C2000 F29H85x Series LaunchPad Development Kit

Description

The LAUNCHXL-F29H85X is a low-cost development board for the Texas Instruments C2000™ Real-Time Microcontroller series of F29H85x devices. The LAUNCHXL-F29H85X is designed around the F29H850TU real-time microcontroller (MCU) and highlights the control, analog, and communications peripherals, as well as the integrated nonvolatile memory. The LaunchPad™ Development Kit also features two independent BoosterPack XL expansion connectors (80-pins), dedicated 12-bit/16-bit differential ADC header, PMIC, on-board Controller Area Network (CAN) transceiver supporting both classic CAN and CAN-FD, two 5V encoder interface (eQEP) connectors, FSI connector, SENT connectors, power-domain isolation, and an on-board XDS110 debug probe.

Get Started

1. Order the Launchxl-F29H85X evaluation board (EVM).
2. Download the latest Code Composer Studio™ integrated development environment (IDE), F29H85X-SDK software development kit, Python, and OpenSSL.

Features

- C2000 series F29H850TU9SRFS (144-pin) microcontroller
  - With Configurable Logic Block (CLB) capability
- On-board XDS110 debug probe
- Two user-controlled LEDs
- One microcontroller reset switch
- Selectable power domains:
  - BoosterPack™ Plug-in Module
  - External power supply (5V or 3.3V)
- 12-bit/16-bit differential Analog-to-Digital Converter (ADC) header
  - The on-board REF5030 supports all 12-bit AND 16-bit ADC modes
- CAN connector with on-board CAN transceiver
  - Supports both classic CAN and CAN-FD
- Two independent Enhanced Quadrature Encoder Pulse (QEP)-based encoder connectors
- FSI peripheral connector
- SENT connectors
- Two independent BoosterPack XL standard connectors (80-pins) featuring stackable headers to maximize expansion through the BoosterPack ecosystem

F29H85x LaunchPad Board Overview

Copyright © 2025 Texas Instruments Incorporated
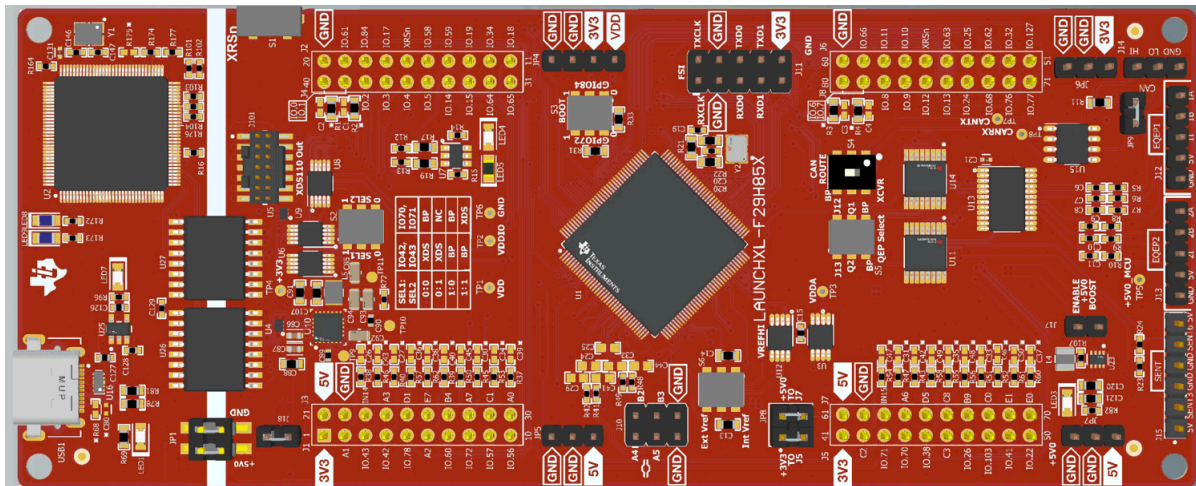
1 Evaluation Module Overview

1.1 Introduction

The F29H85X LaunchPad (LAUNCHXL-F29H85X) from Texas Instruments (TI) provides a great way to learn and experiment with the F29H85x device. The F29H85x device is a member of TI's C2000 family of microcontrollers (MCUs). This 80-pin LaunchPad is intended to provide a well-filtered design capable of working in most environments. This document provides the hardware details of the F29H85x LaunchPad and explains the functions, locations of jumpers, and connectors present on the board.

Note

This kit is designed to explore the functionality of the F29H85x microcontroller. The LaunchPad can be treated as a good reference design and is not intended to be a complete customer design. Full compliance to safety, EMI/EMC, and other regulations are left to the designer of the customer's system.

1.2 Kit Contents

The F29H85x Series LaunchPad Development Kit contains these items:

- C2000 F29H85x Series LaunchPad development board (LAUNCHXL-F29H85X)
- USB Type-A male to USB Type-C™ male cable
- Pinout Map

1.3 Specifications

Table 1-1 summarizes the F29H85X LaunchPad specifications.

Table 1-1. LAUNCHXL-F29H85X Specifications

Parameter	Value
Board Supply Voltage	$5V_{DC}$ from one of the following sources: • USB Connector (USB1) - USB Type-C cable connected to PC or other compatible power source. • BoosterPack 1 • BoosterPack 2 • Auxiliary power connectors $3.3V_{DC}$ from one of the following sources: • BoosterPack 1 • BoosterPack 2 • Auxiliary power connectors
Dimensions	5.67in x 2.3in x 0.75in (14.4cm x 5.84cm x 1.90cm) (L x W x H)
Break-out Power Output	• $3.3V_{DC}$ to BoosterPacks, limited by output of LDO inside the PMIC TPS65036. This 3.3V plane is shared with on-board components. Total output power limit of TPS65036 is 300mA.
Assumed Operating Conditions	This kit is assumed to run at standard room conditions. The EVM can run at approximately standard ambient temperature and pressure (SATP) with moderate-to-low humidity.

WARNING

All external connections to the hardware must stay within the recommended operating conditions and intended usage for all hardware/components connected in the system.

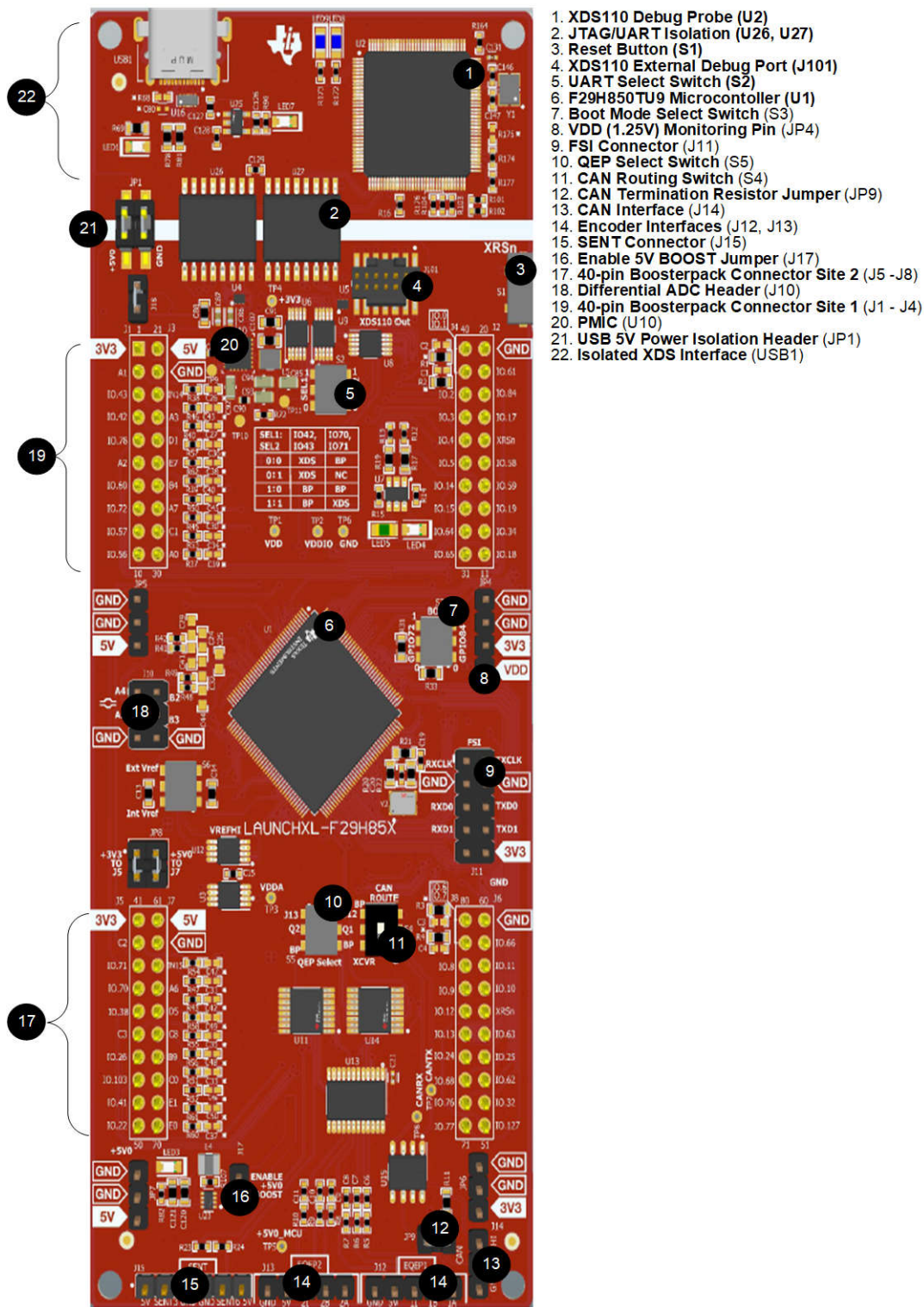Figure 1-1 highlights the key features of the F29H85x LaunchPad.

1. **XDS110 Debug Probe** (U2)
2. **JTAG/UART Isolation** (U26, U27)
3. **Reset Button** (S1)
4. **XDS110 External Debug Port** (J101)
5. **UART Select Switch** (S2)
6. **F29H850TU9 Microcontoller** (U1)
7. **Boot Mode Select Switch** (S3)
8. **VDD (1.25V) Monitoring Pin** (JP4)
9. **FSI Connector** (J11)
10. **QEP Select Switch** (S5)
11. **CAN Routing Switch** (S4)
12. **CAN Termination Resistor Jumper** (JP9)
13. **CAN Interface** (J14)
14. **Encoder Interfaces** (J12, J13)
15. **SENT Connector** (J15)
16. **Enable 5V BOOST Jumper** (J17)
17. **40-pin Boosterpack Connector Site 2** (J5 -J8)
18. **Differential ADC Header** (J10)
19. **40-pin Boosterpack Connector Site 1** (J1 - J4)
20. **PMIC** (U10)
21. **USB 5V Power Isolation Header** (JP1)
22. **Isolated XDS Interface** (USB1)

Figure 1-1. F29H85x LaunchPad Board Overview

ATTENTION
OBSERVE PRECAUTIONS
FOR HANDLING
ELECTROSTATIC
SENSITIVE DEVICES

1.3.1 External Power Supply or Accessory Requirements

Nominal Output Voltage: *5VDC*

Maximum Output Current: *3A*

Efficiency Level V

Note

TI recommends using an external power supply or power accessory **that** complies with applicable regional safety standards such as (by example) UL, CSA, VDE, CCC, PSE, and so on.

1.4 Device Information

The recommended steps for using the F29H85x LaunchPad are:

1. **Follow the instructions in** Section 2.4 **to begin running the LaunchPad demo program.** Within just a few minutes, you can control and monitor the F29H85x LaunchPad with the preprogrammed quick start application. Additionally, the Frequently Asked Questions section included in this document can be helpful if there are any issues that can quickly be addressed.

2. **Experiment with BoosterPacks.** This development kit conforms to the latest revision of the BoosterPack pinout standard. This development kit has two independent BoosterPack sites to enable a variety of expansion opportunities, such as two BoosterPacks being used simultaneously. For more information about the TI LaunchPad and BoosterPack standard, see the TI LaunchPad.

3. **Take the first step towards developing your own control applications**. The F29H85x LaunchPad is supported by the software development kit (SDK). After F29H85x SDK is installed, examples within the directory can be used with minor modifications to run on the LaunchPad. For more details about software development, see Section 2.

a. For applications, features and benefits information, see the *F29H85x, F29P58x, and F29P32x Real-Time Microcontrollers* data sheet.

b. Check out this technical article featuring C2000 Real-Time Microcontrollers about how developers can take advantage of the scalability and sustainability benefits these devices bring.

4. **Customize and integrate the hardware to align to your end application**. This development kit can be used as a reference for building your own custom boards and circuits based on C2000 F29H85x series microcontrollers. The LaunchPad also functions as a foundation for expansion with custom BoosterPacks and other circuits. This document can serve as a starting point for such projects.

5. **Get Trained**. Review and download hours of written and video training materials on C2000 Real-time Microcontrollers and related LaunchPads.

a. See *Getting Started With C2000™ Real-Time Control Microcontrollers (MCUs)*

b. See the TI video library page

c. See the C29x Academy page

2 Software

2.1 Software Description

The F29H85x software development kit (F29H85X-SDK) is a cohesive set of development tools for F29H85x real-time controllers. The SDK includes device-specific drivers, bit-field header files for device registers, and peripheral examples. Also included in the SDK are math, DSP, and control libraries.

2.2 Software Installation

2.2.1 Install SDK

Follow these steps to install the F29H85x SDK:

1. Download the **latest** version of F29H85X-SDK.
2. Start the installer and follow the prompts to complete the installation.
   a. It is recommended to install the SDK in the 'C:\ti' folder.

3. Review the release notes and other documentation included with the SDK.

Note

Other F29 SDKs can optionally be installed, but are not necessary to start evaluation with the LAUNCHXL-F29H85X.

2.2.2 Install Additional Software

To run FLASH builds, Python, and OpenSSL software is also needed.

2.2.2.1 Install Python

Follow these steps to install Python:
1. Download and Install the **latest** version of Python (Version 3.X and greater).
2. Review the README included in the SDK for further information.

2.2.2.2 Install OpenSSL

Follow these steps to install OpenSSL:
1. Download and Install the **latest** version of OpenSSL (Version 3.1.7 and greater).
2. After installation is completed, add the openSSL path into the system PATH environment variables:
   a. Navigate to 'System Properties -> Environment Variables -> System Variables -> Path'.
   b. Add new entry for 'C:\Program Files\OpenSSL-Win64\bin' and move it to be the first entry using 'Move up' button.

3. Review the README included in the SDK for further information.

2.3 Software Development

Code Composer Studio™ is an integrated development environment (IDE) that supports TI's Microcontroller and Embedded Processors portfolio. CCS is used to develop and debug embedded applications on the LaunchPad.

Follow these steps to install TI's CCS:

1. Download the latest version of CCS.
2. Start the installer and follow the prompts to complete the installation.
   a. It is recommended to install CCS in the 'C:\ti' folder.

3. Review the release notes and other documentation included with CCS.

2.4 F29H85x LaunchPad Demo Program

The LAUNCHXL-F29H85X includes a F29H850TU9 device preprogrammed with a demo program. When the LaunchPad is powered on, the demo program begins with an LED blink sequence on LED4 and LED5. After a few seconds, the device switches into an ADC sampling mode.

Every 1 second, the ADC samples pin ADCINA6 and the sampled value is represented as follows: If the sample is above mid-scale (2048), the red LED4 illuminates. If the sample is below mid-scale, the green LED5 illuminates.

By default, the ADC is configured in external voltage reference mode. This is 3.0V high-precision voltage reference device (REF5030). The mid-scale point occurs around 1.5V. For more information about the analog voltage reference, see Section 3.1.1.12.

In addition to the LED indicators, ADC sample results are also displayed on your PC through the USB/UART connection. To view the UART information on your PC, first determine the COM port associated with the LaunchPad (see Figure 2-1). To do this in Windows, open the *Device Manager*. Look for an entry under Ports (COM & LPT) titled "XDS110 Class Application/User UART (COM*X*)", where *X* is a number. Remember this number for when you open a serial terminal.
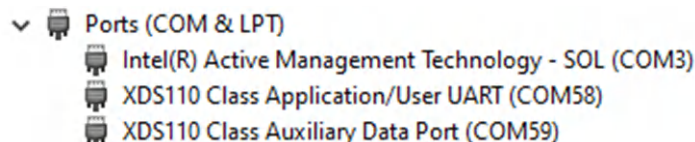
Figure 2-1. LaunchPad XDS110 COM Port

The demo application's UART data was tested using PuTTY, which is a free and open-source terminal emulator. To view the UART data in a serial terminal program, open the COM port found using the Windows *Device Manager* with the following settings:

```
115200 Baud, 8 data bits, no parity, 1 stop bit.
```

After properly opening the serial port in your serial terminal, reset the LaunchPad by pressing the S1 reset button and observe the serial terminal to see the TI logo in ASCII art (see Figure 2-2).
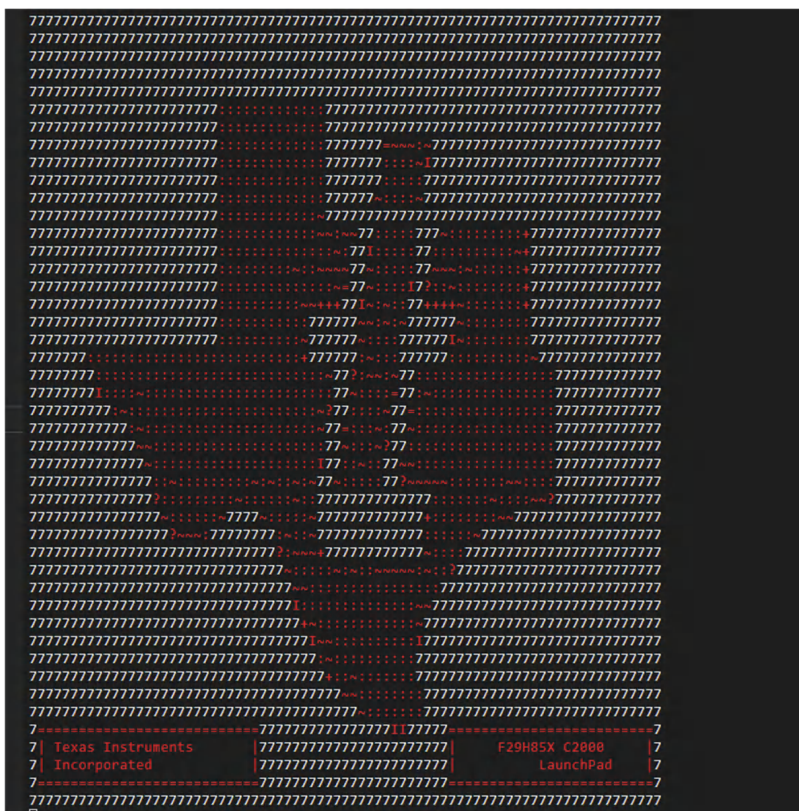

Figure 2-2. LaunchPad Demo Serial Terminal - TI Logo

After a few moments, the ADC value sampled on the ADCINA6 pin appears in the bottom right corner of the terminal and is updated each second (see Figure 2-3). Using a jumper wire, connect the ADCINA6 header to a 3.0V, GND, or other 0-3.0V signal to see the on-screen value change.
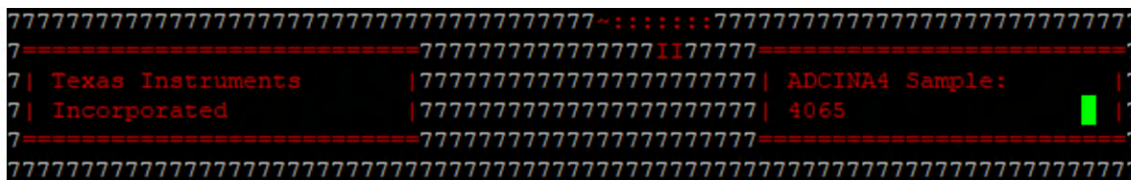

Figure 2-3. LaunchPad Demo Serial Terminal - ADC Sampling

2.5 Programming and Running Other Software on the F29H85x LaunchPad

The software packages described in Section 2.2 include example projects that can be loaded and run on the F29H85x LaunchPad. If you installed the F29H85X-SDK software to the default installation path of *C:\ti* , you can find DriverLib-based example applications in *<sdk_install_dir>\examples*. The on-board XDS110 is used with the On-Chip Flash Programmer tool to program applications to the F29H85x LaunchPad.

Follow these steps to program example applications onto the F29H85x LaunchPad development kit using the on-board XDS110 debug probe:

1. Install Code Composer Studio (CCS) IDE on a PC running Microsoft Windows.
2. Connect the USB-A cable plug in to an available USB port on the PC and plug the USB Type-C plug to the port (USB1) on the F29H85x LaunchPad.
3. Verify the following LEDs are illuminated:
    a. LED1, at the top left of the board, indicating 5V USB power.
    b. LED7 indicating 3.3V power to the XDS110 debug probe.
    c. LED3 indicating 5.0V power to the F29H850TU MCU.
4. Install Windows XDS110 and Virtual COM Port drivers if prompted. Installation instructions can be found at XDS110 Product Page.
5. Run CCS IDE on the PC.
6. Import a F29H85x project from F29H85X-SDK, or another installed software package, into the CCS IDE workspace.
7. Add the *_LAUNCHXL_F29H85X* predefined symbol to the imported DriverLib example project for the software to use relevant F29H85x LaunchPad signals.
    a. Open the projects Properties → Expand the *Build* tab → Expand the *C2000 Compiler* tab → Select *Predefined Symbol* → Add *_LAUNCHXL_F29H85X* pre-define NAME.
8. Right click the project name and select *Rebuild Project* in CCS IDE.
9. Launch the LAUNCHXL-F29H85X Target Configuration file and connect to the F29H85x device. Make sure that the Target Configuration file is configured to use the 4-pin JTAG advanced configuration. For additional details, see FAQ section.
10. Click 'Load Program' and select the program's binary to load. The binary is loaded onto the device and can now be run and debugged.

3 Hardware

3.1 Hardware Description

The F29H85x LaunchPad includes a F29H850TU9SRFS MCU, which is designed for advanced real-time control applications. A large number of these peripherals are made available to users using the on-board accessories and the BoosterPack connectors. This section explains how those peripherals operate and interface to the MCU.

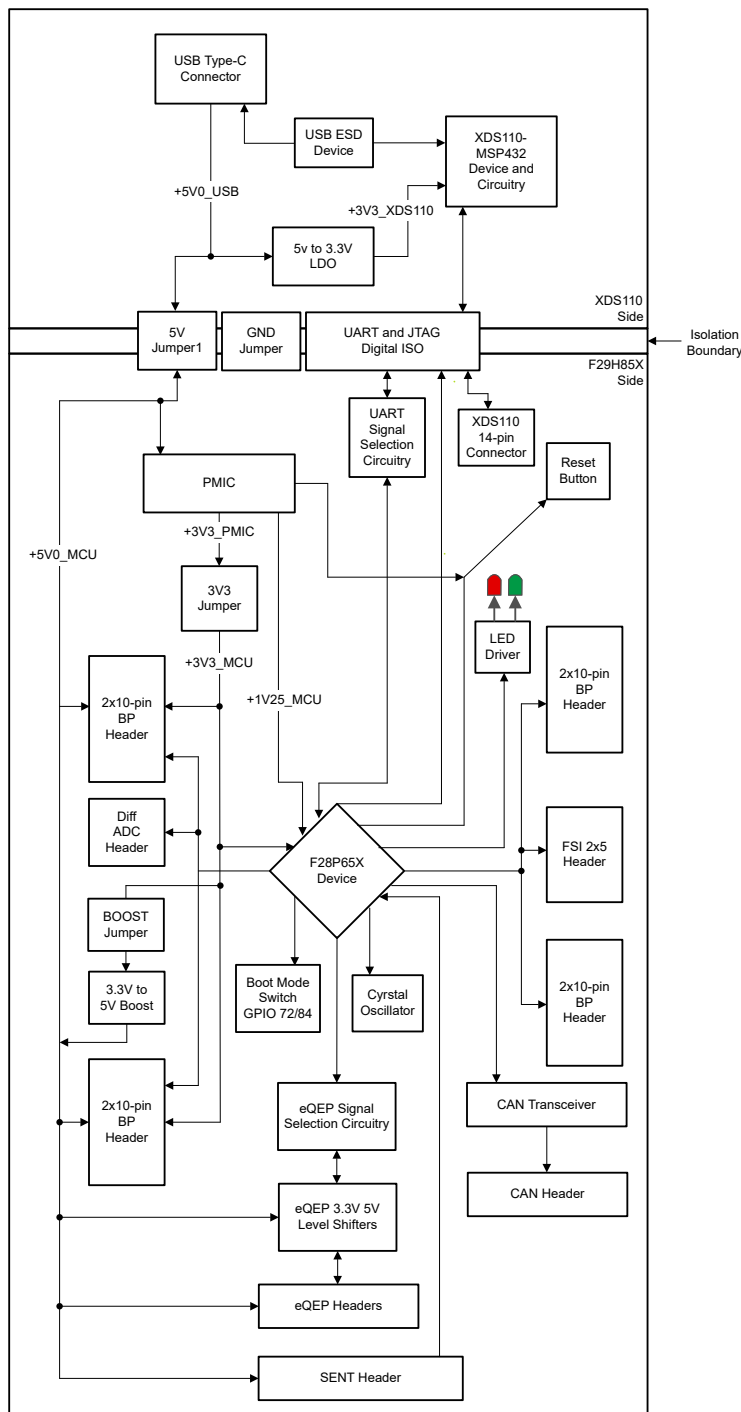Figure 3-1 shows a high-level block diagram of the F29H85x LaunchPad.

Figure 3-1. F29H85x LaunchPad Development Kit Block Diagram

3.1.1 Functional Description and Connections

3.1.1.1 Microcontroller

The F29H850TU9SRFS has 3 C29x 64-bit floating-point microcontroller with 4MB Flash memory, 452KB RAM, and operates at 200MHz. The microcontroller includes advanced control peripherals, differentiated analog, and various communications peripherals. The device has been optimized for high-performance real-time control applications. For more details, see the *F29H85x, F29P58x, and F29P32x Real-Time Microcontrollers* data sheet.

Most of the microcontroller's signals are routed to 0.1 inch (2.54mm) pitch headers laid out to comply with the TI BoosterPack standards, with a few exceptions. The F29H85x MCU's internal multiplexer allows different peripheral functions to be assigned to each of the General-Purpose Input/Output (GPIO) pins. The multiplexing options can be found in the device-specific data sheet. When adding external circuitry, consider the additional load on the development board's power rails.

The F29H85x LaunchPad is factory-programmed with a quick start demo program. The quick start program resides in the on-chip Flash memory and executes each time power is applied, unless the application has been replaced with a user program. For details on the LaunchPad's demo program, see Section 2.4.

3.1.1.2 PMIC

The Launchpad receives power from the 5V input on the USB connector. This 5V input serves as the input to the TPS65036x-Q1 PMIC (orderable part number: TPS65036604RAYQ1). The PMIC on the board generate all the voltage rails required for the Microcontroller. All power supply sequencing and voltage monitoring is handled by the PMIC. Refer to Section 1.3.1 for specific requirements on the 5V input to the Launchpad.

The F29H85x device and the PMIC on the Launchpad can be connected via I2C bus by populating R29 and R32. The I2C bus is used by the device to configure the PMIC and also, when enabled, to service the PMIC's watchdog. The I2C bus GPIOs are also connected to the I2C standard location on J5 by default.

PMIC watchdog is disabled by default. User can enable watchdog by removing R18. Similarly, ERRORSTS is not connected to PMIC by default. By populating R26 and R27, ERRORSTS is connected to GPIO which can be configured as PMIC ESM input.

3.1.1.3 Power Domains

The F29H85x LaunchPad has several power domains that can be connected or isolated from each other with removable shunts. The different 3.3V and 5V power domains are further described in Figure 3-2.
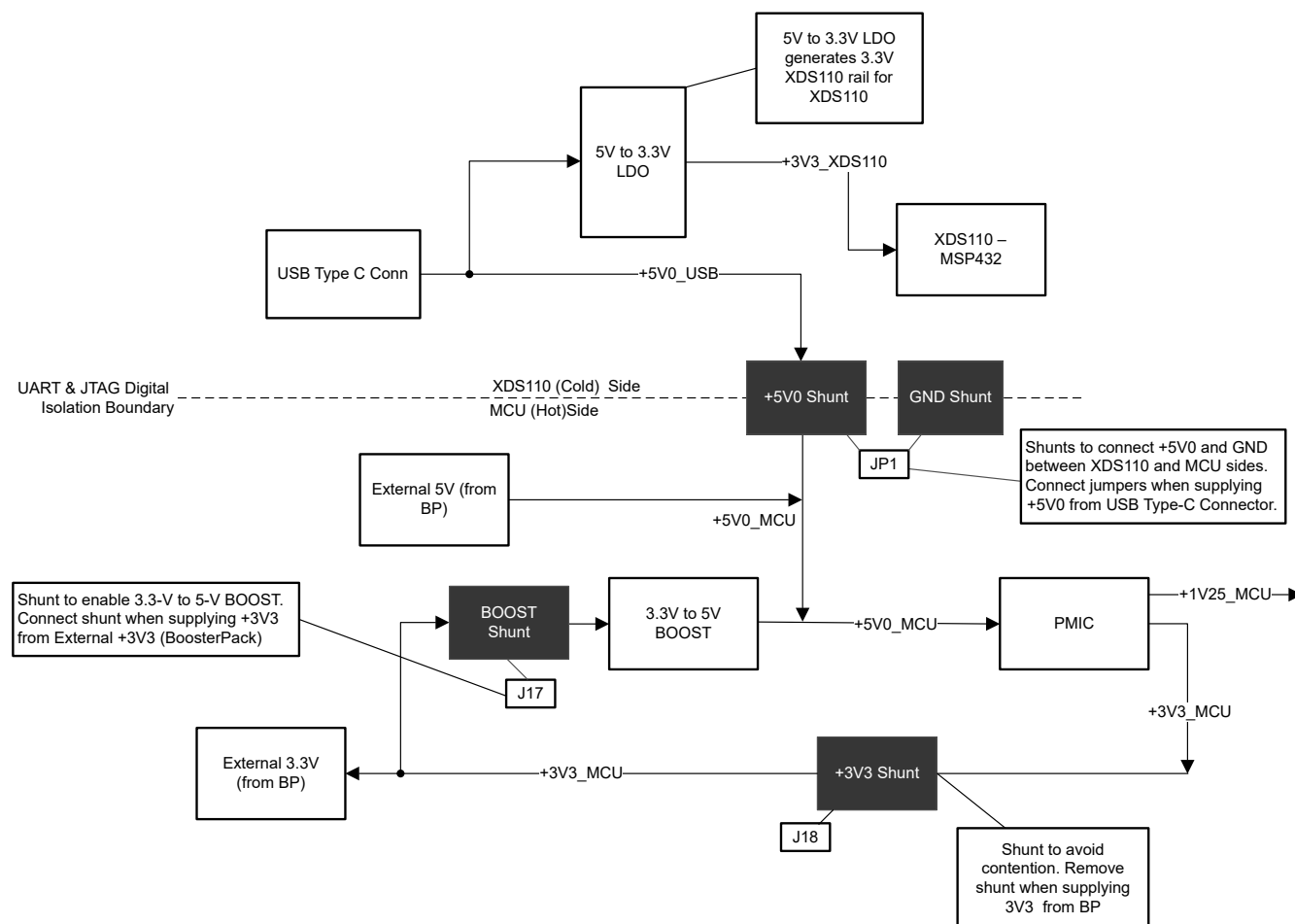

Figure 3-2. LaunchPad Power Distribution Diagram

Table 3-1 describes the usage of the different removable shunts on the LaunchPad board.

Table 3-1. Power Domain Shunts

Shunt Identifier	Usage Description
JP1, +5V0	Connects the +5V power from the USB-C connector (+5V0_USB) to the +5V power on the MCU side of the board (+5V0_MCU). Bridges the power and ground isolations between the two board sides.
JP1, GND	Connects the board Ground on the isolated USB-C connector side of the board (USB_GND) to the rest of the board ground (GND). Bridges the power and ground isolations between the two board sides.
J18	Disconnects the PMIC and on-board 3.3V when powering from Boosterpack
J17	Enables the on-board 3.3V to 5V BOOST regulator to convert the +3.3V power rail to a +5V power rail.

The F29H85x LaunchPad features a flexible power domain scheme that allows users to supply power to the board in a variety of different configurations. Table 3-2 shows the different power configurations and the required shunts that need to be populated to supply power throughout the board.

Table 3-2. Power Configurations

Power Source	Connected Shunts	Description of Power Sources
USB-C Connector	JP1, J16	**+5V0_USB:** supplied from the USB-C connector **+5V0_MCU:** +5V0_USB passes through JP1 and is the same supply as +5V0_MCU **+3V3_MCU:** generated by the MCU-side 5V to 3.3V LDO regulator
External +3.3-V (connected to BoosterPack header)	J17	**+5V0_USB:** If debugging the device, +5V0_USB is supplied through the USB-C connector and is isolated from the MCU side +5V0 rail. Else if not debugging, +5V0_USB is not required. **+5V0_MCU:** generated by the 3.3V to 5V BOOST regulator **+3V3_MCU:** supplied by external +3.3V source
External +5.0-V (connected to BoosterPack header)	J16	**+5V0_USB:** If debugging the device, +5V0_USB is supplied through the USB-C connector and is isolated from the MCU side +5V0 rail. Else if not debugging, +5V0_USB is not required. **+5V0_MCU:** supplied by external +5.0V source **+3V3_MCU:** generated by the MCU-side PMIC

3.1.1.4 LEDs

Power indicator LEDs (red) are included on the F29H85x LaunchPad board. Table 3-3 shows descriptions of each LED.

Table 3-3. Power LED Indication Descriptions

LED Number	Indication Description
LED1	+5V power from the USB Type-C™ connector
LED7	+3.3V power on the XDS110 side of the PCB
LED3	+5V power on the F29H85x side of the PCB

Two user LEDs are provided on the board: LED4 (red) and LED5 (green), see Table 3-4. These user LEDs are connected to GPIO19 and GPIO62 of the F29H850TU, respectively. The signals are connected to the SN74LVC2G07DBVR LED driver IC and are connected in an active-low configuration; that is, drive the GPIO low to turn on the LED and high to turn off the LED. These LEDs are dedicated for use by the software application.

Table 3-4. User-Configurable LED Descriptions

LED Number	Description
LED4	Active-low user-configurable LED controlled by GPIO19
LED5	Active-low user-configurable LED controlled by GPIO62

Two blue LEDs, LED8 and LED9, are connected to the XDS110 debug probe, see Table 3-5. These indicate debugger activity and are not controllable by any application software.

Table 3-5. XDS110 LED Descriptions

LED Number	Indication Description
LED8	Indicates that the XDS110 device is actively connected to a debugger target session (for example, CCS IDE debug session)
LED9	Indicates that the XDS110 device is powered and functional

3.1.1.5 Encoder Connectors

The F29H85x LaunchPad includes two headers, J12 and J13, which are used for connecting linear or rotary incremental encoders. These headers take 5V input signals that are stepped down to 3.3V and wired to the F29H850TU9 MCU. These signals are connected to the eQEP modules on the device when switch S5 is set appropriately, see Table 3-8. Each header has the EQEPA, EQEPB, and EQEPI signals available for each eQEP module (1 and 2) as well as pins for GND and 5V.

3.1.1.6 FSI

The F29H85x MCU features the Fast Serial Interface (FSI) communications peripheral. The FSI enables high-speed communications and is intended to increase the amount of information transmitted while reducing the cost to communicate over an isolation barrier. The FSI signals TXCLK, TXD0, TXD1, RXCLK, RXD0, and RXD1 are available on J11. This header is set up in such a way that adding jumpers on the pins connects the TX to RX channels for external loopback and evaluation. Additionally, there are two GND signals on the connector that can be used for a wrapped-pair connection to an external board with FSI. The GPIOs connected to this header are only routed to the J11 FSI connector on this board, the GPIOs are not routed to the BoosterPack headers.

The LAUNCHXL-F29H85X does not include any on-board isolation devices for the FSI signals. If interested in evaluating the FSI peripheral with isolation devices, or differential drivers/receivers, see the TMDSFSIADAPEVM plug on board.

3.1.1.7 CAN

The F29H85x LaunchPad includes a connector (J14) for a CAN network. GPIO64 and GPIO65 are routed from the F29H850TU to J14 through the on-board CAN Transceiver. Switch S4 is used to route GPIO64 and GPIO65 to either the CAN transceiver and connector or the BoosterPack headers. For more details, see Section 3.1.3.4.

3.1.1.8 SENT

The F29H85x LaunchPad features two SENT channels to enable SENT communication capabilities.

3.1.1.9 CLB

The configurable logic block (CLB) is a collection of blocks that can be interconnected using software to implement custom digital logic functions or enhance existing on-chip peripherals. The CLB is able to enhance existing peripherals through a set of interconnections, which provide a high level of connectivity to existing control peripherals such as enhanced pulse width modulators (ePWM), enhanced capture modules (eCAP), and enhanced quadrature encoder pulse modules (eQEP). The crossbars also allow the CLB to be connected to other internal peripheral signals of the device or external GPIO pins. In this way, the CLB can be configured to perform small logical functions to augment device peripheral inputs and outputs. Through the CLB, functions that can otherwise be accomplished using external logic devices, such as FPGAs or CPLDs, can now be implemented inside the C2000 MCU.

For more information on the CLB, see the C2000™ Configurable Logic Block (CLB) training series video.

3.1.1.10 Boot Modes

The F29H850TU boot ROM contains bootloading software that executes every time the device is powered on or reset. Two pins, GPIO72 and GPIO84, are wired to the Boot Select switch (S3). By default, both pins are set HIGH (1) so the device boots from Flash. For more information on the F29H85x boot modes, see the *F29H85x, F29P58x, and F29P32x Real-Time Microcontrollers* data sheet.

Table 3-6. Boot Select Switch Table - S3

Boot Mode	GPIO72 (LEFT)	GPIO84 (RIGHT)
Boot from Parallel GPIO	0	0
Boot from UART/Wait boot	0	1
Boot from CAN	1	0
Boot from Flash (default)	1	1

3.1.1.11 BoosterPack Sites

The F29H85x LaunchPad features two fully independent BoosterPack XL connectors. Both BoosterPack sites 1 and 2 are compliant with the BoosterPack standard. To expand the functions available to the user on this LaunchPad, some signals are also routed to alternate locations on the board. These alternate routes can be selected by manipulating the on-board switches or by adding / removing 0Ω resistors. This is described in Section 3.1.3.

The GPIO pin numbers as well as the BoosterPack compliant features can be viewed in the LAUNCHXL-F29H85X LaunchPad Pinout Map. Each GPIO has multiple functions available through the F29H85x device's GPIO mux. Some specific functions have been listed in the Pinout Map; the full GPIO mux table can be found in the *F29H85x, F29P58x, and F29P32x Real-Time Microcontrollers* data sheet.

All of the analog signals (denoted ADCIN) of the F29H85x MCU are routed to the J1/J3 and J5/J7 BoosterPack headers on the left side of the board. Close to the respective BoosterPack header each ADC input signal has component pads for a series resistor and parallel capacitor to create an RC filter. By default a 0-ohm resistor is populated and the capacitor is left unpopulated. Users can populate these components with specific values to filter out noise arriving at the device's ADC input.

3.1.1.12 Analog Voltage Reference

The analog subsystem of the F29H85x allows for flexible voltage reference sources. The ADC modules are referenced to the VREFHIx and VREFLOx pin voltages. VREFHIx can either be driven externally or can be generated by an internal bandgap voltage reference. The on-board REF5030 device provides a high-precision 3.0V reference with integrated buffer and is actively connected to the C2000 device. Note that there is no signal conditioning circuitry in place for an external voltage reference. For best performance, some additional circuitry may be required.

3.1.1.13 Differential ADC Header

J10 on the LaunchPad features two pairs of differentially-routed ADC signals to allow for evaluation of the F29H85x's 12-bit differential and 16-bit differential ADC sampling modes. ADCINA4 and ADCINA5 form one pair of differential ADC signals, and ADCINB2 and ADCINB3 form the second pair. Like the single-ended ADC input signals on the board, the differential ADC signals have component pads to allow for the addition of series resistor and parallel capacitor to create an RC filter. In addition, these ADC signals also feature a pad to connect a capacitor across the differential signals. Note: The differentially-routed ADC signals on the F29H85x LaunchPad are not buffered.

3.1.1.14 Other Headers and Jumpers

The LaunchPad has multiple jumpers to select different power sources for the board. This LaunchPad also provides a way to isolate the connected USB from the device, allowing for safe operation and debugging in higher voltage applications.

3.1.1.14.1 XDS Isolation Block

JP1 is provided to enable isolation between the device and the connected XDS in higher-voltage applications. The area of isolation is defined by the white outline on the upper-side of the LaunchPad. JP1 has two removable shunts to separate the GND and 5V power of the USB/XDS region and the F29H85x MCU region of the LaunchPad. By default, both shunts are populated and the power is supplied by the connected USB, meaning that the USB and XDS110 are NOT isolated from the F29H85x MCU region. If power isolation is desired, remove the supplied shunts from JP1. In this configuration, one of the two external power options below are required:

- An external 5-V supply to power the PMIC (TPS65036), which provides 3.3V to the F29H85x MCU region of the board.
- An external 3.3-V supply to power the F29H85x MCU region of the board.

  5V is generated using the on-board 5V BOOST (TPS610333).

In an isolated power application with JP1 shunts removed, make sure the proper shunts on the F29H85x MCU region of the board are populated. For more information, see Section 3.1.1.3.

3.1.1.14.2 BoosterPack Site 2 Power Isolation

JP8 is included to isolate 3.3V and 5V from the BoosterPack site 2 headers. This can be required if two BoosterPacks are simultaneously connected to the LaunchPad and both provide power to the LaunchPad. If this is the case, power can be isolated by removing the shunts on JP8 and there can be no contention between the two BoosterPacks.

3.1.1.14.3 Alternate Power

Additional jumpers are provided outside of the BoosterPack connector for additional external power connections for 3.3V or 5V. These can be used to supply an external board or for powering the LaunchPad with an external supply. When using these connection points, make sure that no other power supplies are connected.

- **JP4 and JP6** are provided as extra connection points for a 3.3V supply to be connected to the LaunchPad.
- **JP5 and JP7** are provided as extra connection points for a 5V supply to be connected to the LaunchPad.

> **CAUTION**
>
> The F29H85x LaunchPad features a VDD pin located on header JP4 that can only be used to monitor the voltage of the 1.25-V VDD rail of the F29H850TU9SRFS device. This pin can not be used to supply power to an external device nor connected to an external supply.

3.1.2 Debug Interface

3.1.2.1 XDS110 Debug Probe

The F29H85x LaunchPad includes an on-board XDS110 Debug Probe. The XDS110 allows for the programming and debugging of the F29H850TU device using Code Composer Studio (CCS) IDE or any other supported tool chains. In the default configuration, the XDS110 is wired to support 4-pin JTAG mode. Due to the isolation scheme used on the F29H85x LaunchPad, 2-wire Serial Wire Debug (SWD) mode is not supported.

3.1.2.2 XDS110 Output

The connector J101 is provided to debug an external target with the on-board XDS110 debug probe. This connector allows the LaunchPad to be used as a stand-alone XDS110 debug probe. For mating connector and cable, see Samtec FFSD or equivalent. UART TX and RX signals from the XDS110 device are also included on this connector.

If the LaunchPad is being used in this manner, the following 0 ohm resistors located on the bottom of the board must be desoldered. This isolates the JTAG signals from going to the F29H850TU9 MCU.

Component	Corresponding Signal
R74	3V3
R76	3V3
R97	TMS
R98	TCK
R99	TDI
R100	TDO
R165	RXD
R171	TXD

3.1.2.3 Virtual COM Port

When plugged into a USB host, the XDS110 enumerates as both a debugger and a virtual COM port. U26 and U27 are reinforced digital isolators which isolate the UART signals between the F29H850TU and the debug probe. The UART signals are then passed from the debug probe on to the USB host. By default, the F29H85x UARTA maps to the virtual COM port of the XDS110 using GPIO42 and GPIO43. Alternately, GPIO70 and GPIO71 can be used for UARTB. This is accomplished by manipulating the on-board switch S2. For the appropriate switch settings, see Section 3.1.3.2.

3.1.3 Alternate Routing

3.1.3.1 Overview

The F29H850TU MCU is a very versatile device in a mid-size package. To balance compatibility with BoosterPack standards as well as showcasing the versatility of the F29H850TU, some complexity was added to the design. Most features aligning with the BoosterPack standard are available by default. The additional functions are configured using switches or static resistors which can be added or removed. This section covers the alternate functions and how to enable them. Note that by enabling certain alternate features, standard BoosterPack functionality can be lost. The switches and resistors are configured such that it is not possible to connect multiple functions to the same header.

3.1.3.2 UART Routing

This LaunchPad allows for one of two sets of pins to be used for the UART routed to the virtual COM port of the XDS110. By default, GPIO43 (UARTA_RX) and GPIO42 (UARTA_TX) are routed to the virtual COM port and not available on the BoosterPack connector. Alternately, GPIO71 (UARTB_RX) and GPIO70 (UARTB_TX) can be routed to the virtual COM port. When UART functionality is not needed at the virtual COM port, the GPIOs can be routed to the BoosterPack connectors for BoosterPack standard functions.

The routing destination of these signal pairs are selected using the on-board switch S2, as described in Table 3-7.

Table 3-7. UART Select Table - S2

SEL1 (Left)	SEL2 (Right)	GPIO42/43	GPIO70/71
0	0	XDS110 COM Port	BP Headers
0	1	XDS110 COM Port	No Connect
1	0	BP Headers	BP Headers
1	1	BP Headers	XDS110 COM Port

3.1.3.3 EQEP Routing

The LaunchPad has the ability to connect to two independent linear or rotary encoders through the F29H85x on-chip eQEP interfaces: Header J12 is connected to eQEP1 and header J13 is connected to eQEP2. By default, this connection is not active and the GPIOs are routed to the BoosterPack connectors. The 5 eQEP input signals from the J12 and J13 connectors are stepped down through a TI SN74LVC8T245 Level Translator (U13) to 3.3V. The signals are then routed through TI SN74LV4053A Triple 2-Channel Analog Multiplexer/Demultiplexer ICs (U11/U14). Switch S5 controls the select inputs of the ICs to configure the eQEP signal destinations to be either the J12/J13 connectors or BoosterPack headers, as described in Table 3-8.

Table 3-8. QEP Select Table - S5

QEP1 SEL (LEFT)	QEP2 SEL (RIGHT)	QEP1 Signals (GPIO84/17/34)	QEP2 Signals (GPIO11/10/32)
0 (down)	0 (down)	J12	J13
0 (down)	1 (up)	J12	BP Headers
1 (up)	0 (down)	BP Headers	J13
1 (up)	1 (up)	BP Headers	BP Headers

3.1.3.4 CAN Routing

The LaunchPad can be connected to a CAN bus through J14. GPIO64 and GPIO65 are routed to the on-board TI TCAN332DR 3.3V CAN Transceiver, U15. By setting S4 to DOWN (XCVR), GPIO64 and GPIO65 are routed to the transceiver. If S4 is set to UP (BP), the GPIOs are routed to the BoosterPack connectors (default case).

3.1.3.5 FSI Routing

One set of GPIOs with available FSI functionality are directly connected to the FSI header, J11. The traces from the device to the FSI header are made short to make sure of higher signal integrity, as FSI signals can switch at frequencies up to 200MHz on F29H85x devices.

3.1.3.6 PWM DAC

In addition to the buffered DAC outputs available on BP pin 30 and BP pin 70, the F29H85x LaunchPad provides up to four PWM DAC signals at the BoosterPack headers using GPIO8 (BP pin 40) and GPIO9 (BP pin 39) on site 1 and GPIO2 (BP pin 80) and GPIO3 (BP pin 79) on site 2. The intended purpose of the PWM DAC signals are to utilize PWMs of the F29H85x device as digital-to-analog converters (DAC). This method involves low-pass filtering the PWM signal to remove its high-frequency components and ideally leave only the DC component.

By default the RC filter is not populated. Instead a 0 ohm resistor is populated and the capacitor is left unpopulated.

3.1.4 Hardware Revisions

This section contains an abbreviated revision history of the LAUNCHXL-F29H85X as well as known issues with each revision.

3.1.4.1 Revision A

The first production revision of the LAUNCHXL-F29H85x was released in June 2025. This revision can be identified by the "MCU119A" silkscreen labeling on the back side of the EVM between the BoosterPack Connector site 1 towards the top of the board.

Issues and concerns that have been identified on the EVM are listed below:

Known issues:

• No issues to report at this time of initial release.

Special notes and considerations to be aware of:

• Nothing to report at this time of initial release.

4 Hardware Design Files

The entire LAUNCHXL-F29H85X design files are available for download at this link: LAUNCHXL-F29H85X design files.

4.1 Schematics

The LaunchPad's schematic can be found at this link: LAUNCHXL-F29H85X Schematic.

4.2 PCB Layouts

The layout source files for the LAUNCHXL-F29H85X are included in the LAUNCHXL-F29H85X design files download.
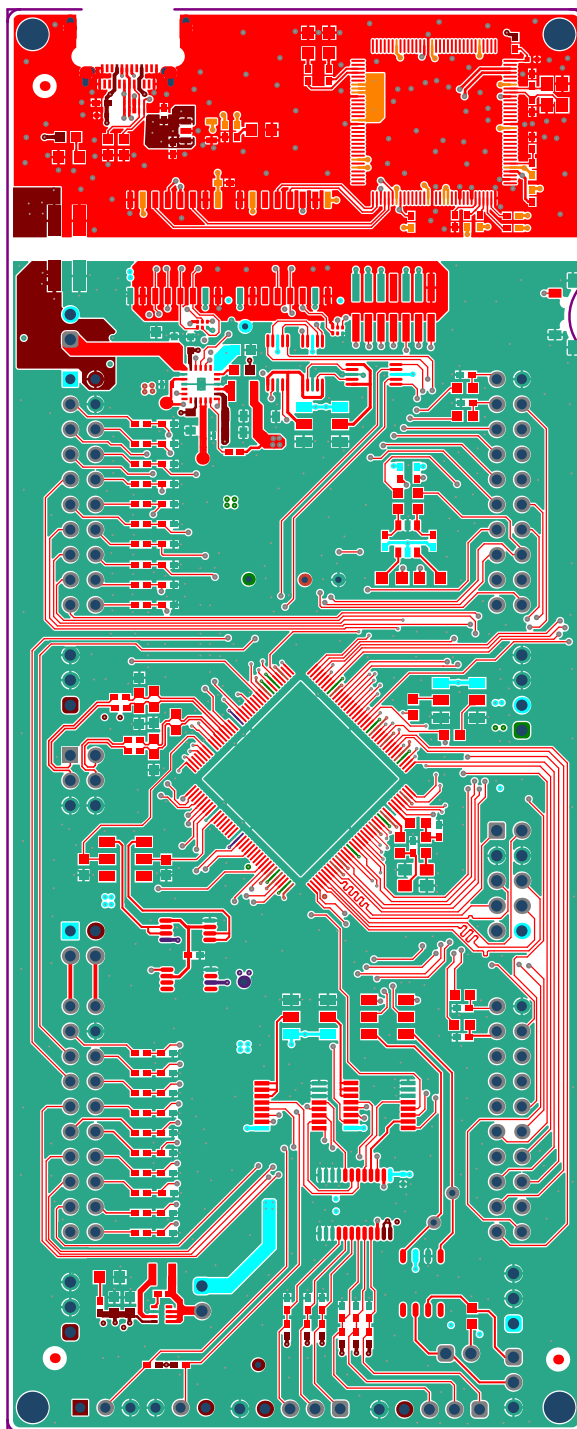
Figure 4-1. Top Signal - Layer 1
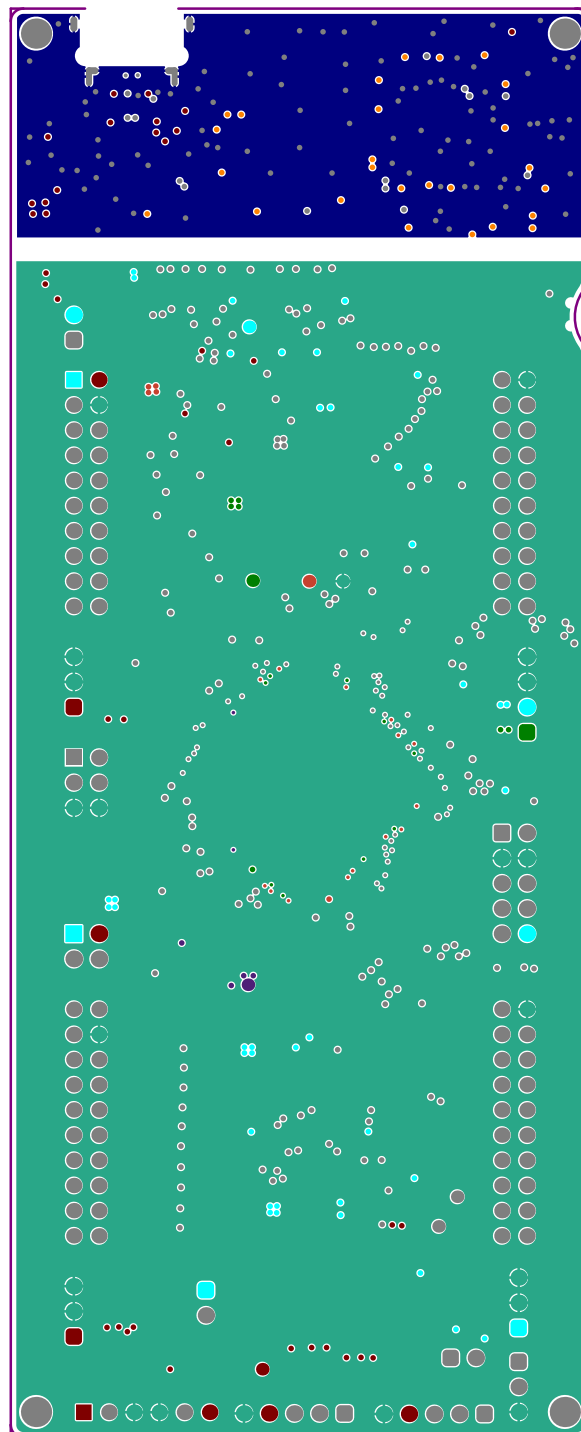
Figure 4-2. GND - Layer 2
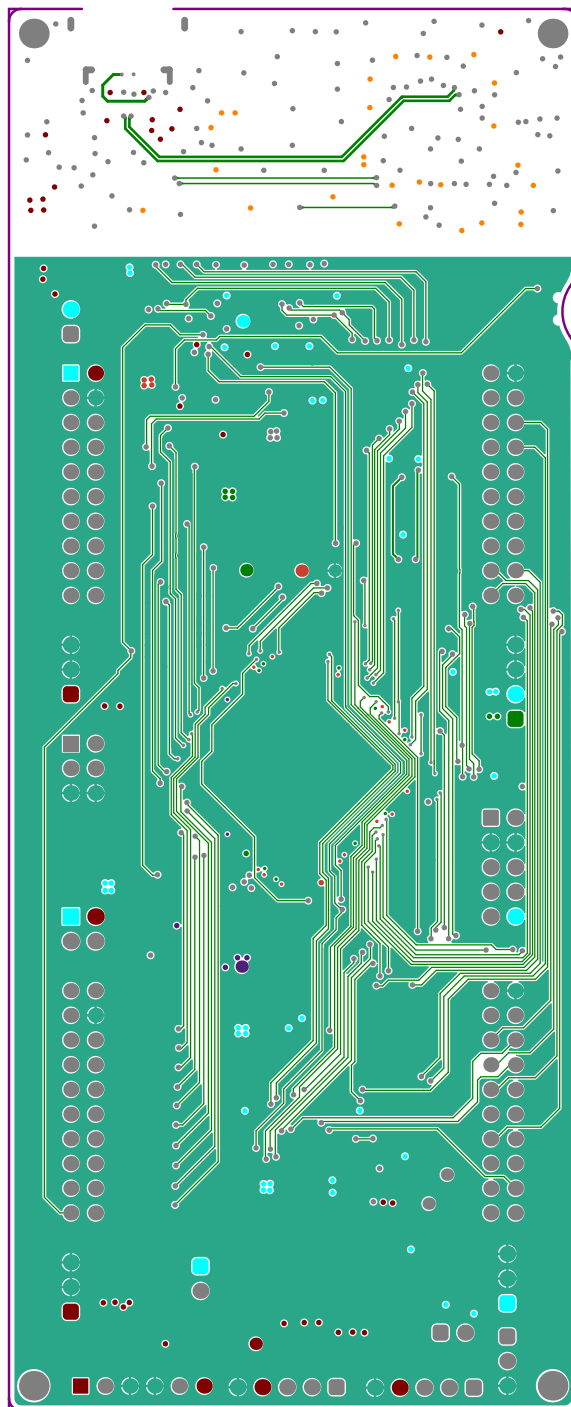
Figure 4-3. Middle Signal- Layer 3
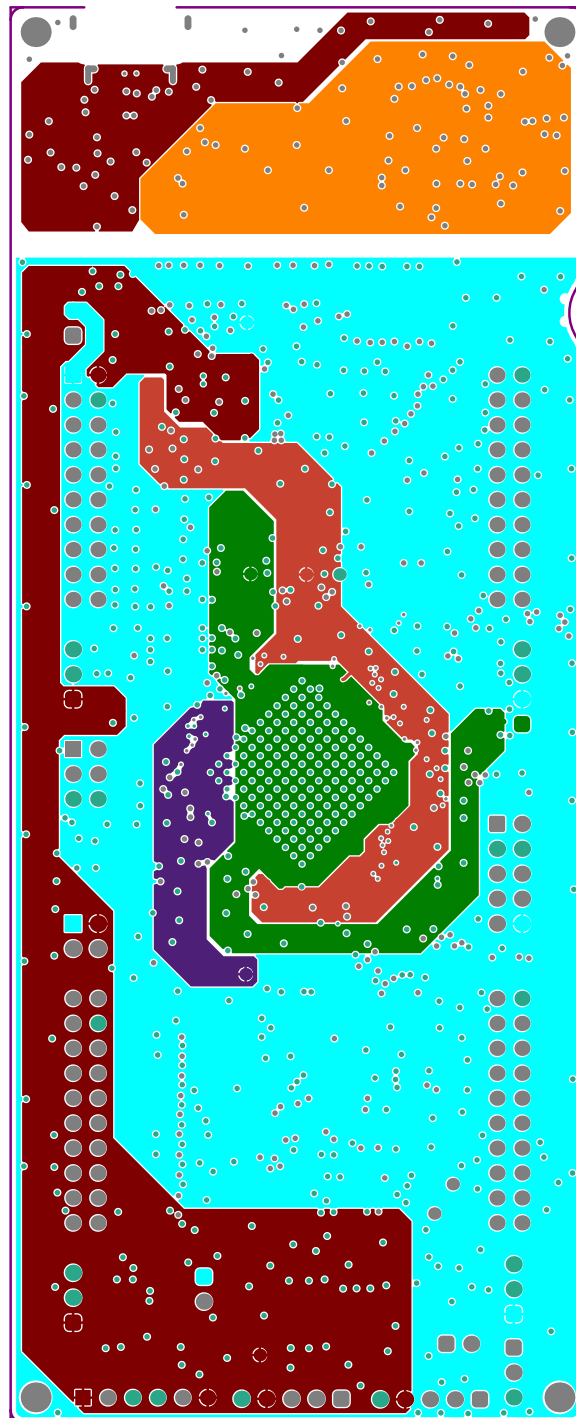
Figure 4-4. PWR - Layer 4
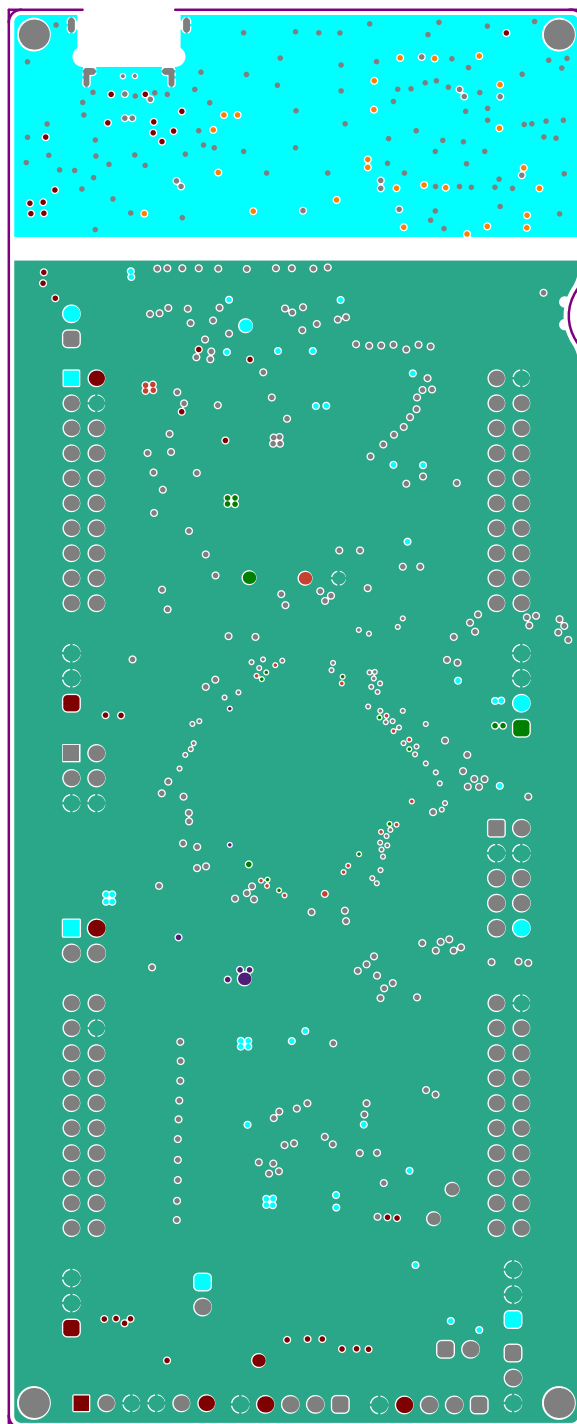
Figure 4-5. GND2- Layer 5
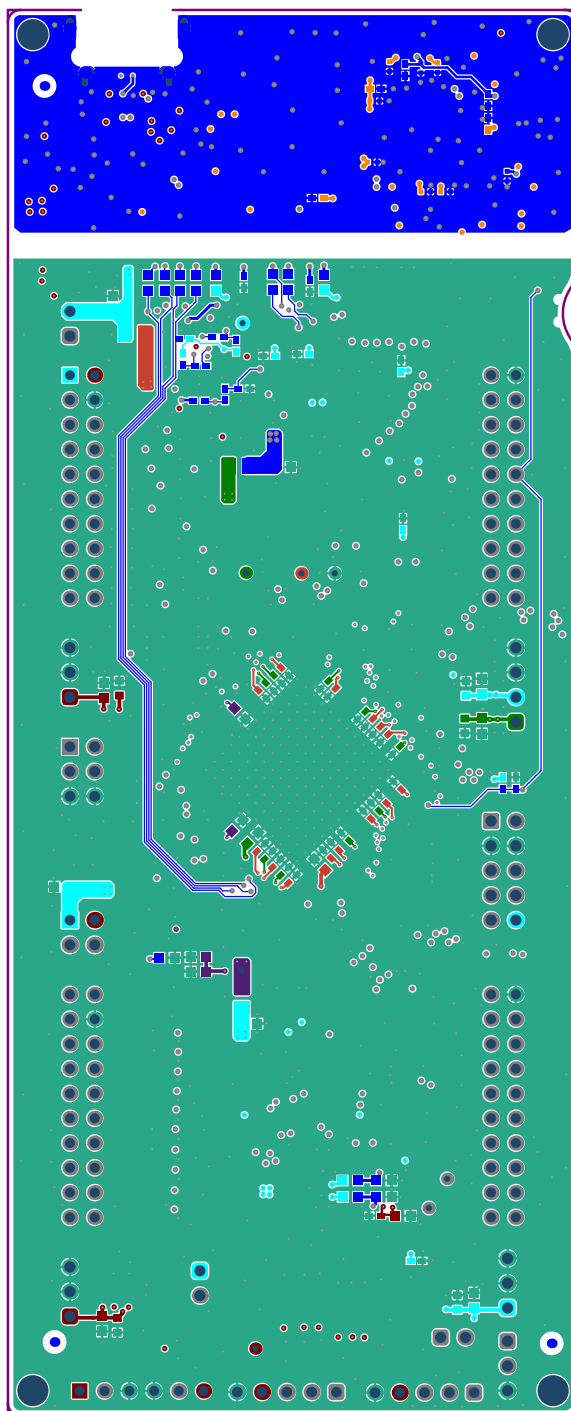
Figure 4-6. Bottom Signal - Layer 6

4.2.1 LAUNCHXL-F29H85X Board Dimensions

Figure 4-7 is a dimensional drawing of the F29H85x LaunchPad that shows the location of selected features of the board as well as the component locations.
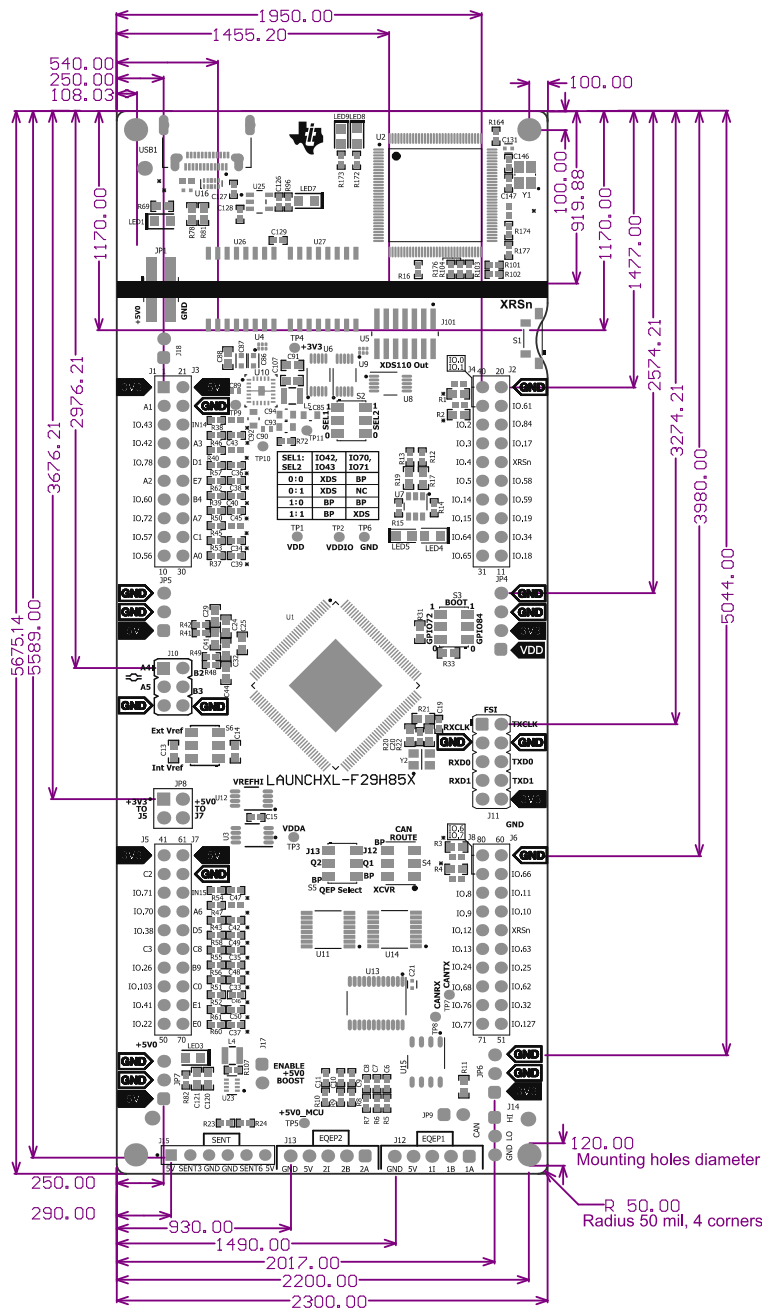
Figure 4-7. F29H85x LaunchPad Dimensions and Component Locations

4.3 Bill of Materials (BOM)

The BOM for the LAUNCHXL-F29H85X is included in the LAUNCHXL-F29H85X design files download.

5 Frequently Asked Questions

1. Why isn't my C2000 device getting any power even though I have the USB-C cable connected to USB1?
   a. Make sure that the USB 5V power and GND jumpers are connected (JP1)
      i. To avoid contention on the 5V rail, make sure that the "ENABLE +5V0 BOOST" jumper (J17) is disconnected when supplying 5V through the USB.

2. Can other programming and debug tools (such as an XDS200 debug probe) be used with the F29H85x LaunchPad?
   a. The F29H85x LaunchPad utilizes an on-board XDS110 debug probe in a 4-pin JTAG configuration.

3. What versions of Code Composer Studio™ IDE can be used to develop software for the F29H85x LaunchPad?
   a. The on-board XDS110 debug probe is compatible with Code Composer Studio development environment version 20.0.0 and later.

4. Why can't I connect to the LaunchPad in Code Composer Studio IDE?
   a. Are shunts present on J101 for TCK and TMS?
   b. Is the XDS110 and the F29H850TU9 MCU powered? Are LED1, LED7 and LED3 illuminated? For further details on powering the LaunchPad, see Section 3.1.1.3.
      i. If JP1 shunts are disconnected, the power provided through the USB is isolated from the rest of the board. Make sure that 3.3V or 5V is supplied to any of the available connectors on the target side of the isolation.
   c. Is the USB-C cable connected to the PCB and is the USB region receiving power? Is LED1 illuminated?
      i. The XDS region must be powered with the 5V from the USB cable. LED1 illuminates when 5V USB power is connected. The XDS-side 5V to 3.3V LDO regulator requires 5V on the USB side to generate the +3V3 rail for the XDS110 device.
   d. Make sure that the target configuration is set up to use JTAG in 4-pin advanced mode. Open the Target Configuration file (.ccxml) in Code Composer Studio IDE. Click on the Advanced tab and select JTAG (1149.1) SWD and cJTAG are disabled. Alternately, a working Target configuration file is included in the launchxl_ex1_f29h85x_demo project "F29H850TU9.ccxml". You can use this without modifications.
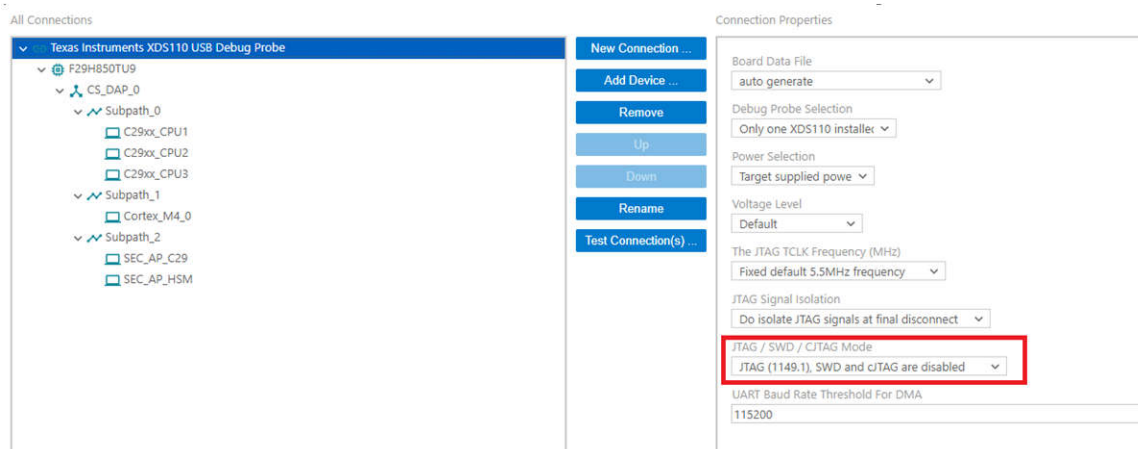

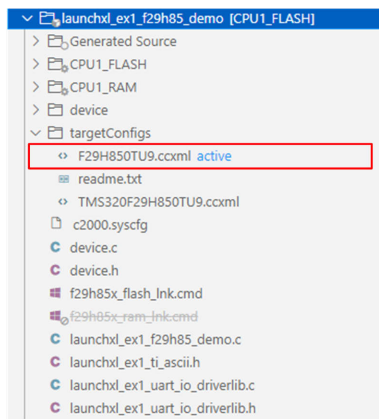**Figure 5-1. Target Configuration Advanced Options**

Figure 5-2. Target Configuration Included in the Demo Project

5. Why is the serial connection not working?
    a. Are you using the correct COM port?
        i. Right click on My Computer and select Properties. Navigate to the Hardware tab in the dialog box and open the device manager. Scroll to Ports (COM and LPT) and expand this entry. Is XDS110 Class Application/User UART listed? If so, read the COM number to the right of the entry; this is the COM number to use.
    b. Are you using the correct baud rate? Most, if not all, of the examples are configured for a baud rate of 115200 when the CPU is running at 200MHz. If you have changed the PLL settings or developed your own code, you can recalculate the baud rate for your specific application. For information on how to do this, see the *F29H85x and F29P58x Real-Time Microcontrollers Technical Reference Manual*.
    c. Does the UART channel wired to the debug probe match the UART channel configured in software?
        i. The F29H85x LaunchPad provides an option for one of two possible UART channels to be routed to the debug probe. Make sure that S2 is configured to the appropriate UART channel for the application software. For details, see Section 3.1.3.2.

6. Why does my F29H85x LaunchPad keep resetting at random?
    a. Check if the boot mode switch (S3) is set to parallel GPIO boot. Using this boot mode when there is no host device driving the corresponding GPIO pins can cause erratic behavior, including MCU reset. S3 must be set according to the desired boot mode.

6 Additional Information

6.1 Trademarks
C2000™, LaunchPad™, BoosterPack™, Code Composer Studio™ are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.

7 References

7.1 Reference Documents

In addition to this document, the following references are available for download at www.ti.com.

- Texas Instruments: *F29H85x, F29P58x, and F29P32x Real-Time Microcontrollers* data sheet
- Texas Instruments: *F29H85x and F29P58x Real-Time Microcontrollers Technical Reference Manual*
- Texas Instruments: *Getting Started With C2000™ Real-Time Control Microcontrollers (MCUs)*
- Texas Instruments: *The Essential Guide for Developing with C2000™ Real-Time Microcontrollers*
- Texas Instruments: *LAUNCHXL-F29H85X LaunchPad Pinout Map*
- *MCU-SDK-F29H85x Development Kit*
- *Texas Instruments Code Composer Studio*
- *Texas Instruments LaunchPad Development Environment*

7.2 Other TI Components Used in This Design

This LaunchPad uses various other TI components for its functions. A consolidated list of these components with links to their TI product pages is shown below.

- *ISO7731 Robust EMC, triple-channel, 2/1, reinforced digital isolator*
- *ISO7741 Robust EMC, quad-channel, 3/1, reinforced digital isolator*
- *TS5A23157 5-V, 2:1 (SPDT), 2-channel analog switch*
- *MSP432E401Y SimpleLink™ Ethernet Microcontroller*
- *REF50xxA-Q1 Low-Noise, Very Low Drift, Precision Voltage Reference*
- *SN74LV4053A Triple 2-Channel Analog Multiplexer/Demultiplexer IC*
- *SN74LVC2G07 Dual Buffer/Driver With Open-Drain Output*
- *SN74LVC8T245 8-Bit Dual-Supply Bus Transceiver with Configurable Voltage-Level Shifting and Three-State Outputs*
- *TCAN332 3.3 V CAN Transceiver*
- *TLV703 300-mA, high-PSRR, low-IQ, low-dropout voltage regulator*
- *TPS61033X 5.5-V 5.5-A 2.4-MHz Fully-Integrated Synchronous Boost Converter, with Output Discharge Function*
- *TPD4E004 4-Channel ESD Protection Array for High-Speed Data Interfaces*
- *TPD4E05U06 4-Channel ESD Protection Array for USB, HDMI & High Speed Interfaces*
- *TPS650362-Q1 Automotive 35V PMIC with three buck converters and one LDO regulator for camera modules up to ASIL-B*

8 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

IMPORTANT NOTICE AND DISCLAIMER