High Speed SelectIO Wizard v2.0

LogiCORE IP Product Guide

Vivado Design Suite

PG188 September 30, 2015

XILINX
ALL PROGRAMMABLE™

Table of Contents

Introduction

The LogiCORE™ IP High Speed SelectIO™ Wizard simplifies the integration of SelectIO technology into high-speed system designs for UltraScale™ devices. This wizard creates a Verilog HDL file that instantiates and configures I/O and clocking logic such as RX_BITSLICE, TX_BITSLICE, RXTX_BITSLICE, BITSLICE_CONTROL and PLL blocks present in the physical-side interface (PHY) architecture. Additionally, this core provides pin planning for the configured interface and updates the register transfer level (RTL) based on constraints.

Features

- User selectable interface type such as TX only, RX only and a mix of TX, RX and Bidir bus directions
- For RX interfaces, the clock/strobe to data relationship is selectable according to the protocol setting.
- Supports serialization factor of 8 (Div4) and 4 (Div2)
- PLL clock source can be from either the GC pin or from the interconnect driven through BUFG.
- Range of the user selectable PLL input clock frequencies for a given data speed
- Configurable I/O delays
- Optional register interface unit (RIU) interface and bitslip logic
- Bank selection and pin planning of all the available pins in the bank with design rule checks to meet SelectIO Logic requirement.

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	UltraScale Architecture
Supported User Interfaces	RIU
Resources	Performance and Resource Utilization web page
Provided with Core	
Design Files	RTL
Example Design	Verilog
Test Bench	Verilog
Constraints File	Xilinx Design Constraints (XDC)
Simulation Model	Not Provided
Supported S/W Driver(2)	N/A
Tested Design Flows(3)	
Design Entry	Vivado® Design Suite
Simulation	For supported simulators, see the Xilinx Design Tools: Release Notes Guide.
Synthesis	Vivado Synthesis
Support	
Provided by Xilinx at the Xilinx Support web page	

Notes:

1. For a complete list of supported devices, see the Vivado IP catalog.
2. Standalone driver details can be found in the software development kit (SDK) directory (<*install_directory*>/doc/usenglish/xilinx_drivers.htm). Linux OS and driver support information is available from the Xilinx Wiki page.
3. For the supported versions of the tools, see the Xilinx Design Tools: Release Notes Guide.

Overview

The High Speed SelectIO™ wizard provides the source HDL wrapper for TX, RX and RXTX bitslices in native mode. The wizard also generates LOC constraints for synthesis and implementation runs.

Feature Summary

Configuration

- Configures bus direction, RX External Clock and Data, Interface speed, PLL clock source, PLL input clock frequency, bank, delay type/delay values on the TX/RX pins, data 3-state and strobe 3-state
- Additionally RIU interface and Bitslip can be configured

Pin Selection

For each byte group, configures the pin selection, bus direction, signal type, data/strobe, and signal name. Allows you to choose the TX/RX/Bidir bus direction with the same configuration.

Each I/O bank contains 52 pins that can be configured as TX/RX/BIDIR. The wizard provides pin level configurability for the following;

- TX/RX/BIDIR
- Single ended/Differential
- Data/strobe/input clock/Clock forward
- Customizable signal name

Applications

This solution is useful for high-speed I/O interface requirements like ASIC emulation and chip-to-chip interaction and any serial protocols operating at line rates < 1,600 Mb/s.

Unsupported Features

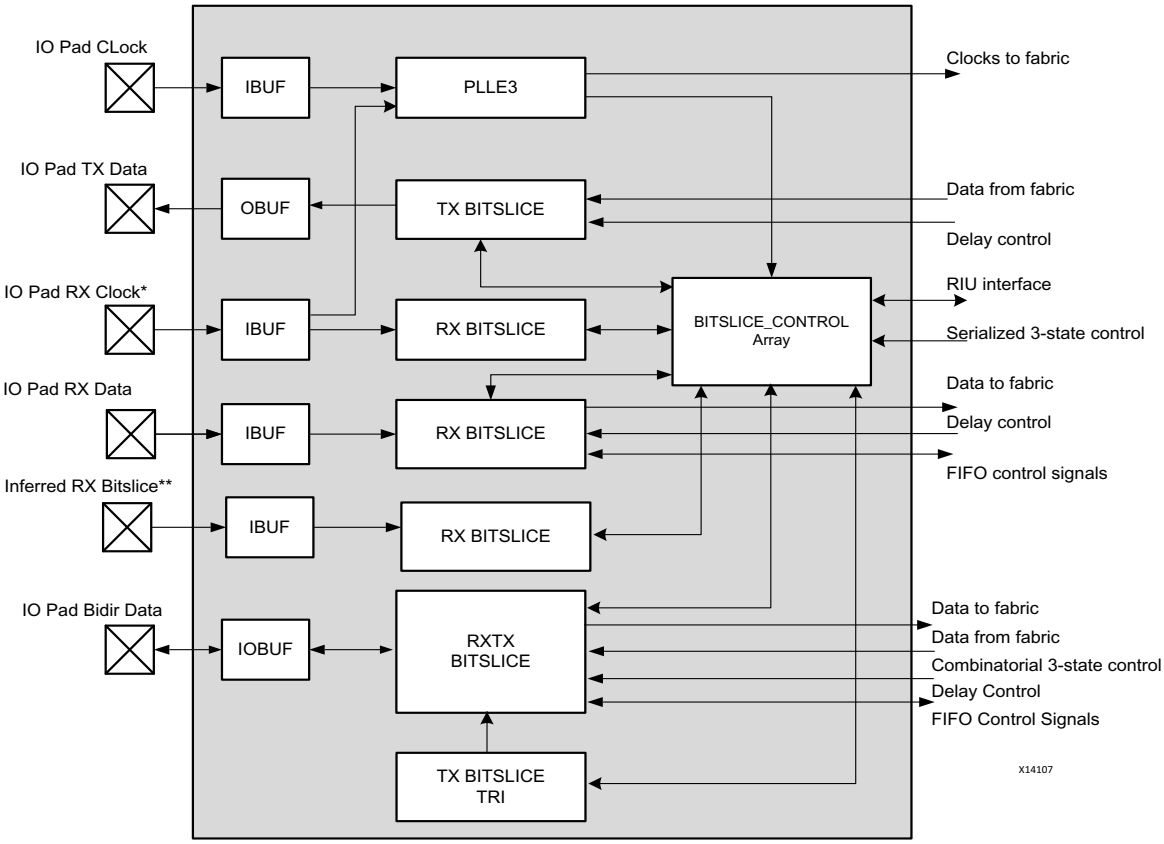
- Count mode (delay format)
- Dynamic Phase Alignment (DPA) feature in the example design
- DRC free partner generation in the example design for a mix of bus directions in device under test (DUT).

Licensing and Ordering Information

This Xilinx LogiCORE™ IP module is provided at no additional cost with the Xilinx Vivado® Design Suite under the terms of the Xilinx End User License. Information about this and other Xilinx LogiCORE IP modules is available at the Xilinx Intellectual Property page. For information about pricing and availability of other Xilinx LogiCORE IP modules and tools, contact your local Xilinx sales representative.

Product Specification

Each I/O bank in UltraScale™ devices contains 52 pins that can be used for input, output, or bidirectional. The High Speed SelectIO™ Wizard provides various options to generate a wrapper using TX_BITSLICE, RX_BITSLICE, RXTX_BITSLICE, and BITSLICE_CONTROL for the user selected configuration in native mode. This wizard also configures clocking circuitry using PLL. A representative design using TX_BITSLICE, RX_BITSLICE, RXTX_BITSLICE and BITSLICE_CONTROL is shown in Figure 2-1.



* IO RX Clock is valid only in Edge DDR/Center DDR and clock pin is on GC_QBC pin (Byte group2 pin0)
** No need to drive data on inferred RX bitslice. Wizard infers a bitslice for strobe propagation.

Figure 2-1: **Representative Design**

The HDL wrapper instantiates all the bitslices related to an I/O bank. Depending on the pin selected, corresponding bitslices are connected to the general interconnect and I/Os. The Wizard also provides LOC constraints to the top level ports.

Performance

The HSSIO Wizard performance is measured by the data speed that it can support, which in turn is the capability realized in the selected I/Os of the device. Refer to the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1] for more details on the performance characteristics.

This core supports a maximum data speed of 1,600 Mb/s for serialization factor 8 and 1,300 Mb/s for serialization factor 4. The general interconnect logic works at 1/8 or 1/4 clock of the interface speed clock.

Resource Utilization

For full details about performance and resource utilization, visit the Performance and Resource Utilization web page

This core uses a flip-flop to generate `clkoutphyen` of the PLL for the circuit stabilization before the actual data is sent on the I/O. Other blocks used are FPGA hard blocks.

Port Descriptions

Table 2-1 and Table 2-2 describe the input and output ports of the High Speed SelectIO Wizard. Availability of ports is controlled by user selection.

Table 2-1 lists ports that are top-level ports connected to the FPGA I/O. The renaming feature is useful for multiple instances of the core in the design to avoid conflicting pin LOC constraints

Table 2-1: **Ports Connected to FPGA I/O**

Port	Direction	Clock Domain	Description
Global Ports			
clk_p	Input	NA	Differential clock input P connected to PLL
clk_n	Input	NA	Differential clock input N connected to PLL
clk	Input	NA	Single ended clock connected to PLL
rst	Input	riu_clk	Global reset pin. Assertion of the reset is asynchronous. Deassertion is synchronous with respect to the RIU clock. Minimum pulse width should be 5 ns, Refer to the PLL Switching Characteristics section in the *Kintex UltraScale Architecture Data Sheet* (DS892)[Ref 2].
bg<x>_pin<y>_<pin num>	Input/ Output/ Inout	NA	Data/Input Clock/Strobe/Clk forward ports connected to I/O pins. Port name of the wizard IP is user-specified through Vivado® IDE appended with the pin number in the bank. To elaborate, the wizard appends the pin_num (range from 0 - 51, number of pins in a full bank) to the port name during generation. The default port name is bg<x>_pin<y>_<pin num>. (You can overwrite the default port name from its default value) In this naming convention, *x* indicates the byte group and its value ranges from 0 to 3 and *y* indicates the pin position within the bytegroup and its value ranges from 0-12. For example, if you select the 13th pin and the signal name is bg1_pin0, the actual port name is bg1_pin0_13 and the corresponding signal name to the interconnect end would be - data_from_fabric_bg1_pin10_13 or data_to_fabric_bg1_pin10_13 depending on the direction of the pin chosen.
bg<x>_pin<y>_nc	Input	NA	Inferred bitslice ports. The wizard infers bitslice0 of a nibble for strobe propagation within the bytegroup; <x> indicates bytegroup (0,1,2,3); <y> indicates bitslice0 position (0 in case of lower nibble, 6 in case of upper nibble.) There is no need to drive any data on these ports.

Table 2-2: **Ports Connected to FPGA General Interconnect Logic**

Port	Direction	Clock Domain	Description
Data Ports			
data_from_fabric_<sig_name>_<y> [sf -1 :0]	Input	pll0_clkout0	Parallel data input from the interconnect logic to TX/Bidir TX bitslices. <sig_name> is the signal name configured for TX/Bidir bus direction during customization. <y> is the pin number (range from 0 -51) appended by the wizard. sf is the serialization factor.
data_to_fabric_<sig_name>_<y>[sf-1 :0]	Output	pll0_clkout0	Parallel data output to general interconnect logic from RX/Bidir RX bitslices. <sig_name> is the signal name configured for RX/Bidir bus direction during customization. <y> is the pin number (range from 0 to 51) appended by the wizard. sf is the serialization factor.
Clock Ports			
pll0_clkout_0	Output	NA	Divided version of clock from PLL CLKOUT0 for the general interconnect logic. The frequency of this is data speed divided by the serialization factor. This clock can be used as a clock for the general interconnect logic.
pll0_clkout1	Output	NA	Divided version of the clock from PLL CLKOUT1 for the general interconnect logic from PLL0. The frequency of this can be set in Vivado IDE from a list of supported frequencies for the data speed chosen.
pll0_clkfbout	Output	NA	Feedback clock from PLL
pll1_clkout0	Output	NA	Divided version of the clock from PLL CLKOUT1 for the general interconnect logic from PLL1. The frequency of this can be set in Vivado IDE from a list of supported frequencies for the data speed chosen.

Table 2-2: **Ports Connected to FPGA General Interconnect Logic *(Cont'd)***

Port	Direction	Clock Domain	Description
RX/Bidir RX Delay Control Ports			
rx_clk	Input	NA	DELAY clock used to sample LOAD, CE and INC for RX.
rx_cntvalue_in_<i> [8:0]	Input	rx_clk/bidir_rx_clk	Counter value from the FPGA logic for dynamically loadable tap value for RX. <i> is the corresponding pin number on which the RX/Bidir bus direction is selected.
rx_cntvalue_out_<i>[8:0]	Output	rx_clk/bidir_rx_clk	Counter value going to FPGA logic for monitoring the tap value for RX. <i> is the corresponding pin number on which the RX/Bidir bus direction is selected.
rx_ce_<i>	Input	rx_clk/bidir_rx_clk	Clock enable for the IDELAY register clock for RX. <i> is the corresponding pin number on which the RX/Bidir bus direction is selected.
rx_en_vtc<i>	Input	rx_clk/bidir_rx_clk	Active-High to enable DELAYCTRL to keep delay over voltage and temp to load new delay for RX Delay Control. <i> is the pin number on which the RX/Bidir bus direction is selected.
rx_inc<i>	Input	rx_clk/bidir_rx_clk	Increment the current delay tap setting for RX. <i> is the pin number on which the RX/Bidir bus direction is selected.
rx_load<i>	Input	rx_clk/bidir_rx_clk	Load the count value from CNTVALUEIN for RX. <i> is the pin number on which the RX/Bidir bus direction is selected.
RX Extended Delay Control Ports			
rx_cntvalue_in_ext_<i>[8:0]	Input	rx_clk	Extended counter value from the FPGA logic for dynamically loadable tap value for RX. <i> is the corresponding pin number on which the RX bus direction is selected.
rx_cntvalue_out_ext_<i>[8:0]	Output	rx_clk	Extended counter value going to the FPGA logic for monitoring the tap value for RX. <i> is the corresponding pin number on which the RX bus direction is selected.
rx_ce_ext_<i>	Input	rx_clk	Extended clock enable for the IDELAY register clock for RX. <i> is the corresponding pin number on which the RX bus direction is selected.

Table 2-2: **Ports Connected to FPGA General Interconnect Logic *(Cont'd)***

Port	Direction	Clock Domain	Description
rx_en_vtc_ext_<i>	Input	rx_clk	Extended active-High to enable DELAYCTRL to keep delay over voltage and temp to load new delay for RX Delay Control. <i> is the pin number on which the RX bus direction is selected.
rx_inc_ext_<i>	Input	rx_clk	Extended increment the current delay tap setting for RX. <i> is the pin number on which the RX bus direction is selected.
rx_load_ext_<i>	Input	rx_clk	Extended load count value from CNTVALUEIN for RX. <i> is the pin number on which the RX bus direction is selected.
TX/Bidir TX Delay Control Ports			
tx_clk	Input	NA	DELAY clock used to sample LOAD, CE INC for TX
tx_cntvalue_in_<i>[8:0]	Input	tx_clk/bidir_tx_clk	Counter value from the FPGA logic for dynamically loadable tap value for TX. <i> is the corresponding pin number on which the TX/Bidir bus direction is selected.
tx_cntvalue_out_<i>[8:0]	Output	tx_clk/bidir_tx_clk	Counter value going to the FPGA logic for monitoring the tap value for TX. <i> is the corresponding pin number on which the TX/Bidir bus direction is selected.
tx_ce_<i>	Input	tx_clk/bidir_tx_clk	Clock enable for the IDELAY register clock for RX. <i> is the corresponding pin number on which the TX/Bidir bus direction is selected.
tx_en_vtc<i>	Input	tx_clk/bidir_tx_clk	Active-High to enable DELAYCTRL to keep delay over voltage and temp to load new delay for TX Delay Control. <i> is the pin number on which the TX/Bidir bus direction is selected.
tx_inc<i>	Input	tx_clk/bidir_tx_clk	Increment the current delay tap setting for TX. <i> is the pin number on which the TX/Bidir bus direction is selected.
tx_load<i>	Input	tx_clk/bidir_tx_clk	Load the count value from CNTVALUEIN for TX. <i> is the pin number on which the TX/Bidir bus direction is selected.

Table 2-2: **Ports Connected to FPGA General Interconnect Logic *(Cont'd)***

Port	Direction	Clock Domain	Description
Bidir Delay Clock Ports			
bidir_tx_clk	Input	NA	DELAY clock used to sample LOAD, CE, INC for TX
bidir_rx_clk	Input	NA	DELAY clock used to sample LOAD, CE, INC for RX
TXBITSLICE_TRI Delay Control Ports			
bidir_tx_bs_tri_clk	Input	bidir_tx_clk	Delay Clock used to sample LOAD, CE, INC
bidir_tx_bs_tri_cntvaluein_<n>[8:0]	Input	bidir_tx_clk	Counter value from the FPGA logic for dynamically loadable tap value, 1 per nibble, <n> indicates the nibble number that varies from 0 to 7.
bidir_tx_bs_tri_cntvalueout_<n>[8:0]	Output	bidir_tx_clk	Counter value to the FPGA logic for monitoring the tap value of the delay control, 1 per nibble, <n> indicates the nibble number that varies from 0 to 7.
bidir_tx_bs_tri_ce_<n>	Input	bidir_tx_clk	Clock Enable for the ODELAY register clock, 1 per nibble, <n> indicates the nibble number that varies from 0 to 7.
bidir_tx_bs_tri_en_vtc_<n>	Input	bidir_tx_clk	Active-High to enable DELAYCTRL to keep delay over voltage and temp to load a new delay for TXBITSLICE TRI Delay Control. <n> indicates the nibble number that varies from 0 to 7.
bidir_tx_bs_tri_inc_<n>	Input	bidir_tx_clk	Increment the current delay tap setting, 1 per nibble. <n> indicates the nibble number that varies from 0 to 7.
bidir_tx_bs_tri_load_<n>	Input	bidir_tx_clk	Load the count value from CNTVALUEIN, 1 per nibble. <n> indicates the nibble number that varies from 0 to 7.
3-State Ports			
tri_tbyte_<n>[3:0]	Input	pll0_clkout0	Serialized 3-state input, 1 per nibble, <n> indicates the nibble number that varies from 0 to 7.
tri_t_<i>	Input	pll0_clkout0	Combinatorial Data 3-State input, 1 per bitslice. <i> is the pin number on which the bidir bus is selected.

Table 2-2: **Ports Connected to FPGA General Interconnect Logic *(Cont'd)***

Port	Direction	Clock Domain	Description
RIU Ports			
riu_addr_bg<m>[5:0]	Input	riu_clk	Address of the RIU register. <m> indicates bytegroup. Varies from 0 to 3.
riu_clk_bg<m>	Input	NA	System clock from the general interconnect. <m> indicates bytegroup. Varies from 0 to 3.
riu_nibble_sel_bg<m>[1:0]	Input	riu_clk	Nibble select to enable RIU read/write for upper (Logic 1 or lower nibble 0). <m> indicates bytegroup. Varies from 0 to 3.
riu_wr_data_bg<m>[15:0]	Input	riu_clk	Input write data to the register. <m> indicates bytegroup. Varies from 0 to 3.
riu_wr_en_bg<m>	Input	riu_clk	Register write enable active-High. <m> indicates bytegroup. Varies from 0 to 3.
riu_rd_data_bg<m>[15:0]	Output	riu_clk	Output read data to the controller. <m> indicates bytegroup. Varies from 0 to 3.
riu_valid_bg<m>	Output	riu_clk	Output read valid to the controller. <m> indicates bytegroup. Varies from 0 to 3.
Status/Control			
pll0_locked	Output	NA	Logic High indicates PLL is locked to the desired clock frequency.
dly_rdy_bsc<n>	Output	riu_clk	Indicates fixed delay calibration completion.
vtc_rdy_bsc<n>	Output	riu_clk	PHY calibration is complete (VTC is ready – after EN_VTC is enabled).
en_vtc_bsc<n>	Input	riu_clk	Active-High to enable DELAYCTRL to keep delay over voltage and temp low to load new delay.
fifo_empty_<i>	Output	pll0_clkout0	FIFO empty flag from each bitslice. <i> is the pin number on which the RX/Bidir RX is selected.
fifo_rd_en_<i>	Input	pll0_clkout0	Fifo read enable for each bit slice. <i> is the pin number on which the RX/Bidir RX is selected.
fifo_rd_clk_<i>	Input	pll0_clkout0	Fifo read clock for each bitslice. <i> is the pin number on which the RX/Bidir RX is selected.

Table 2-2: **Ports Connected to FPGA General Interconnect Logic *(Cont'd)***

Port	Direction	Clock Domain	Description
bitslip_<i>	Input	pll0_clkout0	Control input that makes the data bitslip for the RX/ Bidir RX bus directions.
bitslip_error_<i>	Output	pll0_clkout0	Error output for bitslip, When eight bitslips are performed for 8-bit serialization or four bitslips are performed for 4-bit serialization, this output is pulsed High.
fifo_wr_clk_<n>	Output	NA	Fifo write clock from the bitslice0 of each nibble on which the strobe is available. This pin is available only if **PLL0_FIFO_WRITE_CLK_OUT** is chosen in Vivado IDE.
rst_seq_done	Output	riu_clk	Indicates that the reset sequence is completed and the wizard is ready. When this signal is asserted High, transactions can be initiated from the general interconnect logic.

Designing with the Core

This chapter includes guidelines and additional information to facilitate designing with the core.

General Design Guidelines

This core is for high-speed UltraScale™ architecture designs and can be configured for a data speed range of 300-1,600 Mb/s. The following steps are recommended for all designs using the High Speed SelectIO™ Wizard.

You are expected to have the interface requirements of your application prior to generating the HSSIO wizard specific to your designs — details such as interface speed, clock to data relationship, and system clocking structure. For example, what should be the source of PLL input clock, board pin layout constraints are expected to be specified while generating with the wizard.

There can be two approaches that you can follow when it comes to pin planning. If there are no board layout restrictions, you can do the pin planning through the wizard, generate the wizard and make sure the Vivado® design tools perform synthesis and implementation without any DRCs. You can also try to retrofit the board pin layout into the wizard pin planning. Again it is important to make sure the Vivado design tools complete the synthesis and implementation without any design rule checks (DRCs). In either approach, you must ensure that Vivado synthesis and implementation goes through without DRCs to avoid issues at the later stage of board design.

Clocking

The PLLs associated with each bank are the primary source for clocking the SelectIO resources in the bank. You can select the source of the input reference clock to PLL. The source can be a clock from the global clock pin (GC) or from the general interconnect through BUFG. You can also select the input clock frequency from Vivado IDE, which lists all the supported clocks for a given device.

Clocking of TX_BITSLICE

When transmitting data, the master input clock to the BITSLICE_CONTROL is used. The frequency of this clock determines the serial bit rate of data. Data is transmitted on the serial lines along with the associated clock or strobe. TX uses PLL0 by default unless RX operates in Serial Mode, when TX uses PLL1.

The clock forwarding option in the Wizard can be enabled on any pin in a given bank. Currently, the Wizard supports only one pin as Clock forward. The phase of the forwarded clock with respect to the data can be set in Vivado IDE.

The clock/strobe can be edge aligned or center aligned with the data. Clock/Strobe is generated similar to the data by applying a 10101010 pattern at the D[7:0] input of TX_BITSLICE / RXTX_BITSLICE. The alignment of the strobe to the data is achieved by setting the TX_OUTPUT_PHASE_90 attribute of the TX_BITSLICE/RXTX_BITSLICE.

Clocking of RX_BITSLICE

PLL0 is the primary clock source for RX.There are two modes used to capture data in the RX_BITSLICEs initiated by an attribute of the BITSLICE_CONTROL component (SERIAL_MODE = TRUE/FALSE).

- When the attribute SERIAL_MODE is set to TRUE, the received data is captured using CLKOUTPHY from PLL0. This corresponds to the **Async/None/Fractional** option shown in Vivado IDE for the RX External Clock to Data Alignment. The frequency of CLKOUTPHY is half of the interface speed. In accordance with the clocking rules given in the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1], the Wizard infers a RX_BITSLICE in the 0th position of a nibble related to the Rx data pin.

  Note: When RX operates in Serial Mode, the TX and RX bitslices cannot be combined in the same nibble.

- When the attribute SERIAL_MODE is set to FALSE, the received data is captured using a clock or strobe that is forwarded with the data. This corresponds to the Edge DDR/ Center DDR/Edge DDR Strobe/ Center DDR Strobe options shown in Vivado IDE for the RX External Clock to Data Alignment.

Edge DDR

In this mode, the clock and strobe are the same. This means you have a DDR clock in the system, which can be used for data capture. The clock should be given as an input only on the GC/QBC pin (Bytegroup2 pin0). It is also the input reference clock to PLL; hence it is mandatory for the clock to be free-running and continuous. This clock is forwarded to all the RX data pins using the Inter Byte and Inter Nibble clocking rules as mentioned in the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1]. In accordance with the rules, the Wizard infers a RX_BITSLICE in the GC/QBC pin location (Bytegroup2 pin0). The input clock is connected to the input of RX_BITSLICE. The following figure shows the alignment of clock to data.

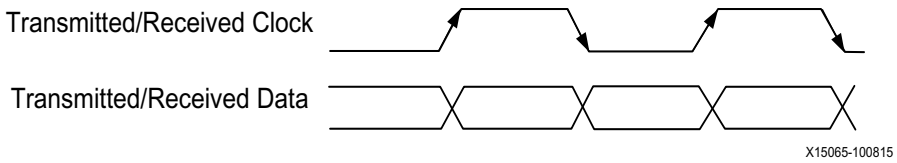


Figure 3-1: **EDGE DDR**

Center DDR

This is the same as in Figure 3-1 except the alignment of Clock to Data is shifted by 90 degrees. The following figure shows the alignment of clock to data.

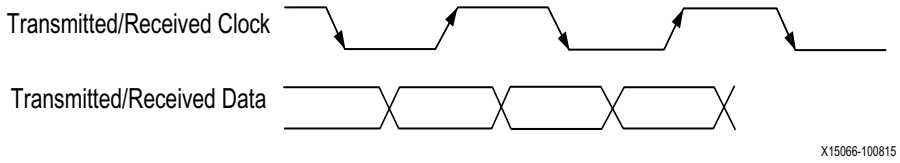


Figure 3-2: **Center DDR**

Edge DDR Strobe

In this mode, the RX data is captured using the incoming strobe. The strobe input can be present on any of the QBC pins present in a bank. The Wizard supports up to 8 strobes in a given bank. The propagation of strobes to RX data pins follows the Inter byte and Inter nibble clocking rules given in the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1]. In accordance with the rules, the Wizard infers a RX_BITSLICE in the 0th position of every nibble where the strobe is to be propagated. The I/Os associated with the inferred bitslices are brought to the top-level wrapper with the "_nc" suffix. It is mandatory to constrain these I/Os in the XDC file for correct functioning of the design.

The strobe pin nearest to an RX data pin is chosen as the associated strobe for the data pin. Special care needs to be taken while pin planning to ensure that the appropriate strobe and data pin positions are chosen.

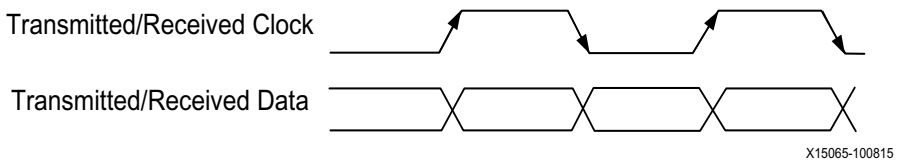


Figure 3-3: **Edge DDR Strobe**

Center DDR Strobe

This is the same as in Edge DDR Strobe except that the strobe to data alignment is shown in Figure 3-4.

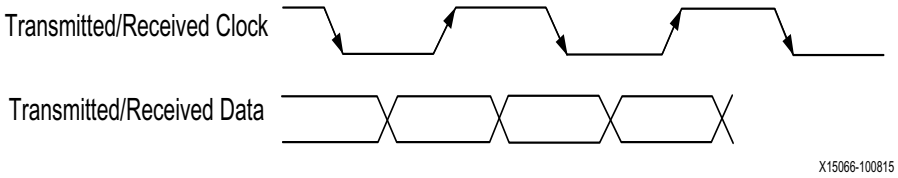


Figure 3-4: **Center DDR Strobe**

Clocking of RXTX_BITSLICE

The TX part of RXTX_BITSLICE follows the same clocking as given in Clocking of TX_BITSLICE.

The RX part of RXTX_BITSLICE supports only EDGE DDR Strobe or Center DDR Strobe mode. This means that the presence of a strobe is mandatory for data capture in RXTX_BITSLICE.

Usage of PLLs

If the design contains a mix of TX, RX and RX_TX bitslices, the usage of PLLs is shown in Table 3-1. When RX operates in Serial Mode, TX and RX bitslices cannot be combined in the same nibble. When RX operates in Serial Mode, Bidirectional (RXTX) operation is not supported.

Table 3-1: **Usage of PLLs**

Design Configuration	External Clock to Data Alignment	PLL Usage	Description
TX + RX	Edge DDR, Center DDR, Edge DDR Strobe, Center DDR Strobe	TX uses PLL0	The master input clock to all BITSLICE_CONTROL modules is from PLL0.
		RX uses PLL0	
TX + RX	Async/None/Fractional	TX uses PLL1	The master input clock to BITSLICE_CONTROL associated with TX is from PLL1.
		RX uses PLL0	The master input clock to BITSLICE_CONTROL associated with RX is from PLL0.
TX + RX + RXTX	Edge DDR Strobe, Center DDR Strobe	TX uses PLL0	The master input clock to all BITSLICE_CONTROL modules is from PLL0.
		RX uses PLL0	
		RXTX uses PLL0	

Resets

The Wizard generates a reset module that is built in the wrapper. The input is an asynchronous reset that triggers resets to all Bitslice and Bitslice Control modules in the design. The following diagram gives the details of the reset state machine.

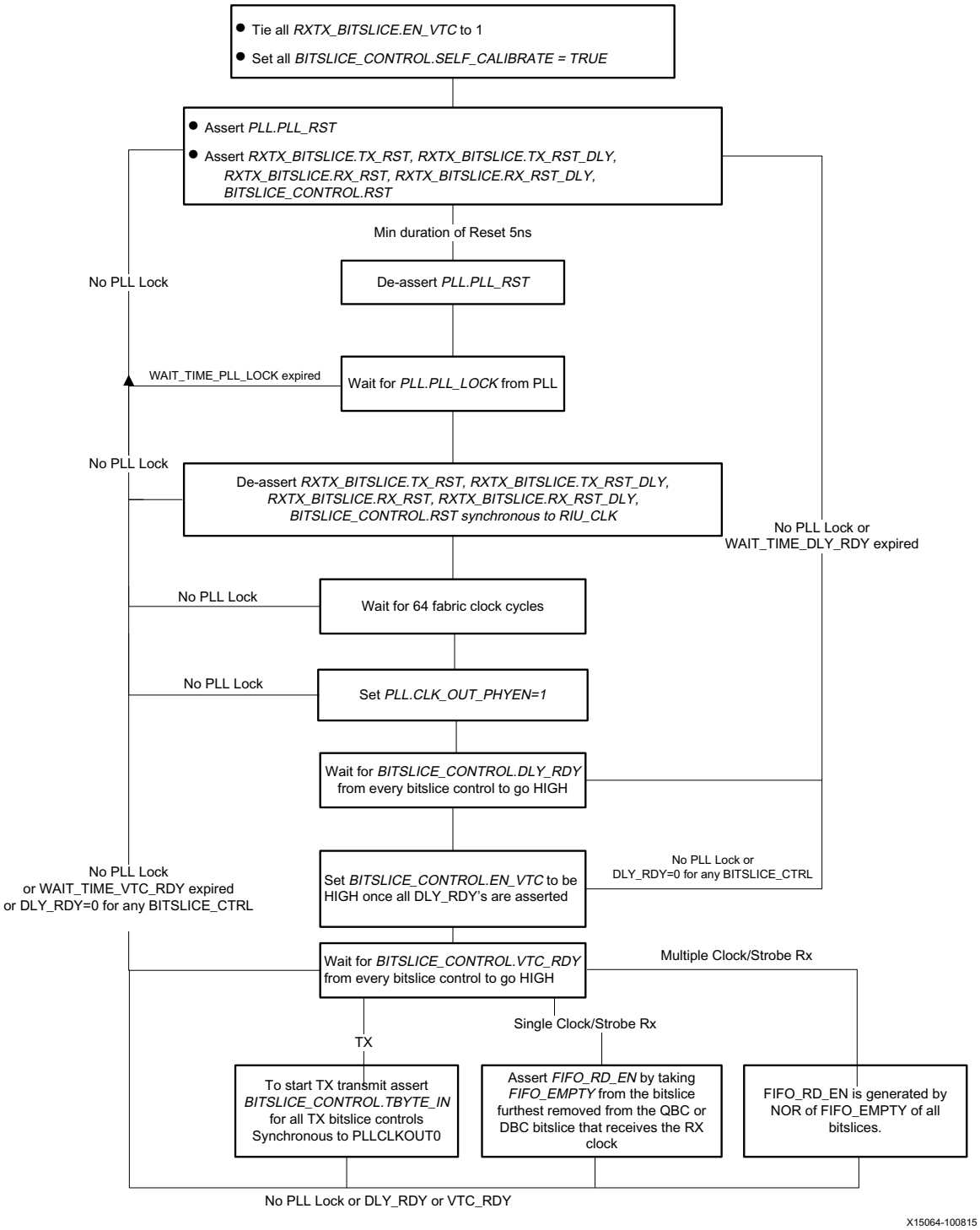


Figure 3-5: **HSSIO Reset Sequence**

When any pin is enabled as Clock Forward, it is mandatory to hold the counterpart design (RX) in reset until the TX is out of reset and `rst_seq_done` of TX is asserted. This will ensure a reliable clock to the RX.

Note: All wait times shown in Figure 3-5 are hard coded to 1 million RIU clock cycles.

Protocol Description

The SelectIO resources can be used to interface any serial protocol operating in the 300 Mb/s to 1,600 Mb/s range. For more details, refer to the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1].

Design Flow Steps

This chapter describes customizing and generating the core, constraining the core, and the simulation, synthesis and implementation steps that are specific to this IP core. More detailed information about the standard Vivado® design flows and the IP integrator can be found in the following Vivado Design Suite user guides:

- *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994) [Ref 3]
- *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4]
- *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 5]
- *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 6]

Customizing and Generating the Core

This section includes information about using Xilinx tools to customize and generate the core in the Vivado Design Suite. You can customize the IP for use in your design by specifying values for the various parameters associated with the IP core using the following steps:

1. Select the IP from the IP catalog.
2. Double-click the selected IP or select the Customize IP command from the toolbar or right-click menu.

For details about starting a Vivado project, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4] and the *Vivado Design Suite User Guide: Getting Started* (UG910) [Ref 5].

For details about output generation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4].

Note: Figures in this chapter are illustrations of the Vivado IDE. The layout depicted here might vary from the current version.

General Vivado IDE Settings

- **Component Name**: Component name is user defined. Component names must not contain any reserved words in Verilog.

Configuration Settings

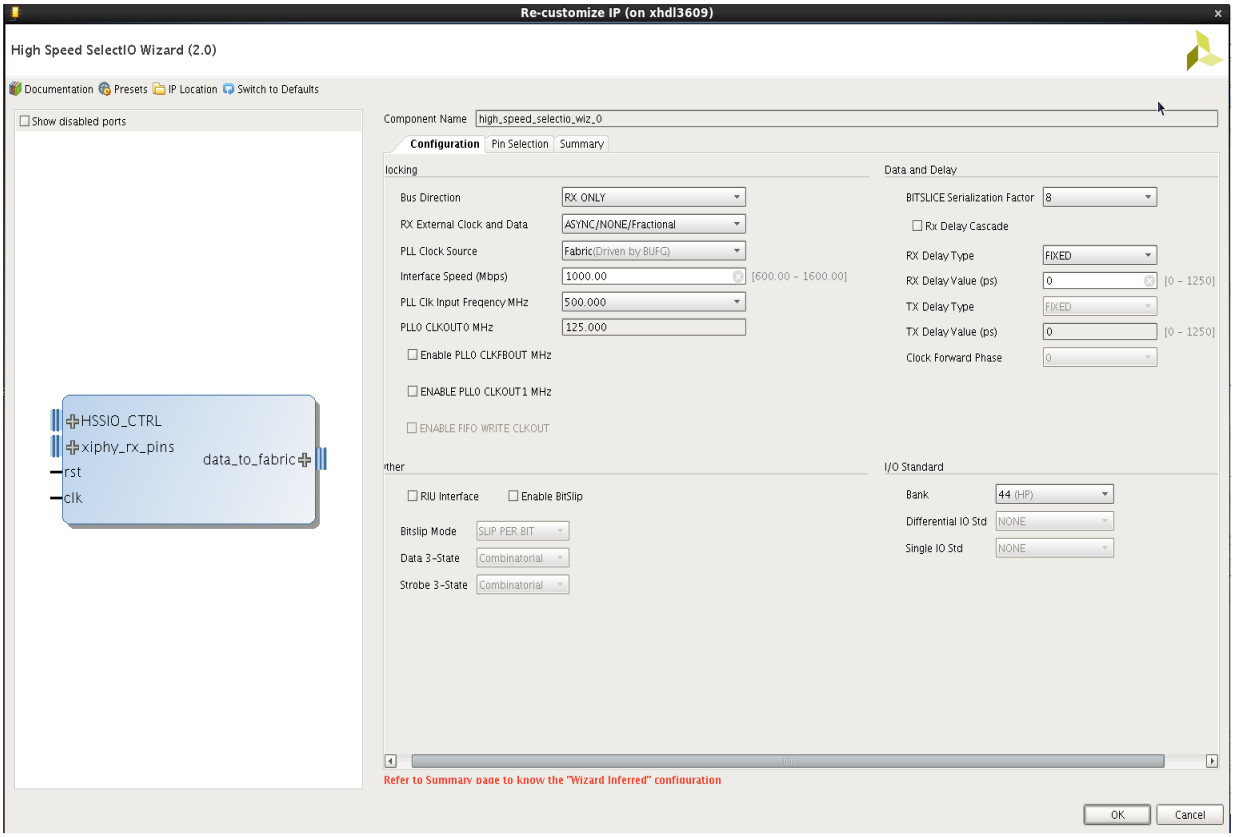


Figure 4-1: **Configuration Settings for the Interface**

Clocking

Bus Direction

Indicates the direction of the pins in the user design.

- **TX ONLY**: Indicates that the user design contains only TX pins.
- **RX ONLY**: Indicates that the user design contains only RX pins.
- **BIDIR or TX+RX or TX+RX+BIDIR**: Indicates that the user design contains a mix of TX, RX and BiDir pins.

RX External Clock and Data

Indicates the alignment of external clock to data. This is applicable for RX and Bidir pins.

- **Edge DDR**: Applicable only when Bus Direction is set to RX Only. Refer to Clocking in Chapter 3 for detailed information.
- **Center DDR**: Applicable only when Bus Direction is set to RX Only. Refer to Clocking in Chapter 3 for detailed information.
- **ASYNC/NONE/Fractional**: Refer to Clocking in Chapter 3 for detailed information.
- **Edge DDR Strobe**: Applicable to RX and BiDir pins. Refer to Clocking in Chapter 3 for detailed information.
- **Center DDR Strobe**: Applicable to RX and BiDir pins. Refer to Clocking in Chapter 3 for detailed information.

PLL Clock Source

The clock is sent to the PLL through the buffer set in this option.

- **GC Pin**: Choose this option if the input clock is available on the GC pin. In this case the input clock goes to PLL via IBUF IBUF (for single-ended clock) or IBUFDS (for differential clock) instantiated by the wizard.
- **Fabric** (Driven by BUFG): Choose this option if the clock is sent from the general interconnect. In this case the input clock goes to PLL via IBUF (for single-ended clock) or IBUFDS (for differential clock). BUFG is expected to be in this scenario.

Interface Speed (Mbps)

Sets the interface speed for the configuration. For 8-bit serialization the range is 300-1,600. For 4-bit serialization the range is 300-1,300.

When RX External Clock and Data is set to ASYNC/NONE/Fractional, the interface speed depends on the Bus Direction selected. Table 4-1 indicates the interface speed ranges when RX External Clock and Data is set to ASYNC/NONE/Fractional.

Table 4-1: **Interface Speed**

RX External Clock and Data	Bus Direction	BITSLICE Serialization Factor	Interface Speed Range
Not Applicable	TX only	8	300-1,600
ASYNC/NONE/Fractional	RX only	8	600-1,600
ASYNC/NONE/Fractional	BIDIR or RX + TX or TX + RX + BIDIR	8	600-1,600
Not Applicable	TX only	4	300-1,300
ASYNC/NONE/Fractional	RX only	4	600-1,300
ASYNC/NONE/Fractional	BIDIR or RX + TX or TX + RX + BIDIR	4	600-1,300

PLL Clk Input Frequency MHz

Sets the input clock frequency for the PLL. Depending on the data speed selected, a range of supported input clock frequencies are listed. For EDGE DDR/CENTER DDR, the input clock frequency is set to half of the data speed.

PLL0 CLKOUT0 MHz

Displays the frequency of the CLKOUT0 of PLL0 (Divided clock from PLL0). The frequency of this clock is Data speed/Serialization factor.

Enable PLL0 CLKFBOUT MHz

Setting this enables the CLKFBOUT port of PLL0 to be brought as output port of the wrapper.

ENABLE PLL0 CLKOUT1 MHz

Setting this, enables the CLKOUT1 port of PLL0 to be brought as the output port of the wrapper.

Note: If two PLLs are being used by the design, `CLKOUT1` of PLL1 is not available as an output port. Only `CLKOUT0` of PLL0 is available if Enable PLL0 CLKOUT0 is enabled.

Enable FIFO WRITE CLKOUT

Enable the FIFO internal divided write clock output to the interconnect logic. This clock is from bitslice 0 of each nibble on which the strobe is available. Refer to *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1] for more information.

Data and Delay Settings

BITSLICE Serialization Factor

Defines the serialization factor for parallel data input/output width from/to the general interconnect. Legal values are 4 and 8. The interface speed maximum range depends on the serialization factor. The maximum data speed range is 1,300 MHz if the serialization factor is 4. 1,600 MHz if the serialization factor is 8. The serialization factor is set to 8 by default.

RX Delay Cascade

Enables cascading of IDELAY and extended delay lines to get a total of 2.5 ns delay on the RX datapath. RX delay cascade is not supported for bidir pins.

RX Delay Type

- **FIXED**: Fixed delay value set through **RX Delay Value** is applied on RX data.
- **VARIABLE**: Delay on RX data can be incremented or decremented from the default value using delay control inputs CE, CLK, and INC.
- **VAR_LOAD**: Delay on RX data can be incremented or decremented from the default set value or loaded with a new value on CNTVALUEIN using delay control inputs CE, CLK, and INC.

RX Delay Value (ps)

Value of the desired RX delay in picoseconds. Maximum value is 1,250 ps when RX Delay Cascade is FALSE. If RX Delay Cascade is enabled, this delay can be configured to 1,250 or 2,500 ps.

Note: For bidirectional PINS RX and TX delay types should be the same. The same is the case with RX and TX delay values.

TX Delay Type

- **FIXED**: Fixed delay value set through **TX Delay Value** is applied on TX data.
- **VARIABLE**: Delay on TX data can be incremented or decremented from the default value using delay control inputs CE, CLK, and INC.
- **VAR_LOAD**: Delay on TX Data can be incremented or decremented from the default set value or loaded with a new value on CNTVALUEIN using delay control inputs CE, CLK, and INC.

TX Delay Value

Value of desired TX delay in picoseconds. The logic to allow conversion of the delay value (in ps) to a certain number of taps is built into the I/O control logic. This logic requires a reference clock. No tap dependent jitter is added by the delay line.

Note: For bidirectional buses, RX/TX delay values should be the same if the Delay type is **FIXED**.

Clock Forward Phase

Available only for the TX pins. Sets the phase between clock forward and TX data. Supported values are 0 and 90.

Other Settings

RIU Interface

Enables Register Interface Unit (RIU) for each bytegroup to access internal registers. Every delay element tap setting can be read with the RIU. Various features, such as clock gating and Voltage Temperature (VT) tracking, can be disabled. With this option, you can dynamically change the FIFO usage (for example, from synchronous to asynchronous to full bypass).

Enable Bitslip

For bus direction RX, Bidir RX, bitslip logic can be enabled to align the RX data with the expected pattern.

Bitslip Mode

SLIP PER BIT: Bitslip the data per bit. When pulsed for a single clock cycle, data is shifted bit per bit. When held High for several clock cycles, the data is shifted by one bit. The Bitslip pulse should be asserted once every 16 general interconnect clock cycles. For details on the implementation, refer to the *Bitslip in Logic Application Note* (XAPP1208) [Ref 7].

Data 3-State

Sets the 3-state control for the bidirectional Data and Strobe pins. Strobes are valid only in receive mode.

- **Combinatorial**: Uses the T pin of the RXTX_BITSLICE. The T input from general interconnect logic directly goes to RXTX_BITSLICE. Each data bitslice has one strobe pin. When Strobe is available, RX data is captured with respect to the strobe.
- **Serialized**: The TBYTE_IN input coming from general interconnect logic goes to the BITSLICE_CONTROL and controls the TBYTE_IN of RXTX_BITSLICE through TX_BITSLICE_TRI. Refer to the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1] on how to use TBYTE_IN.

Strobe 3-State

Sets the 3-state control for the Strobe pins. The DBC/QBC pins can be set as strobe pins and are valid only for receive mode. Refer to the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1] for information on strobes.

- **Combinatorial**: Uses the T pin of the RXTX_BITSLICE. The T input from general interconnect logic directly goes to RXTX_BITSLICE. Each data bitslice has one strobe pin. When Strobe is available, RX data is captured with respect to the strobe.
- **Serialized**: The TBYTE_IN input coming from general interconnect logic goes to the BITSLICE_CONTROL and controls the TBYTE_IN of RXTX_BITSLICE through TX_BITSLICE_TRI. Refer to the *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1] for information on TBYTE_IN.

I/O Standard Settings

Bank

Lists all available High Performance (HP) and High Range (HR) banks for the selected device.

Differential IO Standard

Selection of this standard is disabled in the Vivado IDE. I/O standard constraints should be set in the XDC file.

Single IO Standard

Selection of this standard is disabled in Vivado IDE. I/O standard constraints should be set in the XDC file.

Pin Selection Tab

This tab provides the option to select the pins and for each selected pin allows you to set the pin direction (TX, RX, BIDIR), signal type (Differential/Single ended), signal name, and also select a pin as Data, Strobe, Input Clock or Clock Forward.

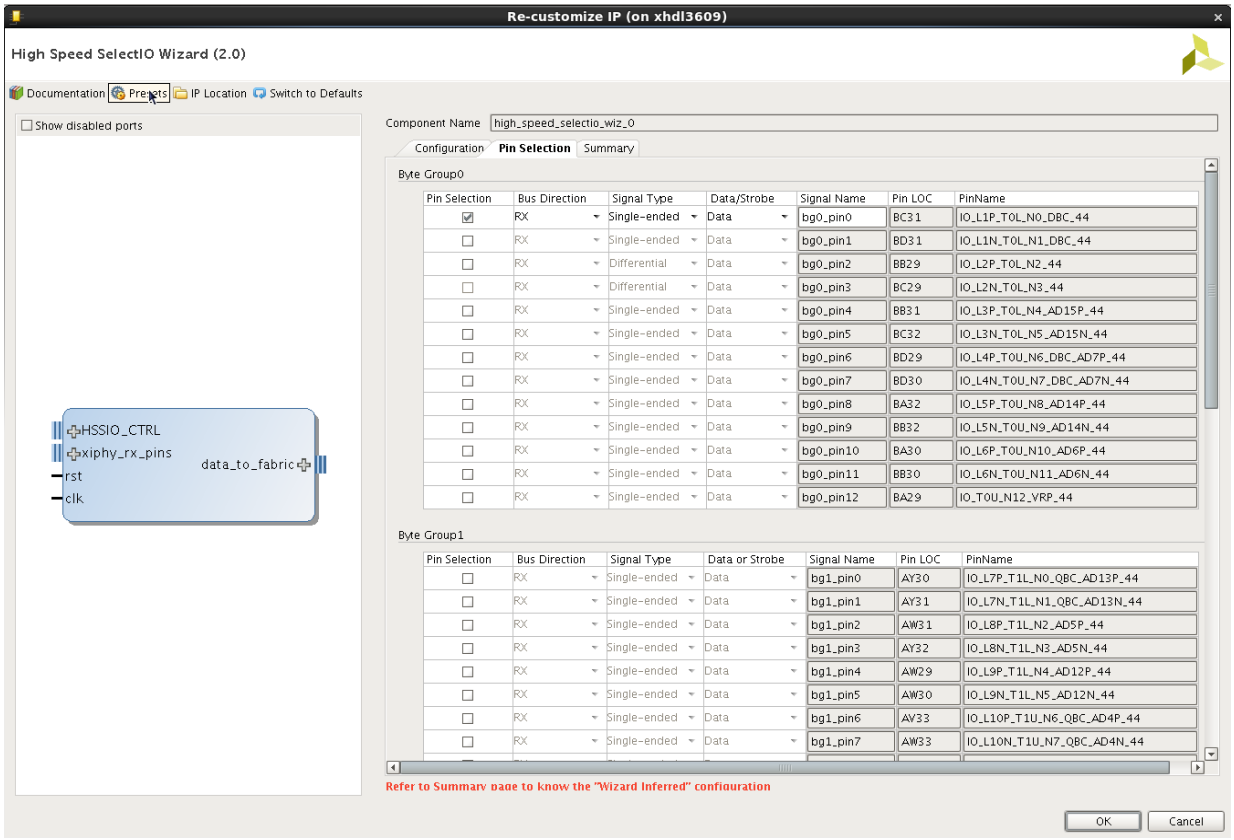


Figure 4-2: **Pin Selection Tab (Only Byte group0 and Byte group1 (partially) are shown)**

Byte Groupx (x is from 0 to 3). Byte group ordering is from Pin 0 to Pin 12 starting from Pin 0. The following description is applicable for all the byte groups unless specified.

Pin Selection

Enables the pin for selection. If P pin is differential, the N pin selection gets grayed out (Pin Name indicates whether pin is P or N, IO_L1P_T0L_). For example, Pin0 is differential, Pin 1 selection gets grayed out. Pin 12 is always a single-ended pin.

Pin Direction

Sets the bus direction on the selected pin. Available options are:

- TX: Pin direction is set to TX; option is not available when **RX only** option is selected for **Bus Direction** under the Configuration tab.
- RX: Pin direction is set to RX; option is not available when **TX only** option is selected for **Bus Direction** under the Configuration tab.
- BIDIR: Pin direction is set to Bidirectional; option is not available in the following cases.
  - **TX only** or **RX only** is selected for **Bus Direction** under the Configuration tab.
  - **ASYNC/NONE/FRACTIONAL** is selected for **RX External Clock and Data** under the Configuration tab.

  Note: In case of differential pins,

  - For the RX bus direction, both the P and N interface ports are available at the general interconnect and I/O interface.
  - For the BIDIR and TX bus directions, both the P and N interface ports are available at the I/O interface and only P ports are available at the general interconnect interface.

Signal Type

Sets the signal type to differential or single ended, If the P pin is differential. the N pin is grayed out and is set to differential (Pin Name indicates whether pin is P or N, IO_L1P_T0L_.).

Data/Strobe

Allows the pin to be selected as data/strobe/input clock/Clk Fwd, If the P pin is differential the N pin gets grayed out and is set to the same value as the P pin.

- Data: Sets the pin as a data pin.
- Strobe: Sets the pin as a strobe pin, Only DBC/QBC/GC_QBC pins can be selected as strobe. Pin Name indicates whether the pin is DBC/QBC/GC_QBC compatible. For example, IO_L1P_T0L_N0_DBC_44 indicates a DBC compatible pin. Only P pins can be set as a strobe.
- Clk Fwd: Valid only for the TX bus direction. Only one of the TX pins can be set as clock forward.
- Input Clock: Sets the pin as a clock pin. The input clock should be on this pin. Only GC/GC_QBC pins can be set as input clock pins. This option is available if **GC** is selected for **PLL ClocK Source** under the Configuration tab.

Signal Name

Allows you to set the signal name. The signal name is appended with the corresponding pin position of the selected pin in the bank. I/O ports appear with this name and for the general interconnect it is appended to the data ports.

For example, if the signal name is `bg1_pin0`, the I/O port appears as `bg1_pin0_13`, 13 is the pin position (starting from 0 from byte group0). For the TX general interconnect interface the data port is `data_from_fabric_bg0_pin0_13`. For RX, the interface data port is `data_to_fabric_bg1_pin0_13`.

Pin LOC

LOC constraint for the pin used for setting the constraints in the XDC. This is grayed out.

Pin Name

Indicates the pin name in the bank, this is grayed out. Provides information about the pin. For example,

- P or N pin
- DBC/QBC/GC_QBC/GC pin
- Pin position in the byte group
- Pin is in Upper/Lower nibble in the byte group

Summary Tab

This tab display the overall summary of the configuration.

- Number of TX/RX/RXTX pins
- General interconnect clock frequency
- ClockOut Phy Frequency from PLL
- Inferred bitslices

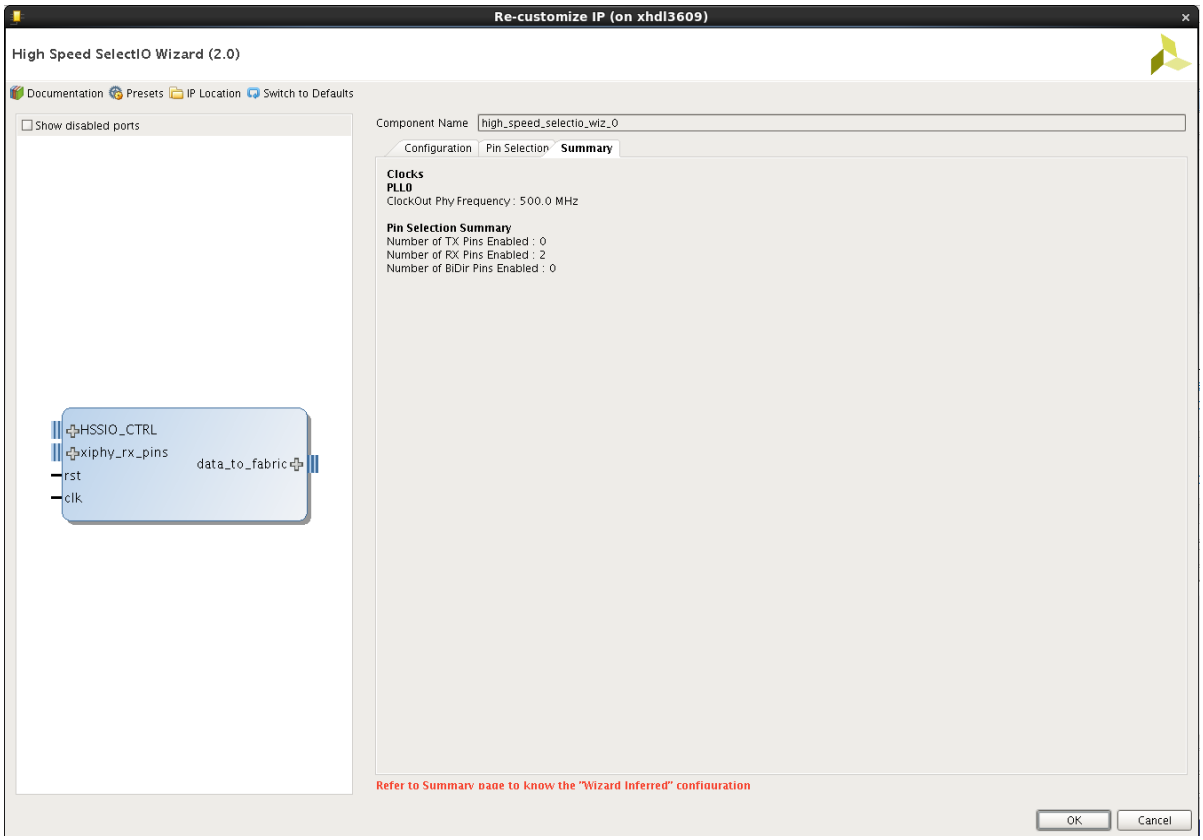


Figure 4-3: **Summary Tab**

.

User Parameters

Table 4-2 shows the relationship between the fields in the Vivado IDE and the User Parameters (which can be viewed in the Tcl Console).

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship**

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Bus Direction Range: ◦ TX_ONLY: 0 ◦ RX_ONLY: 1 ◦ BIDIR or TX+RX or TX+RX+BIDIR: 2	BUS_DIR	0
Interface Speed (Mb/s) Range: 300-1,600 MHz	PLL0_DATA_SPEED	600
PLL Clk Input Frequency MHz Range: 70-1,099 MHz	PLL0_INPUT_CLK_FREQ	156.00
RX External Clock and Data Range: ◦ Edge DDR: 0 ◦ Center DDR: 1 ◦ ASYNC/NONE/Fractional: 2 ◦ Center DDR Strobe: 3 ◦ Edge DDR Strobe: 4	PLL0_RX_EXTERNAL_CLK_TO_DATA	2
ENABLE FIFO WRITE CLKOUT Range: ◦ FALSE: 0 ◦ TRUE: 1	PLL0_FIFO_WRITE_CLK_OUT	0
PLL Clock Source Range: ◦ GC_Pin: IBUF_TO_PLL ◦ Fabric (Driven by BUFG): BUFG_TO_PLL	PLL0_CLK_SOURCE	BUFG_TO_PLL
BITSLICE Serialization Factor Range: 4,8	SERIALIZATION_FACTOR	8
RX Delay Cascade Range: ◦ FALSE: 0 ◦ TRUE: 1	RX_DELAY_CASCADE	0
RX Delay Type Range: ◦ FIXED: 0 ◦ VARIABLE: 1 ◦ VAR_LOAD: 2	RX_DELAY_TYPE	0

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
RX Delay Value Range: 0 to 1,250	RX_DELAY_VALUE	0
TX Delay Type Range: ◦ FIXED: 0 ◦ VARIABLE: 1 ◦ VAR_LOAD: 2	TX_DELAY_TYPE	0
TX Delay Value Range: 0 to 1.250	TX_DELAY_VALUE	0
Clock Forward Phase Range: ◦ FALSE: 0 ◦ TRUE: 1	CLK_FWD_PHASE	0
Bank Range: Varies with Bank	BANK	45
RIU Interface Range: ◦ FALSE: 0 ◦ TRUE: 1	ENABLE_RIU_INTERFACE	0
Enable BitSlip Range: ◦ FALSE: 0 ◦ TRUE: 1	ENABLE_BITSLIP	0
itslip Mode	BITSLIP_MODE	SLIP_PER_BIT
Data 3-state ◦ Serialized: 0 ◦ Combinatorial: 1	DATA_TRISTATE	1
Strobe 3-state ◦ Serialized: 0 ◦ Combinatorial: 1	CLOCK_TRISTATE	1
bus dir1 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN0_BUS_DIR	RX
bus dir2 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN1_BUS_DIR	RX

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
bus dir3 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN2_BUS_DIR	RX
bus dir4 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN3_BUS_DIR	RX
bus dir5 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN4_BUS_DIR	RX
bus dir6 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN5_BUS_DIR	RX
bus dir7 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN6_BUS_DIR	RX
bus dir8 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN7_BUS_DIR	RX
bus dir9 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN8_BUS_DIR	RX
bus dir10 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN9_BUS_DIR	RX

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
bus dir11 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN10_BUS_DIR	RX
bus dir12 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN11_BUS_DIR	RX
bus dir13 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE0_PIN12_BUS_DIR	RX
Data Strobe1 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN0_DATA_STROBE	Data
Data Strobe2 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN1_DATA_STROBE	Data
Data Strobe3 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN2_DATA_STROBE	Data
Data Strobe4 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN3_DATA_STROBE	Data
Data Strobe5 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN4_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe6 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN5_DATA_STROBE	Data
Data Strobe7 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN6_DATA_STROBE	Data
Data Strobe8 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN7_DATA_STROBE	Data
Data Strobe9 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN8_DATA_STROBE	Data
Data Strobe10 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN9_DATA_STROBE	Data
Data Strobe11 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN10_DATA_STROBE	Data
Data Strobe12 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN11_DATA_STROBE	Data
Data Strobe13 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE0_PIN12_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Bus Sig Type1 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN0_SIG_TYPE	Single-ended
Bus Sig Type2 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN1_SIG_TYPE	Single-ended
Bus Sig Type3 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN2_SIG_TYPE	Single-ended
Bus Sig Type4 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN3_SIG_TYPE	Single-ended
Bus Sig Type5 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN4_SIG_TYPE	Single-ended
Bus Sig Type6 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN5_SIG_TYPE	Single-ended
Bus Sig Type7 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN6_SIG_TYPE	Single-ended
Bus Sig Type8 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN7_SIG_TYPE	Single-ended
Bus Sig Type9 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN8_SIG_TYPE	Single-ended
Bus Sig Type10 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN9_SIG_TYPE	Single-ended

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Bus Sig Type11 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN10_SIG_TYPE	Single-ended
Bus Sig Type12 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN11_SIG_TYPE	Single-ended
Bus Sig Type13 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE0_PIN12_SIG_TYPE	Single-ended
Byte Group0 Pin Selection1 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN0	FALSE
Byte Group0 Pin Selection2 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN1	FALSE
Byte Group0 Pin Selection3 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN2	FALSE
Byte Group0 Pin Selection4 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN3	FALSE
Byte Group0 Pin Selection5 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN4	FALSE
Byte Group0 Pin Selection6 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN5	FALSE
Byte Group0 Pin Selection7 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN6	FALSE

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte Group0 Pin Selection8 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN7	FALSE
Byte Group0 Pin Selection9 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN8	FALSE
Byte Group0 Pin Selection10 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN9	FALSE
Byte Group0 Pin Selection11 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN10	FALSE
Byte Group0 Pin Selection12 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN11	FALSE
Byte Group0 Pin Selection13 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE0_PIN12	FALSE
Byte Group0 Signal Name1	BYTE0_PIN0_SIGNAL_NAME	bg0_pin0
Byte Group0 Signal Name2	BYTE0_PIN1_SIGNAL_NAME	bg0_pin1
Byte Group0 Signal Name3	BYTE0_PIN2_SIGNAL_NAME	bg0_pin2
Byte Group0 Signal Name4	BYTE0_PIN3_SIGNAL_NAME	bg0_pin3
Byte Group0 Signal Name5	BYTE0_PIN4_SIGNAL_NAME	bg0_pin4
Byte Group0 Signal Name6	BYTE0_PIN5_SIGNAL_NAME	bg0_pin5
Byte Group0 Signal Name7	BYTE0_PIN6_SIGNAL_NAME	bg0_pin6
Byte Group0 Signal Name8	BYTE0_PIN7_SIGNAL_NAME	bg0_pin7
Byte Group0 Signal Name9	BYTE0_PIN8_SIGNAL_NAME	bg0_pin8
Byte Group0 Signal Name10	BYTE0_PIN9_SIGNAL_NAME	bg0_pin9
Byte Group0 Signal Name11	BYTE0_PIN10_SIGNAL_NAME	bg0_pin10
Byte Group0 Signal Name12	BYTE0_PIN11_SIGNAL_NAME	bg0_pin11
Byte Group0 Signal Name13	BYTE0_PIN12_SIGNAL_NAME	bg0_pin12

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
bus dir1 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN0_BUS_DIR	RX
bus dir2 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN1_BUS_DIR	RX
bus dir3 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN2_BUS_DIR	RX
bus dir4 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN3_BUS_DIR	RX
bus dir5 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN4_BUS_DIR	RX
bus dir6 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN5_BUS_DIR	RX
bus dir7 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN6_BUS_DIR	RX
bus dir8 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN7_BUS_DIR	RX

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
bus dir9 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN8_BUS_DIR	RX
bus dir10 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN9_BUS_DIR	RX
bus dir11 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN10_BUS_DIR	RX
bus dir12 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN11_BUS_DIR	RX
bus dir13 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE1_PIN12_BUS_DIR	RX
Data Strobe1 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN0_DATA_STROBE	Data
Data Strobe2 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN1_DATA_STROBE	Data
Data Strobe3 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN2_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe4 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN3_DATA_STROBE	Data
Data Strobe5 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN4_DATA_STROBE	Data
Data Strobe6 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN5_DATA_STROBE	Data
Data Strobe7 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN6_DATA_STROBE	Data
Data Strobe8 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd ◦ Input Clock: Input Clock	BYTE1_PIN7_DATA_STROBE	Data
Data Strobe9 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN8_DATA_STROBE	Data
Data Strobe10 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN9_DATA_STROBE	Data
Data Strobe11 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd ◦ Input Clock: Input Clock	BYTE1_PIN10_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe12 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN11_DATA_STROBE	Data
Data Strobe13 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE1_PIN12_DATA_STROBE	Data
Bus Sig Type1 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN0_SIG_TYPE	Single-ended
Bus Sig Type2 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN1_SIG_TYPE	Single-ended
Bus Sig Type3 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN2_SIG_TYPE	Single-ended
Bus Sig Type4 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN3_SIG_TYPE	Single-ended
Bus Sig Type5 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN4_SIG_TYPE	Single-ended
Bus Sig Type6 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN5_SIG_TYPE	Single-ended
Bus Sig Type7 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN6_SIG_TYPE	Single-ended
Bus Sig Type8 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN7_SIG_TYPE	Single-ended

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Bus Sig Type9 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN8_SIG_TYPE	Single-ended
Bus Sig Type10 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN9_SIG_TYPE	Single-ended
Bus Sig Type11 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN10_SIG_TYPE	Single-ended
Bus Sig Type12 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN11_SIG_TYPE	Single-ended
Bus Sig Type13 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE1_PIN12_SIG_TYPE	Single-ended
Byte group1 Pin Selection1 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN0	FALSE
Byte group1 Pin Selection2 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN1	FALSE
Byte group1 Pin Selection3 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN2	FALSE
Byte group1 Pin Selection4 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN3	FALSE
Byte group1 Pin Selection5 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN4	FALSE

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte group1 Pin Selection6 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN5	FALSE
Byte group1 Pin Selection7 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN6	FALSE
Byte group1 Pin Selection8 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN7	FALSE
Byte group1 Pin Selection9 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN8	FALSE
Byte group1 Pin Selection10 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN9	FALSE
Byte group1 Pin Selection11 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN10	FALSE
Byte group1 Pin Selection12 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN11	FALSE
Byte group1 Pin Selection13 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE1_PIN12	FALSE
Byte Group1 Signal Name1	BYTE1_PIN0_SIGNAL_NAME	bg1_pin0
Byte Group1 Signal Name2	BYTE1_PIN1_SIGNAL_NAME	bg1_pin1
Byte Group1 Signal Name3	BYTE1_PIN2_SIGNAL_NAME	bg1_pin2
Byte Group1 Signal Name4	BYTE1_PIN3_SIGNAL_NAME	bg1_pin3
Byte Group1 Signal Name5	BYTE1_PIN4_SIGNAL_NAME	bg1_pin4
Byte Group1 Signal Name6	BYTE1_PIN5_SIGNAL_NAME	bg1_pin5
Byte Group1 Signal Name7	BYTE1_PIN6_SIGNAL_NAME	bg1_pin6
Byte Group1 Signal Name8	BYTE1_PIN7_SIGNAL_NAME	bg1_pin7

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte Group1 Signal Name9	BYTE1_PIN8_SIGNAL_NAME	bg1_pin8
Byte Group1 Signal Name10	BYTE1_PIN9_SIGNAL_NAME	bg1_pin9
Byte Group1 Signal Name11	BYTE1_PIN10_SIGNAL_NAME	bg1_pin10
Byte Group1 Signal Name12	BYTE1_PIN11_SIGNAL_NAME	bg1_pin11
Byte Group1 Signal Name13	BYTE1_PIN12_SIGNAL_NAME	bg1_pin12
bus dir1 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN0_BUS_DIR	RX
bus dir2 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN1_BUS_DIR	RX
bus dir3 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN2_BUS_DIR	RX
bus dir4 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN3_BUS_DIR	RX
bus dir5 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN4_BUS_DIR	RX
bus dir6 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN5_BUS_DIR	RX
bus dir7 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN6_BUS_DIR	RX

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
bus dir8 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN7_BUS_DIR	RX
bus dir9 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN8_BUS_DIR	RX
bus dir10 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN9_BUS_DIR	RX
bus dir11 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN10_BUS_DIR	RX
bus dir12 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN11_BUS_DIR	RX
bus dir13 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE2_PIN12_BUS_DIR	RX
Data Strobe1 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd ◦ Input Clock: Input Clock	BYTE2_PIN0_DATA_STROBE	Data
Data Strobe2 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN1_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe3 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd ◦ Input Clock: Input Clock	BYTE2_PIN2_DATA_STROBE	Data
Data Strobe4 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN3_DATA_STROBE	Data
Data Strobe5 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN4_DATA_STROBE	Data
Data Strobe6 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN5_DATA_STROBE	Data
Data Strobe7 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN6_DATA_STROBE	Data
Data Strobe8 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN7_DATA_STROBE	Data
Data Strobe9 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN8_DATA_STROBE	Data
Data Strobe10 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN9_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe11 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN10_DATA_STROBE	Data
Data Strobe12 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN11_DATA_STROBE	Data
Data Strobe13 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE2_PIN12_DATA_STROBE	Data
Bus Sig Type1 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN0_SIG_TYPE	Single-ended
Bus Sig Type2 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN1_SIG_TYPE	Single-ended
Bus Sig Type3 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN2_SIG_TYPE	Single-ended
Bus Sig Type4 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN3_SIG_TYPE	Single-ended
Bus Sig Type5 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN4_SIG_TYPE	Single-ended
Bus Sig Type6 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN5_SIG_TYPE	Single-ended

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Bus Sig Type7 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN6_SIG_TYPE	Single-ended
Bus Sig Type8 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN7_SIG_TYPE	Single-ended
Bus Sig Type9 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN8_SIG_TYPE	Single-ended
Bus Sig Type10 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN9_SIG_TYPE	Single-ended
Bus Sig Type11 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN10_SIG_TYPE	Single-ended
Bus Sig Type12 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN11_SIG_TYPE	Single-ended
Bus Sig Type13 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE2_PIN12_SIG_TYPE	Single-ended
Byte group2 Pin Selection1 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN0	FALSE
Byte group2 Pin Selection2 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN1	FALSE
Byte group2 Pin Selection3 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN2	FALSE

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte group2 Pin Selection4 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN3	FALSE
Byte group2 Pin Selection5 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN4	FALSE
Byte group2 Pin Selection6 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN5	FALSE
Byte group2 Pin Selection7 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN6	FALSE
Byte group2 Pin Selection8 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN7	FALSE
Byte group2 Pin Selection9 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN8	FALSE
Byte group2 Pin Selection10 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN9	FALSE
Byte group2 Pin Selection11 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN10	FALSE
Byte group2 Pin Selection12 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN11	FALSE
Byte group2 Pin Selection13 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE2_PIN12	FALSE
Byte Group2 Signal Name1	BYTE2_PIN0_SIGNAL_NAME	bg2_pin0

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte Group2 Signal Name2	BYTE2_PIN1_SIGNAL_NAME	bg2_pin1
Byte Group2 Signal Name3	BYTE2_PIN2_SIGNAL_NAME	bg2_pin2
Byte Group2 Signal Name4	BYTE2_PIN3_SIGNAL_NAME	bg2_pin3
Byte Group2 Signal Name5	BYTE2_PIN4_SIGNAL_NAME	bg2_pin4
Byte Group2 Signal Name6	BYTE2_PIN5_SIGNAL_NAME	bg2_pin5
Byte Group2 Signal Name7	BYTE2_PIN6_SIGNAL_NAME	bg2_pin6
Byte Group2 Signal Name8	BYTE2_PIN7_SIGNAL_NAME	bg2_pin7
Byte Group2 Signal Name9	BYTE2_PIN8_SIGNAL_NAME	bg2_pin8
Byte Group2 Signal Name10	BYTE2_PIN9_SIGNAL_NAME	bg2_pin9
Byte Group2 Signal Name11	BYTE2_PIN10_SIGNAL_NAME	bg2_pin10
Byte Group2 Signal Name12	BYTE2_PIN11_SIGNAL_NAME	bg2_pin11
Byte Group2 Signal Name13	BYTE2_PIN12_SIGNAL_NAME	bg2_pin12
bus dir1 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN0_BUS_DIR	RX
bus dir2 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN1_BUS_DIR	RX
bus dir3 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN2_BUS_DIR	RX
bus dir4 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN3_BUS_DIR	RX
bus dir5 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN4_BUS_DIR	RX

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
bus dir6 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN5_BUS_DIR	RX
bus dir7 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN6_BUS_DIR	RX
bus dir8 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN7_BUS_DIR	RX
bus dir9 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN8_BUS_DIR	RX
bus dir10 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN9_BUS_DIR	RX
bus dir11 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN10_BUS_DIR	RX
bus dir12 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN11_BUS_DIR	RX
bus dir13 Range: ◦ TX ◦ RX ◦ BIDIR	BYTE3_PIN12_BUS_DIR	RX

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe1 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN0_DATA_STROBE	Data
Data Strobe2 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN1_DATA_STROBE	Data
Data Strobe3 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN2_DATA_STROBE	Data
Data Strobe4 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN3_DATA_STROBE	Data
Data Strobe5 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN4_DATA_STROBE	Data
Data Strobe6 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN5_DATA_STROBE	Data
Data Strobe7 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN6_DATA_STROBE	Data
Data Strobe8 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN7_DATA_STROBE	Data

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Data Strobe9 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN8_DATA_STROBE	Data
Data Strobe10 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN9_DATA_STROBE	Data
Data Strobe11 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN10_DATA_STROBE	Data
Data Strobe12 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN11_DATA_STROBE	Data
Data Strobe13 Range: ◦ Data ◦ Strobe ◦ Clk Fwd: Clk Fwd	BYTE3_PIN12_DATA_STROBE	Data
Bus Sig Type1 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN0_SIG_TYPE	Single-ended
Bus Sig Type2 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN1_SIG_TYPE	Single-ended
Bus Sig Type3 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN2_SIG_TYPE	Single-ended
Bus Sig Type4 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN3_SIG_TYPE	Single-ended

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Bus Sig Type5 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN4_SIG_TYPE	Single-ended
Bus Sig Type6 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN5_SIG_TYPE	Single-ended
Bus Sig Type7 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN6_SIG_TYPE	Single-ended
Bus Sig Type8 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN7_SIG_TYPE	Single-ended
Bus Sig Type9 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN8_SIG_TYPE	Single-ended
Bus Sig Type10 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN9_SIG_TYPE	Single-ended
Bus Sig Type11 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN10_SIG_TYPE	Single-ended
Bus Sig Type12 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN11_SIG_TYPE	Single-ended
Bus Sig Type13 Range: ◦ Differential: DIFF ◦ Single-ended: SINGLE	BYTE3_PIN12_SIG_TYPE	Single-ended
Byte group3 Pin Selection1 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN0	FALSE

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte group3 Pin Selection2 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN1	FALSE
Byte group3 Pin Selection3 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN2	FALSE
Byte group3 Pin Selection4 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN3	FALSE
Byte group3 Pin Selection5 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN4	FALSE
Byte group3 Pin Selection6 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN5	FALSE
Byte group3 Pin Selection7 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN6	FALSE
Byte group3 Pin Selection8 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN7	FALSE
Byte group3 Pin Selection9 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN8	FALSE
Byte group3 Pin Selection10 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN9	FALSE
Byte group3 Pin Selection11 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN10	FALSE

Table 4-2: **Vivado IDE Parameter to User Parameter Relationship *(Cont'd)***

Vivado IDE Parameter/Value(1)	User Parameter/Value(1)	Default Value
Byte group3 Pin Selection12 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN11	FALSE
Byte group3 Pin Selection13 Range: ◦ TRUE ◦ FALSE	ENABLE_BYTE3_PIN12	FALSE
Byte Group3 Signal Name1	BYTE3_PIN0_SIGNAL_NAME	bg3_pin0
Byte Group3 Signal Name2	BYTE3_PIN1_SIGNAL_NAME	bg3_pin1
Byte Group3 Signal Name3	BYTE3_PIN2_SIGNAL_NAME	bg3_pin2
Byte Group3 Signal Name4	BYTE3_PIN3_SIGNAL_NAME	bg3_pin3
Byte Group3 Signal Name5	BYTE3_PIN4_SIGNAL_NAME	bg3_pin4
Byte Group3 Signal Name6	BYTE3_PIN5_SIGNAL_NAME	bg3_pin5
Byte Group3 Signal Name7	BYTE3_PIN6_SIGNAL_NAME	bg3_pin6
Byte Group3 Signal Name8	BYTE3_PIN7_SIGNAL_NAME	bg3_pin7
Byte Group3 Signal Name9	BYTE3_PIN8_SIGNAL_NAME	bg3_pin8
Byte Group3 Signal Name10	BYTE3_PIN9_SIGNAL_NAME	bg3_pin9
Byte Group3 Signal Name11	BYTE3_PIN10_SIGNAL_NAME	bg3_pin10
Byte Group3 Signal Name12	BYTE3_PIN11_SIGNAL_NAME	bg3_pin11
Byte Group3 Signal Name13	BYTE3_PIN12_SIGNAL_NAME	bg3_pin12

Output Generation

The core delivers Verilog RTL for the core logic, example design, and example test bench.

The following files are created when core is configured and output products are generated:

<ComponentName>.v - Top Level wrapper to be instantiated in User Design

- <ComponentName>_high_speed_selectio_wiz_v2_0.v - Top Level file to set the appropriate HDL parameters for a given configuration
  - hssio_wiz_top.v - Top level file
    - bs_top.v - Top level file for Bitslice and Bitslice Control Modules
      - rxtx_bs.v - Instantiates RXTX_BITSLICE module
      - tx_bs_tri.v - Instantiates TX_BITSLICE_TRI module

- tx_bs.v - Instantiates TX_BITSLICE module
- rx_bs.v - Instantiate RX_BITSLICE module
- BitSlipInLogic_Toplevel.v (Optional Module for BitSlip Operation)
- C2BCEtc.v
- C2BCEtc_dwnld.v
- C3BCEtc.v
- C3BCEtc_dwnld.v
- Fdcr.v
- GenPulse.v
- BitSlipInLogic_4b.v
- BitSlipInLogic_8b.v
- BitSlipInLogic_FstCmp_4b.v
- BitSlipInLogic_FstCmp_8b.v
- clk_rst_top.v - Instantiates Clock and Reset Modules
- rst_scheme.v - Reset Logic
- clk_scheme.v - Clocking Logic - Instantiates PLL
- bs_ctrl_top.v - Instantiates BITSLICE_CONTROL module
- iobuf_top.v - Instantiates I/O Buffers for all Clock and Data Pins

If the IP example design project is opened, another core instance with the core name <ip_ex_inst> is instantiated in the `<ComponentName>_exdes.v`. For example design simulation, the `<ComponentName>_tb.v` test bench file is generated.

Constraining the Core

This section contains information about constraining the core in the Vivado Design Suite.

Required Constraints

Refer to the I/O Standard Settings in Configuration Settings.

Device, Package, and Speed Grade Selections

Input clock frequency selection depends on the maximum frequencies supported by the BUFG. Select the device, package and speed grades after referring to the *Kintex UltraScale Architecture Data Sheet* (DS892) [Ref 2] for details on supported maximum frequencies.

Clock Frequencies

The I/O logic for this core works in the range 300-1,600 MHz. The general interconnect logic for this core works up to 200 MHz.

Clock Management

All clocks are generated using PLL.

IMPORTANT: *Core logic should be clocked with the divided version of the PLL output.*

Clock Placement

The input clock can be placed on global clock pins with the name *_GC_*. For EDGE DDR/ CENTER DDR, the clock is placed on *_GC_QBC* pin only.

Banking

This core can be used to configure an I/O circuit for High Range (HR) and High Performance (HP) banks. Available banks for the project part are provided in the Vivado IDE for selection. See Customizing and Generating the Core for more details.

Transceiver Placement

There are no transceiver placement constraints for this core.

I/O Standard and Placement

Steps to set I/O standards in the constraints file

1. After customization, open the XDC file. All the differential and single-ended I/O standards supported for the Bank are listed towards the end of this file.

   There are also two variables listed at the end of the file. These are commented.

   ```
   #set diff_std
   #set sio_std
   ```

2. Uncomment the preceding lines depending on the standard that you are using. Set the desired standard against the variable. For example, if using only a single-ended I/O standard in the configuration and the standard is LVCMOS18, uncomment the second line.

   ```
   #set diff_std
   set sio_std LVCMOS18
   ```

3. Uncomment the I/O standard constraint for each pin

   Example:

   ```
   #set_property IOSTANDARD $sio_std [get_ports bg0_pin6_6]
   #set_property IOSTANDARD $diff_std [get_ports bg0_pin6_6]
   ```

   If a single-ended I/O standard is needed, uncomment the first or if a differential I/O standard is needed, uncomment the second one. `bg0_pin6` is the signal name of the pin given in the Vivado IDE and _6 is the pin number in the bank.

   Note: The Wizard can infer some bitslices for strobe propagation in the RX and Bidir bus configuration; these are displayed under the comment.

   ############ I/O constraints for Wizard inferred RX bitslice ######################

   For these bitslices, only the single-ended I/O standard is applicable. In case only differential pins are selected during configuration, ensure that the appropriate single ended I/O standard is set in the XDC file for inferred RX bit slice.

Refer to *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571) [Ref 1] for the supported I/O standards.

Simulation

For comprehensive information about Vivado simulation components, as well as information about using supported third-party tools, see the *Vivado Design Suite User Guide: Logic Simulation* (UG900) [Ref 6].

Synthesis and Implementation

For details about synthesis and implementation, see the *Vivado Design Suite User Guide: Designing with IP* (UG896) [Ref 4].

Example Design

This chapter contains information about the example design provided in the Vivado® Design Suite.

This core provides an example design with one core instance and one example instance. For each configured pin of the IP instance, a counter part (example design) is generated. When bitslip is enabled, the known pattern `8'h2C` or `4'hC` is compared to align the data at RX. When data is aligned, 0xA5 and 0x5A patterns are sent for 8-bit serialization, and 0x5 and 0xA patterns are sent for 4-bit serialization per bitslice. When bitslip is not enabled, data from RX bitslices are unaligned data, and the example design checks for all possible valid data of RX for a known TX data pattern. If a pattern matches, `data_check_complete` output is asserted from the example design.

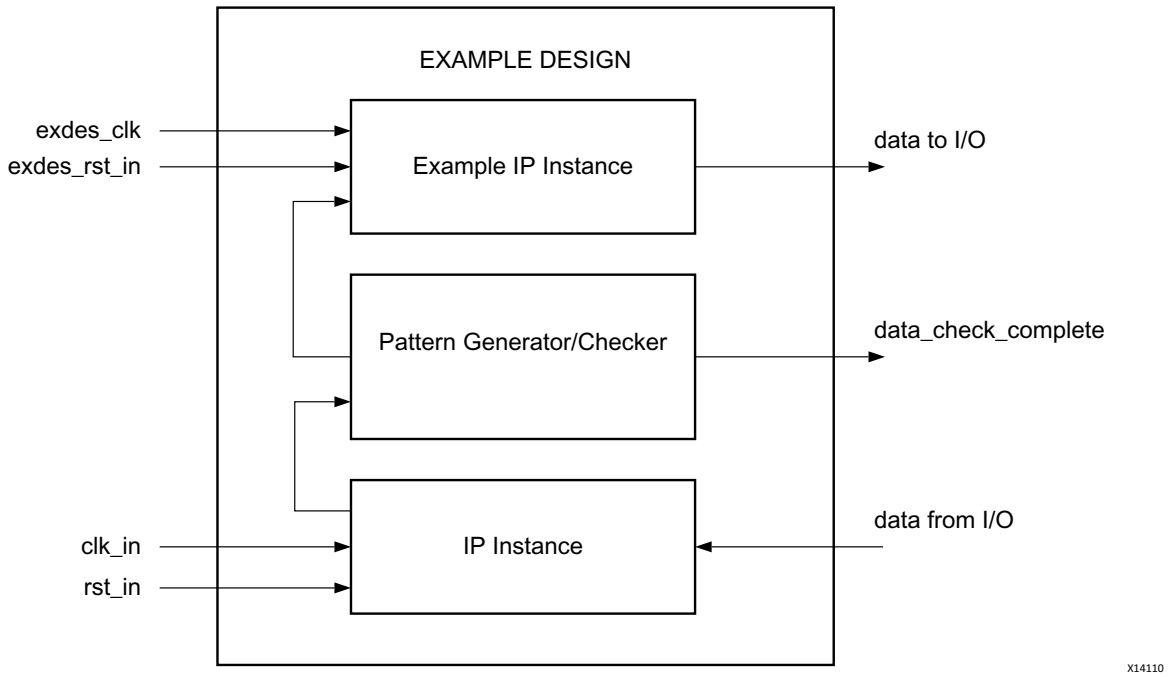


Figure 5-1: **Example Design Block Diagram**

Table 5-1: **DUT and Partner IP Configurations**

DUT Configuration	Partner IP Configuration
TX with no Clock Forward Pin	RX in Serial Mode
TX with Clock Forward Pin and Clock Forward Phase= 0	RX in Edge DDR mode
TX with Clock Forward Pin and Clock Forward Phase= 90	RX in Center DDR mode
RX – Edge DDR/ Center DDR	TX with Clock Forward Pin
RX – Edge DDR Strobe	TX with Clock Forward Pin and Clock Forward Phase = 0
RX – Center DDR Strobe	TX with Clock Forward Pin and Clock Forward Phase = 90
Mix of TX, RX and BiDir	Not supported
Tx with no clock forward pin and interface speed less than 600 Mb/s	Not supported

Test Bench

This chapter contains information about the test bench provided in the Vivado® Design Suite. The test bench is a simple Verilog code to exercise the example design and the core. This test bench performs the following tasks:

- Generates the input clock signals.
- Applies a reset to the example design.
- Example design RX and TX interfaces are looped back; similarly IP BIDIR bus and example design BIDIR buses are looped back. The waveform is shown for the TX/RX loopback for 1 pin.
- If an RX and TX pattern matches, the test bench sends a message for the successful test completion, as shown in Figure 6-1. Otherwise, it waits for 16,000 cycles of input clock and sends a test failure message.

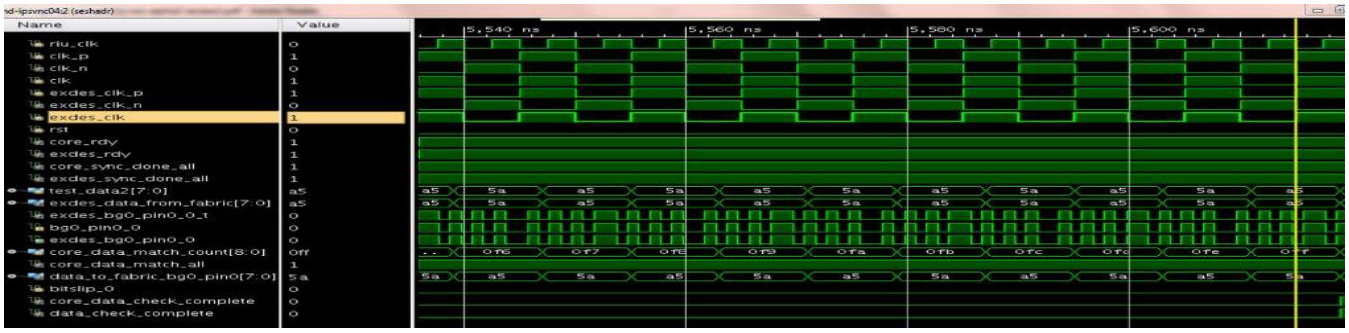


Figure 6-1: **Test Bench Waveform**

Appendix A

Verification, Compliance, and Interoperability

This appendix provides details about how this IP core was tested for compliance with the protocol to which it was designed.

Simulation

This core is verified with incisive enterprise simulator (IES), VCS, Questa and XSIM simulators.

Hardware Testing

Hardware testing was performed on the KCU105 platform for the RX and TX configurations using the MicroBlaze™ processor system to generate the data pattern for the TX and check the data at the RX. The TX and RX pins were connected to an FMC 107 loopback card placed on an HPC connector onboard. Tests were performed on the supported data speed range.

Migrating and Upgrading

Migrating and upgrading are not possible from previous versions of the core. All the port names and Vivado® IDE parameters were changed in v2.0.

Debugging

This appendix includes details about resources available on the Xilinx Support website and debugging tools.

Finding Help on Xilinx.com

To help in the design and debug process when using the High Speed SelectIO Wizard, the Xilinx Support web page contains key resources such as product documentation, release notes, answer records, information about known issues, and links for obtaining further product support.

Documentation

This product guide is the main document associated with the High Speed SelectIO Wizard. This guide, along with documentation related to all products that aid in the design process, can be found on the Xilinx Support web page or by using the Xilinx® Documentation Navigator.

Download the Xilinx Documentation Navigator from the Downloads page. For more information about this tool and the features available, open the online help after installation.

Answer Records

Answer Records include information about commonly encountered problems, helpful information on how to resolve these problems, and any known issues with a Xilinx product. Answer Records are created and maintained daily ensuring that users have access to the most accurate information available.

Answer Records for this core can be located by using the Search Support box on the main Xilinx support web page. To maximize your search results, use proper keywords such as

- Product name
- Tool message(s)
- Summary of the issue encountered

Master Answer Record: 60295

Technical Support

Xilinx provides technical support in the Xilinx Support web page for this LogiCORE™ IP product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support if you do any of the following:

- Implement the solution in devices that are not defined in the documentation.
- Customize the solution beyond that allowed in the product documentation.
- Change any section of the design labeled DO NOT MODIFY.

To contact Xilinx Technical Support, navigate to the Xilinx Support web page.

Vivado Design Suite Debug Feature

The Vivado® Design Suite debug feature inserts logic analyzer and virtual I/O cores directly into your design. The debug feature also allows you to set trigger conditions to capture application and integrated block port signals in hardware. Captured signals can then be analyzed. This feature in the Vivado IDE is used for logic debugging and validation of a design running in Xilinx.

The Vivado logic analyzer is used with the logic debug IP cores, including:

- integrated logic analyzer (ILA) 2.0 (and later versions)
- virtual input output (VIO) 2.0 (and later versions)

See the *Vivado Design Suite User Guide: Programming and Debugging* (UG908) [Ref 8].

Reference Boards

Various Xilinx development boards support the High Speed SelectIO Wizard. These boards can be used to prototype designs and establish that the core can communicate with the system. The UltraScale™ architecture evaluation board KCU105 is supported.

Hardware Debug

Hardware issues can range from link bring-up to problems seen after hours of testing. This section provides debug steps for common issues. The debug feature is a valuable resource to use in hardware debug. The signal names mentioned in the following individual sections can be probed using the Vivado debug feature for debugging the specific problems.

Ensure that all the timing constraints for the core were properly incorporated from the example design and that all constraints were met during implementation.

- Does it work in post-place and route timing simulation? If problems are seen in hardware but not in timing simulation, this might indicate a PCB issue. Ensure that all clock sources are active and clean.
- Ensure that the `rst_seq_done` is asserted by the wizard.
- Does the `rst_seq_done` signal toggle without the assertion of reset? If yes, this indicates a loss of PLL lock or deassertion of the `DLY_RDY` and `VTC_RDY` signals.
- Ensure that the PLL Lock signal is asserted.
- Ensure that the `VTC_RDY` and `DLY_RDY` signals are asserted for all BITSLICE and BITSLICE_CONTROL signals.
- Ensure that the synthesis and implementation of the design is successful without any DRC errors.
- Check that RIU Clock is always present in the system during the deassertion of reset.

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

References

These documents provide supplemental material useful with this product guide:

1. *UltraScale Architecture SelectIO Resources: Advance Specification User Guide* (UG571)
2. *Kintex UltraScale Architecture Data Sheet* (DS892)
3. *Vivado Design Suite User Guide: Designing IP Subsystems using IP Integrator* (UG994)
4. *Vivado Design Suite User Guide: Designing with IP* (UG896)
5. *Vivado Design Suite User Guide: Getting Started* (UG910)
6. *Vivado Design Suite User Guide: Logic Simulation* (UG900)
7. *Bitslip in Logic Application Note* (XAPP1208)
8. *Vivado Design Suite User Guide: Programming and Debugging* (UG908)
9. *Vivado Design Suite User Guide - Implementation* (UG904)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
09/30/2015	2.0	IP Facts table • Added a link to the Performance and Resource Utilization web page. Chapter 1, Overview • Updated Features section on the IP Facts page. • Updated Feature Summary, Applications, and Unsupported Features sections. Chapter 2, Product Specification • Updated Figure 2-1. • Updated Performance section. • Added a link to the Performance and Resource Utilization web page. • Removed all of the device-specific information in the Resource Utilization section. • Completely revised the Port Descriptions section. Replaced Table 2-2 and 2-3 with new tables. Chapter 3, Designing with the Core • Completely revised General Design Guidelines and Clocking sections. • Added Resets section. • Updated Figure 3-1. Chapter 4, Design Flow Steps • Completely revised Customizing and Generating the Core section. • Updated all illustrations and their descriptions. • Replaced Table 4-1 with new table. • Added a new figure to the Output Generation section. • Updated most of the subsections in the Constraining the Core section. Chapter 5, Test Bench Replaced Figure 6-1 with a new waveform.
04/01/2015	1.1	• Added support for bidirectional buses. • Added User Parameters section.
10/01/2014	1.1	• Added support for Bitslip modes. • Removed the if<k>_ext_clk_to_fabric port.
04/02/2014	1.0	Initial Xilinx release.

Please Read: Important Legal Notices