CY8CKIT-046

PSoC® 4 L-Series Pioneer Kit Guide

Doc. #: 002-03344 Rev. *A

Copyrights

Flash Code Protection

Cypress products meet the specifications contained in their particular Cypress PSoC Data Sheets. Cypress believes that its family of PSoC products is one of the most secure families of its kind on the market today, regardless of how they are used. There may be methods, unknown to Cypress, that can breach the code protection features. Any of these methods, to our knowledge, would be dishonest and possibly illegal. Neither Cypress nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as 'unbreakable'.

Cypress is willing to work with the customer who is concerned about the integrity of their code. Code protection is constantly evolving. We at Cypress are committed to continuously improving the code protection features of our products.

Contents

Safety Information

Regulatory Compliance

The CY8CKIT-046 PSoC® 4 L-Series Pioneer Kit is intended for use as a development platform for hardware or software in a laboratory environment. The board is an open system design, which does not include a shielded enclosure. This may cause interference to other electrical or electronic devices in close proximity. In a domestic environment, this product may cause radio interference. In such cases, you may be required to take adequate preventive measures. In addition, this board should not be used near any medical equipment or RF devices.

Attaching additional wiring to this product or modifying the product operation from the factory default may affect its performance and cause interference with other apparatus in the immediate vicinity. If such interference is detected, suitable mitigating measures should be taken.

The PSoC 4 L-Series Pioneer Kit, as shipped from the factory, has been verified to meet with the requirements of CE as a Class A product.

	The PSoC 4 L-Series Pioneer Kit contains ESD-sensitive devices. Electrostatic charges readily accumulate on the human body and any equipment, and can discharge without detection. Permanent damage may occur on devices subjected to high-energy discharges. Proper ESD precautions are recommended to avoid performance degradation or loss of functionality. Store unused kit boards in the protective shipping package.

	End-of-Life/Product Recycling This kit has an end-of life five years from the date of manufacture mentioned on the back of the box. Contact your nearest recycler for discarding the kit.

General Safety Instructions

ESD Protection

ESD can damage boards and associated components. Cypress recommends that you perform procedures only at an ESD workstation. If such a workstation is not available, use appropriate ESD protection by wearing an antistatic wrist strap attached to the chassis ground (any unpainted metal surface) on your board when handling parts.

Handling Boards

PSoC 4 L-Series Pioneer Kit boards are sensitive to ESD. Hold the board only by its edges. After removing the board from its box, place it on a grounded, static-free surface. Use a conductive foam pad if available. Do not slide the board over any surface.

1. Introduction

Thank you for your interest in the CY8CKIT-046 PSoC® 4 L-Series Pioneer Kit. This kit enables customers to evaluate and develop s using the PSoC 4200L device, a member of the PSoC 4 L-Series family.

The PSoC 4 L-Series family is an expansion to the PSoC 4 device family that offers more flash, GPIOs, programmable analog, digital, and communication blocks. PSoC 4200L device is a programmable embedded system-on chip, integrating custom analog and digital peripheral functions, memory, and an ARM® Cortex®-M0 microcontroller on a single chip. The programmable analog and digital peripheral functions allow higher flexibility, in-field tuning of the design, and faster time-to-market. In addition to offering the standard programmable analog, digital, and communications blocks present in the PSoC 4 device family, the PSoC 4 L-Series family adds USB Full-Speed interface to the PSoC 4 family.

The PSoC 4 L-Series Pioneer Kit offers footprint-compatibility with Arduino™ shields and 6-pin Digilent® Pmod™ peripheral modules. The kit offers a unique ability to connect two Arduino shields side by side on the baseboard – dual Arduino footprint. The advantage of dual Arduino footprint is that two communication shields using the same protocol or shields with similar header I/Os can be interfaced simultaneously. In addition, this kit features two CapSense® proximity headers, a CapSense Gesture Pad with a radial slider, audio jack (3.5 mm) with microphone input, audio codec for digital audio streaming, µSD card interface, EZ-BLE™ PRoC™ Module, Energy Harvesting Power Management IC (PMIC) interface, an RGB LED, a push-button switch, an onboard programmer/debugger with USB-UART/I2C bridge functionality (KitProg), and a Cypress F-RAM™ (1 Mb). This kit supports operating voltages of 3.3 V or 5 V.

The kit board consists of two parts – main board and shield board. A perforated area is provided on the board to separate the two boards. When separated (snapped), the main board can act as an Arduino base board. The shield board can be used as a standalone Arduino shield. See Hardware Functional Description on page 58 for details.

You will use PSoC Creator™ to develop and debug your PSoC 4200L device s. PSoC Creator is Cypress' standard integrated design environment (IDE). If you are new to PSoC Creator, see the documentation on the PSoC Creator home page. You can also refer to application note *AN79953 – Getting Started with PSoC 4*, which introduces the PSoC 4 device.

1.1   Kit Contents

The CY8CKIT-046 PSoC 4 L-Series Pioneer Kit contains the following, as shown in Figure 1-1.

- PSoC 4 L-Series Pioneer board
- USB Standard-A to Mini-B cable
- Four jumper wires (4 inches each)
- Two proximity sensor wires (5 inches each)
- Stereo audio earphone with microphone
- Quick Start Guide

Figure 1-1. CY8CKIT-046 PSoC 4 L-Series Pioneer Kit

Inspect the contents of the kit; if you find any part missing, contact your nearest Cypress sales office for help.

1.2   Board Details

The PSoC 4 L-Series Pioneer Kit consists of the following blocks, as shown in Figure 1-2. Figure 1-3 shows the complete kit headers pin mapping.

Figure 1-2. PSoC 4 L-Series Pioneer Kit Markup

PSoC 4 L-Series Pioneer Board Details

1. Power (LED2) and Status (LED1) LEDs
2. KitProg USB connector (J10)
3. 32.768 kHz crystal (Y1)
4. PSoC 5LP programmer and debugger (CY8C5868LTI-LP039, U4)
5. PSoC 5LP I/O header (J8)
6. PSoC 4200L (CY8C4248BZI-L489, U5)
7. PSoC 4200L program and debug header (J6)
8. Digilent® Pmod™ compatible I/O header (J5)*
9. Cypress F-RAM 1 Mb (FM24V10-G, U6)
10. PSoC 4200L reset button (SW1)
11. Arduino™ compatible main board power header (J1)
12. RGB LED (LED3)
13. Arduino compatible main board I/O headers (J2/J3/J4)
14. User button (SW2)
15. PSoC 4200L current measurement jumper (J14)
16. PSoC 4200L USB connector (J11)

17. Audio CODEC (WM8731CLSEFL, U3)
18. Cypress Energy Harvesting PMIC (S6AE101A, U12)*
19. Arduino compatible shield board power header (J17)
20. Character LCD header (J23)*
21. Arduino compatible shield board I/O headers (J18/J19/J20)
22. CapSense Proximity headers (J21, J22)
23. µSD card interface (J25) and Cypress Serial NOR Flash (S25FL512S, U13)*
24. CapSense® Gesture Pad
25. EZ-BLE™ PRoC™ Module (CYBLE-022001-00, U8)*
26. EZ-BLE Programming header (BLE PROG)*
27. Energy Harvesting PMIC header (EH)*
28. EZ-BLE I/O header (BLE)*
29. Audio jack selection (AHJ/OMTP) switch (SW4)*
30. Audio jack - 3.5 mm (J16)
31. 17.2032 MHz crystal (Y2)
32. System power selection jumper (J9) and LDO (3.3V, U2)
33. PSoC 5LP program and debug header (J7)*

*Footprint only, not populated on the board

Figure 1-3. PSoC 4 L-Series Pioneer Kit Pinout

PSoC 4 L-Series Pioneer Board Details

Arduino Uno R3
PSoC 4 L-Series Pioneer Kit
Digilent Pmod
Character LCD
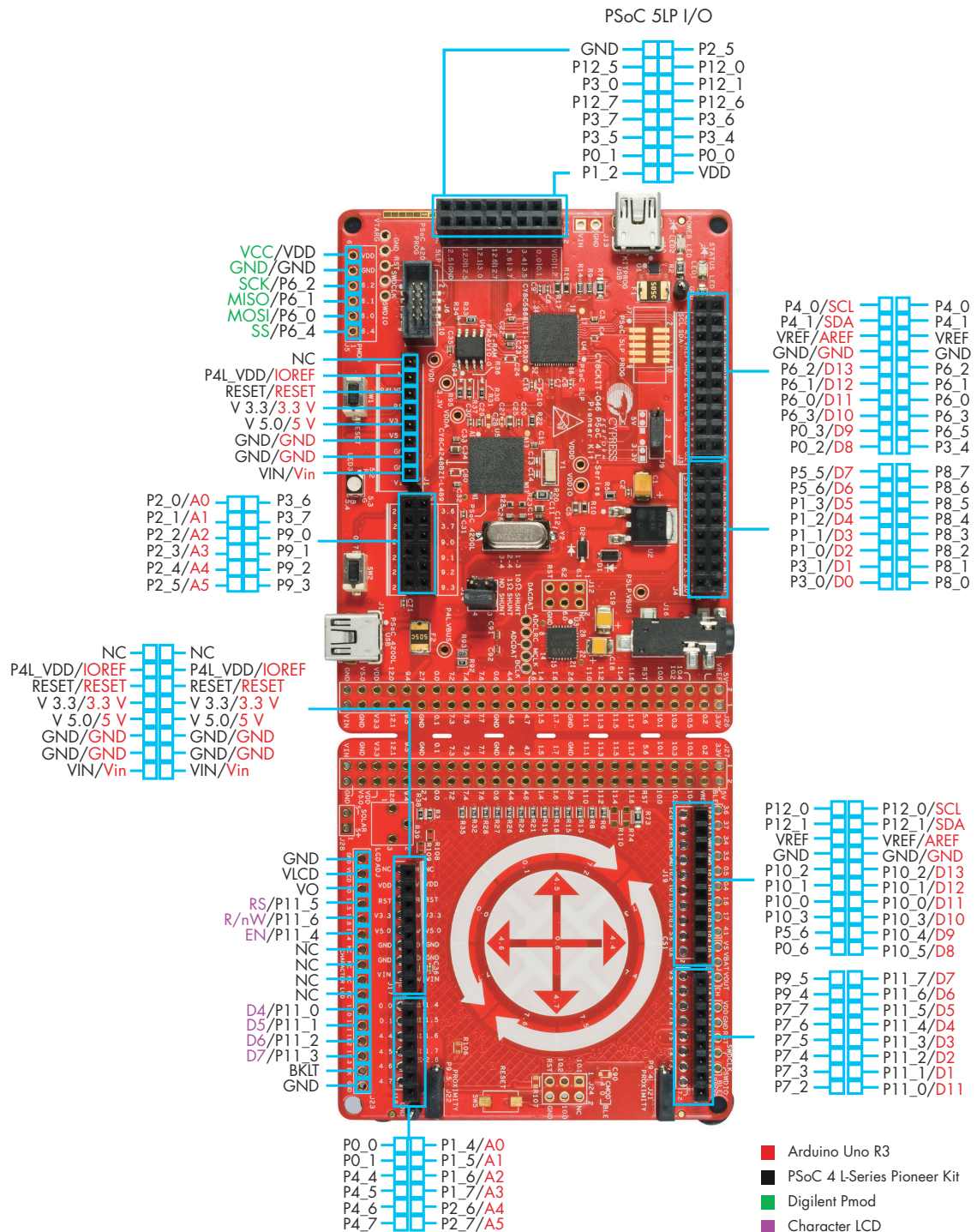
1.3   PSoC Creator

PSoC Creator is a state-of-the-art, easy-to-use IDE. It introduces revolutionary hardware and software co-design, powered by a library of fully verified and characterized PSoC Components™, as shown in Figure 1-4. With PSoC Creator, you can:

1. Drag and drop Components to build your hardware system design in the main design workspace
2. Co-design your application firmware with the PSoC hardware
3. Configure Components using Component Configuration Tools
4. Explore the library of 100+ Components
5. Access Component datasheets

Figure 1-4. PSoC Creator Features
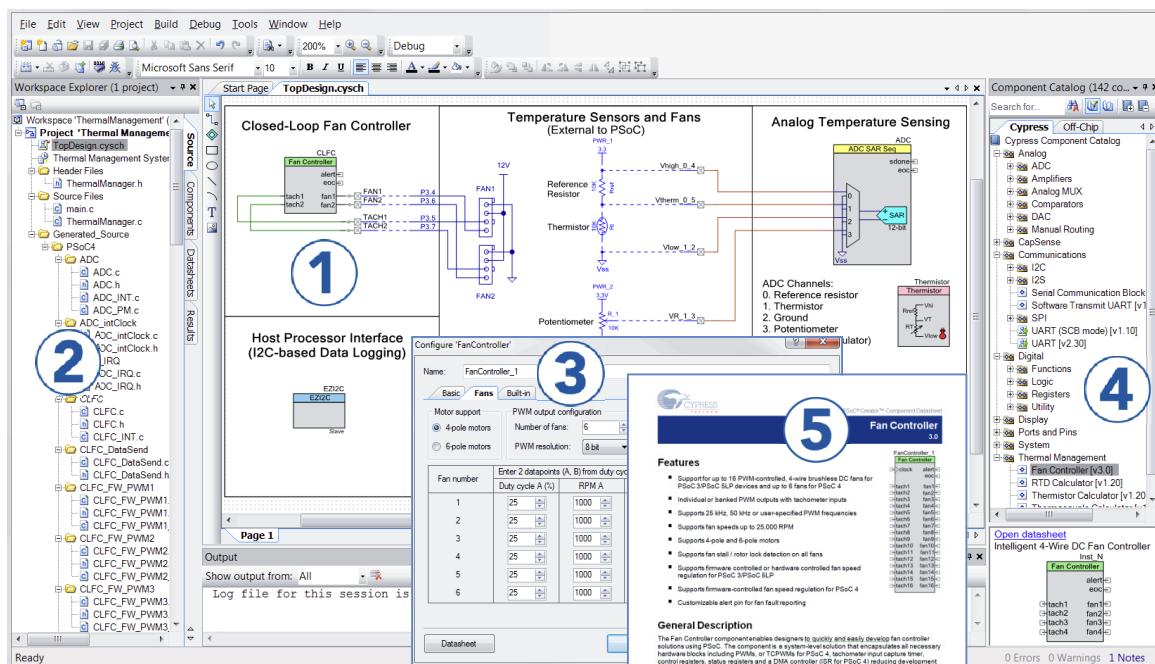
PSoC Creator also enables you to tap into an entire tool ecosystem with integrated compiler chains and production programmers for PSoC devices.

For more information, visit www.cypress.com/psoccreator.

1.3.1   PSoC Creator Code Examples

PSoC Creator includes a large number of code examples. These examples are accessible from the PSoC Creator **File** menu, as shown in Figure 1-5.

Code examples can speed up your design process by starting you off with a complete design, instead of a blank page. The code examples also show how to use PSoC Creator Components for various applications. Code examples and documentation are included, as shown in Figure 1-6.

In the Find Example dialog shown in Figure 1-6, you have several options:

■ Filter for examples based on device family or keyword.

■ Select from the list of examples offered based on the Filter Options.

■ View the documentation for the selection (on the **Documentation** tab).

- View the code for the selection on the **Sample Code** tab. You can also copy and paste code from this window to your , which can help speed up code development.
- Create a new workspace for the code example or add to your existing workspace. This can speed up your design process by starting you off with a complete, basic design. You can then adapt that design to your application.

Figure 1-5. Code Examples in PSoC Creator
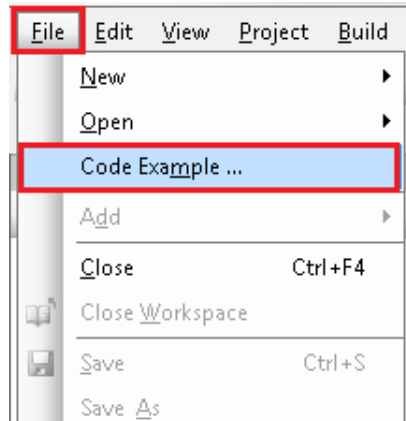
Figure 1-6. Code Examples with Sample Code
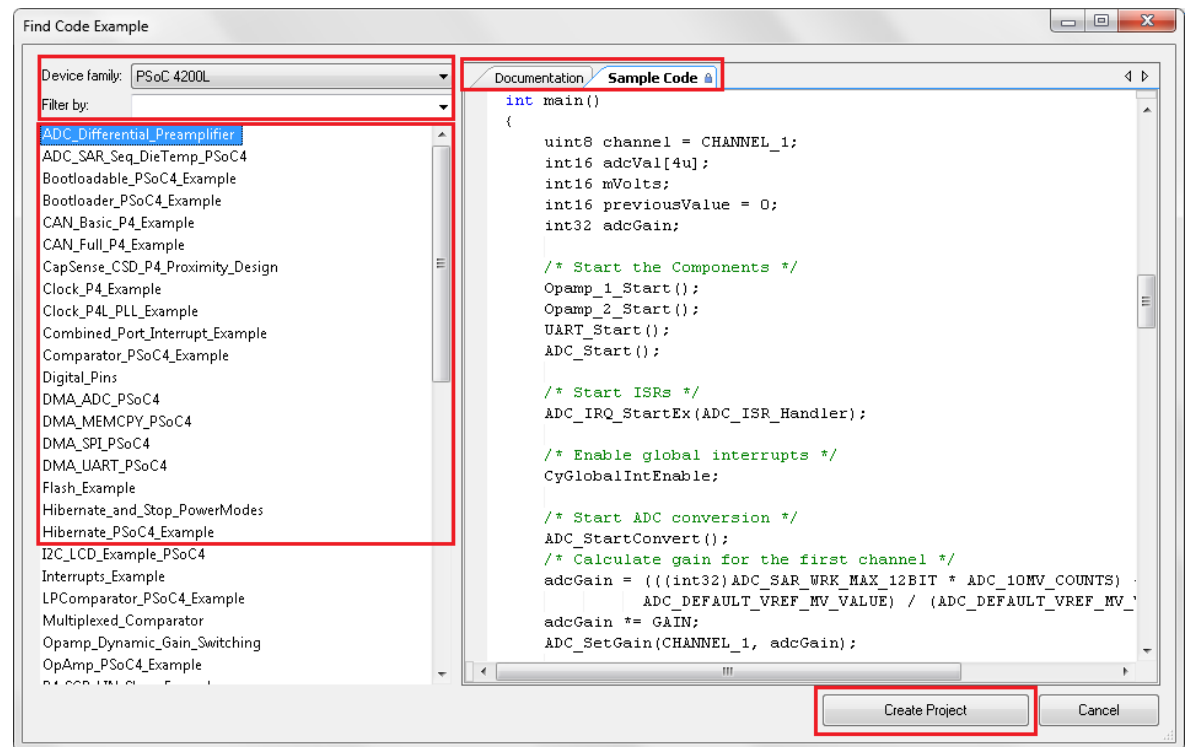
1.3.2   Kit Code Examples

You can access the installed kit code examples from the PSoC Creator Start Page. To access these examples, expand the **Kits** under the section **Examples and Kits**; then, expand the specific kit to see the code examples. Refer to the section Using the Kit Code Examples on page 27 for a detailed explanation on how to access the kit code examples.

1.3.3   PSoC Creator Help

Launch PSoC Creator and navigate to the following items:

■ **Quick Start Guide:** Choose **Help > Documentation > Quick Start Guide**. This guide gives you the basics for developing PSoC Creator s.

■ **Simple Component Example s:** Choose **File > Example s**. These example s demonstrate how to configure and use PSoC Creator Components. To access example s related to a specific Component, place the Component on the TopDesign schematic and right-click on the Component. Select the **Find Example** option in the context menu that appears.

■ **System Reference Guide**: Choose **Help > System Reference > System Reference Guide**. This guide lists and describes the system functions provided by PSoC Creator.

■ **Component Datasheets**: Right-click a Component and select **Open Datasheet.** Visit the PSoC 4 Component Datasheets page for a list of all PSoC 4 Component datasheets.

■ **Document Manager**: PSoC Creator provides a document manager to help you easily find and access the document resources. To open the document manager, choose the menu item **Help > Document Manager**.

1.4   Getting Started

This guide will help you get acquainted with the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit:

■ The Software Installation chapter on page 17 describes the installation of the kit software. This includes the PSoC Creator IDE to develop and debug the applications, and PSoC Programmer to program the *.hex* files onto the device.

■ The Kit Operation chapter on page 20 describes the major features of the PSoC 4 L-Series Pioneer Kit and functionalities such as programming, debugging, and the USB-UART and USB-I2C bridges.

■ The Code Examples chapter on page 27 describes multiple PSoC 4200L code examples that will help you understand how to create your own PSoC 4 s.

■ The Appendix on page 52 provides detailed hardware description and the method to use the onboard F-RAM, kit schematics, and the bill of materials (BOM).

1.5   Additional Learning Resources

Cypress provides a wealth of data at www.cypress.com to help you select the right PSoC device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see KBA86521, How to Design with PSoC 3, PSoC 4, and PSoC 5LP. The following is an abbreviated list for PSoC 4:

■ **Overview**: PSoC Portfolio and PSoC Roadmap.

■ **Product Selectors**: PSoC 4 Product Selector. In addition, PSoC Creator includes a device selection tool.

■ **Datasheets**: Describe and provide electrical specifications for the PSoC 4000, PSoC 4100, PSoC 4200, PSoC 4100M, PSoC 4200M, and PSoC 4200L device families.

■ CapSense Design Guide: Learn how to design capacitive touch-sensing applications with the PSoC 4 family of devices.

■ Application Notes and Code Examples: Cover a broad range of topics, from basic to advanced. Many of the application notes include code examples. Visit the PSoC 3/4/5 Code Examples webpage for a list of all available PSoC Creator code examples. To access code examples from within PSoC Creator – see PSoC Creator Code Examples on page 11.

- Technical Reference Manuals (TRM): Provide detailed descriptions of the architecture and registers in each PSoC 4 device family.
- Development Kits:
  - CY8CKIT-046, CY8CKIT-044, CY8CKIT-042, and CY8CKIT-040 are easy-to-use and inexpensive development platforms. These kits include connectors for Arduino-compatible shields and Digilent Pmod peripheral modules.
  - CY8CKIT-049 and CY8CKIT-043 are very low-cost prototyping platforms for sampling PSoC 4 devices.
  - MiniProg3 device provides an interface for flash programming and debug.
- Cypress Document Manager (CDM): An intuitive, Windows-based tool, which allows users to navigate, filter, search, view, and manage all Cypress documentation available on www.cypress.com.
- Knowledge Base Articles (KBA): Provide design and application tips from experts on using the device.
- PSoC Creator Training: Visit www.cypress.com/go/creatorstart/creatortraining for a comprehensive list of video trainings on PSoC Creator.
- Learning from Peers: Visit www.cypress.com/forums to meet enthusiastic PSoC developers discussing the next generation embedded systems on Cypress Developer Community Forums.

1.6   Technical Support

For assistance, visit Cypress Support or contact customer support at +1 (800) 541-4736 Ext. 2 (in the USA) or +1 (408) 943-2600 Ext. 2 (International).

You can also use the following support resources if you need quick assistance:
- Self-help (Technical Documents)
- Local Sales Office Locations

1.7   Document Conventions

Table 1-1. Document Conventions for Guides

Convention	Usage
Courier New	Displays file locations, user entered text, and source code: `C:\...cd\icc\`
Italics	Displays file names and reference documentation: Read about the *sourcefile.hex* file in the *PSoC Creator User Guide*.
[**Bracketed, Bold**]	Displays keyboard commands in procedures: [**Enter**] or [**Ctrl**] [**C**]
File > Open	Represents menu paths: File > Open > New
Bold	Displays commands, menu paths, and icon names in procedures: Click the **File** icon and then click **Open**.
Times New Roman	Displays an equation: $2 + 2 = 4$
Text in gray boxes	Describes Cautions or unique functionality of the product.

1.8   Acronyms

Table 1-2. Acronyms Used in the Document

Acronym	Description
ADC	analog-to-digital converter
AHJ	American Headphone Jack
BLE	Bluetooth low energy
CMOD	modulator capacitor
CODEC	encoder and decoder
CSD	CapSense Sigma Delta
CTANK	shield tank capacitor
DAC	digital-to-analog converter
DMA	direct memory access
DPDT	double-pole, double-throw
ECO	external crystal oscillator
ESD	electrostatic discharge
F-RAM	Ferroelectric Random Access Memory
GPIO	general-purpose input/output
HID	human interface device
I2C	Inter-Integrated Circuit
I2S	Inter IC Sound
IDE	integrated design environment
ILO	internal low-speed oscillator
IMO	internal main oscillator
LDO	low drop out (voltage regulator)
LFCLK	low-frequency clock
LSB	least significant byte
LSb	least significant bit
MSB	most significant byte
MSb	most significant bit
OMTP	Open Mobile Terminal Platform
PMIC	power management integrated circuit
PRoC	Programmable Radio-on-Chip
PSoC	Programmable Systems-on-Chip
PWM	pulse width modulator
RGB	red green blue
SAR	successive approximation register
SPI	serial peripheral interface
SWD	serial wire debug
TRRS	Tip-Ring-Ring-Sleeve
UART	universal asynchronous receiver transmitter

Table 1-2. Acronyms Used in the Document

Acronym	Description
USB	universal serial bus
USB-IF	USB Implementers Forum
WCO	watch crystal oscillator
WDT	watchdog timer

2. Software Installation

This chapter describes the steps to install the software tools and packages on a PC for using the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit. This includes the PSoC Creator IDE on which the s will be built and used for programming.

2.1 Before You Begin

All Cypress software installations require administrator privileges, but these are not required to run the software after it is installed. Close any other Cypress software that is currently running before installing the kit software.

2.2 Install Software

Follow these steps to install the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit software:

1. Download the kit software from www.cypress.com/CY8CKIT-046. The kit software is available for download in three formats.

   a. **CY8CKIT-046 Kit Complete Setup**: This installation package contains the files related to the kit including PSoC Creator and PSoC Programmer. However, it does not include the Windows Installer or Microsoft .NET framework packages. If these packages are not on your computer, the installer directs you to download and install them from the Internet.

   b. **CY8CKIT-046 Kit Only**: This executable file installs only the kit contents, which include kit code examples, hardware files, and user documents. This package can be used if all the software prerequisites (listed in step 5) are installed on your computer.

   c. **CY8CKIT-046 DVD ISO**: This file is a complete package, stored in a DVD-ROM image format, which you can use to create a DVD or extract using an ISO extraction program such as Winzip® or WinRAR. The file can also be mounted similar to a virtual CD/DVD using virtual drive programs such as Virtual CloneDrive and MagicISO. This file includes all the required prerequisite software, utilities, drivers, hardware files, and user documents.

2. If you have downloaded the ISO file, mount it on a virtual drive. Extract the ISO contents if you do not have a virtual drive to mount. Double-click *cyautorun.exe* in the root directory of the extracted content or mounted ISO if "Autorun from CD/DVD" is not enabled on the computer. The installation window will appear automatically.

   Note: If you are using the "**Kit Complete Setup**" or "**Kit Only**" file, then go to step 4 for installation.

3. Click **Install CY8CKIT-046** to start the kit installation, as shown in Figure 2-1.

Figure 2-1. Kit Installer Screen

4. Select the directory in which you want to install the files related to the PSoC 4 L-Series Pioneer Kit. Choose the directory and click **Next**.

5. The PSoC 4 L-Series Pioneer Kit installer automatically installs the required software if it is not present on your computer. Following are the required software:

   a. PSoC Creator 3.3 Service Pack 1 (SP1) or later: This software is also available at www.cypress.com/psoccreator.

   b. PSoC Programmer 3.24 or later: This is installed as part of PSoC Creator installation (www.cypress.com/programmer).

6. Choose the **Typical**, **Custom**, or **Complete** installation type (select '**Typical**' if you do not know which one to select) in the Product Installation Overview window, as shown in Figure 2-2. Click **Next** after you select the installation type.

Figure 2-2. Product Installation Overview
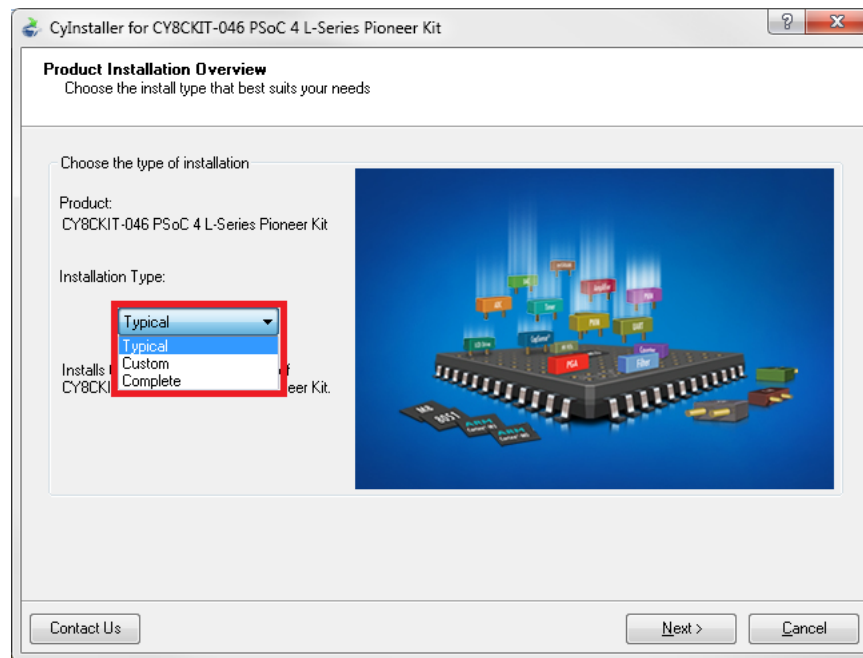
7. Read the License agreement and select **I accept the terms in the license agreement** to continue with the installation. Click **Next**.

8. When the installation begins, a list of packages appears on the installation page. A green check mark appears next to each package after successful installation.

9. Enter your contact information or select the check box **Continue Without Contact Information**. Click **Finish** to complete the kit installation.

10. After the installation is complete, the kit contents are available at the following location:
`<Install_Directory>\CY8CKIT-046 PSoC 4 L-Series Pioneer Kit`

Default location:

Windows OS (64-bit):
`C:\Program Files (x86)\Cypress\CY8CKIT-046 PSoC 4 L-Series Pioneer Kit`

Windows OS (32-bit):
`C:\Program Files\Cypress\CY8CKIT-046 PSoC 4 L-Series Pioneer Kit`

Note: For Windows 7/8/8.1/10 users, the installed files and the folder are read-only. To use the installer code example, follow the steps outlined in the Code Examples chapter on page 27. These steps will create an editable copy of the example in a path that you choose so the original installed example is not modified.

2.3 Uninstall Software

The software can be uninstalled using one of the following methods:

1. Go to **Start** > **All Programs** > **Cypress** > **Cypress Update Manager** and select the **Uninstall** button next to the product that needs to be uninstalled.

2. Go to **Start** > **Control Panel** > **Programs and Features** for Windows 7 or **Add/Remove Programs** for Windows XP; choose the product and select the **Uninstall** button.

3.   Kit Operation

This chapter introduces you to the various features of the PSoC 4 L-Series Pioneer Kit. It primarily includes the kit block overview, programming and debugging functionality, KitProg USB-UART and USB-I2C bridges, and the method to update the KitProg firmware.

3.1   Theory of Operation

The PSoC 4 L-Series Pioneer Kit is built around the PSoC 4200L device. Figure 3-1 shows the block diagram of the PSoC 4200L device. Refer to the PSoC 4200L datasheet for details on device features.

Figure 3-1. PSoC 4200L Block Diagram
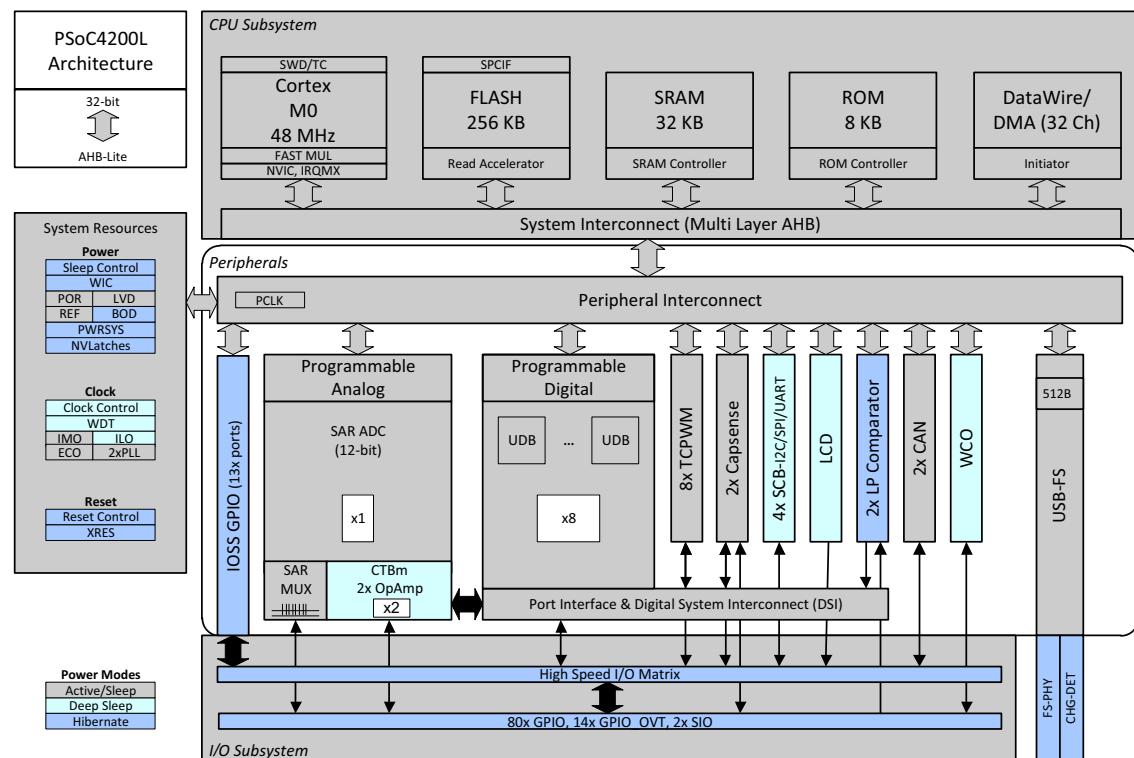
Figure 3-2 shows the block diagram for the PSoC 4 L-Series Pioneer Kit.

Figure 3-2. Block Diagram of PSoC 4 L-Series Pioneer Kit

[1] Arduino UNO baseboard compatible headers (female Arduino headers)
[2] Arduino UNO shield compatible headers (male Arduino headers)
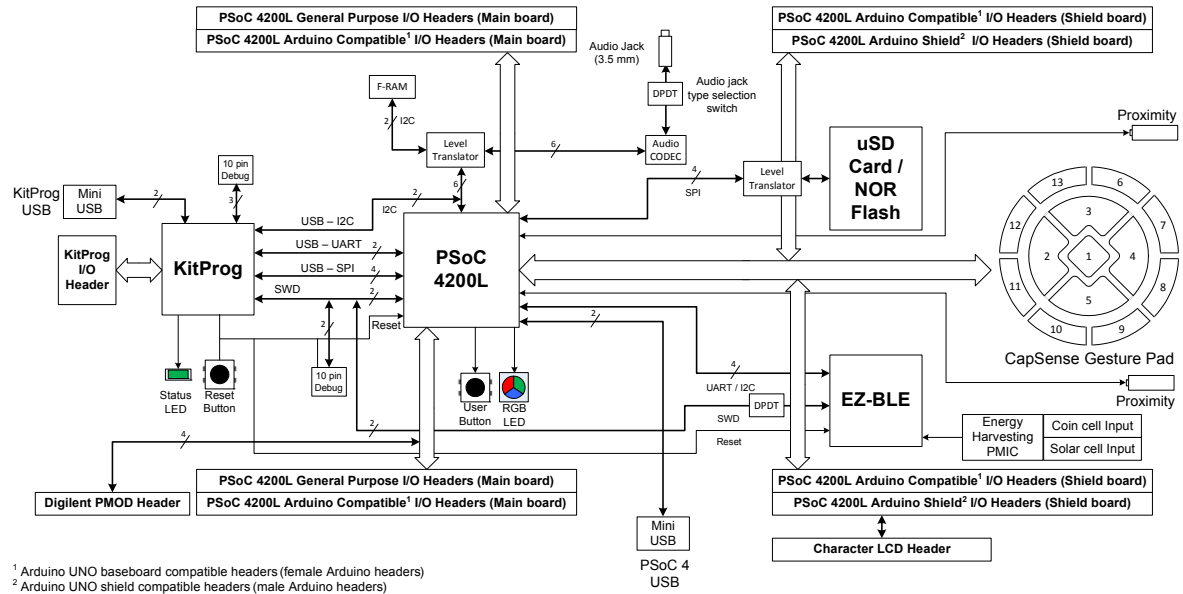
Figure 3-3. PSoC 4 L-Series Pioneer Kit Board Diagram

PSoC 4 L-Series Pioneer Board Details
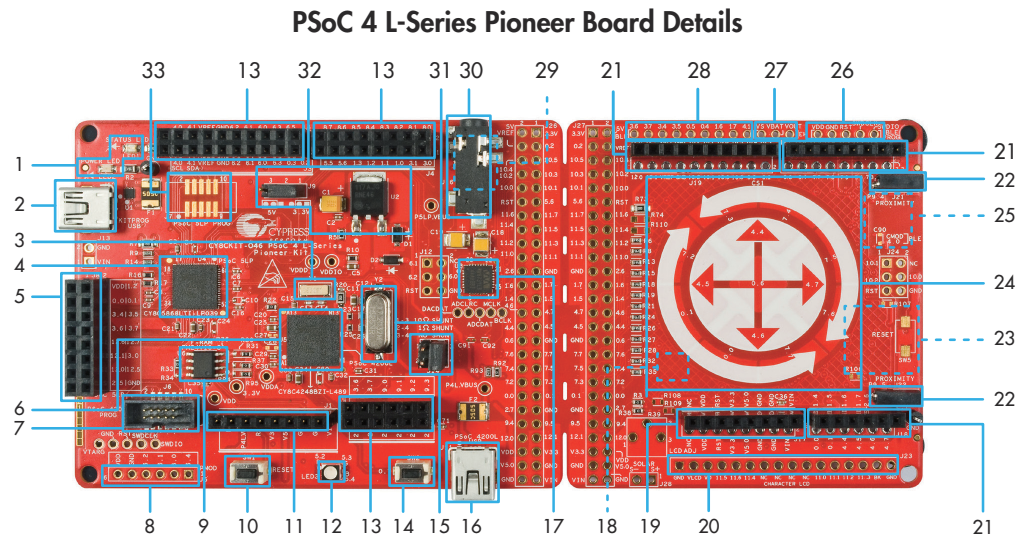
This kit has the following peripherals:

1. **Power (LED2) and Status (LED1) LEDs**: The amber power LED (LED2) is turned on when the kit is connected to a power source. The green status LED (LED1) indicates the status of the KitProg.

2. **KitProg USB Mini-B Connector (J10)**: The USB Mini-B connector J10 connects the KitProg USB to the PC. The USB cable provided along with the PSoC 4 L-Series Pioneer Kit can be used to connect the KitProg USB to the PC.

3. **32.768-kHz Crystal Oscillator (Y1)**: The PSoC 4 L-Series Pioneer Kit includes a 32.768-kHz watch crystal oscillator to provide an accurate low-frequency clock for the 4200L device. This clock sources peripherals such as real-time clock (RTC) and watchdog timer (WDT), which operate in the device's low-power modes.

4. **KitProg (PSoC 5LP) Programmer and Debugger (U4)**: The CY8C5868LTI-LP039 PSoC 5LP device (U4) serving as KitProg, is a multi-functional system, which includes a programmer, debugger, USB-I2C bridge, and a USB-UART bridge. Refer to the *KitProg User Guide* in the installation folder: `<Install_Directory>\CY8CKIT-046 PSoC 4 L-Series Pioneer Kit\<version>\Documentation\KitProg_User_Guide.pdf` for more details.

5. **KitProg (PSoC 5LP) I/O Header (J8)**: This header brings out a few input/output lines of the onboard PSoC 5LP device. This includes the KitProg USB-I2C bridge lines and USB-UART bridge lines. The additional PSoC 5LP pins are direct connections to the internal programmable analog logic of PSoC 5LP. You can also use these pins as GPIO. For more details on the KitProg functionality, refer to the *KitProg User Guide*. For details on the I/O header, see A.2.1.4 Expansion Connectors on page 62.

6. **PSoC 4200L (CY8C4248BZI-L489, U5)**: The PSoC 4200L device (CY8C4248BZI-L489) is the key device featured in the kit. This kit is designed to highlight the features of PSoC 4200L such as USB, CapSense, and more GPIOs. To learn more about PSoC 4200L, visit the PSoC 4 webpage.

7. **PSoC 4200L Program and Debug Header (J6)**: The 10-pin header allows you to program and debug the PSoC 4200L using an external programmer such as MiniProg3. The PSoC 4 L-Series Pioneer Kit also has through-hole test points to allow you to program and debug external PSoC 4 and PSoC 5LP devices using the KitProg. Remove the zero-ohm resistors R71, R70, R64, and R66 to disconnect the KitProg from the onboard PSoC 4200L and EZ-BLE devices before programming an external PSoC.

8. **Digilent Pmod-compatible Header (footprint only, J5)**: This header allows you to connect the PSoC 4200L device to Pmod peripheral modules through I2C and SPI interfaces.

9. **Cypress F-RAM (FM24V10-G, U6)**: This kit features a Cypress F-RAM device of 1 Mb capacity. The F-RAM is connected to the I2C interface of the PSoC 4200L device. The F-RAM device can be used similar to an external EEPROM memory for data logging operations. See A.3 Using FM24V10 F-RAM on page 75 for details on using the F-RAM.

10. **Reset Button (SW1)**: This button is used to reset the PSoC 4200L device. This button also serves as a reset button for the EZ-BLE device, if populated.

11. **Arduino Shield-compatible Main Board Power Header (J1)**: This header powers third-party Arduino shields. The header also has a provision to power the PSoC 4 L-Series Pioneer Kit though the VIN input on the header. The PSoC 4 L-Series Pionner Kit features two Arduino shield-compatible power headers: on the main board (J1) and shield board (J17).

12. **RGB LED (LED3)**: The onboard RGB LED can be controlled from the PSoC 4200L device.

13. **Arduino Shield-compatible Main Board I/O Headers (J2/J3/J4)**: The PSoC 4 L-Series Pioneer Kit is compatible and operates flawlessly with third-party Arduino shields. In addition, the PSoC 4 L-Series Pioneer Kit features two Arduino-compatible headers - one set of headers on the main board (J2/J3/J4) and another set of headers on the shield board (J18/J19/J20). This allows the kit to interface with two Arduino shields simultaneously.

14. **User Button (SW2)**: This button can be used to provide input to the PSoC 4200L. Note that the switch connects the PSoC 4200L pin to ground when pressed; therefore, you need to configure the PSoC 4200L pin as resistive pull-up for detecting the switch press. The switch is also connected to the PSoC 4200L wakeup pin. The wakeup pin can be used to wake up the PSoC 4200L device from the Stop power mode, the lowest power mode (20 nA) in the device.

15. **PSoC 4200L Current Measurement Jumper (J14)**: This jumper can be used to measure the current consumed by the PSoC 4200L device by connecting an ammeter between pins 3 and 4 of the jumper. Remember to connect the jumper between pin 3 and pin 4 if the current measurement functionality is not used.

16. **PSoC 4200L USB Mini-B Connector (J11)**: The USB Mini-B connector J11 connects the PSoC 4200L USB to the PC. This connector can be used to develop USB Full-Speed applications with PSoC 4200L.

17. **Audio codec (U3)**: The kit includes a Cirrus Logic® audio codec (WM8731CLSEFL) for converting the digital audio packets (via I2S) from PSoC 4200L to analog audio output and vice-versa. The codec supports stereo audio output and stereo audio input. The kit also includes a standard 3.5-mm audio jack to interface standard Tip-Ring-Ring-Sleeve (TRRS) headsets with a microphone.

18. **Cypress Energy Harvesting PMIC (footprint only, U12)**: The kit includes a footprint (U12) for S6AE101A, an Energy Harvesting PMIC from Cypress. The PMIC provides power to the EZ-BLE PRoC Module, U8. The chip can harvest solar power when connected to a solar cell through J28. Note that to use this chip, the EZ-BLE PRoC Module should be populated, L4/L3 should be removed, and L5/L6 should be populated. The EZ-BLE PRoC Module can alternatively be powered from a coin cell through connector V1.

19. **Arduino Shield-compatible Shield Board Power Header (J17)**: This is the Arduino shield-compatible power header on the shield board.

20. **Character LCD Header (footprint only, J23)**: The kit includes a footprint for a standard 2x16 character LCD header. The header is not populated by default.

21. **Arduino Shield-compatible Shield Board I/O Headers (J18/J19/J20)**: The odd pins (Pin 1, 3, 5, and so on) on headers J17/J18/J19/J20 provide footprint compatibility for Arduino shield connections. Along with the Arduino headers in the main board, this allows the kit to interface with two Arduino shields simultaneously. When the shield board is snapped, the even pins (Pin 2, 4, 6, and so on) on headers J17/J18/J19/J20 can be used to convert the shield board into an Arduino shield with CapSense Gesture Pad, EZ-BLE, and µSD card, or Serial NOR Flash. See section A.2.3.2 Using the Shield Board on page 72 for details on using the shield board as an Arduino shield.

22. **CapSense Proximity Headers (J22/J21)**: These headers are located next to the CapSense Gesture Pad. They can be used as individual proximity sensors or can be used together to detect CapSense proximity gestures. The code example, Proximity Gestures on page 34, included with this kit demonstrates the detection of wave up and wave down proximity gestures. Remember to connect wires into the proximity headers before using the proximity functionality. Note that if these inputs are not used for proximity sensing, then the connectors and the associated I/Os are available for general-purpose use.

23. **µSD Card Interface and Serial NOR Flash (footprint only)**: The kit includes a footprint (J25) for a standard µSD card interface that can communicate to PSoC 4200L over SPI. The µSD card interface shares the footprint and SPI connection with a Cypress SPI NOR Flash S25FL512S, U13. Only one of the two can be used on the board at any time.

24. **CapSense Gesture Pad**: The CapSense Gesture Pad is a five-element CapSense peripheral and an eight-element radial slider specifically intended for touch gesture recognition and joystick emulation. This peripheral can be used to detect swipe left, swipe right, swipe up, swipe down, clockwise swipe, and counter-clockwise swipe gestures. In addition to the gestures, the Gesture Pad can be used to implement an analog joystick. This implementation is discussed in USB Mouse on page 38.

25. **EZ-BLE PRoC Module (footprint only, U8)**: The kit includes a footprint for populating the Cypress EZ-BLE PRoC Module. The kit also supports mounting a CMOD capacitor required to develop CapSense applications using the EZ-BLE PRoC Module. The EZ-BLE I/O header (marked 28 in Figure 3-3 on page 21) provides access to the available EZ-BLE I/Os for development.

26. **EZ-BLE Programming Header (BLE PROG, footprint only):** The kit includes a footprint for a 5-pin EZ-BLE programming header. This header can be used to program the EZ-BLE PRoC Mod-

ule, when present, through an external SWD programmer such as the MiniProg3. When the shield board is snapped, the programming lines of the EZ-BLE PRoC Module to the KitProg cannot be restored through the 52-pin main board to shield board interface headers J26/J27. The BLE PROG header should be used in such cases to program the EZ-BLE PRoC Module.

27. **Energy Harvesting PMIC Header (EH, footprint only):** The kit provides access to the Energy Harvesting PMIC I/Os (VBAT, VSTORE, and VOUT) through the 3-pin header marked EH. In addition, the solar cell input the PMIC is connected to J28 (2-pin header). These headers are not populated by default.

28. **EZ-BLE I/O Header (BLE, footprint only):** The kit provides access to I/Os from the EZ-BLE PRoC Module through a 9-pin 100-mil header. This header can be used to develop applications with EZ-BLE PRoC Module on the shield board. The header is not populated by default.

29. **Audio Jack Selection Switch (SW4, footprint only)**: The DPDT switch SW4 is used to select the type of headset to be used with the kit. The two types of headsets supported are American Headphone Jack (AHJ) and Open Mobile Terminal Platform (OMTP). Refer to the code example USB Audio on page 43 for details on how to select the switch and identify the headset type. By default, the switch is not populated and the kit supports AHJ type of headsets (R111/R112 populated). Before populating the switch, remove R111/R112.

30. **Audio Jack - 3.5 mm (J16)**: The kit supports a standard 3.5-mm TRRS audio jack J16 for audio playback and recording. The audio signals to and from are interfaced to the onboard audio codec. The TRRS connector supports both AHJ and OMTP headsets. The user is required to select the headset type using the switch (SW4).

31. **17.2032-MHz External Crystal (Y2)**: The PSoC 4 L-Series Pioneer Kit includes a 17.2032-MHz external crystal to provide an accurate external crystal clock input for the PSoC 4200L device. This clock is used to provide an accurate clock for generating standard audio sampling rates.

32. **Power Supply**: The kit can operate at two voltage levels, 3.3 V and 5 V. The power selection jumper J9 is used to select between the two voltage levels. The kit has an onboard 3.3-V low dropout (LDO) regulator that converts the 5-V input to 3.3 V.

33. **10-pin Programming Header for PSoC 5LP (footprint only, J7)**: This header is used to program and debug the PSoC 5LP using an external programmer such as MiniProg3.

See A.2 Hardware Functional Description on page 58 for details on various hardware blocks. The subsequent sections explain the method to interface the PSoC 4 L-Series Pioneer Kit to a PC.

3.2   KitProg

The KitProg is a multi-functional system, which includes a programmer, debugger, USB-I2C bridge, and USB-UART bridge. The PSoC 5LP device is used to implement the KitProg functionality. The KitProg is integrated in most PSoC development kits. For more details, refer to the *KitProg User Guide* available on the kit webpage or at `<Install_Directory>\CY8CKIT-046 PSoC 4 L-Series Pioneer Kit\<version>\Documentation\`. The guide is also accessible from within PSoC Creator by clicking the kit name on the Start Page (left pane) and then selecting **User Guide for KitProg** from the list of available documents (right pane).

3.3   Programming and Debugging the PSoC 4200L Target Device

The target PSoC 4200L device can be programmed and debugged using the built-in KitProg. Before programming the device, ensure that PSoC Creator and PSoC Programmer are installed on your computer. See Install Software on page 17 for more information.

3.3.1   Programming using PSoC Creator

1. Connect the kit to the computer through the PCB USB connector, J10, as shown in Figure 3-4. The kit will enumerate as a composite device when you plug in the PSoC 4 L-Series Pioneer Kit to your computer for the first time. The kit drivers should be automatically installed.

    Note: The kit includes another USB connector (J11), which connects PSoC 4200L USB to the PC and is meant for developing USB Full-Speed applications with PSoC 4200L. For KitProg functionality, always use USB connector J10.

Figure 3-4. Connecting the PSoC 4 L-Series Pioneer Kit (KitProg USB) to a Computer
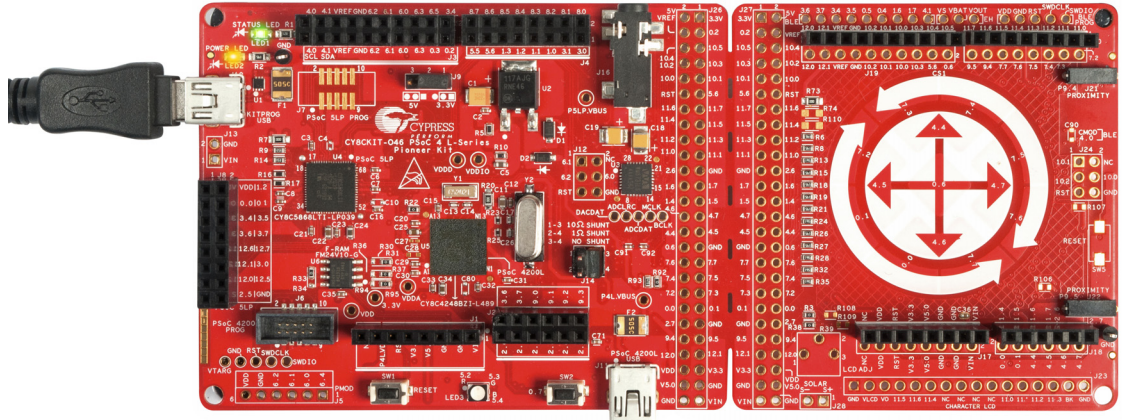
2. Open the desired Project or Workspace in PSoC Creator from **File** > **Open** > **Project/Workspace**. This provides the option to browse and open a previously saved Project or Workspace. If you want to open the code examples provided with the kit, follow the instructions in the Code Examples chapter on page 27.

3. Select **Build** > **Build <Project_Name>** or press **[Shift] [F6]** to build the Project as shown in Figure 3-5.

Figure 3-5. Build an Example
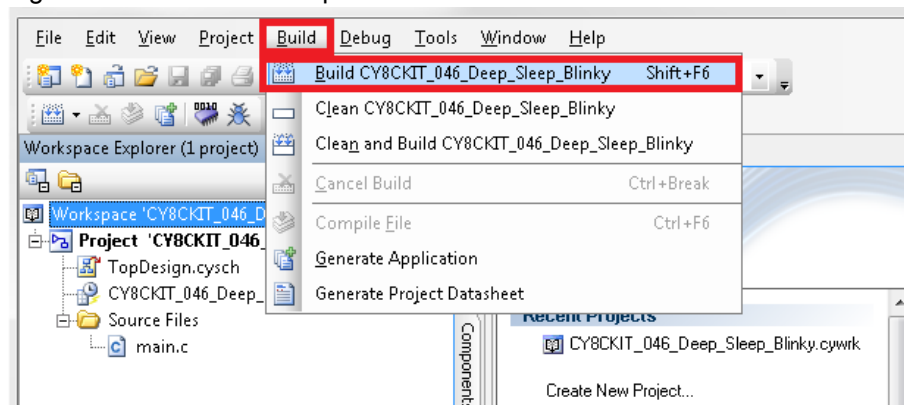
4. If there are no errors during build, program the firmware into the kit by choosing **Debug** > **Program** or press **[Ctrl] [F5]**, as shown in Figure 3-6. This programs the target PSoC 4200L device on the PSoC 4 L-Series Pioneer Kit; the kit is now ready for use.

Figure 3-6. Programming Device From PSoC Creator
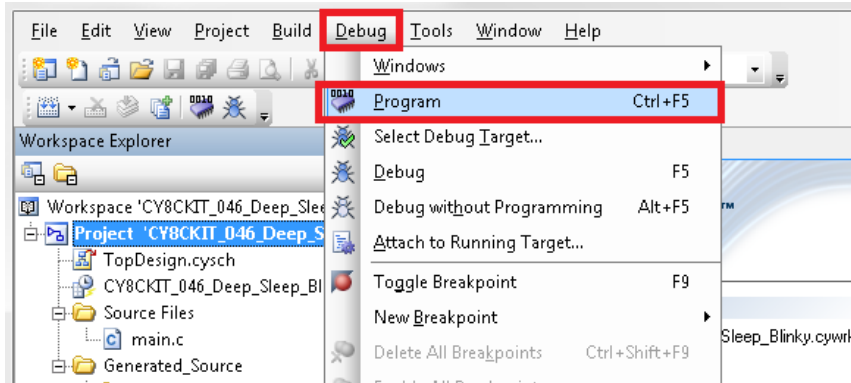
3.3.2 Debugging Using PSoC Creator

To debug the using PSoC Creator, follow steps 1 to 3 from Programming using PSoC Creator on page 25 followed by these steps:

1. Click the **Debug** icon or press **[F5]**. Alternatively, you can choose **Debug** > **Debug**.

2. When PSoC Creator opens in Debug mode, use the buttons on the toolbar to debug your application.

For more details on using the debug features, refer to section 3.2 of the *KitProg User Guide*.

3.3.3 Programming Using PSoC Programmer

PSoC Programmer (3.23.2 or later) can be used to program existing *.hex* files into the PSoC 4 L-Series Pioneer Kit. Refer to section 3.3 of the *KitProg User Guide* for a detailed explanation on how to program using PSoC Programmer.

3.4 USB-UART Bridge

The KitProg can act as a USB-UART bridge. The UART lines between the KitProg and the target are hardwired on the board with UART_RX assigned to P3[1] and UART_TX assigned to P3[0] on PSoC 4200L. For more details on the KitProg USB-UART functionality, refer to section 5 of the *KitProg User Guide*.

3.5 USB-I2C Bridge

The KitProg can function as a USB-I2C bridge and communicate with the Bridge Control Panel (BCP) software utility. The I2C lines on the PSoC 4200L device are P4[1] (SDA) and P4[0] (SCL), which are hardwired on the board to the I2C lines of the KitProg. The USB-I2C supports I2C speeds of 50 kHz, 100 kHz, 400 kHz, and 1 MHz. For more details on the KitProg USB-I2C functionality, refer to section 6 of the *KitProg User Guide*.

3.6 Updating the KitProg Firmware

The KitProg firmware normally does not require any update. You can use PSoC Programmer to update the KitProg firmware. Refer to section 3.5 of the *KitProg User Guide* for more details.

4.   Code Examples

The CY8CKIT-046 PSoC 4 L-Series Pioneer Kit includes five code examples. This chapter explains how to use these code examples. To access the code examples, download and install the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit setup file from www.cypress.com/CY8CKIT-046. After installation, the code examples will be available in the `Firmware` folder in the installation location.

For more code examples, visit the PSoC 3, PSoC 4, and PSoC 5 code examples page, which lists all PSoC Creator code examples available across application notes, kits, and PSoC Creator.

4.1   Using the Kit Code Examples

Follow these steps to open and use the code examples provided with the kit.

1. Launch PSoC Creator from **Start** > **All Programs** > **Cypress** > **PSoC Creator<version>** > **PSoC Creator <version>**.
2. On the **Start Page**, expand **CY8CKIT-046** under **Examples and Kits** > **Kits**. The code examples shipped with the kit appear (see Figure 4-1).
3. Click on the desired code example, select a location to save the example and click **OK**.

Figure 4-1. Open Code Example from PSoC Creator
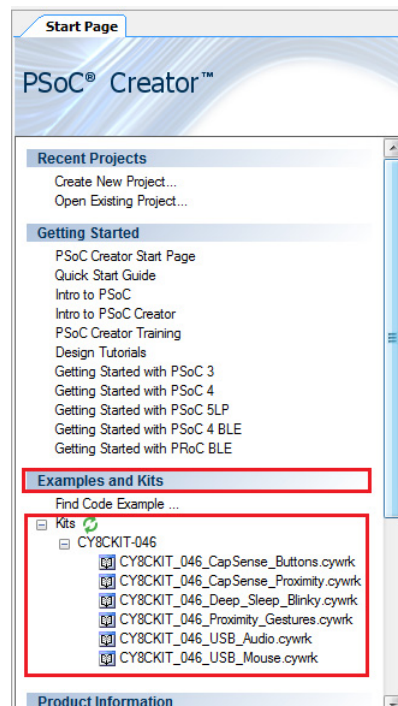
4. Build the code example by choosing **Build** > **Build <Project Name>**, as shown in Figure 4-2. The *CY8CKIT_046_Deep_Sleep_Blinky* code example is used as an example in the following images. A *.hex* file is generated after a successful build process.

Figure 4-2. Build Code Example from PSoC Creator
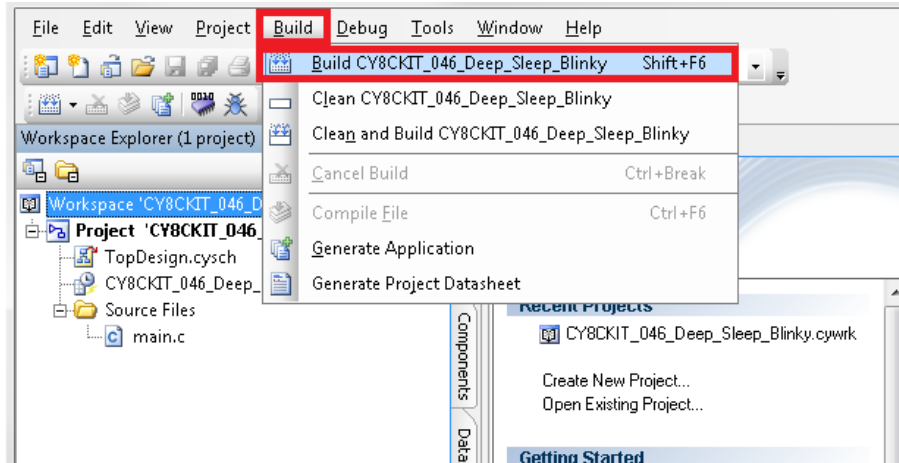
5. Connect the PSoC 4 L-Series Prototyping kit to the PC using the KitProg USB connector, J10, as described in Programming using PSoC Creator on page 25 to program the kit with this code example. If this is the first time the kit is connected, wait for drivers to be installed.

6. Choose **Debug** > **Program** in PSoC Creator as shown in Figure 4-3.

Figure 4-3. Program Device in PSoC Creator
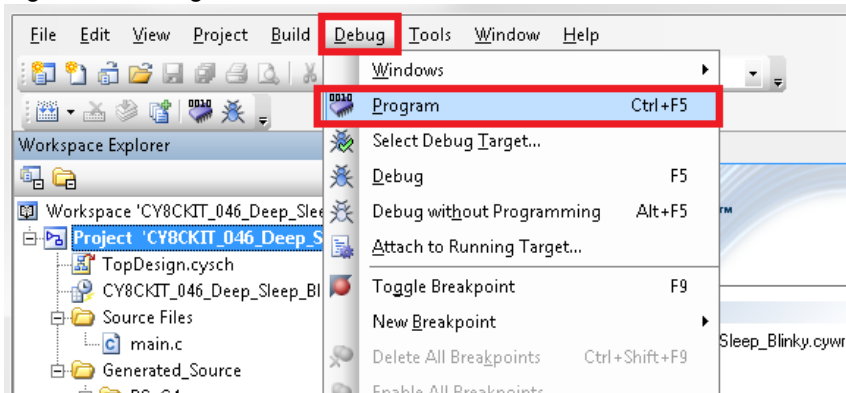
7. PSoC Creator opens the Select Debug Target window if the device is not already acquired. Select **KitProg/<serial_number>** and click the **Port Acquire** button, as shown in Figure 4-4.

Figure 4-4. Port Acquire
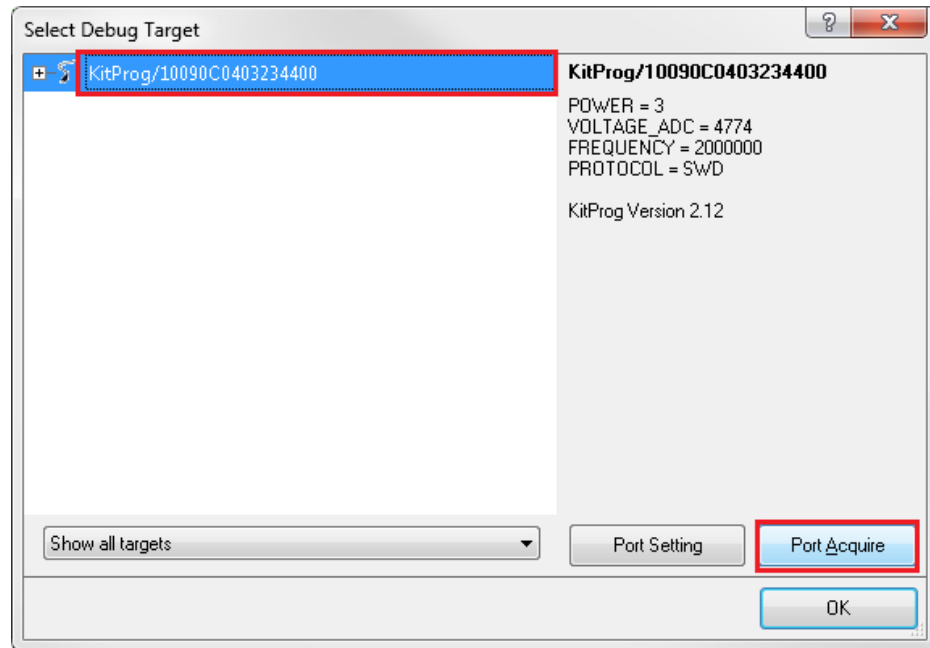
8. After the device is acquired, it is shown in a tree structure below the **KitProg/<serial_number>**. Click the **Connect** button and then **OK** to exit the window and start programming, as shown in Figure 4-5.

Figure 4-5. Connect Device From PSoC Creator and Program
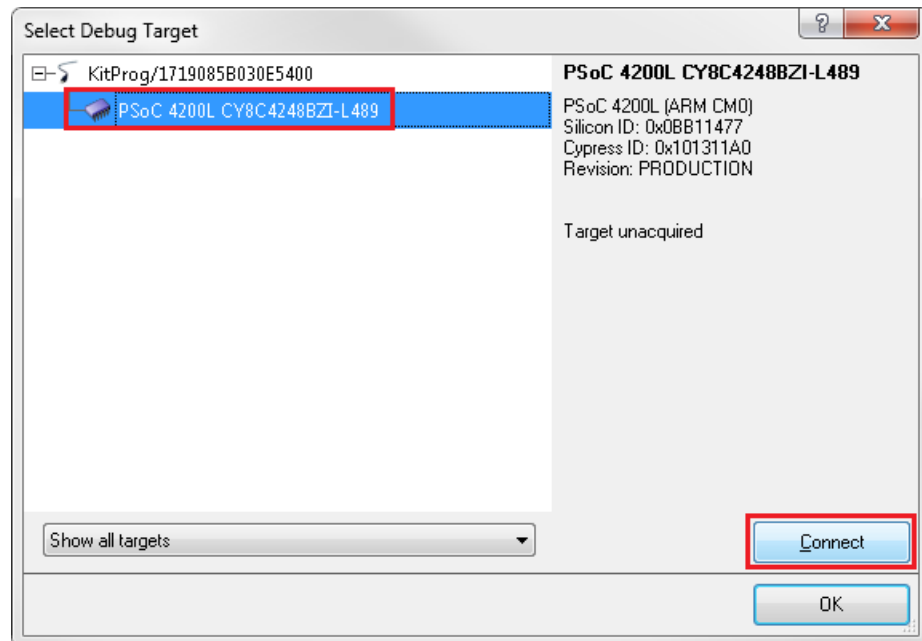
9. After programming is successful, the green LED (LED3) on the board should toggle at a rate of approximately 1 Hz.

4.1.1 Using CapSense Tuner

The CapSense examples (CapSense Proximity, Proximity Gestures, CapSense Buttons, and USB Mouse) include tuner support to tune and view CapSense signals. The CapSense tuner can also be used to measure the signal-to-noise ratio (SNR) for all CapSense elements in the code example. By default, the CapSense tuner is disabled in all examples. To enable tuner support, set the 'TUNER_ENABLE' macro present in *main.h* to 'ENABLED'. Note that enabling tuner in the example will disable all other functionalities and only CapSense tuner can be used in the example. This is done to isolate CapSense tuning in the examples. When CapSense tuner is enabled in the example, you can launch the tuner by navigating to **Tools** > **Component Tuners** > **1 CapSense**; see Figure 4-6. Alternatively, the CapSense tuner can be launched by right-clicking the CapSense component in TopDesign and selecting **Launch Tuner**. Refer to the "Manual Tuning Process" section in AN85951 - PSoC 4 CapSense Design Guide for details on how to use the CapSense tuner.

Figure 4-6. Launching CapSense Tuner
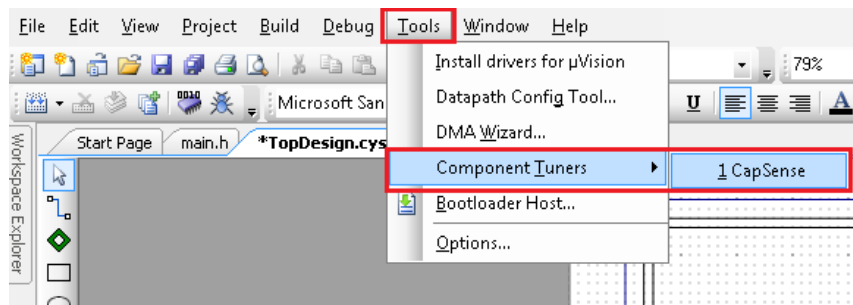
4.1.2 Using UART for Data Monitoring

The USB examples (USB Mouse and USB Audio) include optional data monitoring support over UART. The data sent over UART can be used to observe various USB actions and CapSense gestures in the example. The UART monitoring can be enabled or disabled by commenting or un-commenting the **#define TXDEBUG** line in the *main.h* (for the USB Mouse example) or *Config.h* (for the USB Audio example) files. Figure 4-7 and Figure 4-8 show the UART serial port settings and a sample output from the USB Audio example in Tera Term. The UART TX output is routed to P3[1] in PSoC 4200L, which in turn is connected to the KitProg's USB-UART bridge. The UART output is enabled by default in both the USB Audio and USB Mouse examples.

Figure 4-7. UART Settings (Tera Term)
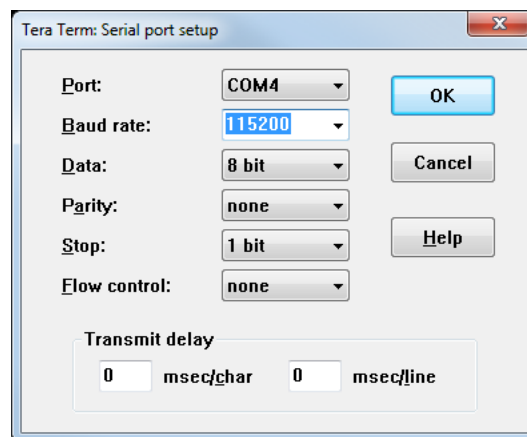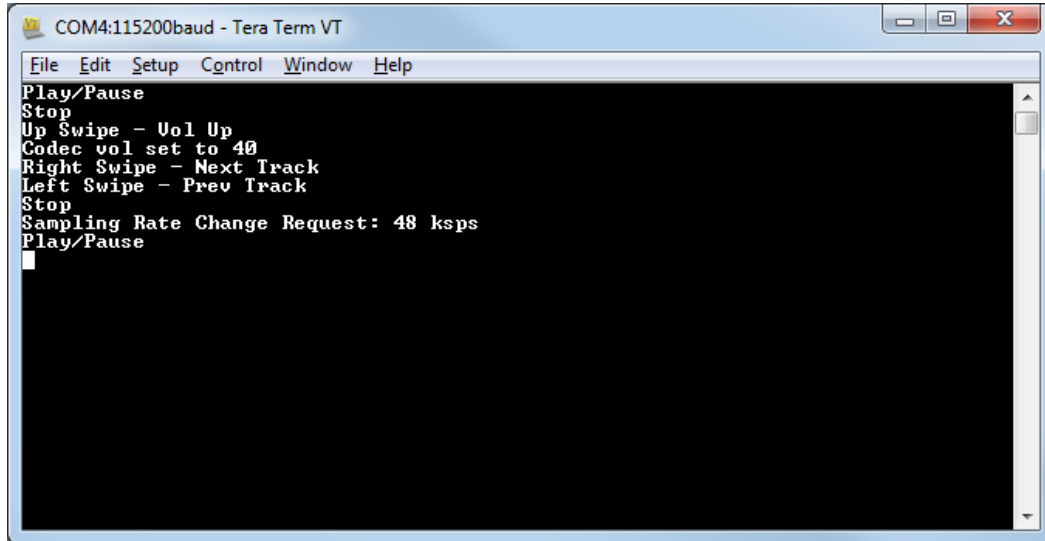
Figure 4-8. USB Audio Example - UART Output

4.2   DeepSleep Blinky

4.2.1   Project Description

This code example demonstrates the DeepSleep low-power mode of the PSoC 4200L device. The device is configured to be in DeepSleep mode and wakes up once every second. The PSoC 4200L toggles the GPIO state each time the device wakes up from DeepSleep mode. The GPIO is connected to the green LED on the kit, which turns on and off depending upon the GPIO state. The example also demonstrates a method to calibrate the ILO clock based on the IMO clock signal for better ILO accuracy. The calibrated LFCLK is used to source the watchdog timer.

4.2.2   Hardware Connections

No specific hardware connections are required for this example. All connections are hardwired on the board.

4.2.3   Verify Output

To verify the DeepSleep Blinky code example, follow these steps.

1. Power the PSoC 4 L-Series Pioneer Kit through USB connector J10.

2. Program the PSoC 4 L-Series Pioneer Kit with the *CY8CKIT_046_Deep_Sleep_Blinky* code example.

3. When programming is successful, you will see the green LED toggling once every second.

4. Power off the PSoC 4 L-Series Pioneer Kit by unplugging the USB cable. Remove the jumper shunt from the power measurement jumper of PSoC 4200L (J14).

5. Connect an ammeter across the terminals 3 and 4 of the power measurement header, as shown in Figure 4-9.

Figure 4-9. Current Measurement When Powered From USB Port
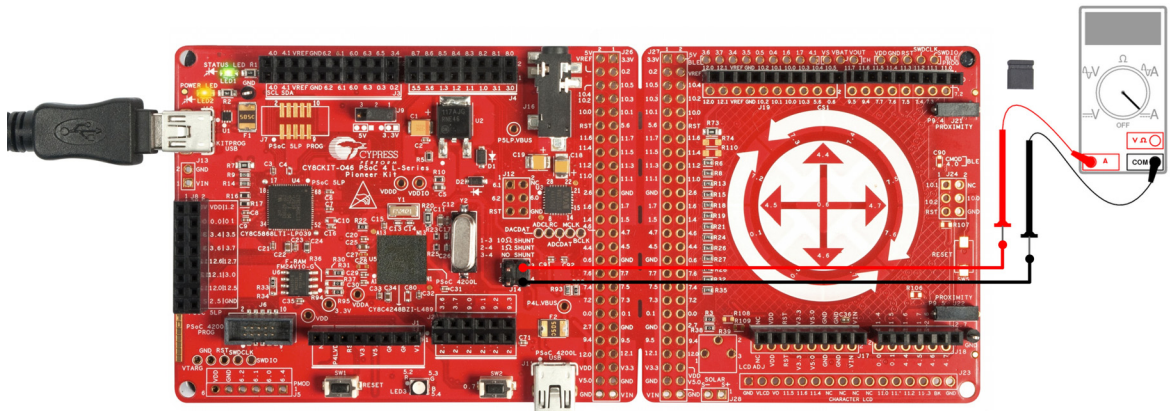
6. Plug in the USB cable again to power the PSoC 4 L-Series Pioneer Kit.

7. Observe the average power consumed by the PSoC 4200L device on the ammeter. The measured DeepSleep current consumption will be less than 3 µA (when the LED is OFF). The LED when ON consumes around 2.5 mA at 3.3 V. This results in an average current consumption of 1.25 mA at 3.3 V for the system including LED current.

Note that the Debug (SWD) port is disabled in the example to reduce power consumption during DeepSleep power mode. The Debug port can be enabled by setting the **Debug Select** option to **SWD** in the **System** tab of the *.cydwr* file, as shown in Figure 4-10. Disabling the debug port disables the ability to debug the code example through SWD.

Figure 4-10. Enabling Debug in a Code Example
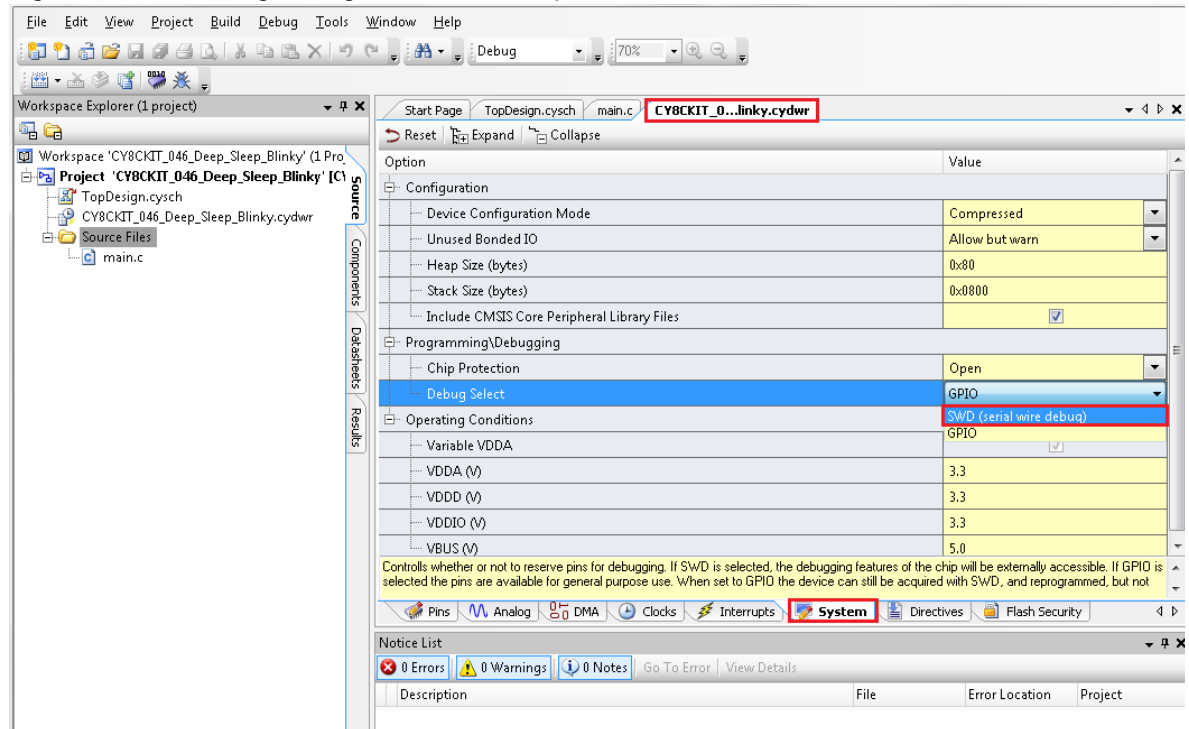
Note: More details on the power measurement header terminals is provided in .

4.3 CapSense Proximity

4.3.1 Project Description

This code example demonstrates the proximity sensing capability of the PSoC 4200L device. The CapSense scans for an approaching hand every 100 ms and enters a low-power mode when proximity is not detected. This allows the PSoC 4200L to operate at lower average power levels. The scanning interval is decreased such that the CapSense scan is performed every 30 ms when proximity is detected.

The PSoC 4200L gradually increases the brightness of the blue LED as a hand approaches the proximity sensor. If proximity is not detected for a period more than 3s, the PSoC 4200L increases the CapSense scanning interval back to 100 ms.

This example supports the CapSense tuner. For more details see Using CapSense Tuner on page 30.

Note: PSoC 4200L supports dual-channel CapSense and the proximity pins (P9[4] and P9[5]) are present on the second CapSense channel (CSD1). CMOD and CTANK (if used) pins should be selected appropriately. See A.2.1.13 CapSense Circuit on page 65 for details on the CMOD and CTANK pin connections for both the CapSense blocks.

Note: The time of 30 ms and 100 ms are dependent on LFCLK accuracy. This example uses the 32.768-kHz WCO in the kit as the LFCLK source. This offers a 250 ppm accuracy on the time. If the WCO is not used or available, the ILO can be used to source LFCLK. In the case of the ILO, the accuracy of the time can vary as much as 60 percent. Use the ILO trim method described in the example DeepSleep Blinky on page 31 to improve the ILO accuracy to 10 percent.

4.3.2 Hardware Connections

Connect a five-inch wire (the green wire provided with the kit) to proximity header J21 on the PSoC 4 L-Series Pioneer Kit, as shown in Figure 4-11.

Figure 4-11. Proximity Sensor Connection on PSoC 4 L-Series Pioneer Kit
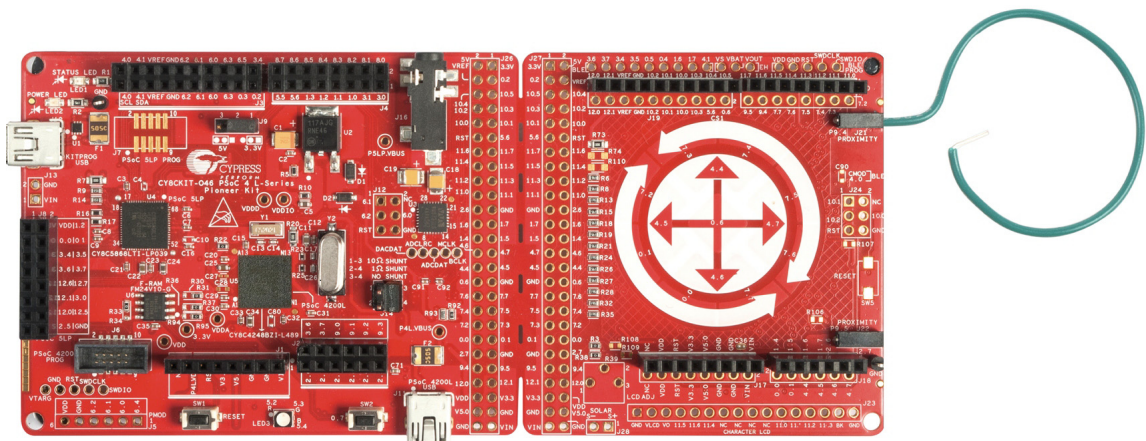
4.3.3 Verify Output

To verify the CapSense Proximity code example, follow these steps:

1. Connect the PSoC 4 L-Series Pioneer Kit to the PC through USB connector J10.

2. Program the PSoC 4 L-Series Pioneer Kit with the *CY8CKIT_046_CapSense_Proximity* code example.

3. Insert the proximity sensor wire to the proximity sensor header as shown in Figure 4-11. You should bend the wire to form a loop, as shown in Figure 4-11 to get better proximity range. Press the reset switch (SW1) when the wire is in place.

Figure 4-12. Hand Close to the Proximity Sensor
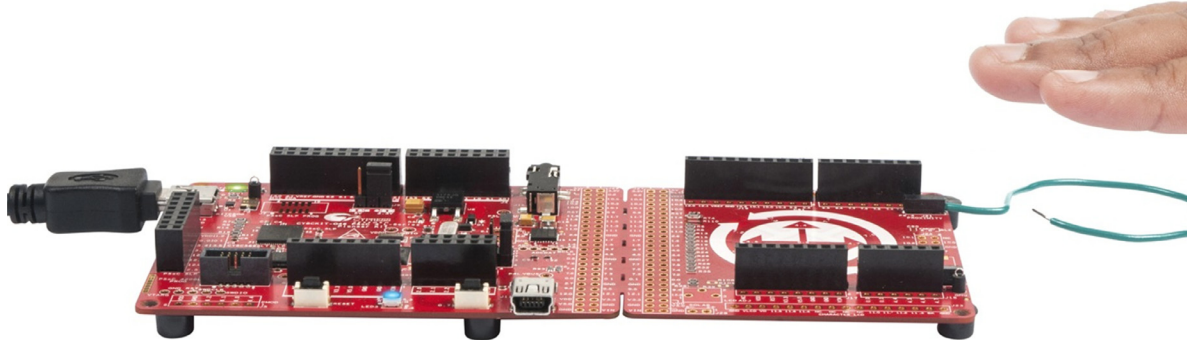
4. Bring the hand close to the proximity sensor, as shown in Figure 4-12. You will see the blue LED turning ON.

5. As you move your hand closer to the proximity sensor, the brightness of the LED increases. As you move your hand away from the sensor, the brightness of the LED decreases.

You will get a proximity range of approximately 5 cm with the wire supplied with this kit. You can achieve a higher proximity range by using a larger wire loop. Also, make sure there are no metal objects near the loop. To test the example and get a maximum proximity range, it is recommended to place the board at the edge of a table and connect wires such that the wire protrudes out from the table. This way, the proximity loop will be free of any interference or performance loss from nearby metals. This is done only for proximity detection using wires, because the proximity sensor is not shielded by any shield or ground signal. In general, proximity detection using PCB trace loops does not require this practice as they are properly shielded.

Power consumed by PSoC 4200L for this example can be measured using an ammeter, as described in DeepSleep Blinky on page 31.

4.4   Proximity Gestures

4.4.1   Project Description

This code example demonstrates the proximity gesture detection capability of the PSoC 4200L device using CapSense. The PSoC 4 L-Series Pioneer Kit has two headers, which are used to connect proximity wires. These wires are used as proximity sensors to detect an approaching hand.

The firmware gangs both proximity sensors and scans them as a single proximity sensor every 200 ms by default. If the ganged proximity sensor detects the presence of an approaching hand, then PSoC 4200L checks the magnitude of the signal produced by the proximity sensor. If the proximity signal is less than the threshold value, the RGB LED on the kit is maintained at a minimum brightness. If the proximity signal is greater than the threshold value, the brightness of the RGB LED is varied according to the magnitude of the proximity signal. The CapSense scanning interval is changed to 30 ms when proximity is detected. The example has a proximity range of approximately 5 cm.

When proximity is detected, the firmware checks for hand gestures. Two types of gestures are recognized by the firmware – Wave UP gesture and Wave DOWN gesture. Figure 4-13 and Figure 4-14 show the Wave UP and Wave DOWN gestures, respectively. The color of the RGB LED changes

based on the recognized gesture. The RGB LED turns OFF if proximity is not detected for more than 3 seconds. The CapSense scanning interval is changed to 200 ms when proximity is not detected.

This example supports the CapSense tuner. For more details see Using CapSense Tuner on page 30.

Note: The gesture is registered only when the hand moves completely from one sensor to the other through both the proximity sensors as shown in Figure 4-13 and Figure 4-14.

Note: The time of 30 ms and 200 ms are dependent on the LFCLK accuracy. This example uses the 32.768-kHz WCO in the kit as the LFCLK source. This offers a 250 ppm accuracy on the time. If the WCO is not used or available, the ILO can be used to source LFCLK. In the case of the ILO, the accuracy of the time can vary as much as 60 percent. Use the ILO trim method provided in the example DeepSleep Blinky on page 31 to improve the ILO accuracy to 10 percent.

Figure 4-13. Proximity Gesture Recognition with PSoC 4 L-Series Pioneer Kit (Wave UP)
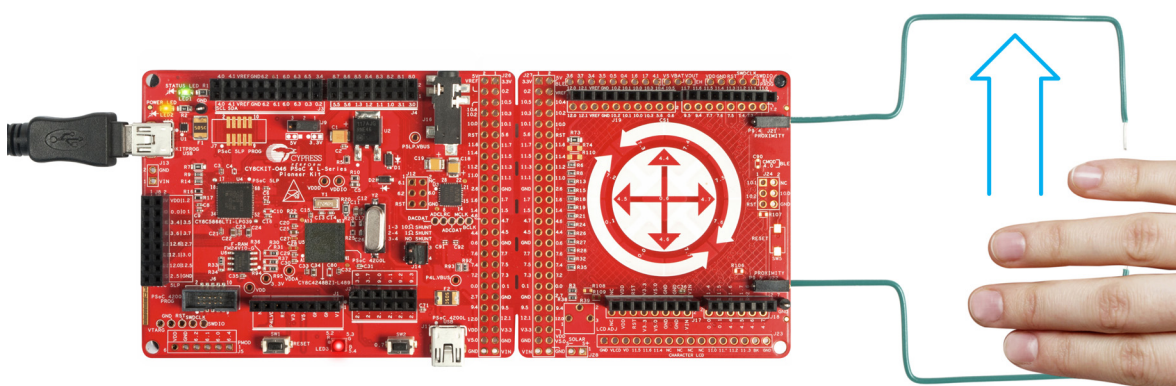
Figure 4-14. Proximity Gesture Recognition with PSoC 4 L-Series Pioneer Kit (Wave DOWN)
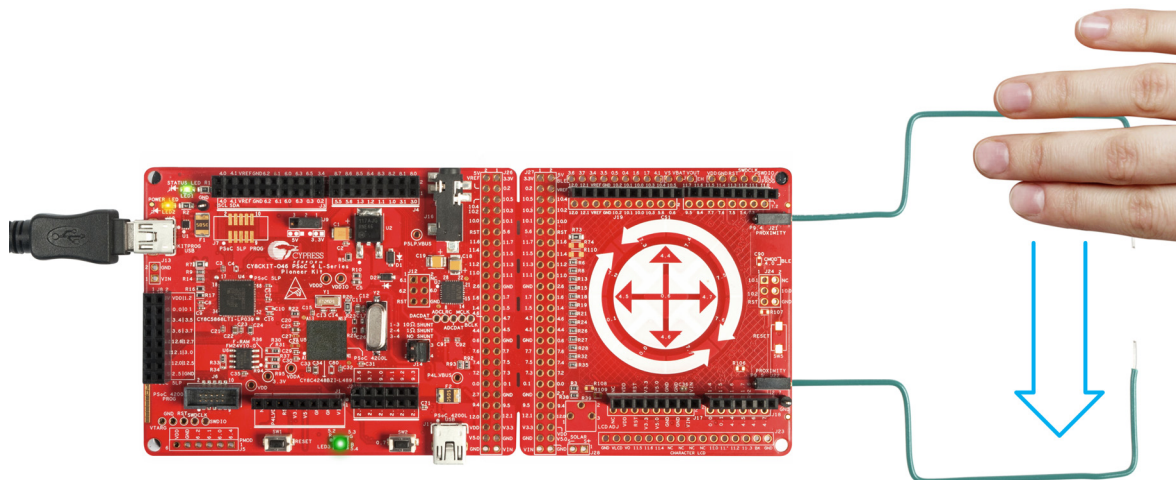
Table 4-1. Proximity Gestures and Actions

Gesture	Action (LED color change)[a]
Wave UP	Red > Green > Blue > Red
Wave DOWN	Red > Blue > Green > Red

a. Each Wave UP/DOWN gesture will trigger an LED color transition.

4.4.2　Hardware Connections

Connect five-inch wires (the green wires provided with the kit) to proximity headers J21 and J22 on the PSoC 4 L-Series Pioneer Kit, as shown in Figure 4-15.

Figure 4-15.　Proximity Sensor (Gestures) Connection on PSoC 4 L-Series Pioneer Kit
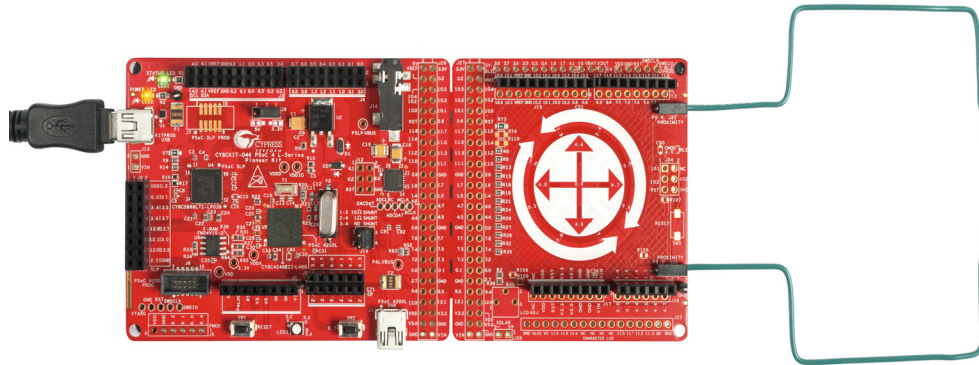
4.4.3　Verify Output

To verify the proximity gestures code example, follow these steps:

1. Connect the PSoC 4 L-Series Pioneer Kit to the PC through USB connector J10.
2. Program the PSoC 4 L-Series Pioneer Kit with the *CY8CKIT_046_Proximity_Gestures* code example.
3. Connect the proximity sensor wires as shown in Figure 4-15. You should bend the wires to form a rectangle, as shown in Figure 4-15 to get better proximity gesture detection. Press the reset switch (SW1) when the wires are in place.
4. Bring your hand close to the proximity sensors. The RGB LED will turn ON in red.
5. The brightness of the LED will increase as you bring your hand closer to the proximity sensors.
6. Wave the hand at a close proximity from the bottom sensor to the top sensor (Wave UP gesture) to see the LED color change in the order red > green > blue > red, one color at a time with each wave.
7. Wave the hand at a close proximity from the top sensor to the bottom sensor (Wave DOWN gesture) to see the LED color change in the order red > blue > green > red, one color at a time with each wave.

The power consumed by the PSoC 4200L device for this example can be measured using an ammeter as described in the DeepSleep Blinky on page 31.

4.5　CapSense Buttons

4.5.1　Project Description

This code example demonstrates basic dual-channel CapSense functionality in the PSoC 4 L-Series Pioneer Kit. The example implements five CapSense buttons using the CSD0 block and a proximity sensor using the CSD1 block. The buttons are used to control the brightness and color of the onboard RGB LED. The signal from the proximity sensor provides an additional brightness control factor. The LED brightness level is controlled by the buttons and the proximity signal provides a multiplication factor to the brightness output from the buttons. Figure 4-16 shows the CapSense Gesture

Pad pattern in the kit. The sensors marked 1 to 5 are used to implement the CapSense buttons. The functionality of each button is listed in Table 4-2.

This example implements dual-channel CapSense, enabling the CapSense buttons and CapSense proximity sensor to be scanned at the same time. This helps reduce the time the device spends in scanning all the sensors using a single-channel CapSense. As a result, the CapSense response time will be faster. In addition, when the device is put into the DeepSleep power mode between scans, the average power consumption of the system comes down. For details on dual-channel capacitive scanning in the PSoC 4 L-Series family, refer to AN85951 - PSoC 4 CapSense Design Guide.

This example supports the CapSense tuner. For more details, see Using CapSense Tuner on page 30.

Figure 4-16. CapSense Gesture Pad Pattern on the PSoC 4 L-Series Pioneer Kit

Pad	Pin
1	P0[6]
2	P4[5]
3	P4[4]
4	P4[7]
5	P4[6]
6	P7[4]
7	P7[5]
8	P7[6]
9	P7[7]
10	P0[0]
11	P0[1]
12	P7[2]
13	P7[3]
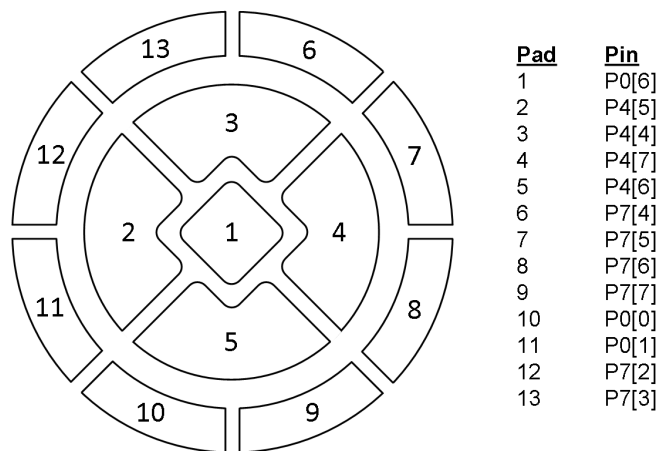
Table 4-2. CapSense Button Functions

Button/Sensor	Function
Center button (Sensor 1)	Turn ON/OFF the RGB LED
Left button (Sensor 2)	Cycles through red > green > blue > red colors on each tap
Right button (Sensor 4)	Cycles through red > blue > green > red colors on each tap
Up button (Sensor 3)	Increases the brightness of the LED by one step on each tap until maximum brightness level is reached
Down button (Sensor 5)	Decreases the brightness of the LED by one step on each tap until minimum brightness level is reached
Proximity sensor	Controls the brightness multiplication factor

4.5.2 Hardware Connections

Connect a five-inch wire (the green wire provided with the kit) to proximity header J21 on the PSoC 4 L-Series Pioneer Kit, as shown in Figure 4-11 on page 33.

4.5.3 Verify Output

To verify the CapSense Proximity code example, follow these steps:

1. Power the PSoC 4 L-Series Pioneer Kit through USB connector J10.

2. Program the PSoC 4 L-Series Pioneer Kit with the *CY8CKIT_046_CapSense_Buttons* code example.

3. Insert the proximity sensor wire to the proximity sensor header as shown in Figure 4-11 on page 33. You should bend the wire to form a loop, as shown in the figure to get better proximity range. Press the reset switch (SW1) when the wire is in place.

4. Tap the centre button to turn ON the RGB LED. By default, the LED should turn ON with Red color. Test the functions for each button captured in Table 4-2.

5. Bring your hand close to the proximity sensor, as shown in Figure 4-12 on page 34. As you move your hand closer to the proximity sensor, the brightness of the LED increases. The brightness decreases when you move your hand away from the sensor.

4.6 USB Mouse

4.6.1 Project Description

This code example demonstrates a simple USB human interface device (HID) implementation (mouse/keyboard) using the CapSense Gesture Pad present in the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit. In addition to emulating mouse/keyboard over USB, the example also controls the RGB LED intensity. Figure 4-16 shows the CapSense Gesture Pad pattern on the PSoC 4 L-Series Pioneer Kit.

Table 4-3 lists various mouse and keyboard actions associated with the gestures. USB is a complex protocol, and it can be difficult for beginners to get a USB-based application up and running quickly. Refer to the following documents to get started with USB and HID device class and to understand the concept presented in this example:

■ *AN57294 - USB 101: An Introduction to Universal Serial Bus 2.0* to get started with USB 2.0

■ *AN57473 - USB HID Basics with PSoC 3 and PSoC 5LP* to learn HID device basics

■ *AN58726 - USB HID Intermediate with PSoC 3 and PSoC 5LP* to learn intermediate HID concepts

■ Device Class Definition for HID v1.11 to learn about the HID class specification from USB-IF

■ HID Usage Tables v1.12 to learn about HID Usage tables - these tables provide additional details on the HID descriptor usage

This example supports the CapSense tuner. For more details see Using CapSense Tuner on page 30.

Table 4-3. CapSense Gestures and Actions
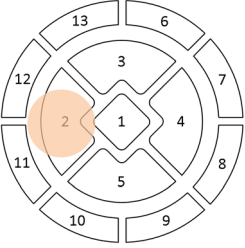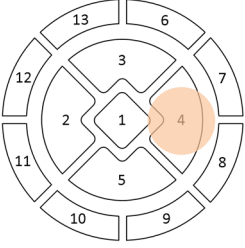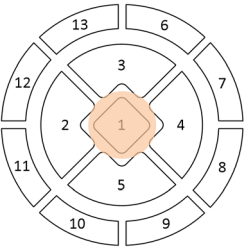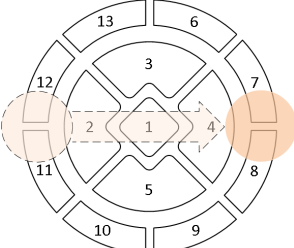
Gesture	Gesture Activation	Action (Mouse/Keyboard)	Action (RGB LED)
Left button tap[a]		Mouse left-click	None
Right button tap		Mouse right-click	None
Middle button tap[b]		Mouse center-click	Turn ON/OFF RGB LED
Right swipe[c]		CTRL+TAB	None

Table 4-3. CapSense Gestures and Actions
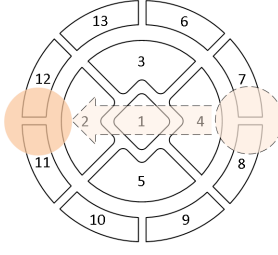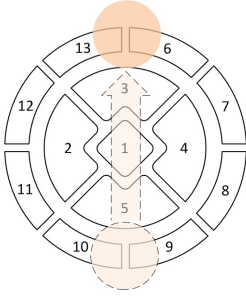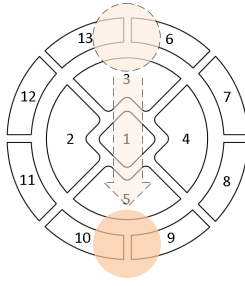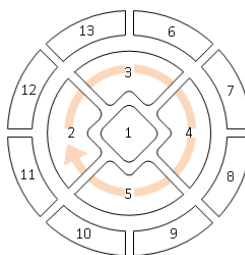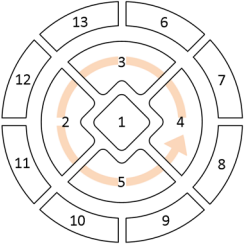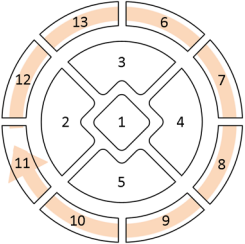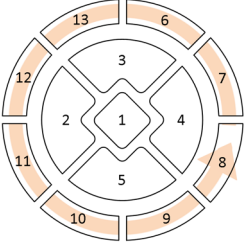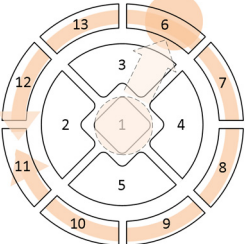
Gesture	Gesture Activation	Action (Mouse/Keyboard)	Action (RGB LED)
Left swipe		CTRL+SHIFT+TAB	None
Up swipe and hold		Scroll up	None
Down swipe and hold		Scroll down	None
Inner clockwise[d]		None	Change RGB LED color: Violet > Indigo > Blue > Green > Yellow > Orange > Red > Violet

Table 4-3. CapSense Gestures and Actions

Gesture	Gesture Activation	Action (Mouse/Keyboard)	Action (RGB LED)
Inner counter-clockwise[d]		None	Change RGB LED color: Red > Orange > Yellow > Green > Blue > Indigo > Violet > Red
Outer clockwise[d]		None	RGB LED brightness control: increase brightness
Outer counter-clockwise[d]		None	RGB LED brightness control: decrease brightness
Outward swipe and hold/rotate		Mouse pointer control - Moves the mouse pointer in the direction of the finger until the finger is released or removed. This action will move the mouse pointer similar to an analog joystick control.	None

a. In the example, a tap is defined as touch and release in less than one second.
b. Middle button tap is activated ONLY when the middle button (Sensor 1) alone is activated/touched.
c. The CapSense series resistors are close to the left side of the Gesture Pad. Therefore, while performing right/left swipe gestures, take care not to touch any of the resistors. Touching them may not detect or report the gestures properly.
d. Inner circular gestures and outer circular gestures are mutually exclusive; moving to the outer circle from the inner circle or vice-versa will NOT report any gestures until the finger is removed from the Gesture Pad and the gestures are tried again.

4.6.2 Hardware Connections

After programming the code example into the kit, connect the USB cable provided with the kit to J11 (PSoC 4200L USB header) on the board as shown in Figure 4-17.

Figure 4-17. USB Connection to J11 (PSoC 4200L USB Header)
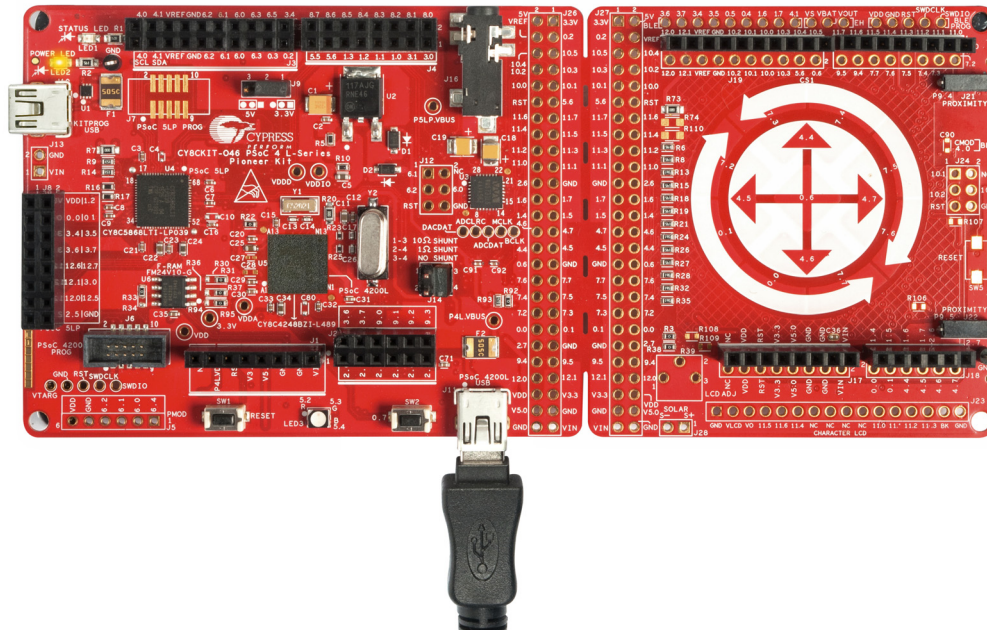
4.6.3 Verify Output

To verify the USB mouse code example, follow these steps:

1. Power the PSoC 4 L-Series Pioneer Kit through USB connector J10.

2. Program the PSoC 4 L-Series Pioneer Kit with the *CY8CKIT_046_USB_Mouse* code example.

3. Connect the PSoC 4200L USB to the PC by plugging the USB cable into the USB connector, J11, as shown in Figure 4-17. USB drivers should install automatically and the kit should enumerate as a USB Composite device as shown in Figure 4-18.

Figure 4-18. PSoC 4200L USB HID Enumeration
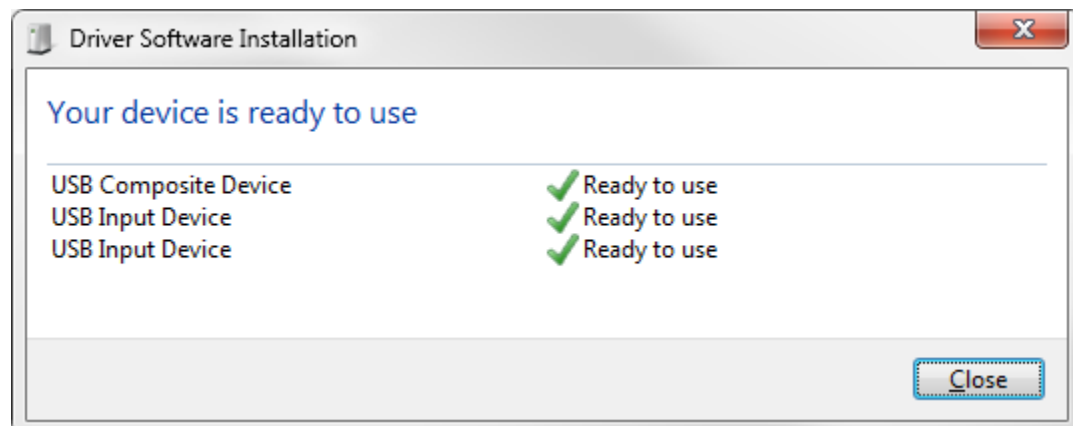
4. After successful driver installation (indicated by the blue LED turning ON), the CapSense Gesture Pad is available as a USB mouse/keyboard. The gestures captured in Table 4-3 can be performed to send the corresponding mouse/keyboard functionality to the PC.

5. In addition to USB mouse, the CapSense Gesture Pad can also be used to control RGB LED color and brightness. Table 4-3 (column 4) lists the supported RGB LED color control gestures.

4.7 USB Audio

4.7.1 Project Description

This code example demonstrates the capability of PSoC 4200L to provide a high-quality audio playback and recording interface to a Windows or Mac PC. The example uses the PSoC 4200L USB to implement an asynchronous USB Audio Class v1.0 compliant device. The PSoC 4 L-Series Pioneer Kit has a standard 3.5-mm TRRS audio jack onboard and an audio codec to convert digital audio stream into analog and vice-versa. These components along with PSoC 4200L implement the complete USB Audio interface.

In addition, the CapSense Gesture Pad is used to provide the volume and media control options required to control audio streaming. The example uses DMA to stream the audio data from USB (PC) to the onboard audio codec through an I2S interface. The codec supports stereo audio-out and stereo audio-in. However, the audio jack supports only mono audio-in. Therefore, the interface implements a single channel stereo audio-out with mono audio-in. Note that the USB Audio IN endpoint is implemented as stereo to make the implementation easy. The same audio-in data (mono data) is replicated on both the stereo channels in the USB Audio IN endpoint. Because the interface is implemented using the standard USB Audio device class, no special drivers are required to support hardware in Windows and Mac OS. To learn more about the USB Audio 1.0 device class, refer to the USB Audio devices class v1.0 specification.

The core of the implementation lies in the data transfer between the USB and I2S interface. The USB to I2S dataflow has multiple stages for streaming as both data rates are different. Audio data packets from the USB arrive once every millisecond whereas the I2S is a fixed stream of data at the configured audio rate. Any breaks in the streaming data from I2S will manifest either as audible pops in the audio or degradation in the quality of the audio. The PSoC implementation handles this such that the audio quality is maintained without any audible pops.

The example supports 24-bit stereo audio at 48 kHz and 44.1 kHz audio rates. This means the device supports two audio channels (stereo left and stereo right). Audio samples in each channel are 24 bits or 3 bytes wide and the number of stereo audio samples (6 bytes) transferred per second can be 48000 or 44100. For a stereo audio output, two channels are required - left and right. Because the audio codec supports stereo audio-in and stereo audio-out, the firmware implements the same – stereo audio-in and stereo audio-out. To generate the stereo audio-in from a mono audio microphone, the mono data is replicated on both channels. To transfer the stereo audio samples at 48 kHz (48000 samples per second), we need to send/receive 48 samples per USB transaction (per millisecond). Each sample contains two channel data and each channel's data is 3 bytes wide. As a result, to implement 48 kHz stereo audio over USB, we need to transfer 288 (48 × 2 × 3) bytes every USB transaction. The data transferred per transaction is referred as a packet. Similarly for the 44.1 kHz sample rate, we need to transfer 264 bytes for nine transactions and 270 bytes every tenth transaction (this averages out to 2646 sample every 10 ms, which is 44.1 kHz stereo audio).

To implement one stereo audio-out channel and one stereo audio-in channel, we will require two USB endpoints (refer to AN57294 - USB 101: An Introduction to Universal Serial Bus 2.0 for details on endpoints) – one for audio-out and one for audio-in. The buffer size of these USB endpoints is derived from the maximum audio rate supported and the type of audio device implemented (synchronous, asynchronous, or adaptive). The example implements an asynchronous audio device at a

maximum of 48 kHz audio rate. This defines the buffer size of each endpoint as 294 (288 + 6) bytes. The additional six bytes (or one audio sample) are for occasional asynchronous control, where the USB host may be required to transmit one additional sample to match the audio streaming rate. The example implements these two primary endpoints along with two control endpoints. One control endpoint communicates between the PC and PSoC 4200L, exchanging various audio controls such as volume and mute. The other control endpoint is an Isochronous Synch Endpoint (refer to section 3.7.2.2 in the USB Audio device class v1.0 specification for details). This endpoint is used to synchronize the audio streaming data rate to the USB transfer data rate. The basic function of this endpoint is to communicate the number of samples transferred to the audio device in an interval synchronized with the USB host clock (typically 1 ms for Full-Speed USB). With this data, the USB host can adjust the number of samples per packet it transfers in subsequent cycles.

The high level data flow is as follows:

1. Every 1 millisecond, transfer the entire USB data packet to a four-packet wide circular buffer with DMA and update the circular buffer write/read pointer.
2. If the circular buffer is at least half full (two packets full), start streaming data to and from the I2S component.
3. The I2S requests data from the circular buffer whenever space is available in the internal FIFO.
4. Every time one-eighth of the circular buffer has transferred to the I2S component, trigger an interrupt and update the circular buffer read/write pointer.

In addition to audio streaming, a USB HID device class is also implemented. The HID device class implements a Consumer Device interface to provide media controls and volume controls through the CapSense Gesture Pad. Note that the audio playback controls supported by the CapSense Gesture Pad are different from the ones supported over the audio control endpoint described earlier along with the audio streaming endpoints. The gestures supported in this example and the respective media control settings are listed in Table 4-4.

Table 4-4. CapSense Gestures and Media Control Actions
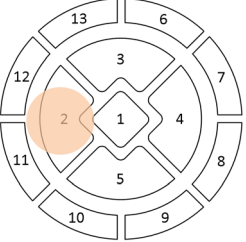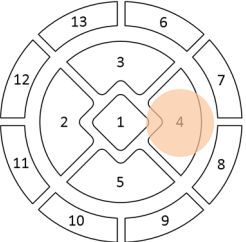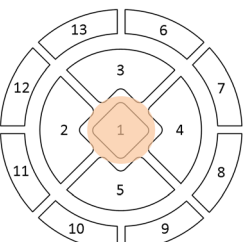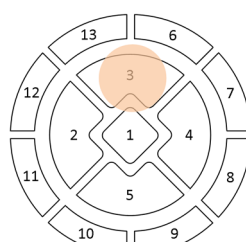
Gesture	Gesture Activation	Media Control
Left button tap		Previous track
Right button tap		Next track
Middle button tap		Play/Pause
Up button tap		Speaker mute

Table 4-4. CapSense Gestures and Media Control Actions
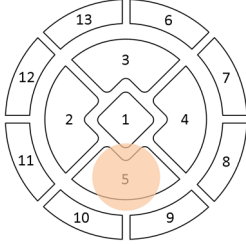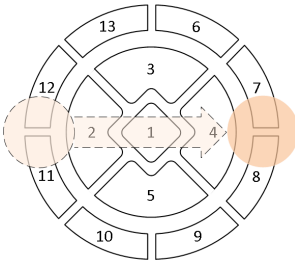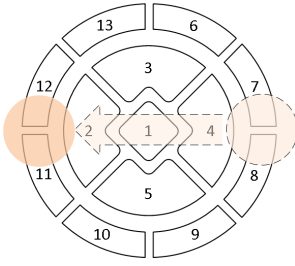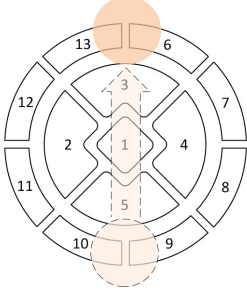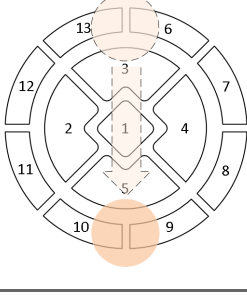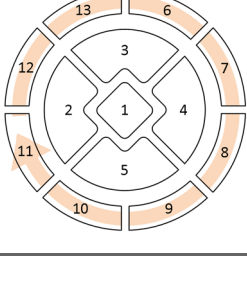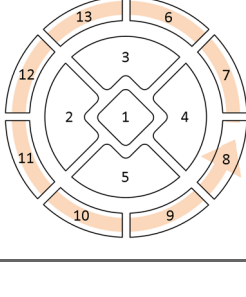
Gesture	Gesture Activation	Media Control
Down button tap		Stop
Right swipe		Next track
Left swipe		Previous track

Table 4-4. CapSense Gestures and Media Control Actions

Gesture	Gesture Activation	Media Control
Up swipe and hold		Volume +
Down swipe and hold		Volume –
Outer clockwise		Volume Increase (depending on the amount of rotation)
Outer counter clock-wise		Volume Decrease (depending on the amount of rotation)

4.7.2   Hardware Connections

This code example requires an external stereo audio device such as headphones, earphones, or speakers for testing. Optionally, the headphone can have a mono microphone output. The kit ships with a stereo audio earphone with microphone, which can be used to test the example. To test the example, perform the following hardware connections and settings.

1. After programming the code example into the kit, connect the USB cable to the PSoC 4200L USB header, J11.

2. Connect the 3.5-mm audio plug from the earphone to the audio jack header, J16, present on the in the kit. The setup is shown in Figure 4-19.

   Note: You can use your own headphone or speaker with the kit, provided it supports the standard 3.5-mm TRRS audio connector. The kit by default supports AHJ type connectors. A footprint for an audio plug selection switch (SW4) is provided on the bottom side of the board. This switch can be populated (after removing R111/R112) and is used to select the audio jack type. The AHJ and OMTP are the two audio jack types available in the market. The AHJ type is the most commonly used; OMTP are legacy headsets. Figure 4-20 shows the TRRS connector interface of both headsets. If you do not know which headset type you have, set the switch to **AHJ** position; if you do not hear a continuous clicking or static noise, then the headset is AHJ type. Set to **OMTP** position otherwise.

Figure 4-19. USB Audio Example Hardware Setup
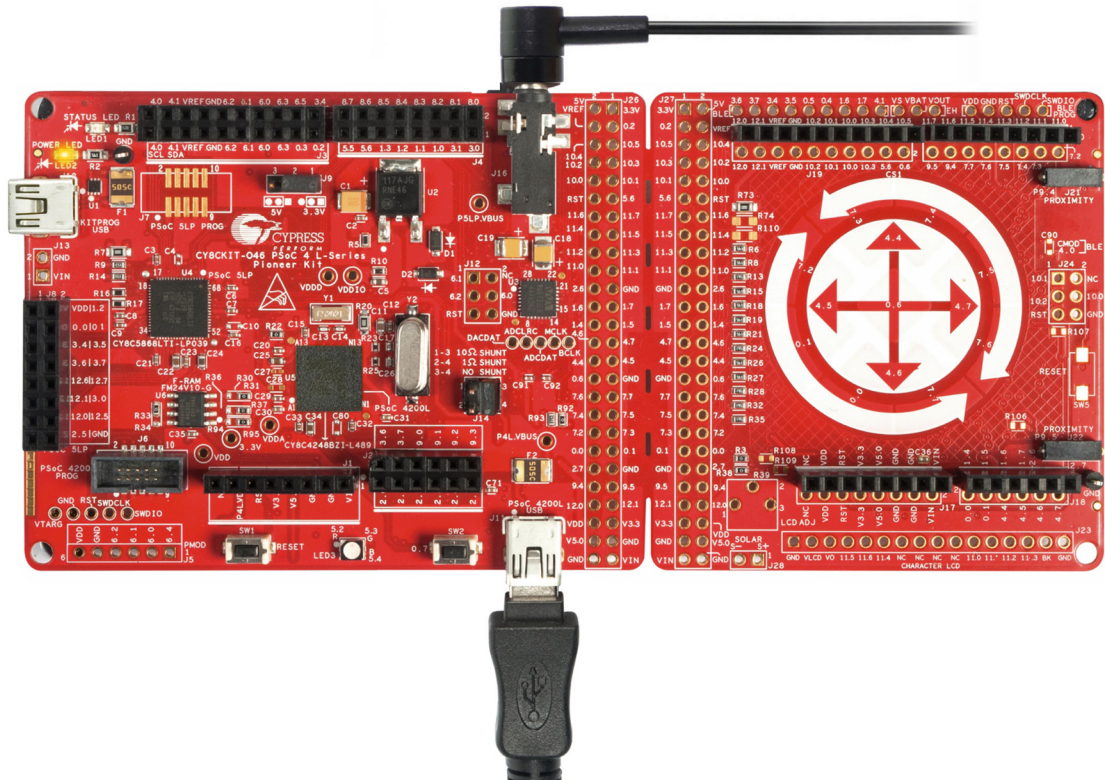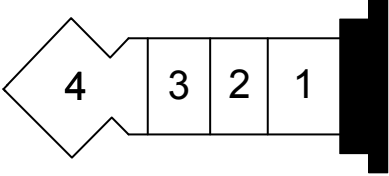
Figure 4-20. Audio Plug/Jack Types

OMTP Connector		AHJ Connector	
Pin	**Description**	**Pin**	**Description**
1 (S)	Ground	1 (S)	Microphone
2 (R)	Microphone	2 (R)	Ground
3 (R)	Right side earpiece	3 (R)	Right side earpiece
4 (T)	Left side earpiece	4 (T)	Left side earpiece

4.7.3   Verify Output

To verify the USB Audio code example, follow these steps:

1. Power the PSoC 4 L-Series Pioneer Kit through USB connector J10.

2. Program the PSoC 4 L-Series Pioneer Kit with the *CY8CKIT_046_USB_Audio* code example.

3. Connect the PSoC 4200L USB to the PC by plugging the USB cable into the USB connector, J11, as shown in Figure 4-19. The kit enumerates as a USB Composite Device (CY8CKIT-046 USB Audio and USB Input Device), as shown in Figure 4-21. Connect the 3.5 mm audio plug from the external audio device (earphone with microphone) to the audio jack, J16.

4. If populated, set the audio jack selection switch (SW4) to the appropriate position as explained in 4.7.2 Hardware Connections on page 47.

Figure 4-21. PSoC 4200L USB Audio Enumeration
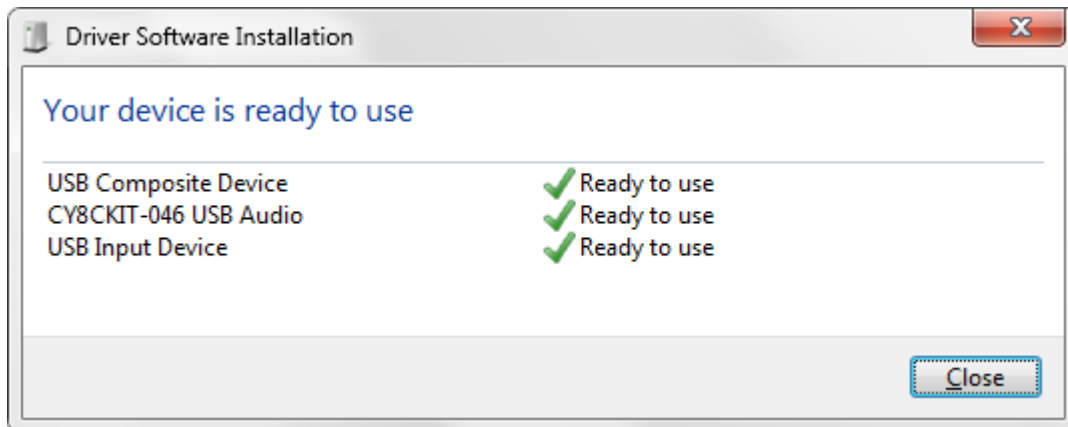
5. After successful driver installation, for stereo audio playback, go to **Start** > **Control Panel** > **Sound** and set **Speakers (CY8CKIT-046 USB Audio)** as default device under **Playback** tab, as shown in Figure 4-22. For audio recording, set **Microphone (CY8CKIT-046 USB Audio)** as the default device in the **Recording** tab, as shown in Figure 4-23.

Figure 4-22. Selecting CY8CKIT-046 USB Audio as Default Playback Device
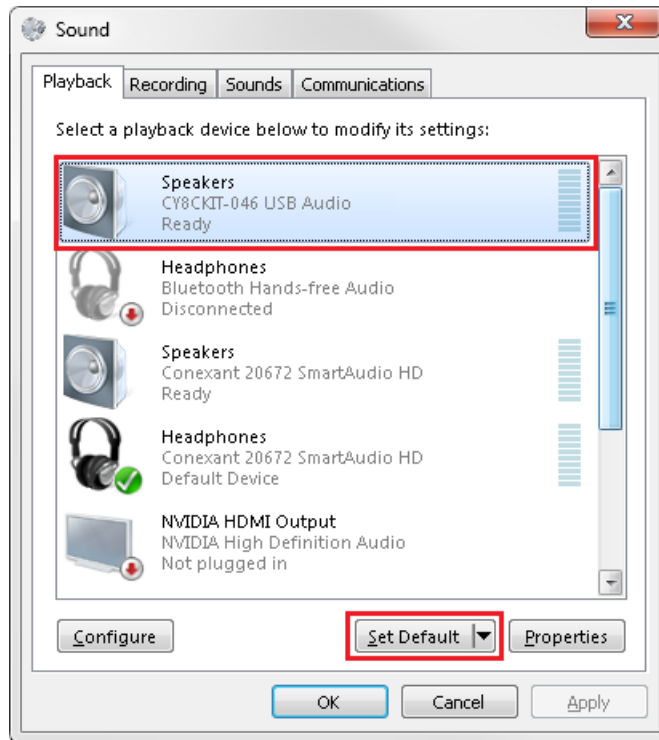
Figure 4-23. Selecting CY8CKIT-046 USB Audio as Default Recording Device
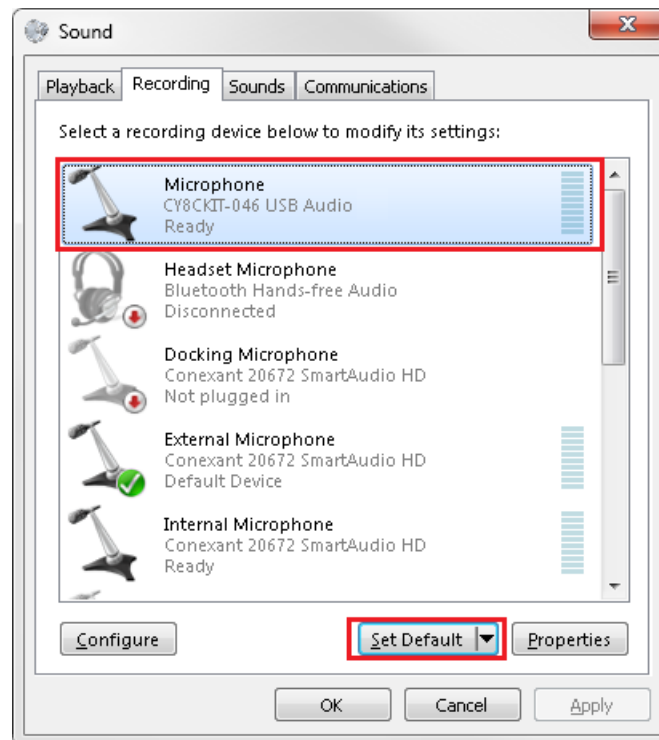
6. Any audio playback software can be used to stream audio to the external audio device. The audio sample rate of the device can be configured by selecting **Start** > **Control Panel** > **Sound** in the **Playback** tab, select **Speakers (CY8CKIT-046 USB Audio)**, and click **Properties**. See Figure 4-24. Similarly, any audio recording software can be used to read the microphone data and record the same. Note that the sample rate for speakers/headphones and microphone of the CY8CKIT-046 USB Audio device cannot be different. However, there is no way to synchronize the data in the **Playback** and **Recording** tab in Sound settings. Hence, when you update the sample rate in any tab of the CY8CKIT-046 USB Audio device, you need to manually update the same sample rate in the other tab as well for proper glitch-free streaming of audio between the kit and the PC.

Figure 4-24. Setting Audio Sample Rate for Speakers
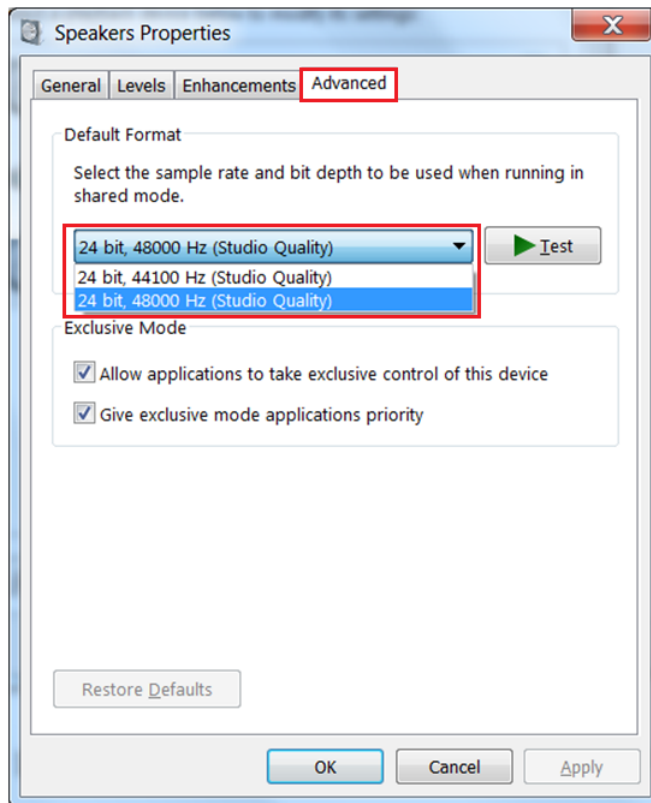
7. Use the gestures described in Table 4-4 to control the audio streaming and playback.

    Note that the volume control between the PC and the playback device depends on the operating system. This is because each operating system uses its own volume to decibel (dB) conversion logic. In general, the volume output from a PC is in the range of –127 dB to +127 dB. However, due to custom implementation, the minimum, maximum, and offset in the volume can be different between operating systems. For instance, Windows 7 sends volume output in the range of +50 dB to +127 dB. In Windows 10, the volume changes from –50 dB to +50 dB. Mac OS, however, uses the full range –127 dB to +127 dB. Because there is no easy way to detect the OS through the USB Audio interface, the VOLUME_CTRL macro, has been provided in the '*Config.h*' file. Set this macro to the appropriate OS to let the volume control module know, which OS will be used. If the OS used is not listed in the list of supported OS macros, use the 'MAC_VOLUME_CTL' macro.

A.   Appendix

A.1   PSoC 4 L-Series Pioneer Kit Schematics

Level translator

F-RAM

Device Select Address

The 8-pin SOIC footprint provided for the F-RAM FM24V10 on the PSoC 4 L-Series Pioneer Kit is compatible with all I2C-based F-RAM devices from Cypress (FM24Vxx, FM24CLxx, and CY15BxxxJ parts). The F-RAM parts with more than 64 KB size support only four addresses (four devices of the same type on the same I2C bus); resistors connected to A1 (R36/R37) and A2 (R30/ R31) pins can be used to select any of the four addresses. The F-RAM parts with less than 64 KB and FM24CLxx parts support eight addresses; resistors connected to A0 (R94/R95), A1 (R36/R37), and A2 (R30/R31) pins can be used to select one of the eight addresses

Audio Codec

Audio Jack Section

Default - AHJ, when R111 and R112 are populated.
For OMTP remove R111, R112 and populate SW4.

Place each pair of caps close to Power Pins

USB-UART

USB-I2C

USB-SPI

USB-SWD

PSoC 5LP and PSoC 4200L Connections

RGB LED

User Switch

Reset Switch

User Interface

Place this on main baord

Load these to make a common I2C bus

NO LOAD - TP33-TP37

PCBA: 121-60244-01
PCB: 600-60276-01
FAB DRW: 610-60265-01
ASSY DRW: 620-60273-01

CYPRESS SEMICONDUCTOR © 2015
Title: CY8CKIT-046 PSoC 4 L-Series Pioneer Kit
Size A3
Document Number: 630-60271-01
Rev 06
Date: Monday, October 12, 2015   Sheet 5 of 8

Character LCD

Level translator

Micro SD Card

SPI NOR Flash Memory

CapSense Gesture Pad

EZ- BLE and PSoC 4200L Connections

SWD line select

No Load; Remove R70, R71 before loading this

Reset Switch

No Load
EZ-BLE PRoC Module

Proximity Headers

Place all TP in 100 mil
spacing above J19 and J20
Arrange Prog line to match
with MiniProg3 5 pin header

CYPRESS SEMICONDUCTOR © 2015

CY8CKIT-046 PSoC 4 L-Series Pioneer Kit

Size A3	Document Number 630-60271-01		Rev 06
Date:	Monday, October 12, 2015	Sheet 6 of 8	

A.2   Hardware Functional Description

This section provides details on individual hardware blocks of the PSoC 4 L-Series Pioneer Kit. The complete schematic of the PSoC 4 L-Series Pioneer Kit is available on the kit webpage or the kit install directory. The kit board can be snapped into two parts (a main board and a shield board) for independent use, if desired. Both these parts are described separately here.

Figure A-1. CY8CKIT-046 PSoC 4 L-Series Pioneer Kit Board Markup

Main board   **Shield board**

1. Power (LED2) and Status (LED1) LEDs
2. KitProg USB Connector (J10)
3. 32.768 kHz crystal (Y1)
4. PSoC 5LP programmer and debugger (CY8C5868LTI-LP039, U4)
5. PSoC 5LP I/O header (J8)
6. PSoC 4200L (CY8C4248BZI-L489, U5)
7. PSoC 4200L program and debug header (J6)
8. Digilent® PMOD™ compatible I/O header (J5)*
9. Cypress F-RAM 1 Mb (FM24V10-G, U6)
10. PSoC 4200L reset button (SW1)
11. Arduino™ compatible main board power header (J1)
12. RGB LED (LED3)
13. Arduino compatible main board I/O headers (J2/J3/J4)
14. User button (SW2)
15. PSoC 4200L current measurement jumper (J14)
16. PSoC 4200L USB connector (J11)
17. Audio CODEC (WM8731CLSEFL, U3)
18. Arduino compatible shield board power header (J17)
19. Character LCD contrast adjustment potentiometer (R39)*
20. Character LCD header (J23)*
21. Arduino compatible shield board I/O headers (J18/J19/J20)
22. CapSense Proximity headers (J22/J21)
23. EZ-BLE PRoC Module reset switch (SW5)*
24. CapSense Gesture Pad
25. EZ-BLE programming header (BLE PROG)*
26. Energy Harvesting PMIC header (EH)*
27. EZ-BLE I/O header (BLE)*
28. Main board to shield board interface connectors (J26/J27)*
29. Audio jack - 3.5 mm connector (J16)
30. 17.2032 MHz crystal (Y2)
31. System power selection jumper (J9) and LDO (3.3V, U2)
32. PSoC 5LP program and debug header (J7)*
33. EZ-BLE PRoC Module (U8)*
34. µSD card interface (J25) and Cypress Serial NOR Flash (S25FL512S, U13)*
35. Level translators (U10/U11)
36. Coin cell battery holder (V1)*
37. Solar cell connector (J28)*
38. Cypress Energy Harvesting PMIC (S6AE101A, U12)*
39. EZ-BLE/PSoC 4200L Programming selection switch (SW3)*
40. Audio jack selection (AHJ/OMTP) switch (SW4)*

Footprint only, not populated on the board

A.2.1   Main Board

A.2.1.1   PSoC 4200L (U5)

This kit uses the PSoC 4200L (CY8C4248BZI-L489) device (marked 6 in Figure A-1). The PSoC 4 L-Series family is an expansion to the PSoC 4 device family that offers more flash, GPIOs, and programmable analog, digital, and communication blocks. The PSoC 4200L device is a true programmable embedded system-on chip, integrating custom analog and digital peripheral functions, memory, and an ARM Cortex-M0 microcontroller on a single chip. The programmable analog and digital peripheral functions allow higher flexibility, in-field tuning of the design, and faster time-to-market. The PSoC 4200L also adds a USB Full-Speed interface to the PSoC 4 family. The PSoC 4200L products are fully compatible with members of the PSoC 4 platform for new applications and design needs. For more details, visit the PSoC 4 webpage; also refer to the PSoC 4 L-Series family datasheet.

A.2.1.2   PSoC 5LP (U4)

The kit features an onboard PSoC 5LP (CY8C5868LTI-LP039) device (marked 4 in Figure A-1) as KitProg to program and debug the PSoC 4200L device. In addition, the PSoC 5LP device provides USB-I2C and USB-UART bridge support to the onboard devices and peripherals. The PSoC 5LP device can also be reprogrammed to provide other user functions. The PSoC 5LP connects to the

USB connector of a PC through a USB Mini-B connector. PSoC 5LP is a true system-level solution providing MCU, memory, analog, and digital peripheral functions in a single chip. The CY8C58LPxx family offers a modern method of signal acquisition, signal processing, and control with high accuracy, high bandwidth, and high flexibility. Analog capability spans the range from thermocouples (near DC voltages) to ultrasonic signals.

For more details on PSoC 5LP, visit the PSoC 5LP webpage; also, refer to the CY8C58LPxx family datasheet. For more details on KitProg functionality and features, refer to the *KitProg User Guide* available on the kit webpage.

A.2.1.3 Power System

The power supply system on this board is versatile, allowing input supply from the following sources:

- 5-V power (V5.0) from onboard USB connectors J10 or J11
- 5-V to 12-V power from Arduino shield or VIN header J13 (VIN)

The PSoC 4 L-Series Pioneer Kit can be powered at 5 V directly from the USB. An LDO (U2, marked 31 in Figure A-1) generates 3.3 V supply. The kit can be powered from either the 5 V supply or the 3.3 V supply. This selection is done through J9 jumper, as shown in Figure A-2. Note that 3.3 V must be selected if VIN is used to power the board and none of the USB connectors are connected to a PC.

Figure A-2. Power Supply Section
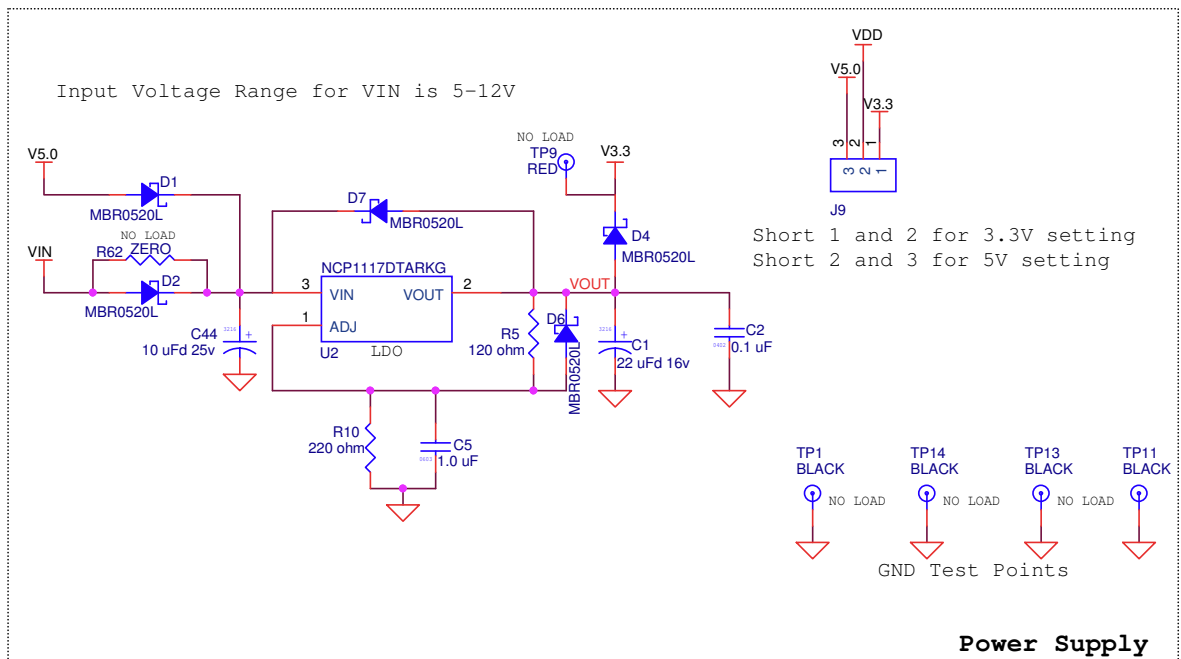
The input to the LDO can be from either the USB or the VIN pin in the Arduino header.

Protection Circuits

The power supply rail has reverse-voltage, overvoltage, short circuit, and excess current protection features, as shown in Figure A-3.

Figure A-3. Power Supply Block Diagram with Protection Circuits
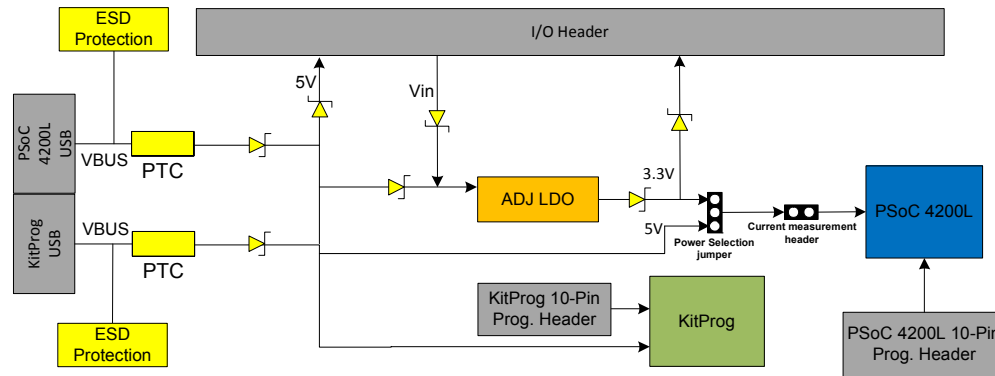
- A PTC resettable fuse is connected to protect the USB Mini-B connectors from shorts and over current.
- ORing diodes prevent damage to components when the board is powered from different voltage sources at the same time.
- ESD protection is provided for the USB Mini-B connectors.

Current Measurement Jumper

To measure the PSoC 4200L power consumption, a header is populated in series with the power supply to the PSoC 4200L. This header is used to measure current using an ammeter without the need to de-solder any components from the board (marked 15 in Figure A-1).

Figure A-4. Current Measurement Jumper Schematics
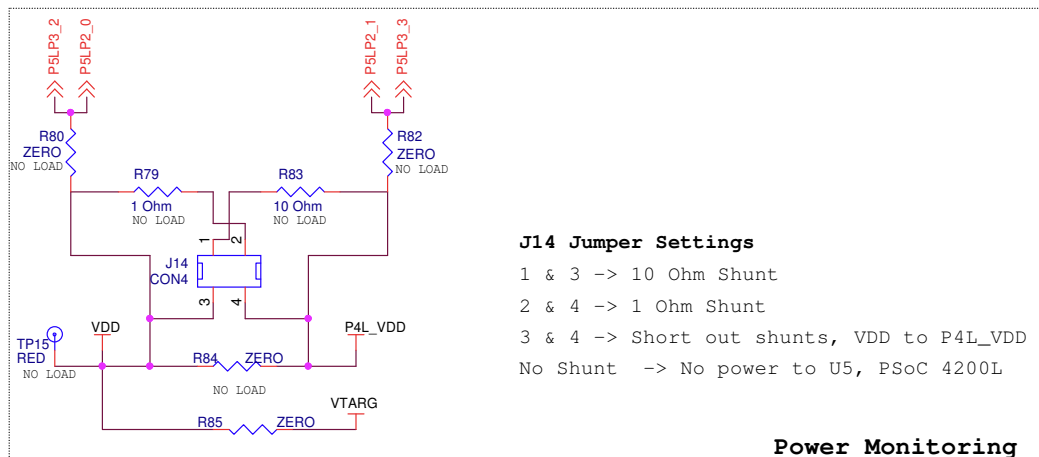
The current consumption of the PSoC 4200L device can be measured using the following methods:

1. When the board is powered through the USB Mini-B connector (J10 or J11), remove jumper J14 and connect an ammeter between terminals 3 and 4, as shown in Figure A-5.

Figure A-5. Current Measurement when Powered from USB Connector
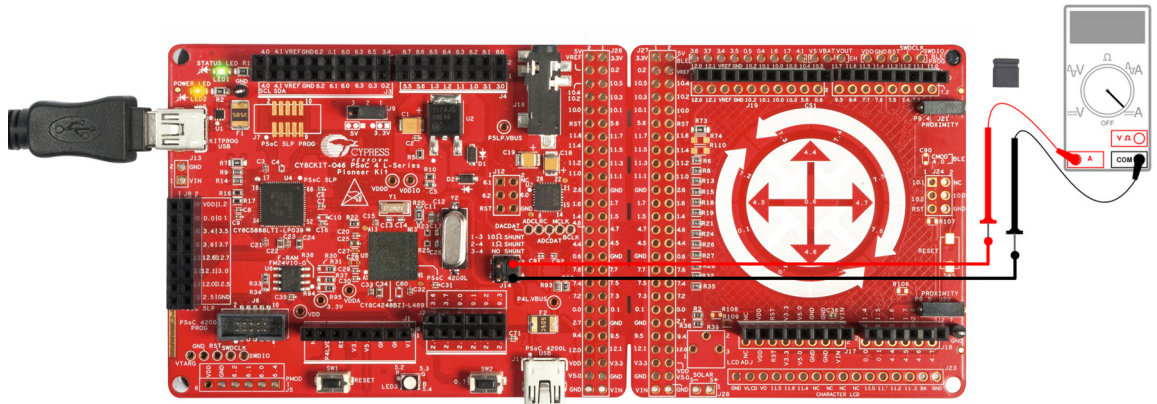
2. When the board is powered from an external voltage supply, remove the USB cable from J10 or J11. Connect the positive terminal of the external voltage supply to the positive terminal of the ammeter and the negative terminal of the ammeter to pin 4 of J14. Connect the negative terminal of the external voltage supply to a GND pin on the board. Figure A-6 shows the required connections.

Note that this powers the PSoC 4200L device directly. Using this method, you can operate the PSoC 4200L device over its complete VDD range (1.8 V to 5.5 V). However, it is not recommended to go below 3.3 V because other peripherals on the board are not designed to operate below 3.3 V.

Figure A-6. Current Measurement when Powered from External Source
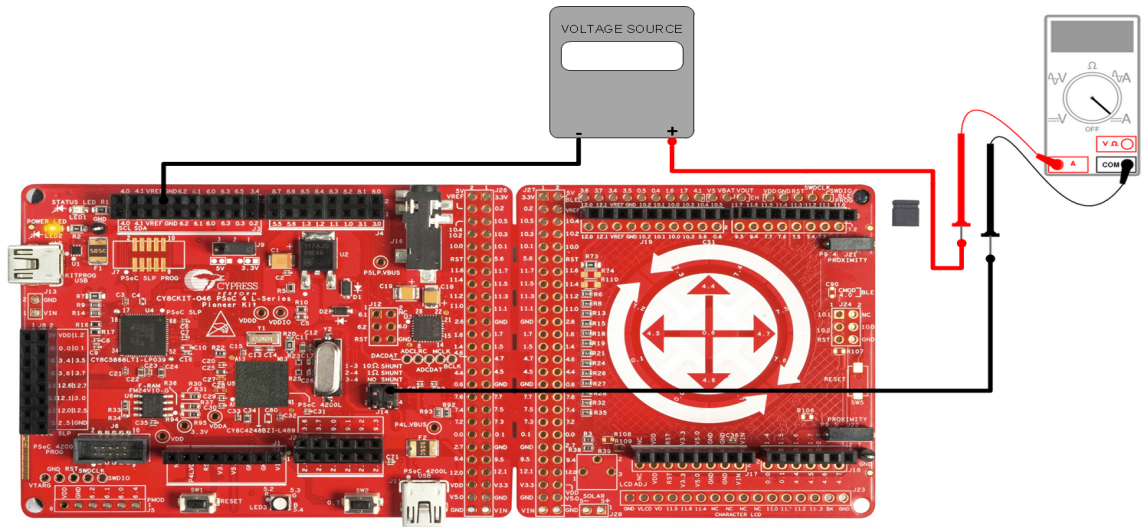
Note that the board includes multiple pull-up resistors and level translators interfacing PSoC 4200L to external peripherals such as F-RAM™, codec, and I2C slaves. As a result, removing the power jumper can potentially back power PSoC 4200L through the device I/Os interfacing these pull-ups and level translators operating on VDD or 3.3 V domain. Hence, it is recommended to use the board with the jumper present or with an ammeter connected across the jumper connections.

A.2.1.4 Expansion Connectors

Arduino-compatible Headers (J1, J2, J3, J4, and J12)

This kit has five Arduino-compatible headers on the main board–J1, J2, J3, J4, and J12. Header J1 is marked 11 and header J2, J3, and J4 are marked 13 in Figure A-1 on page 58. You can develop applications based on the Arduino shield's hardware.

The J1 header contains I/O pins for reset, I/O reference voltage (IOREF), and a power supply line. The J2 header is an analog port that contains I/O pins for SAR ADC, comparator, and opamp. The J3 header has I/O pins that support connection to the PSoC 4200L communication blocks. The J3 header is primarily a digital port that contains I/O pins for PWM, I2C, SPI, and analog reference. The J4 header is also a digital port that contains I/O pins for UART and PWM. The J12 header is an Arduino ICSP-compatible header for the SPI interface and is not populated. Refer to the "No load components" section of "Bill of Materials" on page 86 for the header part number.

Additional Functionality of Header J2, J3, J4

The J2 header is a 6×2 header that supports Arduino shields. Similarly, headers J3 and J4 are 10x2 and 8x2 headers, respectively. On the non-Arduino side (even pins - Pin 2, 4, 6, and so on) of the header, pins from PSoC 4200L are brought to these headers for general-purpose use. See "Onboard Peripherals" on page 83 for details about onboard peripheral connections to the pins on the header.

Functionality of Header J12 (unpopulated)

The J12 header is a 2×3 header that supports Arduino shields. This header is used on a small sub-set of shields and is unpopulated on the PSoC 4 L-Series Pioneer Kit.

Pmod Connector J5 – Digilent Pmod-compatible (unpopulated)

This port supports Digilent Pmod peripheral modules (marked 8 in Figure A-1 on page 58). Pmods are small I/O interfaces that connect with embedded control boards through either 6-pin or 12-pin connectors. The PSoC 4 L-Series Pioneer Kit supports the 6-pin Pmod type 2 (SPI) interface. For more details about Pmods, go to www.digilentinc.com.

This header is not populated on the PSoC 4 L-Series Pioneer board. You must populate this header before connecting Pmod peripheral modules. Refer to the "No load components" section of "Bill of Materials" on page 86 for the header part number.

PSoC 5LP GPIO Header (J8)

An 8x2 header is provided on the board to bring out several pins of PSoC 5LP to support advanced features such as a low-speed oscilloscope and a low-speed digital logic analyzer (marked 5 in Figure A-1 on page 58). This header also contains the USB-UART bridge pins and USB-I2C bridge pins, which can be used when these pins are not accessible on the Arduino headers because of a shield connection.

A.2.1.5 KitProg USB Mini-B Connector (J10)

The KitProg (PSoC 5LP) connects to the PC USB connector through the Mini-B connector J10 (marked 2 in Figure A-1 on page 58), which can also be used to power the PSoC 4 L-Series Pioneer Kit. A resettable poly fuse is used to protect the computer's USB port from shorts and over current. If more than 500 mA is drawn from the USB Mini-B connector, the fuse will automatically break the connection until the short or overload is removed.

A.2.1.6 LEDs

The main board has three LEDs. A green LED (LED1) indicates the status of the KitProg. See the Troubleshooting section in the *KitProg User Guide* for more information on LED states. An amber LED (LED2) indicates the status of power supplied to the board. The kit also has a general-purpose RGB LED (LED3) connected to the PSoC 4200L device for user applications (marked 12 in Figure A-1 on page 58).

A.2.1.7 Push Buttons

The main board contains a reset push button SW1 and a user push button SW2 (marked 10 and 14 in Figure A-1 on page 58). The reset push button is connected to the XRES pin of PSoC 4200L and EZ-BLE PRoC Module, and is used to reset the device. The user push button is connected to P0[7] of the PSoC 4200L device. Both the push buttons connect to ground on activation (active low).

A.2.1.8 Cypress Ferroelectric RAM (F-RAM, U6)

The PSoC 4 L-Series Pioneer Kit contains an F-RAM device (FM24V10-G) (marked 9 in Figure A-1 on page 58) that can be accessed through I2C lines P4[0] and P4[1] of the PSoC 4200L device. The F-RAM has a capacity of 1-Mb (128 KB) with I2C speeds up to 3.4 Mbps. The I2C slave address of the F-RAM device is 7 bits wide, and the second but last and third but last bits are configurable through physical pins. These pins are hardwired to 00 on the board. By default, the address of the F-RAM device used on the kit is 0x50. This address can be modified by changing the R36/R37 (A1) and R30/R31 (A2) resistor pairs. See A.3 Using FM24V10 F-RAM on page 75 for details. The operating voltage range of the F-RAM is between 2 V and 3.6 V. To prevent the F-RAM from operating at voltages greater than 3.6 V, the power supply to the F-RAM is derived from the output of the 3.3 V regulator. The F-RAM I2C lines are connected to the 3.3 V side of an onboard level translator (U11) to help F-RAM communicate with PSoC 4200L operating at 5 V.

A.2.1.9 Serial Interconnection Between PSoC 5LP and PSoC 4200L

In addition to being a programmer/debugger, the KitProg (PSoC 5LP) acts as a USB-Serial interface for the USB-UART and USB-I2C bridges, as shown in Figure A-7. The I2C bus contains firmware-controlled resistive pull-ups using FETs, which can be enabled or disabled using the PSoC 5LP pins. The USB-Serial pins of the PSoC 5LP are also available on the Arduino header; therefore, the PSoC 5LP can also be used to control Arduino shields with an SPI/I2C/UART interface.

Note: The KitProg does not support USB-SPI bridge functionality. The SPI pins of the PSoC 4200L are connected to the PSoC 5LP for enabling SPI communication between PSoC 5LP and PSoC 4200L for custom PSoC 5LP applications. By default, the SPI connections are not enabled on the board. You need to populate R61, R53, R59, and R68 (shown in Figure A-7) to enable these connections.

Figure A-7. KitProg and PSoC 4200L Connections
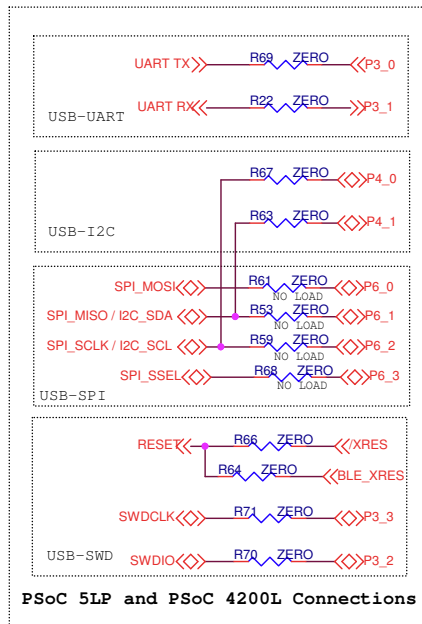
A.2.1.10 PSoC 4200L USB Connector J11

The PSoC 4200L connects to the PC USB through a Mini-B connector J11 (marked 16 in Figure A-1 on page 58), which can also be used to power the PSoC 4 L-Series Pioneer Kit. A resettable poly fuse is used to protect the computer's USB port from shorts and over current. If more than 500 mA is drawn from the USB Mini-B connector, the fuse will automatically break the connection until the short or overload is removed. This connector is used for developing USB-based applications with PSoC 4200L.

A.2.1.11 Audio Playback Circuit

The kit includes a WM8731 series audio codec chip (U3) from Cirrus Logic (marked 17 in Figure A-1 on page 58). The WM8731 is a low-power stereo codec with an integrated headphone driver. The WM8731CLSEFL device is used in the kit for stereo audio playback and mono audio recording. Refer to the codec datasheet for details on the codec specification. See the code example "USB Audio" on page 43 for a demonstration of audio streaming using the PSoC 4200L USB. On the analog audio input/output side, the codec connects to a standard 3.5-mm TRRS audio jack (marked 29 in Figure A-1 on page 58). On the digital audio input/output side, the codec connects to PSoC 4200L over I2C (control commands) and I2S (audio data). The codec operating voltage is from 2.7 V to 3.6 V. Therefore, the codec is configured to always operate at 3.3 V. A level translator, U11, is present on the board to translate the signals between PSoC 4200L (running at 5 V) and the codec (running at 3.3 V). The connections between the codec and PSoC 4200L are listed in Table A-1.

Table A-1. PSoC 4200L and Audio Codec Connections

Codec Connection	PSoC 4200L Connection	Description
SDA	P4[1]	I2C data line
SCL	P4[0]	I2C clock line
ADCDAT	P3[7]	Audio (ADC) digital data output (microphone) from codec
DACDAT	P3[6]	Audio (DAC) digital data input (speakers) to codec

Table A-1. PSoC 4200L and Audio Codec Connections

Codec Connection	PSoC 4200L Connection	Description
ADCLRC	P3[5]	ADC sample rate left/right clock input to codec
DACLRC	P3[5]	DAC sample rate left/right clock input to codec
BCLK	P3[4]	Digital audio bit clock input to codec
MCLK	P5[7]	Master clock input to codec

A.2.1.12   External Crystals

The main board includes a 32.768-kHz (Y1) external crystal for the WCO input and a 17.2032-MHz (Y2) crystal for the ECO input (marked as 3 and 30 in Figure A-1 on page 58). The WCO is used to provide an accurate low-frequency clock to PSoC 4200L for DeepSleep wake up intervals, WDT reset intervals, and RTC tick interval. The 17.2032-MHz ECO is used to provide an accurate high-frequency clock to PSoC 4200L. In addition, the 17.2032-MHz ECO serves as a source to generate the various I2S clock frequencies required for supporting the standard audio sample rates of 32 kHz, 44.1 kHz, and 48 kHz for audio playback and streaming applications.

A.2.1.13   CapSense Circuit

The CMOD and CTANK capacitors essential for CapSense functionality are connected on the main board. PSoC 4200L supports dual-channel capacitive sensing (CSD0 and CSD1 CapSense blocks). This requires two pairs of CMOD and CTANK capacitors, one pair for each block. The 2.2-nF capacitors are present on the CMOD pins, P4[2] for CSD0 and P5[0] for CSD1, for CapSense operation. A 10-nF capacitor is present on the CTANK pin of CSD0 block, P4[3]. The CTANK capacitor of CSD1 block is not populated by default and can be populated if required, on pin P5[1]. The CTANK capacitors help improve the shield (waterproofing) performance. For details on using CapSense, design guidelines and dual-channel capacitive sensing, refer to AN85951 - PSoC 4 CapSense Design Guide.

This kit also supports CapSense designs that enable waterproofing. On this kit, the shield signal is connected to a GPIO pin (P0[2]) or ground by resistors R74 and R73, respectively. By default, R73 is populated on the board, which connects the shield signal to ground. Remove R73 and populate R74 when evaluating waterproofing designs, which will connect the shield to the designated pin, P0[2]. To successfully evaluate waterproof designs, the same should be enabled in the PSoC Creator CapSense Component and the shield signal should be routed to P0[2]. Note that this shield signal is connected to the CSD0 block. Hence, it cannot be used with the CSD1 block operation. In the shield board of the PSoC 4 L-Series Pioneer Kit, only the proximity sensors are routed to the CSD1 block while all other CapSense elements (radial slider and Gesture Pad) are connected to CSD0.

The shield board also supports shield signal connection to P5[6] through resistor R110. This resistor should be populated instead of R74 when the shield board is snapped and used as an Arduino shield and a CapSense shield signal is required. Populating R110 brings the CapSense shield connection to one of the Arduino headers when plugged into a baseboard. For the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit, the shield signal will be connected to P0[3], when the snapped shield board is connected to CY8CKIT-046 Arduino headers.

Figure A-8. CapSense Circuit
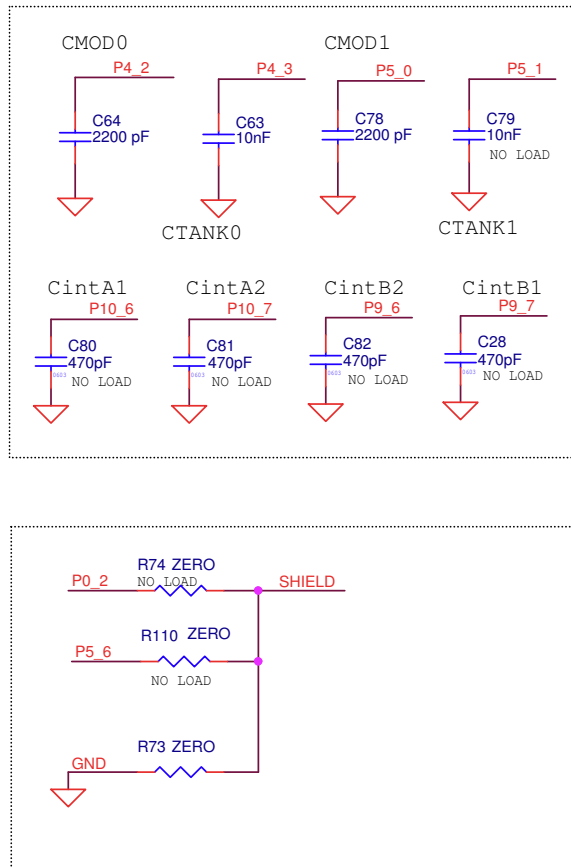
A.2.1.14 *Level Translators (U11 and U10)*

The PSoC 4 L-Series Pioneer Kit has two level translators (marked 35 in): one on the main board (U11) and one on the shield board (U10). The level translators are used to translate digital 5 V logic signals to 3.3 V. The level translator on the main board, U11, translates the I2C and audio interface (I2S/clock) signals from PSoC 4200L to 3.3 V, to be used with the F-RAM and audio codec, as shown in Figure A-9.

The level translator on the shield board, U10, translates the SPI signals from PSoC 4200L to 3.3 V level to be used with µSD card or Serial NOR Flash interface. This is shown in Figure A-9.

Figure A-9. Level Translators in CY8CKIT-046 PSoC 4 L-Series Pioneer Kit
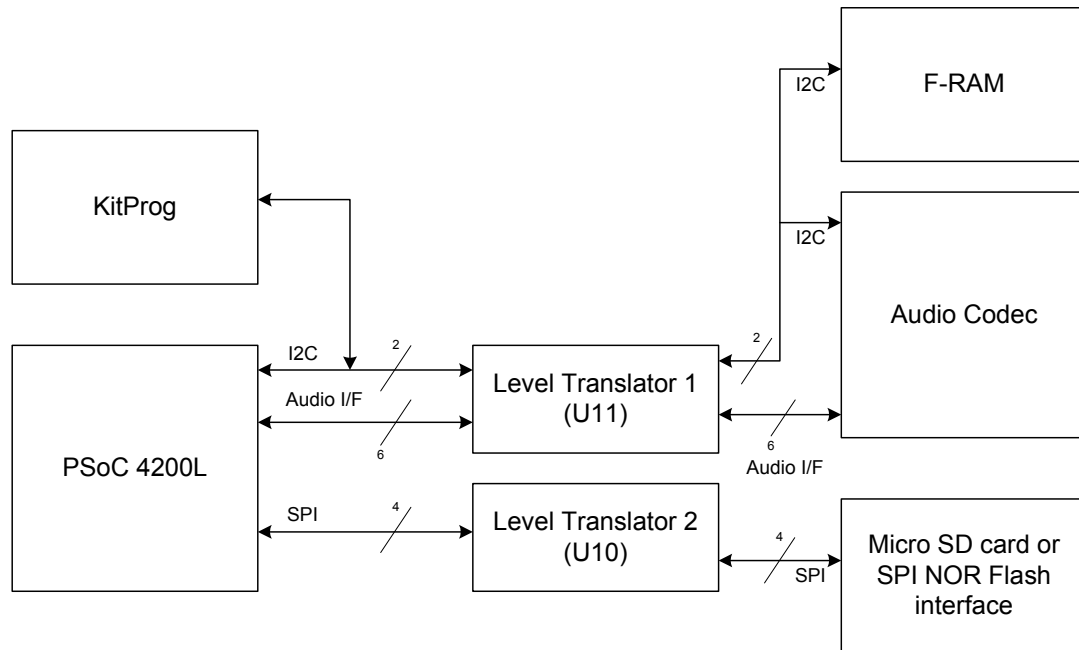
A.2.2   Shield Board

A.2.2.1   *CapSense Circuit*

CapSense Gesture Pad

The shield board includes a five-element CapSense Gesture Pad along with an eight-element radial slider, which are connected to PSoC 4200L (marked 24 in Figure A-1 on page 58). All the pins on the CapSense Gesture Pad are connected to the CSD0 block of PSoC 4200L. The Gesture Pad is used to emulate mouse actions, joystick, LED color/intensity controls, and audio playback controls in the code examples and can be used for general purposes. The CMOD and CTANK capacitors essential for CapSense functionality are connected close to the PSoC 4200L device on the main board. See "CapSense Circuit" on page 65 for details on the CMOD and CTANK capacitors.

Proximity Headers

The shield board contains two headers (J21 and J22) for CapSense proximity wire connection (marked 22 in Figure A-1 on page 58). If the proximity functionality is not being used, these pins can be used as standard GPIO pins.

Note: PSoC 4200L supports dual-channel capacitive sensing and the proximity pins are connected to the second CapSense block (CSD1). As a result, when using the proximity sensors in your design, CMOD1 will be required to implement the capacitive sensing. CMOD1 is populated on the board by default; you need to select the appropriate CMOD pin (P5[0]) in the PSoC Creator to successfully build and use the design.

A.2.2.2   *Expansion Connectors*

Arduino-compatible Headers (J17, J18, J19, J20, and J24)

The PSoC 4 L-Series Pioneer Kit has five additional Arduino-compatible headers on the shield board–J17, J18, J19, J20, and J24. Header J17 is marked 18 and header J18, J19, and J20 are

marked 21 in Figure A-1 on page 58. You can develop applications based on the Arduino shield's hardware.

The J17 header contains I/O pins for reset, I/O reference voltage (IOREF), and power supply line. The J18 header is an analog port that contains I/O pins for SAR ADC, comparator, and opamp. The header J19 has I/O pins that support connection to PSoC 4200L communication blocks. The J19 header is primarily a digital port that contains I/O pins for PWM, I2C, SPI, and analog reference. The J20 header is also a digital port that contains I/O pins for UART and PWM. The J24 header is an Arduino ICSP-compatible header for the SPI interface. See "Arduino-Compatible Headers" on page 79 for details on the headers pinout.

Additional Functionality of Headers J17, J18, J19, J20

The headers J17, J18, J19, and J20 are two row headers. When the shield board is used along with the main board (either without snapping or through J26 and J27), the odd pins on the header (pins 1, 3, 5, and so on) act as Arduino baseboard headers. These pins can interface to any Arduino shield board and can work simultaneously with the main board Arduino shield-compatible headers (J1, J2, J3, J4), as shown in Figure A-10. When the shield board is snapped, the even pins (2, 4, 6, and so on) act as Arduino shield headers. After the board is snapped, these pins can be used to convert the shield board into an Arduino shield. The shield board can then be interfaced with any Arduino base-board such as CY8CKIT-044, CY8CKIT-042, and others. By default, the female Arduino baseboard headers (odd pins on J17, J18, J19, and J20) are populated on the board. After the board is snapped, these female connectors should be removed and alternate connectors provided in the BOM (under No load components) should be populated to use them as Arduino shields. See A.2.3.2 Using the Shield Board on page 72 for details on using the shield board as an Arduino shield.

Figure A-10. CY8CKIT-046 with Two Arduino Shields
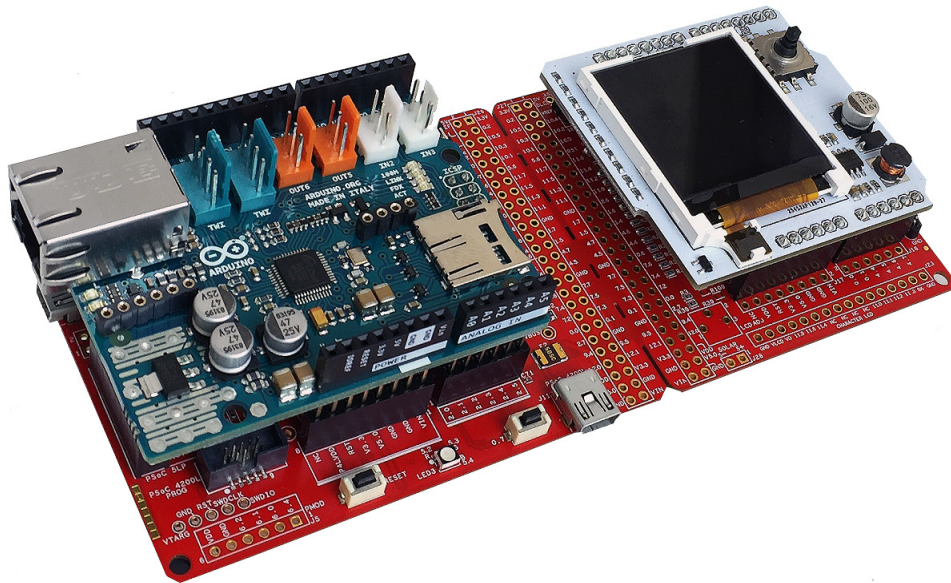
A.2.2.3   EZ-BLE PRoC Module (U8 - footprint only)

The board includes a footprint for mounting an EZ-BLE PRoC Module (marked 28 in Figure A-1 on page 58). The EZ-BLE PRoC Module is a fully integrated, 10×10×1.8 mm, fully certified, program-mable module designed for ease-of-use and reduced time-to-market. It contains Cypress's PRoC BLE chip, two crystals, chip antenna, shield, and passive components. Refer to AN96841 - Getting Started With EZ-BLE PRoC Module for more details. Visit www.cypress.com/ez-bleprocmodule/ to buy the EZ-BLE PRoC Module.

Figure A-11. EZ-BLE Connections
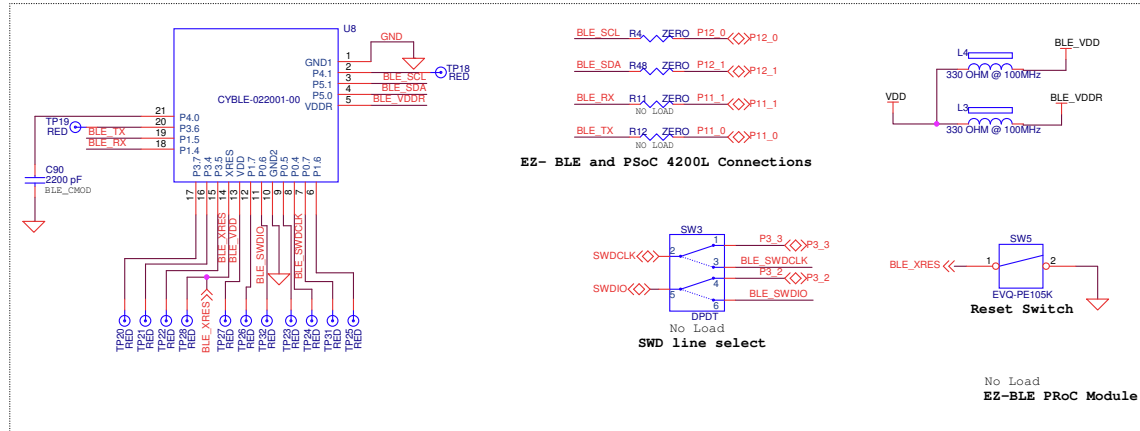
The EZ-BLE PRoC Module includes the following connections to the PSoC 4200L and KitProg:

■ UART connections to PSoC 4200L through zero-ohm resistors R11 and R12. See Figure A-11 and Table A-2.

■ I2C connections to PSoC 4200L through zero-ohm resistors R4 and R48. See Table A-2 for details. The EZ-BLE I2C lines can also be connected to the KitProg I2C lines by populating the zero-ohm resistors R96 and R97 on the main board. The EZ-BLE I2C lines also connect to the Arduino-designated I2C lines on the shield board.

■ SWD connections to KitProg through the DPDT switch SW3. The SWD connections are shared between the PSoC 4200L and EZ-BLE devices. By default, SW3 is not populated and the KitProg SWD connections are hardwired to PSoC 4200L through zero-ohm resistors, R70 and R71.

Table A-2. EZ-BLE Connections

EZ-BLE Connections			
EZ-BLE	PSoC 4200L	KitProg (PSoC 5LP)	Description
BLE_TX, P1.5	P11[0]	–	GPIO/UART_TX
BLE_RX, P1.4	P11[1]	–	GPIO/UART_RX
BLE_SDA, P5.0	P12[1]	–	GPIO/I2C_SDA
BLE_SCL, P5.1	P12[0]	–	GPIO/I2C_SCL
BLE_SWDIO, P0.6	–	SWDIO, P12[2]	SWD I/O
BLE_SWDCLK, P0.7	–	SWDCLK, P12[3]	SWD Clock

To program the EZ-BLE PRoC Module using KitProg, follow these instructions:

1. Populate the DPDT switch, SW3. Refer to A.6 Bill of Materials on page 86 for the part number of the switch to be used.

2. Populate the EZ-BLE PRoC Module, U4 (footprint on the reverse side of the shield board).

3. Remove the zero-ohm resistors R70 and R71.

4. Move the DPDT switch to **PSoC 4200L** position to program PSoC 4200L and to **EZ-BLE** position to program the EZ-BLE PRoC Module.

In addition, the EZ-BLE PRoC Module can be independently programmed and debugged through the 5-pin SWD header (marked 25 in Figure A-1 on page 58). The footprint for populating the CMOD capacitor (C90, 2.2 nF) required to develop CapSense applications with the EZ-BLE PRoC Module

is also provided on the board. In addition, the I/Os from the module are also brought to the EZ-BLE I/O header, marked 27 in Figure A-1 on page 58. When the shield board is snapped, the EZ-BLE PRoC Module can be accessed through the Arduino shield I2C lines or through the EZ-BLE I/O header. The footprint for a reset switch, SW5 (marked 23 in Figure A-1 on page 58), is also provided on the shield board. This switch can be populated and used to reset the EZ-BLE device after the board is snapped.

A.2.2.4   Character LCD Circuit (J23 - footprint only)

The shield board of the PSoC 4 L-Series Pioneer Kit includes a 1x16 header footprint (J23) for interfacing a standard 2x16 character LCD (marked 20 in Figure A-1 on page 58). The character LCD data and control lines are connected to the pins P11[6:0] of PSoC 4200L. A potentiometer, R39 (marked 19 in Figure A-1 on page 58), can be populated if required to control the LCD contrast. By default, resistors R41 and R43 are populated to provide a default contrast for the LCD. These resistors should be removed before populating R39. In addition, the kit also offers LCD power supply selection through resistors R44 and R45. The LCD can operate at 5 V (R44 populated and R45 unpopulated) or 3.3 V (R44 unpopulated and R45 populated). By default, the 3.3 V power supply is selected for the LCD operation.

A.2.2.5   µSD Card Interface (J25 - footprint only)

The kit includes a µSD connector, J25, footprint on the reverse side of the shield board. The µSD card can be interfaced over SPI lines (P10[0] to P10[3]) of PSoC 4200L. The µSD connector shares footprint and connections with the serial NOR Flash memory, U13. Refer to "Bill of Materials" on page 86 for details on the µSD connector part number.

A.2.2.6   Serial NOR Flash Memory (U13 - footprint only)

The kit includes a footprint to connect the S25FL512S (Serial NOR Flash memory from Cypress), U13. The NOR Flash memory shares footprint and SPI connections with the µSD connector, J25. Only one of the two can be used on the board at any time.

A.2.2.7   Energy Harvesting PMIC (U12 - footprint only)

The kit includes a footprint for S6AE101A (an Energy Harvesting PMIC from Cypress), U12. The Energy Harvesting PMIC provides supply to the EZ-BLE PRoC Module, U8. The PMIC can harvest solar power, when connected to a solar cell through J28 and power EZ-BLE PRoC Module. The S6AE101A stores power generated by the solar cell to an output capacitor using built-in switch control. It turns on the power switching circuit while the capacitor voltage is within a preset maximum and minimum range to supply energy to a load. If the power generated from solar cells is not enough, energy can also be supplied similar to solar cells from a coin cell battery (V1) for auxiliary power. The values of resistors (R100, R101, and R102) present in the schematic (Figure A-12) are selected to generate a maximum output voltage (VOUTH) of 3.25 V and minimum output voltage (VOUTL) of 1.91 V. Refer to the "Setting the Operating Conditions" section in the S6AE101A datasheet for details. Note that to use the PMIC, the EZ-BLE PRoC Module should be populated, all the components shown in Figure A-12 should be populated, L4/L3 should be removed, and L5/L6 should be populated.

Figure A-12. Energy Harvester Schematic
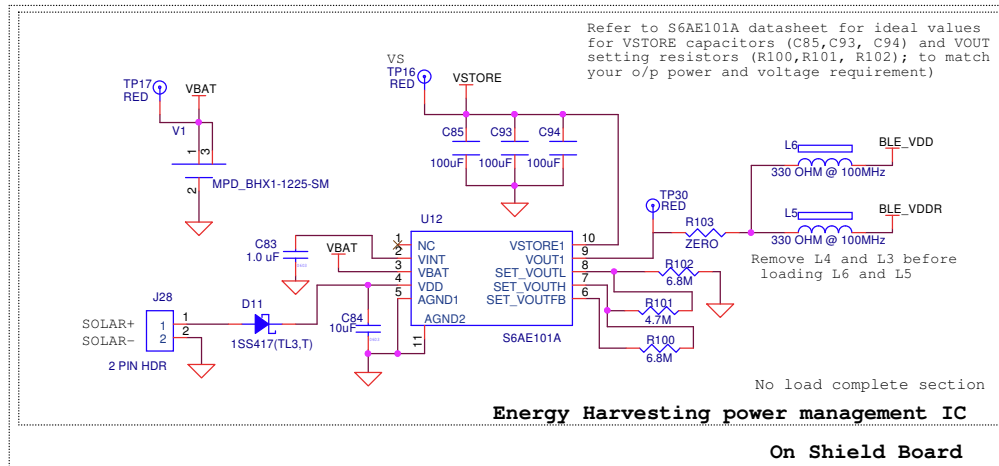
A.2.3 Board Separation (Snapping)

The PSoC 4 L-Series Pioneer Kit consists of two boards – a main board and a shield board (see Figure A-1 on page 58). To separate the two boards for testing or development, break the two boards apart at the built-in perforated edge.

The easiest method of separating the two boards is to place the kit on the edge of a table, where the edge of the table is directly below the perforated edge and the shield board is off the table edge. Press gently on the shield board and snap the two boards apart. If any material is removed from the edge of the boards, use shears to clean up the edge of the kit. Figure A-13 shows the boards after separation.

After snapping out, the main board acts as an Arduino baseboard and the shield board can be used as an Arduino shield board.

Figure A-13. CY8CKIT-046 Board Separation
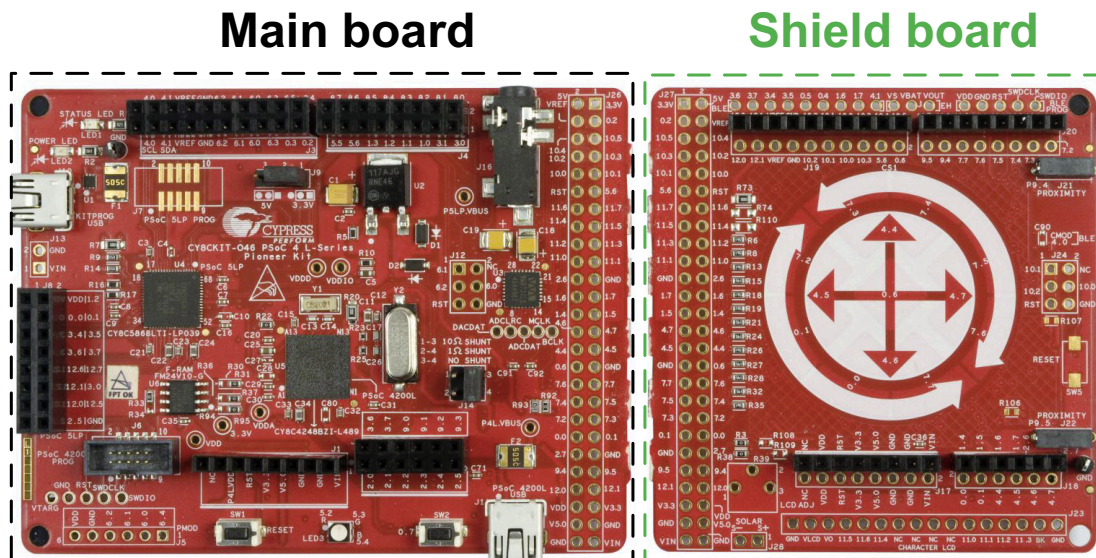
A.2.3.1   Restoring Main Board and Shield Board Connections

Both the main board and shield board can be restored to their original connections by populating the 52-pin female/male headers J26/J27 and connecting them together. This is shown in Figure A-14. Note that this method restores most of the connections between the boards. However, there are some known limitations and differences between the restored board and the actual (not separated) board. The details are captured in "Known Limitations after Board Separation" on page 74.

Figure A-14. CY8CKIT-046 Main Board - Shield Board Interface through J26/J27

A.2.3.2   Using the Shield Board

The separated shield board can be used as an Arduino shield with any Arduino-compatible base-board such as CY8CKIT-042 and CY8CKIT-044. This board can also be used as a shield with the CY8CKIT-046 main board. To use the board as a shield, the even pins on headers J17, J18, J19, and J20 should be populated with male headers that can plug into an Arduino baseboard. Figure A-15 shows the shield board with the male headers populated and connected to the CY8CKIT-046 main board. Figure A-16 shows the shield board used as an Arduino shield with the CY8CKIT-044.

Figure A-15. CY8CKIT-046 Shield Board as Arduino Shield (with CY8CKIT-046 Main Board)

Arduino Shield headers
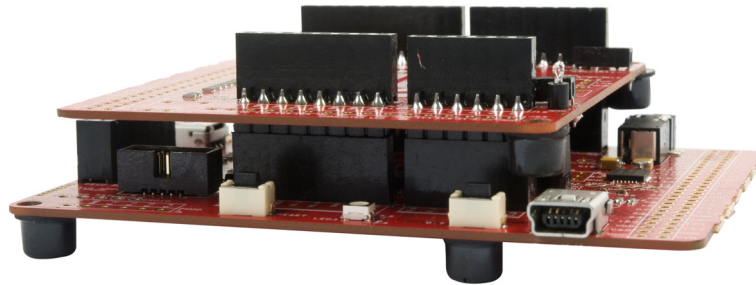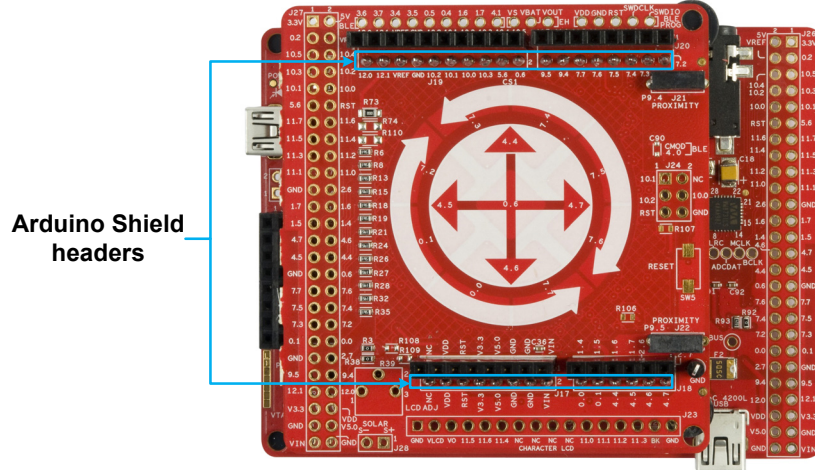
Figure A-16. CY8CKIT-046 Shield Board as Arduino Shield (with CY8CKIT-044)
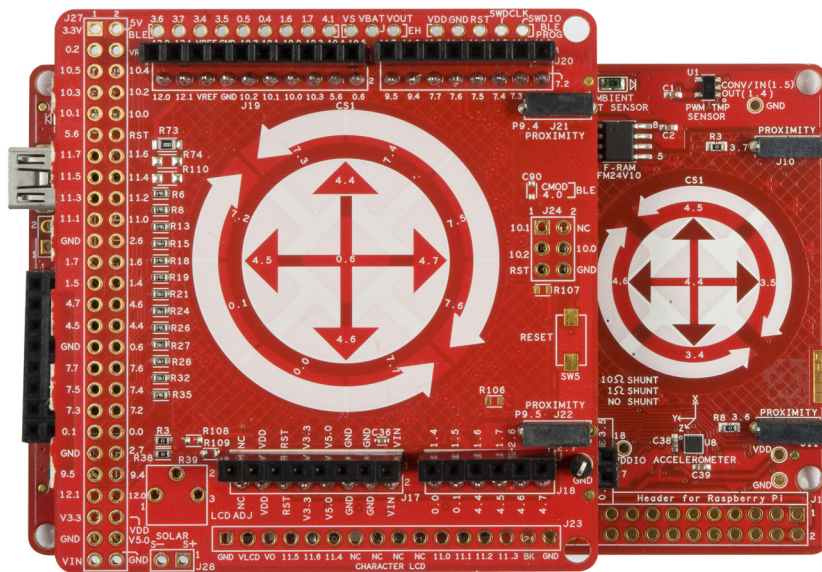
The shield pins that connect to the baseboard are also connected to CapSense Gesture Pad elements (13 sensors), CapSense proximity sensors (two sensors), CapSense shield signal, SPI lines (µSD card or Serial NOR Flash interface), and I2C lines (EZ-BLE PRoC Module interface) on the

shield board; see Figure A-17. In addition, connection to the character LCD header, EZ-BLE I/Os, and Energy Harvesting PMIC outputs are available on the headers marked 20, 27, and 26 in Figure A-1 on page 58. Note that the connection to the character LCD header can be taken from the odd pins on header J20. The CapSense pins on the shield can be used as general-purpose pins to connect to the EZ-BLE I/Os and character LCD pins when CapSense is not used.

Figure A-17. Shield Board Arduino Pin Markup
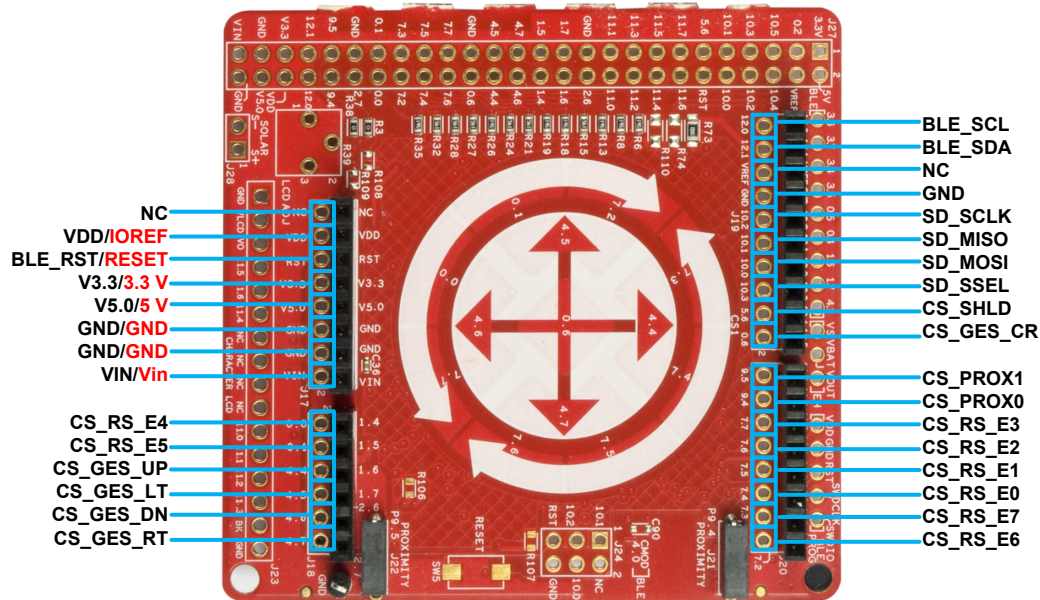
A.2.3.3   Known Limitations after Board Separation

The separated boards have some known limitations:

- When the separated shield board is connected back to the main board through J26 or J27, the EZ-BLE programming lines are not restored. As a workaround, use the EZ-BLE programming header (marked 25 in Figure A-1 on page 58) to program the EZ-BLE device (if populated) on the shield board.

- In REV05 version of the PSoC 4 L-Series Pioneer kit, when the shield board is used as an Arduino shield or connected back to the main board after separation, V5.0 and V3.3 power domains (from the main board) do not connect to the respective power domains of the shield board. As a workaround, connect pin 8 and pin 10 on header J17 to the right pad of R44 and left pad of R45 respectively, using jumper wires to enable V5.0 and V3.3. Figure A-18 shows this. Note that this connection will bypass the onboard protection diodes and any external voltage larger than 3.3 V and 5 V on the respective domains will damage the active components connected on this domain.

Figure A-18. Power Domain Restoration in Shield Board
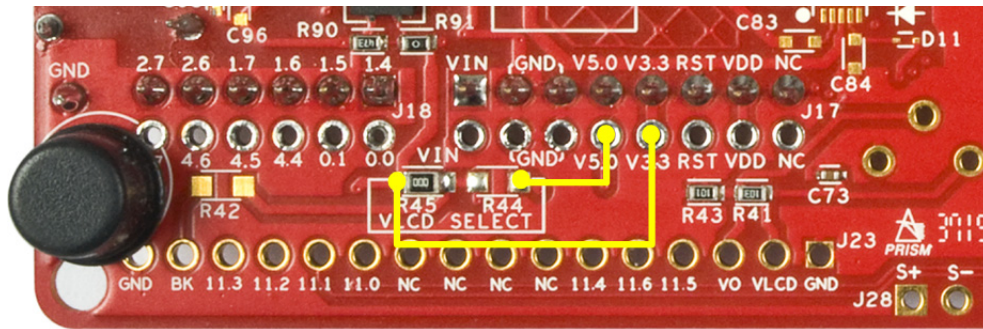
A.3   Using FM24V10 F-RAM

The PSoC 4 L-Series Pioneer board has an onboard ferroelectric RAM chip that can hold up to 1 Mb of data. The chip provides an I2C communication interface for data access. It is hardwired to the I2C interface (P4[0] and P4[1] of the PSoC 4200L device); the same lines are routed to the KitProg I2C interface. Because the F-RAM device is an I2C slave, it can be accessed or shared among various I2C masters on the same lines. For more details on the F-RAM device, refer to the device datasheet.

A.3.1   Address Selection

The slave address of the F-RAM device consists of three parts, as shown in Figure A-19: slave ID, device select, and page select. Slave ID is an F-RAM family-specific ID provided in the datasheet of the particular F-RAM device. For the device (FM24V10-G) used on the PSoC 4 L-Series Pioneer board, the slave ID is 1010b. Device select bits are set using the two physical pins A2 and A1. The setting of these two pins on the PSoC 4 L-Series Pioneer board is controlled by resistors R30/R31 (A2) and R36/R37 (A1). Because the memory location in F-RAM is divided into two pages of 64 KB each, the page select bit is used to refer to one of the two pages in which the read or write operations will take place.

Note: The 8-pin SOIC footprint provided for the F-RAM FM24V10 on the PSoC 4 L-Series Pioneer Kit is compatible with all I2C-based F-RAM devices from Cypress (FM24Vxx, FM24CLxx, and CY15BxxxJ parts). The F-RAM parts with more than 64 KB size support only four addresses (four devices of the same type on the same I2C bus); resistors connected to A1 (R36/R37) and A2 (R30/R31) pins can be used to select any of the four addresses. The F-RAM parts with less than 64 KB and FM24CLxx parts support eight addresses; resistors connected to A0 (R94/R95), A1 (R36/R37), and A2 (R30/R31) pins can be used to select one of the eight addresses.

Figure A-19. F-RAM I2C Address Byte Structure
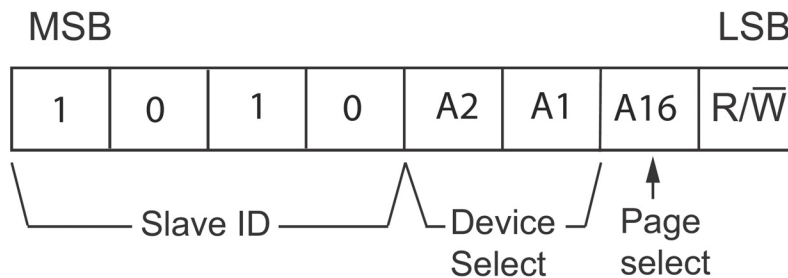
A.3.2   High Speed Mode (Hs-mode)

The FM24V10 supports a 3.4-MHz high-speed mode. A master code (00001XXXb) must be issued to place the device into high-speed mode. Communication between master and slave will then be enabled for speeds up to 3.4 MHz. A STOP condition will exit Hs-mode. Single- and multiple-byte reads and writes are supported.

Figure A-20. F-RAM I2C Data Format for HS Mode
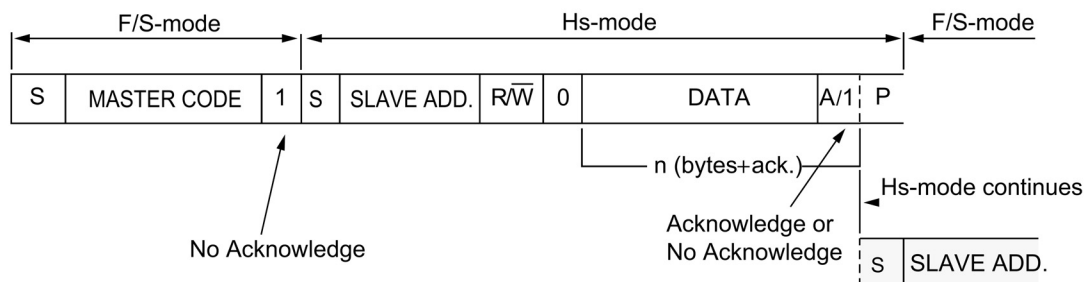
A.3.3   Write/Read Operation

The F-RAM device datasheet includes details on how to perform a write/read operation with F-RAM. Figure A-21 and Figure A-22 show the write/read packet structure as a quick reference.

Figure A-21. F-RAM Write Packet Structure
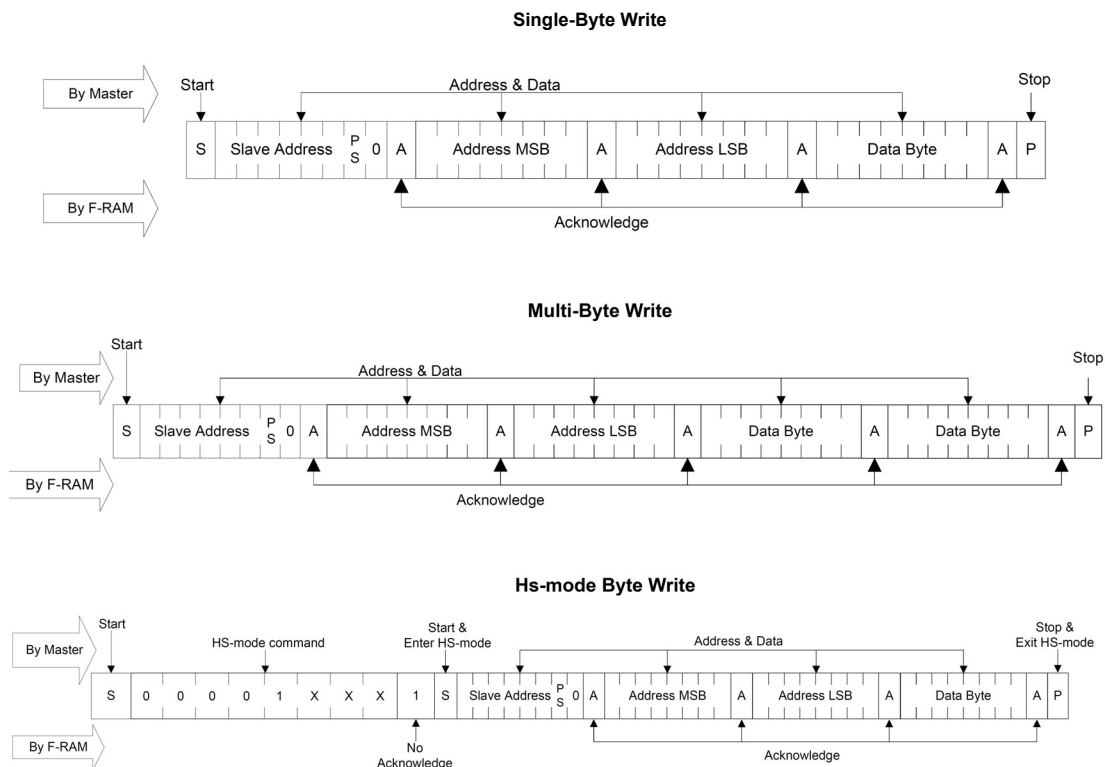
Figure A-22. F-RAM Read Packet Structure

Current Address Read

Sequential Read

Hs-mode Current Address Read
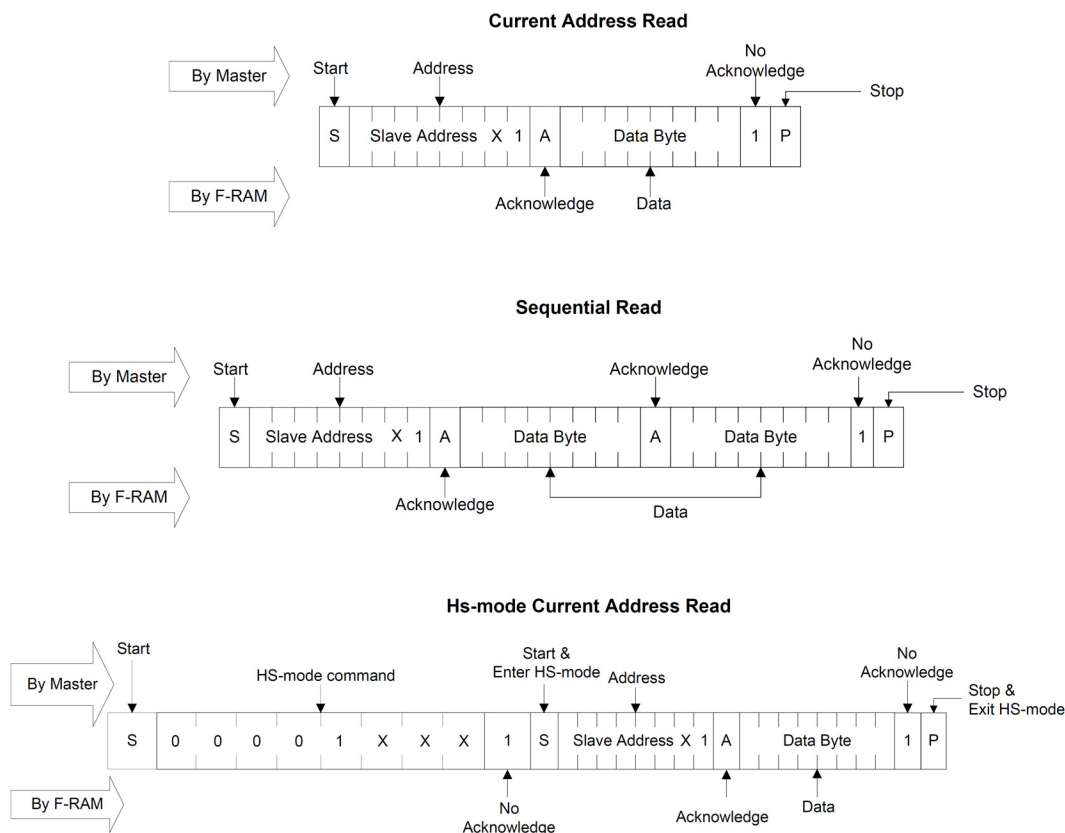
As the figures show, operations start with the slave address followed by the memory address. For write operations, the bus master sends the slave address and memory address followed by one or more data bytes. Each byte of data is written to consecutive locations in the memory, and the memory generates an acknowledgement condition.

For 'Current Address Read' and 'Sequential Read', the bus master sends only the slave address. The memory address used is the same address that was set by the previous 'Write' or 'Selective Read' operation. For 'Selective Read' operations, after receiving the complete slave address and memory address, the memory will begin shifting data from the current address on the next clock

Note: Visit the CY15FRAMKIT-001 kit webpage for code examples and the Arduino library for interfacing I2C F-RAM devices with the PSoC 4 family.

A.4 Migrating s Across Different Pioneer Series Kits

All Cypress Pioneer series kits are Arduino-compatible and have some common onboard peripherals such as RGB LED, CapSense, and user switch. However, the pin mapping in each of the boards is different due to differences in pin functions of the PSoC device used. This section lists the pin mapping of the Pioneer series kits to allow for easy migration of s across different kits.

In some cases, the pins available on the Pioneer kit headers are a superset of the standard Arduino pins. For example, J2 contains only one row of pins on the Arduino pin layout while it contains two rows of pins on many of the Pioneer series kits.

The CY8CKIT-046 PSoC 4 L-Series Pioneer Kit has an additional shield board, which offers another set of Arduino-compatible pins. The pins present in the side board are completely independent of the ones present in the main board and hence offers the ability to use two Arduino shields simultaneously. The side board can be snapped out and used as an Arduino shield. The odd pins on the headers J17, J18, J19, and J20 provide the functionality of Arduino baseboard headers whereas the even pins on these headers, when snapped out, offers Arduino shield functionality.

Figure A-23. Pioneer Series Kits Pin Map
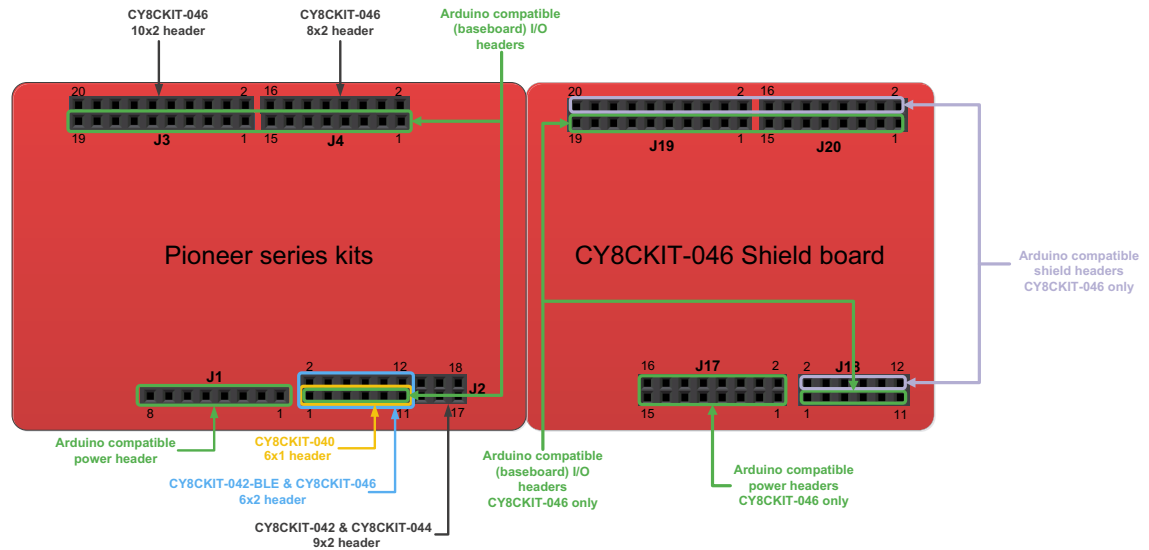
A.4.1 Arduino-Compatible Headers

Table A-3. J1 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	VIN	VIN	VIN	VIN	VIN	VIN
2	GND	GND	GND	GND	GND	GND
3	GND	GND	GND	GND	GND	GND
4	5V	V5.0	V5.0	V5.0	V5.0	V5.0
5	3.3V	V3.3	V3.3	V3.3	V3.3	V3.3
6	RESET	RESET	RESET	RESET	RESET	RESET
7	IOREF	P4.VDD	P4.VDD	BLE.VDD	P4.VDD	P4L.VDD
8	NC	NC	NC	NC	NC	NC

Table A-4. J2 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	A0	P2[0]	P0[0]	P3[0]	P2[0]	P2[0]
2	–	P0[2][a]	–	P2[0]	P2[6][a]	P3[6][a]
3	A1	P2[1]	P0[1]	P3[1]	P2[1]	P2[1]
4	–	P0[3][a]	–	P2[1][a]	P6[5][a]	P3[7][a]
5	A2	P2[2]	P0[2][a]	P3[2]	P2[2]	P2[2]
6	–	P4.VDD	–	P2[2][a]	P0[6][a]	P9[0]
7	A3	P2[3]	P0[4][a]	P3[3]	P2[3]	P2[3]
8	–	P1[5][a]	–	P2[3][a]	P4[4][a]	P9[1]
9	A4	P2[4]	P1[3]	P3[4]	P2[4]	P2[4]
10	–	P1[4][a]	–	P2[4][a]	P4[5][a]	P9[2]
11	A5	P2[5]	P1[2]	P3[5]	P2[5]	P2[5]
12	–	P1[3][a]	–	P2[5][a]	P4[6][a]	P9[3]
13	–	P0[0]	–	–	P0[0]	–
14	–	GND	–	–	GND	–
15	–	P0[1]	–	–	P0[1]	–
16	–	P1[2][a]	–	–	P3[4][a]	–
17	–	P1[0]	–	–	P0[7][a]	–
18	–	P1[1][a]	–	–	P3[5][a]	–

a. These pins are also used for onboard peripheral connections. Refer to the A.4.2 Onboard Peripherals on page 83 section for connection details.

Table A-5. J3 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	D8	P2[6]	P1[4]	P0[5]	P0[2]	P0[2]
2	-	-	-	-	-	P3[4]
3	D9	P3[6]	P1[5]	P0[4]	P0[3]	P0[3]
4	-	-	-	-	-	P6[5]
5	D10	P3[4]	P1[6]	P0[2]	P2[7]	P6[3]
6	-	-	-	-	-	P6[3]
7	D11	P3[0]	P1[1][a]	P0[0]	P6[0]	P6[0]
8	-	-	-	-	-	P6[0]
9	D12	P3[1]	P3[1]	P0[1]	P6[1]	P6[1]
10	-	-	-	-	-	P6[1]
11	D13	P0[6]	P1[7]	P0[3]	P6[2]	P6[2]
12	-	-	-	-	-	P6[2]
13	GND	GND	GND	GND	GND	GND
14	-	-	-	-	-	GND
15	AREF	P1[7]	NC	VREF	P1[7]	VREF
16	-	-	-	-	-	VREF
17	SDA	P4[1]	P1[3]	P3[4]	P4[1]	P4[1]
18	-	-	-	-	-	P4[1]
19	SCL	P4[0]	P1[2]	P3[5]	P4[0]	P4[0]
20	-	-	-	-	-	P4[0]

a. These pins are also used for onboard peripheral connections. Refer to the A.4.2 Onboard Peripherals on page 83 section for connection details.

Table A-6. J4 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	D0	P0[4]	P0[5]	P1[4]	P3[0]	P3[0]
2	-	-	-	-	-	P8[0]
3	D1	P0[5]	P0[6]	P1[5]	P3[1]	P3[1]
4	-	-	-	-	-	P8[1]
5	D2	P0[7][a]	P0[7]	P1[6]	P1[0]	P1[0]
6	-	-	-	-	-	P8[2]
7	D3	P3[7]	P3[2][a]	P1[7]	P1[1]	P1[1]
8	-	-	-	-	-	P8[3]
9	D4	P0[0]	P0[3]	P1[3]	P1[2]	P1[2]
10	-	-	-	-	-	P8[4]
11	D5	P3[5]	P3[0]	P1[2]	P1[3]	P1[3]
12	-	-	-	-	-	P8[5]

Table A-6. J4 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
13	D6	P1[0]	P1[0]	P1[1]	P5[3]	P5[6]
14	-	-	-	-	-	P8[6]
15	D7	P2[7]	P2[0][a]	P1[0]	P5[5]	P5[5]
16	-	-	-	-	-	P8[7]

a. These pins are also used for onboard peripheral connections. Refer to the section for connection details.

Table A-7. J17 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kits
		CY8CKIT-046 Shield Board
1	VIN	VIN
2	VIN	VIN
3	GND	GND
4	GND	GND
5	GND	GND
6	GND	GND
7	5V	V5.0
8	5V	V5.0
9	3.3V	V3.3
10	3.3V	V3.3
11	RESET	RESET
12	RESET	RESET
13	IOREF	P4L.VDD
14	IOREF	P4L.VDD
15	NC	NC
16	NC	NC

Table A-8. J18 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kit
		CY8CKIT-046 Shield Board
1	A0	P1[4]
2	–	P0[0]/CS_RS_E4
3	A1	P1[5]
4	–	P0[1]/CS_RS_E5
5	A2	P1[6]
6	–	P4[4]/CS_GES_UP
7	A3	P1[7]
8	–	P4[5]/CS_GES_LT

Table A-8. J18 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kit
		CY8CKIT-046 Shield Board
9	A4	P2[6]
10	–	P4[6]/CS_GES_DN
11	A5	P2[7]
12	–	P4[7]/CS_GES_RT

Table A-9. J19 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kit
		CY8CKIT-046 Shield Board
1	D8	P10[5]
2	-	P0[6]/CS_GES_CR
3	D9	P10[4]
4	-	P5[6]/CS_SHLD
5	D10	P10[3]
6	-	P10[3]
7	D11	P10[0]
8	-	P10[0]
9	D12	P10[1]
10	-	P10[1]
11	D13	P10[2]
12	-	P10[2]
13	GND	GND
14	-	GND
15	AREF	VREF
16	-	VREF
17	SDA	P12[1]
18	-	P12[1]
19	SCL	P12[0]
20	-	P12[0]

Table A-10. J20 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kit
		CY8CKIT-046 Shield Board
1	D0	P11[0]/LCD_D4
2	-	P7[2]/CS_RS_E6
3	D1	P11[1]/LCD_D5
4	-	P7[3]/CS_RS_E7
5	D2	P11[2]/LCD_D6
6	-	P7[4]/CS_RS_E0
7	D3	P11[3]/LCD_D7

Table A-10. J20 Arduino-Compatible Header Pin Map

#	Arduino Pin	Pioneer Series Kit CY8CKIT-046 Shield Board
8	-	P7[5]/CS_RS_E1
9	D4	P11[4]/LCD_EN
10	-	P7[6]/CS_RS_E2
11	D5	P11[5]/LCD_RS
12	-	P7[7]/CS_RS_E3
13	D6	P11[6]/LCD_RW
14	-	P9[4]/CS_PROX0
15	D7	P11[7]
16	-	P9[5]/CS_PROX1

A.4.2 Onboard Peripherals

Table A-11. CapSense Pin Map

#	CapSense Pin	Pioneer Series Kits				
		CY8CKIT-042 (slider)	CY8CKIT-040	CY8CKIT-042-BLE (slider)	CY8CKIT-044 (Gesture Pad)	CY8CKIT-046 (Gesture Pad with Radial Slider)[a]
1	CapSense Sensor 1	P1[1]/CS_LS_E0	–	P2[1]/CS_LS_E0	P4[4]/CS_GES_CR	P0[6]/CS_GES_CR
2	CapSense Sensor 2	P1[2]/CS_LS_E1	–	P2[2]/CS_LS_E1	P4[5]/CS_GES_UP	P4[5]/CS_GES_LT
3	CapSense Sensor 3	P1[3]/CS_LS_E2	–	P2[3]/CS_LS_E2	P4[6]/CS_GES_LT	P4[4]/CS_GES_UP
4	CapSense Sensor 4	P1[4]/CS_LS_E3	–	P2[4]/CS_LS_E3	P3[4]/CS_GES_DN	P4[7]/CS_GES_RT
5	CapSense Sensor 5	P1[5]/CS_LS_E4	–	P2[5]/CS_LS_E4	P3[5]/CS_GES_RT	P4[6]/CS_GES_DN
6	CapSense Sensor 10	–	–	–	–	P7[4]/CS_RS_E0
7	CapSense Sensor 11	–	–	–	–	P7[5]/CS_RS_E1
8	CapSense Sensor 12	–	–	–	–	P7[6]/CS_RS_E2
9	CapSense Sensor 13	–	–	–	–	P7[7]/CS_RS_E3
10	CapSense Sensor 6	–	–	–	–	P0[0]/CS_RS_E4
11	CapSense Sensor 7	–	–	–	–	P0[1]/CS_RS_E5
12	CapSense Sensor 8	–	–	–	–	P7[2]/CS_RS_E6
13	CapSense Sensor 9	–	–	–	–	P7[3]/CS_RS_E7

Table A-11. CapSense Pin Map

#	CapSense Pin	Pioneer Series Kits				
		CY8CKIT-042 (slider)	CY8CKIT-040	CY8CKIT-042-BLE (slider)	CY8CKIT-044 (Gesture Pad)	CY8CKIT-046 (Gesture Pad with Radial Slider)[a]
14	CMOD[b]	P4[2]	P0[4]	P4[0]	P4[2]	P4[2]
15	CTANK[b]	P4[3]	P0[2]	P4[1]	P4[3]	P4[3]
16	CMOD[b]	–	–	–	–	P5[0]
17	CTANK[b]	–	–	–	–	P5[1]
18	CapSense Shield	P0[1]	–	P1[6]	P0[1]	P0[2]

a. The CapSense elements are present on the CY8CKIT-046 shield board. The radial slider (CapSense sensors 6 to 13) is symmetric and the sensor order can be shifted to fit your requirement, that is, the desired zero position on the slider.
b. CMOD0, CTANK0, CMOD1, and CTANK1 are only present in the CY8CKIT-046 PSoC 4 L-Series Pioneer Kit.

Table A-12. Proximity Header Pin Map

#	Proximity Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	PROX0	–	P2[0]	P2[0]	P3[7]	P9[4]
2	PROX1	–	–	–	P3[6]	P9[5]

Table A-13. RGB LED Pin Map

#	LED Pin	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	Red	P1[6]	P3[2]	P2[6]	P0[6]	P5[2]
2	Green	P0[2]	P1[1]	P3[6]	P2[6]	P5[3]
3	Blue	P0[3]	P0[2]	P3[7]	P6[5]	P5[4]

Table A-14. User Switch Pin Map

#	Switch	Pioneer Series Kits				
		CY8CKIT-042	CY8CKIT-040	CY8CKIT-042-BLE	CY8CKIT-044	CY8CKIT-046
1	SW2	P0[7]	–	P2[7]	P0[7]	P0[7]

Table A-15. Audio Codec Pin Map

#	Codec Pin	Pioneer Series Kit
		CY8CKIT-046 Shield Board
1	SDA	P4[1]
2	SCL	P4[0]
3	ADCDAT	P3[7]
4	DACDAT	P3[6]

Table A-15. Audio Codec Pin Map

| # | Codec Pin | Pioneer Series Kit |
		CY8CKIT-046 Shield Board
5	ADCLRC	P3[5]
6	DACLRC	P3[5]
7	BCLK	P3[4]
8	MCLK	P5[7]

A.5   KitProg Status LED States

The KitProg status LED on the development kit indicates the status of the KitProg operation using different blink rates. Table A-16 shows the KitProg LED states and the corresponding status of the KitProg.

Table A-16. KitProg LED Indications

User Indication	Scenario	Action Required by User
LED blinks fast: Frequency = 4.00 Hz	LED starts blinking at power up, if bootloadable file is corrupt.	Bootload the *KitProg.cyacd* file: In PSoC Programmer, connect to the kit, go to the **Utilities** tab, and press the **Upgrade Firmware** button.
LED blinks slow: Frequency = 0.67 Hz	Entered Bootloader mode by holding the Reset button during kit power-up.	Release the Reset button and re-plug the kit if you entered this mode by mistake. If the mode entry was intentional, bootload the new *.cyacd* file using the Bootloader Host tool available in PSoC Creator.
LED blinks very fast: Frequency = 15.0 Hz	SWD or I2C operation is in progress. The Kit's COM port connect/disconnect event (only one blink).	In PSoC Programmer, watch the log window for status messages for SWD operations. In the BCP, the LED blinks on I2C command requests. In BCP or any other serial port terminal program, distinguish the kit's COM port number by the blinking LED when the port is connected or disconnected.
LED is ON	USB enumeration successful. Kit is in the idle state waiting for commands.	PSoC Creator, PSoC Programmer, BCP, and any serial port terminal program can use the kit functions.
LED is OFF	Power LED is ON	This means that the USB enumeration was unsuccessful. This may happen if the kit is not powered from the USB host. Verify the USB cable and check if PSoC Programmer is installed on the PC.

A.6 Bill of Materials

Item	Qty	Reference	Value	Description	Manufacturer	Mfr Part Number
1	1	600-60276-01	-	67.552x152.333, High Tg, ENIG finish, 6 layer, Mask-RED, Silk-WHITE.	Cypress Semiconductor	600-60276-01
2	1	C1	22 uFd 16v	CAP TANT 22UF 16V 10% 1210	AVX Corporation	TPSB226K016R0600
3	35	C2,C3,C6,C7,C8,C9, C12,C15,C16,C20,C21,C23,C29,C30,C31, C32,C33,C35,C36,C39,C40,C47,C48,C49, C50,C53,C59,C70,C71,C72,C73,C74,C75, C91,C92	0.1 uFd	CAP CER 0.1UF 10V 10% X5R 0402 CAP CER 0.1UF 16V X5R 0402	TDK Corporation	C1005X5R1A104K050BA
4	20	C5,C11,C22,C24,C25, C34,C42,C43,C54,C56,C57,C58,C60,C61, C62,C65,C66,C67,C68,C69	1.0 uFd	CAP CERAMIC 1.0UF 25V X5R 0603 10%	Taiyo Yuden	TMK107BJ105KA-T
5	1	C13	18 pF	CAP CER 18PF 50V 1% NP0 0402	Murata Electronics	GRM1555C1H180FA01D
6	1	C14	36 pF	CAP CER 36PF 50V 5% NP0 0402	Murata Electronics	GRM1555C1H360JA01D
7	2	C17,C26	12 pF	CAP CER 12PF 50V 2% NP0 0402	Johanson Technology Inc	500R07S120GV4T
8	2	C18,C19	220uFd	CAP TANT 220UF 6.3V 20% 1210	AVX Corporation	F950J227MBAAQ2
9	2	C37,C38	0.01 uFd	CAP 10000PF 16V CERAMIC 0402 SMD	TDK Corporation	C1005X7R1C103K050BA
10	1	C41	220 pF	CAP CER 220PF 50V 5% C0G 0603	TDK Corporation	C1608C0G1H221J080AA
11	1	C44	10 uFd 20v	CAP TANT 10UF 20V 10% 1210	AVX Corporation	TAJB106K020RNJ
12	5	C45,C46,C51,C52,C55	10uF	CAP CER 10UF 6.3V 10% JB 0603	TDK Corporation	C1608JB0J106K080AB
13	1	C63	10000 pFd	CAP CER 10000PF 50V 5% NP0 0603	Murata	GRM1885C1H103JA01D
14	2	C64,C78	2200 pFd	CAP CER 2200PF 50V 5% NP0 0603	Murata	GRM1885C1H222JA01D
15	10	D1,D2,D3,D4,D5,D6, D7,D8,D9,D10	MBR0520L	DIODE SCHOTTKY 0.5A 20V SOD-123	Fairchild Semiconductor	MBR0520L
16	2	F1,F2	FUSE	PTC RESETTABLE .50A 15V 1812	Bourns	MF-MSMF050-2
17	3	J1, J17, J20	8x1 RECP	CONN HEADER FEMALE 8POS .1" GOLD	Protectron Electromech	P9401-08-21
18	1	J2	6x2 RECP	CONN HEADER FMAL 12PS.1" DL GOLD	Protectron Electromech	P9403-12-21
19	1	J3	10x2 RECP	CONN HEADER FMAL 20PS.1" DL GOLD	Protectron Electromech	P9403-20-21
20	2	J4,J8	8X2 RECP	CONN HEADER FMAL 16PS.1" DL GOLD	Protectron Electromech	P9403-16-21
21	1	J9	3p_jumper	CONN HEADR BRKWAY .100 3POS STR	Protectron Electromech	P9101-03-12-1
22	2	J10,J11	USB MINI B	MINI USB RCPT R/A DIP	TE Connectivity	1734510-1
23	1	J14	2X2 RECP	CONN HEADER .100 DUAL STR 4POS	Protectron Electromech	P9103-04-12-1
24	1	J16	JACK-EAR-PHONE-ACI	3.5mm SMD EARPHONE JACK,(TSH-3756)	CUI	SJ-43516-SMT
25	2	J21,J22	Prox CON	CONN RCPT 1POS .100" SNGL HORZ	Samtec Inc	BCS-101-L-S-HE

Item	Qty	Reference	Value	Description	Manufacturer	Mfr Part Number
26	1	J6	50MIL KEYED SMD	CONN HEADER 10 PIN 50MIL KEYED SMD	FCI	20021521-00010T1LF
27	1	J18	6x1 RECP	CONN HEADER FMAL 6PS.1" DL GOLD	Sullins Connector Solutions	PPTC061LFBN-RC
28	1	J19	10x1 RECP	CONN HEADER FMAL 10PS.1" DL GOLD	Sullins Connector Solutions	PPTC101LFBN-RC
29	1	LED1	Status LED Green	LED GREEN CLEAR 0805 SMD	Chicago Miniature	CMD17-21VGC/TR8
30	1	LED2	Power LED Amber	LED 595NM AMB DIFF 0805 SMD	Avago Technologies	HSMA-C170
31	1	LED3	RGB LED	LED RED/GREEN/BLUE PLCC4 SMD	Cree, Inc.	CLV1A-FKB-CJ1M1F1BB7R4S3
32	4	L1,L2,L3,L4	330 OHM @ 100MHz	FILTER CHIP 330 OHM 1.5A 0603	Murata Electronics	BLM18SG331TN1D
33	1	R1	820 ohm	RES 820 OHM 1/8W 5% 0805 SMD	Panasonic - ECG	ERJ-6GEYJ821V
34	1	R2	560 ohm	RES 560 OHM 1/8W 5% 0805 SMD	Panasonic - ECG	ERJ-6GEYJ561V
35	29	R3,R4,R22,R31,R33,R37,R38,R48,R49,R51,R63,R64,R66,R67,R69,R70,R71,R72,R75,R78,R85,R89,R91,R92,R95,R111,R112,R113,R114	ZERO	RES 0.0 OHM 1/10W 0603 SMD	Panasonic - ECG	ERJ-3GEY0R00V
36	1	R5	120 ohm	RES 120 OHM 1/10W 1% 0603 SMD	Panasonic - ECG	ERJ-3EKF1200V
37	13	R6,R8,R13,R15,R18,R19,R21,R24,R26,R27,R28,R32,R35	560 ohm	RES 560 OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ561V
38	4	R7,R20,R45,R73	ZERO	RES 0.0 OHM 1/8W 0805 SMD	Panasonic-ECG	ERJ-6GEY0R00V
39	2	R9,R14	15K	RES 15K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ153V
40	1	R10	220 ohm	RES SMD 220 OHM 1% 1/10W 0603	Panasonic - ECG	ERJ-3EKF2200V
41	2	R16,R17	22E	RES 22 OHM 1/10W 1% 0603 SMD	Panasonic - ECG	ERJ-3EKF22R0V
42	1	R23	ZERO	RES SMD 0.0 OHM JUMPER 1/10W 0402	Panasonic - ECG	ERJ-2GE0R00X
43	1	R25	1M	RES SMD 1M OHM 5% 1/10W 0402	Panasonic - ECG	ERJ-2GEJ105X
44	2	R40,R46	100K	RES 100K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEJ104X, ERJ-3GEYJ104V
45	1	R41	10K	RES 10K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ103V
46	1	R43	100 ohm	RES 100 OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ101V
47	7	R47,R52,R87,R57,R58,R98,R99	2.2K	RES 2.2K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ222V
48	5	R50,R76,R77,R90,R93	47K	RES 47K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ473V
49	1	R54	680 ohm	RES 680 OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ681V
50	1	R55	330 ohm	RES 330 OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ331V
51	2	R60,R65	30K	RES 30K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ303V
52	2	R86,R88	1.5K	RES 1.5K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ152V
53	2	SW1,SW2	SW PUSH-BUTTON	SWITCH TACTILE SPST-NO 0.05A 12V	Panasonic - ECG	EVQ-PE105K
54	2	TP1,TP14	BLACK	TEST POINT PC MINI .040"D Black	Keystone Electronics	5001
55	2	U1,U7	PUSBM12VX4-TL,115	TVS DIODE 5.5VWM 12VC 6HXSON	NXP Semiconductors	PUSBM12VX4-TL,115
56	1	U2	NCP1117DTARKG	IC REG LDO ADJ 1A DPAK	ON Semiconductor	NCP1117DTARKG

Item	Qty	Reference	Value	Description	Manufacturer	Mfr Part Number
57	1	U3	WM8731CLSEFL/R	IC CODEC PORTABLE INTERNET 28QFN	Cirrus Logic Inc	WM8731CLSEFL/R
58	1	U4	PSoC 5LP	68QFN PSoC 5LP chip for USB debug channel and USB-Serial interface	Cypress Semiconductor	CY8C5868LTI-LP039
59	1	U5	PSoC 4200L	124uBGA PSoC 4200L	Cypress Semiconductor	CY8C4248BZI-L489
60	1	U6	F-RAM	F-RAM 2-Mbit (256K X 8) I2C interface	Cypress Semiconductor	FM24V10-G
61	1	U9	DUAL PMOS	MOSFET 2P-CH 20V 430MA SOT-563	ON Semiconductor	NTZD3152PT1G
62	1	U10	TXS0104ERGYR	IC 4BIT NON-INV TRANSLTR 14-QFN	Texas Instruments	TXS0104ERGYR
63	1	U11	TXS0108ERGYR	IC XLATR VOLT-LVL 8B BDIR 20QFN	Texas Instruments	TXS0108ERGYR
64	1	Y1	32.768KHz	CRYSTAL 32.768KHZ 12.5PF SMD	ECS Inc	ECS-.327-12.5-39-TR
65	1	Y2	17.2032 MHz	CRYSTAL 17.20320MHz	IQD Frequency	LFXTAL063075
66	1	TVS2	5V 350W	TVS UNIDIR 350W 5V SOD-323	Dioded Inc.	SD05-7
Install on Bottom of PCB As per the Silk Screen in the Corners						
67	6	N/A	N/A	BUMPON CYLINDRICAL .312X.215 BLACK	3M	SJ61A6
Special Jumper Installation Instructions						
68	1	J9A	Install jumper across pins 1 and 2	Rectangular Connectors MINI JUMPER GF 6.0MM CLOSE TYPE BLACK	Kobiconn	151-8010-E
69	1	J14A	Install jumper across pins 3 and 4	Rectangular Connectors MINI JUMPER GF 6.0MM CLOSE TYPE BLACK	Kobiconn	151-8010-E
Label						
70	1	N/A	N/A	LBL, PCA Label, Vendor Code, Datecode, Serial Number 121-60244-01 Rev 01 (YYWWVVXXXXX)	Cypress Semiconductor	
No load components						
71	4	C4,C27,C95,C97	0.1 uFd	CAP .1UF 16V CERAMIC Y5V 0402	TDK Corporation	C1005X5R1A104K050BA
72	2	C10,C83	1.0 uFd	CAP CERAMIC 1.0UF 25V X5R 0603 10%	Taiyo Yuden	TMK107BJ105KA-T
73	4	C28,C80,C81,C82	470 pF	CAP CER 470PF 50V 5% C0G 0603	TDK Corporation	C1608C0G1H471J080AA
74	1	C79	10000 pFd	CAP CER 10000PF 50V 5% NP0 0603	Murata	GRM1885C1H103JA01D
75	3	C96,C98,C84	10uF	CAP CER 10UF 6.3V 10% JB 0603	TDK Corporation	C1608JB0J106K080AB
76	1	C90	2200 pFd	CAP CER 2200PF 50V 5% NP0 0603	Murata	GRM1885C1H222JA01D
77	3	C85,C93,C94	100uF/1206	CAP CER 100UF 6.3V 20% X5R 1206	Murata ENA	GRM31CR60J107ME39K
78	1	D11	1SS417(TL3,T)	DIODE SCHOTTKY 40V 100MA FSC	Toshiba	1SS417(TL3,T)
79	2	J7	50MIL KEYED SMD	CONN HEADER 10 PIN 50MIL KEYED SMD	Samtec	FTSH-105-01-L-DV-K
80	1	J5	6X1 RECP RA	CONN FEMALE 6POS .100" R/A GOLD	Sullins Connector Solutions	PPPC061LGBN-RC
81	2	J12,J24	3x2 RECPT	CONN HEADER MALE 6PS .1" GOLD	Protectron Electromech	P9403-06-21
82	2	J13,J28	2 PIN HDR	CONN HEADER MALE 2POS .1" GOLD	Protectron Electromech	P9401-02-21

Item	Qty	Reference	Value	Description	Manufacturer	Mfr Part Number
83	1	J23	16X1 RECP	CONN HEADER FMAL 16PS.1" GOLD	Sullins Connector Solutions	PPPC161LFBN-RC
84	1	J25	uSD Card Conn	CONN MICRO SD CARD PUSH-PUSH R/A	Molex Inc	5027740891
85	1	J26	CON 26x2	CONN FEMALE 52POS DL .1" R/A GOLD	Sullins Connector Solutions	PPPC262LJBN-RC
86	2	J27	HDR 26x2	CONN HEADER .100 DUAL R/A 52POS	Sullins Connector Solutions	PRPC026DBAN-M71RC
87	3	J17A, J20A	8x1 RECP_HDR	HDR, 2.54mm PITCH, SINGLE ROW, 8POS, 13mm, MATING LENGTH	Protectron Electromech	P9101-08-D32-1
88	1	J18A	6x1 RECP_HDR	HDR, 2.54mm PITCH, SINGLE ROW, 6POS, 13mm, MATING LENGTH	Protectron Electromech	P9101-06-D32-1
89	1	J19A	10x1 RECP_HDR	HDR, 2.54mm PITCH, SINGLE ROW, 10POS, 13mm, MATING LENGTH	Protectron Electromech	P9101-10-D32-1
90	2	L5,L6	330 OHM @ 100MHz	FILTER CHIP 330 OHM 1.5A 0603	Murata Electronics	BLM18SG331TN1D
91	20	R11,R12,R30,R34,R36,R53,R59,R61,R62,R68,R80,R82,R94,R96,R97,R103,R106,R107,R108,R109	Zero Ohm	RES 0.0 OHM 1/10W 0603 SMD	Panasonic - ECG	ERJ-3GEY0R00V
92	1	R39	10K POT	TRIMMER 10K OHM 0.5W PC PIN	Bourns Inc.	3362P-1-103LF
93	5	R42,R44,R74,R84,R110	ZERO	RES 0.0 OHM 1/8W 0805 SMD	Panasonic-ECG	ERJ-6GEY0R00V
94	2	R56,R81	4.7K	RES 4.7K OHM 1/10W 5% 0603 SMD	Panasonic - ECG	ERJ-3GEYJ472V
95	2	R100,R102	6.8M	RES ZERO OHM 1/10W 0603 SMD	Vishay Dale	CRCW06036M80FKEA
96	1	R101	4.7M	RES ZERO OHM 1/10W 0603 SMD	Toshiba	RC0603FR-074M7L
97	2	SW3, SW4	DPDT SW	SWITCH SLIDE DPDT 200MA 12V	Copal Electronics	CL-SB-22B-02T
98	1	SW5	SW PUSH-BUTTON	SWITCH TACTILE SPST-NO 0.05A 12V	Panasonic - ECG	EVQ-PE105K
99	2	TP11,TP13	BLACK	TEST POINT 43 HOLE 65 PLATED BLACK	Keystone Electronics	5001
100	32	TP2,TP3,TP4,TP5,TP6,TP7,TP8,TP9,TP10,TP12,TP15,TP16,TP17,TP18,TP19,TP20,TP21,TP22,TP23,TP24,TP25,TP26,TP27,TP28,TP30,TP31,TP32,TP33,TP34,TP35,TP36,TP37	RED	TEST POINT 43 HOLE 65 PLATED RED	Keystone Electronics	5000
101	1	TVS1	5V 350W	TVS UNIDIR 350W 5V SOD-323	Dioded Inc.	SD05-7
102	1	U8	EZ-BLE	EZ-BLE PRoC BLE Module, 10 mm X 10 mm X 1.8 mm	Cypress Semiconductor	CYBLE-022001-00
103	1	U12	S6AE101A	IC ENERGY HARVESTER SON10	Cypress Semiconductor	S6AE101A
104	1	U13	S25FL512S	SPI Nor Flash	Cypress Semiconductor	S25FL512S
105	1	V1	BHX1-1225-SM	HOLDER COIN CELL FOR CR1225 SMD	MPD	BHX1-1225-SM
106	1	R79	1 ohm	RES SMD 1 OHM 1% 1/4W 1206	Stackpole Electronics Inc	RMCF1206FT1R00
107	1	R83	10 ohm	RES SMD 10 OHM 1% 1/4W 1206	Stackpole Electronics Inc	RMCF1206FT10R0

Revision History

CY8CKIT-046 PSoC® 4 L-Series Pioneer Kit Guide Revision History

Document Title: CY8CKIT-046 PSoC® 4 L-Series Pioneer Kit Guide			
Document Number: 002-03344			
Revision	**Issue Date**	**Origin of Change**	**Description of Change**
**	09/24/2015	MSUR	Initial version of kit guide
*A	12/17/2015	MSUR	Added the following sections: "CapSense Buttons" on page 36, "USB Audio" on page 43, and "Using UART for Data Monitoring" on page 30. Modified "Board Separation (Snapping)" on page 71. Updated the schematics, BOM, and board images