Virtex-6 FPGA Connectivity Kit

Getting Started Guide

UG664 (v1.4) July 6, 2011

XPN 0402826-03

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
02/26/10	1.0	Initial Xilinx release.
06/11/10	1.1	Removed references to specific release numbers for the ISE Design Suite, where applicable. Replaced Figure 1, Figure 22, Figure 23, Figure 24, Figure 25, Figure 26, and Figure 27. Removed update DVD from Connectivity Kit Contents. Added “in loopback mode” to step b, page 31. Removed DDR3 from Raw Data Path bullet in step 2a on page 33. In step 2b on page 33, changed the minimum value of the range from 128 to 64 for the XAUI and Raw Data paths and changed the Raw Data path option to one Packet Size instead of a minimum and a maximum. In step 2c on page 35, indicated to click **Start test**. Added the note under Figure 26. Replaced the “ISE 11.1 Software Installation” and “ISE 11.4 Software Update Installation” sections with a link to the Installation, Licensing, and Release Notes document. In step 8, page 40, removed the ISE Design Suite release number from the path. In Modifying the Virtex-6 FPGA Targeted Reference Design, added the note on page 42. Changed the command in step 4c on page 42. Changed the names of the BIT and MCS files in step 5d on page 43. Removed “double-click” from the Windows based script in step 8c on page 44. Removed sentence about the command shell opening from step 8d on page 44. Added the Next Steps section.
08/10/10	1.2	In step 4c on page 42, changed the filename to `mig3_5.xco` from `mig3_4.xco`. In Table 2, changed the implementation software tool entry to ISE Design Suite.
10/05/10	1.3	Added information for AXI4 protocol.
07/06/11	1.4	Removed descriptions of Virtex-6 FPGA Connectivity TRD being available in non-AXI4 protocol version throughout. Replaced v6_trd_quickstart with v6_trd_lin_quickstart. Added Windows platform to Board and Connectivity Targeted Reference Design Features. Updated step 6c on page 14. Added Install Windows Driver. Updated Figure 21. Added Install Linux Driver heading before step 12 on page 27.

Date	Version	Revision
07/06/11	1.4 *(Cont'd)*	Updated step 1 on page page 33. Removed Figure 25: Launch the Performance Monitor and Status GUI, and Figure 26: Run v6_trd_quickstart. Corrected coregen command in step 4c on page 42. Updated ending step instruction from step 8 to step 6 in Test Setup. Added Windows Driver. Added Linux Driver heading before step 1 on page 56. Updated step 2a and step 3a on page 56. Updated Table 1. Updated Figure 52 Figure 54, and Figure 61.

Table of Contents

Revision History 2

Preface: About This Guide

Additional Resources 7
Additional Support 7

Virtex-6 FPGA Connectivity Kit

Introduction 9
Connectivity Kit Contents 9
What is Inside the Box 9
What is Available Online 10
Getting Started with the Connectivity Targeted Reference Design Demo 10
Board and Connectivity Targeted Reference Design Features 10
Hardware Demonstration Setup Instructions 12
Install Windows Driver 15
Install Linux Driver 27
Evaluating the Virtex-6 FPGA Connectivity TRD 33
Installation and Licensing of ISE Design Suite 36
Downloading and Installing Tool Licenses 36
Modifying the Virtex-6 FPGA Targeted Reference Design 42
Hardware Modifications 42
Test Setup 44
Software Modifications 45
Windows Driver 45
Linux Driver 56
Next Steps 57
Connectivity TRD Modules 57
PCI Express 58
Packet DMA 59
Multiport Virtual FIFO and Memory Controller Block 60
XAUI 61
Software Device Driver and Software Application/GUI Files and Scripts 62
Simulating the Connectivity TRD 63
Reusing the DMA IP from Northwest Logic 63
Modifications to the Connectivity TRD 63
Getting Started with the Virtex-6 FPGA IBERT Reference Design 64
IBERT Hardware Demonstration Setup Instructions 64
Reference Design Files 75
Installation is Complete 75
Warranty 76

About This Guide

This Getting Started Guide describes the contents of the Virtex®-6 FPGA Connectivity Kit and provides instructions on how to start developing connectivity systems using Virtex-6 FPGAs.

Additional Resources

To find additional documentation, see the Xilinx website at:

http://www.xilinx.com/support/documentation/index.htm.

To search the Answer Database of silicon, software, and IP questions and answers, or to create a technical support WebCase, see the Xilinx website at:

http://www.xilinx.com/support.

Use this site for technical support regarding the installation and use of the product license file.

Additional Support

For questions regarding products within your Product Entitlement Account or if the email notification was received in error, send an email message to your regional Customer Service Representative:

- Canada, USA, and South America: isscs_cases@xilinx.com
- Europe, Middle East, and Africa: eucases@xilinx.com
- Asia Pacific including Japan: apaccase@xilinx.com

Virtex-6 FPGA Connectivity Kit

Introduction

The Virtex®-6 FPGA Connectivity Kit provides a comprehensive, high-performance Connectivity Development and Demonstration platform using the Virtex-6 family for high-bandwidth and high-performance applications in multiple market segments. The kit enables designing with common serial standards, such as PCI Express®, XAUI, and proprietary serial standards through the SMA interface.

Note: The screen captures in this document are conceptual representatives of their subjects and provide general information only. For the latest information, see the Xilinx® ISE® Design Suite.

Connectivity Kit Contents

This section describes the kit deliverables provided in the box and indicates what can be found on the Xilinx website.

What is Inside the Box

The kit consists of these elements:

- Virtex-6 LX240T FPGA-based ML605 Evaluation Board with:
  - One FMC Connectivity daughter card
  - One CX4 loopback module
  - Universal 12V power supply
  - Two USB A/Mini-B cables (used for download and debug)
  - One CompactFlash card (2 GB)
  - One DVI-to-VGA adapter
  - One Ethernet Cat5 cable
  - Four SMA cables
  - One SATA cable and one SATA loopback cable
- Xilinx ISE Design Suite DVD, including:
  - ISE Foundation™ software with ISE Simulator
  - PlanAhead™ Design and Analysis Tool
  - Embedded Development Kit (EDK)
  - Xilinx Platform Studio (XPS)
  - Software Development Kit (SDK)
  - ChipScope™ Pro Tool

- One USB stick containing reference designs, documentation, and demos
- Operating System: Fedora 10 LiveCD
- Virtex-6 FPGA Connectivity Kit documentation:
  - Welcome letter
  - Hardware setup guide
  - This Getting Started Guide

What is Available Online

Refer to the Xilinx website for this information:

- License for ISE Design Suite: Embedded Edition
  - http://www.xilinx.com/getproduct
  - http://www.xilinx.com/tools/faq.htm
- Connectivity Kit home page with documentation and reference designs:

  http://www.xilinx.com/v6connkit

  This home page provides information on:
  - USB stick contents (the current version of the data on the USB stick is available here)
  - Schematics, Gerber, and board bill of materials (BOM)
  - Additional detailed documentation
- Technical Support

  http://www.xilinx.com/support

Getting Started with the Connectivity Targeted Reference Design Demo

The Virtex-6 FPGA Connectivity Kit comes with a pre-built demonstration of the Virtex-6 FPGA Connectivity Targeted Reference Design (TRD) available on Platform Flash. The demo can be run before any additional tools are installed for an overview of the features of the ML605 Evaluation Board using a connectivity targeted reference design in the Virtex-6 LX240T FPGA.

Board and Connectivity Targeted Reference Design Features

The Virtex-6 FPGA Connectivity TRD (see Figure 1) demonstrates the main integrated components in a Virtex-6 FPGA. The Integrated Endpoint Block for PCI Express and GTX transceivers work together in an application with additional IP cores, such as a Northwest Logic Packet DMA engine for the PCI Express interface, XAUI LogiCORE™ IP, and Memory controller IP generated using the Memory Interface Generator (MIG).

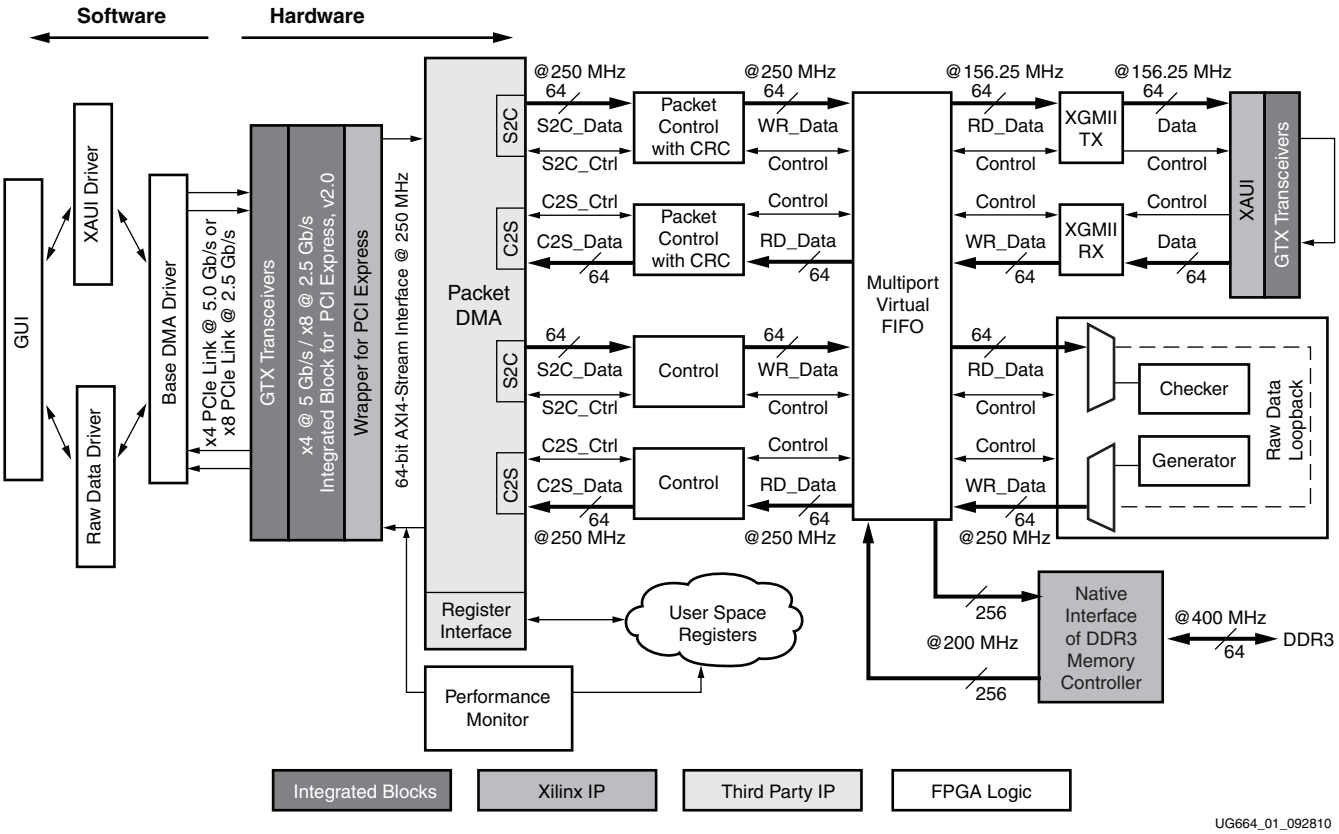


Figure 1: **Block Diagram of the Virtex-6 FPGA Targeted Reference Design**

The Virtex-6 FPGA Connectivity TRD features these components:

- Virtex-6 FPGA Integrated Block for PCI Express core configured as a 4-lane at 5 Gb/s or 8-lane at 2.5 Gb/s Endpoint for PCI Express, v2.0
- A performance monitor tracks the PCIe® data bandwidth by measuring data bus utilization on:
  - AXI4-Stream interface
- Packet DMA for PCI Express from Northwest Logic, a multichannel DMA that:
  - Supports full-duplex operation with independent transmit and receive paths
  - Provides a packetized interface on the backend similar to LocalLink
  - Monitors data transfers in the receive and transmit directions
  - Provides a control plane interface to access user-defined registers
- Multiport Virtual FIFO
  - The Memory Interface Controller is delivered through the Virtex-6 FPGA Memory Interface Generator (MIG) tool.
  - The virtual FIFO is a highly efficient layer around the native interface of the Virtex-6 FPGA Memory Controller and an external DDR3 memory device.
- XAUI LogiCORE IP that utilizes serial I/O transceivers to provide a throughput of up to 10 Gb/s

  XAUI TX and XAUI RX blocks align data as per the XGMII format.
- Control logic to interface between the DMA and the multiport Virtual FIFO.

- Software driver for 32-bit Windows and 32-bit Linux platforms:
  - Configures the hardware design parameters
  - Generates and consumes traffic
  - Provides a Graphical User Interface (GUI) to report status and performance statistics

Hardware Demonstration Setup Instructions

This section describes how to set up the hardware demonstration for the Virtex-6 FPGA Connectivity TRD. This demonstration outlines a bridging function between PCIe and XAUI protocols. It also provides accesses to an onboard DDR3 memory.

1. Equipment Checklist: The following equipment is required to run the hardware demonstration:
   - Virtex-6 FPGA Connectivity Kit
   - PC system with a x8 PCIe slot on the motherboard, CD ROM drive, and a USB port
   - Monitor, keyboard, and mouse
2. Inadequate Equipment: Run the alternate demonstration.

   If there is no access to any of the equipment in step 1, refer to Getting Started with the Virtex-6 FPGA IBERT Reference Design, page 64 to alternately bring up the ML605 board included in the Virtex-6 FPGA Connectivity Kit. Otherwise, continue with the PCIe to XAUI protocol demonstration in step 3.
3. Completion of Hardware Setup Guide Checkpoint:

   If the instructions in the Virtex-6 FPGA Connectivity Kit Hardware Setup Guide have already been completed to bring up the Virtex-6 FPGA Connectivity Kit, proceed to Evaluating the Virtex-6 FPGA Connectivity TRD, page 33; otherwise, continue to step 4.
4. Hardware Setup I: Board setup and configuration.

   The ML605 board is shipped with the FMC Connectivity daughter card attached to the FMC_HPC connector (see Figure 2). To run the Connectivity TRD demonstration, you need to externally loop back the XAUI data through a CX4 loopback connector provided in the connectivity kit.

   a. Verify the switch settings are correct:
      - Switch S1: 1 = OFF, 2 = OFF, 3 = OFF, 4 = ON
      - Switch S2: 1 = ON, 2 = OFF, 3 = OFF, 4 = ON, 5 = ON, 6 = OFF

   b. Verify that Jumper J42 has pins 3-4 shorted.

   c. Plug in the CX4 loopback connector:
      - Remove the plastic pin protector.
      - Plug in the CX4 loopback connector on the FMC Connectivity daughter card's J2 connector (see Figure 3).

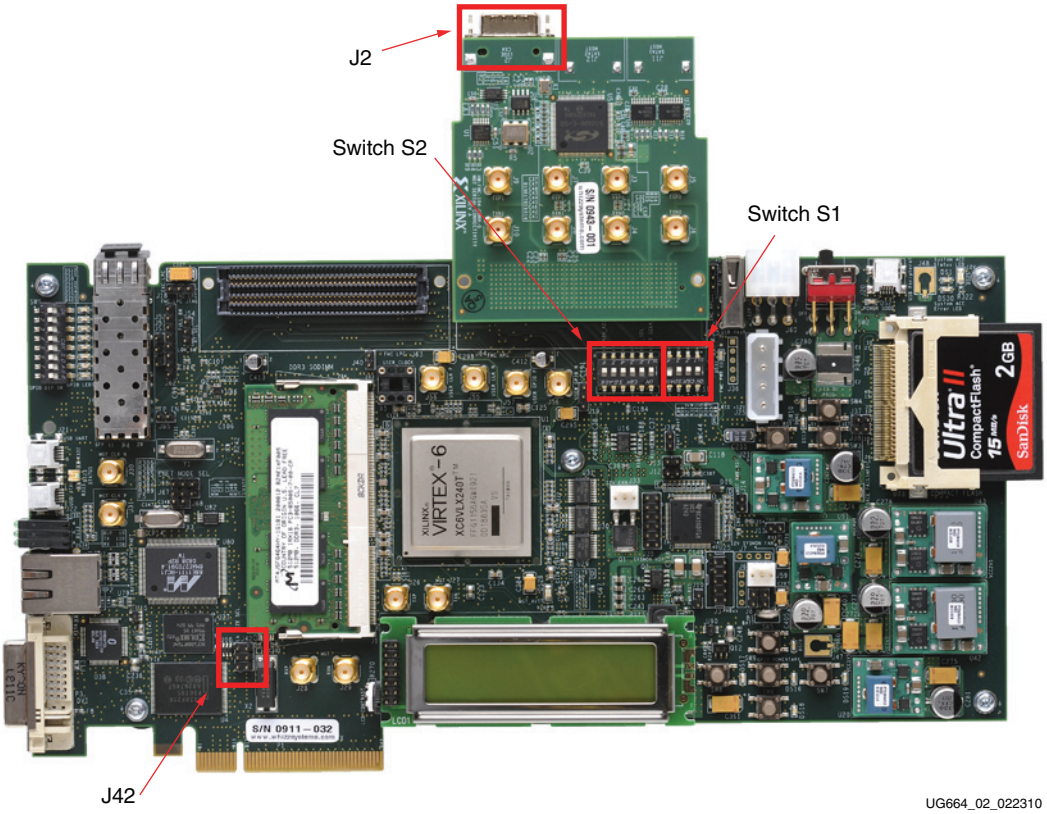


Figure 2: **ML605 and FMC Connectivity Daughter Card**

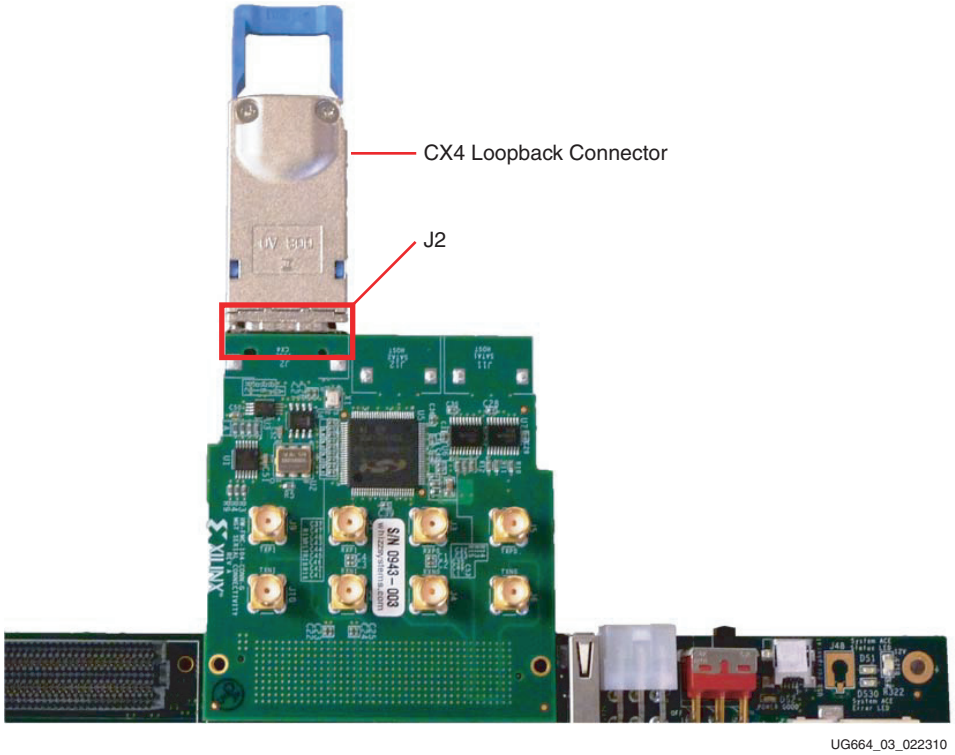


Figure 3: **CX4 Connector**

5. Hardware Setup II: Connect the power connector:
   a. Turn the PC system off.
   b. Connect the 12V ATX power supply's available 4-pin disk-drive-type power connector of the PC system to the board (J25).

   Caution! Using any power supply connector other than the 4-pin inline connector results in damage to the PC system and the ML605 board.

   c. The power switch SW2 should be switched to the ON position (away from the bracket edge of the ML605 board) as shown in Figure 4.

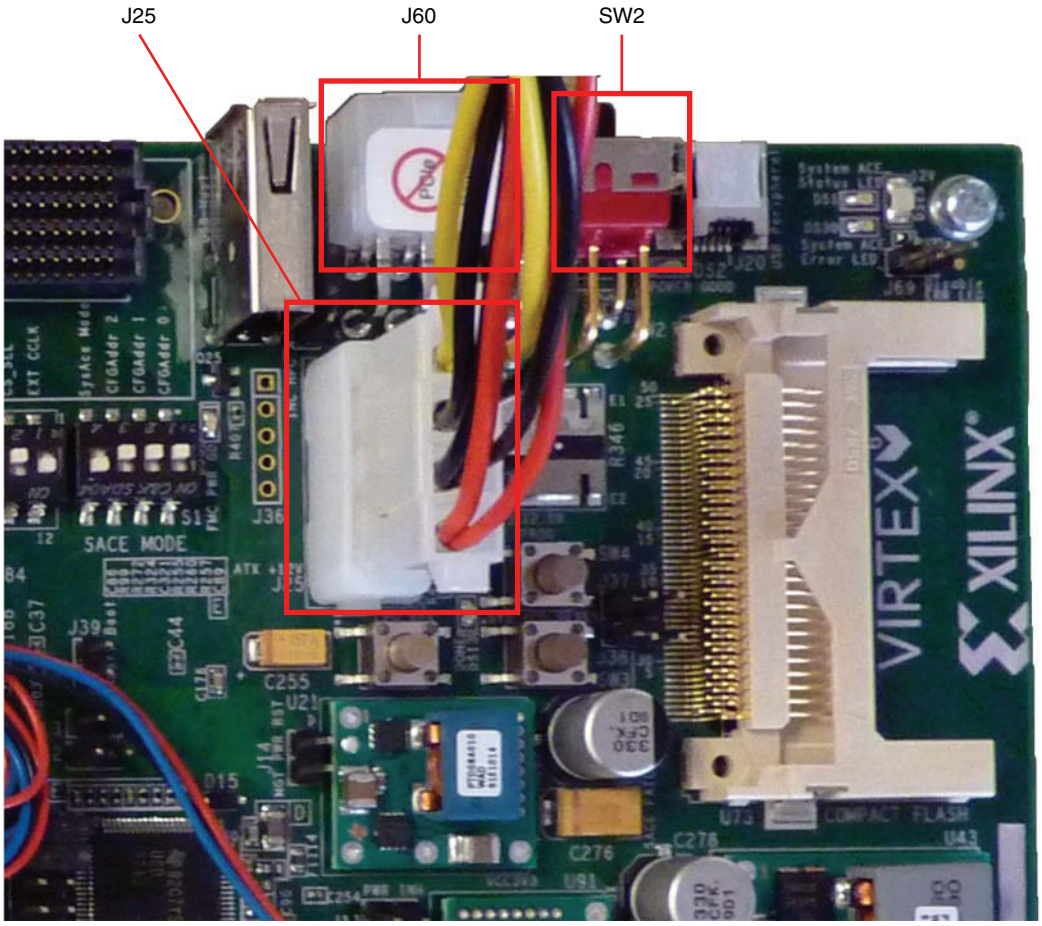
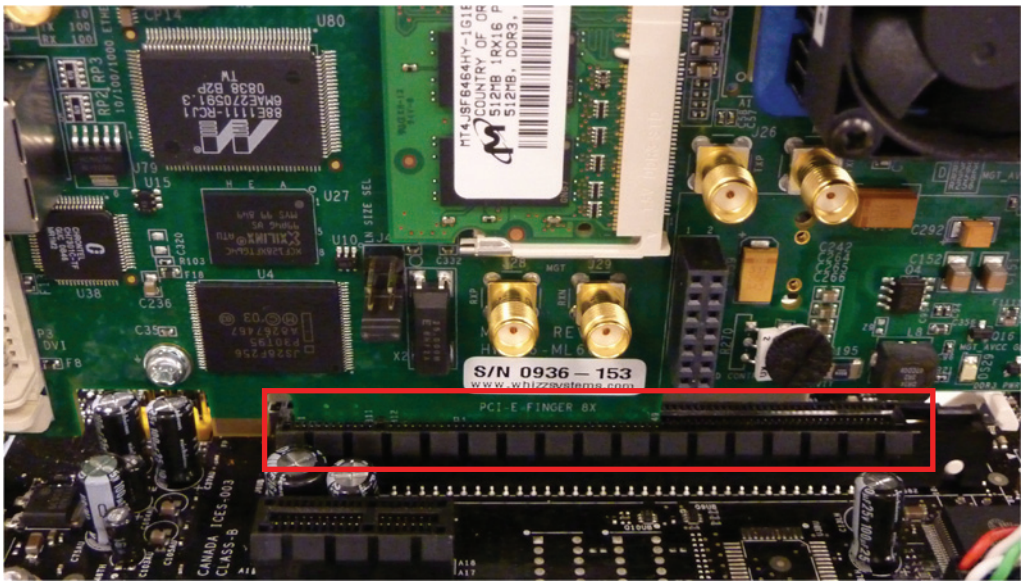


Figure 4: **12V ATX Power Supply Connector**

6. Hardware Setup III: Insert the ML605 board into an empty PCIe slot:
   a. Identify a x8 or a x16 PCIe slot on the PC motherboard.
   b. Insert the ML605 board with FMC daughter card and CX4 loopback module in the PCIe slot through the PCIe x8 edge connector.
   c. On power-up, the connectivity targeted reference design for PCIe to XAUI is loaded from the Platform Flash.

After the hardware setup is complete, the user can choose to install the Windows driver or the Linux driver. Proceed to Install Linux Driver, page 27 to verify the design on the Fedora 10 operating system.



UG664_05_011610

Figure 5: **Insert the ML605 Board into the PCIe Slot**

Install Windows Driver

7. Ignore the Found New Hardware Wizard:
   a. Power the PC system on and wait for the operating system (OS) to load.
   b. The system recognizes a new PCIe endpoint card connected to it and starts the Found New Hardware Wizard.
   c. Click on **Cancel** to close the wizard (Figure 6).

UG664_57_052011

Figure 6: **Ignore Found New Hardware Wizard**

8. Copy the contents of the USB flash drive:
   a. The reference design files are provided on the USB flash drive delivered with the connectivity kit.
   b. Insert the USB flash drive into a USB connector of the PC system and copy the `v6_pcie_10Gdma_ddr3_xaui_axi` folder to the PC system.

Note: Ensure that the path where the `v6_pcie_10Gdma_ddr3_xaui_axi` folder is located does not have spaces.

9. Install the drivers and GUI:
   a. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi` folder.
   b. Double click on `x_v6_trd_setup.exe` (Figure 7).

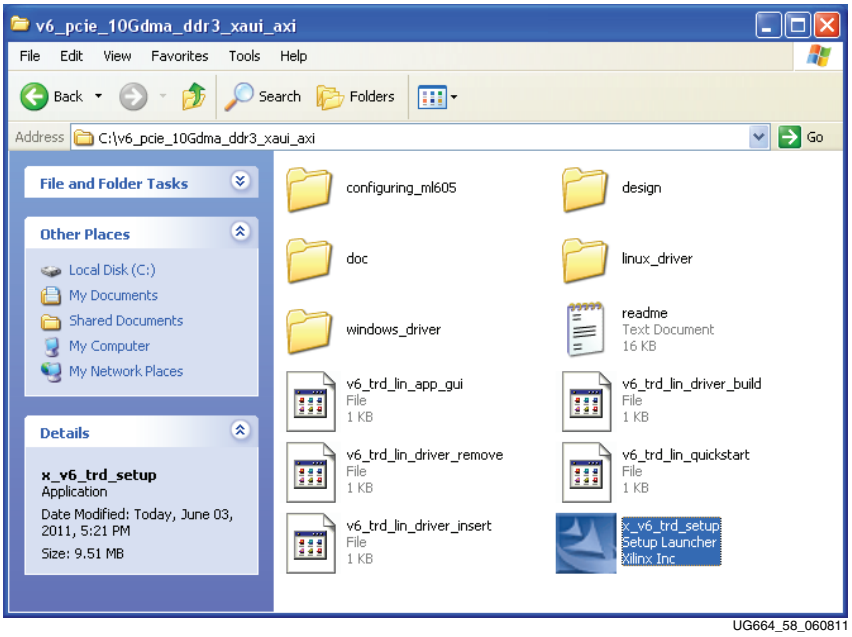


Figure 7: **Run `x_v6_trd_setup.exe`**

c. The InstallShield wizard for the Virtex-6 FPGA Connectivity TRD is launched (Figure 8). Click on **Next** to select the setup type.

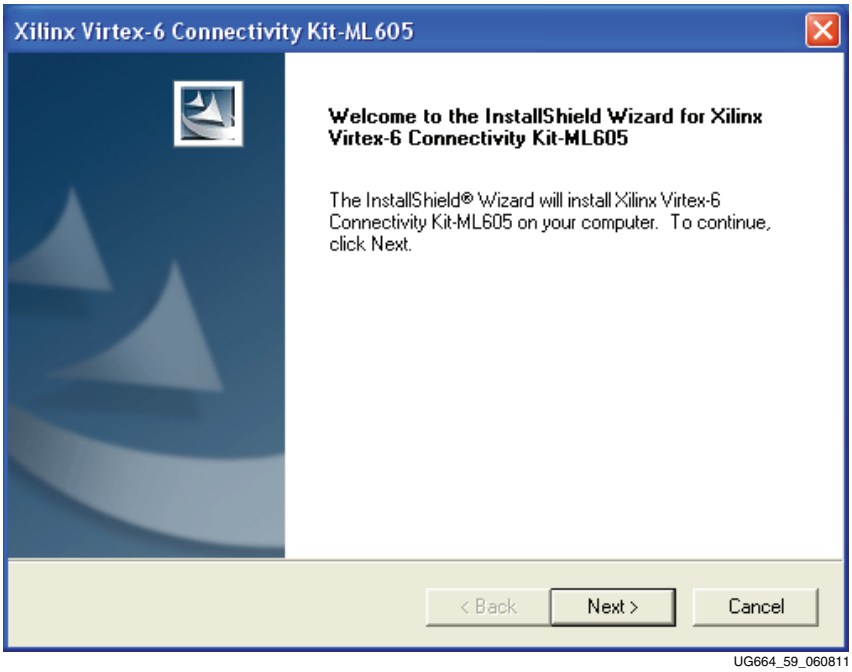


Figure 8: **InstallShield Wizard is Launched**

d. Select **Typical** to set `C:\Program Files` as the destination directory where driver files will reside. Click on **Next** and confirm the Setup Type selection (Figure 9).

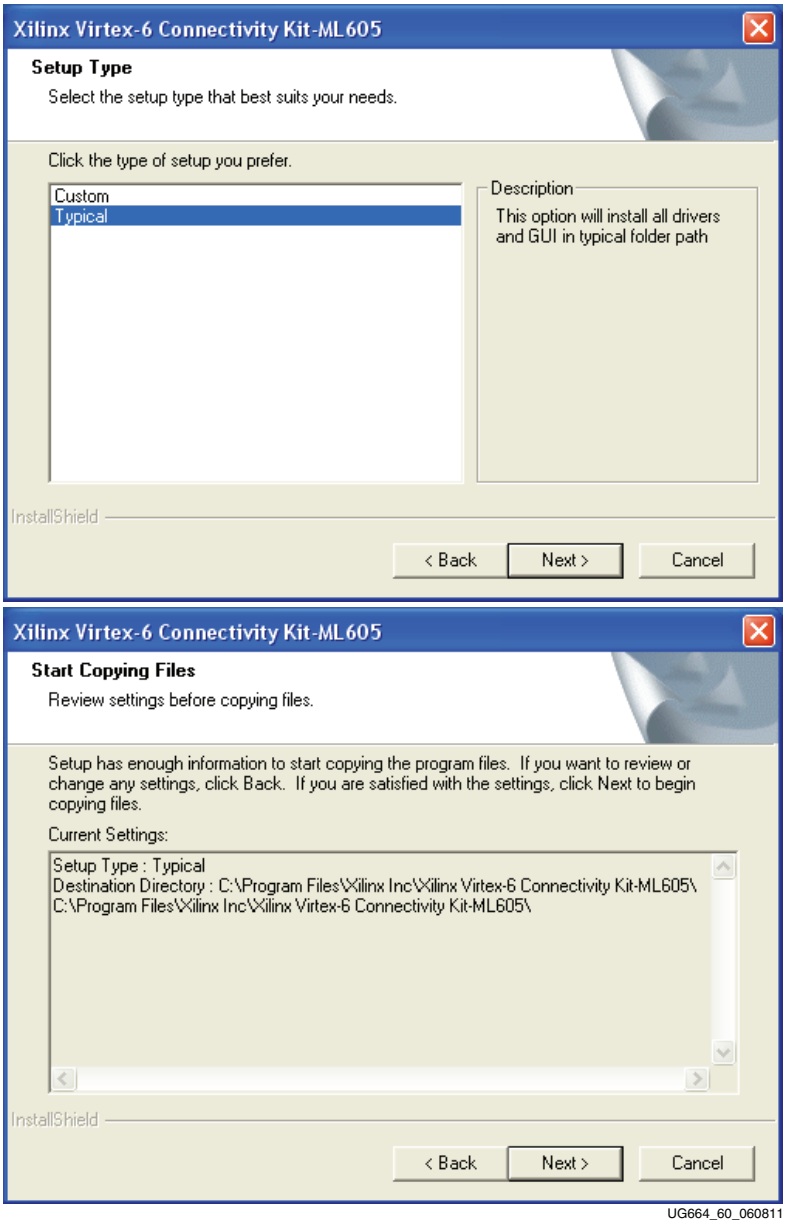


Figure 9: **Set Directory to which the Driver Files are Copied**

e. Click on **Next**. When the InstallShield wizard completes, click on **Finish** (Figure 10). At the end of this install process, the driver and GUI files are copied to the `C:\Program Files\Xilinx Inc\Virtex6` folder. Also, a shortcut to the Xilinx Performance Monitor (`xpmon`) GUI is available on the desktop.

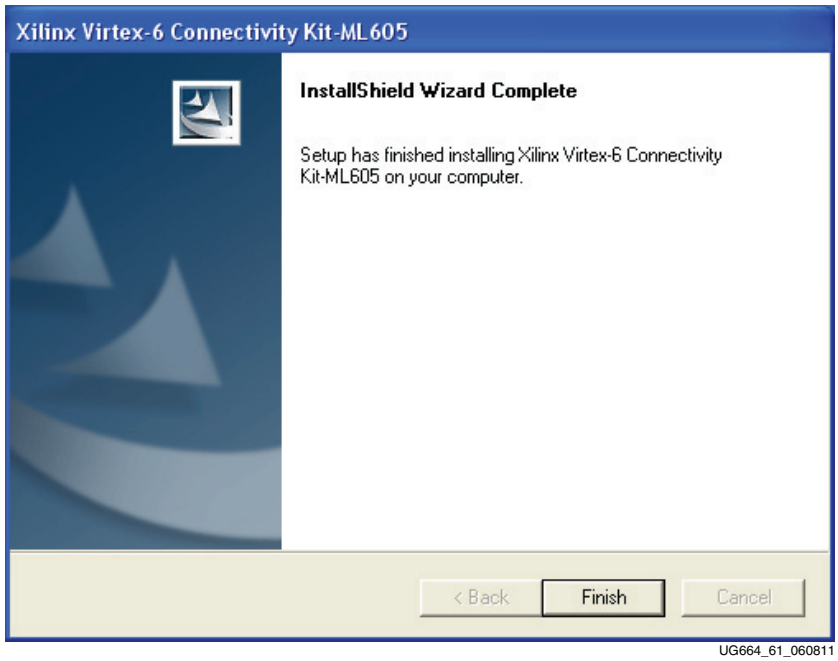


Figure 10: **InstallShield Wizard Copies GUI and Driver Files into Program Files when Done**

10. Load the drivers:
    a. After the InstallShield wizard completes, Add Hardware Wizard is launched (Figure 11). Click on **Next**.

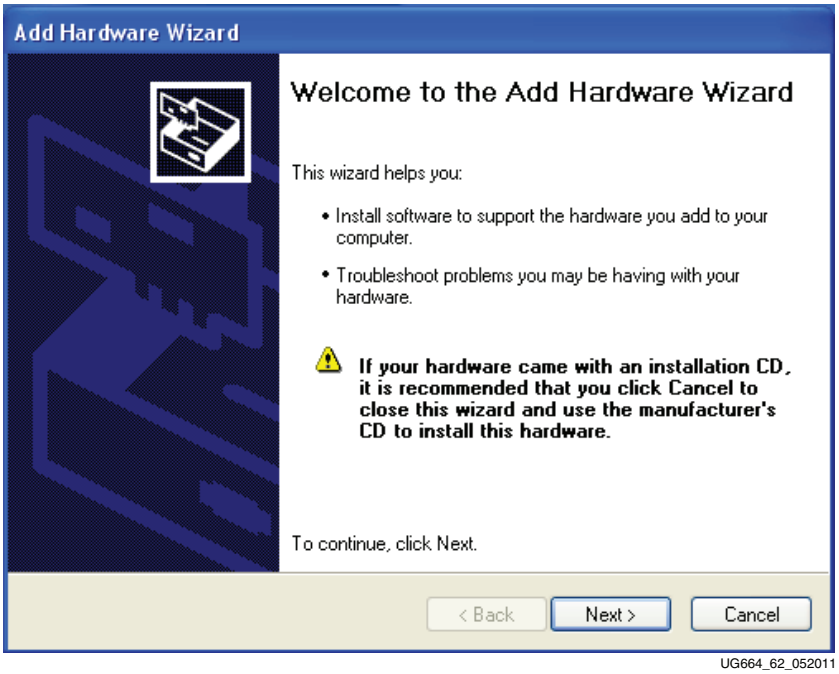


Figure 11: **Launch Add Hardware Wizard**

b. The Hardware wizard finds a PCI Simple Communications Controller VID:10EE DID:6042 and prompts a search of the driver through the Windows Update website (Figure 12). Select **No, not this time** and click on **Next** to get the driver files available on the system.

Note: This window might not appear on all systems.

Figure 12: **Found New Hardware Wizard**

c. Select **Install the software automatically** and click on **Next**. The system associates the Xilinx DMA driver to the PCI Simple Communications Controller. Click on **Finish** to proceed and install the child drivers Raw Data and XAUI (Figure 13).

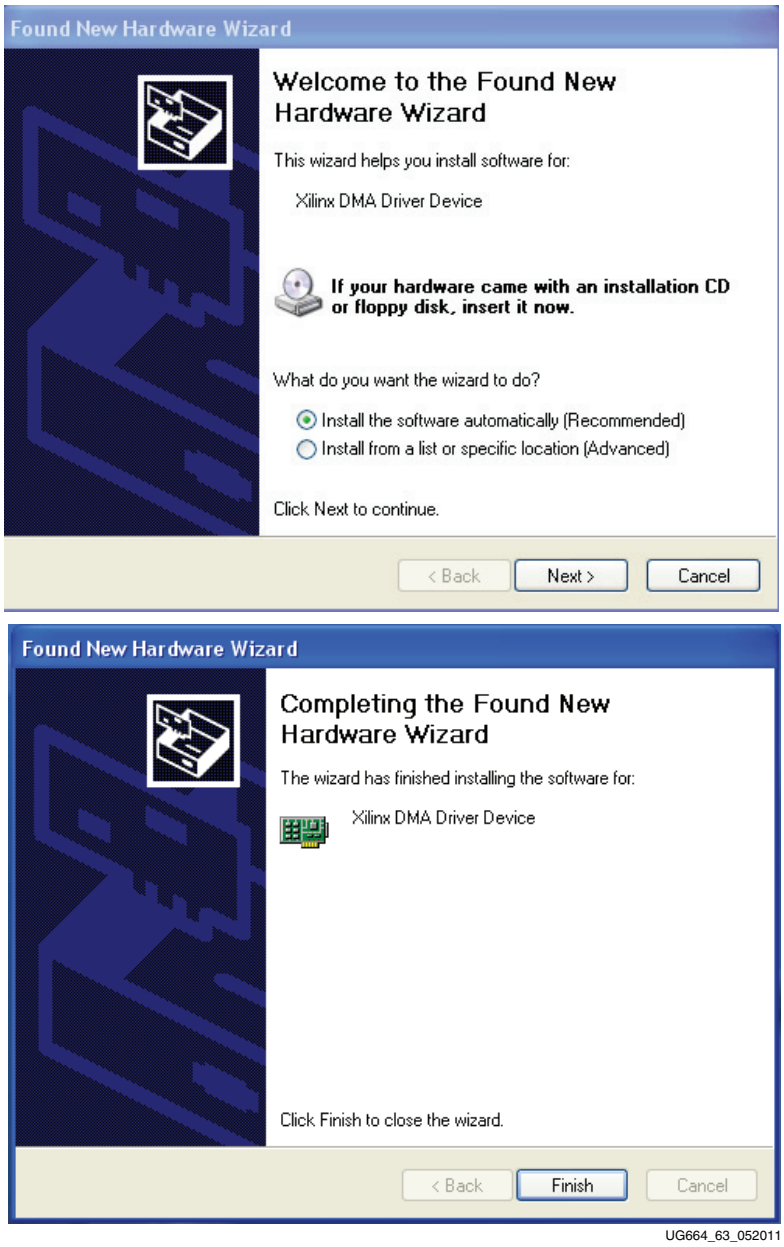


Figure 13: **Load Driver - XDMA**

d. Select **No, not this time** for the driver to be searched on the Windows Update website and click on **Next** to get the driver files available on the system (Figure 14).

Note: This window might not appear on all systems.

Figure 14: **Found New Hardware Wizard**

e. Select **Install the software automatically** and click on **Next** to install the Xilinx Raw Data driver (Figure 15). Click on **Finish** to proceed to installation of the XAUI driver.

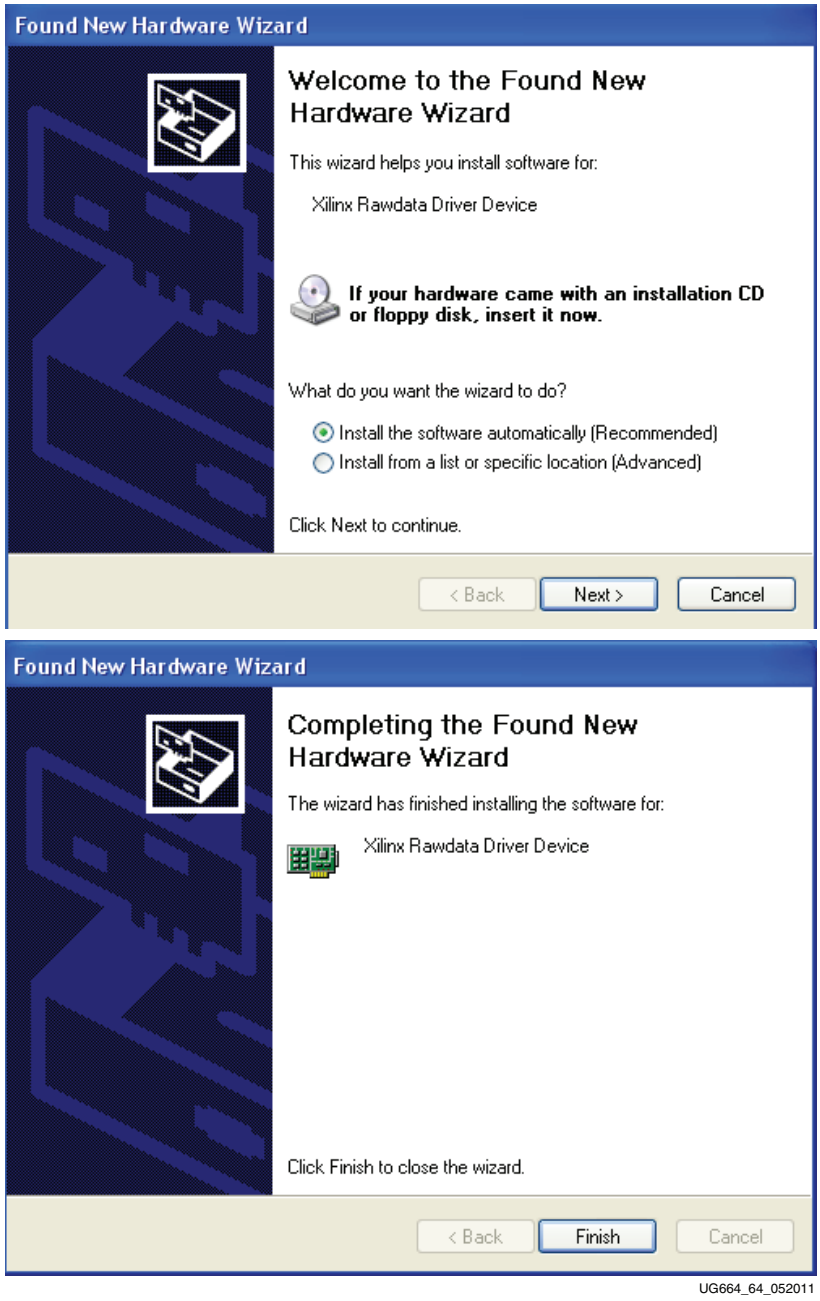


Figure 15: **Load Driver - Raw Data**

f. Select **No, not this time** for the driver to be searched on the Windows Update website and click on **Next** to get the driver files available on the system (Figure 16).

Note: This window might not appear on all systems.

Figure 16: **Found New Hardware Wizard**

g. Select **Install the software automatically** and click on **Next** to install the Xilinx XAUI driver (Figure 17). Click on **Finish**.

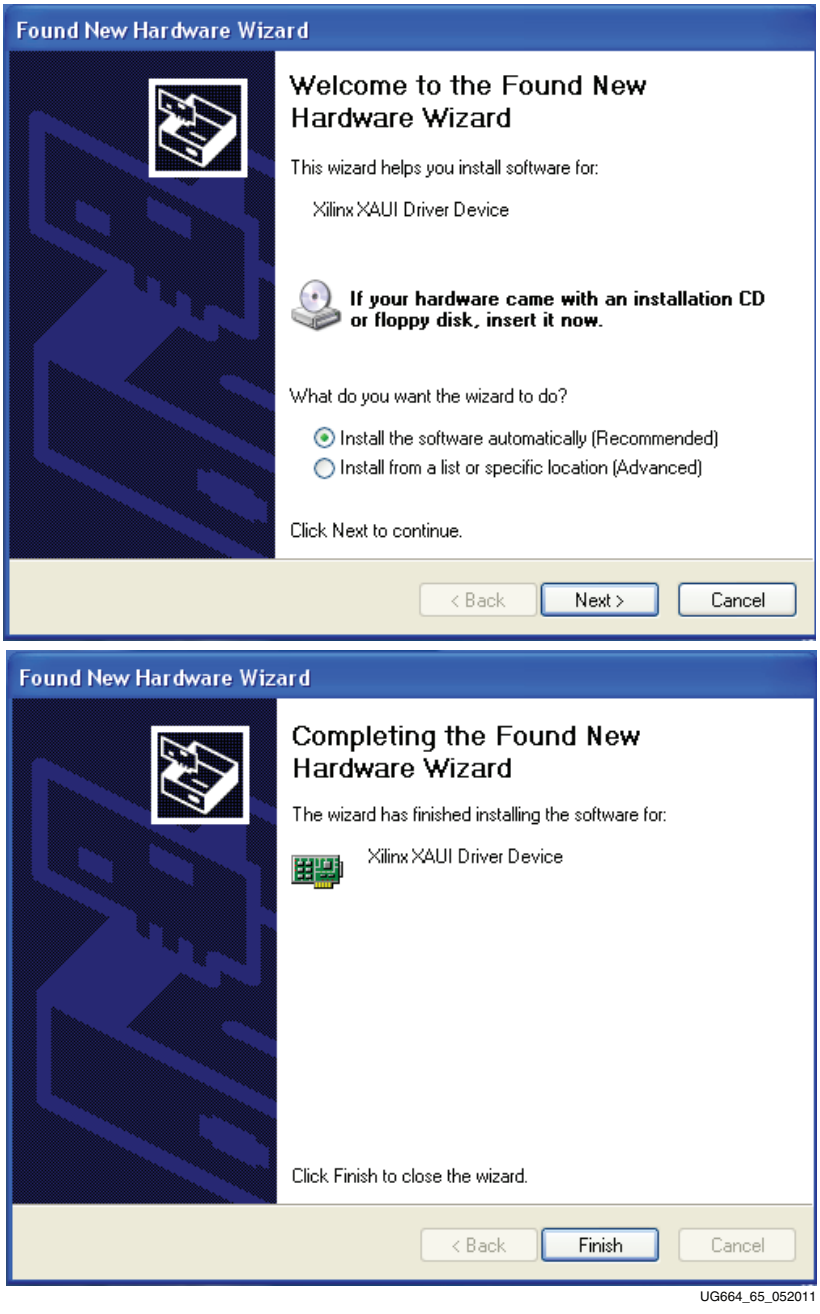


Figure 17: **Load Driver - XAUI**

All the drivers required to run the Virtex-6 FPGA Connectivity TRD are successfully installed. Click **Finish** to exit the Add Hardware Wizard (Figure 18).

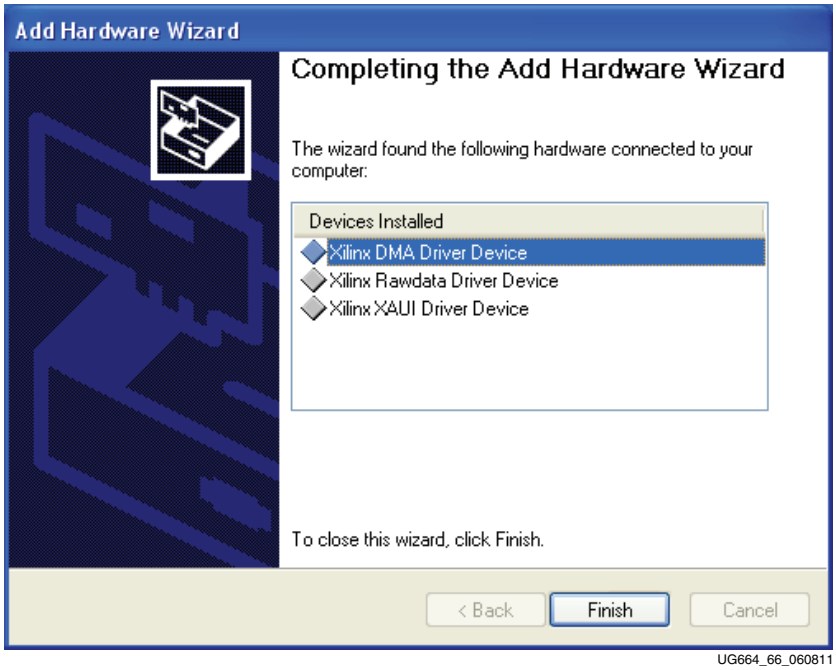


Figure 18: **All Drivers Installed Successfully**

Note: After the drivers are installed, `x_v6_trd_setup.exe` need not be run until the user makes modifications to the software source code. Running `x_v6_trd_setup.exe` again causes drivers to uninstall and clean the install folders.

11. Launch the GUI:
    a. Double-click on the `xpmon` icon on the desktop to launch the Performance Monitor application (Figure 19). Proceed to step 16 to run the application GUI.

       The `xpmon` icon can also be found in the `C:\Program Files\Xilinx Inc\Virtex6` folder.

Figure 19: **Launch GUI**

Install Linux Driver

12. Configure the desktop PC to boot from the CD ROM:
    a. Power the PC system on and watch the initial BIOS screen for a prompt that indicates which key to use for either:

- A boot menu or
- The BIOS setup utility

b. If such a prompt is not visible, refer to the manufacturer's documentation for the PC system. On many systems, the required key is F12, F2, F1, or Delete.

c. Adjust the boot menu or BIOS boot order settings to make sure that the CD ROM is the first drive in the boot order.

d. Eject the CD ROM bay and insert Fedora 10 LiveCD.

e. Save changes and exit the boot menu or BIOS setup.

f. The PC system will boot from the CD ROM.

13. Boot Fedora 10 LiveCD and automatically log in:

a. The images in Figure 20 are displayed on power up. Wait for the login screen to be displayed. This could take two to three minutes or longer depending on system configuration.

b. Click Login to enter. The desktop should appear in a couple of minutes, depending on the system configuration.

Figure 20: **Fedora Screens**

14. Copy the contents of the USB flash drive:

a. The reference design files are provided on the USB flash drive delivered with the connectivity kit.

b. Insert the USB flash drive into a USB connector of the PC system.

c. Wait for the Fedora 10 operating system to mount the USB flash. When the flash is mounted, an icon pops up on the desktop.

d. Double-click the USB flash drive icon and copy the `v6_pcie_10Gdma_ddr3_xaui_axi` folder into the liveuser's home folder directory.

e. Unmount the USB flash. Right-click on the USB flash drive icon and select **Unmount Volume**.

15. Load the driver and launch the Performance Monitor application:
    a. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi` folder.
    b. Double-click **v6_trd_lin_quickstart** to build the kernel objects, load the device driver, and launch the Performance Monitor application.
    c. A window prompt appears as shown in Figure 21. Click **Run in Terminal** to proceed.

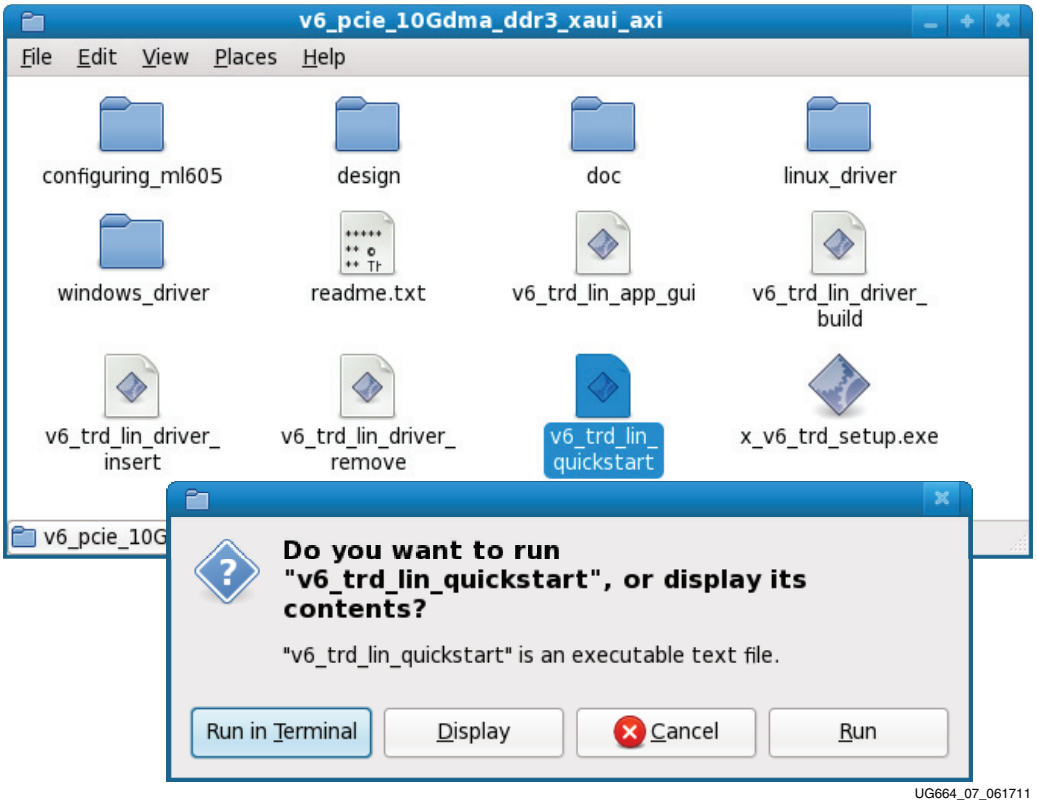


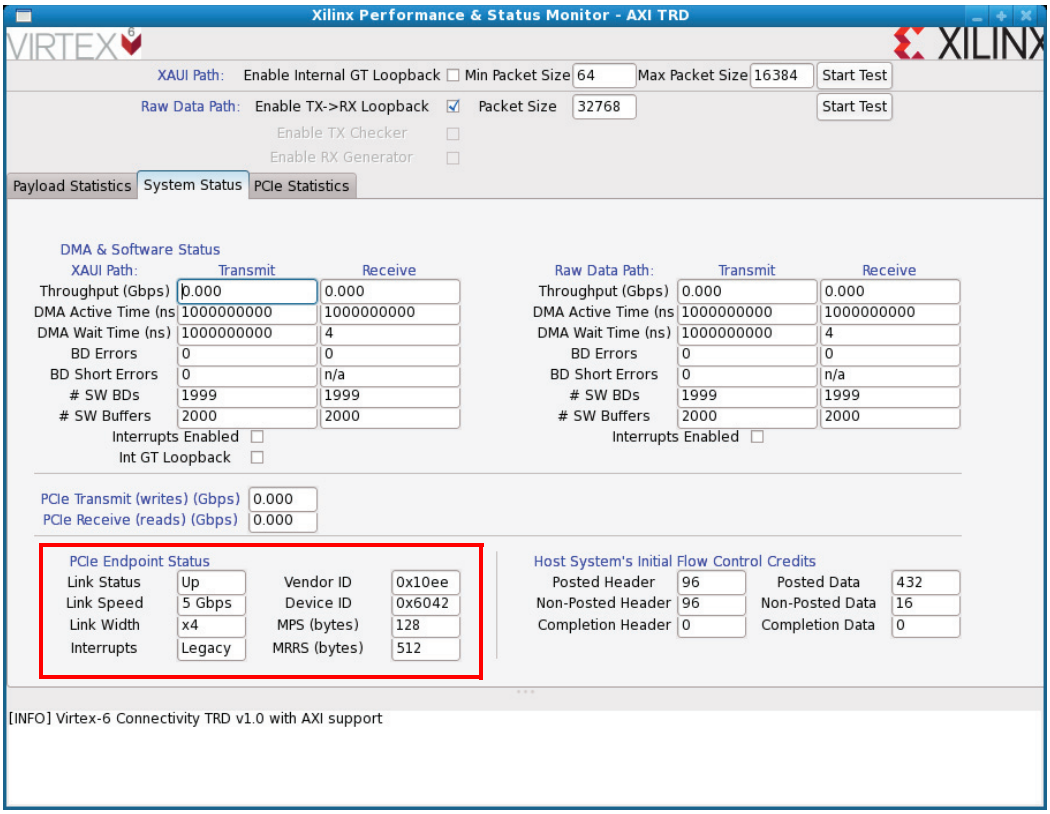
Figure 21: **Load Driver and Launch Performance Monitor Application**

16. Performance Monitor Application: Verify the board status.
    a. Click on the **System Status** tab to verify the status of the ML605 board and the PCIe link (see Figure 22):
        - Link Status: Up

          This confirms that the PCIe link is up and a PCIe connection is established between the Virtex-6 FPGA Endpoint for PCI Express and the PC motherboard chipset.
        - Link Speed: 5.0 Gbps

          This confirms that the PCIe link is operating at line rate speeds per PCI Express, v2.0.
        - Link Width: x4

          This confirms that the PCIe link is trained as a x4 link.



UG664_08_090810

Figure 22: **Verify Board Status in the Performance Monitor**

17. Performance Monitor Application: Start the data traffic.
    a. To enable the XAUI datapath, click **Start Test** as shown in Figure 23.

       This enables the driver to start generating the data traffic for the DMA channel connected to the XAUI path.

    b. To enable the Raw Data path in loopback mode, click **Start Test** as shown in Figure 23.

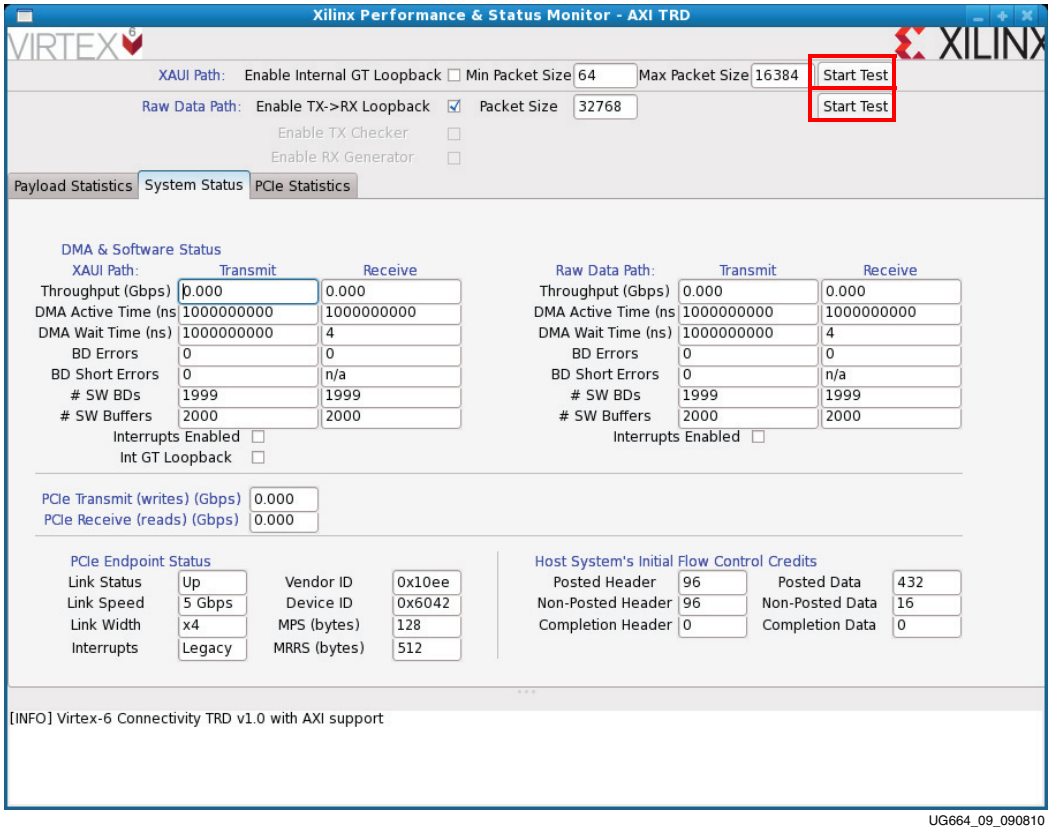


Figure 23: **Start Data Traffic from the Performance Monitor**

18. Performance Monitor Application: In the dialog box shown in Figure 24, verify data throughput and error-free operation:
    a. Verify the PCIe throughput is approximately 10 Gbps.
    b. Verify the DMA channel throughput for the XAUI path and DMA channel throughput for the Raw Data path together is approximately 9 Gbps.
    c. Verify there are no buffer descriptor errors for error-free operation.

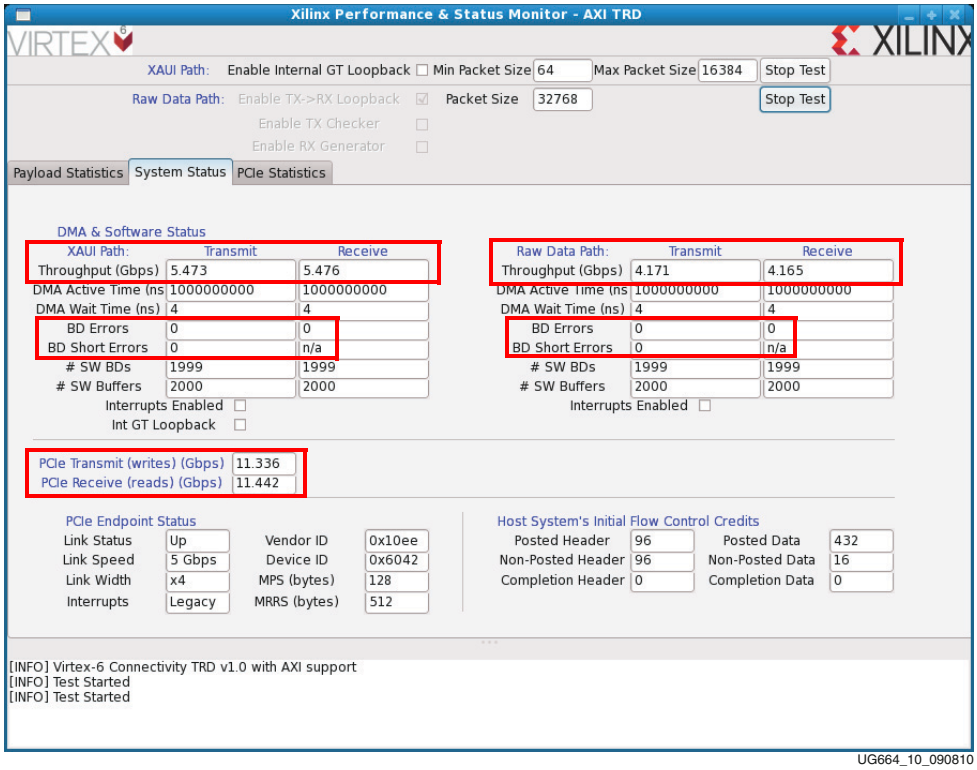


Figure 24: **Verify Error-Free Operation in the Performance Monitor**

Congratulations! The Virtex-6 FPGA Connectivity Kit is now set up. The pre-built connectivity targeted reference design demonstration has been tested, using the built-in block for PCI Express (4-lane 5 GT/s configuration for PCI Express, v2.0), XAUI LogiCORE IP module, a Virtual FIFO memory controller that interfaces to the onboard DDR3 SODIMM device, and a third-party DMA controller for PCI Express.

Evaluating the Virtex-6 FPGA Connectivity TRD

The Virtex-6 FPGA Connectivity TRD provides a Performance and Status monitor application and GUI. The application enables customers to evaluate different system parameter optimizations. This section demonstrates key performance criteria for the PCI Express, XAUI, and Raw Data Path (Memory) interfaces.

To evaluate the Virtex-6 FPGA Connectivity TRD:

1. Launch the Performance Monitor for the Virtex-6 FPGA Connectivity TRD:

   In Windows:

   a. Double-click the **xpmon** icon on the desktop to launch the Performance Monitor application.

   In Linux:

   a. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi` folder.

   b. Double-click **v6_trd_lin_quickstart** to launch the Performance Monitor and Status GUI.

   c. A window prompt appears as shown in Figure 21. Click **Run in Terminal** to proceed.

2. Set up the test in the Performance Monitor:

   a. Two Data Transmission options are provided:

      - XAUI Path
      - Raw Data Path

   b. These Packet Size options are provided:

      - XAUI Path

        Minimum Packet Size: Choose a value between 64 - 16384 bytes

        Maximum Packet Size: Choose a value between 64 - 16384 bytes

      - Raw Data Path

        Packet Size: Choose a value between 64 - 32768 bytes

3. Execute the test and view payload statistics in Performance Monitor:
   a. Click **Start Test** to start the performance test.
   b. View the payload statistics to review data transfers on the XAUI Path and Raw Data Path channels of the DMA engine (see Figure 25).

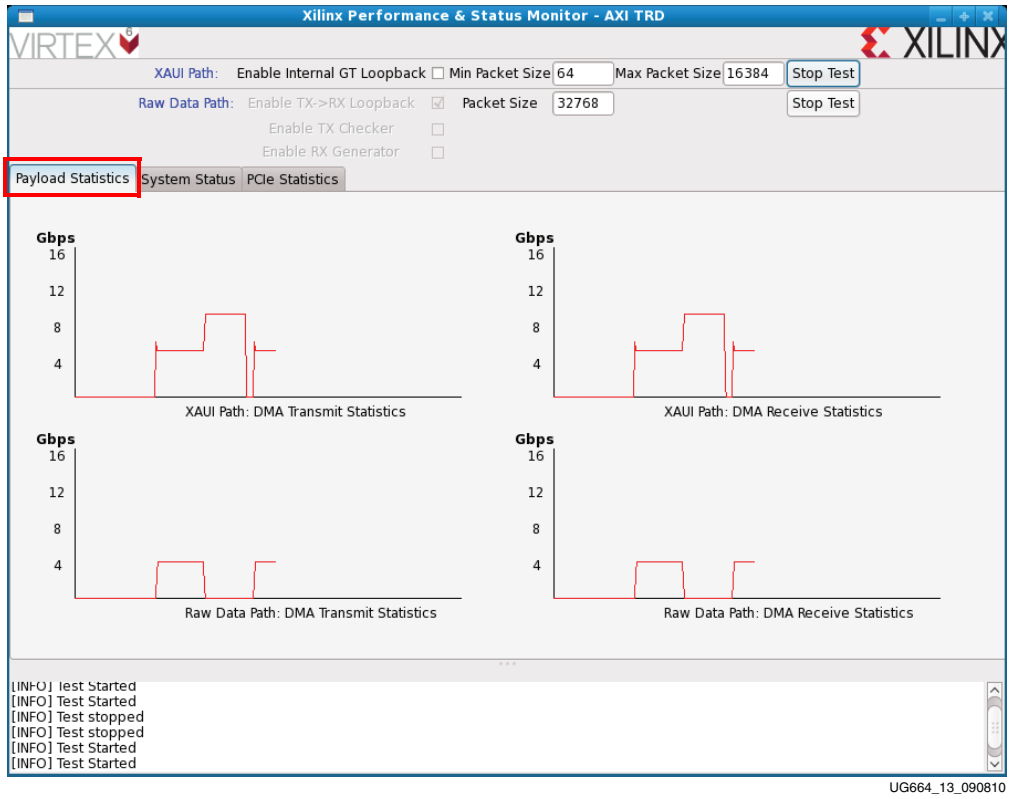


Figure 25: **Payload Statistics**

c. Modify the Packet Size parameters for the XAUI Path and Raw Data Path transfers (see Figure 26) and click **Start test**. Then view the payload statistics to review data transfers on the XAUI Path and Raw Data Path channels of the DMA engine.

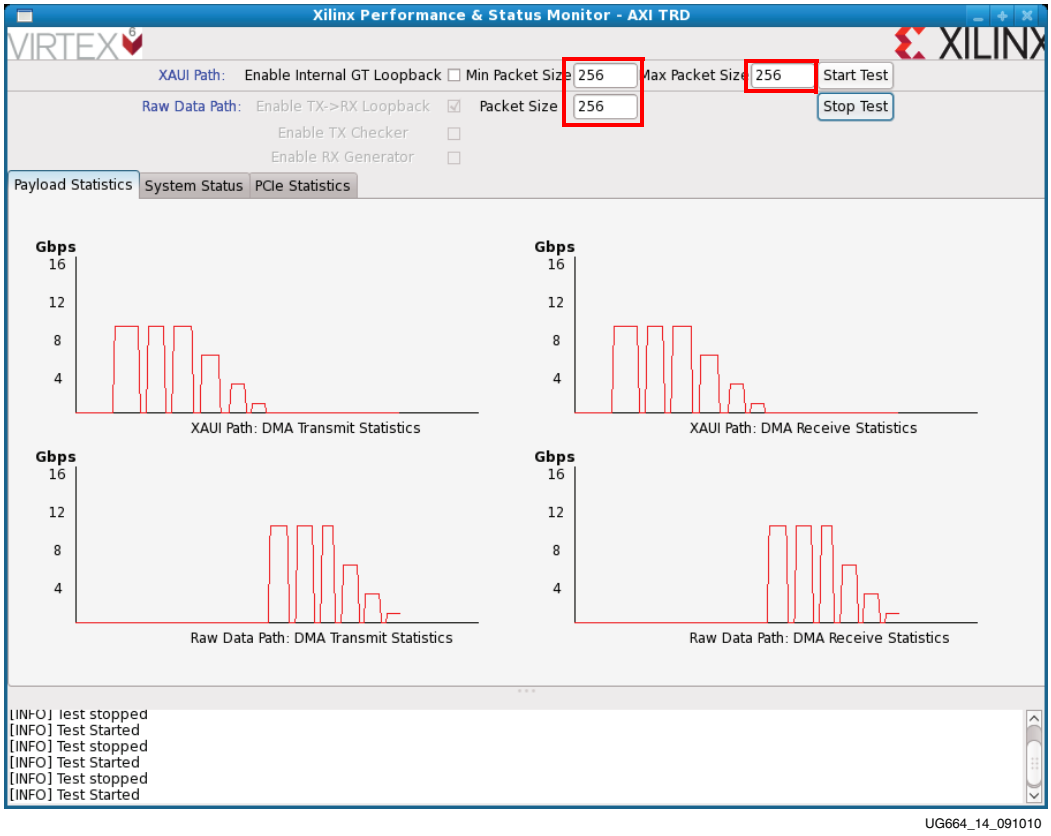


Figure 26: **Packet Size Field in the Payload Statistics Tab**

Note: For packet sizes equal to 64 or 128 bytes, the throughput is reduced and might not be visible on the Payload Statistics tab. The exact values can be viewed on the System Status tab.

4. View the PCIe statistics in Performance Monitor (see Figure 27). Click **PCIe Statistics** to view data transfer numbers on the PCIe interface.

Figure 27: **PCIe Statistics in the Performance Monitor**

Congratulations! The system performance of the Virtex-6 FPGA Connectivity Kit has been evaluated using the pre-built demonstration design. This design includes the built-in integrated block for PCI Express (4-lane, 5 GT/s configuration for PCI Express v2.0), XAUI LogiCORE IP, a Virtual FIFO memory controller designed to interface to the onboard DDR3 SODIMM device, and a third-party DMA controller for PCI Express.

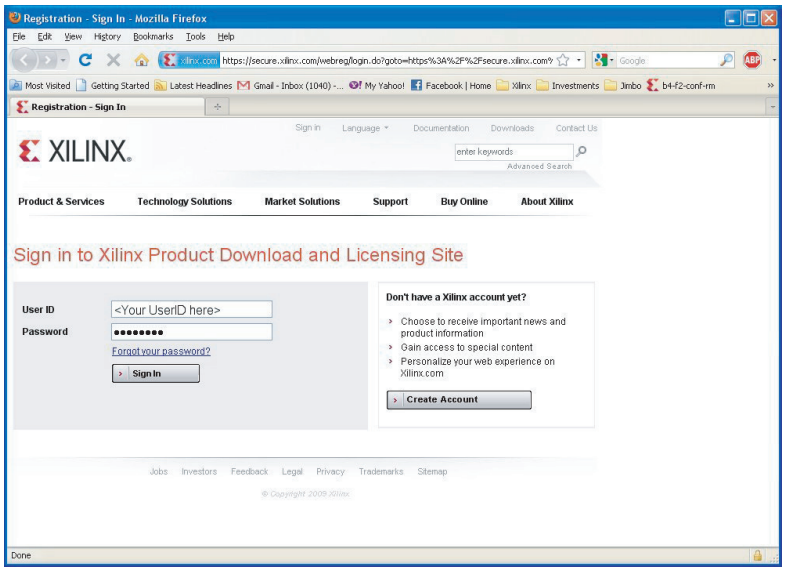
Installation and Licensing of ISE Design Suite

This Virtex-6 FPGA Connectivity Kit comes with an entitlement to a full seat of the ISE Design Suite: Embedded Edition that is device locked to a Virtex-6 LX240T FPGA. This software can be installed from the DVD or the Web installer can be downloaded from http://www.xilinx.com/support/download/index.htm.

For detailed information on licensing and installation, refer to UG631, *ISE Design Suite: Installation, Licensing, and Release Notes*, located on the Xilinx documentation site at http://www.xilinx.com/support/documentation.

Downloading and Installing Tool Licenses

1. Visit the Xilinx software registration and entitlement site at http://www.xilinx.com/getproduct to access the Xilinx product download and licensing site (Figure 28).



UG664_29_011710

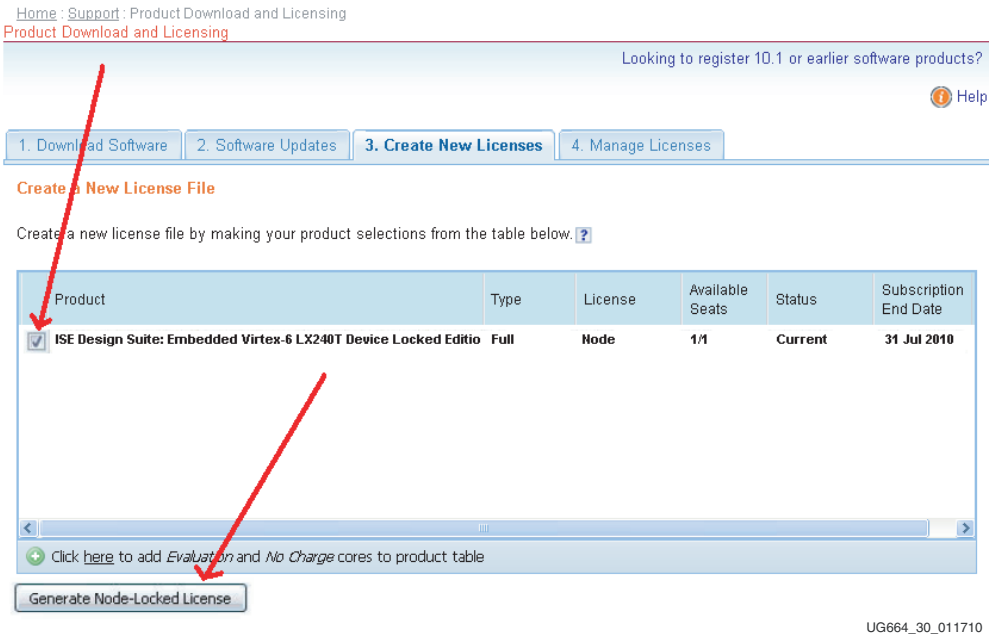
Figure 28: **Sign In to Xilinx Product Download and Licensing Site**

2. Log in to an existing account or create a new account, if needed.

   Note: The user name and password are provided in an email sent when the kit was ordered. If this e-mail is unavailable, contact Xilinx customer service for access to the account at http://www.xilinx.com/support/techsup/tappinfo.htm.

3. After logging in, verify the shipping address, if prompted. Click **Next** after the shipping address has been verified or updated.

4. Check the **ISE Design Suite Embedded Virtex-6 LX240T Device Locked Edition** product box and click **Generate Node-Locked License** as shown in Figure 29.



UG664_30_011710

Figure 29: **Xilinx Entitlement Center**

5. Follow the instructions to generate the license by providing the Host OS information and Host ID (Disk Serial number or Ethernet MAC address) as shown in Figure 30. Click **Next**.

   Note: Laptop users might want to select their Disk ID or Wireless Ethernet card Host ID. Laptops on docking stations might have three Ethernet Host IDs from which to choose. If a docking station Host ID is selected, then the license is available only when docked. It is best to avoid the Host ID of the RJ45 Ethernet connection on laptop computers, because some Ethernet adapters power down when not plugged into the network. To select an Ethernet adapter, it is best to select the wireless card.

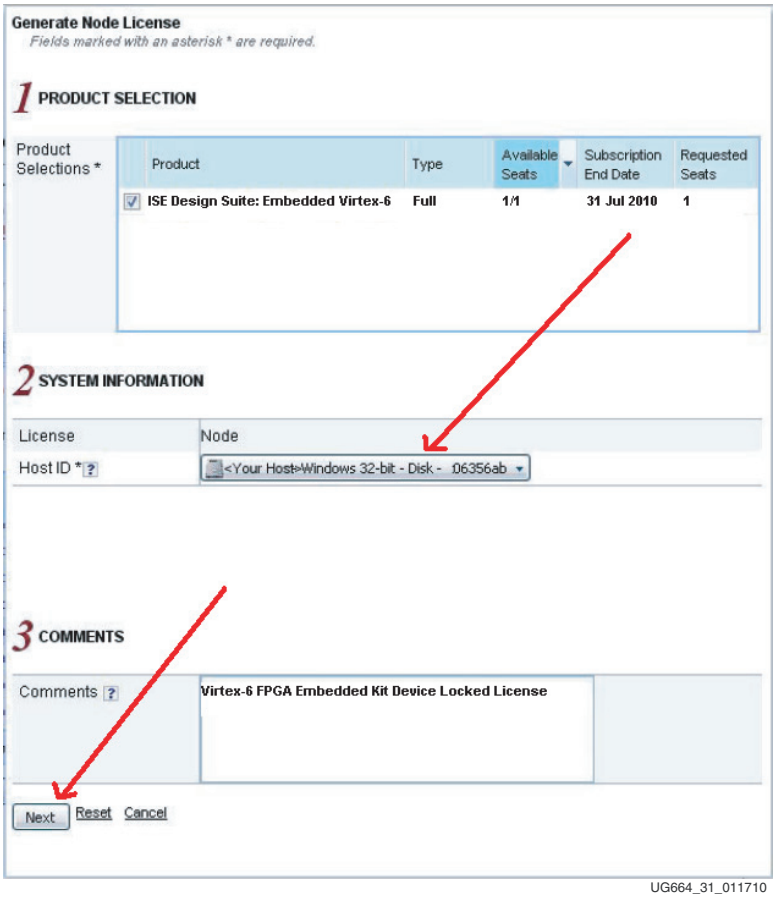


Figure 30: **Selecting the Host ID**

6. Review the license request as shown in Figure 31 and click **Next**.

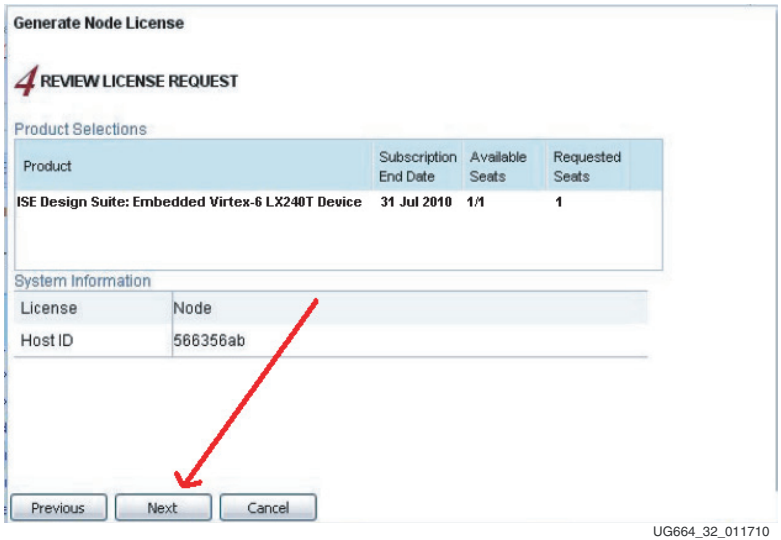


Figure 31: **Reviewing the License Request**

7. The generated license is e-mailed to the user in an e-mail similar to the one shown in Figure 32.

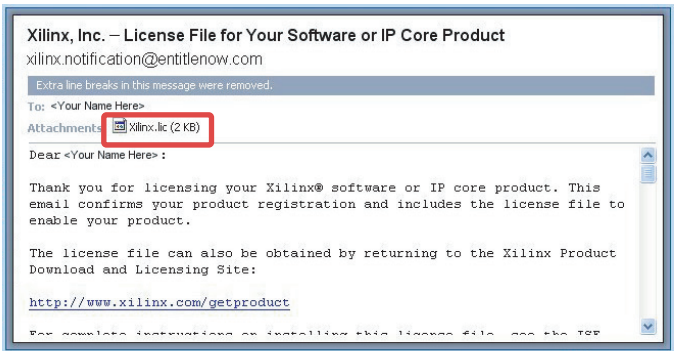
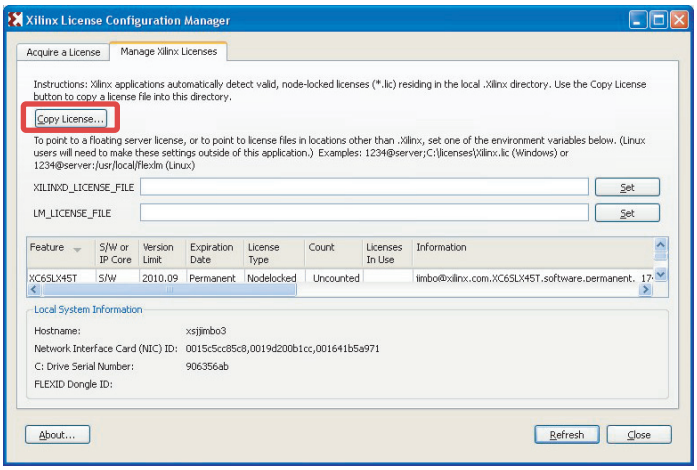


Figure 32: **Xilinx License Notification E-mail**

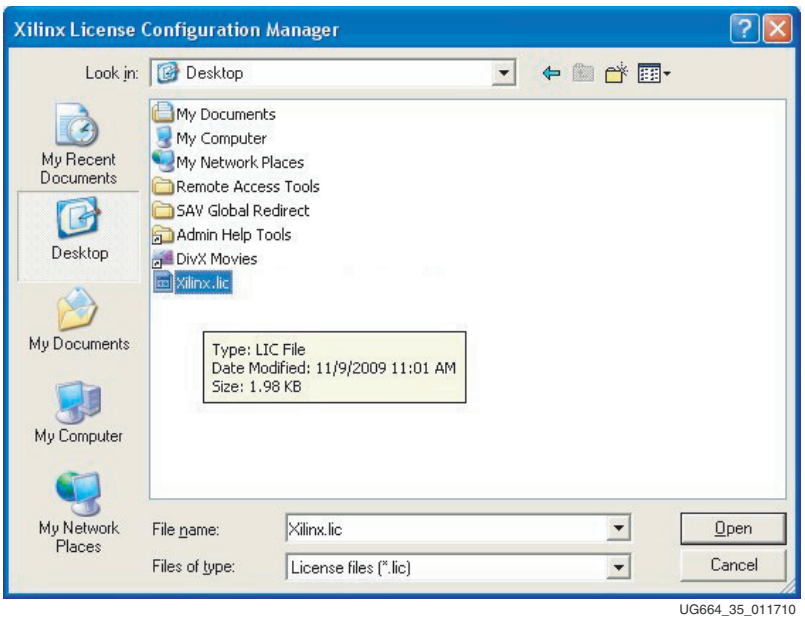
8. Start the Xilinx License Manager (**Start** → **Programs** → **ISE Design Suite** → **Manage Xilinx Licenses**) and click **Copy License** to install the license on the computer.



UG664_34_011710

Figure 33: **Manage Xilinx License Tab**

9. Navigate to the `Xilinx.lic` file location and select it (see Figure 34).



UG664_35_011710

Figure 34: **Select the `Xilinx.lic` File**

10. The ISE software license has been successfully installed. Click **OK** on the Success Dialog (Figure 35) and close the Xilinx License Configuration Manager.

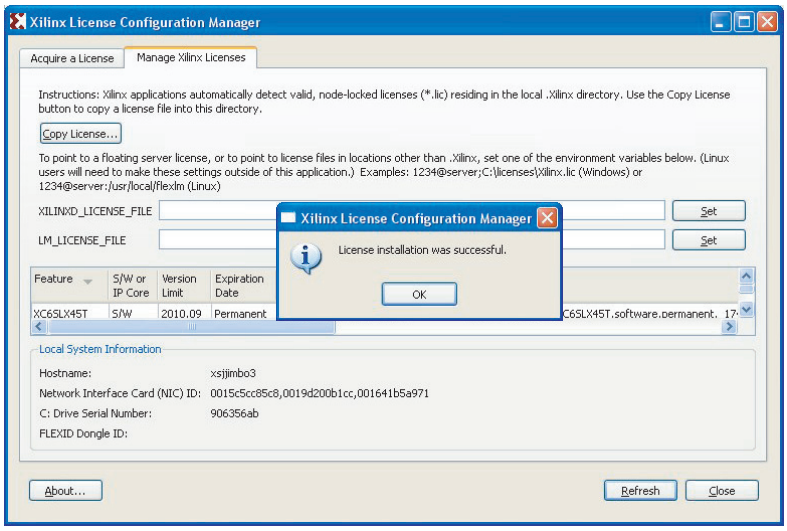


Figure 35: **License Installation Successful**

Congratulations! The ISE Design Suite tools are now installed and the licenses are set up for the Embedded Edition of the tools.

For detailed information on licensing and installation, refer to UG631, *ISE Design Suite: Installation, Licensing, and Release Notes*, located on the Xilinx documentation site at http://www.xilinx.com/support/documentation.

Now that the FPGA-based connectivity demonstration has been set up and evaluated and the ISE Design Suite Logic plus Embedded Edition is completely installed, the connectivity design for the Virtex-6 LX240T FPGA can be modified. This step enables the designer to understand the simplified flow of the Xilinx tools and design methodologies as they apply to the Virtex-6 FPGA Connectivity Kit and the Virtex-6 FPGA Connectivity Targeted Reference Design.

Modifying the Virtex-6 FPGA Targeted Reference Design

This section describes how to modify the design:

- Hardware modifications

  Note: Before running any command line scripts, refer to the "Platform Specific Instructions" section in UG631, *ISE Design Suite: Installation, Licensing, and Release Notes* (http://www.xilinx.com/support/documentation) to learn how to set the appropriate environment variables for the operating system. All scripts mentioned in this document assume the XILINX environment variables are set.

- Software modifications

Hardware Modifications

This section describes how the hardware is modified. This exercise modifies the PCI Express vendor ID.

To make RTL design changes and implement the design, follow these steps:

1. Use the PC system or laptop on which the Xilinx design tools were installed.
2. Copy the contents of the included USB stick into a local directory on this machine.
3. Make design changes:
   a. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi/design/source/` directory.
   b. Edit the `v6_pcie_10Gdma_ddr3_xaui.v` file.
   c. Search for this string: *VENDOR_ID*.
   d. Change the alphanumeric value `10EE` on this line to the vendor ID assigned to the user's company by PCI-SIG (e.g., the vendor ID for Xilinx is `10EE`). Change this value to `19AA`.
   e. Save changes and exit.
4. Generate the MIG IP core required for the Targeted Reference Design:
   a. Open a command or a terminal window.
   b. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi/design/ip_cores/mig` directory.
   c. Execute this command at the command prompt:

      ```
      $ coregen -b mig.xco -p coregen.cgp
      ```

      Wait for this command to complete before proceeding.
5. Build and implement the design:
   a. Open a terminal window.
   b. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi/design/implement/` directory.
   c. Follow the implementation flow steps depending on the operating system:
      - For Linux: Execute this command on the command line:

        ```
        $ source implement.sh x4 gen2
        ```

      - For Windows: Execute this command on the command line:

        ```
        $ implement.bat -lanemode x4gen2
        ```

d. After successful implementation of the design, a `results` folder with these FPGA programming files is generated:
   - FPGA programming bit file: `<filename>.bit` (in this case, it is `v6_pcie_10Gdma_ddr3_xaui.bit`)
   - SPI x4 flash programming MCS file: `<filename>.mcs` (in this case, it is `ML605.mcs`)

6. Program the FPGA:
   a. If the ML605 board is still plugged into the PC system, shut down the PC system and remove the ML605 board.
   b. To program the FPGA using Platform Flash, refer to the jumper settings detailed in step 4 of Hardware Demonstration Setup Instructions, page 12.
   c. For all other ML605 switch and jumper settings, keep them at the factory default configuration as indicated in UG534, *ML605 Hardware User Guide*.
7. Set up the board:
   a. Connect the mini USB cable to the USB-JTAG connector as shown in Figure 36. The other end of the USB cable is connected to the PC system or laptop on which the Xilinx design tools were installed.
   b. Power off the ML605 board (the power switch should be towards the bracket edge). Use the included wall power adapter to provide 12V power to the 6-pin connector. Power on the ML605 board.

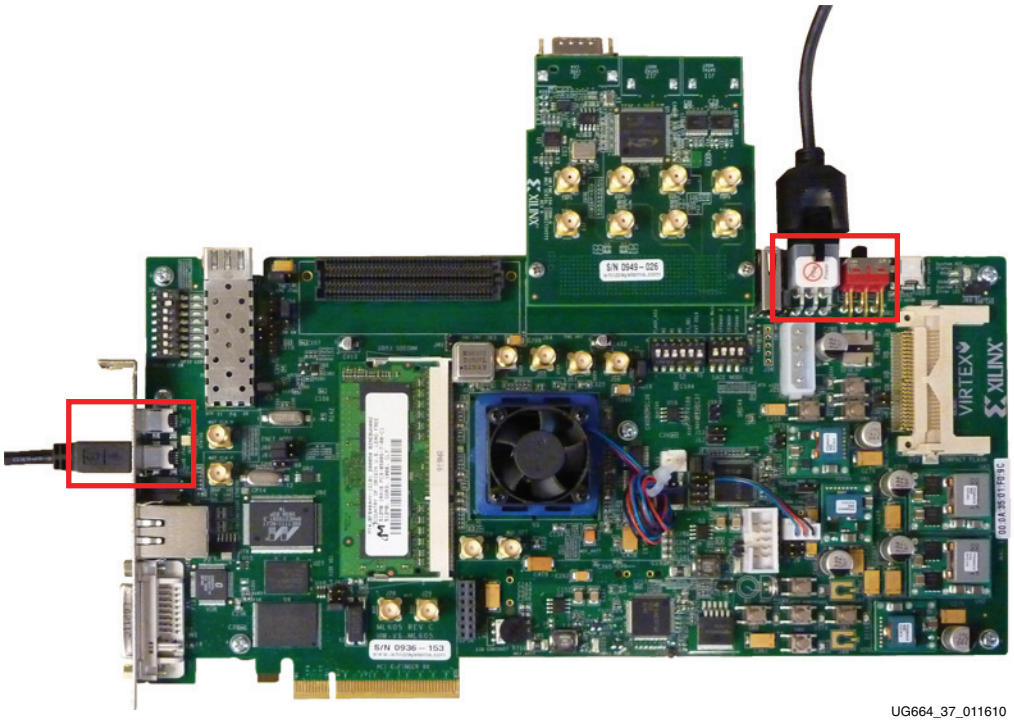


Figure 36: **Setting Up the Board**

8. Program the onboard Platform Flash:
   a. Open a terminal window.
   b. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi/design/implement/results_x4_gen2_240t` directory.
   c. Execute the FPGA programming script at the command prompt. This operation takes approximately 600 to 800 seconds to complete.
      - `$ impact -batch ml605program.cmd` (for Linux based machines)
      - `$ ml605program.bat` (for Windows based machines)
   d. After successful completion, the Programmed successfully message should appear.

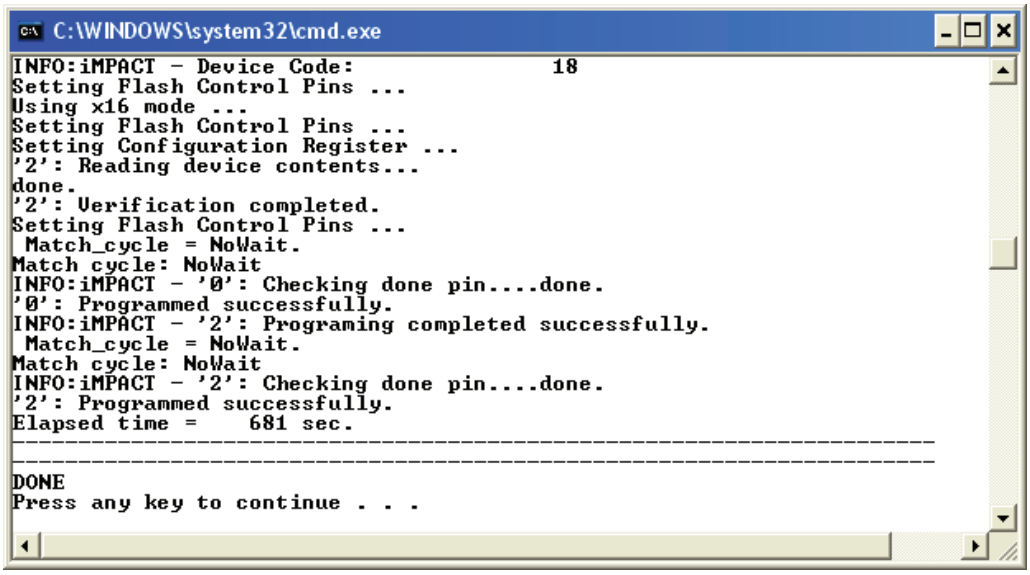


Figure 37: **Programming Was Successful**

   e. Turn off the power switch and remove the power connector.
   f. Carefully remove the mini USB cable.

The Virtex-6 FPGA Connectivity TRD is now modified and programmed into the Platform Flash and will automatically configure at power up.

Test Setup

Follow step 1 through step 6 in Hardware Demonstration Setup Instructions, page 12 to insert the board in the PC system and configure the FPGA with the design changes that were implemented. Software modifications corresponding to hardware modifications are also required to ensure that the TRD is functioning correctly.

Software Modifications

This section describes how to modify the Windows driver and Linux driver software. This exercise modifies the PCI Express vendor ID.

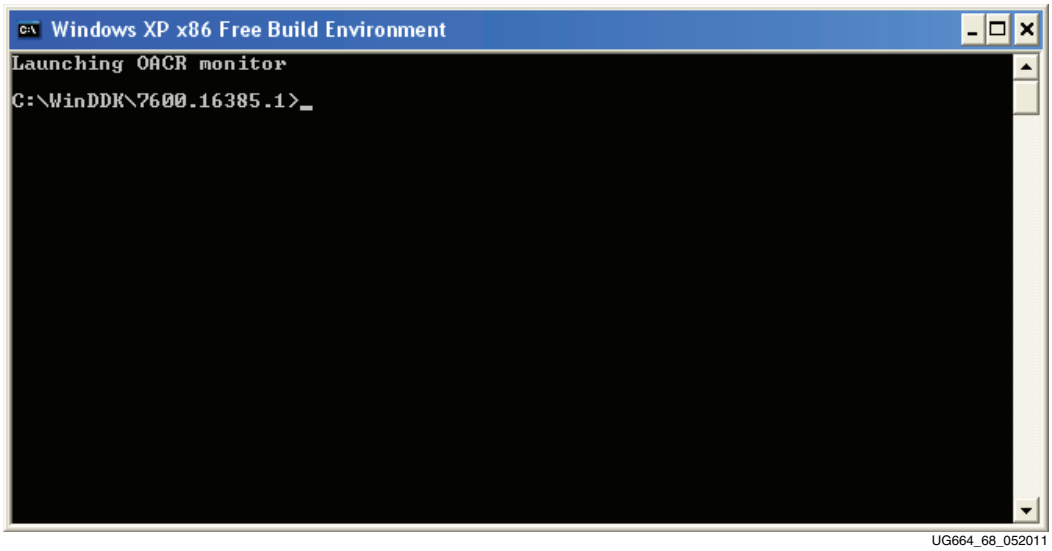
Windows Driver

To make software design changes, follow these steps:

1. Use the PC system on which the ML605 evaluation board is installed.
2. Copy the contents of the included USB stick into a local directory on this machine:
   a. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi/windows_driver/xdma/` directory.
   b. Edit the `xdma.inx` file.
   c. Search for this string: *xdma_Inst*. Add the following entry for Windows XP:

      ```
      %xdma.DRVDESC%= xdma_Inst, PCI\VEN_19aa&DEV_6042
      ```

   d. With this change, the driver also supports a device with Vendor ID 19AA and Device ID 6042. Save the changes and exit.
3. Recompile the drivers:
   a. Windows driver compilation is done using Windows Device Driver Kit (WDK). Prerequisite for this step is WDK installation. Download and install WDK following the instructions available at: http://www.microsoft.com/whdc/devtools/wdk/wdkpkg.mspx.
   b. After WDK installation, invoke the free build environment from **Start → All Programs → Windows Driver Kit → WDK <version> → Build Environments → [Windows XP]**. This invokes the following window (Figure 38).



UG664_68_052011

Figure 38: **Win WDK Build Environment**

   c. Compile DMA code: Navigate to the `windows_driver/xdma` directory and execute this command:

      ```
      build /w  -ceZ
      ```

This command invokes the Microsoft make routines to build the driver components (Figure 39).

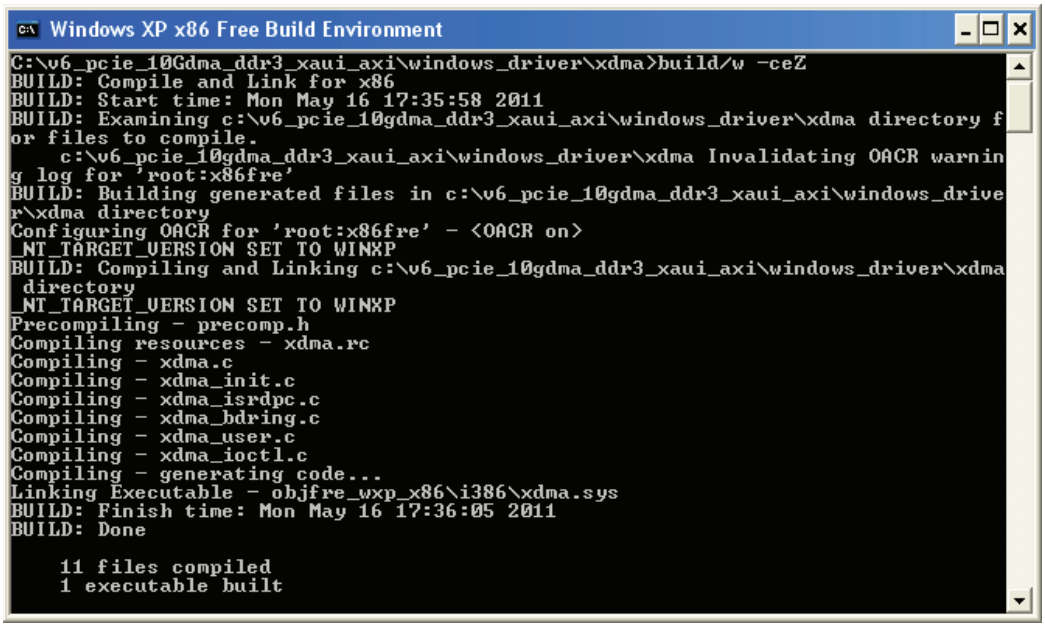


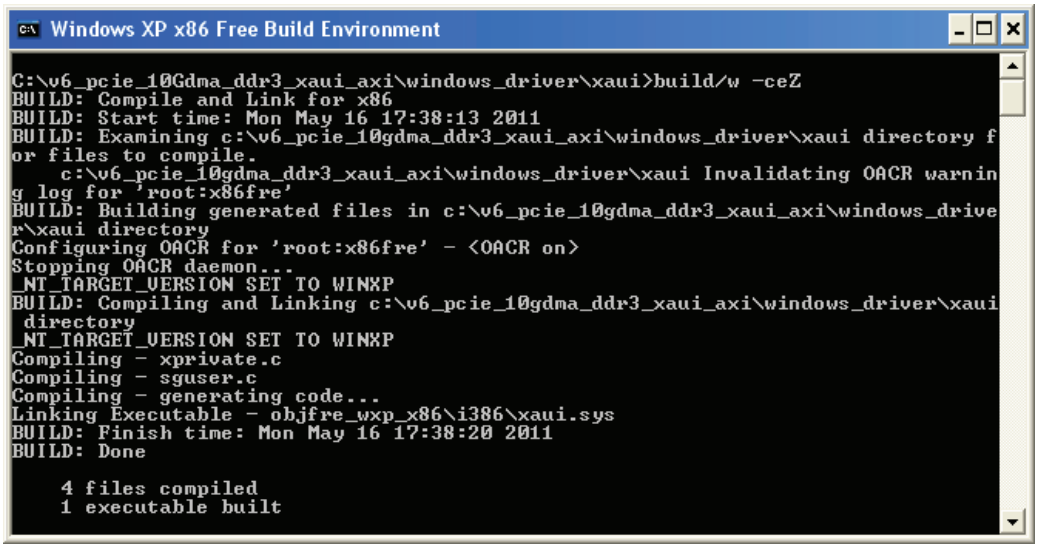
Figure 39: **DMA Driver Compilation**

On successful completion of the build, the driver file (`xdma.sys`) is available under [`objfre_wxp_x86`|`objchk_wxp_x86`]`\i386` depending on the build environment selected. The Setup Information file (`xdma.inf`) is also available in the same directory.

d. Compile XAUI path code: Navigate to the `windows_driver/xaui` directory and execute this command:

```
build /w  -ceZ
```

This command invokes the Microsoft make routines to build the driver components (Figure 40).



UG664_70_052011

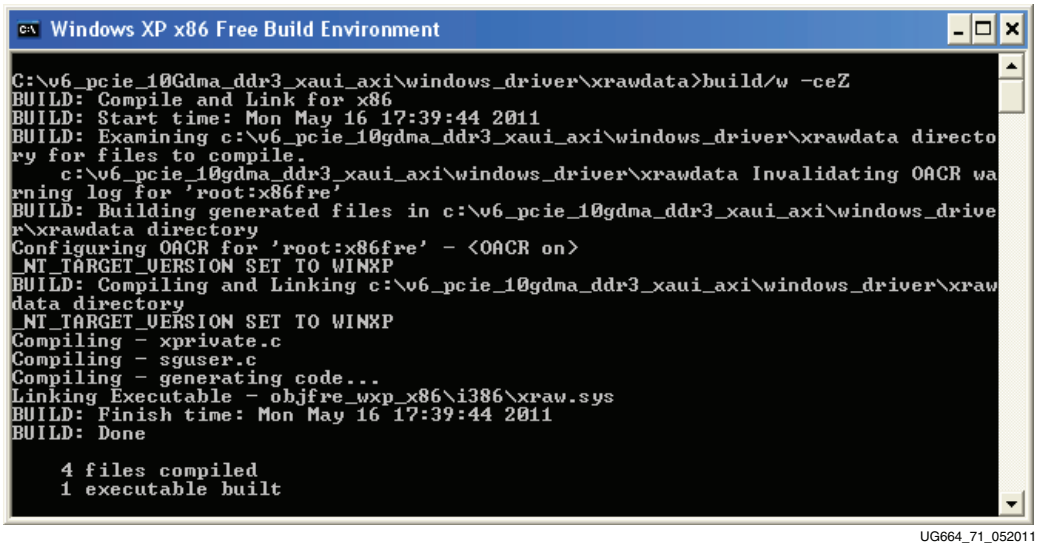
Figure 40: **XAUI Driver Compilation**

On successful completion of the build, the driver file (`xaui.sys`) is available under [`objfre_wxp_x86`|`objchk_wxp_x86`]`\i386` depending on the build environment selected. The Setup Information file (`xaui.inf`) is also available in the same directory.

e. Compile Raw Data path code: Navigate to the `windows_driver/xraw` directory and execute the following

```
build /w  -ceZ
```

This command invokes the Microsoft make routines to build the driver components (Figure 41).



UG664_71_052011

Figure 41: **Raw Data Driver Compilation**

On successful completion of the build, the driver file (`xraw.sys`) is available under [`objfre_wxp_x86`|`objchk_wxp_x86`]`\i386` depending on the build

environment selected. The Setup Information file (`xraw.inf`) is also available in the same directory.

Note: The XAUI and Raw Data drivers are not required to be recompiled because only the Vendor ID was changed. The step d, page 46 and step e, page 47 are shown if other software sections need to be modified.

4. Create the recompiled driver package:

   After all the drivers are compiled, create a folder called `compiled_drivers`. In this folder, do the following:

   a. Make a folder called `xdma` and populate it with `xdma.sys` (type - System File) and `xdma.inf` (type - Setup Information) from the DMA driver compiled area.

   b. Make a folder called `xaui` and populate it with `xaui.sys` (type - System File) and `xaui.inf` (type - Setup Information) from the XAUI driver compiled area.

   c. Make a folder called `xraw` and populate it with `xraw.sys` (type - System File) and `xraw.inf` (type - Setup Information) from the Raw Data driver compiled area.

   d. Copy the GUI executable from the `v6_pcie_10Gdma_ddr3_xaui_axi/windows_driver/xpmon` folder.

   e. Copy `WdfCoInstaller01009.dll` from the `v6_pcie_10Gdma_ddr3_xaui_axi/windows_driver/utils` folder and place it under the `xdma`, `xaui`, and `xraw` folders.

   The folder hierarchy appears as shown in Figure 42.

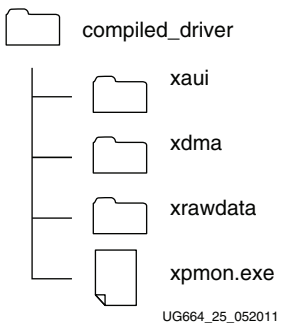


Figure 42: **Compiled Driver Directory Organization**

5. Uninstall previous drivers:

   If previous drivers are installed in the system, they need to be uninstalled. To uninstall previous drivers, run `x_v6_trd_setup.exe`. InstallShield asks for confirmation of driver install. Click **OK** (Figure 43).

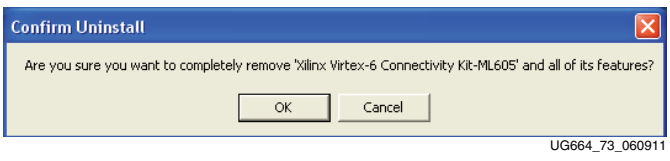


Figure 43: **Confirm Driver Uninstall**

InstallShield successfully uninstalls the previous driver for the Virtex-6 FPGA Connectivity TRD. Click **Finish** to close the InstallShield Wizard (Figure 44).

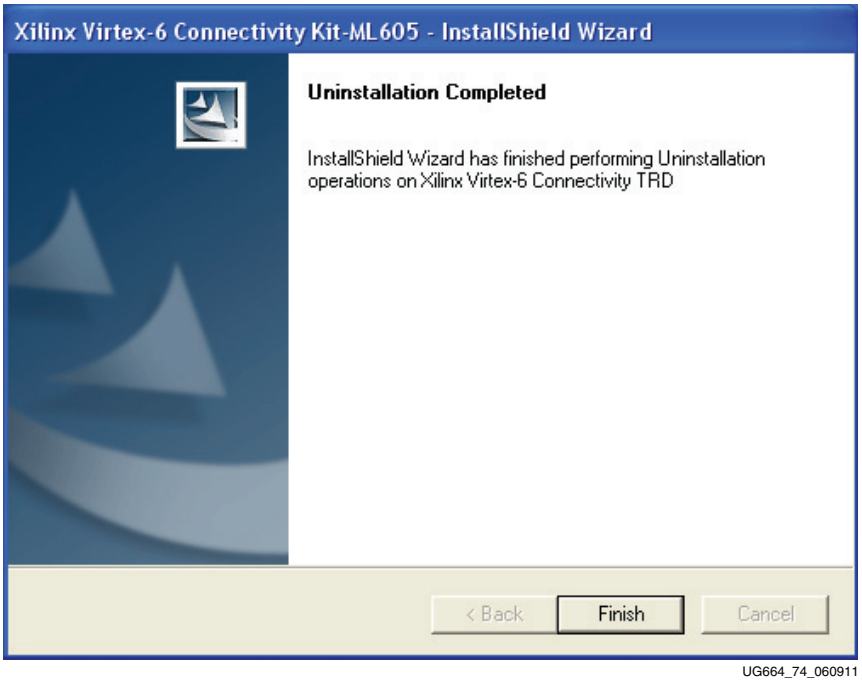


Figure 44: **InstallShield Uninstalls Drivers**

6. Load recompiled drivers:

   To install the recompiled drivers, run `x_v6_trd_setup.exe` again.

   The InstallShield Wizard for the Virtex-6 FPGA Connectivity TRD is launched. Click on **Next** until the InstallShield Wizard completes, and then click on **Finish**. At the end

of this install process, the driver and GUI files are copied to the `C:\Program Files\Xilinx Inc\Virtex6` folder (Figure 45).

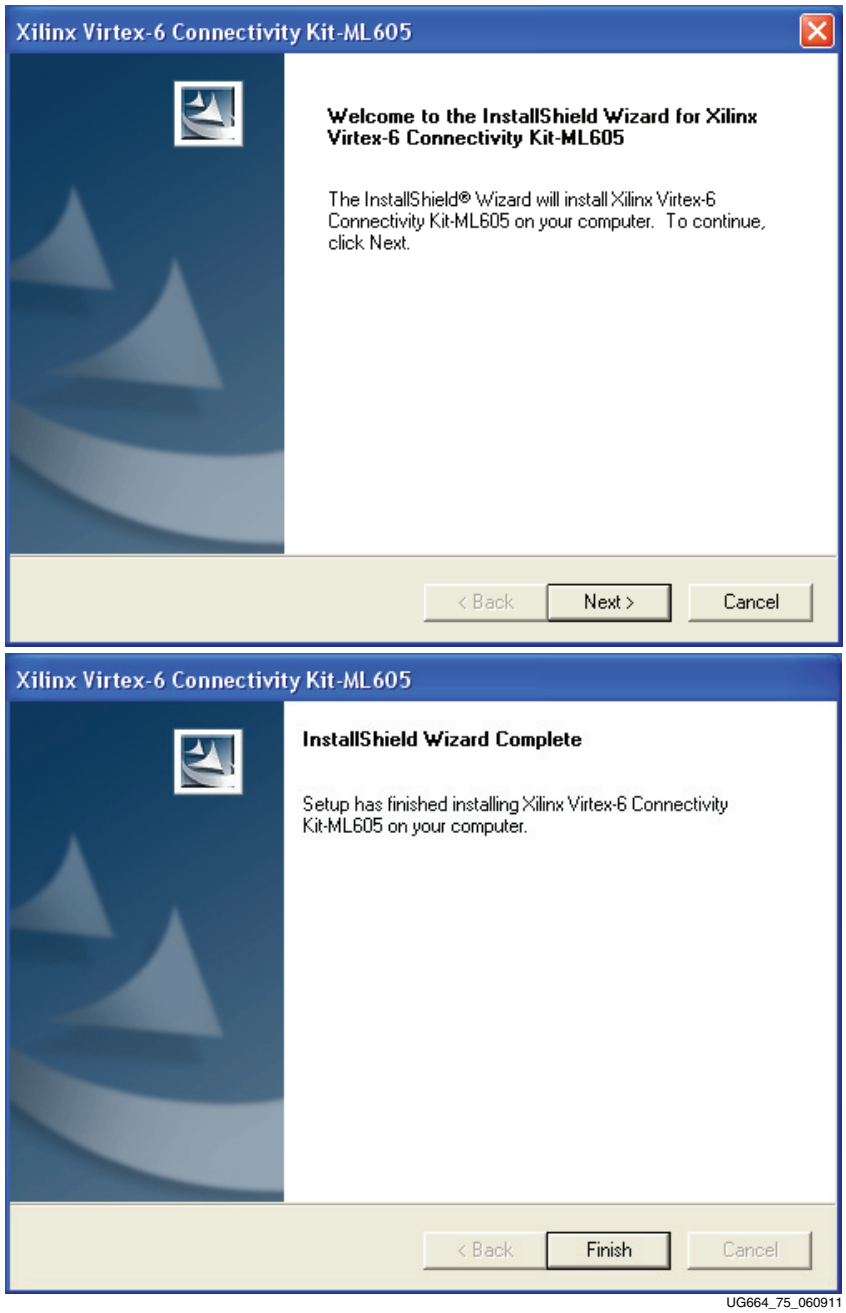


Figure 45: **InstallShield Wizard Copies GUI and Driver Files into Program Files**

After InstallShield Wizard completes, Add Hardware Wizard is launched (Figure 46). Click on **Next**.

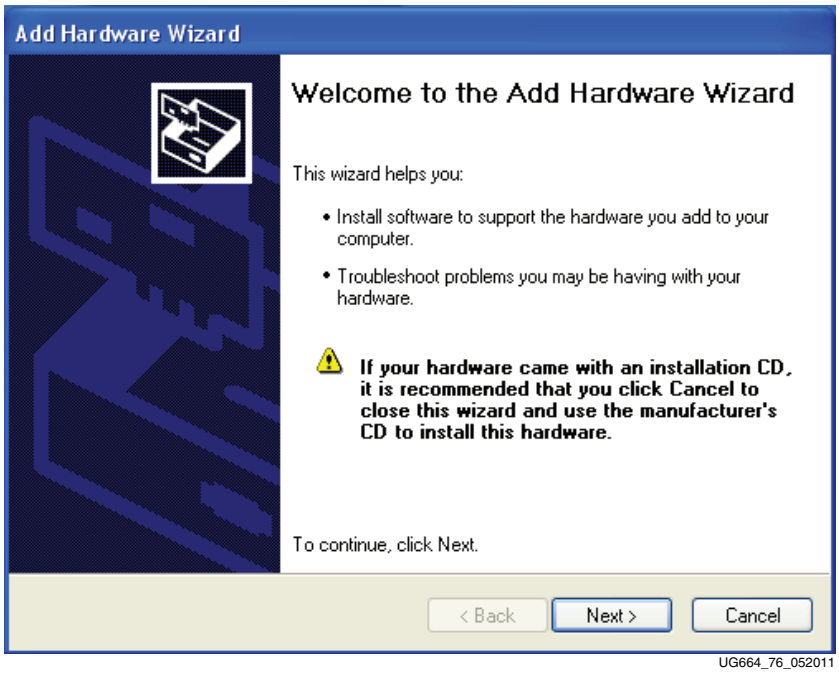


Figure 46: **Launch Add Hardware Wizard**

The xdma driver and child drivers xrawdata and xaui are loaded manually through the Found New Hardware Wizard.

To load the recompiled xdma driver, select **Install from a list or specific location** and click **Next**. Browse and choose `v6_pcie_10Gdma_ddr3_xaui_axi\compiled_drivers\xdma`, and click **Next**. The Xilinx DMA driver is installed and associated with the Connectivity TRD hardware. Click **Finish** to load the next driver (Figure 47).

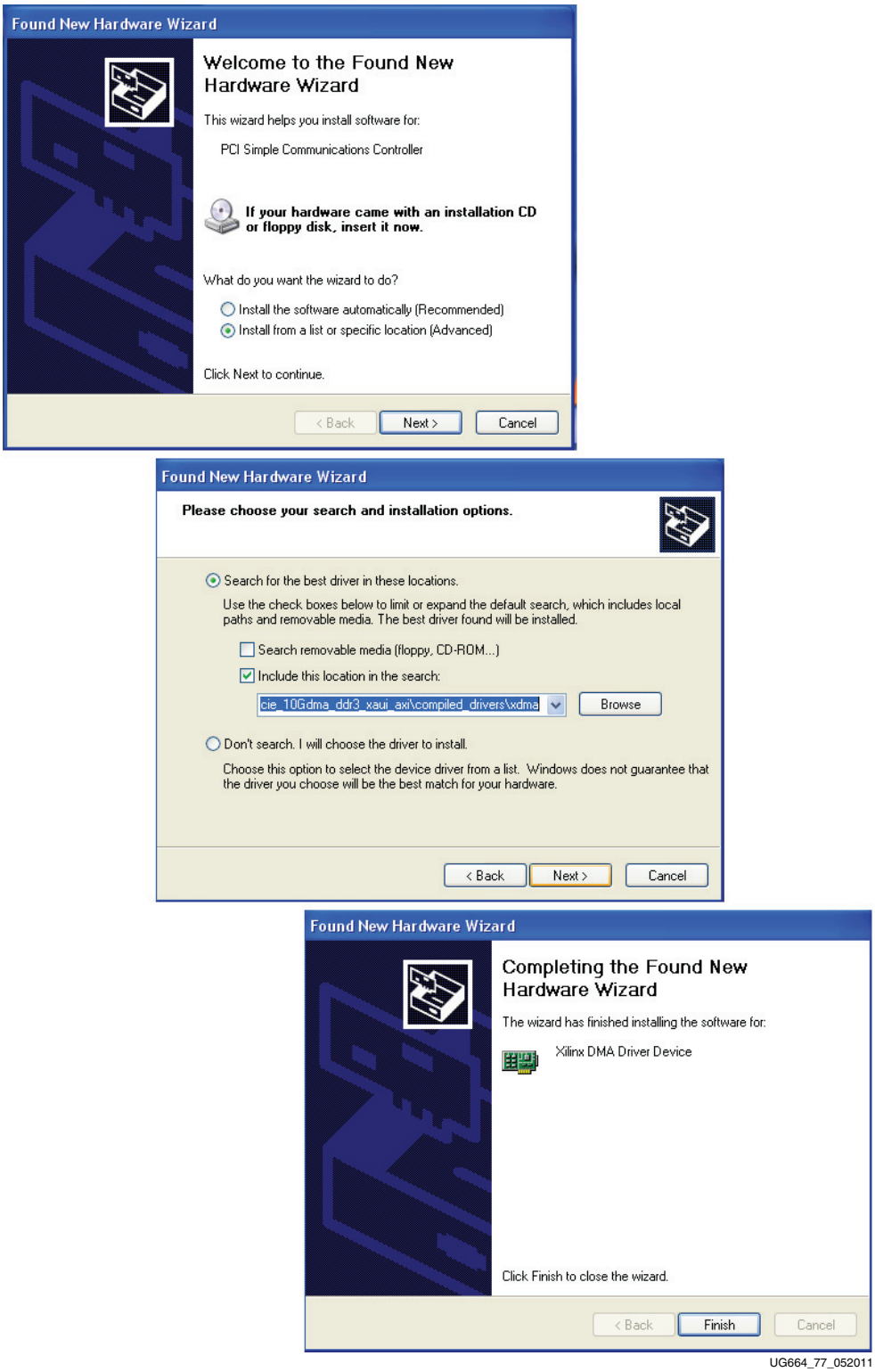


Figure 47: **Load Xilinx DMA Driver**

To load the recompiled xrawdata driver, select **Install from a list or specific location**, and click **Next**. Browse and choose

`v6_pcie_10Gdma_ddr3_xaui_axi\compiled_drivers\xrawdata`, and click **Next**. The Xilinx Raw Data driver is installed. Click **Finish** to load the next driver (Figure 48).

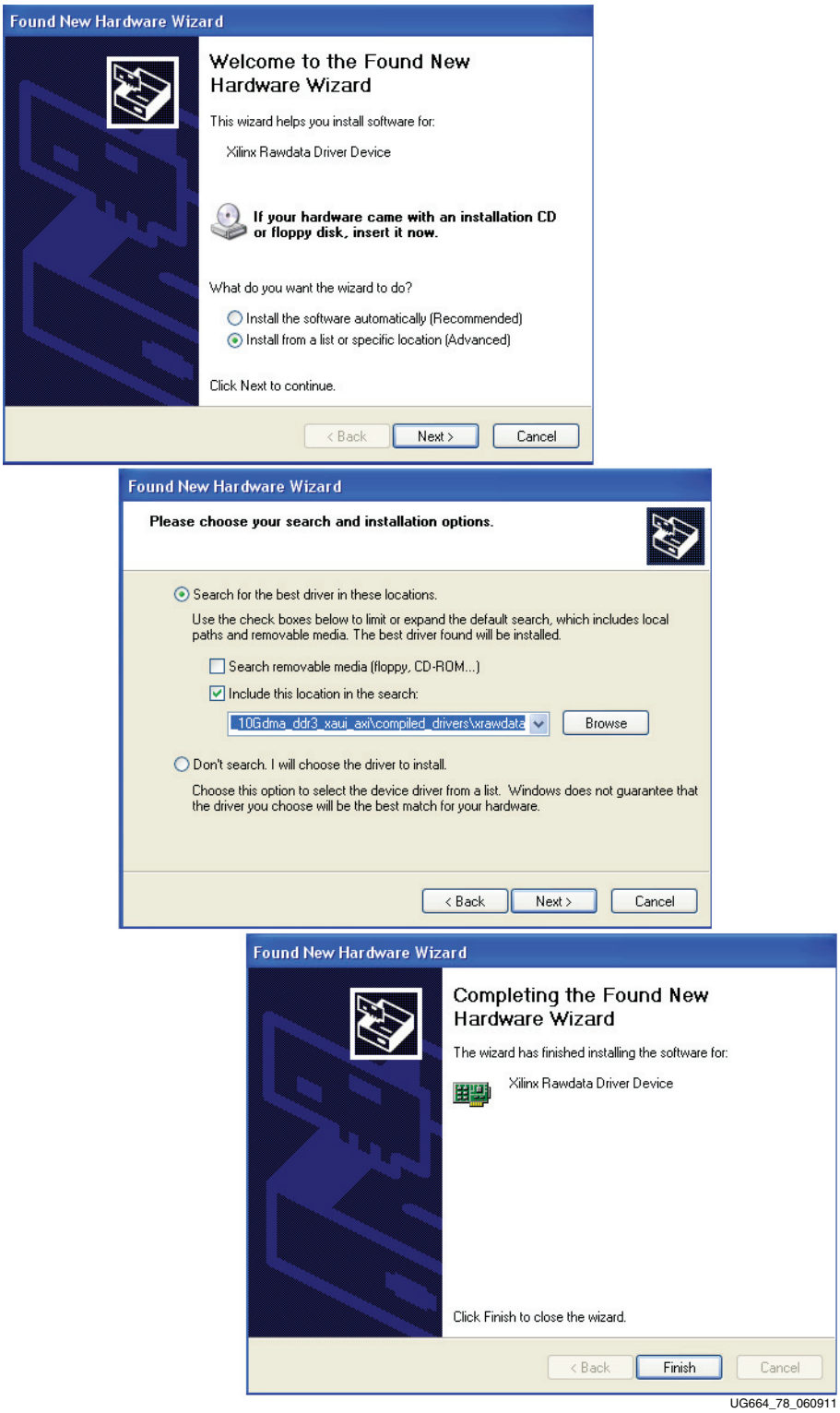


Figure 48: **Load Xilinx Raw Data Driver**

To load the recompiled xaui driver, select **Install from a list or specific location**, and click **Next**. Browse and choose `v6_pcie_10Gdma_ddr3_xaui_axi\compiled_drivers\xaui`, and click **Next**. The Xilinx XAUI driver is installed. Click **Finish** (Figure 49).

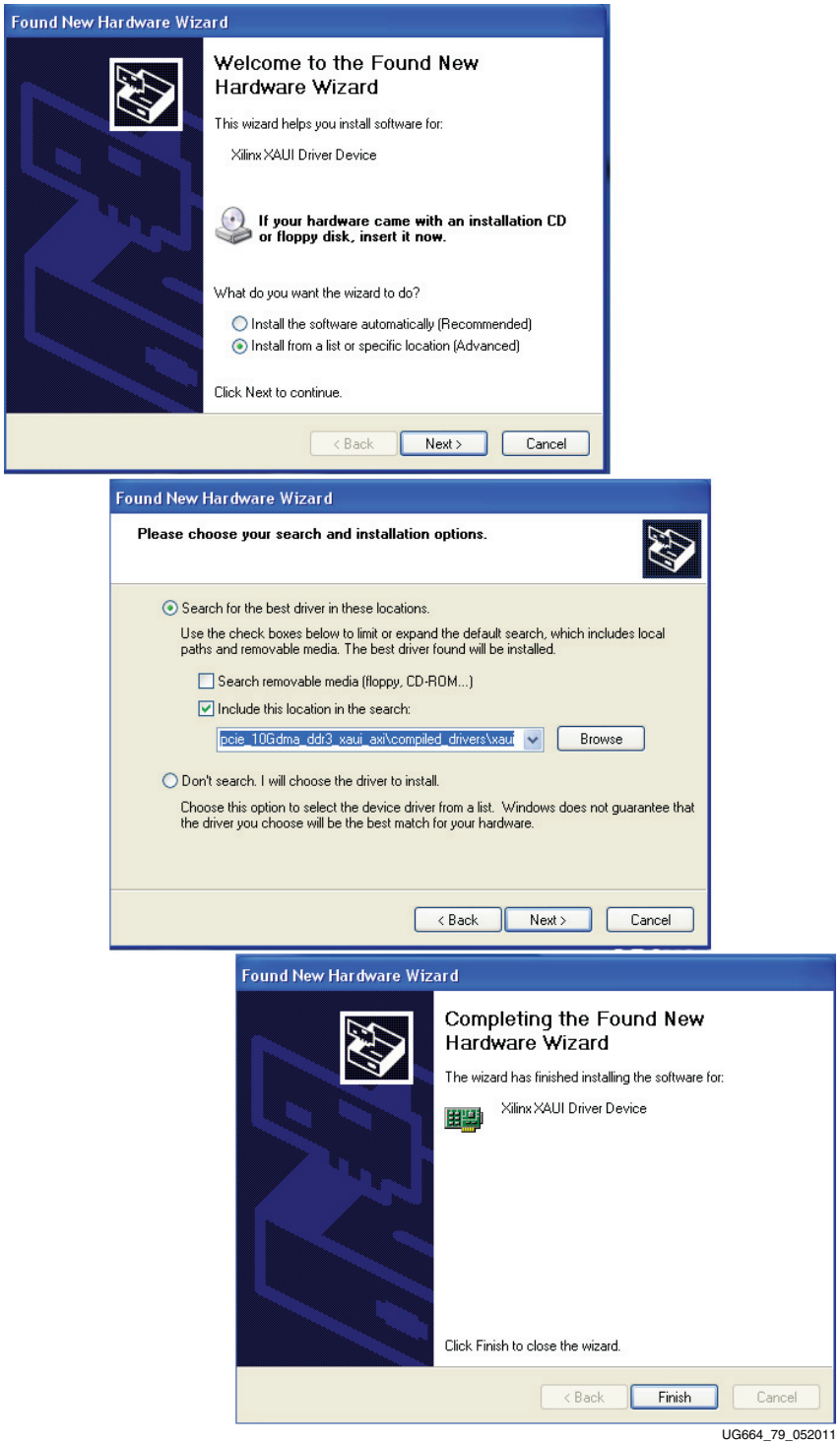


Figure 49: **Load Xilinx XAUI Driver**

All the drivers required to run the Virtex-6 FPGA Connectivity TRD are found. Click **Finish** to exit the Add Hardware Wizard (Figure 50).

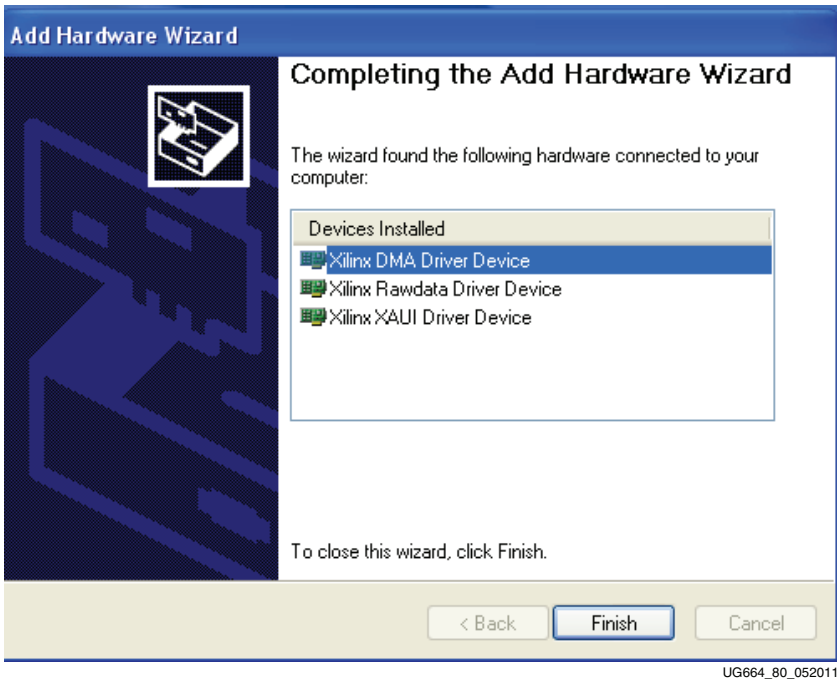


Figure 50: **All Virtex-6 FPGA Connectivity TRD Drivers are Installed**

7. Launch GUI:

   Double-click on the xpmon icon available on the desktop to launch the Performance Monitor application.

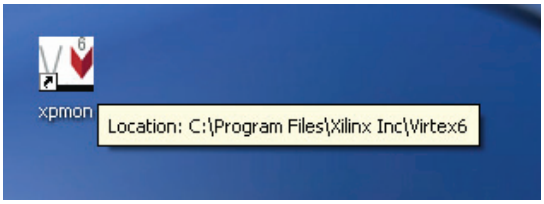


Figure 51: **Launch GUI**

8. Follow step 16, page 30 through step 18, page 32 in Hardware Demonstration Setup Instructions to completely verify the modified settings.
9. Follow step 1, page 33 through step 4, page 36 in Evaluating the Virtex-6 FPGA Connectivity TRD to evaluate the performance for the modified design.
10. Click **System Status** to review:
    a. PCIe link status, Vendor ID, and Device ID information.
    b. The vendor ID displayed on this screen should be equal to `19AA`, corresponding to the hardware change that was performed.

Linux Driver

To make software design changes, follow these steps:

1. Use the PC system on which the ML605 evaluation board is installed.
2. Copy the contents of the included USB stick into a local directory on this machine:
   a. Navigate to the `v6_pcie_10Gdma_ddr3_xaui_axi/linux_driver/xdma/` directory.
   b. Edit the `xdma_base.c` file.
   c. Search for this string: *#define PCI_VENDOR_ID_DMA*.
      - Change the alphanumeric value `10EE` found on this line, with the vendor ID assigned to the user's company by PCI-SIG (e.g., the vendor ID for Xilinx is `10EE`). Change this value to `19AA`.
      - Save the changes and exit.
3. Load the driver and launch the Performance Monitor application:
   a. Navigate to the `v6_pcie_dma_ddr3_xaui_axi` folder.
   b. Double-click **v6_trd_lin_quickstart**.

      This step builds kernel objects, loads the device driver, and launches the Performance Monitor application.
   c. Click **Run in Terminal** to proceed.
4. Follow step 16, page 30 through step 18, page 32 in Hardware Demonstration Setup Instructions to completely verify the modified settings.
5. Follow step 1, page 33 through step 4, page 36 in Evaluating the Virtex-6 FPGA Connectivity TRD to evaluate the performance for the modified design.
6. Review the system status.

   Click **System Status** to review:

   - PCIe link status, Vendor ID, and Device ID information.
   - The vendor ID displayed on this screen should be equal to `19AA`, corresponding to the hardware change that was performed.

Congratulations! The Virtex-6 FPGA Connectivity Kit using the connectivity TRD has been fully set up, and the system performance has been evaluated. The Xilinx design flow has been reviewed for modifying the connectivity TRD. This design includes the built-in integrated block for PCI Express (4-lane, 5 GT/s configuration for PCI Express v2.0), XAUI LogiCORE IP, a Virtual FIFO memory controller designed to interface to the onboard DDR3 SODIMM device, and a third-party DMA controller for PCI Express.

Next Steps

Connectivity TRD Modules

This section outlines the correlation between the design modules and corresponding design source files for the various blocks of the design. Refer to Figure 1, page 11 for the detailed block diagram of the Virtex-6 FPGA Connectivity TRD. Table 1 shows the design file organization per module.

Table 1: **Design File Organization for the Virtex-6 FPGA Connectivity TRD**

Module Name	Source Files/Directories	LogiCORE IP	Connectivity TRD Source
Top-Level Module: Virtex-6 FPGA Connectivity TRD	`v6_pcie_10Gdma_ddr3_xaui_axi`		✓
PCI Express (x4)	`pcie`	✓ (Endpoint for PCI Express core with AXI4-Stream interface - CORE generator output) OR (Endpoint for PCI Express core with transaction interface - CORE generator output)	
Packet DMA	`dma`		Netlist deliverable only (from Northwest Logic)
Multiport Virtual FIFO	`virtual_fifo`		✓
Memory Controller Block	`mig`	✓ (MIG - CORE Generator output)	
XAUI	`xaui`	✓ (XAUI core - CORE Generator output)	
Clocking, Reset, Register Interface	`reset_control`, `registers`		✓
Software Device Driver	`linux_driver`, `windows_driver`		✓
Software Application/GUI	`linux_driver/xpmon`, `windows_driver/xpmon`		✓

For functional details on these modules, refer to the "Functional Description" chapter in UG379, *Virtex-6 FPGA Connectivity Targeted Reference Design with AXI4 Protocol User Guide*.

PCI Express

Figure 52 shows the design module for PCI Express. Figure 53 shows the design file structure.

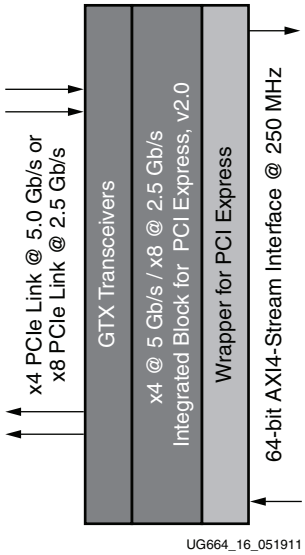


Figure 52: **Design Module for PCI Express**

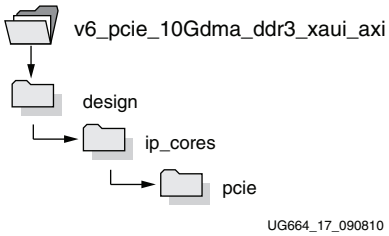


Figure 53: **Design FIles for PCI Express**

Packet DMA

Figure 54 shows the design module for Packet DMA. Figure 55 shows the design file structure.

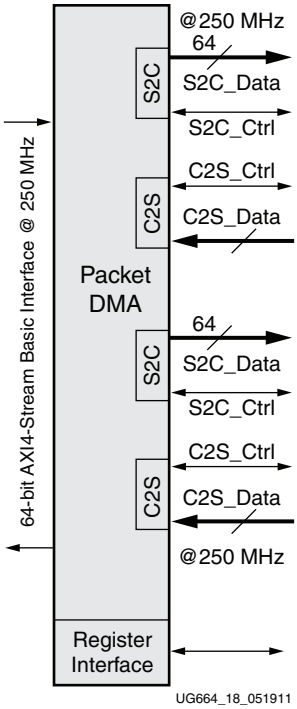


Figure 54: **Packet DMA Design Module**

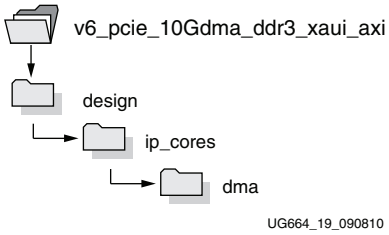


Figure 55: **Packet DMA Design FIles**

Multiport Virtual FIFO and Memory Controller Block

Figure 56 shows the design module for the multiport virtual FIFO and memory controller block. Figure 57 shows the design file structure.

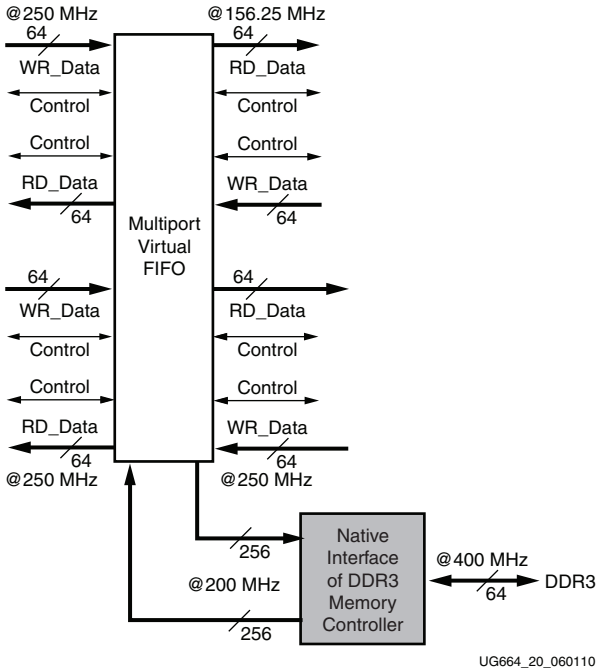


Figure 56: **Multiport Virtual FIFO and Memory Controller Block Design Module**

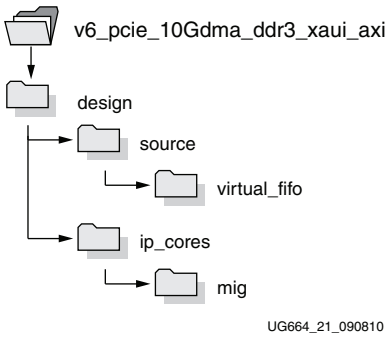


Figure 57: **Multiport Virtual FIFO and Memory Controller Design FIles**

XAUI

Figure 58 shows the XAUI design module. Figure 59 shows the design file structure.

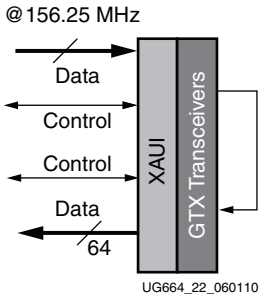


Figure 58: **XAUI Design Module**

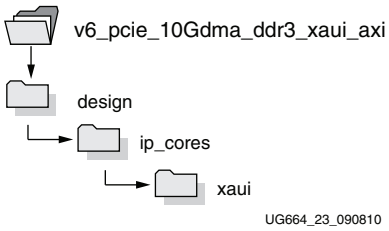


Figure 59: **XAUI Design FIles**

Software Device Driver and Software Application/GUI Files and Scripts

Figure 60 shows the design module for the software device driver and the software application/GUI files and scripts. Figure 61 shows the design file structure for the software device driver and the software application and GUI.

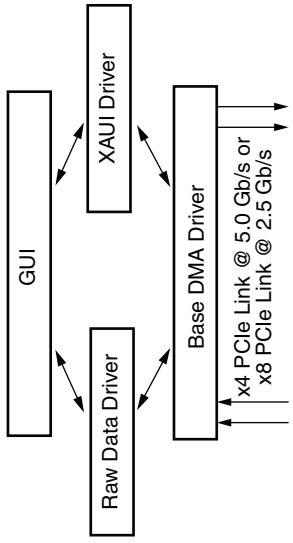


Figure 60: **Software Device Driver and Software Application/GUI Design Module**

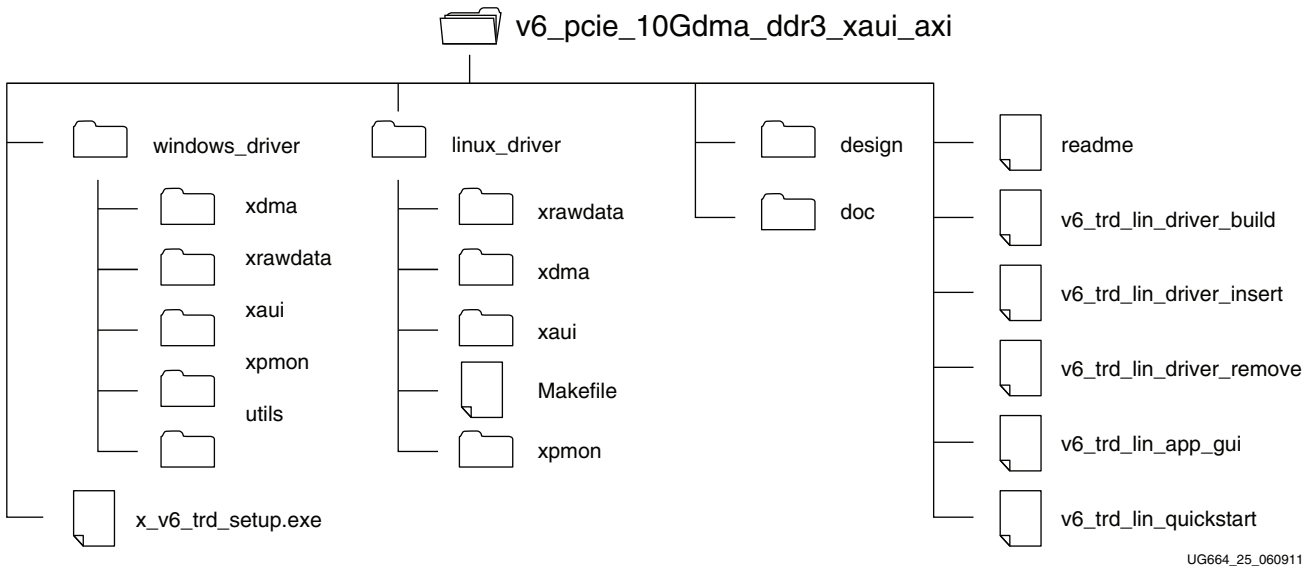


Figure 61: **Software Device Driver and Software Application/GUI Files**

Simulating the Connectivity TRD

A complete simulation environment is provided with the Virtex-6 FPGA Connectivity TRD. For more details on the simulation environment and the associated simulation files, refer to the "Simulation" section in the "Getting Started" chapter in UG379, *Virtex-6 FPGA Connectivity Targeted Reference Design with AXI4 Protocol User Guide*.

Reusing the DMA IP from Northwest Logic

The Packet DMA Controller included in the Virtex-6 FPGA Connectivity Kit is an evaluation version of the Northwest Logic PCIe Packet DMA IP Core. This 64-bit DMA IP core is optimized for the Virtex-6 FPGA architecture. The DMA design deliverables are:

- Simulation model
- Hardware evaluation netlist (time-limited to 12 hours)

Orders for the full production version of the Northwest Logic PCIe Packet DMA IP core can be placed at http://www.nwlogic.com/packetdma.

Modifications to the Connectivity TRD

The Virtex-6 FPGA Connectivity TRD is a framework for system designers to derive extensions or modify their designs. Additional possible design enhancements, modifications, and reconstructions with custom IPs and design blocks are described in the "Designing with the TRD Platform" chapter in UG379, *Virtex-6 FPGA Connectivity Targeted Reference Design with AXI4 Protocol User Guide*.

Getting Started with the Virtex-6 FPGA IBERT Reference Design

This Virtex-6 FPGA Connectivity Kit comes with an Integrated Bit Error Ratio Test (IBERT) reference design available on the CompactFlash. The demonstration shows the capabilities of the Virtex-6 LXT device using the GTX transceivers running at 3.125 Gb/s line rates. The GTX transceivers can successfully operate at line rates from 750 Mb/s to 6.6 Gb/s.

The Virtex-6 FPGA IBERT reference design has these components:

- Virtex-6 FPGA GTX transceivers running at 3.125 Gb/s
- The IBERT v2.0 reference design available through the CORE Generator tool for IP delivery

  The design also includes a pseudo-random bit sequence (PRBS) pattern generator and checker.
- Four GTX transceivers in the Virtex-6 LX240T FPGA are accessed through these channels in the IBERT reference design:
  - SMA (two channels)
  - SATA (two channels)

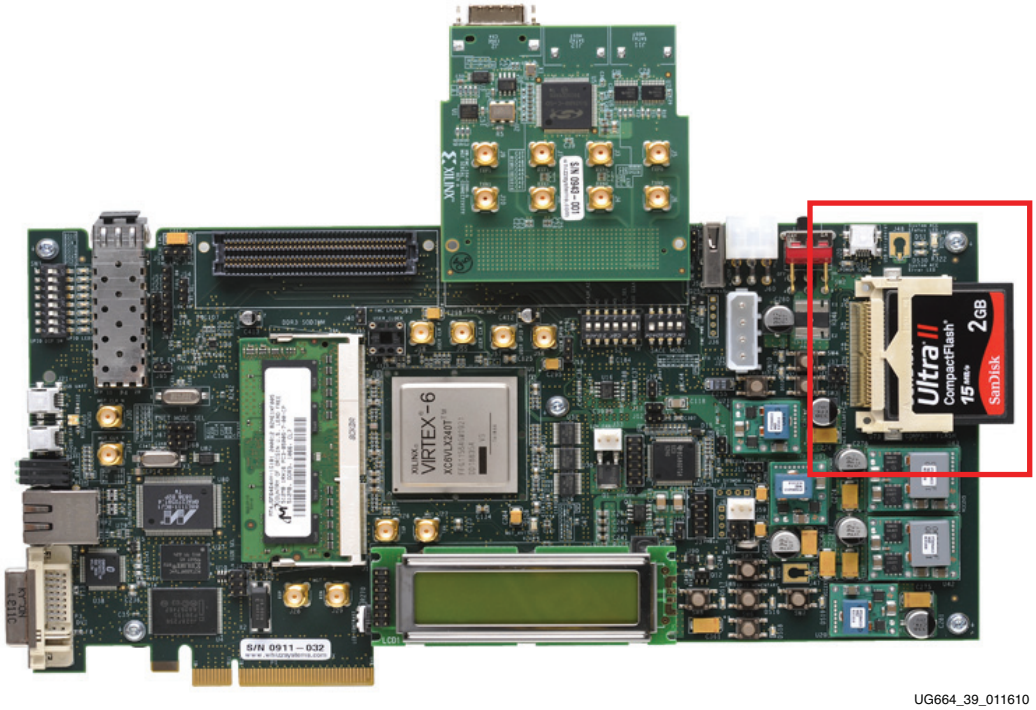
Note: The demonstration is for SMA and SATA external loopback scenarios only.

IBERT Hardware Demonstration Setup Instructions

This section describes how to set up the hardware for the IBERT reference design demonstration. The IBERT reference design is provided as an FPGA programming file on the CompactFlash.

1. This equipment is needed to run the demonstration:
   - Virtex-6 FPGA Connectivity Kit
   - PC system with USB port
   - Monitor, keyboard, and mouse
   - ISE Design Suite installed on the PC system

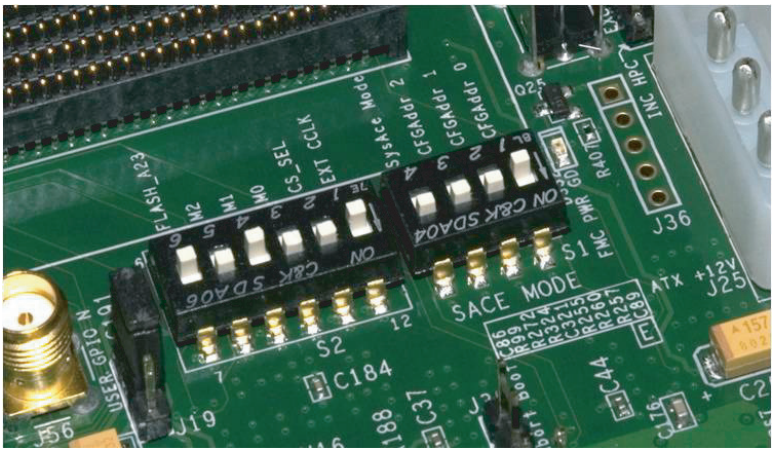
2. Board Setup I – Install the CompactFlash on the ML605 board:
   Use the CompactFlash provided in the kit (see Figure 62).



UG664_39_011610

Figure 62: **Installing the Included CompactFlash on the ML605 Board**

3. Board Setup II – Configure the settings for DIP switches S1 and S2 to load the IBERT design from the CompactFlash (see Figure 63), where `X` = Don't care, `1` = ON, 0 = OFF):
   a. Set S1 to `1110` (Position 4 is the most-significant bit, and Position 1 is the least-significant bit).
   b. Set S2 to `0101XX` (Position 6 is the most-significant bit, and Position 1 is the least-significant bit).



UG664_40_022210

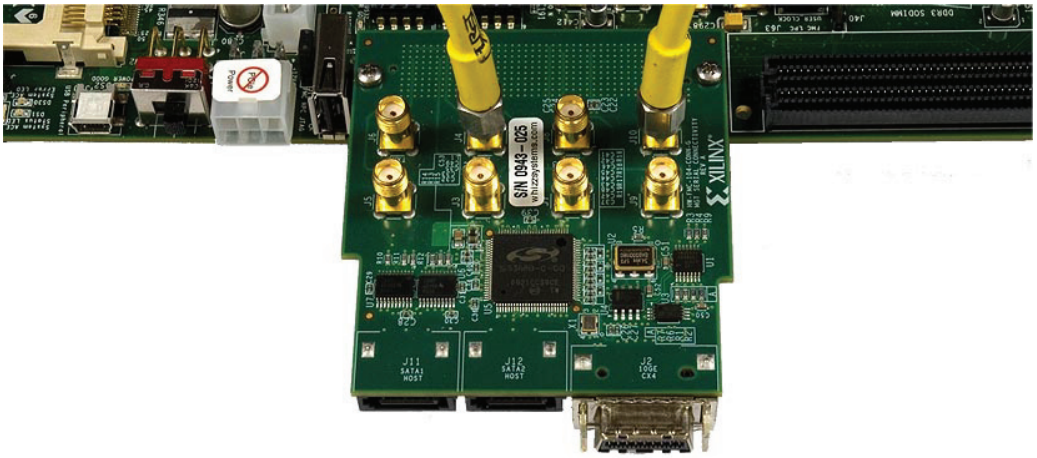
Figure 63: **Configuring the FPGA with the IBERT Design from CompactFlash**

4. Board Setup III – Connect a USB cable to the ML605 board as shown in Figure 64:
   a. Connect the included USB Type-A to Mini-B cable to the USB JTAG connector on the ML605 board.
   b. Connect the other end of this cable to the PC system.

UG664_41_011610

Figure 64: **Connecting the USB Cable to the USB-JTAG Connector of the ML605 Board**

5. Board Setup IV – Use the SMA cables to loop back the transceiver channel pinned to the SMA on the FMC card:
   a. Connect J4 to J10 (see Figure 65).



UG664_42_021810

Figure 65: **Configuring the SMA Transceiver Channel with External Loopback - I**

b. Connect J6 to J8 (see Figure 66).

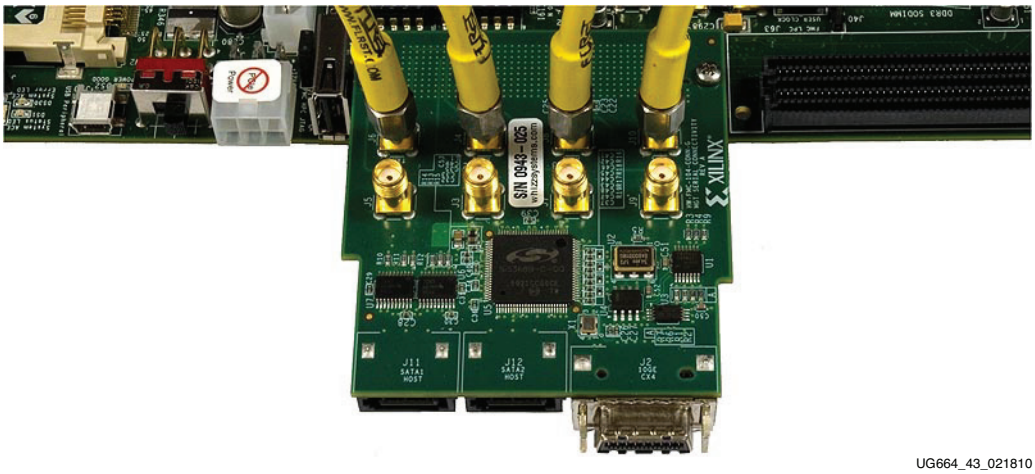


Figure 66: **Configuring the SMA Transceiver Channel with External Loopback - II**

c. Connect J3 to J9 (see Figure 67).

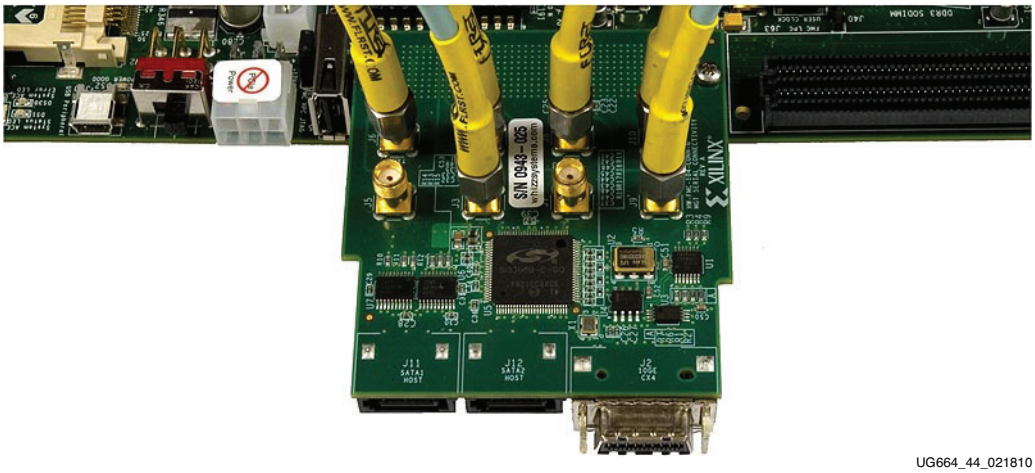
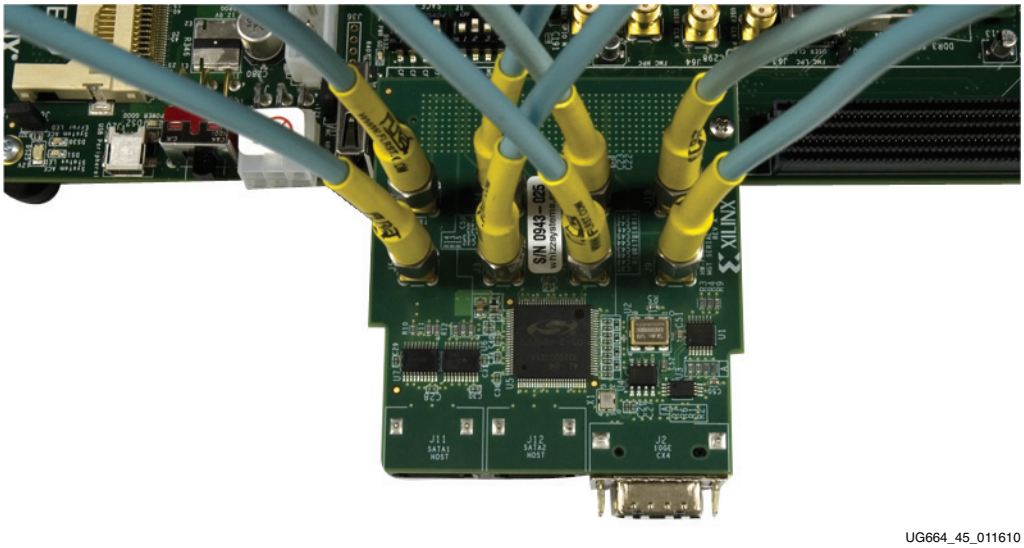


Figure 67: **Configuring the SMA Transceiver Channel with External Loopback - III**

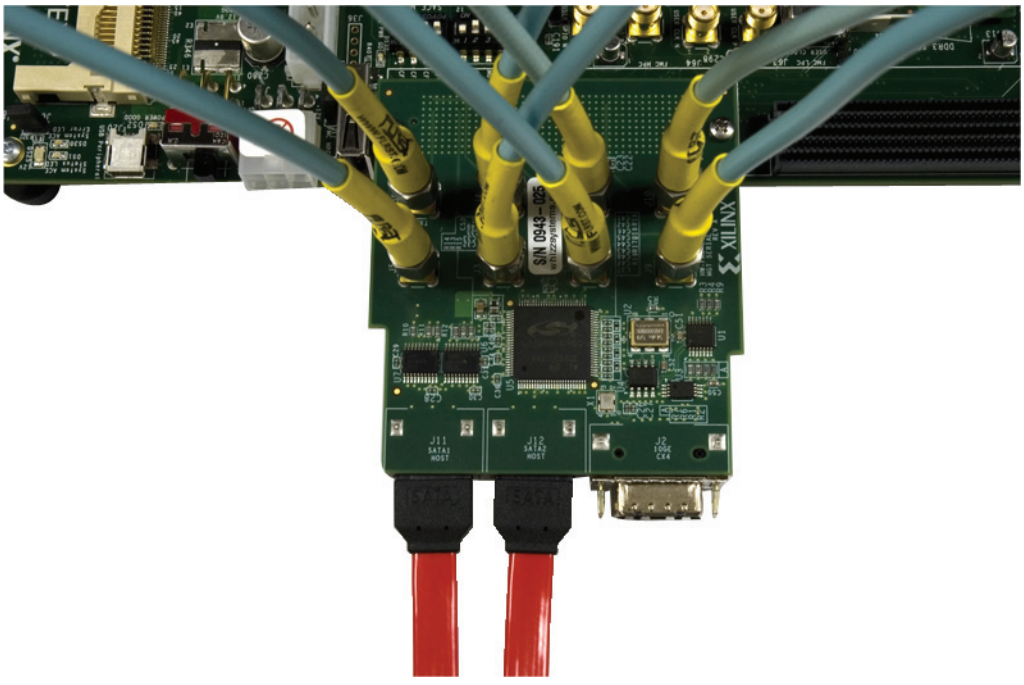
d. Connect J5 to J7 (see Figure 68).



UG664_45_011610

Figure 68: **Configuring the SMA Transceiver Channel with External Loopback - IV**

e. Connect J11 to J12 with a *SATA loopback cable* included in the Virtex-6 FPGA Connectivity Kit (see Figure 69).



UG664_46_011610

Figure 69: **Configuring the SMA Transceiver Channels and SATA Channels with External Loopback - V**

6. Board Setup V – Connect the power connector:
   a. Using the included power supply, connect the power supply connector to the ML605 board as shown in Figure 70.
   b. The power switch SW2 should be switched to the ON position.

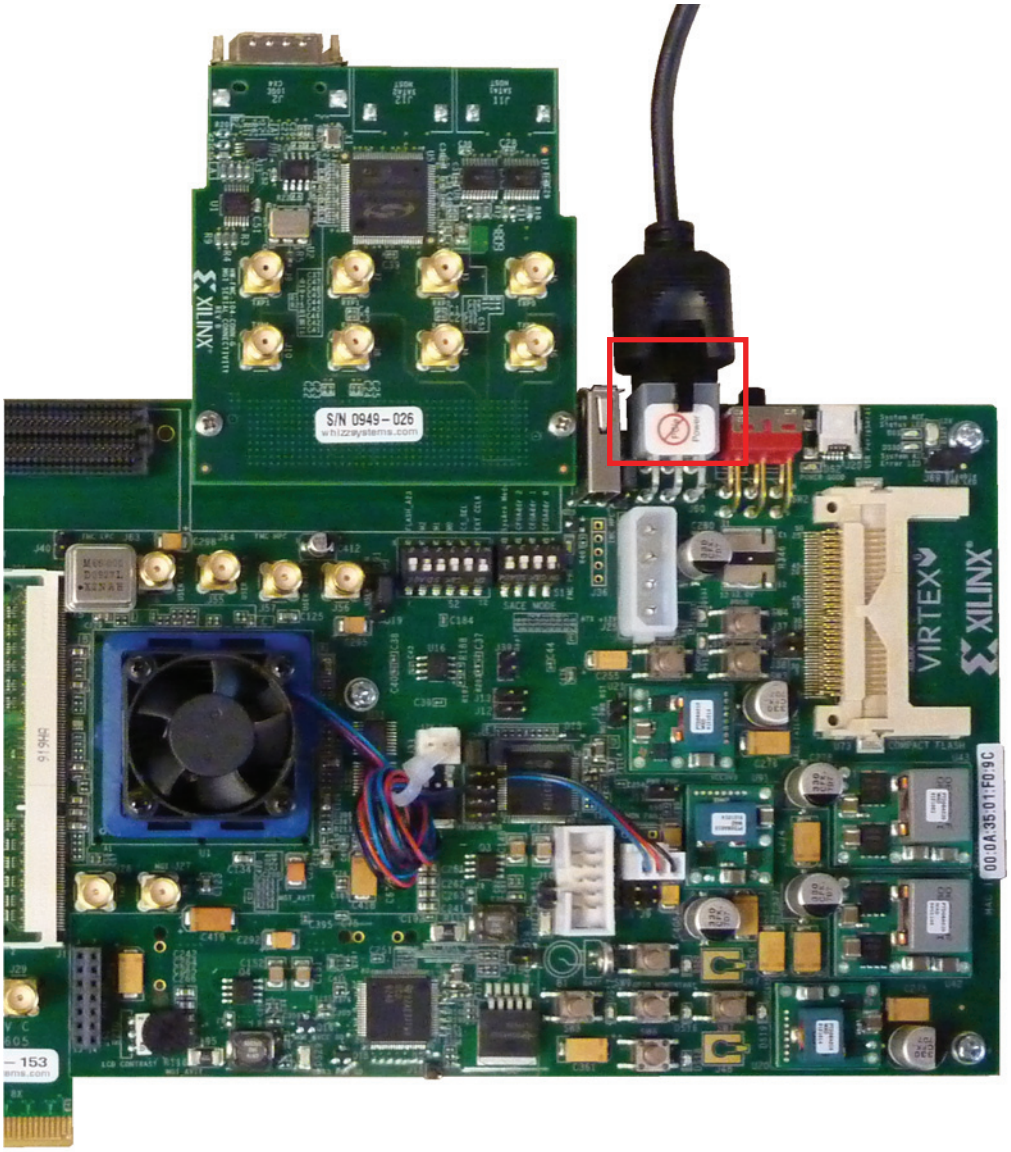
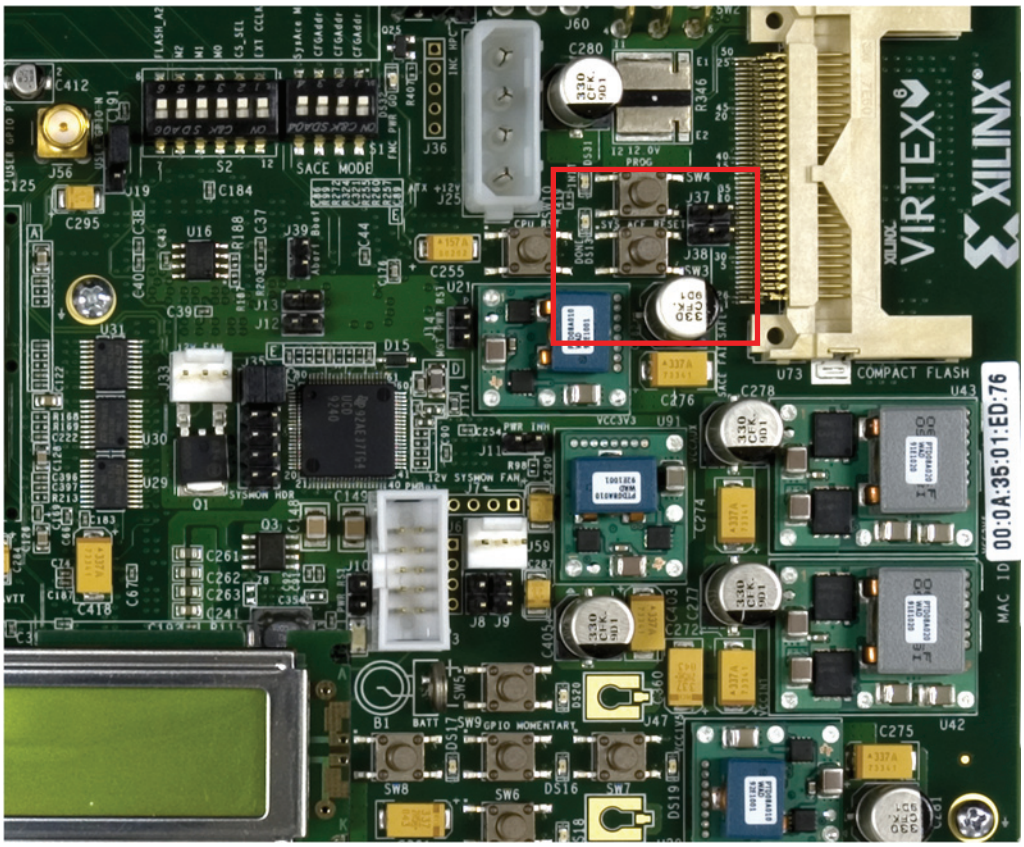


Figure 70: **Powering Up the ML605 Board**

7. Board Setup VI – Load the FPGA with the IBERT design from the CompactFlash:
   a. Press switch SW3 to configure from the CompactFlash.
   b. Verify that the FPGA is loaded with the IBERT design. The DONE LED should be lit.



UG664_48_011610

Figure 71: **FPGA Programmed with the IBERT Reference Design**

8. The IBERT Reference Design files are provided on a USB flash drive delivered as a part of the kit. Copy the contents of the included USB flash drive:
   a. Insert the USB flash drive into a USB connector of the PC system.
   b. Wait for the operating system to mount the USB flash. When the flash is mounted, an icon pops up on the desktop.
   c. Navigate to the USB flash drive and copy the `ML605_FMC_XM104_Ibert_Reference_Design` folder into a local directory.
   d. Eject the USB flash drive.

9. Open the ChipScope Pro Analyzer window:
   a. Click on **Programs** → **Xilinx ISE Design Suite** → **ChipScope Pro** → **Analyzer**.
   b. Click on **Open Cable Button** as shown in Figure 72.

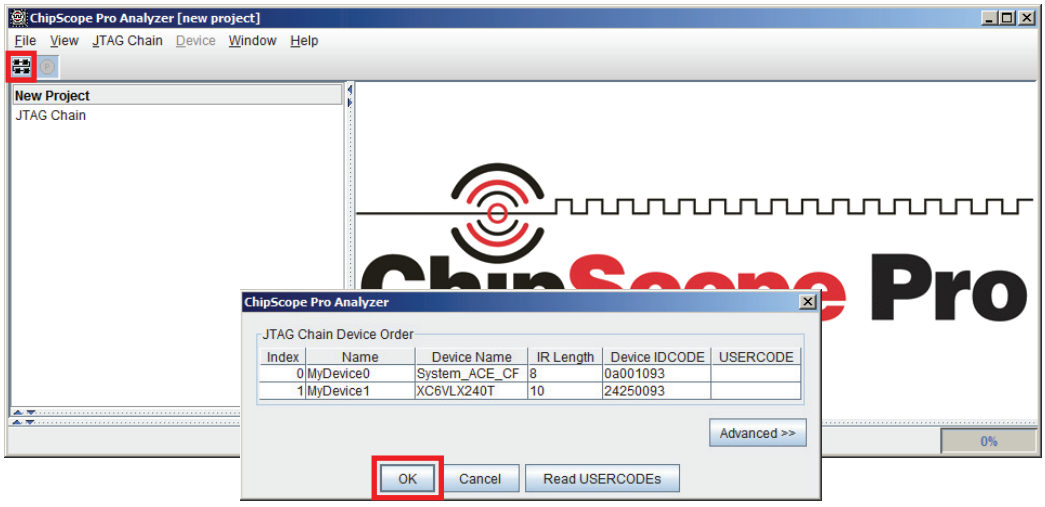


Figure 72: **Launch the ChipScope Pro Analyzer Window**

10. Open the ChipScope Pro Analyzer project (see Figure 73):
    a. Click **File** → **Open Project**.
    b. Navigate to the `ML605_FMC_XM104_Ibert_Reference_Design` folder.
    c. Select **ml605_fmc_xm104_ibert.cpj**.

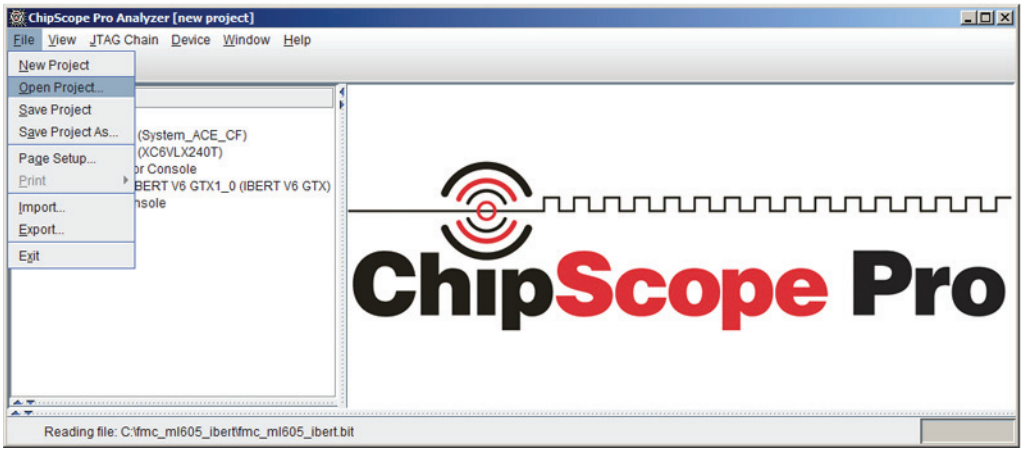
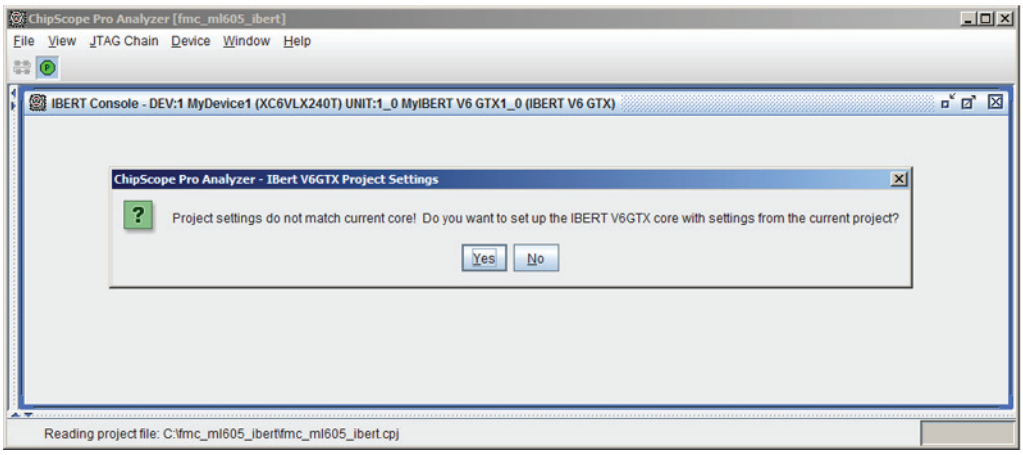


Figure 73: **Open an Existing ChipScope Tool Project in the IBERT Design Package**

11. Load the ChipScope Pro Analyzer project:
    a. Click **Yes** on the dialog box shown in Figure 74.



UG664_51_011710

Figure 74: **Load the ChipScope Tool Project and Communicate with the IBERT Reference Design**

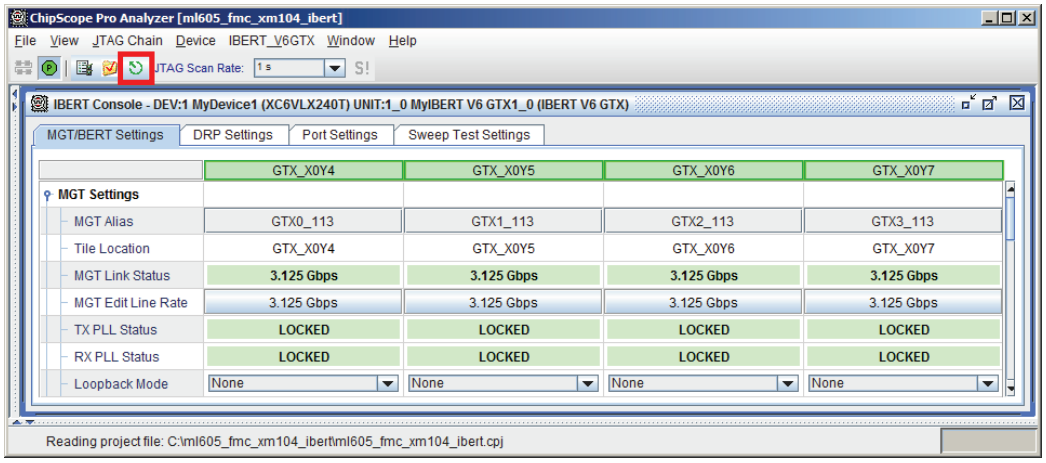
12. Load and reset the IBERT reference design through the GUI (see Figure 75).

    GTX0_113 → FMC Daughter Card connector: DP3 SATA2 Host Channel

    GTX1_113 → FMC Daughter Card connector: DP2 SATA1 Host Channel

    GTX2_113 → FMC Daughter Card connector: DP1 SMA Channel

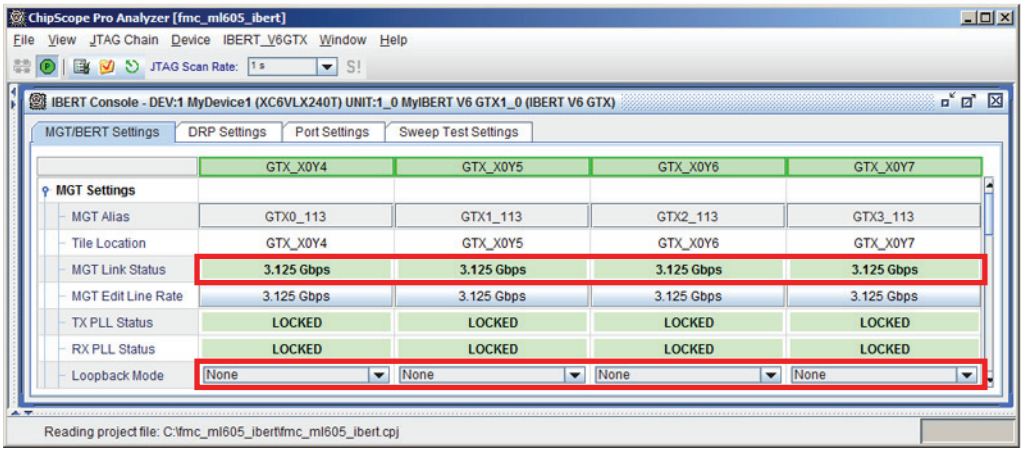
    GTX3_113 → FMC Daughter Card connector: DP0 SMA Channel



UG664_52_021810

Figure 75: **Load and Reset the IBERT Reference Design**

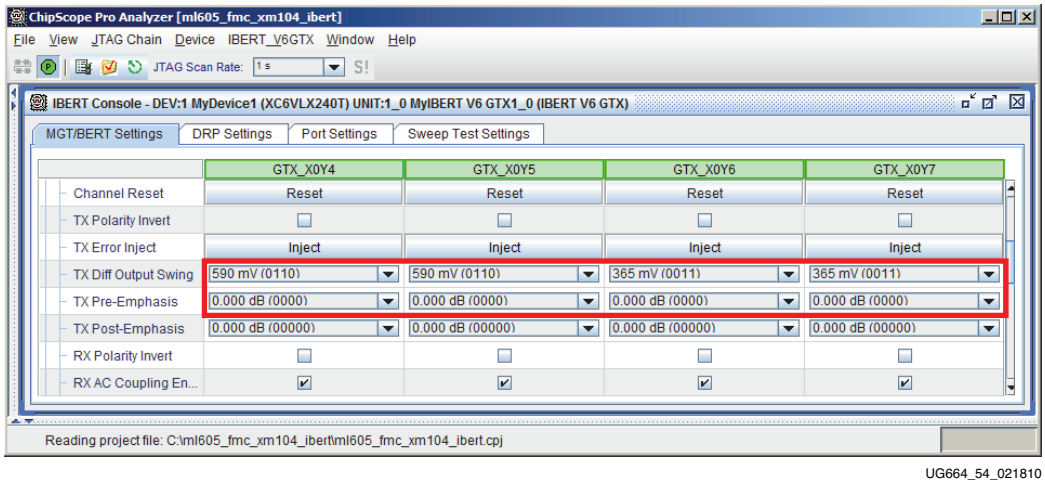
13. Verify the line rates for the GTX transceivers (see Figure 76):
    a. The line rate is set to 3.125 Gb/s for all four GTX transceiver channels instantiated in the design.
    b. The GTX0_113 and GTX1_113 transceiver channels have been looped on external loopback through a SATA cable. Select the loopback mode for these transceiver channels as **None** (no internal loopback). The GTX2_113 and GTX3_113 transceiver channels have been looped on external loopback through an SMA cable. Select the loopback mode for these transceiver channels as **None** (no internal loopback).



UG664_53_011710

Figure 76: **Verify the GTX Transceiver Loopback Configuration and Link Status**

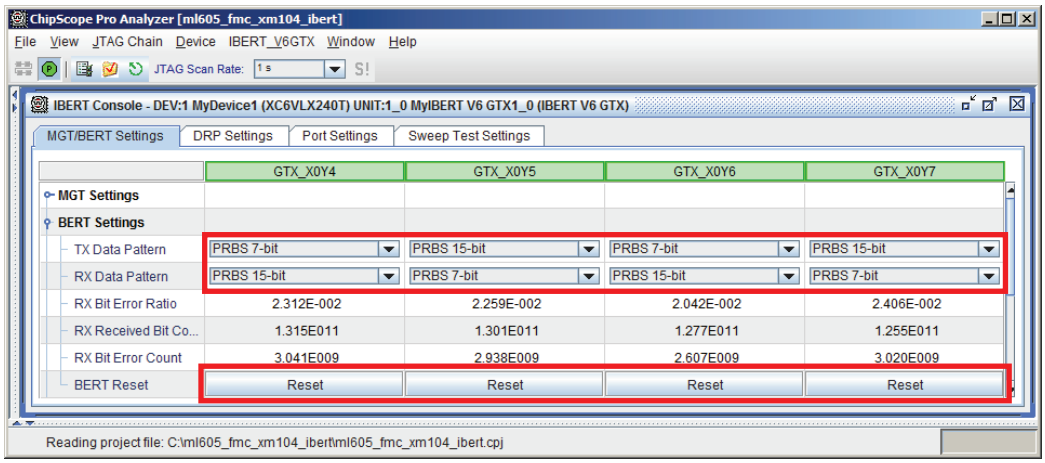
14. Configure the GTX transmit parameter settings (see Figure 77):
    a. Set the *TX Diff Output Swing* parameter to 590 mV (`0110`) for the SATA channels and 365 mV for the SMA channels.
    b. Set the *TX Pre-Emphasis* parameter to 0 dB (`000`).



UG664_54_021810

Figure 77: **Modifying the Transmitter Settings of the GTX Transceiver Channels**

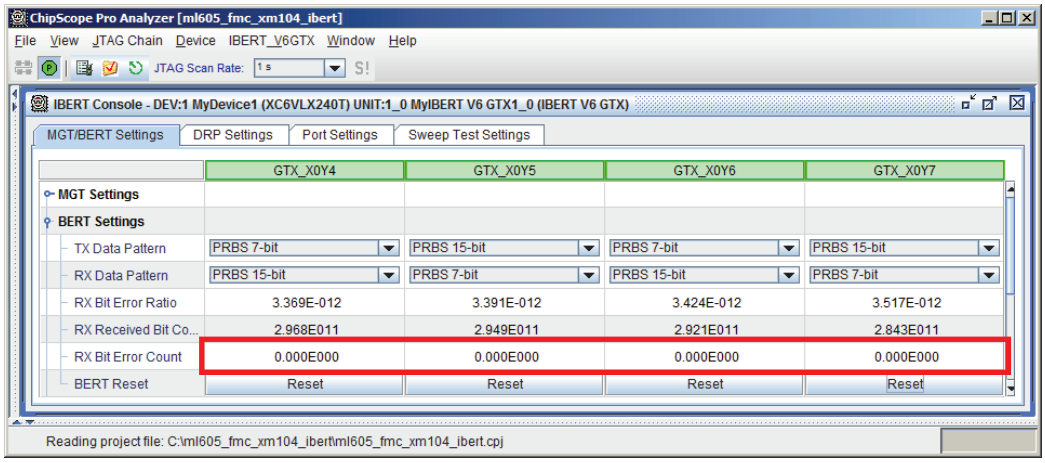
15. Configure the bit error ratio test (BERT) parameter settings (see Figure 78):
    a. Set the TX/RX data patterns to PRBS 7-bit and 15-bit.
    b. Click the BERT **Reset** buttons for each channel.



UG664_55_021810

Figure 78: **Configuring the BERT Settings for the GTX Transceiver Channels**

16. View the reported BERT (see Figure 79). The RX bit error count should be 0.



UG664_56_021810

Figure 79: **Verify the Bit Error Ratio on All Four Transceiver Channels**

Congratulations! The IBERT reference design for the Virtex-6 FPGA Connectivity Kit has been set up and the pre-built demo that uses the GTX transceivers running at 3.125 Gb/s has been tested.

For further details on other example reference designs available for the ML605 board, refer to http://www.xilinx.com/ml605 and click on **ML605 Documentation**.

Reference Design Files

The design checklist in Table 2 includes simulation, implementation, and hardware details for the reference designs. After registration, reference design files are available for download at ug664.zip.

Table 2: **Design Checklist**

Parameter	Description
General	
Developer Name	Xilinx
Target devices (stepping level, ES, production, speed grades)	XC6VLX240T-1-FF1156
Source code provided	Y (for custom logic only)
Source code format	Verilog
Design uses code or IP from an existing reference design or application note, third party, CORE Generator software	Uses code from a third party and LogiCORE IP from the CORE Generator software
Simulation	
Functional simulation performed	Y
Timing simulation performed	N
Testbench used for functional and timing simulations	Y (for functional simulations)
Testbench format	System Verilog (inhouse verification), Verilog (customer deliverable)
Simulator software/version used	ModelSim Questa 6.5a (inhouse verification)/ ModelSim 6.4b (out-of-box simulation support)
SPICE/IBIS simulations	N
Implementation	
Synthesis software tools/version used	XST
Implementation software tools/versions used	ISE Design Suite
Static timing analysis performed	Y
Hardware Verification	
Hardware verified	Y
Hardware platform used for verification	ML605 board and FMC X104 Connectivity daughter card

Installation is Complete

The Xilinx design tools have been successfully installed, the CORE Generator tool flow for IP delivery is better understood, and the FPGA application is ready to be designed and implemented targeting the Virtex-6 LXT architecture.

For updated information on this Virtex-6 FPGA Connectivity Kit, go to http://www.xilinx.com/v6connkit. Check this page regularly for the latest in documentation, FAQs, reference design examples, product updates, and known issues.

Warranty

THIS LIMITED WARRANTY applies solely to standard hardware development boards and standard hardware programming cables manufactured by or on behalf of Xilinx ("Development Systems"). Subject to the limitations herein, Xilinx warrants that Development Systems, when delivered by Xilinx or its authorized distributor, for ninety (90) days following the delivery date, will be free from defects in material and workmanship and will substantially conform to Xilinx publicly available specifications for such products in effect at the time of delivery. This limited warranty excludes:
(i) engineering samples or beta versions of Development Systems (which are provided "AS IS" without warranty); (ii) design defects or errors known as "errata";
(iii) Development Systems procured through unauthorized third parties; and
(iv) Development Systems that have been subject to misuse, mishandling, accident, alteration, neglect, unauthorized repair or installation. Furthermore, this limited warranty shall not apply to the use of covered products in an application or environment that is not within Xilinx specifications or in the event of any act, error, neglect or default of Customer. For any breach by Xilinx of this limited warranty, the exclusive remedy of Customer and the sole liability of Xilinx shall be, at the option of Xilinx, to replace or repair the affected products, or to refund to Customer the price of the affected products. The availability of replacement products is subject to product discontinuation policies at Xilinx. Customer may not return product without first obtaining a customer return material authorization (RMA) number from Xilinx.

THE WARRANTIES SET FORTH HEREIN ARE EXCLUSIVE. XILINX DISCLAIMS ALL OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT, AND ANY WARRANTY THAT MAY ARISE FROM COURSE OF DEALING, COURSE OF PERFORMANCE, OR USAGE OF TRADE. (2008.10)

Do not throw Xilinx products marked with the "crossed out wheelie bin" in the trash. Directive 2002/96/EC on waste electrical and electronic equipment (WEEE) requires the separate collection of WEEE. Your cooperation is essential in ensuring the proper management of WEEE and the protection of the environment and human health from potential effects arising from the presence of hazardous substances in WEEE. Return the marked products to Xilinx for proper disposal. Further information and instructions for free-of-charge return available at: http:\\www.xilinx.com\ehs\weee.htm.

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Optical Sensor Development Tools ***category:***

Click to view products by Pyreos ***manufacturer:***

Other Similar products are found below :

MT9P031I12STCH-GEVB MT9V032C12STCH-GEVB MT9V034C12STCH-GEVB MT9V115EBKSTCH-GEVB ISL76683EVAL1Z AR0130CSSC00SPCAH-GEVB AR0331SRSC00XUEAH-GEVB AS0260HQSC28SUKAD3-GEVK 416015300-3 ISL29102IROZ-EVALZ MT9D131C12STCH-GEVB MT9M021IA3XTMH-GEVB MT9M034I12STMH-GEVB MT9P031I12STMH-GEVB TMD2725-DB AR1820HSSC12SHQAH3-GEVB AR0331SRSC00SUCAH-GEVB MT9M001C12STMH-GEVB MT9V032C12STCD3-GEVK MT9P031I12STCD3-GEVK MT9M034I12STCH-GEVB MT9V013PACSTCHM-GEVB AR1335CSSC11SMKAH3-GEVB AR0331SRSC00SHCAD3-GEVK MAXCAMOV10640# MT9T112PACSTCH-GEVB AR0141CS2M00SUEAH3-GEVB MT9V022IA7ATCH-GEVB AR0331SRSC00SHCAH-GEVB MT9M031I12STMH-GEVB MT9F002I12-N4000H-GEVB AR1337CSSC32SMFAH3-GEVB MT9V013PACSTCHP-GEVB MT9F002I12STCVH-GEVB TMD2620-DB ISL29028AIROZ-EVALZ AR1335CSSC32SMFAH3-GEVB TSL4531-DB TSL2581FNEVM TSL2581CS-DB TMD3700-DB ISL78365EVAL1Z KAI-2020-12-20-A-EVK ASX340AT3C00XPEDH3-GEVB AR0140CS2C00SUEAH3-GEVB AR0430CS2C34SMFAH3-GEVB AR0231AT7R00XUEAH3-GEVB AR0231AT7C00XUEAH3-GEVB AR0231AT7B00XUEAH3-GEVB AR0144CSSC20SUKAH3-GEVB