PCIe Colorbar Demo for Lattice Nexus-based FPGAs

User Guide

FPGA-UG-02149-1.2

May 2023

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults and associated risk the responsibility entirely of the Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
DMA	Direct Memory Access
BAR	Base Address Register
FDSOI	Fully Depleted Silicon on Insulator
FPGA	Field-Programmable Gate Array
LED	Light-emitting diode
MIPI	Mobile Industry Processor Interface
PCIe	PCI Express
PHY	Physical Layer
SPI	Serial Peripheral Interface
USB	Universal Serial Bus

1. Introduction

This guide describes how to set up and run the PCIe Colorbar Demo using devices built on the Lattice Nexus™ platform, specifically CrossLink™-NX, Certus™-NX and CertusPro™-NX devices.

For CrossLink-NX devices, the demo is targeted to the CrossLink-NX PCIe Bridge Board, which features the CrossLink-NX FPGA in the 400-ball caBGA package (LIFCL-40-8BG400C). For Certus-NX devices, the demo is targeted towards the Certus-NX Versa Evaluation Board, which features the Certus-NX FPGA in the 256-ball caBGA package (LFD2NX-40-8BG256C). For CertusPro-NX devices, the demo is targeted to the CertusPro-NX Versa Evaluation Board, which features the CertusPro-NX FPGA in the LFG672 package. Both FPGAs are built on the Lattice Nexus FPGA platform using low power 28 nm FDSOI technology. The above-mentioned FPGAs are built on the Nexus FPGA platform using low power 28 nm FD-SOI technology.

This guide familiarizes you with the process of setting up your PCI Express development environment. It is assumed that you do not have any associated tools installed on your system.

The demo discussed in this document includes the PCI Express Colorbar Demo.

1.1. Learning Objectives

After completing the steps in this guide, you will be able to perform the following:

- Set up and install all applicable development tools and PCI Express demos;
- Establish communication between the FPGA and the system through the PCI Express link;
- Run the PCI Express Colorbar Demo, which displays a series of moving colorbars by streaming the image data using DMA transfers from the FPGA to the host system.
- Modify and rebuild the PCI Express Colorbar Demo.
- Become familiar with the software development tools and major design flow steps employed in this kit.
- Use other existing documentation in conjunction with this guide.

This document assumes that you have already installed the Lattice Radiant™ design software. This document covers some of the basic of function of Lattice Radiant software. If you would like to learn more about Lattice Radiant software, refer to the Lattice Radiant software Help system.

2. Hardware and Software Requirements

2.1. Hardware Requirements

To install the kit design and run the demo software, a computer with a PCI Express ×16, ×8, ×4, or ×1 slot is required. The computer must also have a USB port and be able to run the Lattice Radiant Software. All other hardware and drivers are included in the kit.

- Mini-USB to USB-A cable for programming the bitstream
- 12 V Power Adapter
- Evaluation Board
- CrossLink-NX PCIe Bridge Board for CrossLink-NX devices
- Certus-NX Versa Evaluation Board for Certus-NX devices
- CertusPro-NX Versa Board for CertusPro-NX devices

2.2. Software Requirements

The following software is required to obtain the expected results for the procedures described in this guide:

- Lattice Radiant Software 2.2 or later (available at the Lattice website Design Software and IP page).
- PCIe Colorbar Demo for Windows 10 or Linux
- Windows 10 or Ubuntu 18.04.6
- Bit file for the SPI Flash
- LIFCL40_PCIeColorbarDemo.bit for CrossLink-NX devices
- LFD2NX_PCIeColorbarDemo.bit for Certus-NX devices
- LPCNX_PCIeColorbarDemo_1X1.bit, LPCNX_PCIeColorbarDemo_1X4.bit, LPCNX_PCIeColorbarDemo_2X1.bit, and LPCNX_PCIeColorbarDemo_3X4.bit for CertusPro-NX devices depending on the link speed and lane width

3. Setting Up the Demo

3.1. Hardware Setup

This section covers the steps in programming the demo to the SPI memory of the CrossLink-NX PCIe Bridge Board. Make sure the board is plugged into a PCIe slot or connected to the 12 V input supply.

For CrossLink-NX devices, if the board is plugged into the PCIe slot, external power is provided by the system, and SW2 should be in the *up* position to receive power from the PCIe slot. If external 12 V power is provided, then SW2 should be in the *down* position to receive power from the external 12 V connection. Connect the board to the PC running the Lattice Radiant software with the Mini USB Type A cable as shown in Figure 3.1.

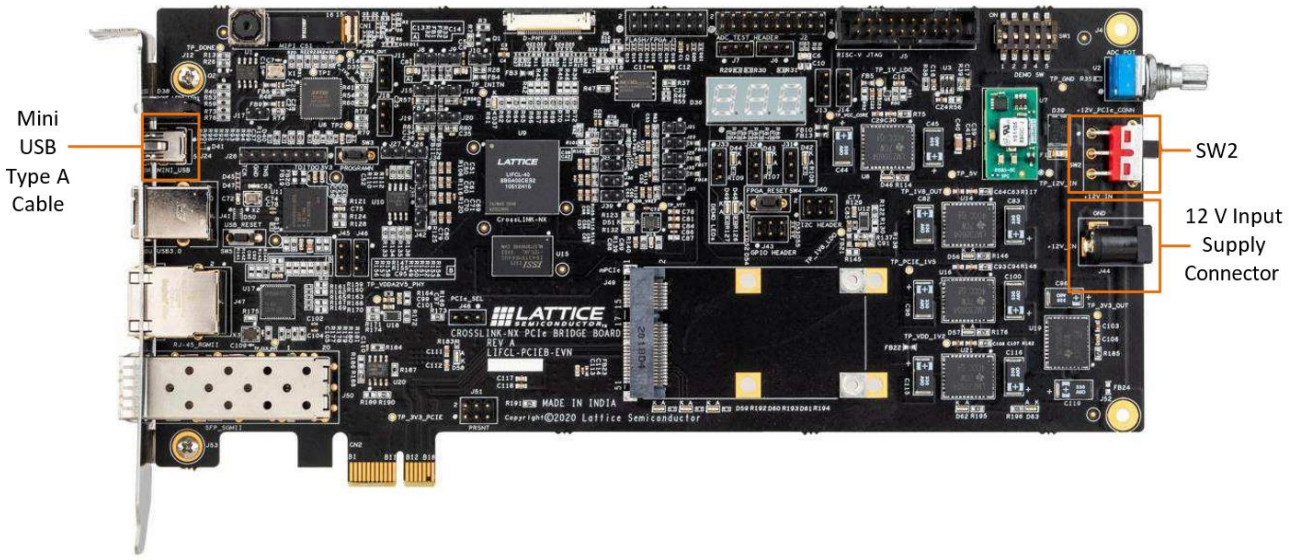


Figure 3.1. CrossLink-NX PCIe Bridge Board Connection

For Certus-NX devices, there are no jumpers to configure beyond the default. Refer to the Certus-NX Evaluation Board User Guide (FPGA-EB-02032) for the default jumper configuration.

There is no toggle switch to select the power source, and the board can be powered through the PCIe slot or from an external 12 V input supply. If the board is powered from the PCIe slot, the external 12 V input supply connector does not need to be connected. Connect the board to the PC running the Lattice Radiant software with the Mini USB Type A Cable as shown in Figure 3.2.

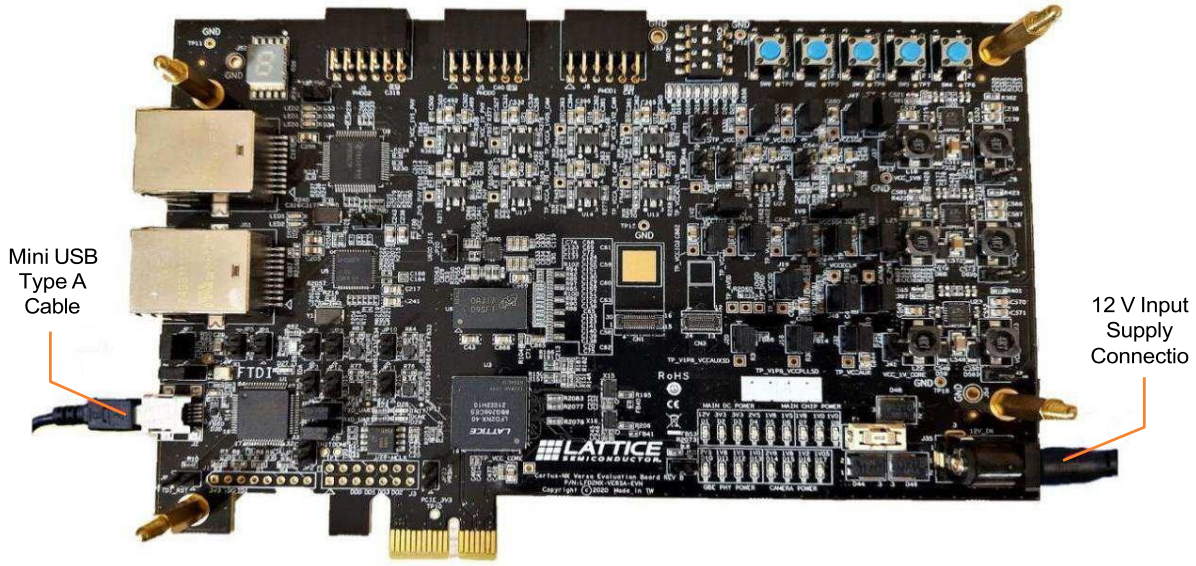


Figure 3.2. Certus-NX PCIe Versa Evaluation Board Connection

For CertusPro-NX devices, there are no jumpers to configure beyond the default. Refer to the CertusPro-NX Versa Board User Guide (FPGA-EB-02053) for a detailed list of jumpers on the board. when the board is plugged into the PCIe slot, external power is provided by the system, and SW6 should be in the up position to receive power from the PCIe slot. Connect the board to the PC running the Lattice Radiant software with the Mini USB Type A Cable as shown in Figure 3.3.

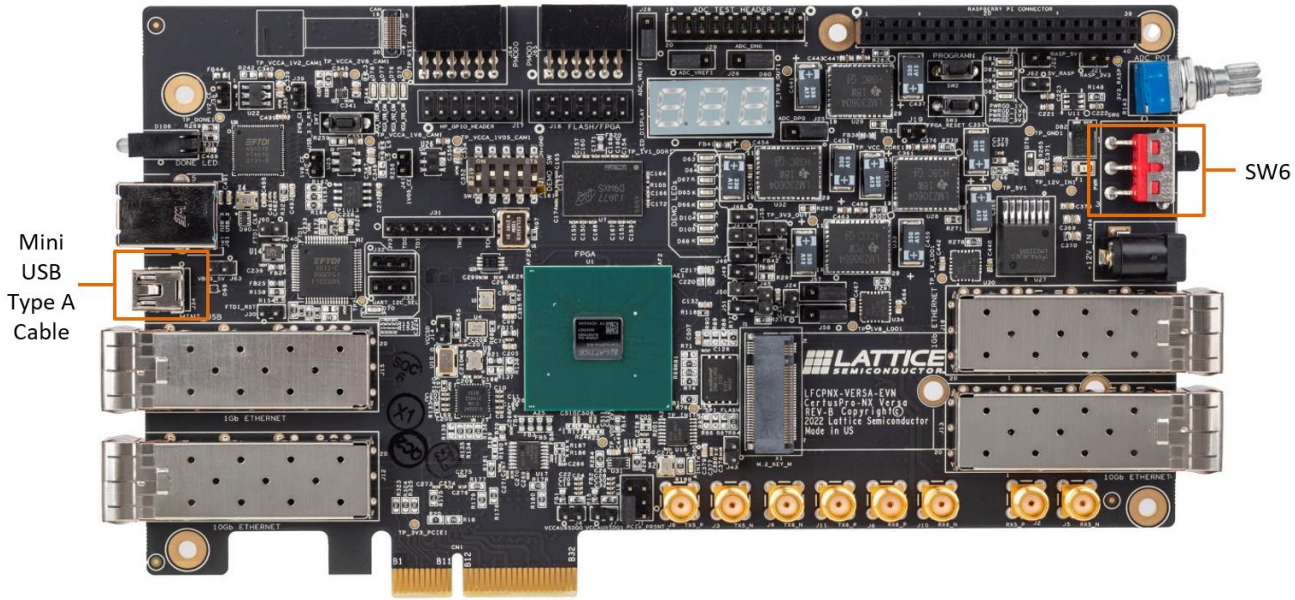


Figure 3.3. CertusPro-NX Versa Board Connection

3.1.1. Programming the FPGA

To program the FPGA device:

1. Create a new project using the Lattice Radiant Programmer software. In the **Getting Started** dialog box, indicate **Project Name** and **Location** as shown in Figure 3.4.
2. Select **Create a new project from scan**. Values are indicated in the **Cable**, **Port**, and **TCK Divider Setting (0-30×)** fields.
3. Click **OK**.

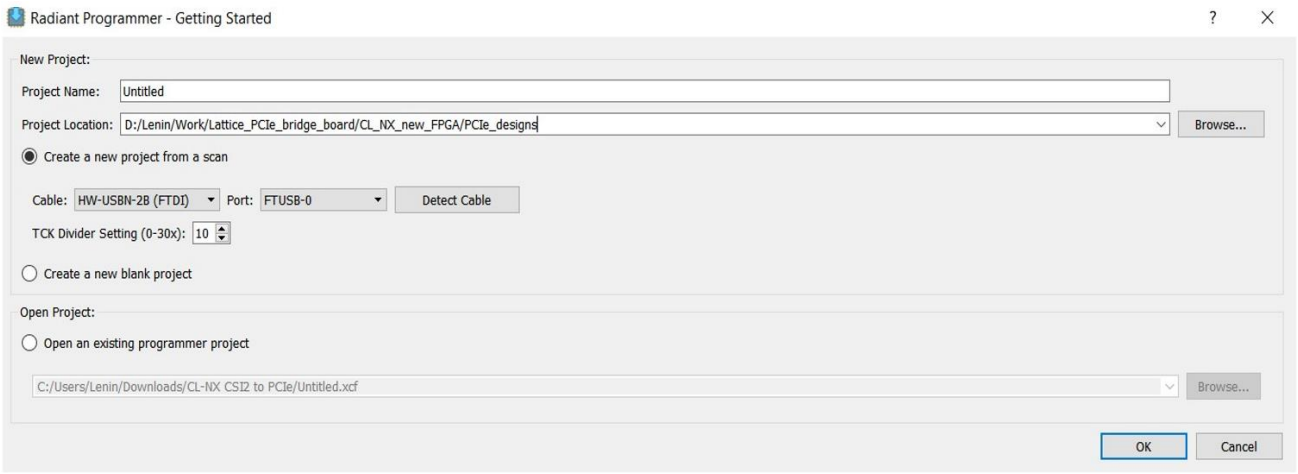


Figure 3.4. Creating a New Project from a Scan

4. The main interface opens as shown in Figure 3.5.

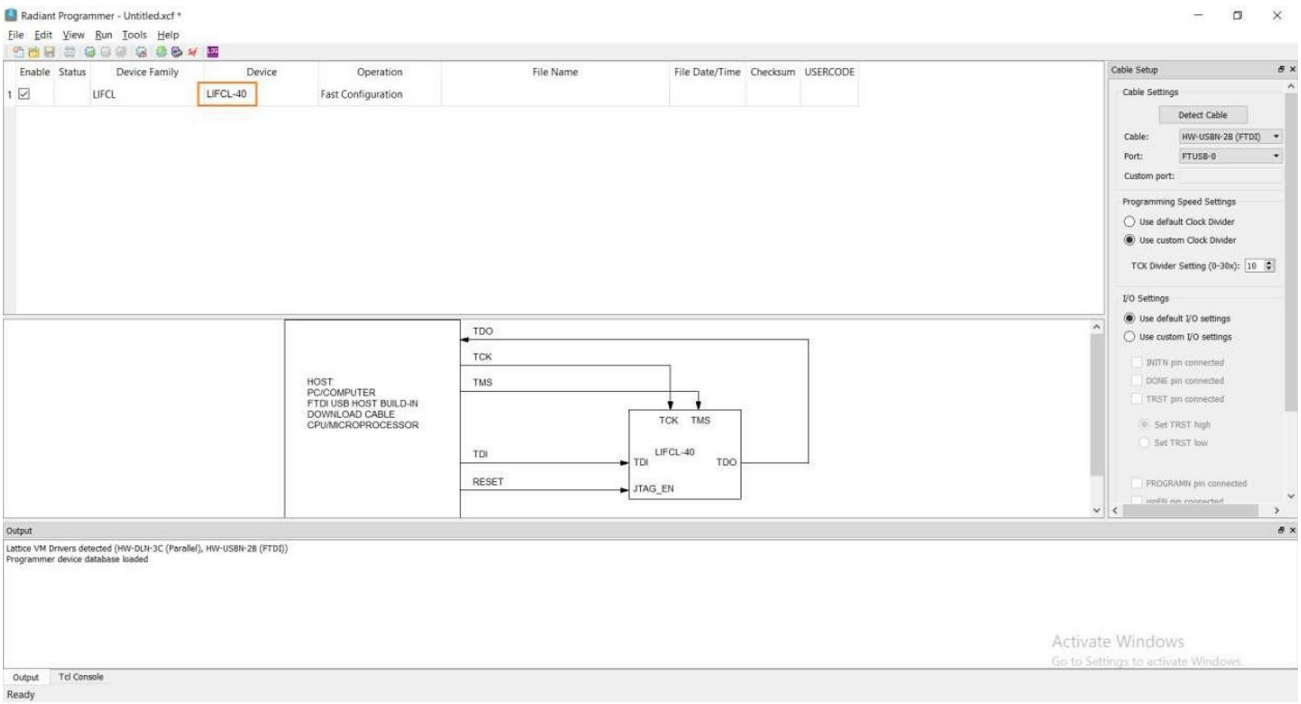


Figure 3.5. Lattice Radiant Programmer Window

5. If the Programmer settings do not match the settings shown in Figure 3.6 for CrossLink-NX devices, Figure 3.7 for Certus-NX devices, or Figure 3.8 for CertusPro-NX devices, select these settings manually from drop down menu.

Figure 3.6. CrossLink-NX FPGA Device Settings

Figure 3.7. Certus-NX FPGA Device Settings

Figure 3.8. CertusPro-NX FPGA Device Settings

To select programming settings:

1. Browse and select the Programming file *Demonstration/Bitstream*.
   a. For CrossLink-NX devices, select LIFCL40_PCIeColorBarDemo.bit from CL-NX_BridgeBoard_PCIe_BMD\Demonstration\Bitstream.
   b. For Certus-NX devices, select LFD2NX_PCIeColorBarDemo.bit from CertusNX _PCIe_ColorbarDemo\Demonstration\Bitstream.
   c. For CertusPro-NX devices, select LPCNX_PCIeColorbarDemo_1X1.bit, LPCNX_PCIeColorbarDemo_1X4.bit, LPCNX_PCIeColorbarDemo_2X1.bit, or LPCNX_PCIeColorbarDemo_3X4.bit depending on link width and link speed from Demonstration\Bitstream.
2. Click **OK**.
3. Double-click under **Operation** to open the **Device Properties** dialog box.

4. For CrossLink-NX devices, select the settings as shown in Figure 3.9.

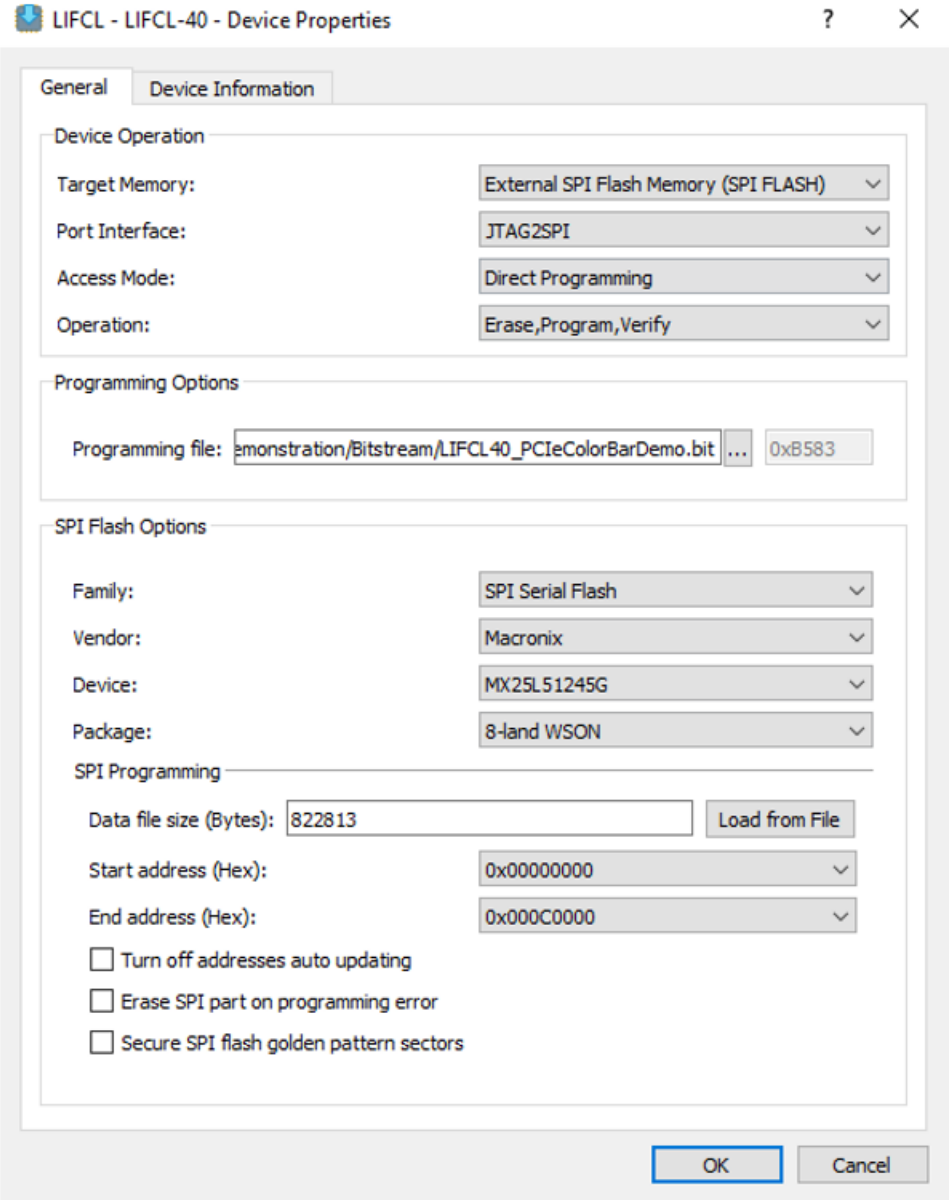


Figure 3.9. Device Properties Window for CrossLink-NX SPI Flash Programming

5. For Certus-NX devices, select the settings as shown in Figure 3.10.

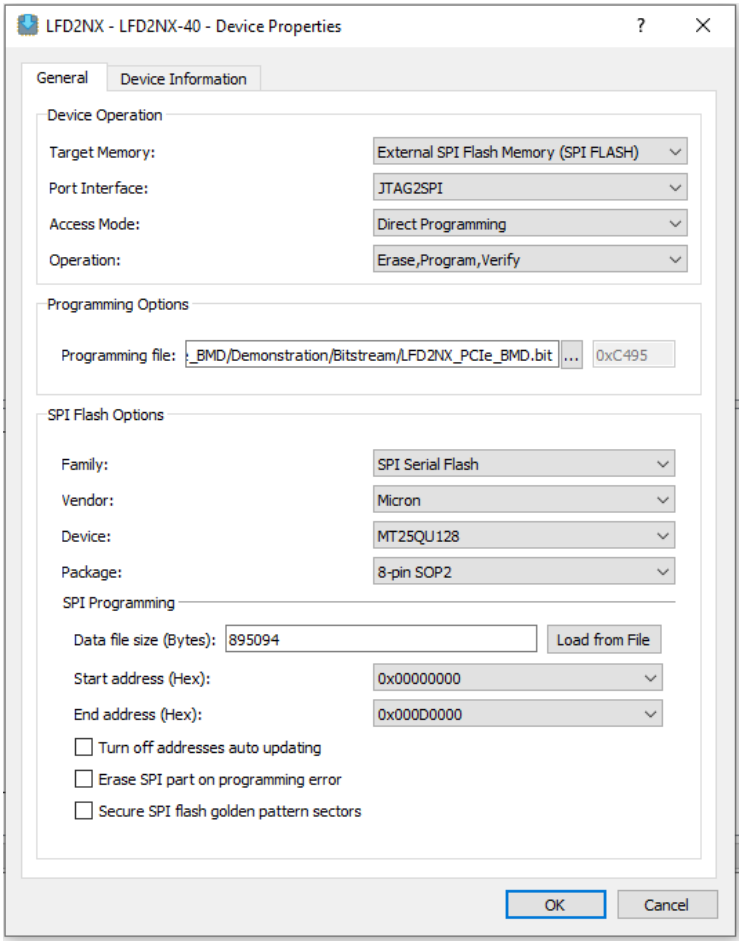


Figure 3.10. Device Properties Window for Certus-NX SPI Flash Programming

6. For CertusPro-NX devices, select the settings as shown in Figure 3.11.

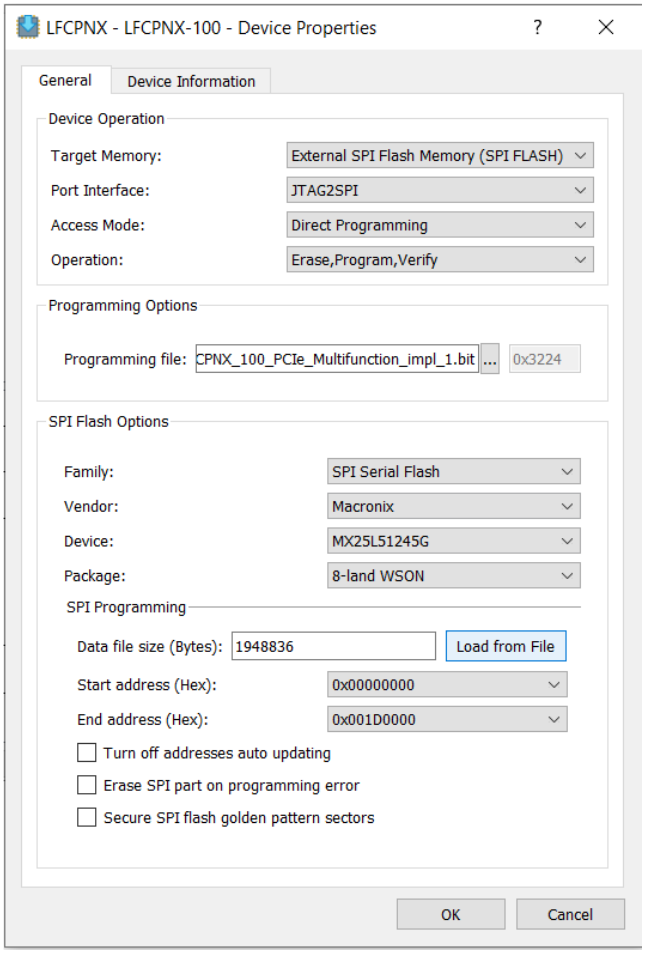


Figure 3.11. Device Properties Window for CertusPro-NX SPI Flash Programming (with Macronix Flash)

7. Click the **Programming** button from the menu bar shown in Figure 3.12 to start programming.

Figure 3.12. Programmer Menu Bar

When the FPGA programming is successful, the output console shows an Operation Successful message as shown in Figure 3.13.

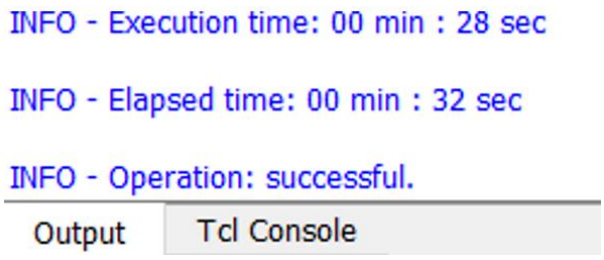


Figure 3.13. Programmer Output Window

If the programming operation is not successful, refer to the Troubleshooting section of this document.

After programming, power cycle the board and check the status LEDs on the board.

- For CrossLink-NX devices, refer to Figure 3.14 and Table 3.1.
- For Certus NX devices, refer to Figure 3.15.
- For CertusPro NX devices, refer to Figure 3.16.

3.1.2. Status LED

The three status LEDs are shown in Figure 3.14.

Figure 3.14. Status LED

The description of each status LED is provided in Table 3.1.

Table 3.1. CrossLink-NX Status LED Description

Sl. No	Name	Description
1	125 MHz Clock present	Green – Blinking when clock present (125 MHz Clock)
2	PCIe Link Up	Green – PCIe link up is successful
		Red – PCIe link up is not successful
3	DONE	Green – Configuration is successful Red – Board is not programmed

For Certus-NX devices, the programming done LED lights up in green if configuration is successful. The LED is located at D29 on the board, as indicated in Figure 3.15.

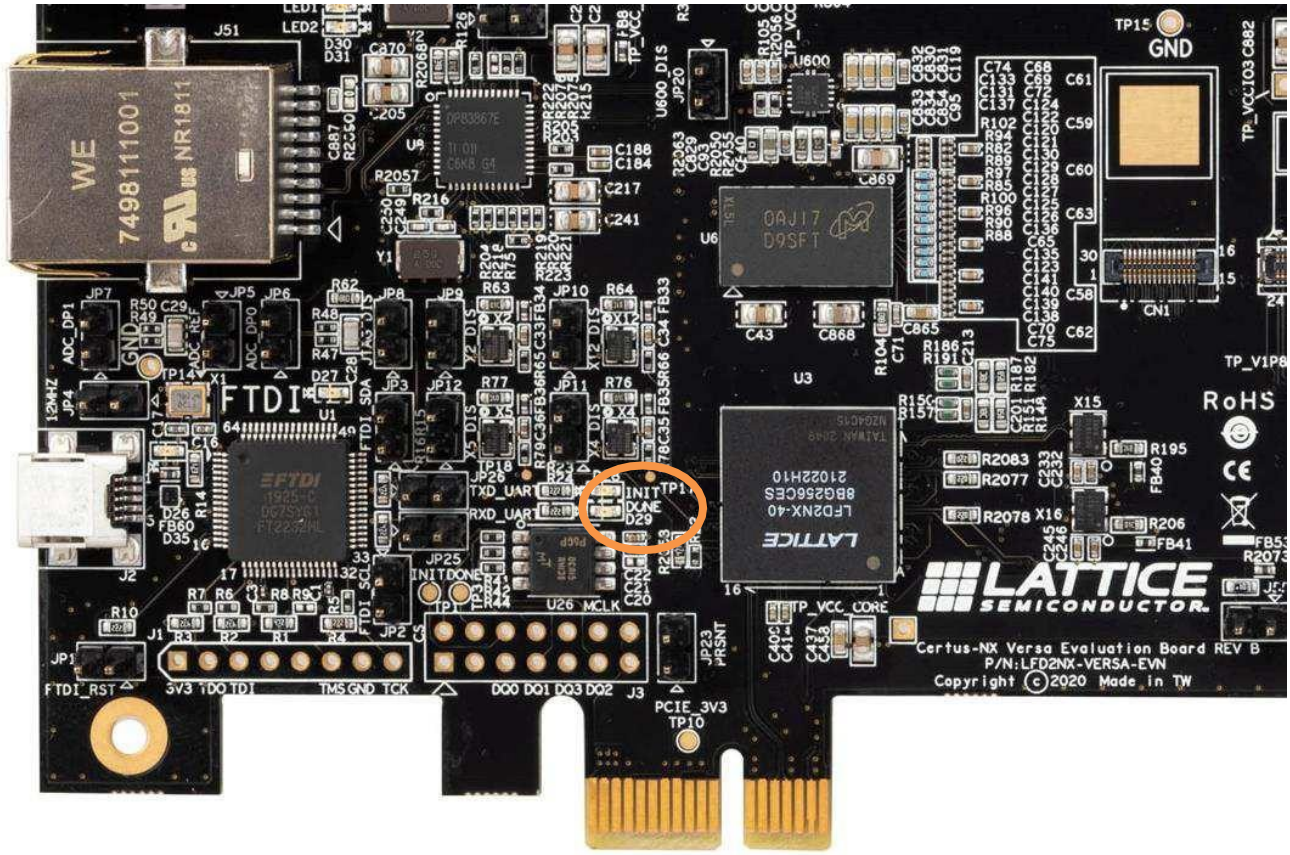


Figure 3.15. Certus-NX Programming Done LED

For CertusPro-NX devices, the programming done LED lights up in green if configuration is successful. The LED is located as shown in Figure 3.16.

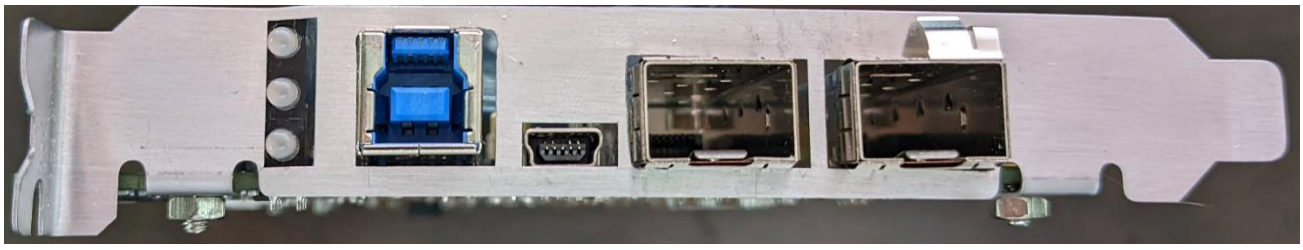


Figure 3.16. CertusPro-NX Programming Done LED

3.2. Software Setup

This section provides the procedure for installing software onto the host machine.

3.2.1. Software Setup and Installation for Windows

Before installing the driver, driver signature enforcement should be disabled.

3.2.1.1. Disabling Driver Signature Enforcement Permanently

To permanently disable driver signature enforcement:

1. Start the Command Prompt as administrator.
2. Enter the following lines and press Enter.

```
bcdedit.exe -set loadoptions DISABLE_INTEGRITY_CHECKS
```

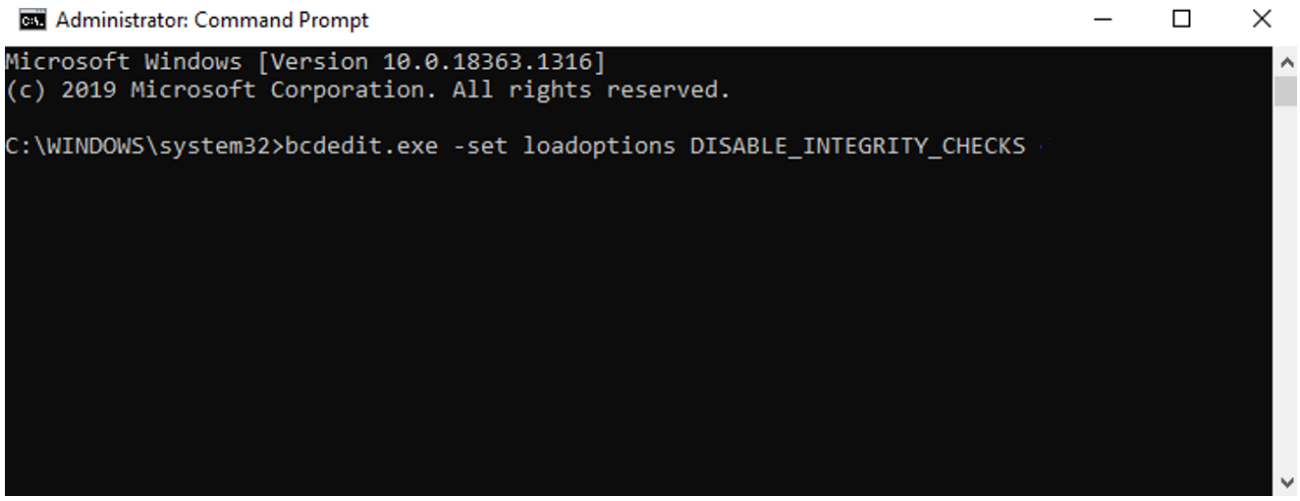


Figure 3.17. Running Disable Integrity Checks Command

```
bcdedit.exe -set TESTSIGNING ON
```

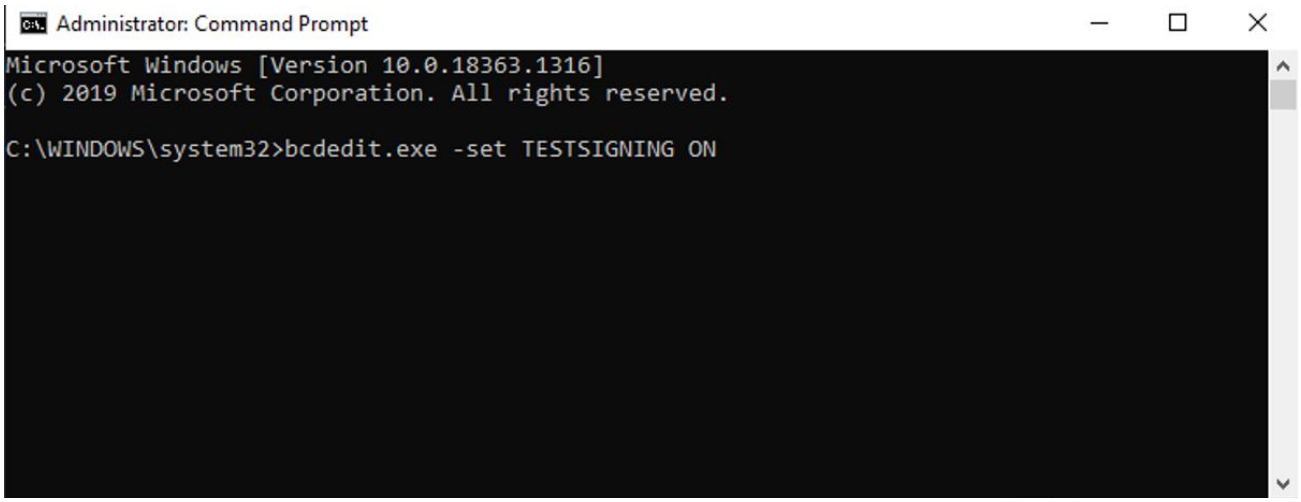


Figure 3.18. Running Test Sign on Command

3. Close the Command Prompt and restart your PC.

3.2.1.2. Disabling Driver Signature Enforcement Temporarily

To disable the driver signature enforcement temporarily on Windows 10:

Note: if Driver Signature Enforcement is already disabled, skip this section and proceed to the Driver Installation section.

1. Press the Windows key and click the power button.
2. Press and hold the Shift key and click **Restart**.
3. When the Choose an option screen appears as shown in Figure 3.19, release the Shift key.
4. Click **Troubleshoot**.

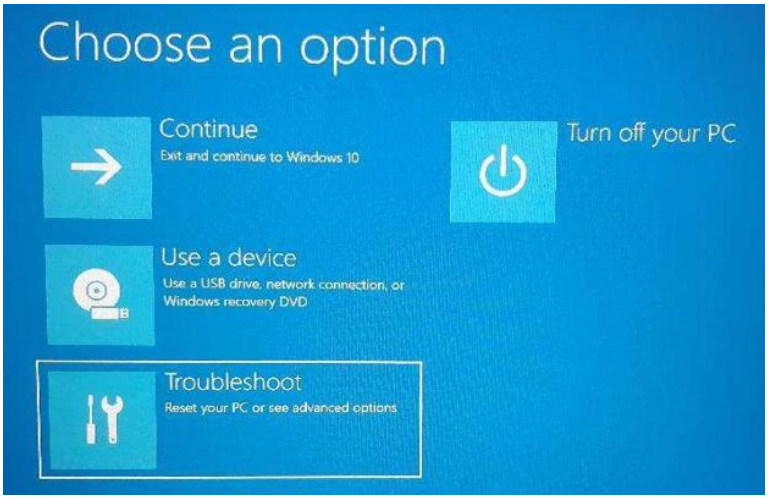


Figure 3.19. Troubleshoot Option

5. Select **Advanced options** and press Enter.

Figure 3.20. Advanced Options

6. Select **Startup Settings** and press Enter.

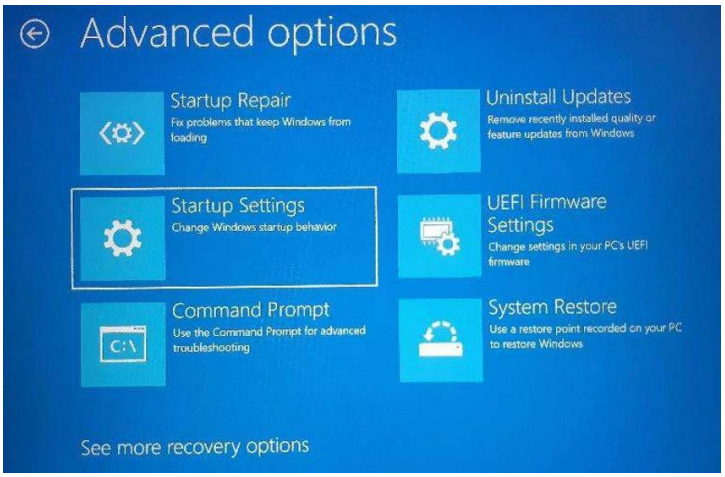


Figure 3.21. Select Startup Settings

7. Press Enter to restart.

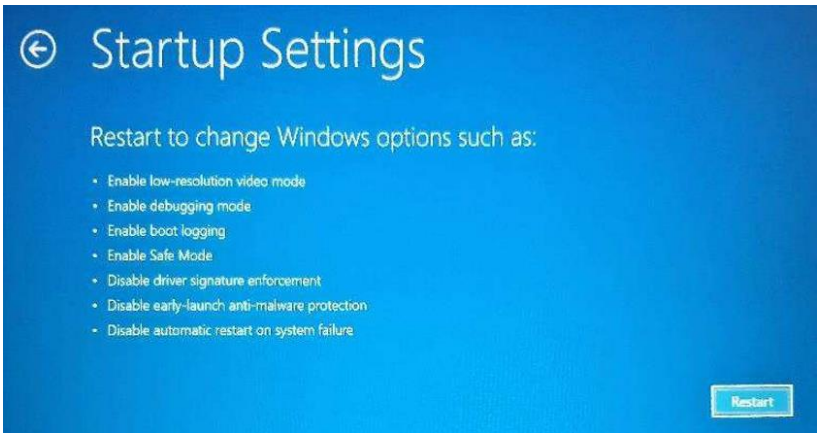


Figure 3.22. Restarting Windows

8. After restarting, select Option 7 to disable driver signature verification.

3.2.1.3. Driver Installation

There are two procedures to install the device driver:

- Install through user interface installer
- Install manually

Installing the Colorbar Demo Device Driver through the User Interface Installer

The Colorbar Demo device driver is installed during the installation of the user interface as described in the following section.

The Installer provides a standard packaging format for applications and a standard method for customizing the applications. The installer helps to install the CrossLink-NX PCIe Bridge Board Colorbar Demo application in your system.

The Framework supported version is Windows 10, WDF 1.25 or earlier. To install the Colorbar Demo device driver through user interface:

1. In the Demonstration\Windows10\Application folder or Demonstration\Windows\Application, double click setup.exe.

2. The welcome page appears. Click **Next**.

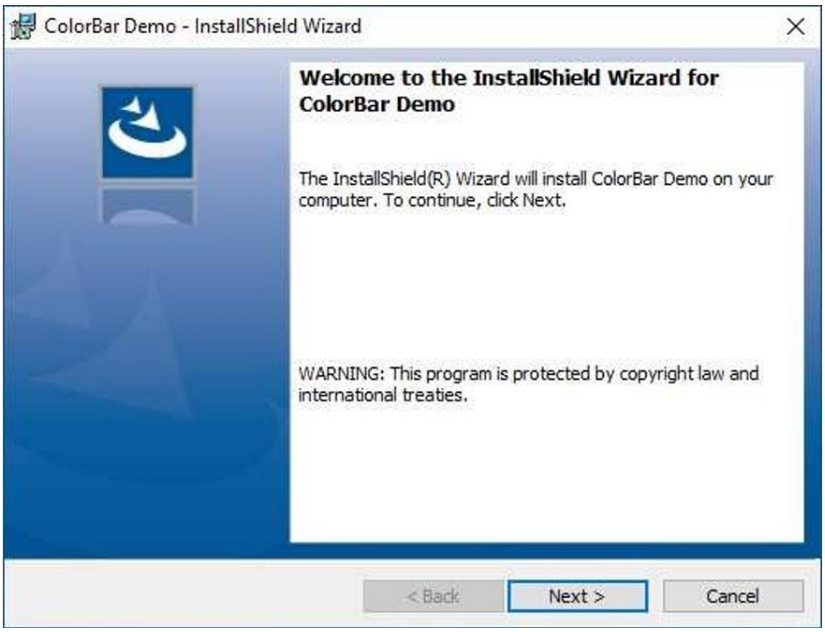


Figure 3.23. Welcome Page

3. Provide the location where you want to install the application. Click **Next**.

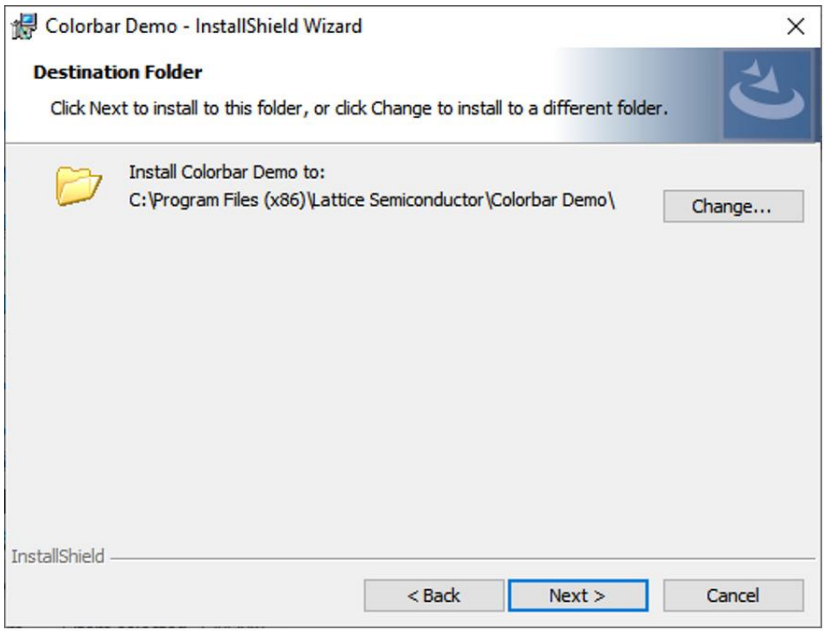


Figure 3.24. Destination Folder Page

4. The installation summary page is shown. Click **Install.**

Note: Administrative access is required to run this command.

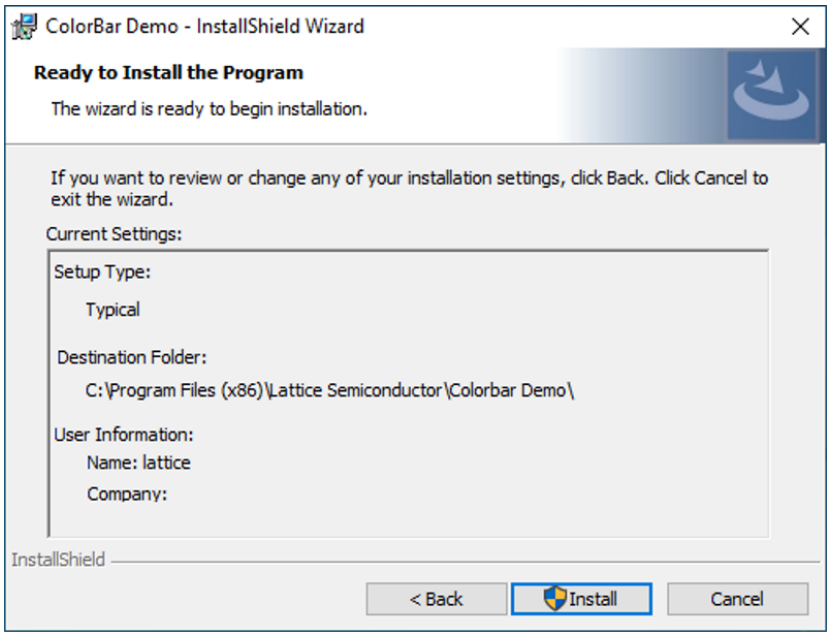


Figure 3.25. Summary Page

5. Installation of the Colorbar Demo application starts. When installation of the software is complete, the drivers are installed.

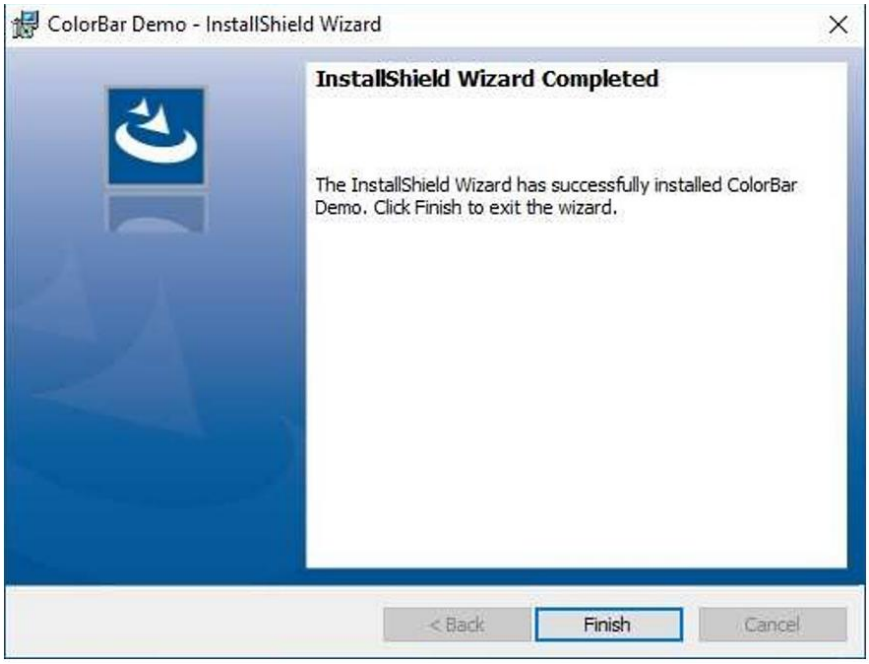


Figure 3.26. Application Installed

6. A message box appears. Click **Yes**.

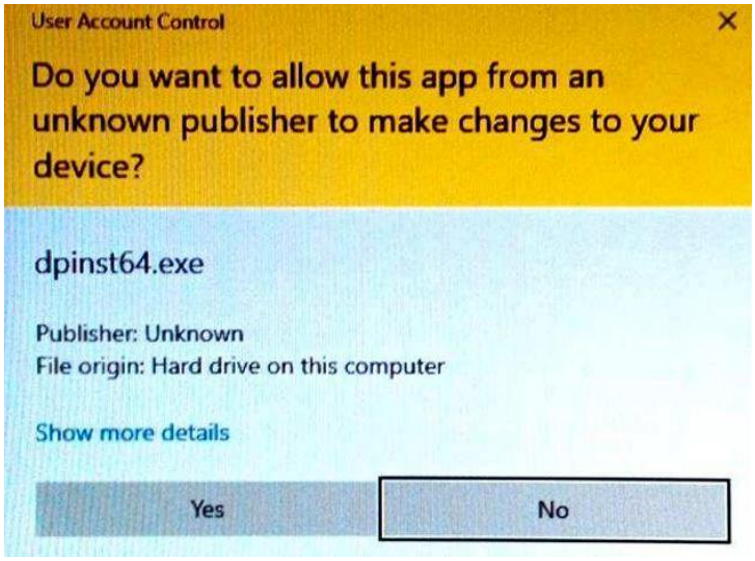


Figure 3.27. Device Configuration Prompt

7. The device driver installation wizard opens. Click **Next**.

Figure 3.28. Device Driver Installation Wizard

8. If you receive a Windows Security prompt, select **Install this driver software anyway** as shown in Figure 3.29.

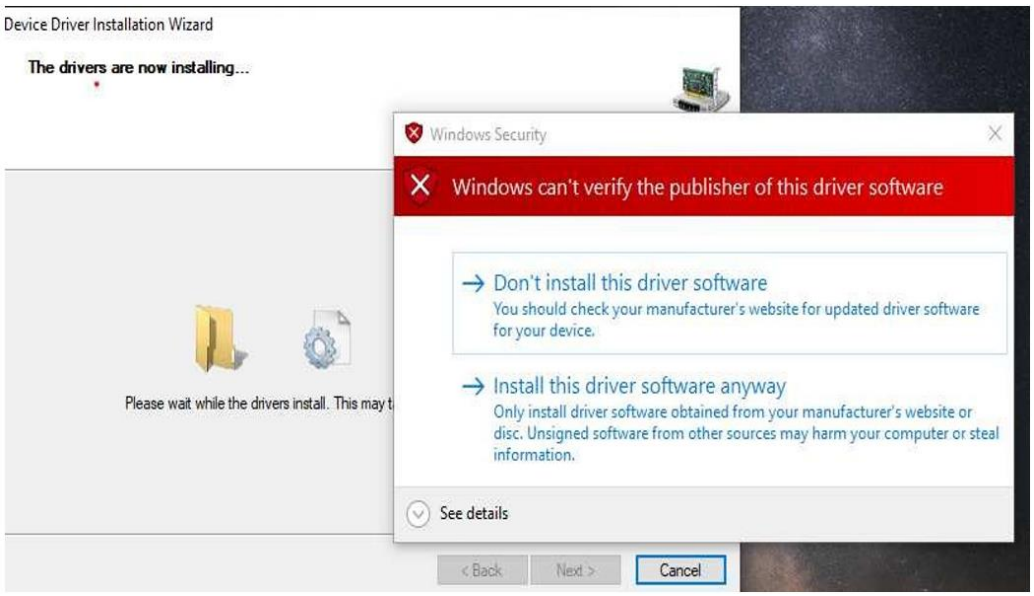


Figure 3.29. Windows Security in Driver Installation

9. If the driver is installed successfully, a message is displayed as shown in Figure 3.30.

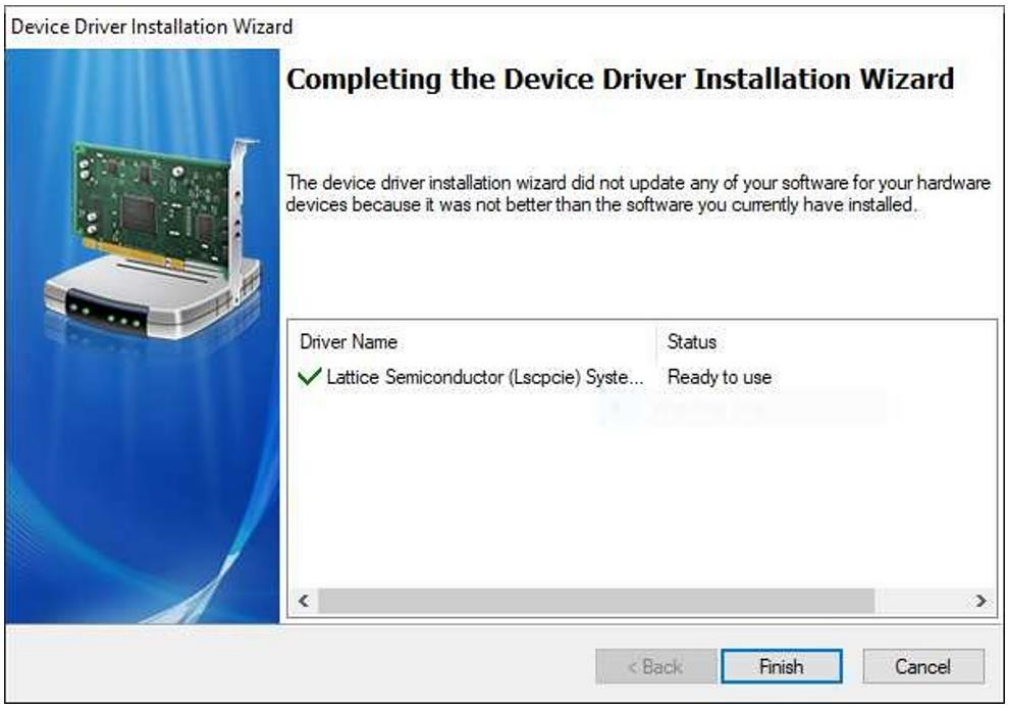


Figure 3.30. Device Driver Installation Completed

Installing Colorbar Demo Device Driver Manually

The drivers for the Colorbar Demo can also be installed manually. The drivers can be found in the *Demonstration > Windows10 > Driver* folder.

To install the device driver manually:

1. Open **Device Manager**, which shows the connected PCIe devices, as shown in Figure 3.31.

Note: If the Device Manager does not show the connected PCIe device, see the Troubleshooting section.

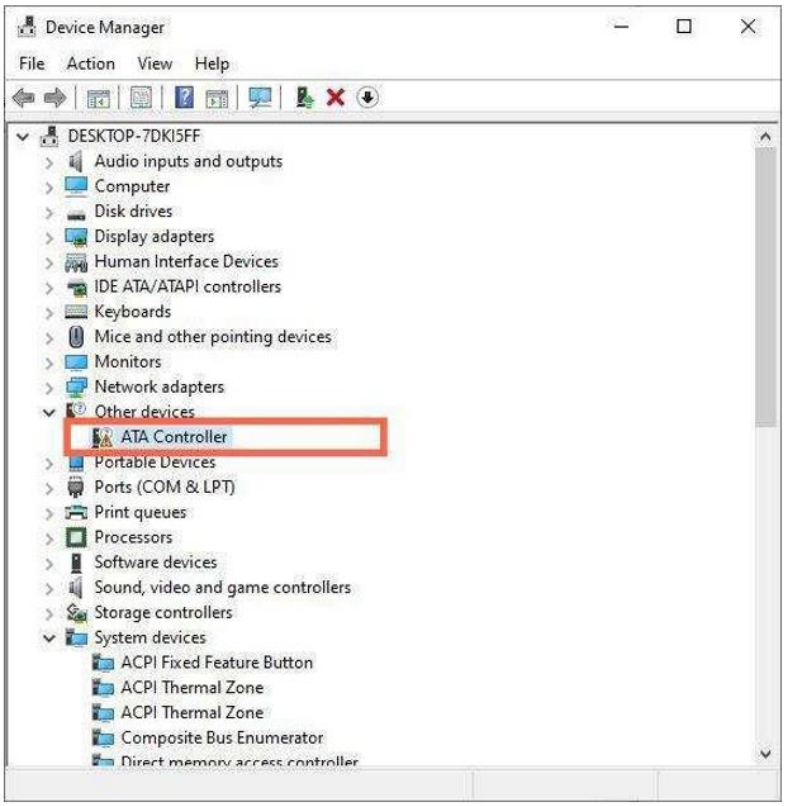


Figure 3.31. Device Manager

2. Right-click the device and select **Properties** as shown in Figure 3.32.

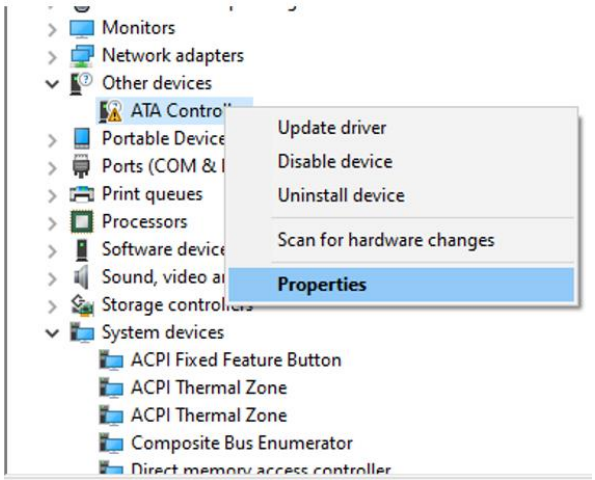


Figure 3.32. Showing Device Properties

3. The information in **Hardware IDs** is needed to install the correct driver for the corresponding device.
   a. For CrossLink-NX devices, the Hardware IDs of the Base System Device for the Colorbar Demo is *PCI\VEN_1204&DEV_9C1C&SUBSYS_E00419AA* as shown in Figure 3.33.

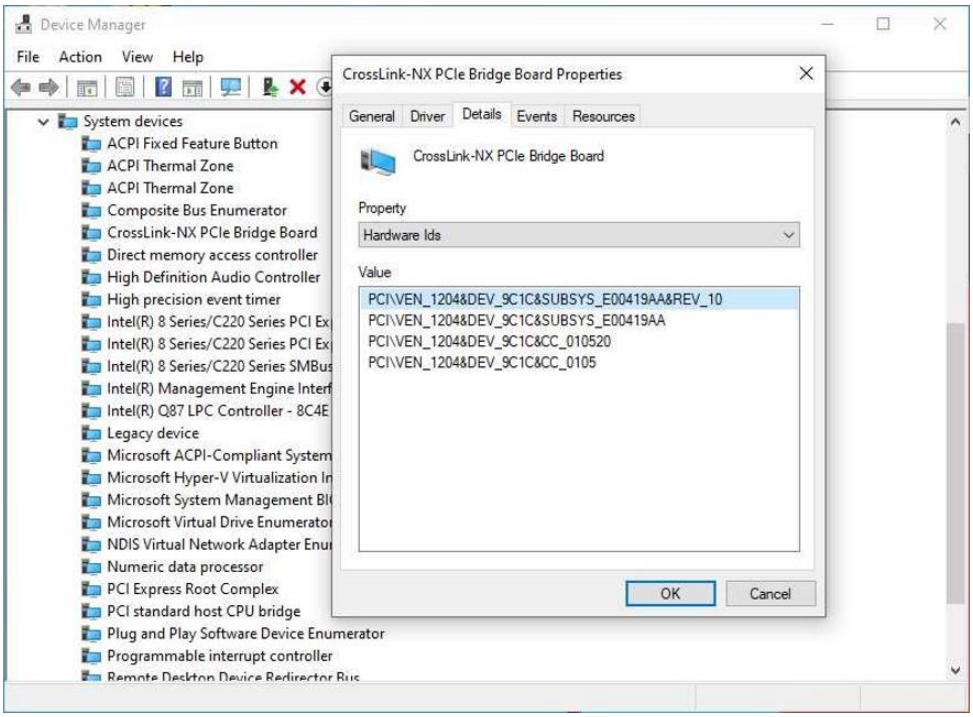


Figure 3.33. Hardware IDs of CrossLink-NX Colorbar Demo Device

   b. For Certus-NX devices, the Hardware IDs of the Base System Device for the Colorbar Demo is *PCI\VEN_1204&DEV_9C2C&SUBSYS_E00419AA* as shown in Figure 3.34.

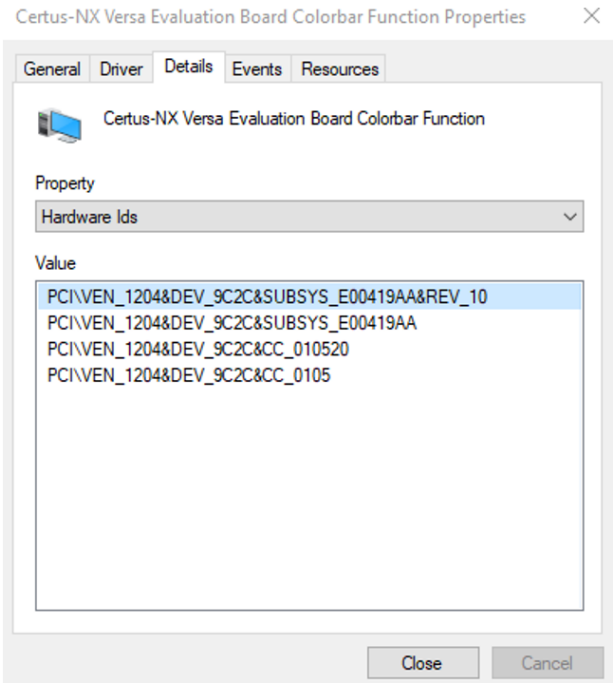


Figure 3.34. Hardware IDs of Certus-NX Colorbar Demo Device

c. For CertusPro-NX devices, the Hardware IDs of the Base System Device for the Colorbar Demo is *PCI\VEN_1204&DEV_9C40&SUBSYS_E00419AA* as shown in Figure 3.35.

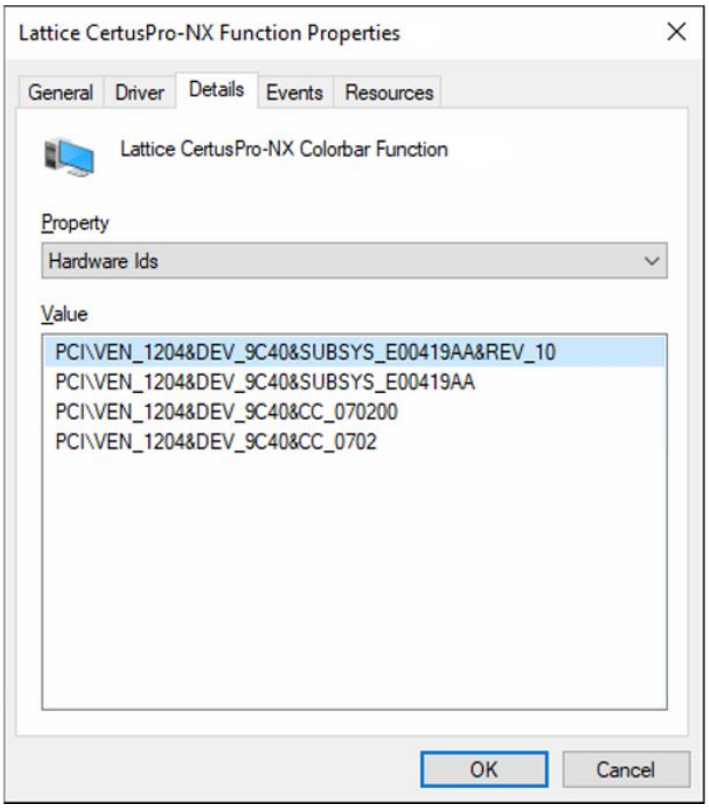


Figure 3.35. Hardware IDs of CertusPro-NX Colorbar Demo Device

4. Right click on the device and select **Update driver**.

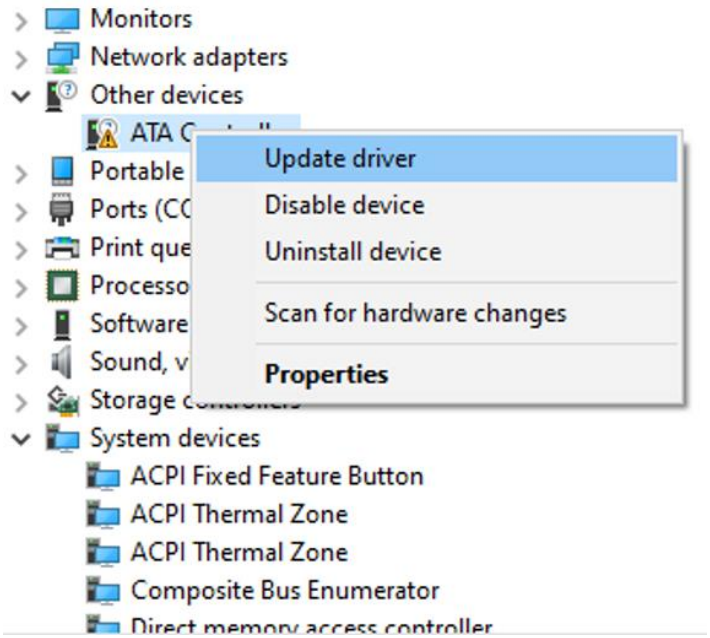


Figure 3.36. Update Driver Menu in Device Manager

5. Select the **Browse my computer for driver software** option as shown in Figure 3.37.

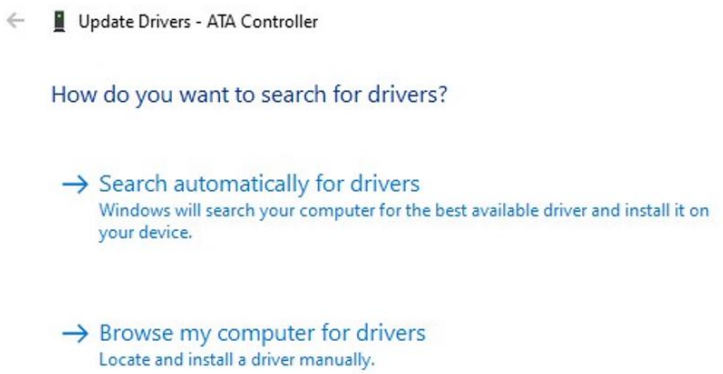


Figure 3.37. Update Driver Options

6. Browse for the Colorbar Demo device driver as shown in Figure 3.38.

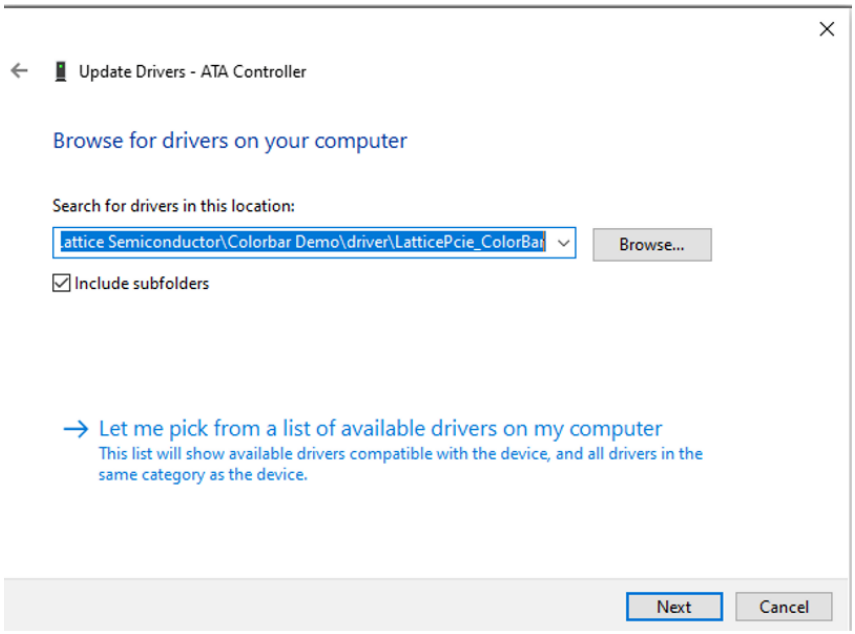


Figure 3.38. Browse the Driver for Device

7. If you receive a Windows Security prompt, select **Install this driver software anyway** as shown in Figure 3.39.

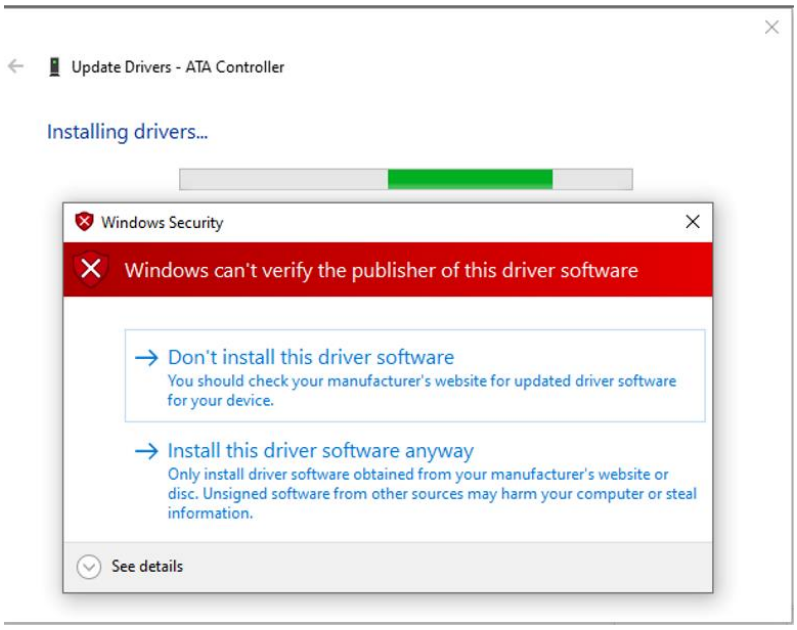


Figure 3.39. Windows Security in Device Manager

8. If the driver is installed successfully, a message is displayed as shown in Figure 3.40 for CrossLink-NX devices, Figure 3.41 for Certus-NX devices, and Figure 3.42 for CertusPro-NX devices.

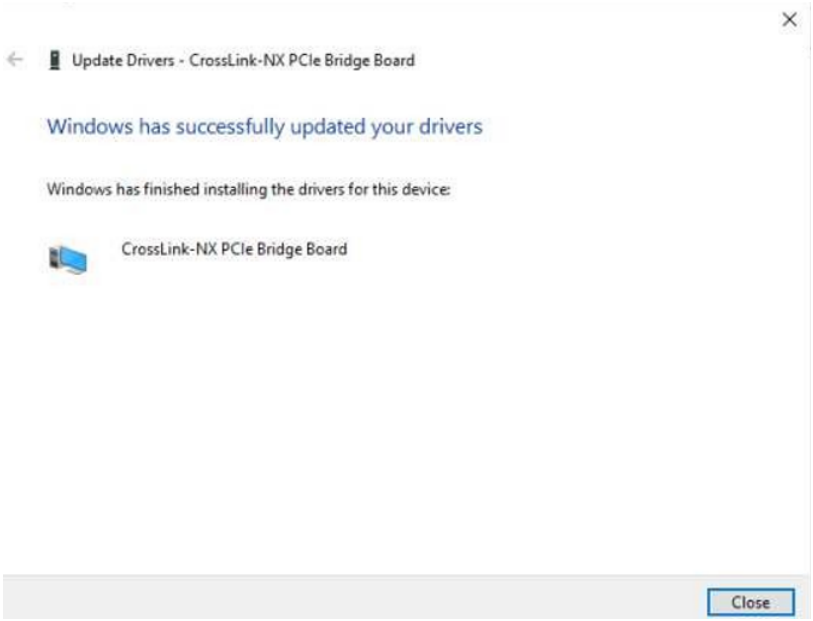


Figure 3.40. CrossLink-NX Driver Installation Status Message

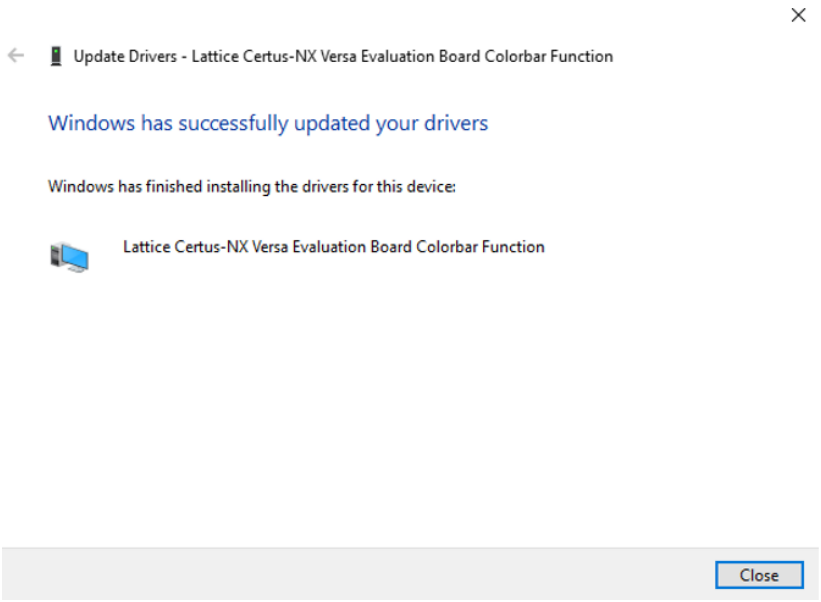


Figure 3.41. Certus-NX Driver Installation Status Message

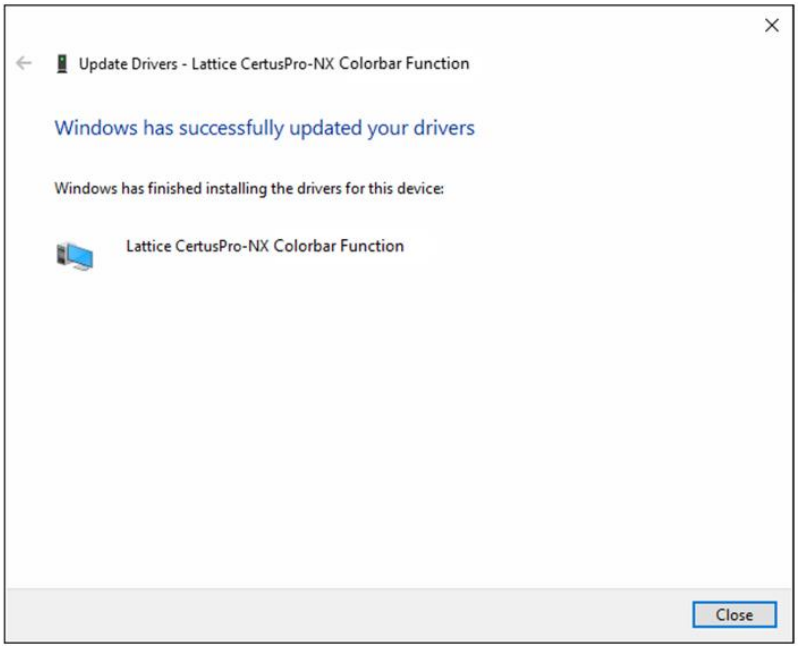


Figure 3.42. CertusPro-NX Driver Installation Status Message

9. After the driver is installed, the device driver name is displayed in Device Manager, as shown in Figure 3.43 for CrossLink-NX devices, Figure 3.44 for Certus-NX devices, and Figure 3.45 for CertusPro-NX devices.

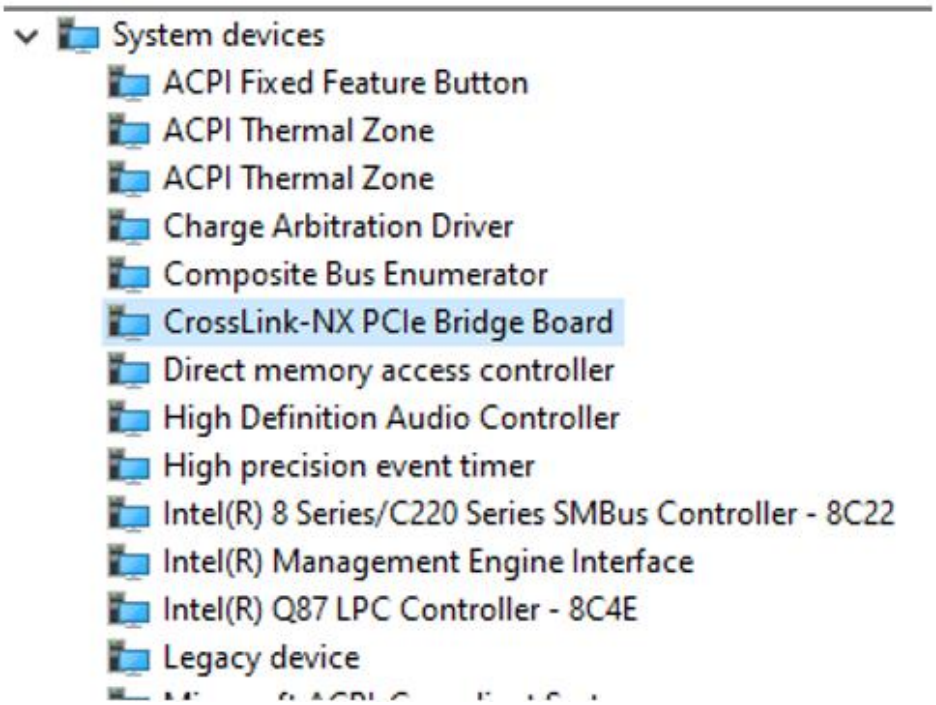


Figure 3.43. CrossLink-NX Colorbar Demo Device Name Displayed in Device Manager

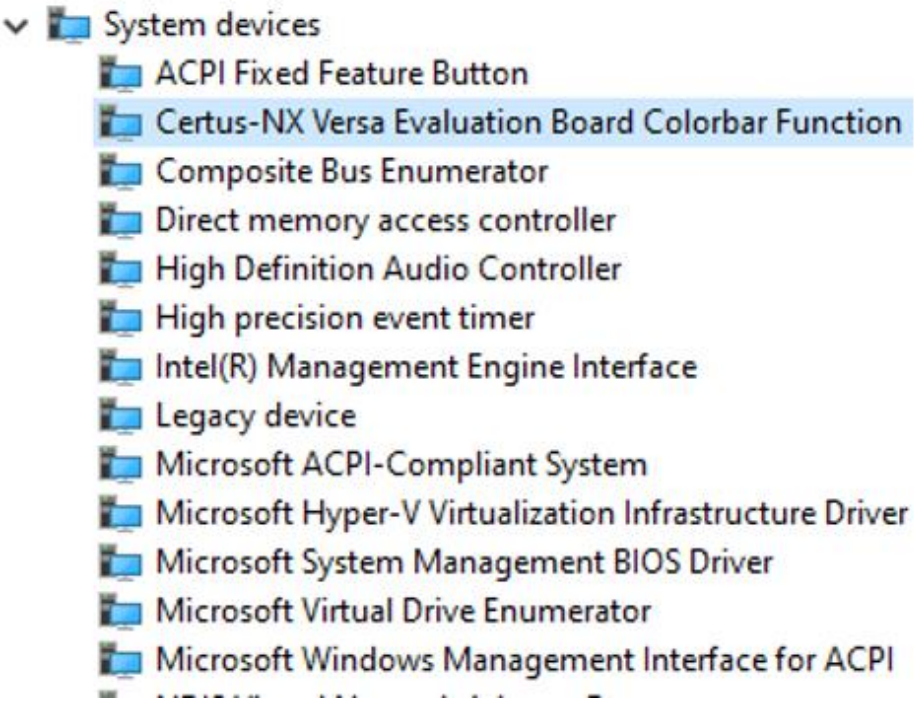


Figure 3.44. Certus-NX Colorbar Demo Device Name Displayed in Device Manager

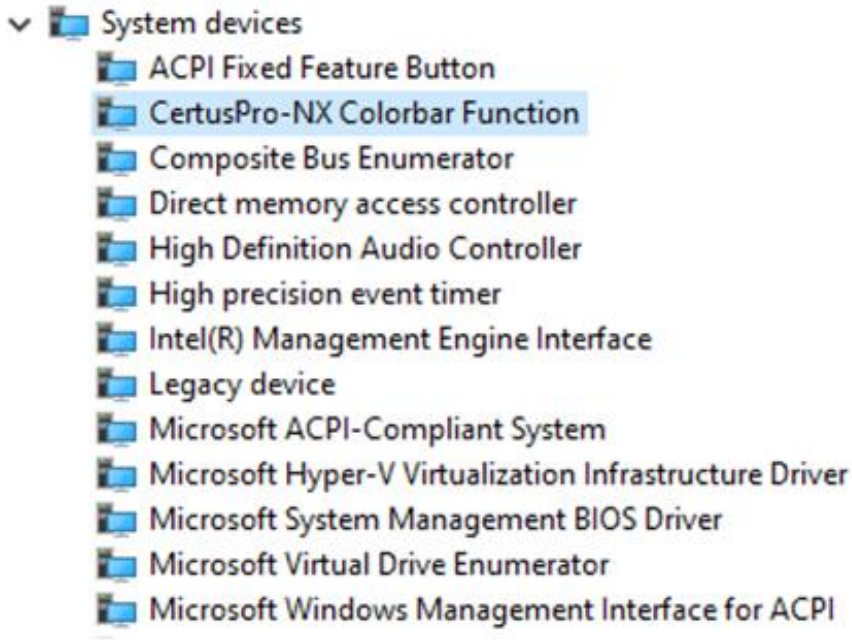


Figure 3.45. CertusPro-NX Colorbar Demo Device Name Displayed in Device Manager

3.2.2. Software Setup for Linux

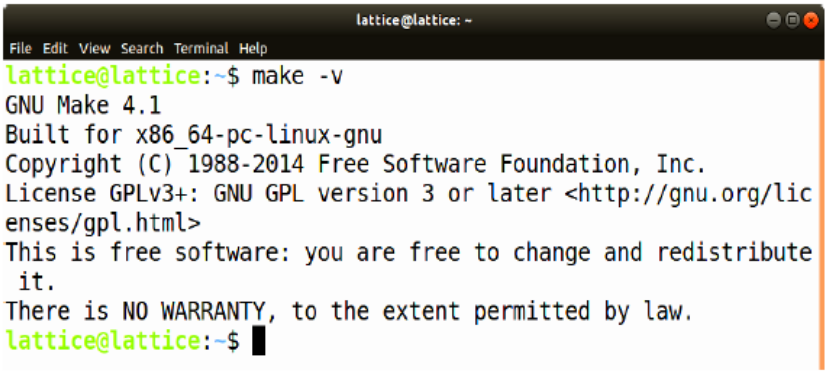
3.2.2.1. Supported Operating System

- Distribution: Ubuntu
- Description: Ubuntu 18.04.6 LTS
- Release: 18.04.6
- OS Type: 64 bit.
- Codename: bionic

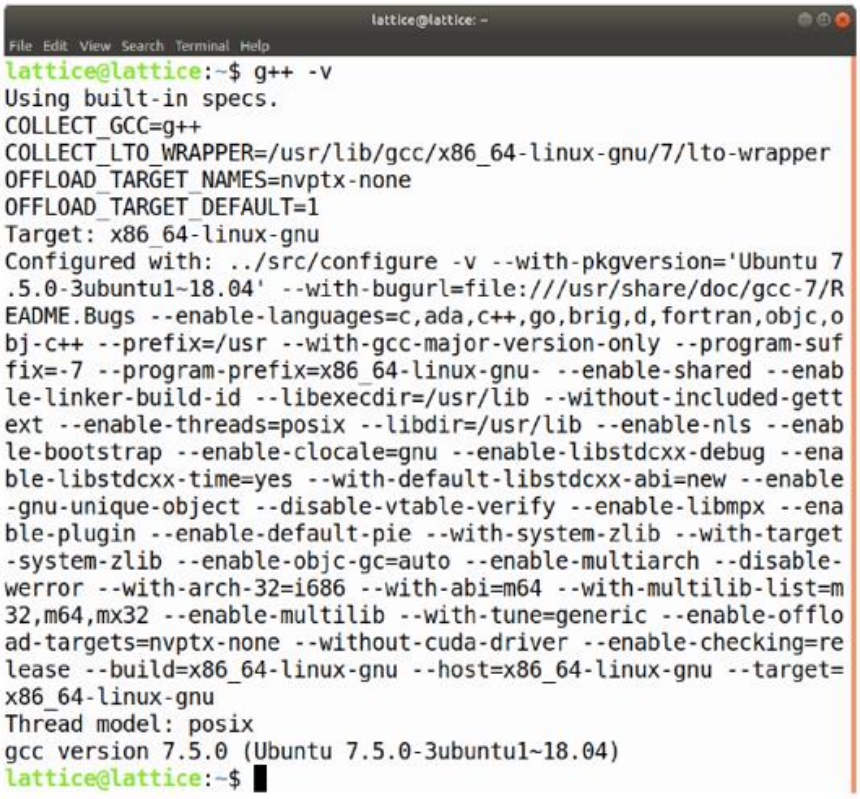
3.2.2.2. Required Packages

To check whether the required packages are installed or not, run the following commands on a terminal as shown below.

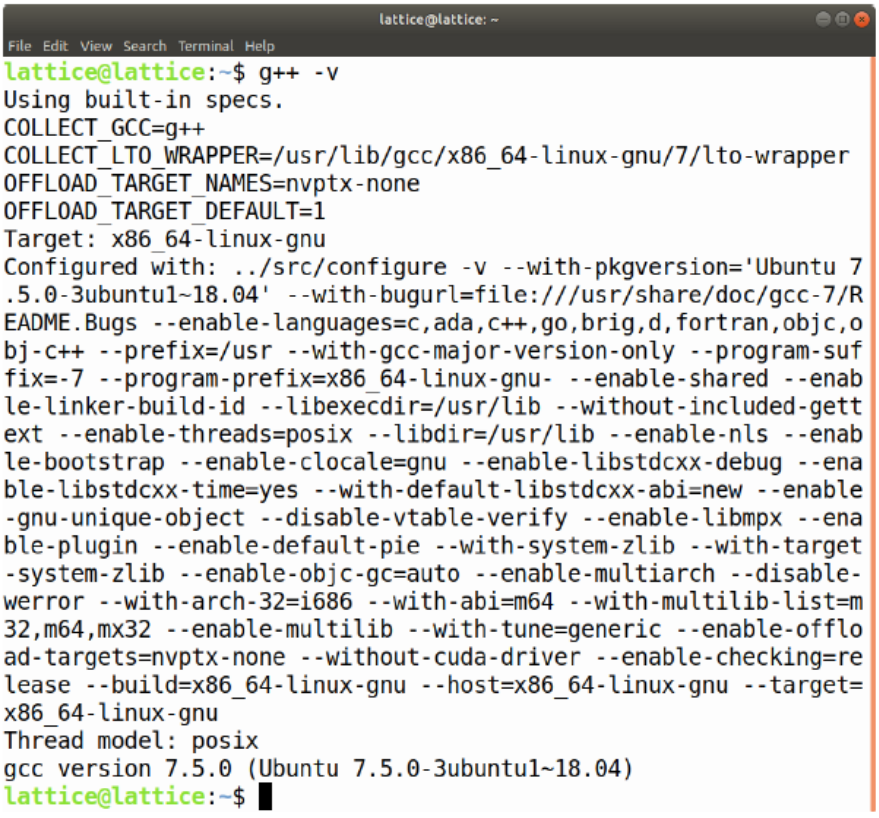
```
make
```




```
gcc
```



g++



3.2.2.3. Packages Installation Steps

Run the following commands on a terminal to install the required packages

```
sudo apt update
sudo apt install build-essential
sudo apt install flex
```

3.2.2.4. Automatic Setup and Installation

To setup the demo in automatic mode:

1. Go to the *Demonstration/Linux* directory.
2. Change the permission of the *install.sh* file by running chmod command.

```
sudo chmod 777 install.sh
```

3. Run the command below, which builds the driver and API library, install the driver, and launch the QT use interface application.

```
sudo ./install.sh
```

4. To uninstall the driver, go to the *Demonstration/Linux* directory and run the command below.

```
sudo ./uninstall.sh
```

3.2.2.5. Manual Setup and Installation

Before installing the driver, the driver must be built. To build the driver:

1. Go to the *Demonstration/Linux* directory.
2. Run the following command on terminal.

```
sudo chmod 777 -R Source_Code
```

3. Go to the *Source_Code* directory.
4. Run the following commands. This builds the driver and API library.

```
sudo make clean
sudo make
```

5. Make sure that the driver is not installed.
6. To remove an existing driver, run the following command in a terminal window:

```
sudo rmmod lcolor_bar_main
```

To install the driver:

1. Go to the drv_src/color_bar_drv/ directory.
2. Run the following commands.

```
sudo insmod lcolor_bar_main.ko
```

3. To launch the user interface application, go to the *app_src/gui/deploy/* directory and run the command below.

```
sudo ./color_bar_demo.sh
```

4. Demo Design Overview

4.1. Theory of Operation

The demo runs on a standard x64 PC and accesses the FPGA Board installed in a PCIe slot. Figure 4.1shows the relationship of the hardware and software components of the demo. The purpose of the demo is to show the streaming performance of the Lattice PCI Express SERDES hardware and PCI Express Endpoint IP core.

In the colorbar demo, image data consisting of different colored bars is transferred from theFPGA to PC memory, where software then displays it on the screen. The bars are rotated after every 50 frames to get a visualization of the moving image. The resolution of the image that can be selected is 800 X 600, 1920x1080 and 3840x2160 and is displayed on the screen at a certain frame rate. Displaying a moving image over a screen is a quick way to illustrate that data is being transferred and that it can run consistently.

After the hardware has transferred the pixel data, the API call returns, and then the software displays the image. This process is repeated in a loop until stopped by the user.

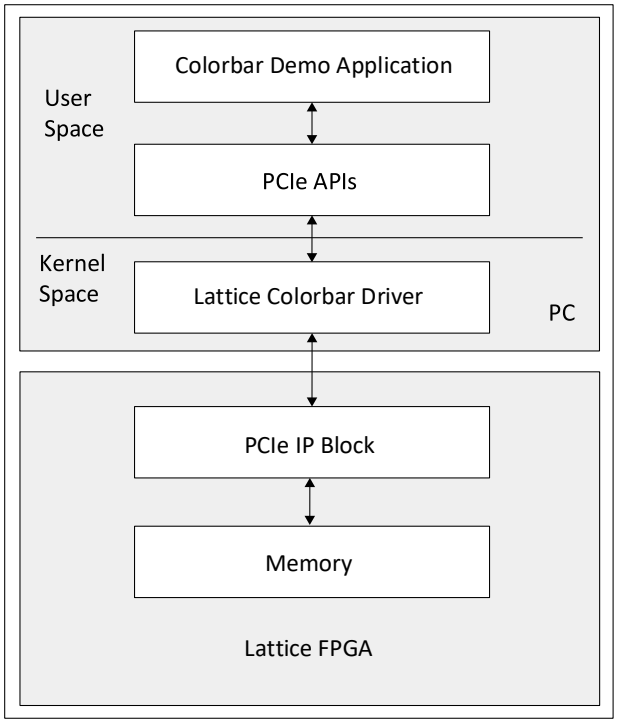


Figure 4.1. Relationship between the Hardware and Software Components

4.2. Design Overview

A user interface application is provided for demonstrating the PCIe Colorbar demo. Application software is developed using a layered architecture consisting of the following layers:

- User interface application
- Driver API
- Device Drivers
- Device Hardware (FPGA Design)

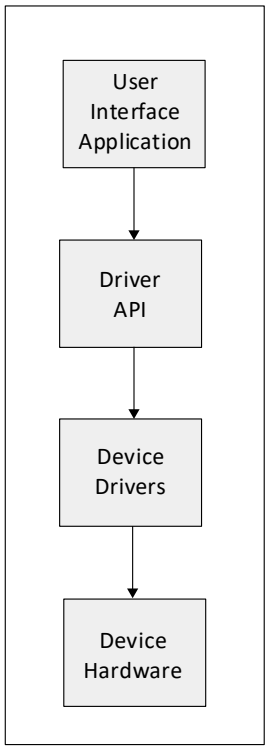


Figure 4.2. PCIe Colorbar Demo SW Design

4.2.1. User Interface Application

The user interface application uses the driver API to communicate with the device hardware. The Driver API is a C++ dynamic library which provides an interface to access the hardware.

4.2.2. Driver API

On Windows, the *MemRwApis.dll* library is a DLL that bridges the user space demo applications to the kernel space driver code and provides routines to control the IP modules.

On Linux, the library is named *libmem_rw.so*.

4.2.3. Device Drivers

The Hardware Drivers provide access to the CrossLink-NX PCIe bridge board. On Windows, the *LatticePcie_BMD_Colourbar.sys* driver supports the PCIe Colorbar demo. On Linux, the *lcolor_bar_main.ko* driver supports the demo.

4.2.4. Device Hardware (FPGA Design)

Figure 4.3 shows the top-level architecture of the FPGA design. DMA support is an option provided by the Lattice PCIe soft IP to enable more efficient data transfer when endpoint is acting as initiator or master. This feature is only available when the AHB-Lite data interface is selected.

To support DMA operations, the FPGA design has two queues implemented in local memory, a descriptor queue and a status queue. To initiate a DMA transfer, the host writes a descriptor into the descriptor queue, which contains the source address, destination address, length, and direction of data transfer. When the operation is completed or aborted, the status of the transfer (including done status, error status, length of transfer, and data address offset) is written into the status queue. The DMA descriptor and status queues can be read via a register space implemented in the design.

A register space is implemented in the design to configure the DMA and to get the status of transfer and throughput achieved. A FIFO wrapper block is used in the design to instantiate a video timing generator and bar pattern source to generate the bar pattern for the DMA write operation. As the video generation will take place throughout regardless of the case whether AHB is ready to receive the data or not, a ping-pong memory wrapper is instantiated in the design and acts as a buffer to combat this issue.

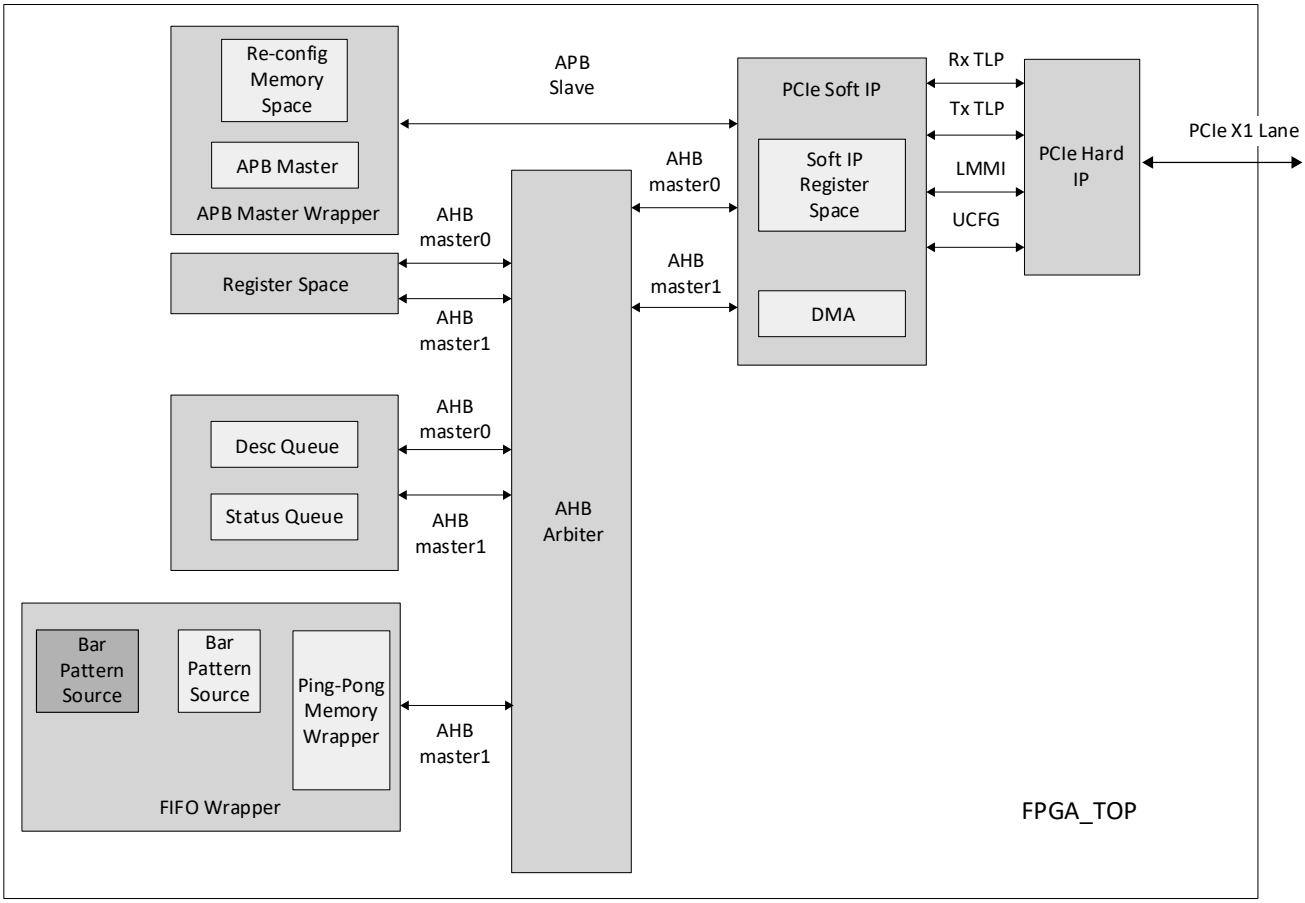


Figure 4.3. Top Level Architecture of FPGA Design

5. Application Overview

The PCIe Colorbar demo is intended to show video data transmitting capability between the Lattice FPGA and a host system. The application takes video frame data continuously from the FPGA and displays it in real time. The video frame being generated by the FPGA is in RGB16 format.

5.1. Running the PCIe Colorbar Demo Application

This section describes how to run the demo software after installation. On Linux, the demo can be launched by running the *./color_bar_demo.sh script*, which is located in the *Demonstration/Linux/Source_Code/app_src/gui/deploy* directory. On Windows, click the Bridge Board Colorbar Demo Icon.

The graphical user interface opens the PCI Express Test Application software with the Device Info tab selected as shown in Figure 5.1.

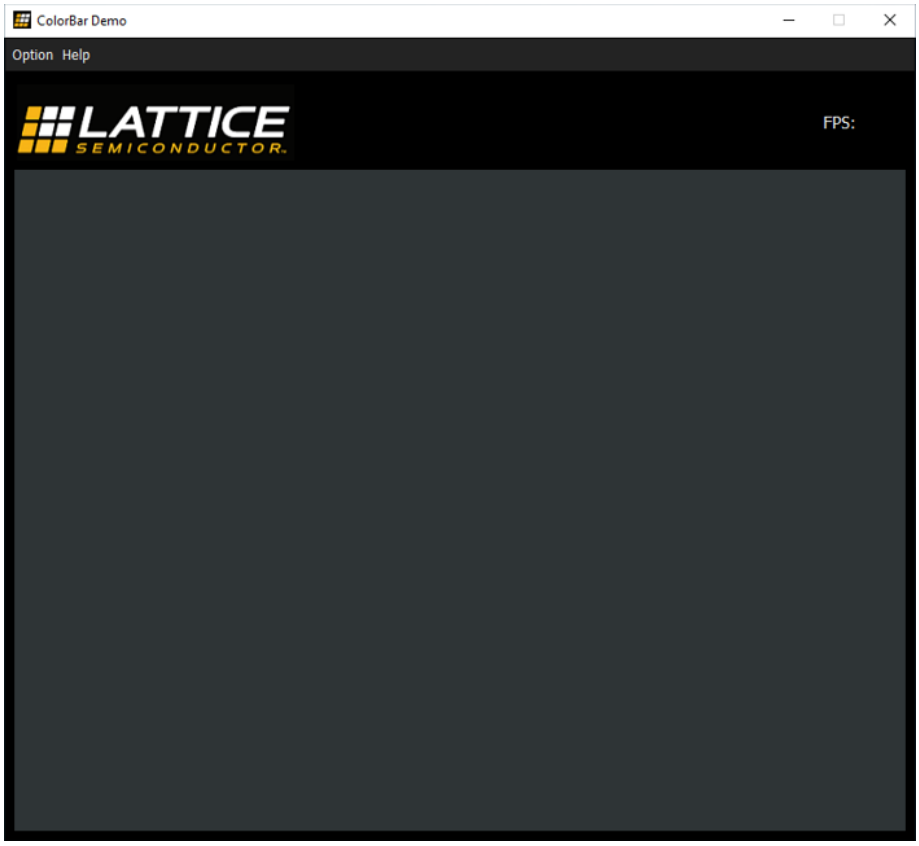


Figure 5.1. PCIe Colorbar Demo Device Info Tab

5.2. Option

The option in menu bar gives the PCIe Info, Play, stop, set video parameter and exit options.

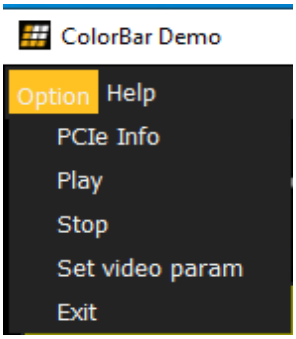


Figure 5.2. Drop-Down Menu

5.2.1. PCIe Info

This is used to show the PCIe interface capability information, and is read-only. Note that some of the information displayed may vary between different FPGA devices.

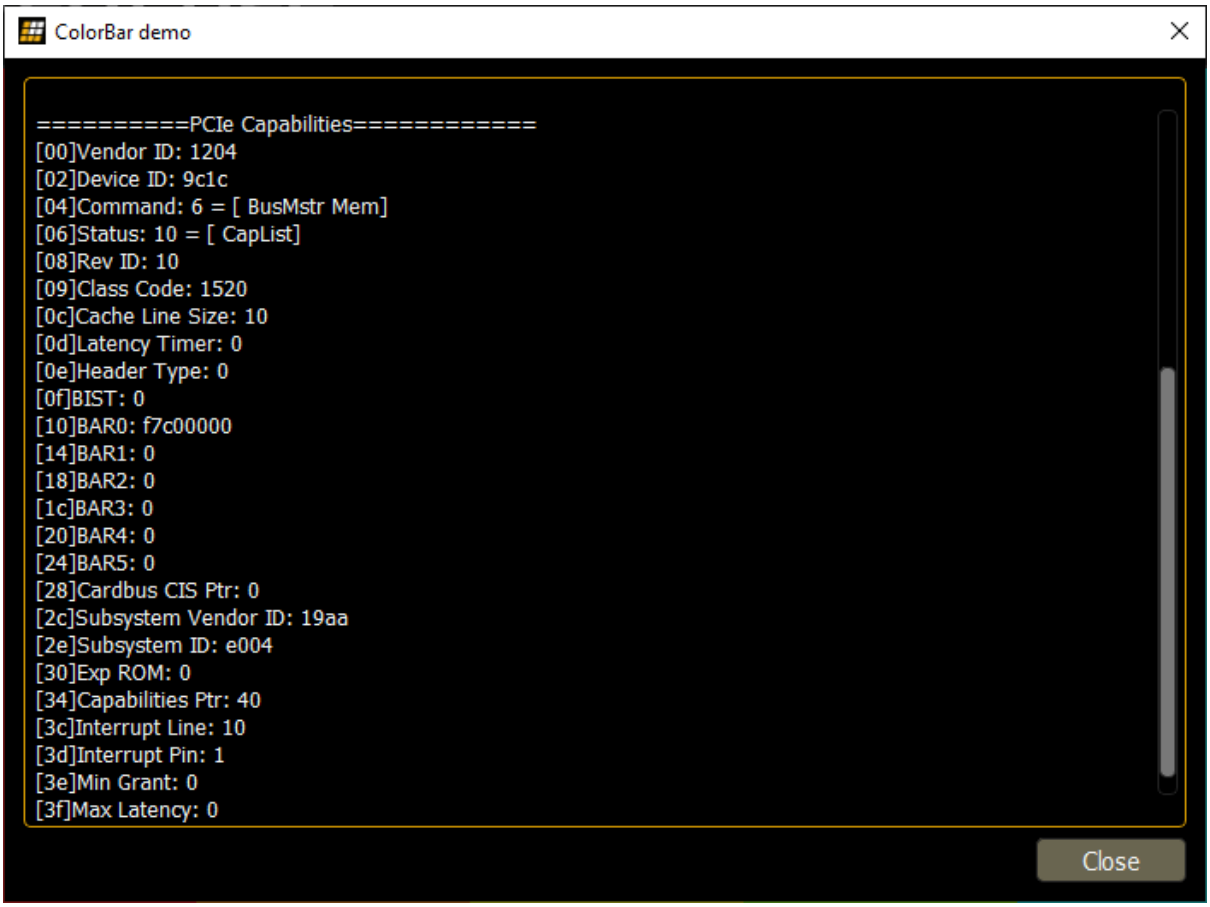


Figure 5.3. PCIe Device Info

5.2.2. Play

This is used to play the colorbar demo. When the demo is running, rotating video frames should appear as shown in Figure 5.4.

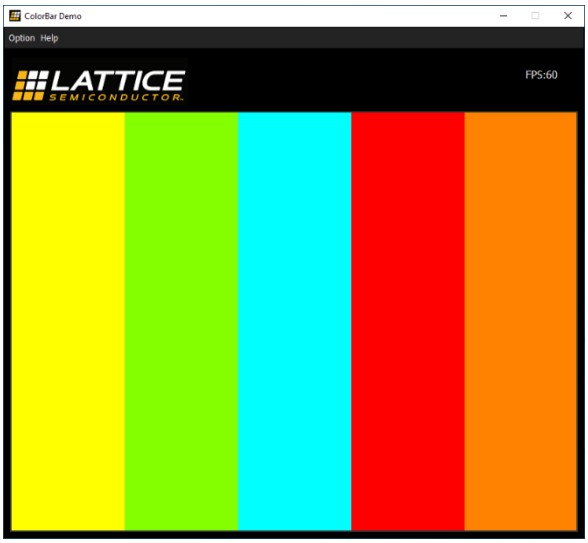


Figure 5.4. Video Frames

5.2.3. Stop

This is used to stop colorbar demo.

5.2.4. Set Video Param

This is used to configure the video resolution and color information as shown in below image. At present, these options are not configurable.

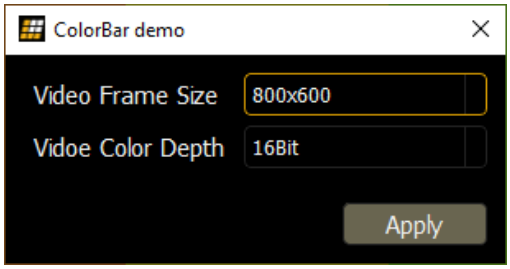


Figure 5.5. Set Video Param

5.2.5. Exit

This is used to exit from the application.

5.2.6. About

Gives a description about the application version number, build date etc.

6. Importing and Building the FPGA Demonstration

The package includes the PCIe IP, .bit file, and synthesis projects using Lattice Radiant Software.

6.1. Hardware Directory Structure

The Hardware folder inside the package contains the following subfolders. ./CL_NX_BridgeBoard_ColorbarDemo/ or ./CertusNX_PCIe_ColorBarDemo/ or /CertusPro-NX_PCIe_Colorbar_Demo/

- IP – Contains the pre-generated IPs used in the design. These IPs can be configured by clicking the .ipx file after opening the project in Radiant.
- Implementation – Contains the Lattice Radiant project (.rdf) file, constraints file (pdc), and implemented design and bit files.
- Source – Contains RTL files required for the design.

6.2. Building Lattice Radiant Project

To generate the bit stream file:

1. Open the Lattice Radiant software.
2. Click **Open project** and browse to the .rdf file

- For CrossLink-NX devices, the file is *LIFCL40_PCIeColorbarDemo.rdf*, which is located in the *Hardware/CL_NX_BridgeBoard_ColorbarDemo/Implementation/LIFCL40_PCIeColorbarDemo* folder
- For Certus-NX devices, the file is LFD2NX_PCIeColorBarDemo.rdf, which is located in the *Hardware/CertusNX_PCIe_ColorBarDemo/Implementation/LFD2NX_PCIeColorBarDemo* folder
- For CertusPro-NX devices, the file is *Implementation_1X1.rdf, Implementation_1X4.rdf, Implementation_2X1.rdf and Implementation_3X4.rdf*, corresponding to different link speed and link width which is located in the

  Hardware/CertusPro-NX_PCIe_Colorbar_Demo/PCIE_ColorBar_demo_Gen1_x1/Implementation1x1 folder or

  Hardware/CertusPro-NX_PCIe_Colorbar_Demo/PCIE_ColorBar_demo_Gen1_x4/Implementation1x4 folder or

  Hardware/CertusPro-NX_PCIe_Colorbar_Demo/PCIE_ColorBar_demo_Gen2_x1/Implementation2x1 folder or

  Hardware/CertusPro-NX_PCIe_Colorbar_Demo/PCIE_ColorBar_demo_Gen3_x4/Implementation3x4 folder

3. Once your project loads, click **Task Detail View**. This shows a list of actions that Lattice Radiant will perform to build the .bit file.
4. Select the files and reports that you want to generate.

   Note: The options needed to regenerate the .bit file are selected by default.
5. After selecting your preferred reports, click **Run All**.

This creates a .bit file with your project's current name in the *impl_1* folder.

7. Troubleshooting

7.1. SPI Flash Update

- If you are getting a verification error while dumping the .bit file, try changing the TCK frequency to a value greater than 4. The TCK Divider Setting option is in the Cable Setup dialog box of the Lattice Radiant Programmer Window, as shown in Figure 7.1. Restart programming by clicking the Program button.

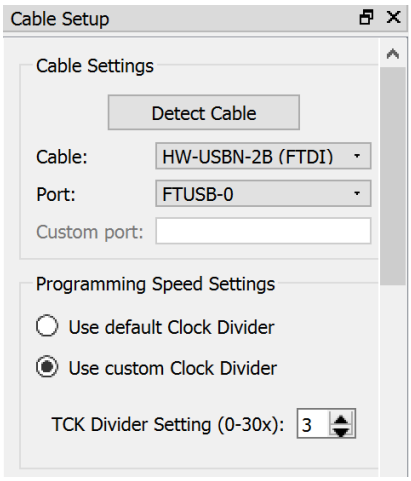


Figure 7.1. TCK Frequency Setting

- If the verification error problem still occurs, press and hold the PROGRAMN push button on the board before clicking the Program button.
- If the device is not getting detected on port FTUSB-0. Click Detect cable and select the port FTUSB-1.

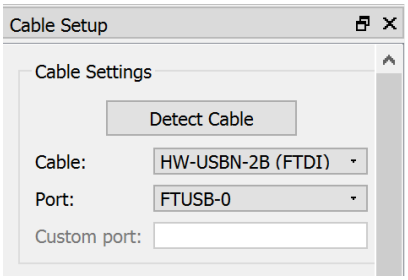


Figure 7.2. Port Selection

7.2. Driver Installation and User Interface Launch for Windows

7.2.1. Problem in Driver Installation

- If the hardware IDs are not found in Device Manager, make sure that the board is properly installed in a PCIe slot.
- Make sure that the board has a valid PCIe bitstream file loaded in the SPI Flash. Shut down the system and try another slot. If the board is present, check the Properties and the Resource tab to verify if memory is assigned to it.

  Also, verify if the Vendor ID and Device ID are valid, as seen by Windows Plug-n-Play. If the values are invalid, perhaps the bitstream file is corrupt and needs to be reloaded into SPI flash. If the PCIe board is shown in the list of Device Manager, install the driver manually.
- If the driver is not being installed even though the device is present in the Device Manager, make sure that the driver signature enforcement is disabled permanently. If not, follow the steps described in the Disabling Driver Signature Enforcement Permanently section.

7.2.2. Problem with Launching User Interface

Make sure that the driver is installed properly. Otherwise, the Device Info section of the user interface displays the message shown in Figure 7.3.

The *Driver Open [FAILED]* message is displayed if the driver is not installed properly. The Device information is present if the drivers are installed properly.

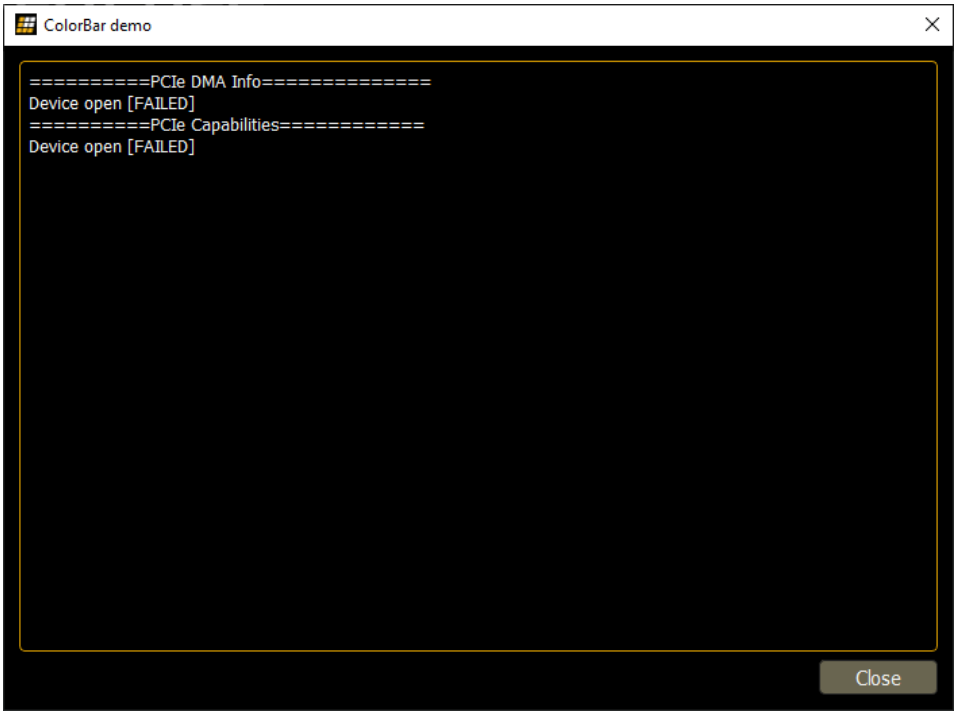


Figure 7.3. User Interface with No Device Driver

If the device is not inserted in the system or the device driver is not installed, any read or write operation is not executed. A sample error message is shown in Figure 7.4.

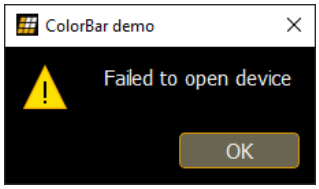


Figure 7.4. Error Message

If any such error message occurs, make sure that the device is properly inserted in the system. After that, verify that drivers are installed. When all the issues are resolved, restart the user application.

It is also possible that there could be a driver conflict due to another demo design being installed. In this case, manual deletion of the driver may be needed. It is recommended to uninstall the device from the system and delete the *'LatticePcie_BMD_Throughput.sys'* and *'LatticePcie_BMD_Colourbar.sys'* drivers from the *Windows\System32\drivers* directory, and then reinstall the Colorbar demo from the start.

If the Colorbar demo shortcut icon is accidentally deleted, then the application can be launched by double clicking on *color_bar_demo.exe* file present in [INSTALLATION FOLDER]/Lattice Semiconductor/Colorbar Demo/bin folder.

7.3. Driver Installation User Interface Launch for Linux

This section describes trouble shooting steps for Linux.

7.3.1. Problem in Driver Loading

After launching the user interface, if the driver is not loaded properly, a *Driver open[Failed]* message is displayed in the Info section.

- Make sure that the board is properly connected to the PC.
- Run the command below in a terminal window to show the list of PCIe devices connected to PC.

  ```
  sudo lspci -vnm
  ```

  Make sure that the Lattice device is present by checking the device and vendor IDs, as shown in Figure 7.5 for CrossLink-NX devices, Figure 7.6 for Certus-NX devices, and Figure 7.7 for CertusPro-NX.

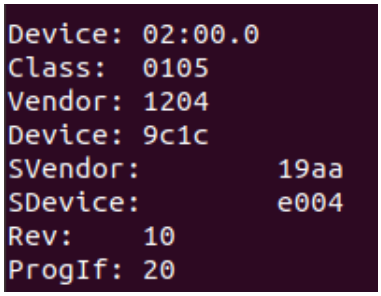


Figure 7.5. lspci -vnm for CrossLink-NX Output Image

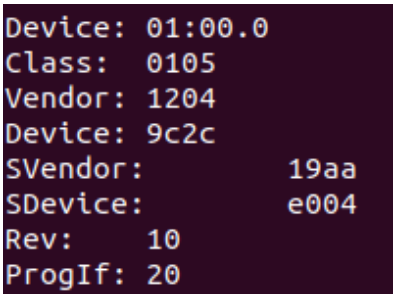


Figure 7.6. lspci -vnm for Certus-NX Output Image

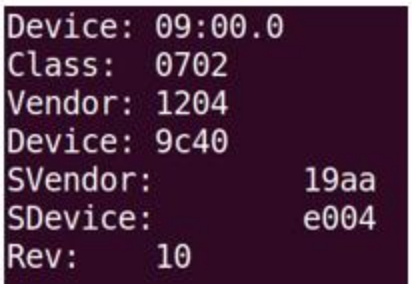


Figure 7.7. lspci -vnm for CertusPro-NX Output Image

- If the device is present, perform the manual setup and installation steps one by one.
- If the driver does not build properly, check for any software or kernel dependencies.

7.3.2. Problem with User Interface Launching

1. Go to the *Demonstration/Linux* directory. Check all content of this directory. Default content is shown in Figure 7.8. Check the permissions of these file.

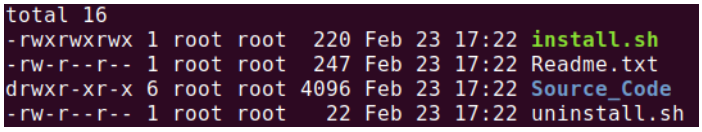


Figure 7.8. Content List of Demonstration/Linux Directory.

2. The script files should have *execute* permission – to set permissions on the file, run the command below in the terminal and try to run the application again.
```
sudo chmod 777 filename
```
3. If the user interface still does not launch, change directory into the Source_Code directory and verify that all files are present in this directory, according to Figure 7.9.

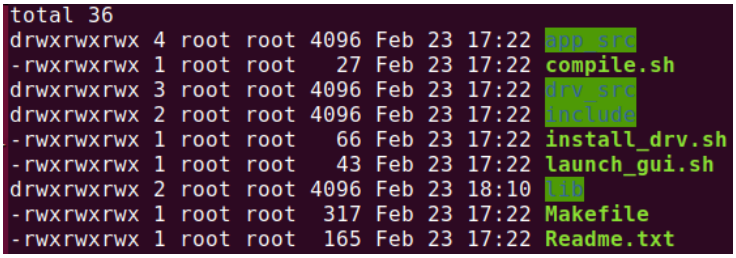


Figure 7.9. Content List of Software/Linux Directory

4. If all files are present in this directory, change directory to *app_src/gui/deploy* and try to run the command below directly.
```
sudo ./color_bar_demo.sh
```

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

Revision History

Revision 1.2, May 2023

Section	Change Summary
All	Updated the contents to be applicable to CertusPro-NX devices.
Technical Support Assistance	Added reference link to the Lattice Answer Database.

Revision 1.1, March 2022

Section	Change Summary
All	• Changed document title to PCIe Colorbar Demo for Lattice Nexus-based FPGAs. • Updated the contents to be applicable to Lattice Nexus-platform devices, specifically CrossLink-NX and Certus-NX for this release.

Revision 1.0, January 2022

Section	Change Summary
All	Initial release.

www.latticesemi.com