3. Configuring Multiple Protocols and Data Rates in Arria II Devices

AIIGX52003-3.0

This chapter describes the configuration of multiple protocols and data rates for Arria® II GX and GZ devices. Each transceiver channel in an Arria II GX or GZ device can run at an independent data rate or protocol mode. Within each transceiver channel, the transmitter and receiver channel can run at different data rates. Each transceiver block consists of two clock multiplier unit (CMU) phase-locked loops (PLLs) that provide clocks to all the transmitter channels within the transceiver block. Each receiver channel contains a dedicated clock data recovery (CDR).

This chapter includes the following sections:

Transceiver PLL Configurations

You can configure each transmitter channel to use one of the two CMU PLLs in the transceiver block. In addition, each transmitter channel has a local divider (/1, /2, or /4) that divides the clock output of the CMU PLL to provide high-speed serial and low-speed parallel clocks for its physical coding sublayer (PCS) and physical medium attachment (PMA) functional blocks.

You can configure the RX CDR present in the receiver channel to a distinct data rate and provide separate input reference clocks. Each receiver channel also contains a local divider that divides the high-speed clock output of the RX CDR and provides clocks for its PCS and PMA functional blocks. To enable transceiver channel settings, the Quartus® II software provides the ALTGX MegaWizard™ Plug-In Manager interface. The ALTGX MegaWizard Plug-In Manager allows you to instantiate a single transceiver channel or multiple transceiver channels in **Receiver and Transmitter**, **Receiver Only**, and **Transmitter Only** configurations.

Creating Transceiver Channel Instances

You can instantiate multiple transceiver channels in the **General** screen of the ALTGX MegaWizard Plug-In Manager in the following two different ways:

- For the **What is the number of channels?** option, select the required value. This method creates the transceiver channels with identical configurations. For examples, refer to "Combining Transceiver Instances in Multiple Transceiver Blocks" on page 3–13.
- For the **What is the number of channels?** option, select **1** and create a single channel transceiver instance. To instantiate additional transceiver channels with an identical configuration, stamp the created ALTGX instance multiple times. If you require additional transceiver channels with different configurations, create separate ALTGX megafunction instances with different settings and use them in your design.

When you create instances using the above methods, you can force the placement of up to four transceiver channels within the same transceiver block by assigning the `tx_dataout` and `rx_datain` ports of the channel instances to a single transceiver bank. If you do not assign pins to the `tx_dataout` and `rx_datain` ports, the Quartus II software chooses default pin assignments. When you compile the design, the Quartus II software combines multiple channel instances within the same transceiver block if the instances meet specific requirements. The following sections describe these requirements for different transceiver configurations.

General Requirements to Combine Channels

When you create multiple ALTGX instances, the Quartus II software requires that you set identical values on the following parameters and signals to combine the ALTGX instances within the same transceiver block or in the transceiver blocks on the same side of the device. The following sections describe these requirements.

Control Signals

The `gxb_powerdown` port is an optional port that you can enable in the ALTGX MegaWizard Plug-In Manager. If enabled, you must drive the `gxb_powerdown` port in the ALTGX instances from the same logic or the same input pin to enable the Quartus II software to assign them in the same transceiver block. If you disable the `gxb_powerdown` port, the Quartus II software ties the port to ground.

Calibration Clock and Power Down

Each calibration block in an Arria II GX or GZ device is shared by multiple transceiver blocks.

If your design uses multiple transceiver blocks, depending on the transceiver banks selected, you must connect the `cal_blk_clk` and `cal_blk_powerdown` ports of all channel instances to the same input pin or logic.

For more information about the calibration block and transceiver banks that are connected to a specific calibration block, refer to the "Calibration Block" section in the *Transceiver Architecture in Arria II Devices* chapter.

Asserting the cal_blk_powerdown port affects the calibration circuit on all transceiver channels connected to the calibration block.

Sharing CMU PLLs

Each Arria II GX and GZ transceiver block contains two CMU PLLs. When you create multiple transceiver channel instances and intend to combine them in the same transceiver block, the Quartus II software checks whether a single CMU PLL can be used to provide clock outputs for the transmitter side of the channel instances. If a single CMU PLL is not sufficient, the Quartus II software attempts to combine the channel instances using two CMU PLLs. Otherwise, the Quartus II software issues a Fitter error.

The following two sections describe the ALTGX instance requirements to enable the Quartus II software to share the CMU PLL.

Only channels combined within the same transceiver block can share the two CMU PLLs available in a transceiver block.

Multiple Channels Sharing a CMU PLL

To enable the Quartus II software to share the same CMU PLL for multiple channels, the following parameters in the channel instantiations must be identical:

- Base data rate (the CMU PLL is configured for this data rate)
- CMU PLL bandwidth setting
- Reference clock frequency
- Input reference clock pin
- pll_powerdown port of the ALTGX instances must be driven from the same logic

Each channel instance can have a different local divider setting. Different settings are useful when you intend to run each channel within the transceiver block at different data rates that are derived from the same base data rate using the local divider values /1, /2, and /4. This is shown in Example 1.

Example 1

Consider a design with four instances in a **Receiver and Transmitter** configuration in the same transceiver block at the following serial data rates. Assume that each instance contains a channel, is driven from the same clock source, and has the same CMU PLL bandwidth settings.

Table 3–1 lists the configuration for Example 1.

Table 3–1. Configuration for Example 1 for Arria II Devices (Part 1 of 2)

User-Created Instance Name	ALTGX MegaWizard Plug-In Manager Settings		
	Number of Channels	Configuration	Effective Data Rate (Gbps)
inst0	1	Receiver and Transmitter	3.75
inst1	1	Receiver and Transmitter	0.9375

Table 3–1. Configuration for Example 1 for Arria II Devices (Part 2 of 2)

User-Created Instance Name	ALTGX MegaWizard Plug-In Manager Settings		
	Number of Channels	Configuration	Effective Data Rate (Gbps)
inst2	1	Receiver and Transmitter	1.875
inst3	1	Receiver and Transmitter	3.75

You can share a single CMU PLL for all four channels:

- One CMU PLL can be configured to run at 3.75 Gbps.
- Each channel can divide the CMU PLL clock output using the local divider and achieve the required data rates of 3.75 Gbps, 1.875 Gbps, and 0.9375 Gbps. Because each receiver channel has a dedicated CDR, the receiver side in each instance can be set up for these three data rates without restrictions.

The following steps describe how to achieve the configuration.

To enable the Quartus II software to share a single CMU PLL for all four channels, set the following values in the **General** screen of the ALTGX MegaWizard Plug-In Manager.

- For inst0:
  - Set **What is the effective data rate?** to 3.75 Gbps
  - Set **Specify base data rate** to 3.75 Gbps
- For inst1:
  - Set **What is the effective data rate?** to 1.875 Gbps
  - Set **Specify base data rate** to 3.75 Gbps
- For inst2:
  - Set **What is the effective data rate?** to 0.9375 Gbps
  - Set **Specify base data rate** to 3.75 Gbps
- For inst3:
  - Set **What is the effective data rate?** to 3.75 Gbps
  - Set **Specify base data rate** to 3.75 Gbps

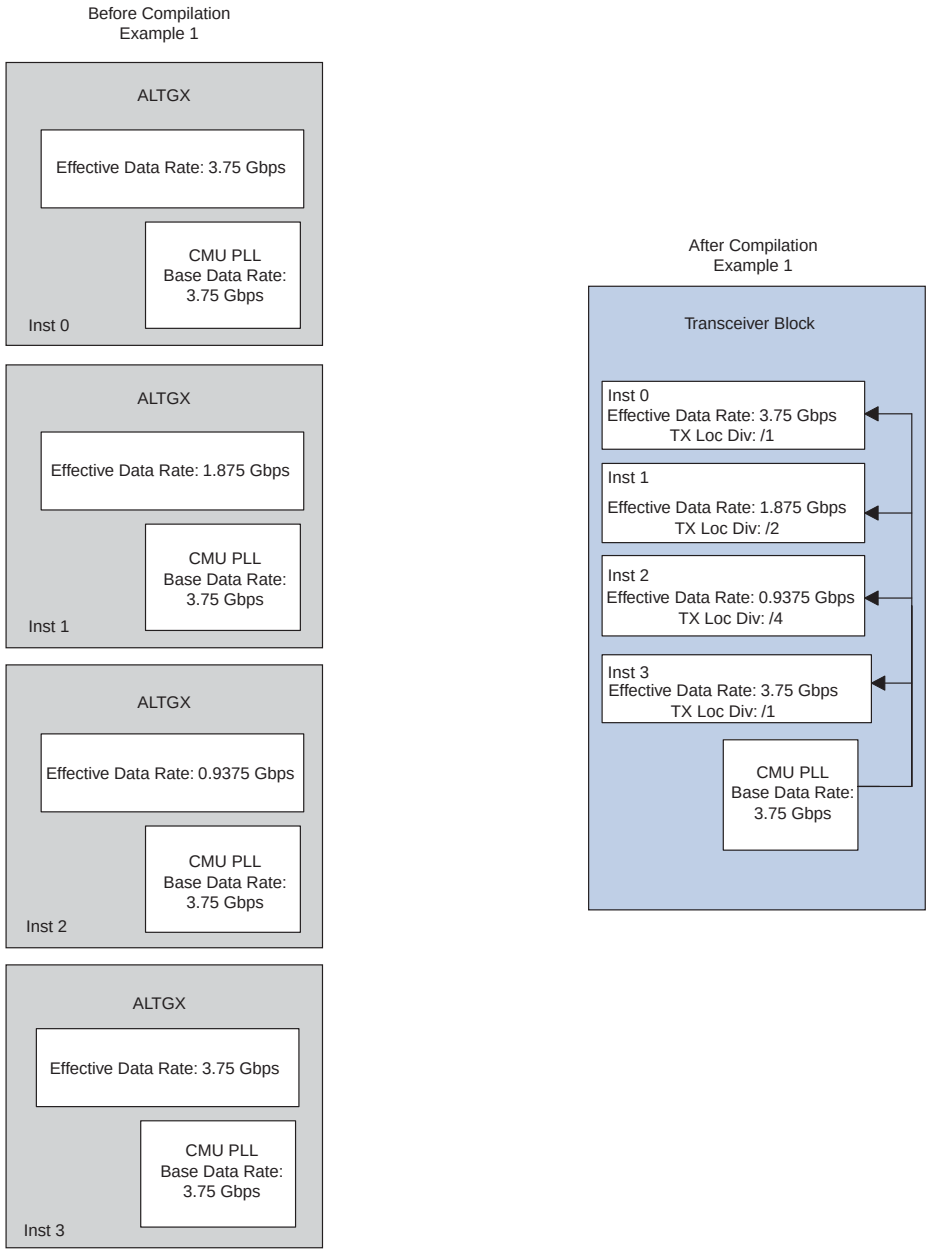
The **Specify base data rate** option is 3.75 Gbps for all four instances. Because the CMU PLL bandwidth setting and input reference clock are the same and the `pll_powerdown` ports are driven from the same logic or pin, the Quartus II software shares a single CMU PLL that runs at 3.75 Gbps.

You can force the placement of transceiver channels to a specific transceiver block by assigning pins to `tx_dataout` and `rx_datain`. Otherwise, the Quartus II software selects a transceiver block.

Figure 3–1 shows the scenario before and after the Quartus II software combines the transceiver channel instances. Because the RX CDR is not shared between channels, only the CMU PLL is shown.

Each ALTGX instance has a `pll_powerdown` port. You must drive the `pll_powerdown` ports of all instances from the same logic to allow the Quartus II software to share the same CMU PLL. If you drive the `pll_powerdown` ports of the ALTGX instance using different logic, the Quartus II software does not use the same CMU PLL even if all the other required parameters of all the ALTGX instances are identical.

Figure 3–1. ALTGX Instances Before and After Compilation for Example 1



Multiple Channels Sharing Two CMU PLLs

In some cases, a single CMU PLL is not sufficient to run the transmitter channels within a transceiver block at the desired data rates.

Use a second CMU PLL if you want to combine channels that require different configurations, such as:

- Quartus II software-defined protocols (for example, Basic, Gbps Ethernet (GbE), SONET/synchronous digital hierarchy [SDH], Serial Digital Interface [SDI], or PCI Express® [PIPE] [PCIe] modes)
- CMU PLL bandwidth settings
- Different input reference clocks

Example 2

Consider a design that requires four channels set up in a **Receiver and Transmitter** configuration in the same transceiver block at the serial data rates shown in Table 3–2.

Table 3–2. Configuration for Example 2 for Arria II Devices

User-Created Instance Name	ALTGX MegaWizard Plug-In Manager Settings		
	Number of Channels	Configuration	Effective Data Rate (Gbps)
inst0	1	Receiver and Transmitter	3.75
inst1	1	Receiver and Transmitter	1.875
inst2	1	Receiver and Transmitter	0.9375
inst3	1	Receiver and Transmitter	2

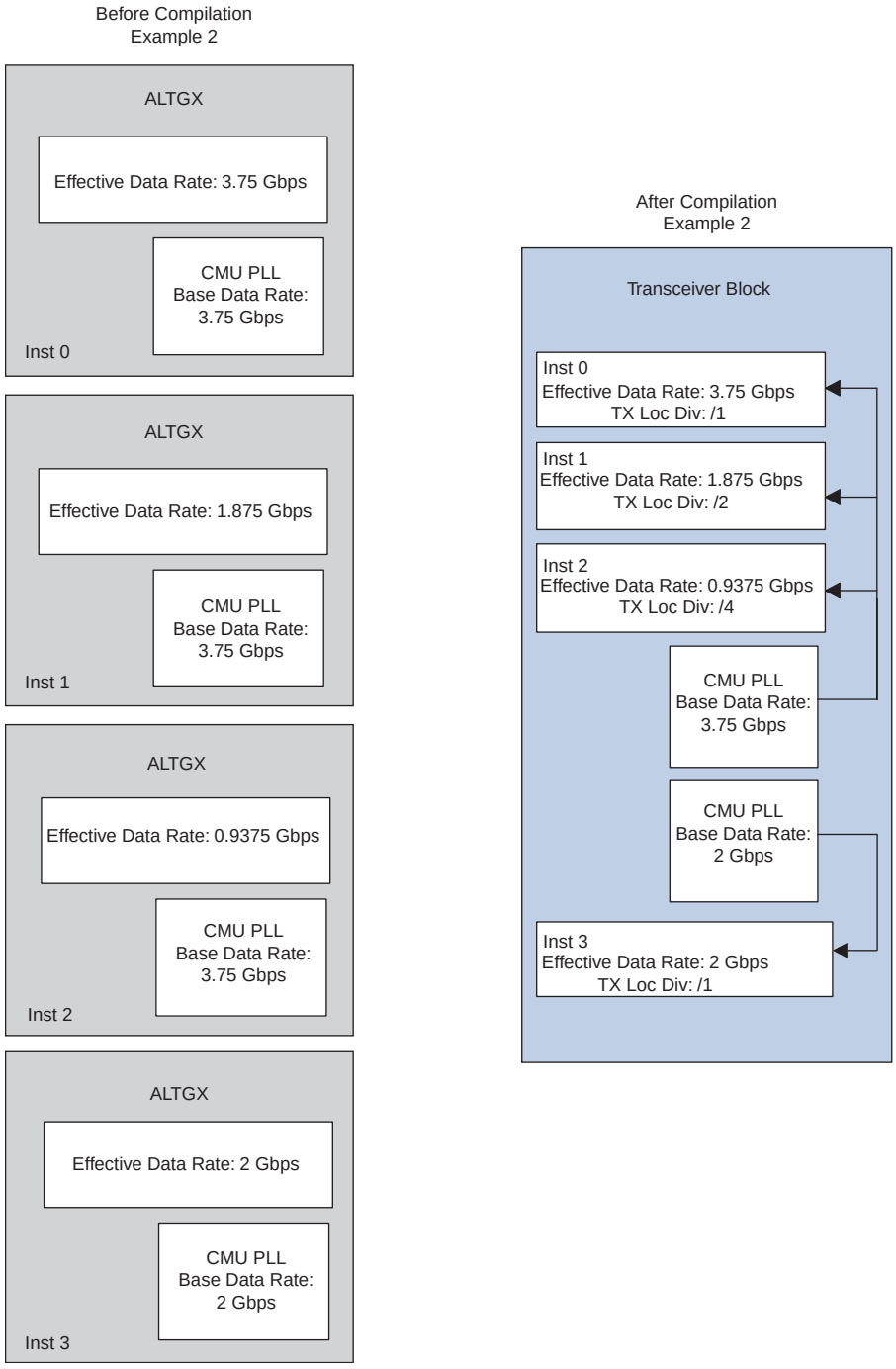
Assume that instance 0, 1, and 2 are driven from the same clock source and have the same CMU PLL bandwidth settings. In this case, you can use one CMU PLL for instance 0, 1, and 2. For the ALTGX MegaWizard Plug-In Manager settings that enable the Quartus II software to share the same CMU PLL, refer to "Example 1" on page 3–3. A second CMU PLL is required for instance 3.

You can force the placement of transceiver channels to a specific transceiver block by assigning pins to the `tx_dataout` and `rx_datain` pins of the four ALTGX instances. Otherwise, the Quartus II software selects a transceiver block.

Figure 3–2 shows the transceiver configuration before and after the Quartus II software combines the transceiver channels within the same transceiver block. Because the RX CDR is not shared between channels, only the CMU PLLs are shown.

You must connect the `pll_powerdown` port of instance 0, 1, and 2 to the same logic output to share the same CMU PLL for these instances.

Figure 3–2. ALTGX Transceiver Channel Instances Before and After Compilation for Example 2



In cases where you have two instances with the same serial data rate but with different CMU PLL data rates, the Quartus II software creates a separate CMU PLL for the two instances. For example, consider the configuration shown in Table 3–3.

Table 3–3. Sample Configuration Where Instances Cannot Be Combined in a Single Transceiver Block for Arria II Devices

User-Created Instance Name	ALTGX MegaWizard Plug-In Manager Settings			
	Number of Channels	Configuration	Effective Data Rate (Gbps)	Base Data Rate (Gbps)
inst0	1	Receiver and Transmitter	1.5	1.5
inst1	1	Receiver and Transmitter	1.5	3.0
inst2	1	Receiver and Transmitter	1	1

Even though the effective data rate of inst1 and inst0 are 1.5 Gbps (3 Gbps/2 = 1.5 Gbps), when you compile the design, the Quartus II software requires two CMU PLLs to provide clocks for the transmitter side of the two instances because their base data rates are different. In this example, you have the third instance (inst2) that requires a third CMU PLL. Therefore, the Quartus II software cannot combine the above three instances within the same transceiver block.

Combining Receiver Only Channels

You can selectively use the receiver in the transceiver channel by selecting the **Receiver Only** configuration in the **What is the Operating Mode?** option on the **General** screen of the ALTGX MegaWizard Plug-In Manager.

You can combine **Receiver Only** channel instances of different configurations and data rates into the same transceiver block. Because each receiver channel contains its own dedicated CDR, each **Receiver Only** instance (assuming one receiver channel per instance) can have different data rates.

For the Quartus II software to combine the **Receiver Only** instances within the same transceiver block, you must connect `gxb_powerdown` (if used) of all the channel instances from the same logic or input pin. For more information, refer to "General Requirements to Combine Channels" on page 3–2.

It is possible to have up to four receiver channels that can run at different data rates by using separate input reference clocks if there are enough clock routing resources available.

If you instantiate the **Receiver Only** configuration, the ALTGX MegaWizard Plug-In Manager does not allow you to enable the rate matching FIFO (clock rate compensation FIFO) in the receiver channel PCS because `tx_clkout` is not available in a **Receiver Only** instance to clock the read side of the rate matching FIFO. If you have to perform clock rate compensation, implement the rate matching FIFO in the FPGA fabric.

If you create a **Receiver Only** instance and do not use the transmitter channel that is present in the same physical transceiver channel, the Quartus II software automatically powers down the unused transmitter channel.

Combining Transmitter Channel and Receiver Channel Instances

You can create a separate transmitter channel instance and a separate receiver channel instance and assign the `tx_dataout` and `rx_datain` pins of the transmitter and receiver instance, respectively, in the same physical transceiver channel. This configuration is useful in cases where you intend to run the transmitter and receiver channel at different serial data rates. To create a transmitter channel instance and a receiver channel instance, select the **Transmitter Only** and **Receiver Only** options in the operating mode (**General** screen) of the ALTGX MegaWizard Plug-In Manager.

Multiple Transmitter Channel and Receiver Channel Instances

The Quartus II software allows you to combine multiple **Transmitter Only** channel and **Receiver Only** channel instances within the same transceiver block. Based on the pin assignments, the Quartus II software combines the corresponding **Transmitter Only** and **Receiver Only** channels in the same physical channel. To enable the Quartus II software to combine the transmitter channel and receiver channel instances in the same transceiver block, follow the rules and requirements outlined in the following sections:

- "General Requirements to Combine Channels" on page 3–2
- "Multiple Channels Sharing a CMU PLL" on page 3–3
- "Multiple Channels Sharing Two CMU PLLs" on page 3–6
- "Combining Receiver Only Channels" on page 3–8

Example 3

Consider that you create four ALTGX instances, as shown in Table 3–4.

Table 3–4. Four ALTGX Instances for Example 3 for Arria II Devices

Instance Name	Configuration	Effective Data Rate (Gbps)	Input Reference Clock Frequency (MHz)
inst0	Transmitter only	3.125	156.25
inst1	Receiver only	2.5	156.25
inst2	Transmitter only	1.25	125
inst3	Receiver only	2	125

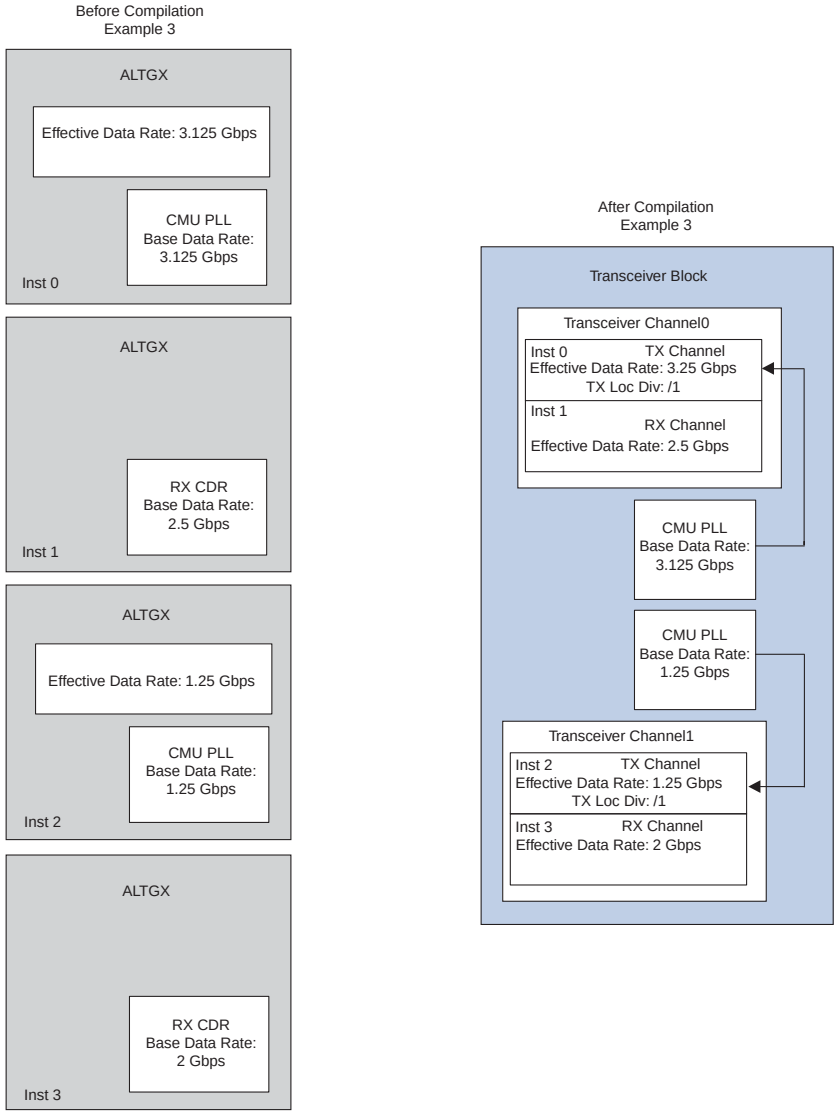
After you create the above instances, if you force the placement of the instances shown in Table 3–5, the Quartus II software combines inst0 and inst1 into physical channel 0 and inst2 and inst3 into physical channel 1.

Table 3–5. Forced Placement of the Instances for Example 3 for Arria II Devices

Instance Name	Physical Channel Pin Assignments in the Same Transceiver Block
inst0	TX pin of channel 0
inst1	RX pin of channel 0
inst2	TX pin of channel 1
inst3	RX pin of channel 1

Figure 3–3 shows the transceiver channel instances before and after compilation.

Figure 3–3. ALTGX Transceiver Channel Instances before and after Compilation for Example 3



Combining Channels Configured in Protocol Functional Modes

The following sections describe combining channels that are configured in protocol functional modes.

Basic ×4 Mode

The ALTGX MegaWizard Plug-In Manager provides a Basic mode with a ×4 option in the **Which sub-protocol will you be using?** option on the **General** screen. If you select this option, all the transmitter channels within the transceiver block receive the high-speed serial and low-speed parallel clock from the CMU0 clock divider block (present in the CMU0 channel). Each receiver channel within the transceiver block is clocked independently by the recovered clock from its receiver CDR.

When you use this mode, the ALTGX MegaWizard Plug-In Manager allows you to select one or more channels from the **What is the number of channels?** option on the **General** screen.

If you select the number of channels to be less than four, the remaining transmitter channels cannot be used within the transceiver block. Therefore, if you have more than one instance configured in **Transmit Only** or **Receiver and Transmitter** mode with the ×4 option enabled, the Quartus II software cannot combine the instances within the same transceiver block.

Only the transmitter channels share a common clock. The receiver channels are clocked independently. Therefore, you can configure the unused receiver channels within a transceiver block in any allowed configuration. For example, assume that you configure the ALTGX MegaWizard Plug-In Manager with the options shown in Table 3–6.

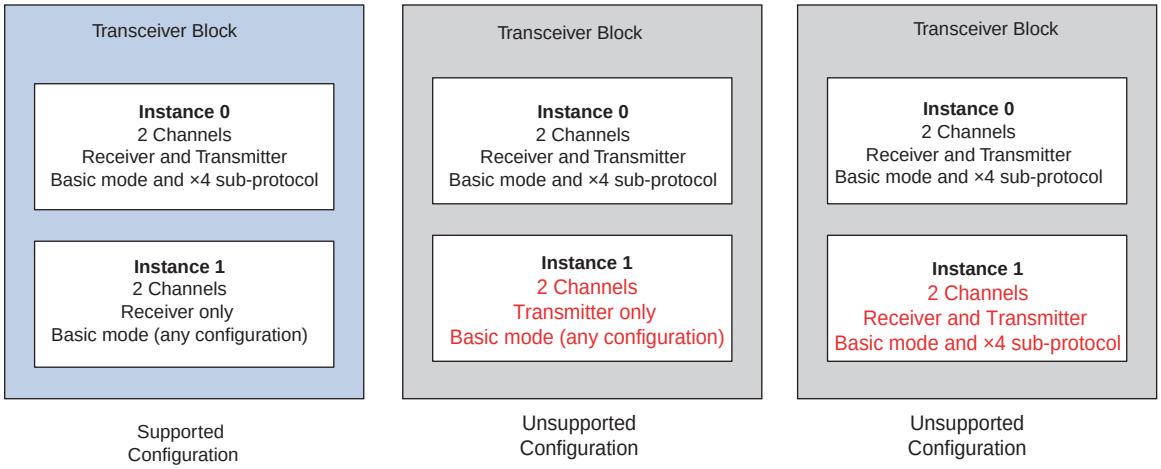
Table 3–6. General Screen Options in Basic ×4 Mode for Arria II Devices

Option	Value
What protocol will you be using?	Basic mode
Which subprotocol will you be using?	×4
What is the operating mode?	Receiver and Transmitter
What is the number of channels?	2

If you create the instance with the above selection, you cannot use the remaining two transmitter channels in the transceiver block. However, you can use the remaining two receiver channels in a different configuration.

Figure 3–4 shows examples of supported and unsupported configurations.

Figure 3–4. Examples of Supported and Unsupported Configurations to Combine Instances in Basic ×4 Mode



Combining Channels Using the PCIe hard IP Block with Other Channels

The Arria II device family contains an embedded PCIe hard IP block that performs the physical, datalink, and transaction layer functionality specified by PCIe base specification 1.1. Each PCIe hard IP block is shared by two transceiver blocks. The PCIe Compiler MegaWizard Plug-In Manager provides you the options to configure the PCIe hard IP block. When enabled, the transceiver channels associated with this block are enabled.

There are restrictions on combining transceiver channels with different functional and/or protocol modes (for example, Basic mode) within two contiguous transceiver blocks with the channels that use the PCIe hard IP block. The restrictions depend on the number of channels used (×1 or ×4) and the number of virtual channels (VC) selected in the PCIe Compiler MegaWizard Plug-In Manager. Table 3–7 lists the restrictions.

Table 3–7. PCIe Hard IP Block Restrictions When Combining Transceiver Channels with Different Functional and/or Protocol Modes for Arria II Devices *(Note 1)*, *(2)*

PCI Express Configuration (PCI Express hard IP Options Enabled in the PCIe Compiler Wizard)			Transceiver Block 0 *(3)*				Transceiver Block 1 *(4)*			
Link Width	Lane (Data Interface Width)	Virtual Channel (VC)	Ch0 *(5)*	Ch1	Ch2	Ch3	CH4	Ch5	Ch6	Ch7
1	64 bit	1	PCIe ×1	Avail.	Avail.	Avail.	Avail.	Avail.	Avail.	Avail.
4	64 bit	1	PCIe ×4	N/A	N/A	N/A	Avail.	Avail.	Avail.	Avail.
8	128 bit	1	PCIe ×8	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Notes to Table 3–7:

(1) Avail.—the channels can be used in other configurations.

(2) N/A—the channels are NOT available for use.

(3) Transceiver block 0—the master transceiver block that provides high-speed serial and low-speed parallel clocks in a PCIe ×4 or ×8 configuration.

(4) Transceiver block 1—the adjacent transceiver block that shares the same PCIe hard IP block with transceiver block 0.

(5) The physical channel 0 in the transceiver block. For more information about physical-to-logical channel mapping in PCIe functional mode, refer to the "×8 Channel Configuration" section in the *Arria II Transceiver Clocking* chapter.

For more information about the PCIe Compiler MegaCore Functions and hard IP implementation, refer to the *PCI Express Compiler User Guide*.

Combining Transceiver Instances Using PLL Cascade Clocks

The Arria II device family provides multiple input reference clock sources to clock the CMU PLLs and RX CDRs in each transceiver block. The following are the input reference clock sources that can clock the CMU PLLs and RX CDRs:

- `refclk`s from the same transceiver block
- Global clock lines
- `refclk`s from transceiver blocks on the same side of the device using the inter-transceiver block (ITB) lines
- PLL cascade clock (this is the cascaded clock output from the PLLs in the FPGA fabric)

If you use the PLL cascade clock to provide input reference clocks to the CMU PLLs or RX CDRs, there are requirements for combining transceiver channels (as described in the following sections).

The Arria II GX and GZ transceiver can cascade the output of the general purpose PLLs to the CMU PLLs and receiver CDRs. The left side of the Arria II GX and GZ device contains a PLL cascade clock network—a single line network that connects the PLL cascade clock to the transceiver block. Similarly, for Arria II GZ devices, the right side PLLs can only be cascaded with the transceivers on the right side of the device. Each side of the Arria II GZ device contains a PLL cascade clock network. This clock line is segmented to allow different PLL cascade clocks to drive the transceiver CMU PLLs and RX CDRs.

The segmentation locations differ based on the device family. Therefore, there are restrictions when you want to combine transceiver channels that use different PLL cascade clocks as input reference clocks.

For more information about using the PLL cascade clock and segmentation, refer to the "PLL Cascading" section in the *Transceiver Clocking in Arria II Devices* chapter.

Combining Transceiver Instances in Multiple Transceiver Blocks

"Creating Transceiver Channel Instances" on page 3–2 describes the method to instantiate multiple transceiver channels using a single ALTGX instance. The following section describes the method to instantiate multiple transceiver channels using multiple transceiver blocks.

When you create a transceiver instance that has more than four transceiver channels, the Quartus II software attempts to combine the transceiver channels in multiple transceiver blocks, as shown in Example 4.

Example 4

Consider that you create two ALTGX instances with the configuration shown in Table 3–8.

Table 3–8. Two ALTGX Instances for Example 4 for Arria II Devices

Instance Name	Number of Transceiver Channels	Configuration	Effective Data Rate (Gbps)	Input Reference Clock (Gbps)
inst0	7	Receiver and Transmitter	3.125	156.25
inst1	1	Receiver and Transmitter	3.125	156.25

In this case, assuming that all the required parameters specified in "Multiple Channels Sharing a CMU PLL" on page 3–3 are identical in inst0 and inst1, the Quartus II software fits inst0 and inst1 in two transceiver blocks.

Figure 3–5 shows the transceiver instances before compilation for Example 4.

Figure 3–5. Transceiver Channel Instances before Compilation for Example 4

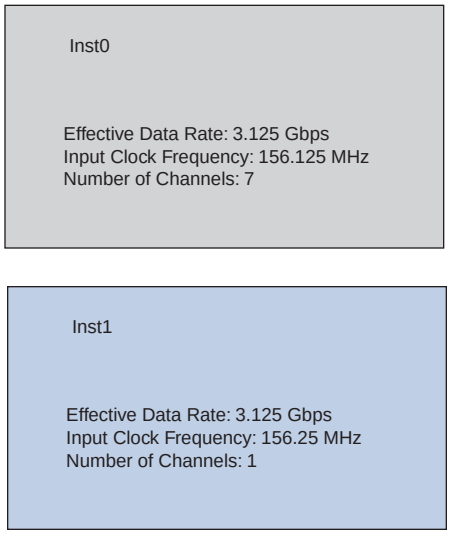
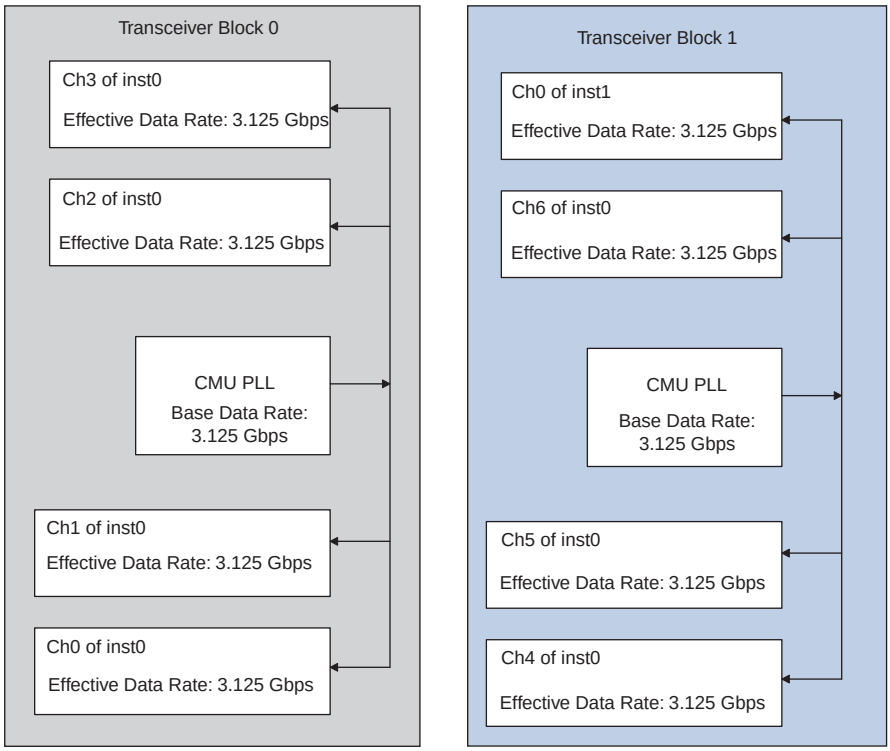


Figure 3–6 shows the transceiver instances after compilation for Example 4.

Figure 3–6. Combined Transceiver Instances after Compilation for Example 4



You can force the placement of the transceiver channels in specific transceiver banks by assigning pins to the `tx_dataout` and `rx_datain` ports of inst0 and inst1.

Even though inst0 instantiates seven transceiver channels, the ALTGX MegaWizard Plug-In Manager provides only one bit for the pll_inclk port for inst0. In your design, provide only one clock input for the pll_inclk port. The Quartus II software uses two transceiver blocks to fit the seven channels and internally connects the input reference clock (connected to the pll_inclk port in your design) to the CMU PLLs of two transceiver blocks.

For inst1, the ALTGX MegaWizard Plug-In Manager provides a pll_inclk port. In this example, it is assumed that a single reference clock is provided for inst0 and inst1. Therefore, connect the pll_inclk port of inst0 and inst1 to the same input reference clock pin to enable the Quartus II software to share a single CMU PLL in transceiver block1 that has three channels of inst0 and one channel of inst1 (shown as ch5, ch6, and ch7 in transceiver block 1) in Figure 3–6 on page 3–14.

For the RX CDRs in inst0, the ALTGX MegaWizard Plug-In Manager provides seven bits for the rx_cruclk port (if you do not select the **Train Receiver CDR from pll_inclk** option in the **PLL/Ports** screen), allowing separate input reference clocks to the RX CDRs of each channel.

Summary

The following summarizes how to configure multiple protocols and data rates in a transceiver block:

- You can run each transceiver channel at independent data rates or protocol functional modes.
- Each transceiver block consists of two CMU PLLs that provide clocks to run the transmitter channels within the transceiver block.
- To enable the Quartus II software to combine multiple instances of transceiver channels within a transceiver block, follow the rules specified in "General Requirements to Combine Channels" on page 3–2 and "Sharing CMU PLLs" on page 3–3.
- You can reset each CMU PLL within a transceiver block using a pll_powerdown signal. For each transceiver instance, the ALTGX MegaWizard Plug-In Manager provides an option to select the pll_powerdown port. If you want to share the same CMU PLL between multiple transceiver channels, connect the pll_powerdown ports of the instances and drive the signal from the same logic.
- If you enable the PCIe hard IP block using the PCIe Compiler, the Quartus II software has certain requirements about using the remaining transceiver channels within the transceiver block in other configurations. For more information, refer to "Combining Channels Using the PCIe hard IP Block with Other Channels" on page 3–12.

Document Revision History

Table 3–9 lists the revision history for this chapter.

Table 3–9. Document Revision History

Date	Version	Changes
December 2010	3.0	■ Updated to add Arria II GZ information. ■ Minor text edits.
July 2010	2.0	■ Updated the "Transceiver PLL Configurations" and "Combining Channels Using the PCIe hard IP Block with Other Channels" sections. ■ Minor text edits.
February 2009	1.0	Initial release.