CY8CKIT-062-BLE

PSoC® 6 BLE Pioneer Kit Guide

Doc. # 002-17040 Rev. *B

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone (USA): 800.858.1810
Phone (Intnl): +1.408.943.2600
www.cypress.com

Copyrights

CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and Company shall and hereby does release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. Company shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.

F-RAM, Programmable System-on-Chip, and PSoC Creator are trademarks, and PSoC and CapSense are registered trademarks of Cypress Semiconductor Corporation.

Contents

Safety Information

Regulatory Compliance

The CY8CKIT-062-BLE PSoC 6 BLE 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 6 BLE Pioneer Kit contains electrostatic discharge (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 PSoC 6 BLE Pioneer kit in the protective shipping package.

End-of-Life/Product Recycling

The end-of life for this kit is five years from the date of manufacture mentioned as bar code 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 an ESD workstation is unavailable, use appropriate ESD protection by wearing an anti-static wrist strap attached to a grounded metal object.

Handling Boards

PSoC 6 BLE Pioneer Kit is 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-062-BLE PSoC 6 BLE Pioneer Kit. The PSoC 6 BLE Pioneer Kit enables you to evaluate and develop your applications using the PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity (hereafter called "PSoC 6 MCU").

PSoC 6 MCU is Cypress' latest, ultra-low-power PSoC specifically designed for wearables and IoT products. PSoC 6 MCU is a true programmable embedded system-on-chip, integrating a 150-MHz ARM® Cortex®-M4 as the primary application processor, a 100-MHz ARM Cortex®-M0+ that supports low-power operations, up to 1 MB Flash and 288 KB SRAM, an integrated BLE 4.2 radio, CapSense® touch-sensing, and programmable analog and digital peripherals that allow higher flexibility, in-field tuning of the design, and faster time-to-market.

The PSoC 6 BLE Pioneer board offers compatibility with Arduino™ shields. The board features a PSoC 6 MCU, a 512-Mb NOR flash, onboard programmer/debugger (KitProg2), USB Type-C power delivery system (EZ-PD™ CCG3), 5-segment CapSense slider, two CapSense buttons, one CapSense proximity sensing header, an RGB LED, two user LEDs, and one push button. The board supports operating voltages from 1.8 V to 3.3 V for PSoC 6 MCU.

The CY8CKIT-062-BLE package includes a CY8CKIT-028-EPD E-INK Display Shield that contains a 2.7-inch E-INK display, a motion sensor, a thermistor, and a PDM microphone. The kit package also contains a CY5677 CySmart BLE 4.2 USB Dongle that is factory-programmed to emulate a BLE GAP Central device, enabling you to emulate a BLE host on your computer.

You can use PSoC Creator™ to develop and debug your PSoC 6 MCU projects. PSoC Creator is Cypress' standard integrated design environment (IDE). PSoC Creator also supports exporting your designs to other third party firmware development tools.

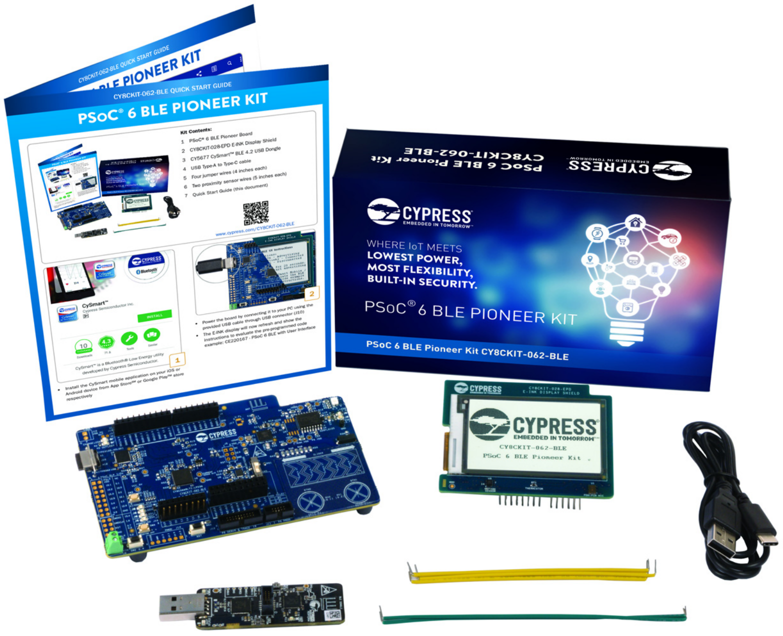
If you are new to PSoC 6 MCU and PSoC Creator IDE, you can find introductions in the application note AN210781 - Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity.

1.1 Kit Contents

The CY8CKIT-062-BLE package has the following contents, as shown in Figure 1-1.

- PSoC 6 BLE Pioneer Board
- CY8CKIT-028-EPD E-INK Display Shield
- CY5677 CySmart BLE 4.2 USB Dongle
- USB Type-A to Type-C cable
- Four jumper wires (4 inches each)
- Two proximity sensor wires (5 inches each)
- Quick Start Guide

Figure 1-1. Kit Contents



Inspect the contents of the kit; if you find any part missing, contact your nearest Cypress sales office for help: www.cypress.com/support.

1.2 Board Details

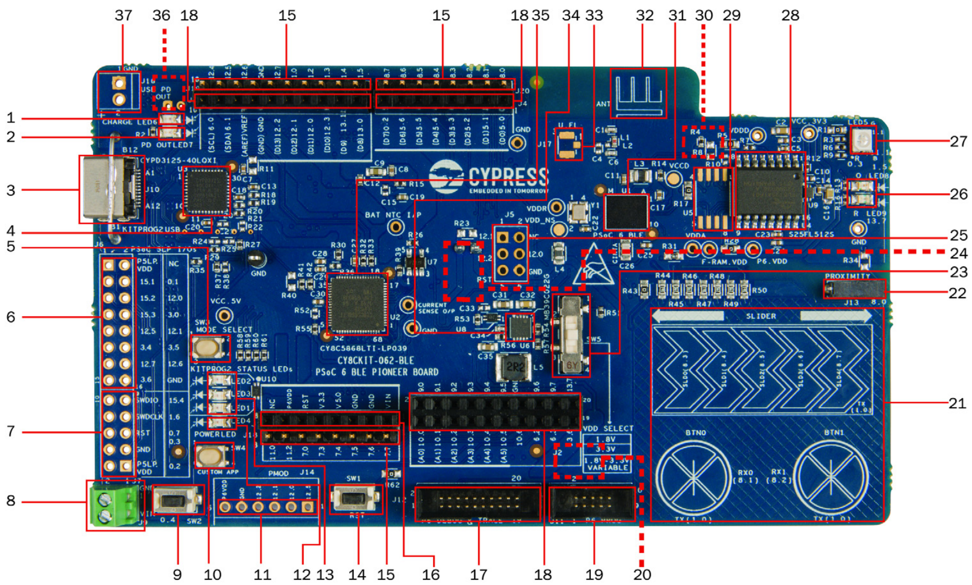
Figure 1-2 shows the Pioneer board that has the following features:

- PSoC 6 MCU with BLE connectivity
- Expansion headers that are compatible with Arduino Uno™ 3.3 V shields[1] and Digilent® Pmod™ modules
- 512-Mbit external quad-SPI NOR Flash that provides a fast, expandable memory for data and code
- KitProg2 onboard programmer/debugger with mass storage programming, USB to UART/I2C/SPI bridge functionality, and custom applications support
- EZ-PD CCG3 USB Type-C power delivery (PD) system with rechargeable lithium-ion polymer (Li-Po) battery support[2]
- CapSense touch-sensing slider (5 elements), two buttons, all of which are capable of both self-capacitance (CSD) and mutual-capacitance (CSX) operation, and a CSD proximity sensor that let you evaluate Cypress' fourth-generation CapSense technology
- 1.8 V to 3.3 V operation of PSoC 6 MCU is supported. An additional 330 mF super-capacitor is provided for backup domain supply (Vbackup)
- Two user LEDs, an RGB LED, a user button, and a reset button for PSoC 6 MCU. Two buttons and three LEDs for KitProg2

[1] 5V shields are not supported

[2] Battery and power-delivery capable USB Type-C to Type-C cable are not included in the kit package and should be purchased separately.

Figure 1-2. Pioneer Board



1. Battery charging indicator (LED6)
2. USB PD output voltage availability indicator (LED7)
3. KitProg2 USB connector (J10)
4. Cypress EZ-PD™ CCG3 Type-C Port Controller with PD (CYPD3125-40LQXI, U3)
5. KitProg2 programming mode selection button (SW3)
6. KitProg2 I/O header (J6)[1]
7. KitProg2 programming/custom application header (J7)[1]
8. External power supply connector (J9)
9. PSoC 6 BLE user button (SW2)
10. KitProg2 application selection button (SW4)
11. Digilent® Pmod™ compatible I/O header (J14)[1]
12. Power LED (LED4)
13. KitProg2 status LEDs (LED1, LED2, and LED3)
14. PSoC 6 BLE reset button (SW1)
15. PSoC 6 BLE I/O header (J18, J19 and J20)
16. Arduino™ Uno R3 compatible power header (J1)
17. PSoC 6 BLE debug and trace header (J12)
18. Arduino™ Uno R3 compatible PSoC 6 BLE I/O header (J2, J3 and J4)
19. PSoC 6 BLE program and debug header (J11)
20. KitProg2 programming target selection switch (SW6)[2]
21. CapSense slider and buttons
22. CapSense proximity header (J13)
23. PSoC 6 BLE VDD selection switch (SW5)
24. PSoC 6 BLE power monitoring jumper (J8)[2]
25. Arduino™ Uno R3 compatible ICSP header (J5)[1]
26. PSoC 6 BLE user LEDs (LED8 and LED9)
27. RGB LED (LED5)
28. Cypress 512-Mbit serial NOR flash memory (S25FL512S, U4)
29. Cypress serial Ferroelectric RAM (U5)[1]
30. Vbackup and PMIC control selection switch (SW7)
31. Cypress PSoC 6 BLE (CY8C6347BZI-BLD53, U1)
32. BLE antenna
33. U.FL connector for external antenna (J17)[1]
34. Cypress main voltage regulator (MB39C022G, U6)
35. KitProg2 (PSoC 5LP) programmer and debugger (CY8C5868LTI-LP039, U2)
36. Battery connector (J15)[1,2]
37. USB PD output voltage (9V/12V) connector (J16)[1]

[1] Footprints only, not populated on the board

[2] Components at the bottom side of the board

Table 1-1. Selection Switches in the Pioneer Board

Switch	Location on the board	Purpose	Default Position
SW5	front	Selects the V_{DD} supply of PSoC 6 MCU between 1.8 V, 3.3 V, and variable 1.8 V to 3.3 V that is controlled by KitProg2.	3.3V
SW6	back	In the “PSoC 6 MCU” position: ■ KitProg2 can program the onboard PSoC 6 MCU ■ In addition, the PSoC 6 MCU can be programmed by an external programmer such as a MiniProg3 connected to J11 Avoid connecting any external devices to J11 in the “PSoC 6 MCU” position, as it can cause programming failure. In the “External Device” position: ■ KitProg2 can program any PSoC 4/5/6 devices connected to J11	PSoC 6 MCU
SW7	back	Selects the Vbackup supply connection of PSoC 6 MCU between V_{DDD} and the super-capacitor. When V_{DDD} is selected, the regulator can be turned ON/OFF by the KitProg2. When the super-capacitor is selected, PSoC 6 MCU can turn the regulator ON/OFF.	V_{DDD} / KitProg2

Figure 1-3 shows the pinout of the Pioneer board.

Figure 1-3. Pioneer board Pinout

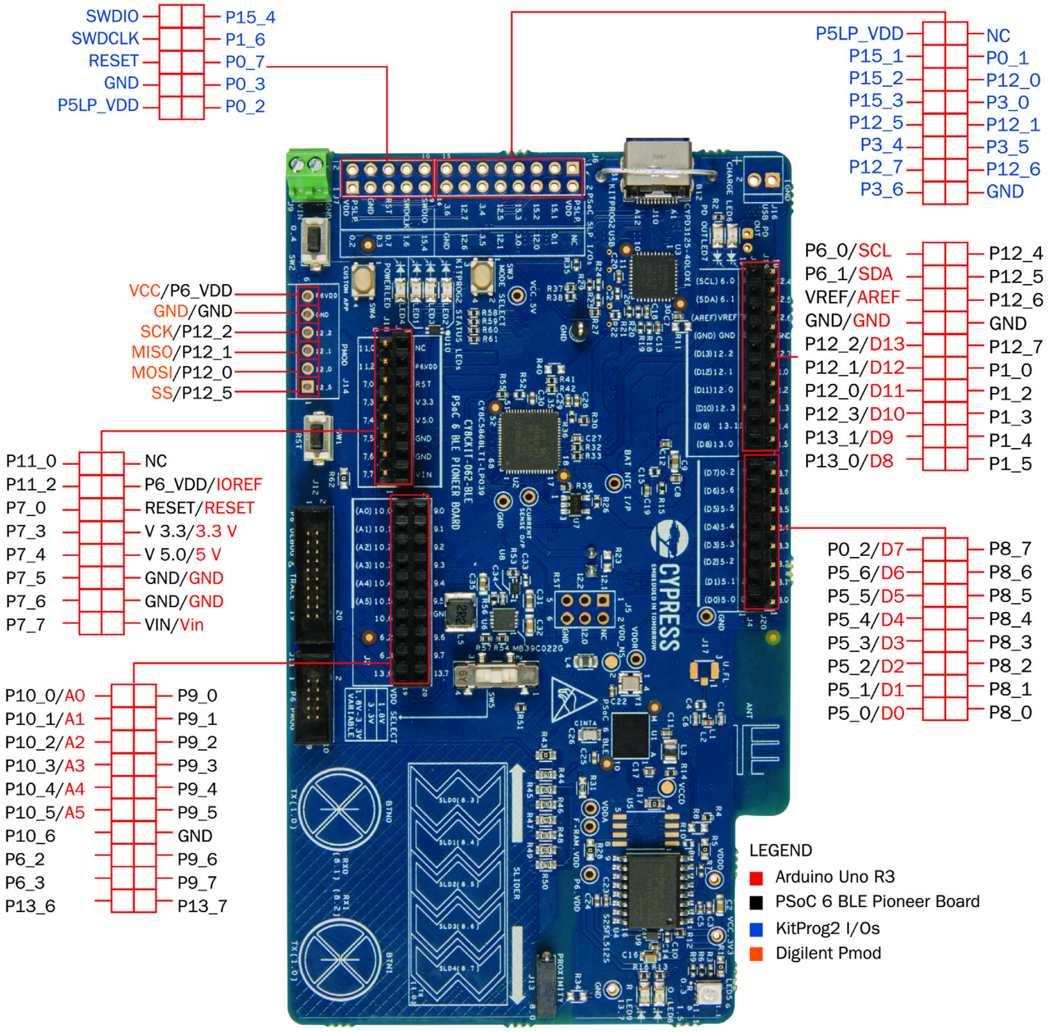


Table 1-2. Pioneer board Pinout

PSoC 6 Pin	Primary On-board Function	Secondary On-board Function	Connection details
ANT	RFIO, Antenna	–	
XI	ECO IN	–	
XO	ECO OUT	–	
XRES	Reset	–	
P0.0	WCO IN	–	

Table 1-2. Pioneer board Pinout *(continued)*

PSoC 6 Pin	Primary On-board Function	Secondary On-board Function	Connection details
P0.1	WCO OUT	–	
P0.2	Arduino header J4.8, D7	–	
P0.3	RGB Red LED	–	
P0.4	User Button with Hibernate wakeup capability	–	
P0.5	PMIC control	–	
P1.0	CapSense Tx	GPIO on non-Arduino header (J19.5)	Connected to primary and secondary functions by default. Remove R101 to disconnect from header or remove R43 to disconnect from CapSense
P1.1	RGB Green LED	–	
P1.2	GPIO on non Arduino header (J19.4)	–	
P1.3	GPIO on non Arduino header (J19.3)	–	
P1.4	GPIO on non Arduino header (J19.2)	–	
P1.5	Orange User LED	GPIO on non-Arduino header (J19.1)	Connected to primary and secondary function by default. Remove R13 to disconnect from LED.
P5.0	Arduino J4.1, D0 UART RX KitProg2 UART TX	–	Remove R120 to disconnect from KitProg2 UART TX
P5.1	Arduino J4.2, D1 UART TX KitProg2 UART RX	–	Remove R119 to disconnect from KitProg2 UART RX
P5.2	Arduino J4.3, D2 UART RTS KitProg2 UART CTS	–	Remove R77 and load R78 to disconnect from KitProg2 UART CTS (This will also disconnect RTS and SPI lines from KitProg2
P5.3	Arduino J4.4, D3 UART CTS KitProg2 UART RTS	–	Remove R77 and load R78 to disconnect from KitProg2 UART CTS (This will also disconnect RTS and SPI lines from KitProg2
P5.4	Arduino J4.5, D4	–	
P5.5	Arduino J4.6, D5	–	
P5.6	Arduino J4.7, D6	–	
P6.0	Arduino J3.10, SCL KitProg2 I2C SCL	–	Remove R109 to disconnect from KitProg2 I2C SCL
P6.1	Arduino J3.9, SDA KitProg2 I2C SDA	–	Remove R114 to disconnect from KitProg2 I2C SDA
P6.2	GPIO on non Arduino header (J2.15)	–	

Table 1-2. Pioneer board Pinout *(continued)*

PSoC 6 Pin	Primary On-board Function	Secondary On-board Function	Connection details
P6.3	GPIO on non Arduino header (J2.17)	CapSense Shield	Remove R144 to disconnect from GND and populate R138 to connect to CapSense shield (hash pattern on board)
P6.4	TDO/SWO	–	
P6.5	TDI	–	
P6.6	TMS/SWDIO	–	Remove R147 to disconnect from KitProg2 SWDIO
P6.7	TCLK/SWCLK	–	Remove R146 to disconnect from KitProg2 SWCLK
P7.0	TRACECLK	GPIO on non-Arduino header (J18.6)	Populate R135 to connect to J18 header
P7.1	CINTA	–	
P7.2	CINTB	CSH	Remove C88 (0.47nF) and populate 10nF for CSH
P7.3	GPIO on non Arduino header (J18.5)	CSH	Remove R88 to disconnect from header and populate C88 (10nF) for CSH
P7.4	TRACEDATA[3]	GPIO on non-Arduino header (J18.4)	Populate R132 to connect to J18
P7.5	TRACEDATA[2]	GPIO on non-Arduino header (J18.3)	Populate R133 to connect to J18
P7.6	TRACEDATA[1]	GPIO on non-Arduino header (J18.2)	Populate R134 to connect to J18
P7.7	CMOD	GPIO on non-Arduino header (J18.1)	Populate R87 to connect to J18
P8.0	Proximity	GPIO on non-Arduino header (J20.1)	Replace R31 with Zero ohm and populate R34 to connect to header
P8.1	CapSense Button0 Rx	GPIO on non-Arduino header (J20.2)	Remove R44 to disconnect CapSense pad and populate R100 to connect to header
P8.2	CapSense Button1 Rx	GPIO on non-Arduino header (J20.3)	Remove R50 to disconnect CapSense pad and populate R103 to connect to header
P8.3	CapSense Silder0 Rx	GPIO on non-Arduino header (J20.4)	Remove R45 to disconnect CapSense pad and populate R99 to connect to header
P8.4	CapSense Silder1 Rx	GPIO on non-Arduino header (J20.5)	Remove R46 to disconnect CapSense pad and populate R97 to connect to header
P8.5	CapSense Silder2 Rx	GPIO on non-Arduino header (J20.6)	Remove R47 to disconnect CapSense pad and populate R105 to connect to header

Table 1-2. Pioneer board Pinout *(continued)*

PSoC 6 Pin	Primary On-board Function	Secondary On-board Function	Connection details
P8.6	CapSense Silder3 Rx	GPIO on non-Arduino header (J20.7)	Remove R48 to disconnect CapSense pad and populate R98 to connect to header
P8.7	CapSense Silder4 Rx	GPIO on non-Arduino header (J20.8)	Remove R49 to disconnect CapSense pad and populate R104 to connect to header
P9.0	GPIO on non Arduino header (J2.2)	–	
P9.1	GPIO on non Arduino header (J2.4)	–	
P9.2	GPIO on non Arduino header (J2.6)	–	
P9.3	TRACEDATA[0]	GPIO on non-Arduino header (J2.8)	Populate R131 to connect to header
P9.4	GPIO on non Arduino header (J2.10)	–	
P9.5	GPIO on non Arduino header (J2.12)	–	
P9.6	GPIO on non Arduino header (J2.16)	–	
P9.7	GPIO on non Arduino header (J2.18)	–	
P10.0	Arduino J2.1, A0	–	
P10.1	Arduino J2.3, A1	–	
P10.2	Arduino J2.5, A2	–	
P10.3	Arduino J2.7, A3	–	
P10.4	Arduino J2.9, A4 PDM_CLK	–	
P10.5	Arduino J2.11, A5 PDM_DAT	–	
P10.6	GPIO on non Arduino header (J2.13)	–	
P11.0	FRAM CS	GPIO on non-Arduino header (J18.8)	Connected to primary and secondary function by default. Remove R17 to disconnect from J18 and load R10 (10K) as FRAM pull-up.
P11.1	RGB Blue LED	–	
P11.2	QSPI FLASH CS	GPIO on non-Arduino header (J18.7)	Populate R8 to connect to J18, remove R5 and R7 to disconnect from Flash and pull-up.
P11.3	QSPI FLASH/ FRAM DATA3	–	

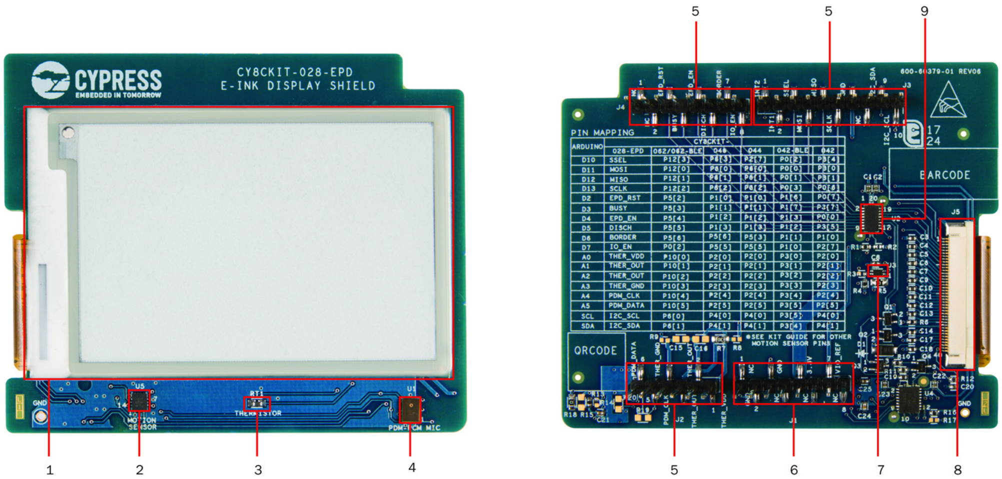
Table 1-2. Pioneer board Pinout *(continued)*

PSoC 6 Pin	Primary On-board Function	Secondary On-board Function	Connection details
P11.4	QSPI FLASH/ FRAM DATA2	–	
P11.5	QSPI FLASH/ FRAM DATA1	–	
P11.6	QSPI FLASH/ FRAM DATA0	–	
P11.7	QSPI FLASH/ FRAM CLK	–	
P12.0	Arduino J3.4, D11 SPI MOSI	–	Remove R77 and load R78 to disconnect from Kit-Prog2_SPI lines (This will also disconnect RTS and CTS lines from KitProg2)
P12.1	Arduino J3.5, D12 SPI MISO	–	Remove R77 and load R78 to disconnect from Kit-Prog2_SPI lines (This will also disconnect RTS and CTS lines from KitProg3)
P12.2	Arduino J3.6, D13 SPI CLK	–	Remove R77 and load R78 to disconnect from Kit-Prog2_SPI lines (This will also disconnect RTS and CTS lines from KitProg4)
P12.3	Arduino J3.3, D10 SPI SELECT	–	Remove R77 and load R78 to disconnect from Kit-Prog2_SPI lines (This will also disconnect RTS and CTS lines from KitProg5)
P12.4	KitProg2 SPI SELECT	GPIO on non-Arduino header (J19.10)	Connected to primary and secondary function by default. Remove R81 to disconnect from J19 or remove R84 to disconnect KitProg2_SPI_SELECT
P12.5	PMOD SPI SELECT	GPIO on non-Arduino header (J19.9)	Connected to primary and secondary function by default. Remove R71 to disconnect from J19 or remove R82 to disconnect PMOD_SPI_SELECT
P12.6	GPIO on non Arduino header (J19.7)	–	
P12.7	GPIO on non Arduino header (J19.5)	–	
P13.0	Arduino J3.1, D8	–	
P13.1	Arduino J3.2, D9	–	
P13.6	GPIO on non Arduino header (J2.19)	CapSense Shield	Remove R144 to disconnect from GND and populate R137 to connect to CapSense shield (hash pattern on board)
P13.7	Red User LED	GPIO on non-Arduino header (J2.20)	Connected to primary and secondary function by default. Remove R16 to disconnect from LED.
VREF	SAR BYPASS, J3.8, AREF	–	

Figure 1-4 shows the E-INK display shield that has the following features:

- A 2.7 inch monochrome E-INK display with a resolution of 264x176. The E-INK display can retain its contents even in the absence of power, which provides an ultra low-power, “always-on” display functionality
- A thermistor that allows temperature-compensation of the display as well as general purpose temperature measurement.
- A 3-axis acceleration and 3-axis gyroscopic motion sensor
- A PDM microphone for voice input
- An I/O level translator that allows the board to operate at any voltage between 1.8 V and 1.8 V ~ 5 V, by providing a constant 3.3 V interface to the E-INK display
- A load switch that can be controlled by the board to toggle the E-INK display’s power.

Figure 1-4. E-INK Display Shield



1. 2.7- inch E-INK display
2. Motion Sensor (U5)
3. Thermistor (RT1)
4. PDM microphone (U1)
5. Arduino compatible I/O headers (J2, J3 and J4)
6. Arduino compatible power and I/O header (J1)
7. E-INK display power control load switch (U3)
8. E-INK display connector (J5)
9. E-INK display I/O voltage translator (U2)

Figure 1-5 shows the pinout of the E-INK display shield.

Figure 1-5. E-INK Shield Pinout

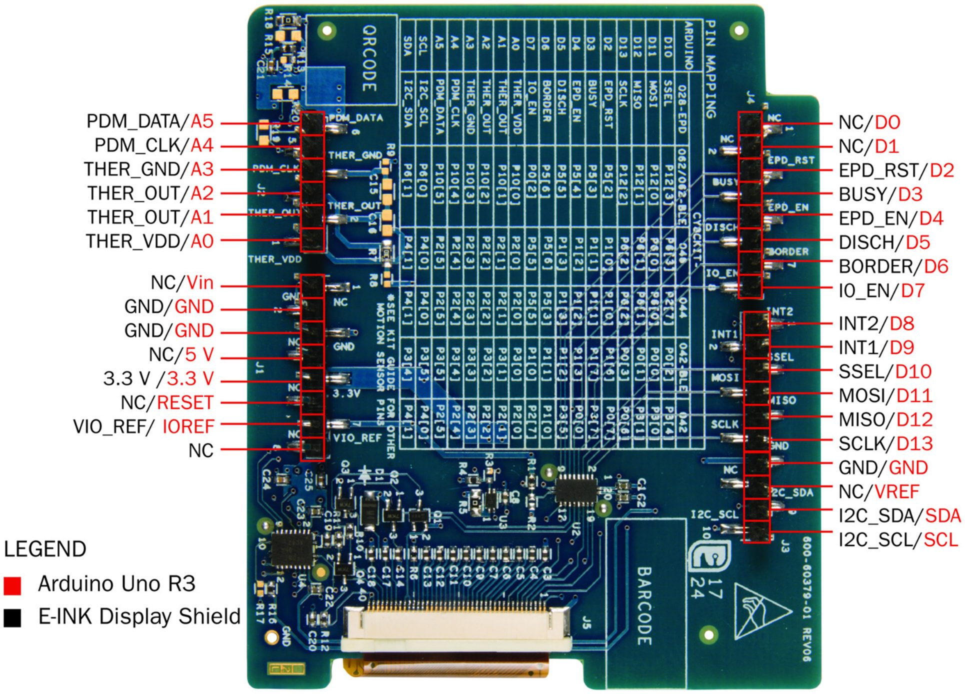


Table 1-3. E-INK Shield Pinout

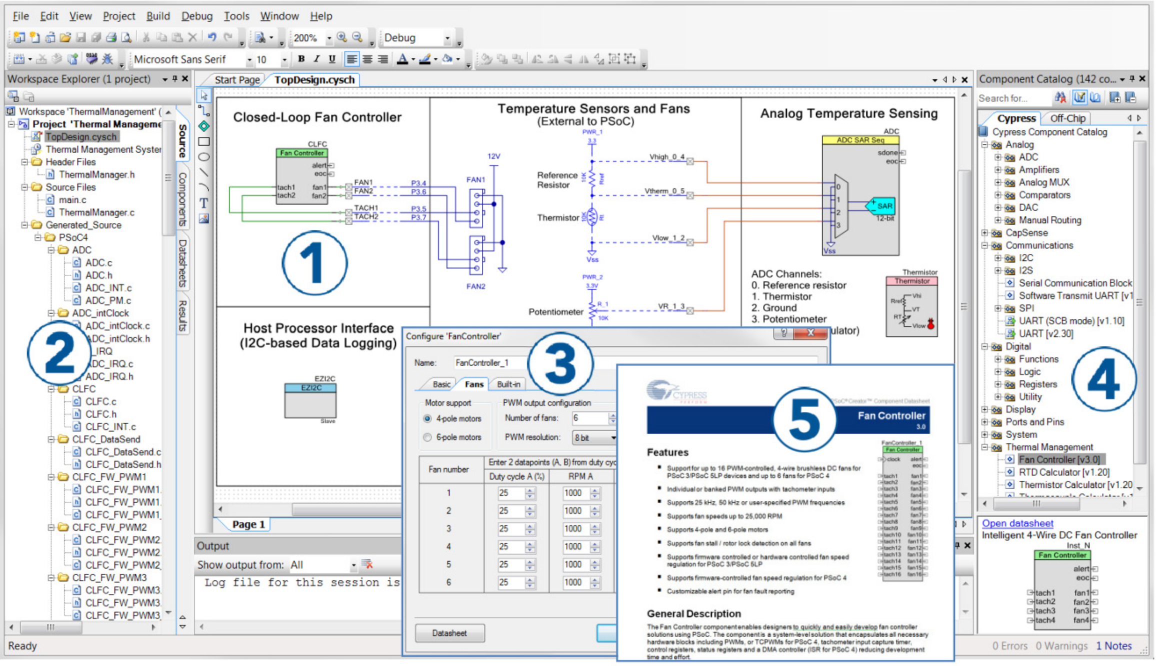
Pin #	Arduino Pin	Arduino Function	E-INK Shield Function	Pioneer Board Connection
J3.1	D8	DIGITAL I/O	IMU INT1	P13[0]
J3.2	D9	PWM	IMU INT2	P13[1]
J3.3	D10	SS/PWM	SSEL	P12[3]
J3.4	D11	MOSI/PWM	MOSI	P12[0]
J3.5	D12	MISO	MISO	P12[1]
J3.6	D13	SCK	SCLK	P12[2]
J3.7	D14	GND	GND	GND
J3.8	D15	analog ref i/p	NC	VREF
J3.9	SCL	SDA	IMU I2C SDA	P6[1]
J3.10	SDA	SCL	IMU I2C SCL	P6[0]
J4.1	D0	RX	NC	NC
J4.2	D1	TX	NC	NC
J4.3	D2	DIGITAL I/O	EPD reset	P5[2]
J4.4	D3	PWM, I/O	BUSY	P5[3]
J4.5	D4	DIGITAL I/O	EPD enable	P5[4]
J4.6	D5	PWM, I/O	DISCHARGE	P5[5]
J4.7	D6	PWM, I/O	BORDER	P5[6]
J4.8	D7	DIGITAL I/O	EPD I/O enable	P0[2]
J2.1	A0	ADC0	THERM VDD	P10[0]
J2.2	A1	ADC1	THERM OUT	P10[1]
J2.3	A2	ADC2	THERM OUT	P10[2]
J2.4	A3	ADC3	THERM GND	P10[3]
J2.5	A4	ADC4	PDM CLK	P10[4]
J2.6	A5	ADC5	PDM DATA	P10[5]

1.3 PSoC Creator

PSoC Creator is a state-of-the-art, easy-to-use IDE. It uses revolutionary hardware and software co-design, powered by a library of fully verified and characterized PSoC Components™ and peripheral driver libraries (PDL), as shown in Figure 1-6. 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 configuration tools or PDL.
4. Explore the library of 100+ Components.
5. Access Component datasheets.
6. Export your design to third-party firmware development tools.

Figure 1-6. PSoC Creator Features


File Edit View Project Build Debug Tools Window Help
Workspace Explorer (1 project)
Start Page
TopDesign.cysch
Component Catalog
Closed-Loop Fan Controller
Temperature Sensors and Fans (External to PSoC)
Analog Temperature Sensing
Host Processor Interface (I2C-based Data Logging)
Configure 'FanController'
Fan Controller
Features
General Description
Output
Ready
1
2
3
4
5

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

1.3.1 PSoC Creator Code Examples

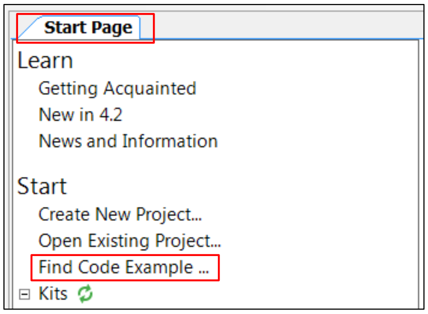
PSoC Creator includes a large number of code examples. These examples are accessible from the PSoC Creator Start Page, as shown in Figure 1-7 or from the menu **File** > **Code Example**.

Code examples can speed up your design process by starting you off with a complete design. The code examples also show how to use PSoC Creator Components for various applications. Code examples and documentation are included.

In the **Find Code Example** dialog, 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 project 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 project, 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 1-7. Code Examples in PSoC Creator



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. For a list of code examples that you can use on this kit, see Code Examples chapter on page 38.

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 projects.
- **Simple Component Code Examples:** Choose **File** > **Code Example**. These examples demonstrate how to configure and use PSoC Creator Components. To access examples related to a specific Component, right-click the Component in the schematic or in the Component Catalog. Select the **Find Code Example** option in the context menu that appears.
- **System Reference Guide:** Choose **Help** > **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**.

1.4 Getting Started

This guide will help you to get acquainted with the PSoC 6 BLE Pioneer Kit:

- The Software Installation chapter on page 23 describes the installation of the kit software. This includes the PSoC Creator IDE and PDL to develop and debug the applications, the PSoC Programmer to program the *.hex* files on to the device, and the CySmart to emulate a host device in the GAP central role.
- The Kit Operation chapter on page 26 describes the major features of the PSoC 6 BLE Pioneer Kit and functionalities such as programming, debugging, and the USB-UART and USB-I2C bridges.
- The Code Examples chapter on page 38 describes multiple PSoC 6 MCU code examples that will help you understand how to create your own PSoC 6 projects.
- The Appendix on page 41 provides a detailed hardware description, methods to use the onboard NOR Flash and onboard EZ-PD™ CCG3 Type-C power delivery system, kit schematics, and the bill of materials (BOM).

1.5 Additional Learning Resources

Cypress provides a wealth of data at www.cypress.com/psoc6 to help you to select the right PSoC device for your design and to help you to quickly and effectively integrate the device into your design.

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 Documentation Conventions

Table 1-4. 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 Project
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 1-5. Acronyms Used in this Document

Acronym	Definition
ADC	Analog-to-Digital Converter
BLE	Bluetooth Low Energy
BOM	Bill of Materials
CINT	Integration Capacitor
CMOD	Modulator Capacitor
CPU	Central Processing Unit
CSD	CapSense Sigma Delta
CTANK	Shield Tank Capacitor
DC	Direct Current
Del-Sig	Delta-Sigma

Table 1-5. Acronyms Used in this Document *(continued)*

Acronym	Definition
ECO	External Crystal Oscillator
EPD	Electronic Paper Display
ESD	Electrostatic Discharge
F-RAM	Ferroelectric Random Access Memory
FPC	Flexible Printed Circuit
GPIO	General-Purpose Input/Output
HID	Human Interface Device
I^2C	Inter-Integrated Circuit
IC	Integrated Circuit
ICSP	In-Circuit Serial Programming
IDAC	Current Digital-to-Analog Converter
IDE	Integrated Development Environment
LED	Light-emitting Diode
PC	Personal Computer
PD	Power Delivery
PDM	Pulse Density Modulation
PTC	Positive Temperature Coefficient
PSoC	Programmable System-on-Chip
PWM	Pulse Width Modulation
RGB	Red Green Blue
SAR	Successive Approximation Register
SPI	Serial Peripheral Interface
SRAM	Serial Random Access Memory
SWD	Serial Wire Debug
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
WCO	Watch Crystal Oscillator

2. Software Installation

This chapter describes the steps to install the software tools and packages on a PC for using the PSoC 6 BLE Pioneer Kit. This includes the IDE on which the projects will be built and used for programming.

2.1 Before You Begin

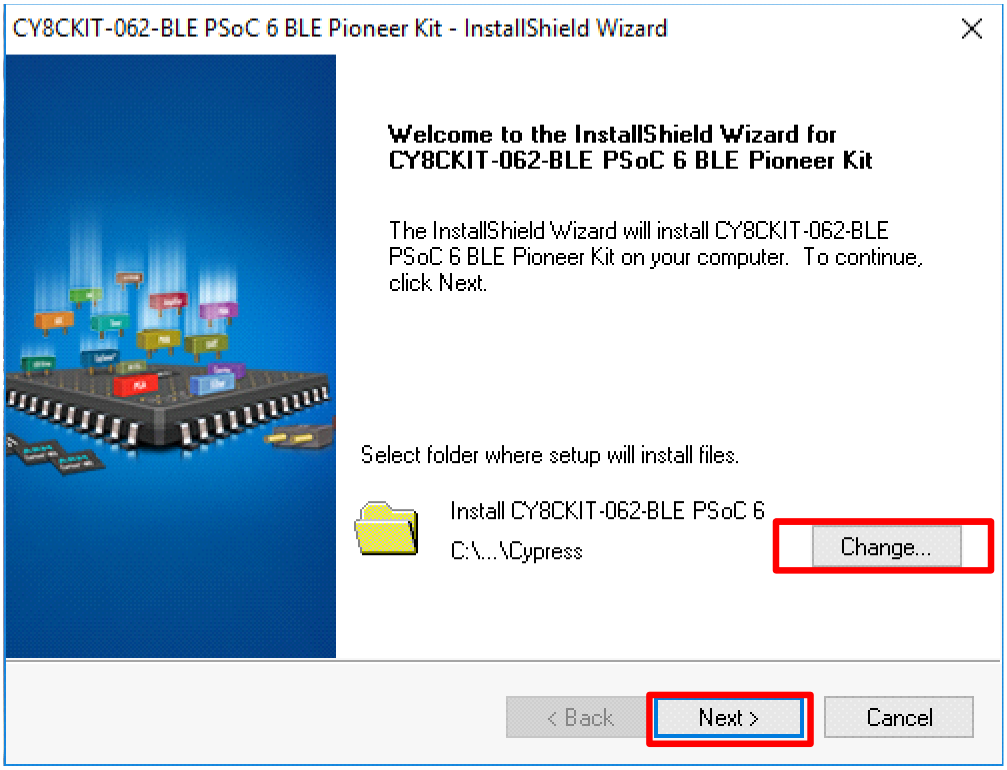
To install Cypress software, you will require administrator privileges. However, they are not required to run the software that is already installed. Before you install the kit software, close any other Cypress software that is currently running.

2.2 Install Software

Follow these steps to install the PSoC 6 BLE Pioneer Kit software:

1. Download and run the CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit software from www.cypress.com/CY8CKIT-062-BLE. The kit software is available in two different formats for download.
   a. **CY8CKIT-062-BLE Kit Complete Setup**: This installation package contains the files related to the Kit including PSoC Creator, PSoC Programmer, PDL, and CySmart. However, it does not include the Windows Installer or Microsoft .NET framework packages. If these packages are not on your computer, the installer will direct you to download and install them from the Internet.
   b. **CY8CKIT-062-BLE 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 PC.
2. Select the folder in which you want to install the PSoC 6 BLE Pioneer Kit-related files. Choose the directory and click **Next**.

Figure 2-1. Kit Installer Screen



CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit - InstallShield Wizard

Welcome to the InstallShield Wizard for CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

The InstallShield Wizard will install CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit on your computer. To continue, click Next.

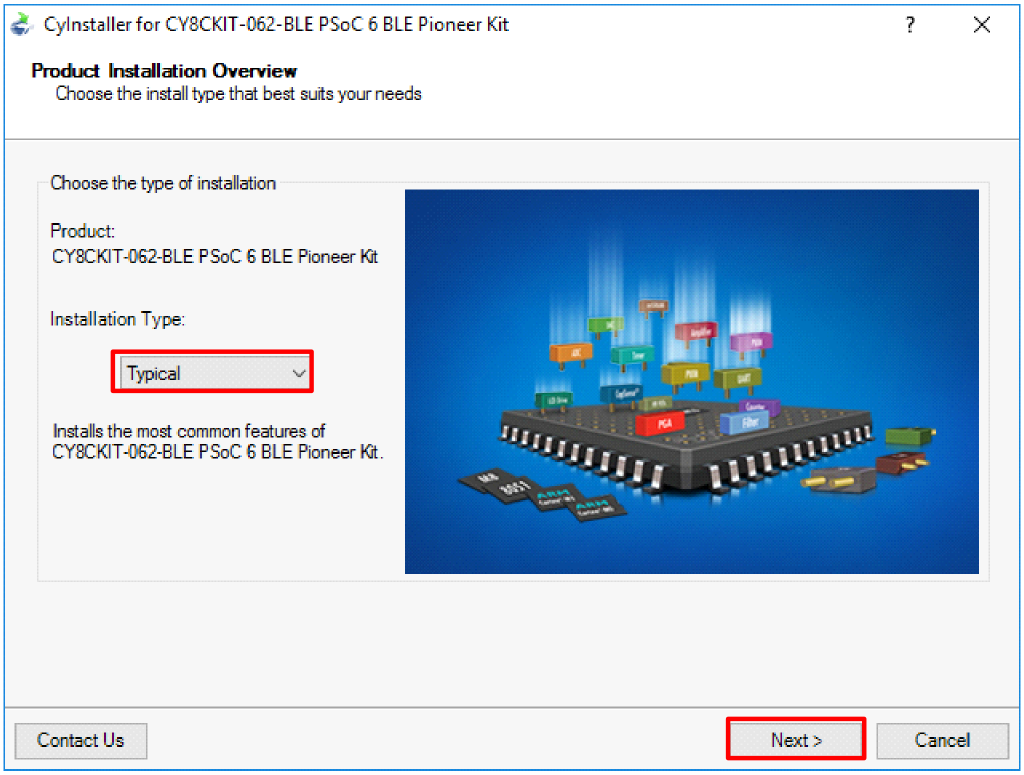
Select folder where setup will install files.

Install CY8CKIT-062-BLE PSoC 6
C:\...\Cypress

3. When you click **Next**, the installer automatically installs the required software, if it is not present on your computer. Following are the required software:
   a. PSoC Creator 4.2: This software is available for download separately at www.cypress.com/psoccreator. PSoC Creator 4.2 installer automatically installs the following additional software:
      PSoC Programmer 3.27.0
      Peripheral Driver Library 2.1.0
      Peripheral Driver Library 3.0.1
   b. CySmart 1.2 or later, available for download separately at www.cypress.com/cysmart.

4. 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 selecting the installation type.

Figure 2-2. Product Installation Overview


CyInstaller for CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

Product Installation Overview

Choose the install type that best suits your needs

Choose the type of installation

Product:
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

Installation Type: Typical

Installs the most common features of CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit.

5. Read the License agreement and select **I accept the terms in the license agreement** to continue with installation. Click **Next**.
6. 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.
7. Enter your contact information or select the check box **Continue Without Contact Information**. Click **Finish** to complete the CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit software installation.
8. After the installation is complete, the kit contents are available at the following location:
   `<Install_Directory>\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit`

   Default location:

   Windows 7 (64-bit): `C:\Program Files (x86)\Cypress\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit`

   Windows 7 (32-bit): `C:\Program Files\Cypress\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit`

Note: For Windows 7/8/8.1/10 users, the installed files and the folder are read-only. To use the installed code examples, follow the steps outlined in Code Examples chapter on page 38. These steps will create an editable copy of the example in a path that you choose, so the original installed example is not modified.

3. Kit Operation

This chapter introduces you to various features of the PSoC 6 BLE Pioneer Kit, including the theory of operation and the onboard programming and debugging functionality, KitProg2 USB-UART, USB-I2C, USB-SPI bridges, and USB Type-C power delivery.

3.1 Theory of Operation

The PSoC 6 BLE Pioneer Kit is built around PSoC 6 MCU. Figure 3-1 shows the block diagram of the PSoC 6 MCU device used in the PSoC 6 BLE Pioneer Kit. For details of device features, see the device datasheet.

Figure 3-1. PSoC 6 MCU Block Diagram

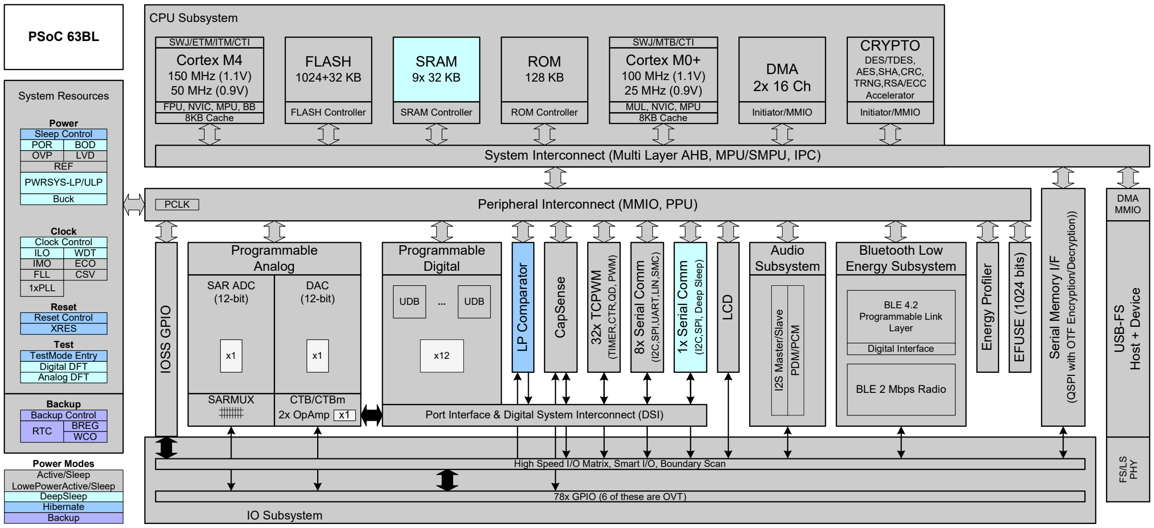
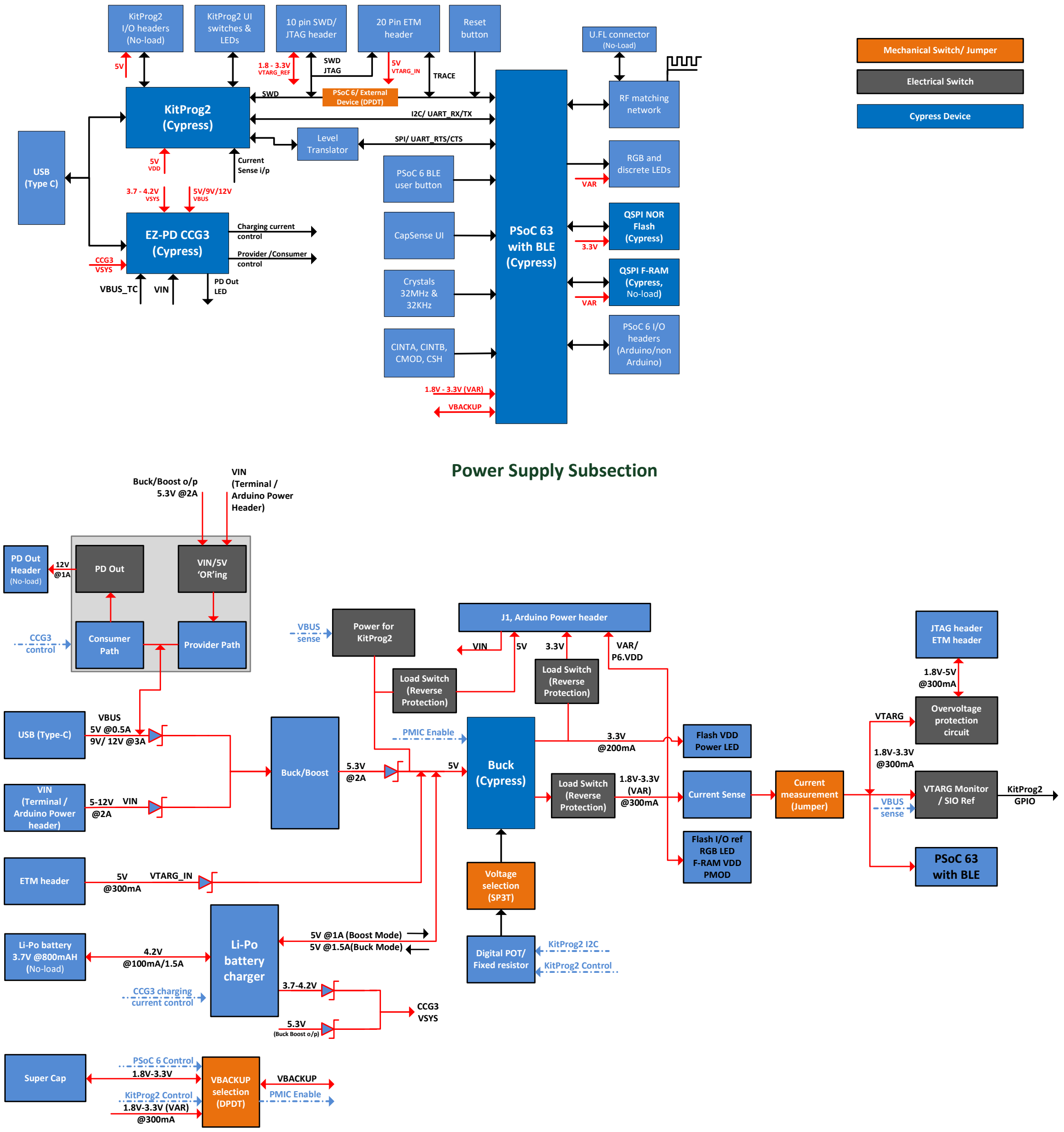


Figure 3-2 shows the block diagram for the Pioneer Board.

Figure 3-2. Block Diagram of Pioneer Board



The CY8CKIT-062-BLE Pioneer Kit comes with the PSoC 6 BLE Pioneer Board with the CY8CKIT-028-EPD E-INK display shield connected, as Figure 3-3 shows.

Figure 3-3. PSoC 6 BLE Pioneer board and E-INK display shield

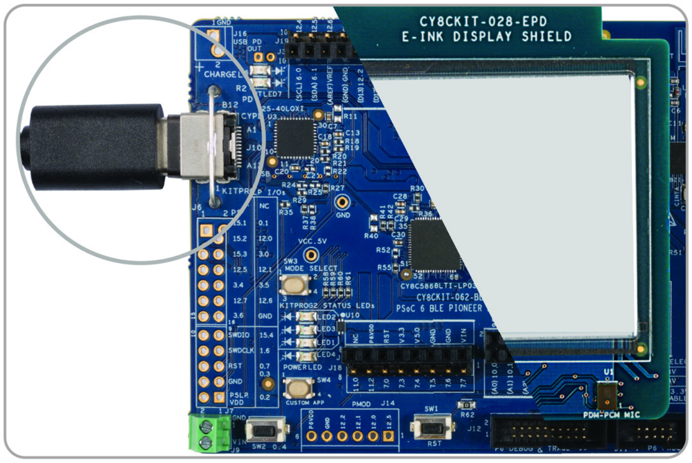
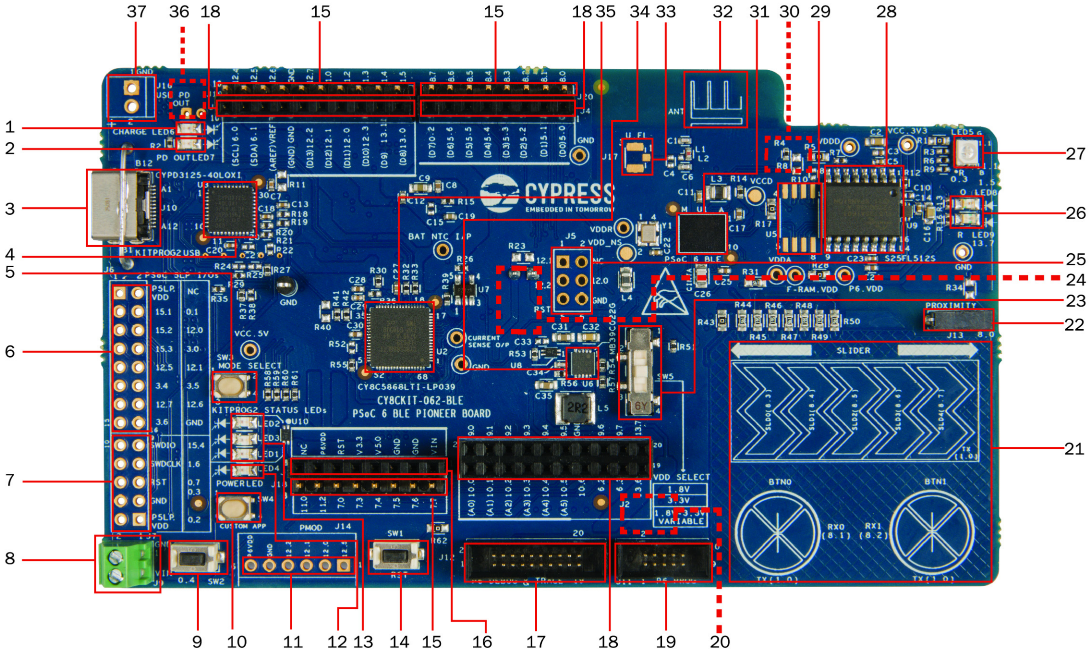


Figure 3-4 shows the markup of the Pioneer board.

Figure 3-4. PSoC 6 BLE Pioneer board - Top View



The PSoC 6 BLE Pioneer board has the following peripherals:

1. **Battery charging indicator (LED7):** This LED turns ON when the onboard battery charger is charging a lithium-iron polymer battery connected to **J15**. Note that the battery connector and battery are not included in the Kit and should be purchased separately if you have to test the battery charging functionality.
2. **USB PD out indicator (LED6):** This LED turns ON when the USB Type-C power delivery output is available for use.
3. **KitProg2 USB connector (J10):** The USB cable provided along with the PSoC 6 BLE Pioneer Kit connects between this USB connector and the PC to use the KitProg2 onboard programmer and debugger and to provide power to the Pioneer board. **J10** is also used for the USB Type-C power delivery system. See EZPD CCG3 Type-C Power Delivery on page 36 for more details.
4. **Cypress EZ-PD™ CCG3 Type-C Port Controller with PD (CYPD3125-40LQXIT, U3):** The Pioneer Board includes a Cypress EZ-PD™ CCG3 USB Type-C Port controller with Power Delivery system. This EZ-PD™ CCG3 device is pre-programmed and can deliver power from a Type-C port to onboard header **J16**, while simultaneously charging a lithium-ion polymer battery connected to **J15**. In addition, the power delivery system can deliver power to a Type-C power sink or consumer such as a cell phone with the power derived from the VIN supply. See EZPD CCG3 Type-C Power Delivery on page 36 for more details.
5. **KitProg2 programming mode selection button (SW3):** This button can be used to switch between various modes of operation of KitProg2 (Proprietary SWD Programming or Mass Storage programming/CMSIS-DAP mode). This button can also be used to provide input to PSoC 5LP in custom application mode. For more details, see the KitProg2 User Guide.
6. **KitProg2 I/O header (J6):** This header brings out several GPIOs of the onboard KitProg2 PSoC 5LP device. This includes the USB-I2C, USB-UART, and USB-SPI bridge lines. The additional PSoC 5LP pins are direct connections to the internal programmable analog logic of the PSoC 5LP. You can also use these pins for custom applications. For more details on the KitProg2, see the KitProg2 User Guide.
7. **KitProg2 programming/custom application header (J7):** This header brings out more GPIOs of the PSoC 5LP, which can be used for custom applications. It also contains a 5-pin SWD programming header for the PSoC 5LP.
8. **External Power Supply VIN connector (J9):** This connector connects an external DC power supply input to the onboard regulators and the USB Type-C power delivery system. The voltage input from the external supply should be between 5 V and 12 V. Moreover, when used as an input to the USB Type-C power delivery system, the external power supply should have enough current capacity to support the load connected via the Type-C port. See EZPD CCG3 Type-C Power Delivery on page 36 for more details.
9. **PSoC 6 MCU user button (SW2):** This button can be used to provide an input to PSoC 6 MCU. Note that by default the button connects the PSoC 6 MCU pin to ground when pressed, so you need to configure the PSoC 6 MCU pin as a digital input with resistive pull-up for detecting the button press. This button also provides a wake-up source from low-power modes of the device. In addition, this button can be used to activate the regulator control output from PSoC 6 MCU.
10. **KitProg2 application selection button (SW4):** This button can be used to switch between KitProg2 programming mode and custom application mode. For more details, see the KitProg2 User Guide.
11. **Digilent® Pmod™ compatible I/O header (J14):** This header can be used to connect Digilent® Pmod™ 1 x 6 pin modules.
12. **Power LED (LED4):** This is the amber LED that indicates the status of power supplied to PSoC 6 MCU.

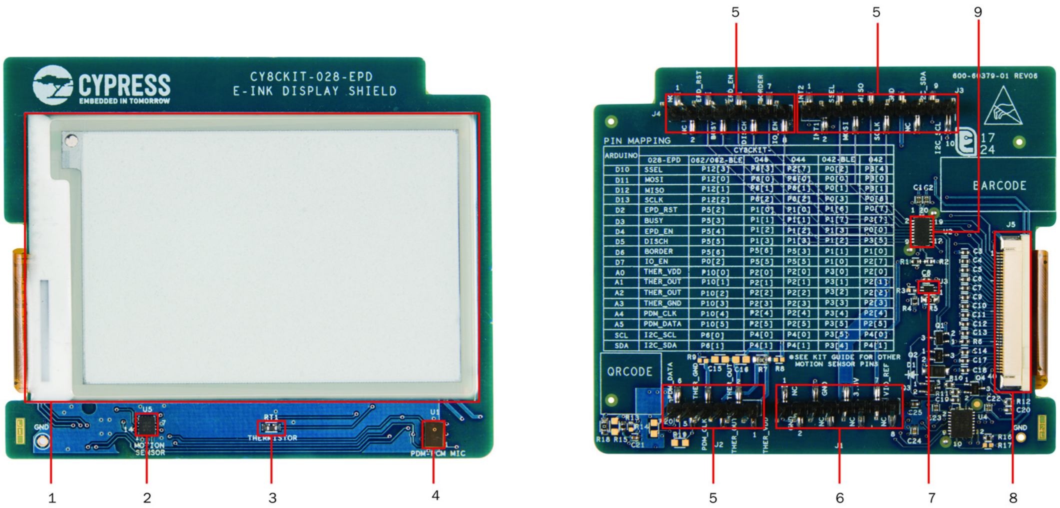
13. **Kitprog2 status LEDs (LED1, LED2, and LED3):** Amber, Red, and Green LEDs (LED1, LED2, and LED3 respectively) indicate the status of KitProg2. For details on the KitProg2 status, see the KitProg2 User Guide.
14. **PSoC 6 MCU reset button (SW1):** This button is used to reset PSoC 6 MCU. This button connects the PSoC 6 MCU reset (XRES) pin to ground.
15. **PSoC 6 MCU I/O headers (J18, J19, and J20):** These headers provide connectivity to PSoC 6 MCU GPIOs that are not connected to the Arduino compatible headers. Majority of these pins are multiplexed with onbroad peripherals and are not connected to PSoC 6 MCU by default. For the detailed information on how to rework the kit to access these pins, see Table 1-2 on page 10.
16. **Arduino compatible power header (J1):** The Arduino-compatible power header powers Arduino shields. This header also has a provision to power the kit though the VIN input.
17. **PSoC 6 MCU debug and trace header (J12):** This header can be connected to an Embedded Trace Macrocell (ETM) compatible programmer/debugger
18. **Arduino Uno R3 compatible I/O headers (J2, J3, and J4):** The Arduino-compatible I/O headers bring out pins from PSoC 6 MCU to interface with the Arduino shields. Few of these pins are multiplexed with onboard peripherals and are not connected to PSoC 6 MCU by default. For a detailed information on how to rework the kit to access these pins, see Table 1-2 on page 10
19. **PSoC 6 MCU program and debug header (J11):** This 10-pin header allows you to program and debug the PSoC 6 MCU using an external programmer such as MiniProg3. In addition, an external PSoC 4/5/6 device can be connected to this header and programmed using KitProg2. To program the external device, **SW6** should be used to select the “External” option.
20. **Kitprog2 programming target selection switch (SW6, on the bottom side of the Board):** This switch selects the programming target of the onboard KitProg2 between the onboard PSoC 6 MCU and an external PSoC 4/5/6 device connected to **J11**.
21. **CapSense slider (SLIDER) and buttons (BTN0 and BTN1):** CapSense touch-sensing slider and two buttons, all of which are capable of both self-capacitance (CSD) and mutual-capacitance (CSX) operation, let you evaluate Cypress’ fourth-generation CapSense technology. The slider and the buttons have a 1-mm acrylic overlay for smooth touch sensing.
22. **CapSense proximity header (J13):** A wire can be connected to this header to evaluate the proximity sensing feature of CapSense.
23. **System Power V_{DD} selection switch (SW5):** This switch is used to select the PSoC 6 MCU’s V_{DD} supply voltage between constant 1.8 V, constant 3.3 V, and variable 1.8 to 3.3 V. In the variable 1.8 to 3.3 V mode, the PSoC programmer software can control the voltage via the KitProg2.
24. **PSoC 6 MCU current measurement jumper (J8, on the bottom side of the Board):** An ammeter can be connected to this jumper to measure the current consumed by the PSoC 6 MCU.
25. **Arduino compatible ICSP header (J5):** This header provides an SPI interface for Arduino ICSP compatible shields.
26. **PSoC 6 MCU user LEDs (LED8 and LED9):** These two user LEDs can operate at the entire operating voltage range of PSoC 6 MCU. The LEDs are active LOW, so the pins must be driven to ground to turn ON the LEDs.
27. **RGB LED (LED5):** This onboard RGB LED can be controlled by the PSoC 6 MCU. The LEDs are active LOW, so the pins must be driven to ground to turn ON the LEDs.
28. **Cypress 512-Mbit serial NOR flash memory (S25FL512SAGMFI011, U4):** This kit features a Cypress NOR flash (S25FL512SAGMFI011) of 512 Mb capacity. The NOR Flash is connected to the serial memory interface (SMIF) of the PSoC 6 MCU. The NOR device can be used for both data and code memory with execute-in-place (XIP) support and encryption.
29. **Cypress 4-Mbit serial Ferroelectric RAM (FM25V10, U5):** Footprint to connect a FM25V10 or any other pin compatible FRAM.

30. **Vbackup and PMIC control selection switch (SW7, on the bottom side of the Board):** This switches the Vbackup supply connection to PSoC 6 MCU between V_{DDD} and the super-capacitor. When V_{DDD} is selected, the regulator ON/OFF is controlled by the KitProg2. When super-capacitor is selected, the regulator ON/OFF is controlled by PSoC 6 MCU.
31. **Cypress PSoC 6 MCU (CY8C637BZI-BLD74, U1):** This kit is designed to highlight the features of the PSoC 6 MCU. For details on PSoC 6 MCU pin mapping, see Table 1-2 on page 10.
32. **BLE antenna:** This is the onboard wiggle antenna for BLE.
33. **U.FL connector (J17):** This connector can be used for conductive measurements and also to connect external antenna.
34. **Cypress main voltage regulator (**MB39C022GPN-G-ERE1**, U6):** This is the main regulator that powers PSoC 6 MCU. This regulator has two output channels. One channel provides fixed LDO-based 3.3 V output from 5 V input and the other channel is a buck DC to DC converter that is configured to provide variable voltage from 1.8 V to 3.3 V.
35. **KitProg2 (PSoC 5LP) programmer and debugger (CY8C5868LTI-LP039, U2):** The PSoC 5LP device (CY8C5868LTI-LP039) serving as KitProg2, is a multi-functional system, which includes a programmer, debugger, USB-I2C bridge, USB-UART bridge, and a USB-SPI bridge. KitProg2 also supports custom applications. For more details, see the KitProg2 User Guide.
36. **Battery connector (J15, on the bottom side of the Board):** This connector can be used to connect a lithium-ion polymer battery. Note that a battery is not included in the kit package and should be purchased separately if you want to demonstrate battery charging.
37. **USB PD output (J16):** This header provides a voltage output when the USB Type-C power delivery system receives power from an external host connected to **J10**. See EZPD CCG3 Type-C Power Delivery on page 36 for more details.

See Hardware Functional Description chapter on page 41 for details on various hardware blocks.

3.1.1 CY8CKIT-028-EPD E-INK Display Shield

Figure 3-5. E-INK Display Shield



The E-INK display shield has the following peripherals:

1. **2.7 inch E-INK display:** This is a monochrome E-INK display with a resolution of 264x176. The E-INK display can retain its contents even in the absence of power, which provides a low-power, “always-on” display functionality.
2. **Motion sensor (U5):** This is a 3-axis acceleration and 3-axis gyroscopic motion sensor that can be used to count steps to emulate a pedometer or similar application.
3. **Thermistor(RT1):** This thermistor can be used for temperature compensation of the display or as a general purpose ambient temperature sensor.
4. **PDM microphone (U1):** This microphone converts voice inputs to Pulse-Density Modulated (PDM) digital signals.
5. **Arduino compatible I/O header (J2, J3 and J4):** This header interfaces with the PSoC 6 MCU GPIOs through header **J2** on the board.
6. **Arduino compatible power & I/O header (J1):** This header receives power from header J1 on the board.
7. **E-INK display power control load switch (U3):** This load switch can be controlled by the board to toggle the E-INK display’s power.
8. **E-INK display connector (J5):** This connector is used to connect the E-INK display to the circuits on the E-INK display shield.
9. **E-INK display I/O voltage translator (U2):** This I/O level translator allows the board to operate at any voltage between 1.8 and 3.3 V by providing a constant 3.3 V interface to the E-INK display.

3.2 KitProg2

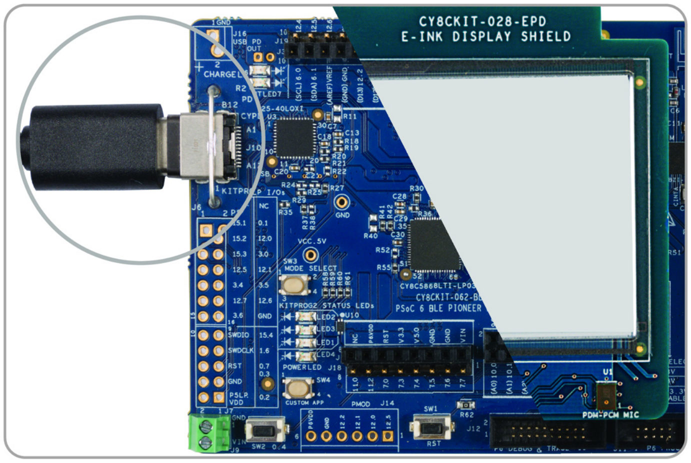
The PSoC 6 BLE Pioneer Kit can be programmed and debugged using the onboard KitProg2. The KitProg2 is a multi-functional system, which includes a programmer, debugger, USB-I2C bridge, USB-UART bridge, and a USB-SPI bridge. KitProg2 also supports mass storage programming and CMSIS-DAP, and custom applications. A Cypress PSoC 5LP device is used to implement KitProg2 functionality. The KitProg2 is integrated in most PSoC development kits. For more details on the KitProg2 functionality, see the KitProg2 User Guide.

Before programming the device, ensure that PSoC Creator and PSoC Programmer software are installed on the computer. See the Install Software chapter on page 23 for more information.

3.2.1 Programming and Debugging using PSoC Creator

1. Connect the PSoC 6 BLE Pioneer Kit to the PC using the USB cable, as shown in Figure 3-6. The kit enumerates as a composite device if you are connecting it to your PC for the first time. The successful enumeration is indicated by the following status LEDs: Amber LED ON, Green LED OFF, and Red LED OFF. If you do not see the desired LED status, see the KitProg2 User Guide for details on the KitProg2 status and troubleshooting instructions. For example, if the Amber LED is showing a breathing effect, press the mode button to switch from mass storage programming mode to SWD programming mode.

Figure 3-6. Connect USB Cable to USB connector on the kit



2. Open the desired project in PSoC Creator. For this, go to **File** > **Open** > **Project/Workspace**. This provides the option to browse and open your saved project.
3. Select the option **Build** > **Build Project** or pressing **[Shift] [F6]** to build the project.
4. If there are no errors during build, select **Debug** > **Program** or press **[Ctrl] [F5]**. This programs the device on the PSoC 6 BLE Pioneer Kit.

PSoC Creator has an integrated debugger. You can start the debugger by selecting **Debug** > **Debug** or by pressing **[F5]**. For a detailed explanation on how to debug using PSoC Creator, see the **Debugging Using PSoC Creator** section in the KitProg2 User Guide

3.2.2 Programming using PSoC Programmer

PSoC Programmer can be used to program existing *.hex* files into the PSoC 6 BLE Pioneer Kit. For a detailed explanation on how to program using PSoC Programmer, see the **Programming Using PSoC Programmer** section in the KitProg2 User Guide.

The KitProg2 firmware normally does not require any update. If necessary you can use the PSoC Programmer software to update the KitProg2 firmware. For a detailed explanation on how to update the KitProg2 firmware, see the **Updating the KitProg2 Firmware** section in the KitProg2 User Guide.

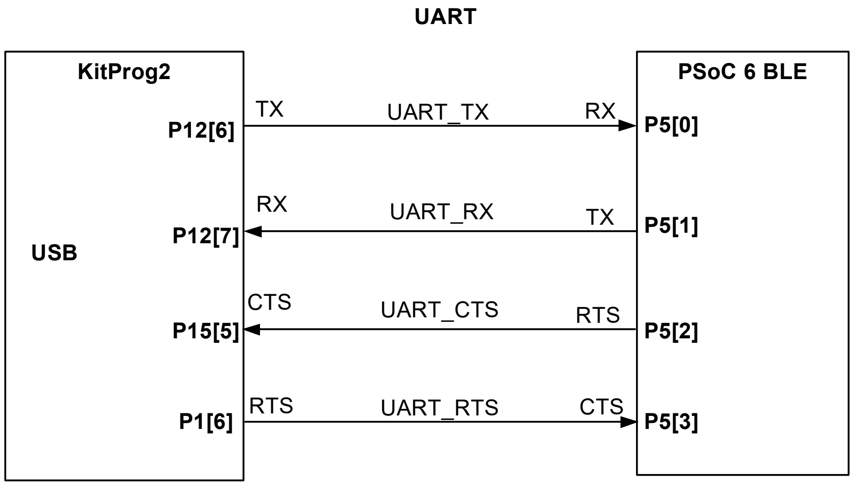
3.2.3 Mass Storage Programmer

The KitProg2 in the PSoC 6 BLE Pioneer Kit supports programming through a USB Mass Storage interface. This interface allows you to program PSoC 6 MCU by copying *.hex* files into an emulated USB Mass Storage device. The user can press the mode button to switch to mass storage mode. At that time, the Amber LED will show a breathing effect. Press the mode button again to switch to the normal programming mode. For more details on KitProg2 Mass Storage Programmer, see the KitProg2 User Guide.

3.2.4 USB-UART Bridge

The KitProg2 on the PSoC 6 BLE Pioneer Kit can act as a USB-UART bridge. The UART and flow-control lines between the PSoC 6 MCU and the KitProg2 are hard-wired on the board, as Figure 3-7 shows. For more details on the KitProg2 USB-UART functionality, see the KitProg2 User Guide.

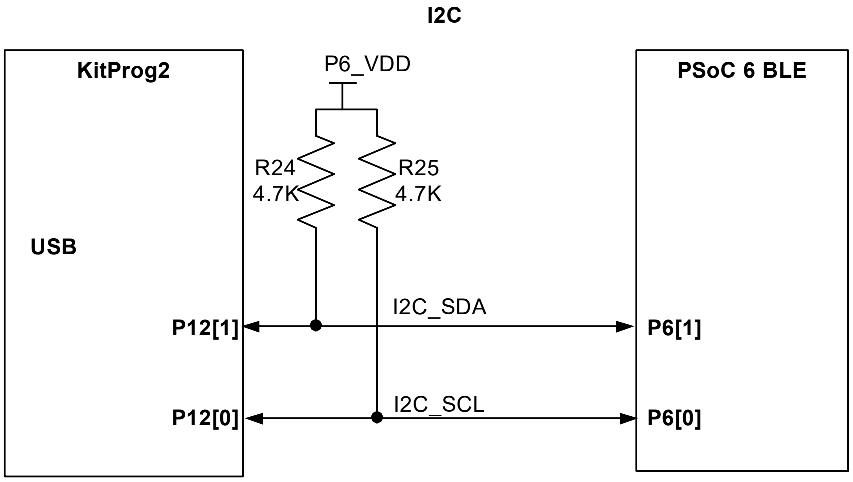
Figure 3-7. UART connection between KitProg2 and PSoC 6



3.2.5 USB-I2C Bridge

The KitProg2 can function as a USB-I2C bridge and communicate with the Bridge Control Panel (BCP) software. The I2C lines on the PSoC 6 MCU are hard-wired on the board to the I2C lines of the KitProg2, with onboard pull-up resistors as Figure 3-8 shows. The USB-I2C supports I2C speeds of 50 kHz, 100 kHz, 400 kHz, and 1 MHz. For more details on the KitProg2 USB-I2C functionality, see the KitProg2 User Guide.

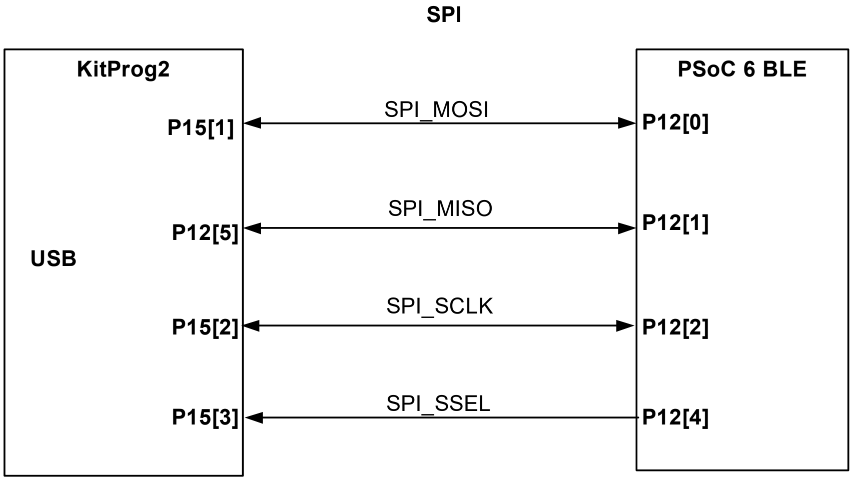
Figure 3-8. I2C connection between KitProg2 and PSoC 6



3.2.6 USB-SPI Bridge

The KitProg2 can function as a USB-SPI bridge. The SPI lines between the PSoC 6 MCU and the KitProg2 are hard-wired on the board, as Figure 3-9 shows. For more details on the KitProg2 USB-SPI functionality, see the KitProg2 User Guide.

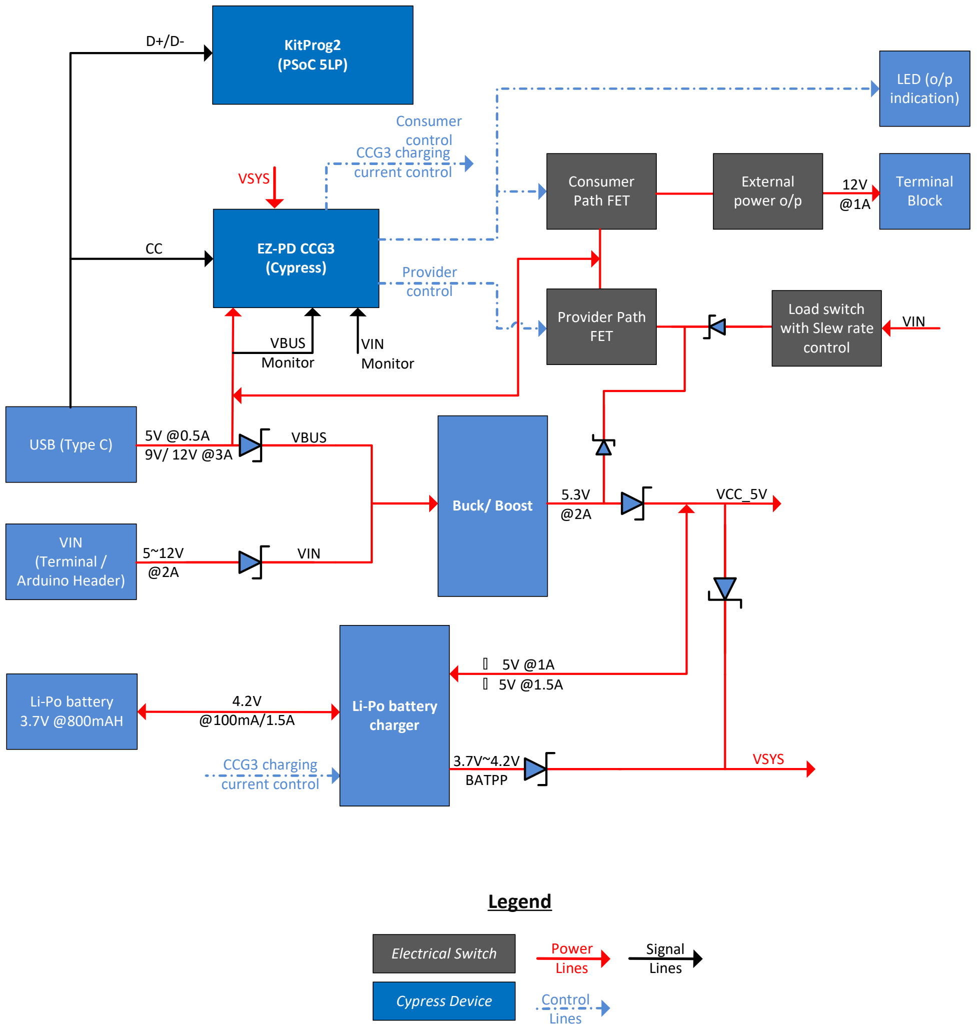
Figure 3-9. SPI connection between KitProg2 and PSoC 6



3.3 EZPD CCG3 Type-C Power Delivery

The Pioneer board includes a Cypress EZ-PD CCG3 power delivery system. This EZ-PD™ CCG3 is pre-programmed and can deliver power from a Type-C port to onboard header **J16** (known as the consumer path), while simultaneously charging a 3.7 V, lithium-ion polymer battery connected to **J15**. In addition, the power delivery system can deliver power to a Type-C peripheral such as a cell phone with the power derived from the VIN **(J9)** supply (known as the provider path). Note that to use the EZ-PD™ CCG3 Type-C power delivery system, a power delivery capable USB Type-C to Type-C cable should be connected to **J10**. This cable is not included in the kit, and should be purchased separately.

Figure 3-10. Type-C Block Diagram



The power delivery system works as follows:

1. If the power delivery system detects a non Type-C power adapter (Legacy USB), CCG3 will charge the battery at 100 mA. CCG3 will also disable the consumer and provider paths.
2. On detection of a Type-C power adapter, CCG3 will request 5 V at 3 A, 9 V at 3 A, or 12 V at 3 A depending on the host capability. Once the power level is successfully negotiated, the Consumer path is enabled by turning on load switch **U12**. This load switch is hardware limited to supply up to 1 A through header **J16** to an external device. CCG3 will use the remaining current to charge the battery connected to **J15** at a higher charging rate up to 1.5 A and PD output voltage availability indicator (**LED6**) will be turned ON.
3. CCG3 will also advertise that it can provide 5 V, 9 V, or 12 V if a DC power supply capable of providing either of these voltages is connected at VIN **(J9)**. The current is limited in this case to 1 A. Note that the external supply must be capable of providing this current. If a connected, Type-C device requests power, the provider path is enabled by turning on load switch **U22**. Table below details the Power delivery scenarios for onboard CCG3.

Table 3-1. Type-C table Power delivery Scenarios

USB Host / consumer capability	VIN	Consumer capability	Provider capability	External USB PD out (J16 header)	Battery Charging current
Non Type-C Power adapter (Legacy USB)	<5V	N/A	0	0	100mA
	>5V	N/A	0	0	0
Type-C, PD power adapter (12V capable)	<12V	12V@3A	0	12V@1A *	1.5A max
	>12V	N/A	0	0	0
Type-C, capable of providing max 9V	<9V	9V@3A	0	9V@1A	1.5A max
	>9V	N/A	0	0	0
Type-C only, capable of providing max 5V	<5	5V@3A	0	5V@1A	1.5A max
	>5	N/A	0	0	0
Type-C, requesting 12V	≠12V	0	5V@1A	0	0
	12V	0	12V@1A	0	0
Type-C, requesting 9V	≠9V	0	5V@1A	0	0
	9V	0	9V@1A	0	0
Type-C, requesting 5V	≠5V	0	5V@1A	0	0
	5V	0	5V@1A	0	0
Type-C, requesting another voltage	5V < VIN <12V	0	5V@1A	0	0

* Due to the voltage drop in series components, the voltage at J16 is ~9 V when 12 V PD power adapter is used. Populate R79 resistor to bypass this drop.

For more information on USB Type-C power delivery with CCG3 device, see the EZ-PD CCG3 web page.

4. Code Examples

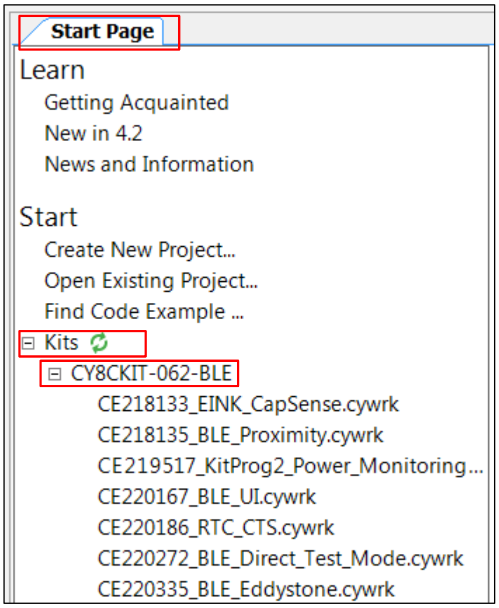
The PSoC 6 BLE Pioneer Kit includes three code examples. To access the code examples, download and install the PSoC 6 BLE Pioneer Kit setup file from www.cypress.com/CY8CKIT-062-BLE. After installation, the code examples will be available from **Start** > **Kits** on the PSoC Creator Start Page.

4.1 Using the Kit Code Examples

Follow these steps to open and use the code examples.

1. Launch PSoC Creator from **Start** > **All Programs** > **Cypress** > **PSoC Creator <version>** > **PSoC Creator <version>**.
2. On the Start Page, click **CY8CKIT-062-BLE** under **Start** > **Kits**. A list of code examples appears, as shown in Figure 4-1.
3. Click the desired code example, select a location to save the project, and click **OK**.

Figure 4-1. Open Code Example from PSoC Creator



Start Page

Learn

Getting Acquainted

New in 4.2

News and Information

Start

Create New Project...

Open Existing Project...

Find Code Example ...

⊟ Kits

⊟ CY8CKIT-062-BLE

CE218133_EINK_CapSense.cywrk

CE218135_BLE_Proximity.cywrk

CE219517_KitProg2_Power_Monitoring...

CE220167_BLE_UI.cywrk

CE220186_RTC_CTS.cywrk

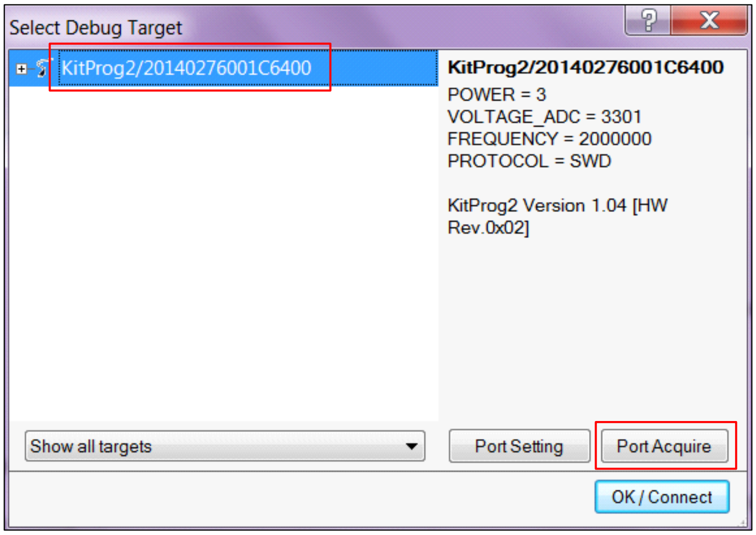
CE220272_BLE_Direct_Test_Mode.cywrk

CE220335_BLE_Eddystone.cywrk

4. Build the code example by choosing **Build** > **Build <Project Name>**. After the build process is successful, a *.hex* file is generated.
5. Connect PSoC 6 BLE Pioneer Kit to the PC using the USB cable, as shown in Figure 3-6, to program the kit with the code example.
6. Choose **Debug** > **Program** in PSoC Creator.

7. If the device is already acquired, programming will complete automatically – the result will appear in the PSoC Creator status bar at the bottom left side of the screen. If the device is yet to be acquired, the Select Debug Target window will appear. Select **KitProg2/<serial_number>** and click **Port Acquire**, as shown in Figure 4-2.

Figure 4-2. Port Acquire



Select Debug Target

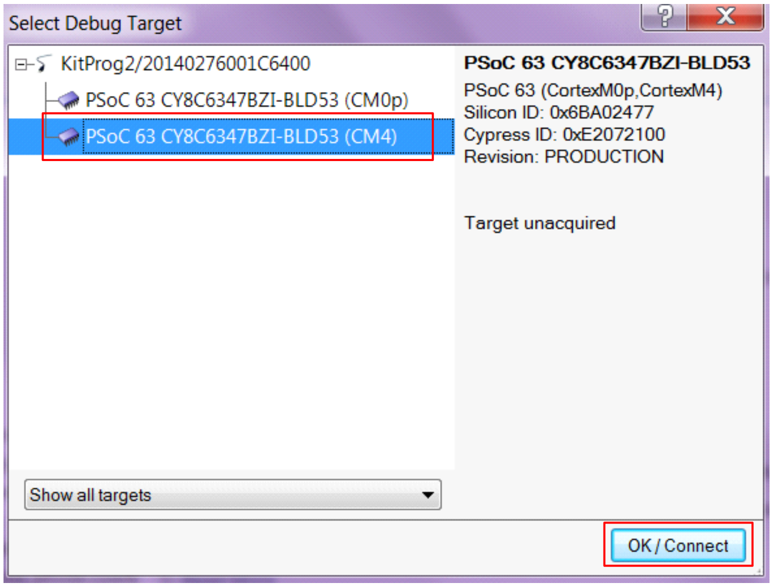
KitProg2/20140276001C6400

KitProg2/20140276001C6400
POWER = 3
VOLTAGE_ADC = 3301
FREQUENCY = 2000000
PROTOCOL = SWD

KitProg2 Version 1.04 [HW Rev.0x02]

8. After the device is acquired, it is shown in a tree structure below the **KitProg2/<serial_number>**. Click **Connect** and then **OK** to exit the window and start programming, as shown in Figure 4-3.
Note: PSoC 6 MCUs have both an ARM Cortex M0+ and ARM Cortex M4 CPUs. To program, you can select any one of them and click **Connect**. To debug, you have to select the CPU which you want to be debugged.

Figure 4-3. Connect Device from PSoC Creator and Program



Select Debug Target

KitProg2/20140276001C6400
PSoC 63 CY8C6347BZI-BLD53 (CM0p)
PSoC 63 CY8C6347BZI-BLD53 (CM4)

PSoC 63 CY8C6347BZI-BLD53
PSoC 63 (CortexM0p,CortexM4)
Silicon ID: 0x6BA02477
Cypress ID: 0xE2072100
Revision: PRODUCTION

Target unacquired

9. After programming is successful, the code example is ready to use.

4.2 Code Examples

Table 4-1 shows a list of code examples that can be used with this kit. See the individual code example document for additional details on a particular example.

Table 4-1. Code Examples in PSoC Creator

#	Project	Description
1	CE218133_EINK_CapSense	This code example shows how to create a user interface solution using an E-INK display and CapSense.
2	CE218135_BLE_Proximity	This code example demonstrates connectivity between PSoC 6 MCU with BLE and CySmart BLE host emulation tool or mobile device running the CySmart mobile application, to transfer CapSense proximity sensing information.
3	CE219517_KitProg2_Power_Monitoring	This code example demonstrates how to create a bootloadable PSoC 5LP (KitProg2) project to monitor the power consumed by the PSoC 6 MCU device on CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit.
4	CE220167_BLE_UI	This code example demonstrates interfacing PSoC 6 MCU with user interface functions such as an E-INK display, RGB LED, and touch sensors based on self and mutual capacitance (CapSense CSD and CSX) with bi-directional BLE connectivity between the PSoC 6 MCU device and a PC running the CySmart BLE Host Emulation tool or a mobile device running the CySmart mobile application.
5	CE220186_RTC_CTS	This code example demonstrates accurate time keeping with PSoC 6 MCU's real time clock (RTC), which is synchronized with a current time server such as an iPhone using the BLE current time service (CTS).
6	CE220272_BLE_Direct_Test_Mode	This code example demonstrates Direct Test Mode (DTM) over the Host Controller Interface (HCI) using PSoC 6 MCU with BLE Connectivity.
7	CE220335_BLE_Eddystone	This code example demonstrates a BLE beacon that broadcasts the core frame types (UID, URL, and TLM) of Google's Eddystone beacon profile.

A. Appendix

A.1 Schematics

Refer to the schematics files available in the kit installation directory under following paths:

1. `<Install_Directory>\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit\1.0\Hardware\CY8CKIT-028-EPD\CY8CKIT-028-EPD Schematic.pdf`
2. `<Install_Directory>\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit\1.0\Hardware\CY8CKIT-062-BLE\CY8CKIT-062-BLE Schematic.pdf`

A.2 Hardware Functional Description

This section explains in detail the individual hardware blocks of the PSoC 6 BLE Pioneer board.

A.2.1 PSoC 6 MCU (U1)

PSoC 6 MCU is Cypress' latest, ultra-low-power PSoC specifically designed for wearables and IoT products. PSoC 6 MCU is a true programmable embedded system-on-chip, integrating a 150-MHz ARM® Cortex®-M4 as the primary application processor, a 100-MHz ARM Cortex®-M0+ that supports low-power operations, up to 1 MB Flash and 288 KB SRAM, an integrated BLE 4.2 radio, CapSense touch-sensing, and custom analog and digital peripheral functions. The programmable analog and digital peripheral functions allow higher flexibility, in-field tuning of the design, and faster time-to-market.

For more information, see the PSoC 6 MCU web page and the datasheet.

A.2.2 PSoC 5LP (U2)

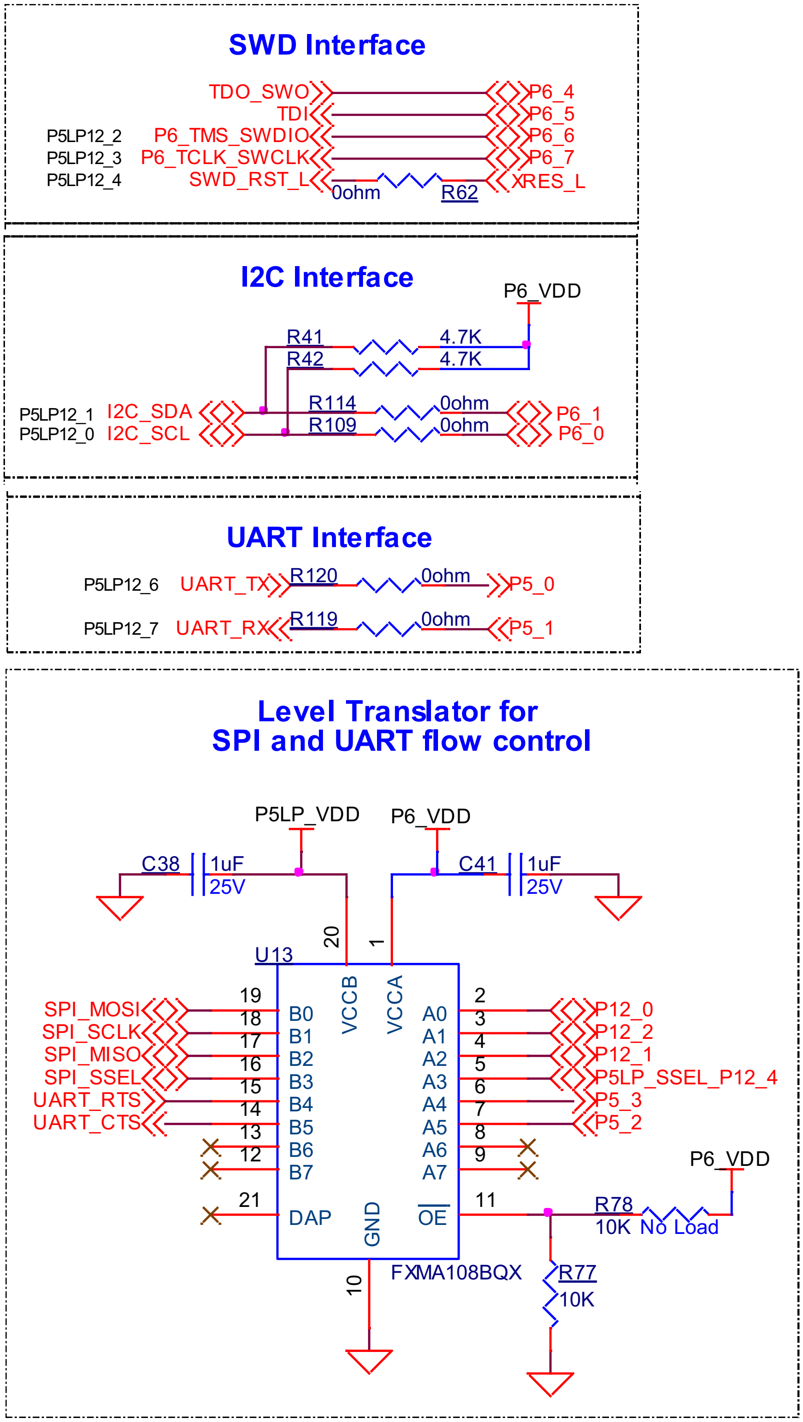
An onboard PSoC 5LP (CY8C5868LTI-LP039) is used as KitProg2 to program and debug PSoC 6 MCU. The PSoC 5LP connects to the USB port of a PC through a USB connector and to the SWD and other communication interfaces of PSoC 6 MCU. The PSoC 5LP is a true system-level solution providing MCU, memory, analog, 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 information, visit the PSoC 5LP web page. Also, see the CY8C58LPxx Family Datasheet.

A.2.3 Serial Interconnection between PSoC 5LP and PSoC 6 MCU

In addition of use as an onboard programmer, the PSoC 5LP functions as an interface for the USB-UART, USB-I2C, and USB-SPI bridges, as shown in Figure A-1. The USB-Serial pins of the PSoC 5LP are hard-wired to the I2C/UART/SPI pins of the PSoC 6 MCU. These pins are also available on the Arduino-compatible I/O headers; therefore, the PSoC 5LP can be used to control Arduino shields with an I2C/UART/SPI interface.

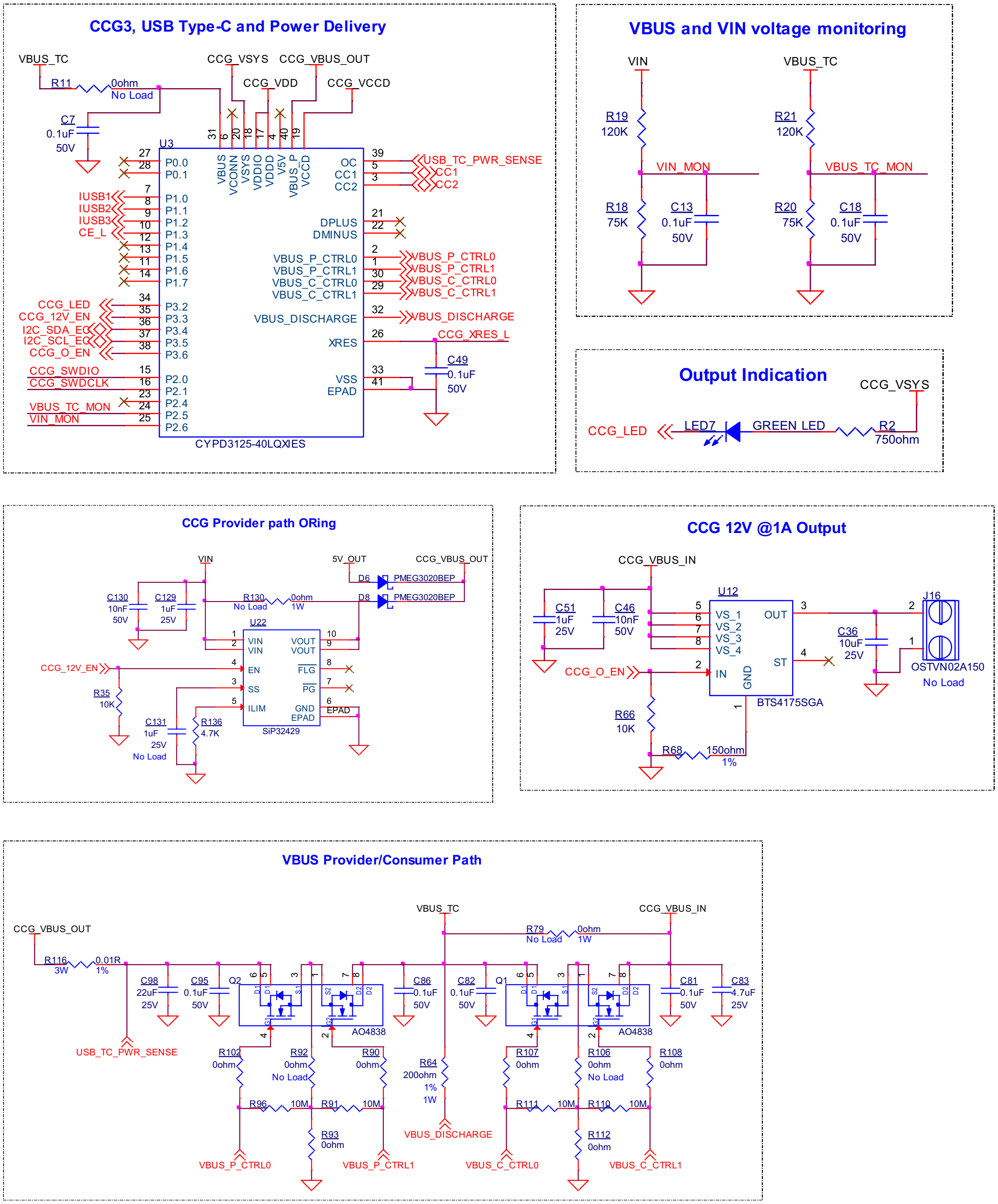
Figure A-1. Schematics of Programming and Serial Interface Connections



A.2.4 EZ-PD CCG3 Power Delivery System

Cypress EZ-PD CCG3 provides a complete solution ideal for power adapters, power banks, Type-C dongles, monitors, docks and notebooks. See “EZPD CCG3 Type-C Power Delivery” on page 36 for more details of the power delivery system implementation in the Pioneer board.

Figure A-2. Schematics of EZ-PD CCG3 Power Delivery System



A.2.5 Power Supply System

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

- 5 V, 9 V, or 12 V from the onboard USB Type-C connector
- 5 V to 12 V power from an Arduino shield or from external power supply through VIN header **J9** or **J1**
- 3.7 V from a rechargeable Li-Po battery connected to **J15**
- 5 V from an external programmer/debugger connected to **J11** and **J12**

The power supply system is designed to support 1.8 V to 3.3 V operation of the PSoC 6 MCU. In addition, an intermediate voltage of 5 V is required for the operation of the power delivery circuitry and KitProg2. Therefore, three regulators are used to achieve 1.8 to 3.3 V and 5 V outputs - a buck boost regulator **(U21)** that generates a fixed 5 V from an input of 5 to 12 V, and a main regulator **(U6)** that generates a variable 1.8 V to 3.3 V and a fixed 3.3 V from the output of **U21**. Figure A-3 shows the schematics of the voltage regulator and power selection circuits. In addition to this, the battery charger **U14** also functions as a boost regulator. **U14** boosts the battery voltage to provide a 5 V to the main regulator **U6**. This feature is enabled only when the VIN and the USB supply are unavailable.

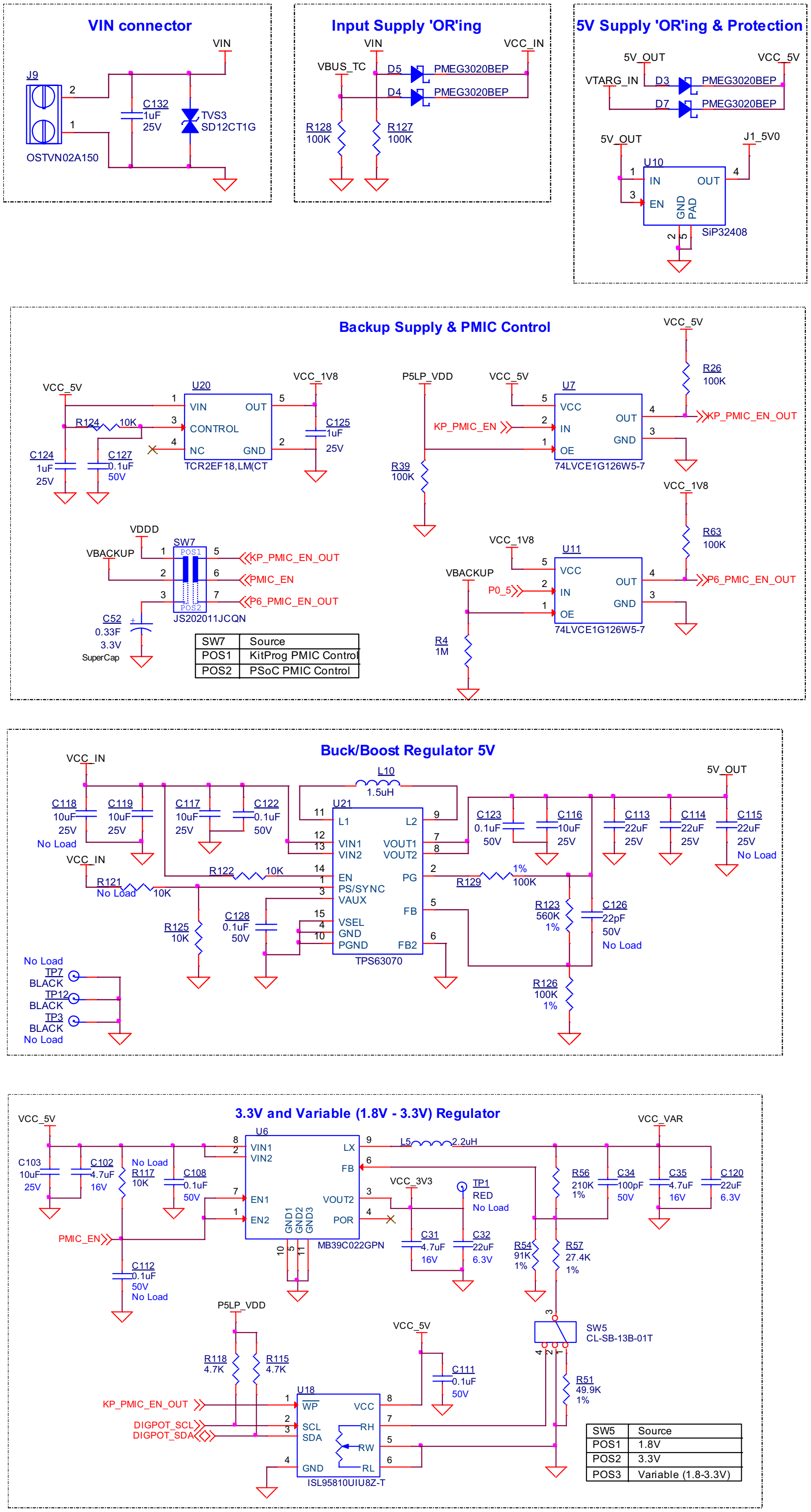
The voltage selection is made through switch **SW5**. In addition, an onboard 330 mF super-capacitor **(C52)** can be used to power the backup domain power (Vbackup) of PSoC 6 MCU. Switch **SW7** selects the Vbackup supply connection of PSoC 6 MCU between V_{DDD} and the super-capacitor. When V_{DDD} is selected, the variable regulator ON/OFF terminal is controlled by KitProg2. When the super-capacitor is selected, the regulator ON/OFF terminal is controlled by PSoC 6 MCU. To ensure proper operation of PSoC 6 MCU, the super-capacitor, when selected, must be charged internally by PSoC 6 MCU before turning OFF the regulator. For more details of the PSoC 6 MCU backup system and power supply, see the PSoC 6 BLE Technical Reference Manual.

Table A-1 details the different powering scenarios for Pioneer board.

Table A-1. Power supply scenarios

Power inputs					**Board condition**		
USB	**VIN**	**ETM header (VTARG_IN)**	**Battery Connected**	**JTAG/SWD header (VTARG_REF)**	**Main Regulator Powered by**	**PSoC Powered by**	**Battery charging**
Non Type-C power adapter (Legacy USB), 5V	<5	N/A	Yes	N/A	Type-C	Main Regulator	100mA
	>5	N/A	N/A	N/A	VIN	Main Regulator	No
Type-C, PD power adapter	< PD power adapter	N/A	Yes	N/A	Type-C	Main Regulator	1.5A
	> PD power adapter, <12V	N/A	N/A	N/A	VIN	Main Regulator	No
0V	5V–12V	N/A	N/A	N/A	VIN	Main Regulator	No
0V	0V	5V	N/A	N/A	ETM (VTARG_IN)	Main Regulator	No
0V	0V	0V	3.2V–4.2V	N/A	Battery	Main Regulator	No
0V	0V	0V	0V	1.8V–3.3V	N/A	JTAG/ SWD (VTARG_ REF)	No

Figure A-3. Schematics of Power Supply System



SW7	Source
POS1	KitProg PMIC Control
POS2	PSoC PMIC Control

SW5	Source
POS1	1.8V
POS2	3.3V
POS3	Variable (1.8-3.3V)

A.2.6 Expansion Connectors

A.2.6.1 Arduino-compatible Headers (J1, J2, J3, J4, and J5)

The board has five Arduino-compatible headers: **J1**, **J2**, **J3**, **J4**, and **J5** (**J5** is not populated by default). You can connect 3.3 V Arduino-compatible shields to develop applications based on the shield's hardware. Note that 5 V shields are not supported and connecting a 5 V shield may permanently damage the board. See Appendix A.3 for details on PSoC 6 MCU pin mapping to these headers.

A.2.6.2 PSoC 6 MCU I/O Headers (J18, J19, and J20)

These headers provide connectivity to PSoC 6 MCU GPIOs that are not connected to the Arduino compatible headers. Majority of these pins are multiplexed with onboard peripherals and are not connected to PSoC 6 MCU by default. For the detailed information on how to rework the kit to access these pins, see "PSoC 6 BLE Pioneer Board Reworks" on page 50.

A.2.6.3 PSoC 5LP GPIO Header (J6)

J6 is a 8x2 header provided on the board to bring out several pins of the PSoC 5LP to support advanced features such as a low-speed oscilloscope and a low-speed digital logic analyzer. This header also contains the USB-UART, USB-I2C, and USB-SPI bridge pins that can be used when these pins are not accessible on the Arduino headers because a shield is connected. The additional PSoC 5LP pins are connected directly to the internal programmable analog logic of PSoC 5LP. This header also has GPIOs for custom application usage. J6 is not populated by default. Note that the SPI, RTS, and CTS lines on these headers are directly from PSoC 5LP (before level translator).

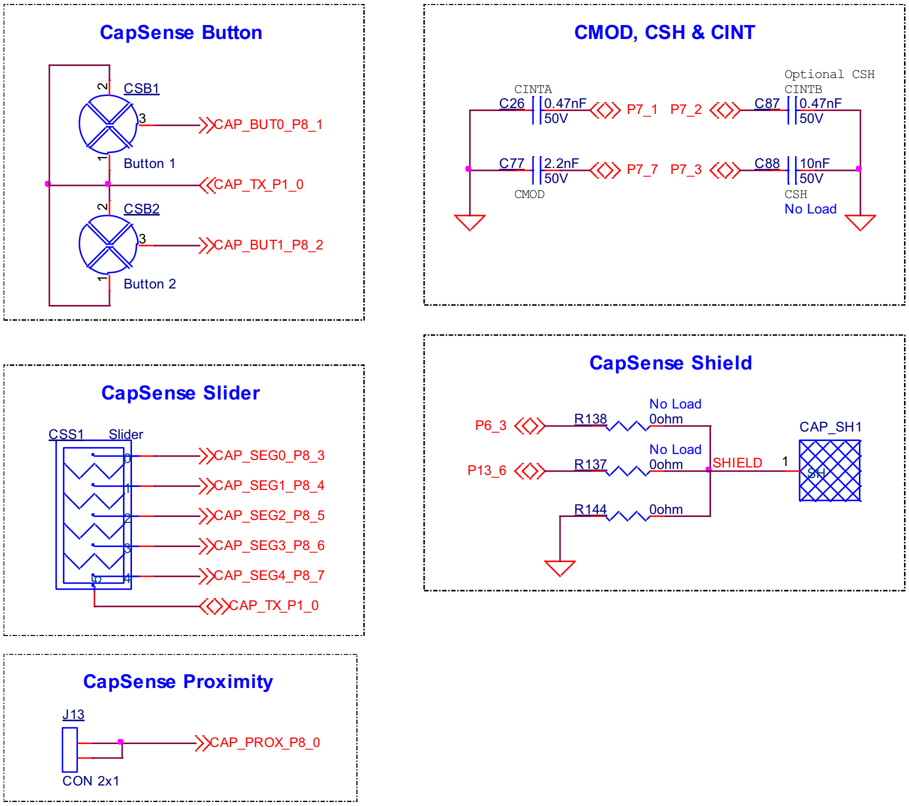
A.2.6.4 KitProg2 Custom Application Header (J7)

A 5x2 header is provided on the board to bring out more GPIOs of PSoC 5LP for custom application usage. This header also brings out the PSoC 5LP programming pins and can be programmed using MiniProg3 and 5-pin programming connector. J7 is not populated by default.

A.2.7 CapSense Circuit

A CapSense slider, two buttons, all of which support both self-capacitance (CSD) and mutual-capacitance (CSX) sensing, and a CSD proximity sensor (header) are connected to PSoC 6 MCU as Figure A-4 shows. Four external capacitors - C_{MOD} and C_{SH} for CSD, C_{INTA} and C_{INTB} for CSX are present on the Pioneer board. Note that CSH is not loaded by default. For details on using CapSense including design guidelines, see the *Getting Started with CapSense Design Guide*.

Figure A-4. Schematics of CapSense Circuit



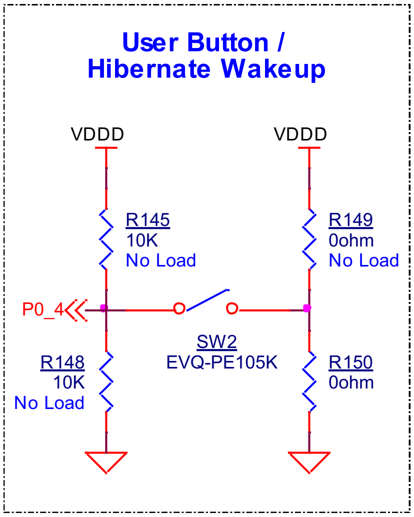
A.2.8 LEDs

LED1, **LED2**, and **LED3** (Red, Amber, and Green respectively) indicate the status of the KitProg2 (See the KitProg2 User Guide for details). **LED4** (amber LED) indicates the status of power supplied to PSoC 6 MCU. **LED7** (Green) indicates the status of power delivery output on **J16**. **LED6** (Red) indicates the battery charger status.

The Pioneer board also has two user controllable LEDs **(LED8** and **LED9)** and an RGB LED **(LED5)** connected to PSoC 6 MCU pins for user applications.

A.2.9 Push Buttons

The PSoC 6 BLE Pioneer Kit has a reset button and three user buttons. The reset button **(SW1)** is connected to the XRES pin of the PSoC 6 MCU, and is used to reset the device. One user button **(SW2)** is connected to pin P0[4] of the PSoC 6 MCU. The remaining two buttons - **SW3** and **SW4** are connected to the PSoC 5LP device for programming mode and custom app selection respectively (Refer to the KitProg2 User Guide for details). All the buttons connect to ground on activation (active low) by default. User button **(SW2)** can be changed to active high mode by changing the zero resistors shown below.



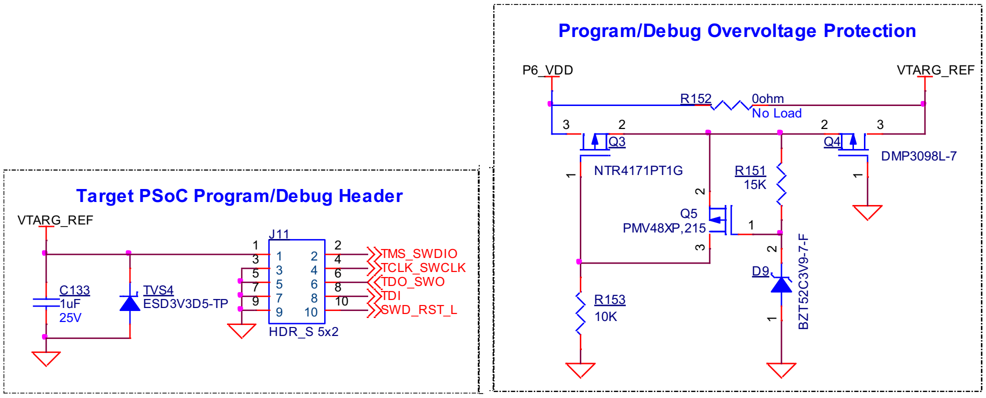
A.2.10 Cypress NOR Flash

The Pioneer board has a Cypress NOR flash memory (S25FL512SAGMFI011) of 512 Mb capacity. The NOR Flash is connected to the serial memory interface (SMIF) of PSoC 6 MCU. The NOR device can be used for both data and code memory with execute-in-place (XIP) support and encryption.

A.3 PSoC 6 BLE Pioneer Board Reworks

A.3.1 Bypassing protection circuit on PSoC 6 MCU Program and Debug Header (J11)

The 10-pin header allows you to program and debug PSoC 6 MCU using an external programmer such as MiniProg3. This header has a protection circuit that cuts-off any voltage greater that 3.4 V on VTARG_REF pin. This is to ensure that PSoC 6 MCU and other 3.3 V devices do not get damaged due to overvoltage.

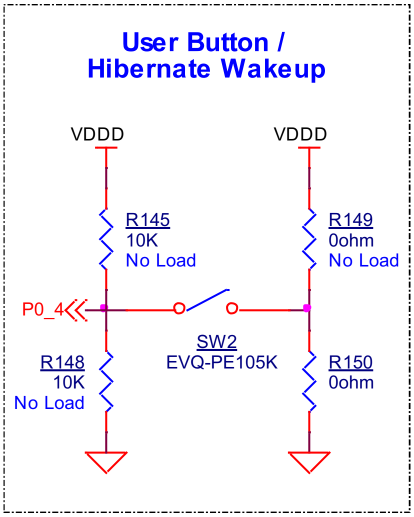


In case the external programmer provides slightly higher voltage, say 3.42 V, and you need to still use the programmer, you can bypass this protection circuit by populating the bypass zero-ohm resistor R152.

Do note that this change will compromise the protection circuit when an external supply is used and will permanently damage any 3.3 V device if the external voltage exceeds absolute maximum limit of devices. For example, 3.6 V for PSoC 6 MCU device, see the respective device datasheet for absolute maximum voltage limits.

A.3.2 PSoC 6 MCU User Button (SW2)

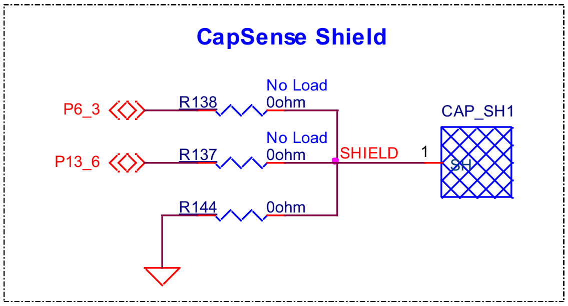
By default, this button connects the PSoC 6 MCU pin to ground when pressed, and you need to configure the PSoC 6 MCU pin as a digital input with resistive pull-up for detecting the button press. In case you need to sense active HIGH on PSoC 6 MCU pin, resistor R150 should be removed and R149 should be populated. This will connect the button connecting the PSoC 6 MCU pin to V_{DDD} when pressed. Additionally, there are footprints provided for pull-up and pull-down resistors that can be populated in case external pull-up is required.



A.3.3 CapSense Shield

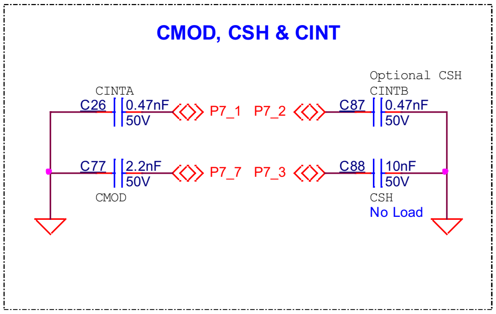
The hatched pattern around the CapSense buttons and slider are connected to ground. In case liquid tolerance is required, this pattern needs to be connected to shield pin. This pattern can be connected to either of the two ports P6.3 or P13.6 populated by R138 or R137, respectively. In both cases, resistor R144 connecting the hatched pattern to ground needs to be removed. These pins need to be configured as shield pin in PSoC creator.

Connecting the hatched pattern to shield instead of ground will also reduce parasitic capacitance of the sensors.



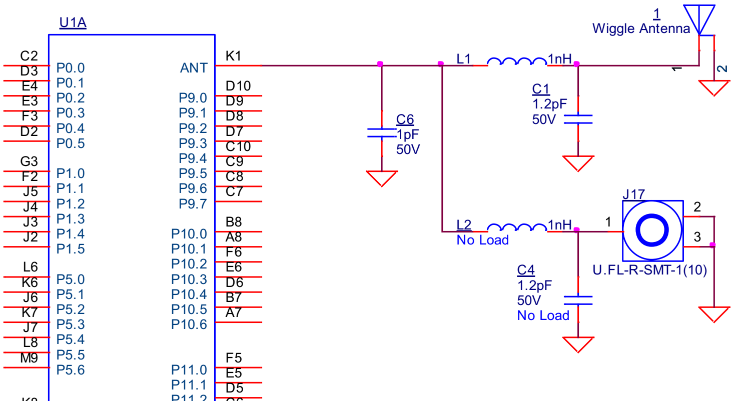
A.3.4 CSH

The shield tank capacitor (CSH) is not populated by default. This capacitor is optional, and can be used for an improved shield electrode driver when CSD sensing is used. You can remove R88 to disconnect port 7.3 from header and populate C88 (10nF) for CSH. See the bill of material (BOM) for recommended part number.



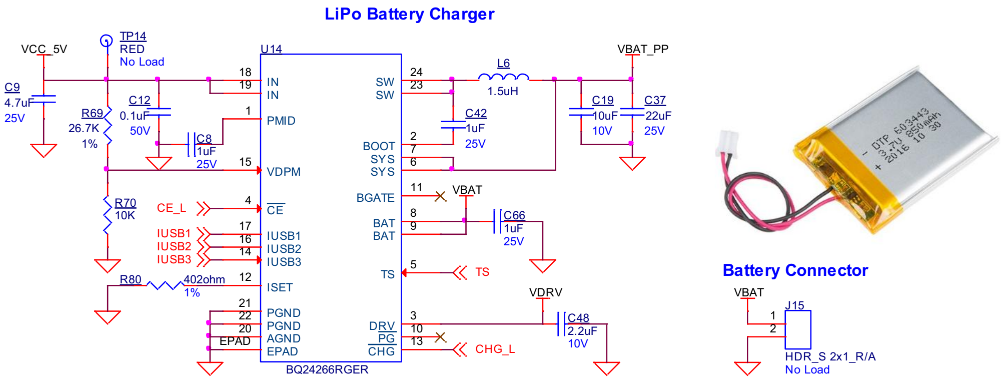
A.3.5 U.FL

This connector can be used for conductive measurements and can also be used to connect external antenna. This is not loaded by default. Remove L1, C1, populate L2, C4 and populate the U.FL connector (J17) to connect ANT pin from PSoC 6 MCU to connector. See the BOM for recommended part numbers.



A.3.6 LiPo Battery Charger

Battery connector (J15) for lithium-ion polymer battery charger is not loaded by default, this need to populate to evaluate battery charging and battery powering option. See the BOM for recommended part numbers. Recommended lithium-ion polymer rate is 3.7 V @850 mAH or higher. SparkFun Electronics PRT-13854 or equivalent. batteries can be used



A.3.7 Multiplexed GPIOs

Some of the PSoC 6 MCU pins are multiplexed with onboard peripherals and are not connected to connectors or other secondary components by default. See the PSoC 6 pin mapping table for details on modification required to access these pins.

A.4 Bill of Materials

Refer to the BOM files in the following paths in the kit software installed:

1. `<Install_Directory>\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit\1.0\Hardware\CY8CKIT-062-BLE\CY8CKIT-062-BLE PCBA BOM.xlsx`
2. `<Install_Directory>\CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit\1.0\Hardware\CY8CKIT-028-EPD\CY8CKIT-028-EPD PCBA BOM.xlsx`

A.5 Frequently Asked Questions

1. I don’t have a Type- C connector on my PC. Can I still connect and use this kit?
2. How does CY8CKIT-062-BLE handle voltage connection when both multiple power sources are plugged in?
3. How can I access Smart I/O and other GPIOs connected to onboard peripherals?
4. Why does the Red LED of RGB LED (LED5) light up when switch SW7 is set to SuperCap position?
5. What can I use the U.FL connector for and what is the typical mating cycle for these connectors?
6. What are the three selection switches on baseboard for?
7. What is the Jumper onboard for?
8. What are the input voltage tolerances? Are there any overvoltage protection on this kit?
9. Why is the voltage of the kit restricted to 3.3 V? Can’t it drive external 5 V interfaces?
10. By mistake, I powered my Arduino board while powering PSoC 6 MCU. Is my PSoC 6 chip alive?
11. What type of battery can I use for this kit?
12. By mistake, I connected the battery with opposite polarity. Did I fry the system?
13. Can I charge any kind of Type-C device using this kit?
14. How can I evaluate the USB Type-C provider and consumer features to get started with?
15. Why is the screen of EPD permanently ON?
16. Why does the screen of EPD glitch from black to white during every transition?
17. When I touch the resistors near the CapSense slider/button, the EPD Shield and LED are triggered. Why does this happen?
18. The EPD shield shows white and black lines after I switched OFF the kit. Is the EPD shield damaged?
19. I am unable to program the target device.
20. Does the kit get powered when I power the kit from another Cypress Kit through the J1 header?
21. What additional overlays can be used with the CapSense?
22. What is PMOD?
23. With what type of shield from Cypress can I use this baseboard?
24. Can I use this Kit as a programmer to program external PSoC devices?
25. Which third-party debuggers does this Kit support?

1. I don’t have a Type- C connector on my PC. Can I still connect and use this kit?

   Yes. To evaluate PSoC 6 MCU features, any PC with USB2.0 connectivity is sufficient. Type-C power adapter is required only to evaluate the CCG3 section of the Kit.

2. How does CY8CKIT-062-BLE handle voltage connection when both multiple power sources are plugged in?

   There are five different options to power the baseboard; Type-C USB connector (J10), External DC supply via VIN connector (J9/ J1), Debug and trace header (J12, VTARG_IN), Program and debug header (J11), and LiPo battery header (J15). Both Type-C and VIN take priority over other supply options. These inputs are ORed using diode and the higher voltage among the two take precedence. Output of ORing diode is given to a buck-boost regulator (**U16**) that generate a constant 5.2 V. This output is ORed with ETM supply (J12) which is typically 5 V. For most of the practical uses, output from the 5.2 V regulator takes priority and the same is given as an input to Cypress buck regulator (U6). LiPo battery voltage is used when all the above sources are absent. Output of Cypress buck regulator (U6) is ORed with supply voltage from the Program and debug header (J11), and higher voltage takes precedence. See the Power supply scenarios table for more details on voltage input and output scenarios.

3. How can I access Smart I/O and other GPIOs connected to onboard peripherals?

   Some of the Smart I/O (Port 8 and Port 9.3) and GPIO connected to onboard peripherals are multiplexed with PSoC 6 MCU I/O headers (**J18**, **J19**, and **J20**). By default, some of these I/Os are connected to onboard peripherals using series resistors. These resistors can be changed to route these I/Os to headers. See Table 1-2 on page 10. Pioneer board Pinout for details on list of resistors that needs to be changed.

4. Why does the Red LED of RGB LED (LED5) light up when switch **SW7** is set to SuperCap position?

   This behavior is observed if SuperCap is charged below 1.5 V. The I/Os referring to this domain will leak current, in this case P0[3]. VBACKUP feature needs to be enabled in silicon before switching **SW7** to SuperCap position. See the RTM/ datasheet on options to enable SuperCap charging.

5. What can I use the U.FL connector for and what is the typical mating cycle for these connectors?

   U.FL can be used for conductive measurements and to connect external antenna. U.FL connectors are not designed for reconnection. They are rated only for approximately 30 mating cycles.

6. What are the three selection switches on baseboard for?

   Table 1-1 on page 9 gives details on all three selection switches. Additionally, each code example documentation explains the selection switch setting required for each code example.

7. What is the Jumper on board for?

   The jumper J8 can be used to measure current of PSoC 6 MCU device without the need to desolder any component from the board. An ammeter can be connected across this jumper to measure the current consumed by the PSoC 6 MCU device. Remove the Jumper on J8, connect an ammeter (+ve terminal of ammeter to Pin 2), and power the kit though USB connector J10.

8. What are the input voltage tolerances? Are there any overvoltage protection on this kit?

   Input voltage level are as follows:

Table A-2. Input voltage levels

Supply	Typical i/p voltage	Absolute max (overvoltage protection)
USB Type-C connector (J10)	4.5V to 12V	15V
VIN connector (J9/J1)	5V to 12V	15V
Debug and trace header (J12)	5V	5.5V
Program and Debug header (J11)	1.8V to 3.3V	3.6V
Li-Po battery connected (J15)	3.2V to 4.2V	5V

9. Why is the voltage of the kit restricted to 3.3 V? Can't it drive external 5 V interfaces?

   PSoC 6 is not meant to be powered for more than 3.6 V. Powering PSoC 6 to more than 3.3 V will damage the chip. You cannot drive IO system with > 3.3 V supply voltages.

10. By mistake, I powered my Arduino board while powering PSoC 6 MCU. Is my PSoC 6 chip alive?

    Yes. The 3.3 V and 5 V on Arduino power header are not input pins and have protection circuit to prevent the voltage from entering the board. VIN is an input pin and this is routed to the regulator that is capable of taking an absolute maximum of 15 V. P6.V_{DD} pin is not protected and care should be taken not to supply voltage to this pin.

11. What type of battery can I use for this kit?

    Recommended lithium-ion polymer rating is 3.7 V @850 mAH or higher. SparkFun Electronics PRT-13854 or equivalent batteries can be used. The LiPo battery charger can charge at 100 mA or 1.5 mA based on whether the USB connection is a legacy device or PD capable.

12. By mistake, I connected the battery with opposite polarity. Did I fry the system?

    There are relevant protection circuits to protect the system from permanent damage. Prolonged connection may lead to damage.

13. Can I charge any kind of Type-C device using this kit?

    Kit is programmed to advertise the VIN voltage with 1 A current rating. 9 V and 12 V devices are the recommended options. VIN needs to be 9 V and 12 V respectively for this to work.

14. How can I evaluate the USB Type-C provider and consumer features to get started with?

    You can use any kind of Type-C laptop, mobile phone, or PD adapters based on the feature that you are trying to evaluate. To use as consumer, note that devices like laptop may be able to provide only 5 V out and may not support 9 V/12 V without a docking station. To use as provider, any 5 V/9 V/12 V device that has a current requirement of less that 1 A may be used. Additionally, Cypress has its own USB Type-C evaluation Kit which can be used to evaluate the provider and consumer features and many more. Visit http://www.cypress.com/products/usb-type-c-and-power-delivery for details on these kits.

15. Why is the screen of EPD permanently ON?

    Electronic paper and e-paper are display devices that mimic the appearance of ordinary ink on paper. Unlike conventional backlit flat panel displays that emit light, electronic paper displays reflect light like paper. This may make it more comfortable to read and provide a wider viewing angle than most light-emitting displays. The contrast ratio in electronic displays available as of 2008 approaches newspaper, and newly developed displays are slightly better. An ideal e-paper display can be read in direct sunlight without the image fading. Many electronic paper technologies hold static text and images indefinitely without electricity. To know the details, see https://en.wikipedia.org/wiki/Electronic_paper.

16. Why does the screen of EPD glitch from black to white during every transition?

    The screen of EPD shield refreshes at every transition. It first clears all the cells of the display by spreading it with white pixels and then posts the images.

17. When I touch the resistors near the CapSense slider/button, the EPD Shield and LED are triggered. Why does this happen?

    Those are the series limiting resistors which interface CapSense lines with ICs I/O lines. As you are touching CapSense lines, it triggers EPD shield which responds to CapSense.

18. The EPD shield shows white and black lines after I switched OFF the kit. Is the EPD shield damaged?

    EPD retains the image when you may have powered OFF the kit. In case the kit is powered OFF during transition between images, lines might appear.

19. I am unable to program the target device.

    a. Check SW6 to ensure it is in PSoC 6 MCU position.
    b. Make sure that no external devices are connected to J11.
    c. Update your KitProg2 Version in programmer to 1.04 or later using the steps mentioned in KitProg2 User Guide.
    d. Ensure that device used in PSoC Creator is CY8C6347BZI-BLD53

20. Does the kit get powered when I power the kit from another Cypress Kit through the J1 header?

    Yes, VIN pin on J1 header is the supply input/output pin and can take up to 12 V.

21. What additional overlays can be used with the CapSense?

    Any kind of overlays (up to 5 mm thickness) like wood, acrylic, and glass can be used with this CapSense. Note that additional tuning may be required when the overlay is changed.

22. What is PMOD?

    PMOD interface or Peripheral Module interface is an open standard defined by Digilent Inc in the Digilent Pmod Interface Specification for peripherals used with FPGAs or microcontrollers. Several types of modules are available from simple push buttons to more complex modules with network interfaces, analog to digital converters or LCD displays. PMODs are available from multiple vendors such as Diligent, Maxim Integrated, Analog Devices and a variety of hobby sites. This Kit supports only 1 x 6 pin PMOD modules.

23. With what type of shield from Cypress can I use this baseboard?

    Any Arduino Uno shield which supports 3.3 V operation is compatible with this Pioneer board. Following cypress shields are pin compatible with this pioneer board:

    a. CY3280-MBR3
    b. CY8CKIT-022
    c. CY8CKIT-024
    d. CY8CKIT-026
    e. CY8CKIT-040
    f. CY8CKIT-046
    g. CY8CKIT-048

24. Can I use this Kit as a programmer to program external PSoC devices?

    Yes, the onboard KitProg2 can program any PSoC 4/5/6 devices connected to J11 header. Switch SW6 should be switched to “External Device” position to program devices connected via J11 header.

25. Which third-party debuggers does this Kit support?

    Multiple third-party IDEs are supported; IAR and uVision are some examples. For more details on all supported devices and procedures to export to these IDEs, see PSoC Creator ‘Help’ menu.

Revision History

Document Revision History

Document Title: CY8CKIT-062-BLE, PSoC® 6BLE Pioneer Kit Guide **Document Number: 002-17040**				
Revision	**ECN Number**	**Issue Date**	**Origin of Change**	**Description of Change**
*B	5866901	09/13/2017	RKPM / NIDH	Initial version for public release.