MIPI Migration Guide

Application Note

FPGA-AN-02119-1.0

July 2026

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
APB	Advanced Peripheral Bus
AXI4-Lite	Advanced eXtensible Interface 4 Lite
B2P	Byte-to-Pixel
BPC	Bits per Color
BPP	Bits per Pixel
CPP	Colors per Pixel
CRC	Cyclic Redundancy Check
CSI	Camera Serial Interface
CSR	Control and Status Register
D-PHY	MIPI D-PHY Physical Layer
DSI / DSI-2	Display Serial Interface / Display Serial Interface-2
ECC	Error Correction Code
FE	Frame End
FIFO	First In, First Out
FPGA	Field-Programmable Gate Array
FS	Frame Start
FSM	Finite State Machine
GUI	Graphical User Interface
HS	High-Speed
I/O	Input/Output
IP	Intellectual Property
LE	Line End
LMMI	Lattice Memory Mapped Interface
LP	Low Power
LS	Line Start
MBSI	MIPI Byte Streaming Interface
MIPI	Mobile Industry Processor Interface
P2B	Pixel-to-Byte
PDC	Physical Design Constraints
PHY	Physical Layer
PLL	Phase-Locked Loop
PPC	Pixels per Clock
RTL	Register Transfer Level
RX	Receive
SDC	Synopsys Design Constraints
SOF	Start of Frame
SoTp	Start of Transmission Pattern
TX	Transmit
UVSI	Unified Video Streaming Interface
WC	Word Count

1. Introduction

The new unified MIPI CSI/DSI IP consolidates four legacy Lattice IPs into a single configurable core that covers both receive (RX) and transmit (TX) flows in the same Lattice Radiant™ package. The unified IP simplifies integration through standardized data and control interfaces, reducing implementation effort and improving user experience.

This guide helps engineers who are already familiar with the legacy IPs to move an existing design to the new IP while keeping the application logic largely intact. Note that you can only perform the migration using the Lattice Radiant software version 2025.2 and later.

Table 1.1. Referenced IP Version

IP Name	Version
MIPI CSI/DSI IP	4.1.0
CSI-2/DSI D-PHY Receiver IP	2.2.0
CSI-2/DSI D-PHY Transmitter IP	2.4.1
Byte-to-Pixel Converter IP	1.9.3
Pixel-to-Byte Converter IP	1.9.3

Note: This document is limited to the scope of the MIPI CSI/DSI IP version listed in the table above. Additional features and enhancements are planned for future releases of the MIPI CSI/DSI IP and will be documented in subsequent revisions of this guide.

Throughout this guide, the term *Legacy* or *Old IPs* refers to any of the four legacy IPs (CSI-2/DSI D-PHY Receiver IP, CSI-2/DSI D-PHY Transmitter IP, Byte-to-Pixel Converter IP, Pixel-to-Byte Converter IP), and *New* refers to the MIPI CSI/DSI IP.

1.1. Key Differences

- The new MIPI CSI/DSI IP is a single parameterized core. A top-level *PHY Direction* parameter selects RX or TX; you no longer instantiate two legacy cores for a full flow.
- The pixel-domain interface is the Unified Video Streaming Interface (UVSI) (axis_vid_*) AXI4-Stream. No native-pixel interface is available on the unified IP.
- The byte-domain interface is the MIPI Byte Streaming Interface (MBSI) (axis_byte_*) AXI4-Stream. No native-byte interface is available on the unified IP.
- Packet parsing (RX) and packet formatting (TX) are built in. These features are always present and not user-toggleable.

1.2. Architecture Comparison

1.2.1. TX Path

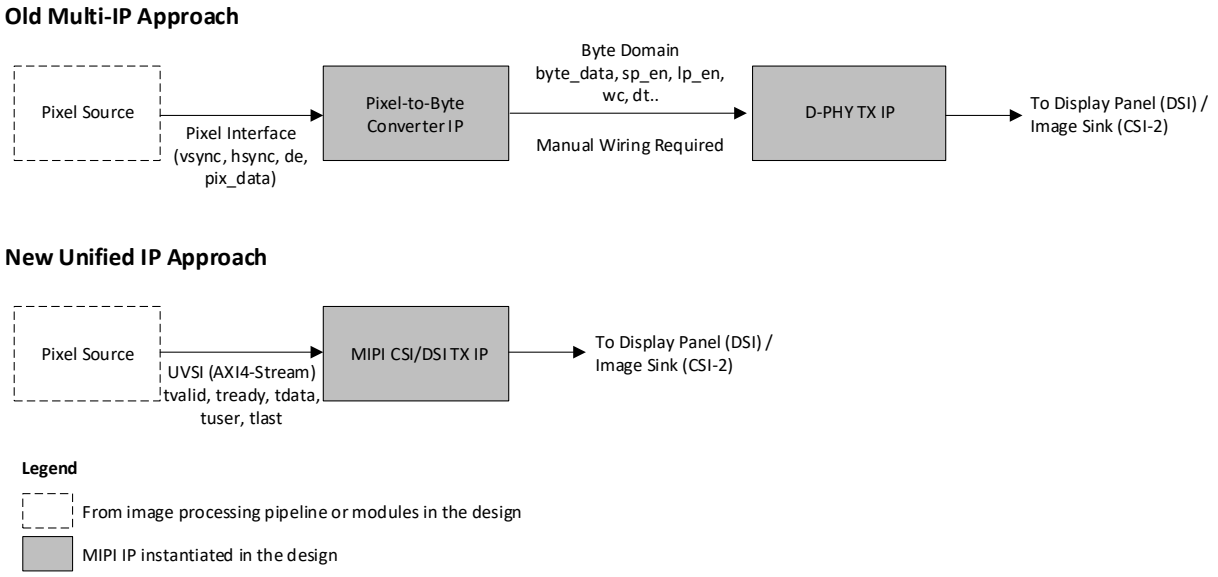


Figure 1.1. TX Path Architecture Comparison

1.2.2. RX Path

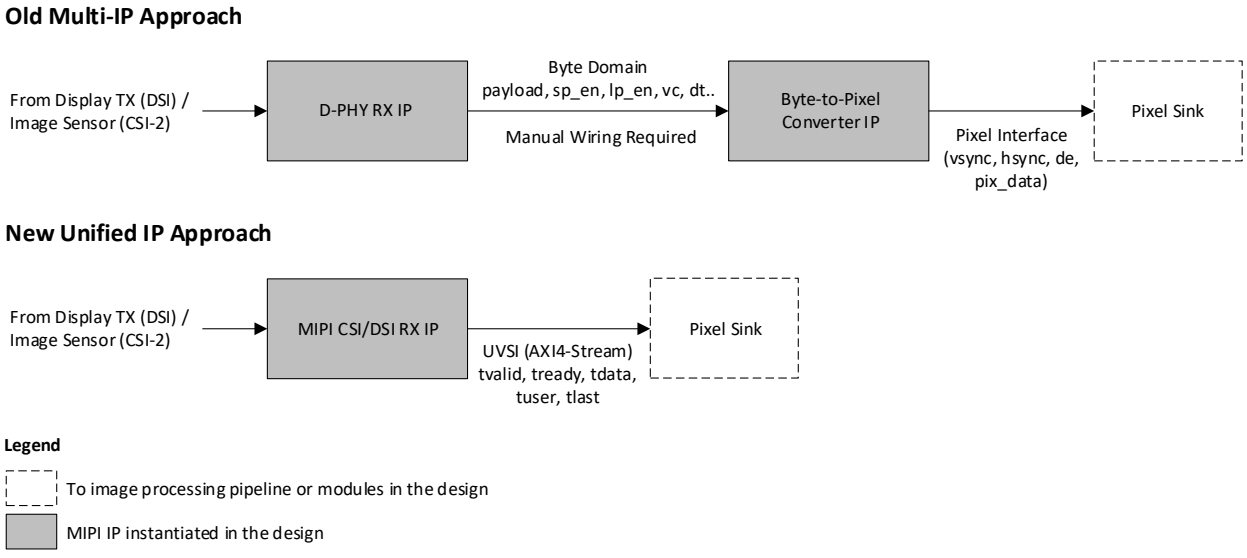


Figure 1.2. RX Path Architecture Comparison

2. Migration Scope

2.1. Overview

Before migrating, you must verify that your existing design configuration is supported by the new MIPI CSI/DSI IP. This section provides a comprehensive eligibility check covering device families, PHY type, lane count, data rate, color format, and feature dependencies.

2.2. Supported Device Families

Table 2.1. Supported Device Families

Device Family	Old IPs	New MIPI CSI/DSI IP
CrossLink™-NX	Supported	Supported
Certus™-NX	Supported	Supported
CertusPro™-NX	Supported	Supported
MachXO5™-NX	Supported	Supported
Lattice Avant™	Supported	Available in a future release
Certus-N2	Supported	Available in a future release

Note: Migration support for designs targeting Lattice Avant and Certus-N2 devices is planned for a future release.

2.3. Supported Configurations Comparison

Table 2.2. Supported Configurations Comparison

Parameter	Old IPs	New MIPI CSI/DSI IP	Migration Impact
PHY Mode	Soft D-PHY, Hard D-PHY (CrossLink-NX)	Soft D-PHY, Hard D-PHY (CrossLink-NX)	No Impact
TX Line Rate	Soft: up to 1,500 Mbps; Hard: up to 2,500 Mbps	Soft: up to 1,500 Mbps; Hard: up to 2,500 Mbps	No impact
RX Line Rate	Soft: up to 1,500 Mbps; Hard: up to 2,500 Mbps	Soft: up to 1,500 Mbps; Hard: up to 2,500 Mbps	No impact
TX Protocol	CSI-2, DSI	CSI-2 only	DSI TX cannot migrated
RX Protocol	CSI-2, DSI	CSI-2, DSI-2	Supported
Register Access	LMMI, AXI4-Lite, APB	AXI4-Lite	Change the register access mechanism
Dynamic Reconfiguration	Available	Available	No impact
RX FIFO Type	OFF/Single/PingPong/Queue	Line buffer (UVSI) Byte buffer (MBSI)	Verify internal buffering is adequate
RX Error Detection	ECC/CRC checking available through either LMMI or native signals	ECC/CRC checking available only through register access	Access registers through AXI4-Lite to query the status

2.4. Supported Color Formats

Table 2.3. Supported Color Formats

Color Format	Old IPs	New MIPI CSI/DSI IP
RGB888	Supported	Supported
RGB565	Supported	Supported
RAW10	Supported	Supported

Color Format	Old IPs	New MIPI CSI/DSI IP
RAW12	Supported	Supported
RAW8	Supported	Available in a future release
RAW14	Supported	Available in a future release
RAW16	Supported	Available in a future release
RGB666	Supported	Available in a future release
RGB444	Supported	Available in a future release
RGB555	Supported	Available in a future release
YUV420 8-bit	Supported	Available in a future release
YUV420 10-bit	Supported	Available in a future release
YUV422 8-bit	Supported	Available in a future release
YUV422 10-bit	Supported	Available in a future release

Note: Designs using color formats not supported by the MIPI CSI/DSI IP v4.1.0 cannot migrate to the new IP; migration support is planned for a future release.

2.5. Word Count Alignment Requirement

2.5.1. TX Word Count

On the transmit path, the new MIPI CSI/DSI IP derives the Word Count (the long-packet payload length, in bytes, that the IP writes into each MIPI packet header) automatically from the active video resolution and the selected color format. Word Count is not a user-entered parameter; no manual Word Count entry or adjustment is required during migration.

The new MIPI CSI/DSI IP does not require the Word Count to be a multiple of 32 bytes. The internal pixel-to-byte data path and line buffer operate on 256-bit (32-byte) words, but the IP correctly handles resolutions which Word Count is not a multiple of 32 by transmitting the final partial word. When the Word Count is not a multiple of 32 bytes, ensure a minimum gap of four axis_vid_clk_i cycles between successive packet streams. Refer to the *Design Considerations* section in the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).

A minimum Word Count applies on the transmit path. Refer to the TX Minimum Word Count section.

The table below illustrates how the Word Count is computed for the supported color formats.

Table 2.4. Word Count Determined by the Video Resolution and Color Format

Format	Word Count Formula	Width (px)	Example Word Count (Bytes)
RGB888	Width × 3	1920	1920 × 3 = 5760
RGB565	Width × 2	1920	1920 × 2 = 3840
RAW10	Width × 10 / 8	1920	1920 × 10 / 8 = 2400
RAW12	Width × 12 / 8	1920	1920 × 12 / 8 = 2880
RGB888	Width × 3	1366	1366 × 3 = 4098
RAW10	Width × 10 / 8	800	800 × 10 / 8 = 1000

Note: The Word Count does not need to be a multiple of 32 bytes. The 1366-pixel RGB888 (4098-byte) and 800-pixel RAW10 (1000-byte) rows are examples of valid, non-32-multiple Word Counts that the IP transmits correctly, subject to the four-cycle inter-packet gap.

2.5.2. RX Word Count

On the receive path, the new MIPI CSI/DSI IP enforces a minimum Word Count on each incoming long packet. The receiver depacketizer enforces a minimum Word Count requirement based on the selected configuration. The minimum Word Count depends on the protocol and the controller mode and applies to the supported streaming interfaces. Refer to the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321). The old IPs do not impose minimum Word

Count. For designs that rely on small and long packets (for example, narrow regions of interest or short DSI-2 packets), check before migrating. The table below lists the minimum Word Count for each configuration.

Table 2.5. RX Minimum Word Count Limitation

PHY Mode	Protocol	Controller Mode	Minimum Word Count (Bytes)
Soft D-PHY or Hard D-PHY	DSI-2	32-bit	14
Soft D-PHY or Hard D-PHY	CSI-2	32-bit	6
Hard D-PHY	DSI-2	64-bit	34
Hard D-PHY	CSI-2	64-bit	10

Note:

1. 64-bit controller mode is only supported with Hard D-PHY in a 4-lane configuration.

Note: Before migrating an RX design, verify that the smallest long packet your source produces meets or exceeds the applicable minimum Word Count in the table above. Refer to the *MIPI CSI/DSI Receiver Word Count Support Limitation* table in the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).

3. Functional Area Migration

3.1. IP GUI Configuration Mapping

This section maps the old IP GUI parameters to the equivalents in the new MIPI CSI/DSI IP.

3.1.1. TX Path: Old IP (D-PHY TX IP and Pixel-to-Byte IP) to New IP (MIPI CSI/DSI IP)

Table 3.1. D-PHY TX IP Parameters

Old Parameter (D-PHY TX IP)	New Parameter (MIPI CSI/DSI IP)	Notes
TX Interface Type (CSI-2/DSI)	Protocol Mode	CSI-2 only for TX in new IP
Number of TX Lanes (1–4)	Number of D-PHY Lanes	1, 2, or 4 only (no 3-lane)
TX Line Rate	Line Rate Per Lane (Mbps)	Soft D-PHY: 160–1,500 Mbps Hard D-PHY: 320–2,500 Mbps
D-PHY Clock Mode (Continuous/Non-Continuous)	PHY Clock Mode	Same concept
Packet Formatter (Enabled/Disabled)	—	Always integrated; no separate option
LMMI Interface (Enabled/Disabled)	Enable AXI4-Lite Interface	Register access has changed from LMMI to AXI4-Lite
Protocol timing (tLPX, tHS-PREPARE, and so on)	Protocol timing parameters	Similar names in new IP GUI

Table 3.2. Pixel-to-Byte IP Parameters

Old Parameter (Pixel-to-Byte IP)	New Parameter (MIPI CSI/DSI IP)	Notes
Data Type (RGB888 and so on)	Enable <format> Color Format checkboxes	Checkbox instead of drop-down box
TX Gear (8/16)	PHY Data Width (8/16)	Fixed at 8 for Soft D-PHY, 16 for Hard D-PHY
Number of TX Lanes	Number of D-PHY Lanes	Inherited from PHY configuration
Word Count	—	Automatically derived from video resolution and format from UVSI stream
FIFO Depth/Width/Threshold	Line Buffer Depth	Simplified single parameter
AXI4-Stream Enable/Disable	—	UVSI is always the interface
APB Interface	Enable AXI4-Lite Interface	Register access has changed from APB to AXI4-Lite
Manual Adjust FIFO settings	—	Removed; not available, as the UVSI is capable of back pressure

Note:

1. These parameters are valid only when the Unified Video Streaming Interface (UVSI) is checked while the MIPI Byte Streaming Interface (MBSI) is unchecked.

3.1.2. RX Path: Old IP (D-PHY RX IP and Byte-to-Pixel IP) to New IP (MIPI CSI/DSI IP)

Table 3.3. D-PHY RX IP Parameters

Old Parameter (D-PHY RX IP)	New Parameter (MIPI CSI/DSI IP)	Notes
RX Interface Type (CSI-2/DSI)	Protocol Mode	CSI-2 or DSI-2
Number of RX Lanes (1–4)	Number of D-PHY Lanes	1, 2, or 4 only
RX Line Rate	Line Rate Per Lane (Mbps)	Soft D-PHY: 80–1,500 Mbps Hard D-PHY: 160–2,500 Mbps
Packet Parser (ON/OFF)	—	Always integrated; no separate option
LMMI Interface (Enabled/Disabled)	Enable AXI4-Lite Interface	Register access has changed from LMMI to AXI4-Lite

Old Parameter (D-PHY RX IP)	New Parameter (MIPI CSI/DSI IP)	Notes
Protocol timing parameters	Protocol timing parameters	Similar names in new IP GUI

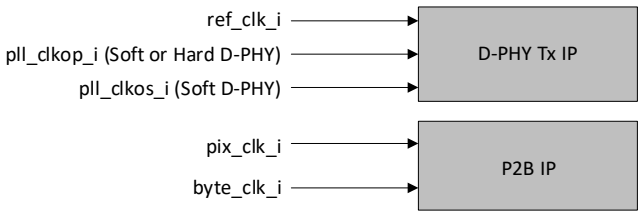
Table 3.4. Byte-to-Pixel IP Parameters

Old Parameter (Byte-to-Pixel IP)	New Parameter (MIPI CSI/DSI IP)	Notes
Data Type (RGB888 and so on)	Enable <format> Color Format checkboxes	Select one of the supported formats
RX Gear (8/16)	PHY Data Width (8/16)	Fixed at 8 for Soft D-PHY, 16 for Hard D-PHY
FIFO settings	Pixel Buffer Depth	Simplified single parameter
AXI4-Lite Interface	Enable AXI4-Lite Interface	Same interface, but register address is different
Debug Ports	—	Removed; not available

3.2. Clocking Migration

This section describes how to set up clocking with the new MIPI CSI/DSI IP. The AXI4-Lite clocking (s_axi_aclk_i) is omitted in this section.

3.2.1. TX Clocking



(byte_clk is often derived from external PLL or D-PHY Tx output)

Figure 3.1. TX Clocking for Old IPs (D-PHY TX IP and Pixel-to-Byte IP)

Table 3.5. Requirements for Old IPs

Signal	Description
ref_clk_i	D-PHY TX reference clock.
pix_clk_i	Pixel-to-Byte pixel-domain clock.
byte_clk_i	Pixel-to-Byte byte-domain clock (typically routed from D-PHY TX or an external PLL).
pll_clkop_i	Primary input reference clock. Available when D-PHY TX IP = *Soft D-PHY* and *Enable Edge Clock Synchronizer and Divider* = checked, or D-PHY TX IP = *Hard D-PHY* and *D-PHY PLL Mode* = *External*.
pll_clkos_i	Primary input reference clock with 90 Degree phase offset. Available when D-PHY TX IP = *Soft D-PHY*.

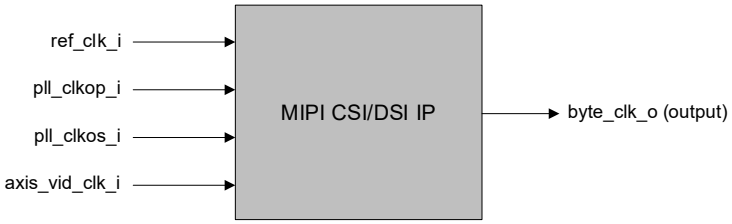


Figure 3.2. TX Clocking for New IP (MIPI CSI/DSI IP)

Table 3.6. Requirements for New IP

Signal	Description
ref_clk_i	Reference clock for D-PHY. Must be 60–200 MHz.
pll_clkop_i	Primary input reference clock. Available only for Soft D-PHY.
pll_clkos_i	Primary input reference clock with 90 degree phase offset. Available only for Soft D-PHY.
axis_vid_clk_i	Pixel/video clock for the UVSI interface. Available only in the UVSI mode.
byte_clk_o	Output clock (no user routing required). Equals *Line Rate Per Lane (Mbps) / PHY Data Width*. The *Actual Byte Clock Frequency* operates in the range 10–187.5 MHz (typical design point is 125 MHz) for the new IP.

Key difference: The new IP exposes axis_vid_clk_i as the pixel/video clock input and provides byte_clk_o as an output. For the application that works only with pixel data, you no longer need to route a byte clock between IPs. The byte_clk_o can be used for byte data manipulation when using the IP in MBSI mode, only if the clock is free running (that is, D-PHY clock lane is in continuous clock mode) and higher than 60 MHz.

3.2.2. RX Clocking

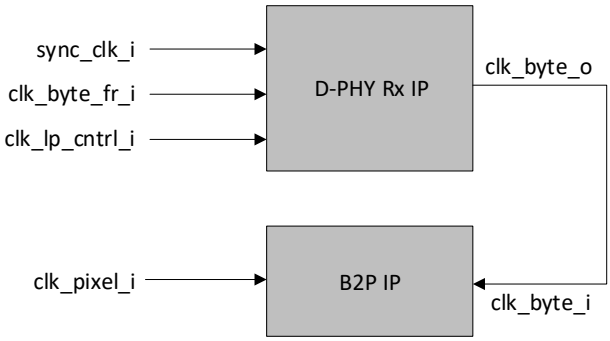


Figure 3.3. RX Clocking for Old IPs (D-PHY RX IP and Byte-to-Pixel IP)

Table 3.7. Requirements for Old IPs

Signal	Description
sync_clk_i	Low speed or oscillator clock
clk_byte_fr_i	D-PHY RX byte-domain free-running clock
clk_lp_ctrl_i	D-PHY RX low power (LP) control clock
clk_byte_o	D-PHY RX byte clock output, routed to B2P clk_byte_i
clk_pixel_i	B2P pixel-domain clock

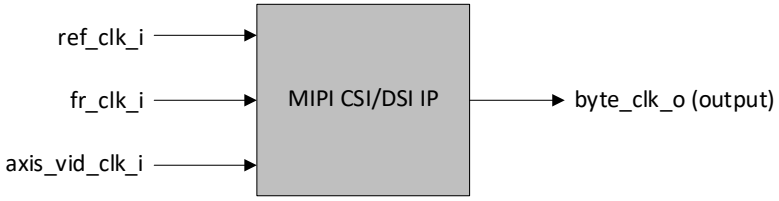


Figure 3.4. RX Clocking for New IP (MIPI CSI/DSI IP)

Table 3.8. Requirements for New IP

Signal	Description
ref_clk_i	Reference clock for D-PHY. Must be 60–200 MHz.
fr_clk_i	Free-running clock (replaces clk_byte_fr_i and clk_lp_ctrl_i). Must be ≥ *Actual Byte Clock Frequency (MHz)*, minimum 60 MHz.
axis_vid_clk_i	Pixel/video clock for the UVSI interface (replaces B2P clk_pixel_i). Must be ≥ *Actual Byte Clock Frequency (MHz)*, and ≥ (*Number of D-PHY Lanes* × *PHY Data Width*) / (*Number of Pixel per Clock (PPC)* × BPP) × *Actual Byte Clock Frequency (MHz)* when that quotient is greater than the byte-clock value.
byte_clk_o	Geared down output clock from the D-PHY clock lane. Equals *Line Rate Per Lane (Mbps)* / *PHY Data Width*.

Key difference: fr_clk_i consolidates the functions of clk_byte_fr_i and clk_lp_ctrl_i. The separate pixel clock domains previously managed across the B2P IP are replaced by a single axis_vid_clk_i.

3.3. Reset Migration

3.3.1. TX Reset

Table 3.9. TX Reset for Old IPs

Signal	IP	Description
reset_n_i	D-PHY TX	Active-low reset for D-PHY TX
rst_n_i	P2B	Active-low reset for P2B (internally generates reset_byte and reset_pixel)

Table 3.10. TX Reset for New IP

Signal	Description
ref_rst_n_i	Active-low reset for the reference clock domain
axis_vid_rst_n_i	Active-low reset for the UVSI / pixel clock domain

Guidance: Both resets must be asserted together when performing a full reset during an active transaction. There is no required sequencing between the two domains, but all domains must be fully flushed before de-asserting reset.

3.3.2. RX Reset

Table 3.11. RX Reset for Old IPs

Signal	IP	Description
reset_n_i	D-PHY RX	Active-low reset for D-PHY RX
reset_byte_n_i	B2P	Byte-domain reset
reset_pixel_n_i	B2P	Pixel-domain reset
axis_sresetn_i	B2P	AXI4-Stream target reset (if AXI4-Stream interface is used)
axis_mresetn_i	B2P	AXI4-Stream controller reset (if AXI4-Stream interface is used)

Table 3.12. RX Reset for New IP

Signal	Description
ref_rst_n_i	Active-low reset for the reference clock domain
fr_rst_n_i	Active-low reset for the free-running clock domain
axis_vid_rst_n_i	Active-low reset for the UVSI / pixel clock domain

Guidance: All three resets must be asserted together for a complete reset. The consolidated reset scheme replaces the complex multi-reset arrangement of the B2P IP. No specific sequencing is required, but all domains must be flushed.

3.4. User Interface Migration to UVSI

3.4.1. UVSI Overview

The new IP uses the Unified Video Streaming Interface (UVSI), based on the AXI4-Stream protocol. This replaces the variety of interface options available in the old IPs (native pixel interface, AXI4-Stream). Use this interface to work with the video data in the pixel format.

UVSI handshaking follows standard AXI4-Stream conventions as follows:

- All UVSI signals are synchronous to axis_vid_clk_i.
- A beat transfers on rising edges of axis_vid_clk_i where tvalid and tready are both 1.
- When tvalid goes high, the signal must stay high, and the payload/qualifiers must not change, until the ready handshake completes. The source may retract tvalid only after the handshake.
- axis_vid_tuser[0] marks start-of-frame and is valid only on the first accepted beat of a frame.
- axis_vid_tlast marks end-of-line and must be asserted on the last beat of each image line.
- axis_vid_tdata carries Number of Pixel per Clock (PPC) packed pixels per beat. The bus width is controlled by the internal TDATA_WIDTH and depends on PPC and the enabled format checkboxes.

For comprehensive details on UVSI signal handshaking and behavior for both TX and RX, refer to the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).

3.4.2. TX Signal Mapping

Table 3.13. TX Interface Signals for Old IPs and New UVSI Signals

Old Signal	Old IP	New UVSI Signal	Notes
vsync_i	P2B	axis_vid_tuser_i[0]	SOF — assert on first pixel of frame
hsync_i	P2B	(derived from blanking)	Horizontal sync is implicit in UVSI
de_i (data enable)	P2B	axis_vid_tvalid_i	Asserts when pixel data is valid
fv_i (frame valid)	P2B	axis_vid_tuser_i[0]	Maps to SOF
lv_i (line valid)	P2B	axis_vid_tlast_i	Asserts on last pixel of each line
pix_data_i	P2B	axis_vid_tdata_i	Pixel data; see TDATA mapping in the TDATA Pixel Mapping section
—	—	axis_vid_tready_o	New signal, backpressure from IP to user

The table below shows the signals you managed in the old IPs. These signals are now internal in the new IP.

Table 3.14. Eliminated TX Signals

Old Signal	Old IP	Purpose	Status in New IP
byte_data_i	D-PHY TX	Byte data input	Internal
byte_data_en_i	D-PHY TX	Byte data enable	Internal
sp_en_i	D-PHY TX	Short packet enable	Auto-generated by Video FSM
lp_en_i	D-PHY TX	Long packet enable	Auto-generated by Video FSM
vc_i	D-PHY TX	Virtual channel	Hard-Coded to 0
dt_i	D-PHY TX	Data type	Internal
wc_i	D-PHY TX	Word count	Auto-derived
d_hs_en_i	D-PHY TX	HS mode enable	Internal
c2d_ready_o	D-PHY TX	Controller-to-device ready	Replaced by TREADY
ready_o	D-PHY TX	Ready indicator	Replaced by TREADY

Old Signal	Old IP	Purpose	Status in New IP
ld_pyld_o	D-PHY TX	Load payload indicator	Internal
pkt_format_ready_o	D-PHY TX	Packet formatter ready	Internal
d_hs_rdy_o	D-PHY TX	HS ready indicator	Internal
byte_data_o	P2B	Byte data to D-PHY	Internal
byte_en_o	P2B	Byte enable to D-PHY	Internal
wc_o	P2B	Word count to D-PHY	Internal
data_type_o	P2B	Data type to D-PHY	Internal
txfr_req_o	P2B	Transfer request to D-PHY	Internal

3.4.3. RX Signal Mapping

Table 3.15. RX Interface Signals for Old IPs and New UVSI Signals

Old Signal	Old IP	New UVSI Signal	Notes
vsync_o	B2P	axis_vid_tuser_o[0]	SOF on first pixel of frame
hsync_o	B2P	(derived from blanking)	Implicit in UVSI
de_o (data enable)	B2P	axis_vid_tvalid_o	Asserts when pixel data is valid
fv_o (frame valid)	B2P	axis_vid_tuser_o[0]	Maps to SOF
lv_o (line valid)	B2P	axis_vid_tlast_o	Asserts on last pixel of line
pd_o / pd_red_o / pd_green_o / pd_blue_o	B2P	axis_vid_tdata_o	Pixel data; see TDATA mapping in the TDATA Pixel Mapping section
—	—	axis_vid_tready_i	New signal, you must assert to accept data

The table below shows the signals that are visible to you in the old IPs. These signals are now internal in the new IP.

Table 3.16. Eliminated RX Signals

Old Signal	Old IP	Purpose	Status in New IP
payload_o	D-PHY RX	Byte payload output	Internal
payload_en_o	D-PHY RX	Payload enable	Internal
sp_en_o	D-PHY RX	Short packet indicator	Internal
lp_en_o	D-PHY RX	Long packet indicator	Internal
lp_av_en_o	D-PHY RX	Long packet active video enable	Internal
vc_o	D-PHY RX	Virtual channel output	Internal
dt_o	D-PHY RX	Data type output	Internal
wc_o	D-PHY RX	Word count output	Internal

3.4.4. TDATA Pixel Mapping

Understanding the TDATA bus width is critical for correct migration. The width depends on the color format and the number of pixels per clock (PPC).

Table 3.17. TDATA Width per Format

Format	CPP	BPC	BPP	Bits per Pixel Slot (ceil(BPP/8) × 8)	TDATA_WIDTH (PPC=1)	TDATA_WIDTH (PPC=2)	TDATA_WIDTH (PPC=4)
RGB888	3	8	24	24	24	48	96
RGB565	3	8 (actual 5, 6)	24	24	24	48	96
RAW10	1	10	10	16 (padded)	16	32	64

Format	CPP	BPC	BPP	Bits per Pixel Slot (ceil(BPP/8) × 8)	TDATA_WIDTH (PPC=1)	TDATA_WIDTH (PPC=2)	TDATA_WIDTH (PPC=4)
RAW12	1	12	12	16 (padded)	16	32	64

Notes:

1. TDATA_WIDTH = ceil(BPP / 8) × 8 × PPC.
2. RAW10/RAW12: Each pixel occupies a 16-bit slot with zero-padding in the upper bits. For RAW10, bit[9:0] carry data and bit[15:10] are zero. For RAW12, bit[11:0] carry data and bit[15:12] are zero. Refer to the *Unified Video Streaming Interface (UVSI)* section in the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).
3. RGB565: Each color component occupies an 8-bit slot, where the lower bits are zero-padding. Refer to the *Unified Video Streaming Interface (UVSI)* section in the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).

Comparison with the old interface:

In the old B2P/P2B IPs, pixel data was presented on separate buses (for example, pd_red_o[7:0], pd_green_o[7:0], pd_blue_o[7:0] for RGB888) or as a packed pd_o bus. In the new IP, all pixel data for all PPC pixels is packed into a single axis_vid_tdata bus. You must update your pixel data packing and unpacking logic accordingly.

3.4.5. TX UVSI Behavior

The new TX IP includes an internal line buffer with configurable depth. The line buffer behavior affects the UVSI backpressure as follows:

- With sufficient horizontal blanking: The line buffer has time to drain between lines. The IP does not assert backpressure (axis_vid_tready_o remains high), and you can stream pixels continuously during active video.
- Without sufficient horizontal blanking (or continuous streaming): The line buffer may fill up, causing the IP to deassert axis_vid_tready_o. The user logic must pause pixel transmission when tready goes low and resume when tready goes high again.

You must correctly size the *Line Buffer Depth* based on your video resolution and blanking intervals. An undersized buffer with insufficient blanking causes frequent backpressure events.

3.4.6. RX UVSI Behavior

The new RX IP includes an internal pixel buffer with configurable depth. Key considerations are as follows:

- You must assert axis_vid_tready_i to accept incoming pixel data.
- If axis_vid_tready_i remains low for too long, the internal pixel buffer overflows and data is lost.
- You must ensure your downstream logic can consume pixels fast enough to prevent overflow.

3.5. User Interface Migration to MBSI

3.5.1. MBSI Overview

The new IP offers another interface called the MIPI Byte Streaming Interface (MBSI), based on the AXI4-Stream protocol. This replaces the variety of interface options available in the old IPs (native byte interface, AXI4-Stream). Use this interface to work with the video data in the raw byte format.

MBSI handshaking follows standard AXI4-Stream conventions as follows:

- All MBSI signals are synchronous to fr_clk_i when *PHY Direction* is set to *RX*, and axis_vid_clk_i when *PHY Direction* is set to *TX*. Both clocks are used to operate with video data in the byte domain.
- A beat transfers on rising edges of fr_clk_i where tvalid and tready are both 1.
- When tvalid goes high, the signal must stay high, and the payload/qualifiers must not change, until the ready handshake completes. The source may retract tvalid only after the handshake.
- axis_byte_tuser[5:2] or axis_byte_tuser[43:40] marks the virtual channel ID of the current burst.
- axis_byte_tlast marks end of the high-speed packet, which is the last byte of the CRC footer.
- axis_byte_tdata carries the full MIPI packet, including the packet header and CRC footer. Note the SoTp (B8 pattern) is not included in the axis_byte_tdata.

For comprehensive details on MBSI signal handshaking and behavior for both TX and RX, refer to the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).

3.5.2. TX Signal Mapping

Table 3.18. TX Interface Signals for Old IPs and New MBSI Signals

Old Signal	Old IP	New MBSI Signal	Notes
byte_data_i	D-PHY TX	axis_byte_tdata_i	Includes packet header and CRC footer, without SoTp and HS-Trail
dt_i	D-PHY TX	axis_byte_tdata_i	Part of TDATA
wc_i	D-PHY TX	axis_byte_tdata_i	Part of TDATA
byte_data_en_i	D-PHY TX	axis_byte_tvalid_i	Replaced by TVALID
ready_o	D-PHY TX	axis_byte_tready_i	Replaced by TREADY
c2d_ready_o	D-PHY TX	axis_byte_tready_i	Replaced by TREADY
vc_i	D-PHY TX	axis_byte_tuser_i[3:2] or axis_byte_tuser_i[41:40]	Replaced by TUSER
vcx_i	D-PHY TX	axis_byte_tuser_i[5:4] or axis_byte_tuser_i[43:42]	Replaced by TUSER
—	—	axis_byte_tready_o	New signal, backpressure from IP to user

The table below shows the signals you managed in the old IPs. These signals are now internal in the new IP.

Table 3.19. Eliminated TX Signals

Old Signal	Old IP	Purpose	Status in New IP
sp_en_i	D-PHY TX	Short packet enable	Auto-generated based on TDATA
lp_en_i	D-PHY TX	Long packet enable	Auto-generated based on TDATA
d_hs_en_i	D-PHY TX	HS mode enable	Internal
ld_pyld_o	D-PHY TX	Load payload indicator	Internal
pkt_format_ready_o	D-PHY TX	Packet formatter ready	Internal
d_hs_rdy_o	D-PHY TX	HS ready indicator	Internal

3.5.3. RX Signal Mapping

Table 3.20. RX Interface Signals for Old IPs and New MBSI Signals

Old Signal	Old IP	New MBSI Signal	Notes
payload_o	D-PHY RX	axis_byte_tdata_o	Includes packet header and CRC footer, without SoTp and HS-Trail
dt_o	D-PHY RX	axis_byte_tdata_o	Part of TDATA
wc_o	D-PHY RX	axis_byte_tdata_o	Part of TDATA
payload_en_o	D-PHY RX	axis_byte_tvalid_o	Replaced by TVALID
vc_o	D-PHY RX	axis_byte_tuser_o[3:2] or axis_byte_tuser_i[41:40]	Replaced by TUSER
vcx_o	D-PHY RX	axis_byte_tuser_o[5:4] or axis_byte_tuser_i[43:42]	Replaced by TUSER
—	—	axis_byte_tready_i	New signal, you must assert to accept data

The table below shows the signals that are visible to you in the old IPs. These signals are now internal in the new IP.

Table 3.21. Eliminated RX Signals

Old Signal	Old IP	Purpose	Status in New IP
sp_en_o	D-PHY RX	Short packet indicator	Internal
lp_en_o	D-PHY RX	Long packet indicator	Internal
lp_av_en_o	D-PHY RX	Long packet active video enable	Internal

3.5.4. ECC handling in TDATA

When the *PHY Direction* attribute is set to *TX*, and the *ECC Insert (TX) / ECC Check (RX)* attribute is unchecked, you are responsible for calculating the ECC values and embedding them correctly as part of TDATA. If the *ECC Insert (TX) / ECC Check (RX)* attribute is checked, you must populate the ECC field with placeholder or dummy values. The transmitter then recalculates the correct ECC values with the remaining packet header information.

When the *PHY Direction* attribute is set to *RX*, and the *ECC Insert (TX) / ECC Check (RX)* attribute is unchecked, you must manually calculate the ECC from the remaining packet header and compare it with the received ECC field. When *ECC Insert (TX) / ECC Check (RX)* attribute is checked, you can then access the CSR at address 0x020 through the AXI4-Lite interface to check for any ECC errors.

3.5.5. CRC handling in TDATA

When the *PHY Direction* attribute is set to *TX*, and the *CRC Insert (TX) / CRC Check (RX)* attribute is unchecked, you are responsible for calculating the CRC values and embedding them correctly as part of TDATA. If the *CRC Insert (TX) / CRC Check (RX)* attribute is checked, you must populate the CRC field with placeholder or dummy values. The transmitter then recalculates the correct CRC values with the remaining payload information.

When the *PHY Direction* attribute is set to *RX*, and the *CRC Insert (TX) / CRC Check (RX)* attribute is unchecked, must manually calculate the CRC from the remaining packet header and compare it with the received CRC field. When *CRC Insert (TX) / CRC Check (RX)* attribute is checked, you can then access the CSR at address 0x020 through the AXI4-Lite interface to check for any CRC errors.

3.5.6. TX MBSI Behavior

The new IP with the transmitter mode includes an internal byte buffer with configurable depth. The byte buffer behavior affects the MBSI backpressure as follows:

- With sufficient horizontal blanking: The byte buffer has time to drain between lines. The IP does not assert backpressure (axis_byte_tready_o remains high), and you can stream byte continuously during active video.
- Without sufficient horizontal blanking (or continuous streaming): The byte buffer may fill up, causing the IP to deassert axis_byte_tready_o. The user logic must pause byte transmission when tready goes low and resume when tready goes high again.

You must correctly size the *Byte Buffer Depth* based on your video resolution and blanking intervals. An undersized buffer with insufficient blanking causes frequent backpressure events. Ensure the operating frequency of axis_vid_clk_i is equal to the *Actual Byte Clock Frequency (MHz)* as reported in the GUI.

3.5.7. RX MBSI Behavior

The new RX IP includes an internal byte buffer with configurable depth. Key considerations are as follows:

- You must assert axis_byte_tready_i to accept incoming byte data.
- If axis_byte_tready_i remains low for too long, the internal byte buffer overflows and data is lost.
- You must ensure your downstream logic can consume byte data fast enough to prevent overflow. Ensure the operating frequency of fr_clk_i is set to be greater than or equal to the *Actual Byte Clock Frequency (MHz)* attribute, with a minimum value of 60 MHz.

4. Migration Considerations

4.1. New Requirements

The following features and behaviors exist in the new IP but do not exist (or work differently) in the old IPs. You must account for these differences during migration.

4.1.1. Line Buffer Depth Configuration (TX with UVSI)

The new TX IP includes an internal line buffer with a configurable **Line Buffer Depth** parameter when you choose to use the UVSI interface to transmit video data. You must correctly size this buffer based on your video resolution and horizontal blanking intervals. Refer to the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321) for the formula.

- A larger buffer provides more tolerance for blanking variations but consumes more FPGA resources.
- An undersized buffer with insufficient blanking causes frequent backpressure (TREADY deassertion).
- The old P2B IP has a FIFO with separate depth, width, and threshold parameters, which are replaced by the single *Line Buffer Depth* setting.

4.1.2. Pixel Buffer Depth Configuration (RX with UVSI)

The new RX IP includes an internal pixel buffer with a configurable **Pixel Buffer Depth** parameter when you choose to use the UVSI interface to receive video data.

- You must ensure your downstream logic does not hold axis_vid_tready_i low for extended periods.
- If the pixel buffer overflows, incoming data is lost.
- The old B2P IP has separate FIFO configuration options, which are replaced by the single *Pixel Buffer Depth* setting.

4.1.3. Byte Buffer Depth Configuration (TX or RX with MBSI)

The new IP includes an internal byte buffer with a configurable **Byte Buffer Depth** parameter when you choose to use the MBSI interface.

- A larger buffer is more robust for a wide range of applications but consumes more FPGA resources.
- When setting *PHY Direction* to *TX,* an undersized buffer with insufficient blanking causes frequent backpressure (TREADY deassertion).
- When setting *PHY Direction* to *RX*, you must ensure your downstream logic does not hold axis_byte_tready_i low for extended periods.
- If the byte buffer overflows, incoming data is lost.
- The legacy MIPI CSI-2/DSI Receiver IP has separate FIFO configuration options (that is, depth, type, and read delay), and now all are automatically handled by the new MIPI IP.

4.1.4. Video FSM and Automatic Sync Packet Generation (TX)

The new TX IP includes a built-in Video FSM that automatically generates the following packets:

- Frame Start (FS) short packets at the beginning of each frame
- Frame End (FE) short packets at the end of each frame
- Line Start (LS) short packets (optional, configurable)
- Line End (LE) short packets (optional, configurable)

When using the old IPs, you are responsible for manually driving the D-PHY TX signals sp_en_i, lp_en_i, vc_i, dt_i, and wc_i to trigger sync packet generation. This is no longer necessary when using the new IP.

Table 4.1. Configuration Parameters for Video FSM using the New IP

Parameter	Description
Number of Active Lines Per Frame	Tells the Video FSM how many lines constitute a frame
Line Start and Line End	Enables or disables LS or LE short packet insertion
Line Number Insertion	Enables or disables line number in LS packets

4.1.5. TX Minimum Word Count

The new MIPI CSI/DSI IP enforces a minimum Word Count on the transmit path. This IP does not impose a 32-byte alignment requirement; the Word Count is derived automatically from the video resolution and color format, as described in the Word Count Alignment Requirement section. The minimum Word Count is 32 bytes for the supported transmit configurations (Soft D-PHY with CSI-2 in 32-bit controller mode, and Hard D-PHY with CSI-2 in 64-bit controller mode) and applies to the MBSI and UVSI streaming interfaces. Refer to the *MIPI CSI/DSI Transmitter Word Count Support Limitation for MBSI and UVSI* table in the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321). For typical video resolutions, the Word Count is far above this minimum, hence most migrating designs are unaffected. Designs that transmit small and long packets must verify that the smallest Word Count meets the minimum.

4.1.6. TDATA size of the MBSI

In the legacy MIPI IPs (CSI-2/DSI D-PHY Receiver IP or CSI-2/DSI D-PHY Transmitter IP), the byte data width (for example, payload_i or byte_data_i) is dependent on the number of lanes and gearing at compile time. To seamlessly support dynamic reconfiguration for number of lanes, the new MIPI CSI/DSI IP fixes the TDATA width to either be 32-bit or 64-bit wide. 64-bit is only applicable when selecting Hard D-PHY with 4 lanes at compile time.

4.2. Timing and Physical Constraints

4.2.1. Timing Constraints (SDC)

The new IP generates different SDC constraint files compared to the old IPs. During migration, follow these steps:

1. Review any user-defined timing constraints that referenced old IP clock or signal names.
2. Update the clock or signal names to reference the new IP signal names.

4.2.2. Physical Constraints (PDC)

MIPI pin assignments and I/O type definitions between the new and legacy IPs must be compatible with each other.

5. Example Migration

This example compares the old two-IP RX approach with the new single-IP approach for a 2-lane CSI-2 RAW10 sensor at Line Rate Per Lane = 800 Mbps, gear = 8, PPC = 1.

5.1. Configuration for Old IP

Table 5.1. CSI-2/DSI D-PHY Receiver IP Parameters (Subset)

Parameter	Value
RX Interface Type	CSI-2
D-PHY RX IP	Soft D-PHY
Number of RX Lanes	2
RX Gear	8
RX Line Rate (Mbps)	800
Sync Clock Frequency (MHz)	60 (minimum for Soft D-PHY)
Enable AXI4-Stream Interface	Unchecked
Enable Packet Parser	Checked

Table 5.2. Byte-to-Pixel Converter IP Parameters (Subset)

Parameter	Value
Data Type	RAW10
RX Interface	CSI-2
Number of RX Lanes	2
RX Gear	8
Byte Clock Frequency (MHz)	100 (= 800 / 8)
Number of Output Pixel Lanes	1
Pixel Clock Frequency (MHz)	160
Pixel-Side Transmitter Interface	Native Interface

Sketch instantiation (wrapper ports only):

```
// Legacy: CSI-2/DSI D-PHY Receiver (Soft D-PHY, CSI-2, 2 lanes, gear 8)
csi2_dsi_dphy_rx u_dphy_rx (
    .clk_p_io        (mipi_clk_p),
    .clk_n_io        (mipi_clk_n),
    .d_p_io          (mipi_d_p[1:0]),
    .d_n_io          (mipi_d_n[1:0]),
    .sync_clk_i      (sync_clk_60mhz),
    .clk_byte_fr_i   (fr_byte_clk),
    .reset_byte_fr_n_i(fr_rst_n),
    .clk_byte_o      (byte_clk),
    // Parser outputs (Native byte interface, AXI4 disabled)
    .payload_en_o    (rx_payload_en),
    .payload_o       (rx_payload),       // [NUM_RX_LANE*RX_GEAR-1:0] = [15:0]
    .dt_o            (rx_dt),            // [5:0]
    .vc_o            (rx_vc),            // [1:0]
    .wc_o            (rx_wc),            // [15:0]
    .sp_en_o         (rx_sp_en),
    .lp_en_o         (rx_lp_en),
    .lp_av_en_o      (rx_lp_av_en)
    // ... other optional ports omitted ...
);
```

```
// Legacy: Byte to Pixel Converter (RAW10, Native pixel TX)
byte2pixel u_b2p (
    .clk_byte_i      (byte_clk),
    .reset_byte_n_i  (byte_rst_n),
    .payload_en_i    (rx_payload_en),
    .payload_i       (rx_payload),
    .dt_i            (rx_dt),
    .wc_i            (rx_wc),
    .sp_en_i         (rx_sp_en),
    .lp_av_en_i      (rx_lp_av_en),
    .clk_pixel_i     (pix_clk_160mhz),
    .reset_pixel_n_i (pix_rst_n),
    // Native pixel output (CSI-2 sync style)
    .vsync_o         (vid_vsync),
    .hsync_o         (vid_hsync),
    .de_o            (vid_de),
    .pd_o            (vid_pixel)         // [PD_BUS_WIDTH*NUM_TX_CH - 1 : 0]
    // ...
);
```

The user design must route byte_clk from the D-PHY receiver into the Byte-to-Pixel Converter, manage a separate pixel_clk, and carry the payload_o, dt_o, wc_o, vc_o, sp_en_o, lp_en_o, lp_av_en_o byte-domain bundle intact between the two IPs.

5.2. Configuration for New IP

Table 5.3. MIPI CSI/DSI IP Parameters (Subset, RX Direction, RAW10 Only)

Parameter	Value
PHY Direction	RX
PHY Mode	Soft D-PHY
PHY Data Width	8
Number of D-PHY Lanes	2
Line Rate Per Lane (Mbps)	800
Protocol Mode	CSI-2
PHY Clock Mode	Continuous
Reference Clock Frequency (MHz)	60
Enable RGB888 Color Format	Unchecked
Enable RGB565 Color Format	Unchecked
Enable RAW10 Color Format	Checked
Enable RAW12 Color Format	Unchecked
Number of Pixel per Clock (PPC)	1
Pixel Buffer Depth	2048
Maximum Buffered Pixels	2048

Sketch instantiation (wrapper ports only):

```
// New: MIPI CSI/DSI IP (PHY Direction = RX, CSI-2, 2 lanes, Soft D-PHY, RAW10, PPC=1)
mipi_csi_dsi u_mipi (
    // D-PHY lanes
    .d_p_io            (mipi_d_p[1:0]),
    .d_n_io            (mipi_d_n[1:0]),
    .clk_p_io          (mipi_clk_p),
    .clk_n_io          (mipi_clk_n),
```

```
    // Reference and free-running clocks for Soft D-PHY RX
    .fr_clk_i          (fr_byte_clk),      // >= Actual Byte Clock, >= 60 MHz
    .fr_rst_n_i        (fr_rst_n),
    .ref_clk_i         (refclk),
    .ref_rst_n_i       (ref_rst),
    // UVSI pixel-domain output
    .axis_vid_clk_i    (pix_clk),          // >= Actual Byte Clock Frequency (MHz)
    .axis_vid_rst_n_i  (pix_rst_n),
    .axis_vid_tready_i (sink_ready),
    .axis_vid_tvalid_o (sink_valid),
    .axis_vid_tdata_o  (sink_data),        // [(TDATA_WIDTH-1):0], 10-bit RAW packed by
PPC=1
    .axis_vid_tuser_o  (sink_tuser),       // [1:0], [0] = sof
    .axis_vid_tlast_o  (sink_tlast)        // end of line

);
```

In the new IP, the user design only needs to provide axis_vid_clk_i and consume the UVSI handshake. Byte-clock domain work (byte unpack, packet parse, line buffering) is inside the IP.

5.3. Migration Steps for This Example

For this migration example, follow these steps:

1. Replace the two old MIPI IP (csi2_dsi_dphy_rx, byte2pixel) instances with a single mipi_csi_dsi instance configured in Table 5.3.
2. Remove the byte-domain harness between the two legacy cores (payload_*, dt_*, wc_*, vc_*, sp_en_*, lp_*).
3. Replace the native pixel outputs (vsync_o, hsync_o, de_o, pd_o) of the Byte-to-Pixel Converter with a UVSI consumer on axis_vid_tvalid_o, axis_vid_tdata_o, axis_vid_tuser_o, axis_vid_tlast_o and drive axis_vid_tready_i.
4. Derive a valid axis_vid_clk_i frequency using the rule axis_vid_clk_i $\geq$ *Actual Byte Clock Frequency (MHz)*. For 2 lanes, RAW10, PPC = 1, 800 Mbps, this rule and the throughput formula yield a minimum of ≈ 160 MHz.
5. Verify that *Reference Clock Frequency (MHz)* is within 60–200 MHz for Soft D-PHY.

Appendix A. Migration Checklist

This appendix condenses the guidance in Section 1 through 5 into a sequential, actionable checklist for migrating a design from the four legacy MIPI IPs (CSI-2/DSI D-PHY Transmitter, CSI-2/DSI D-PHY Receiver, Pixel-to-Byte Converter, and Byte-to-Pixel Converter) to the unified MIPI CSI/DSI IP (IP version 4.1.0).

How to use this checklist:

- Complete the phases in order. Phase 1 is a gating assessment; if any item marked (Blocking) cannot be satisfied, the design cannot be migrated to the current IP release.
- Each item references the section of this guide that provides the details.
- Apply the checklist once per IP instance, separately for each transmit (TX) path and each receive (RX) path.

A.1. Phase 1 — Pre-Migration Eligibility Assessment

Confirm the existing design is supported before any RTL changes are made (refer to the Migration Scope section).

☐ **(Blocking)** The target device family is CrossLink-NX, Certus-NX, CertusPro-NX, or MachXO5-NX. Designs targeting Lattice Avant or Certus-N2 cannot migrate at this time (refer to the Supported Device Families section).

☐ **(Blocking)** For a transmit path, the protocol is CSI-2. DSI transmit is not supported by the new IP (refer to the Supported Configurations Comparison section).

☐ The receive path protocol is CSI-2 or DSI-2; both are supported (refer to the Supported Configurations Comparison section).

☐ **(Blocking)** Every active color format is one of RGB888, RGB565, RAW10, or RAW12. Designs using RAW8, RAW14, RAW16, RGB666, RGB444, RGB555, or any YUV420 or YUV422 format cannot migrate (refer to the Supported Color Formats section).

☐ The PHY mode (Soft D-PHY or Hard D-PHY) and the per-lane line rate are within the supported ranges (refer to the Supported Configurations Comparison section).

☐ The lane count is 1, 2, or 4. Three-lane configurations are not supported by the new IP (refer to the IP GUI Configuration Mapping section).

☐ **(RX, Blocking)** The smallest long packet produced by the source meets or exceeds the enforced minimum Word Count for the applicable PHY mode, protocol, and controller mode (refer to the Word Count Alignment Requirement section). The legacy IPs do not impose this floor.

☐ The change of register access from LMMI or APB to AXI4-Lite is understood and planned for in the host and driver software (refer to the Supported Configurations Comparison section).

☐ RX ECC status and CRC status are available only through register access over AXI4-Lite, not through native status signals (refer to the Supported Configurations Comparison section).

A.2. Phase 2 — IP Instantiation and GUI Configuration

Replace each legacy two-IP pairing with a single core (refer to the IP GUI Configuration Mapping section).

☐ Replace the legacy IP pair with one MIPI CSI/DSI IP instance (TX: D-PHY TX and Pixel-to-Byte; RX: D-PHY RX and Byte-to-Pixel).

☐ Set the **PHY Direction** parameter to TX or RX.

☐ Map the legacy D-PHY parameters: Protocol Mode, Number of D-PHY Lanes, Line Rate Per Lane, PHY Clock Mode, and protocol timing parameters (refer to the IP GUI Configuration Mapping section).

☐ Select the color format using the Enable <format> Color Format check boxes rather than a drop-down menu and confirm the **PHY Data Width** (8 for Soft D-PHY, 16 for Hard D-PHY) (refer to the IP GUI Configuration Mapping section).

☐ Confirm that the packet formatter (TX) and packet parser (RX) are always integrated; there is no separate enable option.

☐ Confirm that the Word Count is no longer entered manually on the transmit path. Word Count is derived automatically from the resolution and color format (refer to the Word Count Alignment Requirement section).

☐ Enable the AXI4-Lite interface if register access is required.

☐ Remove the legacy manual FIFO controls which are replaced by single buffer-depth parameters (refer to Phase 6 — New Requirements and Buffer Sizing).

A.3. Phase 3 — Clocking Migration

Update the clock topology (refer to the Clocking Migration section).

☐ **(TX)** Provide ref_clk_i within 60–200 MHz for Soft D-PHY, and axis_vid_clk_i for the UVSI pixel domain.

☐ **(TX)** Remove the manually routed byte clock between the legacy IPs. byte_clk_o is now an IP output and requires no user routing for pixel-only applications.

☐ **(RX)** Provide ref_clk_i (60–200 MHz for Soft D-PHY) and a single free-running clock fr_clk_i to consolidate the legacy clk_byte_fr_i and clk_lp_ctrl_i.

☐ **(RX)** Ensure fr_clk_i is greater than or equal to the Actual Byte Clock Frequency, with a minimum of 60 MHz.

☐ **(RX)** Ensure axis_vid_clk_i satisfies both the byte-clock floor and the throughput formula in the Clocking Migration section.

☐ Remove the separate legacy pixel-clock domains. The domains are replaced by the single axis_vid_clk_i.

A.4. Phase 4 — Reset Migration

Consolidate the reset scheme (refer to the Reset Migration section).

☐ **(TX)** Replace the legacy resets with ref_rst_n_i and axis_vid_rst_n_i, and assert both together for a full reset.

☐ **(RX)** Replace the legacy multi-reset arrangement with ref_rst_n_i, fr_rst_n_i, and axis_vid_rst_n_i, and assert all three together for a full reset.

☐ Confirm that no specific reset sequencing is required, but that all clock domains are fully flushed before reset is de-asserted.

A.5. Phase 5 — User Data Interface Migration

Select one data interface and rewire the user logic (refer to User Interface Migration to UVSI and User Interface Migration to MBSI sections). Use the UVSI to work with pixel data, or the MBSI to work with raw MIPI byte data.

If migrating to the UVSI (pixel domain):

☐ Map the legacy converter pixel signals to the UVSI: vsync or fv to axis_vid_tuser[0] (start-of-frame), de to axis_vid_tvalid, lv to axis_vid_tlast, and the pixel buses to axis_vid_tdata (refer to the User Interface Migration to UVSI section).

☐ Implement the new TREADY handshake (axis_vid_tready) for backpressure. Horizontal sync is now implicit.

☐ Repack the pixel data into the single axis_vid_tdata bus per the TDATA width rule (ceil(BPP / 8) × 8 × PPC), including 16-bit zero-padded slots for RAW10 and RAW12 and 16-bit packing for RGB565 (refer to the User Interface Migration to UVSI section).

☐ Remove all eliminated byte-domain and handshake signals that are now internal to the IP (refer to the User Interface Migration to UVSI section).

If migrating to the MBSI (byte domain):

☐ Map the legacy byte signals to the MBSI: byte data, data type, and word count to axis_byte_tdata (the full MIPI packet, including the packet header and CRC footer, but excluding SoTp and HS-Trail); the enable signals to axis_byte_tvalid; the ready signal to axis_byte_tready; and the virtual channel to axis_byte_tuser (refer to the User Interface Migration to MBSI section).

☐ Confirm the MBSI clock domain: axis_vid_clk_i for TX and fr_clk_i for RX.

☐ Implement user-side detection of short and long packets and of the footer byte position on the 32-bit or 64-bit TDATA bus.

☐ Define the ECC and CRC handling (refer to the User Interface Migration to MBSI section): when insertion or checking is disabled, you compute and embed (TX) or validate (RX) these fields; when enabled, populate placeholder fields (TX) or read the error status from the register interface (RX).

A.6. Phase 6 — New Requirements and Buffer Sizing

Account for behaviors that do not exist, or that work differently, in the legacy IPs (the New Requirements section).

☐ **(TX, UVSI)** Size the Line Buffer Depth for the resolution and horizontal blanking which replaces the legacy Pixel-to-Byte FIFO depth, width, and threshold parameters.

☐ **(RX, UVSI)** Size the Pixel Buffer Depth and ensure downstream logic does not hold axis_vid_tready_i low long enough to overflow it.

☐ **(MBSI)** Size the Byte Buffer Depth to hold at least one full long packet and set axis_vid_clk_i (TX) or fr_clk_i (RX) to at least the Actual Byte Clock Frequency.

☐ **(TX)** Configure the built-in Video FSM (Number of Active Lines Per Frame, the optional Line Start and Line End packets, and the optional Line Number Insertion). Remove the legacy manual driving of sp_en_i, lp_en_i, vc_i, dt_i, and wc_i.

☐ **(TX)** Confirm that the smallest transmitted Word Count meets the enforced minimum of 32 bytes for the supported transmit configurations; most video resolutions far exceed this (the New Requirements section).

☐ **(TX, non-32-byte Word Count)** Ensure a minimum gap of four axis_vid_clk_i cycles between successive packet streams (refer to the Word Count Alignment Requirement section).

☐ Note that the MBSI TDATA width is fixed at 32 bits, or 64 bits only for Hard D-PHY with four lanes, to support dynamic lane reconfiguration (the New Requirements section).

A.7. Phase 7 — Register Access and Error Handling

Update the host and driver software (refer to IP GUI Configuration Mapping and User Interface Migration to MBSI sections).

☐ Port the register transactions from LMMI or APB to the AXI4-Lite interface, and supply s_axi_aclk_i.

☐ Update all register addresses. The legacy addresses do not carry over. Refer to the *Register Description* section in the MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321).

☐ **(RX)** Where ECC or CRC checking is enabled, read the error status from the CONTROLLER_STATUS_REG (0x020) over AXI4-Lite instead of from legacy native status signals.

A.8. Phase 8 — Timing and Physical Constraints

Refresh the constraint files (refer to the Timing and Physical Constraints section).

☐ Update any user-defined timing constraints (SDC) that referenced legacy IP clock or signal names to reference to the new IP names.

☐ Verify the MIPI pin assignments and I/O type definitions (PDC) which remain compatible between the legacy and new IPs.

A.9. Phase 9 — Verification and Sign-Off

☐ Confirm that the design builds cleanly with the single MIPI CSI/DSI IP instance and the legacy byte-domain harness is fully removed.

☐ Verify frame integrity from end to end (start-of-frame, end-of-line, resolution, and color format).

☐ Confirm that there is no buffer overflow (RX) or unexpected backpressure (TX) under worst-case blanking.

☐ Confirm that register reads and writes and the error status behave as expected over AXI4-Lite.

☐ Verify dynamic reconfiguration (for example, lane count) if this feature is used by the design.

☐ Record the final IP configuration and retain the configuration with the design documentation.

References

- MIPI CSI/DSI IP User Guide (FPGA-IPUG-02321)
- CSI-2/DSI D-PHY Rx IP User Guide (FPGA-IPUG-02081)
- CSI-2/DSI D-PHY Tx IP User Guide (FPGA-IPUG-02080)
- Byte-to-Pixel Converter IP User Guide (FPGA-IPUG-02079)
- Pixel-to-Byte Converter IP User Guide (FPGA-IPUG-02094)
- Lattice Radiant Timing Constraints Methodology (FPGA-AN-02059)
- CrossLink-NX web page
- Certus-NX web page
- Certus-N2 web page
- CertusPro-NX web page
- MachXO5-NX web page
- Avant-E web page
- Avant-G web page
- Avant-X web page
- MIPI CSI/DSI IP Core web page
- CSI-2/DSI D-PHY Receiver IP Core web page
- CSI-2/DSI D-PHY Transmitter IP Core web page
- Byte to Pixel Converter IP Core web page
- Pixel to Byte Converter IP Core web page
- Lattice Radiant Software web page
- Lattice Insights for Lattice Semiconductor training courses and learning plans

Technical Support Assistance

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

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

Revision History

Revision 1.0, July 2026

Section	Change Summary
All	Initial release.

www.latticesemi.com