UG0948
User Guide
PolarFire MIPI DSI Transmitter

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.

50200948. 1.0 7/21

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

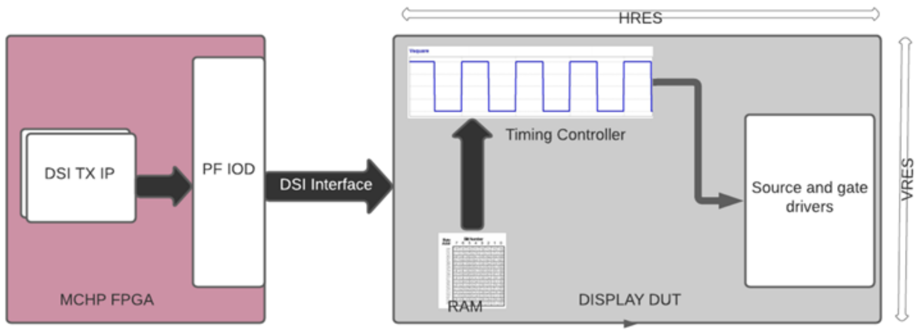
The first publication of this document.

2 MIPI DSI Transmitter

2.1 Introduction

Mobile Industry Processor Interface (MIPI) Display Serial Interface (DSI) is a standard specification defined by the MIPI Alliance display working group. The DSI specification defines an interface between a display device and a host processor. This user guide describes the MIPI DSI transmitter IP developed in Microchip FPGAs. There are two modes of video transmission to the display as per the DSI specification - video mode and command mode. The MIPI DSI transmitter IP supports only command mode.

Figure 1 • **A typical use case of Microchip DSI TX IP**



Microchip FPGAs have high-speed serial interface capabilities with their IOD blocks. The DSI TX IP interfaces with a video timing generator on the input side and an IOD block on the output side to drive a command mode display. The DSI IP transmitter encodes the pixel data compliant to the MIPI DSI standard. This IP Core supports 4 lanes RGB-888 data type and operates in two modes on the physical layer - high-speed mode and low-power mode. In high-speed mode, MIPI DSI supports the transmission of image data using short and long packets. Short packets are used to send control information, and long packets are used to send video content on the DSI lanes. For video data, one long packet is equivalent to one image data line.

2.2 Hardware Implementation

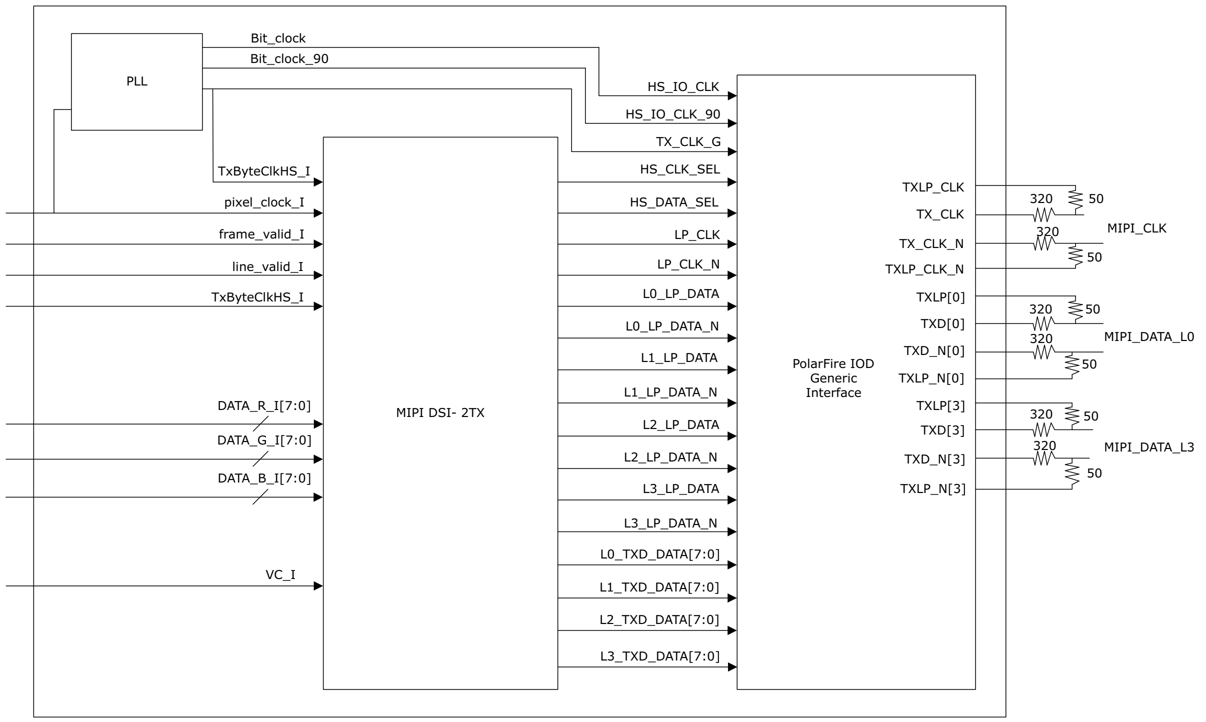
The following illustration shows the MIPI DSI2 Transmitter solution that contains the MIPI DSI TX IP. This IP is used in conjunction with the PolarFire® MIPI IOD generic interface block and PLL. The illustration shows the pin connections from the MIPI DSI TX IP to the PolarFire IOD. In clocking architecture, DSI IP needs the following clocks:

- **Pixel clk**: parallel video pixel clock
- **TxByteClkHs_I**: byte clk (3/4th of pixel clk for four lane configuration and RGB-888 data format)
- **Bitclk**: high-speed serial clock for PF IOD
- **Bitclk90**: high-speed 90° shifted serial clock for PF IOD

A PLL is required to generate the TxByteClkHs_I clock (Byte clock). The PLL is configured to produce the TxByteClkHs_I clock, MIPI high-speed bitclk, and 90° phase shifted bitclk.

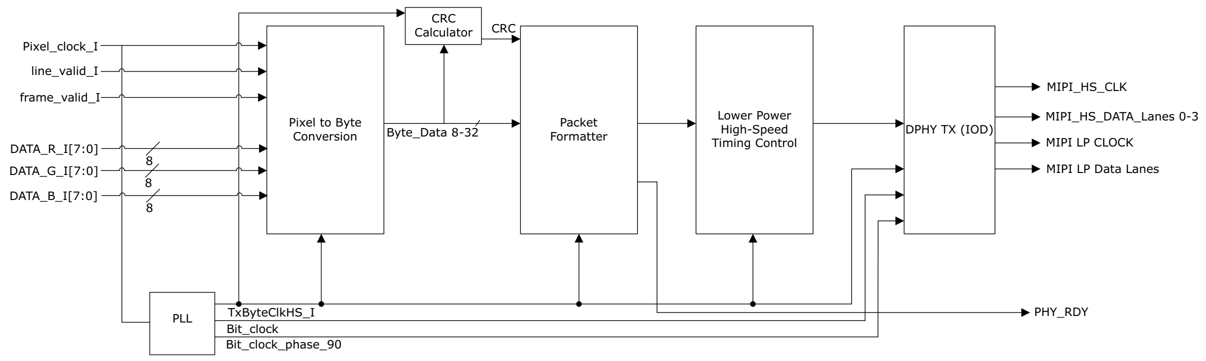
An external resistor network is needed to accommodate Low Power (LP) and High Speed (HS) mode transitioning on the same signal pairs, as shown in the following figure. It is also needed to set the voltage swing to 200 mV during HS clock and data transfers.

Figure 2 • **Architecture of MIPI DSI Transmitter Solution**



The following illustration shows the architecture of the MIPI DSI transmitter core.

Figure 3 • **Implementation of MIPI DSI Transmitter Core**



2.3 Design Description

This section describes the different internal modules of the MIPI DSI transmitter core.

2.3.1 Pixel to Byte Conversion

The DSI lanes transmit data using a high-speed bit clock, though the video data is generated using a pixel clock. The video is transmitted on the lanes in the format of bytes. The DSI IP contains a module to convert the incoming pixel data to bytes. The user is expected to transmit the pixels along with the control signals line_valid_i, frame_valid_i, and virtual channel number (vc_i). An internal clock crossing FIFO is used to convert the incoming data from pixel clock to byte clock domain. The pixel to byte conversion module also generates the byte enable signal, which indicates the valid byte data.

There are some initial settings (see: Packetizer, page 4), which the DSI IP is required to complete before accepting video data from the video generator. There is a PHY_RDY signal from the DSI IP, which should be used to enable the video timing generator. Sending video to the DSI IP before this ready signal is active may give shifted or rolling image outputs.

2.3.2 Packetizer

The packetizer module in DSI TX IP implements a packet forming state machine. The packets contain header and payload data. In each packet, it appends ECC in headers and CRC in the payload data. It generates long and short packets as required by the DCS (Display Command Set). In initial states, this state machine generates the following packets:

- Display mode setting packet
- Display sleep-out packet
- Display switch-on packet
- Display default brightness and backlight on packets
- Pixel format setting packet
- Columns and page address setting packets

Once these basic settings are done, the packetizer is ready for sending the video stream to the display. At this time, it generates a PHY_RDY signal which is used to enable a video timing generator. The packetizer writes this video data to the right-hand side of the display RAM. This process is followed for each line in each frame. The memory start command and memory continue commands are used as per the DSI specification.

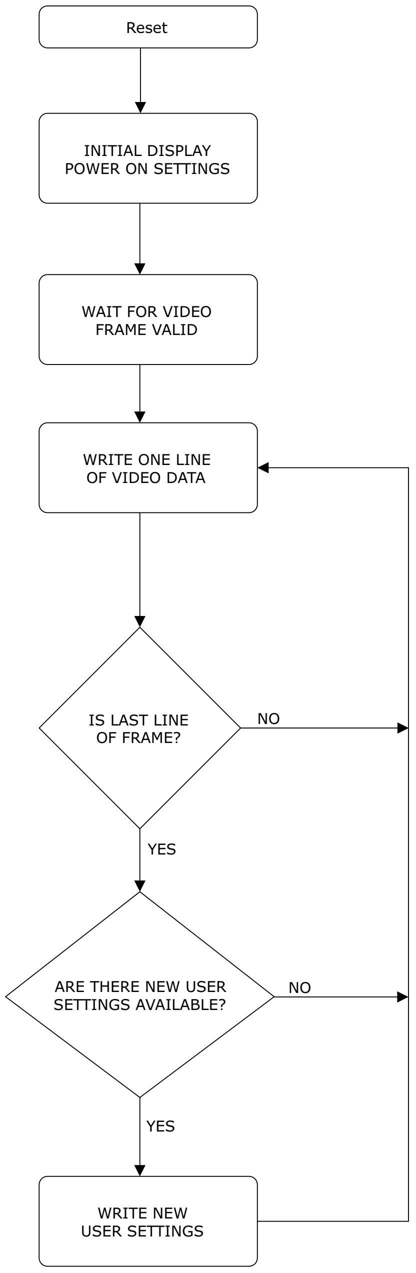
- Write memory start (DCS command 0x2C)
- Write memory continue (DCS command 0x3C)

The packetizer module also writes new brightness and backlight values to the display registers. These new values are given by the user using a command, address and data interface defined in different sections on this document. It checks for these values every frame and updates during the vertical blanking period.

Figure 4 • **Output Packets from Packetizer**

DATAID 0X39	WC[7:0]	WC[15:8]	ECC	DCS CMD	PAYLOAD DATA	CRC[7:0]	CRC[15:8]

Figure 5 • **FSM Implementation of High-Speed Data Generation**


Reset
INITIAL DISPLAY POWER ON SETTINGS
WAIT FOR VIDEO FRAME VALID
WRITE ONE LINE OF VIDEO DATA
IS LAST LINE OF FRAME?
NO
YES
ARE THERE NEW USER SETTINGS AVAILABLE?
NO
YES
WRITE NEW USER SETTINGS

2.3.3 PLL

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 (PolarFire IOD). TxByteClkHs_I must be configured such that the output MIPI DSI compliant packets sent on the interface are sampled. The following equations show the relation between Pixel_clock_i and TxByteClkHs_I, depending on the number of lanes configured.

TxByteClkHs_i = (Pixel_clock_i × Bits per pixel) / (Number of Lanes × 8).
MIPI bit clock = 4 × TxByteClkHs_I

Bits per pixel = 24

Number of lanes = 4

Two serial MIPI bit clocks are required- 0° and 90° phase shifted.

MIPI DSI TX IP supports RAW8 data types.

2.3.4 Low Power/High-Speed

Once powered up, the DSI IP must complete the initialization sequence of the display. The low power module in the IP follows the DPHY specification and takes all the clock and data lanes from low power mode to high-speed mode. Currently, all the transactions are done in high-speed mode. Transition to high-speed mode follows the following sequence: LP-11, LP-01, LP-00, HS0/1. It indicates the HS request path and following the timing based on MIPI DPHY Specification version 1.1.

Once in high-speed mode, the clock always remains in that mode. The data lanes move to the high-speed mode when there is data to send, and they return to low power mode when not sending DSI packets. An example is when in the video blanking period, the data lanes remain in low power mode.

The low power module generated SoT and EoT sequences as per the DPHY specification.

This IP does not support LP requests, Escape mode and Turn around modes.

2.3.5 DPHY TX

This module uses PolarFire IOD generic blocks to convert Byte data to serial data. A gearing ratio of four is used to convert the parallel data to serial data. It generates both HS and LP signals (for both clock and data). It also switches between HS and LP modes using the HS_CLK_SEL and HS_DATA_SEL signals.

In the PF IOD configuration, we need to specify various parameters as required by the supported data rates by the display.

2.3.6 CRC Calculator

This module uses the bytes generated from the pixel to byte conversion module and calculates the 16-bit CRC for the generated bytes. This 16-bit CRC is sent to the packetizer, which appends the value at the end of the long packet.

2.3.7 User Control

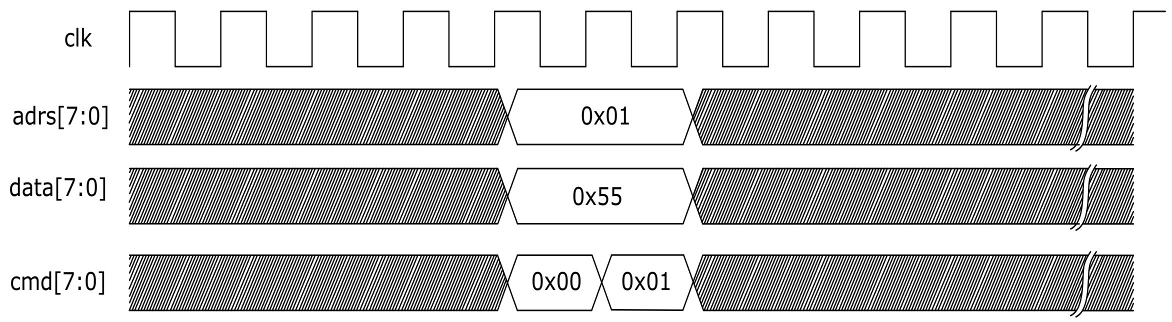
The DSI IP has a user control module. The user control module has a command, data, and address interface. This module provides a user interface to write brightness, backlight, and display on-off command to the display. A simple way to change these parameters is to set the address and data values on the ports and then change the command signal LSB bit from 0 (low) to 1 (high). Corresponding register details and waveform examples are shown as follows for clarity.

Table 1 • **Register Details of User Control**

Register Address (Hex)	Register name	Description	Details
0X00	Display_ONOFF_cmd	Switches the display ON or OFF	Bit 0: '0' display is OFF Bit 0: '1' display is ON
0X01	Display_Brightness_cmd	This register controls the brightness of the display.	0x00: minimum brightness 0xFF: maximum brightness
0X02	Display_Backlight_cmd	This register switches display backlight ON or OFF.	Bit 0: '0' display backlight is OFF Bit 0: '1' display backlight is ON

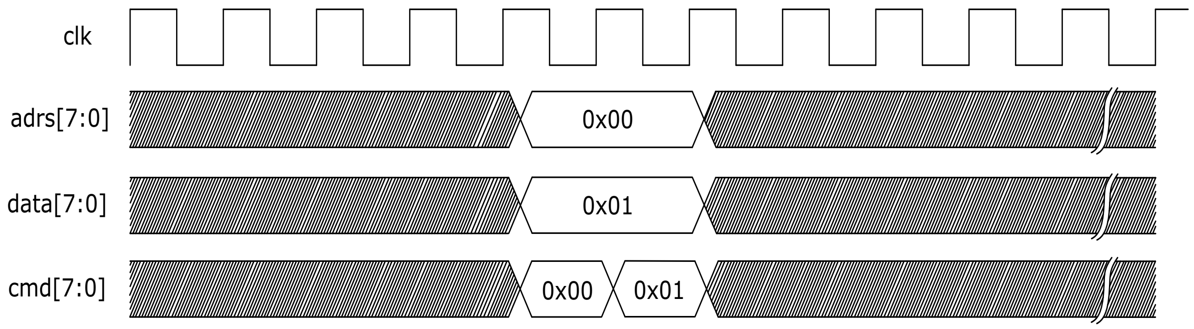
The following is an example of changing display brightness to 0x55.

Figure 6 • **Display Brightness**



The following is an example of writing display on command.

Figure 7 • **Writing Display ON Command**



2.4 Inputs and Outputs

The following table lists the input and output ports of the MIPI DSI TX configuration parameters.

Table 2 • **Input and Output Ports of the MIPI DSI Transmitter**

Signal Name	Direction	Width	Description
RESET_n_I	Input	1	Active low asynchronous reset signal to design
Pixel_clock_i	Input	1	Input clock with which incoming pixels are sampled
TxByteClkHs_I	Input	1	TX Byte clock (gearing ratio 4). This clock must be configured such that the pixels sent on the MIPI DSI interface are sampled according to it.
VC_I	Input	[1:0]	Virtual Channel Identifier. The IP supports only one virtual channel with 0
DATA_R_I	Input	[7:0]	Input Pixel Data Red
DATA_G_I	Input	[7:0]	Input Pixel Data Green
DATA_B_I	Input	[7:0]	Input Pixel Data Blue
FRAME_VALID_I	Input	1	Asserts high for every valid frame
LINE_VALID_I	Input	1	Asserts high when the valid packet is available
L(0-3)_LP_DATA	Output	1	Low power data (P side)
L(0-3)_LP_DATA_N	Output	1	Low power data (N side)
LP_CLK	Output	1	Low power clock (P side)
LP_CLK_N	Output	1	Low power clock (N side)
L(0-3)_TXD_DATA	Output	[7:0]	Lane0 to Lane3 transmit bytes
HS_CLK_SEL	Output	1	Selects HS clock or LP clock mode
HS_DATA_SEL	Output	1	Selects HS data or LP data mode
DSI_RST	Output	1	Active low Reset to the display device
USER_CNTRL_CMD_I	INPUT	[7:0]	User interface command, a toggle on bit 0 will write the data
USER_CNTRL_ADRS_I	INPUT	[7:0]	User interface address
USER_CNTRL_DATA_I	INPUT	[7:0]	User interface data
CLK_ALIGN_I	INPUT	[3:0]	Connect to CLK_ALIGN signal of PF IOD
PHY_RDY	OUTPUT	1	Asserts when DSI IP is ready to take the video data

2.5 Configuration Parameters

The following table lists the configuration parameters used in the hardware implementation of the MIPI DSI transmitter block. These are generic parameters and can vary based on the application requirements.

Table 3 • **Configuration Parameters**

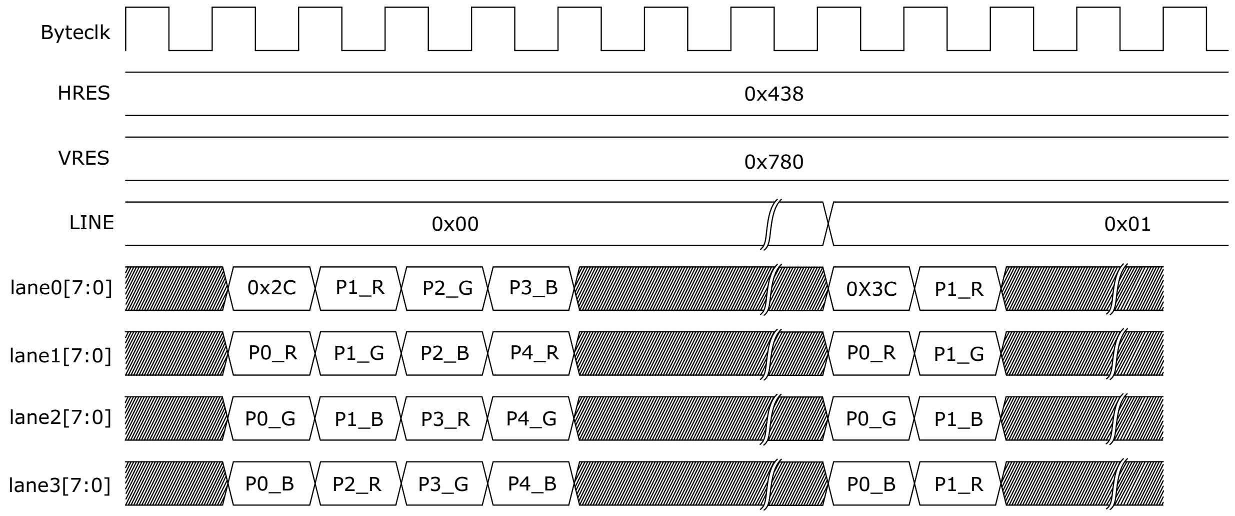
Name	Description
g_HORIZONTAL_RESOLUTION	Active horizontal resolution
g_VERTICAL_RESOLUTION	Active vertical resolution
g_PIXELCLK_FREQ_MHZ	Input pixel clock frequency

2.6 Timing Diagrams

2.6.1 Long Packet

The following illustration shows a portrait 1080x1920 display of 24-bit RGB data packets using long packets.

Figure 8 • **24-bit RGB Data Packets using Long Packets**



2.7 License

MIPI DSI transmitter requires encrypted license.

2.8 Resource Utilization

The following table lists the resource utilization of a sample MIPI DSI transmitter core implemented in a PolarFire MPF300T-1FCG1152I device for RAW8, four lanes configuration.

Table 4 • **Resource Utilization of the MIPI DSI Transmitter**

Element	Usage
DFFs	796
Fabric LUTs	926
LSRAM	10
Differential IO	5 pairs