



XAPP1228 (v1.0.2)
February 28, 2017

Demonstration of a 16 Lane 10G Aurora 64B/66B Link on VC7203

Author: K Krishna Deepak, Pankaj Kumbhare

Summary

This application note explains the steps required to validate the Xilinx LogiCORE™ IP Aurora 64B/66B IP core working at 10.3125 Gb/s serial line rate and configured as a 16-lane link on the Virtex-7 FPGA VC7203 Characterization Kit. Aurora 64B/66B is a scalable, lightweight, high-data rate, link-layer protocol for high-speed serial communication. Aurora IP cores are designed to enable easy implementation of Xilinx transceivers using an intuitive wizard interface while providing a light-weight user interface on top of which designers can build a serial link. The Aurora protocol specification is open and available upon request. The Aurora core is available free of charge in the Vivado® IP catalog and is licensed for use in Xilinx silicon devices.

Included Systems

The reference design is created using the 2014.3 Vivado Design Suite: System Edition. The Vivado design tools help simplify the task of instantiating, configuring, and connecting IP blocks together to form complex embedded systems. The design also includes VIO and ILA cores to probe the signals.

Introduction

This application note demonstrates the maximum bandwidth of 165 Gb/s that is achievable through the core with the configuration of a 16-lane design with each lane working at 10.3125 Gb/s. This application note details the steps required to configure the Aurora 64B/66B core using the Vivado Design Suite, to validate the operation of the core using the VIO and ILA cores to probe design signals, and to know the core status.

Guidance on 16-Lane Design

As the number of lanes is 16, the Aurora 64B/66B core requires two GT (serial transceiver) reference clocks (only for cores with more than 12 lanes). Any suitable conditioned clock source meeting the GT reference clock specification (*7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 1]) can be used to replicate the example design demo created in this application note. In this application note, a 156.250 MHz GT (serial transceiver) reference clock is chosen which can be sourced from the SuperClock-2 module provided with the VC7203 characterization kit. Based on the completed example design, You can build more complex systems.

The example test setup uses two clock sources from the SuperClock-2 module on each of the VC7203 boards to generate 156.250 MHz clock signals for the 16-lane example. The `INIT_CLOCK` is sourced from the onboard 200 MHz LVDS system clock.

The XC7VX485T-3FFG1761E FPGA on the VC7203 board has a total of 7 GTX transceiver quads available to configure the Aurora 64B/66B core. As this application note is targeted at using a 16-lane Aurora 64B/66B design, any four continuous quad selections are required; each quad contains four channels. You can either choose any four consecutive active quads or have the option of choosing the 16-lanes across the available GT locations as long as they meet the `RefClk` requirements as stated in the *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 1]. Also if the lane-assignment is spread across multiple non-consecutive quads, care has to be taken to meet the Aurora 64B/66B lane skew tolerance from the *LogiCORE IP Aurora 64B/66B Product Guide* (PG074) [Ref 2],



RECOMMENDED: *Ensure that the lane assignment is continuous and the choice of GT reference clocks satisfies the North/South clocking constraints as documented in the 7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 1].

In this application note, the lane assignment is contiguous and the selected quads are Bank 113/114/115/116. This is shown for Aurora 64/66B in the IP catalog as GTXQ0/GTXQ1/GTXQ2/GTXQ3. Ensure that the VC7203-to-VC7203 boards are connected properly with the bulls-eye connectors. If you choose to use a different lane assignment, update the connections accordingly.

For programming of the SuperClock-2 module, refer to the *VC7203 IBERT Getting Started Guide* (UG847) [Ref 3]. Xilinx recommends that IBERT testing be performed before testing the Aurora 64B/66B IP core so that any physical link related issues can be identified and resolved.

Hardware Requirements

For the 16 lane Aurora 64B/66B example demonstration, the following hardware is required.

- Two Virtex-7 FPGA VC7203 characterization boards to demonstrate 16 lanes
- Four bulls-eye to bulls-eye connectors to connect the two FPGA GTs
- Two JTAG platform USB cables
- Two onboard SuperClock-2 programming modules to source the stable GT reference clock for the GTs

The Virtex-7 VC7203 characterization kit is shown in [Figure 1](#).

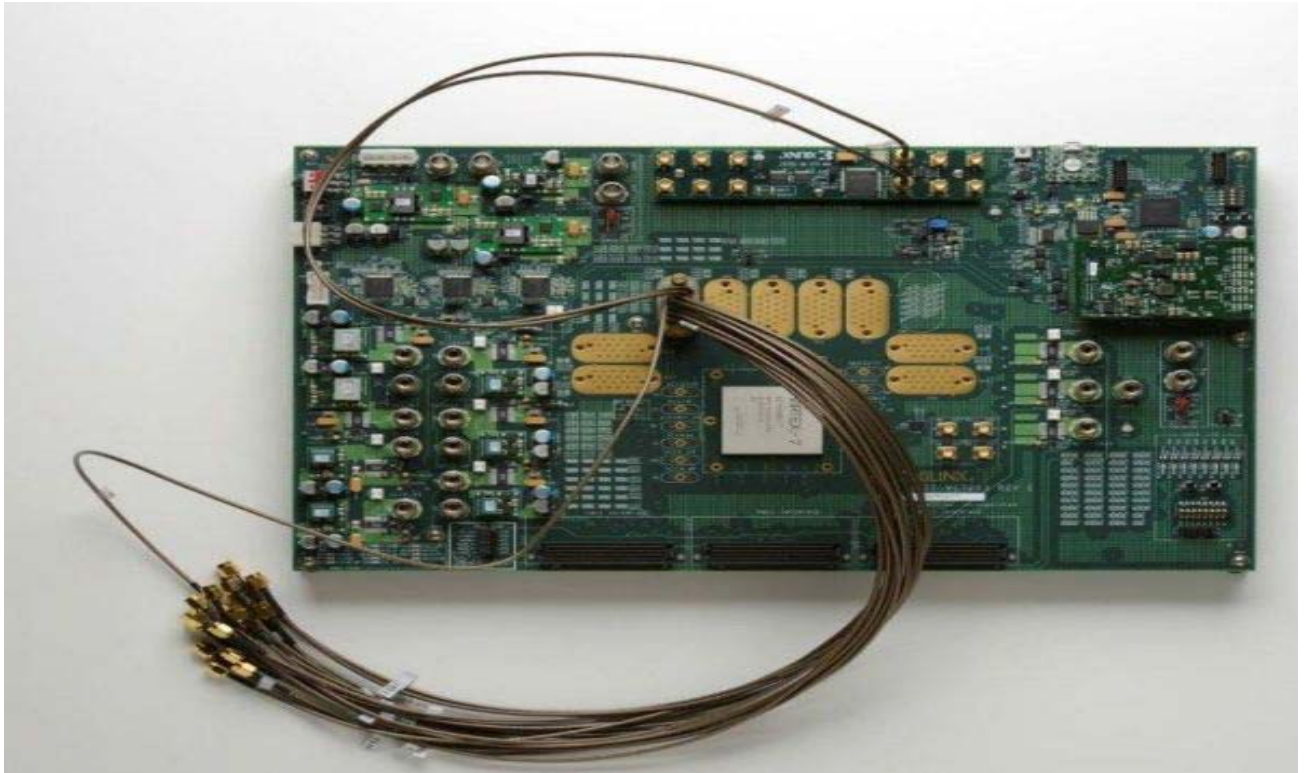


Figure 1: VC7203 Characterization Kit

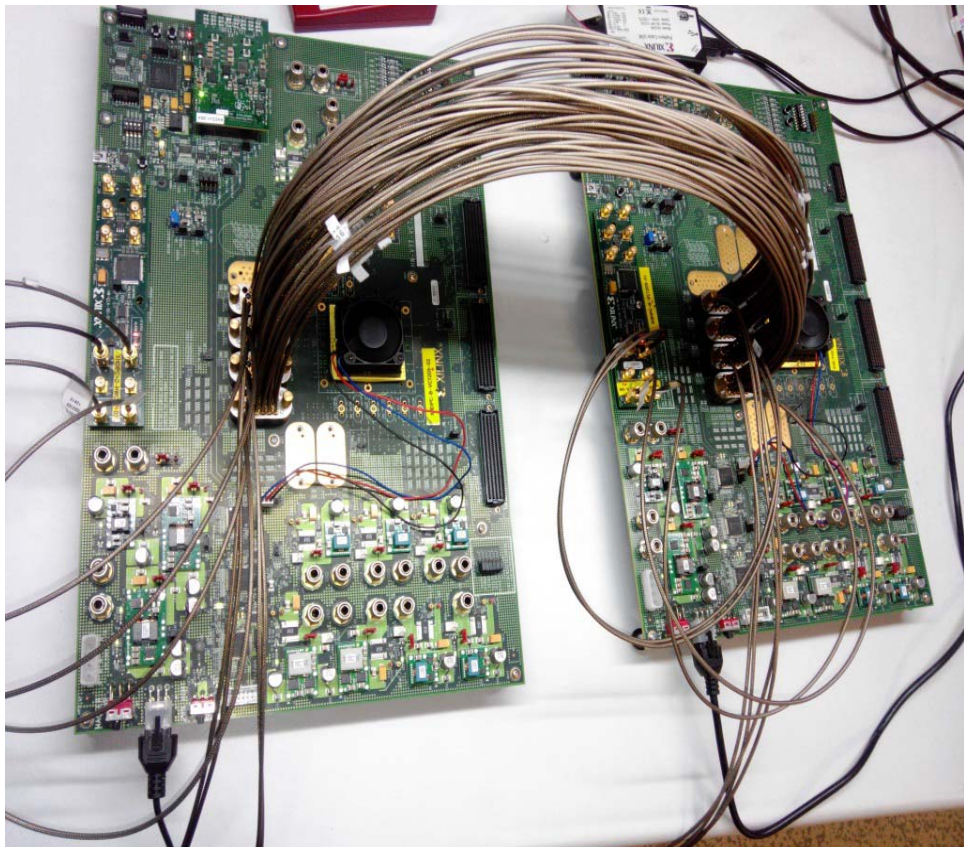


Figure 2: VC7203 Characterization Board Setup for the 16-Lane Aurora 64B/66B Design

Software Requirements

Vivado Design Suite 2014.3: System Edition.

Aurora 64B/66B Core Configuration

- 16 Lane
 - Duplex
 - 10.3125 Gb/s line rate
 - 156.25 MHz GT reference clock (External SuperClock-2 module)
 - Framing Mode
-

Building Hardware

Customizing the Aurora Core

The steps to customize and generate the Aurora 64B/66B IP core are listed in following steps.

1. Open the 2014.3 Vivado Design Suite.
2. Select **Create New Project** and click **Next**.
3. Select the project path and name and click **Next**.

4. Select **RTL Project**, as you will be running an example design. Select the option **Do not specify sources at this time**. Click **Next**.

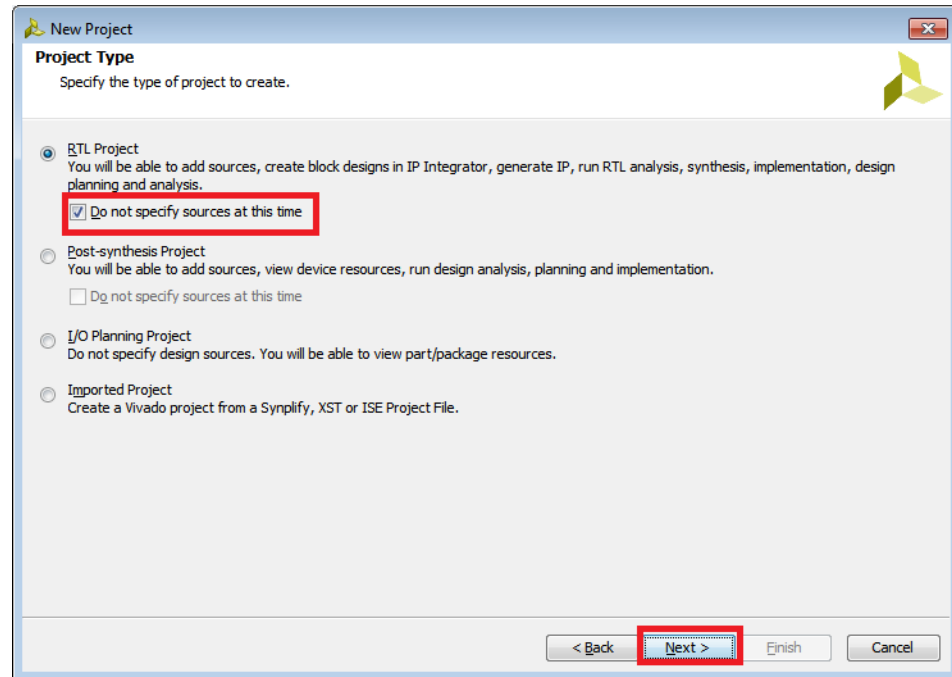


Figure 3: Project Name Page

5. Select the part number, **xc7vx485tffg1761-3** and click **Next**.

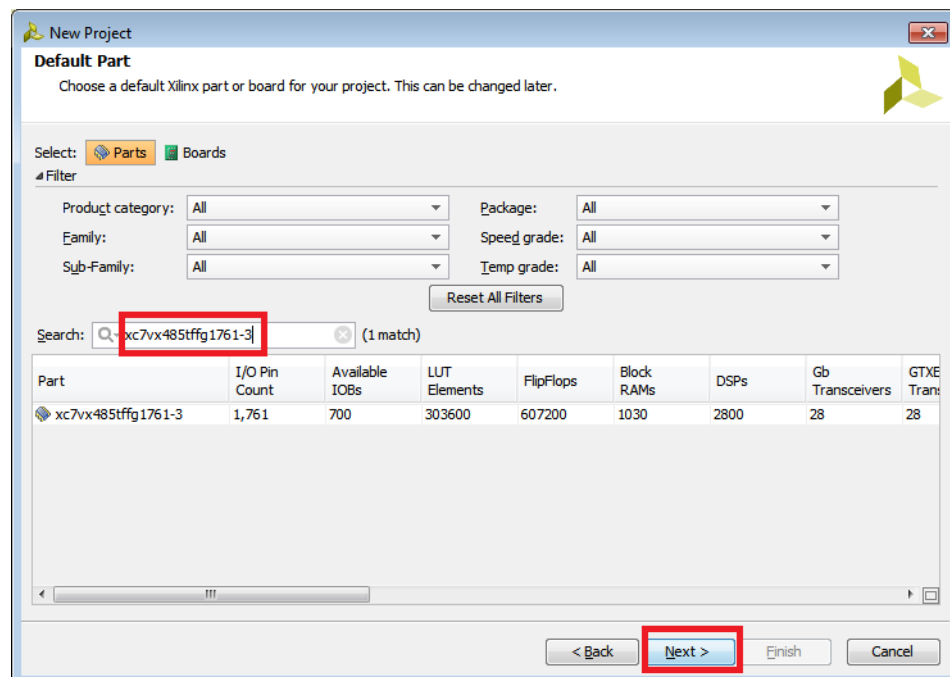


Figure 4: Project Part Selection

- Click **Finish**. The Project Manager screen opens.
- In the Project Manager window, click **IP Catalog**.

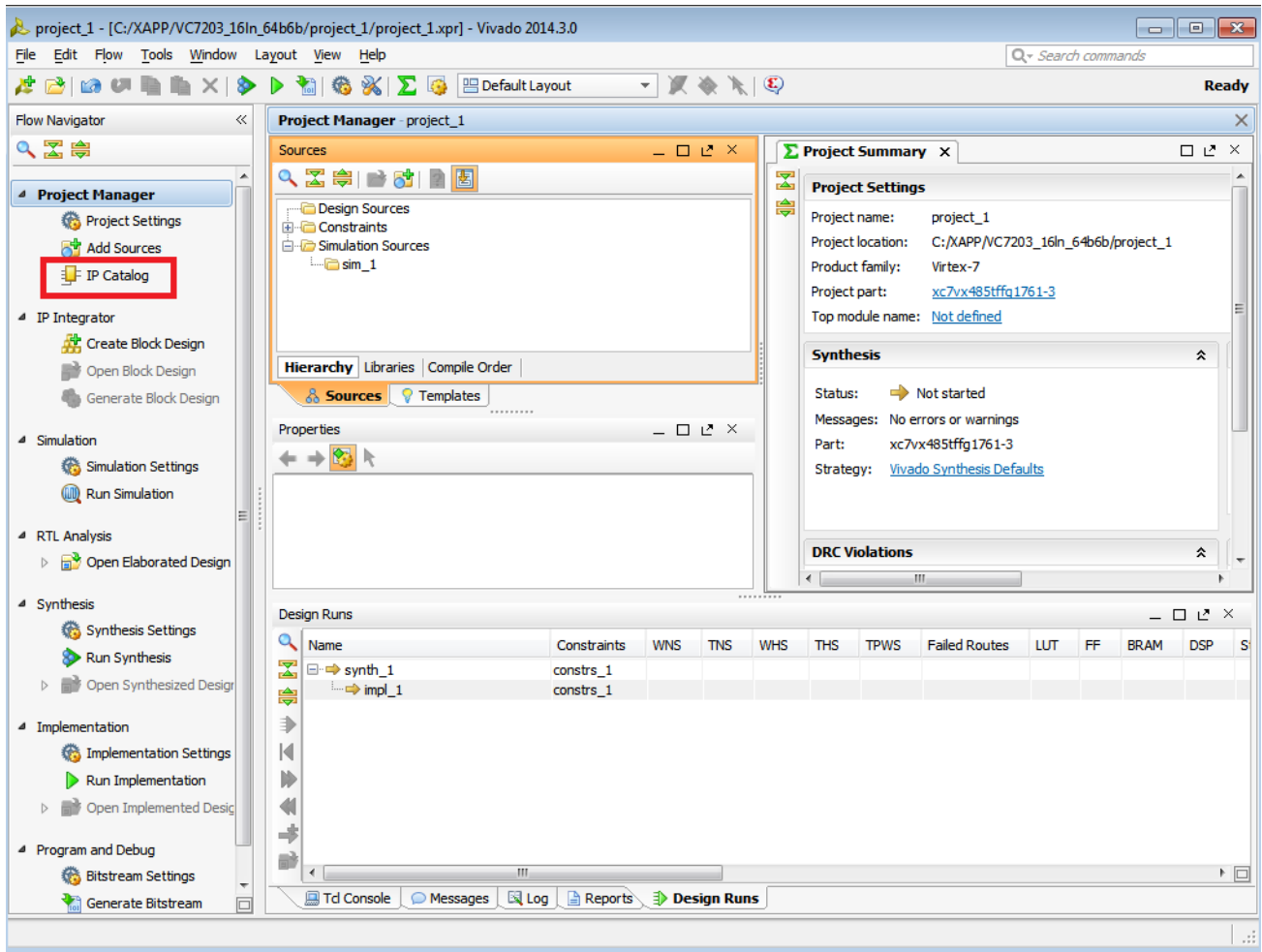


Figure 5: Project Manager

8. Locate **Aurora 64b66b** in the IP Catalog.

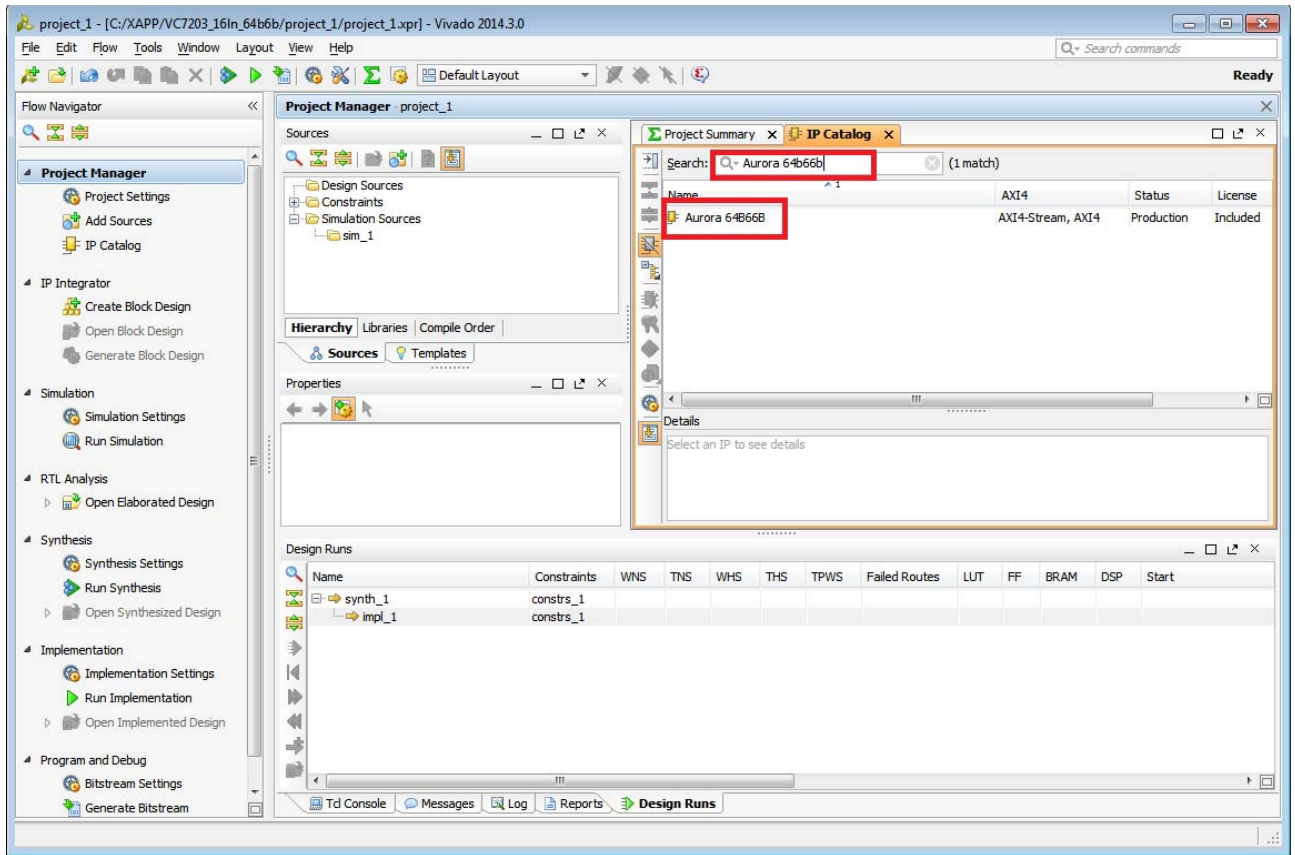


Figure 6: IP Catalog Core Selection Tab

- Double-click **Aurora 64b66b** or alternatively right-click and select the Customize IP. See [Figure 7](#).

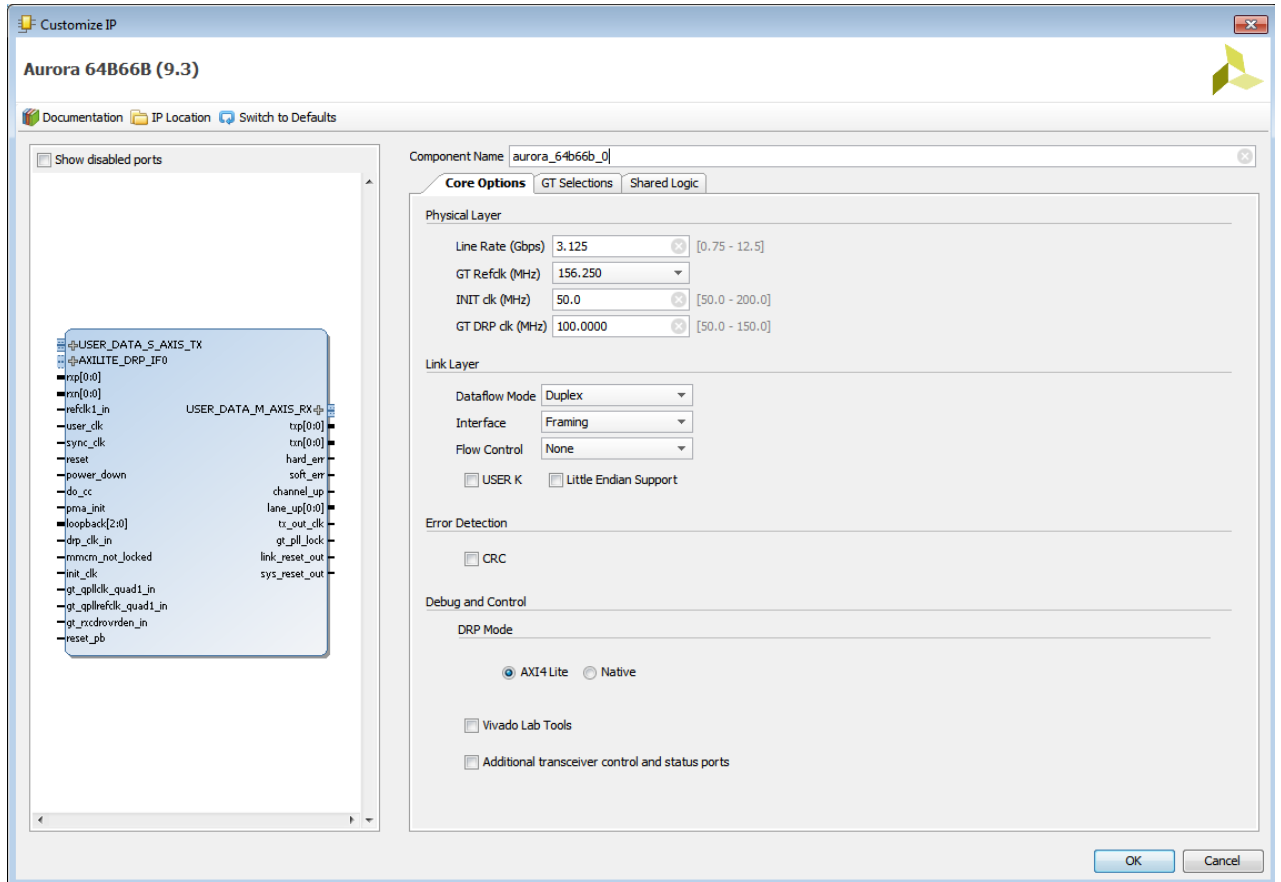


Figure 7: Core Options Tab

10. In the tab **Core Options**, check the **Vivado Lab Tools** option and update the Line Rate to be 10.3125. (You can also select other valid required configurations at this tab.)

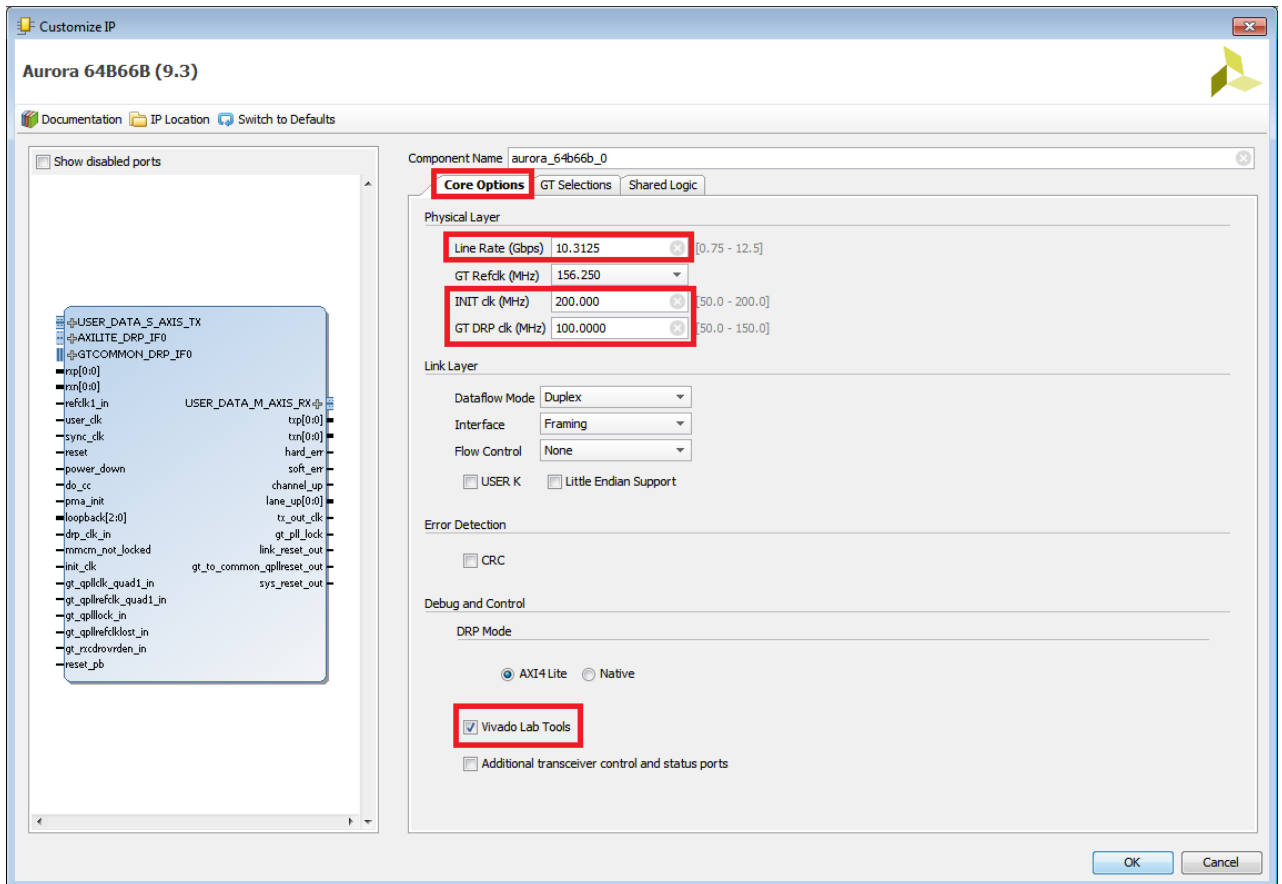


Figure 8: Core Options Tab

- Click the **GT Selections** tab. Customize to have the Lanes as 16 and select the lane assignment as shown in [Figure 9](#).

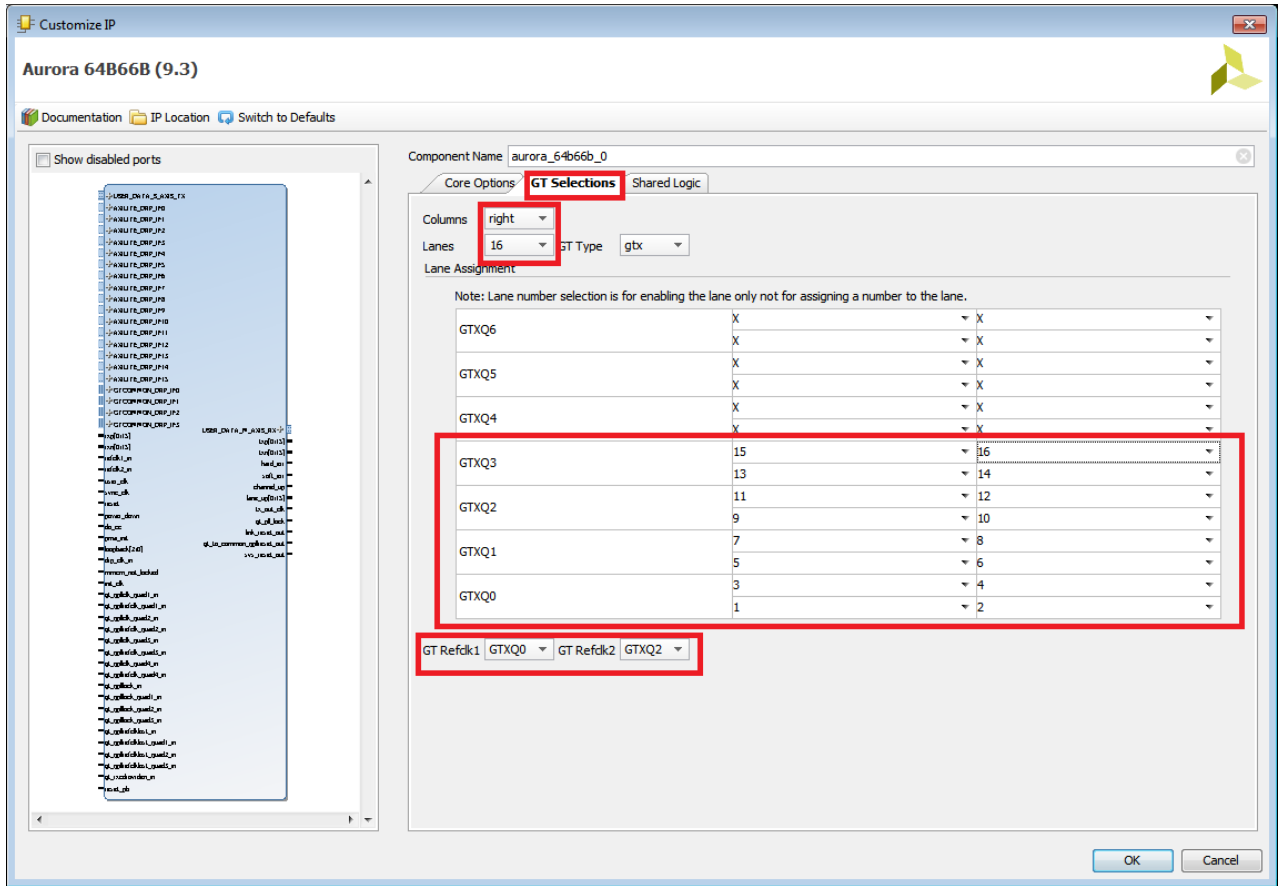


Figure 9: GT Selections Tab

12. Click the **Shared Logic** tab and select any option that you require.

In this application note because the example design is used, the location of the shared logic does not make any difference; however, if you choose to use only the Aurora core, make sure that every GT channel is associated with its appropriate GT common and other shared resources.

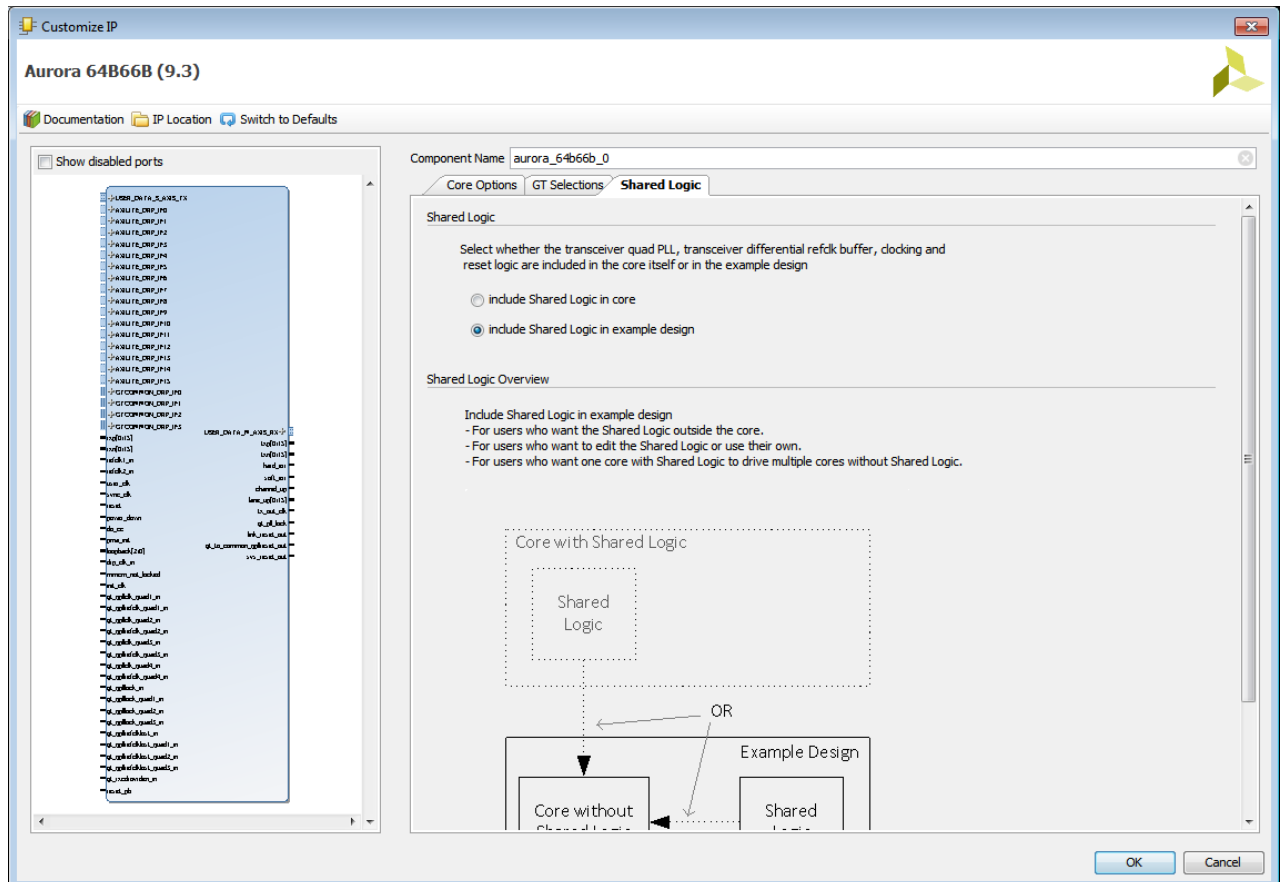


Figure 10: Shared Logic Tab

13. Review the core configuration by checking all three tabs and click **OK**.

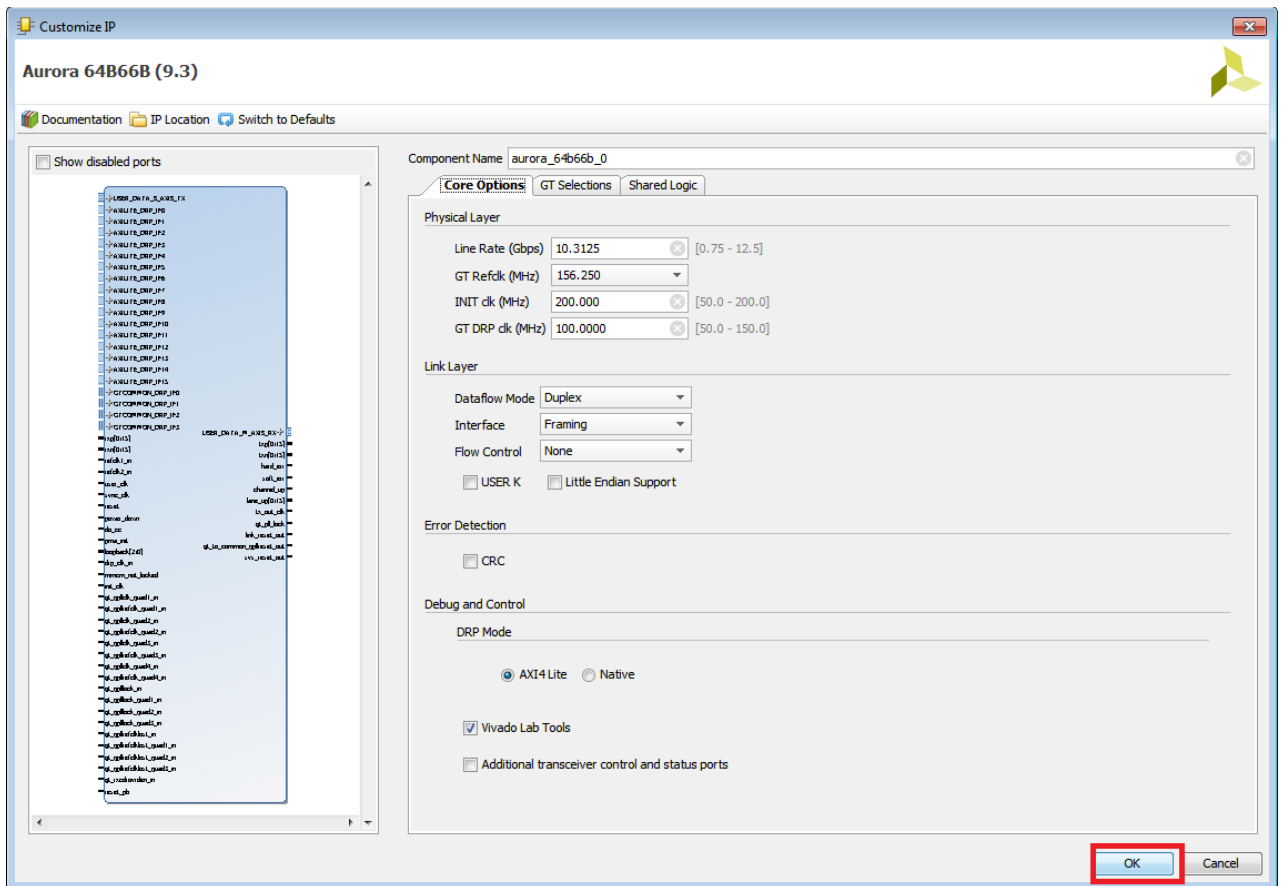


Figure 11: Customize IP, Core Customization Summary Tabs

14. A new page opens. By default **Generate Synthesized checkpoint (.dcp)** is selected in the **Out-of-Context Settings**, This can be confirmed after clicking **Out-of-Context Settings**. If not selected, select **Generate Synthesized checkpoint (.dcp)**, and click **Generate**.

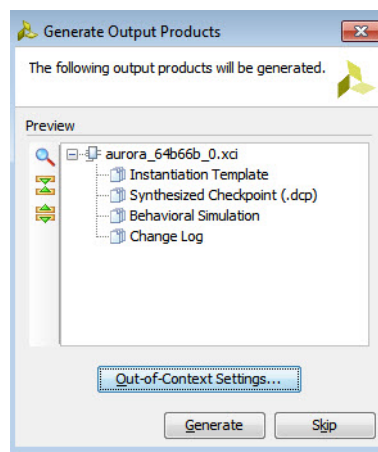


Figure 12: Out-of-Context Settings

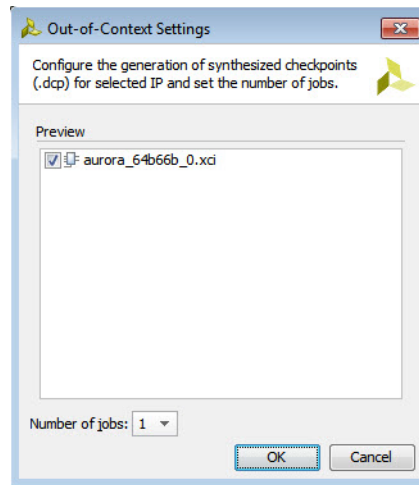


Figure 13: Generate Synthesis Checkpoint

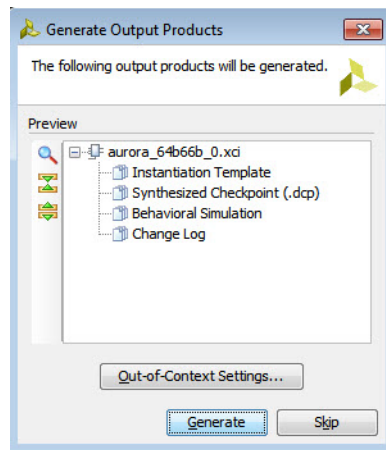


Figure 14: Generate Output Products

Generating the Aurora 64B/66B Example Design Project

1. When the core is generated, In the Project Manager window, right-click the core and select **Open IP Example Design**.
2. Click **OK** after checking the **Overwrite the existing example project**. See [Figure 15](#) and [Figure 16](#).

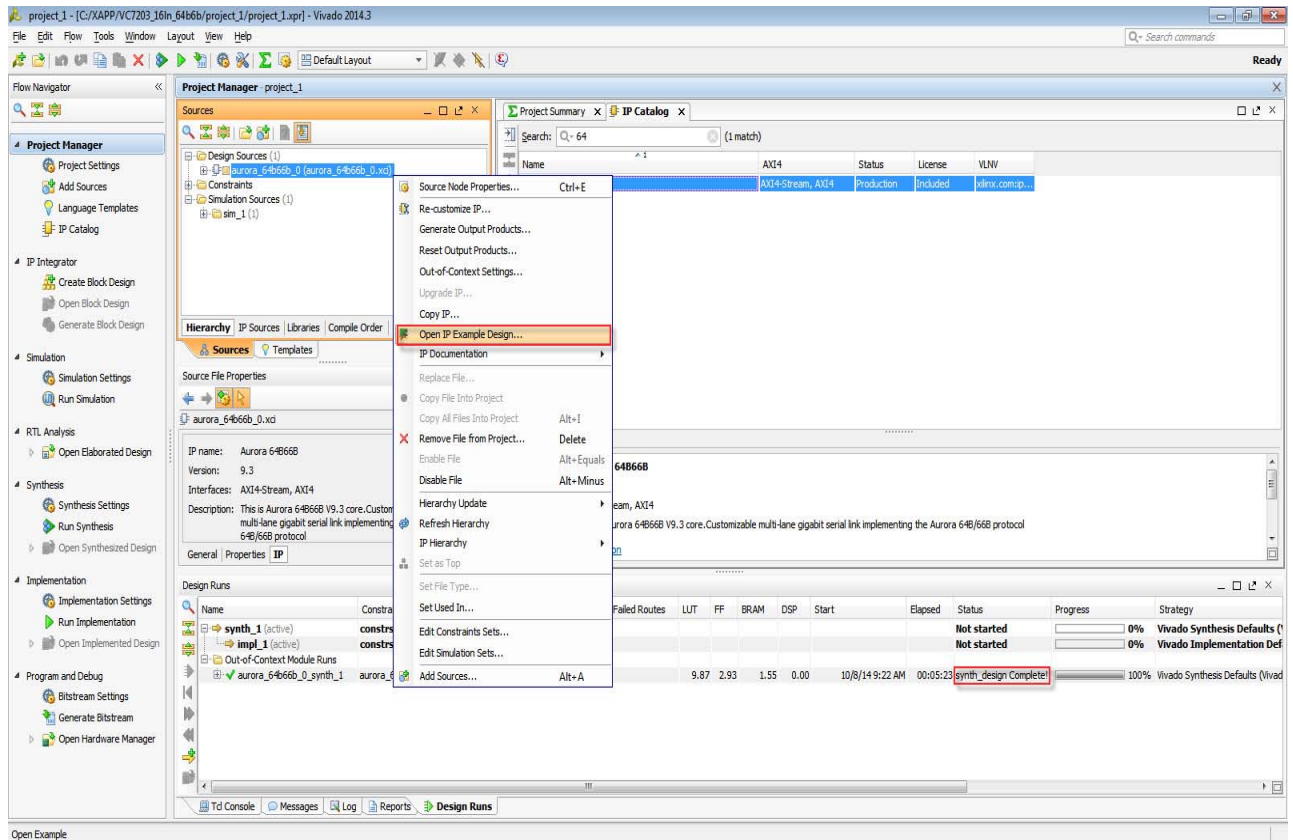


Figure 15: Open Example Design

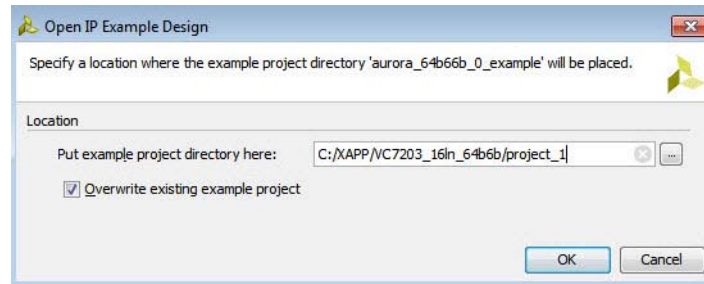


Figure 16: Path for Example Design

A new Vivado IDE will open with the core example design files.

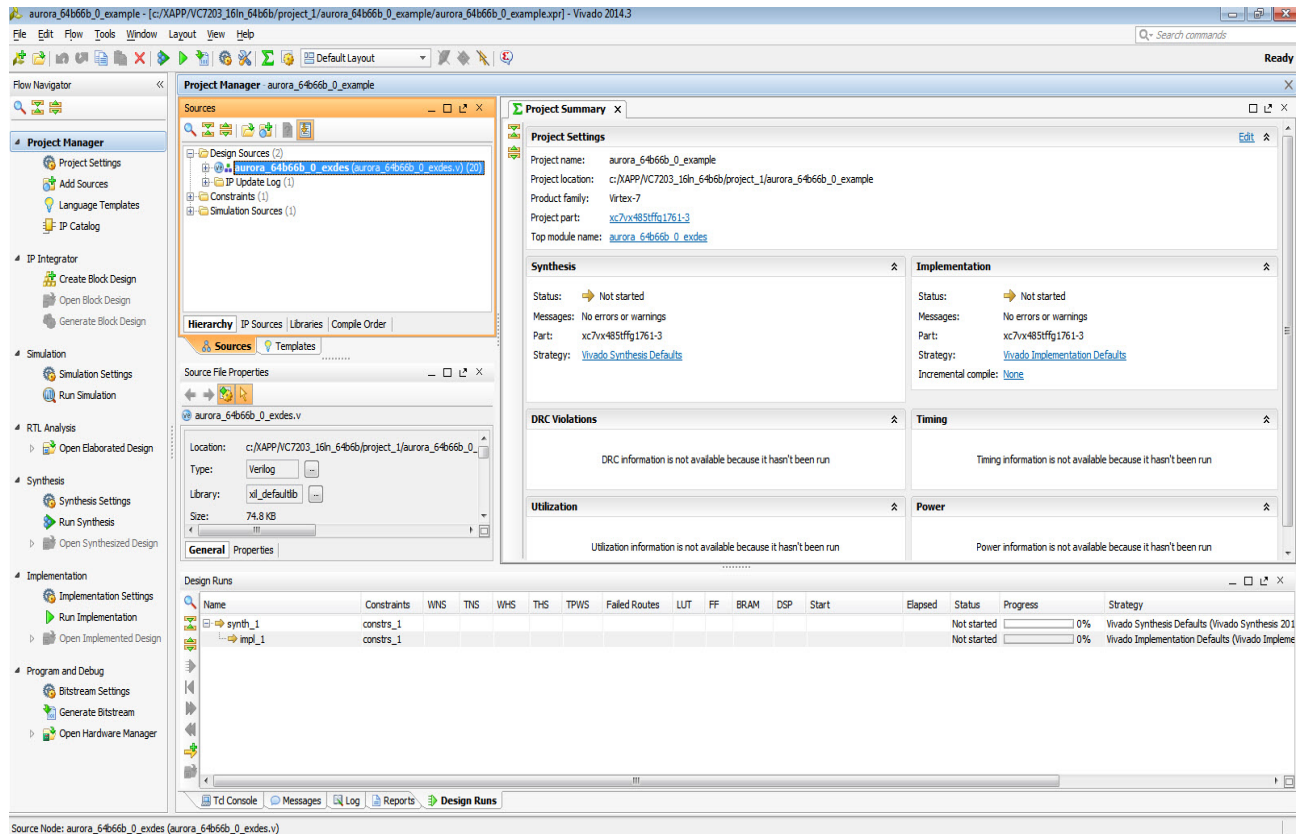


Figure 17: Generated Example Design

Updating the XDC Constraints for VC7203 and Bitstream Generation

1. In the newly-opened Vivado IDE window, expand the **Constraints** entry in the Sources panel of the Project Manager section.
2. Right-click the constraints file (`aurora_64b66b_0_exdes.xdc`) and select **Open file** (or alternatively double-click the file to open).

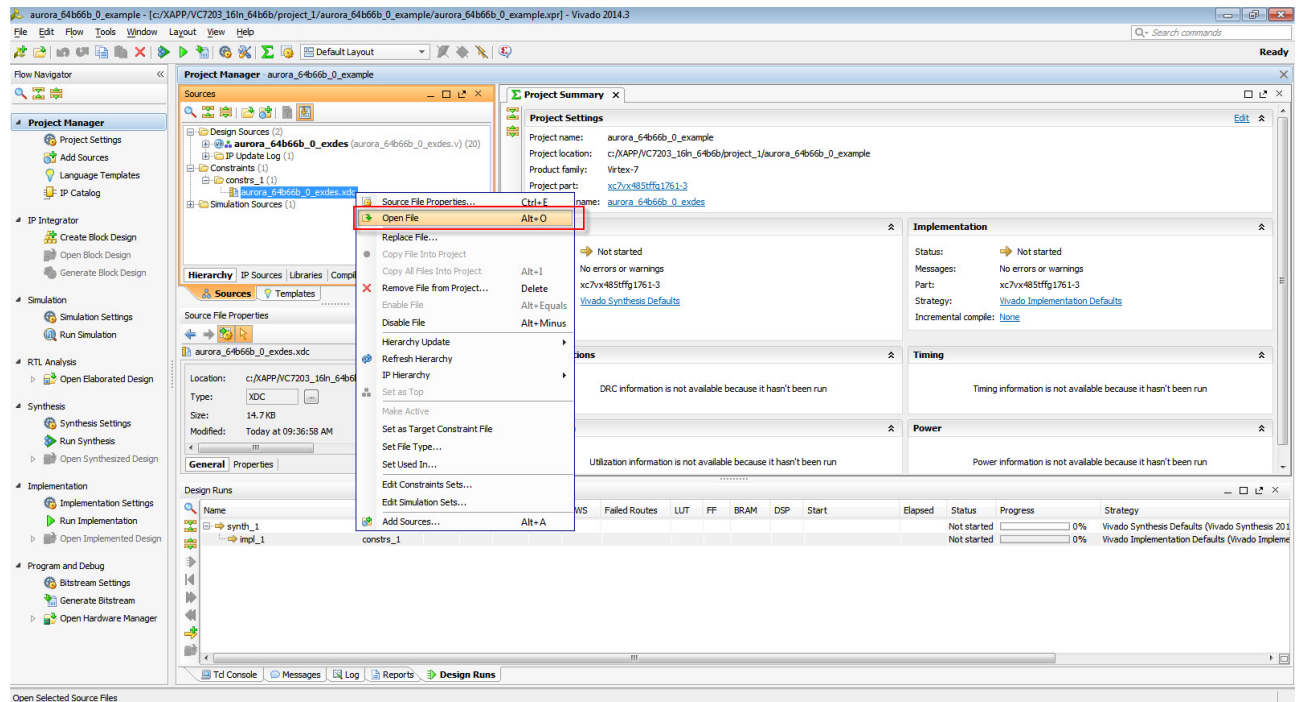


Figure 18: XDC Updates

3. Assign the pin locations for the Aurora 64B/66B core ports to those shown in [Table 1](#).

Table 1: Pin Locations

Pin Name	Location	Board LOC Identification
INIT_CLK_P	E19	On board 200MHz sys diff clk
INIT_CLK_N	E18	
RESET	P41	SW5
PMA_INIT	N41	SW4
GTXQ0_P	AH8	MGTREFCLK0P_Quad113
GTXQ0_N	AH7	MGTREFCLK0N_Quad113
GTXQ1_P	Y8	MGTREFCLK0P_Quad115
GTXQ1_N	Y7	MGTREFCLK0N_Quad115
DRP_CLK_IN	M32	FMC2_LA01_CC_P. Because the GTX transceiver has no DRP operations, this can be floating.
CHANNEL_UP	P40	APP_LED8
LANE_UP[0]	R40	APP_LED7 (DS14)
LANE_UP[1]	M39	APP_LED6 (DS13)
LANE_UP[2]	N38	APP_LED5
LANE_UP[3]	P42	APP_LED4
LANE_UP[4]	R42	APP_LED3
LANE_UP[5]	M38	APP_LED2
LANE_UP[6]	M37	APP_LED1
LANE_UP[7]	R29	
LANE_UP[8]	K35	
LANE_UP[9]	J35	
LANE_UP[10]	K32	
LANE_UP[11]	N30	
LANE_UP[12]	M31	
LANE_UP[13]	P30	
LANE_UP[14]	N31	
LANE_UP[15]	M28	
HARD_ERR	M29	
SOFT_ERR	R28	
DATA_ERR_COUNT[0]	P28	
DATA_ERR_COUNT[1]	H30	
DATA_ERR_COUNT[2]	L29	
DATA_ERR_COUNT[3]	L30	
DATA_ERR_COUNT[4]	J31	
DATA_ERR_COUNT[5]	H31	
DATA_ERR_COUNT[6]	L31	
DATA_ERR_COUNT[7]	N28	

The example design generated XDC file contains indicative PIN LOC and IO STANDARD constraints, Refer to the VC7203 schematic and update these accordingly to match the board requirements. Make sure that all the output pins of the Aurora 64B/66B example design are assigned with LOC and IOSTANDARD.

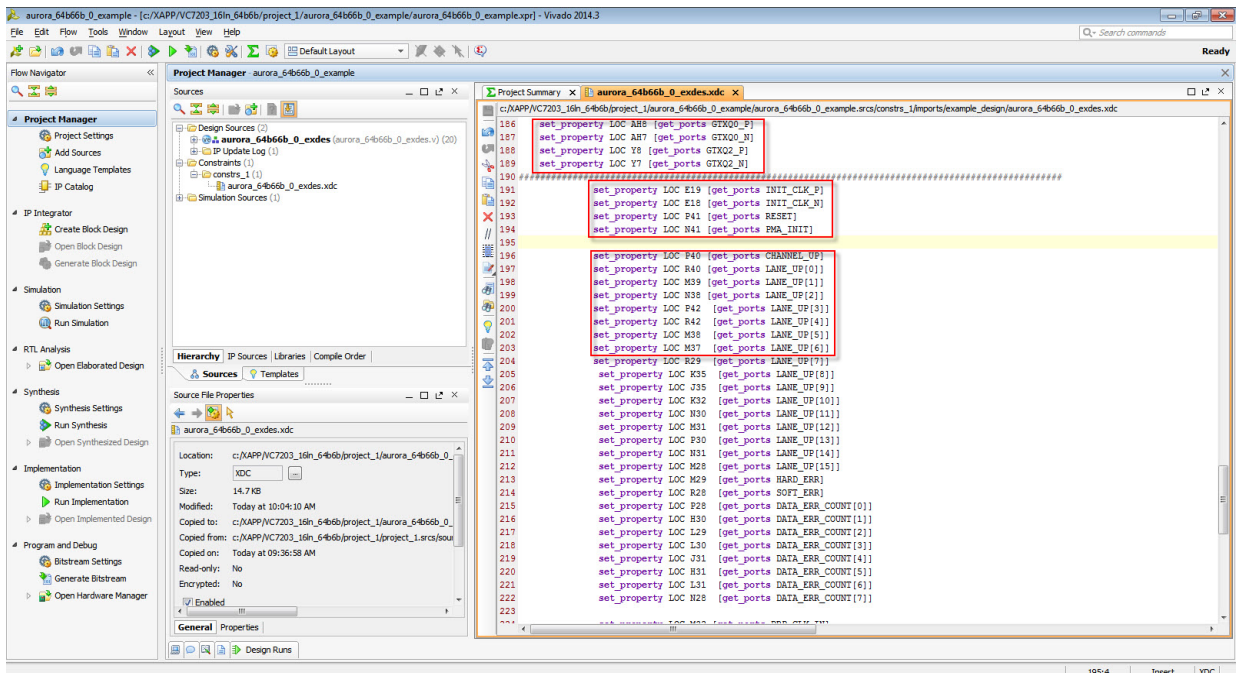


Figure 19: Final XDC

4. Right-click within the constraints file editor and select **Save File**. Close the constraints file editor window.
5. Save the files and run **Generate Bitstream**. A pop up dialog box displays.
6. Click **Yes** to launch synthesis and implementation and proceed with bit file generation.

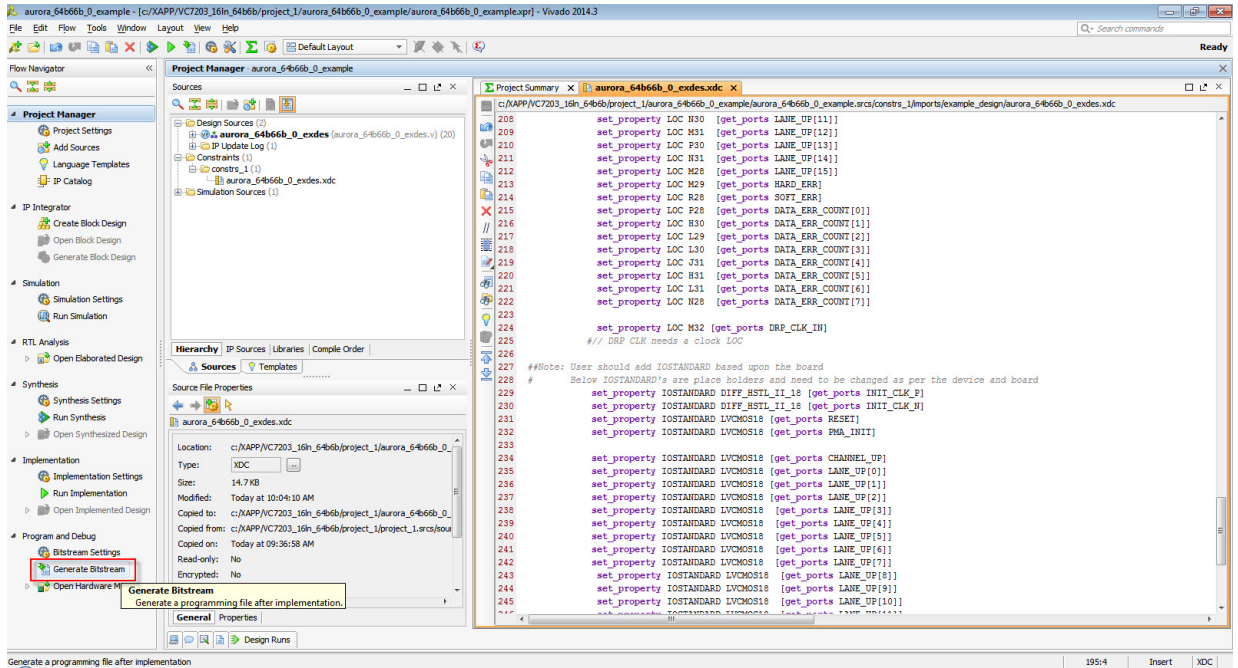


Figure 20: Generate Bitstream

7. Wait until the write bitstream process is completed.

Setting up the Hardware Session

1. Click **Flow > Open Hardware Session** or click **Open Hardware Manager** under **Program and Debug** in the Project Manager panel as shown in [Figure 21](#).

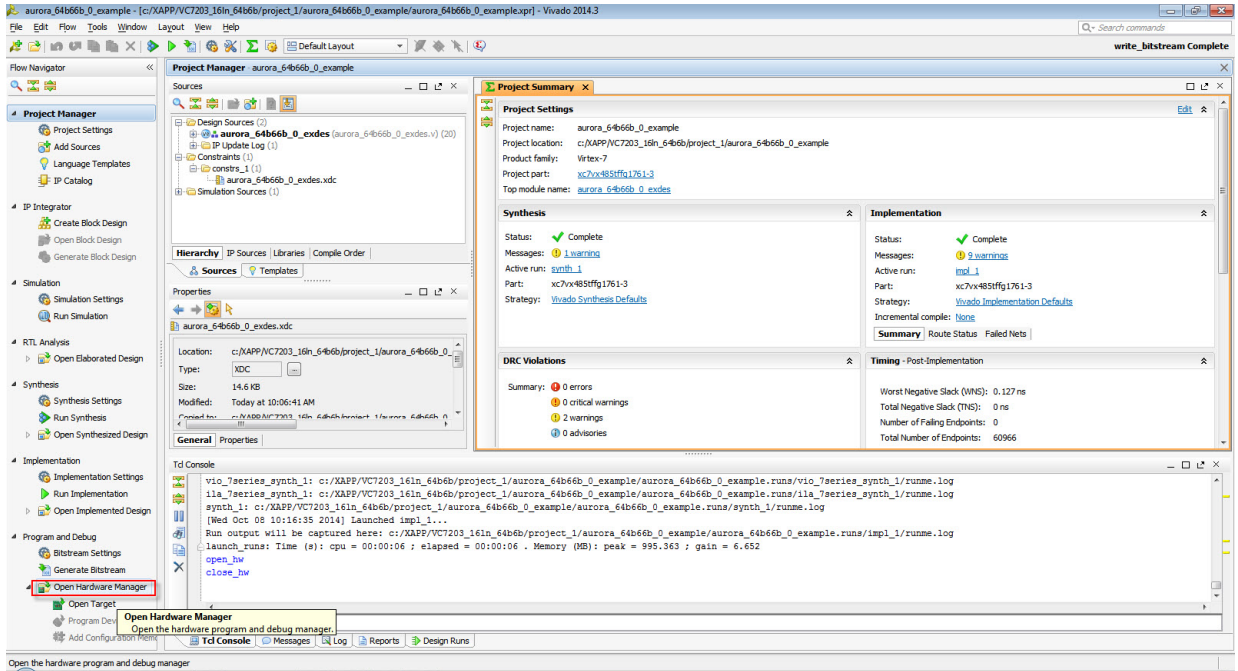


Figure 21: Open Hardware Session

2. Click **Open a new hardware target**.

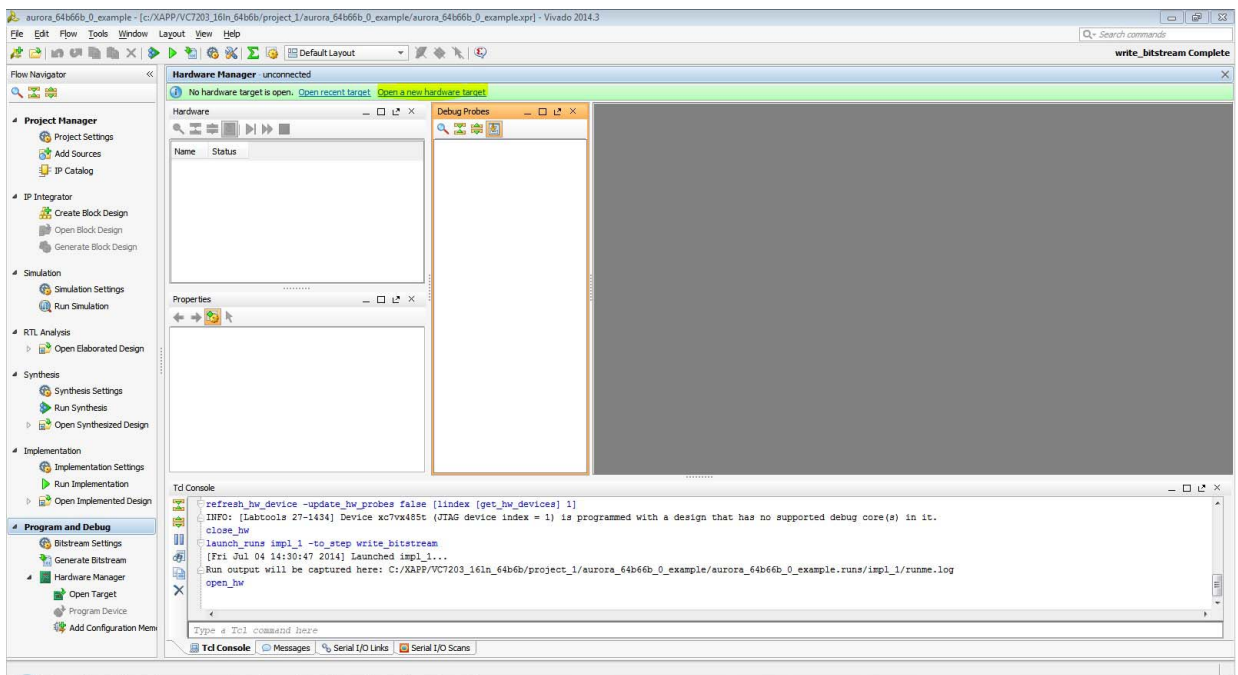


Figure 22: Hardware Session

3. When the Open New Hardware Target window opens, click **Next**.
4. Depending on the board connection, perform one of the following.
 - If the board is connected to a remote system, you need to run **hw_server** on that particular system. In the Hardware Server Settings window, select Remote server and enter the server name., after selecting the Remote Server enter the system name.

Or

- If the board is connected to the same system, you can select the **Local Server** option and click **Next**. Selecting this option automatically connects to the server and detects all boards connected to the Server.

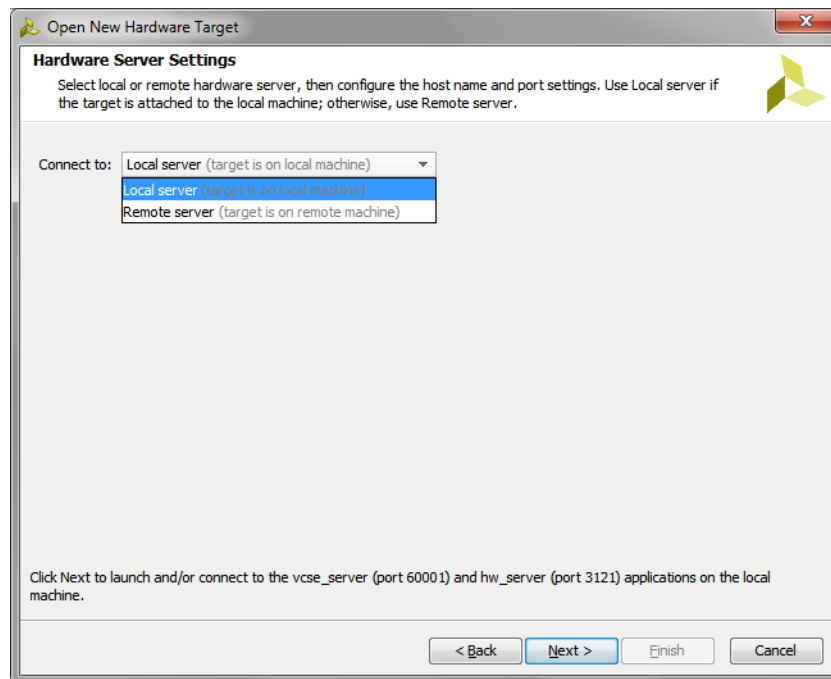


Figure 23: Hardware Server Settings

The Open Hardware Target window will open with all the targeted boards connected to the server as show in [Figure 24](#).

5. Select the board that you want to program and click **Next** to set the targeted hardware JTAG properties.

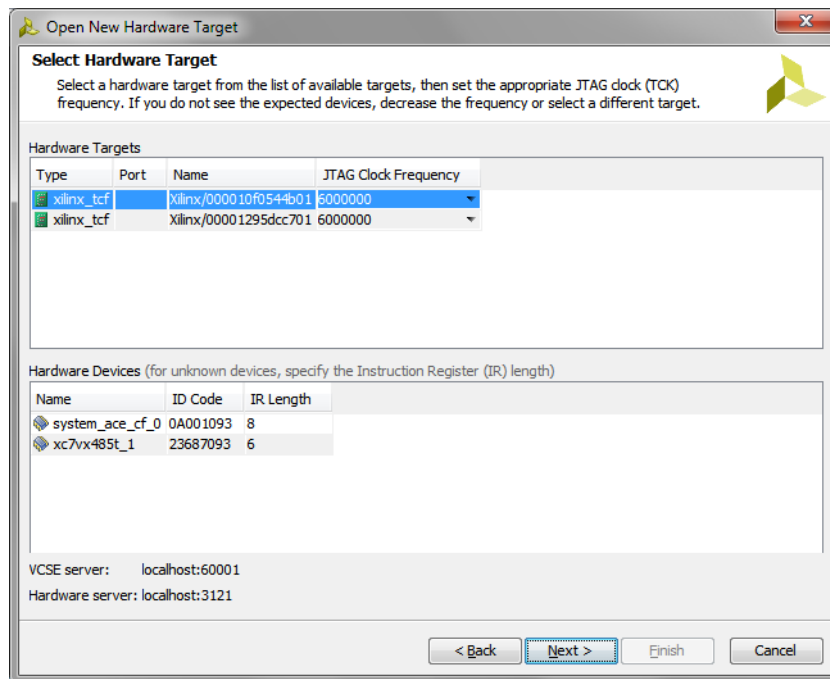


Figure 24: Select Hardware Target

- Review the targeted hardware summary and Click **Finish**. You are returned to the Hardware Manager screen.

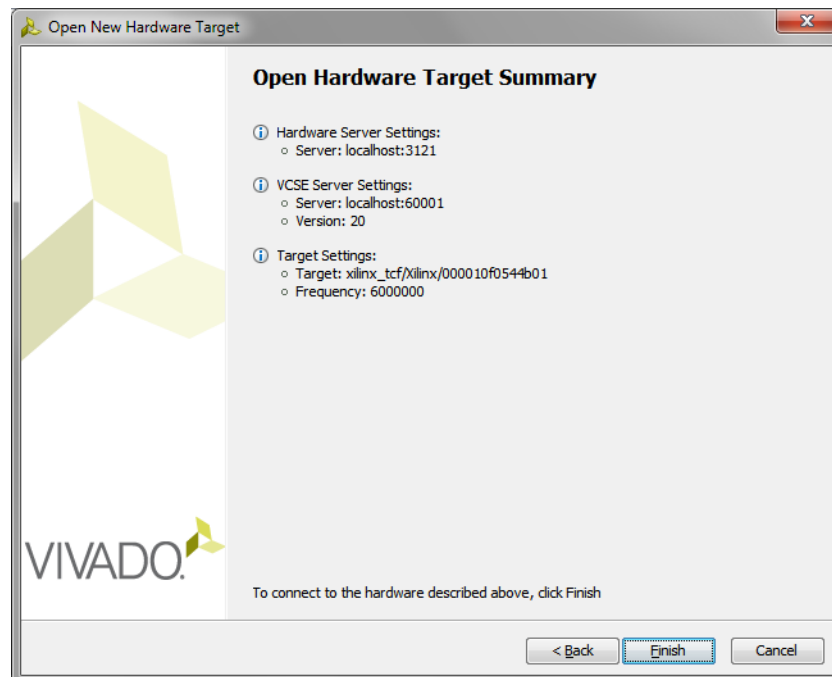


Figure 25: Hardware Target Summary

- Right-click the device, and click the **Program Device**.

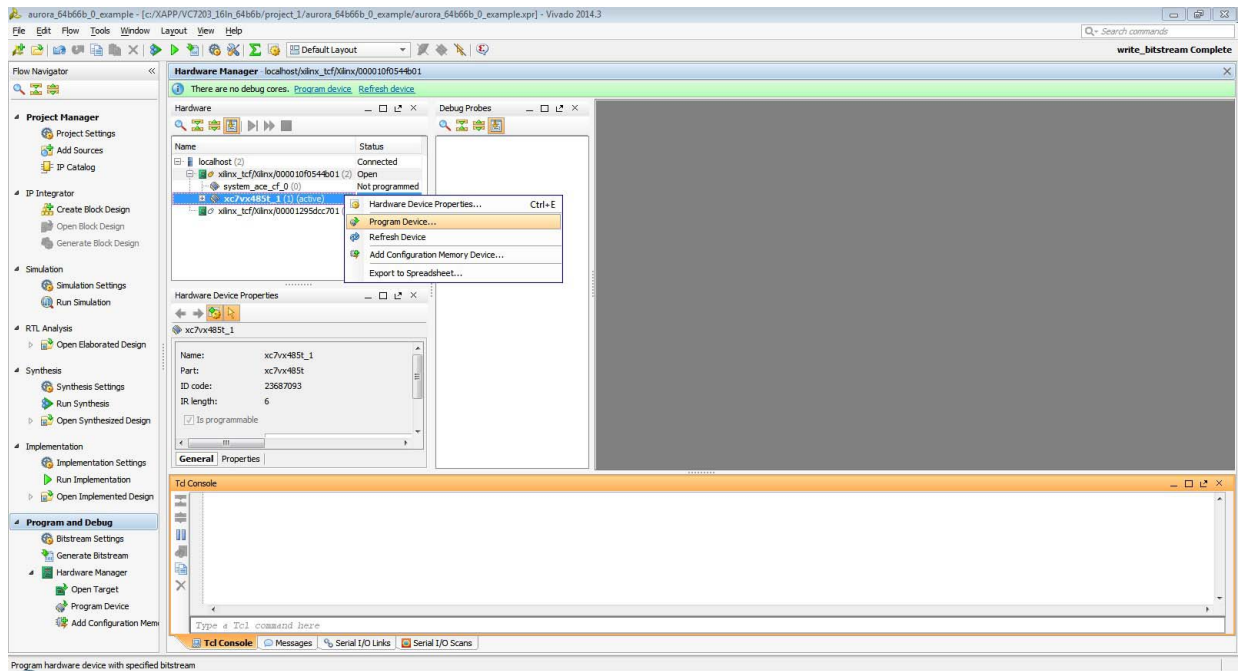


Figure 26: Hardware Manager, Program Device

- Browse for the `superclk2_vc7203.bit` and `superclk2_vc7203_debug_nets.ltx` files that are attached with the application note and download the bitstream. Then run the `setup_scm2_156.25.tcl` script which programs the SuperClock-2 module Si5324 and Si5700 clocks to 156.25 MHz. Additional guidance can be found in the *VC7203 IBERT Getting Started Guide* (UG847) [Ref 3].

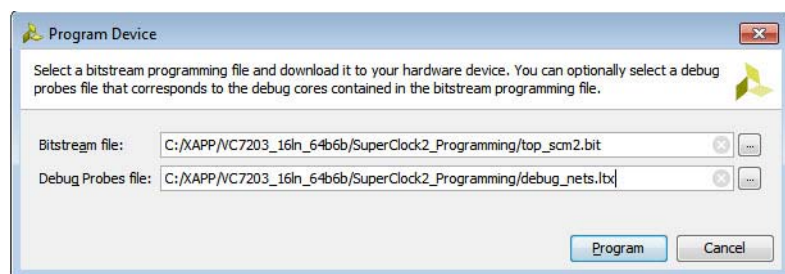


Figure 27: SuperClock-2 Module Programming

After the SuperClock-2 module is programmed, you can observe the DS1 status LED on the SuperClock-2 module turning off from indicating the proper configuration.

9. Close the hardware target and open the other target as shown in [Figure 29](#) and [Figure 30](#).

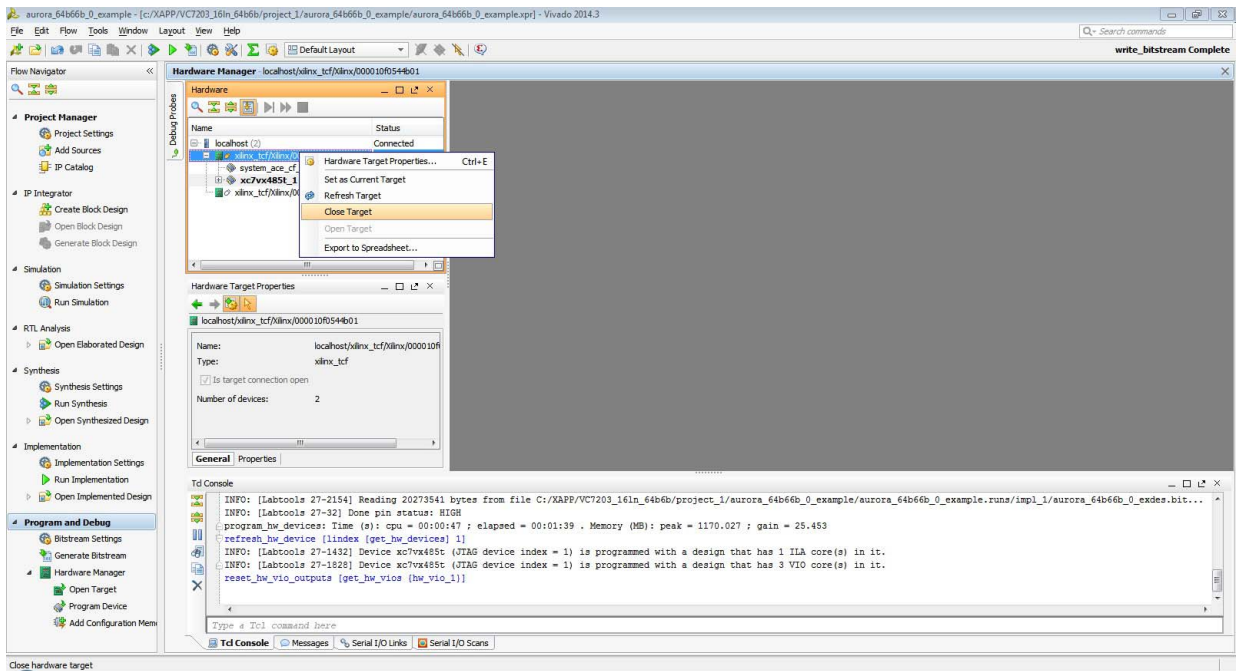


Figure 28: Close Target on Board A

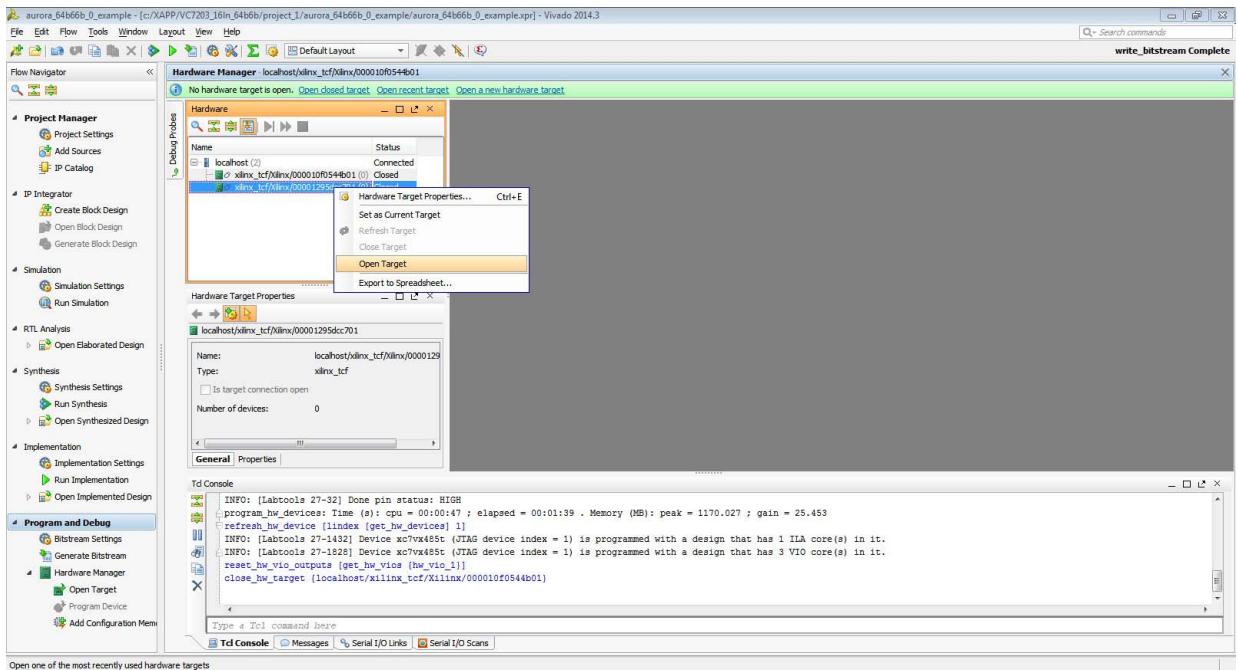


Figure 29: Open target on Board B

10. Repeat [step 8](#) and [step 9](#) to program the second board with the SuperClock-2 module bit so that the stable 156.25 GT MHz `refclk` can be sourced by the Aurora 64B/66B example designs.

After successful programming of the SuperClock-2 module bit files on both the boards, observe that the DS1 status LED on the SuperClock-2 module will turn off indicating proper SuperClock-2 module programming.

Programming and Testing the Aurora64B/66B Example Project

1. Right-click the device and select **Program Device** and then choose the Bitstream file and Debug Probes file from the implemented project as shown in Figure 31 and click **Program**.

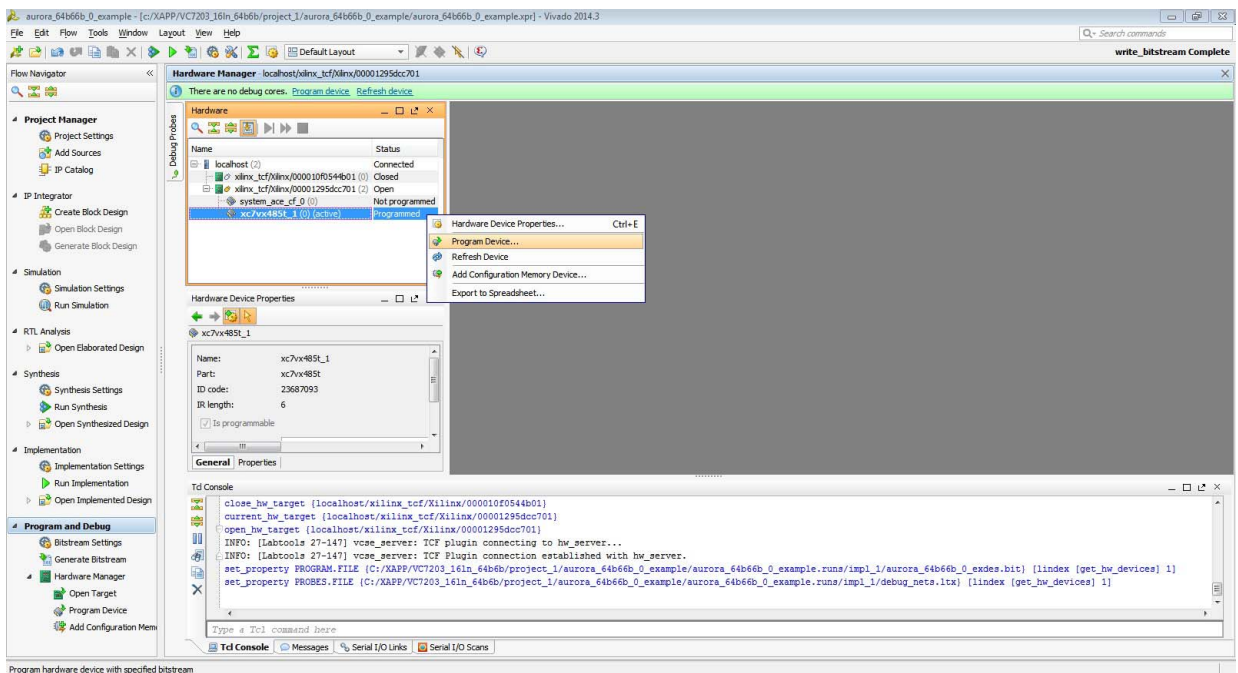


Figure 30: Programming the Aurora 64B/66B core on Board B

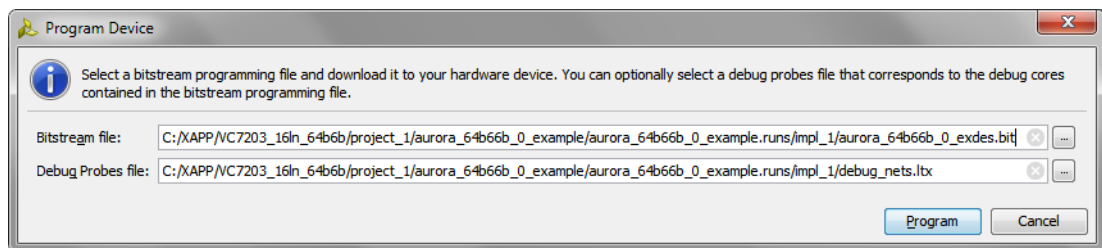


Figure 31: Programming the Aurora 64B/66B core on Board B

2. Right-click the **hw_vio_1** instance or the other VIO instance in which signals, **channel_up_in_initclk** and **lane_up_vio_i** are present. This might be either hw_vio_1/hw_vio_2/hw_vio_3 or other depending on the implementation of the debug hub core and then select **Add Probes to VIO Window**.

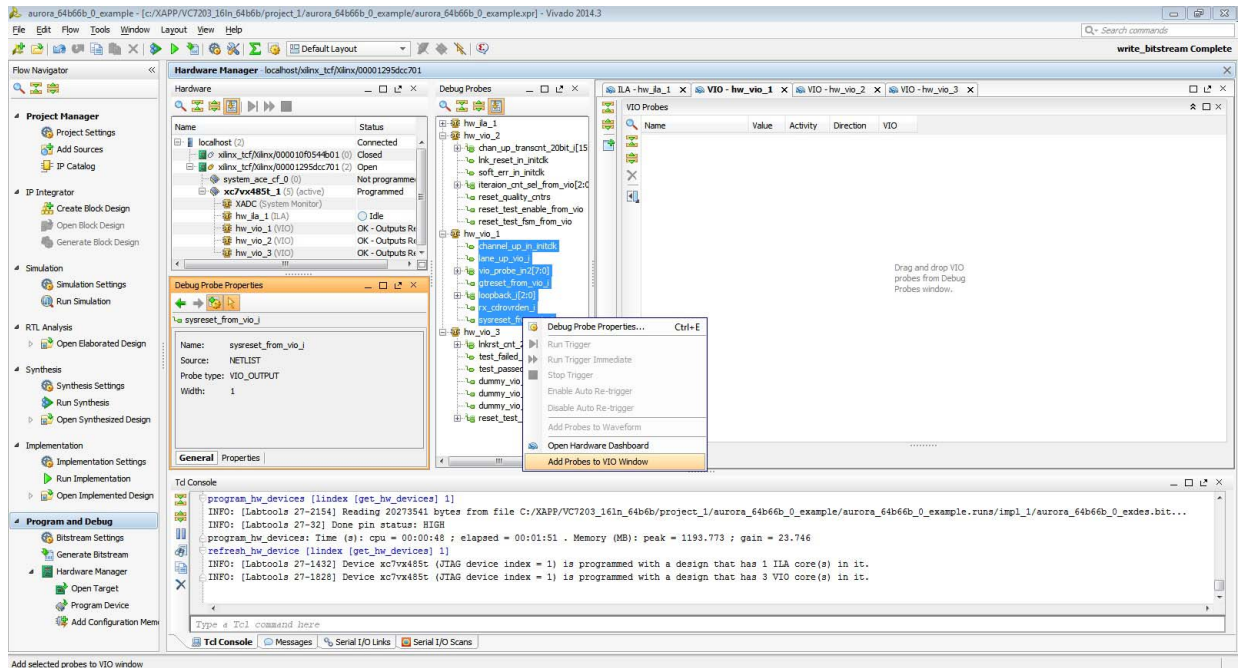


Figure 32: Channel-up/Lane-up VIO Monitors

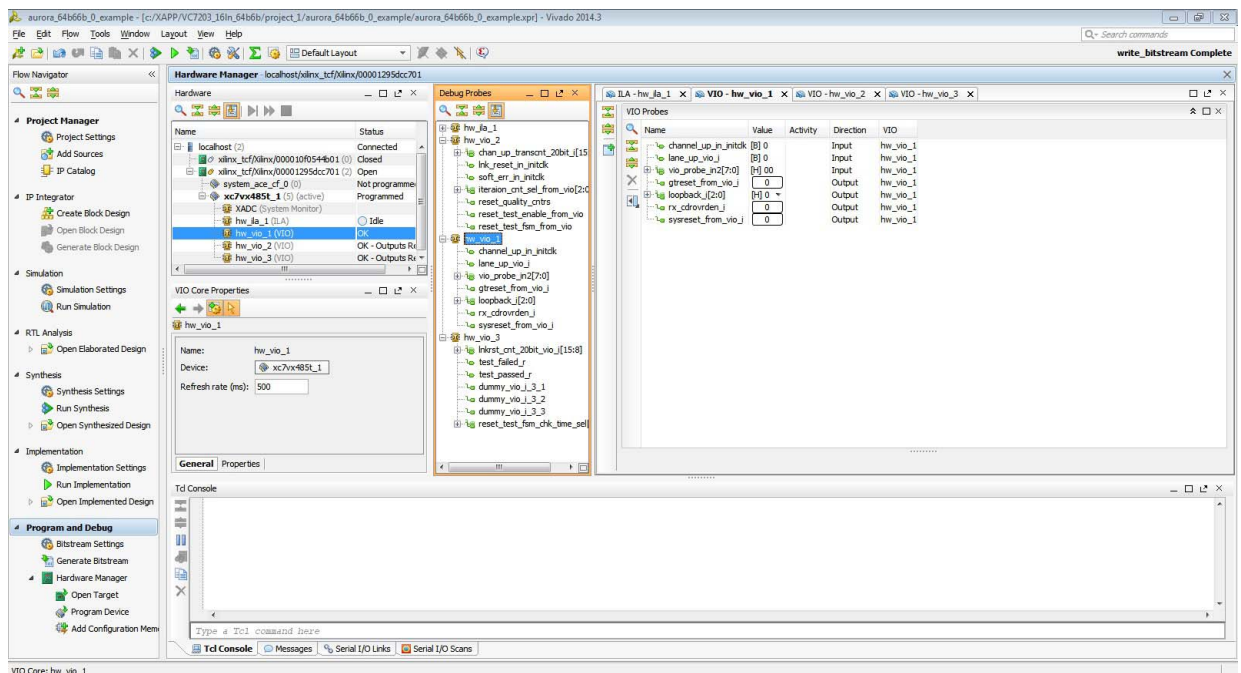


Figure 33: Channel-up/Lane-up VIO Monitors

- Using the VIOs, follow the recommended reset sequence, assert `sysreset_from_vio_i`, assert `gtreset_from_vio_i`. Then deassert `gtreset_from_vio_i` and then `sysreset_from_vio_i`.

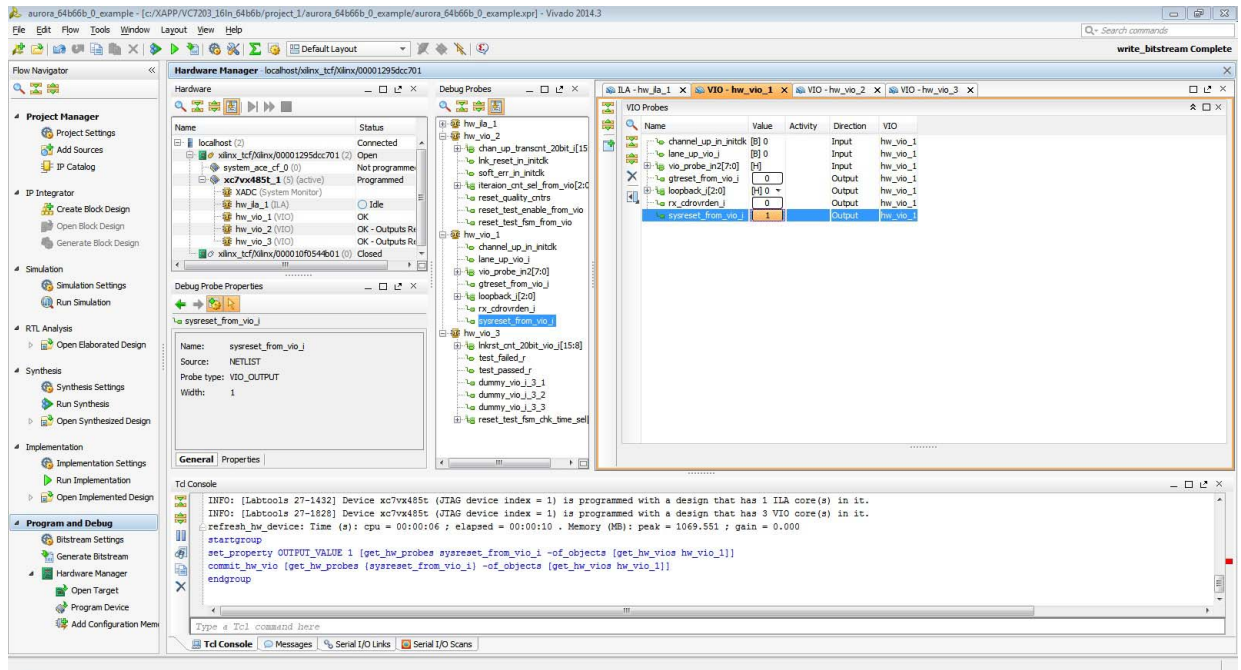


Figure 34: Assert Reset Sequence

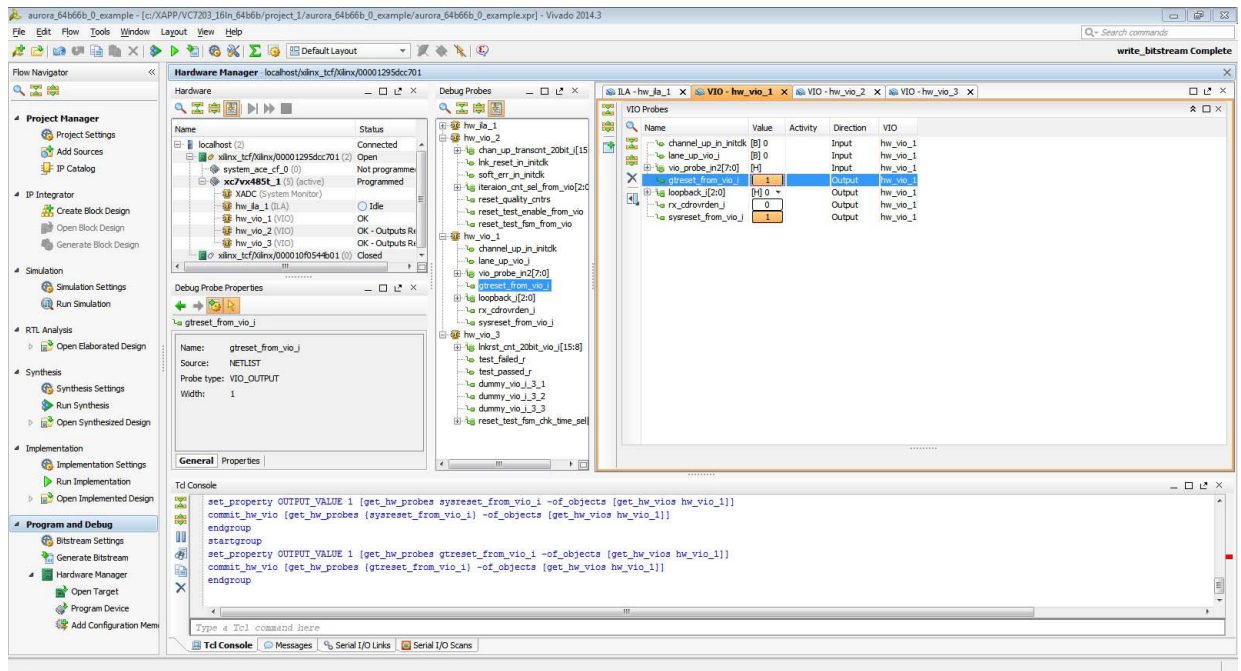


Figure 35: Assert Reset Sequence

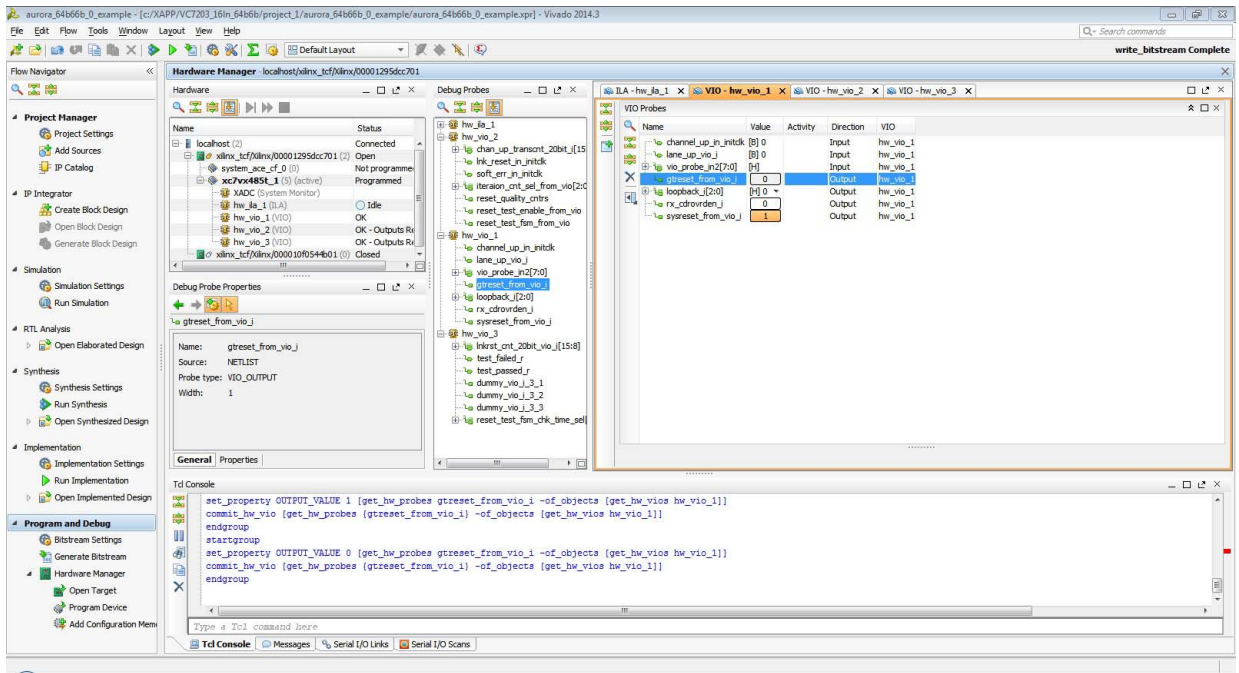


Figure 36: Deassert Reset Sequence

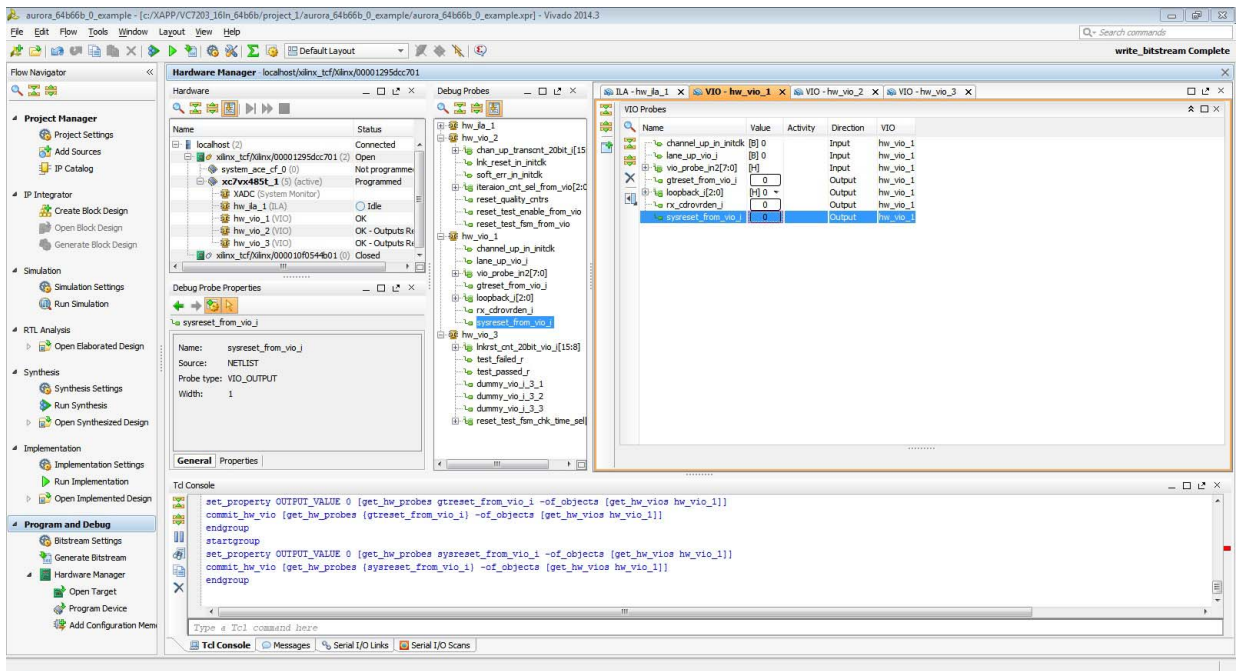


Figure 37: Deassert Reset Sequence

4. Program the second board with the Aurora 64B/66B example design project bitfile. Close the hardware target. Right-click the device and click **Close Target** as shown in Figure 38.

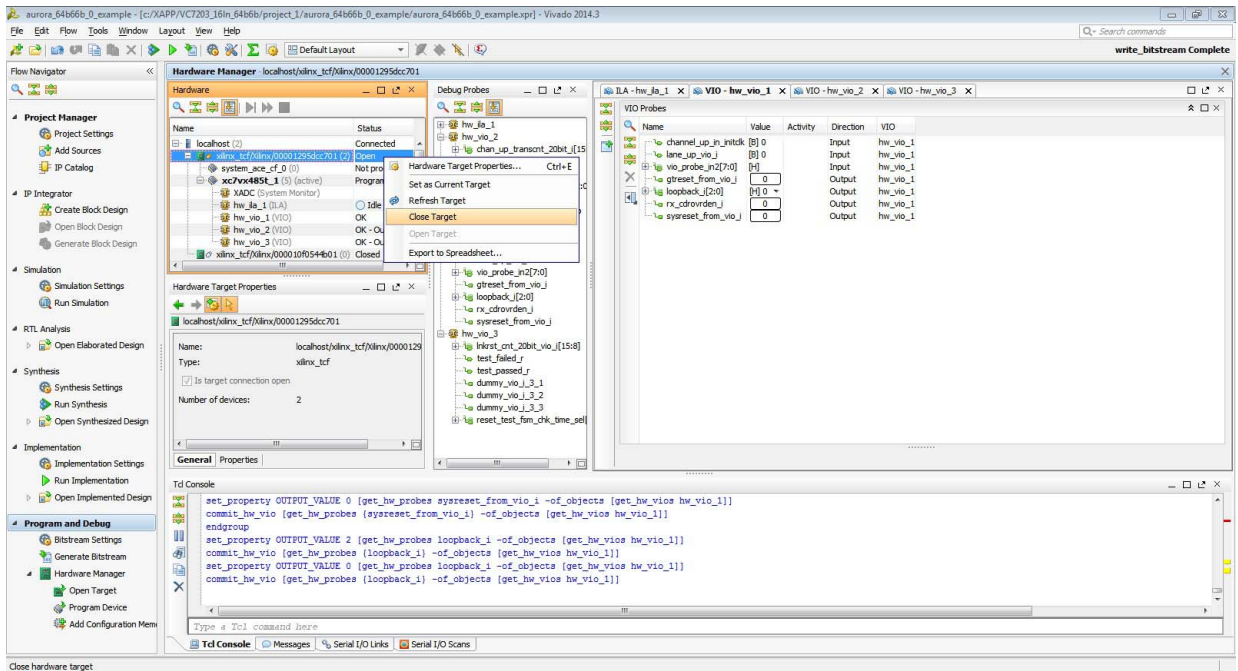


Figure 38: Close Target

5. Right-click the other hardware target and click **Open Target**.

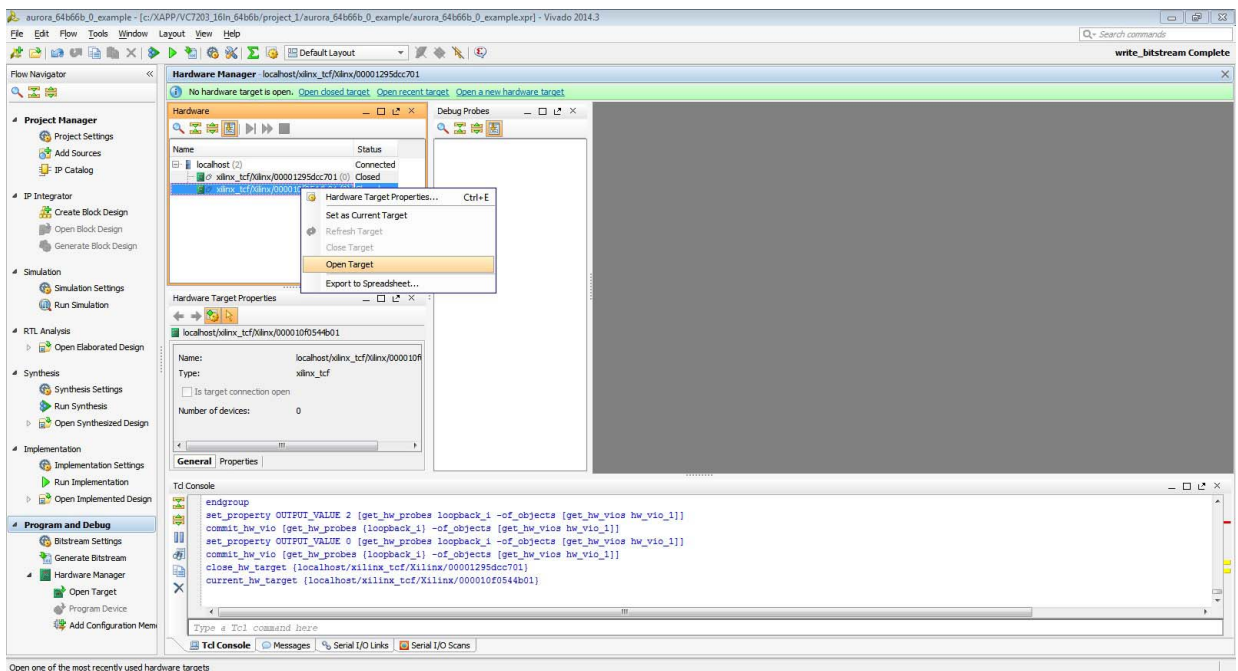


Figure 39: Open the Other Target

6. Repeat [step 2](#), [step 3](#), and [step 4](#) to program the second hardware target.

Observe that `channel_up_in_initclk` and `lane_up_vio_i` go High.

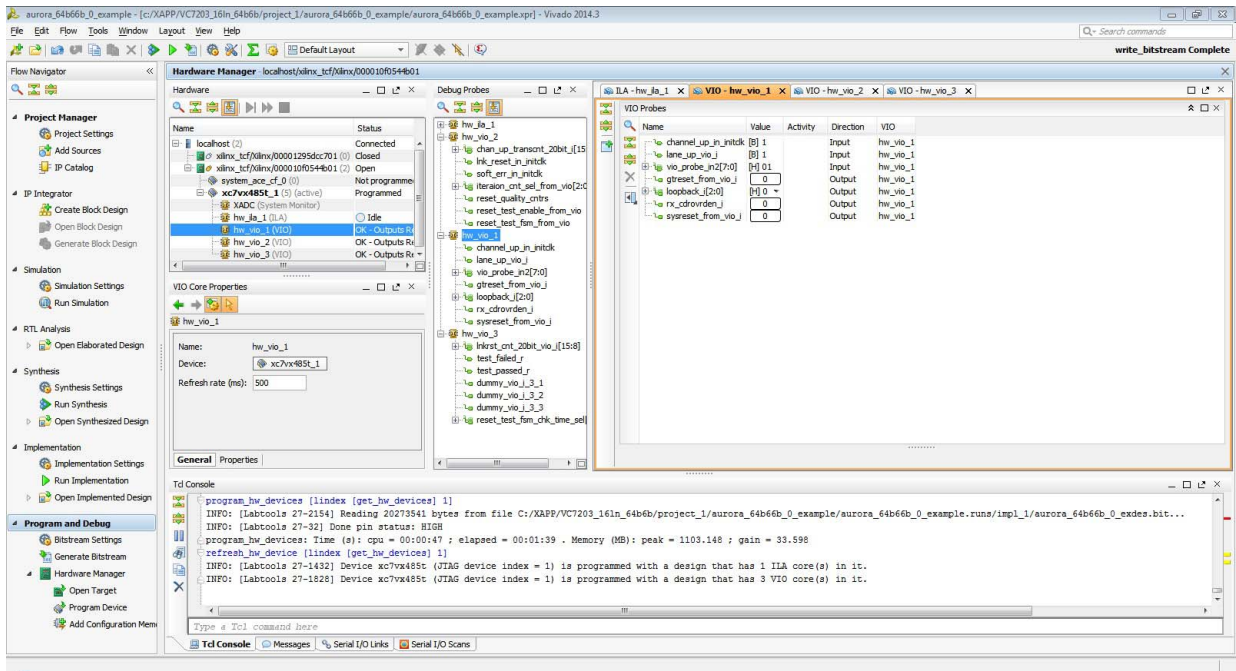


Figure 40: Channel-up/Lane-up VIO Monitoring

For the testing the link, you can try asserting and deasserting `sysreset_from_vio_i` or by following the recommended reset sequence as described in [step 3](#). Also, you can use the push-buttons, SW4(PMA_INIT), SW5(RESET). Apart from the VIO status, channel-up/lane-up LEDs going High can also be observed.

a. Testing by asserting only `sysreset_from_vio_i`

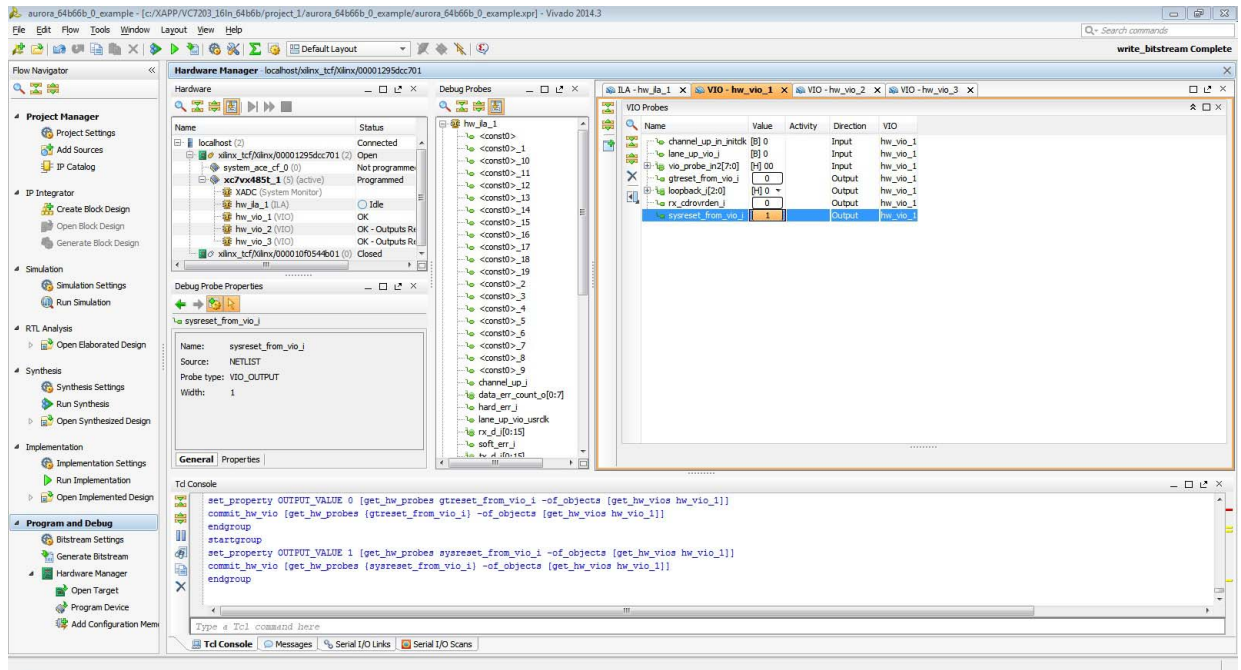


Figure 41: Assert Reset Sequence

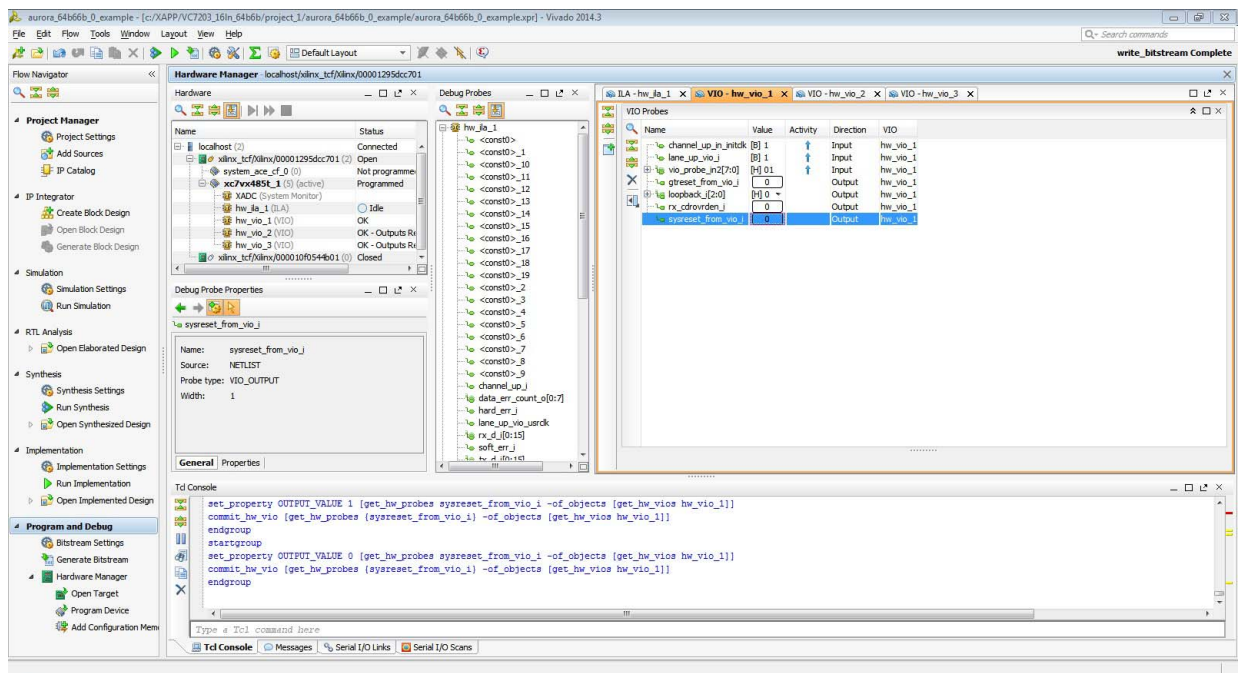


Figure 42: Deassert Reset Sequence and Observe Channel-up/Lane-up

b. Testing by following the reset sequence as discussed in [step 4](#).

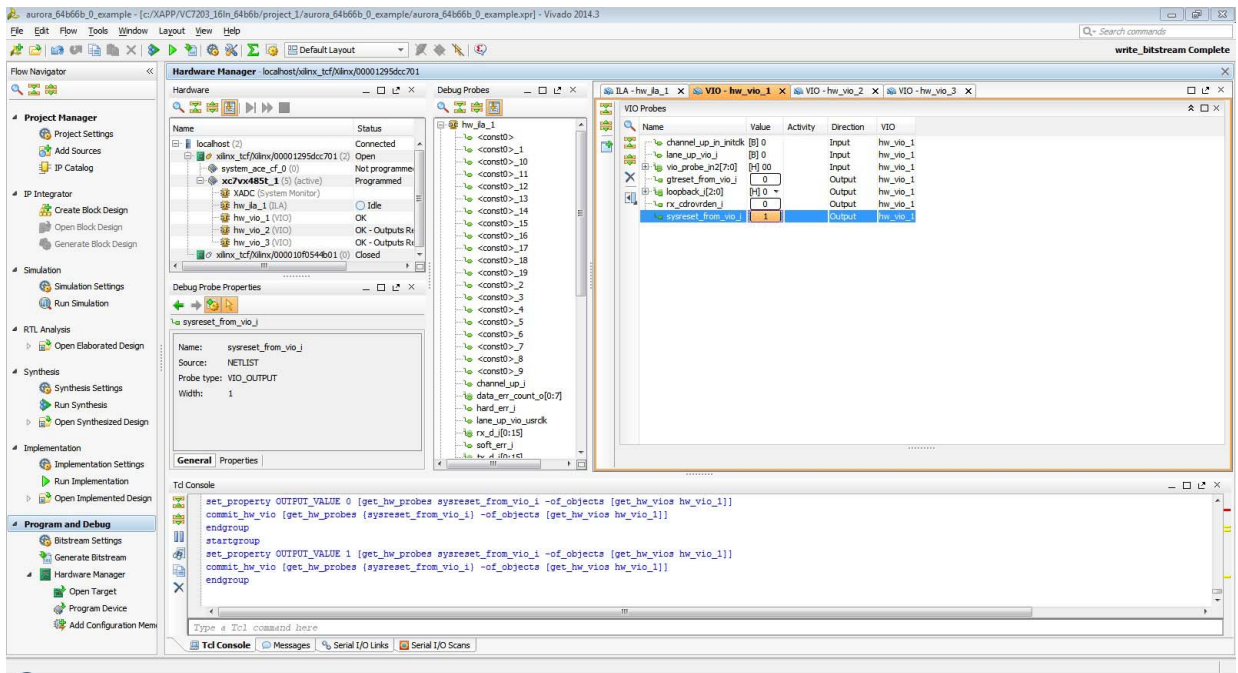


Figure 43: Channel-up/Lane-up VIO Monitoring

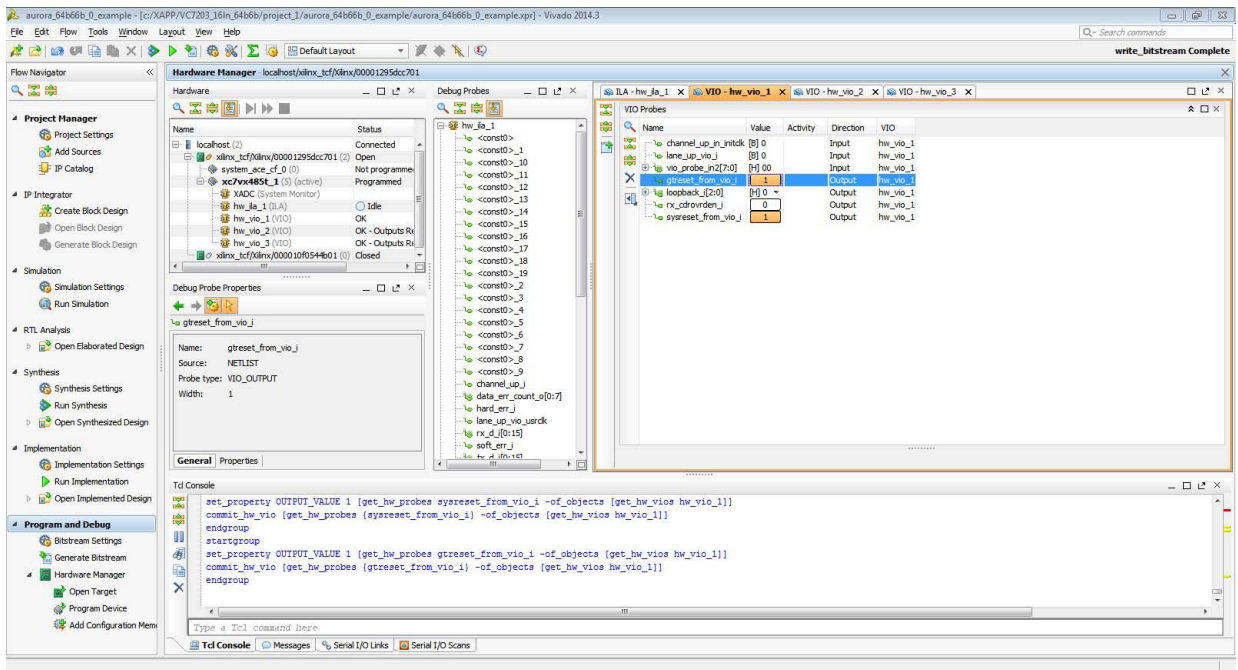


Figure 44: Channel-up/Lane-up VIO Monitoring

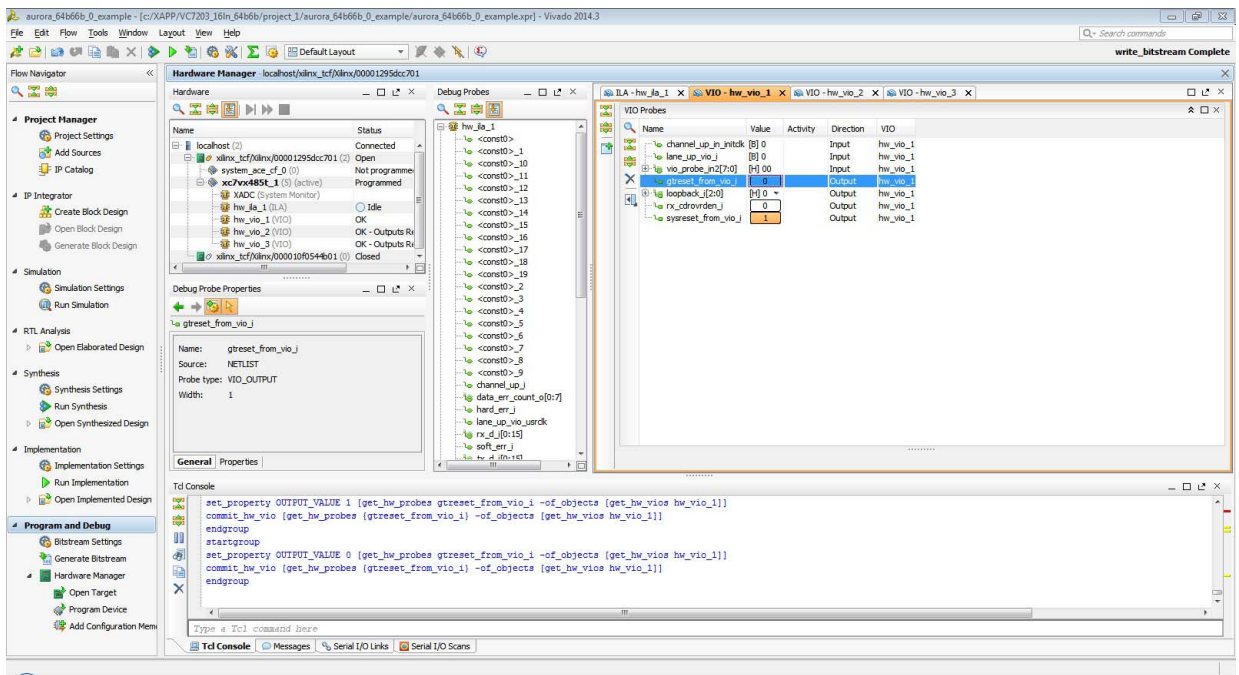


Figure 45: Channel-up/Lane-up VIO Monitoring

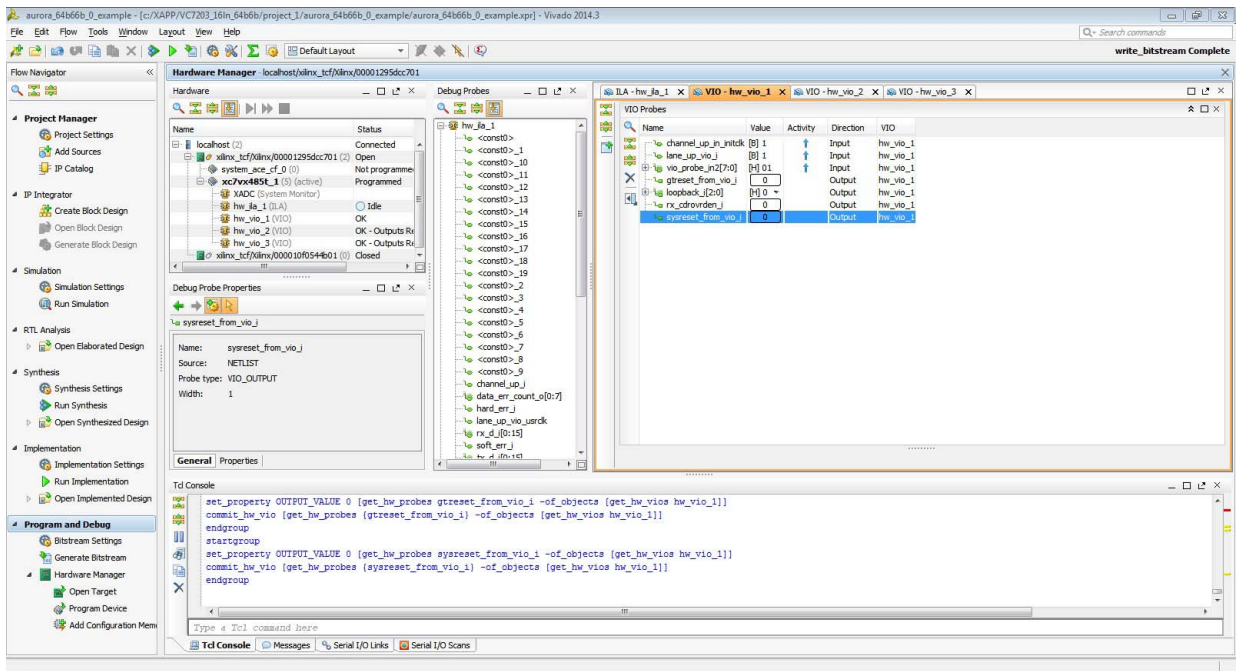


Figure 46: Channel-up/Lane-up VIO Monitoring

Reference Design

The reference design files for this application note are generated when the Aurora 64B/66B core is customized from the Vivado IP catalog. The reference design has been fully verified and tested on hardware. The design includes details on the various functions of the different modules. The SuperClock-2 module bit files are provided for configuring the SuperClock-2 module add-on board to 156.25 MHz.

You can download the [Reference Design Files](#) for this application note from the Xilinx website.

The reference design checklist is shown in [Table 2](#).

Table 2: Reference Design Checklist

Parameter	Description
General	
Target devices (stepping level, ES, production, speed grades)	Virtex7 (VC7203) FPGA (XC7VX485T-3FFG1761E)
Source code provided	No (Core generated from 2014.3 IP catalog)
Source code format	Verilog
Design uses code/IP from existing Xilinx application note/reference designs, CORE Generator™ software, or 3rd-party	Reference design provided by the Aurora 64B/66B core generated from the 2014.3 Vivado IP catalog
Simulation	
Functional simulation performed	No (Example design provides the default test bench)
Timing simulation performed	No
Test bench used for functional and timing simulations	N/A
Test bench format	N/A
Simulator software/version used	N/A
Implementation	
Synthesis software tools/version used	Vivado Design Suite 2014.3
Implementation software tools/versions used	Vivado Design Suite 2014.3
Static timing analysis performed	Yes (passing timing with the Vivado Design Suite Implementation Tool)
Hardware Verification	
Hardware verified	Yes
Hardware platform used for verification	VC7203 Characterization board

Conclusion

The Virtex-7 FPGA VC7203 characterization kit provides an excellent platform to implement and test the LogiCORE IP Aurora 64B/66B core for various supported configurations, the `GT_refclk` can be sourced from the provided SuperClock-2 module. Various configurations can be quickly evaluated using only the VC7203 characterization kit and the Vivado Design Suite. Detailed steps are provided to demonstrate that the 16-lane Aurora 64B/66B will work at 10.3125 Gb/s. This provides a high-speed AXI user link with 1024 bit data width.

References

Refer to these documents for additional details:

1. *7 Series FPGAs GTX/GTH Transceivers User Guide* ([UG476](#))
2. *LogiCORE IP Aurora 64B/66B Product Guide* ([PG074](#))
3. *VC7203 IBERT Getting Started Guide* ([UG847](#))
4. *VC7203 Virtex-7 FPGA GTX Transceiver Characterization Board User Guide* ([UG957](#))
5. *VC7203 Schematics* (Rev 1.0)
6. *Vivado Design Suite User Guide: Designing with IP* ([UG896](#))
7. *Vivado Design Suite User Guide: Programming and Debugging* ([UG908](#))
8. *Virtex-7 FPGAs Data Sheet: DC and Switching Characteristics* ([DS183](#))

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
02/28/2017	1.0.2	Fixed broken link on page 34 to the Reference Design Files.
10/08/2015	1.0.1	Updated link in step 6 on page 30 to be step 3 instead of step 4. Updated link in step 3 on page 16 to be Table 1 instead of Figure 1.
11/04/2014	1.0	Initial Xilinx release

Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at <http://www.xilinx.com/legal.htm#tos>.

Automotive Applications Disclaimer

XILINX PRODUCTS ARE NOT DESIGNED OR INTENDED TO BE FAIL-SAFE, OR FOR USE IN ANY APPLICATION REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS APPLICATIONS RELATED TO: (I) THE DEPLOYMENT OF AIRBAGS, (II) CONTROL OF A VEHICLE, UNLESS THERE IS A FAIL-SAFE OR REDUNDANCY FEATURE (WHICH DOES NOT INCLUDE USE OF SOFTWARE IN THE XILINX DEVICE TO IMPLEMENT THE REDUNDANCY) AND A WARNING SIGNAL UPON FAILURE TO THE OPERATOR, OR (III) USES THAT COULD LEAD TO DEATH OR PERSONAL INJURY. CUSTOMER ASSUMES THE SOLE RISK AND LIABILITY OF ANY USE OF XILINX PRODUCTS IN SUCH APPLICATIONS.