

This chapter describes the Altera-recommended basic design flow that simplifies Stratix® IV GX transceiver-based designs.

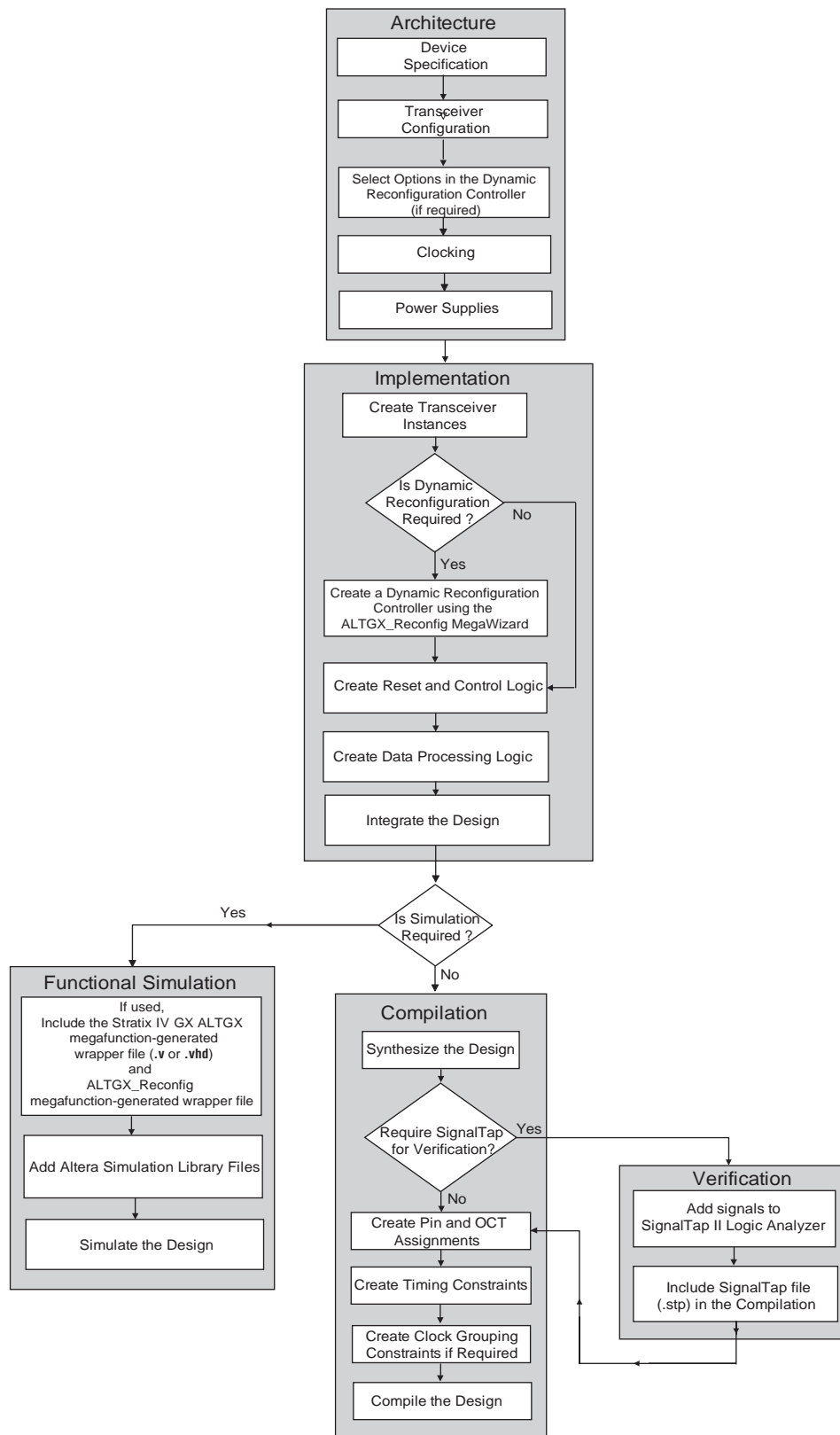
Use the following design flow techniques to simplify transceiver implementation. The “[Guidelines to Debug Transceiver-Based Designs](#)” on page 2–14 provides guidelines to trouble-shoot transceiver-based designs. An example of a fibre channel protocol application is also described in this chapter.

The transceiver-based design is divided into phases and are detailed in the following sections:

- “Architecture” on page 2–3
- “Implementation and Integration” on page 2–6
- “Compilation” on page 2–10
- “Verification” on page 2–12
- “Functional Simulation” on page 2–12
- “Example 1: Fibre Channel Protocol Application” on page 2–17

Figure 2–1 shows the design flow chart of the different stages of the design flow. The design flow stages include architecture, functional simulation, compilation, and verification. Each stage of the design flow is explained in the sections that follow.

Figure 2-1. Flow Chart of the Different Stages in a Transceiver-Based Design



Architecture


The first step in creating a transceiver-based design is to map your system requirements with the Stratix IV GX device supported features. The Stratix IV GX device contains multiple transceiver channels that you can configure in multiple data rates and protocols. It also provides multiple transceiver clocking options. For your design, identify the transceiver capabilities and clocking options to ensure that the transceiver meets your system requirements.

This section describes the critical parameters that you need to identify as part of this architecture phase.

Device Specification

The following device specifications must meet your requirements:

- Refer to the device data sheet to ensure that the transceivers meet the data rate and electrical requirements for your target high-speed interface application; for example, the jitter specification and voltage output differential (V_{OD}) range.
- Check whether the device family that you select supports your design requirements; for example, the number of transceiver channels, FPGA logic density, memory elements, and DSP blocks.
- If you intend to migrate to a higher logic density or higher transceiver count device in the future, ensure that the migration device is available.


 For information about device characteristics, refer to the “Transceiver Performance Specifications” section in the *DC and Switching Characteristics for Stratix IV Devices* chapter. For information about transceiver resources, refer to the *Stratix IV Device Family Overview* chapter.

Transceiver Configuration



Use the ALTGX MegaWizard™ Plug-In Manager interface to configure the Stratix IV transceiver channel’s features and options.

When selecting a transceiver configuration, check for the following parameters:

- Check whether the transceiver physical coding sublayer (PCS) and physical medium attachment (PMA) functional blocks comply with your system requirements. For example, check whether the rate match (clock rate compensation) FIFO in the receiver channel PCS meets the parts per million (PPM) specifications required for your application.

 For more information about transceiver specifications, refer to the “Transceiver Performance Specifications” section of the *DC and Switching Characteristics for Stratix IV Devices* chapter.




- Select a configuration that meets your latency requirements. If your system has maximum latency requirements through the transceiver data path, consider the appropriate functional configuration. The Stratix IV GX transceiver supports various configurations that differ in latency (for example, low latency PCS mode and Basic [PMA direct] mode).

- In some configurations, specific functional blocks in the transceiver are disabled or bypassed. Before you select a transceiver configuration, understand the functional blocks that must be implemented in the FPGA fabric. For example, Basic (PMA direct) mode provides reduced latency but does not have PCS functional blocks enabled (for example, word aligner and 8B/10B encoder). Therefore, implement these functional blocks in the FPGA fabric if you need them in your application. Some examples of functional blocks that you may need to implement in the FPGA fabric are shown in [“Create Data Processing and Other User Logic”](#) on page 2-8.
-  For more information about the ALTGX MegaWizard Plug-In Manager, refer to the [ALTGX Transceiver Setup Guide for Stratix IV Devices](#) chapter.
 - Check whether the loopback features are available for your selected functional mode. The Stratix IV GX transceiver provides diagnostic loopback features between the transmitter channel and the receiver channel at the transceiver PCS and PMA interfaces. These loopback features help in debugging your design.
 - If your design uses multiple transceiver channels within the same transceiver block, based on the transceiver channel configurations, the Quartus® II software might impose restrictions on combining these channels.
-  For more information about these restrictions, refer to the [Configuring Multiple Protocols and Data Rates in Stratix IV Devices](#) chapter.

Dynamic Reconfiguration

Use the Stratix IV transceivers in multiple-link interconnect environments by dynamically reconfiguring the PMA controls (for example, V_{OD} , Pre-emphasis, Equalization, DC gain, and the transceiver channel configuration). You can also reconfigure the PMA controls without affecting any other transceiver channel or the logic in the FPGA fabric.

Use the transceiver channel reconfiguration to dynamically switch a transceiver channel to multiple protocols and data rates. The Quartus II software allows you to generate a memory initialization file (.mif) that stores unique transceiver settings and provides a dynamic reconfiguration controller, which is soft logic that controls the transceiver reconfiguration with minimal user interface logic. You can generate this soft logic using the ALTGX_RECONFIG MegaWizard interface.

-  For more information about the ALTGX_RECONFIG interface, refer to the [ALTGX_RECONFIG Megafunction User Guide for Stratix IV Devices](#) chapter.
-  All receiver channels in the Stratix IV GX device require offset cancellation to counter offset variations in process, voltage, and temperature (PVT) on the receiver. The dynamic reconfiguration controller initiates the sequence to perform offset cancellation on the receiver channels. Therefore, if you configure the Stratix IV GX transceiver channel in **Receiver only** or **Transmitter and Receiver** configuration, you must instantiate a dynamic reconfiguration controller.
-  For more information about offset cancellation or dynamic reconfiguration of PMA controls or channel configuration, refer to the [“Offset Cancellation Feature”](#) section in the [Dynamic Reconfiguration in Stratix IV Devices](#) chapter.

Clocking

The Stratix IV GX transceiver is clocked by various input reference clocks, for example:

- Dedicated transceiver reference clock (`refclk`) pins. Altera recommends using `refclk` pins whenever possible because the `refclk` pins yield reduced jitter on the transmitted data.
- Clock sources connected to global clock lines.
- Clock outputs from the phase-locked loops (PLLs) in the FPGA fabric.

Identify the transceiver channels input reference clock sources, for example:

- Ensure that your selected device has the required number of input reference clock resources to implement your design.
- Ensure that the transceiver clock input supports the required I/O standards.
- Ensure that the clocking restrictions work with your selected device:
 - Check whether the allowed frequencies for the transceiver input reference clocks meet your system requirements.
 - If you use the PLL cascade clock, understand its restrictions.
 - If you are using the auxiliary transmit (ATX) PLL, understand the recommendations for the input reference clock sources and the restrictions on data rate ranges supported by the ATX PLL.

For transceiver-FPGA interface clocking:

- Ensure that the transceiver-FPGA interface clock frequency limits meet your system requirements.



For information about transceiver specifications, refer to the *DC and Switching Characteristics for Stratix IV Devices* chapter.

- Identify the clocking scheme to clock the transceiver data to the logic in the FPGA fabric. For example, if your design has multiple transceiver channels that run at the same data rate and are connected to the one upstream link, you might be able to use a single transceiver-FPGA clock to provide clocks to the transceiver data path, which can conserve clock routing resources.
- If you are using Basic (PMA direct) mode, determine whether you require a left/right PLL to provide phase shifted clocks to the FPGA fabric. The left/right PLL clocks the data received and transmitted between the transceiver and the FPGA fabric interface and may be required to meet the timing requirements of the data transfer.



For information about transceiver clocking, refer to the *Transceiver Clocking in Stratix IV Devices* chapter.

After you identify the required transceiver parameters, start the implementation and integration phase.

Power Supplies

The Stratix IV GX device requires multiple power supplies. The pin connection guidelines provide specific recommendations about the type of power supply regulator (linear or switching) and the voltage supply options and restrictions. For example, the transmitter buffer supply VCCHTx has two options—1.5 V and 1.4 V. There are specific data rate restrictions when using 1.5 V. You must understand these restrictions when you select a power supply value.

- For more information, refer to the *Stratix IV GX and Stratix IV E Device Family Pin Connection Guidelines*.

Estimate the power required to run your design. This estimation allows you to select the appropriate power supply modules and to design the power distribution network on your board.

Use the Early Power Estimator tool to estimate the transient current requirements.

- For more information about the Early Power Estimation tool, refer to the *Stratix III, Stratix IV, Stratix V, HardCopy III, and HardCopy IV PowerPlay Early Power Estimator*.

If your design is already complete, use the power optimization features available in the Stratix IV Devices.

- For more information about optimizing power in Stratix IV FPGA devices, refer to *AN 514: Power Optimization in Stratix IV FPGAs*.

Board Design Requirements

For improved signal integrity on the high-speed serial interface, follow the best design practices for your power distribution network, PCB design, and stack up.

- For detailed guidelines and recommendations about your power distribution network, PCB design, and stack up, refer to the [Board Design Resource Center](#) web site.
- For more information about the Stratix IV GX design process, refer to *AN 519: Stratix IV Design Guidelines*.

Implementation and Integration

There are three steps to the implementation and integration phase:

- “Create Transceiver Instances” on page 2-7
- “Create Reset Logic to Control the FPGA Fabric and Transceivers” on page 2-34
- “Create Data Processing and Other User Logic” on page 2-36

Create Transceiver Instances

The ALTGX MegaWizard Plug-In Manager creates the transceiver instance. In the architecture phase, you identified the transceiver configuration for your design. Using the ALTGX MegaWizard Plug-In Manager, select the appropriate parameters that apply to your architecture requirements.

Reset and Status Signals

The ALTGX MegaWizard Plug-In Manager provides various reset and status signals:



- Reset signals—`tx_digitalreset`, `rx_digitalreset`, `rx_analogreset`, and `pll_powerdown` are required to reset the transceiver PCS and PMA functional blocks.
- Status signals—`rx_freqlocked` and `pll_locked` indicate the state of the receiver CDR and transmitter PLL, respectively. Use these reset and status signals to implement the transceiver reset control logic in the FPGA fabric. For more information, refer to “[Create Reset and Control Logic](#)” on page 2-8.

If you determine that your application requires dynamic reconfiguration, select the options in the **Reconfig** screen of the ALTGX MegaWizard interface.

If you intend to dynamically reconfigure the channel into other protocol modes or data rates, the **Reconfig** screen provides multiple options (for example, the **channel interface** and **Use alternate PLL** options) to enable this feature.

-  To understand the **logical channel addressing**, **logical PLL index**, and **type of reconfiguration to select** options in the **Reconfig** screen, refer to the “[Channel and CMU PLL Reconfiguration Mode Details](#)” section in the *Dynamic Reconfiguration in Stratix IV Devices* chapter.


Depending on your system, when you use multiple transceiver channels, you might be able to share the transmitter and receiver parallel clocks of one channel with the other channels. If your design requires sharing a clock resource, select the `tx_coreclk` and `rx_coreclk` ports.

-  Transceiver-FPGA fabric interface clock sharing conditions are provided in the *Transceiver Clocking in Stratix IV Devices* chapter.
-  For more information about using the ALTGX MegaWizard Plug-In Manager and the functionality of the different options and signals available, refer to the *ALTGX Transceiver Setup Guide for Stratix IV Devices* chapter.

Create Dynamic Reconfiguration Controller Instances


Use the ALTGX_RECONFIG MegaWizard interface to create the dynamic reconfiguration controller instance. If you intend to use the channel and CMU PLL reconfiguration feature, select the relevant options in the ALTGX_RECONFIG Megawizard Plug-In Manager.

-  For descriptions of the options in the ALTGX_RECONFIG megafunction, refer to the *ALTGX_RECONFIG Megafunction User Guide for Stratix IV Devices* chapter.

 For more information about using the signals, refer to the *Dynamic Reconfiguration in Stratix IV Devices* chapter.

Create Reset and Control Logic

The reset sequence is important for initializing the transceiver functional blocks to proper operating condition. Altera recommends a reset sequence for different transceiver configurations and protocol functional modes. The ALTGX MegaWizard Plug-In Manager provides the `tx_digitalreset`, `rx_analogreset`, `rx_digitalreset`, and `p11_powerdown` signals to reset the different functional blocks of the transceiver. You can reset the CMU PLL or the ATX PLL (based on your selection) using the `p11_powerdown` signal. For transceiver instances that share the same CMU PLL or ATX PLL, the `p11_powerdown` port of these instances must be driven by the same logic.

 For more information about reset sequences, refer to the *Reset Control and Power Down in Stratix IV Devices* chapter.

Create Data Processing and Other User Logic

A typical transceiver-based design consists of custom data processing and other user logic that must be implemented in the FPGA fabric based on your application requirements. In addition to application-specific logic, for specific transceiver configurations, you may need additional logic to interface with the transceivers. This section provides examples of such logic.

PPM Detector When the Receiver CDR Is Used in Manual Lock Mode

Each receiver channel contains a clock data recovery (CDR) that you can use in automatic or manual lock mode.

If you use the receiver CDR in manual lock mode, you can control the timing of the CDR to lock to the input reference clock using the `rx_locktorefc1k` port or lock to the recovered data using the `rx_locktodata` port.

When you use the receiver CDR in manual lock mode, you may need to implement the PPM detector in the FPGA fabric to determine the PPM difference between the upstream transmitter and the Stratix IV GX receiver.

Synchronization State Machine in Manual Word Alignment Mode

Each receiver channel contains a synchronization state machine in the PCS that you can enable in certain functional modes. The synchronization state machine triggers the loss of synchronization status to the FPGA fabric based on invalid 8B/10B code groups.

However, the synchronization state machine in the PCS is not available in some functional modes. You may need to implement custom logic in the FPGA fabric to indicate the loss-of-synchronization status of the received data.

Gear Boxing Logic

Some protocols require a wider data path than provided by the transceiver interface; for example, the Interlaken Protocol requires 64/67-bit encoding and decoding, but the maximum data path interface in the Stratix IV GX transceiver is 40 bits. Therefore, you must implement gear box logic to interface the 64/67-bit encoder-decoder with the transceiver interface.

Functional Blocks to Interface with the Transceiver Configured in Basic (PMA Direct) Mode

In Basic (PMA direct) mode, all the PCS functional blocks in the transceiver channel are disabled. Therefore, you may need to implement the following blocks in the FPGA fabric:

- Word Alignment—To align the byte boundary on the received data.
- Byte Deserializer—To increase the data path width to the rest of the user logic and to reduce the clock frequency of the data path by two.
- Phase Compensation FIFO (for bonded channel applications)—In bonded channel applications in which multiple transceiver channels are connected to the same upstream system (for example, one Interlaken Protocol link using 24 transceiver channels). To minimize the global clock routing resources you use, implement a phase compensation FIFO to interface the receiver side of the transceiver interface with the logic in the FPGA Fabric.
 - Use the recovered clock from each channel to clock the write side of the phase compensation FIFO.
 - Use the recovered clock from any of the channels to clock the read side of the phase compensation FIFO.

With this method, you only use one clock resource and the subsequent receive-side logic in the FPGA fabric can operate in this single clock domain.

- Deskew Logic (for bonded channel applications)—In bonded channel applications in which multiple transceiver channels are connected to the same upstream system, the data received between multiple channels are not aligned due to potential skew in the interconnect and the upstream transmitter system. To compensate for the skew, use deskew logic in the FPGA fabric.
- Encoding/Decoding or Scrambling/Descrambling—Many protocols require the transmitter data to be encoded or scrambled to maintain signal integrity. This logic may be required in the FPGA fabric based on your application requirements.

Integrate the Design

After you implement all of the required logic, integrate the transceiver instances with the remaining logic and provide the appropriate transceiver-FPGA fabric interface clocking. Synthesize the design using third-party synthesis tools, such as Synopsys Synplicity or the Quartus II software synthesis tool. This allows you to detect syntax errors in your design.



If you are using the transceiver in Basic (PMA direct) mode, you must develop all the PCS functionality in the FPGA fabric.

Compilation

When you compile your design, the Quartus II software generates an SRAM Object File (.sof) or programmer object file (.pof) that you can download to the Stratix IV GX hardware. Typically, the first step in compiling the design is assigning pin locations for the I/Os and clocks. Use the pin planner tool in the Quartus II software to assign pins.



For a basic tutorial about the Quartus II software, open the Quartus II software, click the **Help** menu and select **Tutorial**.

- Stratix IV GX transceivers support a variety of I/O standards for the input reference clocks and serial data pins. Assign pins and the logic level standard (for example, 1.5-V PCML and LVDS) for the input and output pins.



For more information, refer to the *I/O Features in Stratix IV Devices* chapter.

- If you share the same transceiver-FPGA fabric interface clocks for multiple transceiver channels (tx_coreclk and rx_coreclk) in your design, set the **0 ppm** constraints. These constraints enable the Quartus II software to relax the legality check restrictions on clocking.



For more information, refer to the “Common Clock Driver Selection Rules” section of the *Transceiver Clocking in Stratix IV Devices* chapter.

- For transceiver serial pins and refclk pins, set the on-chip termination (OCT) resistor settings.



For more information about supported OCT settings, refer to “Transmitter Output Buffer” section of the *Transceiver Architecture in Stratix IV Devices* chapter.

- Create timing constraints for the clocks and data paths. Use the TimeQuest Timing Analyzer to set timing constraints.



For more information about the TimeQuest Timing Analyzer, refer to the *Quartus II Development Software Handbook*.

- Compile the design. This generates a .sof that can be downloaded in the FPGA.

The Quartus II software generates multiple report files that contain information such as transceiver configuration and clock resource utilization. The following section describes the report files relevant to using transceivers and clock resource.

Report Files

The Quartus II software provides a report file in the synthesis, fitter, map, placement, and assembler stages. The report file provides useful information on the device and transceiver configuration generated by the Quartus II software. This section only describes the reports provided in the fitter stage. To access the report, click on the **Processing** menu, select the **Compilation Report** option and expand the **Fitter** tab.

Fitter Summary

The fitter summary provides high-level information on the FPGA fabric resources and transceiver channels used by your design. For example, to ensure that the Quartus II software has created the number of transceiver channels as specified in your design, refer to the GXB Receiver channels and GXB Transmitter channels field at the bottom of the report. For detailed information on resource utilization, expand the **Fitter** tab.

Pin-Out File

Select the **Pin-Out** file option under the **Fitter** tab. The Quartus II software displays the I/O standards and bank numbers of all the pins (used and unused) needed to connect to the board. The Quartus II software also generates a PIN file (**.pin**) with the above information. Altera recommends using the **.pin** as a guideline. Use the pin connection guidelines for board layout.



For more information about pin connection guidelines for board layout, refer to [Stratix IV GX and Stratix IV E Device Family Pin Connection Guidelines](#).

Resource Section

Expand the **Resource Section** option under the **Fitter** tab to view the following tabs:

- The **GXB Transmitter channel** tab—Provides generated settings for all the transmitter channels instantiated in your design.
- The **GXB Transmitter PLL** tab—Provides generated settings for all the transmitter PLLs instantiated in your design.
- The **GXB Receiver channel** tab—Provides generated settings for all the receiver channels instantiated in your design.
- The **Global and other fast signals** tab—Displays the list of clock and other signals in your design that are assigned to the global and regional clock resources.

You can use the report file to verify whether the transceiver settings (for example, data rate), are generated per your settings in the ALTGX MegaWizard Plug-In Manager.


Verification

The SignalTap® Logic Analyzer allows you to verify design functionality using the on-chip logic analyzer. SignalTap provides options to create multiple sets of signals that can be sampled using different trigger clocks. You can add the signals to the SignalTap Logic Analyzer and save the file as an STP file (.stp). When you include this .stp along with the design files and compile the design, the Quartus II software creates an .sof that allows you to verify the functionality of the signals that you added in the SignalTap Logic Analyzer file.

You can run the .stp that connects to the device through the JTAG port and displays the signal transitions using the Quartus II software. Because the JTAG port is required to run SignalTap, consider designing the board with the JTAG interface for debugging your system.

 For more information about using SignalTap, refer to the *Design Debugging Using the SignalTap II Embedded Logic Analyzer* section in volume 3 of the *Quartus II Development Software Handbook*.

To verify the functionality of the PCS and PMA blocks, the Stratix IV GX transceiver provides diagnostic loopback features between the transmitter and the receiver channels.


 For more information, refer to the “Loopback Modes” section in the *Transceiver Architecture in Stratix IV Devices* chapter.

Functional Simulation

Use the ALTGX MegaWizard Plug-In Manager-generated wrapper file to simulate the instantiated transceiver configuration in third-party simulation software such as ModelSim. For simulation, specific Altera® simulation library files are required (listed in Table 2-1). The following library files are available in VHDL and Verilog versions:

- 220pack
- 220model
- altera_mf_components
- altera_mf
- sgate_pack
- sgate
- stratixiv_hssi_component
- stratixiv_hssi_atoms

These simulation files are available under the following folder in the Quartus II installation directory: *<Quartus II installation folder>/eda/sim_lib*

 The stratixiv_hssi_component library file is only applicable if the transceiver instance is created using VHDL.

For VHDL simulation using ModelSim, create the following libraries in your ModelSim project:

- lpm
- sgate
- altera_mf
- stratixiv_hssi

These simulation files are available under *<Quartus II installation folder\quartus\eda\sim_lib>*.

Compile the simulation files into the libraries specified in [Table 2-1](#).

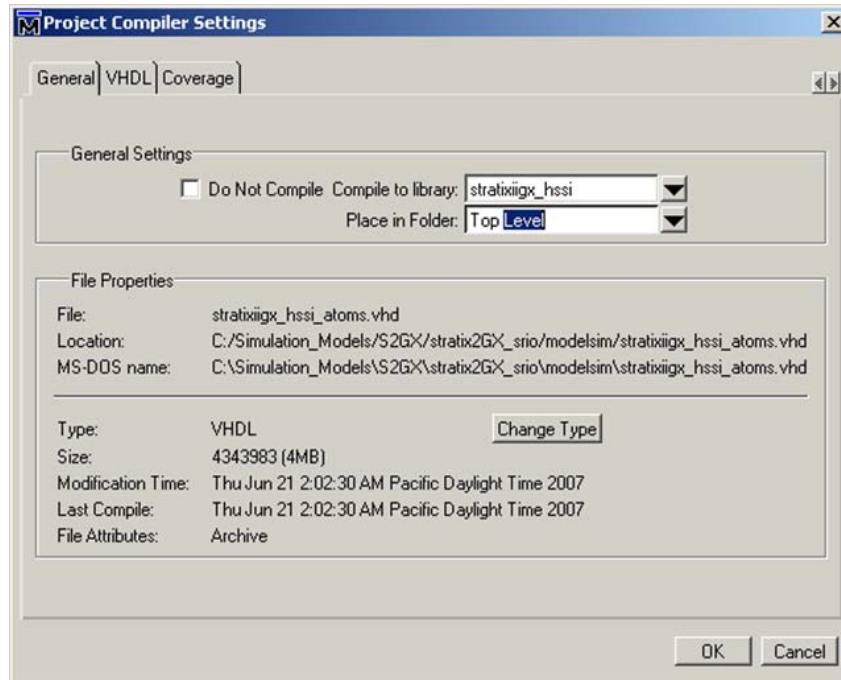
Table 2-1. Library to Compile Simulation Files

Altera Simulation Files	Library
220pack	lpm
220model	lpm
sgate pack	sgate
sgate	sgate
altera_mf_components	altera_mf
altera_mf	altera_mf
stratixiv_hssi_component	stratixiv_hssi
stratixiv_hssi_atoms	stratixiv_hssi
user design files	work

For example, to compile a file into a specific library using ModelSim, right click on the file, select **Properties**, then click the **General** tab.

In the **Compile to library** option, select the corresponding library for the file selected. [Figure 2-2](#) shows the ModelSim window compilation of files in a specific library for the Stratix II GX device.


Figure 2-2. ModelSim Option to Compile Files in a Specific Library



Include all the libraries in the search path. Add the ALTGX and ALTGX_RECONFIG MegaWizard Plug-In Manager-generated wrapper files (.v or .vhd) and all of the design files to the library. Compile all the library files first, then the design files, and lastly run the simulation.

For Verilog simulation, add the ALTGX and ALTGX_RECONFIG MegaWizard Plug-In Manager-generated Verilog wrapper files (.v), the Altera library files, and all of the design files. Compile all the library files first, then the simulation model file, followed by the design files. Lastly, run the simulation.

These guidelines are further described in [“Example 1: Fibre Channel Protocol Application”](#) below.

 For more information about functional register transfer level (RTL) simulation or post-fit simulation, refer to the [Simulation](#) chapter in volume 3 of the *Quartus II Handbook*.


Guidelines to Debug Transceiver-Based Designs

This section provides guidelines to debug transceiver-based designs. If a system failure occurs, the first step is to ensure the functionality of the logic within the FPGA. Use the following information when you observe a system failure.


Guidelines to Debug the FPGA Logic and the Transceiver Interface

Before checking the functionality in silicon, perform functional simulation to ensure the basic functionality of the RTL and the transceiver-FPGA fabric interface.


- Understand the limitations of functional simulation. If you intend to simulate timing parameters, consider post-fit simulation. The functional simulation model for transceivers does not model timing-related parameters or uncertainties in the transceiver data path. For example, the PPM difference in the rate matcher clocks (clock rate compensation) or the phase differences between the read and write side of the phase compensation FIFO are not modeled.

 For information about functional RTL simulation or post-fit simulation, refer to the *Simulation* chapter in volume 3 of the *Quartus II Handbook*.

- Check whether the compiled design has timing violations in the TimeQuest Timing Analyzer report. Set the appropriate timing constraints on the failing paths.

 For information about using the TimeQuest Timing Analyzer, refer to the *The Quartus II TimeQuest Timing Analyzer* chapter in volume 3 of the *Quartus II Handbook*.

- Verify the functionality of the transmitter and receiver data path with serial loopback. Dynamically control the serial loopback through the `rx_serial_lpbken` port. When this signal is asserted, data from the transmitter serializer is looped back to the receiver CDR of the channel.
- Use SignalTap to verify the behavior of the user logic and the transceiver interface signals. If you have FPGA I/O pins available for debug, you can also use the external logic analyzer to debug the functionality of the device.


 For more information, refer to the *In-System Debugging Using External Logic Analyzers* chapter in volume 3 of the *Quartus II Handbook*.

 To use these features, you must connect the JTAG configuration pins in the FPGA.


- Verify the interconnect on the receive side by configuring the transceiver in reverse serial loopback mode. In this case, the recovered data from the receiver channel is sent to the transmitter buffer. To configure a transceiver channel operating in a different configuration to reverse serial loopback mode, use the dynamic reconfiguration controller.
- Check whether the transceiver FPGA fabric interface clocking schemes follow the recommendations provided in the “FPGA Fabric-Transceiver Interface Clocking” section in the *Transceiver Clocking in Stratix IV Devices* chapter.
- Ensure that you have used the recommended transceiver reset sequence.

Guidelines to Debug System Level Issues


If you have determined that the logic in the FPGA fabric is functionally correct, check for system level issues:

- Check the voltage ripple across the 2 k Ω resistor that is connected to the RREF pin. The voltage ripple must be less than 60 mv.
 - Measure the eye on the near-end and far-end of the transmitter to understand the jitter added by the transmitter and interconnect.
 - Ensure that the high-speed scopes you use for measurement have sufficient bandwidth (the bandwidth rating on the scope and cables must be at least three times the serial data rate).
 - Check whether the eye meets the eye-mask requirements if specified by the protocol application.
 - Use scopes that provide information on the different jitter components to understand the possible source of the increased jitter. For example, increased intersymbol interface (ISI) indicates potential bandwidth limitations on the interconnect.
-  Some scopes, such as Agilent 86100C DCA, require pre-defined patterns (for example, PRBS7 or PRBS23) to provide jitter components.
- Measure signals on the traces (no connector) using a high-impedance differential probe with short leads.
 - Ensure that characteristic impedance on the interconnect matches the source and load systems.
 - Check for impedance discontinuities on the trace by Time Domain Reflectometry (TDR).
 - Revisit the board design, layout, and routing for any inconsistencies that can cause impedance discontinuities.
 - Check whether the termination schemes on the Stratix IV GX device and on the upstream system are matched. Altera recommends using OCT in the Stratix IV GX device instead of external termination to improve signal integrity.
 - Change the transmit output differential voltage to improve eye amplitude.
 - Compensate for high frequency losses in the interconnect by changing the equalization settings of the Stratix IV GX device and check for improvement of the bit error rate. If the upstream system does not have an equalization feature, increase the pre-emphasis (1st post tap) of the Stratix IV GX transmitter. In cases where there are multiple interconnects between the Stratix IV GX device and the upstream system, use the pre-tap and 2nd post tap. Altera provides tools to select the pre-emphasis.

- Measure the increase in jitter at the near end and far end with one channel turned on at a time if you have multiple transceiver channels connected to the upstream system. This helps to observe the effect of cross talk from adjacent channels on the victim channel.
 - Check the board layout and routing to ensure that you have implemented the design practices to mitigate cross talk.
- Ensure that the input voltage and duty cycle of the input reference clock source provided to the transmitter PLLs meet the input reference clock requirements.
- Check whether the voltage drop on the power supplies is within the specified tolerance range.
 - Measure the voltage at the via beneath the power supply pin using a high-impedance probe.
 - Check whether the voltage regulator specifications meet the Stratix IV GX power supply requirements.
 - Revisit the power distribution scheme for the supply voltage to ensure that it is designed to handle the transient current requirements of the transceiver.

 For the tolerance values of the different power supplies, refer to the *DC and Switching Characteristics for Stratix IV Devices* chapter.

- Check for periodic modulation of other frequency components on the transmit data. Send a high-frequency pattern (1010) from the transmitter side and connect the transmitter serial output to a spectrum analyzer.

 For more information about debugging Stratix IV GX transceivers, refer to *AN 553: Debugging Transceivers*.

Example 1: Fibre Channel Protocol Application

Assume that you want to implement a fibre channel protocol application using three transceiver channels. Consider the following system requirements:

- You need three transceiver channels
- All the channels need to be placed in the same transceiver block
- All the channels need to have independent control to reset their PCS and PMA functional blocks

Table 2-2 lists the transceiver channel configuration for Example 1.

Table 2-2. Transceiver Channel Configuration for Example 1

Channels	Mode of Operation	Data Rate	Input Reference Clock Frequency (MHz)
0	Receiver and Transmitter	FC4G (4.25 Gbps)	106.25
1	Receiver and Transmitter	FC1G (1.0625 Gbps)	53.125
2	Transmitter Only	FC4G (4.25 Gbps)	106.25

Phase 1—Architecture

In this phase, check whether the Stratix IV GX device supports or meets your design requirements.

Device Specification

Consider the questions listed in [Table 2-3](#) before setting device-specific parameters.

Table 2-3. Device Specific Parameters

Questions	Answer
Do the parameters meet the fibre channel protocol electrical requirements?	Yes For more information, refer to the “Transceiver Performance Characteristics” section in the <i>DC and Switching Characteristics for Stratix IV Devices</i> chapter
Are three transceiver channels available?	Yes
Is there support for 4.25 Gbps and 1.0625 Gbps data rates?	Yes Two CMU PLLs are available within each transceiver block to support two different transmitter data rates. Each receiver channel contains a dedicated receiver CDR that supports 4.25 Gbps and 1.0625 Gbps data rates.

 For the maximum data rates supported, refer to the “Transceiver Performance Specifications” section in the *DC and Switching Characteristics for Stratix IV Devices* chapter.

Transceiver Configuration

The fibre channel protocol uses an 8B/10B encoder and requires clock rate compensation.

Functional Blocks

Consider the questions listed in [Table 2-4](#) before configuring the transceiver.

Table 2-4. Configuring the Transceiver

Questions	Answer
Is the 8B/10B encoder in the PCS block fibre channel compliant?	No The fibre channel protocol consists of two different End-of-Frame (EOFt) ordered sets. The correct EOFt ordered set sent by the user logic depends on the ending disparity of the word preceding the EOFt. The Stratix IV GX transceiver does not provide running disparity flags to the user logic. Therefore, the user logic might not be able to select the correct EOFt ordered set.
Is there a workaround?	Yes Implement the 8B/10B encoder in the FPGA fabric.
Is the clock rate compensation block in the PCS available without an 8B/10B encoder?	No You can implement this in the FPGA fabric.



The design requires a **Transmitter and Receiver** configuration for two channels and a **Transmitter Only** configuration for one channel (Table 2-5).

Table 2-5. Multiple Channels


Questions	Answer
Does the Stratix IV GX transceiver support these two configurations and allow you to combine them within the same transceiver block	Yes The available FPGA fabric interface width is 20 or 40 bits to support 4.25 Gbps and 1.0625 Gbps data rates, respectively. This FPGA fabric interface facilitates 8B/10B encoding and decoding in the FPGA fabric without additional re-arrangement of the received parallel data to a 10-bit boundary.

Dynamic Reconfiguration

If your application requires you to dynamically reconfigure the transceiver PMA controls, ensure that you understand the settings, options, and user logic required to enable this feature.

-  For more information, refer to the “Interfacing ALTGX and ALTGX_RECONFIG Instances” section in the *Dynamic Reconfiguration in Stratix IV Devices* chapter.
-  For more information about initiating read and write transactions, refer to the “Dynamically Reconfiguring PMA Controls” section in the *Dynamic Reconfiguration in Stratix IV Devices* chapter.

If you are using the channel reconfiguration feature, enable the appropriate options in the ALTGX and ALTGX_RECONFIG MegaWizard Plug-In Managers.

-  You can dynamically use the reconfiguration modes to reconfigure different functional blocks in a transceiver channel using **.mifs**. For information about generating **.mifs**, refer to the “Channel and CMU PLL Reconfiguration Mode Details” section in the *Dynamic Reconfiguration in Stratix IV Devices* chapter.

Clocking

Consider the questions listed in Table 2-6 before configuring clocking.

Table 2-6. Configuring Clocking (Part 1 of 2)

Questions	Answer
Is there support for two different input reference clocks?	Yes The Stratix IV GX transceiver has two <code>refclk</code> pins for each transceiver block.
Do the <code>refclk</code> pins support the required frequency range?	Yes The minimum frequency range of <code>refclk</code> is 50 MHz; the maximum frequency range is 622.08 MHz.

Table 2-6. Configuring Clocking (Part 2 of 2)

Questions	Answer
Can transceiver-FPGA fabric interface clocking be shared?	No The design requires independent control on all channels, so you must not share the transceiver-FPGA fabric interface clock of one channel with another channel. Each of the channels must use its own <code>tx_clkout</code> and <code>rx_clkout</code> signals to clock the data between the transceiver channels and the FPGA fabric.
Does the Stratix IV GX transceiver support this feature?	Yes


 For more information about clocking the transmitter and receiver channel data path for this type of configuration, refer to the “Transmitter Channel Datapath Clocking” section of the *Transceiver Clocking in Stratix IV Devices* chapter.

Figure 2-3 shows the transmitter side of the transceiver setup for Example 1.


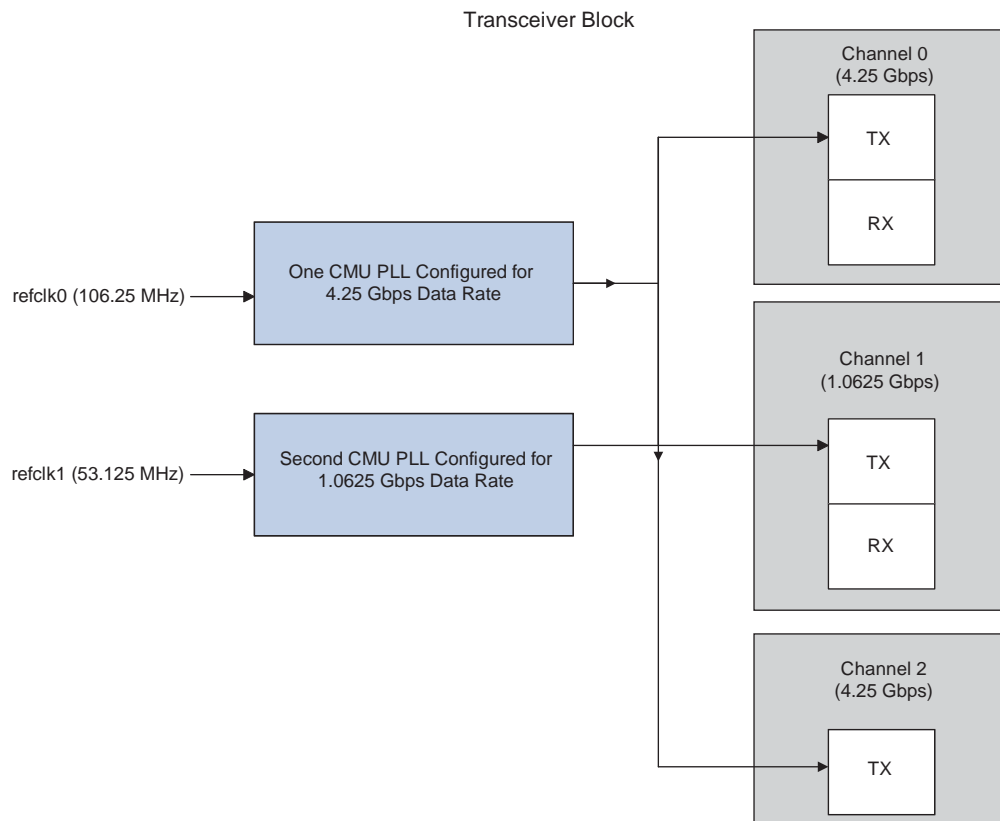
 The transmitter side receives its clocks from the clock multiplier unit (CMU) PLLs. The receiver side contains its dedicated CDR that provides the high-speed serial and low-speed parallel clocks to its PMA and PCS blocks, respectively.

Figure 2-3. Top-Level Transceiver Setup—Transmitter-Side Only

Phase 2—Implementation

Create the transceiver instance using the ALTGX MegaWizard Plug-In Manager.

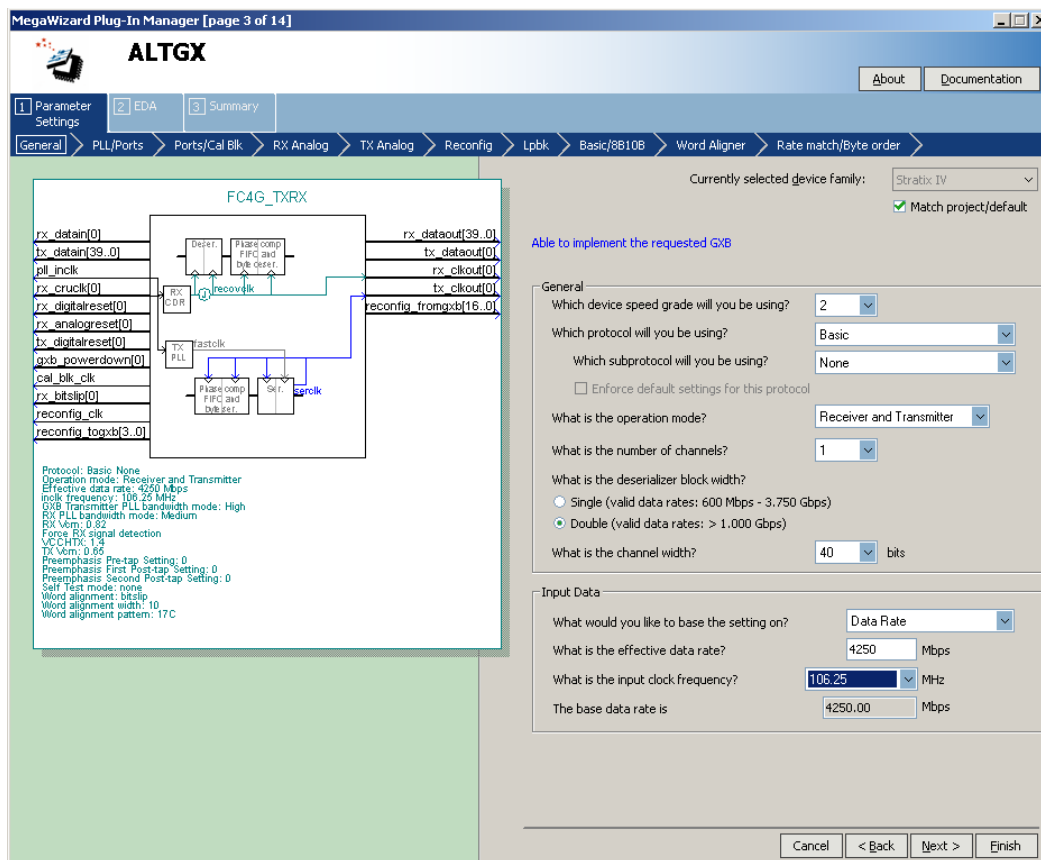
For a description of the individual options, refer to the *ALTGX Transceiver Setup Guide for Stratix IV Devices* chapter.

Create the Transceiver Instance for an FC4G Configuration (Channel 0)

Figure 2-4 through Figure 2-14 show the different options available in the ALTGX MegaWizard Plug-In Manager to create the transceiver channel instance for the FC4G data rate. Use this instance for channel 0, with the following settings:

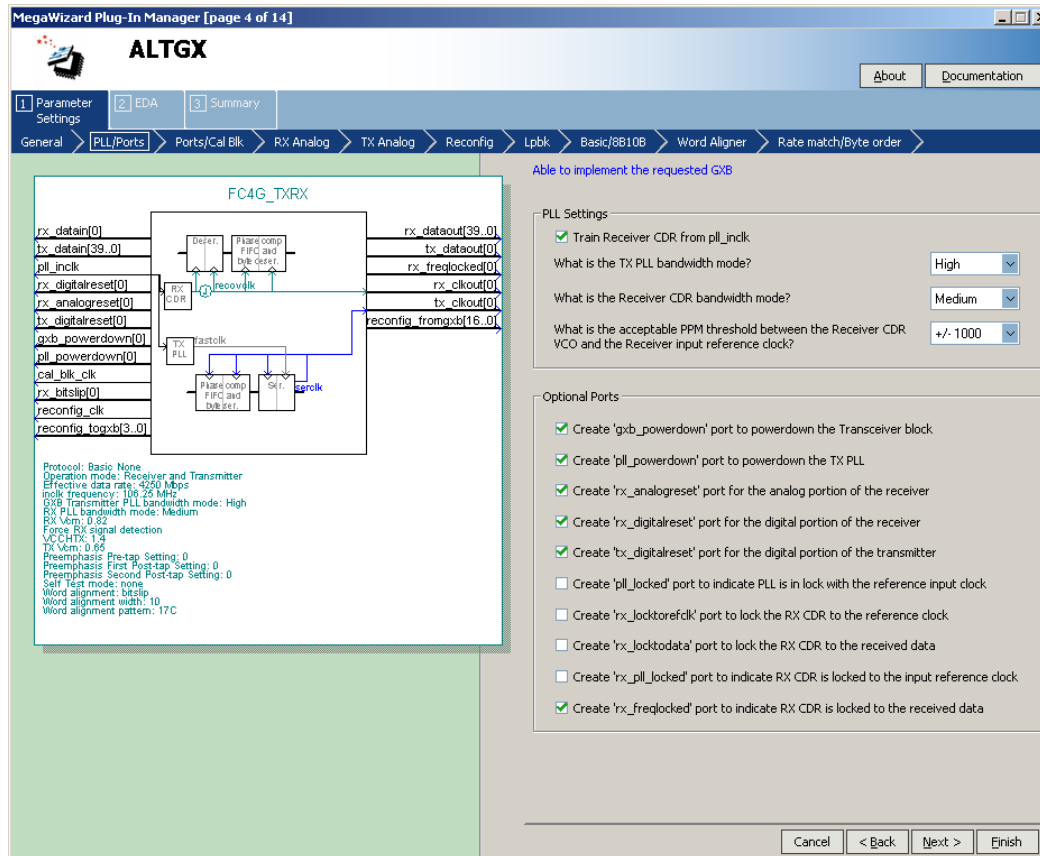
- **General** screen—You can configure the Stratix IV GX transceiver for fibre channel protocol using Basic mode. Set the options with the values shown in Figure 2-4.

Figure 2-4. FC4G Instance Settings (General Screen)



- **PLL/Ports screen**—Check the **Train Receiver CDR from PLL inclk** option, as shown in Figure 2-5. When you select this option, the same input reference clock used for the CMU PLL is provided as a training clock to the receiver CDR.

Figure 2-5. FC4G Instance Settings (PLL/Ports Screen)

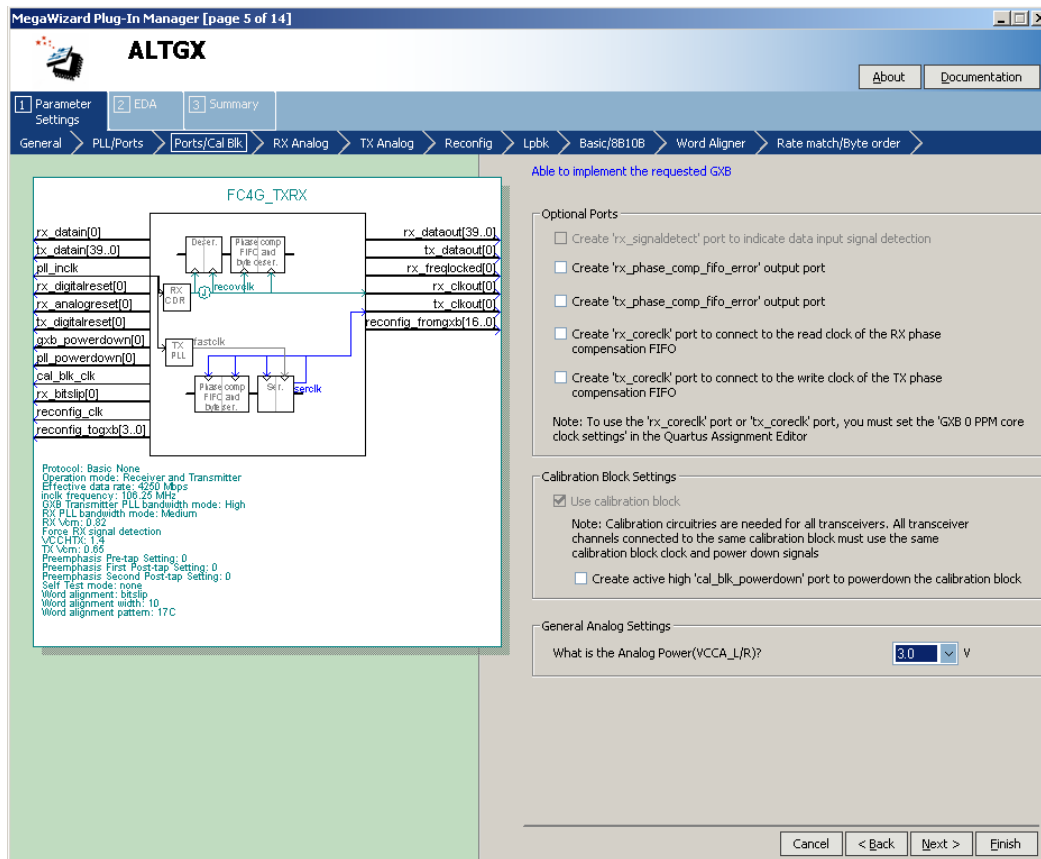


- Check the `pll_powerdown` signal. This signal allows you to power down the CMU PLL. Use this signal as part of your reset sequence.
- Check the `pll_locked` signal. This signal indicates whether the CMU PLL is locked to the input reference clock. The user logic waits until the `pll_locked` signal goes high before transmitting data.
- Check the `rx_freqlocked` signal. This signal indicates whether the receiver CDR is locked to data. When the receiver CDR is configured in automatic lock mode, assert the `rx_digitalreset` signal if the `rx_freqlocked` signal goes low to keep the receiver PCS under reset. Altera recommends specific transceiver reset sequences to ensure proper device operation.

For more information about receiver CDR and lock modes, refer to the “Receiver Channel Datapath” section of *Transceiver Architecture in Stratix IV Devices* chapter.

- **Ports /Cal Blk screen**—The calibration block is required so it is always enabled. Select the options shown in [Figure 2-6](#).

Figure 2-6. FC4G Instance Settings (Ports/Cal Blk Screen)



- RX Analog screen—Select the options shown in [Figure 2-7](#).

Figure 2-7. FC4G Instance Settings (RxAnalog Screen)

The screenshot displays the MegaWizard Plug-In Manager (ALTGX) interface for configuring the RX Analog settings of an FC4G instance. The window title is "MegaWizard Plug-In Manager [page 6 of 14]". The main menu includes "Parameter Settings", "EDA", and "Summary". The "RX Analog" tab is selected, with other tabs like "General", "PLL/Ports", "Ports/Cal Blk", "TX Analog", "Reconfig", "Lpbk", "Basic/8B10B", "Word Aligner", and "Rate match/Byte order" visible.

The central block diagram shows the "FC4G_TXRX" block with various input and output signals. The input signals on the left include: rx_datain[0], tx_datain[39..0], pll_inclk, rx_digitalreset[0], rx_analogreset[0], tx_digitalreset[0], gxb_powerdown[0], pll_powerdown[0], cal_bk_clk, rx_bitslip[0], reconfig_clk, and reconfig_togxb[3..0]. The output signals on the right include: rx_dataout[39..0], tx_dataout[0], rx_freqlocked[0], rx_clkout[0], tx_clkout[0], and reconfig_fromgxb[16..0].

The "Receiver Analog Settings" panel on the right is titled "Able to implement the requested GXB". It contains the following settings:

- Enable static equalizer control
- A signal strength indicator with a scale from 0 (Low) to High.
- What is the DC gain?
- What is the Receiver Common Mode Voltage (RX Vcm)? V
- Force signal detection
- What is the signal detect and signal loss threshold?
- Use external Receiver termination
- What is the Receiver termination resistance? Ohms


At the bottom of the panel are buttons for "Cancel", "< Back", "Next >", and "Finish".

Below the block diagram, the following protocol and configuration details are listed:

```

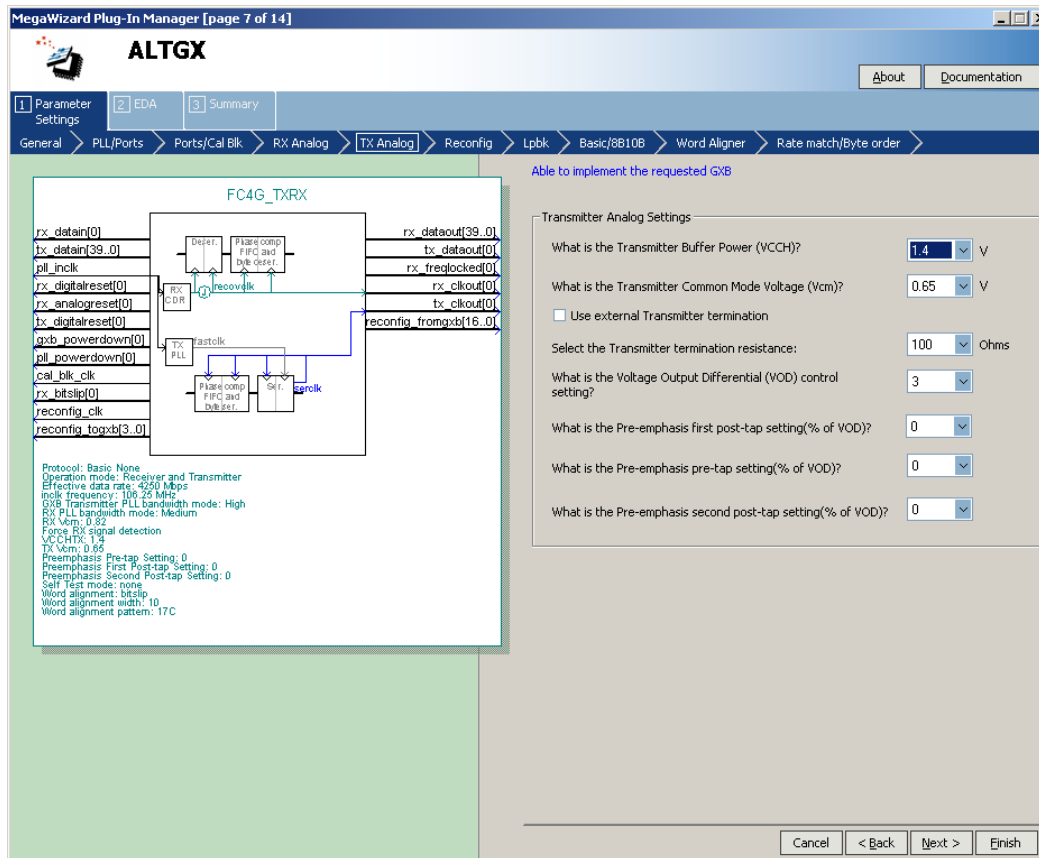
Protocol: Basic None
Operation mode: Receiver and Transmitter
Effective data rate: 4250 Mbps
Inclk frequency: 10.26 MHz
GXB Transmitter PLL bandwidth mode: High
RX PLL bandwidth mode: Medium
RX Vcm: 0.82
Force RX signal detection
VDCHTX: 1.4
TX Vcm: 0.86
Preemphasis Pre-tap Setting: 0
Preemphasis First Post-tap Setting: 0
Preemphasis Second Post-tap Setting: 0
Self Test mode: none
Word alignment: bitslip
Word alignment width: 10
Word alignment pattern: 17C

```

 For a description of the individual options, refer to the *ALTGX Megafunction User Guide for Stratix IV Devices* chapter.

- **TX Analog** screen—Select the output differential voltage and common mode voltage values that meet the fibre channel protocol specification. If you intend to transmit data through faulty interconnects, select the pre-emphasis settings shown in Figure 2-8.

Figure 2-8. FC4G Instance Settings (TX Analog Screen)



For more information about pre-emphasis settings, refer to the *DC and Switching Characteristics for Stratix IV Devices* chapter.

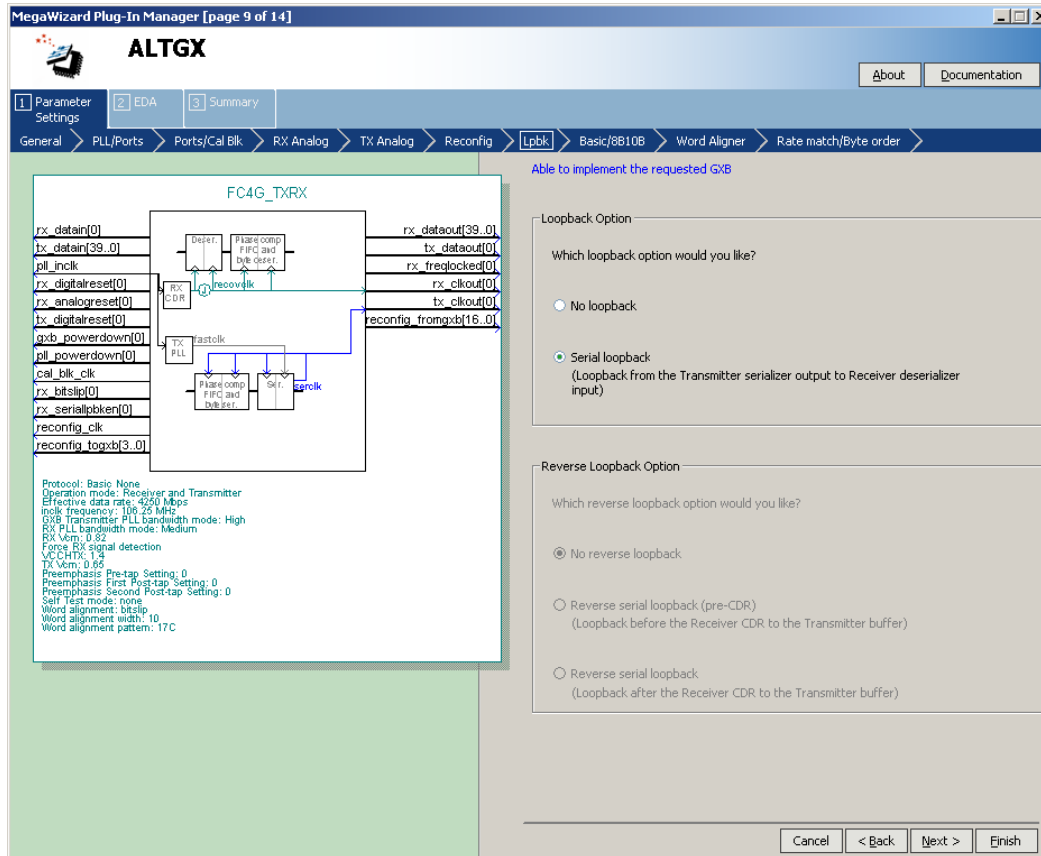
- Reconfig screen**—Set the starting channel number to 0. Because offset cancellation is required for receiver channels, the **Offset Cancellation for Receiver Channels** option is automatically enabled. Ensure that you connect the `reconfig_fromgxb` and `reconfig_togxb` ports with the dynamic reconfiguration controller (Figure 2-9).

Figure 2-9. FC4G Instance Settings (Reconfig Screen)

For more information about the starting channel numbers, refer to the “Logical Channel Addressing” section of the *Stratix IV Dynamic Reconfiguration* chapter.

- Lpbk screen—The serial loopback option is enabled, as shown in Figure 2-10.

Figure 2-10. FC4G Instance Settings (Lpbk Screen)



- **Basic/8B10B** screen—The Basic/8B10B screen is shown in Figure 2-11. The 8B/10B encoder is not compatible with the fibre channel protocol application; therefore, this option is unchecked.

Figure 2-11. FC4G Instance Settings (Basic 8B/10B)

The screenshot shows the MegaWizard Plug-In Manager window for the ALTGX instance. The 'Basic/8B10B' tab is selected, and the 'FC4G_TXRX' block diagram is visible on the left. The diagram shows the internal components of the transceiver, including the RX CDR, TX PLL, Phase Comp FIFO and byte deser., and Ser. blocks. The configuration options on the right are as follows:

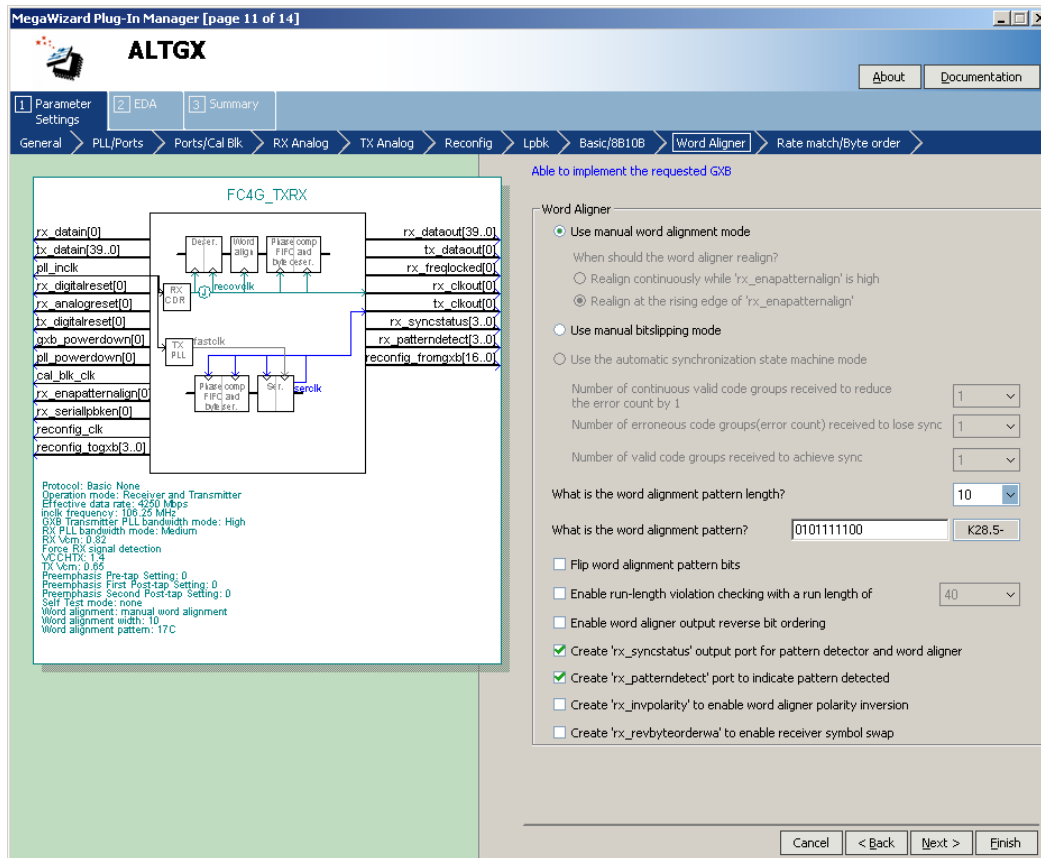
ABLE TO IMPLEMENT THE REQUESTED GXB

- Enable low latency PCS mode
Note: This option disables all the PCS blocks except the Phase Comp FIFO and the optional byte serializer
- 8b10b Decoder/Encoder**
 - Enable 8b10b decoder/encoder
 - Create 'tx_forcedisp' to enable force disparity and use 'tx_dispsval' to code up the incoming word using positive or negative disparity
 - Create 'rx_ctrlrdetect' port to indicate 8b10b decoder has detected a control code
 - Create 'rx_errdetect' port to indicate 8b10b decoder has detected an error code
 - Create 'rx_dispers' port to indicate 8b10b decoder has detected a disparity error
 - Create 'rx_runningdisp' port to indicate the current running disparity of the 8b10b decoded byte
- Flip Receiver output data bits
- Flip Transmitter input data bits
- Enable Transmitter bit reversal
- Create 'tx_invpolarity' to allow Transmitter polarity inversion

At the bottom of the window, there are buttons for 'Cancel', '< Back', 'Next >', and 'Finish'.

- Word Aligner screen**—The fibre channel protocol requires that you use K28.5 to align the byte boundary. In the **What is the word alignment pattern?** option, set one of the 10-bit disparity values to K28.5. The word aligner automatically detects when the other disparity value is received.

Figure 2-12. FC4G Instance Settings (Word Aligner Screen)



- Select the `rx_patterndetect` and `rx_syncstatus` signals. The `rx_patterndetect` signal indicates whenever the word alignment pattern is detected in the word boundary.
- Click **Finish** to exit the ALTGX MegaWizard Plug-In Manager.

Create the Transceiver Instance for an FC1G Configuration (Channel 1)

Creating the instance for FC1G is very similar to that of the FC4G configuration, with the following changes:

- **General screen**—Set the values shown in [Figure 2-13](#).

Figure 2-13. FC1G Instance (Channel 1) Settings (General Screen)

The screenshot displays the MegaWizard Plug-In Manager interface for configuring an FC1G instance. The left pane shows a schematic diagram of the FC1G_TXRX block with various input and output pins. The right pane shows the configuration settings for the instance, categorized into General, Input Data, and Output Data sections.

General Settings:

- Which device speed grade will you be using? 2
- Which protocol will you be using? Basic
- Which subprotocol will you be using? None
- Enforce default settings for this protocol
- What is the operation mode? Receiver and Transmitter
- What is the number of channels? 1
- What is the deserializer block width?
 - Single (valid data rates: 600 Mbps - 3.750 Gbps)
 - Double (valid data rates: > 1.000 Gbps)
- What is the channel width? 40 bits

Input Data Settings:

- What would you like to base the setting on? Data Rate
- What is the effective data rate? 1062.5 Mbps
- What is the input clock frequency? 53.125 MHz
- Specify base data rate 1062.50 Mbps

Output Data Settings:

- What is the output clock frequency? 53.125 MHz
- What is the output data rate? 1062.50 Mbps

Buttons at the bottom: Cancel, < Back, Next >, Finish

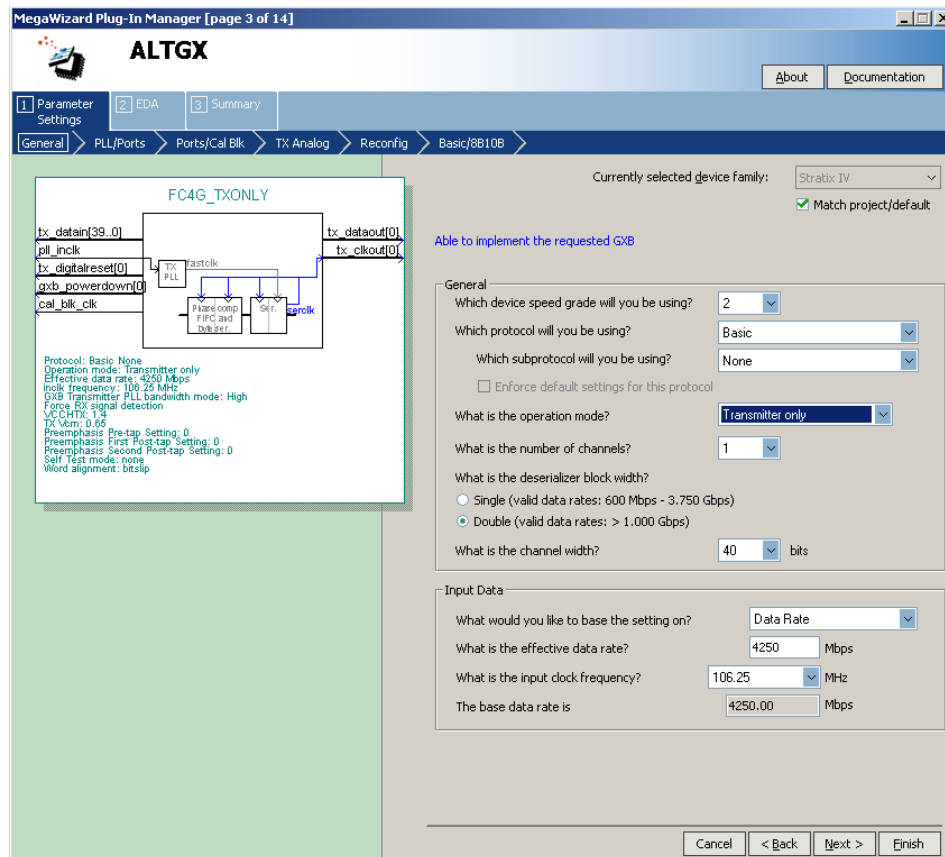
- **Reconfig screen**—Set the starting channel number to 4.

Create the Instance for an FC4G Configuration—Transmitter Only Mode (Channel 2)

This configuration is similar to the channel 0 configuration, with the following changes:

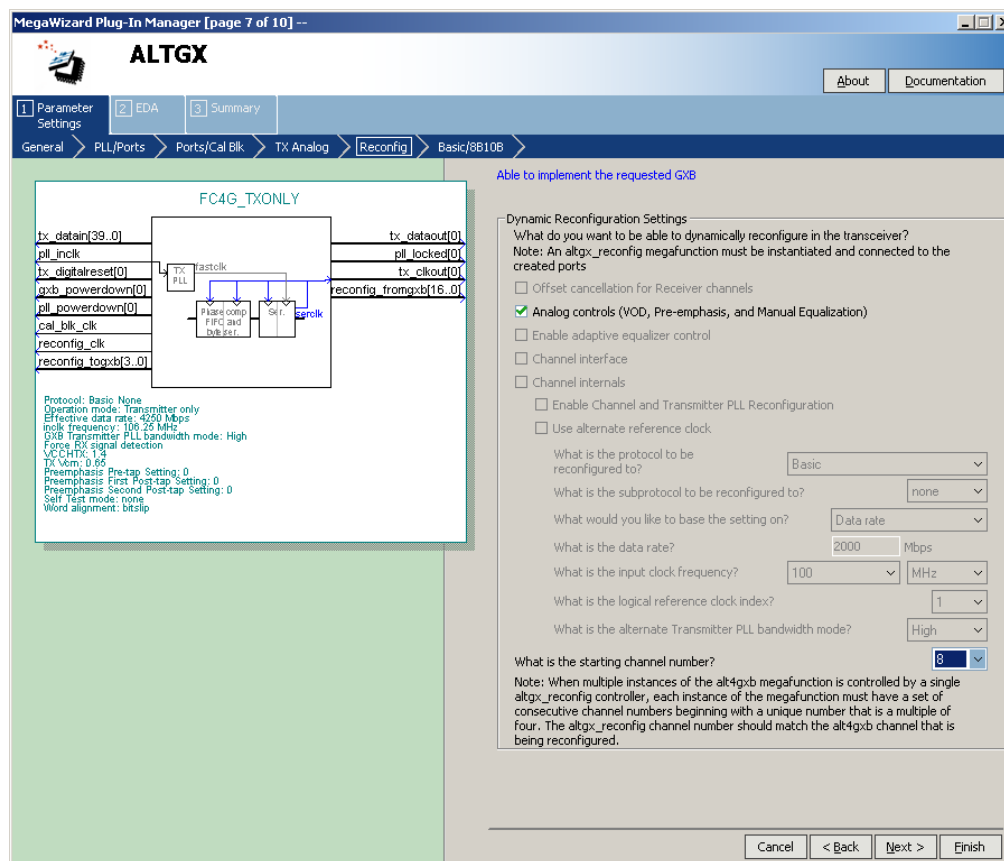
- Set the operation mode to **Transmitter Only**, as shown in Figure 2-14. Because this is a **Transmitter Only** instance, all the options relevant to the receiver are not available in the ALTGX MegaWizard Plug-In Manager.

Figure 2-14. FC4G_TXONLY Instance (Channel 1) Settings (General Screen)



- **Reconfig screen**—Set the starting channel number to 8. Select the **Analog controls** option even if you do not intend to dynamically reconfigure the PMA controls, as shown in Figure 2-15. Selecting this option is required for this example scenario because:
 - For a **Transmitter Only** instance, offset cancellation is not available; therefore, the `reconfig_fromgxb` and `reconfig_togxb` ports are not available.
 - The other two instances (containing a receiver channel) have these ports available because offset cancellation is automatically enabled.
 - If one transceiver instance has the `reconfig_fromgxb` and `reconfig_togxb` ports enabled, the Quartus II software requires the other transceiver instances to have these ports enabled to combine them in the same transceiver block. Therefore, for this **Transmitter Only** instance, the **Analog options...** must be selected.

Figure 2-15. FC4G_TXONLY Instance (Reconfig) Screen

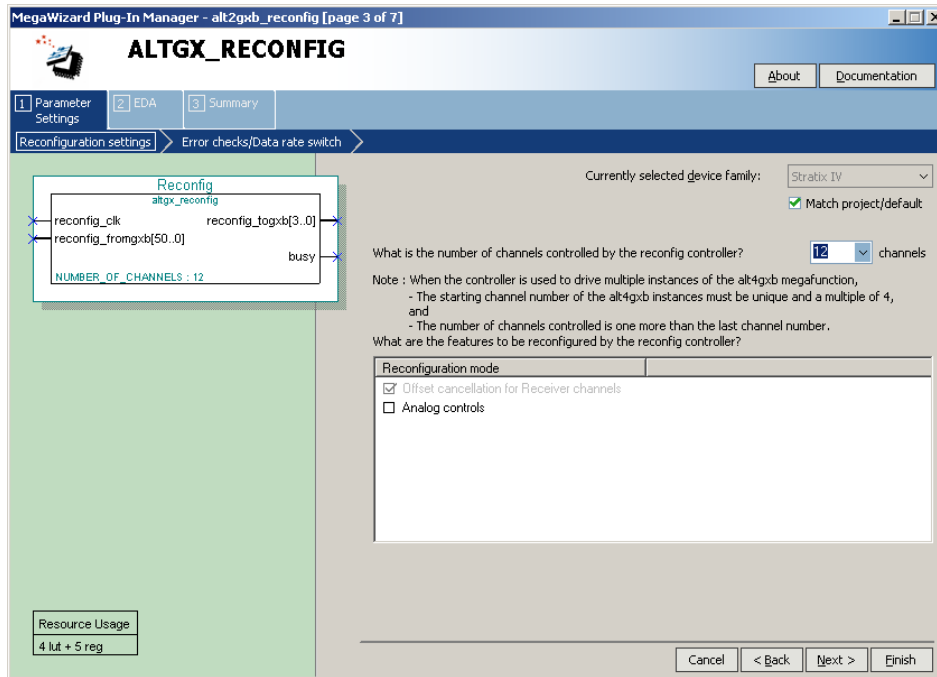


- For more information about the requirements to combine multiple transceiver instances, refer to the “Combining Transceiver Instances in Multiple Transceiver Blocks” section in the *Configuring Multiple Protocols and Data Rates in Stratix IV Devices* chapter.

Create the Dynamic Reconfiguration Controller (ALTGX_Reconfig) Instance

This section only describes the relevant options that must be set to implement the application.

Figure 2–16. ALTGX_Reconfig Settings (Reconfiguration Settings Screen)



For more information, refer to the *Dynamic Reconfiguration in Stratix IV Devices* chapter.

Figure 2–16 shows the options that you must set (assuming that you do not require dynamic reconfiguration of the PMA controls in the transceiver channels).

For more information about selecting the **Number of Channels** option, refer to the “Total Number of Channels Option in the ALTGX_RECONFIG Instance” section in the *Dynamic Reconfiguration in Stratix IV Devices* chapter.

Connect the following:

- `reconfig_fromgxb[16:0]` of the ALTGX_RECONFIG instance to the FC4G instance (channel0)
- `reconfig_fromgxb[33:17]` to the FC1G instance (channel1)
- `reconfig_fromgxb[50:34]` to the FC4G **Transmitter Only** instance (channel2)
- `reconfig_togxb[3:0]` of the ALTGX_RECONFIG instance to all three transceiver instances

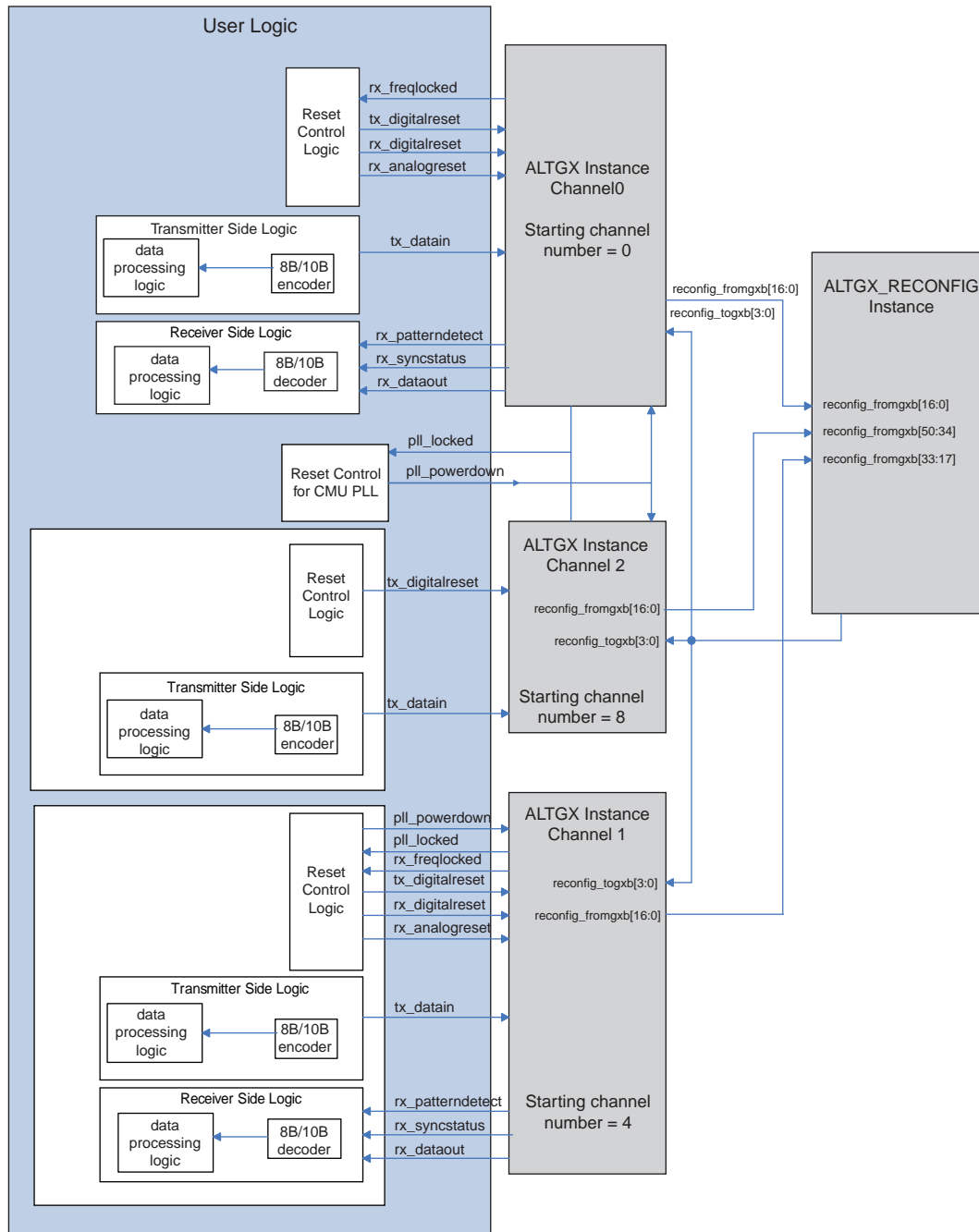
Create Reset Logic to Control the FPGA Fabric and Transceivers

The design requires independent control on each channel. Altera recommends creating independent reset control logic for each channel.

In this design, channel 0 and channel 2 share the same CMU PLL (because they are configured at the same data rate) and channel 1 uses the second CMU PLL. When you create a **Transmitter Only** or **Receiver and Transmitter** instance, the ALTGX MegaWizard Plug-In Manager provides a `p11_powerdown` signal to reset the CMU PLL that provides clocks to the transmitter channel. In this design example, because channels 0 and 2 share the same CMU PLL, drive the `p11_powerdown` port of channel 0 and channel 2 in the ALTGX instance from the same logic.

Channels 0, 1, and 2 have separate rx_digitalreset, rx_analogreset, and tx_digitalreset signals. Figure 2-17 shows the interface between the three transceiver instances and the FPGA fabric.

Figure 2-17. Transceiver—FPGA Fabric Interface



Create Data Processing and Other User Logic

For this example, you must implement the 8B/10B encoder and decoder in the FPGA fabric. [Figure 2-17 on page 2-35](#) shows the logic on the transmitter and receiver side and the system logic controls for all channels in the FPGA fabric. This block diagram is a representation of a typical system and may not exactly show the different blocks in a practical application. Interface all the logic blocks with the transceiver.

If you would like to add SignalTap for verification, first complete synthesis, then add the transceiver-FPGA fabric or other user logic signals in SignalTap. Lastly, compile the design to generate the .sof.

Phase 3—Compilation

Assign pins for the input and output signals in your design. The Quartus II software versions 8.1 and earlier do not allow pin assignments for the Stratix IV GX device.

Set the OCT values for the transceiver serial pins, add timing constraints for the clocks and data paths in your logic, then compile the design.

Phase 4—Simulating the Design

To simulate the design, follow the steps outlined in [“Functional Simulation” on page 2-12](#).

Document Revision History

[Table 2-7](#) lists the revision history for this chapter.

Table 2-7. Document Revision History (Part 1 of 2)

Date	Version	Changes
February 2011	4.1	<ul style="list-style-type: none"> ■ Applied new template. ■ Updated chapter title. ■ Minor text edits
November 2009	4.0	<ul style="list-style-type: none"> ■ Added Table 2-3, Table 2-4, Table 2-5, and Table 2-6. ■ Minor text edits.
June 2009	3.1	<ul style="list-style-type: none"> ■ Updated the “Introduction”, “Power Supplies”, “Transceiver Configuration”, “Clocking”, “Create Transceiver Instances”, “Create Dynamic Reconfiguration Controller Instances”, “Create Data Processing and Other User Logic”, “Functional Simulation” sections. ■ Added the “Board Design Requirements”, “Gear Boxing Logic”, “Guidelines to Debug the FPGA Logic and the Transceiver Interface”, and “Guidelines to Debug System Level Issues” sections. ■ Added introductory sentences to improve search ability.
March 2009	3.0	<ul style="list-style-type: none"> ■ Add “Power Supplies” on page 2-6 ■ Updated “Dynamic Reconfiguration” on page 2-4 ■ Text edits

Table 2-7. Document Revision History (Part 2 of 2)

Date	Version	Changes
November 2008	2.0	<ul style="list-style-type: none"> ■ Added “Transceiver Configuration” on page 2-3 ■ Added “Create Dynamic Reconfiguration Controller Instances” on page 2-8 ■ “Dynamic Reconfiguration” on page 2-15 ■ Updated “Create the Instance for an FC4G Configuration—Transmitter Only Mode (Channel 2)” on page 2-28 ■ Added “Create the Dynamic Reconfiguration Controller (ALTGX_Reconfig) Instance” on page 2-30 ■ Updated Figure 2-1, Figure 2-4, Figure 2-5, Figure 2-6, Figure 2-7, Figure 2-8, Figure 2-10, Figure 2-11, Figure 2-12, Figure 2-13, and Figure 2-14 ■ Added Figure 2-9, Figure 2-15, and Figure 2-16
May 2008	1.0	Initial release.

