

## Overview

The IBERT Reference Design for Transceiver Based Kits from Avnet Electronics Marketing provides engineers with everything needed to develop products with high speed transceivers. This document describes how to use Avnet Electronic Marketing's new flow utilizing GIT for an expedited



Figure 1 – PicoZed Logo

## Objectives

This tutorial is a guide for how to:

- Retrieve the design files from the public Avnet git repository
- Build the reference design
- Execute the reference design on hardware

## Reference Design Overview

Utilizing the built in debug cores, this document will help provide a method for producing an IBERT design as quickly as possible in order to allow an analysis of the transceivers for your hardware helping to expedite your design.

## Experiment Setup

This tutorial makes use of Xilinx Vivado Design Suite in graphical (GUI) mode in order to create a project, then if request produce eye diagrams for further analysis and modification. If you have not already configured your transceiver clock, you will need to visit PicoZed.org. There, under Reference Designs / Tutorials, you will find “Transceiver Clock Programming Reference Design” which will help you through setting up a reference clock for use with this project. The default configuration used in that document will provide you the necessary 250MHz on the MGT1 transceiver clock input.

## Software

The software required to build, and execute the reference design is:

- Windows-7 64-bit
- Xilinx Vivado Design Suite 2015.4

## Optional Software

The software required to customize the clock configuration is:

- [Timing Commander](#)

## Hardware

The hardware required to build, and execute the reference design is:

- PC with minimum amount of additional RAM available for Xilinx tools as specified at [www.xilinx.com/design-tools/vivado/memory.htm](http://www.xilinx.com/design-tools/vivado/memory.htm) for either the XC7Z015 or XC7Z030 device
  - 4GB required but 8GB recommended
- Avnet PicoZed 7015 or 7030 SOM
- Avnet PicoZed FMC2 Carrier Card (AES-PZCC-FMC-V2-G)
- JTAG Programming Cable (Xilinx Platform Cable, Digilent HS1, HS2, or HS3 cables)
  - If you don't already have a JTAG Cable, Avnet recommends the Digilent HS3 Cable
  - <http://www.em.avnet.com/itaghs3>
- SMA cables (2)

## Optional Hardware

The optional hardware setup for this reference design is:

- SFP+ module supporting up to 6.6Gbps data rate and optical cables or SFP+ loopback adapter
  - Avago AFBR-709SMZ
  - Finisar FTLF8528P3BCV

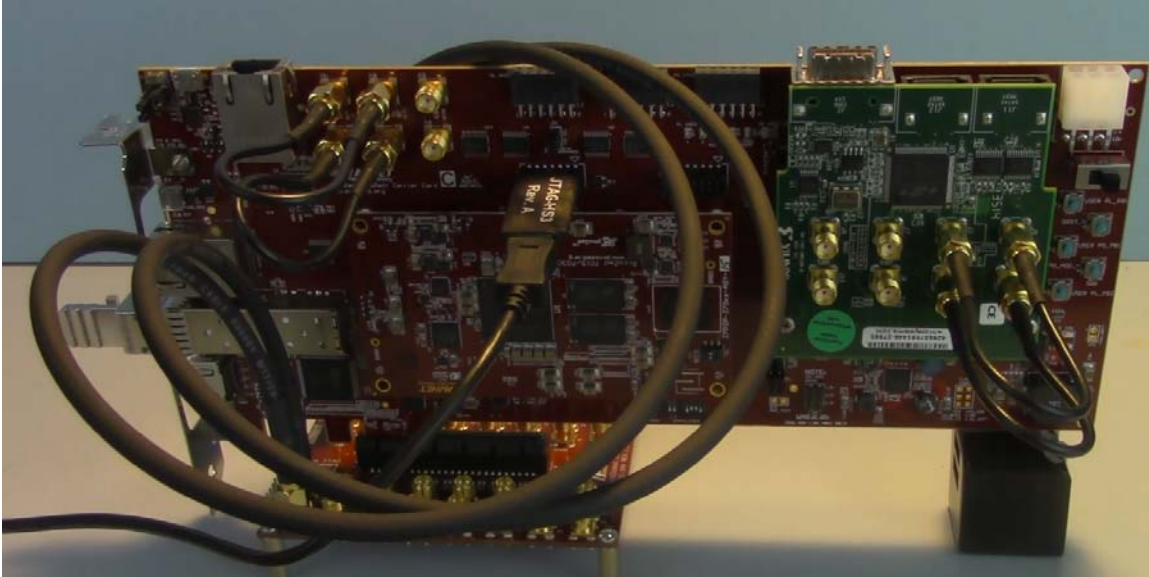
- HighTechGlobal PCIe loopback adapter
- XM104 FMC Loopback Adapter
  - SMA cables (2) for use with the XM104

## Setting up the Board

1. Enable 250MHz clock
  - a. Refer to the PicoZed FMC Carrier Card V2 Reference Design and Tutorial page and follow “Transceiver Clock Programming Reference Design”
  - b. <http://picozed.org/support/design/13076/106>
2. Remove all connectors on JP5, or configure for 1.8V – especially if using a PicoZed 7030 SOM (1.8V ONLY supported!!).
3. If using SFP+, install a jumper on the AES-PZCC-FMC-V2-G JP2.
4. Plug the Zynq PicoZed SOM onto the carrier card via JX1/JX2/JX3 connectors.
5. Connect 12V power supply to J2 – NOTE this is NOT ATX compatible!.
6. Plug an SFP+ module into the SFP+ socket (P1) on the carrier card
  - a. If needed, use an optical loopback cable to connect the SFP TX port to the SFP RX port or insert an SFP loopback adapter into the SFP+ socket.
7. Connect J3 SMA connector to J4 SMA connector using an SMA cable.
8. Connect J5 SMA connector to J6 SMA connector using an SMA cable.
9. Plug XM104 into the FMC LPC connector CONN1
  - a. Connect XM104-J3 SMA connector to XM104-J5 SMA connector using an SMA cable.
  - b. Connect XM104-J4 SMA connector to XM104-J6 SMA connector using an SMA cable.
10. Connect SMA cables to HighTechGlobal PCIe loopback adapter
  - a. Connect HTG-RX0P SMA connector to HTG-TX0P SMA connector using an SMA cable.
  - b. Connect HTG-RX0N SMA connector to HTG-TX0N SMA connector using an SMA cable.
11. Plug the entire AES-PZCC-FMC-V2-G into the HighTechGlobal PCIe loopback adapter

**Note:** If the AES-PZCC-FMC-V2-G is used with the HTG Loopback Adapter card, it might be necessary to support the card with a non-conductive box, roughly the height of the HighTech Global card

12. Slide the SW7 power switch to the ON position on the AES-PZCC-FMC-V2-G.



---

## Reusable Components

The tutorial does not use any components that are readily available for reuse, although all the IP that is used is provided as part of the Vivado Suite install.

## Experiment 1: Retrieve the design files

In this section, the design files for the reference design will be retrieved from the Avnet git repository.

1. Navigate to the following web site : <https://github.com/Avnet/hdl>
2. Click the **branch:master** button
3. Specify the following search criteria : ibert
4. Click the **Tags** tab

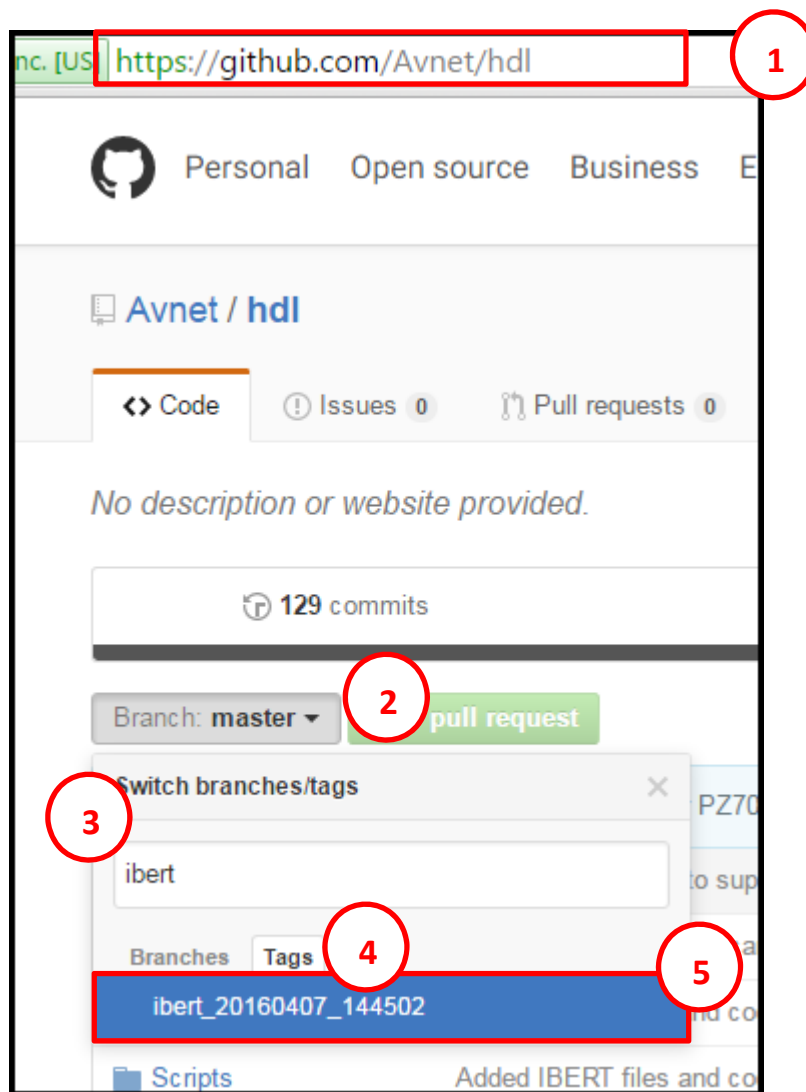


Figure 2 – Avnet GitHub repository – Retrieving specific version with tag

5. Select the **ibert\_20160407\_144502** tag

This will retrieve a known working version of the design files for the IBERT reference design for both PicoZed SOMs using either FMC Carrier Card target.

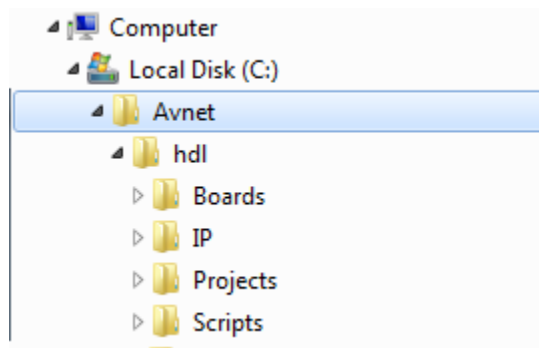
6. Click the Download ZIP file button



**Figure 3 – Avnet GitHub repository – Download ZIP**

7. Create an “Avnet” directory in your root C:\ drive
8. Save **hdl-ibert\_20160407\_144502.zip** file to the **C:\Avnet** directory, and extract the contents of the zip file in this directory
9. Rename the “hdl-ibert\_20160407\_144502” directory to “hdl”

You should see the following directory structure



**Figure 4 – Extracted C:\Avnet\hdl directory structure**

**NOTE:** The exact directory name is not critical, but it must remain short on Windows machines, due to the directory length limitation of Windows

The **C:\Avnet\hdl** repository contains the following sub-directories:

Directory	Content Description
C:\Avnet\hdl\Boards	contains board related files
C:\Avnet\hdl\IP	contains the IP cores used by ref designs
C:\Avnet\hdl\Projects	contains project related files
C:\Avnet\hdl\Scripts	contains scripts used to automatically build the designs

For the IBERT reference design, the following content is of interest:

Directory	Content Description
C:\Avnet\hdl\Projects\ibert	Folder containing files for IBERT reference design
C:\Avnet\hdl\Scripts\make_ibert.tcl	TCL script to launch the build of the IBERT reference design

By default, the script will build the design for all four combinations of the PicoZed 7015 and 7030 SOM, combined with either a FMC Carrier Card or the FMC V2 Carrier Card.

10. Edit the **make\_ibert.tcl** script to only build for your PicoZed SOM, as well as enable JTAG to allow the script to take the build all the way to configuring your board set.

As an example, if you have a PicoZed 7015 SOM, with FMC2 Carrier Card comment out the build for both PicoZed 7030 SOMs as well as the , PicoZed 7015 with FMC Carrier Card, as shown below:

```
# Set Build Variables
# set argv [list board=PZ7015_FMCCC project=ibert jtag=no version_override=yes]
# set argc [llength $argv]
# source ./make.tcl -notrace

# set argv [list board=PZ7030_FMCCC project=ibert jtag=no version_override=yes]
# set argc [llength $argv]
# source ./make.tcl -notrace

set argv [list board=PZ7015_FMC2 project=ibert jtag=yes version_override=yes]
set argc [llength $argv]
source ./make.tcl -notrace

# set argv [list board=PZ7030_FMC2 project=ibert jtag=no version_override=yes]
# set argc [llength $argv]
# source ./make.tcl -notrace
```

**Figure 5 – Editing the make script to build only for PicoZed 7015 SOM, with FMC2 Carrier Card**



## Experiment 2: Build the reference design

In this section, the Vivado project will be created and built with TCL scripts, implementing the IBERT Design.

1. From the Start menu, open the “Vivado 2015.4” GUI
2. In the TCL Console, Change to the **C:\Avnet\hdl\Scripts** directory

```
cd c:/Avnet/hdl/Scripts/
```

**Figure 6 – Vivado 2015.4 TCL Console – Changing to C:/Avnet/hdl/Scripts directory**

3. Launch the build with the “**source ./make\_ibert.tcl**” TCL command

```
source ./make_ibert.tcl
# set argv [list board=PZ7015_FMC2 project=ibert jtag=yes version_override=yes]
# set argc [llength $argv]
# source ./make.tcl -notrace
```

[illegible]

Setting	Configuration
Board	PZ7015_FMC2
Project	ibert
No Close Project	no
Version override	yes

Overriding Version Check, Please Check the Design for Validity!

### Figure 7 – Vivado 2015.4 TCL Console – Launching the build

The build will perform the following steps:

- Create and build the hardware design with Vivado 2015.4, including the Xilinx Coregen Integrated Bit Error Ratio Tester (IBERT)

- Perform a JTAG of the design of your design to your board set

[illegible]

### Figure 8 – Vivado 2015.4 TCL Console – Scripted Build Complete

---

## Experiment 3: Execute the reference design on hardware

If you did not opt to have the script JTAG your SOM, it is now time to perform the JTAG and begin analysis using Eye Scans. Please refer to the Avnet “Tech Tip - Transceiver Tools 102: We have an IBERT bit stream, now what?” at 3:06 seconds in. Here we walk you through performing a JTAG of the IBERT design to the PL of a PicoZed 7015 SOM. This same process is used for each configuration of the SOM. If you would rather have a printed document, please see the next section for a step-by-step procedure for programming your IBERT design.

If you would like to configure a customized IBERT, please refer to “Tech Tip - Transceiver Tools 101: Intro to IBERT” video for a step-by-step in performing a custom IBERT configuration.

You can find this video and many more at the PicoZed.org website under:  
Support → Training and Videos

For more information on tuning your transceivers, please refer to the other videos in the Avnet Transceiver Tech Tips Series.

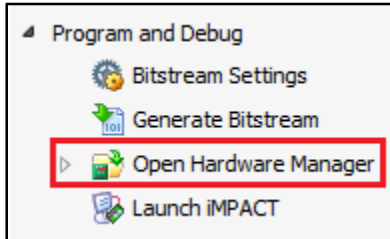
Tech Tip - Transceiver Tools 103: Now that we are running, what are all these adjustments?

Tech Tip - Transceiver Tools 104: Getting More Margin

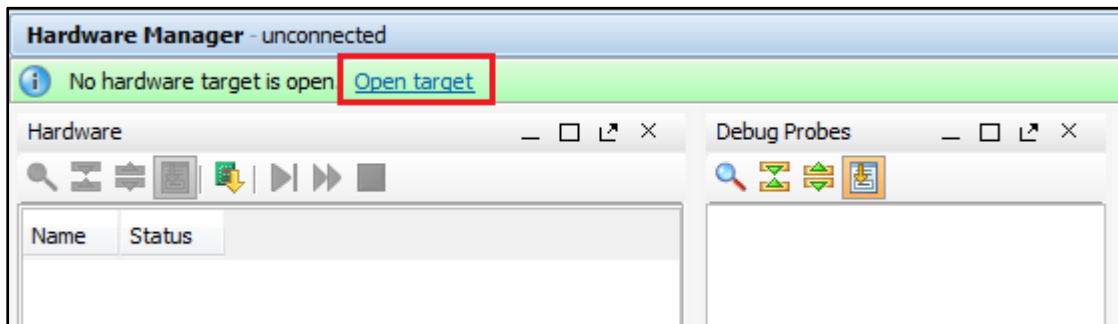
You can also view these videos on YouTube

## Running the IBERT Design on the PicoZed Carrier Card V2

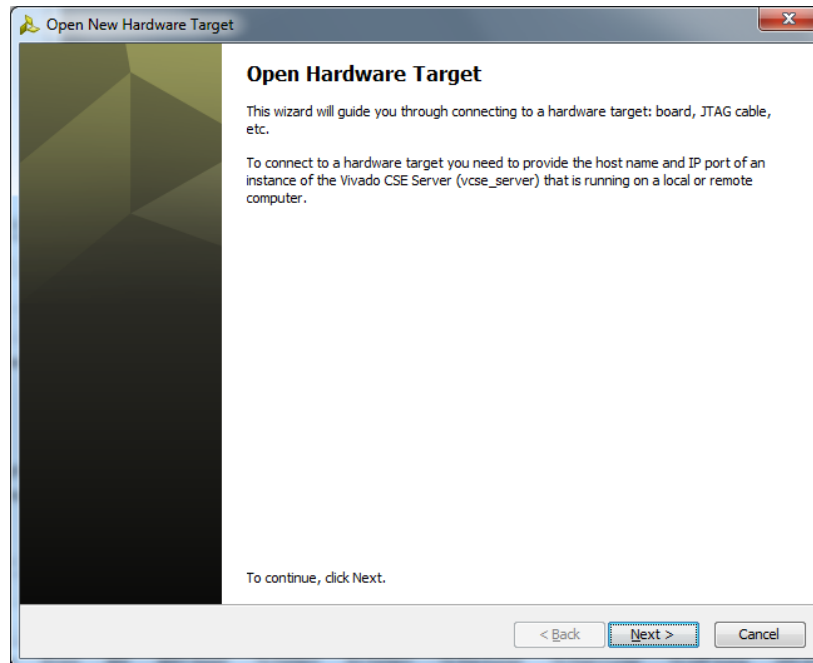
1. Click on the **Open Hardware Manager** in the Vivado GUI as shown in the following figure.



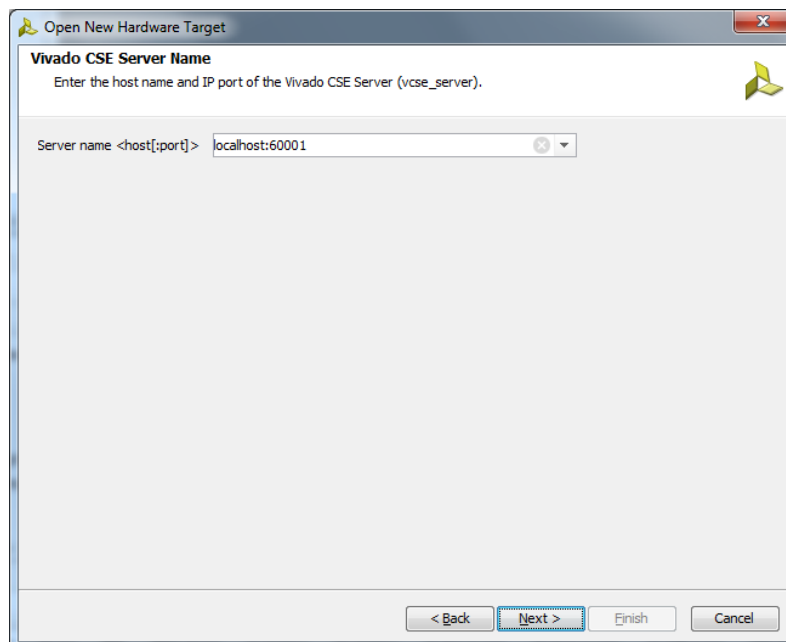
2. Click on the **Open target** and then select **Open New Target** in the Vivado GUI as shown in the following figure.



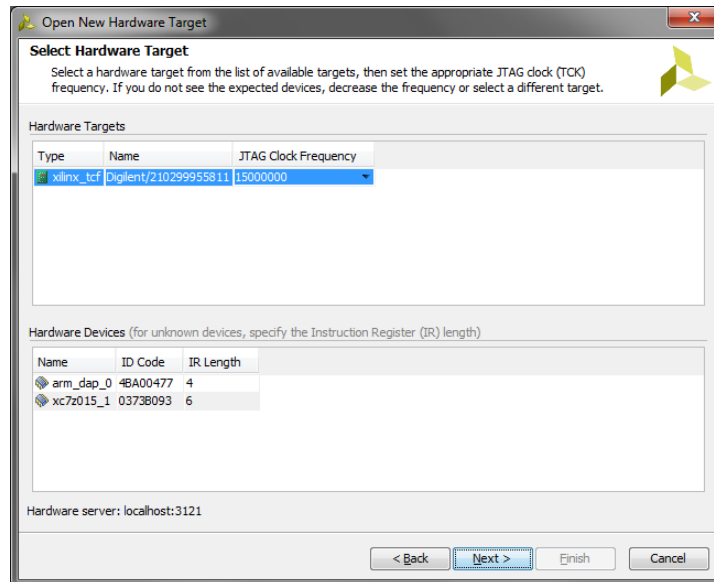
3. Click **Next** to continue.



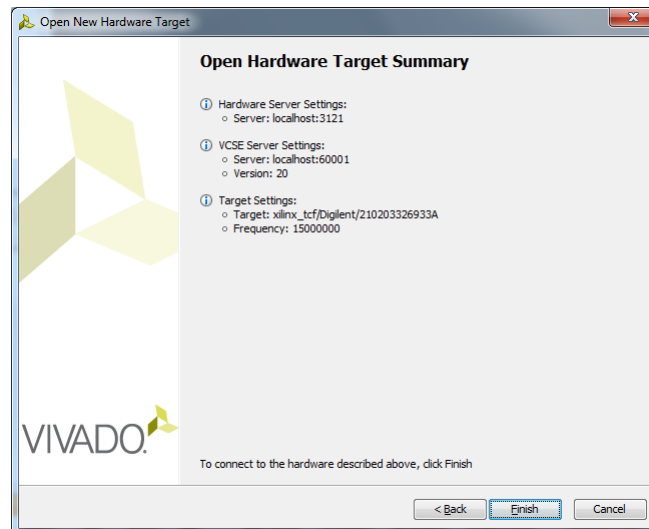
4. Click **Next** to continue.



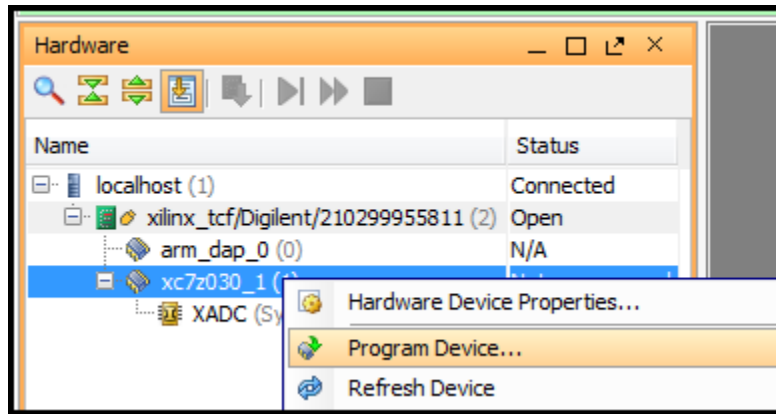
5. Click **Next** to continue.



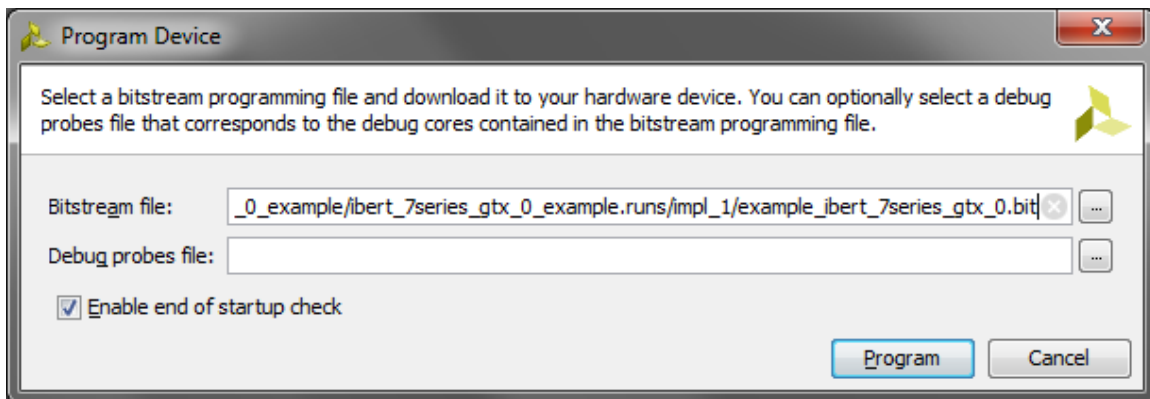
6. Click **Finish** to continue.



7. Right-click on the **XC7Z015\_1** or **XC7Z030\_1** device and select Program Device as shown in the following figure.

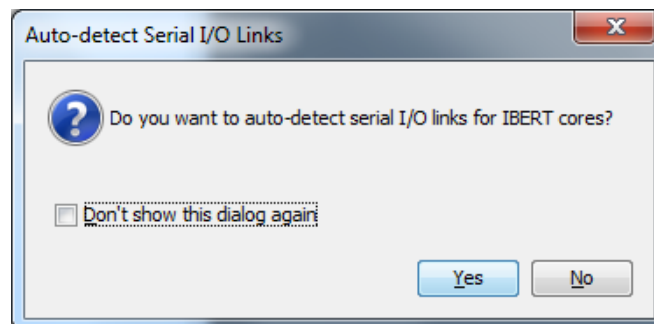


8. When the following dialog box appears, click **Program** to continue.

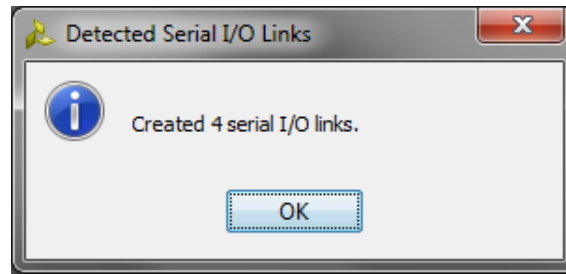


**Note:** Previous versions of Vivado always had a Debug probes file listed. This is not true for Vivado 2015.2.1

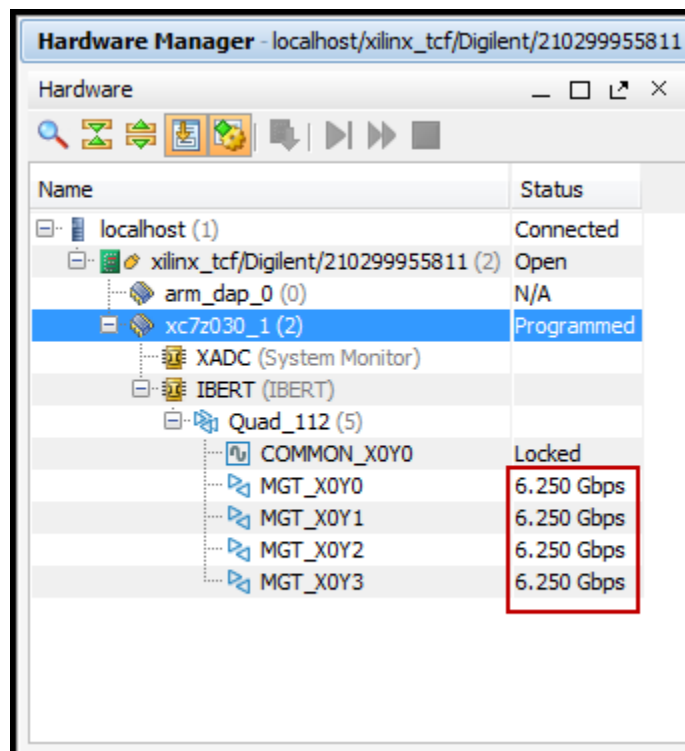
9. When the following dialog box appears, click **Yes** to continue.



10. When the following dialog box appears, click **OK** to continue.



11. The Vivado GUI should look as shown in the following figure. The MGT\_X0Y1 is connected to the SFP socket on the MMP baseboard while the MGT\_X0Y3 is connected to the SMA connectors on the MMP baseboard. The MGT\_X0Y0 and MGT\_X0Y2 in the GTX Quad 109 are not used in this IBERT example design.



12. You should see the following in the Vivado GUI. As seen, link0 through link3 are running at 6.25Gbps without any errors using PRBS 7-bit data pattern for a PicoZed 7030 SOM. You should see 3.75Gbps for a PicoZed 7015 SOM



Serial I/O Links

Name	TX	RX	Status	Bits	Errors	BER	BERT Reset	TX Pattern	RX Pattern
Ungrouped Links (0)									
Found Links (4)									
Found 0	MGT_X0Y0/TX	MGT_X0Y0/RX	6.252 Gbps	5.996E11	0E0	1.668E-12	Reset	PRBS 7-bit	PRBS 7-bit
Found 1	MGT_X0Y1/TX	MGT_X0Y1/RX	6.250 Gbps	5.998E11	0E0	1.667E-12	Reset	PRBS 7-bit	PRBS 7-bit
Found 2	MGT_X0Y2/TX	MGT_X0Y2/RX	6.250 Gbps	5.998E11	0E0	1.667E-12	Reset	PRBS 7-bit	PRBS 7-bit
Found 3	MGT_X0Y3/TX	MGT_X0Y3/RX	6.250 Gbps	6E11	0E0	1.667E-12	Reset	PRBS 7-bit	PRBS 7-bit

13. Set the **TX Pattern** and **RX Pattern** to **PRBS 31-bit** using the drop-down box to stress the links and then click on the **BERT Reset** to reset the Error counter as shown in the following figure.

Serial I/O Links										
Name	TX	RX	Status	Bits	Errors	BER	BERT Reset	TX Pattern	RX Pattern	
Ungrouped Links (0)										
Found Links (4)										
Found 0	MGT_X0Y0/TX	MGT_X0Y0/RX	6.250 Gbps	6.882E10	0E0	1.453E-11	Reset	PRBS 31-bit	PRBS 31-bit	
Found 1	MGT_X0Y1/TX	MGT_X0Y1/RX	6.250 Gbps	6.905E10	0E0	1.448E-11	Reset	PRBS 31-bit	PRBS 31-bit	
Found 2	MGT_X0Y2/TX	MGT_X0Y2/RX	6.250 Gbps	6.919E10	0E0	1.445E-11	Reset	PRBS 31-bit	PRBS 31-bit	
Found 3	MGT_X0Y3/TX	MGT_X0Y3/RX	6.250 Gbps	6.942E10	0E0	1.44E-11	Reset	PRBS 31-bit	PRBS 31-bit	

## Revision History

Date	Version	Revision
04 Jun 2015	2015.4	Initial release for PicoZed targets
14 May 2016	2015.4	Update for Git Flow including all PZ Carriers and SOMs