Dynamic Reconfiguration of Transceiver Channels Using Multiple PLLs in Stratix IV Devices

AN-607-1.2 Application Note

This application note describes how you can dynamically reconfigure your Stratix® IV transceiver channels using the multiple phase-locked loop (PLL) dynamic reconfiguration feature. This feature allows you to reconfigure a transceiver channel to switch among PLLs (CMU PLLs) within the transceiver block and the PLLs (CMU and [auxiliary transmit] ATX PLLs) located outside the transceiver block. Using this feature, you can independently switch a transceiver channel to at least four independent and distinct data rates typically required by universal front end (UFE) applications.

- The CMU PLL-based dynamic reconfiguration feature allows you to reconfigure the transmitter side of a transceiver channel independently by either switching to one of the two clock multiplier unit PLLs (CMU PLLs) within the transceiver block or by reconfiguring a CMU PLL to the data rate you require. However, with this CMU PLL reconfiguration approach, the other channels within the transceiver block listening to the same CMU PLL are also affected. Dynamically reconfiguring using multiple PLLs overcomes this issue.
- The focus of this application note is about reconfiguring the transmitter side of the transceiver channel. Because each receiver channel has a dedicated clock and data recovery (CDR) that can be reconfigured to any supported data rate, the reconfiguration of the receiver side is not affected by any PLL resource.

This application note assumes prior knowledge of the dynamic reconfiguration feature and the Quartus® II software flow used to implement this feature.

For more information about the dynamic reconfiguration feature, refer to the "Memory Initialization File (**.mif**)" and "Quartus II MegaWizard Plug-In Manager Interfaces to Support Dynamic Reconfiguration" sections in the following handbooks:

- *Stratix IV Dynamic Reconfiguration* chapter in volume 2 of the *Stratix IV Device Handbook*
- *HardCopy IV GX Dynamic Reconfiguration* chapter in volume 3 of the *HardCopy IV Device Handbook*.

The following sections describe this feature using an example design.

101 Innovation Drive
San Jose, CA 95134
www.altera.com

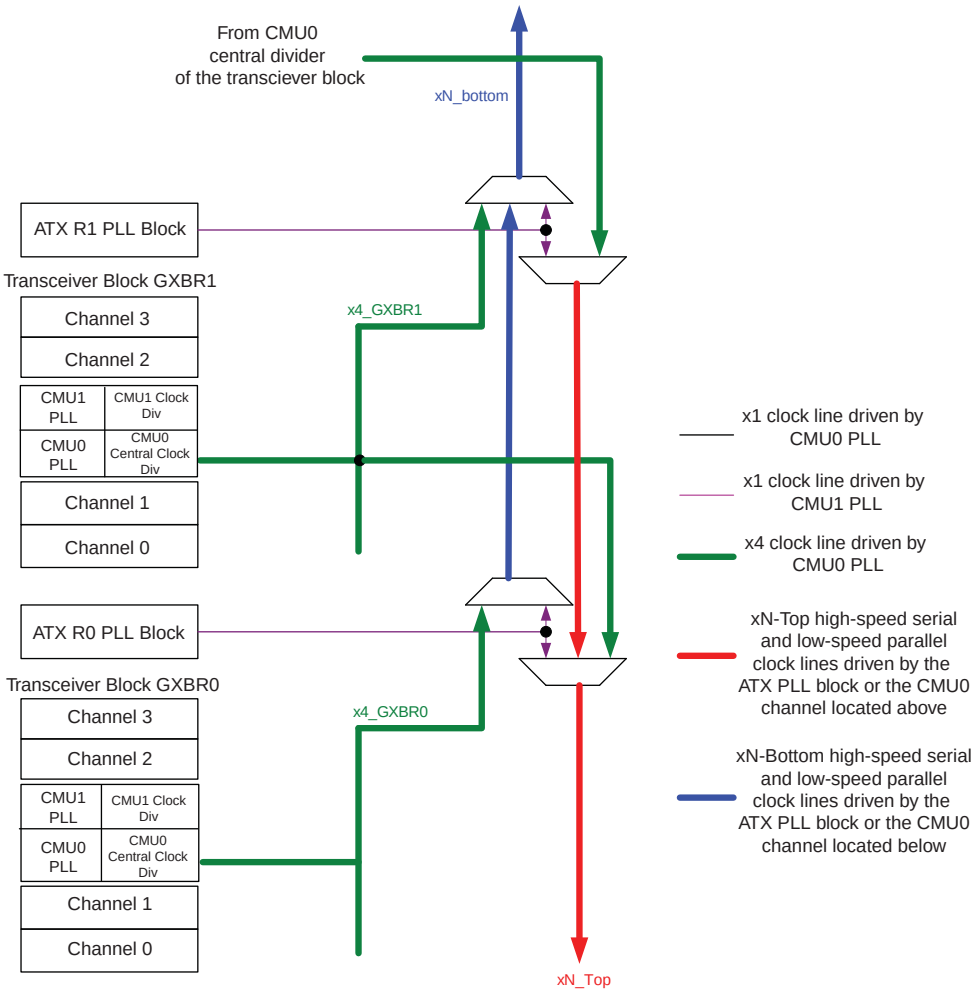
Multiple PLL Dynamic Reconfiguration Overview

To use the dynamic reconfiguration feature with multiple PLLs, you must understand the different sources and clock lines that generate the high-speed clocks to the transmitter channels.

Each side of the Stratix IV GX and GT devices and HardCopy IV devices has two high-speed ×N clock lines (×N top and ×N bottom) that span the entire side. The drivers of the ×N line are the CMU0 central clock divider (driven by the CMU1 PLL or CMU0 PLL) and the ATX PLL. Therefore, a transmitter channel can acquire only one high-speed clock either from the CMU PLL or the ATX PLL above its transceiver block (because only one can drive the ×N top line reaching the channel). The same rule applies for a transmitter channel acquiring its high-speed clock from below its transceiver block.

For example, as shown in Figure 1, the transmitter channels in GXBR1 can get their high-speed clock from either the ATX R0 PLL block or the CMU0 central divider block (driven by CMU0 or CMU1) in GXBR0. The channels in GXBR1 can also acquire their high-speed clock from either the ATX R1 PLL or from the CMU0 central divider block in GXBR2 (GXBR2 is not shown in Figure 1).

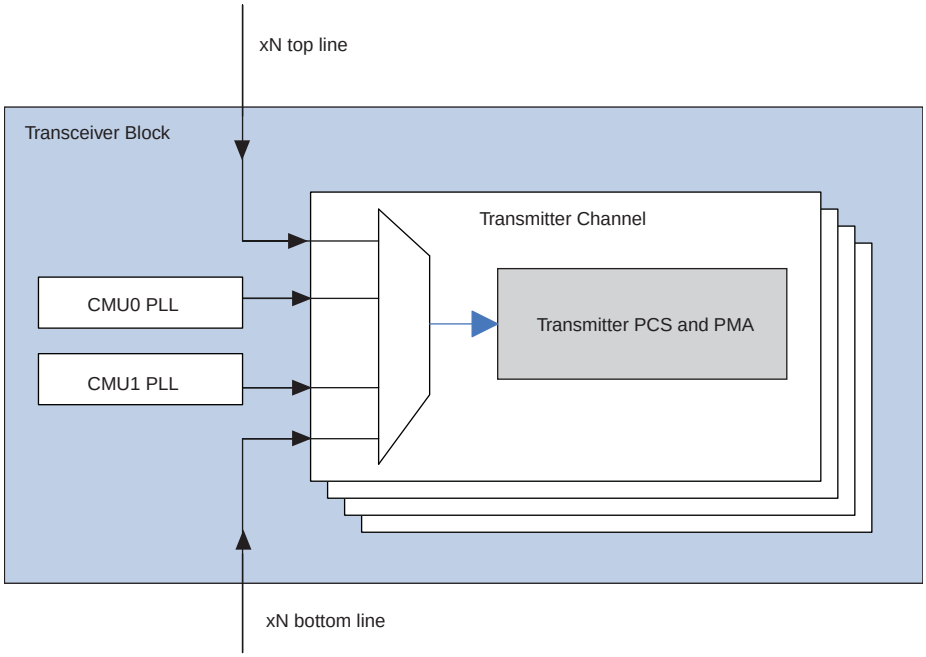
Figure 1. ×N Line Connections



As shown in Figure 1, a transceiver block in the center gets the highest number of transmitter PLLs—two CMU PLLs within the same transceiver block—one from above and one from below the transceiver block (refer to Figure 2).

The central clock divider (in the CMU PLL or ATX PLL) is always required to drive the ×N line.

Figure 2. Number of PLLs Connected to a Transmitter Channel



Functional Modes that Support Multiple PLL Switching

The dynamic reconfiguration feature using multiple PLLs is supported in the following functional modes:

- All non-bonded functional modes; for example, SONET, GIGE, and Basic ×1
- Basic (PMA Direct) ×N and ×1 modes

The multiple PLL dynamic reconfiguration feature is not supported in bonded functional modes, such as XAUI, Basic ×4 and ×8, and PCI Express (PIPE) (PCIe) ×4 and ×8.

Enabling the Multiple PLL Dynamic Reconfiguration Feature

You can enable the multiple PLL dynamic reconfiguration feature in the ALTGX MegaWizard™ Plug-In Manager by selecting the **Use additional CMU/ATX Transmitter PLLs from outside the Transceiver Block** option in the **Reconfiguration Settings** tab. A summary of the settings used in this example design is shown in Table 5 on page 10.

Select the number of additional PLLs (index 0 to 3) from the **How many additional PLLs used?** option. The maximum number of PLLs that you can select is 4, as shown in Figure 2. Set up the parameters for the main and additional PLLs in the subsequent screens.

The two key options in the PLL screens on the **Reconfiguration Settings** tab are the following:

- Option 1: **Use central clock divider to drive the ×4/×N lines**—select this option if the PLL must provide high-speed clocks to the transmitter channel that resides outside the transceiver block.
- Option 2: **Use Auxiliary Transmit (ATX) PLL**—select this option to enable the ATX PLL to provide clocks to the transmitter channel(s) in the instance. Because the ATX PLL always requires a central clock divider, you must select Option 1 to enable this option.

These two options impact the logical addressing scheme. If you select a transceiver instance with the (Option 1) **Use central clock divider to drive the ×4/×N lines** option enabled using the CMU PLL (Option 2 is not selected), you must assign a separate logical channel address in the next multiple of four for the CMU PLL. During run time, use this logical channel address to reconfigure this CMU PLL. For example, if your starting channel number is 8, and if the number of channels in the instance is 1, when you select Option1, the logical address of the channel and the CMU PLL are 8 and 12, respectively.

You cannot reconfigure the ATX PLL. Therefore, if you use the ATX PLL, a separate logical channel address for the ATX PLL is not required.

If you do not select the (Option 1) **Use central clock divider to drive the ×4/×N lines** option, the PLL must be the CMU PLL WITHIN the transceiver block (it cannot be the ATX PLL). In this case, a separate logical channel address for the PLL is not required. If you use more than one channel in the instance, you can use the logical address of one of the channels as the address of the CMU PLL within the transceiver block. For example, if your starting channel number is 0, and the number of channels in the instance is 3, and you did not select Option 1, the logical channel addresses of the three channels are 0, 1, and 2, respectively. You can use 0, 1, or 2 to address the CMU PLL within the transceiver block.

The other option that you must set is the **PLL logical reference index** option using the **What is the PLL logical reference index?** option in each PLL tab. This index value must be consistent across the multiple instances that share the same PLL.

The logical channel addressing scheme and the PLL logical reference index are described further using the following example design.

Multiple PLL Dynamic Reconfiguration Using an Example Design

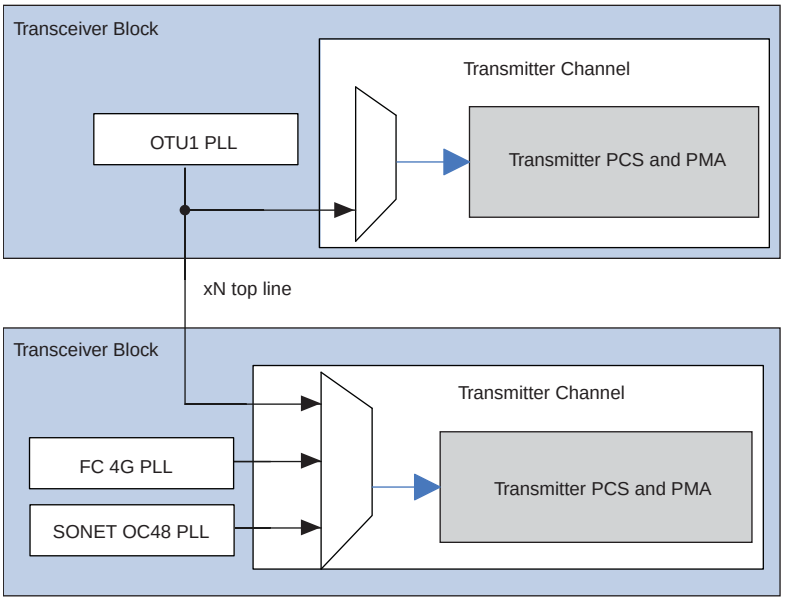
The following example design has these requirements:

- One channel in OTU1 with a 2.666 Gbps data rate.
- One channel that switches between Fibre channel 4G (4.25 Gbps), SONET OC48 (2.488 Gbps), and OTU1 (2.666 Gbps) data rates. To meet the second design requirement, three PLLs are required to provide clocks to this transmitter channel.
- These two channels must be placed in different transceiver blocks.

Also, the data path interface to support these protocols is different. For example, SONET OC48 requires a 16-bit data path interface while the Fibre channel 4G at 4.25 Gbps requires a 40-bit data path (you cannot use 8B/10B in the hard physical coding sublayer [PCS] for Fibre channel because it is not compliant). You can dynamically change the channel PCS using channel reconfiguration and TX PLL select mode.

Figure 3 shows a possible placement to achieve this design fit.

Figure 3. Example Design Placement Scheme



Use the settings in the following sections to set up the instances to implement the dynamic reconfiguration feature using the multiple PLLs scheme shown in Figure 3. Only the parameters that are relevant for this feature are specified. The files generated by the MegaWizard Plug-In Manager that correspond to these instances are also shown. Three ALTGX instances are required for this example design—Fibre channel 4G, SONET OC48 (only for **.mif** generation, as described in "SONET OC48 Instance (ALTGX_OC48.v)" on page 9), and OTU1 instance.

In addition to the transceiver parameters, to ensure that the dynamic reconfiguration feature using multiple PLLs operates correctly, the PLL logical index and input reference clocks must be consistent across the multiple instances.

Table 1 lists the input clock and logical reference index of the different PLLs.

Table 1. Input Clock and Logical PLL Reference Index

PLL	Logical Reference Index	Input Reference Clock Index
FC 4G PLL	0	0
OTU1 PLL	2	2
SONET OC48 PLL	1	1

OTU1 Instance (ALTGX_OTUDEFAULT.v)

Table 2 lists the **General** tab settings for the OTU1 instance.

Table 2. General Tab Settings for the OTU1 Instance

Option	Setting
Which protocol will you be using?	Basic
Which sub protocol will you be using?	none
Operation mode?	Receiver and Transmitter
Number of channels?	1
What is the deserializer block width?	Single
What is the channel width?	16
What would you like to base the setting on?	data rate
Effective data rate?	2666 Mbps
Input reference clock frequency?	166.625 (selected for illustration)

Reconfiguration Settings Tab

In the **Reconfiguration Settings** tab, select the following options:

- **Enable Channel and Transmitter PLL reconfiguration**
- **Channel interface**
- **Use additional CMU/ATX Transmitter PLLs....**
- **Number of additional PLLs?** Set to **0**. The channel listening to OTU1 does not require multiple PLLs, as noted in requirement 1 listed above.
- **How many input clocks are used?** Set to **3**. Even though you do not require three clocks for the OTU1 instance, this option is set to 3 in the example design to make the input reference clocking consistent with other instances.
- **Starting channel number** Set to **0**.

Main PLL Tab

In the **Main PLL** tab, select the following options:

- **Use central clock…**. This is required because this PLL is shared with the Fibre channel instance. Refer to Figure 3.
- **PLL logical reference index** set to either **2** or **3**. For this example design, **2** is selected.

When you select the **Use central clock…** option, the available PLL logical reference index options are 2 or 3.

- **What is the selected input clock source for the PLL?** Set to **2**.

The ALTGX MegaWizard Plug-In Manager provides the logical address of the channel and the PLL (in text) in each PLL tab. In this case, the logical address of the channel is **0** because the starting channel number used is 0 and the logical address of the PLL is 4. (This text is at the bottom of the tab). Therefore, the starting channel number of the Fibre channel instance must be set to **8**.

Click **Finish** to complete the MegaWizard Plug-In Manager setup.

Fibre Channel Instance (ALTGX_FC4GDEFAULT.v)

Table 3 lists the **General** tab settings for the Fibre channel instance.

Table 3. General Tab Settings for the Fibre Channel Instance

Option	Setting
Which protocol will you be using?	Basic
Which sub protocol will you be using?	none
Operation mode?	Receiver and Transmitter
Number of channels?	1
What is the deserializer block width?	double
What is the channel width?	40
What would you like to base the setting on?	data rate
Effective data rate?	4250 Mbps
Input reference clock frequency?	106.25 (selected for illustration)

Reconfiguration Settings Tab

In the **Reconfiguration Settings** tab, select the following options:

- **Enable Channel and Transmitter PLL reconfiguration**
- **Channel interface**
- **Use additional CMU/ATX Transmitter PLLs....**
- **Number of additional PLLs?** Set to **2**.
- **How many input clocks are used?** Set to **3**.
- **Starting channel number** Set to **8**.

Main PLL Tab

In the **Main PLL** tab:

- Do NOT select the **Use central clock…** option. As shown in Figure 3, the FC 4G PLL is not shared with any other transmitter channels outside the transceiver block. Therefore, this option is not required.

Do select the following options:

- **PLL logical reference index** set to either **0** or **1**. For this example design, **0** is selected.
- **What is the selected input clock source for the PLL?** Set to **0**.

PLL1 Tab

This example design assumes that PLL1 is SONET OC48.

In the **PLL1** tab:

- Do NOT select the **Use central clock…** option. As shown in Figure 3, the SONET OC48 PLL does not provide clocks to transmitter channels outside the transceiver block.

Select the following options:

- **PLL logical reference index** Set to **1** because 0 is already selected for the main PLL.
- **What is the selected input clock source for the PLL?** Set to **1**.
- **Protocol** Set to **SONET OC 48**.
- **Input clock frequency** Set to **155.52**.

PLL2 Tab

This example design assumes that PLL2 is OTU1 PLL.

In the **PLL2** tab, select the following options:

- **Use central clock…**. As shown in Figure 3, the OTU1 PLL also provides clock to the FC 4G instance outside the transceiver block. Therefore, you must select this option.
- **PLL logical reference index** Set to **2** because 2 is already selected for the first OTU1 instance.

  The logical reference index of the PLLs that are shared across instances must be the same. For the summary of the PLL index and input reference clock settings, refer to the Table 1 on page 6.

- **What is the selected input clock source for the PLL?** Set to **2**.
- **Protocol** Set to **Basic** (for OTU1).
- **Data rate** Set to **2666** (for OTU1).
- **Input clock frequency** Set to **166.625** (for OTU1).

The logical address of the channel is **8** and the logical address of PLL2 is **12** (the next multiple of four, as described in "Multiple PLL Dynamic Reconfiguration Overview" on page 2).

Click **Finish** to complete the MegaWizard Plug-In Manager setup.

SONET OC48 Instance (ALTGX_OC48.v)

This instance is required only to generate **.mif** files. As stated in requirement 2 of the example design (refer to "Multiple PLL Dynamic Reconfiguration Using an Example Design" on page 5), the default configuration is FC4G and OC48 is one of the configurations to switch to. Therefore, this instance is not required in the main design.

Table 4 lists the **General** tab settings for the OC48 instance.

Table 4. General Tab Settings for the OC48 Instance

Option	Setting
Which protocol will you be using?	SONET
Which sub protocol will you be using?	OC48
Operation mode?	Receiver and Transmitter
Number of channels?	1
What is the deserializer block width?	single
What is the channel width?	16
What would you like to base the setting on?	data rate
Effective data rate?	2488 Mbps
Input reference clock frequency?	155.52 (selected for illustration)

Reconfiguration Settings Tab

In the **Reconfiguration Settings** tab, select the following options:

- **Enable Channel and Transmitter PLL reconfiguration**
- **Channel interface**
- **Use additional CMU/ATX Transmitter PLLs....**
- **Number of additional PLLs?** Set to **2**.
- **How many input clocks are used?** Set to **3**.
- **Starting channel number** Set to **8**.

Main PLL Tab

In the **Main PLL** tab:

- Do NOT select the **Use central clock…** option. As shown in Figure 3, the OC48 PLL is not shared with any other transmitter channels outside the transceiver block. Therefore, this option is not required.

Select the following options:

- **PLL logical reference index** Set to **1** because **0** is selected for the FC4G PLL.
- **What is the selected input clock source for the PLL?** Set to **0**.

PLL1 Tab

In the **PLL1** tab:

- Do NOT select the **Use central clock…** option. As shown in Figure 3, the FC PLL does not provide clocks to transmitter channels outside the transceiver block.

Select the following options:

- **PLL logical reference index** Set to **0** because 1 is already selected for the OC48 PLL.
- **What is the selected input clock source for the PLL?** Set to **0**.
- **Protocol** Set to **Basic**.
- **Data rate** Set to **4250**.
- **Input clock frequency** Set to **106.25**.

PLL2 Tab

This example design assumes that PLL2 is OTU1 PLL.

In the **PLL2** tab, select the following options:

- **Use central clock…**. As shown in Figure 3, the OTU1 PLL also provides clock to the FC 4G instance outside the transceiver block. Therefore, you must select this option.
- **PLL logical reference index** Set to **2** because 2 is already selected for the first OTU1 instance.

  The logical reference index of the PLLs that are shared across instances must be the same. For the summary of the PLL index and input reference clock settings, refer to the Table 1 on page 6.

- **What is the selected input clock source for the PLL?** Set to **2**.
- **Protocol** Set to **Basic** (for OTU1).
- **Data rate** Set to **2666** (for OTU1).
- **Input clock frequency** Set to **166.625** (for OTU1).

The logical address of the channel is **8** and the logical address of PLL2 is **12** (the next multiple of four, as described in "Multiple PLL Dynamic Reconfiguration Overview" on page 2).

Click **Finish** to complete the MegaWizard Plug-In Manager setup.

Table 5 lists the key multiple PLL dynamic reconfiguration settings for the three transceiver instances in this example design.

Table 5. Multiple PLL Dynamic Reconfiguration Settings ***(Note 1)***

Instance	Starting Channel Number	Logical PLL Addresses
ALTGX_OTU1DEFAULT	0	4 for the OTU1 PLL in this instance
ALTGX_FC4GDEFAULT	8	■ No separate address for the OC48 and FC4G PLLs ■ 12 for the OTU1 PLL in this instance

Note to Table 5:

(1) For the PLL logical index and reference clock index, refer to Table 1 on page 6.

The next step is to set up the ALTGX_RECONFIG instance.

ALTGX_RECONFIG Instance

In the **ALTGX_RECONFIG** tab, select the following options:

- **Channel and TX PLL select/reconfig** (because you must reconfigure the channels).
- **Analog Controls** (because typically you must change the PMA controls when you switch between different data rates to meet the signal integrity requirements).
- **What is the number of channels controlled by the reconfig controller?** Set to **16**. To find out the value for this option, use the following method:

  Match the number of `reconfig_fromgxb` ports in the **ALTGX_RECONFIG** instance to that of the ALTGX instances in the design. In this example design, the OTU1 instance has `reconfig_fromgxb[33:0]` and the FC4G instance has `reconfig_fromgxb[33:0]`. Therefore, in total you require `reconfig_fromgxb[67:0]` (34 + 34 = 68 bits).

Create the Reset Control Logic

Create the reset control logic to control the transceivers according to the protocol requirements. A simplified reset sequence is provided in this example design.

Create the Memory Initialization Files (.mifs)

You require three **.mifs** to switch the channel between FC 4G, SONET OC48, and OTU1. Using your selected instance, create the **.mifs**. The attached project contains **top.v** that was used to generate the **.mifs**.

Simulating the Multiple PLL Dynamic Reconfiguration Feature

The attached **simulation.zip** file contains all the required files to simulate the design.

Create a ModelSim® project and add the following Stratix IV libraries in your work directory:

sgate.v, **altera_mf.v**, **altera_primitives.v**, **220model.v**, **stratixiv_hssi_atoms.v**, **stratixiv_atoms.v**

The Quartus II version used for this design is 9.1. The library files are located in *\Quartus project directory\quartus\eda\sim_lib*.

Design Files

Add the following design files:

- **tb.v** (the testbench)—instantiates the top-level file that contains the transceiver instances, reset logic, reconfiguration controller, and reconfiguration control logic
- **top_sim.v**—top-level file for the design
- **reconfig_user_logic.v**—contains the control logic for the reconfiguration controller
- **reconfig.v**—ALTGX_RECONFIG instance
- **rst_controller.v**—reset controller logic

- **header.settings.v**—includes header settings
- **mem_RAM, mem_RAM1, mem_RAM2**—memory instances to hold **.mifs**

These **.mifs** are generated by compiling the **top.v** file that contains the FC4G, OC48, and OTU1 instances. For more information about generating **.mifs**, refer to the "Memory Initialization File (**.mif**)" section in the following handbooks:

- *Stratix IV Dynamic Reconfiguration* chapter in volume 2 of the *Stratix IV Device Handbook*
- *HardCopy IV GX Dynamic Reconfiguration* chapter in volume 3 of the *HardCopy IV Device Handbook*.

The following sections describe this feature using an example design

- **ALTGX_FC4GDEFAULT.v**—Fibre channel 4G transceiver instance
- **ALTGX_OTUDEFAULT.v**—OTU1 transceiver instance

Compiling and Running the Simulation

Add the design files, library files, and the three **.mifs** under the *reconfig_mif* directory to the project and compile the design.

Run the design using the following command (in Modelsim):

```
>> log -r /*; run 175 us
```

The test bench performs the following sequence:

1. The FC4G instance is configured to the OC48 data rate. In simulation, you can see this at 72 µs. The `channel_reconfig_done` signal is asserted and the frequency of `rx_clkout` and `tx_clkout` changes to 155.52 MHz.
2. The FC4G instance is next configured to the OTU1 data rate to show reconfiguration to the PLL outside the transceiver block. In simulation, the operation is complete at approximately 175 µs. The `channel_reconfig_done` signal is asserted and the frequency of `rx_clkout` and `tx_clkout` changes to 166.6 MHz.

The Quartus II software version 9.1 SP1 and earlier software versions have a known issue in simulation for multiple PLL dynamic reconfiguration. To workaround the issues, follow these two steps.

1. Reconfigure the OTU1 PLL (PLL2) using the CMU PLL reconfiguration. The logical channel address of this PLL is **12**. To observe this workaround in simulation, at 83 µs, `reconfig_mode_sel` changes to 100 (the TX PLL reconfiguration). For more information, refer to "PLL2 Tab" on page 8.
2. After the CMU PLL reconfiguration, reconfigure the channel using the channel and TX PLL select reconfiguration mode. The logical channel address corresponding to the channel is **8**.

If you use the multiple PLL reconfiguration feature, you must enable the `logical_tx_pll_sel[1:0]` port in the ALTGX_RECONFIG instances. During run time, set the `logical_tx_pll_sel` value. In the example design, the values for this port is set in **tb.v**.

The workaround mentioned in the first and second steps are only required for functional simulation.

To implement this design in the hardware, you must force the placement of the OTU1 PLL to enable the Quartus II software to use a single OTU1 PLL between the two transceiver instances. For more information about forcing PLL placement for the multiple PLL dynamic reconfiguration feature, refer to this Solution.

Because OTU1 PLL merging is forced in the compilation, when you run the design you can bypass the first step and directly go to the second step mentioned above.

The project contains a **SIM.do** file that contains all the relevant signals for observation. Because the data path width between the FC4G (40 bits) and the SONET OC48/OTU1 (16 bits) are different, to help observe the signals, the `tx_datainfull` and `rx_dataoutfull` buses are grouped for both SONET OC48 (`TX_SONET_OTU_bits15_0` and `rx_dataout_SONET_OTU`) and FC4G (`TX_FC_bits39_0`).

In simulation, only the ALTGX_FC4GDEFAULT instance is reconfigured because the OTU1 instance does not require reconfiguration, as explained in "Multiple PLL Dynamic Reconfiguration Using an Example Design" on page 5.

Document Revision History

Table 6 lists the revision history for this application note.

Table 6. Document Revision History

Date	Version	Changes
August 2010	1.2	■ Updated to improve searchability. ■ Included the HardCopy IV device family. ■ Minor text edits.
May 2010	1.1	Updated the "Compiling and Running the Simulation" section.
April 2010	1.0	Initial release.