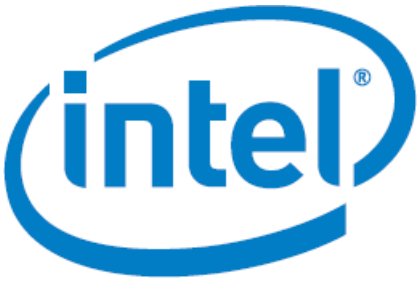

Cyclone 10 GX Dynamic Reconfiguration with Transmitter PLL Switching by using Nios II processor Reference Design 17.1 User Guide

Date: January 2018

Revision: 1.0

Table of Contents

Introduction

The objective of this design example is to demonstrate the implementation of Cyclone 10 GX Native PHY Transmitter PLL switching, channel reconfiguration with embedded streamer or direct write method as well as recalibration. The two ATX PLLs or FPLL are used to support two different data rates which could not be achieved with TX local divider. The purpose of this design example is to assist users to have quick start with the Cyclone 10 GX transceiver dynamic reconfiguration. The design starts with transceiver channel running at 2Gbps data rate and then reconfigure to 1.5Gbps using Transmitter PLL switching and channel reconfiguration. After reconfiguration is completed, a channel recalibration is performed. Incremental data is sent from the TX and loopback to the RX for monitoring. The dynamic reconfiguration and recalibration commands are performed through Nios II processor.

Requirement

The reference design requires the following hardware and software to run the test:

• Quartus® Prime Software Version: 17.1 Pro Edition

• Cyclone 10 GX Development Kit

https://www.altera.com/products/boards_and_kits/dev-kits/altera/cyclone-10-gx-development-kit.html

• FPGA Mezzanine Card (FMC) loopback card

Theory of Operation

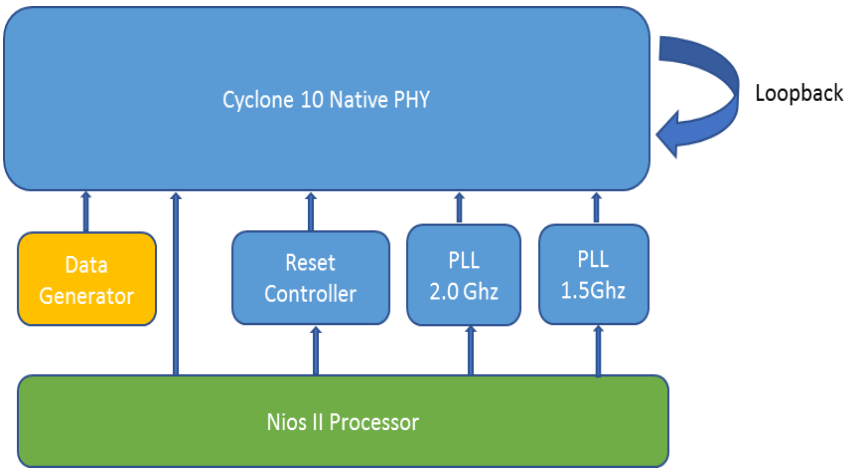

Figure 1. Block diagram of modules in the reference design

Figure 1 shows the high-level modules in the reference design as well as the interfaces among the modules. The Cyclone 10 GX transceiver Native PHY is used to configure and implement the hard transceiver channels. The Reset Controller is used to perform the reset to the transceiver channel and transmitter PLL. The Nios II processor is used to interface with the Native PHY to execute the commands for transceiver dynamic reconfiguration and recalibration. Incremental counter value is fed to the TX and loopback to RX through FMC loopback card. The design comes with frequency counters which are used to monitor the tx_clkout and rx_clkout frequencies to verify if the data rate change is done successfully.

After power up, by default the transceiver channel will be running at 2Gbps. With the PCS-PMA interface width = 10bits, the expected tx_clkout and rx_clkout frequencies are 200MHz. User can use the on-board push button to select the desire data rate. If push button 0 is pressed, the transceiver rate change is done by switching the transmitter PLL that the Native PHY is listening to. After reconfiguring the transceiver channel to 1.5Gbps, the expected tx_clkout and rx_clkout frequencies are 150MHz. User can also use the push button to select either using Direct Write Method or embedded steamer to perform the reconfiguration.

Push Button	Function
0 (S8)	Switching to 1.5Gbps
1 (S10)	Switching to 2.0Gbps
2(S11)	Direct Write Method or embedded streamer

Table 1. function of the on-board push button

The following shows the high-level steps to perform the transceiver dynamic reconfiguration with transmitter PLL switching:

- Assert reset to the Native PHY
- Request access to AVMM bus of targeted channel
- Perform transmitter PLL switching
- Perform a channel reconfiguration by switching the profile using embedded streamer or direct write method
- Perform channel recalibration
- Release the reset of the Native PHY

For further details on transmitter PLL switching dynamic reconfiguration, you may refer to the Intel Cyclone 10 GX Transceiver PHY User Guide -> "Switching Transmitter PLL" section.

For further details on transceiver channel reconfiguration, you may refer to the Intel Cyclone 10 GX Transceiver PHY User Guide -> “CDR/CMU PLL Recalibration” and “PMA Recalibration” sections.

How to Setup the Development Kits for Link test

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

1. Connect the FMC loopback card to the FMC port on the Cyclone 10 GX Development Kit
2. Use the default switching settings of the development kit
3. Connect the Micro USB cable to the USB Blaster connector on the development kit
4. Connect the power adapter shipped with the development board to power supply jack
5. Turn On the power for the Cyclone 10 GX Development Kit. The hardware system is now ready

How to Reconstruct and Running the Reference Design

Hardware setup:

1. Follow the instruction in the Design Store to prepare the design template and load the design into your Quartus software
2. By default, the design use ATX PLL, to change to use fPLL as transmitter PLL, open the top.sv file, and change the "ATXPLL" parameter to 0.
3. Perform full compilation with the Intel Quartus Prime design
4. Do the following to reduce the JTAG frequency to 16M to avoid JTAG connection issue

   a. Launch NIOS II command shell from Windows Intel Quartus Prime Pro v17.1 startup menu

   b. Follow the commands in Figure 2 to configure the JTAG frequency

```
                    /cygdrive/c/intelFPGA_pro/17.1
$ jtagconfig --getparam 1 JtagClock
24M

                    /cygdrive/c/intelFPGA_pro/17.1
$ jtagconfig --setparam 1 JtagClock 16M

                    /cygdrive/c/intelFPGA_pro/17.1
$ jtagconfig --getparam 1 JtagClock
16M

                    /cygdrive/c/intelFPGA_pro/17.1
$ jtagconfig
1) USB-BlasterII [USB-1]
  031820DD   10M08SA(.|ES)/10M08SC
  02E120DD   10CX220Y
```

Figure 2. Commands to Reduce the JTAG Frequency to 16M

5. Only execute step 5 if your C10 GX Development Kit Board belongs to 1st prototype build with board serial number within 0000001-0000030 else skip to step 6. For 1st prototype board, by default after power up the development kit board, the programmable clock generator (Si5332) used for the transceiver channel in this design is disabled. Do the following to enable and configure the output clock frequencies correctly

   a. Download and unzip "Kit Collateral.zip" design package from below link
   https://www.altera.com/products/boards_and_kits/dev-kits/altera/cyclone-10-gx-development-kit.html

   b. Launch Quartus software first then run the ClockController.exe from cyclone-10-gx-kit-collateral\examples\board_test_system

   c. By default, the Clock Controller GUI should look like Figure 3

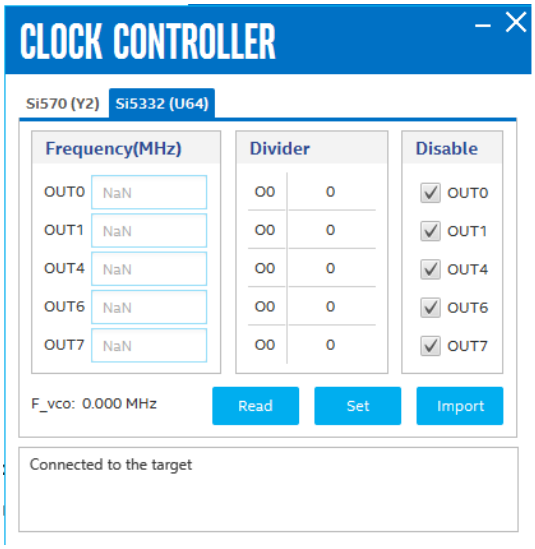

Figure 3. Default Clock Controller GUI for Si5332

d. Click the "Import" button in Clock Controller GUI and select the "U64-Registers.txt" in cyclone-10-gx-kit-collateral\examples\board_test_system folder to configure the output clock frequencies

e. After successful configuration, the Clock Controller GUI should look like Figure 4

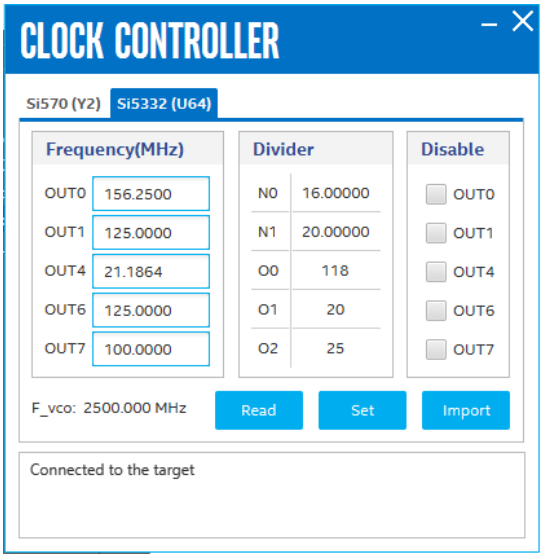

Figure 4. Clock Controller GUI for Si5332 after Successful Configuration

f. Close the Clock Controller application

g. Note that you would need to reconfigure the Si5332 using step 5(b) -> 5(f) each time the development kit is power-cycled

6. Program the SOF file generated into the Cyclone 10 GX Development Kit

Nios II Eclipse setup:

1. Open Nios II Eclipse
2. From Eclipse menu -> File -> New -> Nios II Application and BSP from Template
3. Load the SOPC information file.
    a. For ATX PLL design use “Nios_Native.sopcinfo” that located at Nios_Native folder
    b. For fPLL design use “Nios_Native_fPLL.sopcinfo” that located at Nios_Native_fPLL folder
4. Select a blank project and provide a project name
5. Click next, and then Finish
6. Copy the main.c file from Software folder into the project
7. Go to the “Nios II BSP properties” by right click the <<project_name>>_BSP folder at the Eclipse project explorer panel, and select properties
8. Enable the support of small C library and then click Apply. The Nios II BSP Properties should look like Figure 5.

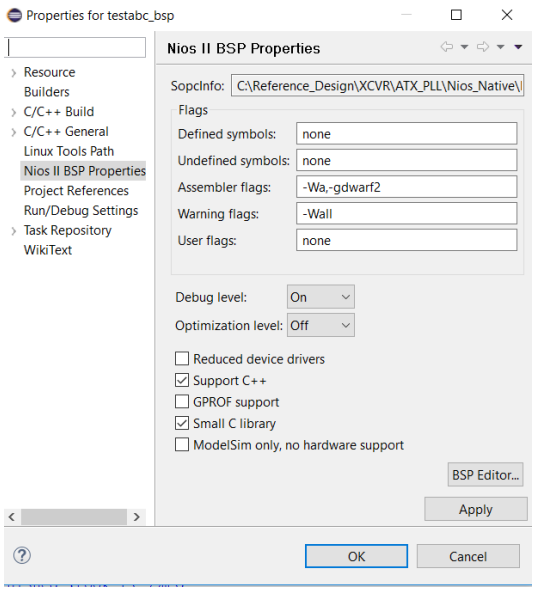

Figure 5. Nios II BSP Properties

9. From the Eclipse project explorer panel, right click the project folder and select Build Project
10. From the Eclipse project explorer panel, right click the project folder and select Run As -> Nios II Hardware
11. From the Run Configurations GUI, go to Target Connection and tick the Ignore Mismatched system ID and Ignore mismatched system timestamp, and then click Apply follow by Run
12. After successfully run the software, you will see the push button option at Nios II Console as Figure 6

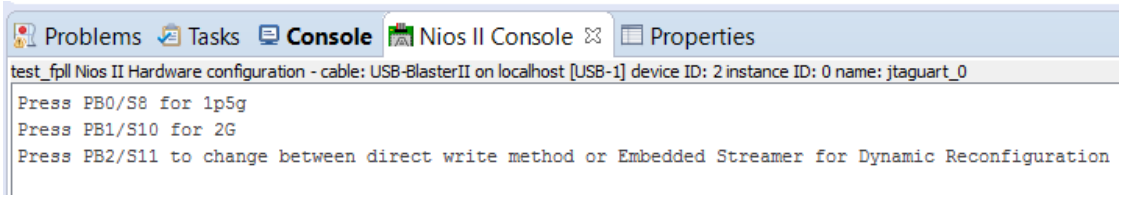

Figure 6. Nios II Console

13. By default, this design use Embedded Streamer Dynamic Reconfiguration. To change the Dynamic Reconfiguration method, user can press Push Button 2.
14. When Push Button 0 is pressed, the design starts dynamic reconfiguration to 1.5Gpbs, and after the reconfiguration is completed, the RX_CLKOUT and TX_CLKOUT are changed to 150Mhz as Figure 7

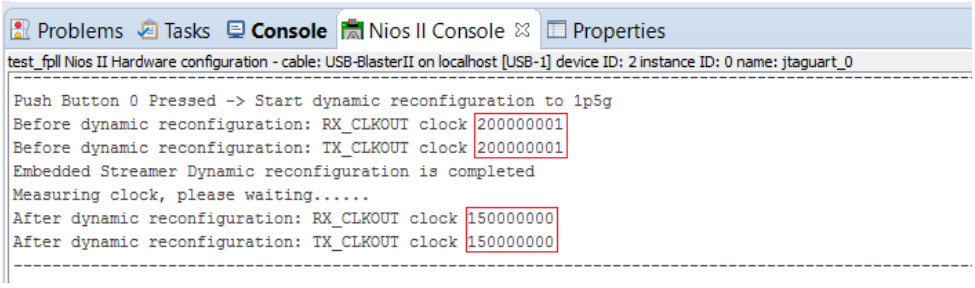

Figure 7. Result of Push Button 0 is pressed

15. When Push Button 1 is pressed, the design starts dynamic reconfiguration to 2Gbps, and after the reconfiguration is completed, the RX_CLKOUT and TX_CLKOUT are changed to 200Mhz as Figure 8

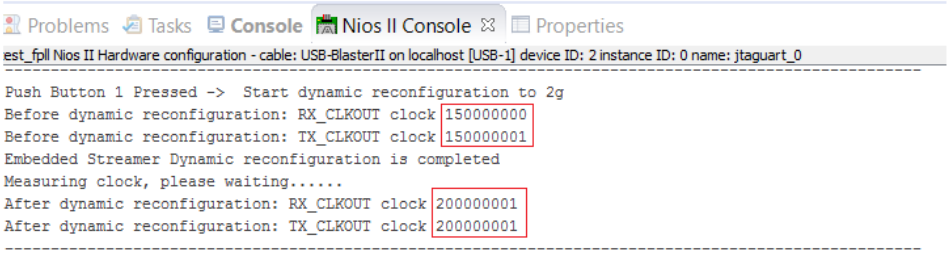

Figure 8. Result of Push Button 1 is pressed

Conclusion

The design example provides a reference on how to perform dynamic transceiver rate change with Transmitter PLL switching, channel reconfiguration with embedded streamer or direct write method as well as recalibration.

References

- Intel Cyclone 10 GX Transceiver PHY User Guide
  https://www.altera.com/documentation/hki1486507600636.html

Revision History

Date	Version	Changes
Jan, 2018	**1.0**	**Initial Release**