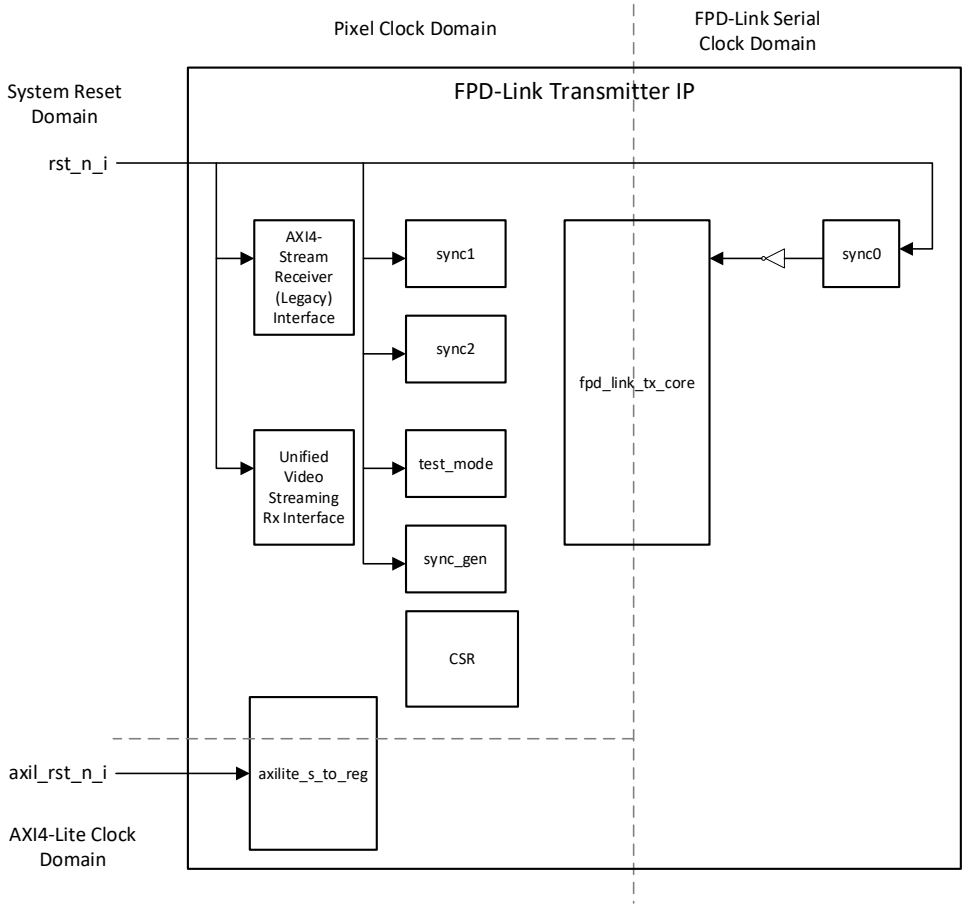

Figure 2.4. FPD-Link Tx IP Reset Domain Block Diagram

2.3.1. Reset Overview

The rst_n_i signal is an asynchronous active low signal but with synchronous release. This is the system reset input connected to the FPD-Link Tx module. This reset is also connected to the pixel and serial clock domains. A separate axil_rst_n_i signal is used in the AXI4-Lite interface and registers in the CSR module.

2.3.2. Initialization and Reset Sequence

The following is the initialization and reset sequence:

1. Assert active low system reset (rst_n_i) for at least three clock cycles of the slowest clock (sync clock). It is expected that the input clock is stable after reset. Clock synchronization starts immediately after the release of system reset.
2. If *Enable Miscellaneous Signals* is checked, wait for rdy_o to be asserted before sending the valid data to give time for the FPD-Link Tx clock synchronization to complete. rdy_o is used to indicate that the IP clock synchronization is done. Only when rdy_o is asserted can valid data be sampled and correctly transmitted by the FPD-Link Tx IP.
   If *Enable Miscellaneous Signals* is unchecked, wait for the pixel clock (pixel_clk_o) to be stable before sending the valid data to give time for Tx clock synchronization to complete. Refer to the Clocking Overview section for more details on how pixel clock frequency is computed.

2.4. User Interfaces

Table 2.1 lists the available user interfaces and supported protocols of the FPD-Link Transmitter IP.

Table 2.1. User Interfaces and Supported Protocols

User Interface	Supported Protocols	Description
Native Pixel	Standard Parallel Video Interface	Receives pixel data.
LVDS Transmitter	OpenLDI/FPD-Link (LVDS 7:1) Serial Video Interface	Outputs data in the OpenLDI serial video format.
AXI4-Stream Receiver	AXI4-Stream Interface	Receives pixel data.
Unified Video Streaming Rx	AXI4-Stream Interface	Receives pixel data.
Control	AXI4-Lite Interface	Allows access to the control and status registers of the FPD-Link Tx IP.

2.4.1. Native Pixel Interface

The Native Pixel interface follows the standard parallel video interface protocol transmitting pixel data with VSYNC, HSYNC, and Data Enable, all clocked by the pixel clock. Figure 2.5 shows the timing of the data format when the Native Pixel interface is enabled.

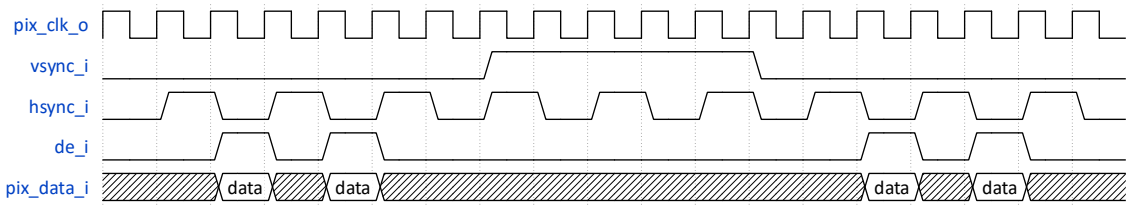

Figure 2.5. Parallel Video Input Timing Diagram

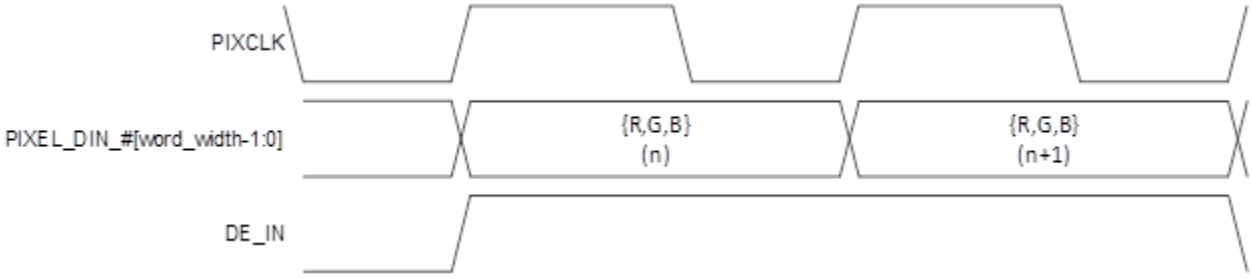

Figure 2.6. Input Pixel Data RGB Arrangement

Table 2.2. Input Pixel Data Summary

Number of FPD-Link Channels	Gear	No. of Input Pixel Data	Pixel Clock
1	7	1	pix_clk_o
2	7	2	pix_clk_o

2.4.2. LVDS Transmitter Interface

Figure 2.7 shows the timing of the LVDS transmitter interface. There is a 2-bit offset between the rising edge of the LVDS clock and the word boundary. Each word is 7 bits long.

DATAOUT0, DATAOUT1, DATAOUT2, and DATAOUT3 are the data lanes. CLOCKOUT is the LVDS clock lane. For every 7-bit data packet, LSB is the first output serial data to the receiver. A processor sends parallel video packet data to the FPGA chip. Each data lane is serialized using the DDR primitive. One channel of FPD-Link transmitter has a maximum of five lanes. Each channel consists of one LVDS clock pair and four LVDS data pairs (RGB888) or three LVDS data pairs (RGB666).

The clock lane is generated by feeding constant 1100011. The clock is edge-aligned against data. The clock runs at 1/7th of the data rate, as per the standard for the FPD-Link interface. Seven bits of data are transmitted in one LVDS clock cycle. The default mode for the LVDS operating system is unbalanced, as this is commonly used. The maximum supported data rate per lane for LVDS is 945 Mb/s for Nexus devices and 1,050 Mb/s for Avant devices. A maximum of two FPD-Link channels can be used. When dual channel is selected, additional data lanes are activated and two additional specific control signals may be transmitted, CNTLE and CNTLF. In this IP, the CNTLE and CNTLF control bits are unused and equate to 0.

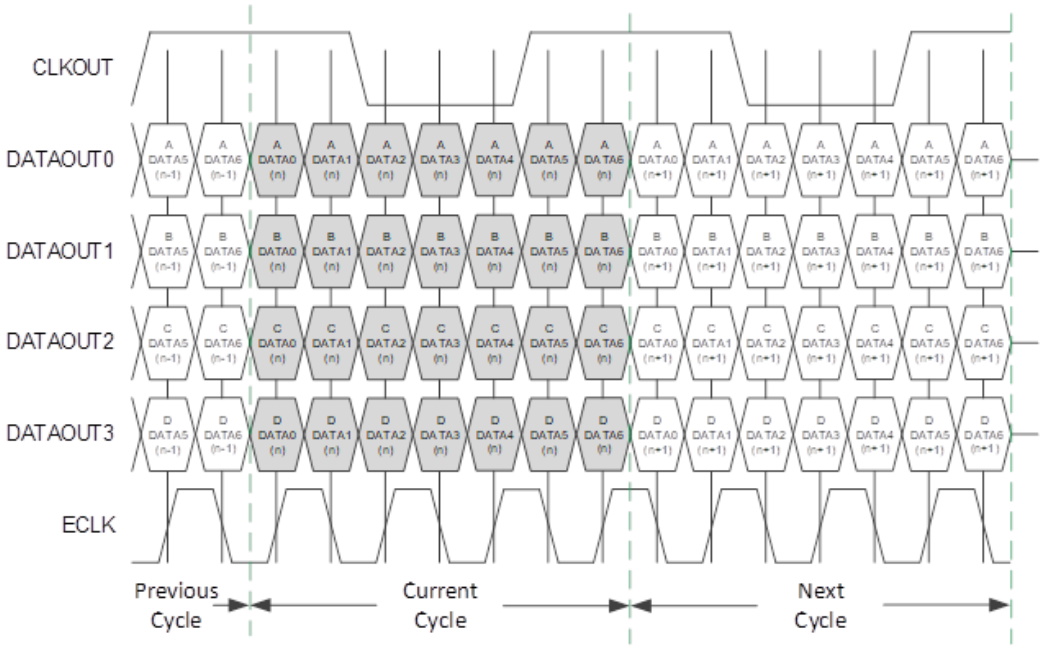

Figure 2.7. OpenLDI/FPD-Link/LVDS Output Bus Waveform

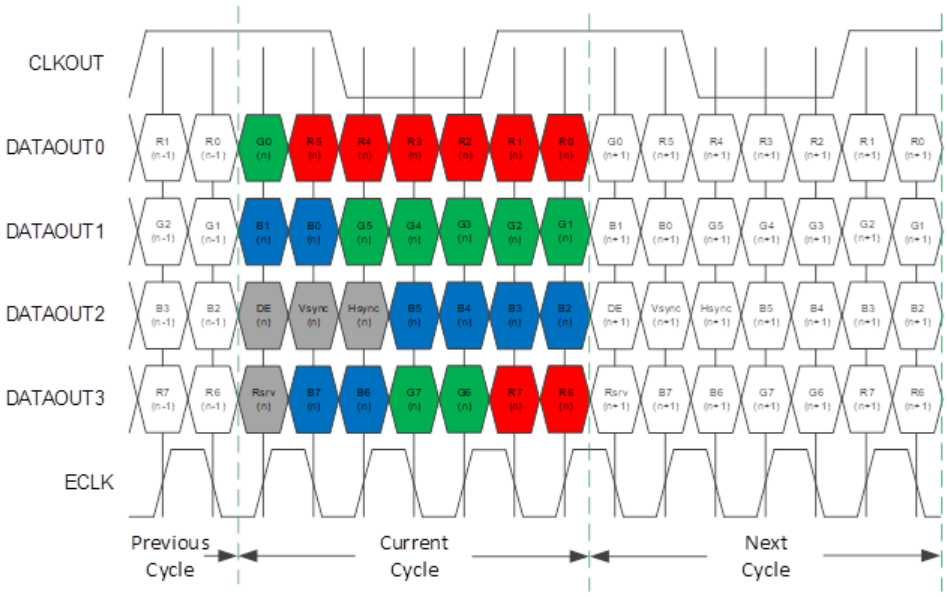

Figure 2.8. Single Channel OpenLDI/FPD-Link/LVDS Output Bus Waveform for RGB888 Format (VESA)

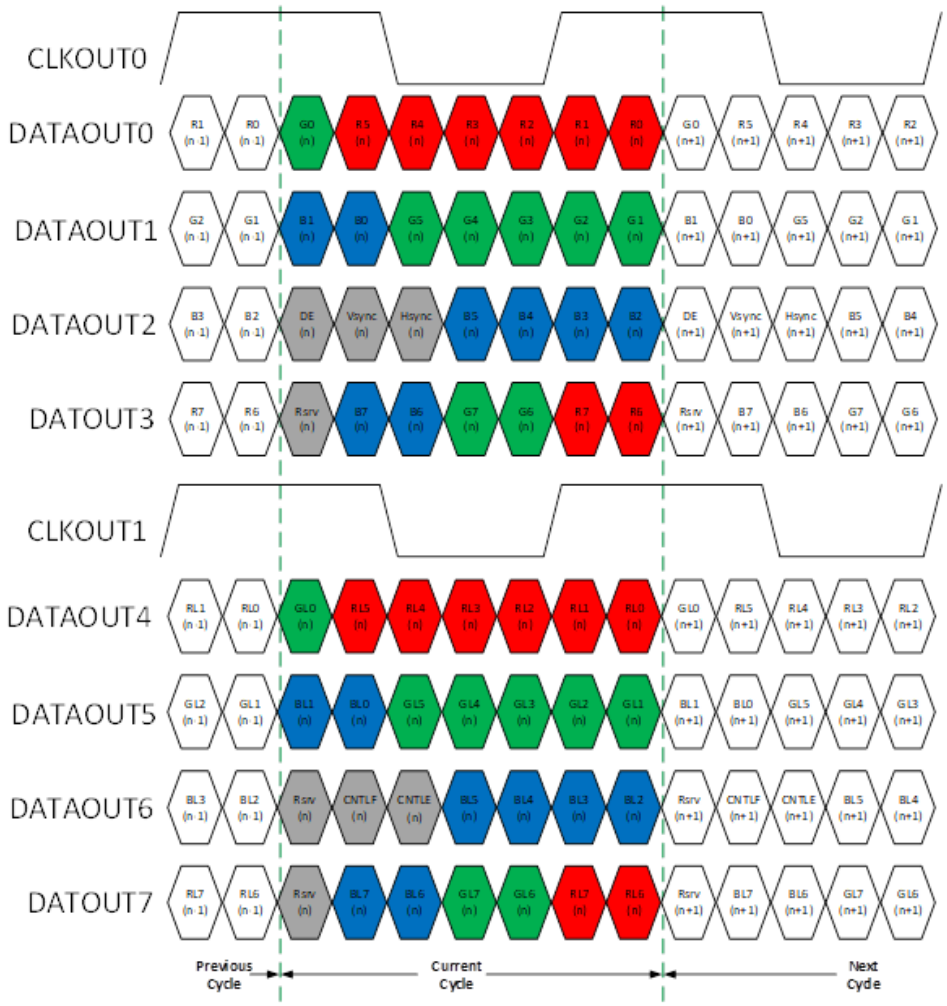

Figure 2.9. Dual Channel OpenLDI/FPD-Link/LVDS Output Bus Waveform for RGB888 Format (VESA)

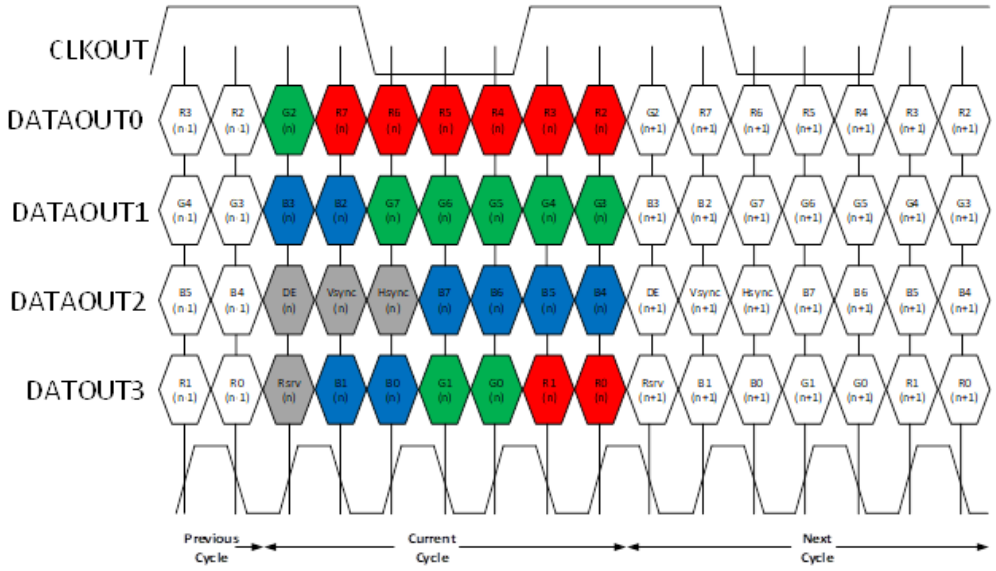

Figure 2.10. Single Channel OpenLDI/FPD-Link/LVDS Output Bus Waveform for RGB888 Format (JEIDA)

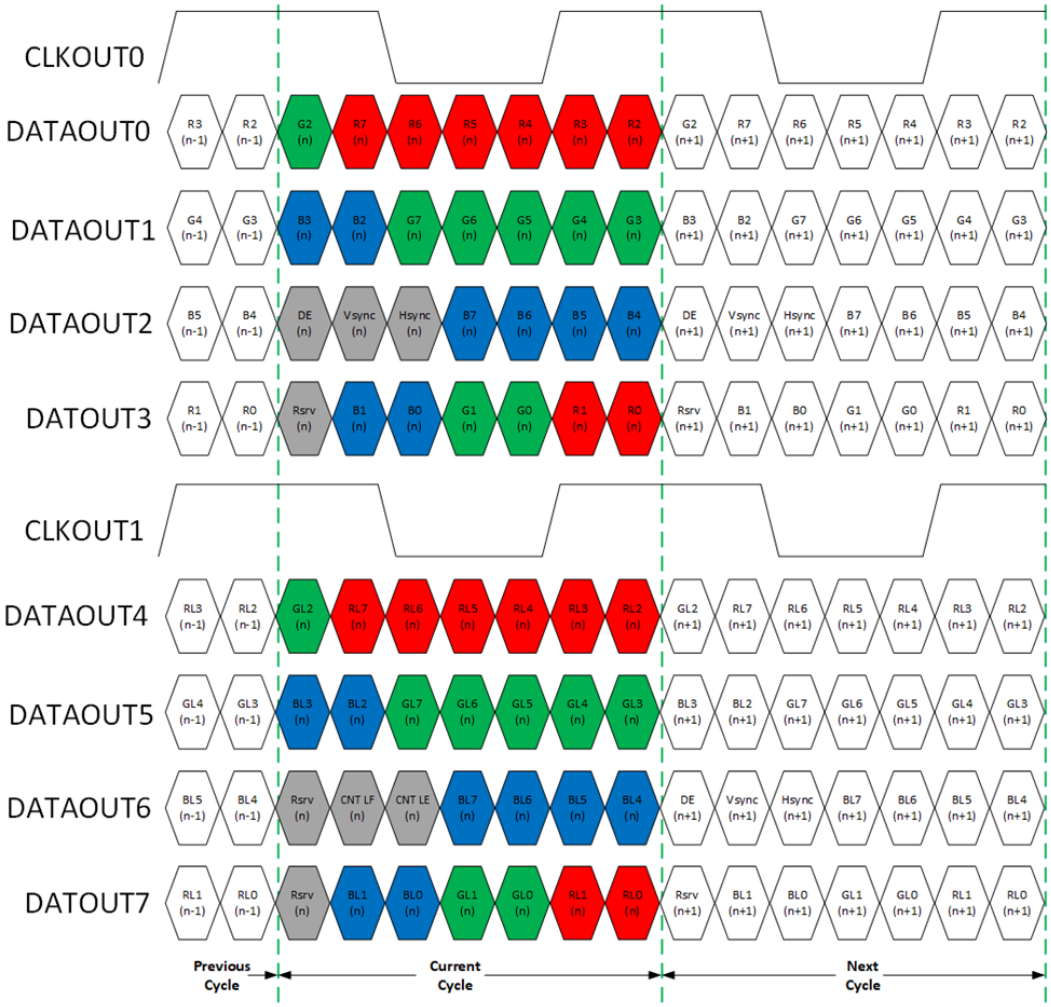

Figure 2.11. Dual Channel OpenLDI/FPD-Link/LVDS Output Bus Waveform for RGB888 Format (JEIDA)

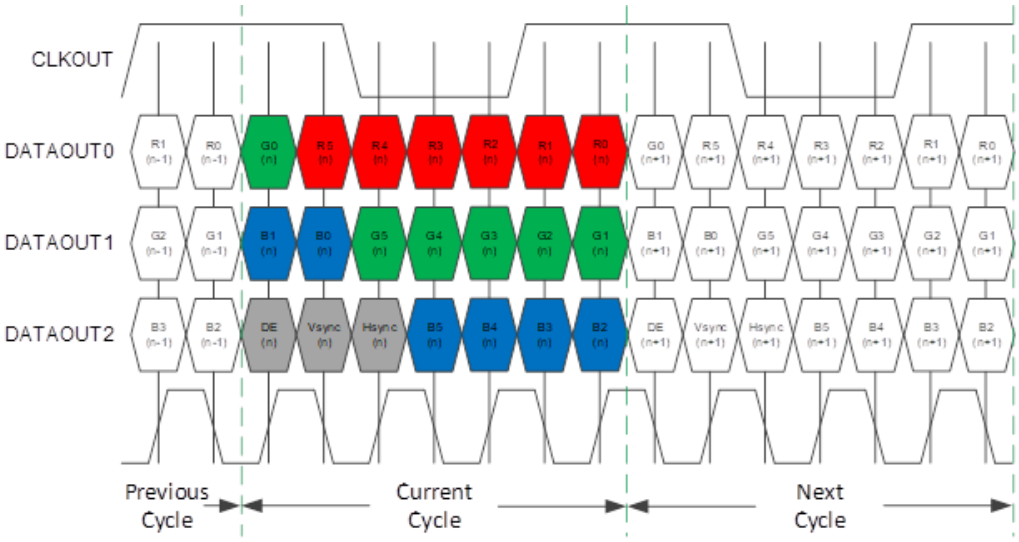

Figure 2.12. Single Channel OpenLDI/FPD-Link/LVDS Output Bus Waveform for RGB666 Format (JEIDA/VESA)

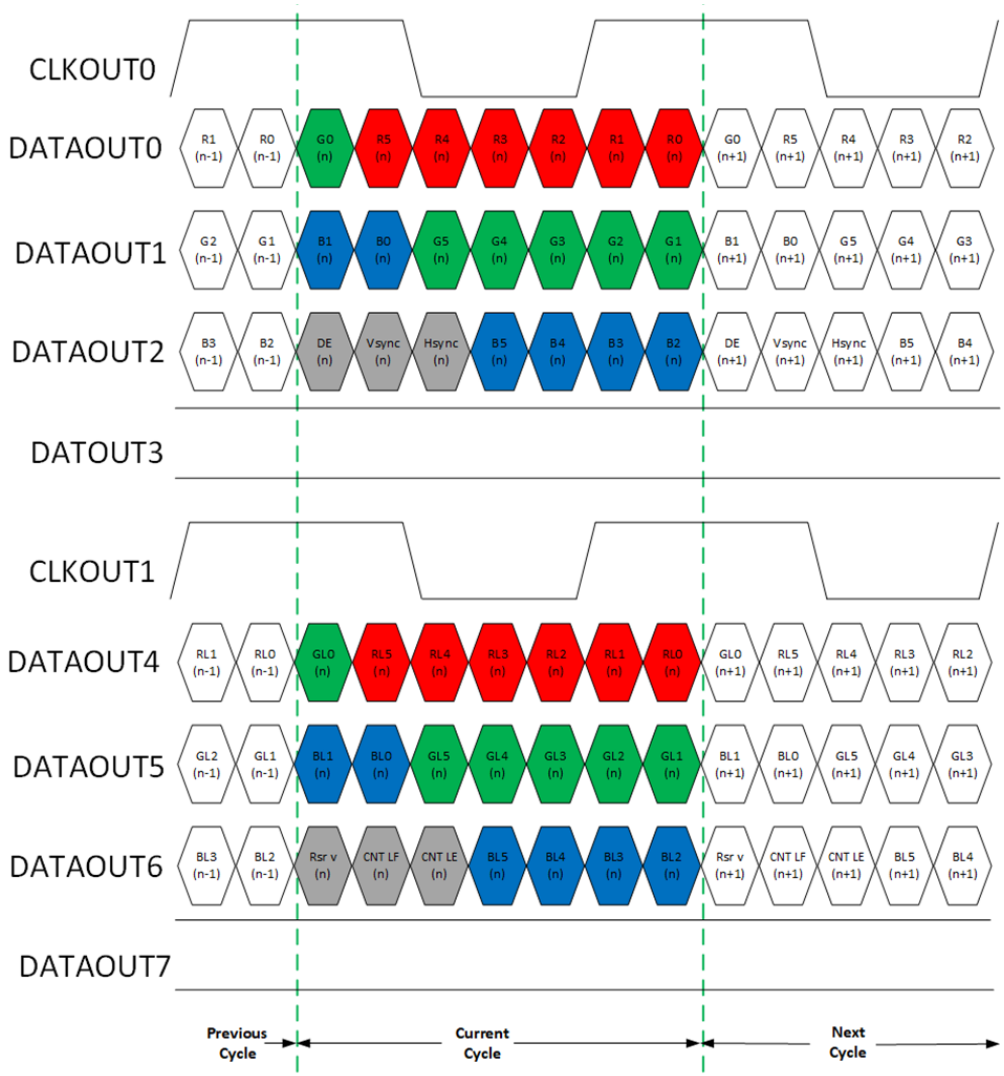

Figure 2.13. Dual Channel OpenLDI/FPD-Link/LVDS Output Bus Waveform for RGB666 Format (JEIDA/VESA)

2.4.3. AXI4-Stream Receiver (Legacy) Interface

The AXI4-Stream Receiver interface is a legacy interface for receiving pixel data. Figure 2.14 shows the timing of the data format when the AXI4-Stream Receiver interface is enabled.

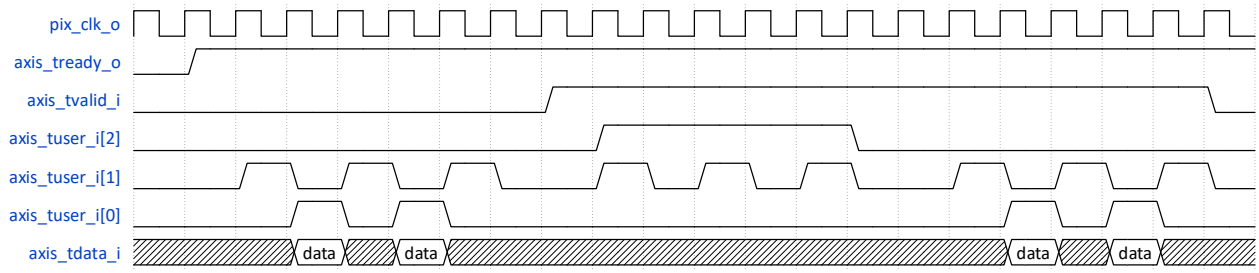

Figure 2.14. Data Format Timing when AXI4-Stream Receiver Interface is Enabled

If the AXI4-Stream Receiver interface is not enabled, the signals are mapped to top-level input signals as shown in Table 2.3.

Table 2.3. Data Mapping of AXI4-Stream Receiver (Legacy) Interface Ports from Native Pixel Interface

Native Pixel Interface	Equivalent Port
vsync_i	axis_tuser_i[2]
hsync_i	axis_tuser_i[1]
de_i	axis_tuser_i[0]
pix_data0_i	axis_tdata_i
pix_data1_i	axis_tdata_i

2.4.4. Unified Video Streaming Rx Interface

The Unified Video Streaming Rx interface is an interface for receiving pixel data. This interface is compliant with the AMBA AXI4-Stream protocol specification. axis_vid_tdata_i is mapped from the pixel data of the Native Pixel interface and axis_vid_tuser_i[0] indicates the assertion of the first valid data per frame. This interface is compliant with the AMBA AXI4-Stream protocol specification. Figure 2.15 shows the timing of the data format when the Unified Video Streaming Rx interface is enabled.

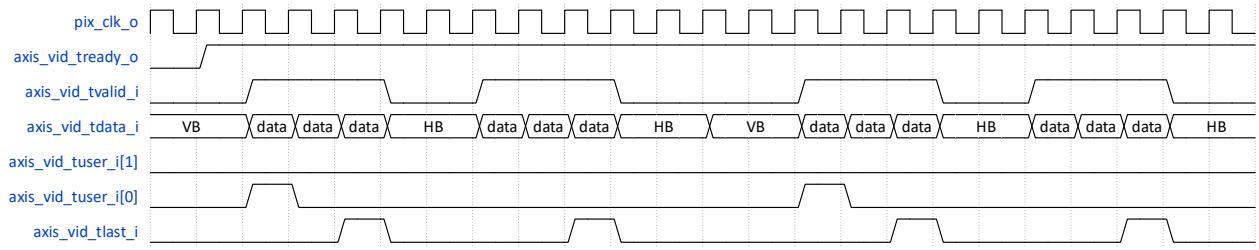

Figure 2.15. Data Format Timing when Unified Video Streaming Interface is Enabled

If the Unified Video Streaming Rx interface is not enabled, the signals are mapped to top-level input signals as shown in Table 2.4.

Table 2.4. Data Mapping of Unified Video Streaming Interface Ports from Native Pixel Interface

Native Pixel Interface	Equivalent Port
de_i	axis_vid_tvalid_i
pix_data0_i	axis_vid_tdata_i
pix_data1_i	axis_vid_tdata_i

The size of axis_vid_tdata_i varies depending on tdata width per pixel (TD_WD) and pixels per clock (PPC). This IP only supports 1 and 2 PPC. TD_WD is dependent on the following attributes:

- BPP – Bits per pixel. Calculated as CPP × BPC.
- CPP – Colors per pixel. Only three colors are supported namely red, green, and blue.
- BPC – Bits per color. Minimum width of each color component is 1 byte (8 -bits). RGB888 and RGB666 use 8 bits and 6 bits per color, respectively. These bits are known as active bits.

TD_WD = ceil(BPP/8) × 8 and TDATA = TD_WD × PPC.

When the active bits per color is less than the minimum BPC which is 8 bits, the additional bits or least significant bits (LSBs) of each color are padded with 0s. For RGB666, with six active bits per color, the two LSBs of each color component are padded with 0s. Figure 2.16 and Figure 2.17 show sample TDATA mapping for the RGB888 and RGB666 formats.

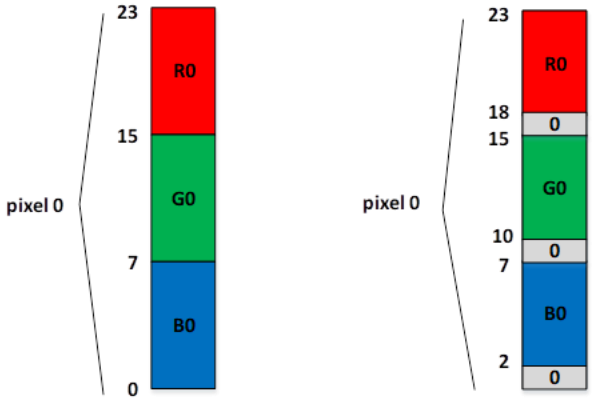

Figure 2.16. Unified Video Streaming Rx TDATA Mapping of RGB888 and RGB666 when PPC = 1

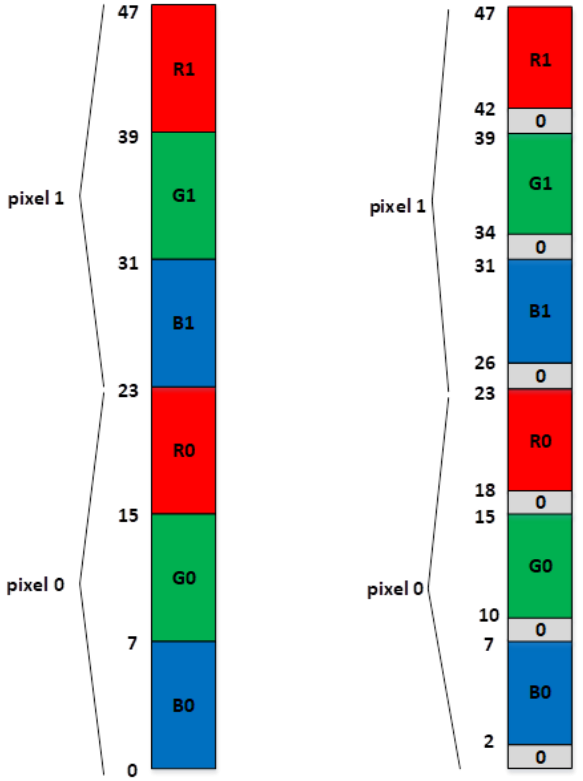

Figure 2.17. Unified Video Streaming Rx TDATA Mapping of RGB888 and RGB666 when PPC = 2

2.4.5. AXI4-Lite Interface

The AXI4-Lite Manager module through the AXI4-Lite interface is used for configuring the control and status registers of the FPD-Link Tx IP. Not all registers are configurable, and access varies depending on the register. It is recommended that registers be accessed after clock synchronization is done (rdy_o = 1 or pixel_clk_o = pixel clock frequency) to obtain accurate values from the registers. Refer to the Register Description section for more information on the registers.

2.5. Other IP Specific Blocks/Layers/Interfaces

2.5.1. FPD-Link Tx Core Module

The fpd_link_tx_core module instantiates the GDDR_SYNC, lvds_oddr, and lvds_clk_tree modules. The GDDR_SYNC module is required to initialize and synchronize DDR clock and tolerate the large skew between stop and reset of the DDR components.

Parallel data are fed to the I/O logic DDR71 register in the lvds_oddr module. This module is used to convert the incoming parallel data into serial format. The lvds_clk_tree module is used to generate the clocks needed by the system. The SCLK is used as the output pixel clock of the DDR71 IP.

2.5.2. Lane Distribution Module

The lane_distr module is used for distributing pixel data to different LVDS lanes for OpenLDI unbalanced format.

2.5.3. Test Mode Tx Module

The test_mode_tx module is used to automatically generate test data inside the design specified through *Test Mode Expected Data in Hex Format*, and to transmit the data. Data comparison should be done outside the design. Refer to the Debug Mode section for details on how to enable test mode.

2.5.4. Synchronizer Module

The synchronizer module is a two-level synchronizer used to sync the input data into a different clock domain. In the design, this is used to synchronize the system reset into different clock domains before it is used in the system.

2.5.5. Sync Generator Module

The sync generator module produces sync signals (VSYNC and HSYNC) to be used in the lane distribution module when the Unified Video Streaming Rx interface is used as the video data interface. To generate the sync signals, when the AXI4-Lite interface is enabled, video information from the CSR module is used; when the AXI4-Lite interface is disabled, video information is deduced from the user based on attribute settings. The process of sync signals generation from the video information only takes place after rdy_o signal assertion.

2.6. Dynamic Reconfiguration

This IP supports dynamic reconfiguration for bits per color by programming the CFG_REG register through the AXI4-Lite interface. Writing 1 in the register CFG_REG[0] switches the data format from RGB888 (JEIDA/VESA) to RGB666.

Writing 1 in the register CFG_REG[0] can be performed at any time but the change takes effect only on the next frame of pixel data. Dynamic reconfiguration is only supported for the Native Pixel and Unified Video Streaming Rx interfaces. The TDATA mapping for both Native Pixel and Unified Video Streaming Rx interfaces is the same. Refer to the Unified Video Streaming Rx Interface section for TDATA mapping details.

Dynamic reconfiguration of data type in either direction will only be supported if the initial Data Type is set to RGB888.

3. IP Parameter Description

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

Wherever applicable, default values are in bold.

3.1. General

Table 3.1. General Attributes

Attribute	Selectable Values	Description
Pixel Interface		
Data Type	**RGB888**, RGB666	Specifies the data type of the input pixel.
Number of Input Pixels per Clock	**1**, 2	Specifies the number of input pixels per clock. *Number of Input Pixels per Clock = Number of TX Channels*
Video Data Interface	**Native**, AXI4-Stream Rx (Legacy), Unified Video Streaming Rx	Specifies the interface to be used in the pixel domain.
Transmitter Interface		
Number of TX Channels	**1**, 2	Specifies how many LVDS links are used.
TX Interface	**LVDS**	Specifies the I/O interface. This attribute is not configurable.
Data Mapping Format	Format 1 (JEIDA), **Format 2 (VESA)**	Specifies the data mapping format of the output.
Number of TX Lanes	3, **4**	Specifies the number of data lanes per LVDS Tx link. 3 *Data Type* = RGB666 4 *Data Type* = RGB888
TX Gear	**7**	Specifies the DDR71 gearing used. This attribute is not configurable.
Clock		
TX Total Aggregate Bandwidth (Mbps)	210–7560[1], **3780** 210–7560[2], **3780**	Tx total line rate. This attribute is not configurable.
TX Line Rate per lane (Mbps)	70–**945**[1] 70–1050[2], **945**	Target line rate per lane.
Pixel Clock Frequency (MHz)	10–**135**[1] 10–150[2], **135**	Pixel clock. This attribute is not configurable.
LVDS Output Clock Frequency (MHz)	10–**135**[1] 10–150[2], **135**	LVDS clock. This attribute is not configurable.
LVDS ECLK Frequency (MHz)	35–**472.5**[1] 35–525[2], **472.5**	LVDS edge clock. This attribute is not configurable.
Sync Clock Frequency (MHz)	10–**135**[1] 10–150[2], **135**	Sync clock. Must be equal or slower than the slowest clock in the system.
Miscellaneous		
Register Interface	**Off**, AXI4-Lite	Specifies the register interface used. Not configurable when *Video Data Interface* is set to *AXI4-Stream Rx (Legacy).*
Enable Miscellaneous Signals	**Checked**, Unchecked	Enables miscellaneous signals. When enabled, select internal signals are available to the top-level IP wrapper.
Enable Test Mode	Checked, **Unchecked**	Enables test mode. When enabled, Tx automatically generates test data inside the system specified through *Test Mode Expected Data in Hex Format* and transmits the data.
Test Mode Expected Data in Hex Format	**18'h00000** - 24'hFFFFFF	Test data value used when *Enable Test Mode* is checked.

Notes:

2. Supported range for Nexus devices.
3. Supported range for Avant devices.

3.2. Video

Table 3.2. Video Attributes

Attribute	Selectable Values	Description
Video Information		
Active Width (Pixels)	2–65535, **1920**	Determines the active width of the video streaming data. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Active Height (Lines)	2–65535, **1080**	Determines the active height of the video streaming data. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Horizontal Front Porch (Pixels)	2–65535, **88**	Determines the horizontal front porch. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Horizontal Sync Width (Pixels)	2–65535, **44**	Determines the horizontal sync width. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Horizontal Back Porch (Pixels)	2–65535, **148**	Determines the horizontal back porch. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Vertical Front Porch (Lines)	2–65535, **4**	Determines the vertical front porch. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Vertical Sync Width (Lines)	2–65535, **5**	Determines the vertical sync width. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*
Vertical Back Porch (Lines)	2–65535, **36**	Determines the vertical back porch. Available when *Register Interface* is set to *AXI4-Lite* and *Video Data Interface* is set to *Unified Video Streaming Rx.*

4. Signal Description

This section describes the FPD-Link Tx IP ports.

4.1. FPD-Link Tx Interface

Table 4.1. FPD-Link Tx Interface Ports

Port	Type	Description
Clock and Reset		
rst_n_i	Input	Asynchronous active low system reset.
sync_clk_i	Input	Clock for GDDR_SYNC. Must be slower than all the clocks in the system.
eclk_i	Input	Edge clock for LVDS side.
pix_clk_o	Output	Output pixel clock.
clk_ch0_p_o	Output	Positive LVDS output clock to FPD-Link Tx channel 0.
clk_ch1_p_o	Output	Positive LVDS output clock to FPD-Link Tx channel 1.
Native Pixel Interface		
pix_data0_i	Input	Input pixel data 0. Bus width depends on the data type selected.
pix_data1_i[1,2]	Input	Input pixel data 1. Bus width depends on the data type selected.
de_i	Input	Input data enable for parallel interface.
hsync_i	Input	Input horizontal sync for parallel interface.
vsync_i	Input	Input vertical sync for parallel interface.
OpenLDI/FPD-Link Interface		
d0_ch0_p_o	Output	Positive LVDS output data lane 0 to FPD-Link Tx channel 0.
d1_ch0_p_o	Output	Positive LVDS output data lane 1 to FPD-Link Tx channel 0.
d2_ch0_p_o	Output	Positive LVDS output data lane 2 to FPD-Link Tx channel 0.
d3_ch0_p_o[3]	Output	Positive LVDS output data lane 3 to FPD-Link Tx channel 0.
d0_ch1_p_o[4]	Output	Positive LVDS output data lane 0 to FPD-Link Tx channel 1.
d1_ch1_p_o[4]	Output	Positive LVDS output data lane 1 to FPD-Link Tx channel 1.
d2_ch1_p_o[4]	Output	Positive LVDS output data lane 2 to FPD-Link Tx channel 1.
d3_ch1_p_o[3,4]	Output	Positive LVDS output data lane 3 to FPD-Link Tx channel 1.

Notes:

1. Available only when *Number of Input Pixels per Clock* is more than 1.
2. These pixel data are transmitted by channel 1 when dual channel is selected.
3. Available only when *Data Type* is set to *RGB888*.
4. LVDS channel 1 output ports are not available when single LVDS channel is selected.

4.2. Miscellaneous Status Interface

Table 4.2. Miscellaneous Status Interface Ports

Port	Type	Description
Miscellaneous		
tstmode_en_i[1]	Input	Enables or disables test mode. 0 – Disable test mode 1 – Enable test mode
rdy_o[2]	Output	0 – DDR synchronization has not started or is still in progress. 1 – Indicates that DDR synchronization is already done.

Notes:

1. Available if *Register Interface* is set to *Off* and *Enable Test Mode* is checked. Refer to the Debug Mode section for details.
2. Available if *Register Interface* is set to *Off* and *Enable Miscellaneous Signals* is checked.

4.3. AXI4-Stream Receiver (Legacy) Interface

Table 4.3. AXI4-Stream Receiver (Legacy) Interface Ports[1]

Port	Type	Description
AXI4-Stream Receiver (Legacy) Interface		
axis_tdata_i	Input	AXI4-Stream Receiver interface data mapped from pix_data0_i/pix_data1_i from the Native Pixel interface.
axis_tvalid_i	Input	AXI4-Stream Receiver (Legacy) interface valid data
axis_tuser_i	Input	AXI4-Stream Receiver (Legacy) interface user-defined data input mapped to the Native Pixel interface. [0] – de_i [1] – hsync_i [2] – vsync_i
axis_tready_o	Output	AXI4- Stream Receiver (Legacy) interface ready output signal.

Note:

1. Available if *Video Data Interface* is set to *AXI4-Stream Rx (Legacy)*.

4.4. Unified Video Streaming Rx Interface

Table 4.4. Unified Video Streaming Rx Interface Ports[1]

Port	Type	Description
Unified Video Streaming Rx Interface		
axis_vid_tdata_i	Input	Unified Video Streaming Rx interface data mapped from pix_data0_i or pix_data1_i of the Native Pixel interface.
axis_vid_tvalid_i	Input	Unified Video Streaming Rx interface valid data signal.
axis_vid_tuser_i	Input	Unified Video Streaming Rx interface user-defined additional information data. [0]: Start of Frame 0 – Not the start of new frame 1 – Start of new frame [1]: Unused
axis_vid_tlast_i	Input	Unified Video Streaming Rx Interface signal that indicates end of line of the data.
axis_vid_tready_o	Output	Unified Video Streaming Rx interface ready output signal.

Note:

1. Available if *Video Data Interface* is set to *Unified Video Streaming Rx*.

4.5. AXI4-Lite Interface

Table 4.5. AXI4-Lite Interface Ports[1]

Port	Type	Description
AXI4-Lite Interface		
axil_clk_i	Input	AXI4-Lite input clock. Supported range is 10-135 MHz.
axil_rst_n_i	Input	AXI4-Lite asynchronous active low reset.
s_axil_awvalid_i	Input	AXI4-Lite write valid address signal.
s_axil_awaddr_i	Input	AXI4-Lite write address signal.
s_axil_wvalid_i	Input	AXI4-Lite write valid signal.
s_axil_wdata_i	Input	AXI4-Lite write data signal.
s_axil_wstrb_i	Input	AXI4-Lite write strobe signal.
s_axil_bready_i	Input	AXI4-Lite write response ready signal. Indicates that transfer on the write response channel is accepted.
s_axil_arvalid_i	Input	AXI4-Lite read valid address signal.
s_axil_araddr_i	Input	AXI4-Lite read address signal.
s_axil_rready_i	Output	AXI4-Lite read ready signal. Indicates that transfer on the read data channel is accepted.
s_axil_awready_o	Output	AXI4-Lite write address ready output signal. Indicates that the write valid address signal is asserted, input write address is valid, and transfer is accepted on write address channel.
s_axil_wready_o	Output	AXI4-Lite write ready output signal. Indicates that the data written in the write data signal is valid.
s_axil_bvalid_o	Output	AXI4-Lite write response valid signal.
s_axil_bresp_o	Output	AXI4-Lite write response signal. Indicates the status of a write transaction.
s_axil_arready_o	Output	AXI4-Lite read address ready signal.
s_axil_rvalid_o	Output	AXI4-Lite read valid signal.
s_axil_rdata_o	Output	AXI4-Lite read data signal.
s_axil_rresp_o	Output	AXI4-Lite read response signal.

Note:

1. Available if *Register Interface* is set to *AXI4-Lite*.

5. Register Description

The FPD-Link Transmitter IP supports register programming and status reading. All registers listed are accessible through the AXI4-Lite interface.

5.1. Register Address Map

Table 5.1. Configuration Register Address Map

Address Offset	Name	Description	Access	Default
0x0000	CFG_REG	Configuration register	RW	32'h0
0x0004	TESTMODE_REG	Test mode register	RW	32'h0
0x0008	MISC_REG	Miscellaneous signals register	RO	32'h0
0x0200	ACT_WIDTH	Active width register	RW	32'h0780
0x0204	ACT_HEIGHT	Active height register	RW	32'h0438
0x0208	H_FPORCH	Horizontal front porch register	RW	32'h0058
0x020C	H_SYNCWD	Horizontal sync width register	RW	32'h002C
0x0210	H_BPORCH	Horizontal back porch register	RW	32'h0094
0x0214	V_FPORCH	Vertical front porch register	RW	32'h0004
0x0218	V_SYNCWD	Vertical sync width register	RW	32'h0005
0x0218	V_BPORCH	Vertical back porch register	RW	32'h0024

5.2. FPD-Link Tx Configuration Registers

Table 5.2. CFG_REG Register

Field	Name	Description	Access	Default
[31:9]	rsvd	Reserved	RO	23'h0
[8]	data_frmt	Data mapping format register bit. 0 – VESA 1 – JEIDA	RO	Depends on the *Data Mapping Format* attribute. 1'b1 – JEIDA 1'b0 – VESA
[7:1]	rsvd	Reserved	RO	7'h0
[0]	act_bpc	Dynamic reconfiguration bit for bits per color (BPC). 0 – RGB888 1 – RGB666	RW	Depends on the *Data Type* attribute. 1'b1 – RGB666 1'b0 – RGB888

Table 5.3. TESTMODE_REG Register

Field	Name	Description	Access	Default
[31:25]	rsvd	Reserved	RO	7'h0
[24]	testmode_en	Enable or disable test mode. 0 – Disable test mode 1 – Enable test mode	RW	1
[23:0]	testmode_data	Pre-defined test data that drives channel 0 and channel 1 (if enabled). 24 bits and 18 bits ([17:0]) are used for RGB888 and RGB666, respectively.	RW	0

Table 5.4. MISC_REG Register

Field	Name	Description	Access	Default
[31:1]	rsvd	Reserved	RO	31'h0
[0]	rdy_reg	1-bit DDR synchronization status. 0 – DDR synchronization is not yet started or still in progress. 1 – DDR synchronization is done.	RO	1'h0

Table 5.5. ACT_WIDTH Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	act_width	Active width in units of pixels.	RW	16'h0780

Table 5.6. ACT_HEIGHT Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	act_height	Active height in units of lines.	RW	16'h0438

Table 5.7. H_FPORCH Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	h_fporch	Horizontal front porch in units of pixels.	RW	16'h0058

Table 5.8. H_SYNCWD Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	h_syncwd	Horizontal sync width in units of pixels.	RW	16'h002C

Table 5.9. H_BPORCH Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	h_bporch	Horizontal back porch in units of pixels.	RW	16'h0094

Table 5.10. V_FPORCH Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	v_fporch	Vertical front porch in units of lines.	RW	16'h0004

Table 5.11. V_SYNCWD Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	v_syncwd	Vertical sync width in units of lines.	RW	16'h0005

Table 5.12. V_BPORCH Register

Field	Name	Description	Access	Default
[31:16]	rsvd	Reserved	RO	16'h0
[15:0]	v_bporch	Vertical back porch in units of lines.	RW	16'h0024

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 them into the device architecture. The steps below describe how to generate the FPD-Link Transmitter IP in the Lattice Radiant software.

To generate the FPD-Link Transmitter IP:

1. Create a new Lattice Radiant software project or open an existing project.
2. Click the **IP Catalog** button to view the **IP Catalog** pane.
3. On the **IP on Local** tab, double-click **OpenLDI/FPD-Link/LVDS Transmitter** under the **IP/…/Audio_Video_and_Image_Processing** category. The **Module/IP Block Wizard** opens.

   Note: If the IP is not available on the **IP on Local** tab, download the IP from the **IP on Server** tab.

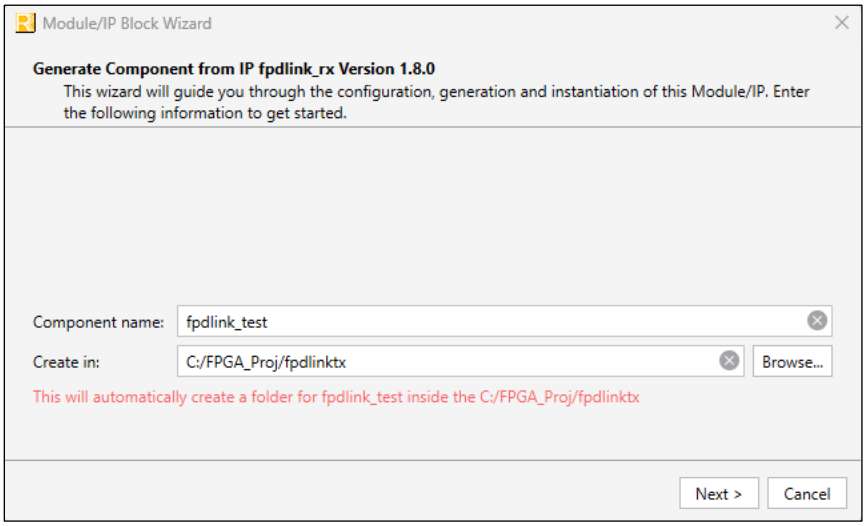

Figure 6.1. Module/IP Block Wizard

4. Enter values in the **Component name** and **Create in** fields, then click **Next**.
5. Customize the selected FPD-Link Transmitter IP using drop-down lists and check boxes. Figure 6.2 shows an example configuration of the FPD-Link Transmitter IP. For details on the configuration options, refer to the IP Parameter Description section.

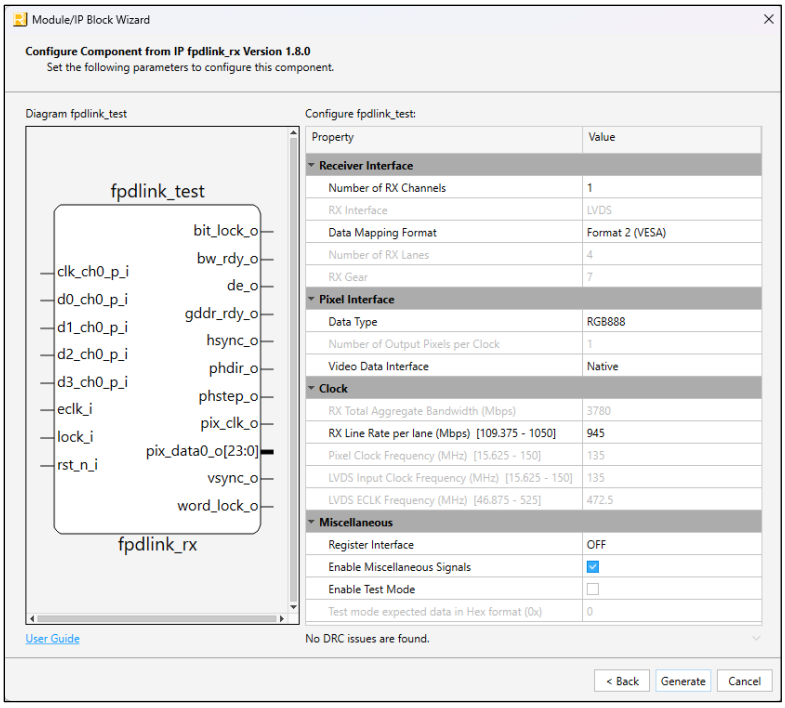

Figure 6.2. IP Configuration

6. Click **Generate**. The **Check Generated Result** window opens. This window shows design block messages and results.

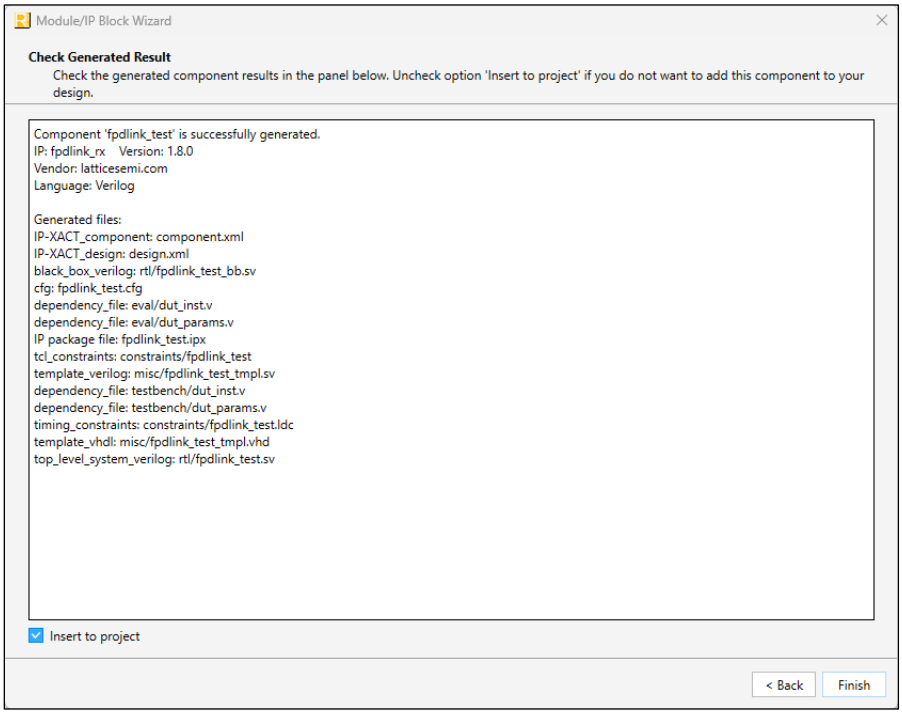

Figure 6.3. Check Generated Result

7. Click **Finish**. All generated files are placed in the directory specified by the **Component name** and **Create in** fields shown in Figure 6.1.

6.1.1. Generated Files and File Structure

The generated FPD-Link Transmitter IP Core 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 IP core is also provided. You may also use this example as the starting template for your top-level design. The generated files are listed in Table 6.1.

Table 6.1. Generated File List

Attribute	Description
<Component Name>.ipx	This file contains information on the files associated with the generated IP.
<Component Name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	This file contains the ipxact: component information of the IP.
design.xml	This file 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.
constraints/<Instance Name>.ldc	This file contains examples on how to constrain the IP pre-synthesis.
eval/dut_inst.v testbench/dut_inst.v	These files provide the list of ports used in the IP configuration.
eval/dut_inst.v testbench/dut_params.v	These files provide the list of parameters used in the IP configuration.

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 FPD-Link Transmitter IP generates the following constraint files:

- A constraint file in SDC format (*<ip_instance_path>/constraints/constraint.sdc*) that contains post-synthesis IP constraints. These constraints are automatically used and propagated by the software tool starting from the Lattice Radiant software version 2024.1.
- A constraint file in PDC format (*<ip_instance_path>/eval/constraint.pdc*) that contains clock constraints. These constraints can be modified according to the clock frequencies in your design.

```
#-------------------------------------------------------------------------------
# SETTINGS
#-----

set IP_INST_LVDS_ECLK_PRD [expr {double(round(1000000/$IP_INST_LVDS_CLK_MHZ))/1000}]
set IP_INST_SYNC_CLK_PRD  [expr {double(round(1000000/$IP_INST_SYNC_CLK_MHZ))/1000}]
set IP_INST_AXI4_LITE_PRD  10

#-------------------------------------------------------------------------------
# CLOCKS
#-----

# -- LVDS ECLK CLOCK --#
create_clock -name {eclk_i} -period $IP_INST_LVDS_ECLK_PRD [get_ports eclk_i]

# -- SYNC CLOCK --#
create_clock -name {sync_clk_i} -period $IP_INST_SYNC_CLK_PRD [get_ports sync_clk_i]


# -- AXI4-LITE CLOCK -- #
if {$IP_INST_AXI4L=="ON"} {
  create_clock -name {axil_clk_i} -period $IP_INST_AXI4_LITE_PRD [get_ports axil_clk_i]
}
```

Figure 6.4. Constraint File in PDC

Note: During synthesis, you can ignore clock related warnings as the IP does not include clock-related constraints at pre-synthesis level.

Note: During post-synthesis, warnings related to dropped constraints may be shown. As the IP supports many configurations and parameter combinations, some default constraints may not be applicable to the selected configuration.

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

6.4. Specifying the Strategy

The Radiant software provides two predefined strategies: area and timing. It 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.

6.5. Running Functional Simulation

You can run functional simulation after the IP is generated.

To run functional simulation:

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

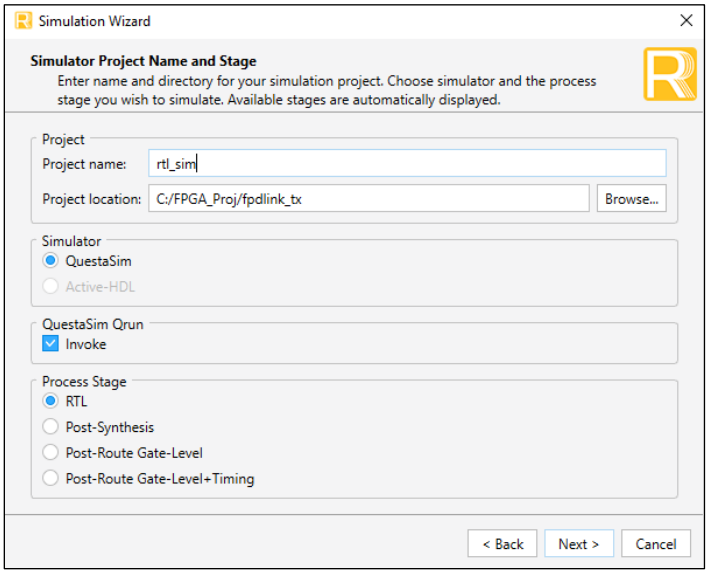

Figure 6.5. Simulation Wizard

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

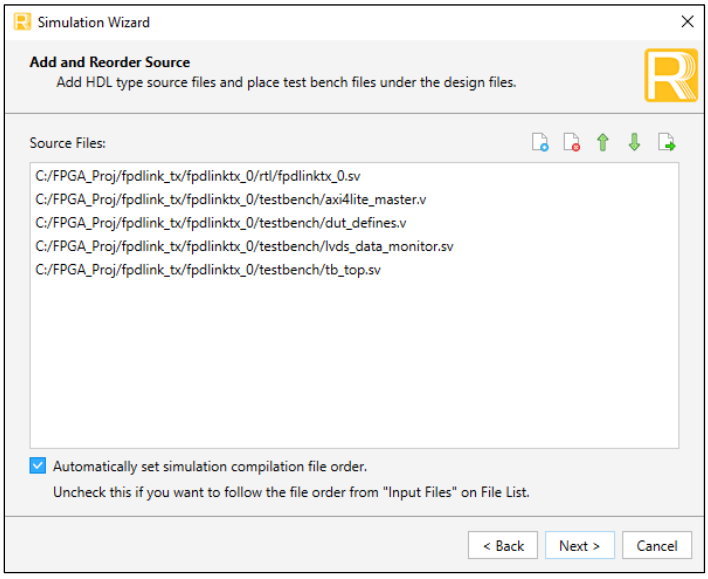

Figure 6.6. Add and Reorder Source

3. Click **Next**. The **Summary** window is shown.
4. Click **Finish** to run the simulation.

The waveform in Figure 6.7 shows an example simulation result.

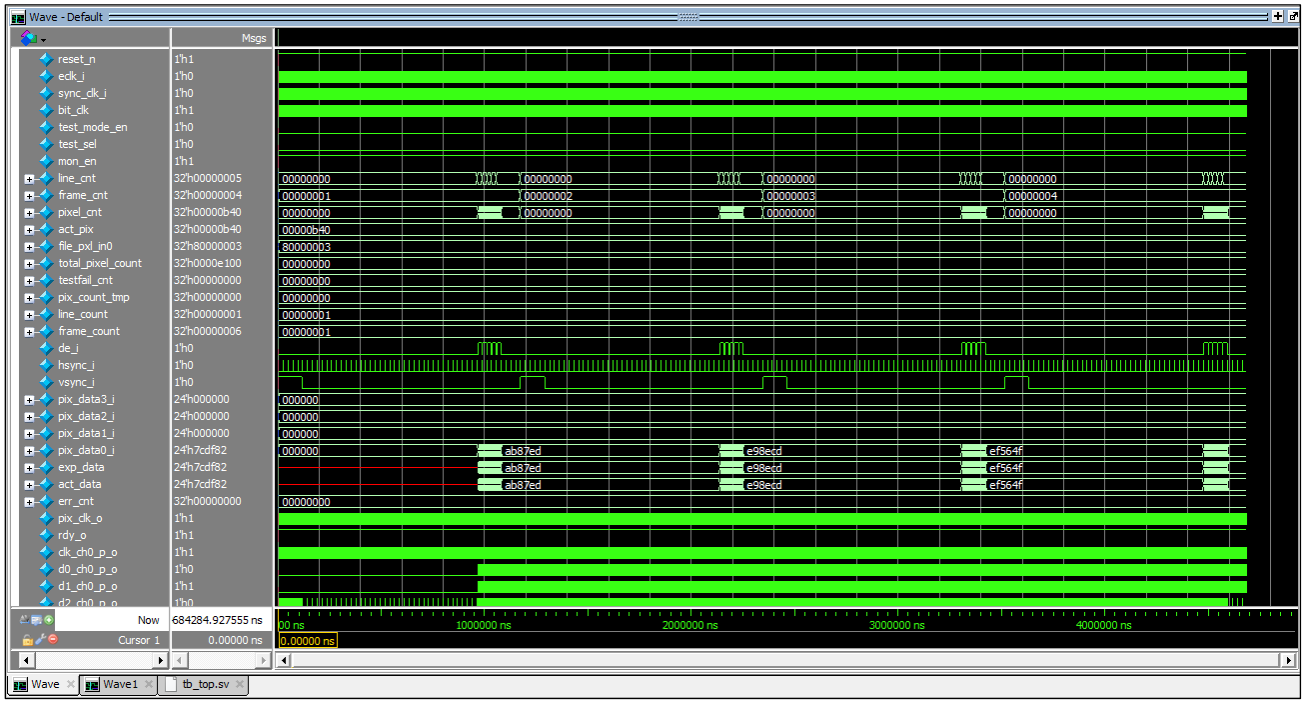

Figure 6.7. Simulation Waveform

6.5.1. Simulation Results

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

```
#### DATA COMPARISON IS STARTED #####
 ------------------------------------------------
 ------------------------------------------------
  Test fail count :            0

 ------------------------------------------------------------
 ------------------ SIMULATION PASSED -----------------------
 ------------------------------------------------------------
 4684284928000 TEST END.

 4684284928000 Simulation is done.
```

Figure 6.8. Passing Simulation Log

7. Debugging

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

7.1. Debug Methods

7.1.1. Debug Mode

This debug feature is used for automatic transmission of test data that is specified through the *Test Mode Expected Data in Hex Format* attribute. The test mode module drives channel 0 and channel 1 (if enabled) with predefined test data. Data comparison should be done outside the design.

To enable test mode:

1. Make sure *Enable Test Mode* is checked and *Test Mode Expected Data in Hex Format (0x)* is configured during IP generation. Pre-defined data is equivalent to {R[7:0], G[7:0], B[7:0]}.
2. Drive tstmode_en_i to 1'b1.
3. Follow this initialization and reset sequence:
    a. Assert active low system reset for at least three clock cycles of the slowest clock (sync clock). It is expected that the input clock is already stable immediately after reset. Clock synchronization starts immediately after the release of system reset.

       Note: Active low system reset is used in the design with synchronous release. This is the system reset input connected to the FPD-Link Tx module.
    b. If *Enable Miscellaneous Signals* is checked, wait for rdy_o to be asserted. rdy_o is used to indicate that the FPD-Link Tx clock synchronization is done. Only when rdy_o is asserted can valid data be sampled and correctly transmitted by the FPD-Link Tx IP.
    c. If *Enable Miscellaneous Signals* is unchecked, wait for rdy_o to be asserted before sending the valid data to allow Tx clock synchronization to complete.
4. Monitor data transmission over the LVDS lanes (test data is sent continuously with DE = high, VSYNC = HSYNC = RSVD = 0).

8. Design Considerations

8.1. Limitations

- For dual channel configuration, odd multiple number of pixels is not supported.
- For dual channel configuration, only the clock from channel 0 (clk_ch0_p_o) is used.
- AXI4-Lite Response signals (BRESP, RRESP) only supports 2'b00 (OKAY).
- AXI4-Stream Rx (Legacy) does not support back-pressure.
- AXI4-Stream Rx (Legacy) does not support the AXI4-Lite interface.
- Dynamic reconfiguration is only supported for Native Pixel and Unified Video Streaming interfaces.
- Resetting the AXI4-Lite interface (axil_rst_n_i) separately from system reset (rst_n_i) causes replay read in the AXI4-Lite interface. s_axil_bvalid_o is asserted until all data are flashed out. If this is not intended, always toggle axil_rst_n_i and rst_n_i together.

Appendix A. Resource Utilization

Table A.1 and Table A.2 show the device information, tool information, and resource utilization of the FPD-Link Transmitter IP core on a CertusPro-NX device. Table A.3 and Table A.4 show the device information, tool information, and resource utilization of the FPD-Link Transmitter IP on a Lattice Avant device. Table A.5 and Table A.6 show the device information, tool information, and resource utilization of the FPD-Link Transmitter IP on a Certus-N2 device. The default configuration is used, and some attributes are changed from the default value to show the effect on resource utilization.

Table A.1. CertusPro-NX Device and Tool Information

Software Version	Lattice Radiant software 2025.2 production build
Device Used	LFCPNX-100-7LFG672C
Performance Grade	7_High-Performance_1.0V
Synthesis Tool	Synplify Pro, October 2025

Table A.2. Resource Utilization on CertusPro-NX Device

Configuration	LVDS Clk (MHz)	Sync Clk (MHz)	Video Data Interface	Register Interface	f_{MAX} (MHz)[1]	Slice Registers	LUTs	EBRs	High Speed I/O Resources
Default	945	10	Native	Off	200	18	28	0	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	Native	AXI4-Lite	200	682	707	0	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Default	945	10	Legacy	Off	200	18	31	0	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	Legacy	Off	200	18	31	0	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Default	945	10	UVSI	Off	200	155	243	1	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	UVSI	AXI4-Lite	200	876	2077	1	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC

Note:

1. f_{MAX} is generated using multiple iterations of place and route.

Table A.3. Avant Device and Tool Information

Software Version	Lattice Radiant software 2025.2 production build
Device Used	LAV-AT-E70ES1-1LFG676C
Performance Grade	1
Synthesis Tool	Synplify Pro, October 2025

Table A.4. Resource Utilization on Avant Device

Configuration	LVDS Clk (MHz)	Sync Clk (MHz)	Video Data Interface	Register Interface	f_{MAX} (MHz)[1]	Slice Registers	LUTs	EBRs	High Speed I/O Resources
Default	945	10	Native	Off	250	18	28	0	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	Native	AXI4-Lite	250	682	685	0	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Default	945	10	Legacy	Off	250	18	31	0	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	Legacy	Off	250	18	31	0	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Default	945	10	UVSI	Off	250	131	222	1	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	UVSI	AXI4-Lite	250	838	1979	1	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC

Note:

1. f_{MAX} is generated using multiple iterations of place and route.

Table A.5. Certus-N2 Device and Tool Information

Software Version	Lattice Radiant software 2025.2 production build
Device Used	LN2-CT-20ES-1ASG410C
Performance Grade	1
Synthesis Tool	Synplify Pro, October 2025

Table A.6. Resource Utilization on Certus-N2 Device

Configuration	LVDS Clk (MHz)	Sync Clk (MHz)	Video Data Interface	Register Interface	f_{MAX} (MHz)	Slice Registers	LUTs	EBRs	High Speed I/O Resources
Default	945	10	Native	Off	250	18	28	0	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	Native	AXI4-Lite	250	682	685	0	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Default	945	10	Legacy	Off	250	18	31	0	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	Legacy	Off	250	18	31	0	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Default	945	10	UVSI	Off	250	131	222	1	5 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC
Number of TX Channels = 2, *Data Type* = RGB666, Others = Default	945	135	UVSI	AXI4-Lite	250	838	1979	1	8 x ODDR71, 1 x ECLKDIV, 1 x ECLKSYNC

References

- OpenLDI/FPD-Link/LVDS Transmitter IP Release Notes (FPGA-RN-02012)
- Avant-E web page
- Avant-G web page
- Avant-X web page
- Certus-N2 web page
- Certus-NX web page
- CertusPro-NX web page
- CrossLink-NX web page
- MachXO5-NX web page
- Lattice Radiant FPGA design software
- Lattice Solutions IP Cores web page
- Lattice Solutions Reference Designs web page
- Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059)
- 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 1.6, IP v1.8.0, December 2025

Section	Change Summary
All	• Added a note on the IP version in the Quick Facts and Revision History sections. • Made minor editorial fixes.

Introduction	• In Table 1.1. Summary of the FPD-Link Transmitter IP:
	  • Updated IP core version and software version in *Lattice Implementation*.
	• In the Licensing and Ordering Information section:
	  • Updated the licensing and ordering information.
	  • Removed the Ordering Part Number section.

Designing with the IP	• Added note on IP version in GUI.
	• In the Generating and Instantiating the IP section:
	  • Split Step 2 into Steps 2 and 3.
	  • Updated Steps 2 through 7.
	  • Updated Figure 6.1. Module/IP Block Wizard, Figure 6.2. IP Configuration, and Figure 6.3. Check Generated Result.

Appendix A. Resource Utilization	• In Table A.1. CertusPro-NX Device and Tool Information and Table A.3. Avant Device and Tool Information:
	  • Updated software version.
	  • Updated synthesis tool date.
	• In Table A.5. Certus-N2 Device and Tool Information:
	  • Updated software version.
	  • Updated device used.
	  • Updated synthesis tool date.
	• In Table A.2. Resource Utilization on CertusPro-NX Device:
	  • Updated LUTs for native interface with 135-MHz Sync Clk, UVSI interface with 10-MHz Sync Clk, and UVSI interface with 135-MHz Sync Clk.
	  • Added note on f_{MAX} in relation to place and route.
	• In Table A.4. Resource Utilization on Avant Device:
	  • Updated LUTs for native interface with 135-MHz Sync Clk and UVSI interface with 10-MHz Sync Clk.
	  • Updated Slice Registers and LUTs for UVSI interface with 135-MHz Sync Clk.
	  • Added note on f_{MAX} in relation to place and route.
	• In Table A.6. Resource Utilization on Certus-N2 Device:
	  • Updated LUTs for native interface with 135-MHz Sync Clk and UVSI interface with 10-MHz Sync Clk.
	  • Updated Slice Registers and LUTs for UVSI interface with 135-MHz Sync Clk.

Revision 1.5, IP v1.7.0, July 2025

Section	Change Summary

Introduction	• In Table 1.1. Summary of the FPD-Link Transmitter IP:
	  • Renamed *Supported FPGA Family* to *Supported Devices* and incorporated *Targeted Devices* information into row.
	  • Removed *Targeted Devices* row.
	  • Added IP core version to *Lattice Implementation*.
	• In Table 1.3. Ordering Part Number:
	  • Updated header name from *Multi-Site Perpetual* to *Single Seat Perpetual*.

Section	Change Summary
Designing with the IP	Updated Figure 6.1. Module/IP Block Wizard, Figure 6.2. IP Configuration, Figure 6.3. Check Generated Result, Figure 6.5. Simulation Wizard, and Figure 6.6. Add and Reorder Source.

Appendix A. Resource Utilization	• In Table A.1. CertusPro-NX Device and Tool Information:
	  • Updated software version.
	  • Updated synthesis tool date.
	• In Table A.2. Resource Utilization on CertusPro-NX Device:
	  • Updated f_{MAX}, slice registers, and LUTs for native interface with 135-MHz Sync Clk and UVSI interface with 10- and 135-MHz Sync Clk.
	  • Updated high speed I/O resources for legacy and UVSI interfaces with 135-MHz Sync Clk.
	• In Table A.3. Avant Device and Tool Information and Table A.5. Certus-N2 Device and Tool Information:
	  • Updated software version.
	  • Updated device used.
	  • Updated synthesis tool date.
	• In Table A.4. Resource Utilization on Avant Device:
	  • Removed term *Pixel* from *Native* interface.
	  • Updated f_{MAX}, slice registers, and LUTs for native interface with 135-MHz Sync Clk and UVSI interface with 135-MHz Sync Clk.
	  • Updated slice registers and LUTs for UVSI interface with 10-MHz Sync Clk.
	  • Updated high speed I/O resources for native interface with 10-MHz Sync Clk and legacy interface with 135-MHz Sync Clk.
	• In Table A.6. Resource Utilization on Certus-N2 Device:
	  • Removed term *Pixel* from *Native* interface.
	  • Updated slice registers and LUTs for native interface with 135-MHz Sync Clk and UVSI interface with 10- and 135-MHz Sync Clk.

Revision 1.4, IP v1.6.0, December 2024

Section	Change Summary
Cover	Added IP version.
Abbreviations in This Document	Updated section title, description, and table header.

Introduction	• Added Certus-N2 to list of targeted devices in description.
	• In Table 1.1. Summary of the FPD-Link Transmitter IP:
	  • Added Certus-N2 to *Supported FPGA Family*.
	  • Added IP Changes row.
	  • Added Lattice Implementation row and moved IP version information into *Lattice Implementation*.
	  • Added IP core v1.6.0 to *Lattice Implementation*.
	  • Removed IP Version row.
	  • Added LFD2NX-9, LFD2NX-28, and LN2-CT-20 to *Targeted Devices.*
	• Added the IP Support Summary section.
	• Removed the IP Validation Summary section.
	• Added part numbers for Certus-N2 in Table 1.3. Ordering Part Number.
	• Added the Hardware Support section.

Functional Description	Updated Figure 2.1. FPD-Link Transmitter IP Block Diagram.
Signal Description	Updated description for *axis_vid_tlast_i* in Table 4.4. Unified Video Streaming Rx Interface Ports1.
Designing with the IP	Updated Figure 6.1. Module/IP Block Wizard, Figure 6.2. IP Configuration, and Figure 6.3. Check Generated Result.
Appendix A	• Updated description. • Added Table A.5. Certus-N2 Device and Tool Information and Table A.6. Resource

Section	Change Summary
	Utilization on Certus-N2 Device.
References	• Added OpenLDI/FPD-Link/LVDS Transmitter IP Release Notes and Certus-N2 web page. • Removed Lattice Radiant Software User Guide.

Revision 1.3, June 2024

Section	Change Summary
All	Made editorial changes.
Acronyms and Abbreviations in This Document	• Renamed section. • Added items.

Introduction	• Reworked section content and added the following subsections:
	  • 1.1 Overview of the IP
	  • 1.4 Licensing and Ordering Information
	  • 1.5 IP Validation Summary
	  • 1.6 Minimum Device Requirements
	• Reworked subsection IP Evaluation and section Ordering Part Number into subsection 1.4 Licensing and Ordering Information.
	• Updated IP Version, Supported User Interface, and Lattice Implementation in Table 1.1. Summary of the FPD-Link Transmitter IP in subsection 1.2 Quick Facts.
	• Updated key features in subsection 1.3 Features.

Functional Description	• Updated and reworked section content and added the following subsections:
	  • 2.1 IP Architecture Overview
	  • 2.2 Clocking
	  • 2.3 Reset
	  • 2.4 User Interfaces
	  • 2.5 Other IP Specific Blocks/Layers/Interfaces

IP Parameter Description	• Added this section. • Reworked subsection Attribute Summary and added into this section.
Signal Description	• Added this section. • Reworked subsection Signal Description and added into this section.
Register Description	Added this section.
Designing with the IP	• Added this section. • Reworked section IP Generation, Simulation, and Validation and added into this section.
Debugging	• Added this section. • Reworked subsection Debug Mode and added into this section.
Design Considerations	• Added this section. • Updated limitations and moved into this section. • Added limitations.
Appendix A	Reworked and updated this section.

Revision 1.2, December 2023

Section	Change Summary
All	• Retitled document from OpenLDI/FPD-LINK/LVDS Transmitter IP Core - Lattice Radiant Software to OpenLDI/FPD-LINK/LVDS Transmitter IP Core. • Made minor fixes to comply with Lattice's inclusive language guidelines.
Disclaimers	Updated boilerplate.
Inclusive Language	Added this section.
Introduction	Updated Target Devices and Lattice Implementation in Table 1.1. OpenLDI/FPD-LINK/LVDS Transmitter IP Quick Facts.
Functional Descriptions	• In Table 2.1. OpenLDI/FPD-LINK/LVDS Transmitter IP Core Signal Description, removed

Section	Change Summary
	duplicate rows for *axis_tuser_i*, *axis_tready_o*, *axis_tdata_i*, and *axis_tvalid_i* and updated description for *axis_tuser_i*. • Updated Figure 2.1. Functional Block Diagram and Table 2.1. OpenLDI/FPD-LINK/LVDS Transmitter IP Core Signal Description. • Updated table header from *Dependency on Other Attributes* to *Description* in Table 2.3. • Updated figure titles for Figure 2.9, Figure 2.10, Figure 2.13, and Figure 2.14. • Added Figure 2.5. Single Channel OpenLDI/FPD-LINK/LVDS Output Bus Waveform for RGB888 Format (VESA), Figure 2.6. Dual Channel OpenLDI/FPD-LINK/LVDS Output Bus Waveform for RGB888 Format (VESA), Figure 2.7. Single Channel OpenLDI/FPD-LINK/LVDS Output Bus Waveform for RGB888 Format (JEIDA), Figure 2.8. Dual Channel OpenLDI/FPD-LINK/LVDS Output Bus Waveform for RGB888 Format (JEIDA), and Figure 2.15. AXI4-Stream Receiver Interface Diagram. • Updated description in Module Description. • Added section 2.7.5. AXI4-Stream Receiver Interface.
IP Generation, Simulation, and Validation	• Updated Figure 3.1., Figure 3.3., Figure 3.4., and Figure 3.5. • Updated Table 3.1. Generated File List. • Updated step 2 in the Running Functional Simulation section. • Updated the Constraining the IP section. • Added Table 3.2. Testbench Files Description. • Updated the IP Evaluation section. • Updated Figure 3.6. Simulation Waveform and its corresponding notes.
Ordering Part Number	Updated list of ordering part numbers.
Resource Utilization	Updated *LAV-AT-500E-3LFG1156C* to *LAV-AT-E70-3LFG1156C*.
References	Updated this section with relevant links.

Revision 1.1, May 2023

Section	Change Summary
All	Added support for CertusPro-NX, MachXO5-NX, and Lattice Avant devices support.
Introduction	• Added support for CertusPro-NX, MachXO5-NX, and Lattice Avant devices in the general description. • Updated Supported FPGA Families, Target Devices, and Resources in Table 1.1. OpenLDI/FPD-LINK/LVDS Transmitter IP Quick Facts reflecting the new devices support.
Functional Description	• Added AXI4-Stream Slave Interface data to Table 2.1. OpenLDI/FPD-LINK/LVDS Transmitter IP Core Signal Description. • Updated Table 2.2. Attributes Table and Table 2.3. Attributes Description.
IP Generation, Simulation, and Validation	• Updated section header to the current. • Removed the original Licensing the IP section. • Changed the section header to the current Generating the IP. • Updated Figure 3.2. Configure User Interface of selected OpenLDI/FPD-LINK/LVDS Transmitter IP Core. • Changed the Constraining the IP section header to the current. • Updated the description showing the updated .ldc file location.
Ordering Part Number	Added Ordering Part Number (OPN) for CertusPro-NX, MachXO5-NX, and Lattice Avant families.

Revision 1.0, March 2023

Section	Change Summary
All	Initial release.

www.latticesemi.com