UG0850
User Guide
PolarFire FPGA Video Solution

Microsemi
a Microchip company

a Microchip company

Microsemi Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
Email: sales.support@microsemi.com
www.microsemi.com

About Microsemi

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Learn more at www.microsemi.com.

50200850. 1.0 2/19

Contents

Figures

Tables

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 1.0

The first publication of this document.

2 Overview

Libero® SoC PolarFire Design Suite includes all the IP cores for building prototype solutions quickly. Thus, PolarFire FPGA-based video solutions save design time and cost without compromising performance.

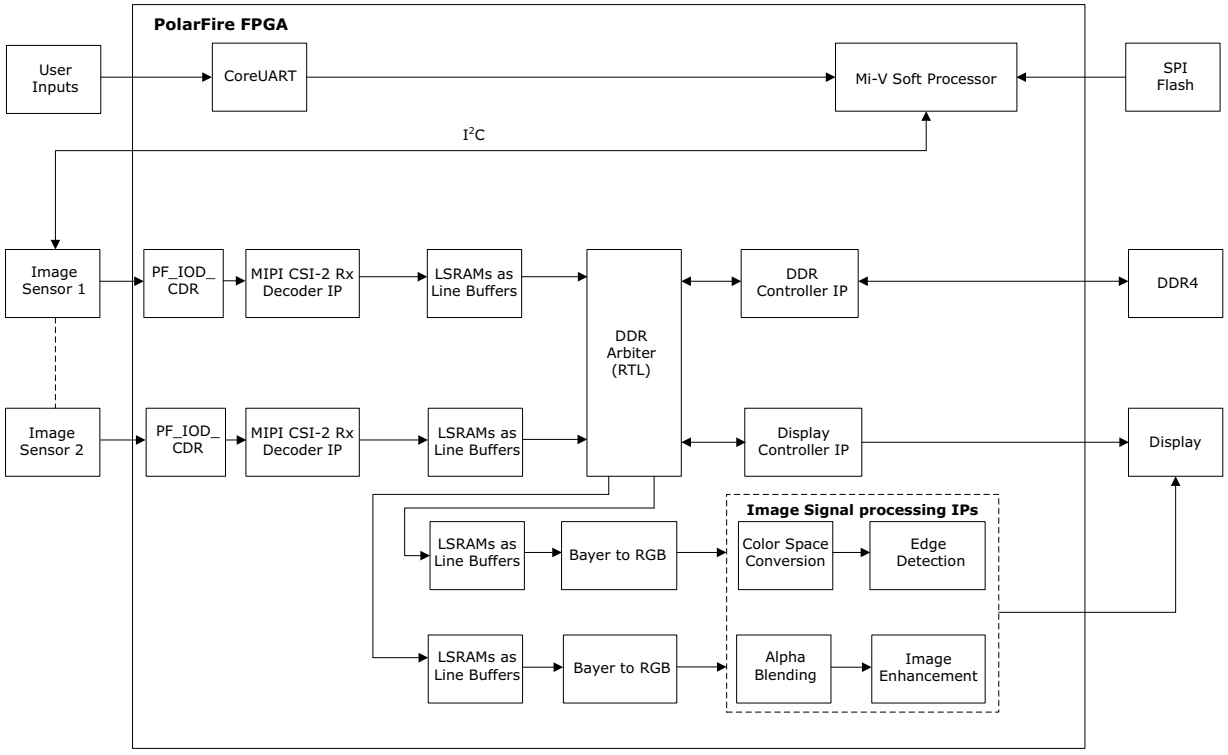
The prototype solutions can be validated on the PolarFire Video Kit, which features PolarFire MPF300T FPGA, SPI Flash, DDR4 memories, SDI, HDMI, and FTDI hardware. The on-board MIPI-CSI-2 interface can be used to connect to a dual camera daughter module. The on-board FMC connector can be used for connecting to daughter cards. For more information about the video kit, see *UG0856: PolarFire FPGA Video Kit User Guide*.

This document describes the following protocols for video solution:

- MIPI CSI-2, page 4
- HDMI, page 14
- SDI, page 15

The following figure shows the PolarFire video solution architecture.

Figure 1 • **Imaging and Video Architecture**



Note: In this solution, the camera data is received using the Microsemi's MIPI CSI-2 RX IP. The data is processed and transmitted to a display device using the HDMI interface. Alternately, Microsemi's SDI or HDMI RX and TX IPs can also be used to receive and transmit the data.

The following points summarize the video solution:

- The Mi-V soft processor is booted by a user application stored in SPI Flash. The user application initializes the camera sensors and processes the user requests from host PC via the UART interface.
- The PF_IOD_GENERIC_RX block receives the high-speed MIPI CSI-2 serial data from the camera module. The IOD gear box component converts the serial data to parallel 8-bit format.
- The MIPI CSI2 RxDecoder PF IP decodes the MIPI byte packets and extracts the pixel data along with control signals.
- The raw pixel data is stored in LSRAMs as line buffers.
- The DDR Arbiter module writes these line buffers and stores video frames in DDR4.
- The DDR arbiter reads the line data and passes it to the other two LSRAMs from where the ISP modules like Bayer Conversion IP read and process the data.

Note: Other signal processing requests like edge detection or alpha blending are also processed by the respective ISP module on demand. These requests come from host PC to Mi-V soft processor via the CoreUART block.

- After processing, the ISP module passes the resultant data to the display device.

3 MIPI CSI-2

The MIPI CSI-2 (Camera Serial Interface) is the standard specification to transmit and receive the camera sensor raw pixel data in High speed serial format. The MIPI CSI-2 uses MIPI D-PHY as its PHY layer and sends data over it. MIPI DPHY supports 1 Lane to 4 Lanes. MIPI CSI-2 uses High- Speed (HS) and Low-Power (LP) modes, which are differentiated by the voltage levels as shown in Figure 3, page 4. In High-Speed mode Actual High Speed video packets are transferred. MIPI CSI-2 uses two kinds of High-Speed packets, Long packet and short packet. Long packets are used to transmit burst video data and short packets are used to send the control signals of the video (e.g frame start (FS), frame end (FE), Line Start and Line end). Figure 2, page 4 shows the different High speed packets and their occurrence.

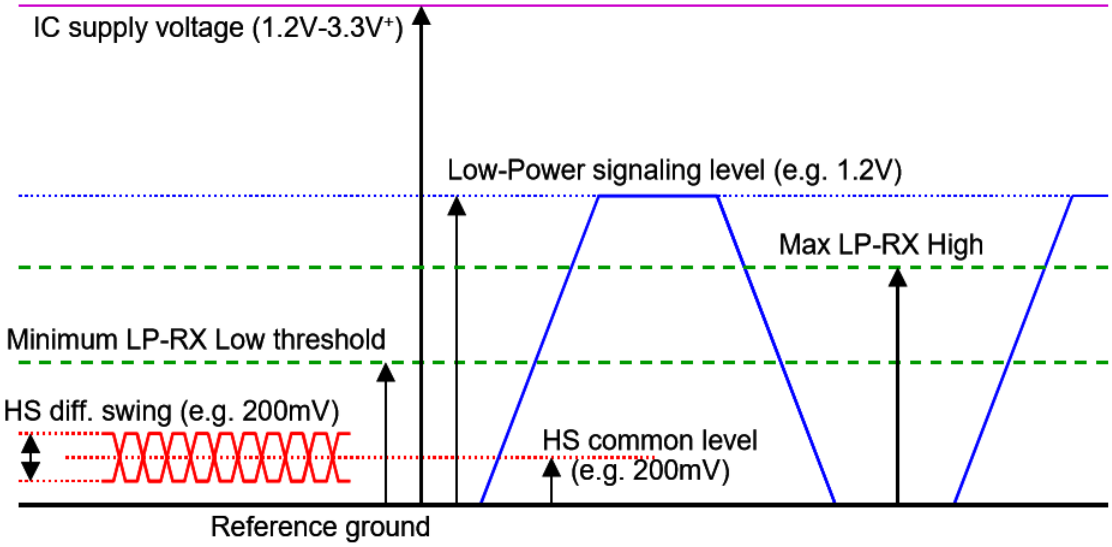
Figure 2 • **High-Speed Video Packet Structure**

Note: The following points describe the terms used in Figure 2, page 4.

- FS: Frame Start Packet (short packet)
- Image: Pixel data of the image embedded in Long Packet
- FE: Frame End Packet (short Packet)

MIPI DPHY uses differential SLVS200 Signaling to transmit high-speed data and LVCMOS12 to transmit Low Power data. MIPI D-PHY switches between Differential High Speed mode and single ended Low Power mode on the fly on the same pair of I/Os. During Low Power mode it operates in very less speed so that it consumes low power.

Figure 3 • **Low-Power Signal Processing**



3.1 MIPI Packet Format

This section introduces the MIPI packet formats.

3.1.1 Short Packet

The following figure shows the short packet structure.

Figure 4 • **Short Packet Structure**

SoT or Sync Code (B8)	Data ID	Frame Number[7:0]	Frame Number[15:8]	ECC

3.1.2 Long Packet

The following figure shows the short packet structure.

Figure 5 • **Long Packet Structure**

SoT B8	Data ID	Word Count[7:0]	Word Count[15:8]	ECC	Payload	CRC[7:0]	CRC[15:8]

3.2 MIPI CSI-2 Based Solution

Implementation of MIPI CSI-2 based video solutions in FPGAs requires IP and design flows that reduce development time, and utilize minimal device resources to meet performance, power, and cost goals.

Microsemi offers MIPI CSI2 RxDecoder PF and MIPI CSI2 Transmitter PF IP cores, image signal processing (ISP) IP solutions, PF_IOD_GENERIC_RX/TX hard IPs, high-speed transceiver hard IP, and other soft IP cores for the implementation of imaging and video solutions for PolarFire FPGAs. These IP cores are available in Libero® SoC PolarFire Design Suite for building prototype solutions quickly.

The following sections describe the building blocks of this solution and the implementation of receive and transmit interfaces.

3.2.1 Building Blocks of MIPI CSI-2 Video Solution

The following Microsemi IP cores form a complete MIPI CSI-2 video solution:

- Mi-V soft processor to configure the MIPI CSI-2 camera
- MIPI CSI-2 Rx and Tx IPs to decode and encode MIPI packets
- PF_IOD_GENERIC_RX/TX to receive the high-speed serial MPI CSI-2 camera data
- PLLs to generate fabric, receive and transmit clocks
- DDR Controller to store and retrieve the data
- Image Signal Processing Cores—Image Edge Detection, Bayer Conversion, and Image Enhancement IPs
- Transceiver blocks (Rx and Tx) to receive and send the data

For more information about the video solution, see https://www.microsemi.com/product-directory/technology/3861-imaging#software-and-ip.

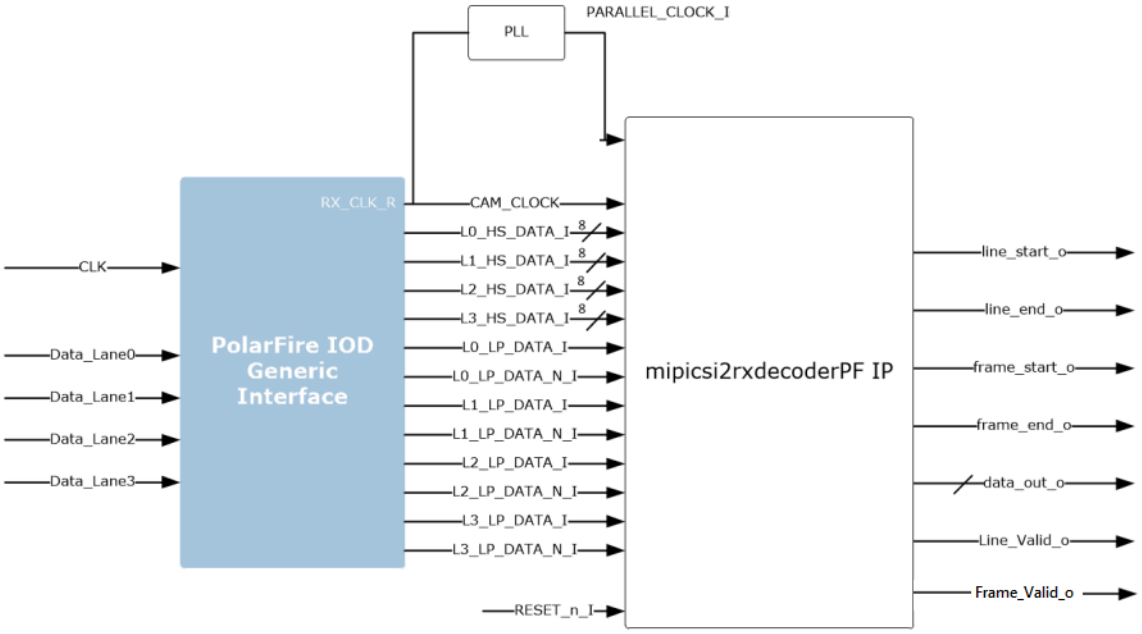
3.2.2 Implementing MIPI CSI-2 Receive Interface

As shown in Figure 1, page 2, the PF_IOD_GENERIC_RX hard IP interfaces with the camera sensor module and passes the data to the MIPI CSI-2 Receiver Decoder IP. The PF_IOD_GENERIC_RX and MIPI CSI2 RxDecoder PF IP form the receive path of the solution.

The PF_IOD_GENERIC_RX is set to the MIPI mode to receive high-speed and low-power data. The high-speed MIPI CSI-2 serial data is passed through IOD gear box to convert the data into parallel 8-bit format. The MIPI CSI2 RxDecoder PF decodes the MIPI byte packets and extracts the pixel data along with the control signals.

The following figure shows the data and clock lines of the MIPI CSI-2 receiver interface solution.

Figure 6 • **MIPI CSI-2 Receiver Interface**



MIPI CSI2 RxDecoder PF IP decodes both high-speed short and long packets[1]. It supports RAW8, RAW10, and RAW12 data types for 1-Lane, 2-Lane, and 4-Lane interfaces.

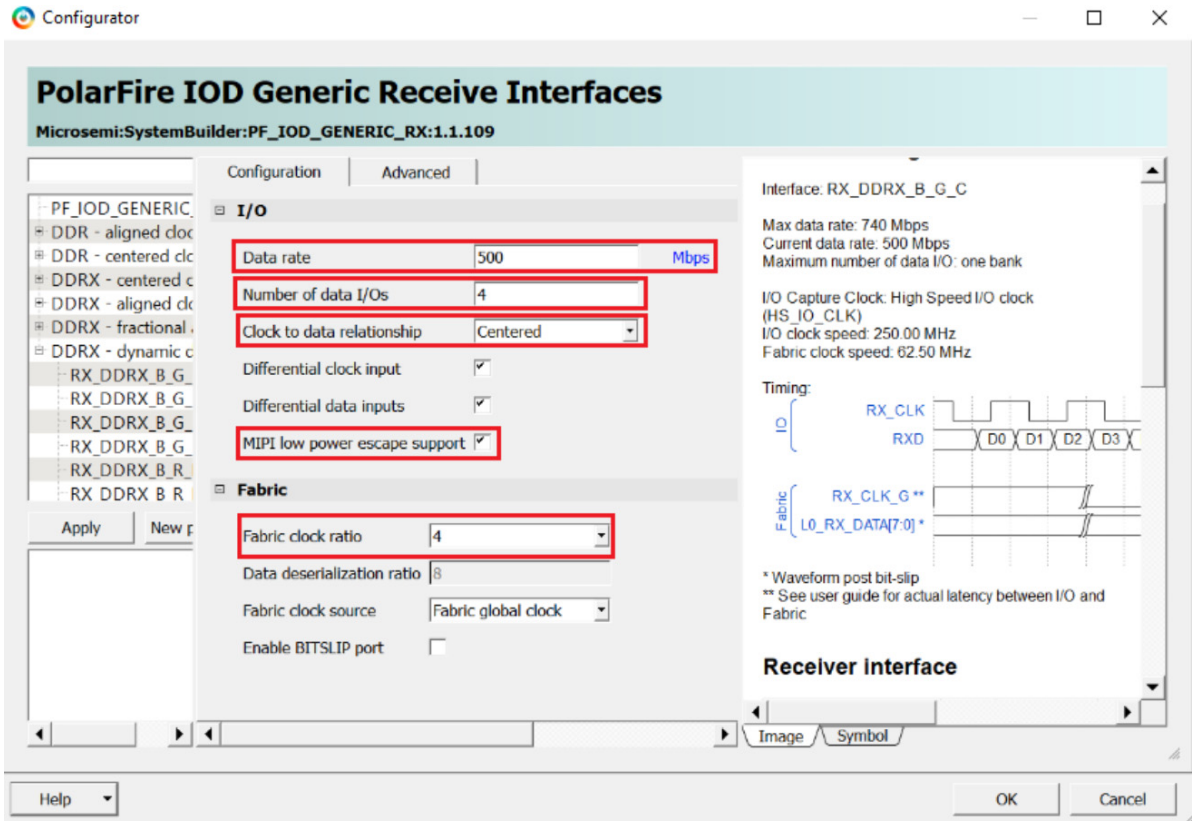
3.2.3 Configuring the MIPI CSI-2 Receive Interface IP Cores

This section describes the configuration settings used for the PF_IOD_GENERIC_RX and MIPI CSI2 RxDecoder PF IP cores in Libero.

The following figure shows the data rate, number of data I/Os, and other settings used for PF_IOD_GENERIC_RX.

1.For more information about the packet format, see MIPI Packet Format, page 19.

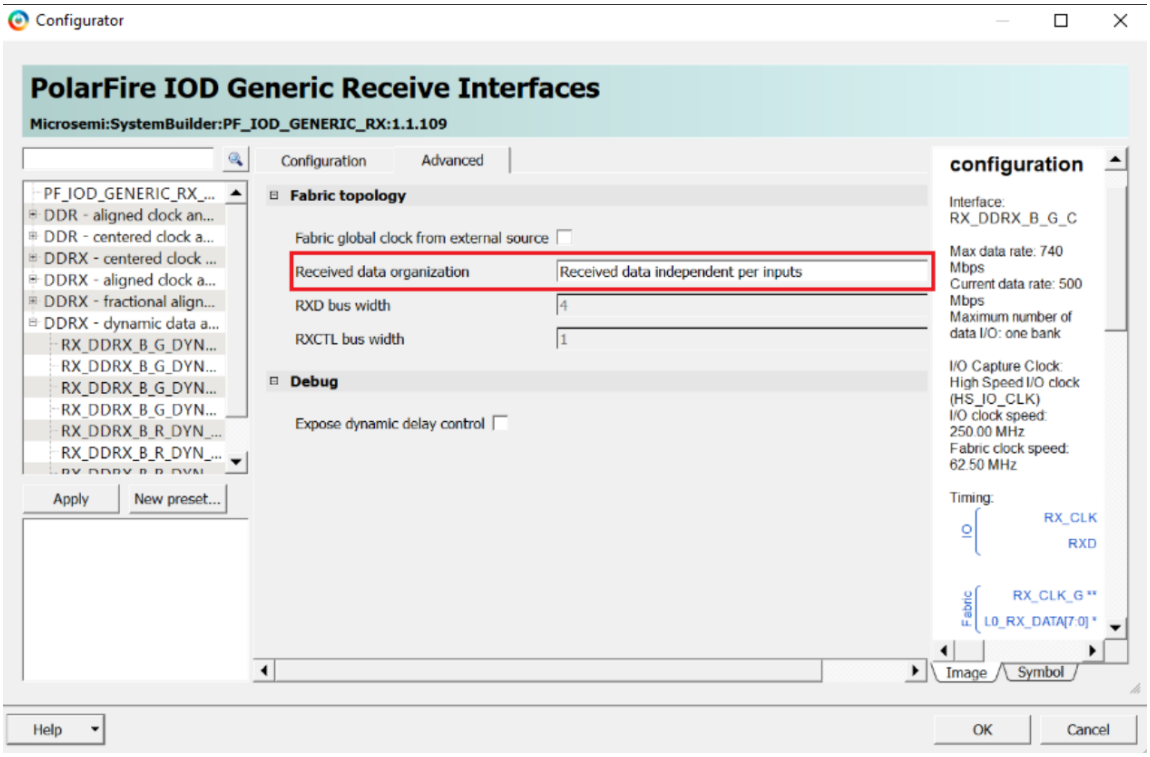
Figure 7 • **PF_IOD_GENERIC_RX in MIPI Mode**



The following points must be considered while configuring PF_IOD_GENERIC_RX:

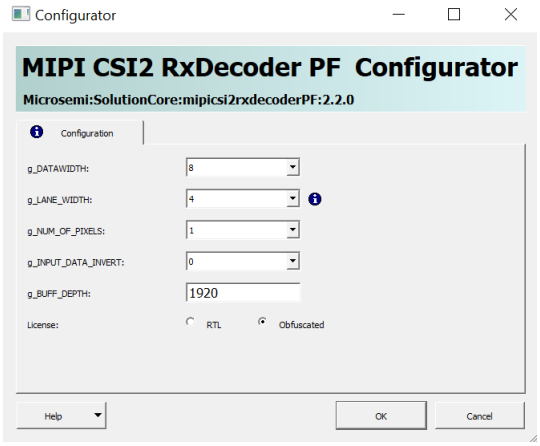
- Set the IOD data rate according to the per lane data rate of the MIPI camera.
- Set the number of Data I/Os according to the number of lanes used by the MIPI camera.
- Set the Clock to data relationship option to **Centered** because MIPI is a centered-aligned interface.
- Enable the MIPI mode by selecting the **MIPI Low power escape support** check box. By enabling the MIPI Mode all the Low Power Ports are added to IOD.
- Set the Fabric clock ratio to 4.
- In the Advanced tab, set **Received data organization** to **Received data independent per inputs** as shown in the following figure.

Figure 8 • **PF_IOD_GENERIC_RX Advanced Configuration**



The MIPI CSI2 RxDecoder PF IP is configured as shown in the following figure.

Figure 9 • **MIPI CSI-2 RxDecoder Configuration**



The following points must be considered while configuring MIPI CSI2 RxDecoder PF:

- Set **g_DATAWIDTH** to **8** for RAW8 data type. For RAW10/12 set **g_DATAWIDTH** to **10/12**.
- Set **g_LANE_WIDTH** according to the number of lanes used by the MIPI camera.
- Set **g_BUFF_DEPTH** according to the active horizontal resolution configured for the MIPI sensor.
- Set **g_NUM_OF_PIXELS** according to the requirement. If this parameter is set to 1, only one pixel per clock is given as output. If this parameter is set to 4, four pixels per clock are given as output.

A PLL is required to generate the parallel clock (pixel clock). The RX_CLK clock of PF_IOD_GENERIC_RX serves as the input clock to the PLL. The PLL must be configured to produce the parallel clock based on the MIPI_bit_clk and the number of lanes used. The following equations are used to calculate the parallel clock:

$$CAM\_CLOCK\_I = (MIPI\_bit\_clk)/4$$

EQ-1

$$PARALLEL\_CLOCK = (CAM\_CLOCK\_I \times Num\_of\_Lanes \times 8)/(g\_DATAWIDTH \times g\_NUM\_OF\_PIXELS)$$

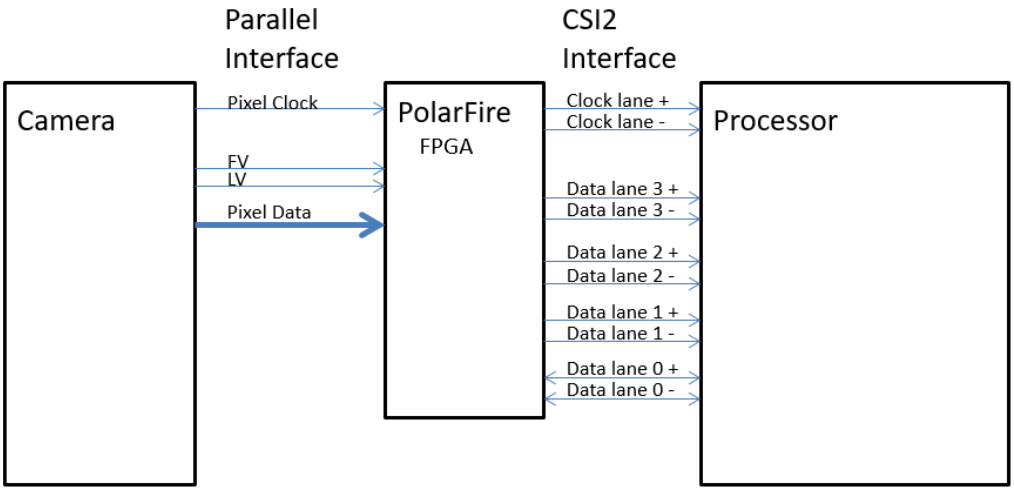
EQ-2

For more information about the MIPI CSI2 RxDecoder IP, see *UG0806: MIPI CSI-2 Receiver User Guide for PolarFire*.

3.2.4 Implementing MIPI CSI-2 Transmit Interface

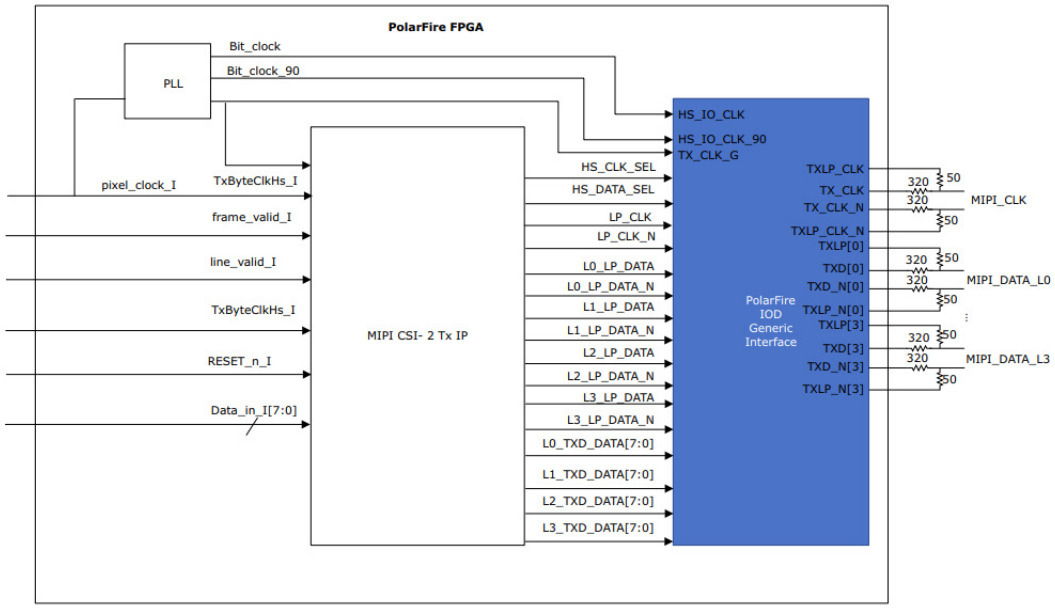
The MIPI CSI2 Transmitter PF IP converts the parallel video data into MIPI CSI-2 Byte packets. The typical system diagram for the MIPI CSI-2 transmitter interface is shown in the following diagram. The MIPI CSI2 Transmitter bridge is used to deliver data to a MIPI CSI-2 compatible receiver such as an ISP from standard Parallel video interface.

Figure 10 • **MIPI CSI-2 Transmitter Interface Use Case**



The PF_IOD_GENERIC_TX IP (with I/O gearing blocks) and the MIPI CSI2 Transmitter IP form the MIPI CSI-2 Transmitter interface in PolarFire devices, as shown in the following figure.

Figure 11 • **MIPI CSI-2 Transmit Interface**



MIPI CSI2 Transmitter PF IP supports both high-speed short packets and long packets. It supports RAW8 and RAW10 data types for 1-Lane, 2-Lane, and 4-Lane interfaces.

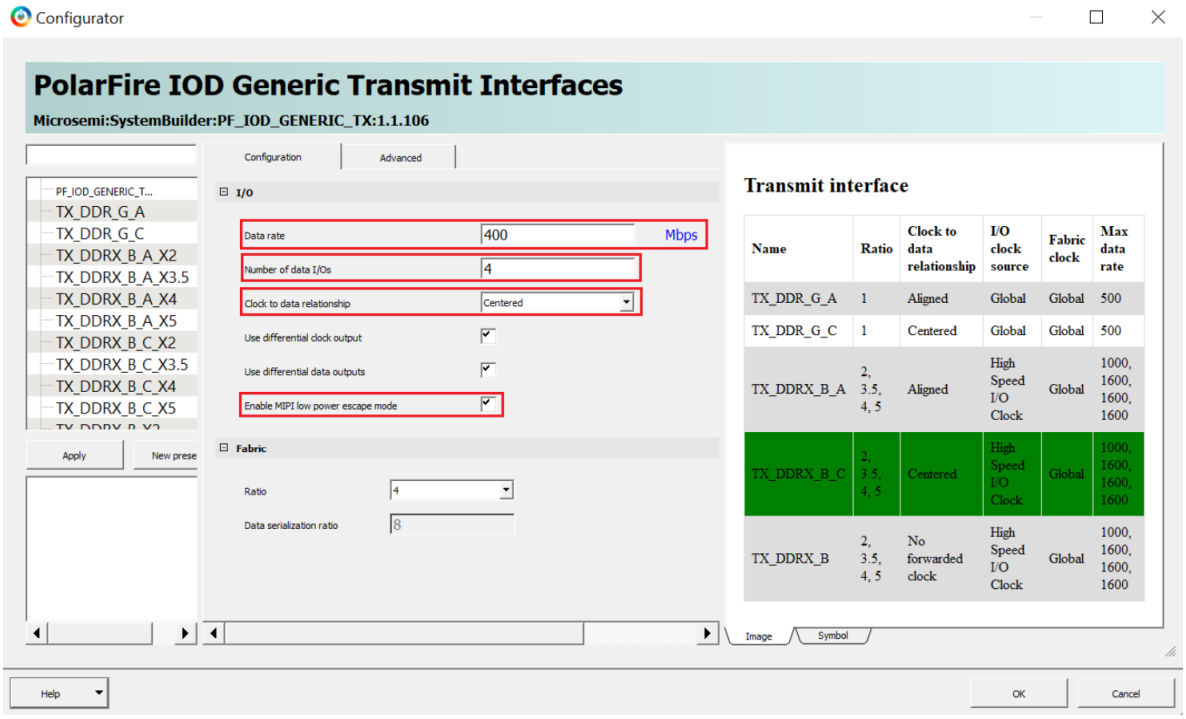
An external resistor network shown in Figure 11, page 10 is required to accommodate the low-power and high-speed mode transitioning on the same signal pairs and also to bring down the voltage swing to 200 mV during high-speed clock and data transfers.

3.2.5 Configuring the MIPI CSI-2 Transmit Interface IP Cores

This section describes the configuration settings used for the PF_IOD_GENERIC_TX and MIPI CSI2 Transmitter PF IP cores in Libero.

The following figure shows the data rate, number of data I/Os, and other settings used for PF_IOD_GENERIC_TX.

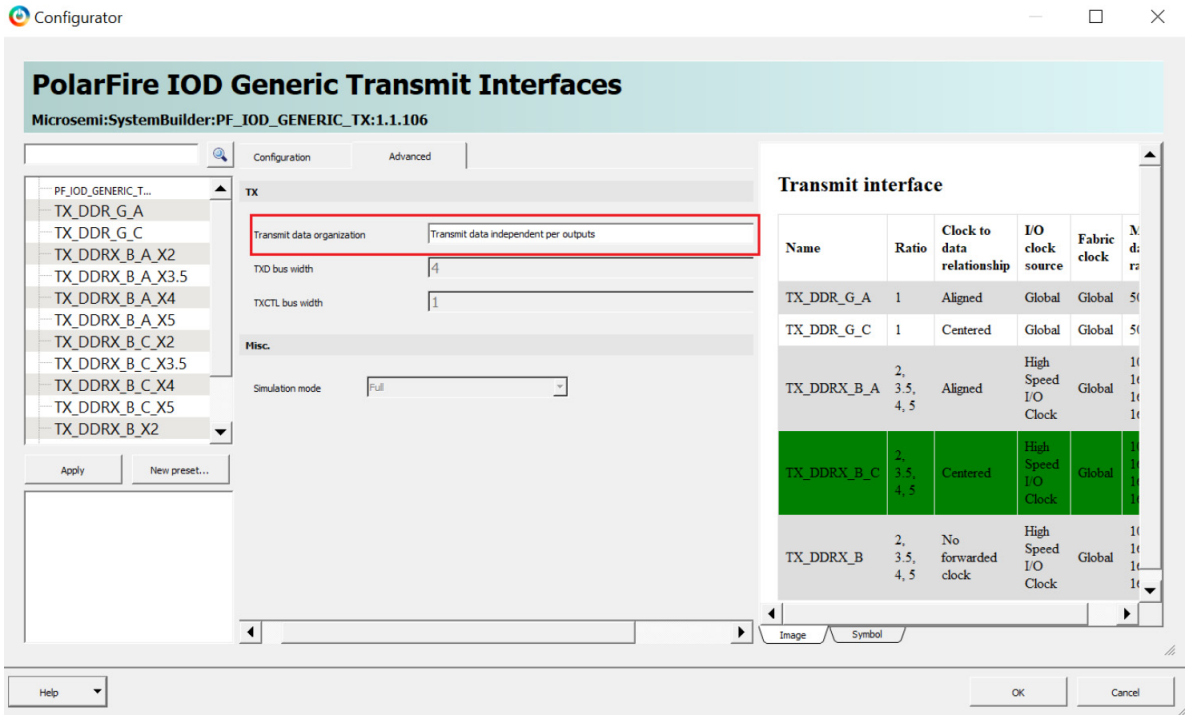
Figure 12 • **PF_IOD_GENERIC_TX in MIPI Mode**



The following points must be considered while configuring PF_IOD_GENERIC_TX:

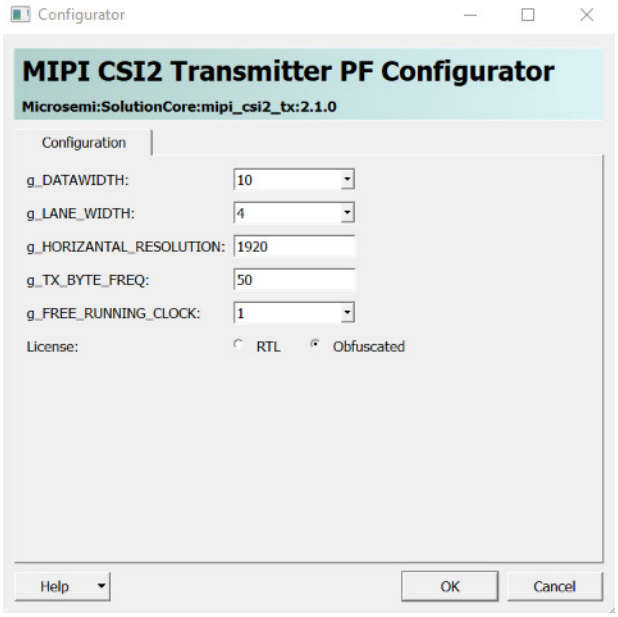
- Set **Data rate** according to the required MIPI data rate per lane.
- Set **Number of data I/Os** according to the number of MIPI Lanes.
- Set **Clock to data relationship** to **Centered** because MIPI DPHY is a centered-aligned interface.
- Enable the MIPI low power escape mode check box.
- In the Advanced tab, set **Transmit data organization** to **Transmit data independent per outputs** as shown in the following figure.

Figure 13 • **PF_IOD_GENERIC_TX Advanced Configuration**



The MIPI CSI2 Transmitter PF IP is configured as shown in the following figure.

Figure 14 • **MIPI CSI-2 Transmitter Configuration**



The following points must be considered while configuring MIPI CSI2 Transmitter PF:

- Set **g_DATAWIDTH** to **10** for RAW10 data type. For RAW8 set **g_DATAWIDTH** to **8**.
- Set **g_LANE_WIDTH** according to the number of lanes used by the MIPI camera.
- Set **g_HORIZONTAL_RESOLUTION** as the active resolution of the transmitted video.
- g_TX_BYTE_FREQ is calculated using the following equation:

$$Pixel\_Clock\_I = TxByteClkHs\_I \times Number\ of\ Lanes \times 8\ Bits\ per\ Pixel$$

Pixel_clock_i is the input clock with which incoming pixels are sampled. A PLL is used to generate the Byte clock (TxByteClkHs_I) and bit clocks used by the MIPI DPHY block (PF_IOD_GENERIC_TX). TxByteClkHs_I must be configured such that the output MIPI CSI-2 packets sent on the interface are sampled. The following equations shows the relation between Pixel_clock_I and TxByteClkHs_I depending on the number of lanes configured.

$$Pixel\_Clock\_I = TxByteClkHs\_I \times Number\ of\ Lanes \times 8\ Bits\ per\ Pixel$$

EQ-3

$$MIPI\ Bit\ Clock = 4 \times TxByteClkHs\_I$$

EQ-4

There are two MIPI bit clocks as per EQ-4, and these clocks are phase shifted by 90°. For more information about the MIPI CSI-2 Transmitter PF IP, see*UG0826: MIPI CSI-2 Transmitter User Guide for PolarFire*.

4 HDMI

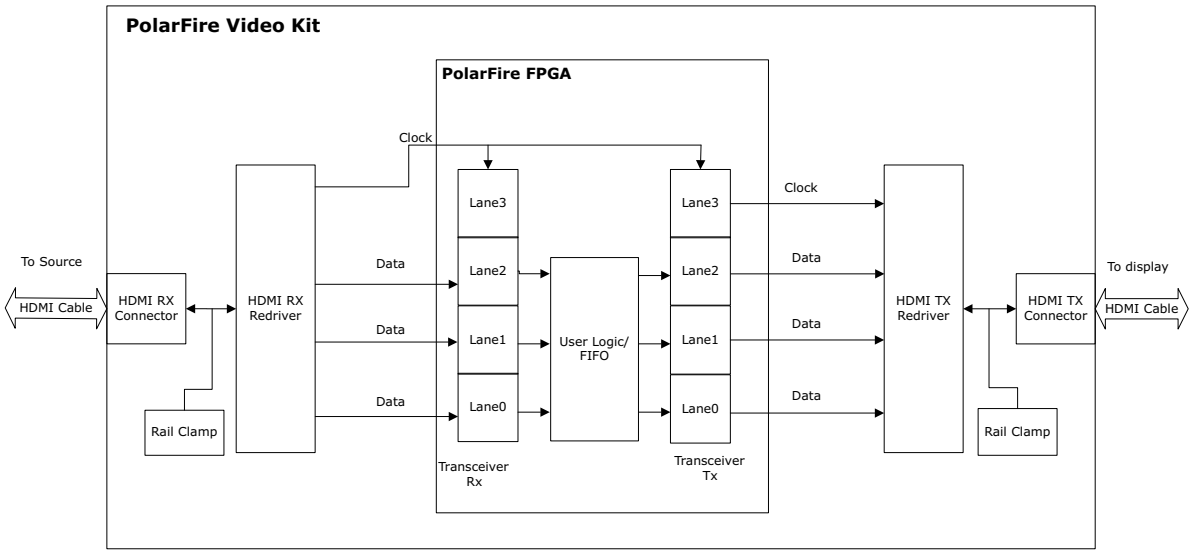
HDMI and Display port solutions can be enabled on PolarFire devices using high speed transceivers. For more about configuring the high-speed transceivers, see *UG0677: PolarFire FPGA Transceiver User Guide*.

The TMDS signals in HDMI are DC coupled, with different differential impedances for sources and sinks. Devices such as PI3HDX1204E and PI3HDX1204B can be used for DC coupling and to ensure that impedances are at the right levels. Refer to the video board schematic for details <link>.

For HDMI RX, the EDID ROM can be implemented as an I2C slave in the FPGA. Similarly, an I2C master implementation in the FPGA can be used to fetch details of a display connected to HDMI TX.

An example of how the transceiver can be configured for a HDMI 2.0 loopback is shown in the following block diagram. The transceiver should be configured in the PMA only mode.

Figure 15 • **HDMI Loopback Block Diagram**



5 SDI

Microsemi's CoreSDIRX and CoreSDITX IP cores support SDI-based video solutions. CoreSDIRX and CoreSDITX IP cores support HD-SDI and 3G-SDI modes.

The SDI based video solution uses the PolarFire transceiver block, which converts the high-speed serial data into parallel data and passes this data to the FPGA fabric.

These two IP cores have a 32-bit CFG_SDI register that must be configured for the required mode and color chroma sub-sampling.

This section tabulates the various segments of the CFG_SDI register. For more information, see CoreSDIRX and CoreSDITX handbooks from Libero Catalog.

Table 1 • **SDI Configuration Register**

Address	Register Name	Type	Width	Reset Value	Description
0x00	CFG_SDI	R/W	32	0x00000082	SDI Configuration Register

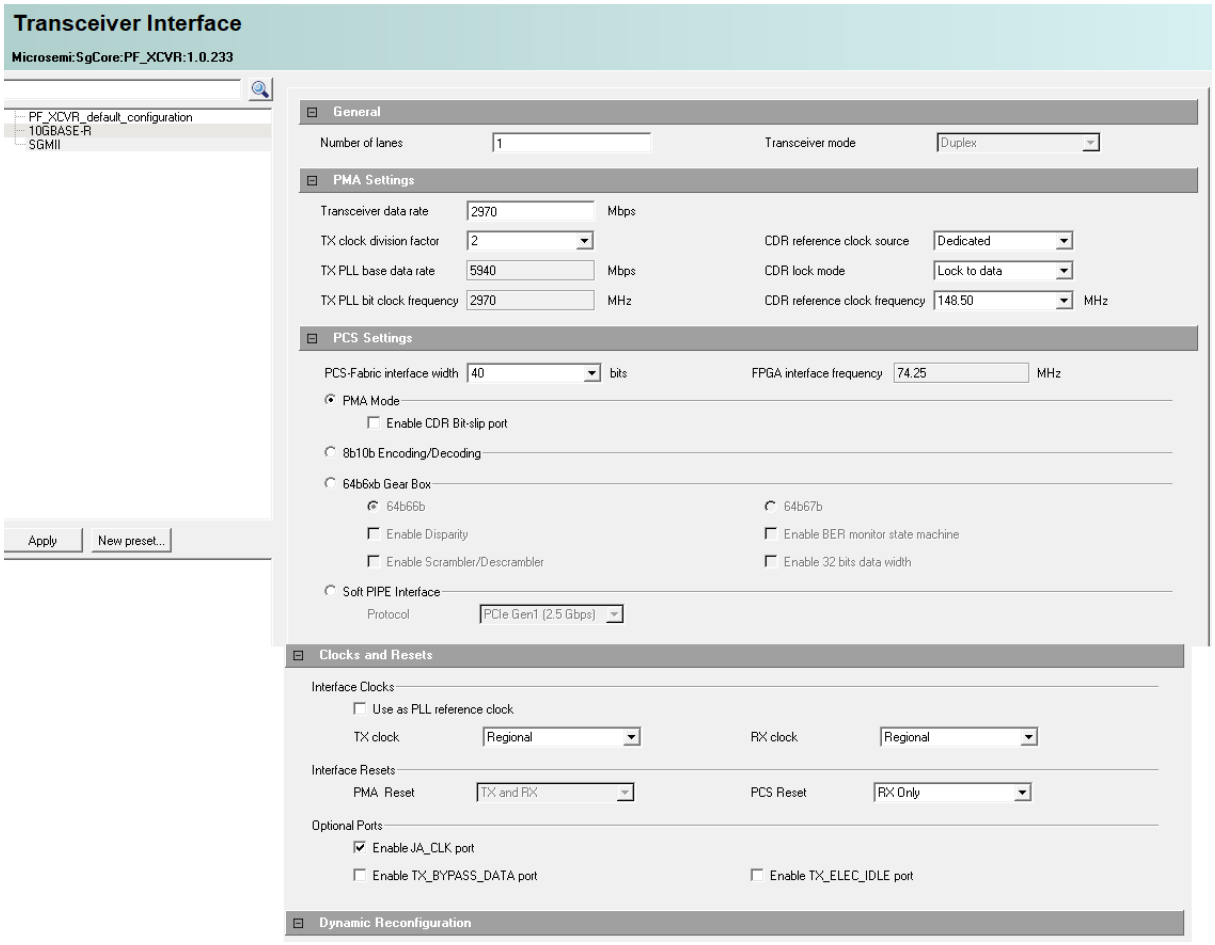
Table 2 • **SDI Configuration Register Bit Field Description**

Bit	Name	Type	Reset Value	Description
31:25	Reserved	-	-	-
24	VALID	R/W	1	Valid bit. Active high. Indicates that the configuration data is valid.
23:10	Reserved	-	-	-
9:6	MODE	R/W	0010	Type of SDI Mode: 0010: HD-SDI 0101: 3G-SDI - Level A Other values are reserved.
5:3	Reserved	-	-	-
2:0	COLOR	R/W	010	Type of SDI Color. • 000: RGB (color coding), 4:4:4 (chroma sub-sampling) • 001: YCbCr (color coding), 4:4:4 (chroma sub-sampling) • 010: 001: YCbCr (color coding), 4:2:2 (chroma sub-sampling) Other values are reserved.

5.1 XCVR Configuration for 3G-SDI Mode

The following figure shows the transceiver data rate, PMA and PCS, and clock and resets settings for the 3G-SDI mode.

Figure 16 • **XCVR Configuration For 3G-SDI**



5.2 XCVR Configuration For HD-SDI

Figure 17, page 17 shows the transceiver data rate, PMA and PCS, and clock and resets settings for the HD-SDI mode.

For more about configuring the high-speed transceivers, see *UG0677: PolarFire FPGA Transceiver User Guide*.

Figure 17 • **XCVR Configuration For HD-SDI**