Bitec

DSP Solutions for Industry & Research

DisplayPort IP Core Lattice Integration Manual

Version 1.0

Table of Contents

1. Revision history 3
2. Scope 4
3. General 4
4. Device Family Support 4
5. DisplayPort Feature support 4
6. Functional Description 5
7. Lattice PCS Interface 6
8. Port signals 7
    8.1.1. Port: tx_gxb_conduit 7
    8.1.2. Port: tx_reconfig_conduit 7
9. Port signals 9
    9.1.1. Port: rx_gxb_conduit 9
    9.1.2. Port: rx_reconfig_conduit 10

1. Revision history

Version	Released	Comment
1.0	03/22/2018	First release

2. Scope

This user manual refers to the Bitec DisplayPort IP Core version 6.3.

3. General

This document is an addendum to the Bitec “DisplayPort IP Core Reference Manual”.

The purpose of this document is to describe Bitec DisplayPort (DP) Core functionality which is relevant only when the Core is instantiated in a Lattice FPGA..

For general Core functionality, please refer to the Bitec “DisplayPort IP Core Reference Manual” and “DisplayPort IP Core Software Manual”. Unless otherwise specified in the rest of this document, the functionality described in the above mentioned manuals holds also for Lattice FPGA applications.

4. Device Family Support

The DisplayPort IP core is designed to work with the following Lattice FPGA devices:

- ECP5 Devices

5. DisplayPort Feature support

The Lattice FPGA family does not support the full feature set of DisplayPort. The features limitations are shown below.

- Support for 1,2 & 4-lane
- Support DisplayPort 1.4
- Support 1.62 or 2.7Gbps link rate
- Dual/quad symbol modes
- Single/dual/quad pixel modes
- 4,8,10,12 & 16 bit color support
- Supports RGB, YCbCr Colorimetric Formats
- Autonomous AUX channel or GPU driven
- AUX debug channel
- Optional HDCP Support
- 8-Channel Audio
- eDP Features supported

6. Functional Description

The Bitec DisplayPort IP Core provides Sink and Source functionality which can be instantiated independently. The core has been designed with a rich set of parameters to allow device resource optimizations.

The Core Top level is shown in Figure 2. The DisplayPort Encoder and Decoder use a common Transceiver controller and MM Slave interface port. The core ports are configurable under parameterization using either a RTL wrapper instantiation or via the Clarity Designer Tool.

The core itself is device independent. All architectural sensitive components are instantiated externally to the encrypted core. This provides added flexibility for system designers.

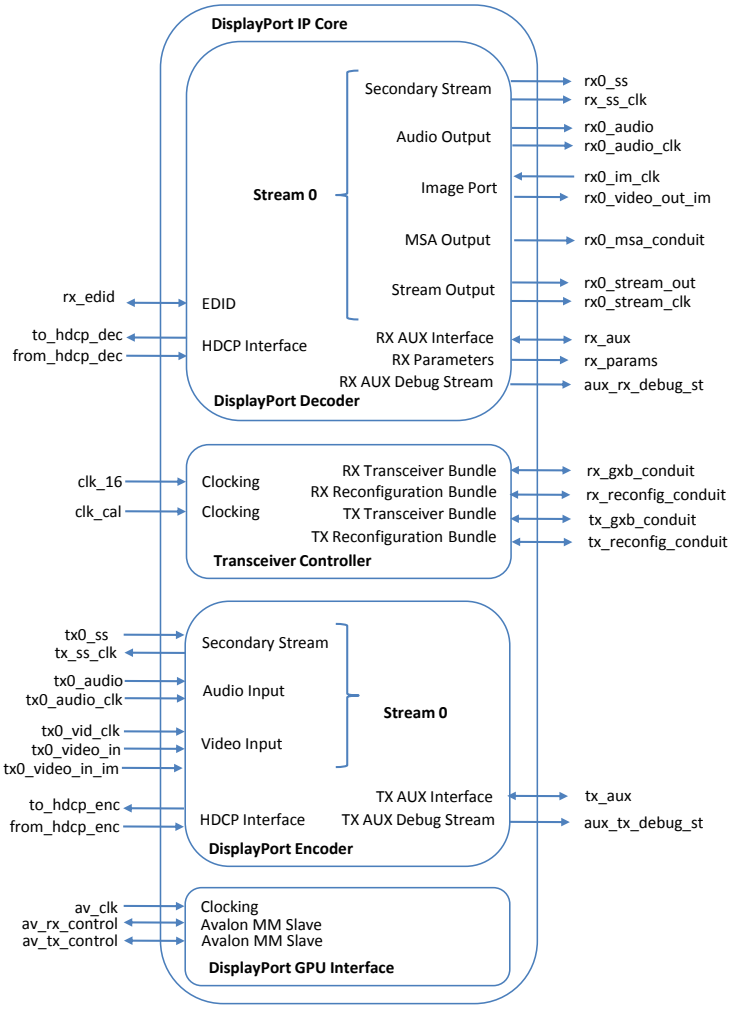

Figure 1. DisplayPort IP Core top level diagram.

7. Lattice PCS Interface

When the Bitec DP Core is used in Lattice FPGA devices, internal Lattice-specific wrappers are used between the Generic MAC blocks and the Lattice transceivers.

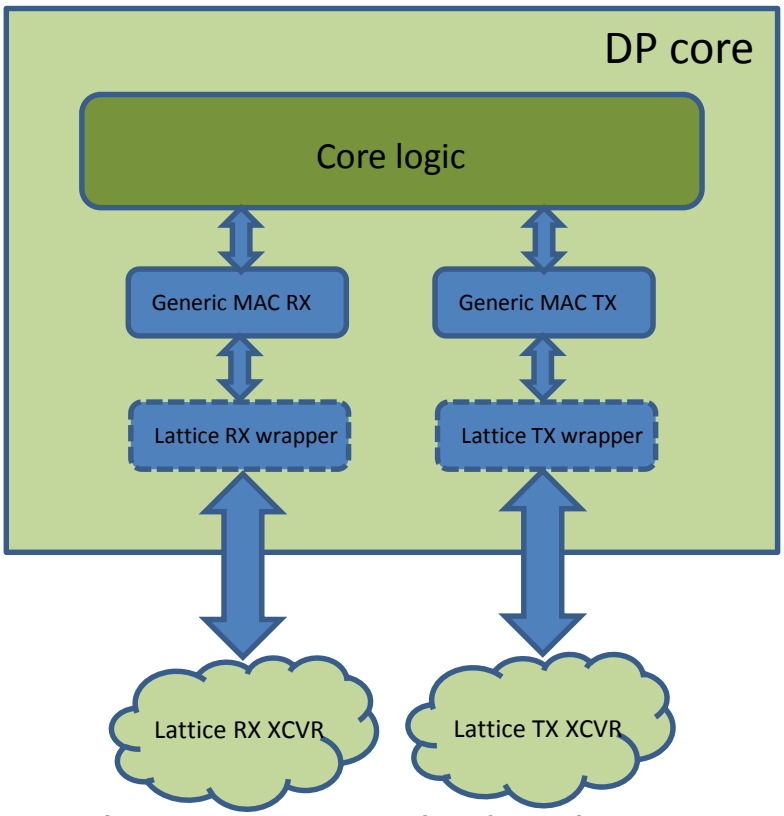

Figure 2. DP Core connections for Lattice PMAs.

8. Port signals

The Source signal ports specific to Lattice FPGA applications are detailed in Table 1. Note, some of these interfaces are present or may be missing according to the parameter selections in the DisplayPort Qsys GUI or RTL Wrapper instantiation.

Table 1. Source port groups.

Port	Clock domain	Description
tx_gxb_conduit	**tx_clkout**	Signal bundle for connection to TX Transceiver
Signal	**Direction**	**Description**
conduit_from_gxb_tx[1:0]	Input	See Section 8.1.1
conduit_to_gxb_tx[164:0]	Output	See Section 8.1.1
Port	**Clock domain**	**Description**
tx_reconfig_conduit	**clk_cal**	Signal bundle for reconfiguration of TX Transceiver
Signal	**Direction**	**Description**
conduit_from_tx_reconfig	Input	See section 8.1.2
conduit_to_tx_reconfig[]	Output	See section 8.1.2

8.1.1. Port: tx_gxb_conduit

The DisplayPort Source core can be set to operate in dual-symbol mode, where 20 bits (2 symbols) are forwarded to the Transceiver within one clock cycle or in quad-symbol mode, where 40 bits (4 symbols) per clock cycle are used. The choice has to be made according to the used device capabilities and the maximum main link bitrate desired.

The `tx_gxb_conduit` is a signal bundle used to interface to the device TX Transceivers. The bundle dimensions will change according to the number of lanes and the symbol mode used within the design. The tables below illustrate the bundle contents.

Table 2. conduit_to_gxb_tx port signals (dual-symbol mode).

conduit_to_gxb_tx			
Transceiver signal	**4-lanes**	**2-lanes**	**1-lane**
Parallel data to XCVR	conduit_to_gxb_tx[79:0]	conduit_to_gxb_tx[39:0]	conduit_to_gxb_tx[19:0]
Reset to TX XCVR	conduit_to_gxb_tx[83:80]	conduit_to_gxb_tx[41:40]	conduit_to_gxb_tx[20]
Power down to TX XCVR	conduit_to_gxb_tx[84]	conduit_to_gxb_tx[42]	conduit_to_gxb_tx[21]

Table 3. conduit_to_gxb_tx port signals (quad-symbol mode).

conduit_to_gxb_tx			
Transceiver signal	**4-lanes**	**2-lanes**	**1-lane**
Parallel data to XCVR	conduit_to_gxb_tx[159:0]	conduit_to_gxb_tx[79:0]	conduit_to_gxb_tx[39:0]
Reset to TX XCVR	conduit_to_gxb_tx[163:160]	conduit_to_gxb_tx[81:80]	conduit_to_gxb_tx[40]
Power down to TX XCVR	conduit_to_gxb_tx[164]	conduit_to_gxb_tx[82]	conduit_to_gxb_tx[41]

Table 4. conduit_from_gxb_tx port signals.

conduit_from_gxb_tx			
Transceiver signal	**4-lanes**	**2-lanes**	**1-lane**
XCVR locked input	conduit_from_gxb_tx[0]	conduit_from_gxb_tx[0]	conduit_from_gxb_tx[0]
XCVR parallel data clock	conduit_from_gxb_tx[1]	conduit_to_gxb_tx[1]	conduit_to_gxb_tx[1]

8.1.2. Port: tx_reconfig_conduit

The Transceiver reconfiguration is necessary on the Source side to allow link rate, voltage and pre-emphasis changes during DisplayPort link training. This port bundle dimensions and signals are dependent on the lane count parameterization. Each lane in the design will have a corresponding 2-

bit voltage (vod_lanex) and 2-bit pre-emphasis (pre_lanex). All lanes must be configured at the same time using the rising edge of the reconfig_analog_req signal (for vod_lanex and pre_lanex) and the rising edge of the reconfig_linkrate_req signal (for tx_link_rate).

Table 5 and Table 6 illustrate the conduit mappings. The bundle signals should be used to drive and return the appropriate signals from a Transceiver reconfiguration component.

Table 5. conduit_to_tx_reconfig port signals.

conduit_to_tx_reconfig			
Bundle signal	**4-lanes**	**2-lanes**	**1-lane**
reconfig_analog_req	conduit_to_tx_reconfig[0]	conduit_to_tx_reconfig[0]	conduit_to_tx_reconfig[0]
reconfig_linkrate_req	conduit_to_tx_reconfig[1]	conduit_to_tx_reconfig[1]	conduit_to_tx_reconfig[1]
tx_link_rate[7:0]	conduit_to_tx_reconfig[9:2]	conduit_to_tx_reconfig[9:2]	conduit_to_tx_reconfig[9:2]
vod_lane0[1:0]	conduit_to_tx_reconfig[11:10]	conduit_to_tx_reconfig[11:10]	conduit_to_tx_reconfig[11:10]
vod_lane1[1:0]	conduit_to_tx_reconfig[13:12]	conduit_to_tx_reconfig[13:12]	
vod_lane2[1:0]	conduit_to_tx_reconfig[15:14]		
vod_lane3[1:0]	conduit_to_tx_reconfig[17:16]		
pre_lane0[1:0]	conduit_to_tx_reconfig[19:18]	conduit_to_tx_reconfig[15:14]	conduit_to_tx_reconfig[13:12]
pre_lane1[1:0]	conduit_to_tx_reconfig[21:20]	conduit_to_tx_reconfig[17:16]	
pre_lane2[1:0]	conduit_to_tx_reconfig[23:22]		
pre_lane3[1:0]	conduit_to_tx_reconfig[25:24]		

reconfig_linkrate_req is asserted for at least one clk_cal clock cycle when the Transceiver must be reconfigured to the tx_link_rate bitrate. reconfig_analog_req is asserted for at least one clk_cal clock cycle when the Transceiver voltage and pre-emphasis must be reconfigured.

It should be noted that the 2-bit voltage and pre-emphasis may need mapping to 5 or 6-bit values according to the FPGA device family being used. tx_link_rate is expressed in multiples of 270 Mbps (e.g. 6 = 1.62Gbps, 10 = 2.7Gbps, 20 = 5.4Gbps, etc.).

Table 6. conduit_from_tx_reconfig port signals.

conduit_from_tx_reconfig			
reconfig signal	**4-lane**	**2-lane**	**1-lane**
busy	conduit_from_tx_reconfig[0]	conduit_from_tx_reconfig[0]	conduit_from_tx_reconfig[0]

The busy signal should be connected to a signal coming from the Transceiver, asserted when the Transceiver is busy during reconfiguration and/or calibration.

Signal clk_cal is a 50 MHz clock input. This clock should be synchronous with the clock used for the Transceiver Reconfiguration block (mgmt_clk_clk, external to the Sink Core). For devices using a 50 MHz mgmt_clk_clk, connect the same clock directly also to clk_cal. For devices using a 100 MHz mgmt_clk_clk, connect the same clock to clk_cal through a by-2 divider.

9. Port signals

The Sink signal ports specific to Lattice FPGA applications are detailed in Table 7. Note, some of these interfaces are present or may be missing according to the parameter selections in the DisplayPort Qsys GUI or RTL Wrapper instantiation.

Table 7. Sink port groups.

Port	Clock domain	Description
rx_gxb_conduit	**rx_clkout[]**	Signal bundle for connection to RX Transceiver
Signal	**Direction**	**Description**
conduit_from_gxb_rx[168:0]	Input	See Section 9.1.1
conduit_to_gxb_rx[20:0]	Output	See Section 9.1.1
Port	**Clock domain**	**Description**
rx_reconfig_conduit	**clk_cal**	Signal bundle for reconfiguration of RX Transceiver
Signal	**Direction**	**Description**
conduit_from_rx_reconfig	Input	See section 9.1.2
conduit_to_rx_reconfig[]	Output	See section 9.1.2

9.1.1. Port: rx_gxb_conduit

The DisplayPort Sink core can be set to operate in dual-symbol mode, where 20 bits (2 symbols) are read from the Transceiver within one clock cycle or in quad-symbol mode, where 40 bits (4 symbols) per clock cycle are used. The choice has to be made according to the used device capabilities and the maximum main link bitrate desired.

The `rx_gxb_conduit` is a signal bundle used to interface to the device RX Transceivers. The dimensions and internal mappings are dependent on the number of lanes and the symbol mode used within the design. The bundle contents and their associated Transceiver port mapping are shown in the tables below.

Table 8. conduit_to_gxb_rx port signals.

conduit_to_gxb_rx			
Transceiver signal	**4-lanes**	**2-lanes**	**1-lane**
XCVR Digital reset	conduit_to_gxb_rx[3:0]	conduit_to_gxb_rx[1:0]	conduit_to_gxb_rx[0]
XCVR Analog reset	conduit_to_gxb_rx[7:4]	conduit_to_gxb_rx[3:2]	conduit_to_gxb_rx[1]
XCVR Power down	conduit_to_gxb_rx[8]	conduit_to_gxb_rx[4]	conduit_to_gxb_rx[2]
XCVR CDR Lock to Ref	conduit_to_gxb_rx[12:9]	conduit_to_gxb_rx[6:5]	conduit_to_gxb_rx[3]
XCVR CDR Lock to data	conduit_to_gxb_rx[16:13]	conduit_to_gxb_rx[8:7]	conduit_to_gxb_rx[4]
XCVR Bit-slip command	conduit_to_gxb_rx[20:17]	conduit_to_gxb_rx[10:9]	conduit_to_gxb_rx[5]
XCVR Main Reset	conduit_to_gxb_rx[21]	conduit_to_gxb_rx[11]	conduit_to_gxb_rx[6]

Table 9. conduit_from_gxb_rx port signals (dual-symbol mode).

conduit_from_gxb_rx			
Transceiver signal	**4-lanes**	**2-lanes**	**1-lane**
XCVR Parallel data	conduit_from_gxb_rx[79:0]	conduit_from_gxb_rx[39:0]	conduit_from_gxb_rx[19:0]
XCVR CDR PLL Locked	conduit_from_gxb_rx[83:80]	conduit_from_gxb_rx[41:40]	conduit_from_gxb_rx[20]
XCVR CDR data locked	conduit_from_gxb_rx[87:84]	conduit_from_gxb_rx[43:42]	conduit_from_gxb_rx[21]
XCVR data clock	conduit_from_gxb_rx[88]	conduit_from_gxb_rx[44]	conduit_from_gxb_rx[22]

Table 10. conduit_from_gxb_rx port signals (quad-symbol mode).

conduit_from_gxb_rx			
Transceiver signal	**4-lanes**	**2-lanes**	**1-lane**
XCVR Parallel data	conduit_from_gxb_rx[159:0]	conduit_from_gxb_rx[79:0]	conduit_from_gxb_rx[39:0]
XCVR CDR PLL Locked	conduit_from_gxb_rx[163:160]	conduit_from_gxb_rx[81:80]	conduit_from_gxb_rx[40]
XCVR CDR data locked	conduit_from_gxb_rx[167:164]	conduit_from_gxb_rx[83:82]	conduit_from_gxb_rx[41]
XCVR data clock	conduit_from_gxb_rx[168]	conduit_from_gxb_rx[84]	conduit_from_gxb_rx[42]

9.1.2. Port: rx_reconfig_conduit

The Transceiver reconfiguration is necessary on the Sink side to allow link rate changes during DisplayPort link training. All lanes must be configured at the same time using the rising edge of the `reconfig_linkrate_req` signal.

Table 11 and Table 12 illustrate the conduit mappings. The bundle signals should be used to drive and return the appropriate signals from a Transceiver reconfiguration component.

Table 11. conduit_to_rx_reconfig port signals.

conduit_to_rx_reconfig			
Bundle signal	**4-lanes**	**2-lanes**	**1-lane**
reconfig_linkrate_req	conduit_to_rx_reconfig[0]	conduit_to_rx_reconfig[0]	conduit_to_rx_reconfig[0]
rx_link_rate[7:0]	conduit_to_rx_reconfig[8:1]	conduit_to_rx_reconfig[8:1]	conduit_to_rx_reconfig[8:1]
reconfig_analog_req	conduit_to_rx_reconfig[9]	conduit_to_rx_reconfig[9]	conduit_to_rx_reconfig[9]
vod_lane0[1:0]	conduit_to_rx_reconfig[11:10]	conduit_to_rx_reconfig[11:10]	conduit_to_rx_reconfig[11:10]
vod_lane1[1:0]	conduit_to_rx_reconfig[13:12]	conduit_to_rx_reconfig[13:12]	
vod_lane2[1:0]	conduit_to_rx_reconfig[15:14]		
vod_lane3[1:0]	conduit_to_rx_reconfig[17:16]		
pre_lane0[1:0]	conduit_to_rx_reconfig[19:18]	conduit_to_rx_reconfig[15:14]	conduit_to_rx_reconfig[13:12]
pre_lane1[1:0]	conduit_to_rx_reconfig[21:20]	conduit_to_rx_reconfig[17:16]	
pre_lane2[1:0]	conduit_to_rx_reconfig[23:22]		
pre_lane3[1:0]	conduit_to_rx_reconfig[25:24]		

`reconfig_linkrate_req` is asserted for at least one `clk_cal` clock cycle when the Transceiver must be reconfigured to the `rx_link_rate` bitrate. `Rx_link_rate` is expressed in multiples of 270 Mbps (e.g. 6 = 1.62Gbps, 10 = 2.7Gbps, 20 = 5.4Gbps, etc.).
`reconfig_analog_req` is asserted for at least one `clk_cal` clock cycle when the main link voltage and pre-emphasis values are changed by the connected Source during Link Training. The Transceiver may use this information to tune its equalizer and adapt to the new main link analog values.

Table 12. conduit_from_rx_reconfig port signals.

conduit_from_rx_reconfig	
reconfig signal	
busy	conduit_from_rx_reconfig[0]

The `busy` signal should be connected to a signal coming from the Transceiver, asserted when the Transceiver is busy during reconfiguration and/or calibration.

Signal `clk_cal` is a 50 MHz clock input. This clock should be synchronous with the clock used for the Transceiver Reconfiguration block (`mgmt_clk_clk`, external to the Sink Core). For devices using a 50 MHz `mgmt_clk_clk`, connect the same clock directly also to `clk_cal`. For devices using a 100 MHz `mgmt_clk_clk`, connect the same clock to `clk_cal` through a by-2 divider.

Bitec
Urb. Los Angeles
C/ Guadaiza 25
Edificio Diz
San Pedro Alcantara
Malaga, 29670, Spain
Tel : +34 91 632 69 66
Fax : +34 91 790 50 27
E-mail: info@bitec-dsp.com
Internet: www.bitec-dsp.com

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Development Software ***category:***

Click to view products by Lattice ***manufacturer:***

Other Similar products are found below :

RAPPID-560XBSW RAPPID-567XFSW DG-ACC-NET-CD SRP004001-01 SW006021-1NH SW163052 SYSWINEV21 Core429-SA SW500006-HPA CWP-BASIC-FL W128E13 CWP-PRO-FL SYSMACSE210L SYSMACSE203L AD-CCES-NODE-1 NT-ZJCAT1-EV4 CWA-BASIC-FL RAPPID-567XKSW CWA-STANDARD-R SW89CN0-ZCC CWA-LS-DVLPR-NL VDSP-21XX-PCFLOAT RAPPID-563XMSW IPS-EMBEDDED SWR-DRD-L-01 SDAWIR-4532-01 SYSMAC-SE201L MPROG-PRO535E AFLCF-08-LX-CE060-R21 WS02-CFSC1-EV3-UP SYSMAC-STUDIO-EIPCPLR LIB-PL-PC-N-1YR-DISKID SYSMACSE2XXL LS1043A-SWSP-PRM 1120270005 1120270006 MIKROBASIC PRO FOR FT90X (USB DONGLE) MIKROC PRO FOR AVR (USB DONGLE LICENSE) MIKROC PRO FOR FT90X (USB DONGLE) MIKROBASIC PRO FOR AVR (USB DONGLE LICEN MIKROBASIC PRO FOR FT90X MIKROC PRO FOR DSPIC30/33 (USB DONGLE LI MIKROC PRO FOR FT90X MIKROC PRO FOR PIC32 (USB DONGLE LICENSE 52202-588 MIKROPASCAL PRO FOR ARM (USB DONGLE LICE MIKROPASCAL PRO FOR FT90X MIKROPASCAL PRO FOR FT90X (USB DONGLE) MIKROPASCAL PRO FOR PIC32 (USB DONGLE LI SW006021-2H