Ti375 PCIe® Early Access Board User Guide

Ti375-PCIeEA-BRD-UG-v1.1
August 2024
www.efinixinc.com

Contents

Introduction **3**
- What's in the Box? 3
- Register Your Kit 3
- Download the Efinity® Software 3
- Installing the Windows USB Drivers 4
- Installing the Linux USB Driver 4

Board Functional Description **5**
- Features 5
- Overview 6
- Power On 7
- Reset 7
- Configuration 7
- Clock Sources 8
- Headers 8
  - Header J124 (ATX Power Supply) 8
  - Header J34 and J35 (JTAG) 9
  - Header USB1 (USB FTDI FT2232H) 9
  - PCIe Edge Card Connector 9
  - PCIe Edge Card Socket 9

Ti375 PCIe Early Access Board Example Design **10**
- Set Up the Hardware 11
- Run the Demonstration Design 12
  - Read the PCIe Link Status 12
  - Reset the Design and PCIe Link 13
  - Check the Bandwidth 13
  - Write to PCIe Registers through APB Interface 14
  - View Traffic Waveform 14
- Debugging Transceivers (Beta) 14
  - Launching the Transceiver Debugger 15
  - Using the Transceiver Debugger 15
  - Interpreting the Results 16

Creating Your Own Design **17**

Restoring the Demonstration Design **17**

Revision History **17**

Introduction

Thank you for choosing the Ti375 PCIe Early Access Board (part number: Ti375-PCIE-EA). This early access board lets you see the Ti375 PCIe transceivers in action and get a sneak peak at the capabilities of the Ti375 FPGA. You can connect the Ti375 PCIe Early Access Board's PCIe edge card connector and edge card socket with a PCIe extender cable to do a loopback test. You can connect the board to your computer with a USB cable and use the Efinity® Transceiver Debugger to test the link quality. Additionally, you can connect this board to a motherboard with the PCIe extender cable.

Important: This board is truly early access: it does not have all the features you would expect from one of our development kits. However, for those of you who want to get your hands on the transceiver-capable silicon today, the Ti375 PCIe Early Access Board provides that opportunity.

The Ti375 PCIe Early Access Board has two 256 Mbit SPI NOR flash devices for SPI active configuration.

Warning: The board can be damaged without proper anti-static handling.

What's in the Box?

The box includes:

- Ti375 PCIe Early Access Board
- Micro-USB cable
- 12 V, 5 A power adapter with ATX 6-pin converter cable
- PCIe crossed extender cable

Register Your Kit

When you purchase an Efinix development kit, you also receive a license for the Efinity® software, plus 1 year of software upgrades and patches. After the first year you can request a free maintenance renewal. The Efinity® software is available for download from the Support Center.

To download the software, first register at our Support Center (**https://www.efinixinc.com/register**), then register your development kit.

Download the Efinity® Software

To develop your own designs for the Ti375 FPGA on the board, you must install the Efinity® software. You can obtain the software from the Efinix Support Center under Efinity Software (**www.efinixinc.com/support/**).

The Efinity® software includes tools to program the device on the board. Refer to the Efinity® Software User Guide for information about how to program the device.

Learn more: Efinity® documentation is installed with the software (see **Help** > **Documentation**) and is also available in the Support Center under Documentation (**www.efinixinc.com/support/**).

Installing the Windows USB Drivers

The Ti375 PCIe Early Access Board development board has an FTDI FT2232H chip to communicate with the USB port.

Note: If you have another Efinix board and are using the Ti375 PCIe Early Access Board, you must manage drivers accordingly. Refer to **AN 050: Managing Windows Drivers** for more information.

On Windows, you use software from Zadig to install drivers. Download the Zadig software (version 2.7 or later) from **zadig.akeo.ie**. (You do not need to install it; simply run the downloaded executable.)

To install the driver:

1. Connect the board to your computer with the appropriate cable and power it up.
2. Run the Zadig software.
3. Choose **Options > List All Devices**.
4. Select **Titanium Ti375 PCIe EA Board (Interface 1)**.
5. Select **libusb-win32** in the **Driver** drop-down list.
6. Click **Replace Driver**.
7. Close the Zadig software.

Installing the Linux USB Driver

The following instructions explain how to install a USB driver for Linux operating systems.

1. Disconnect your board from your computer.
2. In a terminal, use these commands:

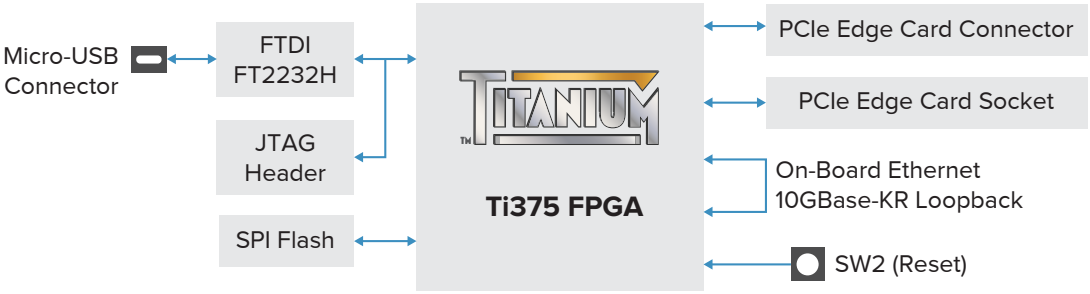
```
> sudo <installation directory>/bin/install_usb_driver.sh
> sudo udevadm control --reload-rules
```

Note: If your board was connected to your computer before you executed these commands, you need to disconnect it, then re-connect it.

Board Functional Description

The Ti375 PCIe Early Access Board contains a variety of components to help you build designs for the Titanium Ti375 FPGA.

Figure 1: Ti375 PCIe Early Access Board Block Diagram



The main purpose of this board is to demonstrate the PCIe Gen4x4 capability. However, the board also has a direct connection between the non-PCIe transceiver banks, which can be used for an Ethernet 10GBase-KR loopback. However, Efinix will not be providing an example design to demonstrate that functionality.

Features

- Efinix Ti375N1156 FPGA in an 1,156-ball FineLine BGA package
- PCIe edge card connector
- PCIe edge card socket
- Two 256-Mbit SPI NOR flash devices:
  - Supports single, dual and quad mode for each SPI flash
  - Supports x8 with both SPI flash devices in quad mode
- One micro-USB connector to configure the development board
- 12.0 V, 5 A ATX power supply connector for standalone operation[1]
- Power good and Ti375 configuration done LEDs

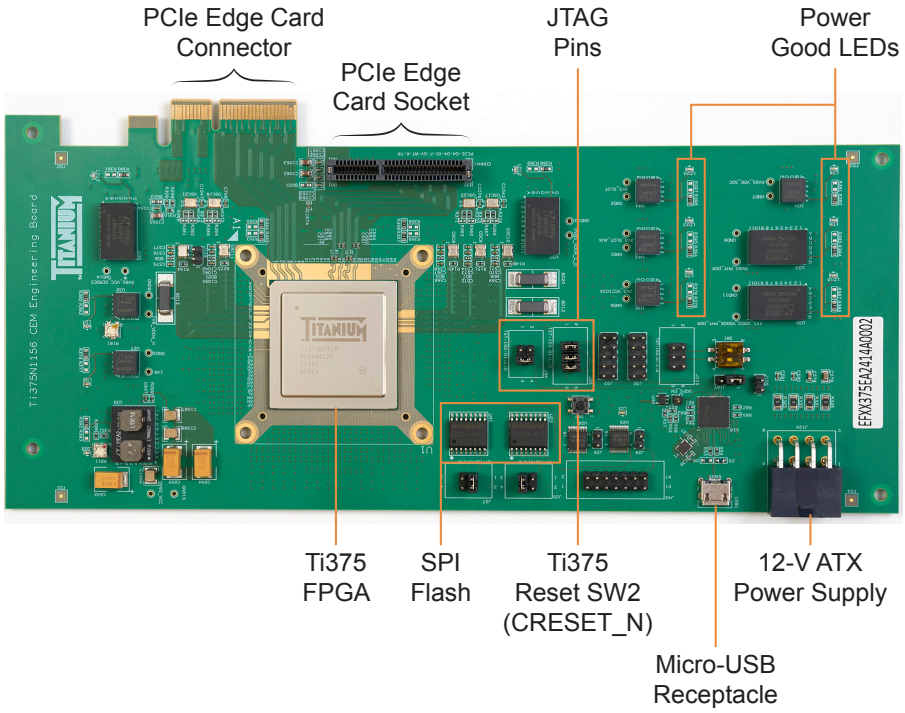
[1] The connector has eight pins, but the power cable only uses 6 of them. The cable only plugs in one way, so you cannot connect the wrong pins by accident.

Overview

The board features the Efinix Ti375 FPGA in a N1156-ball FBGA package, which is fabricated using Efinix Quantum® technology. The Quantum®-accelerated programmable logic and routing fabric is wrapped with an I/O interface in a small footprint package. Ti375 FPGAs also include embedded memory blocks and DSP blocks. You create designs for the Ti375 FPGA in the Efinity® software, and then download the resulting configuration bitstream to the board using the USB connection.

Learn more: For more information on Ti375 FPGAs, refer to the Ti375 Data Sheet.

Figure 2: Ti375 PCIe Early Access Board Components



The board includes a micro-USB port for the FTDI interface. The FTDI FT2232H module supports the following interfaces:

- *FTDI interface 0*—SPI interface.
- *FTDI interface 1*—JTAG inteface.

The FTDI module receives the Ti375 configuration bitstream from a USB host and writes to the Ti375 FPGA in JTAG configuration. You can write a configuration bitstream to the on-board SPI NOR flash memory with SPI active mode using a JTAG bridge.

The SPI NOR flash memory stores the configuration bitstream. The Ti375 FPGA uses this configuration bitstream when it is in active configuration mode (default).

Learn more: Refer to the Ti375 PCIe Early Access Board Schematics and BOM for more information about the components used.

Power On

You turn on the board by attaching the power cable to the board and a power source. The power good LEDs illuminate, giving you a visual confirmation of the status. Do not connect multiple active power sources to the board at the same time.

Note: The board's PCIe edge card connector can receive power for the board. However, the required crossed extender cable cannot supply power. Therefore, if you connect the Ti375 PCIe Early Access Board to a motherboard you need to use the ATX power cable to supply power.

Reset

The Ti375 FPGA is typically brought out of reset with the `CRESET_N` signal. Upon power up, the Ti375 FPGA is held in reset until `CRESET_N` toggles high-low-high.

Note: You can manually assert the high-low-high transition with pushbutton switch SW2.

`CRESET_N` has a pull-up resistor. When you press SW2, the board drives `CRESET_N` low; when you release SW2, the board drives `CRESET_N` high. Thus, a single press of SW2 provides the required high-low-high transition.

After toggling `CRESET_N`, the Ti375 FPGA goes into configuration mode and reads the configuration bitstream from the flash memory. When configuration completes successfully, the FPGA drives the `CDONE` signal high. `CDONE` is connected to a LED (LD1), which turns on when the Ti375 FPGA enters user mode.

Important: For the Ti375 PCIe Early Access Board, **you must press SW2 twice** whenever you power cycle the board for the Ti375 FPGA to configure correctly. Alternatively, configure the FPGA twice with the bitstream.

Configuration

You can configure the Ti375 FPGA using the SPI passive, JTAG, and SPI active via JTAG bridge configuration modes (see Programming the Development Board).

- JTAG or SPI passive[2]. Use to program the FPGA.
- SPI active (up to x8) via JTAG bridge.

The Ti375 PCIe Early Access Board does not support internal reconfiguration for remote updates.

Note: For more details on JTAG bridge mode, refer to Efinity Software User Guide.

[2] For SPI passive mode, remove the jumpers connecting the pins on J28 and J29.

Clock Sources

Nine on-board oscillators (25 and 100 MHz) are available to drive the Ti375 PLL input pins and transceiver clock inputs.

Table 1: Oscillator and Clock Generator Sources

Clock Source	Label	Ti375 Pin Name	Resource
25 MHz oscillator	OSC13	GPIOB_P_26_EXTSPICLK	for SPI clock
100 MHz oscillator	OSC8	GPIOR_88_PLLIN1	PLL TR1
100 MHz oscillator	OSC9	GPIOR_101_PLLIN1	PLL TR2
100 MHz oscillator	OSC10	GPIOR_93_PLLIN1	PLL TR0
100 MHz oscillator	OSC11	GPIOR_145_PLLIN1	PLL BR1
156 MHz oscillator	OSC16	Q1_REFCLK0_PN	SERDES1
156 MHz oscillator	OSC17	Q3_REFCLK0_PN	SERDES3
100 MHz oscillator	OSC21	Q0_REFCLK0_PN	SERDES0
100 MHz oscillator	OSC22	Q2_REFCLK0_PN	SERDES2

Headers

The board contains a limited set of headers to provide power inputs, signal inputs, and outputs, and to communicate with the Efinity software on your computer.

Table 2: Ti375 PCIe Early Access Board Headers

Reference Designator	Description
J124	12 V DC ATX power supply input jack
J34, J35	JTAG header
USB1	Micro-USB receptacle (FTDI FT2232H)
GF1	PCIe edge card connector
CONN41	PCIe edge card socket

Header J124 (ATX Power Supply)

J124 is a 12 V DC power supply input jack. J124 supplies power to regulators on the board that power the Ti375. The maximum current supply to this input jack is 3 A. J124 is an 8-pin ATX connector. You use a 6-pin cable to connect to J124. The cable only fits in one way; you cannot plug it into the wrong slots.

Important: To prevent unexpected failures, you should only use the 6-pin ATX power cable that comes with the board.

Header J34 and J35 (JTAG)

J34 and J35 are headers for the JTAG interface. You can access the Ti375 JTAG pins through this interface.

Table 3: JTAG Pin Assignments

Header	Pin Number	Signal Name
J34	1	TDO
J35	2	TDI
J35	4	TCK
J35	6	TMS

Header USB1 (USB FTDI FT2232H)

USB1, a micro-USB receptacle, is the interface between the board and your computer for communication through the FTDI FT2232H chip. Connect the micro-USB cable for configuring the Ti375 FPGA and NOR flash. Refer to Overview for more details on the FTDI FT2232H module interface.

PCIe Edge Card Connector

This connector is a standard PCIe edge card connector that fits into an edge card socket. This connector goes to the Ti375 FPGA's PCIe transceiver bank (quad 0).

Important: Use the PCIe extender cable provided with the Ti375 PCIe Early Access Board to connect the board to a motherboard. DO NOT connect the Ti375 PCIe Early Access Board directly to the motherboard.

PCIe Edge Card Socket

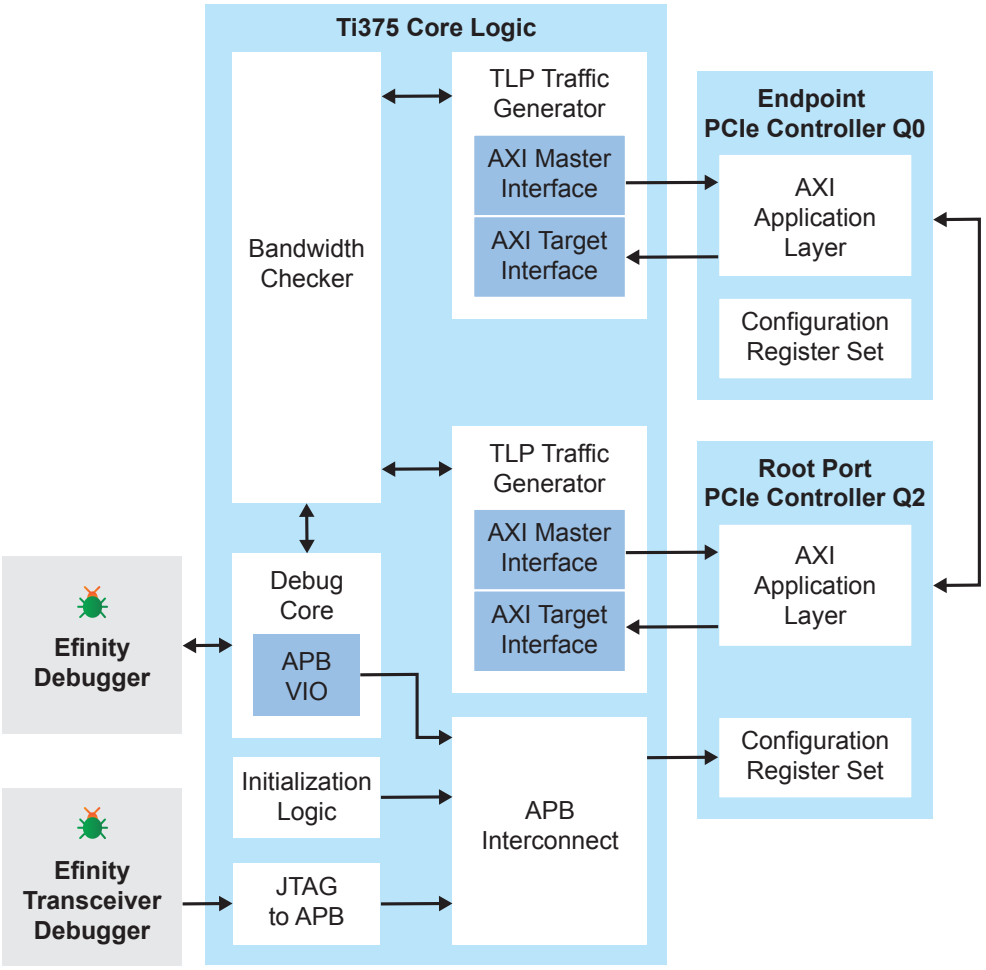
This connector is a standard PCIe socket that accepts an edge card connector. This socket connects to the Ti375 FPGA's PCIe transceiver bank (quad 2).

Ti375 PCIe Early Access Board Example Design

Efinix preloads the Ti375 PCIe Early Access Board with an example design that demonstrates the following functions:

- Two-way transaction layer packet traffic generator with AXI master and slave logic as well as a pattern generator and checker.
- Use the Efinity Transceiver Debugger to test the link quality. The Transceiver Debugger is a new tool available with the Efinity software v2024.1.
- Use the Efinity Debugger and a virtual I/O debug core to:
  - Check the PCIe bandwidth.
  - Use the APB interface to access the PCIe Controller's configuration registers for debugging.

Figure 3: Example Design Block Diagram Overview

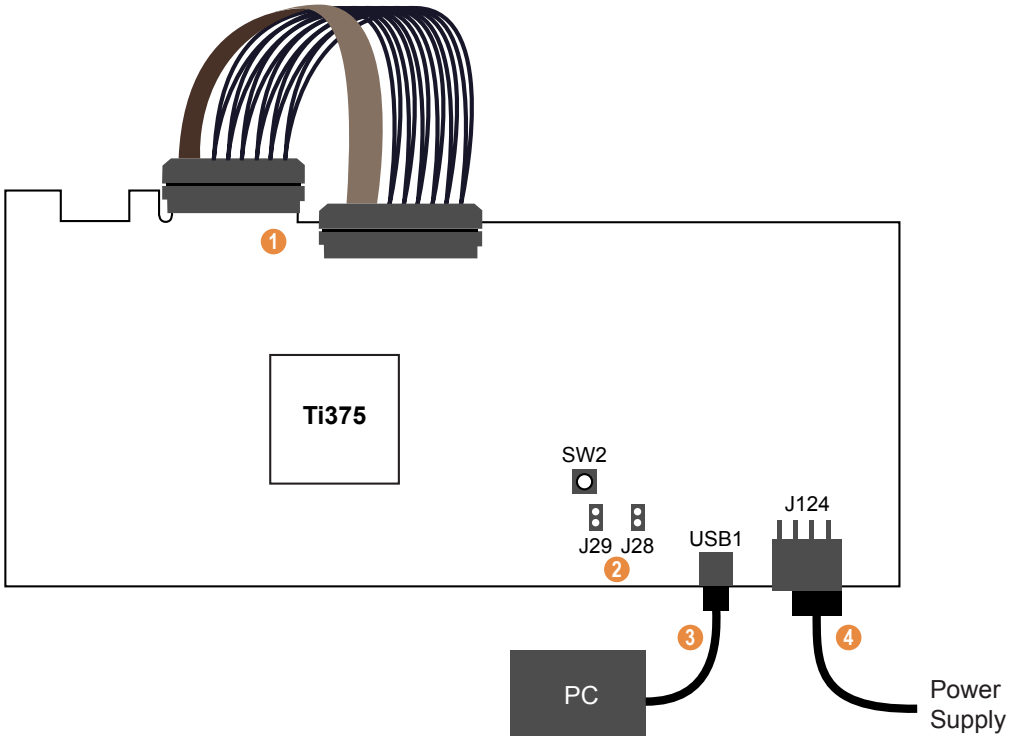


In this design, a PCIe link is configured with transceiver quad 2 as the root port and transceiver quad 0 as the endpoint. Using AXI interface as a top-level wrapper, the design establishes bidirectional basic Transaction Layer Packet (TLP) traffic. The design includes a debug core to show the traffic waveform, PCIe link status, data throughput, and to provide access to the root port's configuration registers. Additionally, you can use this design to view the PCIe signal quality with the Efinity Transceiver Debugger.

Set Up the Hardware

The following figure shows the hardware setup steps:

Figure 4: Hardware Setup



Important: Always remove the power supply before attaching or detaching cables.

1. Connect the PCIe crossed extender cable to the PCIe edge card connector and PCIe edge card socket.
2. Remove all jumpers from the 2-pin headers J28 and J29.
3. Connect the micro-USB cable to the board and to a USB port on your computer.
4. Connect the 12 V ATX power cable to the board connector and a power source. The connector has eight pins, but the cable only has six. The cable only fits into the connector one way, so you cannot connect it incorrectly.

Table 4: Board LED Outputs

LED	Description
LD2, LD5, LD6, LD7, LD8, LD10, LD12, LD13, LD22, LD23, LD24 turned on	Power good
LD1 turned on	FPGA configuration done

Run the Demonstration Design

Before you run the demonstration design, download the example design files from the Efinix Support Center. You need to use the debug profile included with the design files.

1. Go to **Support Center > Examples > Design Example: Ti375 PCIe Early Access Board** (direct link).
2. Download the **efx_ti375n1156_ea_oob-v**<*version*>**.zip** file.
3. Extract the design files into your project folder.

Use these steps to run the demonstration design:

1. Press pushbutton SW2 **twice** to toggle the `CRESET_N` signal and load the design from the flash device.
2. Wait for the Ti375 FPGA to enter user mode, indicated by LED LD1 illuminating.
3. Launch the Efinity software and open the Debugger.
4. Choose **File > Open Debug Profile**.
5. Browse to the example design directory and choose **debug_profile.json**. Click **Open**.
6. Select JTAG **USER1** and connect the Debugger (refer to the Efinity Software User Guide for complete instructions on using the Debugger).

Read the PCIe Link Status

The Debugger's **vio0** tab contains the information for the PCIe link status. It has these sources and probes:

Table 5: Debugger vio0 Tab

Name	Type	Width	Description
pcie_rst	Source	1	Active-high reset. Reset the RTL design and trigger a hot reset in the PCIe Controller.
q2_p00_rate	Probe	2	PIPE link signal rate for transceiver quad 2. Selects the data rate. 2'b00: PCIe Gen1 2'b01: PCIe Gen2 2'b10: PCIe Gen3 2'b11: PCIe Gen4
q2_ltssm_state	Probe	6	LTSSM state for transceiver quad 2. Refer to "Appendix C: LTSSM State Encoding" in the Titanium PCIe® Controller User Guide for more information.
q2_link_status	Probe	2	Status of the quad 2 PCIe link. 2'b00: No receivers detected. 2'b01: Link training in progress. 2'b10: Link up, DL initialization in progress. 2'b11: Link up, DL initialization completed.
q2_cmn_ready	Probe	1	Common ready for transceiver quad 2.
q0_p00_rate	Probe	2	PIPE link signal rate for transceiver quad 0. Selects the data rate.

Name	Type	Width	Description
q0_ltssm_state	Probe	6	LTSSM state for transceiver quad 0. Refer to "Appendix C: LTSSM State Encoding" in the Titanium PCIe® Controller User Guide for more information.
q0_link_status	Probe	2	Status of the PCIe link for transceiver quad 0. 2'b00: No receivers detected. 2'b01: Link training in progress. 2'b10: Link up, DL initialization in progress. 2'b11: Link up, DL initialization completed.
q0_cmn_ready	Probe	1	Common ready for transceiver quad 0.
negotiated_link_speed	Probe	4	Negotiated link speed of the device. 4'b0001: 2.5 GT/s per lane. 4'b0010: 5.0 GT/s per lane. 4'b0011: 8.0 GT/s per lane. 4'b0100: 16.0 GT/s per lane.
negotiated_link_width	Probe	6	Negotiated link width of the device. 6'd1: x1 width 6'd2: x2 width 6'd3: x3 width 6'd4: x4 width
error_cnt	Probe	32	Number of corrupted data in the link.

Reset the Design and PCIe Link

Use the `pcie_rst` source in the Debugger's **vio0** tab to reset the RTL design and perform a hot reset on the PCIe link. Set `pcie_rst` to `1` to perform a reset.

Alternatively, you can press SW2 to force the FPGA to reload the bitstream, thereby causing a reset.

Note: Reloading the bitstream triggers a cold reset in the PCIe Controller. Refer to "Reset Types" in the Titanium PCIe® Controller User Guide for more information.

Check the Bandwidth

You use the bandwidth checker to evaluate the design's data throughput. To access the bandwidth calculator, click the Debugger's **vio1** tab. The calculator has these sources and probes:

Table 6: Debugger vio1 Tab

Name	Type	Width	Description
ReadWrite_pattern	Source	1	AXI traffic selector. 1: Write traffic 0: Read traffic
bandwidth_MBps	Probe	16	PCIe bandwidth in MBps.

Write to PCIe Registers through APB Interface

The Debugger's **apbvio** tab gives you access to the PCIe Controller's configuration register set through the APB interface. This access is useful for debugging purposes.

Note: Refer to "Configuring Registers with the APB interface" in the Titanium PCIe® Controller User Guide for more information.

Table 7: Debugger apbvio Tab

Name	Type	Width	Description
apb_write	Source	1	APB transfer selector. 1: Write transfer 0: Read transfer
apb_paddr	Source	24	APB address input.
apb_pwdata	Source	32	APB write data input.
apb_pstrb	Source	4	APB write strobe input.
apb_start	Source	1	Start APB transfer operation. Toggle (0 to 1) to begin transfer.
apb_prdata	Probe	32	Read data output.

View Traffic Waveform

The Debugger has two logic analyzers to view the traffic waveform:

- **Laq2** contains the transceiver quad 2 master AXI interface and transceiver quad 0 target AXI interface.
- **Laq0** contains the transceiver quad 0 master AXI interface and transceiver quad 2 target AXI interface.

Note: Refer to the Efinity Debugger Tutorial for instructions on using the logic analyzer.

Debugging Transceivers (Beta)

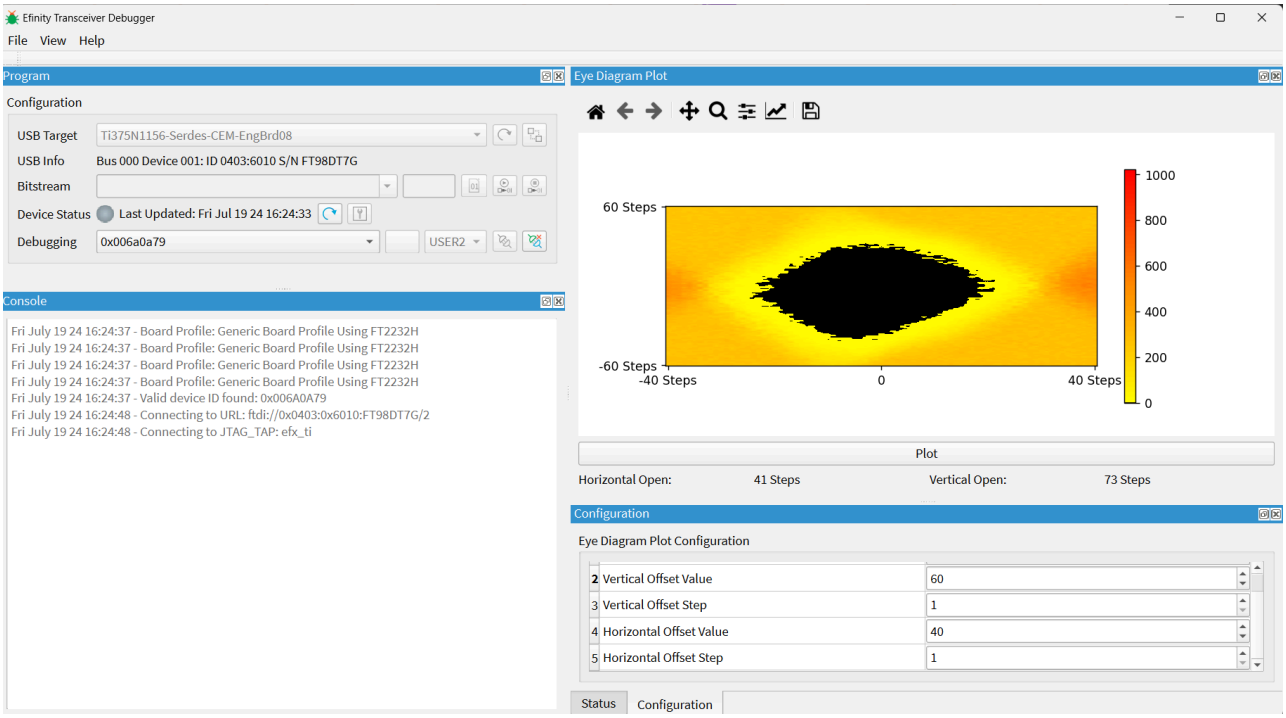
The Efinity software v2024.1 and higher includes the Efinity Transceiver Debugger tool that tests and displays the signal quality of the transceiver signals; it does not test the actual data itself. This tool is **not** an oscilloscope.

Important: This tool is Beta in v2024.1, and has limited features. For example, it can only access channel 0 for the eye diagram plot.

The left side of the Transceiver Debugger has programming options and buttons. Its functionality is similar to the Programmer, only simplified. A Console displays messages such as the device ID, board profile, and device status. It also shows any error messages.

The right side of the window has an Eye Diagram Plot. The **Status** tab shows the PHY and link status; the **Configuration** tab has options to customize the plot.

Figure 5: Efinity Transceiver Debugger



Launching the Transceiver Debugger

In the Efinity software v2024.1 you launch the Transceiver Debugger using a batch file (Windows) or shell script (Linux).

Windows—Execute the file *<Efinity version>***\debugger\serdes_debug_tool\bin\efinity_serdes_dbg.bat**.

Linux—Execute the file *<Efinity version>***/debugger/serdes_debug_tool/bin/efinity_serdes_dbg.sh**.

Using the Transceiver Debugger

This topic assumes you already know how to program an FPGA using the Efinity Programmer. To use the Transceiver Debugger:

1. Connect an Efinix transceiver-capable board, e.g., a board with the Ti375 N1156 FPGA. If you are using the Ti375 PCIe Early Access Board, it comes pre-loaded with an example design.
2. Click Refresh USB Target if the **USB Target** field does not display the board name.
3. If the FPGA is not programmed already, select a bitstream file and program it as you normally would.
4. Select **JTAG > USER2** and connect the Transceiver Debugger.

   Note: Disconnect all other debug cores before connecting the Transceiver Debugger.

5. Review the **PCIE Status** table in the **Status** tab. This table shows the link status and speed.

6. Review the **PHY Status** table in the **Status** tab. This table show whether the lanes are locked and ready.
7. In the **Configuration** tab, adjust the settings for the eye diagram plot.

Setting	Description
Vertical Offset Value	Indicates how many vertical positions there are to the right and left of center. Higher values result in a plot with high resolution, at the expense of longer drawing time. Default: 40
Vertical Offset Step	Indicates how many vertical positions to include in the same color pixel. For example, for an offset value of 40 and a step of 2, the software plots a total of 20 blocks on each side of the center line. A higher number results in less drawing time. Default: 2
Horizontal Offset Value	Indicates how many horizontal positions there are to the right and left of center. Higher values result in a plot with high resolution, at the expense of longer drawing time. Default: 40
Horizontal Offset Step	Indicates how many horizontal positions to include in the same color pixel. For example, for an offset value of 40 and a step of 2, the software plots a total of 20 blocks on each side of the center line. A higher number results in less drawing time. Default: 2
Accumulation Period	The accumulation period is how long to perform the test for each pixel. A higher value provides more saturated results. Each period in the nanoseconds range.

8. When you are finished setting values, click **Plot**. The tool begins sampling data and displaying the results on the plot.

Interpreting the Results

The **PCIE Status** table shows information about the link. You can review the data to see whether the link is operating at the correct speed with the correct number of lanes.

The eye plot gives a visual representation of the link quality. Areas that are black have no errors reported. The yellow, orange, and red colors indicate how many errors are being found (yellow is less while red is more).

You can use the plot to see if the error-free area (or eye) is sufficient for your application. For example, for a PCIe Gen4x4 interface, the unit interval (UI) for PCIe Gen4 is 62.5 picoseconds (refer to the PCI-SIG's PCI Express® Base Specification for more information). In the PCIe Gen4x4 plot:

- One horizontal step is approximately three millivolts.
- One unit interval is 64 steps. A UI is a full window (between the X of the eye diagram).
- A step, therefore, is close to 1 picosecond.

Creating Your Own Design

The Ti375 PCIe Early Access Board allows you to create and explore designs for the Ti375 FPGA. Efinix® provides example code and designs to help you get started:

- Our Support Center (www.efinixinc.com/support) includes examples targeting the board.
- The Efinity® software includes also example designs that you can use as a starting point for your own project and includes a step-by-step tutorial.

Restoring the Demonstration Design

After you have used the board for other designs, you may want to go back to the original pre-loaded example design. The preloaded example design project file is available in the Support Center . To restore the example design, you need to program the board's SPI flash device with the Ti375 example design bitstream.

Note: The example design available in the Support Center requires Efinity software v2024.1 or later.

The example design zip file includes a bitstream file to get you started quickly. Download it to the board using these steps:

1. Download the file **efx_ti375n1156_ea_oob-v<*version*>.zip** from the Support Center.
2. Open the project (**ti375n1156_ea_oob.xml**) in the Efinity software. The project is located in the **efx_ti375n1156_ea_oob-v<*version*>** directory.
3. Review the design.
4. Connect the Ti375 PCIe Early Access Board to your computer using a USB cable.
5. Use the Efinity® Programmer to download the bitstream file, **bitstream/ti375n1156_ea_oob.hex**, to your board using SPI active via JTAG bridge mode. Set the **Starting Flash Address** to `0x000000`.

   Note: You use SPI active mode because you need to reset the FPGA. Efinix also includes the **ti375n1156_ea_oob.bit** file to be used with JTAG mode. Using JTAG mode requires you to reprogram the bitstream into the board when you reset the board.

Learn more: Instructions on how to use the Efinity® software and board documentation are available in the Support Center .

Revision History

Table 8: Revision History

Date	Version	Description
August 2024	1.1	Corrected link to schematics and BOM.
July 2024	1.0	Initial release.