Using the Stratix V Reconfiguration Controller to Perform Dynamic Reconfiguration

AN-664-1.0 Application Note

The ability to change the behavior and characteristic of a transceiver in real time provides enormous benefits to the overall operation of any digital system. The benefits of reconfiguration include the ability to adapt to changing electrical parameters, data rate requirements, and protocol requirements. The Stratix® V Reconfiguration Controller provides an easy-to-use interface that allows you to dynamically modify specific functionality of a Stratix V transceiver.

This application note describes the steps required to perform a transceiver reconfiguration using the Reconfiguration Controller. The following examples of reconfiguration are shown in this document:

- Changing the transmitter PMA settings
- Changing the transceiver data rate by selecting between two instantiated transmitter PLLs
- Changing the transceiver data rate by streaming a Memory Initialization File (**.mif**)

This application note is accompanied by a sample design that demonstrates the easy and efficient method by which the Reconfiguration Controller can dynamically change the functionality of a Stratix V PHY IP. Using either the built-in System Console tool or any other embedded controller, you can issue simple commands to the Reconfiguration Controller that allow any system to dynamically adapt to changing system requirements.

The sample design is created with the Quartus® II software version 12.0.

Design Overview

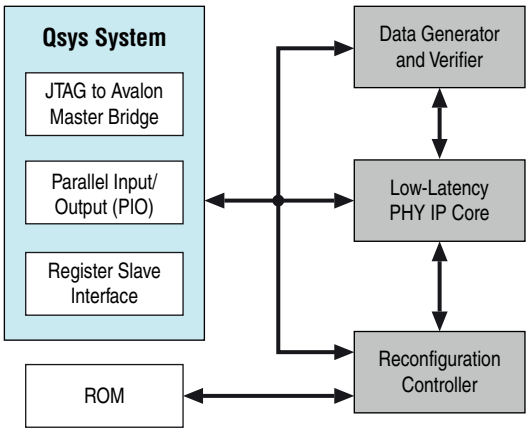
A sample design is used to demonstrate the functionality of the Reconfiguration Controller. It consists of the following modules:

- Data generators and verifiers
- Qsys system
- Stratix V Low Latency Transceiver
- Stratix V Reconfiguration Controller
- ROM containing the **.mif** for reconfiguration

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

Figure 1 shows a system level diagram of the reconfiguration sample design and how the different parts interface with each other.

Figure 1. System Diagram



The sample design is created such that you can dynamically reconfigure a Low Latency PHY IP core. The reconfiguration commands are controlled through the System Console tool that ships with the Quartus II software.

All design files mentioned in this application note are located in the appropriate **source** directory in the **sv_reconfiguration_design_example.zip** file.

Creating the Data Generators and Verifiers

The sample design contains data generators and verifiers. The data generator is capable of generating PRBS7, PRBS15, counter, and fixed value (40'hF0F0F0F0F0) data patterns. The data verifiers are used to verify the PRBS7 and PRBS15 data received by the design.

The data generators and verifiers are defined in the **data_generator_verifier.v** file located in the **original_design/source/** directory.

Creating the Qsys System

The reconfiguration sample design uses a simple Qsys system that consists of three components: the JTAG to Avalon Master Bridge, the External Slave Interface, and the PIO core. The Qsys system connects the three slave components together: the data generator and verifier, the Low Latency PHY IP core, and the Reconfiguration Controller.

To examine the Qsys system, perform the following steps:

1. Launch the Quartus II software
2. On the File menu, click **Open**
3. Browse and select the **console_interface.qsys** file located in the **original_design/source/** directory
4. Click **Open**

After clicking **Open**, the Qsys tool launches and shows all the components used in the design. The following components make up the Qsys system:

- The **JTAG to Avalon Master Bridge** component acts as the master in the design. It is the main communication channel between the System Console tool and the slaves in the design. The System Console tool issues Avalon® reads and writes to the Reconfiguration Controller to carry out reconfiguration of the PHY IP.
- The **External Slave Interface** component exports all required Avalon signals to the top-level design. With the Avalon signals exported, the Qsys system can interface with any Avalon-compliant component that resides outside the Qsys component library.
- The sample design uses two Avalon compliant components: the **Low Latency PHY IP core** and the **Reconfiguration Controller**. Two External Slave Interface components are therefore required. The two External Slave Interface components connect to the JTAG to Avalon Master Bridge.
- The **PIO** component monitors and drives individual status and control bits in the system. The PIO connects to the JTAG to Avalon Master Bridge.

The Stratix V PHY IP components are not supported in the Qsys tool in the Quartus II software version 12.0. To interface with a Stratix PHY IP in a Qsys system, you must use an external slave interface.

Figure 2 shows the component map of the Qsys system.

Figure 2. Qsys System

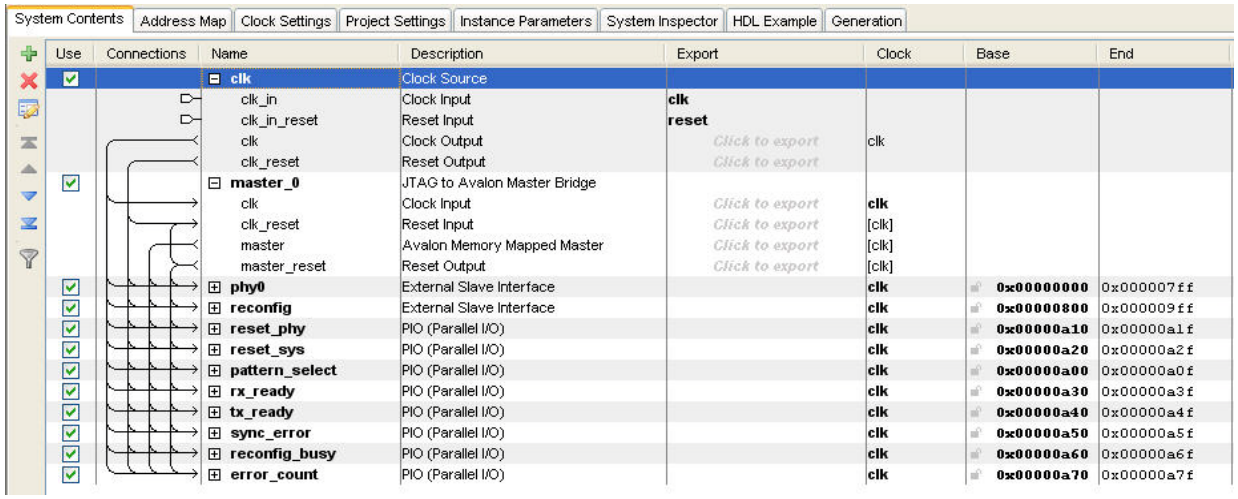


Table 1 lists the memory map for the Qsys system.

Table 1. Qsys Memory Map (Part 1 of 2)

Name	Component Name	Base Address	Description
phy0	External Slave Interface	0x000	Exports Avalon signals to interface with the Low Latency PHY IP core
reconfig	External Slave Interface	0x800	Exports Avalon signals to interface with the Reconfiguration Controller

Table 1. Qsys Memory Map (Part 2 of 2)

Name	Component Name	Base Address	Description
reset_phy	Parallel I/O (PIO)	0xa10	Controls the reset signal to the Low Latency PHY IP core
reset_sys	PIO	0xa20	Controls the reset signal to the system
pattern_select	PIO	0xa00	Controls the pattern type for the generator and the verifier
rx_ready	PIO	0xa30	Monitors the rx_ready signal from the Low Latency PHY IP core
tx_ready	PIO	0xa40	Monitors the tx_ready signal from the Low Latency PHY IP core
sync_error	PIO	0xa50	Monitors the sync signal from the verifier
reconfig_busy	PIO	0xa60	Monitors the reconfig_busy signal from the Reconfiguration Controller
error_count	PIO	0xa70	Monitors the error count from the data verifier
inject_error	PIO	0xa80	Injects an error into the PRBS generator
rx_is_lockedtoref	PIO	0xa90	Monitors the locked-to-reference signal

For more information about the Qsys and Qsys components, refer to the Qsys System Integration Tool Support web page.

Creating the Low Latency PHY IP Core

The sample design uses the Low Latency PHY IP core as a single duplex transceiver channel. The Low Latency PHY IP core uses two transmit PLLs. The first transmit PLL, TX PLL 0, is configured to run at a data rate of 1250 Mbps and the second transmit PLL, TX PLL 1, is configured to run at a data rate of 2457.6 Mbps. With the Reconfiguration Controller connected, you can selectively determine which transmit PLL or data rate is used.

For more information about the Low Latency PHY IP core, refer to the *Low Latency PHY IP Core* chapter in the *Altera Transceiver PHY IP Core User Guide*.

To set the parameters required by the Low Latency PHY IP core to switch between the two PLLs, perform the following steps:

1. Launch the Quartus II software
2. On the Tools menu, click **MegaWizard Plug-In Manager**
3. Click **Edit an existing custom megafunction variation**
4. Click **Next**
5. Select the **my_low_latency_phy.v** file located in the **original_design/source** directory

6. Click **Next**

When the Low Latency PHY IP core opens in the MegaWizard™ Plug-In Manager, the **General** tab appears first. This tab contains the general settings for the Low Latency PHY IP core.

The initial data rate set for the design is 1250 Mbps and uses an input reference clock frequency of 625 MHz. Figure 3 shows the **General** tab of the Low Latency PHY IP core.

Figure 3. Low Latency PHY IP Core Generate Tab

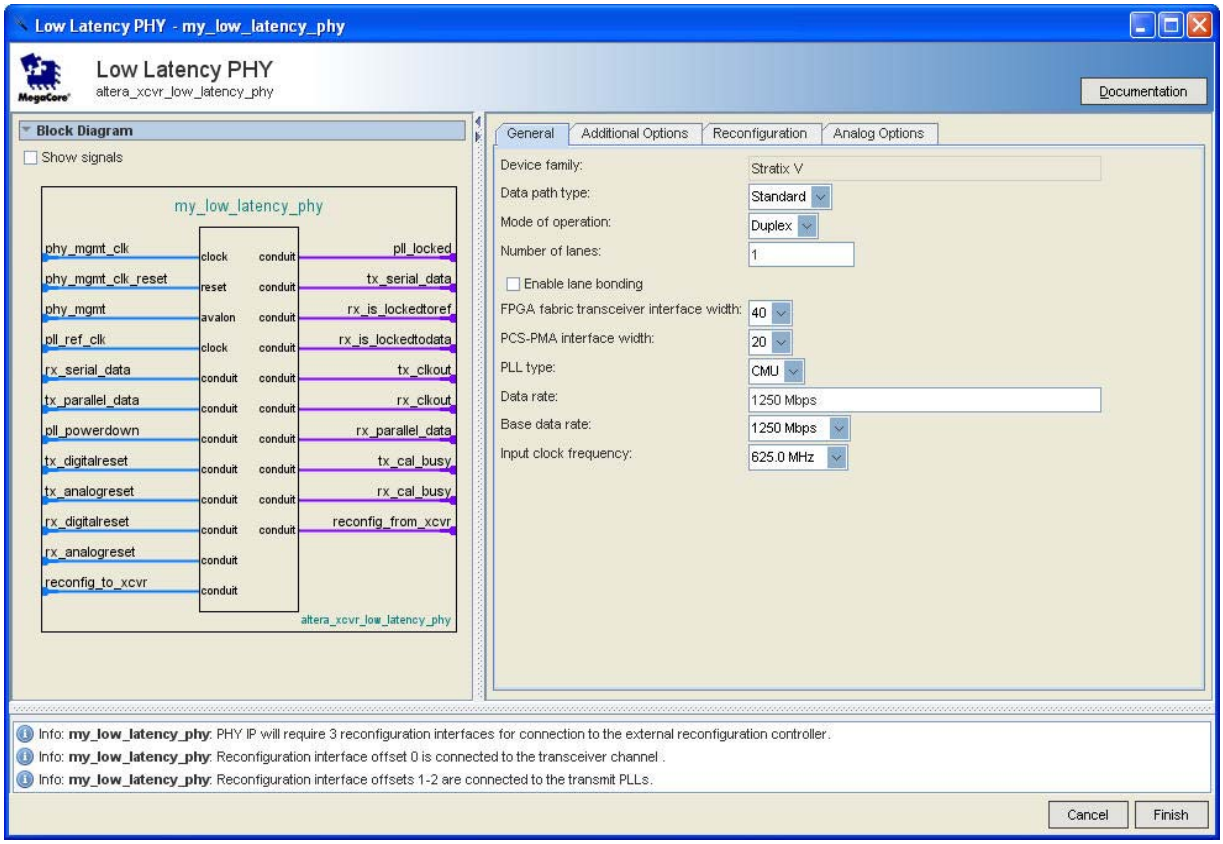


Table 2 lists the parameters in the **General** tab for the Low Latency PHY IP core and their settings.

Table 2. Low Latency PHY IP Core General Tab Parameters (Part 1 of 2)

Parameter	Value
Data path type	Standard
Mode of operation	Duplex
Number of lanes	1
Enable lane bonding	Disabled
FPGA fabric transceiver interface width	40
PCS-PMA interface width	20
PLL type	CMU

Table 2. Low Latency PHY IP Core General Tab Parameters (Part 2 of 2)

Parameter	Value
Data rate	1250 Mbps
Base data rate	1250 Mbps
Input clock frequency	625.0 MHz

7. Click the **Additional Options** tab in MegaWizard Plug-In Manager

   The **Additional Options** tab contains optional ports and features that can be enabled. In this sample design, no options are set in the **Additional Options** tab. Figure 4 shows the **Additional Options** tab of the Low Latency PHY IP core.

Figure 4. Low Latency PHY IP Core Additional Options Tab

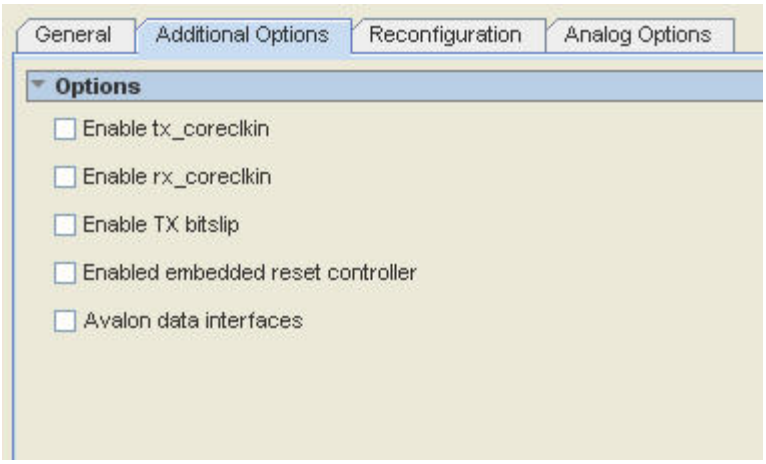


Table 3 lists the parameters in the **Additional Options** tab and their settings for the Low Latency PHY IP core.

Table 3. Low Latency PHY IP Core Additional Options Tab Parameters

Parameter	Value
Enable `tx_coreclkin`	Disabled
Enable `rx_coreclkin`	Disabled
Enable TX bitslip	Disabled
Enable embedded reset controller	Disabled
Avalon data interfaces	Disabled

8. Click the **Reconfiguration** tab in the MegaWizard Plug-In Manager

   The **Reconfiguration** tab contains all the reconfiguration features in the Low Latency PHY IP core. In order to change the data rate, this design needs to specify two transmit PLLs. The first transmit PLL is used for the initial data rate of 1250 Mbps. The second transmit PLL is used for the final data rate of 2457.6 Mbps. Also, each transmit PLL uses different reference clocks. Figure 5 shows the **Reconfiguration** tab of the Low Latency PHY IP core.

Figure 5. Low Latency PHY IP Core Reconfiguration Tab

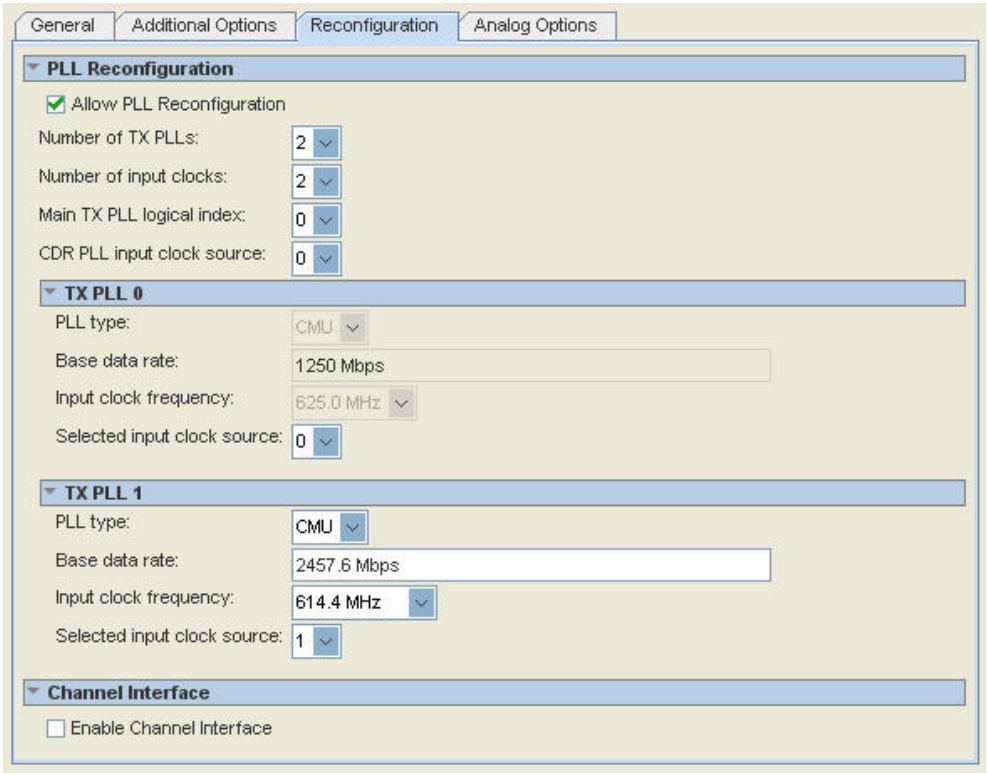


Table 4 lists the parameters in the PLL Reconfiguration section of the **Reconfiguration** tab and their settings.

Table 4. Low Latency PHY IP Core Reconfiguration Tab Parameters

Parameter	Value
Allow PLL Reconfiguration	Enabled
Number of TX PLLs	2
Number of input clocks	2
Main TX PLL logical index	0
CDR PLL input clock source	0

The parameters in the TX PLL 0 section are not editable because the main TX PLL logical index is set to 0. The parameters for TX PLL 0 are derived from the **General** tab. In this case, the data rate is set to 1250 Mbps with a reference clock of 625 MHz. The only parameter that can be modified is **Selected input clock source**, which in the sample design is set to 0. Table 5 lists the parameters for the TX PLL 0 section and their settings.

Table 5. TX PLL 0 Parameters (Part 1 of 2)

Parameter	Value
PLL type	CMU
Base data rate	1250 Mbps

Table 5. TX PLL 0 Parameters (Part 2 of 2)

Parameter	Value
Input clock frequency	625.0 MHz
Selected input clock source	0

The parameters in the TX PLL 1 section specify the values for the second transmit PLL. For this design, the transmit PLL generates a data rate of 2457.6 Mbps with a reference clock frequency of 614.4 MHz. The reference clock used is different from the one used for transmit PLL 0. To specify a different reference clock, the **Selected input clock source** parameter is set to 1. Table 6 lists the parameters for the TX PLL 1 section and their settings.

Table 6. TX PLL 1Parameters

Parameter	Value
PLL type	CMU
Base data rate	2457.6 Mbps
Input clock frequency	614.4 MHz
Selected input clock source	1

After all parameters have been specified, you can generate the Low Latency PHY IP core.

9. Click **Finish** in the MegaWizard Plug-In Manager

Creating the Reconfiguration Controller

The Reconfiguration Controller is used to control the dynamic reconfiguration of all Stratix V PHY IPs.

To set up the Reconfiguration Controller to dynamically control the PMA settings, change the PLL section, and stream a **.mif**, perform the following steps:

1. Launch the Quartus II software
2. On the Tools menu, click **MegaWizard Plug-In Manager**
3. Click **Edit an existing custom Megafunction variation**
4. Click **Next**
5. Browse and select the **my_reconfig.v** file located in the **original_design/source** directory

The Low Latency PHY IP core created in "Creating the Low Latency PHY IP Core" on page 4 requires three reconfiguration interfaces. Because the Low Latency PHY IP core uses a single channel, you do not need to specify a value in the optional grouping field. Figure 6 shows the Reconfiguration Controller.

Figure 6. Reconfiguration Controller

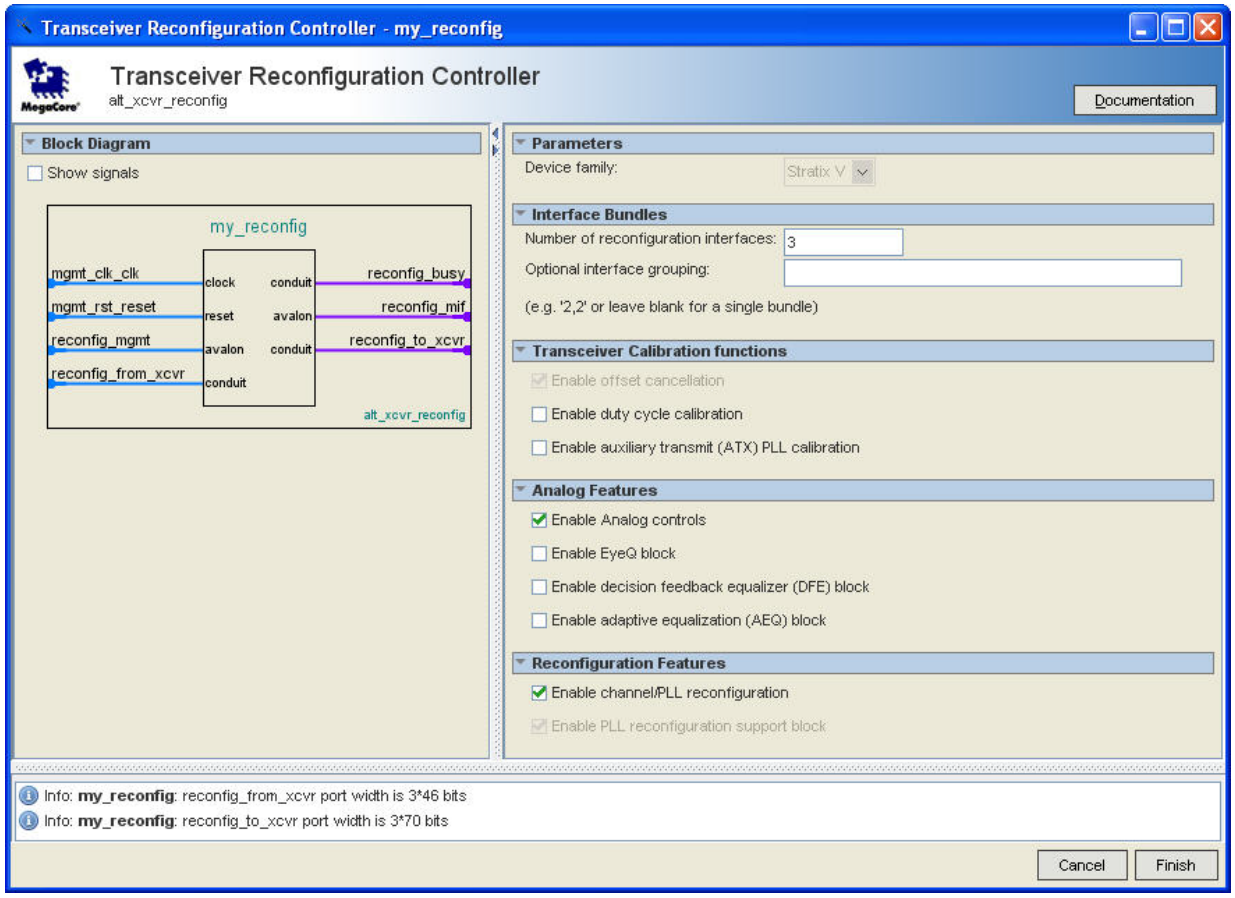


Table 7 lists the parameters in the **Interface Bundles** section of the Transceiver Reconfiguration Controller.

Table 7. Interface Bundles Parameters

Parameter	Value
Number of reconfiguration interfaces	3
Optional interface grouping	—

The **Transceiver Calibration functions** section specifies the blocks that can be enabled to improve the signal quality of the transceiver. The design does not require any of the calibration functions; therefore, they are all disabled. Table 8 lists the parameters in the **Transceiver Calibration functions** section.

Table 8. Transceiver Calibration Functions Parameters

Parameter	Value
Enable duty cycle calibration	Disabled
Enable auxiliary transmit (ATX) PLL calibration	Disabled

The **Analog Features** section specifies the features that can be enabled to tune and monitor the analog blocks of the transceiver. To change the analog settings of the PMA, turn on the transceiver **Enable Analog controls** feature. The other analog features are not used and can be disabled. Table 9 lists the parameters in the **Analog Features** section.

Table 9. Analog Features Parameters

Parameter	Value
Enable Analog controls	Enabled
Enable EyeQ block	Disabled
Enable decision feedback equalizer (DFE) block	Disabled
Enable adaptive equalization (AEQ) block	Disabled

The design changes the PLL that drives the transceiver so the Channel/PLL reconfiguration feature must be enabled. Table 10 lists the parameters in the **Reconfiguration Features** section.

Table 10. Reconfiguration Features Parameters

Parameter	Value
Enable channel/PLL reconfiguration	Enabled
Enable PLL reconfiguration support block	Enabled

When all parameters have been specified, you can generate the Reconfiguration Controller.

6. Click **Finish** in the MegaWizard Plug-In Manager

Creating a ROM that Contains the .mif for Reconfiguration

There are two methods for performing dynamic reconfiguration of the Low Latency PHY IP core:

- A **register-based** reconfiguration is carried out by writing to a specific set of memory-mapped registers in the Reconfiguration Controller. For example, to modify the transmitter's VOD setting, you must write to a specific set of memory-mapped registers in the Reconfiguration Controller.
- A **streamer-based** reconfiguration is carried out by streaming a **.mif**, which contains the reconfiguration data, to the Reconfiguration Controller.

Altera recommends that you have two separate designs when generating a **.mif** for reconfiguration:

- The first design or the original design is the functional design that generates the programming file for the Stratix V device.
- The second design or **.mif** design contains the PHY IP settings for reconfiguration. The **.mif** design is used only to generate the **.mif** for the original design. The following steps describe how to generate the **.mif** design used in the sample design.

☞ Use two different directories to compile the original design and the **.mif** design. This precaution prevents inadvertently deleting or modifying a design file.

The **.mif** design is the original design with different settings specified for the Low Latency PHY IP core. In the original design, the initial data rate is set to 1250 Mbps. After **.mif** reconfiguration, the data rate is 2457.6 Mbps. This change is accomplished by changing the Low Latency PHY IP core settings. Only the settings contained in the **General** and **Reconfiguration** tabs must be changed. Perform the following steps to change the Reconfiguration Controller:

1. Launch the Quartus II software
2. On the Tools menu, click **MegaWizard Plug-In Manager**
3. Click **Edit an existing custom Megafunction variation**
4. Click **Next**
5. Browse and select the **my_low_latency_phy.v** file located in the **mif_design/source** directory

   The initial data rate for the **.mif** design must be set to 2457.6 Mbps with an input reference clock frequency of 614.4 MHz. For this design, the original design and the **.mif** design use different reference clock frequencies. Table 11 lists the **.mif** design parameter values set in the **General** tab of the Low Latency PHY IP core.

Table 11. Low Latency PHY IP Core General Tab Parameters

Parameter	Original Design Value	.mif Design Value
Data path type	Standard	Standard
Mode of operation	Duplex	Duplex
Number of lanes	1	1
Enable lane bonding	Disabled	Disabled
FPGA fabric transceiver interface width	40	40
PCS-PMA interface width	20	20
PLL type	CMU	CMU
Data rate	1250 Mbps	2457.6 Mbps
Base date rate	1250 Mbps	2457.6 Mbps
Input clock frequency	625.0 MHz	614.4 MHz

6. Click the **Reconfiguration** tab in the MegaWizard Plug-In Manager

   Parameters under the **Reconfiguration** tab must be changed to specify the new initial data rate of 2457.6 Mbps. Table 12 lists the parameters set in the PLL Reconfiguration section of **Reconfiguration** tab.

Table 12. Low Latency PHY IP Core Reconfiguration Tab Parameters

Parameter	Original Design Value	.mif Design Value
Allow PLL reconfiguration	Enabled	Enabled
Number of TX PLLs	2	2
Number of input clocks	2	2
Main TX PLL logical index	0	1
CDR PLL input clock source	0	1

Setting the **Main TX PLL logical index** parameter to 1 keeps the same PLL settings as the original design.

The parameters in the TX PLL 0 section are changed to match the TX PLL 0 from the original design. Table 13 lists the parameters for the TX PLL 0 section.

Table 13. TX PLL 0 Parameters

Parameter	Original Design Value	.mif Design Value
PLL type	CMU	CMU
Base date rate	1250 Mbps	1250 Mbps
Input clock frequency	625.0 MHz	625.0 MHz
Selected input clock source	0	0

You cannot edit the parameters in the TX PLL 1 section because the **Main TX PLL logical index** is set to 1, so the parameter values set under the **General** tab are used. Table 14 lists the parameters for the TX PLL 1 section.

Table 14. TX PLL 1 Parameters

Parameter	Original Design Value	.mif Design Value
PLL type	CMU	CMU
Base date rate	2457.6 Mbps	2457.6 Mbps
Input clock frequency	614.4 MHz	614.4 MHz
Selected input clock source	1	1

7. After all parameters have been specified, you can generate the Low Latency PHY IP core for the **.mif** design

Compiling the Design

This section describes the compilation process to generate a programming file, such as a SOF, for the Stratix V device. Two designs currently exist: the original design and the **.mif** design.

The **.mif** design is compiled first because the **.mif** generated is used by the original design to configure from one data rate to another. Perform the following steps to compile the design:

1. Open the **.mif** project in the Quartus II software
2. On the Processing menu, click **Start Compilation**

After a successful compilation, a **reconfig_mif** directory is created in the **.mif** design's project directory. The **my_low_latency_phy_inst_channel.mif** file is used in the original design compilation.

For more information about the various **.mif** files contained in the **reconfig_mif** directory, refer to the *Altera Transceiver PHY IP Core User Guide*.

The original design is compiled after the **.mif** design. However, before the original design is compiled, you must specify the **.mif** created by the **.mif** design. The module **mif_rom** is used to store the **.mif**. To specify the **.mif**, perform the following steps:

1. Open the original project in the Quartus II software.
2. On the Tools menu, click **MegaWizard Plug-In Manager**.
3. From the MegaWizard Plug-In Manager, browse to the **original_design/source** directory and select **mif_rom.v**.
4. All parameters are the same except that you must specify the **.mif**. Specify the **mif_design/reconfig_mif/my_low_latency_phy_inst_channel.mif** file by browsing to the **.mif**.
5. Click **Finish** to generate the new **mif_rom** module.

After the **.mif** has been specified, the original design is ready for compilation. To compile the design, perform the following steps:

1. Open the original project in the Quartus II software
2. On the Processing menu, click **Start Compilation**

After a successful compilation, a **top.sof** file exists in the original design directory. This SOF is used to program the Stratix V device.

Performing Reconfiguration

With the Avalon to JTAG Master Bridge, reconfiguration commands are directly streamed to the Reconfiguration Controller through the JTAG port. Use the System Console tool to issue commands that initiate a dynamic reconfiguration of the Low Latency PHY IP core. This section describes how to use the System Console tool to perform reconfiguration.

For more information about the System Console tool, refer to the *Analyzing and Debugging Designs with the System Console* chapter in volume 3 of the *Quartus II Handbook*.

Program the Stratix V device with the SOF generated in the "Compiling the Design" section before launching the System Console.

Before any reconfiguration can take place, you must first launch the System Console tool. To launch the System Console, perform the following steps:

1. Program the Stratix V Device with the SOF generated from the original design
2. Launch the Quartus II software
3. From the Quartus II software, on the Tools menu, click **Qsys**
4. From the Qsys tool, on the Tools menu, click **System Console**

From the System Console tool, type the following commands in the Tcl console:

```
set master_path [ lindex [ get_service_paths master ] 0 ]
open_service master $master_path
```

These commands allow the System Console tool to claim a service master. The service master allows the System Console to communicate directly with the Avalon to JTAGA Bridge Master, which in turn communicates with the Reconfiguration Controller.

After the service has been claimed, reconfiguration commands can be issued with the System Console. The following sections describe the steps necessary to carry out a reconfiguration.

For more information about the specific address map associated with the Reconfiguration Controller, refer to the *Transceiver Reconfiguration Controller* chapter in the *Altera Transceiver PHY IP Core User Guide*.

Changing the Transceiver PMA Settings

This section describes the necessary steps to modify the analog settings within the transceiver's PMA block.

To change the PMA settings for the Low Latency PHY IP core, write to the PMA Analog Controls memory-mapped registers in the Reconfiguration Controller.

For a description of all available addresses, refer to the "PMA Analog Controls" section of the *Transceiver Reconfiguration Controller* chapter in the *Altera Transceiver PHY IP Core User Guide*.

To set the VOD setting to 40 on channel 0, perform the following steps:

1. Set the reconfiguration controller base address as specified in the Qsys system:

```
set base 0x800
```

2. Add a delay between writes:

```
after 100
```

3. Set the logical channel to 0:

```
master_write_32 $master_path [expr $base + 0x8*4] 0x0
```

4. Add a delay between writes:

```
after 100
```

5. Set the offset value to VOD:

```
master_write_32 $master_path [expr $base + 0xB*4] 0x0
```

6. Add a delay between writes:

```
after 100
```

7. Set the data register to 40:

```
master_write_32 $master_path [expr $base + 0xC*4] 0x28
```

8. Add a delay between writes:

```
after 100
```

9. Write all the data:

```
master_write_32 $master_path [expr $base + 0xA*4] 0x1
```

After the `reconfig_busy` signal goes low, the VOD on channel 0 is set to a value of 40.

Changing the Transmitter PLL Via Memory-Map Writes

This section describes the steps necessary to change the PLL selection.

To change the PLL selection for the Low Latency PHY IP core, you must write to a specific memory-mapped register in the Reconfiguration Controller.

For a description of all available addresses, refer to the "PLL Reconfiguration" section of the *Transceiver Reconfiguration Controller* chapter in the *Altera Transceiver PHY IP Core User Guide*.

To change the PLL selection with the memory-mapped interface, perform the following steps:

1. Set the reconfiguration controller base address as specified in the Qsys system:

```
set base 0x800
```

2. Set the logical channel to 0:

```
master_write_32 $master_path [expr $base + 0x40*4] 0x0
```

3. Add a delay between writes:

```
after 100
```

4. Set the offset to PLL reference selection:

```
master_write_32 $master_path [expr $base + 0x43*4] 0x1
```

5. Add a delay between writes:

```
after 100
```

6. Set the data register to logical PLL reference 1:

```
master_write_32 $master_path [expr $base + 0x44*4] 0x1
```

7. Add a delay between writes:

```
after 100
```

8. Write all the data:

```
master_write_32 $master_path [expr $base + 0x42*4] 0x1
```

After the `reconfig_busy` signal goes low, the PLL selection on channel 0 is set to a value of 1.

Streaming a .mif to Perform PLL Reconfiguration

This section describes the steps necessary to change the PLL selection. Unlike the procedure described in the "Changing the Transmitter PLL Via Memory-Map Writes" section, the PLL selection is carried out by streaming a **.mif** that contains the PLL selection.

For a description of the Streamer module, refer to the "Streamer Module" section of the *Transceiver Reconfiguration Controller* chapter in the *Altera Transceiver PHY IP Core User Guide*.

To change the PLL selection by streaming a **.mif**, perform the following steps:

1. Set the reconfiguration controller base address as specified in the Qsys system:

```
set base 0x800
```

2. Set the logical channel to 0:

```
master_write_32 $master_path [expr $base + 0x38*4] 0x0
```

3. Add a delay between writes:

```
after 100
```

4. Set the Streamer mode to 0:

```
master_write_32 $master_path [expr $base + 0x3A*4] 0x0
```

5. Add a delay between writes:

```
after 100
```

6. Set the Streamer offset register to the **.mif** base address 0x0:

```
master_write_32 $master_path [expr $base + 0x3B*4] 0x0
```

7. Add a delay between writes:

```
after 100
```

8. Set the data register with the **.mif** base address:

```
master_write_32 $master_path [expr $base + 0x3C*4] 0x0
```

9. Add a delay between writes:

```
after 100
```

10. Write all the data to the Streamer:

```
master_write_32 $master_path [expr $base + 0x3A*4] 0x1
```

11. Add a delay between writes:

```
after 100
```

12. Set the Streamer offset for Start **.mif** stream:

```
master_write_32 $master_path [expr $base + 0x3B*4] 0x1
```

13. Add a delay between writes:

```
after 100
```

14. Set the data register with 0x1 to setup for streaming:

```
master_write_32 $master_path [expr $base + 0x3C*4] 0x3
```

15. Add a delay between writes:

```
after 100
```

16. Write all the data to the Streamer to start streaming the **.mif**:

```
master_write_32 $master_path [expr $base + 0x3A*4] 0x1
```

After the `reconfig_busy` signal goes low, the Streamer module is complete and the PLL selection is set to 1 for channel 0.

Document Revision History

Table 15 lists the revision history for this document.

Table 15. Document Revision History

Date	Version	Changes
July 2012	1.0	Initial release.