

MSP432P4111 SimpleLink™ Microcontroller LaunchPad™ Development Kit (MSP-EXP432P4111)

The SimpleLink™ MSP-EXP432P4111 LaunchPad™ development kit is an easy-to-use evaluation module (EVM) for the SimpleLink MSP432P4111 microcontroller (MCU). It contains everything needed to start developing on the SimpleLink MSP432P4111 low-power 32-bit Arm® Cortex®-M4F MCU with precision ADC, including onboard emulation for programming, debugging, and energy measurements. The MSP432P4111 MCU supports low-power applications requiring increased CPU speed, memory, analog, and 32-bit performance.



Figure 1. MSP-EXP432P4111 LaunchPad Development Kit

Contents

1	Getting Started	4
1.1	Introduction	4
1.2	Key Features	4
1.3	What's Included	4
1.4	First Steps: Out-of-Box Experience	4
1.5	Next Steps: Looking Into the Provided Code	5
2	Hardware.....	5
2.1	Block Diagram.....	6
2.2	SimpleLink MSP432P4111 MCU	6
2.3	XDS110-ET Onboard Emulator	8
2.4	Special Features	13
2.5	Power	17
2.6	Measure MSP432 Current Draw	18
2.7	Clocking	18
2.8	BoosterPack Plug-in Module Pinout	19
2.9	Design Files	21
2.10	Hardware Change Log.....	21
3	Software and Additional Resources	21
3.1	TI Resource Explorer	21
3.2	SimpleLink MSP432P4 SDK.....	21
3.3	Integrated Development Environments.....	21
3.4	LaunchPad Websites	22
3.5	MSP432P4111 MCU.....	23
3.6	Community Resources	23
4	FAQ	23
5	Schematics	27

List of Figures

1	MSP-EXP432P4111 LaunchPad Development Kit.....	1
2	MSP-EXP432P4111 Overview.....	5
3	MSP-EXP432P4111 Block Diagram	6
4	MSP432P4111IPZ Pinout	7
5	XDS110-ET Emulator	8
6	XDS110-ET Isolation Block.....	9
7	Application Backchannel UART in Device Manager.....	10
8	EnergyTrace Technology Preferences	12
9	EnergyTrace Windows	13
10	LCD Segment Layout.....	14
11	LMT70 Output Transfer Function	16
12	R8 Location	17
13	MSP-EXP432P4111 Power Block Diagram	17
14	LaunchPad-to-BoosterPack Connector Pinout	20
15	SWD Mode Settings	24
16	Target Configurations.....	24
17	Launch Selected Configuration.....	24
18	Show All Cores	25
19	Connect Target	25
20	MSP432_Factory_Reset Script.....	26
21	Schematics (1 of 6)	28
22	Schematics (2 of 6)	29
23	Schematics (3 of 6)	30

24	Schematics (4 of 6)	31
25	Schematics (5 of 6)	32
26	Schematics (6 of 6)	33

List of Tables

1	Isolation Block Connections	8
2	LCD FH-1138P Segment Mapping	14
3	LCD-to-MSP Connections.....	15
4	Default Clock Operation	19
5	Hardware Change Log	21
6	How MSP Device Documentation is Organized.....	23

Trademarks

SimpleLink, LaunchPad, BoosterPack, Code Composer Studio, EnergyTrace, E2E are trademarks of Texas Instruments.

Arm, Cortex, Keil, μ Vision are registered trademarks of Arm Limited.

IAR Embedded Workbench is a registered trademark of IAR Systems.

All other trademarks are the property of their respective owners.

1 Getting Started

1.1 Introduction

The SimpleLink [MSP-EXP432P4111 LaunchPad development kit](#) is an easy-to-use evaluation module (EVM) for the [MSP432P4111 MCU](#). It contains everything needed to start developing on the SimpleLink MSP432 low-power and performance Arm 32-bit Cortex-M4F MCU, including onboard emulation for programming, debugging, and energy measurements. The [MSP432P4111 MCU](#) supports low-power applications requiring increased CPU speed, memory, analog, and 32-bit performance. The board features onboard buttons and LEDs for quick integration of a simple user interface and an liquid crystal display (LCD) that showcases the integrated driver that can drive up to 320 segments.

Rapid prototyping is simplified by access to the 40-pin headers and a wide variety of BoosterPack™ plug-in modules that enable technologies such as wireless connectivity, graphical displays, environmental sensing, and many more. Free software development tools are also available such as TI's Eclipse-based [Code Composer Studio™ IDE](#), [IAR Embedded Workbench® IDE](#), and [Keil® µVision® IDE](#).

Code Composer Studio IDE supports [EnergyTrace™ technology](#) when paired with the MSP432P4111 LaunchPad development kit. For more information about the LaunchPad kit, the supported BoosterPack plug-in modules, and the available resources, visit the [TI LaunchPad development kit portal](#). To get started quickly and find available resources in the SimpleLink MSP432 software development kit (SDK), visit the [TI Cloud Development Environment](#).

1.2 Key Features

- Low-power Arm Cortex-M4F MSP432P4111
- 40-pin LaunchPad standard that leverages the BoosterPack ecosystem
- XDS110-ET, an open-source onboard debugger featuring EnergyTrace+ technology and application UART
- Two buttons and two LEDs for user interaction
- Segmented LCD
- LMT70 precision analog temperature sensor
- Backchannel UART through USB to PC

1.3 What's Included

1.3.1 Kit Contents

- One MSP-EXP432P4111 LaunchPad development kit
- One Micro USB cable
- One quick start guide

1.3.2 Software Examples (See [Section 3](#))

- SimpleLink MSP432P4 SDK

Includes both low-level examples and more in-depth demos that use various middleware and libraries

1.4 First Steps: Out-of-Box Experience

An easy way to get familiar with the EVM is by using its preprogrammed out-of-box code. It demonstrates some key features of the LaunchPad from a user level, showing how to use the pushbutton switches together with onboard LEDs and basic serial communication with a computer.

For a more detailed explanation of the out-of-box demo, see [Section 3](#).

1.5 Next Steps: Looking Into the Provided Code

It is now time to start exploring more features of the EVM!

<http://www.ti.com/tool/msp-exp432p4111>

To get started, you need an integrated development environment (IDE) to explore and start editing the code examples. See [Section 3](#) for more information on IDEs and where to download them.

The out-of-box source code and more code examples can be found in TI Resource Explorer or downloaded from <http://www.ti.com/tool/msp-exp432p4111>. Find what code examples are available and more details about each example in [Section 3](#). All code is licensed under BSD, and TI encourages reuse and modifications to fit specific needs.

2 Hardware

[Figure 2](#) shows an overview of the EVM hardware.

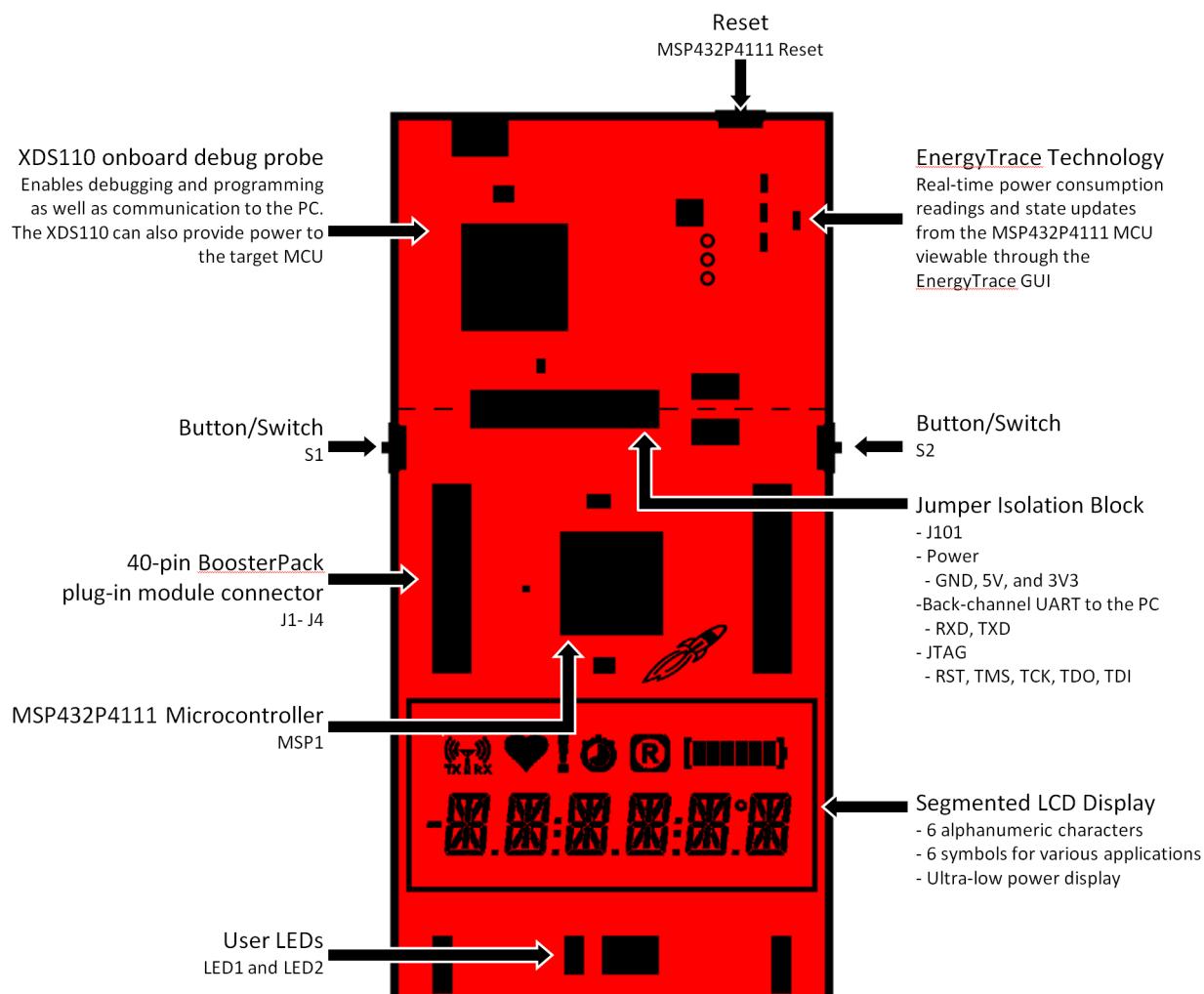


Figure 2. MSP-EXP432P4111 Overview

2.1 Block Diagram

Figure 3 shows the block diagram.

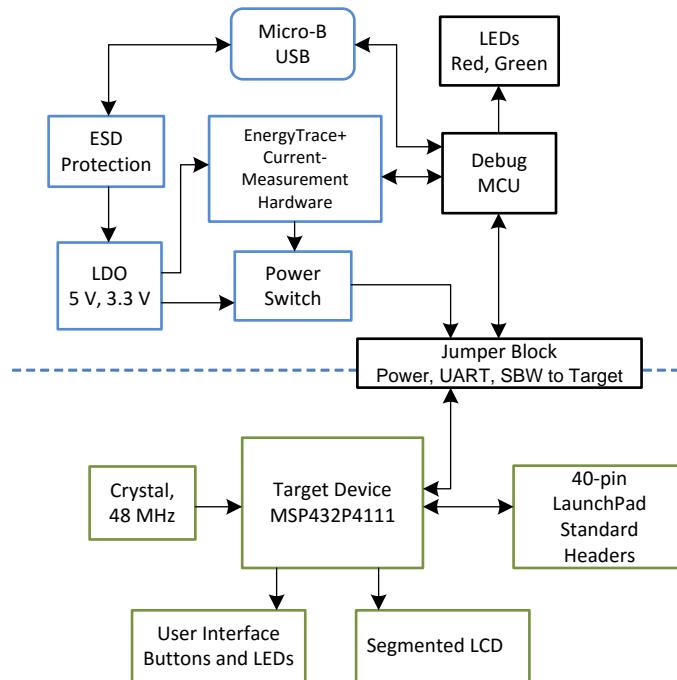
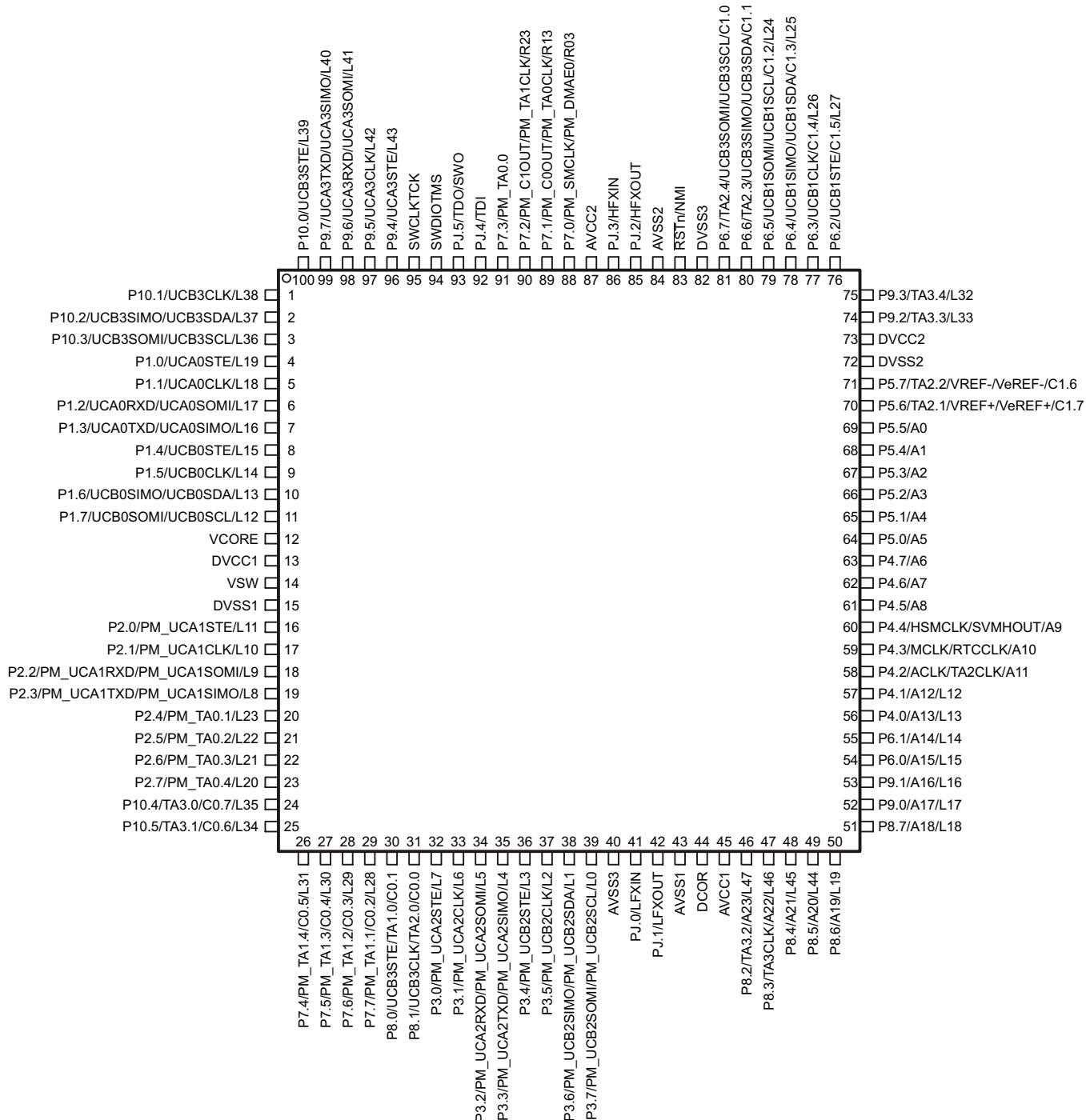


Figure 3. MSP-EXP432P4111 Block Diagram

2.2 SimpleLink MSP432P4111 MCU

The MSP432P4111 is the next device in the MSP432 family featuring low-power performance with an Arm Cortex-M4F core. Device features include:

- Low-power Arm Cortex-M4F MSP432P4111 MCU
- Up to 48-MHz system clock
- 2MB of flash memory, 256KB of SRAM, and 32KB of ROM with SimpleLink MSP432 SDK Libraries
- Four 16-bit timers with capture, compare, or PWM; two 32-bit timers; and RTC
- Up to eight serial communication channels (I²C, SPI, UART, and IrDA)
- Analog: Precision ADC, two comparators, integrated LCD driver
- Digital: AES256, CRC, μ DMA
- 100-pin LQFP (PZ) package (see Figure 4)


Figure 4. MSP432P4111IPZ Pinout

2.3 XDS110-ET Onboard Emulator

To keep development easy and cost effective, the TI LaunchPad development kits integrate an onboard emulator, which eliminates the need for expensive programmers. The MSP-EXP432P4111 has the XDS110-ET emulator, which is a simple and low-cost debugger that supports nearly all TI Arm device derivatives.

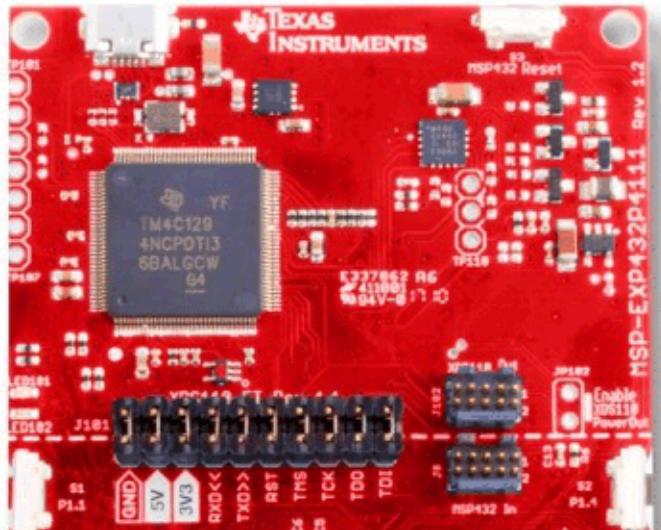


Figure 5. XDS110-ET Emulator

The XDS110-ET hardware can be found in the schematics in [Section 5](#) and in the MSP-EXP432P4111 Hardware Design Files.

2.3.1 XDS110-ET Isolation Block

The J101 isolation block is composed of 10 jumpers (see [Table 1](#)). The J101 isolation block allows the user to connect or disconnect signals that cross from the XDS110-ET domain into the MSP432P4111 target domain. This crossing is shown by the dotted line across the LaunchPad development kit through J101. No other signals cross this domain, so the XDS110-ET can be decoupled from the MSP432P4111 target side. This includes XDS110-ET power and GND signals, UART, and JTAG signals.

[Table 1](#) lists the signals that are controlled at the isolation block (also see [Figure 6](#)).

Table 1. Isolation Block Connections

Signal	Description
GND	GND power connection between XDS110 and MSP432 target GND planes. The GND jumper is populated to connect the separate GND planes. This connection is required for proper operation with 3V3, 5V, and JTAG
5V	5-V power rail, VBUS from USB
3V3	3.3-V power rail, derived from VBUS by an LDO in the XDS110-ET domain
RXD <<	Backchannel UART: The target MCU receives data through this signal. The arrows indicate the direction of the signal.
TXD >>	Backchannel UART: The target MCU sends data through this signal. The arrows indicate the direction of the signal.
RST	MCU RST signal (active low)
TCK_SWCLK	Serial wire clock input (SWCLK); JTAG clock input (TCK)
TMS_SWDIO	Serial wire data input/output (SWDIO); JTAG test mode select (TMS)
TDO_SWO	Serial wire trace output (SWO); JTAG trace output (TWO) (also PJ.5)
TDI	JTAG test data input (also PJ.4)

Reasons to open these connections:

- To remove any and all influence from the XDS110-ET emulator for high accuracy target power measurements
- To control 3-V and 5-V power flow between the XDS110-ET and target domains
- To expose the target MCU pins for other use than onboard debugging and application UART communication
- To expose the programming and UART interface of the XDS110-ET so that it can be used for devices other than the onboard MCU.

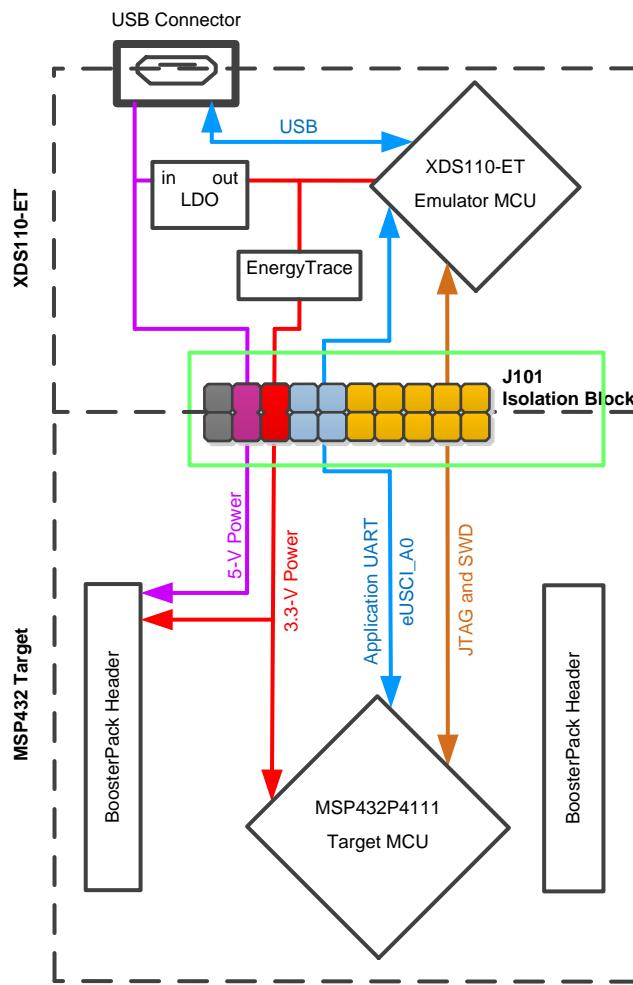


Figure 6. XDS110-ET Isolation Block

2.3.2 Application (or “Backchannel”) UART

The XDS110-ET provides a “backchannel” UART-over-USB connection with the host, which can be very useful during debugging and for easy communication with a PC.

The backchannel UART allows communication with the USB host that is not part of the main functionality of the target application. This is very useful during development, and also provides a communication channel to the PC host side. This can be used to create GUIs and other programs on the PC that communicate with the LaunchPad development kit.

The pathway of the backchannel UART is shown in [Figure 7](#). The backchannel UART eUSCI_A0 is independent of the UART on the 40-pin BoosterPack plug-in module connector eUSCI_A2.

On the host side, a virtual COM port for the application backchannel UART is generated when the LaunchPad enumerates on the host. You can use any PC application that interfaces with COM ports, including terminal applications like Hyperterminal or Docklight, to open this port and communicate with the target application. You need to identify the COM port for the backchannel. On Windows PCs, Device Manager can assist.

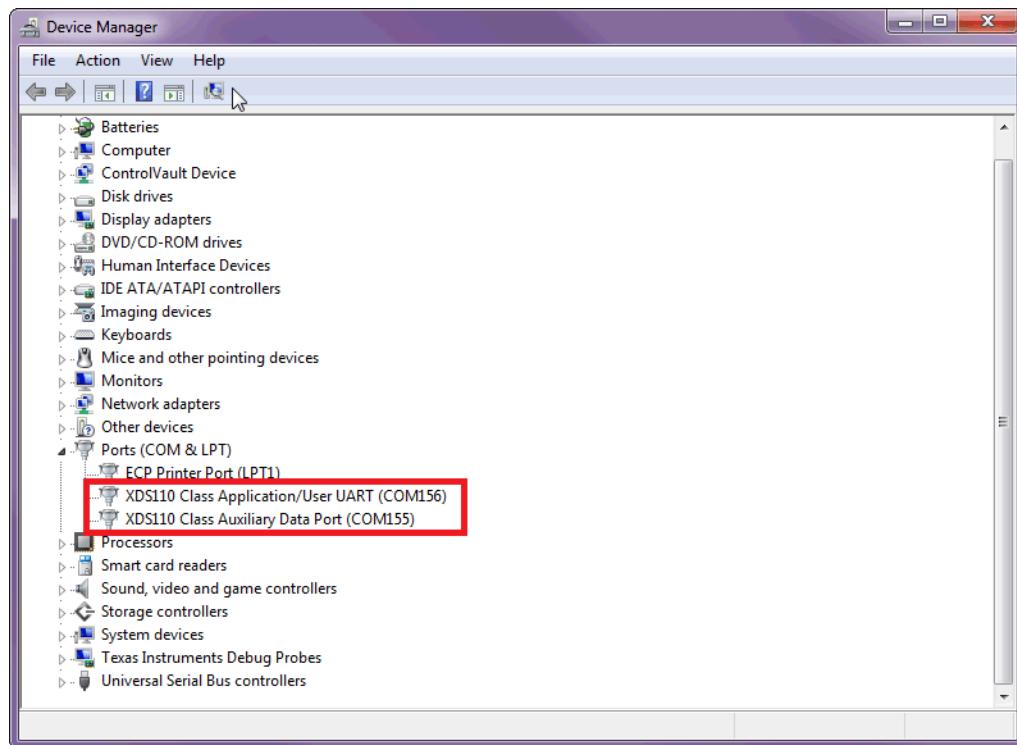


Figure 7. Application Backchannel UART in Device Manager

The backchannel UART is the XDS110 Class Application/User UART port. In this case, [Figure 7](#) shows COM156, but this port can vary from one host PC to the next. After you identify the correct COM port, configure it in your host application according to its documentation. You can then open the port and begin communication to it from the host.

The XDS110-ET has a configurable baud rate; therefore, it is important that the PC application configures the baud rate to be the same as what is configured on the eUSCI_A0 backchannel UART.

2.3.3 Using an External Debugger Instead of the Onboard XDS110-ET

Many users have a specific debugger that they prefer to use, and may want to bypass the XDS110-ET to program the MSP432 target MCU. This is enabled by jumpers on isolation block J101, and the connector J8. Using an external debug probe is simple, and full JTAG access is provided through J8.

1. Remove jumpers on the JTAG signals on the J101 isolation block, including RST, TMS, TCK, TDO, and TDI.
2. Plug any Arm debugger into J8.
 - a. J8 follows the Arm Cortex Debug Connector standard outlined in [Cortex-M Debug Connectors](#)
3. Plug USB power into the LaunchPad, or power it externally
 - a. Ensure that the jumpers across 3V3 and GND are connected if using USB power
 - b. External debuggers do not provide power, the Vcc pin is a power sense pin
 - c. See [Section 2.5](#) for more details on powering the LaunchPad development kit.

2.3.4 Using the XDS110-ET Emulator With a Different Target

The XDS110-ET emulator on the LaunchPad can interface to most Arm Cortex-M devices, not just the onboard MSP432P4111 device. This functionality is enabled by the J102 [10-pin Cortex-M JTAG connector](#) and a 10-pin cable, such as the [FFSD-05-D-06.00-01-N](#) (sold separately from the LaunchPad development kit).

Header J102 follows the Cortex-M Arm standard; however, pin 1 is not a voltage sense pin. The XDS110-ET outputs only 3.3-V JTAG signals. If another voltage level is needed, the user must provide level shifters to translate the JTAG signal voltages. Additionally, 3.3 V of output power can be sourced from the XDS110-ET when jumper JP102 is connected. This allows the XDS110-ET to power the external target at 3.3 V through pin 1. By default, JP102 is not populated as it does not explicitly follow the standard.

1. Remove jumpers on the JTAG signals on the isolation block, including RST, TMS, TCK, TDO, and TDI.
2. Plug the 10-pin cable into J102, and connect to an external target.
 - a. J102 allows the Arm Cortex Debug Connector standard outlined in [Cortex-M Debug Connectors](#).
3. Plug USB power into the LaunchPad development kit, or power it externally.
 - a. JTAG levels are 3.3 V only.
 - b. 3.3-V power can be sourced through J102 by shorting the JP102 jumper.

2.3.5 EnergyTrace Technology

EnergyTrace technology is an energy-based code analysis tool that measures and displays the energy profile of the application and helps to optimize it for ultra-low power consumption.

MSP432 devices with built-in EnergyTrace+[CPU State] (or in short EnergyTrace+) technology allow real-time monitoring of internal device states while user program code executes.

EnergyTrace+ technology is supported on the LaunchPad development kit MSP432P4111 device and XDS110-ET debug probe. EnergyTrace technology is available as part of Code Composer Studio IDE. During application debug, additional windows are available for EnergyTrace technology.

To enable EnergyTrace technology:

1. Click Window > Preferences > Code Composer Studio > Advanced Tools > EnergyTrace™ Technology.
2. Check the Enable Auto-Launch on target connect box.

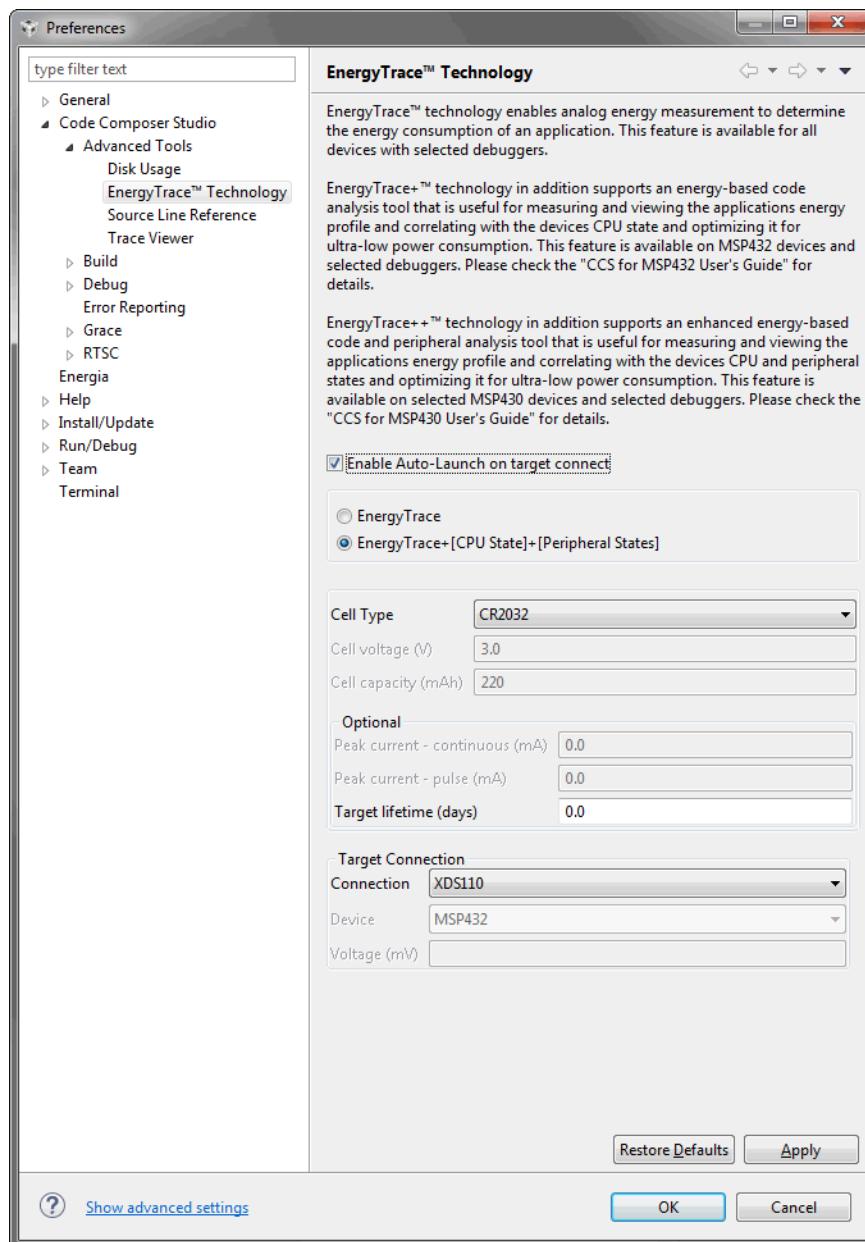


Figure 8. EnergyTrace Technology Preferences

Starting a debug session opens the EnergyTrace technology windows. These windows show energy, power, profile, and states to give a full view of the energy profile of the application.

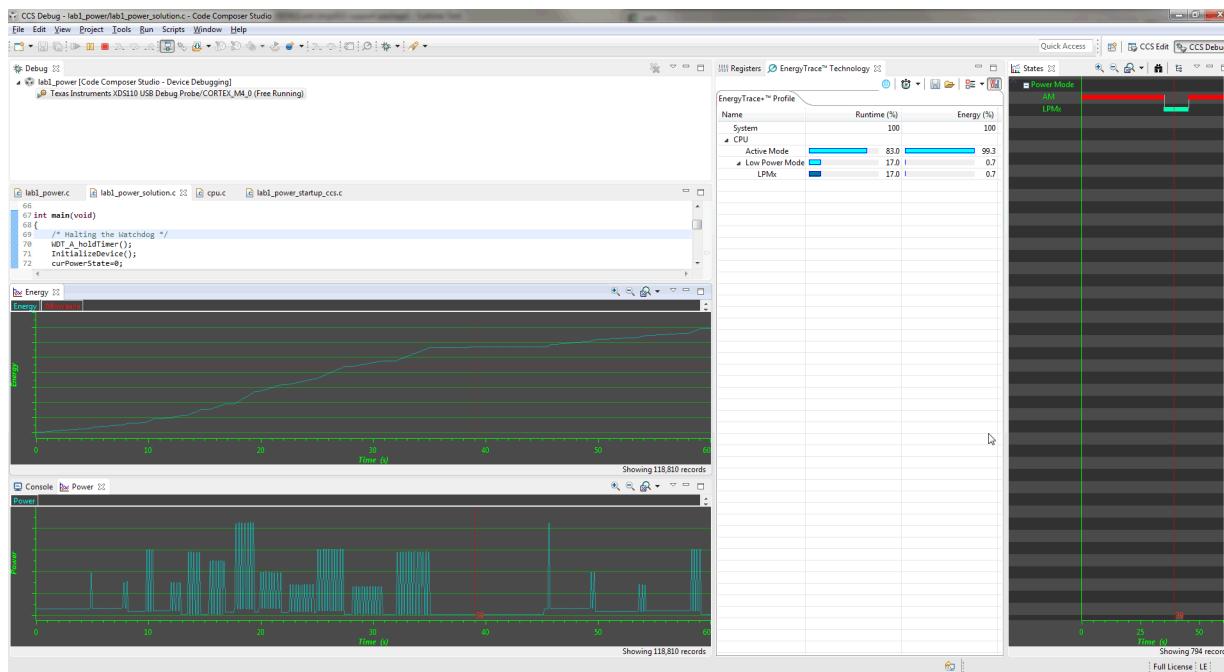


Figure 9. EnergyTrace Windows

This data allows the user to see exactly where and how energy is consumed in their application. Optimizations for energy can be quickly made for the lowest power application possible.

On the LaunchPad development kit, EnergyTrace technology measures the current that enters the target side of the LaunchPad. This includes all BoosterPacks plug-in modules plugged in, and anything else connected to the 3V3 power rail. For more information about powering the LaunchPad kit, see [Section 2.5](#).

For more information about EnergyTrace technology, see <http://www.ti.com/tool/energytrace>.

For more details and questions about setting up and using EnergyTrace technology with the MSP432P4111, see the [Code Composer Studio™ IDE 7.1+ for SimpleLink™ MSP432™ Microcontrollers User's Guide](#).

2.4 Special Features

2.4.1 Liquid Crystal Display

The MSP432P4111 LaunchPad development kit features an onboard LCD. This LCD is driven by the internal LCD driver on the MSP432P4111 device.

Many LCD segments are available, including six full alpha-numeric numbers or letters in addition to several symbols at the top for various modes or applications. [Figure 10](#) shows the layout of the LCD, and [Table 2](#) and [Table 3](#) list the mapping of these segments.

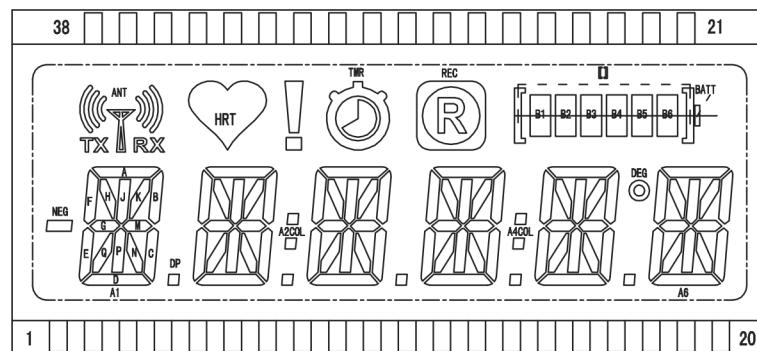


Figure 10. LCD Segment Layout

Table 2. LCD FH-1138P Segment Mapping

Pin	COM3	COM2	COM1	COM0
1	A1E	A1F	A1G	A1M
2	A1A	A1B	A1C	A1D
3	A1Q	NEG	A1N	A1DP
4	A1H	A1J	A1K	A1P
5	A2E	A2F	A2G	A2M
6	A2A	A2B	A2C	A2D
7	A2Q	A2COL	A2N	A2DP
8	A2H	A2J	A2K	A2P
9	A3R	A3F	A3G	A3M
10	A3A	A3B	A3C	A3D
11	A3Q	ANT	A3N	A3DP
12	A3H	A3J	A3K	A3P
13	A4R	A4F	A4G	A4M
14	A4A	A4B	A4C	A4D
15	A4Q	A4COL	A4N	A4DP
16	A4H	A4J	A4K	A4P
17	A5E	A5F	A5G	A5M
18	A5A	A5B	A5C	A5D
19	A5Q	DEG	A5N	A5DP
20	A5H	A5J	A5K	A5P
21	COM3	-	-	-
22	-	COM2	-	-
23	-	-	COM1	-
24	-	-	-	COM0
25	-	-	-	-
26	-	-	-	-
27	-	-	-	-
28	-	-	-	-
29	-	-	-	-
30	-	-	-	-
31	-	-	-	-
32	TMR	HRT	REC	!
33	B6	B4	B2	BATT
34	B5	B3	B1	□

Table 2. LCD FH-1138P Segment Mapping (continued)

Pin	COM3	COM2	COM1	COM0
35	A6E	A6F	A6G	A6M
36	A6A	A6B	A6C	A6D
37	A6Q	TX	A6N	RX
38	A6H	A6J	A6K	A6P

Table 3. LCD-to-MSP Connections

MSP432 Pin Function	MSP432 LCD Memory Register	MSP432 Port Pin	FH-1138P LCD Pin	COM3	COM2	COM1	COM0
L47	LCDM47	P8.2	38	A6H	A6J	A6K	A6P
L46	LCDM46	P8.3	37	A6Q	TX	A6N	RX
L45	LCDM45	P8.4	36	A6A	A6B	A6C	A6D
L44	LCDM44	P8.5	35	A6E	A6F	A6G	A6M
L43	LCDM43	P9.4	12	A3H	A3J	A3K	A3P
L42	LCDM42	P9.5	11	A3Q	ANT	A3N	A3DP
L41	LCDM41	P9.6	10	A3A	A3B	A3C	A3D
L40	LCDM40	P9.7	9	A3R	A3F	A3G	A3M
L39	LCDM39	P10.0	16	A4H	A4J	A4K	A4P
L38	LCDM38	P10.1	15	A4Q	A4COL	A4N	A4DP
L37	LCDM37	P10.2	14	A4A	A4B	A4C	A4D
L36	LCDM36	P10.3	13	A4R	A4F	A4G	A4M
L35	LCDM35	P10.4	8	A2H	A2J	A2K	A2P
L34	LCDM34	P10.5	7	A2Q	A2COL	A2N	A2DP
L33	LCDM33	P9.2	6	A2A	A2B	A2C	A2D
L32	LCDM32	P9.3	5	A2E	A2F	A2G	A2M
L31	LCDM31	P7.4	20	A5H	A5J	A5K	A5P
L30	LCDM30	P7.5	19	A5Q	DEG	A5N	A5DP
L29	LCDM29	P7.6	18	A5A	A5B	A5C	A5D
L28	LCDM28	P7.7	17	A5E	A5F	A5G	A5M
L27	LCDM27	P6.2	23	-	-	COM1	-
L26	LCDM26	P6.3	24	-	-	-	COM0
L25	LCDM25	P6.4					
L24	LCDM24	P6.5					
L23	LCDM23	P2.4					
L22	LCDM22	P2.5					
L21	LCDM21	P2.6					
L20	LCDM20	P2.7					
L19	LCDM19	P8.6	4	A1H	A1J	A1K	A1P
L18	LCDM18	P8.7	3	A1Q	NEG	A1N	A1DP
L17	LCDM17	P9.0	2	A1A	A1B	A1C	A1D
L16	LCDM16	P9.1	1	A1E	A1F	A1G	A1M
L15	LCDM15	P1.4/P6.0					
L14	LCDM14	P1.5/P6.1					
L13	LCDM13	P1.6/P4.0					
L12	LCDM12	P1.7/P4.1					
L11	LCDM11	P2.0					
L10	LCDM10	P2.1					

Table 3. LCD-to-MSP Connections (continued)

MSP432 Pin Function	MSP432 LCD Memory Register	MSP432 Port Pin	FH-1138P LCD Pin	COM3	COM2	COM1	COM0
L9	LCDM9	P2.2					
L8	LCDM8	P2.3					
L7	LCDM7	P3.0					
L6	LCDM6	P3.1	22	-	COM2	-	-
L5	LCDM5	P3.2					
L4	LCDM4	P3.3					
L3	LCDM3	P3.4	21	COM3	-	-	-
L2	LCDM2	P3.5	32	TMR	HRT	REC	!
L1	LCDM1	P3.6	34	B5	B3	B1	□
L0	LCDM0	P3.7	33	B6	B4	B2	BATT

2.4.1.1 LCD Design Considerations

The LCD_F peripheral on the MSP432P4111 allows each LCD pin to be configured as either COMMON or SEGMENT functionalities. Particular care was taken to determine LCD pin connections between the microcontroller and the LCD on the MSP-EXP432P4111 LaunchPad. Pin connections were chosen to optimize software ease-of-use while minimizing hardware layout complexity.

The LCD_F peripheral also features hardware animation, and up to 8 segments can be put into a repetitive animation pattern. In 4-mux mode, the animation feature is supported on LCD pins L0 and L1. The battery symbol on the MSP-EXP432P4111 onboard LCD consists of 8 segments and serves as a good example to demonstrate the LCD_F animation feature. Therefore, the battery segments are connected to L0 and L1 (see [Table 3](#)).

Pins P3.5, P3.6, and P3.7 from the MSP432P4111 are connected to both the BoosterPack plug-in module connector and the LCD. If these pins are driven with a digital voltage from either the MSP432P4111 or a BoosterPack plug-in module, the LCD segments connected to these pins may be damaged.

2.4.2 LMT70 Temperature Sensor

The MSP432P4111 LaunchPad development kit features an LMT70 ultra-small high-precision low-power CMOS analog temperature sensor. The analog output voltage of the LMT70 correlates directly to the ambient temperature (see [Figure 11](#)).

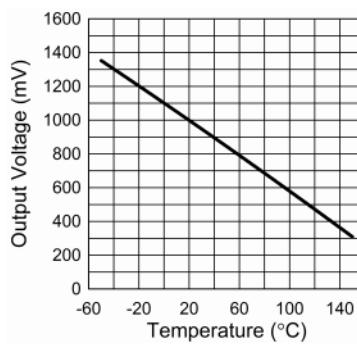


Figure 11. LMT70 Output Transfer Function

The analog output pin of the LMT70 is connected to Analog Input A2 (P5.3) on the MSP432P4111. The VCC pin on the LMT70 is connected to the target side 3.3 V on the LaunchPad development kit through a 0-ohm resistor, R8. By default, R8 is not-populated to enable accurate low-power measurements of the MSP432P4111 on the LaunchPad development kit. To enable the LMT70 to make analog temperature measurements, populate R8 with a 0402 0-ohm resistor. [Figure 12](#) shows the location of R8.



Figure 12. R8 Location

2.5 Power

The board was designed to accommodate various powering methods, including through the onboard XDS110-ET and from an external source or BoosterPack module.

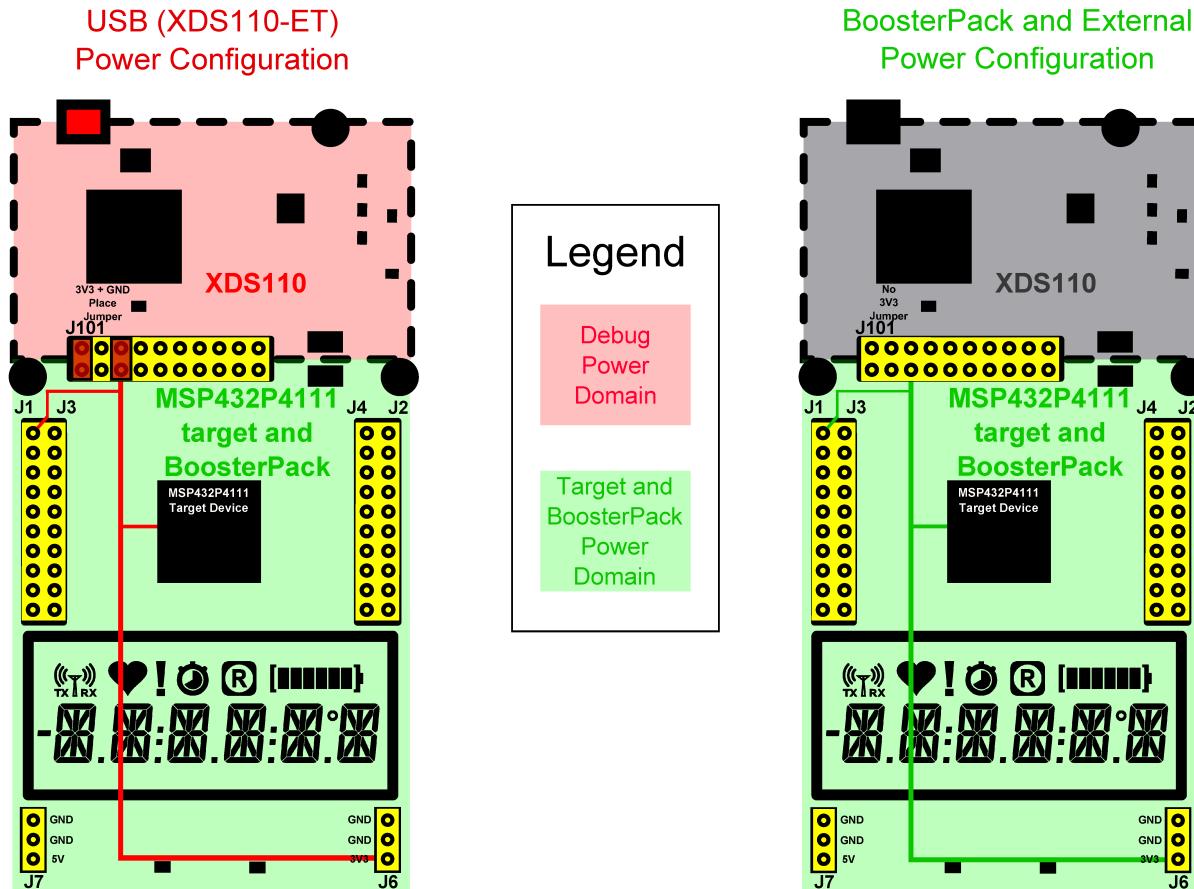


Figure 13. MSP-EXP432P4111 Power Block Diagram

2.5.1 XDS110-ET USB Power

The most common power-supply scenario is from USB through the XDS110-ET debugger. This provides 5-V power from the USB and also regulates this power rail to 3.3 V for XDS110-ET operation and 3.3 V to the target side of the LaunchPad. Power from the XDS110-ET is controlled by the isolation block 3V3 jumper, ensure this jumper is connected for power to be provided to the target MCU side.

Under normal operation, the LDO on the XDS110-ET can supply up to 500 mA of current to the target side including any BoosterPack plug-in modules plugged in. However, when debugging and using the EnergyTrace technology tool, this current is limited to 75 mA total. Be aware of this current limitation when using EnergyTrace technology.

2.5.2 BoosterPack Plug-in Module and External Power Supply

Header J6 is present on the board to supply external power directly. It is important to comply with the device voltage operation specifications when supplying external power. The MSP432P4111 has an operating range of 1.62 V to 3.7 V. More information can be found in the *MSP432P4x1xl SimpleLink Mixed-Signal Microcontroller* data sheet.

2.6 Measure MSP432 Current Draw

To measure the current draw of the MSP432P4111, use the 3V3 jumper on the jumper isolation block. The current measured includes the target device and any current drawn through the BoosterPack headers.

To measure ultra-low power:

1. Remove the 3V3 jumper in the isolation block, and attach an ammeter across this jumper.
2. Consider the effect that the backchannel UART and any circuitry attached to the MSP432P4111 may have on current draw. Disconnect these at the isolation block if possible, or at least consider their current sinking and sourcing capability in the final measurement.
3. Make sure there are no floating input I/Os. These cause unnecessary extra current draw. Every I/O should either be driven out or, if it is an input, should be pulled or driven to a high or low level.
4. Begin target execution.
5. Measure the current. Keep in mind that if the current levels are fluctuating, it may be difficult to get a stable measurement. It is easier to measure quiescent states.

For a better look at the power consumed in your application use EnergyTrace+ Technology. EnergyTrace+ Technology allows the user to see energy consumed as the application progresses. For more details about EnergyTrace+ technology, see [Section 2.3.5](#).

2.7 Clocking

The MSP-EXP432P4111 provides external clocks in addition to the internal clocks in the device.

- Q1: 32-kHz crystal (LFXTCLK)
- Q2: 48-MHz crystal (HFXTCLK)

The 32-kHz crystal allows for lower LPM3 sleep currents and higher precision clock source than the default internal 32-kHz REFOCLK. Therefore, the presence of the crystal allows the full range of low-power modes to be used.

The 48-MHz crystal allows the device to run at its maximum operating speed for MCLK and HSMCLK.

The MSP432P4111 device has several internal clocks that can be sourced from many clock sources. Most peripherals on the device can select which of the internal clocks to use to operate at the desired speed.

Table 4 lists the default configuration of the internal clocks in the device.

Table 4. Default Clock Operation

Clock	Default Clock Source	Default Clock Frequency	Description
MCLK	DCO	3 MHz	Master clock. Sources CPU and peripherals.
HSMCLK	DCO	3 MHz	Subsystem master clock. Sources peripherals.
SMCLK	DCO	3 MHz	Low-speed subsystem master clock. Sources peripherals.
ACLK	LFXT (or REFO if no crystal present)	32.768 kHz	Auxiliary clock. Sources peripherals.
BCLK	LFXT (or REFO if no crystal present)	32.768 kHz	Low-speed backup domain clock. Sources LPM peripherals.

For more information about configuring internal clocks and using the external oscillators, see the [MSP432P4xx SimpleLink™ Microcontrollers Technical Reference Manual](#).

2.8 BoosterPack Plug-in Module Pinout

The MSP-EXP432P4111 LaunchPad development kit adheres to the 40-pin LaunchPad development kit pinout standard. A standard was created to aid compatibility between LaunchPad and BoosterPack tools across the TI ecosystem.

The 40-pin standard is compatible with the 20-pin standard that is used by other LaunchPads like the [MSP-EXP430FR4133](#). This allows some subset of functionality of 40-pin BoosterPacks to be used with 20-pin LaunchPads.

While most BoosterPacks are compliant with the standard, some are not. The MSP-EXP432P4111 LaunchPad kit is compatible with all 20-pin and 40-pin BoosterPack plug-in modules that are compliant with the standard. If the reseller or owner of the BoosterPack plug-in module does not explicitly indicate compatibility with the MSP-EXP432P4111 LaunchPad development kit, compare the schematic of the candidate BoosterPack plug-in module with the LaunchPad development kit to ensure compatibility. Keep in mind that sometimes conflicts can be resolved by changing the MSP432P4111 device pin function configuration in software. More information about compatibility can also be found at <http://www.ti.com/launchpad>.

Pins P3.5, P3.6, and P3.7 from the MSP432P4111 are connected to both the BoosterPack plug-in module connector and the LCD. If these pins are driven with a digital voltage from either the MSP432P4111 or a BoosterPack plug-in module, the LCD segments connected to these pins may be damaged.

[Figure 14](#) shows the 40-pin pinout of the MSP-EXP432P4111 LaunchPad development kit.

Software configuration of the pin functions plays a role in compatibility. The MSP-EXP432P4111 LaunchPad development kit side of the dashed line in [Figure 14](#) shows all of the functions for which the MSP432P4111 device pins can be configured. This can also be seen in the MSP432P4111 data sheet. The BoosterPack plug-in module side of the dashed line shows the standard. The MSP432P4111 function whose color matches the BoosterPack function shows the specific software-configurable function by which the MSP-EXP432P4111 LaunchPad kit adheres to the standard.

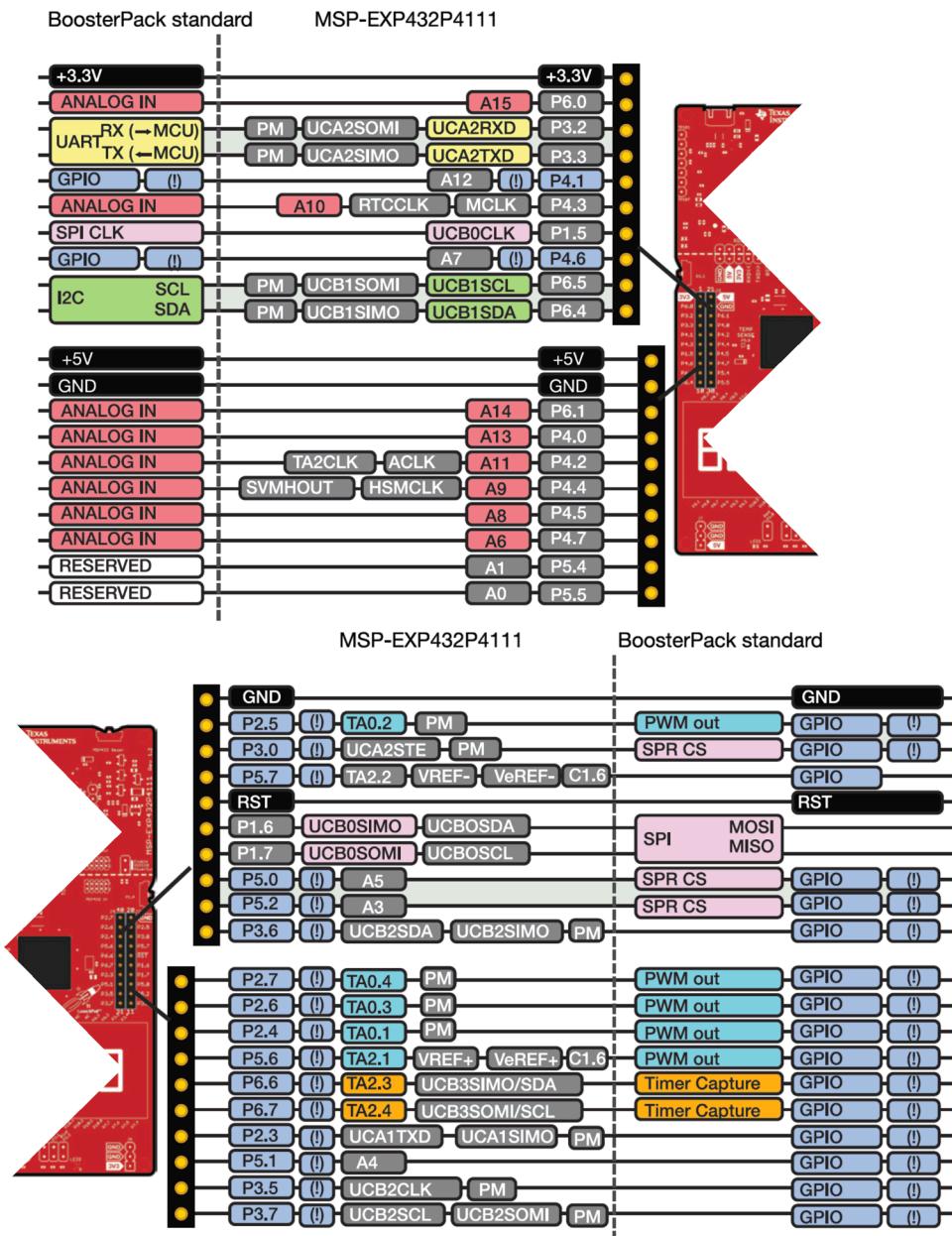


Figure 14. LaunchPad-to-BoosterPack Connector Pinout

2.9 Design Files

2.9.1 Hardware Design Files

Schematics can be found in Section 5. All design files including schematics, layout, bill of materials (BOM), Gerber files, and documentation are available in the MSP-EXP432P4111 Hardware Design Files zip folder.

2.10 Hardware Change Log

[Table 5](#) lists the change history of the MSP-EXP432P4111 hardware.

Table 5. Hardware Change Log

PCB Revision	Date	Description	MSP432P4111 Device Revisions
Rev 1.2	May 2017	Preproduction release	XMS432P4111 Rev A
Rev 1.3	November 2017	Production release	XMS432P4111 Rev A or MSP432P4111 Rev A

3 Software and Additional Resources

3.1 TI Resource Explorer

TI Resource Explorer is a cloud-enabled repository that consolidates everything you need to start your development. Using TI Resource Explorer, you will find code examples, documentation, hardware design files, training, and more. TI Resource Explorer is context-aware, delivering relevant material as you navigate the folder structure to the left.

It is easy to find all of the relevant material associated to your development kit, device, or SDK. Use the search bars to look for your specific LaunchPad kit, and the content in TI Resource Explorer filters appropriately. The content is currently broken up into three main sections: Device, Development Tools, and Software.

Visit the [List of Examples specific to the MSP-EXP432P4111 LaunchPad kit](#).

3.2 SimpleLink MSP432P4 SDK

SimpleLink MSP432P4 SDK is a complete software development kit for MSP432P4 devices. Within the SDK, you will find code examples, drivers, middleware, documentation, migration guides & more.

For more information, visit [www.ti.com/simplelink](#).

3.2.1 SimpleLink Academy

Get started with SimpleLink Academy inside the SimpleLink SDK. It provides a comprehensive collection of trainings & exercises to get you up and running with the SimpleLink MSP432P4 SDK.

Visit the [SimpleLink Academy Overview](#) page.

The SimpleLink Academy also provides a LaunchPad Out-Of-Box walk-through to get you started on your MSP-EXP432P4111 LaunchPad development kit.

Visit the [MSP-EXP432P4111 LaunchPad Out-Of-Box](#) page.

3.3 Integrated Development Environments

Although the source files can be viewed with any text editor, more can be done with the projects if they are opened with a [development environment](#) like Code Composer Studio IDE, Keil µVision IDE, IAR Embedded Workbench IDE, or Energia IDE.

3.3.1 Code Composer Studio IDE

Code Composer Studio Desktop is a professional integrated development environment that supports TI's Microcontroller and Embedded Processors portfolio. Code Composer Studio comprises a suite of tools used to develop and debug embedded applications. It includes an optimizing C/C++ compiler, source code editor, project build environment, debugger, profiler, and many other features.

NOTE: To use the SimpleLink MSP432P4 SDK, Code Composer Studio IDE version 7.3.0 or later is required.

Learn more about the Code Composer Studio IDE and download it at <http://www.ti.com/tool/ccstudio>.

3.3.2 IAR Embedded Workbench for Arm IDE

IAR Embedded Workbench for Arm IDE is another very powerful integrated development environment that allows you to develop and manage complete embedded application projects. It integrates the IAR C/C++ Compiler, IAR Assembler, IAR ILINK Linker, editor, project manager, command line build utility, and IAR C-SPY® Debugger.

NOTE: To use the SimpleLink MSP432 SDK, IAR Embedded Workbench for Arm IDE version 8.11.2 or later is required.

Learn more about the IAR Embedded Workbench IDE and download it at <https://www.iar.com/iar-embedded-workbench/arm>.

3.3.3 Keil µVision IDE

The µVision IDE is an embedded project development environment included in Keil's Microcontroller Development Kit Version 5, that provides a source code editor, project manager, and make utility tool. µVision supports all the Keil tools including C/C++ Compiler, Macro Assembler, Linker, Library Manager, and Object-HEX Converter.

NOTE: To use the SimpleLink MSP432 SDK, Keil µVision IDE version 5.23 or later is required.

Learn more about the Keil µVision IDE and download it at <http://www.keil.com/arm/IDE.asp>.

3.3.4 Energia IDE

Your device is also supported by the Energia IDE, an Arduino-compatible framework of APIs and libraries.

Energia is a simple, open-source community-driven code editor that is based on the [Wiring](#) and [Arduino](#) framework. Energia provides unmatched ease of use through very high level APIs that can be used across hardware platforms. Energia is a light-weight IDE that doesn't have the full feature set of CCS, IAR, or Keil. However, Energia is great for anyone who wants to get started very quickly or who doesn't have significant coding experience.

Learn more about the Energia IDE and download it at www.energia.nu.

3.4 LaunchPad Websites

More information about the LaunchPad development kits, supported BoosterPack plug-in modules, and available resources can be found at the following links:

- [MSP-EXP432P4111 Tool Folder](#): resources specific to this particular LaunchPad development kit
- [TI LaunchPad development kit portal](#): information about all LaunchPad development kits from TI

3.5 MSP432P4111 MCU

3.5.1 Device Documentation

At some point, you will probably want more information about the MSP432P4111 MCU. [Table 6](#) describes the organization of the documentation for this MCU.

Table 6. How MSP Device Documentation is Organized

Document	For MSP432P4111	Description
Device family user's guide	MSP432P4xx SimpleLink™ Microcontrollers Technical Reference Manual	Architectural information about the device, including all modules and peripherals such as clocks, timers, ADC, and so on
Device-specific data sheet	MSP432P411x, MSP432P401x SimpleLink™ Mixed-Signal Microcontrollers	Device-specific information and all parametric information for this device

3.5.2 MSP432 Application Notes and TI Designs

Many application notes can be found at www.ti.com/msp432, in addition to [TI reference designs](#) with practical design examples and topics.

3.6 Community Resources

3.6.1 TI E2E™ Community

Search the forums at e2e.ti.com. If you cannot find your answer, post your question to the community.

3.6.2 Community at Large

Many online communities focus on the LaunchPad development kits; for example, <http://www.43oh.com>. You can find additional tools, resources, and support from these communities.

4 FAQ

Q: I can't program my LaunchPad; the IDE can't connect to target. What's wrong?

A: Check the following:

1. Are the JTAG jumpers populated (GND, RST, TMS, TCK, TDO, TDI)?
2. Check for power to the target
 - a. Are the 3V3 and GND jumpers on J101 populated and USB cable plugged in?
 - b. If using an external debug probe, is USB power provided as shown above? Otherwise, is external power provided to the target?
3. Check the debug probe settings: change to *Serial Wire Debug (SWD) without SWO*.
 - a. Under *targetconfigs*, double-click the *.ccxml file.
 - b. Click the *Advanced* tab at the bottom.
 - c. Click *Texas Instruments XDS110 USB Debug Probe*.
 - d. Under *Connection Properties*, change *SWD Mode* settings to *Use SWD Mode with SWO Trace Disabled*.

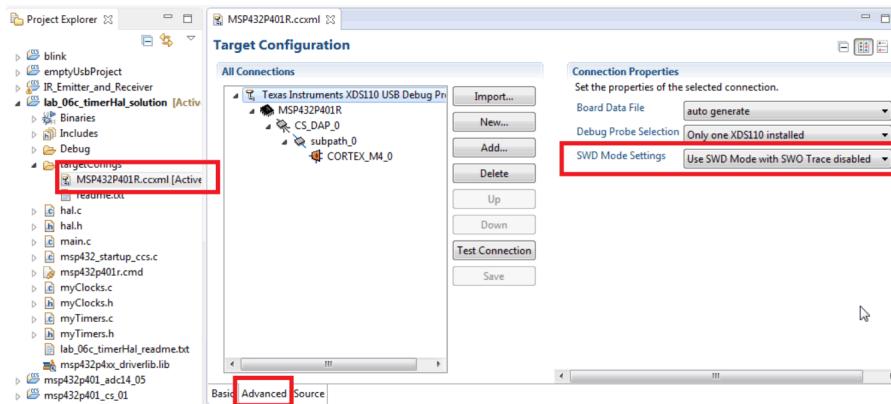


Figure 15. SWD Mode Settings

4. When the settings of Port J (PJSEL0 and PJSEL1 bits) are changed, full JTAG access is prevented on these pins. Changing to use SWD allows access through the dedicated debug pins only.
5. If even this can't connect, reset the device to factory settings
6. Select **View -> Target Configurations**. CCS will show the target configuration

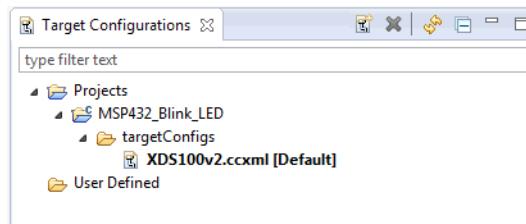


Figure 16. Target Configurations

If using the onboard debug probe, XDS110-ET is shown.

7. Right click **Launch Selected Configuration**

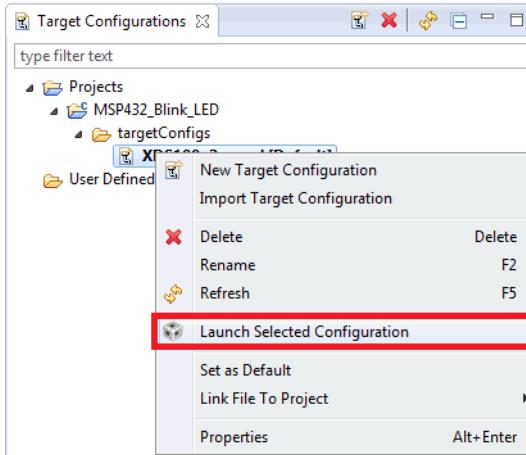


Figure 17. Launch Selected Configuration

8. The debug probe now connects to the device (which is still possible), but does not try to halt the CPU, write to registers or even download code (which would not be possible). The Debug view that is spawned shows the CPU core, but marks it as disconnected.
9. Right click **Show all cores**

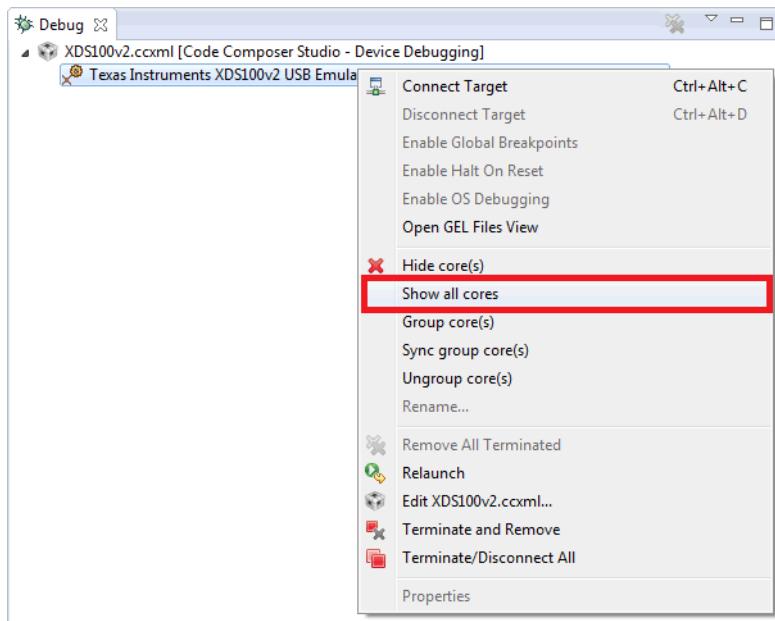


Figure 18. Show All Cores

The MSP432 Debug Access Port, or DAP, is shown under **Non Debuggable devices**.

10. Right click **Connect Target**

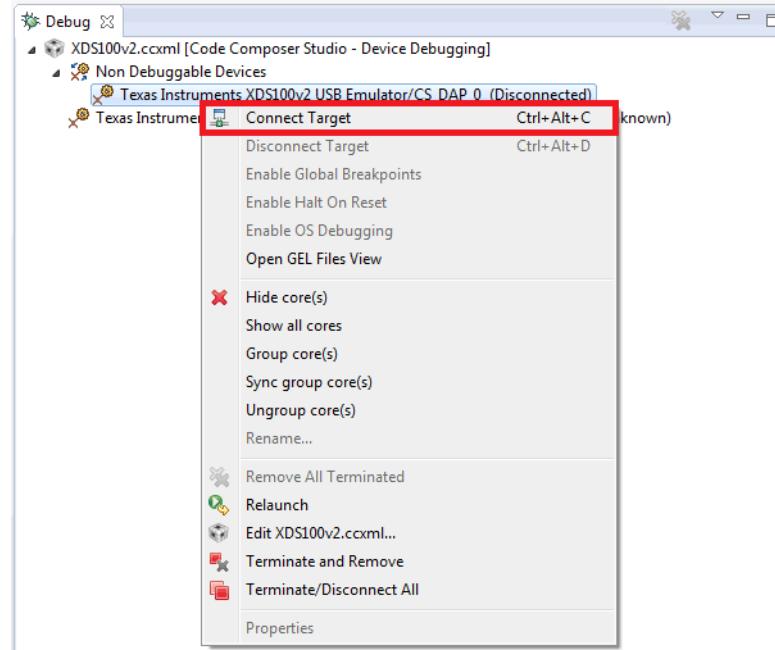


Figure 19. Connect Target

11. Run the following script to return the device back to factory settings (see [Figure 20](#)):

Scripts > default > MSP432_Factory_Reset

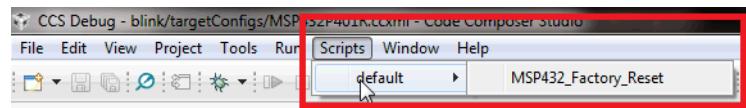


Figure 20. MSP432_Factory_Reset Script

- The preceding instructions are generally the same for all IDEs, but the exact steps may vary slightly by IDE. See the following IDE user's guides for additional details:
 - [Code Composer Studio™ IDE 7.1+ for SimpleLink™ MSP432™ Microcontrollers User's Guide](#)
 - [Arm® Keil® MDK Version 5 for SimpleLink™ MSP432™ Microcontrollers User's Guide](#)
 - [IAR Embedded Workbench for Arm 7.x for SimpleLink™ MSP432™ Microcontrollers User's Guide](#)

Q: How do I use the LaunchPad development kit and my Segger J-Link to debug the target externally? It won't connect to the onboard connector?

A: The Segger J-Link does not come with an adapter for the 10-pin small-pitch Arm connector. The adapter cable is available from [Segger](#).

Q: Problems plugging the MSP432 LaunchPad into a USB3.0 Port

A: It has been observed that when the MSP432 LaunchPad development kit is connected to USB3.0 ports provided by a certain combination of USB3.0 host controller hardware and associated device drivers that the IDE is unable to establish a debug session with the LaunchPad development kit, resulting in an error message like “CS_DAP_0: Error connecting to the target: (Error -260 @ 0x0) An attempt to connect to the XDS110 failed.” in the case of Code Composer Studio. In this case the CCS-provided low-level command line utility ‘xdsdfu’ will also not be able to establish a connection with the LaunchPad.

Specifically, this issue was observed on PCs running Windows 7 that show the “Renesas Electronics USB 3.0 Host Controller” and the associated “Renesas Electronics USB 3.0 Root Hub” in the device manager. After updating the associated Windows USB drivers to more recent versions obtained from the hardware vendor the issue went away. There might be other USB3.0 hardware and device driver combinations that will lead to the same issue. If you think you might be affected try contacting your PC vendor or try locating and installing more recent versions of the USB3.0 device drivers. Alternatively, connect the LaunchPad development kit to an USB2.0 port on your PC if available.

Q: I can't get the backchannel UART to connect. What's wrong?

A: Check the following:

- Do the baud rate in the host's terminal application and the eUSCI settings match?
- Are the appropriate jumpers in place on the isolation jumper block?
- Probe on RXD and send data from the host. If you don't see data, it might be a problem on the host side.
- Probe on TXD while sending data from the MSP432. If you don't see data, it might be a configuration problem with the eUSCI module.

5 Schematics

The following figures show the schematics for the MSP-EXP432P4111. These schematics and additional information are also available in the hardware design files zip file in the [MSP-EXP432P4111 tool folder](#).

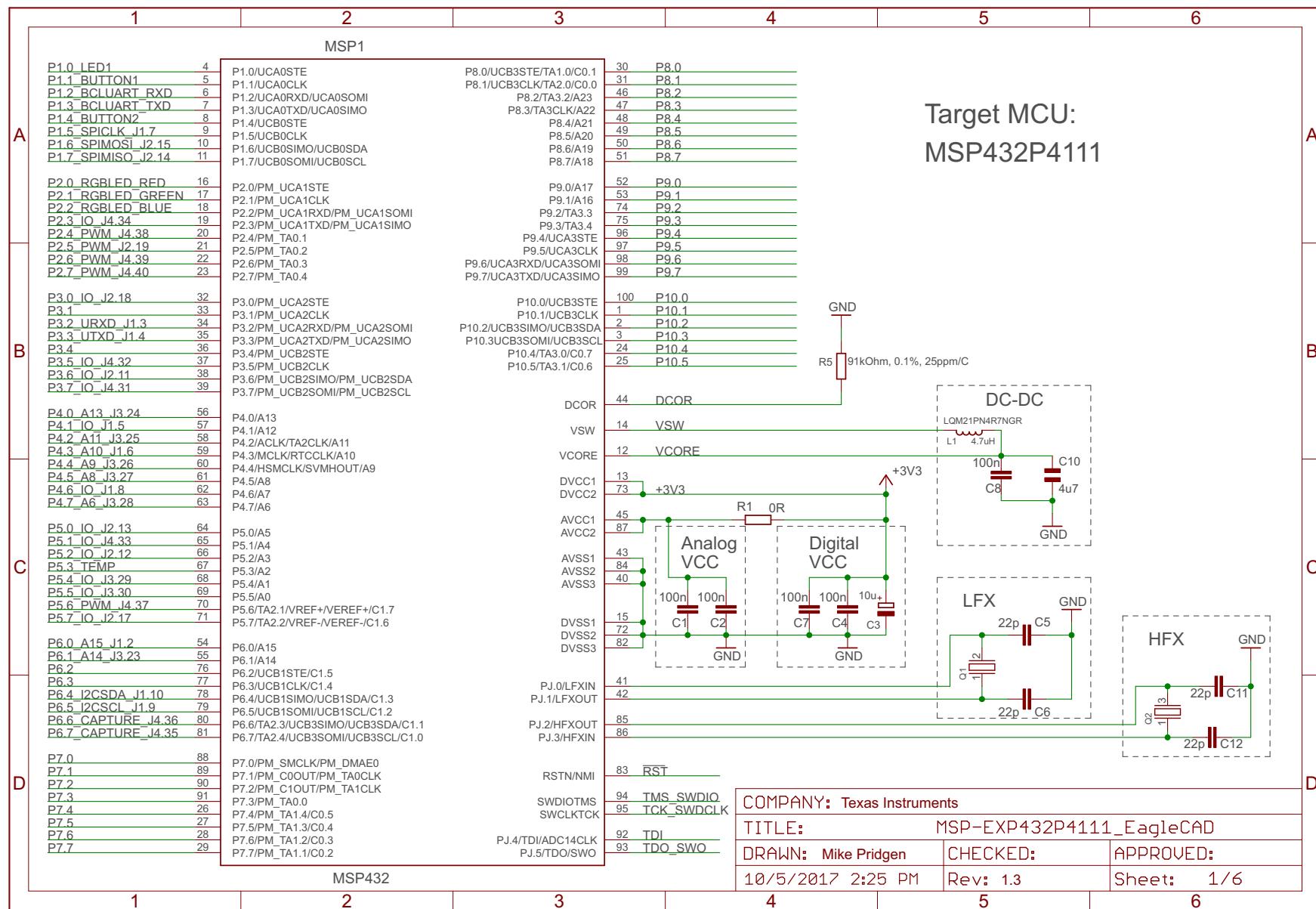
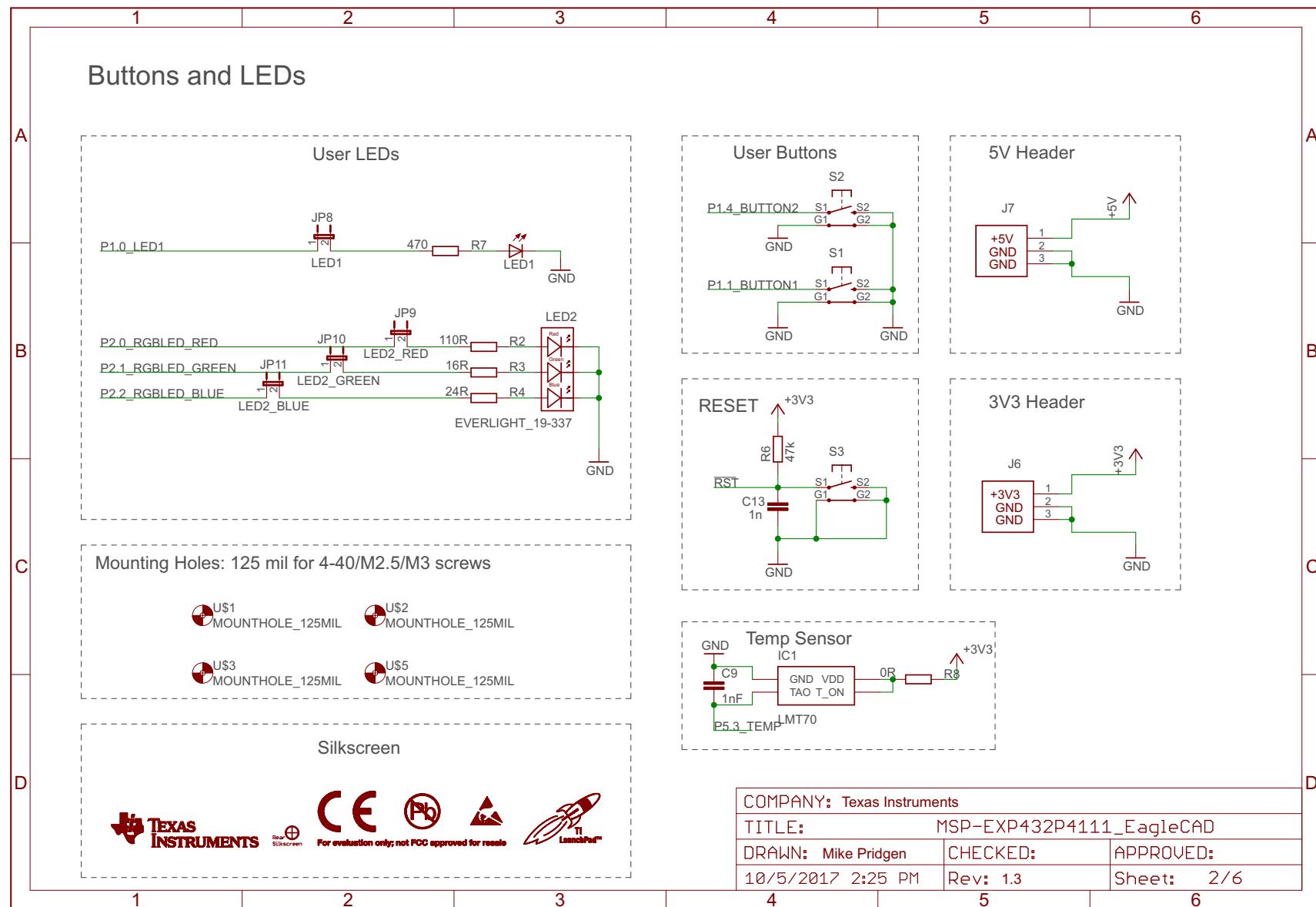


Figure 21. Schematics (1 of 6)



Copyright © 2016, Texas Instruments Incorporated

Figure 22. Schematics (2 of 6)

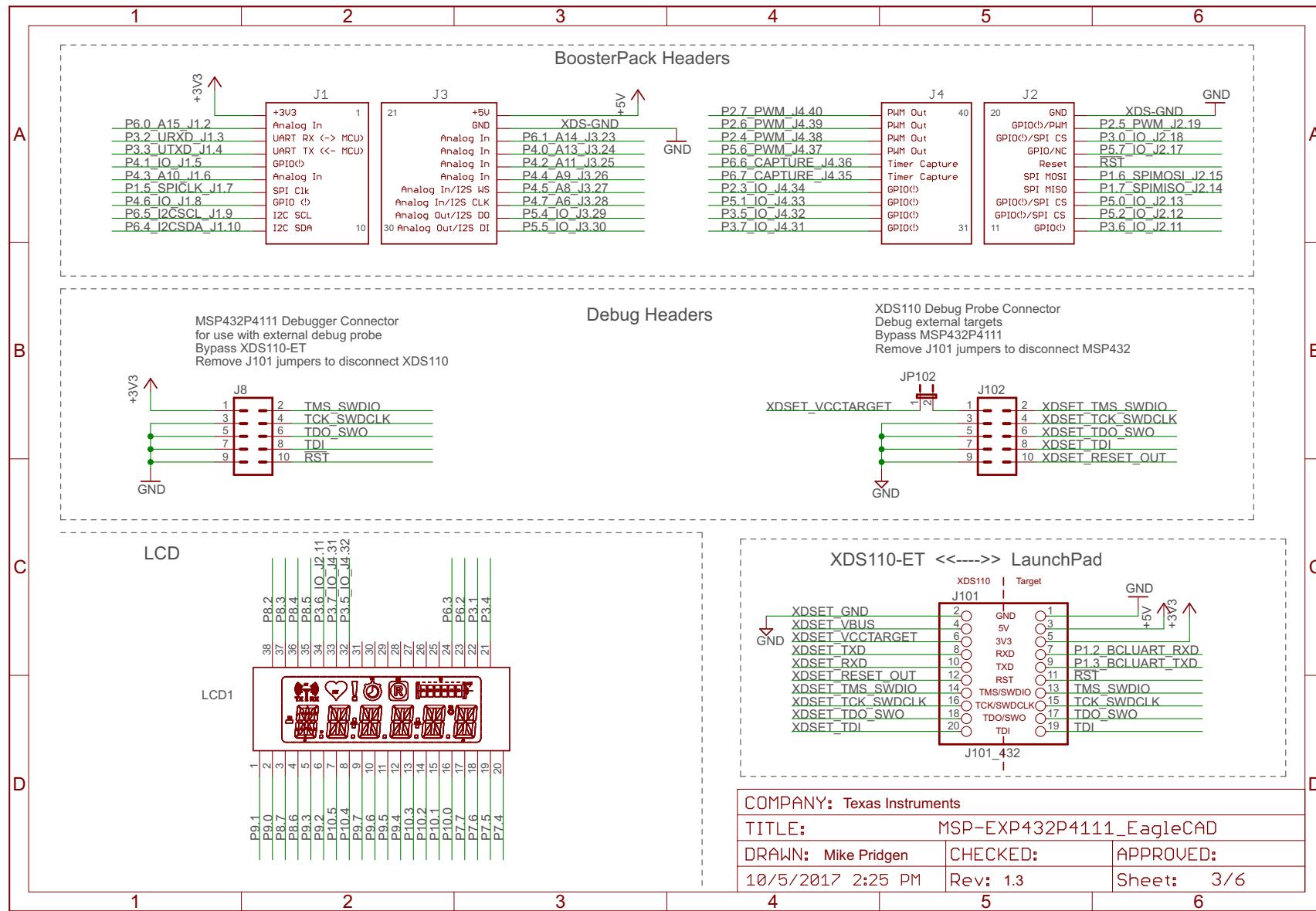


Figure 23. Schematics (3 of 6)

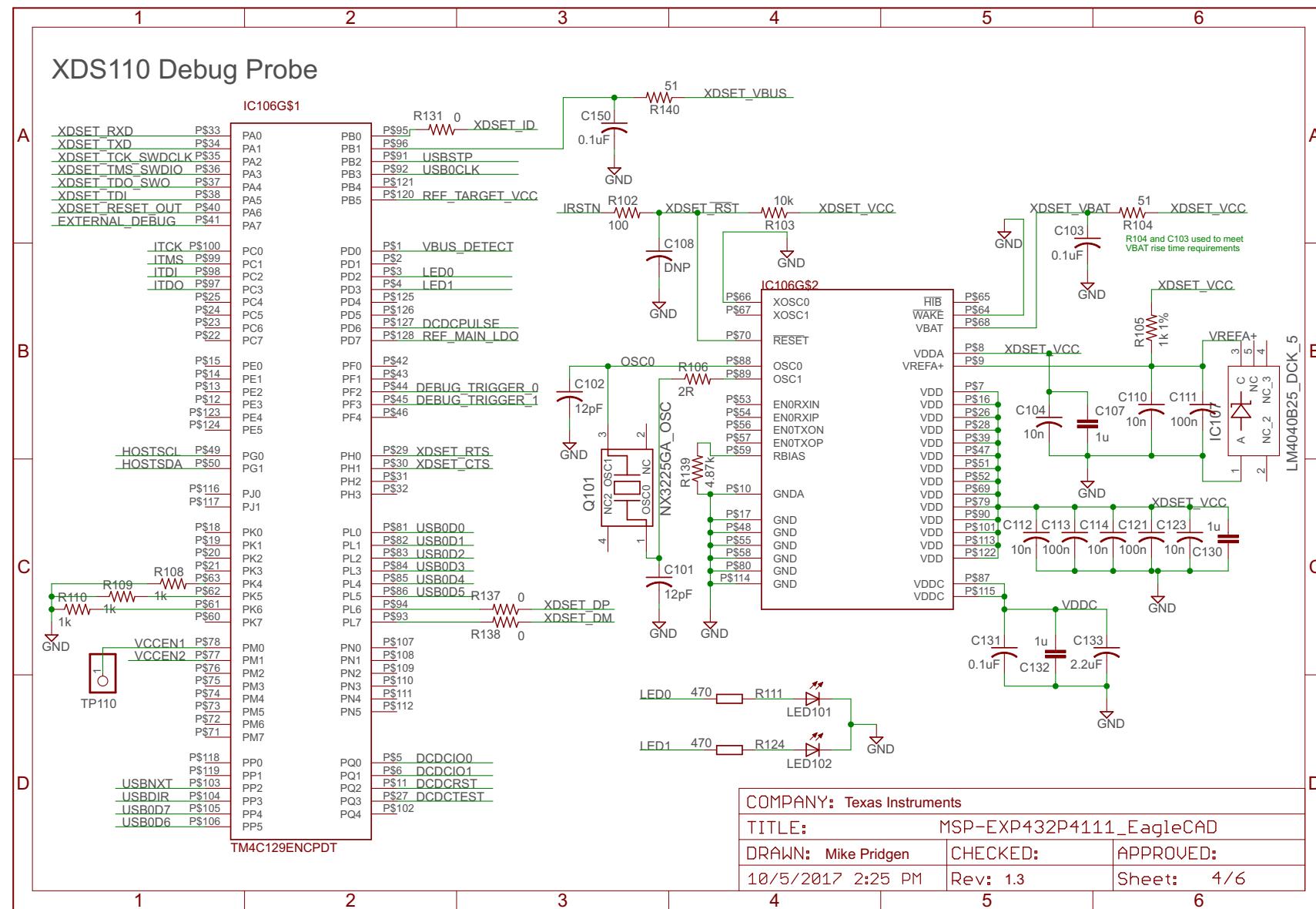


Figure 24. Schematics (4 of 6)

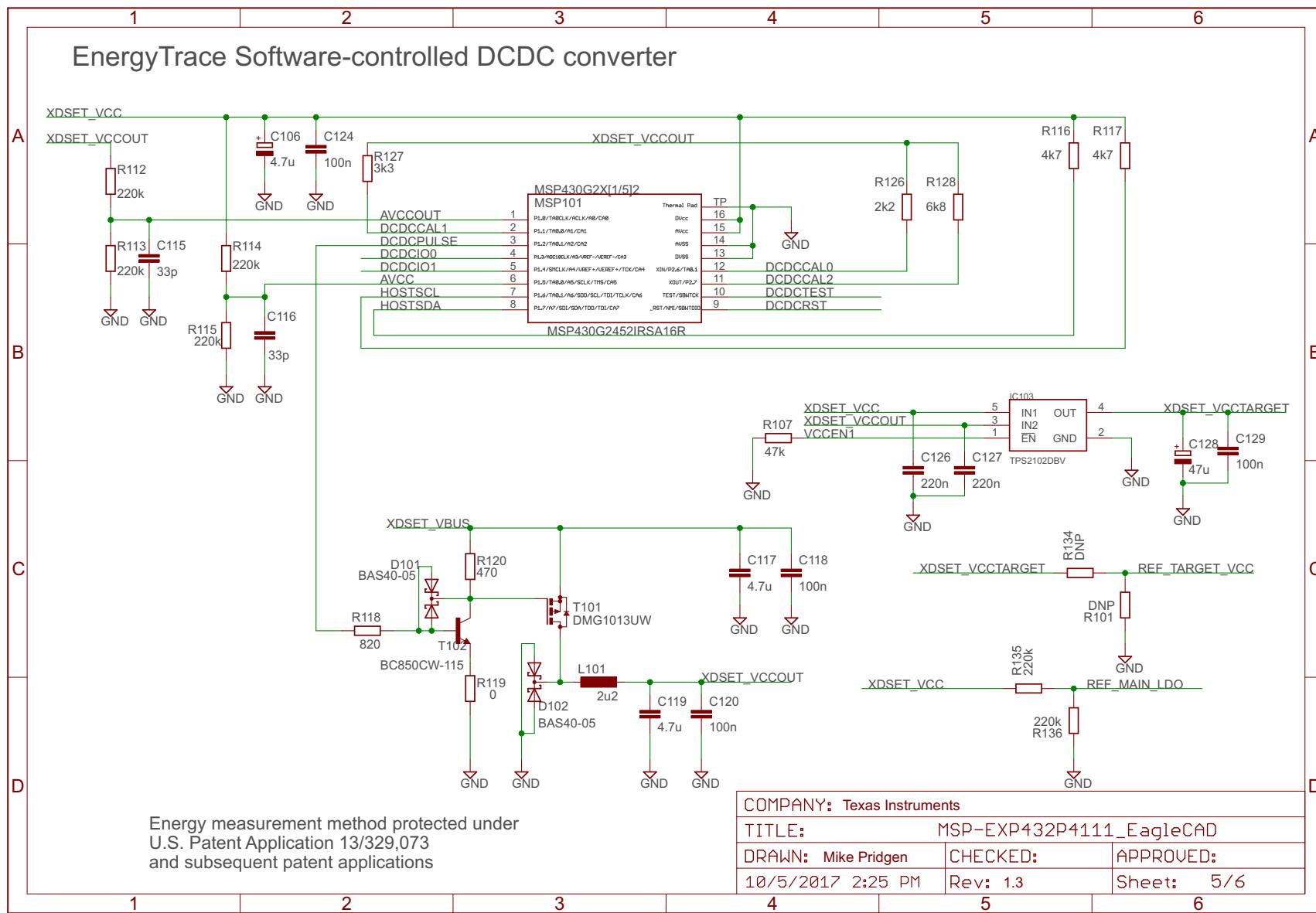
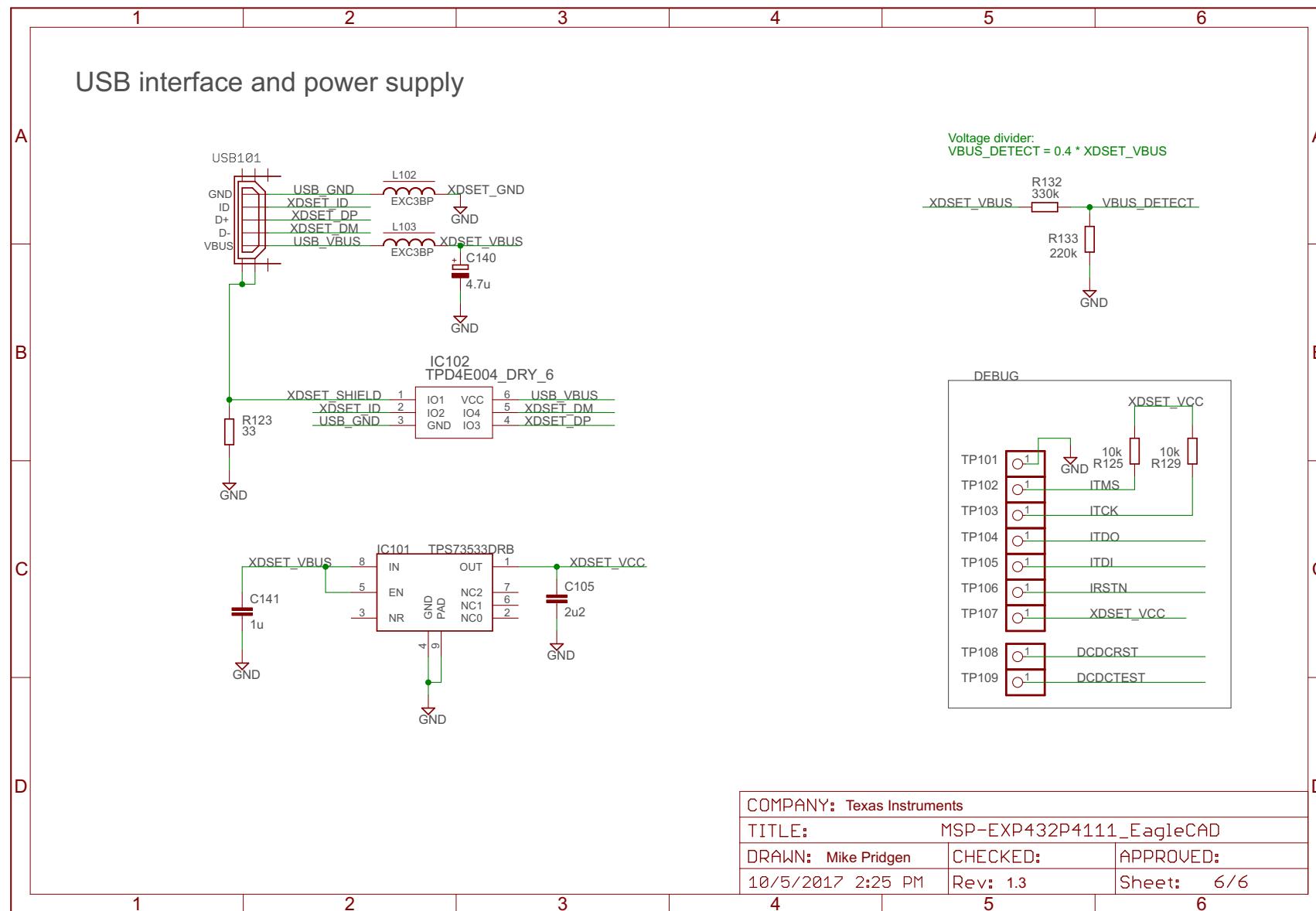


Figure 25. Schematics (5 of 6)



Copyright © 2016, Texas Instruments Incorporated

Figure 26. Schematics (6 of 6)

Revision History

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

Changes from October 24, 2017 to March 8, 2018	Page
• Changed the link to the 10-pin Cortex-M JTAG cable in Section 2.3.4, Using the XDS110-ET Emulator With a Different Target	11
• Updated the answer to the question that begins "Q: How do I use the LaunchPad development kit and my Segger J-Link..." in Section 4, FAQ	26

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ("TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products (<http://www.ti.com/sc/docs/stdterms.htm>), evaluation modules, and samples (<http://www.ti.com/sc/docs/samptersms.htm>).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2018, Texas Instruments Incorporated