Table of Contents

Introduction ........................................................................................................................................... 3
Requirements ......................................................................................................................................... 3
Theory of Operation .............................................................................................................................. 4
How to Setup the Hardware for Link Test ......................................................................................... 8
How to Reconstruct and Running the Reference Design ................................................................. 8
Conclusion ............................................................................................................................................ 10
References .......................................................................................................................................... 10
Revision History ................................................................................................................................. 10
Introduction

The objective of this design example is to demonstrate how to real-time enable and disable different loopback modes supported in the Arria® 10 transceiver using direct reconfiguration flow. Arria 10 devices have three loopback modes - serial loopback, reverse serial loopback pre-CDR and reverse serial loopback post-CDR. These loopback modes are generally used in transceiver hardware debugging to narrow down issue due to signal integrity, external signal source and CDR block. The design will also show link tests between two transceiver channels with these loopback modes in Arria 10 GX Transceiver Signal Integrity Development Kit.

Requirements

The reference design requires the following hardware and software to run the test:
- Quartus® Prime Software Version: 16.0
- Two pairs of matched-length 2.4mm SMA cables

Figure 1. Arria 10 GX Transceiver Signal Integrity Development Kit
Theory of Operation

Figure 2. Block diagram of modules in the reference design

Figure 2 shows the high-level modules in the reference design as well as the interfaces among the modules. Blue colored modules refer to IPs generated by Quartus while orange colored modules refer to user coded logics. In this design, two transceiver channels are instantiated. The transmitter of Channel 1 – TX1 is connected to the receiver of Channel 0 – RX0. The transmitter of Channel 0 – TX0 is then connected to the receiver of Channel 1 – RX1. The loopback controller is a state machine which allows user to real-time control the ON/OFF of the three loopback modes.

Arria 10 devices support three loopback modes:

- Serial Loopback
- Reverse Serial Loopback (Pre-CDR)
- Reverse Serial Loopback (Post-CDR)

Figure 3 shows the data path of the serial loopback mode. In this loopback mode, the serial data from the transmitter serializer is loopback directly to the CDR within the same transceiver channel. The serial data from the receiver serial input pin is ignored.
The following shows the high-level steps to enable the serial loopback mode using direct write reconfiguration flow:

- Request access to AVMM bus of targeted channel
- Perform a read-modify-write to address 0x2E1 to set bit 0 to 1'b1
- Release the AVMM bus to PresICE

The following shows the high-level steps to disable the serial loopback mode using direct write reconfiguration flow:

- Request access to AVMM bus of targeted channel
- Perform a read-modify-write to address 0x2E1 to set bit 0 to 1'b0
- Release the AVMM bus to PresICE

Note that you can also enable the serial loopback mode by turning on Enable rx_serialpbken port in the Native PHY IP Parameter Editor and driving the port to 1'b1.

Figure 4 shows the data path of the Reverse Serial Loopback (Pre-CDR) mode. In this loopback mode, the serial data received through the RX input buffer is loopback to the TX output buffer. Note that the received data is also available to the FPGA fabric through the RX parallel data output signals.
Figure 4. Reverse Serial Loopback (Pre-CDR) Data Path

Figure 5 shows the data path of the Reverse Serial Loopback (Post-CDR) mode. In this loopback mode, the serial data received pass through the RX CDR and then loopback to the TX output buffer. Note that the received data is also available to the FPGA fabric through the RX parallel data output signals.

Figure 5. Reverse Serial Loopback (Post-CDR) Data Path

To enable the reverse serial loopback (pre-CDR) mode, perform read-modify-write operation to set the follow bit values:

<table>
<thead>
<tr>
<th>Address</th>
<th>Bit Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x137[7]</td>
<td>1'b1</td>
</tr>
<tr>
<td>0x13C[7]</td>
<td>1'b0</td>
</tr>
<tr>
<td>0x132[5:4]</td>
<td>2'b00</td>
</tr>
<tr>
<td>0x142[4]</td>
<td>1'b1</td>
</tr>
<tr>
<td>0x11D[0]</td>
<td>1'b1</td>
</tr>
</tbody>
</table>
To enable the reverse serial loopback (post-CDR) mode, perform read-modify-write operation to set the follow bit values:

<table>
<thead>
<tr>
<th>Address</th>
<th>Bit Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x137[7]</td>
<td>1'b0</td>
</tr>
<tr>
<td>0x13C[7]</td>
<td>1'b1</td>
</tr>
<tr>
<td>0x132[5:4]</td>
<td>2'b01</td>
</tr>
<tr>
<td>0x142[4]</td>
<td>1'b0</td>
</tr>
<tr>
<td>0x11D[0]</td>
<td>1'b0</td>
</tr>
</tbody>
</table>

To disable the reverse serial loopback mode (pre-CDR and post-CDR), perform read-modify-write operation to set the follow bit values:

<table>
<thead>
<tr>
<th>Address</th>
<th>Bit Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x137[7]</td>
<td>1'b0</td>
</tr>
<tr>
<td>0x13C[7]</td>
<td>1'b0</td>
</tr>
<tr>
<td>0x132[5:4]</td>
<td>2'b00</td>
</tr>
<tr>
<td>0x142[4]</td>
<td>1'b0</td>
</tr>
<tr>
<td>0x11D[0]</td>
<td>1'b0</td>
</tr>
</tbody>
</table>

The loopback controller module which performs the above register writing process in the design is reverse_lpbk_en.v. Note that this simple controller is just for reference only and it is recommended for you to code your own controller. The SEL[2:0] input of the controller allows you to select the operation modes that you would like to perform. Table 1 shows the mapping of the SEL[2:0] to the loopback controller operation mode. After selecting the operation mode, you can assert the source[3] of the ISSP to start the operation.

The following table shows the mapping of the SEL[2:0]:

<table>
<thead>
<tr>
<th>SEL[2:0]</th>
<th>Operation Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>3'b000</td>
<td>OFF serial loopback</td>
</tr>
<tr>
<td>3'b001</td>
<td>ON serial loopback</td>
</tr>
<tr>
<td>3'b010</td>
<td>ON reverse serial loopback (pre-CDR)</td>
</tr>
<tr>
<td>3'b011</td>
<td>ON reverse serial loopback (post-CDR)</td>
</tr>
<tr>
<td>3'b100</td>
<td>OFF reverse serial loopback (pre-CDR and post-CDR)</td>
</tr>
</tbody>
</table>

| Table 1. Loopback Controller SEL[2:0] Operation Modes |

In the design, data monitoring is done using SignalTap II Logic Analyzer. The In-System Source and Probe (ISSP) is used to allow real time interface with the design through JTAG. ISSP Editor is used to enable the loopback in this design. Fixed data pattern of 0x0BC is fed to TX1 and 0x0CC is fed to TX0 to ease the data monitoring.
The following are the expected observation with different loopback modes:

- If the serial loopback mode is successfully enabled, we should observe the fixed data pattern of 0x0CC or data with correct serial bit stream of 0x0CC at the parallel output of RX0 and RX1.
- If the reverse serial loopback (pre-CDR or post-CDR) mode is successfully enabled, we should observe the fixed data pattern of 0x0BC or data with correct serial bit stream of 0x0BC at the parallel output of RX0 and RX1.

**How to Setup the Hardware for Link Test**

Follow these steps to setup the hardware to run the reference design:

1. Connect the one pair of 2.4mm SMA cables from GXB_1G_TX1 (TX1) to GXB_1G_RX0 (RX0) on the Arria 10 GX Transceiver Signal Integrity Development Kit.
2. Connect another pair of 2.4mm SMA cables from GXB_1G_TX0 (TX0) to GXB_1G_RX1 (RX1) on the Arria 10 GX Transceiver Signal Integrity Development Kit.
3. Use the default switching settings. For more details about default switching settings, please refer to Arria 10 GX Transceiver Signal Integrity Development Kit user guide.
4. Connect the USB cable to the USB Blaster connector on the development kit.
5. Connect the power adapter shipped with the development board to power supply jack.
6. Turn On the power for the Arria 10 GX Transceiver Signal Integrity Development Kit. The hardware system is now ready for programming.

**How to Reconstruct and Running the Reference Design**

Follow these steps to reconstruct, compile and run the design:

1. Follow the instruction in the Design Store to prepare the design template and load the design into your Quartus software.
2. Please note that the design downloaded from Design Store does not enable the Signaltap II Logic Analyzer in Quartus project. To include the Signaltap II file into Quartus project, navigate to the Assignments menu and select Settings.
3. Click on the Signaltap II Logic Analyzer under Category, and then check the Enable Signaltap II Logic Analyzer checkbox.
4. Browse the stp1.stp file in the project directory and hit OK button.
5. Perform full compilation with the design.
6. Program the SOF file generated into the Arria 10 GX Transceiver Signal Integrity Development Kit.
7. After the programming is completed, open the SignalTap and establish connection to the device.
8. Click on the Autorun Analysis to start signal sampling.
9. Go to Tools -> In-System Source and Probe Editor, launch the ISSP Editor and establish connection to the device.
10. Perform a reset to the transceiver channels by writing 1’b1 followed by 1’b0 to the source[0] in the ISSP Editor.
11. Go back to the SignalTap and you should observe the default RX parallel data output for RX0 and RX1 as shown in Figure 6
12. Go to ISSP Editor. Write 3’b001 to the source[6:4] to select the loopback controller operation mode of “ON serial loopback”. Source[6:4] is connected to SEL[2:0] in the design
13. Then, write 1’b1 to the source[3] to enable the serial loopback in Channel 0
14. Go back to the SignalTap and now you should observe RX0 parallel data output is showing the correct serial bit stream of the 0x0C sent but with different word boundary as in Figure 7
15. To disable the serial loopback, go back to ISSP Editor. Write 3’b000 to the source[6:4] to select the loopback controller operation mode of “OFF serial loopback”
16. Then, write 1’b1 to the source[3] to disable the serial loopback in Channel 0
17. To enable the reverse serial pre-CDR loopback, go to ISSP Editor, write 3’b010 to the source[6:4] to select the loopback controller operation mode of “ON reverse serial loopback (pre-CDR)”
18. Then, write 1’b1 to the source[3] to enable the reverse serial pre-CDR loopback in Channel 0
19. You should observe RX1 parallel data output is showing the correct serial bit stream of the 0x0BC sent but with different word boundary as in Figure 8
20. To disable the reverse serial loopback, go back to ISSP Editor. Write 3’b100 to the source[6:4] to select the loopback controller operation mode of “OFF reverse serial loopback (pre-CDR and post-CDR)”
21. Then, write 1’b1 to the source[3] to disable the reverse serial loopback in Channel 0
22. You may repeat Steps 16 – 18 to enable the reverse serial post-CDR loopback by writing 3’b011 to the source[6:4]
23. You should then observe RX1 parallel data output is showing the correct serial bit stream of the 0x0BC sent but with different word boundary as in Figure 9

Figure 6. SignalTap Result Without Any Loopback Modes Enabled

Figure 7. SignalTap Result After Enabling Serial Loopback Mode
Conclusion

The design example provides a reference on how to real-time enable and disable different loopback modes supported in the Arria 10 transceiver using direct reconfiguration flow.

References

- Arria 10 Transceiver PHY User Guide
  

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>March 28, 2017</td>
<td>1.0</td>
<td>Initial Release</td>
</tr>
</tbody>
</table>