The Altera® Transceiver Reconfiguration Controller dynamically reconfigures the transceiver PHY in Arria® V and Cyclone® V devices. You can use the dynamic reconfiguration features to reconfigure the transceiver channels to support multiple or different data rates and physical medium attachment (PMA) settings without interrupting adjacent transceiver channels or powering down the transceiver channels.

The reconfiguration methods are similar between Arria V, Cyclone V, and Stratix® V devices. The features supported in Arria V and Cyclone V devices are a subset of those supported in Stratix V devices.

Related Information

Altera Transceiver PHY IP Core User Guide

Reconfiguration Methods

You can dynamically change the transceiver setting using either register-based or streamer-based reconfiguration. Both methods use a sequence of Avalon®-MM writes and reads to update the transceiver settings.

Related Information

Avalon Interface Specifications
Refer to the read and write transfer timing diagrams.

Register-Based Reconfiguration

Register-based reconfiguration does not require any MIF files during the reconfiguration process. It uses a set of dedicated reconfiguration addresses to carry out a specific reconfiguration function. You use a specific flow to carry out this reconfiguration.

The design example in this application note demonstrates the following:

- The analog (PMA) reconfiguration update on the VOD settings
- The method to trigger the duty cycle distortion (DCD) calibration

Related Information

- Arria V GX Dynamic Reconfiguration Design Example on page 3
- Altera Transceiver PHY IP Core User Guide
  For the register-based reconfiguration read and write flow.
**Streamer-Based Reconfiguration**

The streamer-based reconfiguration mode supports the reconfiguration features that are not achievable with the register-based method.

There are two supported modes: **MIF Streaming** and **Direct Write**. Both modes use the streamer module in the Reconfiguration Controller. The streamer module uses the same address to carry out reconfiguration. However, the data values are different and you must specify that to the Reconfiguration Controller.

- **MIF Streaming mode (Mode 0):**
  - Streams the entire content of a MIF
  - Uses the streamer module

The advantage of this mode is you only need to use one command to execute the write process of the entire MIF. You do not need to manually control the write process to dedicated reconfiguration addresses such as PMA settings, reference clock selection, and PLL selection.

The design example demonstrates the streamer-based reconfiguration mode when switching the TX PLL connected to the transceiver channel.

- **Direct Write mode (Mode 1):**
  - No MIF streaming is required
  - You need to selectively write the reconfiguration data
  - May require multiple writes and reads

The advantage of this mode is to access the reconfiguration address that is not supported by the register-based method.

**Related Information**

- [Arria V GX Dynamic Reconfiguration Design Example](#) on page 3
- [Altera Transceiver PHY IP Core User Guide](#)

**Transceiver Calibration Function**

The Reconfiguration Controller supports two calibration functions: **offset cancellation** and **duty cycle distortion** (DCD) calibration.

The design example shows how to execute the DCD calibration from the Reconfiguration Controller.

**Related Information**

- [Arria V GX Dynamic Reconfiguration Design Example](#) on page 3
- [Duty Cycle Distortion Calibration](#) on page 17

Duty Cycle Distortion (DCD) calibration is used to calibrate the TX duty cycle to compensate for the skew introduced by different clock networks.
Unsupported Reconfiguration Modes

The Reconfiguration Controller in Arria V and Cyclone V devices does not support the following modes:

- Switching between a receiver-only channel and a transmitter-only channel
- Switching between one PHY IP to another PHY IP (for example, switching from a deterministic latency PHY IP to a custom PHY IP)
- Switching between PMA Direct mode to non-PMA Direct mode
- Bonded mode configuration
- TX PLL reconfiguration if the TX PLL is connected to bonded channels

Arria V GX Dynamic Reconfiguration Design Example

The design example uses the Reconfiguration Controller to dynamically reconfigure a Native PHY IP to support multiple data rates of 2500 Mbps and 5000 Mbps by switching the external PLL connected to the transceiver channel. The design example uses a 5AGXFB3H4F35C5 device and is compiled with the Quartus® II 12.1sp1 software.

The reconfiguration commands are controlled through the System Console tool that ships with the Quartus II software. This design example demonstrates the following reconfiguration methods:

- Streamer-based reconfiguration
  - The MIF streaming reconfiguration is used to switch the TX PLLs that are connected to the transceiver channel.
- Register-based reconfiguration
  - Changing VOD setting
  - Triggering DCD calibration manually

The design example consists of the following modules. The numbers refer to the position of the modules in the following figure. The system-level diagram shows how the different modules interact in the reconfiguration design example.

1. Arria V GX Transceiver Native PHY IP
2. Transceiver Reconfiguration Controller
3. Qsys system
4. PHY Reset Controller
5. CMU PLL – Transceiver PLL
6. Fractional PLL (fPLL) – Altera fPLL
7. ROM containing the MIF for reconfiguration
8. In-System Sources and Probes (ISSP)

The design example also contains a PRBS data generator and checker. The data generator generates a PRBS15 data pattern. The data checker verifies the PRBS15 data received.
Related Information

AN 676 Reference Design Example

Creating the Qsys System

The reconfiguration design example uses a simple Qsys system that consists of three components: the JTAG to Avalon Master Bridge, the External Slave Interface, and the PIO.

Follow the steps below to examine the Qsys system:

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/` directory
4. Click Open
The Qsys System Components

The Qsys tool launches and shows all components used.

The Qsys system contains the following components:

- The **JTAG to Avalon Master Bridge component** acts as the master in the design example and is the main communication channel between the System Console tool and the external slave interface 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 of the Qsys component library.

- The **Transceiver Reconfiguration Controller** is an Avalon-compliant component. Therefore, the External Slave Interface component must be connected to the JTAG to Avalon Master Bridge.

- The **PIO component** uses external input as control bits in the system. In this design example, the PIO is connected to the `rate_select` port, which is used to trigger the rate change of the channel from 2500 Mbps to 5000 Mbps. The PIO connects to the JTAG to Avalon Master Bridge. You can also connect any PIO to status bits to be monitored externally.

**Note:** The Arria V and Cyclone V PHY IP components are not supported in the Qsys tool in the Quartus II software. To interface with an Arria V or Cyclone V PHY IP in a Qsys system, you must use an external slave interface.

![Component Map of the Qsys System](image)

![Memory Map of the Qsys System](image)

**Table 1: Memory Map of the Qsys System**

<table>
<thead>
<tr>
<th>Name</th>
<th>Component Name</th>
<th>Base Address</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>reconfig</td>
<td>External Slave Interface</td>
<td>0x000</td>
<td>Exports Avalon signals to interface with the Reconfiguration Controller</td>
</tr>
<tr>
<td>pio_0</td>
<td>PIO (Parallel I/O)</td>
<td>0x800</td>
<td>Export the <code>rate_select</code> port for TX PLL selection</td>
</tr>
</tbody>
</table>
Creating the Transceiver Native PHY IP

The design example uses the Arria V Native PHY IP as a single duplex transceiver channel. Unlike other PHY IP, the Native PHY IP does not include the Avalon-MM interface. Instead, it exposes all signals directly as ports. In this design example, the Native PHY IP interfaces with the Reset Controller, Reconfiguration Controller, and the ISSP.

The Native PHY is created such that two transmit PLLs are used to clock the data channels. Both transmit PLLs are instantiated using external transceiver PLLs. The CMU PLL and fPLL are selected as the external transceiver PLLs. Follow the steps in the following figures to set up the parameters required by the Native PHY to switch between the two external transceiver PLLs.

Figure 3: Datapath Options, TX PMA, and TX PLL0 Settings in Native PHY IP

1. Check this option to use the external TX PLL
2. Use 2 external PLLs
3. Indicate that TX PLL 0 is follo the PLL setting in the PMA tab
4. TX PLL 0 is set to 2500 Mbps
Figure 4: TX PLL 1 and RX PMA settings in Native PHY IP

1. This option remains checked
2. You can change the TX PLL 1 setting in this field. In this example, you set it to 5,000 Mbps
3. Check this option to change the CDR setting dynamically to support a different data rate

Turn on the Enable CDR dynamic reconfiguration option to allow the data rate change of the CDR during streamer-based reconfiguration. With the Reconfiguration Controller connected, you can selectively determine which transmit PLL is used.

Refer to the gxb_duplex.v file in the design example for the standard PCS settings.

Related Information
Altera Transceiver PHY IP Core User Guide
For more information on how to instantiate the Native PHY IP Datapath, Standard PCS, and RX PMA options.

Creating the Reconfiguration Controller
The Reconfiguration Controller controls the dynamic reconfiguration of Arria V and Cyclone V PHY IPs.

The following steps describe how to set up the Reconfiguration Controller to dynamically control the PMA settings, change the PLL selection by streaming a MIF, and trigger DCD calibration manually.

The Native PHY IP created in the previous section requires two reconfiguration interfaces, one for the REGULAR RX/TX Channel and one for the CDR TX PLL, as shown in the following figure. You can verify the logical interface information in the Transceiver Reconfiguration Report.
The Transceiver Reconfiguration Report is located under Fitter Report > GXB Report.

Refer to the parameters setting in the figure below to set up the Interface Bundles, Transceiver Calibration functions, Analog Features, and Reconfiguration Features functions.

The Interface Bundles section specifies two interface bundles. The first interface is connected to the RX/TX channels as shown in Figure 5. The second interface is connected to the CMU PLL.

The following table shows the Interface Bundles connection in the top-level design file a5_top.v.
Table 2: Interface Bundles Parameters

<table>
<thead>
<tr>
<th>Reconfiguration Ports</th>
<th>Native PHY/ CMU PLL Ports</th>
<th>Connected to</th>
</tr>
</thead>
<tbody>
<tr>
<td>[69:0] ch0_0_to_xcvr</td>
<td>[69:0] reconfig_to_xcvr</td>
<td>Connected to RX/TX channel</td>
</tr>
<tr>
<td>[45:0] ch0_0_from_xcvr</td>
<td>[45:0] reconfig_from_xcvr</td>
<td></td>
</tr>
<tr>
<td>[69:0] ch1_1_to_xcvr</td>
<td>[69:0] reconfig_to_cmu</td>
<td>Connected to CMU PLL</td>
</tr>
<tr>
<td>[45:0] ch1_1_from_xcvr</td>
<td>[45:0] reconfig_from_cmu</td>
<td></td>
</tr>
</tbody>
</table>

In the Transceiver Calibration Functions section, turn on the Enable duty cycle calibration option.

In the Analog Features section, turn on the Enable Analog controls option to enable VOD setting reconfiguration.

In the Reconfiguration Features section, turn on the Enable channel/PLL reconfiguration option to allow the streamer-based reconfiguration process. This reconfiguration mode reconfigures the TX/RX data path, CDR settings, and TX PLL selection.

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

Related Information

Duty Cycle Distortion Calibration on page 17

Duty Cycle Distortion (DCD) calibration is used to calibrate the TX duty cycle to compensate for the skew introduced by different clock networks.

Creating the CMU PLL Using an Arria V Transceiver PLL

The design example uses the Arria V Transceiver PLL to clock the transceiver channel at 5000 Mbps.

When you turn on the Use external TX PLL option in the Native PHY IP, you can connect to this external transceiver PLL. This transceiver PLL is referring to the CMU PLL as illustrated in the System Diagram. You can instantiate this IP in the MegaWizard™ Plug-in Manager > Interface > Transceiver PHY > Arria V Transceiver PLL v12.1.

Refer to the following figure to set the parameters for the CMU PLL.

Figure 7: Arria V Transceiver PLL Parameters Setting When Configured as CMU PLL

Select the CMU PLL type
Set the data rate to 5000 MHz
You do not have to turn on the Enable PLL reconfiguration option if you are not dynamically reconfiguring the PLL parameter settings. This option allows you to change the PLL settings to support different data rates.

**Note:** If you are using an fPLL in your design, if you want to switch the fPLL with another CMU PLL, you must instantiate the CMU PLL using the Arria V Transceiver PLL IP. Turn on the option Use external TX PLL in the Native PHY IP to instantiate a CMU PLL.

### Creating a Fractional PLL (fPLL) using Altera PLL

The design example uses the Altera PLL v12.1 to configure an fPLL to clock the transceiver channel at 2500 Mbps.

To connect the Native PHY IP to the fPLL, you must turn on the Use external TX PLL option in the Native PHY IP. You can instantiate this IP in the MegaWizard Plug-in Manager > IO > Altera PLL v12. Refer to the figure below to set the parameters in the fPLL.

**Figure 8: Altera PLL Parameters Setting when Configured as an fPLL**

![Altera PLL Parameters Setting](image)

- **Enable the PLL locked signal**
- **Set the desired frequency to 1250 MHz because the VCO in the PLL is running at half the serial data rate of 2500 Mbps**

### Creating the Transceiver PHY Reset Controller

The design example uses the Transceiver PHY Reset Controller to control the reset sequence of the transceiver channel.

As shown in the figure below, set the Number of TX PLLs field to 2. In this design example, you switch the TX PLL between the CMU PLL and fPLL. Therefore, you must connect both PLL locked signals, `pll_locked[1:0]`, to the reset controller to indicate the release of `tx_digitalreset`. The reset controller releases `tx_digitalreset` whenever there is an assertion on either of the `pll_locked[1:0]` signals. Leave the remaining settings in the PHY Reset Controller to their default values.
Figure 9: Transceiver Reset Controller Parameter Settings

Set this option to 2. There will be indexing on the pll_locked[1:0] port. Connect the locked signal from the fPLL and CMU PLL to this port.

Creating a ROM that Contains the MIF for Reconfiguration

Dynamic reconfiguration of the Native PHY can be performed using one of two methods: register-based and streamer-based. The register-based reconfiguration is carried out by writing to a specific set of memory-mapped registers in the transceiver channel.

The streamer-based reconfiguration is carried out by streaming a MIF that contains the reconfiguration data to the Reconfiguration Controller. The steps below describe how to generate the MIF for reconfiguration for the design example.

Note: Two different design directories should be used to compile the original design and the MIF design. This practice prevents inadvertently deleting or modifying the design files. The MIF design can be as simple as a design with just the Native PHY IP instantiation file. You can also use the original design as suggested in following section to generate the MIF.
The MIF design is the original design with different settings specified for the Native PHY IP. In the original design, the initial data rate is set to 2500 Mbps. Change the Native PHY IP settings so that after MIF reconfiguration the data rate is 5000 Mbps. To generate the MIFs, use the table below for the settings in the Native PHY IP. Only the settings in the TX PMA tab change.

Table 3: MIFs Generation in Reference with Native PHY IP Settings

<table>
<thead>
<tr>
<th>MIF #</th>
<th>MIFs (Mbps)</th>
<th>PMA</th>
<th>TX PMA</th>
<th>TX PLL 0</th>
<th>TX PLL 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data Rate (Mbps)</td>
<td>TX Local Clock Division Factor</td>
<td>TX PLL Base Data Rate (Mbps)</td>
<td>Use external TX PLL</td>
<td>Number of TX PLLs</td>
</tr>
<tr>
<td>1</td>
<td>2500</td>
<td>2500</td>
<td>1</td>
<td>2500</td>
<td>Enabled</td>
</tr>
<tr>
<td>2</td>
<td>5000</td>
<td>5000</td>
<td>1</td>
<td>5000</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

Compiling the Design Example

The compilation process generates an .sof programming file for the Arria V device.

At this point in the procedure, two designs exist: the original design and the MIF design.

The MIF design is compiled first because the MIF generated will be used by the original design. Specifically, the MIF generated by the MIF design is used in the original design to configure from one data rate to another. To compile a design:

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

The changes you make in the MIF design are the Native PHY IP parameters listed in the "MIFs Generation in Reference with Native PHY IP Settings" table. Generate a 5000 Mbps design by setting the parameters in your Native PHY IP to create a mif_5000.mif file. Next, generate a 2500 Mbps design by setting the parameters in your Native PHY IP to create a mif_2500.mif file.

Note: If you get an error message related to a missing .mif (before the MIF is generated and specified in the ROM MegaWizard Plug-In Manager), select the “No, leave it blank” option at the Memory Initialization tab of the MegaWizard Plug-In Manager.

After a successful compilation, a reconfig_mif directory is created in the MIF design’s project directory. The following MIFs are used for each data rate:

- 2500 Mbps – mif_2500.mif
- 5000 Mbps – mif_5000.mif

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. Follow the steps below to specify the MIF:
1. Open the original project in the Quartus II software.
2. Launch the MegaWizard Plug-In Manager from the Tools menu.
3. From the MegaWizard, browse to the `original_design/` 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/mif_2500.mif` file by browsing to the MIF.
5. Click **Finish** to generate the new `mif_rom` module.
6. Repeat these steps for the ROM to store the MIF for 5000 Mbps.

After the MIF has been specified, the original design is ready to be compiled. Follow the steps below to compile the design.

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

After a successful compilation, a file named `a5_top.sof` will exist in the `original/output_files/` directory. This SOF is used to program the Arria V GX device.

**Note:** There is only one `.qar` project in this design example. Use this `.qar` project as both the original design and the MIF design to generate the MIF files. To create the MIF design, duplicate the original design.

**Related Information**

*Creating a ROM that Contains the MIF for Reconfiguration* on page 11

Refer to the “MIFs Generation in Reference with Native PHY IP Settings” table.

### Creating In-System Sources and Probes (ISSP)

The ISSP is instantiated to control the PHY reset, enable serial loopback, and align word boundaries on received data.

The Qsys system communicates with the ISSP to control the Native PHY.

**Table 4: ISSP and Its Control in the Design Example**

<table>
<thead>
<tr>
<th>Bit</th>
<th>ISSP</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[2]</td>
<td><code>rx_std_wa_pattern</code></td>
<td>Aligns the word boundaries in manual alignment mode</td>
</tr>
<tr>
<td>[1]</td>
<td><code>rx_seriallpbken</code></td>
<td>Enables the serial loopback of the transceiver channel</td>
</tr>
<tr>
<td>[0]</td>
<td><code>hssi_reset</code></td>
<td>Used as a system reset</td>
</tr>
</tbody>
</table>

### Performing Reconfiguration with the System Console Tool

With the Avalon to JTAG Master Bridge, reconfiguration commands are directly streamed to the Reconfiguration Controller through the JTAG port. The System Console tool issues commands to initiate dynamic reconfiguration of the Native PHY IP.

This design example uses a Tcl script called `main.tcl` that consists of several different procedures with different functionality.
Note: Program the Arria V GX device with the SOF generated in the previous section before launching the System Console. Having both the programmer and System Console open simultaneously can cause programming errors.

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 Arria 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
5. Ensure that the present working directory contains main.tcl

The following table lists the procedures in main.tcl. You can type in a procedure name and its value to execute the reconfiguration process. Verify your results with the signal tap file (stp1.stp) by looking at the signals listed in the following table.

Table 5: Description of Procedures in main.tcl

<table>
<thead>
<tr>
<th>Command Name</th>
<th>&lt;Value&gt;</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>txpll_register</td>
<td>0</td>
<td>Select logical TX PLL 0 as TX PLL (fPLL). Only the tx_std_clkout frequency is updated.</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Select logical TX PLL 1 as TX PLL (CMU PLL). Only the tx_std_clkout frequency is updated.</td>
</tr>
<tr>
<td>txpll_mif</td>
<td>2500</td>
<td>Select logical TX PLL 0 as TX PLL (fPLL). Both the tx_std_clkout and rx_std_clkout frequencies are updated.</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>Select logical TX PLL 1 as TX PLL (CMU PLL). Both the tx_std_clkout and rx_std_clkout frequencies are updated.</td>
</tr>
<tr>
<td>reset</td>
<td>N/A</td>
<td>System Reset</td>
</tr>
<tr>
<td>sloopback</td>
<td>1</td>
<td>Enable serial loopback. Verify with the rx_seriallpbken port in the signal tap file</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>Disable serial loopback. Verify with the rx_seriallpbken port in the signal tap file</td>
</tr>
<tr>
<td>read_vod</td>
<td>N/A</td>
<td>Read back VOD value (read back data in hexadecimal value)</td>
</tr>
<tr>
<td>write_vod</td>
<td>0-63</td>
<td>Writing VOD value with valid settings of 0-63</td>
</tr>
<tr>
<td>Command Name</td>
<td>&lt;Value&gt;</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>patternalign &lt;Value&gt;</td>
<td>1</td>
<td>Enable pattern alignment detection. To resynchronize to the new word boundary after each MIF based reconfiguration, use this command to create a 0-to-1 transition to the rx_std_wa_patternalign signal.</td>
</tr>
<tr>
<td>patternalign &lt;Value&gt;</td>
<td>0</td>
<td>Disable pattern alignment detection. Set this signal to 0 to re-assert the rx_std_wa_patternalign signal.</td>
</tr>
</tbody>
</table>

To reconfigure the transceiver channel, type the reconfiguration commands as shown below in the Tcl Console. For example:

source main.tcl

sloopback 1
>> Enable serial loopback

txpll_mif 2500
>> TX PLL switch to fPLL. Data channel at 2500 Mbps

reset
>> Reset the transceiver channel after each streamer-based reconfiguration

txpll_mif 5000
>> TX PLL switch to CMU PLL. Data channel at 5000 Mbps

reset
>> Reset the transceiver channel after each streamer-based reconfiguration

These commands allow the System Console to communicate directly with the Avalon to JTAG Bridge Master, which in turn communicates with the Reconfiguration Controller.

Related Information
- Analyzing and Debugging Designs with System Console
- Altera Transceiver PHY IP Core User Guide
  For more information on the specific address map associated with the Reconfiguration Controller

Stream a MIF to Perform Channel Reconfiguration
Switch the TX PLL to change the transceiver channel from 2500 Mbps to 5000 Mbps. You can switch the PLL by streaming a MIF.

Table 6: Steps for Using the Streamer-Based Reconfiguration Mode

<table>
<thead>
<tr>
<th>Step</th>
<th>Reconfiguration Step</th>
<th>Memory Map Address</th>
<th>Value Written</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Write to the logical channel register</td>
<td>0x38</td>
<td>0h</td>
<td>Logical channel 0 selected (Physical ch0)</td>
</tr>
</tbody>
</table>
### Manual Trigger for DCD Calibration IP via Register-based Reconfiguration

You can trigger the DCD calibration IP manually.

The following table lists the steps to access the reconfiguration address reserved for DCD calibration IP. You must trigger the DCD calibration IP when you switch from 2500 Mbps to 5000 Mbps because it switches the clock network and the channel data rate is >4915.2 Mbps. In the design example, the DCD calibration is triggered when the data channel is running at 5000 Mbps and after TX PLL switching happens. You can refer to the `txpll_mif` procedure in `main.tcl` for more details.

### Table 7: Using the Register-Based Reconfiguration Method to Trigger DCD Calibration

<table>
<thead>
<tr>
<th>Step</th>
<th>Reconfiguration Step</th>
<th>Memory Map Address</th>
<th>Value Written</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Write to the logical channel register</td>
<td>0x48</td>
<td>0h</td>
<td>Logical channel 0 selected (Physical ch0)</td>
</tr>
<tr>
<td>2</td>
<td>Write to the data offset register</td>
<td>0x4B</td>
<td>0h</td>
<td>Select DCD calibration mode</td>
</tr>
</tbody>
</table>
### Performing VOD Reconfiguration via Register-Based Reconfiguration

You can reconfigure the transceiver channel to change the VOD settings.

To verify the new settings, perform a write to the channel and read back the VOD setting.

#### Table 8: Using the Register-Based Reconfiguration Method to Reconfigure VOD Settings

<table>
<thead>
<tr>
<th>Step</th>
<th>Reconfiguration Step</th>
<th>Memory Map Address</th>
<th>Value Written</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Write to the logical channel register</td>
<td>0x08</td>
<td>0h</td>
<td>Logical channel 0 selected (Physical ch0)</td>
</tr>
<tr>
<td>2</td>
<td>Write to the PMA offset register</td>
<td>0x0B</td>
<td>0h</td>
<td>Select VOD settings</td>
</tr>
<tr>
<td>3</td>
<td>Write VOD valid settings</td>
<td>0x0C</td>
<td>0-63</td>
<td>Set the VOD settings with entered valid settings</td>
</tr>
<tr>
<td>4</td>
<td>Write to the &quot;write&quot; bit of the control and status register</td>
<td>0x0A</td>
<td>1h</td>
<td>Trigger &quot;write&quot; operation</td>
</tr>
</tbody>
</table>

**Note:** Reset your channel after each manual DCD calibration.

### Duty Cycle Distortion Calibration

Duty Cycle Distortion (DCD) calibration is used to calibrate the TX duty cycle to compensate for the skew introduced by different clock networks.

You must turn on the DCD calibration IP when you switch from 2500 Mbps to 5000 Mbps, because TX PLL switching causes a different clock network to be used.
Enable the DCD calibration IP for Arria V and Cyclone V devices if either of the following conditions is applicable:

- Data rate is $\geq 4915.2$ Mbps
- Clock network switching (TX PLL switching) and the data rate is $\geq 4915.2$ Mbps

The DCD calibration features and options are summarized in the following table. Refer to the usage condition to enable the DCD Calibration IP.

### Table 9: DCD Calibration Features and Their Options

<table>
<thead>
<tr>
<th>Transceiver Calibration Function</th>
<th>Option</th>
<th>Description</th>
<th>Usage Condition</th>
</tr>
</thead>
</table>
| Enable duty cycle calibration    | Enabled    | Use DCD calibration IP either during power up or user mode                   | • Data rate $\geq 4915.2$ Mbps  
• Clock network switching (TX PLL switching) and the data rate is $\geq 4915.2$ Mbps |
|                                  | Disabled   | Disabled DCD calibration features                                           | Data rate $< 4915.2$ Mbps                                                      |
| Calibrate duty cycle during power up | Enabled    | DCD calibration IP process at power up mode and during user mode (manual DCD calibration) | Data rate $\geq 4915.2$ Mbps                                                   |
|                                  | Disabled   | DCD calibration IP will not start at power up mode, but can still be triggered during user mode if the Enable duty cycle calibration option is enabled | • Data rate change from $< 4915.2$ Mbps to $\geq 4915.2$ Mbps  
• Clock network switching (TX PLL switching) and the data rate is $\geq 4915.2$ Mbps |
| Create optional calibration status ports | N/A        | tx_cal_busy, rx_cal_busy, cal_busy_in ports exposed                         | tx_cal_busy should be connected to cal_busy_in port if you are using more than one Reconfiguration Controller per side of the device |

**Note:** Do not enable DCD calibration for applications running at $< 4915.2$ Mbps.

The following table lists the data rate usage conditions and when to enable the power up and manual DCD calibration IP.
### Table 10: DCD Calibration Usage

<table>
<thead>
<tr>
<th>Data Rate Switch</th>
<th>From</th>
<th>To</th>
<th>Power Up</th>
<th>Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>&lt; 4915.2</td>
<td>&lt; 4915.2</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Case 2</td>
<td>&lt; 4915.2</td>
<td>≥ 4915.2</td>
<td>x</td>
<td>Enabled</td>
</tr>
<tr>
<td>Case 3</td>
<td>≥ 4915.2</td>
<td>&lt; 4915.2</td>
<td>Enabled</td>
<td>x</td>
</tr>
<tr>
<td>Case 4</td>
<td>≥ 4915.2</td>
<td>≥ 4915.2</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

| Example 1        | 6144  | 9830.4| Enabled  | Enabled |
| Example 2        | 9830.4| 6144  | Enabled  | x      |

<table>
<thead>
<tr>
<th>Clock Network Switch (TX PLL Switching)</th>
<th>From</th>
<th>To</th>
<th>Power Up</th>
<th>Manual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>&lt; 4915.2</td>
<td>&lt; 4915.2</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Case 2</td>
<td>&lt; 4915.2</td>
<td>≥ 4915.2</td>
<td>x</td>
<td>Enabled</td>
</tr>
<tr>
<td>Case 3</td>
<td>≥ 4915.2</td>
<td>&lt; 4915.2</td>
<td>Enabled</td>
<td>x</td>
</tr>
<tr>
<td>Case 4</td>
<td>≥ 4915.2</td>
<td>≥ 4915.2</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
</tbody>
</table>

**Note:** If you have channels that need to enable DCD calibration IP from both the left and right sides of the device, you must use one transceiver reconfiguration controller per side of the device. This applies to both power-up and manual DCD mode.

If your design is using more than one Reconfiguration Controller (per side of the device) and the data channels are running ≥ 4915.2 Mbps, you must chain the Reconfiguration Controller. The following figure shows the chaining method. The chaining is required from the first Reconfiguration Controller until the last Reconfiguration Controller (per side of the device), as long as the DCD calibration IP is enabled.

The purpose of chaining the Reconfiguration Controller is to allow the DCD calibration process to execute sequentially. For example, in the following figure, by chaining the tx_cal_busy signal in the upstream Reconfiguration Controller to the cal_busy_in port of the downstream Reconfiguration Controller, the DCD is calibrated for the transceiver channels connected to the upstream controller before the transceiver channels connected to the downstream controller.

The cal_busy_in port will reset the downstream Reconfiguration Controller when the upstream Reconfiguration Controller is busy with the DCD calibration. The cal_busy_in port of the upstream Reconfiguration Controller is connected to ground. The tx_cal_busy port of the downstream Reconfiguration Controller can be left unconnected.

**Note:** If you are using only one Reconfiguration Controller with DCD calibration enabled, the cal_busy_in port must be tied to ground.

This process is handled by the Reconfiguration Controller internally. You only need to connect the ports as described for proper functionality.
As this design example demonstrates, the Reconfiguration Controller provides an easy and efficient method to dynamically change the Arria V GX Native PHY's settings, including TX PLL switching, VOD setting updates, and triggering the DCD calibration process during user mode.

### Document Revision History

#### Table 11: Document Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>December 2015</td>
<td>2015.12.04</td>
<td>Updated URLs for links in several sections.</td>
</tr>
<tr>
<td>March 2015</td>
<td>2015.03.04</td>
<td>Corrected the &quot;Value Written&quot; entry in step 5 of the &quot;Using the Register-Based Reconfiguration Method to Trigger DCD Calibration&quot; table.</td>
</tr>
<tr>
<td>April 2014</td>
<td>2014.04.01</td>
<td>Added a link to the reference design example in the &quot;Arria V GX Dynamic Reconfiguration Design Example&quot; section.</td>
</tr>
</tbody>
</table>
| January 2014| 2014.01.21 | • Updated the "Creating a ROM that Contains the MIF for Reconfiguration" section.  
• Updated the "Compiling the Design Example" section.  
• Updated the "Manual Trigger for DCD Calibration IP via Register-based Reconfiguration" section. |
<p>| October 2013| 2013.10.11 | Updated the &quot;Using the Register-Based Reconfiguration Method to Reconfigure VOD Settings&quot; table. |</p>
<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
</table>
| April 2013| 2013.04.11| • Updated the "Using the Register-Based Reconfiguration Method to Trigger DCD Calibration" table.  
• Added a note after the "DCD Calibration Usage" table. |
| March 2013| 2013.03.01| Initial release.                                                          |