Titanium Ti375 C529 Development Kit User Guide

Ti375C529-DK-UG-v1.0
June 2024
www.efinixinc.com

Contents

Introduction

Thank you for choosing the Titanium Ti375 C529 Development Kit (part number: Ti375C529-DK), which allows you to explore the features of the Ti375 FPGA.

The Titanium Ti375 C529 Development Kit has everything you need to develop and prototype designs for the Ti375 FPGA. This kit showcases the Ti375's hardened RISC-V block with a 32-bit CPU featuring the ISA RISCV32I with M, A, C, F, and D extensions, and six pipeline stages. The hard processor has 4 CPUs each with a dedicated FPU and custom instructions. Additionally, the Ti375's soft-core MIPI D-PHY supports up to 1.5 Gbps per lane, and has hardened LPDDR4/4x controllers.

The Titanium Ti375 C529 Development Board has two (2) 512 Mbit SPI NOR flash devices, which support SPI active (x1, x2, x4, x8 modes) to configure the FPGA and store other user data such as RISC-V software. One SPI flash device (U48) also supports SPI passive mode. A highly integrated Ethernet PHY is on-board supporting RGMII at up to 1,000 Mbps.

One key feature is the on-board USB-to-dual JTAG device that supports simultaneous debugging of the Ti375 FPGA and the RISC-V SoC. Additionally, the board has an array of on-board interfaces such as an FMC (LPC) interface and three (3) PMOD interfaces, which let you expand the board's functionality.

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

What's in the Box?

The Titanium Ti375 C529 Development Kit includes:

- Titanium Ti375 C529 Development Board
- 1 USB type-C cable
- 12 V, 3 A universal power adapter with 5.5 mm DC power converter
- 4 standoffs, 4 screws, 8 washers, and 8 nuts
- 12 V cooling fan
- Heat sink
- Quick Start Guide

Register Your Kit

When you purchase an Efinix development kit, you also receive a license for the Efinity® software plus one 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**) and 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 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 and re-connect it.

Installing the Windows USB Drivers

The Titanium Ti375 C529 Development Board development board has an FTDI FT4232H chip to communicate with the USB port.

Note: If you have another Efinix board and are using the Titanium Ti375 C529 Development 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 **Ti375 C529 Development Kit (Interface 1)**.

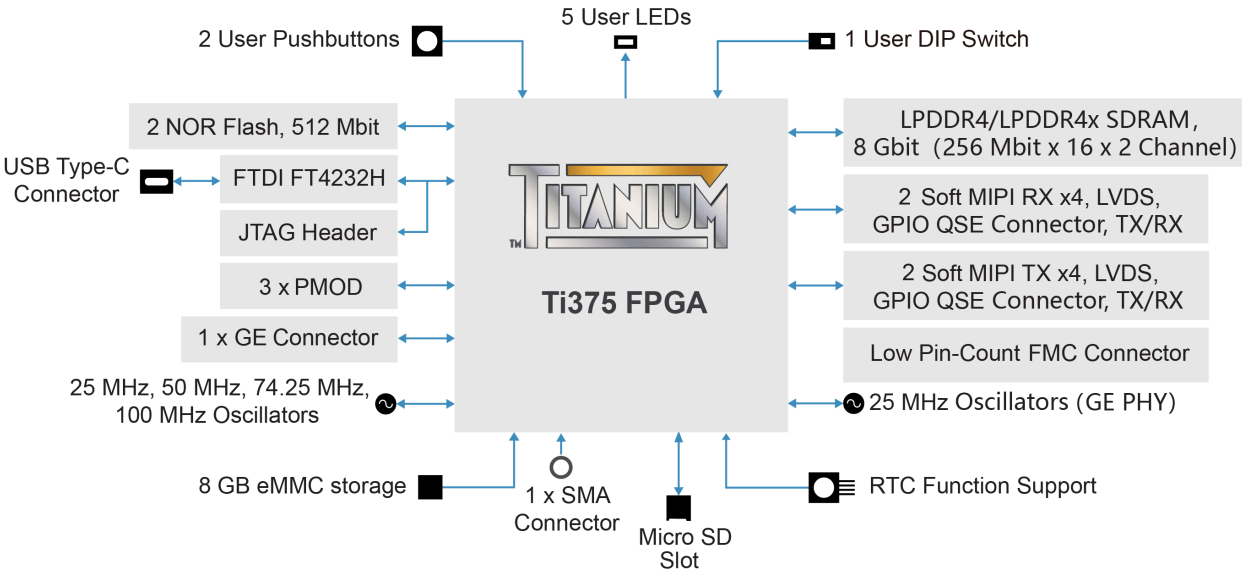
   Note: You are required to use the default driver for Interface 2 when you connect the Titanium Ti375 C529 Development Board (which uses UART interface) to your computer.

5. Select **libusb-win32** in the **Driver** drop-down list.
6. Click **Replace Driver**.
7. Close the Zadig software.

Board Functional Description

The Titanium Ti375 C529 Development Board contains a variety of components to help you build designs for the Titanium Ti375 FPGA.

Figure 1: Titanium Ti375 C529 Development Board Block Diagram



Features

- Efinix Ti375C529C4[1] FPGA in a 529-ball FineLine BGA package
- 8 Gbit (256 Mbit x 16 x 2 channels) LPDDR4/4x SDRAM:
  - Supports x32 data width
  - Read/write speed up to 3.3 Gbps
- 2 512 Mbit SPI NOR flash memories
  - Supports single, dual, and quad mode for each SPI flash
  - Supports x8 with both SPI flash devices in quad mode
- 8 GByte EMMC
- 2 high-speed QSE connectors that support up to 4 channels (4 data lanes + 1 clock lane, 1.5 Gbps per lane) TX/RX soft-core MIPI interfaces or LVDS/GPIO[2]
- 4 high-speed QSE connectors that support MIPI, LVDS, and GPIO[2]
- 3 PMOD interfaces with extended functionality
- Gigabit Ethernet interface supporting RGMII standard and complies with 10 Base-T, 100 Base-TX, and 1,000 Base-T IEEE 802.3 standards
- Micro-SD card slot
- FPGA mezzanine card (FMC) with low pin-count connector (LPC)
- USB type-C connector to configure the development board
- SMA interface
- Supports RTC function for time measurement and calendar
- 25, 50, 74.25, and 100 MHz oscillators for Ti375 clock input
- User LEDs and switches:
  - 5 LEDs
  - 2 pushbutton switches
- Power:
  - 12.0 V power supply connector
  - On-board regulator sources: 0.6 V, 0.85 V, 0.95 V, 1.0 V, 1.1 V, 1.2 V, 1.8 V, 3.3 V, 5.0 V
- Power good and Ti375 configuration done LEDs

Compatible Daughter Cards

The following are daughter cards that are compatible with the Titanium Ti375 C529 Development Board:

- HDMI Daughter Card
- MIPI and LVDS Expansion Daughter Card
- Dual MIPI to DSI Converter Daughter Card
- Raspberry Pi Camera Connector Daughter Card
- Dual Raspberry Pi Camera Connector Daughter Card
- Coral Camera Connector Daughter Card

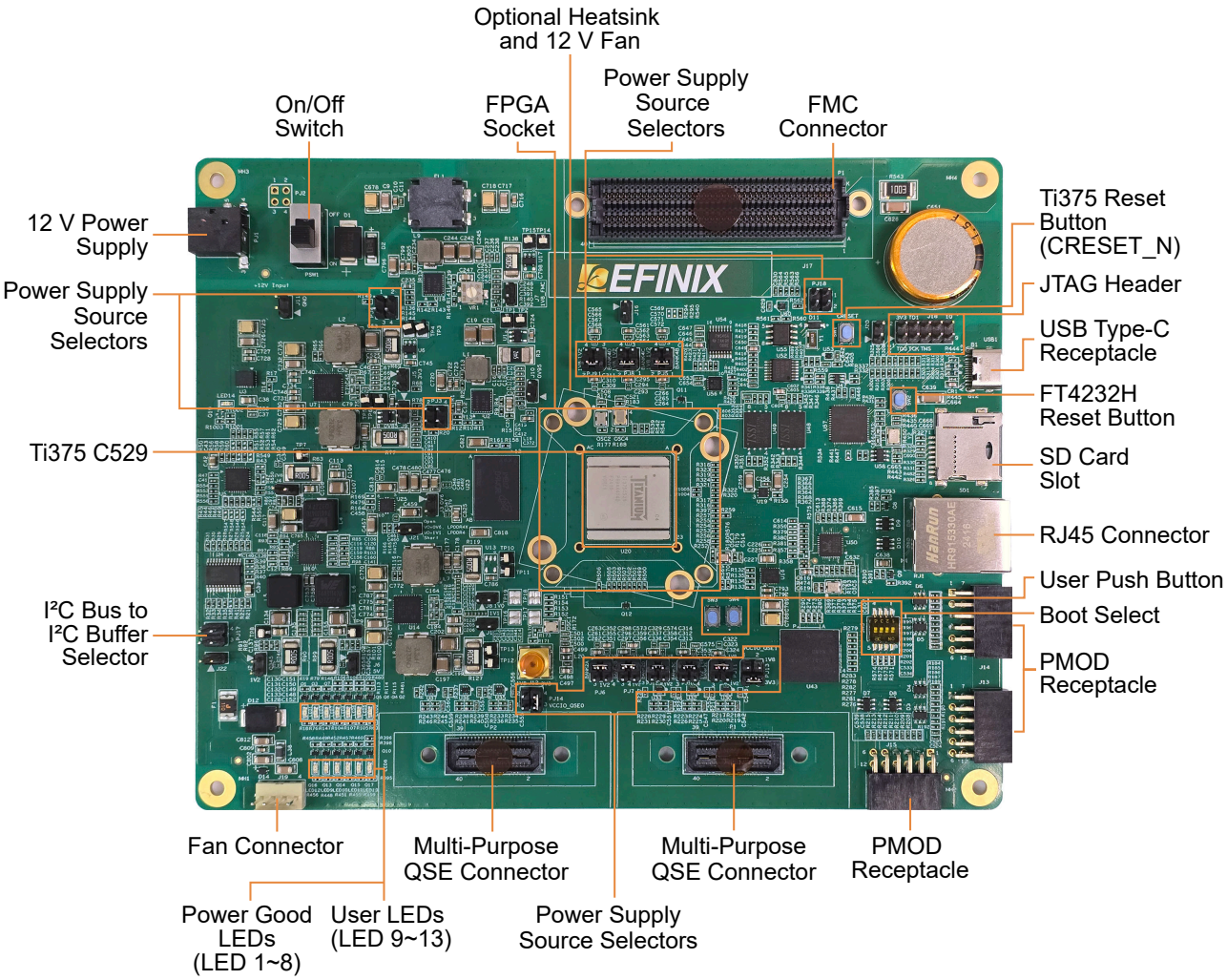
(1) The FPGA speed grade may vary and is subject to availability.
(2) Compatible with the daughter cards from other Efinix development kits

Overview

The board features the Efinix Ti375 FPGA in a 529-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: Titanium Ti375 C529 Development Board Components



The Titanium Ti375 C529 Development Board provides two multi-purpose 0.8 mm high-speed ground plane sockets. These sockets can be used for GPIO, MIPI CSI-2 TX/RX, and LVDS TX/RX. The board includes a USB type-C port for the FTDI interface.

The FTDI FT4232H module supports the following interfaces:

- *FTDI interface 0*—JTAG for RISC-V debugging
- *FTDI interface 1*— JTAG for FPGA and RISC-V debugging
- *FTDI interface 2*— UART Communication
- *FTDI interface 3*— PMIC Configuration/Reserved

The FTDI module receives the Ti375 configuration bitstream from a USB host and writes to the Ti375 FPGA in SPI active configuration. You can write a configuration bitstream to the on-board SPI NOR flash memory through JTAG with the **JTAG SPI Flash Loader Core**. Additionally, it supports a UART interface to the Ti375.

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 **Titanium Ti375 C529 Development Board Schematics and BOM** for more information about the components used.

Power On

To turn on the development board, turn on switch PSW1. The 12 V DC power is input to the on-board regulators to generate the required 0.6 V, 0.85 V, 0.95 V, 1.0 V, 1,1 V, 1.2 V, 1.8 V, 3.3 V, and 5.0 V for components on the board. When these voltages are up and stable, the power-good LEDs, LED1 - LED8 illuminate, giving you a visual confirmation of the status.

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 SW1.

`CRESET_N` has a pull-up resistor. When you press SW1, the board drives `CRESET_N` low; when you release SW1, the board drives `CRESET_N` high. Thus, a single press of SW1 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 (LED1), which turns on when the Ti375 FPGA enters user mode.

FTDI Reset

Pushbutton SW2 is the FTDI FT4232H chip reset button. All board communications through the FTDI FT4232H chip disconnect when you press pushbutton SW2, and reconnect when you release it.

Configuration

You can configure the Ti375 FPGA using the following configuration modes:

- JTAG
- SPI Active (up to x8) via JTAG Bridge

You can use the JTAG bridge mode when programming the flash. See **Programming the Development Board** for SPI Active mode programming operations. You must use the JTAG Bridge when programming the flash because the SPI active signals are not routed directly to FT4232H on the Titanium Ti375 C529 Development Board. When generating bitstreams for you own design, ensure that you select the **Active** option in the **Bitstream Generation** tab of the Efinity **Project Editor**. Refer to **Programming the Development Board** for instructions on using SPI active mode.

The Titanium Ti375 C529 Development Board does not support internal reconfiguration for remote updates.

Note: For more details on the JTAG SPI bridge loader, refer to **Efinity Software User Guide**.

Clock Sources

Four on-board oscillators (25, 50, 74.25, and 100 MHz) are available to drive the Ti375 PLL input pin and clock input. Additionally, there is a dedicated 25 MHz on-board oscillator as the Ethernet PHY clock source. The Titanium Ti375 C529 Development Board also provides an external clock input through the J12 (SMA) connector. Refer to **Header J12 (SMA)** to enable the external clock source.

Table 1: Oscillator and Clock Generator Sources

Clock Source	Ti375 Pin Name	PLL Resource
25 MHz oscillator or SMA1	GPIOT_P_50_PLLIN0	TR0
50 MHz oscillator	GPIOT_P_36_CLK19_P	N/A
74.25 MHz oscillator	GPIOL_32_PLLIN1	BL1
100 MHz	GPIOL_25_PLLIN1	BL0

Note: The Efinity Interface Designer issues an unrouted clkmux input error if more than 8 GCLK resources are used on the left side of Ti375 FPGA. To solve this, assign one of the clocks, for example MIPI clock TX0, to use the RCLK instead of the GCLK. For more information, refer to the Clock and Control Network section of the **Ti375 Data Sheet**.

Real Time Clock (RTC)

The Titanium Ti375 C529 Development Board supports a Real Time Clock (RTC) function. The I^2C protocol is used for the communication between the FPGA and the RTC. You need to install the capacitor, C651 to use the RTC.

Note: For more information on the RTC, refer to the Titanium Ti375 C529 Development Board Schematics and BOM

eMMC

The Titanium Ti375 C529 Development Board provides an 8 GB eMMC 1.8 V function on board. This function allows the data transfer of 200 MBps using clock frequency of 200 MHz in HS200 mode.

Note: For more information on the eMMC, refer to the Titanium Ti375 C529 Development Board Schematics and BOM.

Ethernet PHY

The Titanium Ti375 C529 Development Board provides an Ethernet PHY function. The operating characteristics of the Ethernet PHY are as follows:

- The Gigabit Ethernet (GE) PHY works in RGMII to copper mode.
- The receiver side of the GE PHY (`RXDELAY`) has a 2 ns delay, which is added to `RXC` for `RXD` latching by pulling up `ETH0_RXD0`.
- In the data trasmission of `CFG_LDO[1:0]=2bit10`, the RGMII works with a supply of 1.8 V.
- The power supplies for the GE PHY are as follows:
  - DVDD33, AVDD33: 3.3 V
  - DVDD10, AVDD10: 1.0 V
  - DVDD_RG: RGMII I/O Pad Power

Note: For more information on the Ethernet PHY, refer to the Titanium Ti375 C529 Development Board Schematics and BOM

PMOD

The Titanium Ti375 C529 Development Board provides three 3.3 V PMOD connectors.

Note: For more information on the RTC, refer to the Titanium Ti375 C529 Development Board Schematics and BOM

Thermal Monitor

The Titanium Ti375 C529 Development Board has a temperature monitor to check the working temperature on the development board. The temperature monitor uses two 3.3 V temperature sensors. A 12 V fan is used to cool the FPGA if needed. See Cooling Fan Control on page 12 for information on the cooling fan.

Note: For more information on the temperature control circuits, refer to the **Titanium Ti375 C529 Development Board Schematics and BOM**

Cooling Fan Control

The Titanium Ti375 C529 Development Board provides a cooling fan to lower the temperature on the development board. A 12 V fan is used to cool the FPGA and is connected to the board through the J19 header. One GPIO pin of the thermal monitor is connected to the PWM signal of the fan. The speed of the fan can be fixed and configured by the FPGA using PWM.

Note:

- The 12 V fan cools down the FPGA if the FPGA temperature is high.
- For more information on the temperature control circuits, refer to the **Titanium Ti375 C529 Development Board Schematics and BOM**.

Headers

The board contains a variety of headers to provide power inputs, signal inputs, and outputs, and to communicate with external devices or boards.

Table 2: Titanium Ti375 C529 Development Board Headers

Reference Designator	Description
P1	40-pin multi-purpose high-speed QSE connector for MIPI TX
P2	40-pin multi-purpose high-speed QSE connector for MIPI RX
PJ1	12 V DC power supply input jack
PJ3	User selectable programming mode for DC module (0.95 V)
PJ4	User selectable programming mode for FMC_1V8
PJ5	User selectable VCCIO for BANK 4B (1.2 V or 1.8 V)
PJ6	User selectable VCCIO for BANK 2A (1.2 V or 1.8 V)
PJ7	User selectable VCCIO for BANK 2B (1.2 V or 1.8 V)
PJ8	User selectable VCCIO for BANK 4C (1.2 V or 1.8 V)
PJ9	User selectable VCCIO for BANK 4D (1.2 V or 1.8 V)
PJ10	User selectable VCCIO for BANK 2C (1.2 V or 1.8 V)
PJ11	User selectable VCCIO for BANK 2E (1.2 V or 1.8 V)
PJ12	User selectable VCCIO for BANK 2D (1.2 V or 1.8 V)
PJ13	User selectable VCCIO for QSE1 (P1) GPIO interface (1.8 V or 3.3 V)
PJ14	User selectable VCCIO for QSE2 (P2) GPIO interface (1.8 V or 3.3 V)
PJ17, J22	I^2C bus to I^2C buffer selector
PJ18	User selectable VCCIO for FMC_SCL and FMC_SDA (1.8 V or 3.3V)
J1	1.1 V external power input connector
J2	1.2 V external power input connector
J3	0.6 V external power input connector
J4	1.8 V external power input connector
J5	3.3 V external power input connector
J6	5 V external power input connector
J7	1.8 V external power input connector (FMC)
J8	1 V external power input connector (Ethernet PHY)
J9	0.85 V external power input connector
J10	0.95 V external power input connector
J11	Ground test pin
J12	SMA connector
J13, J14, J15	PMOD connector
J16	User configurable for I^2C buffer (CSI and CSO)
J17	FMC connector

Reference Designator	Description
J18	JTAG header
J19	12 V fan connector
J20	User configurable for programming FPGA using FMC daughter card
J21	User configurable input voltage for LPDDR4 (1.1V) or LPDDR4x (0.6V)
RJ1	RJ-45 Connector
SD1	Micro SD card slot
USB1	USB Type-C receptacle (FTDI FT4232H)
TP1, TP2	0.95 V test point
TP3, TP4	3.3 V test point
TP5, TP6	0.85 V test point
TP7	1.8 V test point
TP8	1.2 V test point
TP9	5 V test point
TP10, TP11	1 V test point (Ethernet PHY)
TP12, TP13	1.1 V test point (LPDDR4)
TP14, TP15	1.8 V test point (FMC)
TP16 - TP19	PLL output test point (DDR PLL)

Headers P1 and P2 (Multi-Purpose)

P1 and P2 are multi-purpose high-speed QSE interface connectors for MIPI TX and MIPI RX respectively that support 2 clock lanes and 8 data lanes.

Table 3: P1 Pin Assignments

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
1	3V3	N.C.	2	MIPI1_TXDP0	GPIOT_P_33 _CLK16_P
3	5V	N.C.	4	MIPI1_TXDN0	GPIOT_N_33 _CLK16_N
5	GND	N.C.	6	GND	N.C.
7	MIPI0_TXDP0	GPIOT_P_27	8	MIPI1_TXDP1	GPIOT_P_29
9	MIPI0_TXDN0	GPIOT_N_27	10	MIPI1_TXDN1	GPIOT_N_29
11	GND	N.C.	12	GND	N.C.
13	MIPI0_TXDP1	GPIOT_P_26	14	MIPI1_TXCLKP2	GPIOT_P_35 _CLK18_P
15	MIPI0_TXDN1	GPIOT_N_26	16	MIPI1_TXCLKN2	GPIOT_N_35 _CLK18_N
17	GND	N.C.	18	GND	N.C.
19	MIPI0_TXCLKP2	GPIOT_P_25	20	MIPI1_TXDP3	GPIOT_P_28
21	MIPI0_TXCLKN2	GPIOT_N_25	22	MIPI1_TXDN3	GPIOT_N_28
23	GND	N.C.	24	GND	N.C.
25	MIPI0_TXDP3	GPIOT_P_23_PLLIN0	26	MIPI1_TXDP4	GPIOT_P_34 _CLK17_P
27	MIPI0_TXDN3	GPIOT_N_23	28	MIPI1_TXDN4	GPIOT_N_34 _CLK17_N
29	GND	N.C.	30	GND	N.C.
31	MIPI0_TXDP4	GPIOT_P_24_EXTFB	32	QSE1_GPIO_3 _1V8P3V3	GPIOT_P_21
33	MIPI0_TXDN4	GPIOT_N_24	34	QSE1_GPIO_4 _1V8P3V3	GPIOT_N_21
35	GND	N.C.	36	GND	N.C.
37	QSE1_GPIO_1 _1V8P3V3	GPIOT_P_20	38	QSE1_GPIO_5 _1V8P3V3	GPIOT_P_22
39	QSE1_GPIO_2 _1V8P3V3	GPIOT_N_20	40	QSE1_GPIO_6 _1V8P3V3	GPIOT_N_22

Table 4: P2 Pin Assignments

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
1	3V3	N.C.	2	MIPI1_RXDP0	GPIOT_P_16
3	5V	N.C.	4	MIPI1_RXDN0	GPIOT_N_16
5	GND	N.C.	6	GND	N.C.
7	MIPI0_RXDP0	GPIOT_P_10 _CLK27_P	8	MIPI1_RXDP1	GPIOT_P_17
9	MIPI0_RXDN0	GPIOT_N_10 _CLK27_N	10	MIPI1_RXDN1	GPIOT_N_17
11	GND	N.C.	12	GND	N.C.
13	MIPI0_RXDP1	GPIOT_P_09 _CLK26_P	14	MIPI1_RXCLKP2	GPIOT_P_18
15	MIPI0_RXDN1	GPIOT_N_09 _CLK26_N	16	MIPI1_RXCLKN2	GPIOT_N_18
17	GND	N.C.	18	GND	N.C.
19	MIPI0_RXCLKP2	GPIOT_P_11 _CLK30_P	20	MIPI1_RXDP3	GPIOT_P_15
21	MIPI0_RXCLKN2	GPIOT_N_11 _CLK30_N	22	MIPI1_RXDN3	GPIOT_N_15
23	GND	N.C.	24	GND	N.C.
25	MIPI0_RXDP3	GPIOT_P_13_PLLIN0	26	MIPI1_RXDP4	GPIOT_P_14_EXTFB
27	MIPI0_RXDN3	GPIOT_N_13	28	MIPI1_RXDN4	GPIOT_N_14
29	GND	N.C.	30	GND	N.C.
31	MIPI0_RXDP4	GPIOT_P_12 _CLK31_P	32	QSE0_GPIO_3 _1V8P3V3	GPIOT_P_07_EXTFB
33	MIPI0_RXDN4	GPIOT_N_12 _CLK31_N	34	QSE0_GPIO_4 _1V8P3V3	GPIOT_N_07
35	GND	N.C.	36	GND	N.C.
37	QSE0_GPIO_1 _1V8P3V3	GPIOT_P_05	38	QSE0_GPIO_5 _1V8P3V3	GPIOT_P_08_PLLIN0
39	QSE0_GPIO_2 _1V8P3V3	GPIOT_N_05	40	QSE0_GPIO_6 _1V8P3V3	PGIOT_N_08

Header PJ1 (Power Supply)

PJ1 is a 12 V DC power supply input jack. PJ1 supplies power to regulators on the board that power the Ti375. The maximum current supply to this input jack is 3 A.

Header PJ3 and PJ4 (Programming Mode)

PJ3 and PJ4 are 4-pin headers used to enable the programming mode of the 0.95 V power module and the FMC_1V8 power module.

To enable the programming mode, pin 3 and pin 4 of both PJ3 and PJ4 headers must be connected.

Table 5: PJ3 and PJ4 (Power Module Programming Mode Enable)

Header	Description
PJ3	0.95 V power module programming mode enable
PJ4	FMC_1V8 power module programming mode enable

Header PJ5, PJ6, PJ7, PJ8, PJ9, PJ10, PJ11, and PJ12 (Bank VCCIO Selector)

PJ5, PJ6, PJ7, PJ8, PJ9, PJ10, PJ11, and PJ12 are 4-pin headers used to select the VCCIO power supply for banks 4B, 2A, 2B, 4C, 4D, 2C, 2E, and 2D. By default, the jumpers connect to pins 1 and 2 at 1.8 V. Connect the jumpers as shown in the following table to change the voltages.

Table 6: PJ5, PJ6, PJ7, PJ8, PJ9, PJ10, PJ11 and PJ12 Pin Assignments

Jumper	PJ5 4B Bank	PJ6 2A Bank	PJ7 2B Bank	PJ8 4C Bank	PJ9 4D Bank	PJ10 2C Bank	PJ11 2E Bank	PJ12 2D Bank
Connect Pins 1 and 2	1.8 V (default)							
Connect Pins 3 and 4	1.2 V							

Header PJ13 and PJ14 (QSE GPIO Power Supply Selector)

PJ13 and PJ14 are 4-pin headers used to select the power supplies for the GPIO pins of the QSE interface.

To select the desired voltage supplies, connect these pins :

- For GPIO voltage of 1.8 V, connect pin 1 and pin 2.
- For GPIO voltage of 3.3 V, connect pin 3 and pin 4.

When the GPIOs are configured as inputs, you need to turn on the weak pull-up in the Efinity Interface Designer.

Table 7: PJ13 and PJ14 (QSE GPIO Power Supply Selector)

Header	Description
PJ13	QSE1 (P1) GPIO power supply source selector
PJ14	QSE0 (P2) GPIO power supply source selector

Header PJ17 and J22 (I^2C Bus to I^2C Buffer Connector)

PJ17 is a 4-pin header while J22 is a 2-pin header for connecting the I^2C bus to the I^2C buffer.

To connect the I^2C bus to the I^2C buffer, connect these pins:

- For PJ17 header, connect pin 1 and pin 2.
- For J22 header, connect pin 3 and pin 4.

Table 8: PJ17 and J22 (I^2C Bus to I^2C Buffer Connector)

Header	Description
PJ17	Connects the I^2C bus to the I^2C buffer.
J22	

Header PJ18 (FMC Power Supply Selector)

PJ18 is a 4-pin header for selecting a power supply of 1.8 V or 3.3 V for FMC_SCL and FMC_SDA.

Table 9: PJ18 (FMC Power Supply Selector)

Connect	Voltage
1 and 2	1.8 V
3 and 4	3.3 V

Header J1, J2, J3, J4, J5, J6, J7, J8, J9, and J10 (Power Supply Source Selector)

J1, J2, J3, J4, J5, J6, J7, J8, J9, and J10 are 2-pin rows of pins for power measurements. See **Titanium Ti375 C529 Development Board Schematics and BOM**. To use the external power supply, input the external power via the power supply source selectors. On the development board, the external power supply selector is near the output of each DC module.

Table 10: Header J1, J2, J3, J4, J5, J6, J7, J8, J9, and J10 (Power Supply Source Selector)

Header	Description
J1	1.1 V external power supply source
J2	1.2 V external power supply source
J3	0.6 V external power supply source
J4	1.8 V external power supply source
J5	3.3 V external power supply source
J6	5 V external power supply source
J7	1.8V_FMC external power supply source
J8	Ethernet PHY 1.0 V external power supply source
J9	0.85 V external power supply source
J10	0.95V external power supply source

Header J11 (GND Test)

J11 is a 2-pin row of pins for ground testing.

Table 11: Header J11 (GND Test)

Header	Description
J11	Single Board GND.

Header J12 (SMA)

The Titanium Ti375 C529 Development Board includes an SMA connector to connect to an optional external clock source. The J12 pin 1 connects to the GPIO pin in bank 2D.

By default, the SMA interface is not connected and the OSC1 oscillator is connected to the GPIOT_P_50_PLLIN0 pin.

CAUTION: Refer to the Titanium Ti375 C529 Development Board Schematics and BOM for details. Consult your nearest FAEs or contact Efinix technical personnel if you have any uncertainties when connecting the SMA to an external clock.

Table 12: Header J12 (SMA)

Pin Number	Ti375 Pin Name
1	GPIOT_P_50_PLLIN0

Header J13, J14, and J15 (PMOD)

The Titanium Ti375 C529 Development Board includes three 12-pin PMOD connectors of J13, J14, and J15. The PMOD interface support all PMOD types at 3.3 V.

Table 13: J13 (PMOD0) Pin Assignment

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375Pin Name
1	PMOD_A_IO_0	GPIOR_88_PLLIN1	7	PMOD_A_IO_4	GPIOR_92
2	PMOD_A_IO_1	GPIOR_89	8	PMOD_A_IO_5	GPIOR_93_PLLIN1
3	PMOD_A_IO_2	GPIOR_90	9	PMOD_A_IO_6	GPIOR_94
4	PMOD_A_IO_3	GPIOR_91	10	PMOD_A_IO_7	GPIOR_95_CLK12
5	GND	N.C.	11	GND	N.C.
6	3V3	N.C.	12	3V3	N.C.

Table 14: J14 (PMOD1) Pin Assignment

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375Pin Name
1	PMOD_B_IO_0	GPIOR_96_CLK13	7	PMOD_B_IO_4	GPIOR_100
2	PMOD_B_IO_1	GPIOR_97	8	PMOD_B_IO_5	GPIOR_101_PLLIN1
3	PMOD_B_IO_2	GPIOR_98	9	PMOD_B_IO_6	GPIOR_102
4	PMOD_B_IO_3	GPIOR_99	10	PMOD_B_IO_7	GPIOR_103
5	GND	N.C.	11	GND	N.C.
6	3V3	N.C.	12	3V3	N.C.

Table 15: J15 (PMOD2) Pin Assignment

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375Pin Name
1	PMOD_C_IO_0	GPIOR_104	7	PMOD_C_IO_4	GPIOR_108
2	PMOD_C_IO_1	GPIOR_105	8	PMOD_C_IO_5	GPIOR_140_PLLIN1
3	PMOD_C_IO_2	GPIOR_106	9	PMOD_C_IO_6	GPIOR_141
4	PMOD_C_IO_3	GPIOR_107	10	PMOD_C_IO_7	GPIOR_142_CLK8
5	GND	N.C.	11	GND	N.C.
6	3V3	N.C.	12	3V3	N.C.

Header J16 (I^2C Buffer - CSI and CSO)

J16 is a header used for daisy-chain loading of multiple FPGAs. For example, user can daisy-chain 2 FPGAs on 2 different boards.

Table 16: Header J16

Header	Description
J16	J16 is used for daisy-chain loading of multiple FPGAs.

Header J17 (FMC Connector)

J17 is a 400-pin FMC LPC interface connector for connecting the FMC daughter cards or adapter cards.

Note: Only the J17-1, J17-2, and J17-3 sub-pins are connected to the I/O pins in the Ti375. J17-4 and J17-5 are unused.

Table 17: J17-1 Pin Assignments

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
C1	GND	N.C.	D1	PG_FMC_1V8	N.C.
C2	N.C.	N.C.	D2	GND	N.C.
C3	N.C.	N.C.	D3	GND	N.C.
C4	GND	N.C.	D4	N.C.	N.C.
C5	GND	N.C.	D5	N.C.	N.C.
C6	GPIOB32_TO_FMC _P_P0CTL4	GPIOB_P_32	D6	GND	N.C.
C7	GPIOB32_TO_FMC _N_P1CTL4	GPIOB_N_32	D7	GND	N.C.
C8	GND	N.C.	D8	FMC_GLOBAL _CLK_2_P	GPIOB_P_44 _CLK11_P
C9	GND	N.C.	D9	FMC_GLOBAL _CLK_2_N	GPIOB_N_44 _CLK11_N
C10	GPIOB34_TO_FMC_P	GPIOB_P_34	D10	GND	N.C.
C11	GPIOB34_TO_FMC_N	GPIOB_N_34	D11	GPIOB41_TO_FMC_P	GPIOB_P_41
C12	GND	N.C.	D12	GPIOB41_TO_FMC_N	GPIOB_N_41
C13	GND	N.C.	D13	GND	N.C.
C14	GPIOB30_TO_FMC_P	GPIOB_P_30 _CDI9_PLLIN0	D14	GPIOB42_TO_FMC_P	GPIOB_P_42
C15	GPIOB30_TO_FMC_N	GPIOB_N_30_CDI8	D15	GPIOB42_TO_FMC_N	GPIOB_N_42
C16	GND	N.C.	D16	GND	N.C.
C17	GND	N.C.	D17	GPIOB24_TO_FMC_P	GPIOB_P_24_CDI15
C18	GPIOB29_TO_FMC_P	GPIOB_P_29 _CDI11_EXTFB	D18	GPIOB24_TO_FMC_N	GPIOB_N_24_CDI14
C19	GPIOB29_TO_FMC_N	GPIOB_N_29_CDI10	D19	GND	N.C.
C20	GND	N.C.	D20	FMC_GLOBAL_CLK_3 _P_FMC_P0SDA	GPIOB_P_13_CDI21 _CLK0_P
C21	GND	N.C.	D21	FMC_GLOBAL_CLK_3 _N_FMC_P0SCL	GPIOB_N_13_CDI20 _CLK0_N
C22	FMC_GLOBAL_CLK_1 _P_FMC_SPI_SCK	GPIOB_P_15 _CDI17_CLK2_P	D22	GND	N.C.

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
C23	FMC_GLOBAL_CLK_1_N_FMC_FPGA_SSN	GPIOB_N_15_CDI16_CLK2_N	D23	GPIOB14_TO_FMC_P	GPIOB_P_14_CDI19_CLK1_P
C24	GND	N.C.	D24	GPIOB14_TO_FMC_N	GPIOB_N_14_CDI18_CLK1_N
C25	GND	N.C.	D25	GND	N.C.
C26	GPIOB09_TO_FMC_P	GPIOB_P_09	D26	GPIOB12_TO_FMC_P	GPIOB_P_12
C27	GPIOB09_TO_FMC_N	GPIOB_N_09	D27	GPIOB12_TO_FMC_N	GPIOB_N_12
C28	GND	N.C.	D28	GND	N.C.
C29	GND	N.C.	D29	FMC_TCK	N.C.
C30	FMC_SCL	N.C.	D30	FMC_TDI	N.C.
C31	FMC_SDA	N.C.	D31	FMC_TDO	N.C.
C32	GND	N.C.	D32	3V3	N.C.
C33	GND	N.C.	D33	FMC_TMS	N.C.
C34	GND	N.C.	D34	N.C.	N.C.
C35	12V	N.C.	D35	GND	N.C.
C36	GND	N.C.	D36	3V3	N.C.
C37	12V	N.C.	D37	GND	N.C.
C38	GND	N.C.	D38	3V3	N.C.
C39	3V3	N.C.	D39	GND	N.C.
C40	GND	N.C.	D40	3V3	N.C.

Table 18: J17-2 Pin Assignments

Pin Number	Signal Name	Pin Number	Signal Name
E1	GND	F1	N.C.
E2	N.C.	F2	GND
E3	N.C.	F3	GND
E4	GND	F4	N.C.
E5	GND	F5	N.C.
E6	N.C.	F6	GND
E7	N.C.	F7	N.C.
E8	GND	F8	N.C.
E9	N.C.	F9	GND
E10	N.C.	F10	N.C.
E11	GND	F11	N.C.
E12	N.C.	F12	GND
E13	N.C.	F13	N.C.
E14	GND	F14	N.C.
E15	N.C.	F15	GND

Pin Number	Signal Name	Pin Number	Signal Name
E16	N.C.	F16	N.C.
E17	GND	F17	N.C.
E18	N.C.	F18	GND
E19	N.C.	F19	N.C.
E20	GND	F20	N.C.
E21	N.C.	F21	GND
E22	N.C.	F22	N.C.
E23	GND	F23	N.C.
E24	N.C.	F24	GND
E25	N.C.	F25	N.C.
E26	GND	F26	N.C.
E27	N.C.	F27	GND
E28	N.C.	F28	N.C.
E29	GND	F29	N.C.
E30	N.C.	F30	GND
E31	N.C.	F31	N.C.
E32	GND	F32	N.C.
E33	N.C.	F33	GND
E34	N.C.	F34	N.C.
E35	GND	F35	N.C.
E36	N.C.	F36	GND
E37	N.C.	F37	N.C.
E38	GND	F38	GND
E39	1V8_FMC	F39	GND
E40	GND	F40	1V8_FMC

Table 19: J17-3 Pin Assignments

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
G1	GND	N.C.	H1	FMC_VREF	N.C.
G2	FMC_M2C_CLK1_P	GPIOB_P_17_CLK4_P	H2	FMC_PRESENT_n	GPIOB_P_26 _EXTSPICLK
G3	FMC_M2C_CLK1_N	GPIOB_N_17_CLK4_N	H3	GND	N.C.
G4	GND	N.C.	H4	FMC_M2C_CLK2_P	GPIOT_P_40 _CLK23_P
G5	GND	N.C.	H5	FMC_M2C_CLK2_N	GPIOT_N_40 _CLK23_N
G6	FMC_GLOBAL _CLK_4_P	GPIOB_P_20_CLK7_P	H6	GND	N.C.

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
G7	FMC_GLOBAL _CLK_4_N	GPIOB_N_20_CLK7_N	H7	GPIOB35_TO_FMC_P	GPIOB_P_35
G8	GND	N.C.	H8	GPIOB35_TO_FMC_N	GPIOB_N_35
G9	GPIOB27_TO_FMC_P	GPIOB_P_27	H9	GND	N.C.
G10	GPIOB27_TO_FMC_N	GPIOB_N_27	H10	GPIOB21_TO_FMC_P	GPIOB_P_21
G11	GND	N.C.	H11	GPIOB21_TO_FMC_N	GPIOB_N_21
G12	GPIOB31_TO_FMC_P	GPIOB_P_31	H12	GND	N.C.
G13	GPIOB31_TO_FMC_N	GPIOB_N_31	H13	GPIOB18_TO_FMC_P	GPIOB_P_18_CLK5_P
G14	GND	N.C.	H14	GPIOB18_TO_FMC_N	GPIOB_N_18_CLK5_N
G15	GPIOB23_TO_FMC_P	GPIOB_P_23_CBSEL1	H15	GND	N.C.
G16	GPIOB23_TO_FMC_N	GPIOB_N_23_CBSEL0	H16	GPIOB19_TO_FMC_P	GPIOB_P_19 _NSTATUS_CLK6_P
G17	GND	N.C.	H17	GPIOB19_TO_FMC_N	GPIOB_N_19 _TEST_N_CLK6_N
G18	GPIOB11_TO_FMC_P	GPIOB_P_11	H18	GND	N.C.
G19	GPIOB11_TO_FMC_N	GPIOB_N_11	H19	GPIOB06_TO_FMC_P	GPIOB_P_06 _CDI26_EXTFB
G20	GND	N.C.	H20	GPIOB06_TO_FMC_N	GPIOB_N_06_CDI25
G21	GPIOB33_TO_FMC _P_FMC_P1SDA	GPIOB_P_33	H21	GND	N.C.
G22	GPIOB33_TO_FMC _N_FMC_P1SCL	GPIOB_N_33	H22	GPIOB25_TO_FMC _P_FMCSPI_nCS0	GPIOB_P_25_CDI13
G23	GND	N.C.	H23	GPIOB25_TO_FMC _N_FMCSPI_nCS1	GPIOB_N_25_CDI12
G24	GPIOB22_TO_FMC _P_P0CTL0	GPIOB_P_22	H24	GND	N.C.
G25	GPIOB22_TO_FMC _N_P0CTL1	GPIOB_N_22	H25	GPIOB28_TO_FMC _P_P1CTL1_CDONE	GPIOB_P_28
G26	GND	N.C.	H26	GPIOB28_TO_FMC _N_P1CTL0 _INIT_RST_N	GPIOB_N_28
G27	GPIOB08_TO_FMC _P_P0CTL2	GPIOB_P_08 _CDI22_EXTFB	H27	GND	N.C.
G28	GPIOB08_TO_FMC _N_P0CTL3	GPIOB_N_08	H28	GPIOB43_TO_FMC _P_P1CTL3	GPIOB_P_43 _CSO_CLK10_P
G29	GND	N.C.	H29	GPIOB43_TO_FMC _N_P1CTL2_PROG_N	GPIOB_N_43 _CSI_CLK10_N
G30	GPIOB10_TO_FMC _P_P0CTL5	GPIOB_P_10	H30	GND	N.C.
G31	GPIOB10_TO_FMC _N_P0CTL6	GPIOB_N_10	H31	GPIOB05_TO _FMC_P_P1CTL5	GPIOB_P_05 _CDI28_PLLIN0

Pin Number	Signal Name	Ti375 Pin Name	Pin Number	Signal Name	Ti375 Pin Name
G32	GND	N.C.	H32	GPIOB05_TO _FMC_N_P1CTL6	GPIOB_N_05_CDI27
G33	GPIOB16_TO_FMC _P_FMCSPIDAT1	GPIOB_P_16_CLK3_P	H33	GND	N.C.
G34	GPIOB16_TO_FMC _N_FMCSPIDAT0	GPIOB_N_16_CLK3_N	H34	GPIOB04_TO_FMC _P_FMCSPIDATA3	GPIOB_P_04 _CDI30_EXTFB
G35	GND	N.C.	H35	GPIOB04_TO_FMC _N_FMCSPIDATA2	GPIOB_N_04_CDI29
G36	GPIOB07_TO_FMC _P_FMCSPIDAT5	GPIOB_P_07 _CDI24_PLLIN0	H36	GND	N.C.
G37	GPIOB07_TO_FMC _N_FMCSPIDAT4	GPIOB_N_07_CDI23	H37	GPIOB03_TO_FMC _P_FMCSPIDATA7	GPIOB_P_03_PLLIN0
G38	GND	N.C.	H38	GPIOB03_TO_FMC _N_FMCSPIDATA6	GPIOB_N_03_CDI31
G39	1V8_FMC	N.C.	H39	GND	N.C.
G40	GND	N.C.	H40	1V8_FMC	N.C.

Header J18 (JTAG)

J18 is a 10-pin JTAG interface. You can access the Ti375 JTAG pins through this interface.

Table 20: J18 Pin Assignment

Pin Number	Signal Name
1	TDO
2	3.3V
3	TCK
4	TDI
5	TMS
6	FTDI_RST
7	N.C.
8	CRESET_N
9	GND
10	GND

Header J19 (Fan Connector)

J19 is a 4-pin 12 V fan connector.

Table 21: J19 Pin Assignment

Pin Number	Pin Name	Description
1	GND	DC power negative
2	+12 V	DC power positive
3	TACHO	Fan speed detection
4	PWM	Fan motor speed control

Header J20 (FPGA Programming via FMC)

J20 is a 2-pin connector to allow the programming of the FPGA through the FMC daughter cards. To enable programming through the FMC connector, use a jumper to connect pins 1 and 2.

Header J21 (LPDDR4/4x)

J21 is a 2-pin header to configure the input voltage to support either LPDDR4 or LPDDR4x. To use LPDDR4, use the jumper to connect pins 1 and 2. To use LPDDR4x, leave the J21 header unconnected and connect both R475 and R165.

Header RJ1 (Gigabit Ethernet Port)

Titanium Ti375 C529 Development Board provides a Gigabit Ethernet transceiver from Renesas (RTL8211FDI), which is compatible with 10 Base-T, 100 Base-TX, and 1,000 Base-T IEEE 802.3 standards. The chip supports:

- RGMII MAC interface
- Support for 120 meters of 1,000 Base-T CAT.5 cable
- Automatic polarity correction
- Low drop voltage regulator

The RJ1 header is connected to Ethernet PHY (U50).

Table 22: U50 ETH0 PHY Pin Assignment

Pin Name	Signal Name	Description
TXC	ETH0_TXC	Transmit reference clock can be 125 MHz, 25 MHz or 2.5 MHz at different rates.
TXD0	ETH0_TXD0	Data is transferred from the MAC to the PHY via TXD[3:0].
TXD1	ETH0_TXD1	Data is transferred from MAC to PHY via TXD[3:0].
TXD2	ETH0_TXD2	Data is transmitted from MAC to PHY via TXD[3:0].
TXD3	ETH0_TXD3	Data is transmitted from MAC to PHY via TXD[3:0].
TXCTL	ETH0_TXCTL	Signal is received from MAC.
RXC/PHYAD1	ETH0_RXC_10M	Continuously receiving reference clock can be 2.5 MHz, 25 MHz, or 125 MHz from the receiving data stream.
RXD0/RXDLY	ETH0_RXD0	Receives data. Data is transferred from PHY to MAC via RXD[3:0].
RXD1/TXDLY	ETH0_RXD1	Receives data. Data is transferred from PHY to MAC via RXD[3:0].
RXD2/PLLOFF	ETH0_RXD2	Receives data. Data is transmitted from PHY to MAC via RXD[3:0].
RXD3/PHYAD0	ETH0_RXD3	Receives data. Data is transferred from PHY to MAC via RXD[3:0].
RXCTL/PHYAD2	ETH0_RXCTL	Transmits control signal to the MAC.
PHYRSTB	ETH0_RESETN	Hardware reset active low. A complete PHY reset requires this pin to be pulled low for at least 10 ms. All registers are cleared after a hardware reset.
MDC	ETH0_MDC	Manages the data clock.
MDIO	ETH0_MDIO	Manages data input/output. 3.3, 2.5, 18, and 1.5 V RGMII. Pull-up 3.3,2.5, 1.8, and 1.5 V respectively.
INTB/PMEB	ETH0_INTN	Interrupt (supports 3.3 V pull-up). Set to low if status changes; active low. Power management event (supports 3.3 V pull-up). Set to low if a packet or wake-up frame is received; active low. Keep this pin floating if unused.
CLKOUT	GE_PHY_CLKOUT	The 125 or 25 MHz reference clock is generated by the internal PLL. Keep the pin floating if the clock is unused in the MAC.
XTAL_OUT/EXT_CLK	ETH0_REFCLK_25MHZ	25 MHz crystal output. If the 25 MHz oscillator is used, connect XTAL_OUT/EXT_CLK to the oscillator output.

SD1 (Micro-SD Card Slot)

The Titanium Ti375 C529 Development Board includes a micro-SD card slot, SD1. SD1 connects to GPIO pins in bank TR. The micro-SD supports data rate of up-to 25 MBps.

Table 23: SD1 Pin Assignments

Pin Name	Signal Name	Ti375 Pin Name
DATA2	SD_DATA2	GPIOR_170
DATA3	SD_DATA3	GPIOR_171
CMD	SD_CMD	GPIOR_167
VDD	3V3	-
CLK	SD_CLK	GPIOR_166
GND	GND	-
DATA0	SD_DATA0	GPIOR_168
DATA1	SD_DATA1	GPIOR_169
Card-detect	TFCard_Detect	GPIOT_P_49_EXTFB
CGND	GND	-
CGND	GND	-
CGND	GND	-
CGND	GND	-

Header USB1 (USB FTDI FT4232H)

USB1, a type-C USB receptacle, is the interface between the board and your computer for communication through the FTDI FT4232H chip. Connect the type-C USB cable for configuring the Ti375 FPGA and NOR flash. Refer to Overview on page 8 for more details on the FTDI FT4232H module interface.

Headers TP1, TP2, TP3, TP4, TP5, TP6, TP7, TP8, TP9, TP10, TP11, TP12, TP13, TP14, and, TP15 (Power Supply Source Selector)

TP1, TP2, TP3, TP4, TP5, TP6, TP7, TP8, TP9, TP10, TP11, TP12, TP13, TP14, and TP15 are headers for power measurements. Refer to the **Titanium Ti375 C529 Development Board Schematics** for more information.

Table 24: TP1, TP2, TP3, TP4, TP5, TP6, TP7, TP8, TP9, TP10, TP11, TP12, TP13, TP14, and TP15 (Power Supply Source Selector)

Header	Description
TP1, TP2	0.95 V power supply source selector
TP3, TP4	3.3 V power supply source selector
TP5, TP6	0.85 V power supply source selector
TP7	1.8 V power supply source selector
TP8	1.2 V power supply source selector
TP9	5 V power supply source selector
TP10, TP11	Ethernet PHY 1.0 V power supply source selector
TP12, TP13	LPDDR4 1.1 V power supply source selector
TP14, TP15	FMC Interface 1.8 V power supply source selector

Headers TP16, TP17, TP18, and TP19 (DDR PLL Selector)

TP16, TP17, TP18, and TP19 are PLL output test points for DDR. Refer to the **Titanium Ti375 C529 Development Board Schematics and BOM** for more information.

Table 25: TP16, TP17, TP18, and TP19 (DDR PLL Selector)

Header	Description
TP16, TP17, TP18, TP19	PLL output test points for DDR

User Outputs

The board has 5 user LEDs that are connected to I/O pins in Ti375 banks BL2, 2B, 2D, and 2E. By default, the Ti375 I/O connected to these LEDs are active high. To turn a given LED on, pull the corresponding I/O signal high.

Note: When adding these GPIO in the Efinity® Interface Designer, configure them as output pins.

Table 26: User Outputs

Reference Designator	Ti375 Pin Name	Active
LED9	GPIOL_23	High
LED10	GPIOL_24	High
LED11	GPIOT_N_50	High
LED12	GPIOT_N_19	High
LED13	GPIOT_N_64_CLK15_N	High

User Inputs

The board has 2 pushbutton switches that you can use as inputs to the Ti375 FPGA. The pushbuttons are connected to I/O pins in Ti375 bank 2B and TL1. When building designs using these switches, turn on an internal pull up for these pins in the Interface Designer.

When you press the pushbutton switches the signal drives low, indicating user input.

Table 27: User Pushbuttons

Reference Designator	Ti375 Pin Name	Active
SW3	GPIOL_52_PLLIN1	Low
SW4	GPIOT_P_19	Low

Installing Standoffs

Before using the board, attach the standoffs with the screws and nuts provided in the kit. The following table lists the standoffs, screws, and nuts required for standoffs installation.

Note: Always power off the development board before attaching the standoffs.

Table 28: Standoffs, Screws, and Nuts for Standoff Installation

Standoff			Screw			Nut	
Size	Length	Qty	Size	Length	Qty	Size	Qty
M3	12 mm	4	M3	18 mm	4	M3	8

Warning: You can damage the board if you over tighten the screws. Tighten all screws to a torque between 4 ± 0.5 kgf/cm and 5 ± 0.5 kgf/cm.

Heat Dissipation

You can use the Power Estimator to estimate the power consumption of your design and estimate the junction temperature based on the thermal coefficient of the package. If the junction temperature is likely to be greater than the junction temperature specified in the specification, you should consider cooling the board.

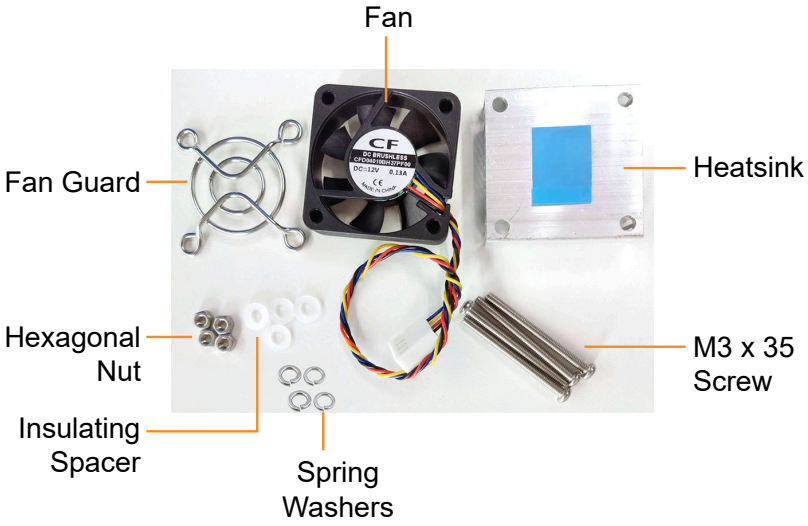
You can choose to install a heatsink only or a heatsink and a fan.

Note: You need a license to obtain the **Power Estimator** Excel file from Efinix® Support Center.

Installing Heatsink and Fan

Materials to prepare: Heatsink, fan, shield, M3 x 35 screws, insulating spacers, spring washers, and hexagonal nuts.

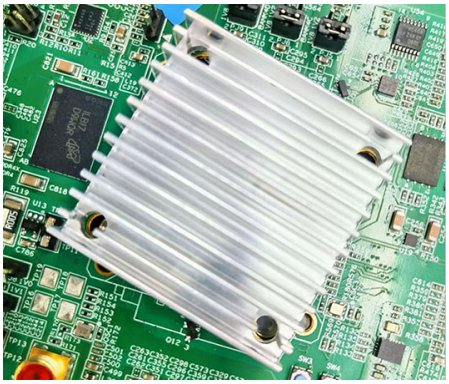
Figure 3: Materials for Installing Heatsink and Fan



Follow these steps to install the heatsink and fan:

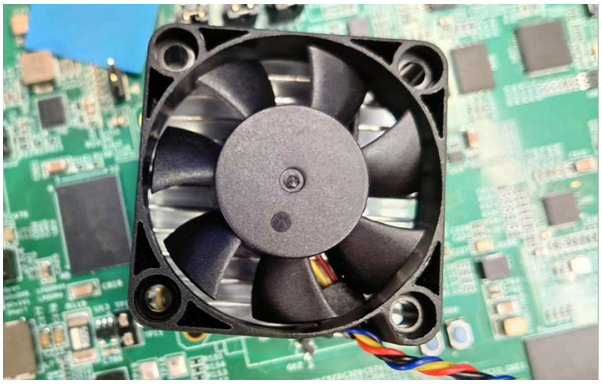
1. Remove the protective film from the heatsink, align the heatsink with the fan mounting holes on the PCB, and place the heatsink correctly on the FPGA.

   Figure 4: Mounting the Heatsink on the PCB

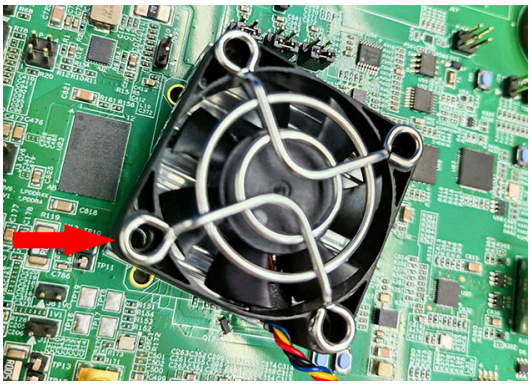
2. Place the side of the fan (with logo) on the heatsink, and properly align the fan's mounting holes with the heatsink.

Figure 5: Place Fan (with Logo) on the Heatsink



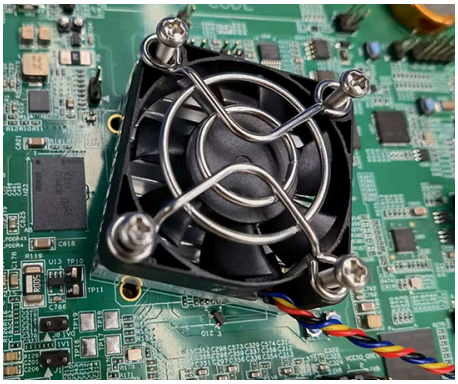
3. Place the fan guard over the fan with the mounting holes properly aligned with the heatsink and fan.

Figure 6: Place Fan Guard on the Fan



4. Insert four M3 x 35 screws into the aligned holes on the PCB.

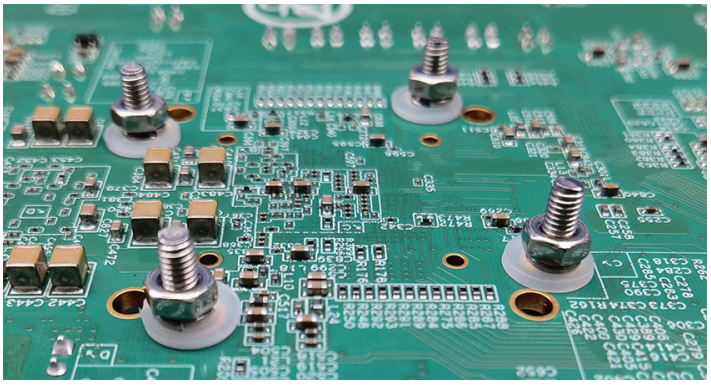
Figure 7: Place four M3 x 35 screws into the Aligned Holes



5. Hold the screws in place and turn the PCB upside down. Insert the insulating spacers, followed by spring washers and hex nuts on all four screws accordingly. Then, tighten the screws with equal force.

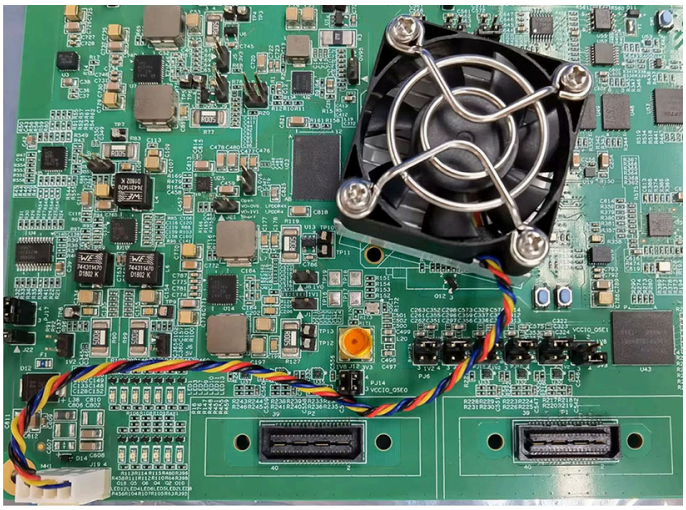
Warning: You can damage the board if you over-tighten the screws. Tighten all screws to a torque between 4 ± 0.5 kgf/cm and 5 ± 0.5 kgf/cm.

Figure 8: Insert Insulating Spacers, Spring Washers, and Hex Nuts



6. Plug in the fan's power supply.

Figure 9: Completed Heatsink and Fan Installation



Installing Heatsink

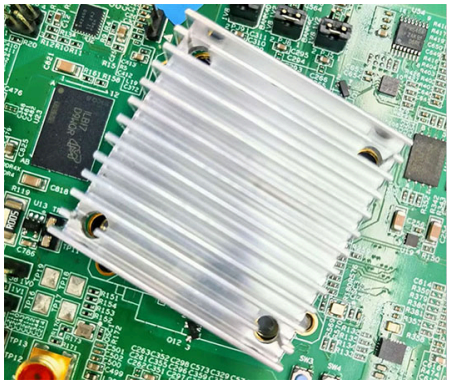
Materials to prepare: Heatsink, M3 x 20 screws, insulating spacers, and hexagonal nuts.

Figure 10: Materials for Installing Heatsink

Follow these steps to install the heatsink:

1. Remove the protective film from the heatsink, align the heatsink with the fan mounting holes on the PCB, and place the heatsink correctly on the FPGA.

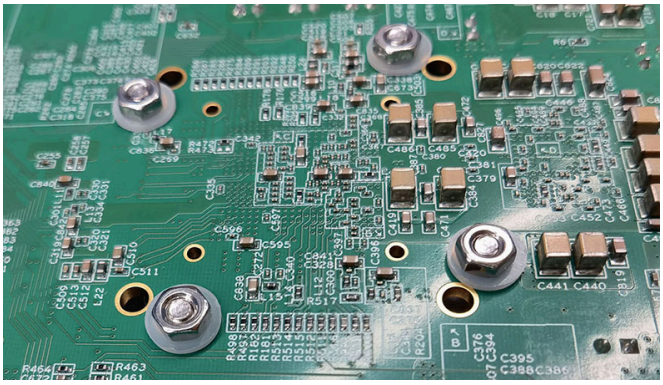
   Figure 11: Mounting the Heatsink on the PCB

2. Insert four M3 x 20 screws into the aligned holes on the PCB. Hold the screws in place and turn the PCB upside down. Insert the insulating spacers, followed by hexagonal nuts on all four screws accordingly. Then, tighten the screws with equal force. When tightening the screws, tighten them diagonally in turn to ensure a balanced force on the heat sink.

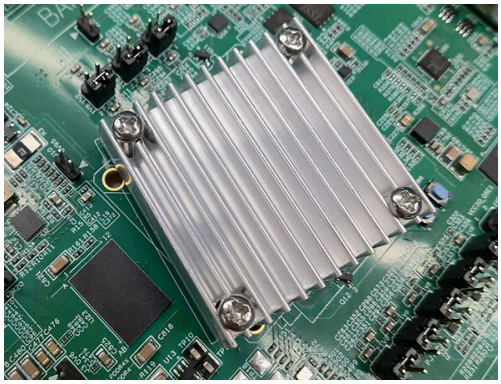
Warning: You can damage the board if you over-tighten the screws. Tighten all screws to a torque between 4 ± 0.5 kgf/cm and 5 ± 0.5 kgf/cm.

Figure 12: Insert Insulating Spacers, Spring Washers, and Hex Nuts



The installation is complete. See the following figure.

Figure 13: A Completely Installed Heatsink

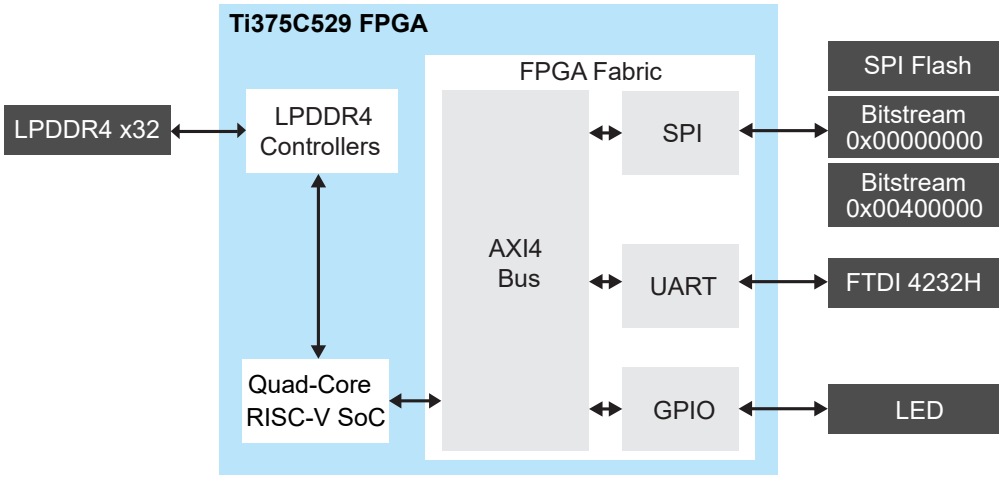


Titanium Ti375 C529 Development Board Example Design

Efinix preloads the Titanium Ti375 C529 Development Board with an example design that demonstrates the following functions:

- Using CPU 0 to calculate the moving positions of a spinning donut and render the image through the UART.
- Using CPU 1 to make an alternate blink on 5 LEDs to show a running light effect by blinking the LEDs in an alternate pattern.

Figure 14: Example Design Block Diagram Overview



The Spinning Donut Program

This program was invented by Andy Sloane in 2006. It draws a spinning donut in ASCII characters and sends the data through a UART to a PC terminal. Andy Sloane explained the math behind the program to do the 3D rendering and its movement in his article, **Donut math: how donut.c works**.

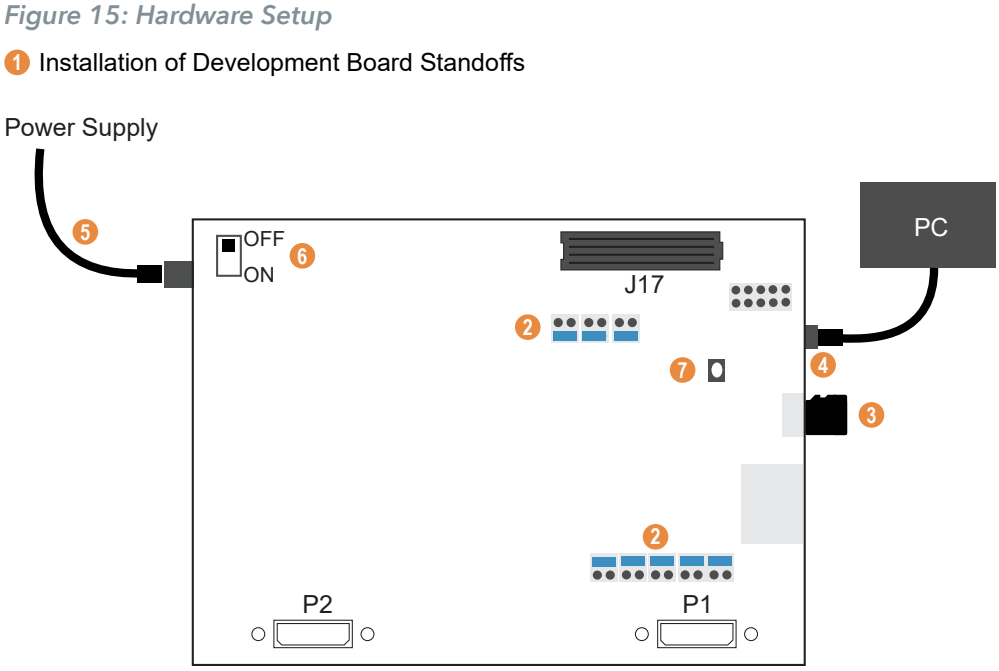
The Running Light Program

This program demonstrates that the quad-core RISC-V example design is actively running on the development board. The LEDs light up in series with a short delay between each one. the default direction is from left to right.

Set Up the Hardware

The following figure shows the hardware setup steps:

Figure 15: Hardware Setup



Important: Always switch off the power supply and board's power switch before attaching or detaching cables and daughter cards.

1. Install standoffs on the Titanium Ti375 C529 Development Board as described in Installing Standoffs.
2. Ensure that the jumpers of the development board are set as follows:

Board	Header	Pins to Connect
Titanium Ti375 C529 Development Board	PJ15, PJ6, PJ7, PJ8, PJ9, PJ10,PJ11, PJ12	1 - 2

3. Connect the USB header to a USB port on your computer.
4. Ensure the power supply and board's power switch (PSW1) are turned off, then connect the 12 V power cable to the board connector and a power source.
5. Turn on the board's power switch (PSW1).
6. Press push button SW1 (CRESET_N).

The board LEDs start blinking in series from left to right.

Table 29: Board LED Outputs

LED	Description
LED1 - LED8 turned on	Power good
LED9 blinking	FPGA configuration done
LED10 blinking	Configuration done
LED11 blinking	Configuration done
LED12 blinking	Configuration done

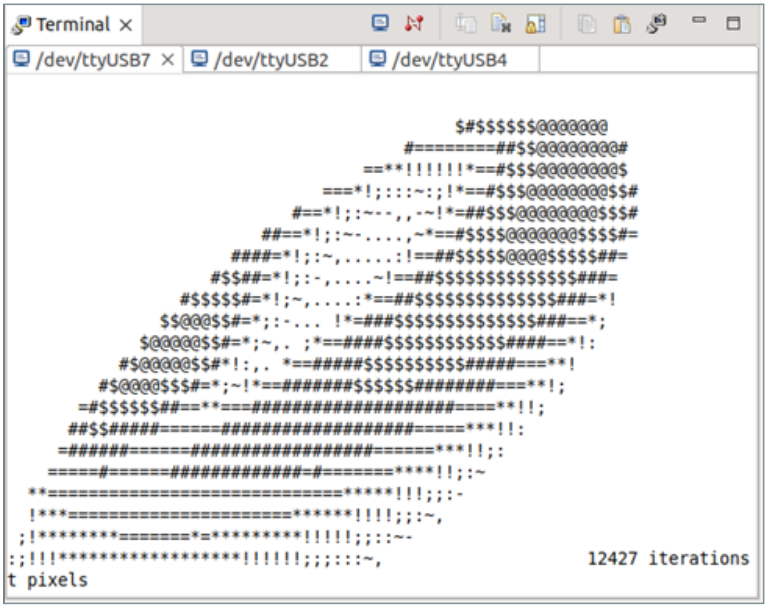
LED	Description
LED13 blinking	Configuration done

Using the Spinning Donut Program

You need to set up a terminal on your computer before using this program.

1. Open terminal software on the computer. You can use any Windows or Linux terminal application such as PuTTy, Tera Term, Minicom, and others.
2. Select the available USB serial COM port and set the following:
   - Baud rate: 115200
   - Data: 8-bit
   - Parity bit: No
   - Stop bit: 1
3. Press push button SW4 to start streaming the donut program through the UART as shown in the following display.

Figure 16: The Spinning Donut Output



Using the Running Light Program

By default, the LEDs blink from left to right. To toggle the direction of the blinking LEDs, press the push button SW4 on the Titanium Ti375 C529 Development Board. Each LED has a default blinking period of 500 ms; the period changes after the fifth LED blinks. The blinking period is 500 ms, 250 ms, 100 ms, and 50 ms. Then, the blinking restarts again with a 500 ms interval. The LED blinks faster every time the interval period changes to a smaller value.

Creating Your Own Design

The Titanium Ti375 C529 Development 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.

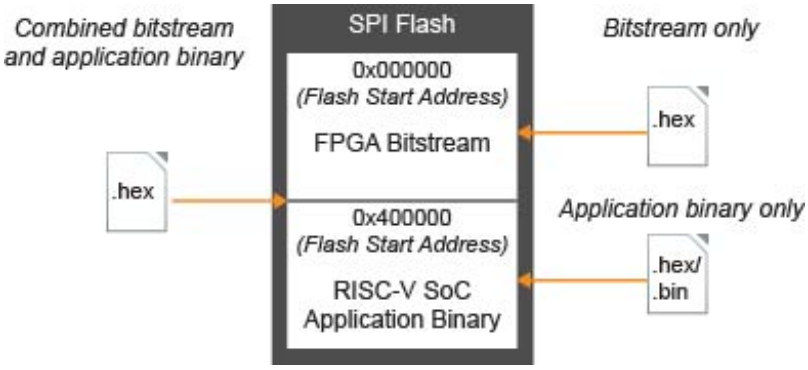
Note: The preloaded example design project file is available in the Titanium Ti375 C529 Development Board Demonstration Design page. The example design page indicates the required Efinity software version to compile the design.

The example design consists of the FPGA bitstream and the RISC-V SoC application binary. You need to program these files into the flash device on the board if you want to restore the example design. There are two ways of programming the files into the SPI flash:

- Program the FPGA bitstream and the application binary together as a combined bitstream
- Program the FPGA bitstream and the application binary separately

Generally, you use the combined bitstream to restore the entire example design. Use the separate files if you want to restore either the FPGA bitstream or the application binary portion only.

Figure 17: Example Design SPI Flash Content



Example Design Files

The example design includes the following design files.

Table 30: Design Example File and Directories

File or Directory	Description
bsp\TI375C529_DK\ti375_oob.xml	Example design project file.
bsp\TI375C529_DK\Bitstream\RestoreBitstream \FPGA_TI375C529_oob.hex	FPGA bitstream only. Program this file into the SPI flash using SPI active configuration mode.
bsp\TI375C529_DK\Bitstream\RestoreBitstream \FW_TI375C529_oob.hex	RISC-V SoC application binary only. Program this file into the SPI flash using SPI active configuration mode.
bsp\TI375C529_DK\Bitstream\RestoreBitstream \FPGA_TI375C529_oob.bit	FPGA bitstream only. Use this file to configure the Ti375 FPGA using JTAG mode configuration.
bsp\TI375C529_DK\Bitstream\FlashLoader \jtag_spix8_flash_loader_ti375c529.bit	Pre-compiled JTAG bridge image file.
bsp\TI375C529_DK\Bitstream\SocFW\ti375_oob.elf	Pre-compiled example design application binary file to run using OpenOCD Debugger.
bsp\TI375C529_DK\Bitstream\SocFW\ti375_oob.bin	Pre-compiled example design application binary file to program into SPI flash using OpenOCD Debugger.
bsp\TI375C529_DK\Bitstream\BootLoader \bootloader.hex	Pre-compiled SPI flash bootloader binary file. The example design does not use the default High-Performance Sapphire RISC-V SoC SPI flash bootloader. Use this bootloader to overwrite the default SPI flash bootloader if you regenerate the Sapphire RISC-V SoC in IP Manager.
bsp\TI375C529_DK\Bitstream\BootLoader \bootloader.elf	Pre-compiled bootloader binary file to run using OpenOCD Debugger.
bsp\TI375C529_DK\embedded_sw\soc_ti375_oob	RISC-V SoC workspace. Use the files in **soc_ti375_oob** if you are using the Efinity RISC-V IDE.
bsp\TI375C529_DK\embedded_sw\soc_ti375_oob \software\standalone\ti375_oob	RISC-V SoC example design project files.
bsp\TI375C529_DK\embedded_sw\soc_ti375_oob \software\standalone\bootloader	RISC-V SoC bootloader project files.

Programming the Development Board

Before programming the board, connect the Titanium Ti375 C529 Development Board to your computer using a USB cable and power-on the board. Refer to Table 30: Design Example File and Directories on page 41 .

Combined Bitstream

To program the combined bitstream into the SPI flash:

1. Choose the **USB Target**.
2. In the **Image** box, click the **Select Image File** button and select **Combine_TI375C529_oob.hex**.
3. Choose the **SPI Active using JTAG Bridge (New)** or **SPI Active x8 using JTAG Bridge (New)** programming mode.
4. Turn on the **Auto configure JTAG Bridge Image** option, click the **Select Image File** button and select the generated **.bit** file.
5. Ensure that the **Starting Flash Address** is set to `0x000000`.
6. Click **Start Program** button. The Programmer first configures the FPGA and then programs the flash device.

Separate FPGA Bitstream and Application Binary

To program the FPGA bitstream into the SPI flash:

1. Choose the **USB Target**.
2. In the **Image** box, click the **Select Image File** button and select **FPGA_TI375C529_oob.hex**.
3. Choose the **SPI Active using JTAG Bridge (New)** or **SPI Active x8 using JTAG Bridge (New)** programming mode.
4. Turn on the **Auto configure JTAG Bridge Image** option, click the **Select Image File** button and select the generated **.bit** file.
5. Under the **SPI Active Options (x1)** or **SPI Active Options (x8)**, select **Lower for Select Flash**.
6. Ensure that the **Starting Flash Address** is set to `0x000000`.
7. Click **Start Program** button. The Programmer first configures the FPGA and then programs the flash device.

To program the application binary into the SPI flash:

1. Choose the **USB Target**.
2. In the **Image** box, click the **Select Image File** button and select **FW_TI375C529_OOB.hex**.
3. Choose the **SPI Active using JTAG Bridge (New)** or **SPI Active x8 using JTAG Bridge (New)** programming mode.
4. Turn on the **Auto configure JTAG Bridge Image** option, click the **Select Image File** button and select the generated **.bit** file.
5. Under the **SPI Active Options (x1)** or **SPI Active Options (x8)**, select **Upper for Select Flash**.
6. Ensure that the **Starting Flash Address** is set to `0x000000`.
7. Click **Start Program** button. The Programmer first configures the FPGA and then programs the flash device.

Revision History

Table 31: Revision History

Date	Version	Description
June 2024	1.0	Initial release.