CY8CEVAL-062S2 PSoC™ 62S2 evaluation kit guide

About this document

Scope and purpose

This document serves as a guide for using the CY8CEVAL-062S2 PSoC™ 62S2 evaluation kit. The document explains about the kit operation, describes the out-of-the-box example and its operation, and hardware details of the board.

Intended audience

This evaluation board is intended for all technical specialists who are familiar with connectivity and this board is intended to be used under laboratory conditions.

Reference documents

This user guide should be read in conjunction with the following documents:

- **AN228571 - Getting started with PSoC™ 6 MCU on ModusToolbox™ software**
- **PSoC™ 62 CY8C62x8, CY8C62xA datasheet**
- **Sterling - LWB5+ M.2 module from Laird Connectivity datasheet**

Table of contents

Table of contents

Safety and regulatory compliance information

This kit is intended for development purposes only. Users are advised to test and evaluate this kit in an RF development environment.

Safety evaluation for this kit is done in factory default settings using default accessories shipped with the kit. All evaluations for safety are carried out using a 5-V (USB 2.0, @ 500 mA) supply. 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.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required authorizations are first obtained. Contact **support@cypress.com** for details.

	These kits contain electrostatic discharge (ESD) sensitive devices. Electrostatic charges readily accumulate on the human body and any equipment, which can cause a 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 kits in the protective shipping package.
	End-of-Life/Product Recycling The end-of-life cycle for this kit is five years from the date of manufacture mentioned on the back of the box. Contact your nearest recycler to discard the kit.

General safety instructions

ESD protection

ESD can damage boards and associated components. Infineon 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 the chassis ground (any unpainted metal surface) on your board when handling parts.

Handling boards

This board 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 CY8CEVAL-062S2 PSoC™ 62S2 evaluation kit. The PSoC™ 62S2 evaluation kit enables you to evaluate and develop your applications using the **PSoC™ 62 series MCU** (hereafter called "PSoC™ 6 MCU") and M.2 connectivity modules based on AIROC™ Wi-Fi & Bluetooth® combos.

PSoC™ 6 MCU is an ultra-low-power PSoC™ device 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 2 MB flash and 1 MB SRAM , Secure Digital Host Controller (SDHC) supporting SD/SDIO/eMMC interfaces, 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 evaluation board carries a PSoC™ 6 MCU and a M.2 interface connector for interfacing radio modules based on AIROC™ Wi-Fi & Bluetooth® combos. In addition, the board features an on-board programmer/debugger (KitProg3), a 512-Mbit QSPI NOR flash, a Quad SPI F-RAM, OPTIGA™ Trust M security controller, a micro-B connector for USB device interface, add-on board interface compatible with mikroBUS by Mikroelektronika, headers compatible with Arduino Uno R3, a 5-segment CAPSENSE™ slider, two CAPSENSE™ buttons, an RGB LED, two user LEDs, one potentiometer, and two push buttons. The board supports operating voltages from 1.8 V to 3.3 V for PSoC™ 6 MCU.

You can use ModusToolbox™ software to develop and debug your PSoC™ 6 MCU projects. **ModusToolbox™ software** is a set of tools that enable you to integrate these devices into your existing development methodology.

If you are new to PSoC™ 6 MCU and ModusToolbox™ software, see the application note **AN228571 – Getting started with PSoC™ 6 MCU on ModusToolbox™ software** to help you familiarize with the PSoC™ 6 MCU and help you create your own design using the Eclipse IDE for ModusToolbox™ software.

1.1 Kit contents

- PSoC™ 62S2 evaluation board
- **Sterling-LWB5+ Wi-Fi/Bluetooth® M.2 radio module** from Laird Connectivity (plugged into the PSoC™ 62S2 evaluation board) (453-00048)
- **FlexPIFA antenna** from Laird Connectivity (001-0021)
- USB Type-A to Micro-B cable
- Four jumper wires (4 inches each)
- Two jumper wires (5 inches each)
- Quick start guide

Figure 1 Kit contents

Inspect the contents of the kit; if you find any part missing, go to **www.cypress.com/support**.

1.2 Getting started

This guide will help you to get acquainted with this evaluation kit:

- The **Kit operation** chapter describes the major features of the PSoC™ 62S2 evaluation board and functionalities such as programming, debugging, the USB-UART and USB-I2C bridges.
- The **Hardware** chapter provides a detailed hardware description, kit schematics, and the bill of materials (BOM).
- Application development using PSoC™ 62S2 evaluation kit is supported in ModusToolbox™ software. ModusToolbox™ software is a free development eco-system that includes the Eclipse IDE for ModusToolbox™ software and the PSoC™ 6 SDK with PSoC™ 6 MCU. Using ModusToolbox™ software, you can enable and configure device resources, middleware libraries; write C/assembly source code; and program and debug the device. You can download the software from the **ModusToolbox™ home page**. See the ModusToolbox™ software installation guide for additional information.

- There are wide range of code examples to evaluate the PSoC™ 62S2 evaluation board. These examples help you familiarize PSoC™ 6 MCU and create your own design. These examples can be accessed through ModusToolbox™ Project Creator tool. Alternatively, you can also visit Infineon's code example page to access these examples:
  - **Code examples for ModusToolbox™ software**

1.3 Board details

The PSoC™ 62S2 evaluation board has the following features:

- PSoC™ 6 MCU – CY8C624ABZI-S2D44. See the device **datasheet**.
- M.2 interface connector to connect radio modules based on AIROC™ Wi-Fi & Bluetooth® combos. By default, a Sterling-LWB5+ M.2 module from Laird Connectivity is connected to M.2 connector. See the "Kit operation" chapter for details on using the Sterling-LWB5+ M.2 module from Laird Connectivity with this evaluation board.
- Add-on board interface compatible with mikroBUS by Mikroelektronika
- 512-Mbit external Quad SPI NOR Flash that provides a fast, expandable memory for data and code.
- 4-Mbit Quad SPI F-RAM
- KitProg3 on-board SWD programmer/debugger, USB-UART and USB-I2C bridge functionality
- CAPSENSE™ touch-sensing slider (5 elements), two buttons, based on self-capacitance (CSD)- and mutual-capacitance (CSX)-based sensing
- A micro-B connector for USB device interface
- Selectable input supply voltages of 1.8 V, 2.5 V, and 3.3 V for the PSoC™ 6 MCU. The PSoC™ 6 MCU should be connected to 2.5 V when eFuse programming is required. Otherwise, the input voltage should be connected either to 1.8 V or 3.3 V.
- Two user LEDs, an RGB LED, two user buttons, and a reset button for the PSoC™ 6 MCU
- A potentiometer which can be used to simulate analog sensor output
- A mode button and a mode LED for KitProg3

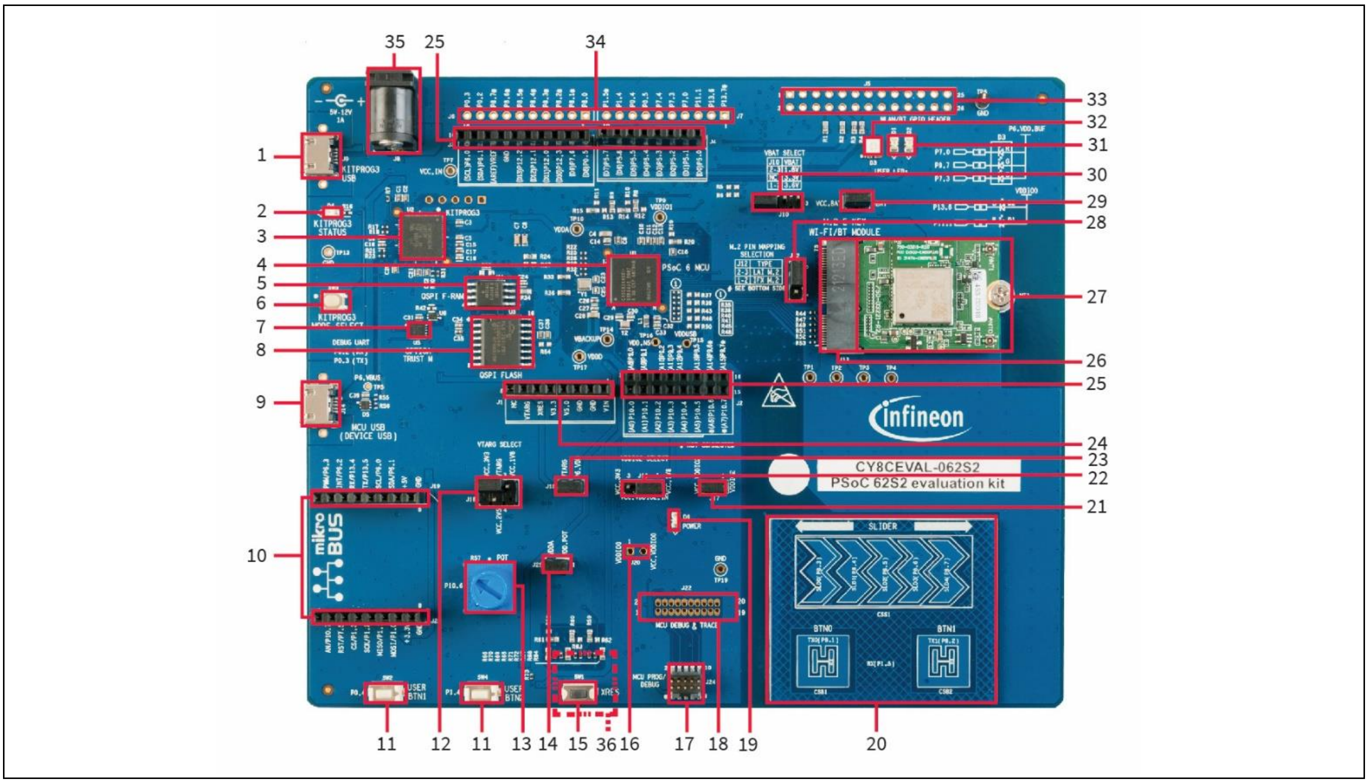


Figure 2 PSoC™ 62S2 evaluation board – top view

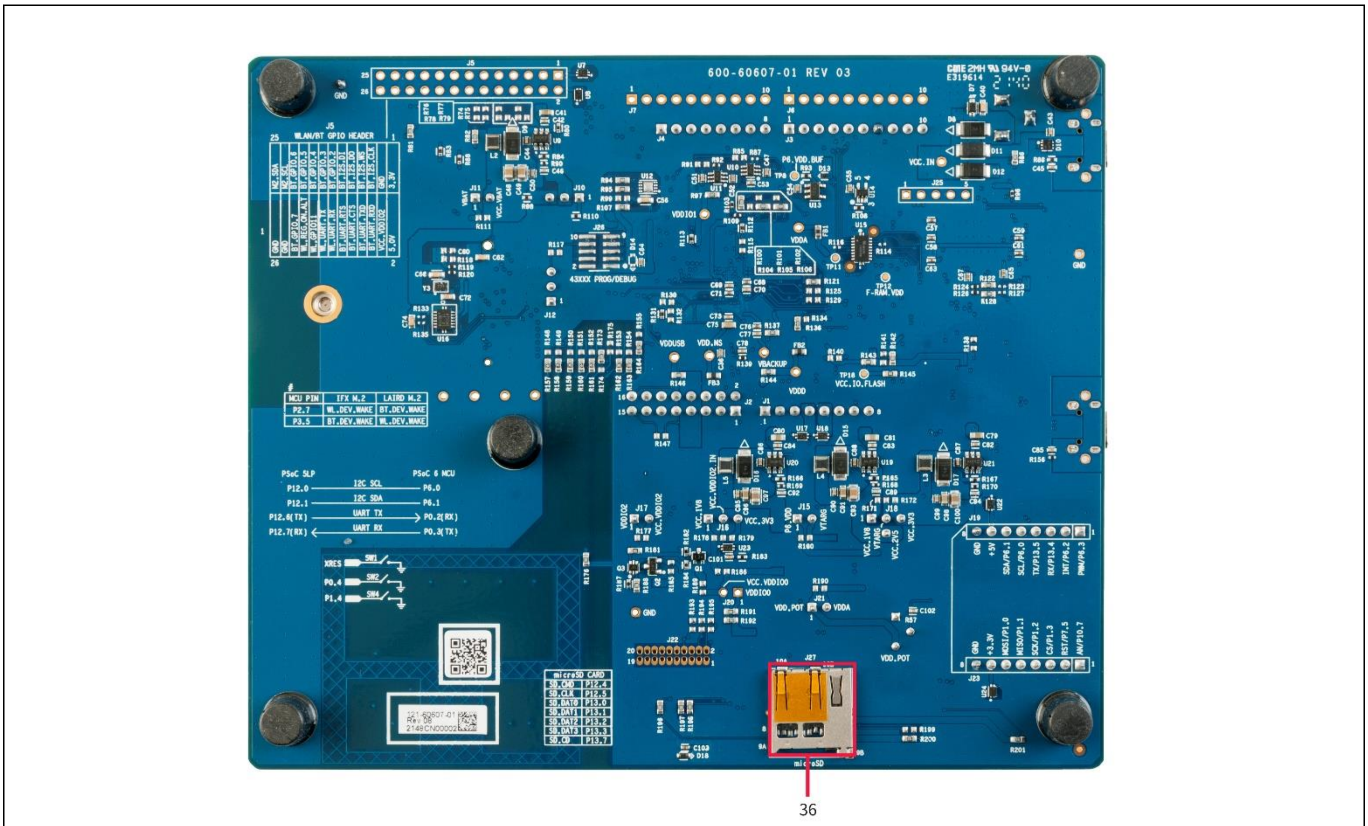


Figure 3 PSoC™ 62S2 evaluation board – bottom view

1. KitProg3 USB connector (J9)
2. KitProg3 status LED (D4)
3. KitProg3 (PSoC™ 5LP) programmer and debugger (CY8C5868LTI-LP039, U2)
4. PSoC™ 6 MCU (CY8C624ABZI-S2D44, U1)
5. QSPI F-RAM (CY15B104QSN, U4)
6. KitProg3 programming mode selection button (SW3)
7. OPTIGA™ Trust M security controller (SLS32AIA, U5)
8. 512-Mbit serial NOR flash memory (S25FL512S, U3)
9. PSoC™ 6 MCU USB device connector (J14)
10. Headers compatible with mikroBUS by Mikroelektronika (J19, J23)
11. PSoC™ 6 MCU user buttons (SW2 and SW4)
12. System power (VTARG) selection jumper (J18)
13. Potentiometer (R57)
14. Potentiometer connection jumper (J21)
15. PSoC™ 6 MCU reset button (SW1)
16. PSoC™ 6 MCU VDDIO0 current measurement jumper (J20)*
17. PSoC™ 6 MCU 10-pin SWD/JTAG program and debug header (J24)
18. PSoC™ 6 MCU debug and trace header (J22)*
19. Power LED (D6)
20. CAPSENSE™ slider (SLIDER) and buttons (BTN0 and BTN1)
21. PSoC™ 6 MCU VDDIO2 current measurement jumper (J17)
22. PSoC™ 6 MCU VDDIO2 power selection jumper (J16)
23. PSoC™ 6 MCU VTRAG current measurement jumper (J15)
24. Power header compatible with Arduino Uno R3 (J1)
25. I/O headers compatible with Arduino Uno R3 (J2, J3, and J4)
26. M.2 interface connector (J13)
27. Sterling-LWB5+ M.2 module from Laird Connectivity
28. Custom M.2 interface selection jumper (J12)
29. VBAT current measurement jumper (J11)
30. VBAT power selection jumper (J10)
31. PSoC™ 6 MCU user LEDs (D1 and D2)
32. RGB LED (D3)

33. AIROC™ Wi-Fi & Bluetooth® combo GPIO header (J5)*

34. PSoC™ 6 MCU I/O headers (J6, J7)*

35. External power supply VIN connector (J8)

36. microSD card holder (J27)

* Footprint only, not populated on the board

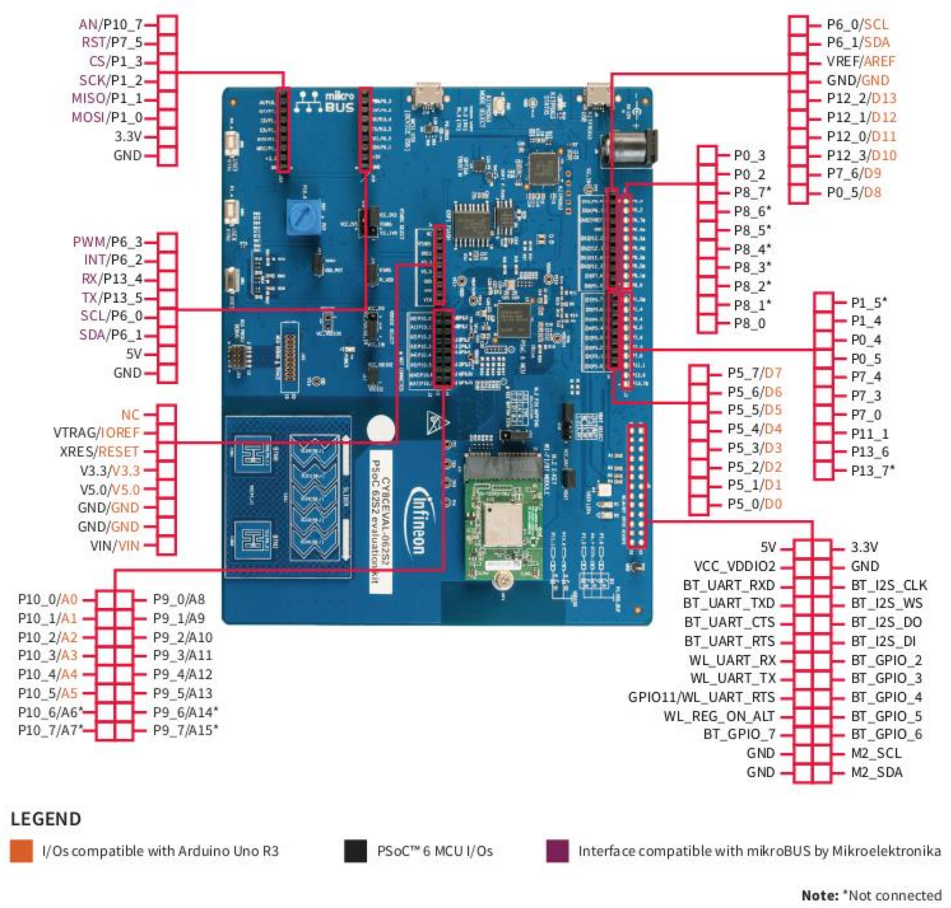


Figure 4 PSoC™ 62S2 evaluation board pinout

Table 1 PSoC™ 62S2 evaluation board pinout

Pin	Primary on-board function	Secondary on-board function	Connection details
PSoC™ 6 MCU pin			
XRES	Hardware reset	–	Remove R196 to disconnect it from KitProg3 reset.
P0[2]	Debug UART_RX	Generic GPIO header (J6.9)	This pin is connected to KitProg3 UART Tx pin.
P0[3]	Debug UART_TX	Generic GPIO header (J6.10)	This pin is connected to KitProg3 UART Rx pin.

Pin	Primary on-board function	Secondary on-board function	Connection details
P0[4]	User button with hibernate wakeup capability	Generic GPIO header (J7.8)	
P0[5]	D8 – compatible with Arduino (J3.1)	Generic GPIO header (J7.7)	
P1[0]	SPI MOSI pin in the interface compatible with mikroBUS by Mikroelektronika (J23.6)		
P1[1]	SPI MISO pin in the interface compatible with mikroBUS by Mikroelektronika (J23.5)		
P1[2]	SPI SCK pin in the interface compatible with mikroBUS by Mikroelektronika (J23.4)		
P1[3]	SPI CS pin in the interface compatible with mikroBUS by Mikroelektronika (J23.3)		
P1[4]	User button with Hibernate wakeup capability	Generic GPIO header (J7.9)	
P1[5]	CAPSENSE™ Rx for buttons	Generic GPIO header (J7.10)	Remove R164 to disconnect from the CAPSENSE™ RX pin and populate R155 to connect to J7.10.
P5[0]	Pin D0 of connector J4.1 compatible with Arduino	Debug UART_RX	Remove R8 and populate R12 to connect to KitProg3 UART Tx pin.
P5[1]	Pin D1 of connector J4.2 compatible with Arduino	Debug UART_TX	Remove R14 and populate R10 to connect to KitProg3 UART Rx pin.
P5[2]	Pin D2 of connector J4.3 compatible with Arduino	Debug UART_RTS	Populate R91 to connect to KitProg3 UART CTS pin.
P5[3]	Pin D3 of connector J4.4 compatible with Arduino	Debug UART_CTS	Populate R85 to connect to KitProg3 UART RTS pin.
P5[4]	Pin D4 of connector J4.5 compatible with Arduino		
P5[5]	Pin D5 of connector J4.6 compatible with Arduino		
P5[6]	Pin D6 of connector J4.7 compatible with Arduino		
P5[7]	Pin D7 of connector J4.8 compatible with Arduino		

Pin	Primary on-board function	Secondary on-board function	Connection details
P6[0]	I2C SCL - Common I2C SCL pin for KitProg3 USB-I2C bridge, I2C interface compatible with mikroBUS by Mikroelektronika, OPTIGA™ Trust M device	I2C SCL pin on Arduino header (J3.10)	Remove R122 to disconnect from KitProg3 I2C SCL pin.
P6[1]	I2C SDA - Common I2C SDA pin for KitProg3 USB-I2C bridge, I2C interface compatible with mikroBUS by Mikroelektronika, OPTIGA™ Trust M device	I2C SDA pin Arduino header (J3.9)	Remove R128 to disconnect from KitProg3 I2C SDA pin.
P6[2]	INT pin in the interface compatible with mikroBUS by Mikroelektronika (J19.2)		
P6[3]	PWM pin in the interface compatible with mikroBUS by Mikroelektronika (J19.1)		
P6[4]	TDO_SWO		
P6[5]	TDI		
P6[6]	SWDIO		
P6[7]	SWCLK		
P7[0]	RGB red LED	ETM Clock	To connect to ETM clock, remove R20 and populate R19.
P7[1]	CAPSENSE™ CINTA		
P7[2]	CAPSENSE™ CINTB		
P7[3]	RGB blue LED	Generic GPIO header (J7.5)	
P7[4]	GPIO on non-Arduino header (J7.6)	CAPSENSE™ shield	Remove R173 to disconnect from non-Arduino header pin (J7.6). Populate R174 to connect to CAPSENSE™ shield.
P7[5]	RST pin in the interface compatible with mikroBUS by Mikroelektronika (J23.2)		
P7[6]	Arduino D9 (J3.2)		
P8[0]	USB VBUS Detect	Generic GPIO header (J6.1)	
P8[1]	CAPSENSE™ Button0 Tx	Generic GPIO header (J6.2)	Remove R163 to disconnect from CAPSENSE™. Populate R154 to connect to GPIO on non-Arduino header.

Pin	Primary on-board function	Secondary on-board function	Connection details
P8[2]	CAPSENSE™ Button1 Tx	Generic GPIO header (J6.3)	Remove R162 to disconnect from CAPSENSE™. Populate R153 to connect to GPIO on non-Arduino header.
P8[3]	CAPSENSE™ Slider0	Generic GPIO header (J6.4)	Remove R161 to disconnect from CAPSENSE™. Populate R152 to connect to GPIO on non-Arduino header.
P8[4]	CAPSENSE™ Slider1	Generic GPIO header (J6.5)	Remove R160 to disconnect from CAPSENSE™. Populate R151 to connect to GPIO on non-Arduino header.
P8[5]	CAPSENSE™ Slider2	Generic GPIO header (J6.6)	Remove R159 to disconnect from CAPSENSE™. Populate R150 to connect to GPIO on non-Arduino header.
P8[6]	CAPSENSE™ Slider3	Generic GPIO header (J6.7)	Remove R158 to disconnect from CAPSENSE™. Populate R149 to connect to GPIO on non-Arduino header.
P8[7]	CAPSENSE™ Slider4	Generic GPIO header (J6.8)	Remove R157 to disconnect from CAPSENSE™. Populate R148 to connect to GPIO on non-Arduino header.
P9[0]	A8 – extended header compatible with Arduino (J2.2)	TRACEDATA [3]	Remove R9 to disconnect from A8 of the extended header compatible with Arduino (J2.2). Populate R13 to connect to ETM Trace header.
P9[1]	A9 – extended header compatible with Arduino (J2.4)	TRACEDATA [2]	Remove R106 to disconnect from A9 of the extended header compatible with Arduino (J2.4). Populate R102 to connect to ETM Trace header.
P9[2]	A10 – extended header compatible with Arduino (J2.6)	TRACEDATA [1]	Remove R15 to disconnect from A10 of the extended header compatible with Arduino (J2.6).

Pin	Primary on-board function	Secondary on-board function	Connection details
			Populate R11 to connect to ETM Trace header.
P9[3]	A11 – extended header compatible with Arduino (J2.8)	TRACEDATA [0]	Remove R105 to disconnect from A11 of the extended header compatible with Arduino (J2.8). Populate R101 to connect to ETM Trace header.
P9[4]	A12 – extended header compatible with Arduino (J2.10)		
P9[5]	A13 – extended header compatible with Arduino (J2.12)		
P9[6]	OPTIGA™ Trust M VCC enable pin	A14 – extended header compatible with Arduino (J2.14)	Remove R100 to disconnect from OPTIGA™ Trust M VCC enable pin. Populate R104 to connect to A14 of the extended header compatible with Arduino (J2.14).
P9[7]	RGB green LED	A15 – extended header compatible with Arduino (J2.16)	Remove R112 to disconnect from RGB green LED. Populate R115 to connect to A15 of the extended header compatible with Arduino (J2.16).
P10[0]	A0 – header compatible with Arduino (J2.1)		
P10[1]	A1 – header compatible with Arduino (J2.3)		
P10[2]	A2 – header compatible with Arduino (J2.5)		
P10[3]	A3 – header compatible with Arduino (J2.7)		
P10[4]	A4 – header compatible with Arduino (J2.9)		
P10[5]	A5 – header compatible with Arduino (J2.11)		
P10[6]	Potentiometer (POT) output	A6 – extended header compatible with Arduino (J2.13)	Remove R121 to disconnect from the potentiometer. Populate R125 to connect to A6 of the extended header compatible with Arduino (J2.13).

Pin	Primary on-board function	Secondary on-board function	Connection details
P10[7]	AN pin in the interface compatible with mikroBUS by Mikroelektronika (J23.1)	A7 – extended header compatible with Arduino (J2.15)	Populate R147 to connect to A7 of the extended header compatible with Arduino (J2.15).
P11[0]	QSPI F-RAM SSEL		
P11[1]	Red user LED (D1)	Generic GPIO header (J7.3)	Remove R83 to disconnect from the red user LED (D1).
P11[2]	QSPI Flash SSEL		
P11[3:6]	QSPI IO[3:0]		
P11[7]	QSPI CLK		
P12[0]	D11 – header compatible with Arduino (J3.4)		
P12[1]	D12 – header compatible with Arduino (J3.5)	microSD card detect GPIO	Populate R129 to connect to microSD card detect GPIO.
P12[2]	D13 – header compatible with Arduino (J3.6)		
P12[3]	D10 – header compatible with Arduino (J3.3)		
P12[4]	microSD card SD CMD		Remove R61 to disconnect from the microSD card connector.
P12[5]	microSD card SD CLK		Remove R60 to disconnect from the microSD card connector.
P12[6]	ECO Crystal XIN		
P12[7]	ECO Crystal XOUT		
P13[0]	microSD card SD DATA0	microSD card SPI MOSI	Remove R61 to disconnect from microSD (SDHC interface). Populate R58 to connect to microSD (SPI interface).
P13[1]	microSD card SD DATA1	microSD card SPI MISO	Remove R59 to disconnect from microSD (SDHC interface). Populate R62 to connect to microSD (SPI interface).
P13[2]	microSD card SD DATA2	microSD card SPI SCLK	Remove R60 to disconnect from microSD (SDHC interface). Populate R63 to connect to microSD (SPI interface).
P13[3]	microSD card SD DATA3	microSD card SPI SSEL	

Pin	Primary on-board function	Secondary on-board function	Connection details
P13[4]	UART Rx pin in the interface compatible with mikroBUS by Mikroelektronika (J19.3)		
P13[5]	UART Tx pin in the interface compatible with mikroBUS by Mikroelektronika (J19.4)		
P13[6]	Orange user LED (D2)	Generic GPIO header (J7.2), QSPI Flash Reset	Remove R131 to disconnect from the orange user LED (D2). Populate R132 to connect to QSPI flash reset.
P13[7]	SD card chip detect	Generic GPIO header (J7.1), QSPI Flash Interrupt (INT)	Remove R136 to disconnect from microSD card detect. Populate R134 to connect to QSPI flash INT. Populate R54 to connect to QSPI flash INT.
M.2 radio module pin			
BT_UART_TXD	UART interface with the host MCU (PSoC™ 6)	UART interface with KitProg3	Remove R2 to disconnect from PSoC™ 6 MCU UART Rx. Populate R76 to connect to KitProg3 Secondary UART Rx.
BT_UART_RXD	UART interface with the host MCU (PSoC™ 6)	UART interface with KitProg3	Remove R1 to disconnect from PSoC™ 6 MCU UART Tx. Populate R77 to connect to KitProg3 Secondary UART Tx.
BT_UART_CTS	UART interface with the host MCU (PSoC™ 6)	UART interface with KitProg3	Remove R3 to disconnect from PSoC™ 6 MCU UART RTS. Populate R75 to connect to KitProg3 Secondary UART RTS.
BT_UART_RTS	UART interface with the host MCU (PSoC™ 6)	UART interface with KitProg3	Remove R4 to disconnect from PSoC™ 6 MCU UART CTS. Populate R74 to connect to KitProg3 Secondary UART CTS.
BT_I2S_CLK	I2S serial clock	–	
BT_I2S_WS	I2S serial word select	–	
BT_I2S_DO	I2S serial data out	–	
BT_I2S_DI	I2S serial data in	–	

Pin	Primary on-board function	Secondary on-board function	Connection details
BT_GPIO_2	Bluetooth® GPIO	–	
BT_GPIO_3	Bluetooth® GPIO	–	
BT_GPIO_4	Bluetooth® GPIO	–	
BT_GPIO_5	Bluetooth® GPIO	–	
BT_GPIO_6	Bluetooth® GPIO	–	
BT_GPIO_7	Bluetooth® GPIO	–	
WL_UART_RX	Wi-Fi debug UART Rx pin	–	Populate R79 to connect to KitProg3 secondary UART Tx.
WL_UART_TX	Wi-Fi debug UART Tx Pin	–	Populate R78 to connect to KitProg3 secondary UART Rx.
WL_GPIO11	Programable GPIO Pin	–	
WL_REG_ON _ALT	Alternate WL Regulator ON pin	–	
M.2 SCL	I2C SCL pin to EEPROM SCL	–	Remove R95 to disconnect from the EEPROM SCL pin. Populate R6 to connect to P6_0 (SCL pin of PSoC™ 6 MCU).
M.2 SDA	I2C SDA pin to EEPROM SDA	–	Remove R94 to disconnect from the EEPROM SDA pin. Populate R5 to connect to P6_1 (SDA pin of PSoC™ 6 MCU).

1.4 Additional learning resources

Infineon 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.

1.5 Technical support

For assistance, go to **www.cypress.com/support**. Visit **community.infineon.com** to ask your questions in Infineon developer community.

You can also use the following support resources if you need quick assistance:

- **Self-help (Technical Documents)**
- **Local Sales Office Locations**

1.6 Documentation conventions

Table 2 Document conventions for guides

Convention	Usage
`Courier New`	Displays user-entered text and source code
Italics	Displays file names and reference documentation: Read about the *sourcefile.hex* file in the *PSoC™ Creator user guide*.
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.

1.7 Abbreviations and definitions

Table 3 Abbreviations

Abbreviation	Definition
ADC	Analog-to-Digital Converter
BOM	Bill of Materials
CINT	Integration Capacitor
CMOD	Modulator Capacitor
CPU	Central Processing Unit
CSD	CAPSENSE™ Sigma Delta
CSX	CAPSENSE™ Crosspoint
CTANK	Shield Tank Capacitor
DC	Direct Current
ECO	External Crystal Oscillator
ESD	Electrostatic Discharge
FPC	Flexible Printed Circuit
GPIO	General-Purpose Input/Output
IC	Integrated Circuit
IDE	Integrated Development Environment

Abbreviation	Definition
IoT	Internet of Things
I2C	Inter-Integrated Circuit
I2S	Inter-IC Sound
LED	Light-emitting Diode
LPO	Low Power Oscillator
PC	Personal Computer
PDL	Peripheral Driver Library
PDM	Pulse Density Modulation
PSoC™	Programmable System-on-Chip
QSPI	Quad Serial Peripheral Interface
SDHC	Secure Digital Host Controller
SDIO	Secure Digital Input Output
SDK	Software Development Kit
SMIF	Serial Memory Interface
SPI	Serial Peripheral Interface
SRAM	Static Random-Access Memory
SWD	Serial Wire Debug
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
WCO	Watch Crystal Oscillator

2 Kit operation

2.1 Theory of operation

The PSoC™ 62S2 evaluation board is built around PSoC™ 6 MCU. For details of device features, see the device **datasheet**.

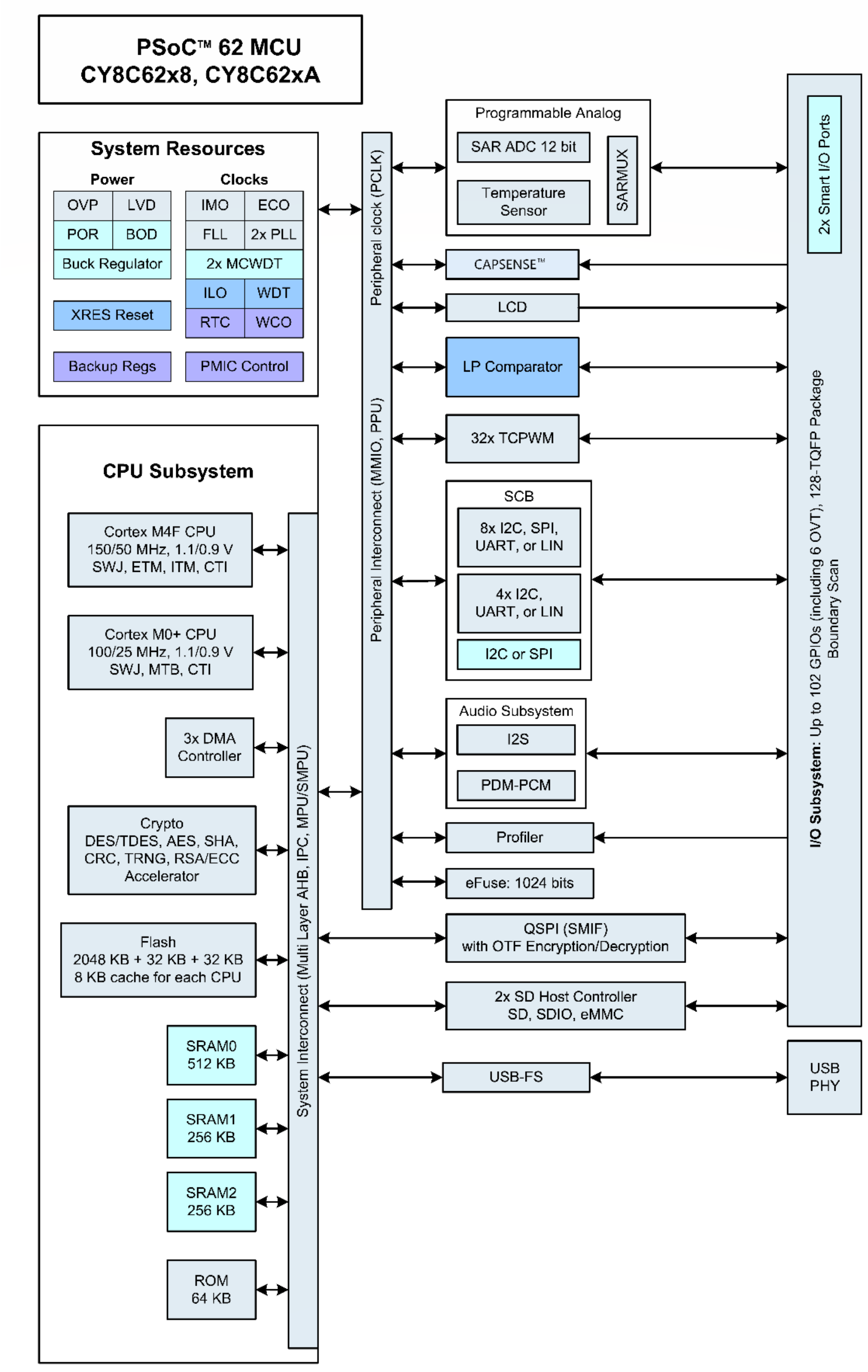


Figure 5 **PSoC™ 62 MCU block diagram**

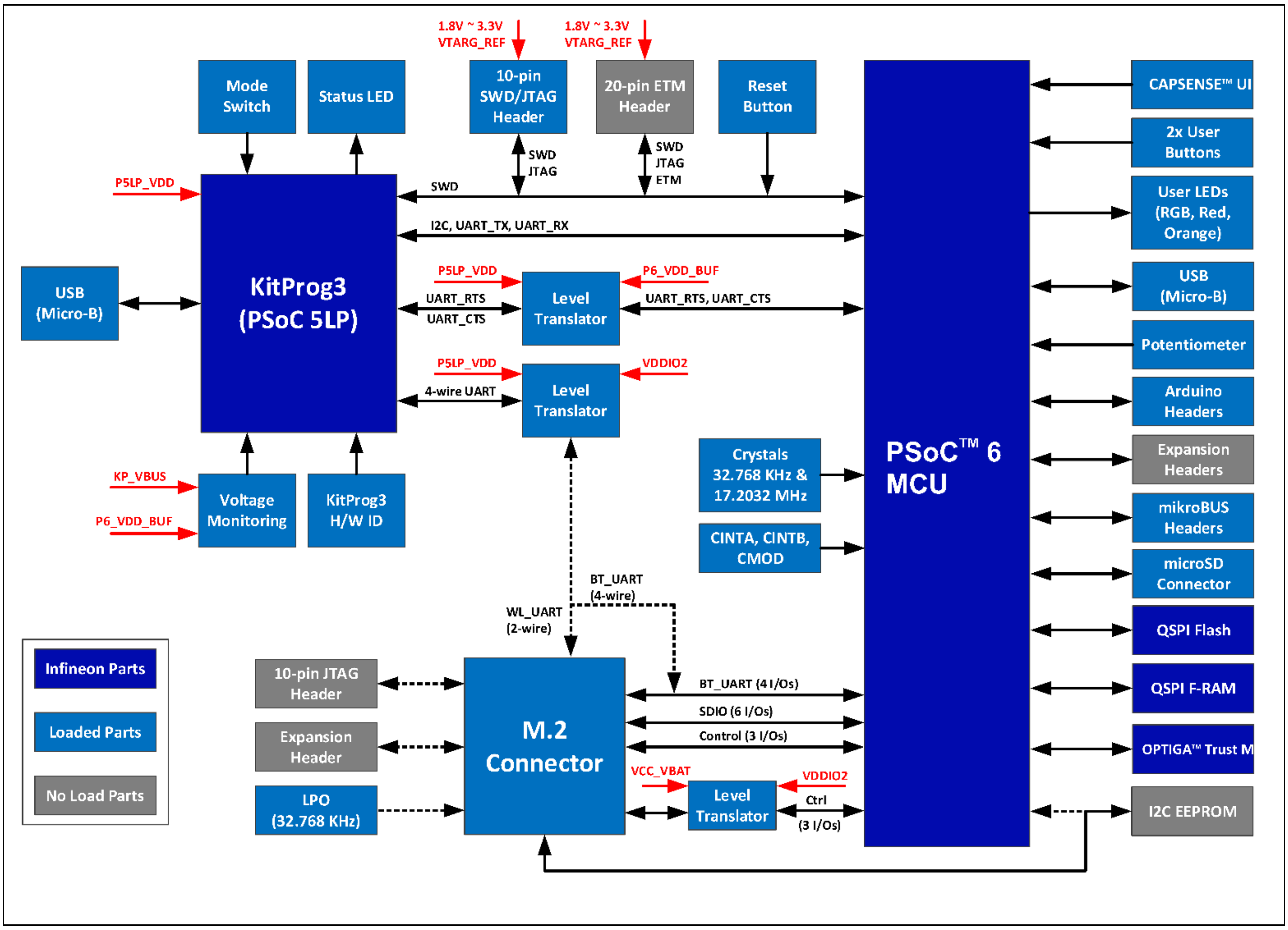


Figure 6 Block diagram of PSoC™ 62S2 evaluation board

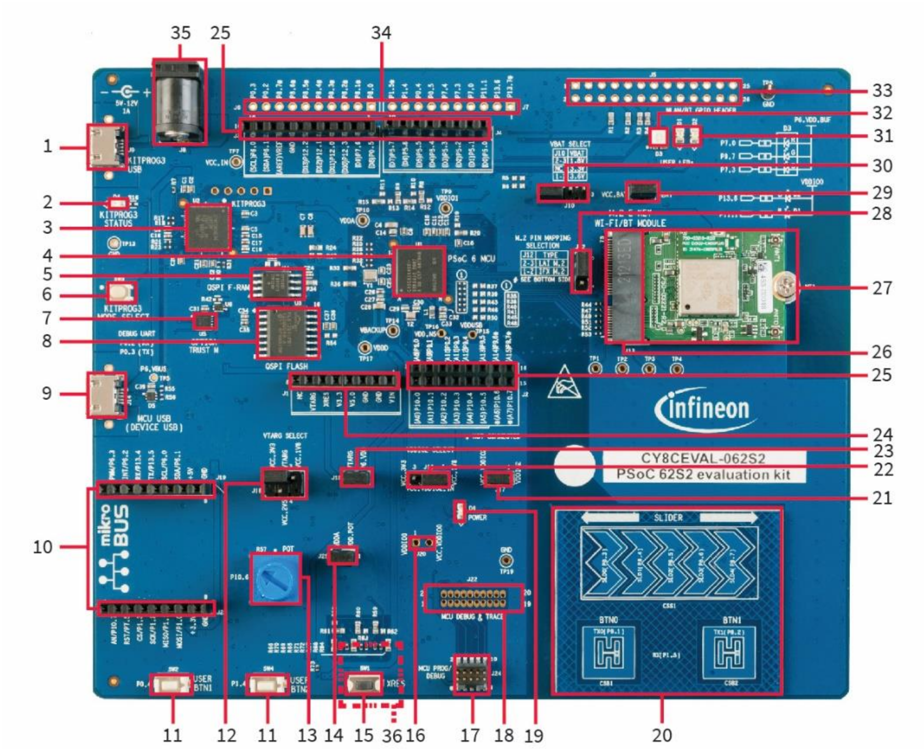


Figure 7 **PSoC™ 62S2 evaluation board – top view**

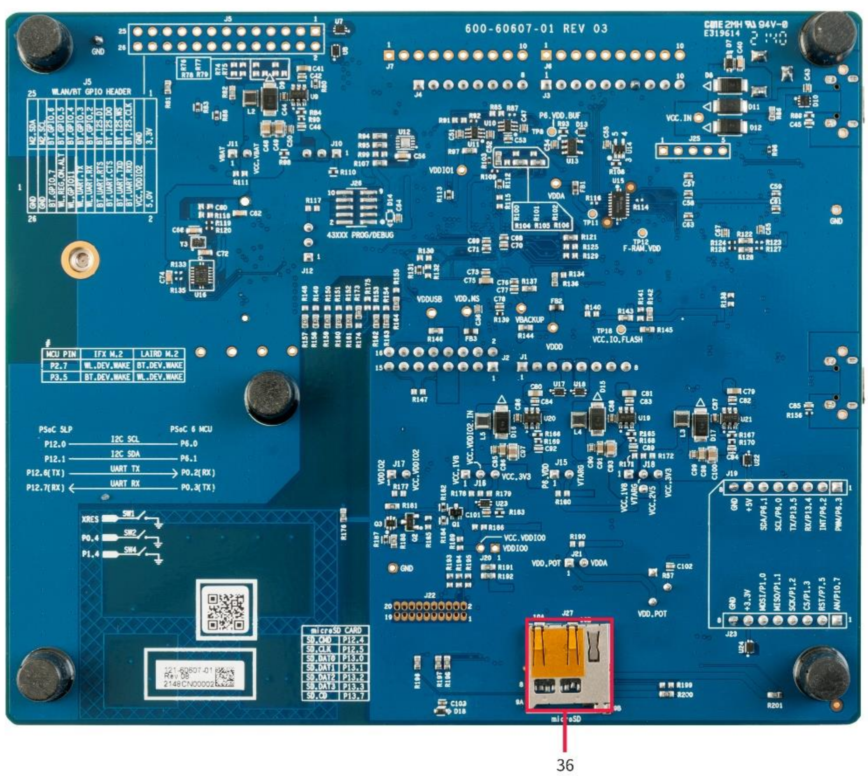


Figure 8 **PSoC™ 62S2 evaluation board – bottom view**

The PSoC™ 62S2 evaluation board has the following peripherals:

Table 4 Peripheral details

Sl. No.	Peripheral	Description
1.	KitProg3 USB connector (J9)	Connect to a PC to use the KitProg3 on-board programmer and debugger and to provide power to the board.
2.	KitProg3 status LED (D4)	Amber LED (D4) indicates the status of KitProg3. For details on the KitProg3 status, see the **KitProg3 user guide**.
3.	KitProg3 (PSoC™ 5LP) programmer and debugger (CY8C5868LTI-LP039, U2)	The PSoC™ 5LP device (CY8C5868LTI-LP039) serving as KitProg3, is a multi-functional system, which includes a SWD programmer, debugger, USB-I2C bridge and USB-UART bridge. For more details, see the **KitProg3 user guide**.
4.	PSoC™ 6 MCU	This kit is designed to highlight the features of the PSoC™ 6 MCU. For details on PSoC™ 6 MCU pin mapping, refer to **Table 1**.
5.	QSPI F-RAM (CY15B104QSN, U4)	4-Mbit nonvolatile memory that performs reads and writes similar to a RAM. It provides reliable data retention for 151 years while eliminating the complexities, overhead, and system-level reliability problems caused by serial flash and other nonvolatile memories.
6.	KitProg3 programming mode selection button (SW3)	Use this button to switch between various modes of operation of KitProg3. Note that this board supports only CMSIS-DAP BULK mode. For more details, see the **KitProg3 user guide**. This button function is reserved for future use.
7.	OPTIGA™ Trust M security controller (SLS32AIA, U5)	The OPTIGA™ Trust M is a high-end security solution that provides an anchor of trust for connecting IoT devices to the cloud, giving every IoT device its own unique identity.
8.	512-Mbit serial NOR flash memory (S25FL512S, U3)	Connected to the serial memory interface (SMIF) of the PSoC™ 6 MCU. The NOR flash device can be used for both data and code memory with execute-in-place (XIP) support and encryption.
9.	PSoC™ 6 MCU USB device connector (J14)	Use this USB connector to connect to a PC for using the PSoC™ 6 MCU USB device applications.
10.	Headers compatible with mikroBUS by Mikroelektronika (J19, J23)	Interfaces add-on boards compatible with mikroBUS by Mikroelektronika.
11.	PSoC™ 6 MCU user buttons (SW2 and SW4)	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. These buttons also provide a wake-up source from low-power modes of the device.

Sl. No.	Peripheral	Description
12.	System power (VTARG) selection jumper (J18)	Selects the PSoC™ 6 MCU VDD supply voltage between 1.8 V, 2.5 V, and 3.3 V.
13.	Potentiometer (R57)	10-kΩ potentiometer connected to PSoC™ 6 MCU pin P10[6]. It can be used to simulate a sensor output to PSoC™ 6 MCU.
14.	Potentiometer connection jumper (J21)	Connects the PSoC™ 6 MCU VDDA supply to the potentiometer.
15.	PSoC™ 6 MCU reset button (SW1)	Resets PSoC™ 6 MCU. It connects the PSoC™ 6 MCU reset (XRES) pin to ground.
16.	PSoC™ 6 MCU VDDIO0 current measurement jumper (J20)	Connect an ammeter to this jumper to measure the current consumed by the PSoC™ 6 MCU VDDIO0 power domain. Not loaded by default.
17.	PSoC™ 6 MCU 10-pin SWD/JTAG program and debug header (J24)	This 10-pin header allows you to program and debug the PSoC™ 6 MCU using an external programmer such as MiniProg4.
18.	PSoC™ 6 MCU debug and trace header (J22)	Connect to an Embedded Trace Macrocell (ETM)-compatible programmer/debugger. This is not loaded by default.
19.	Power LED (D6)	Amber LED that indicates the status of the power supplied to board.
20.	CAPSENSE™ slider (SLIDER) and buttons (BTN0 and BTN1)	The CAPSENSE™ touch-sensing slider and two buttons, all of which are capable of both self-capacitance (CSD) and mutual-capacitance (CSX) operation, allow you to evaluate Infineon’s fourth-generation CAPSENSE™ technology. The slider and buttons have a 1-mm acrylic overlay for smooth touch sensing.
21.	PSoC™ 6 MCU VDDIO2 current measurement jumper (J17)	Connect an ammeter to this jumper to measure the current consumed by the PSoC™ 6 MCU VDDIO2 power domain.
22.	PSoC™ 6 MCU VDDIO2 power selection jumper (J16)	VDDIO2 power supply of PSoC™ 6 MCU can be set to either 1.8 V or 3.3 V.
23.	PSoC™ 6 MCU VTRAG current measurement jumper (J15)	Connect an ammeter to this jumper to measure the current consumed by the PSoC™ 6 MCU VDD power domain.
24.	Power header compatible with Arduino Uno R3 (J1)	Powers the shields compatible with Arduino. It also has a provision to power the kit though the VIN input.
25.	I/O headers (J2, J3, and J4) compatible with Arduino Uno R3	Bring out pins from PSoC™ 6 MCU to interface with shields compatible with Arduino. Some of these pins are multiplexed with on-board 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**.
26.	M.2 interface connector (J13)	M.2 E-Key socket to interface compatible AIROC™ Wi-Fi & Bluetooth® combo M.2 radio modules like the Sterling-LWB5+ M.2 module from Laird Connectivity.

Sl. No.	Peripheral	Description
27.	Sterling-LWB5+ M.2 module from Laird Connectivity	M.2 radio module based AIROC™ CYW4373E with SDIO interface. There are two MHF4 connectors on the M.2 board, which can use **certified antennas** to support antenna diversity.
28.	Custom M.2 interface selection jumper (J12)	Supports Laird Connectivity custom M.2 interface pin mapping. By default, this jumper is set to POS 2-3 to support the Sterling- LWB5+ M.2 module from Laird Connectivity. For other M.2 modules compatible with Infineon's standard M.2 pin mappings (see section **3.2.2 M.2 Interface Connector**), this jumper must be set to POS 1-2.
29.	VBAT current measurement jumper (J11)	Connect an ammeter to this jumper to measure the current consumed by the VBAT power domain of the connected radio module.
30.	VBAT power selection jumper (J10)	Selects the VBAT supply for the connected radio module. VBAT can be set to either 1.8 V or 3.3 V or 3.6 V.
31.	PSoC™ 6 MCU user LEDs (D1 and D2)	The user LEDs can operate at the entire operating voltage range of the PSoC™ 6 MCU. The LEDs are active LOW, so the pins must be driven to ground to turn ON the LEDs.
32.	RGB LED (D3)	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.
33.	AIROC™ Wi-Fi & Bluetooth® combo GPIO header (J5)	Brings out a few I/Os of the AIROC™ Wi-Fi & Bluetooth® combo based M.2 module. This is not loaded by default.
34.	PSoC™ 6 MCU I/O headers (J6, J7)	Provide connectivity to PSoC™ 6 MCU GPI/Os. Some of these I/Os are also connected to on-board peripherals. See **Table 1** for pin mapping. This is not loaded by default.
35.	External power supply VIN connector (J8)	Connects an external DC power supply input to the on-board regulators.
36.	microSD Card holder (J27)	Provides SDHC interface with microSD cards with the option to detect the presence of the card.

See **3.2 Hardware functional description** for details on various hardware blocks.

2.2 M.2 wireless connectivity module

The PSoC™ 62S2 evaluation kit provides the option to connect AIROC™ Wi-Fi & Bluetooth® combo radio modules to the M.2 interface connector on the board. The combination of PSoC™ 6 host MCU and radio module enables you to evaluate Wi-Fi/Bluetooth® IoT applications. The M.2 radio module is not required if your application does not require Wi-Fi/Bluetooth® connectivity.

For some M.2 radio modules, it is required to connect an external antenna certified by the module vendor. The Sterling-LWB5+ M.2 module from Laird Connectivity provided with this kit requires an external antenna. Connect the FlexPIFA antenna from Laird Connectivity provided with this kit as follows:

1. Connect the FlexPIFA antenna from Laird Connectivity to the MHF4L connector, ANT0.

   Note: For the single antenna use case, ANT1 is not a valid option.

2. The FlexPIFA antenna from Laird Connectivity comes with a 3M adhesive. You can either stick the antenna to the base board as shown in **Figure 9** or leave the antenna hanging freely.

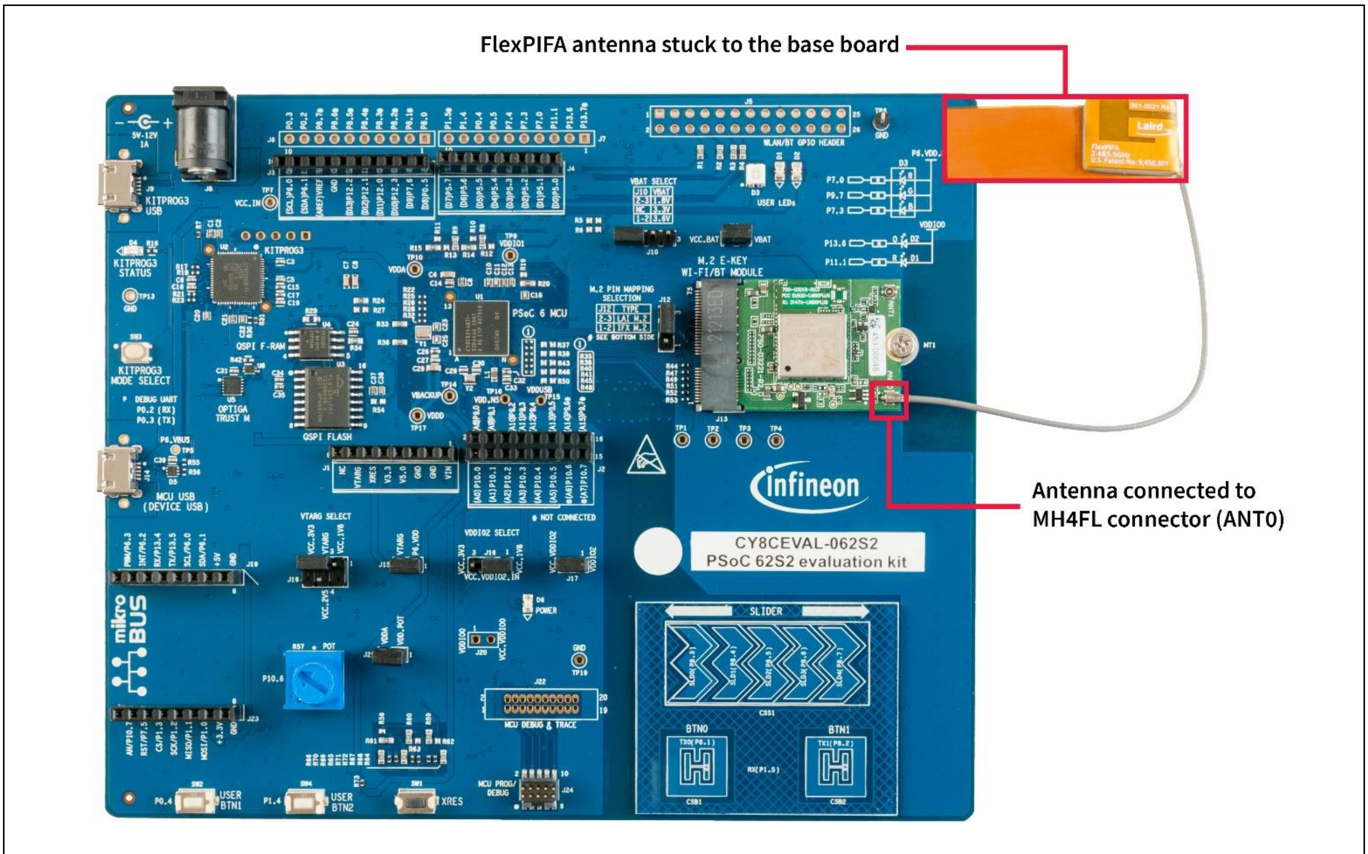


Figure 9 PSoC™ 62S2 evaluation board + Sterling-LWB5+ M.2 module from Laird Connectivity: Antenna connection

2.3 BSP selection

The PSoC™ 62S2 evaluation kit comes with two board support packages (BSP). Choose the BSP based on the type of application you need to run on this board.

Table 5 BSP selection

BSP	Application type
CY8CEVAL-062S2	PSoC™ 6 MCU-based non-connectivity examples
CY8CEVAL-062S2-LAI-4373M2	Connectivity (Wi-Fi or Bluetooth®) examples based on PSoC™ 6 MCU and the Sterling-LWB5+ M.2 module from Laird Connectivity. *Note: PSoC™ 6 MCU-based non-connectivity examples are also supported on this BSP.*

This board can support other M.2 based connectivity modules which use an SDIO interface like **Embedded Artists 1DX**, **Embedded Artists 1LV** M.2 modules, etc. To run applications on this board with other M.2-based connectivity modules, create your own BSP for the appropriate combination of base board and connectivity module by following the instructions given in the **Board Support Packages** section of the **ModusToolbox™ user guide**.

Note: BSPs with support for additional M.2 based connectivity modules will be added in the future.

2.4 KitProg3: on-board programmer/debugger

The PSoC™ 62S2 evaluation board can be programmed and debugged using the on-board KitProg3. KitProg3 is an on-board programmer/debugger with USB-UART, USB-I2C, and USB-SPI Bridge (not supported on this board) functionality. KitProg3 supports CMSIS-DAP only and does not support mass storage. A PSoC™ 5LP device is used to implement the KitProg3 functionality. For more details on the KitProg3 functionality, see the **KitProg3 user guide**.

2.4.1 Programming and debugging using ModusToolbox™ software

1. Connect the board to the PC using the provided USB cable through the KitProg3 USB connector, as shown in **Figure 10**. It enumerates as a USB Composite Device if you are connecting it to your PC for the first time.
2. KitProg3 on this kit supports CMSIS-DAP Bulk mode (default) and CMSIS-DAP Bulk with two UARTs. The status LED (amber) is always ON in the CMSIS-DAP Bulk mode. If you do not see the desired LED status, see the **KitProg3 user guide** for details on the KitProg3 status and troubleshooting instructions.

Note: The programming can be done in either of the KitProg3 programming modes but it is recommended to program the kit in CMSIS-DAP Bulk mode.

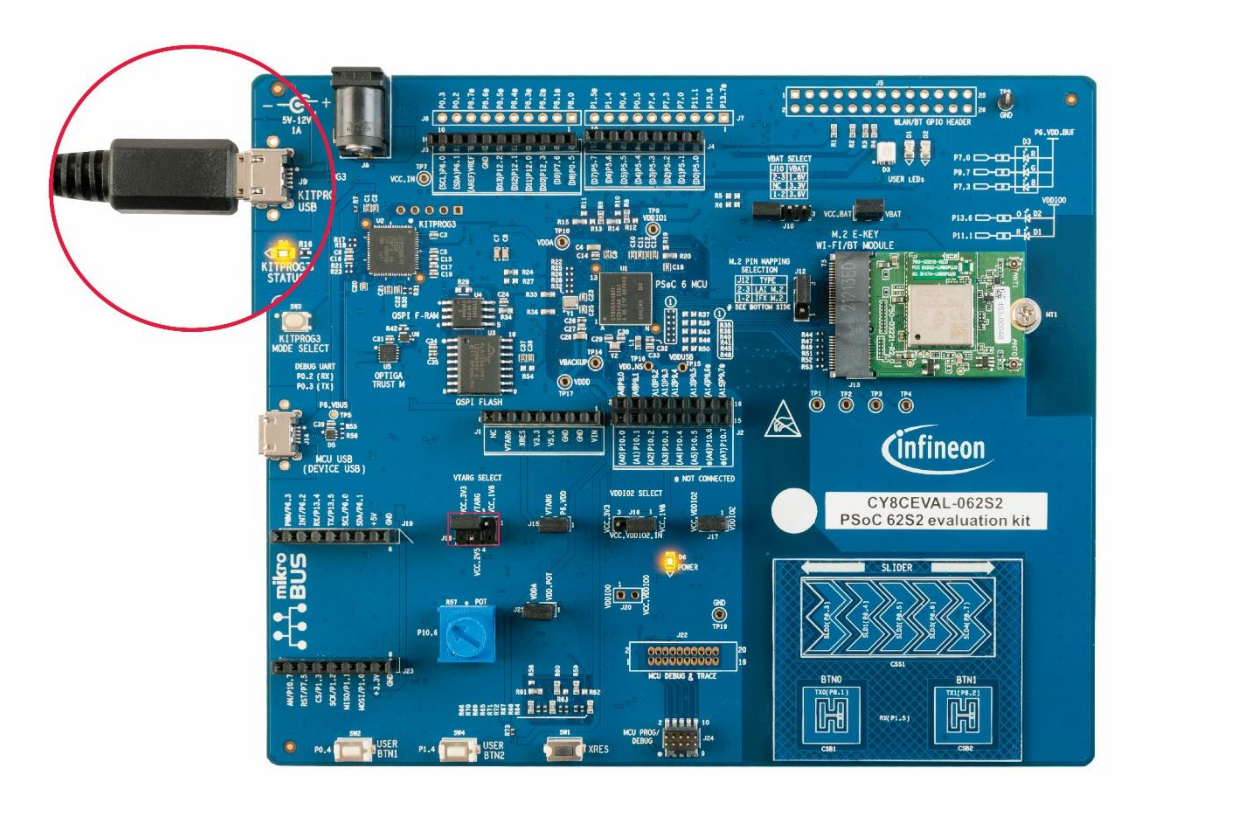


Figure 10 Connect USB cable to USB connector on the board

3. In the Eclipse IDE for ModusToolbox™ software, import the desired code example (application) into a new workspace.

a) Click on **New Application** from **Quick Panel**.

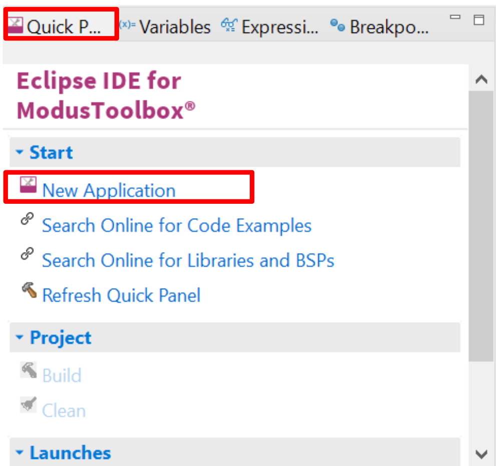


Figure 11 Create new application

b) Select the BSP in the "Choose Board Support Package" window and click **Next**.
As noted in section **2.3 BSP selection**, the BSP selection should be based on the combination of baseboard and radio module used. The rest of the steps assumes there is no radio module connected to the baseboard and uses the CY8CEVAL-062S2 BSP for the sake of explanation.

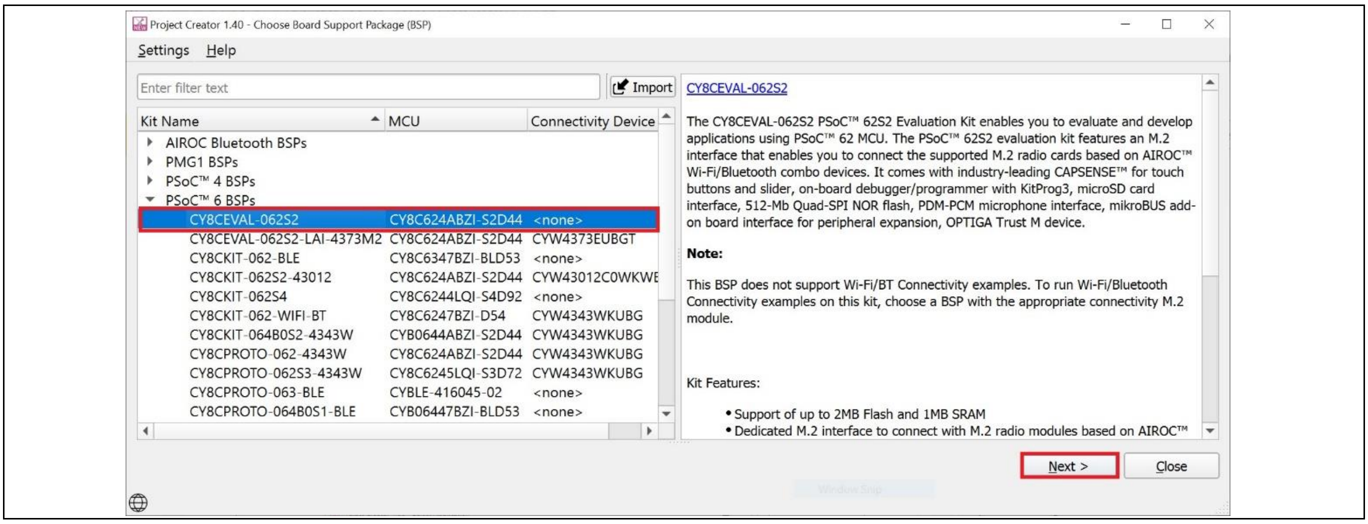


Figure 12 Creating a new application: Choose Board Support Package

c) Select the application in the "Select Application" window and click **Create**.

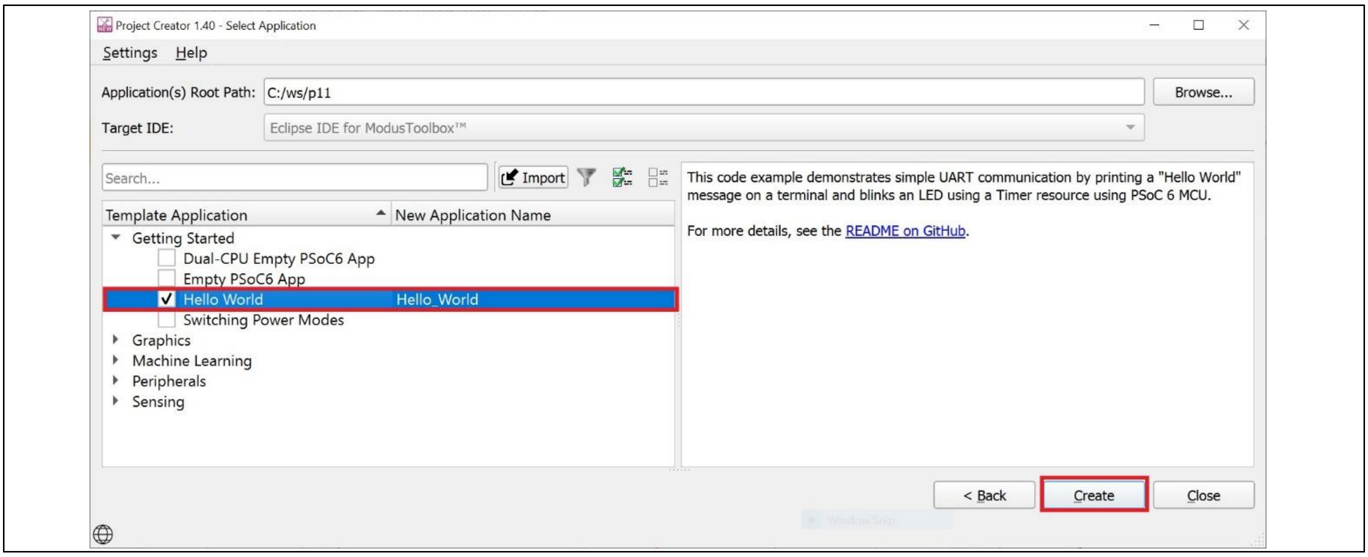


Figure 13 Creating a new application: Select Application

Kit operation

4. To build and program a PSoC™ 6 MCU application, in the Project Explorer, select **<App_Name>** project. In the Quick Panel, scroll to the Launches section and click the **<App_Name> Program (KitProg3_MiniProg4)** configuration as shown in **Figure 14**.

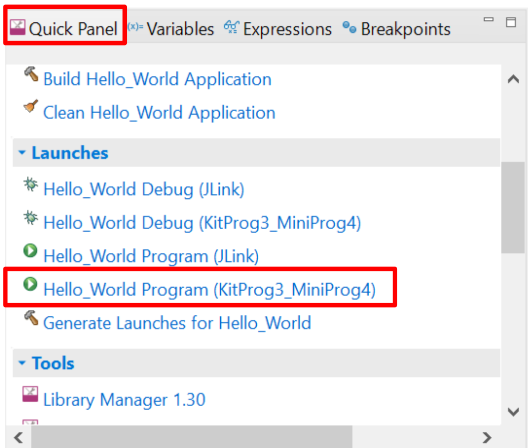


Figure 14 Programming in ModusToolbox™ software

5. ModusToolbox™ software has an integrated debugger. To debug a PSoC™ 6 MCU application, in the Project Explorer, select **<App_Name>** project. In the Quick Panel, scroll to the **Launches** section and click the **<App_Name> Debug (KitProg3_MiniProg4)** configuration as shown in **Figure 15**. For more details, see the "Program and debug" section in the **Eclipse IDE for ModusToolbox™ user guide**.

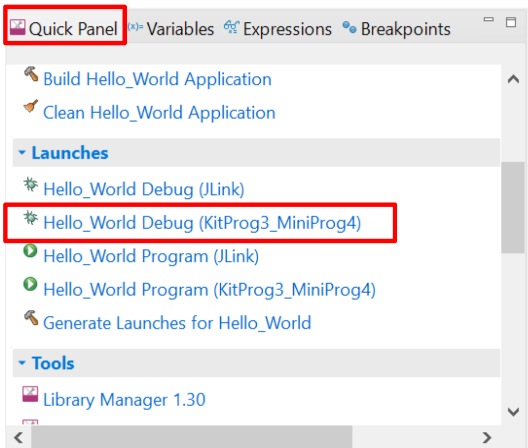


Figure 15 Debugging in ModusToolbox™ software

2.4.1.1 Using the OOB example – PSoC™ 6 MCU: Hello World

The PSoC™ 62S2 evaluation board is by default programmed with the code example: *PSoC™ 6 MCU: Hello World*. The steps below describe on how to use the example. For a detailed description of the project refer to the example's README.md file in the GitHub repository. The README.md file is also in the application directory once the application is created.

Note: At any point of time, if you overwrite the OOB example, you can restore it by programming the PSoC™ 6 MCU: Hello World.

1. Connect the board to your PC using the provided USB cable through the KitProg3 USB connector.
2. Open a terminal program and select the KitProg3 COM port. Set the serial port parameters to 8N1 and 115200 baud. The image below shows COM7 but your port will likely be different.
3. Press the reset button (SW1) on the board and confirm that terminal application displays code example title and other text **Figure 16**.

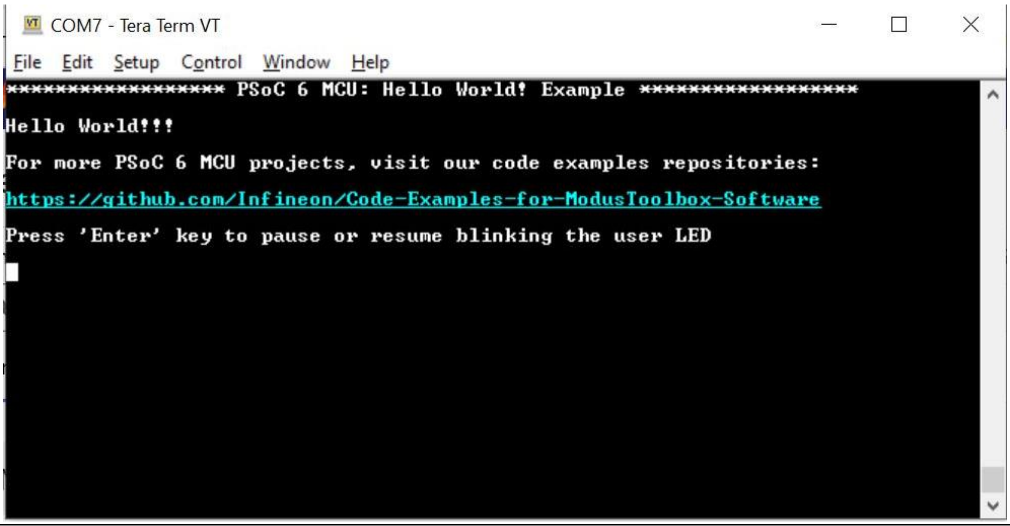


Figure 16 "Hello World" message in the terminal

4. Confirm that the kit LED blinks at 1 Hz.
5. Press the **Enter** key. Confirm that the kit LED stops blinking. The terminal displays the message "LED blinking paused".
6. Press the **Enter** key again. Confirm that the kit LED resumes blinking at 1 Hz. The message displayed on the terminal is updated to "LED blinking resumed".

You can repeat steps 5 and 6 indefinitely.

2.4.2 USB-UART bridge

KitProg3 on the PSoC™ 62S2 evaluation board can act as a USB-UART bridge.

The UART Rx and Tx pins of KitProg3 are connected to the PSoC™ 6 MCU UART pins, as follows:

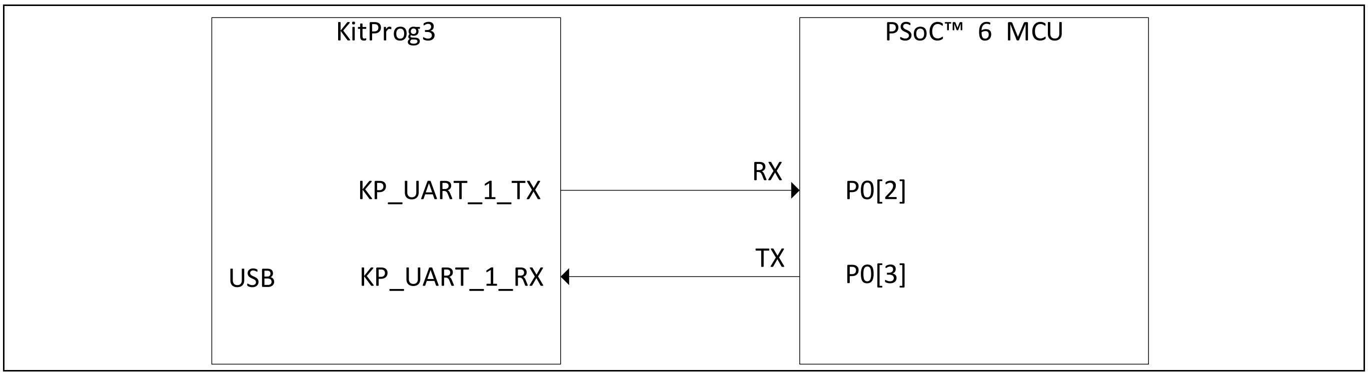


Figure 17 UART connection between KitProg3 and PSoC™ 6 MCU

Figure 18 shows the secondary UART connection between the AIROC™ Wi-Fi & Bluetooth® combo-based M.2 module and KitProg3. These wires are not connected on the board by default. See section **3.3.7 Secondary UART interface between PSoC™ 5LP and M.2 module** to see how to re-work the board to connect KitProg3 USB-UART bridge with AIROC™ Wi-Fi & Bluetooth® combo-based M.2 module.

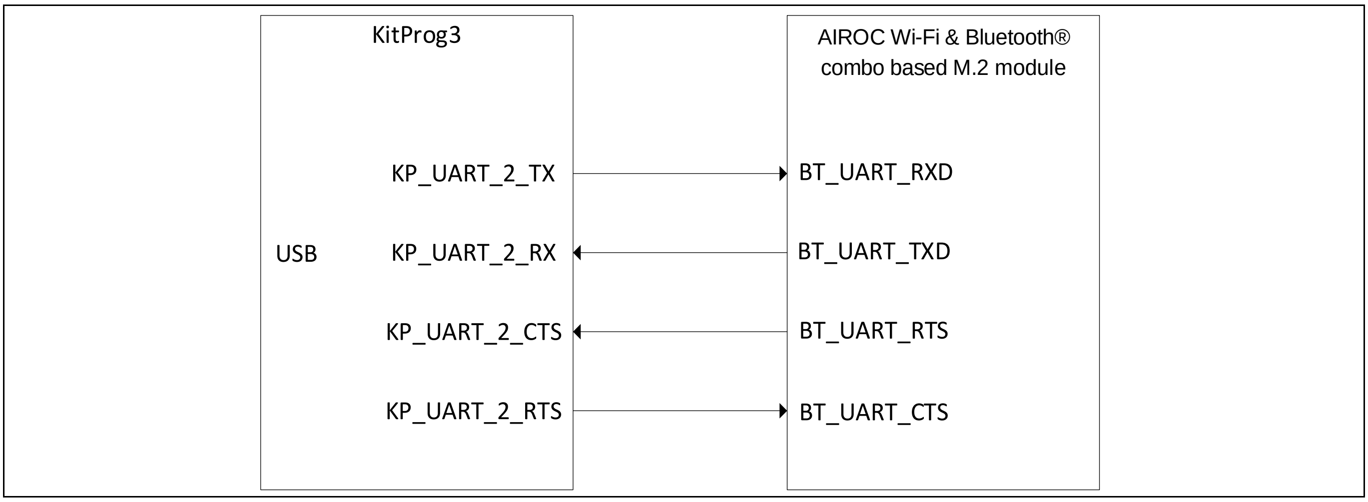


Figure 18 UART connection between KitProg3 and AIROC™ Wi-Fi & Bluetooth® combo-based M.2 module

For more details on the KitProg3 USB-UART functionality, see the **KitProg3 user guide**.

2.4.3 USB-I2C bridge

The KitProg3 can function as a USB-I2C bridge and communicate with an I2C master such as Bridge control panel (BCP). The I2C lines on the PSoC™ 6 MCU are hard-wired on the board to the I2C lines of the KitProg3, with on-board pull-up resistors. The USB-I2C supports I2C speeds of 50 kHz, 100 kHz, 400 kHz, and 1 MHz. For more details on the KitProg3 USB-I2C functionality, see the **KitProg3 user guide**.

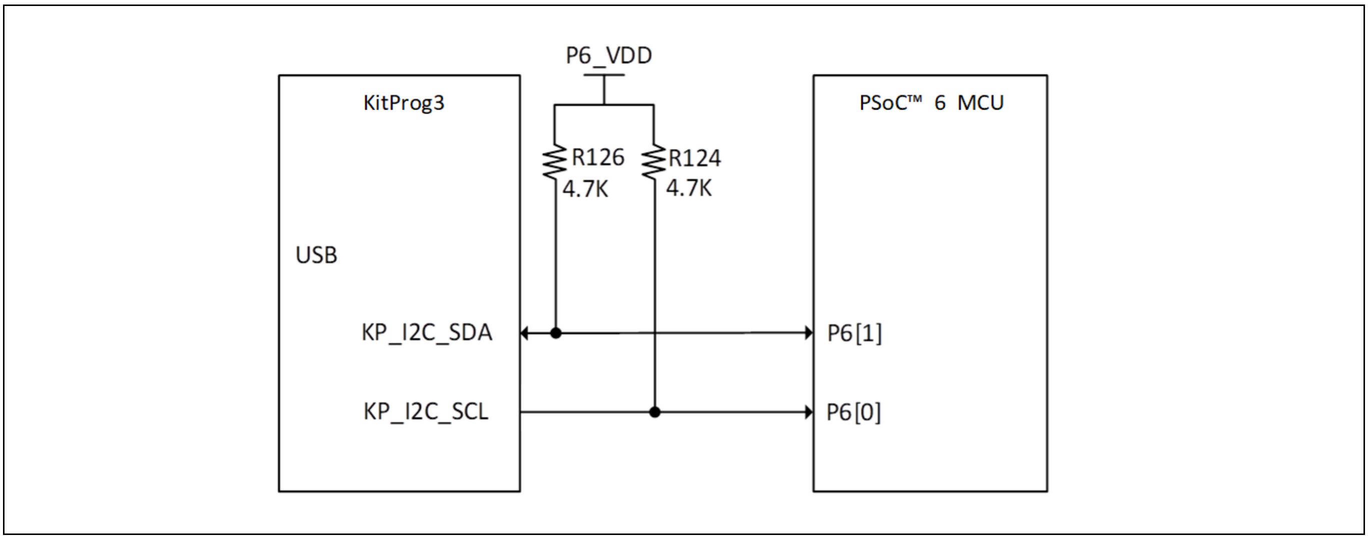


Figure 19 I2C connection between KitProg3 and PSoC™ 6 MCU

3 Hardware

3.1 Schematics

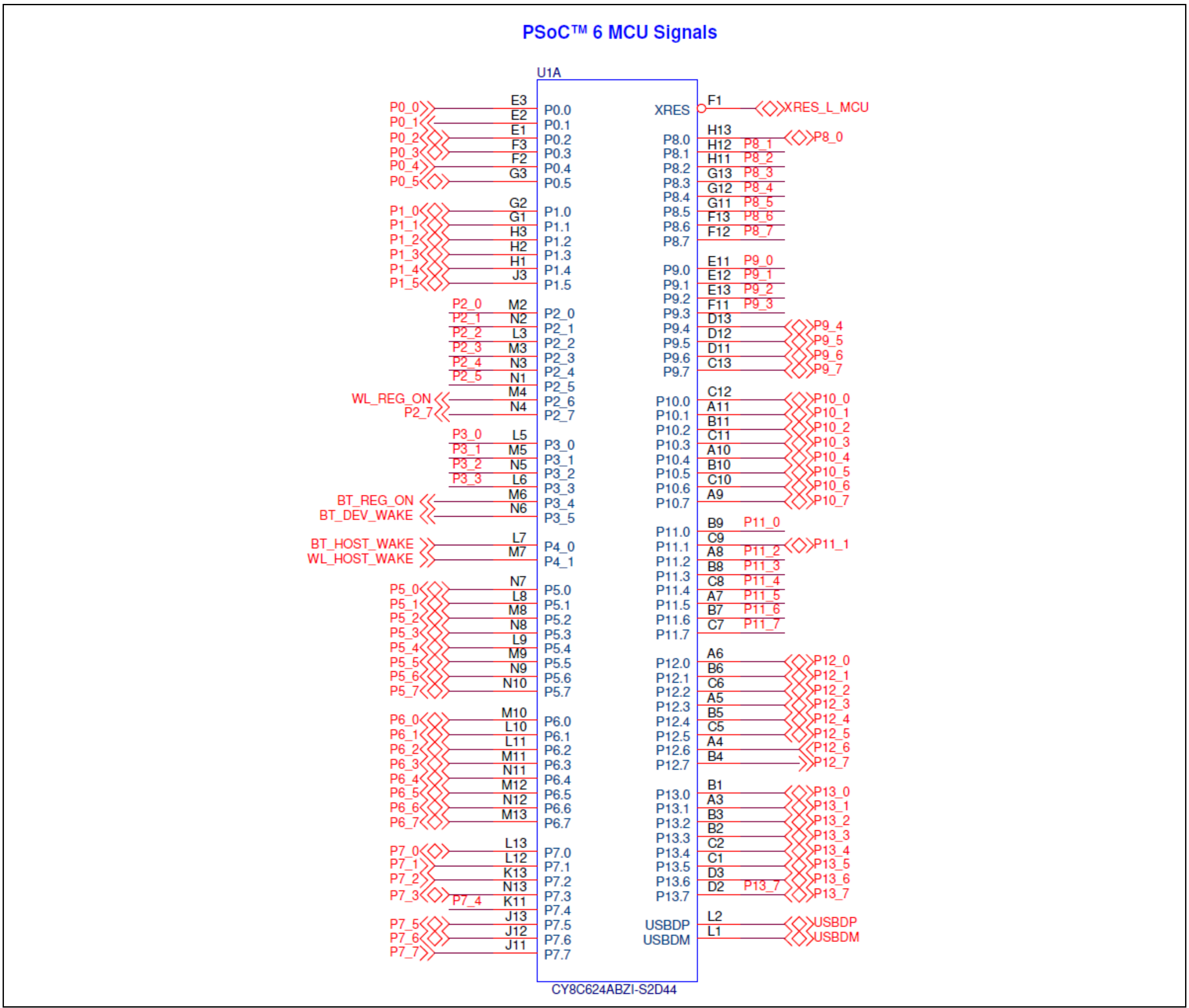
See the schematic files available in the **kit webpage**.

3.2 Hardware functional description

3.2.1 PSoC™ 6 MCU (U1)

PSoC™ 6 MCU is a high-performance, ultra-low-power and secured MCU platform, purpose-built for IoT applications. This device used on this board belongs to the CY8C62x8/A product line, based on the PSoC™ 6 MCU platform. It is a combination of a dual-CPU microcontroller with low-power flash technology, digital programmable logic, high-performance analog-to-digital conversion and standard communication and timing peripherals.

For more information, see the **PSoC™ 6 MCU datasheet**.



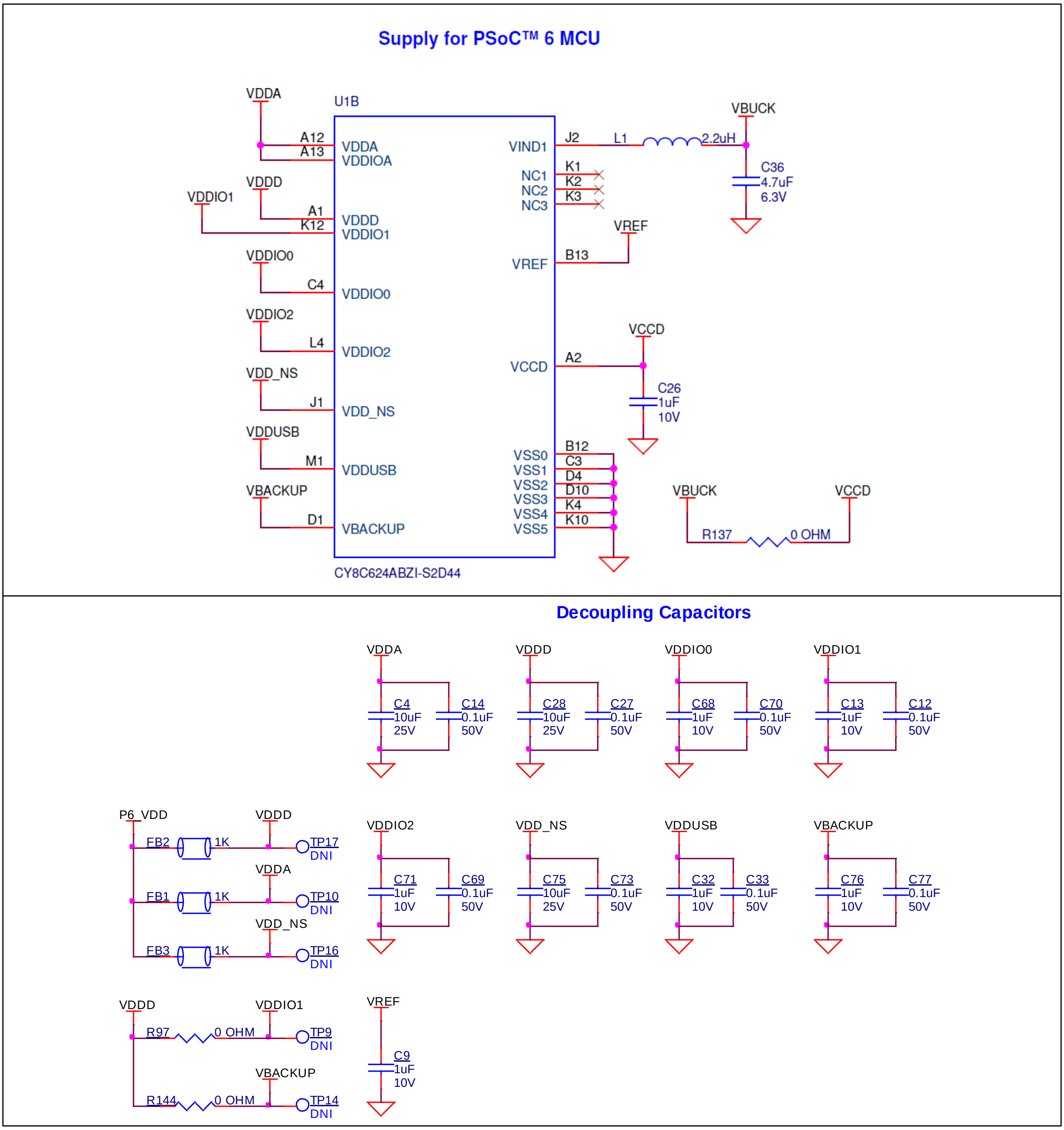


Figure 20 PSoC™ 6 MCU

3.2.2 M.2 interface connector (J13)

The M.2 E-key connector adds radio connectivity to PSoC™ 6 MCU as required. Any M.2 E-key radio module, compatible with Infineon M.2 connector pin mapping, can be interfaced to the PSoC™ 62S2 evaluation kit. The WLAN interface to the PSoC™ 6 MCU is SDIO, with an option to rework to SPI if required. The Bluetooth® interface to PSoC™ 6 MCU is UART. The VBAT supply for the module is provided from the PSoC™ 62S2 evaluation kit, which is 1.8 V/3.3 V/3.6 V selectable depending on the M.2 radio module being used. Refer to the appropriate M.2 radio module datasheet for valid operating voltage.

Table 6 Infineon M.2 connector pin mapping

Group	Signal name	M.2 connector pin
WLAN interface	SDIO_CLK	9
	SDIO_CMD	11
	SDIO_DATA0	13
	SDIO_DATA1	15
	SDIO_DATA2	17
	SDIO_DATA3	19
	WL_HOST_WAKE	21
	WL_REG_ON	56
	WL_DEV_WAKE	66
Bluetooth® interface	BT_UART_TXD	22
	BT_UART_RXD	32
	BT_UART_RTS	34
	BT_UART_CTS	36
	BT_HOST_WAKE	20
	BT_DEV_WAKE	42
	BT_REG_ON	54
Clock	LPO_IN	50
Power	VBAT	2, 4, 72, 74
	WL_VDDIO *Not connected by default	64

*Note: Some modules from Laird Connectivity like Sterling-LWB5+ does not follow the Infineon M.2 interface standard. For such modules, use the jumper settings mentioned in **Table 7**.*

Table 7 Pin mapping for Sterling LWB5+ M.2 module from Laird Connectivity

Signal name	Infineon M.2 connector pin (J12 POS 1-2)	Laird M.2 connector pin (J12 POS 2-3)
WL_DEV_WAKE	66	42
BT_DEV_WAKE	42	40

Hardware

There are additional optional signals which may not be present in all modules. See schematics for more details.

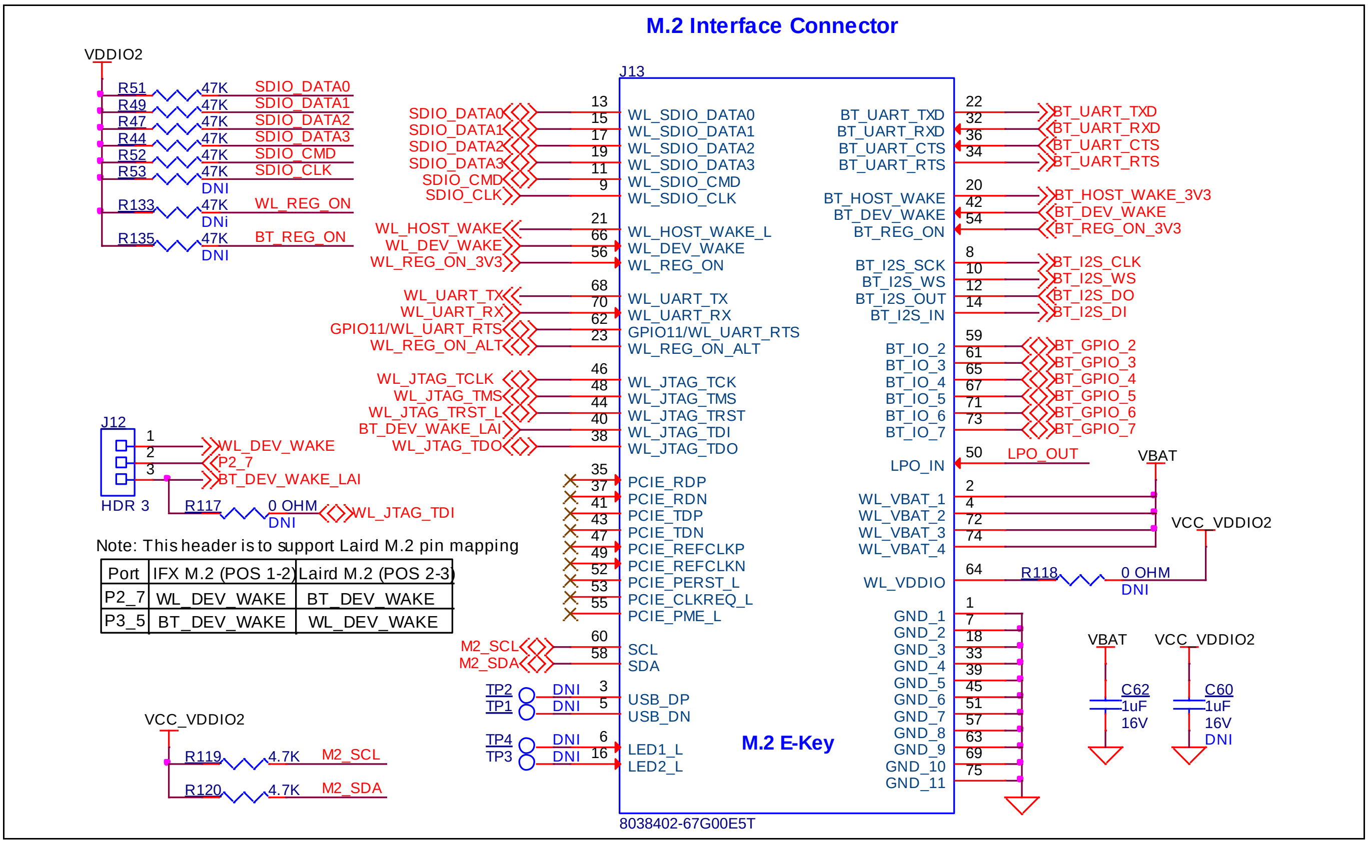


Figure 21 M.2 interface connector

3.2.3 PSoC™ 5LP-based KitProg3 (U2)

An on-board PSoC™ 5LP (CY8C5868LTI-LP039) is used as KitProg3 to program and debug PSoC™ 6 MCU. The PSoC™ 5LP device connects to the USB port of a PC through a USB connector and to the SWD and other communication interfaces of PSoC™ 6 MCU.

Visit the **PSoC™ 5LP web page** and see the **CY8C58LPxx family datasheet for more information**.

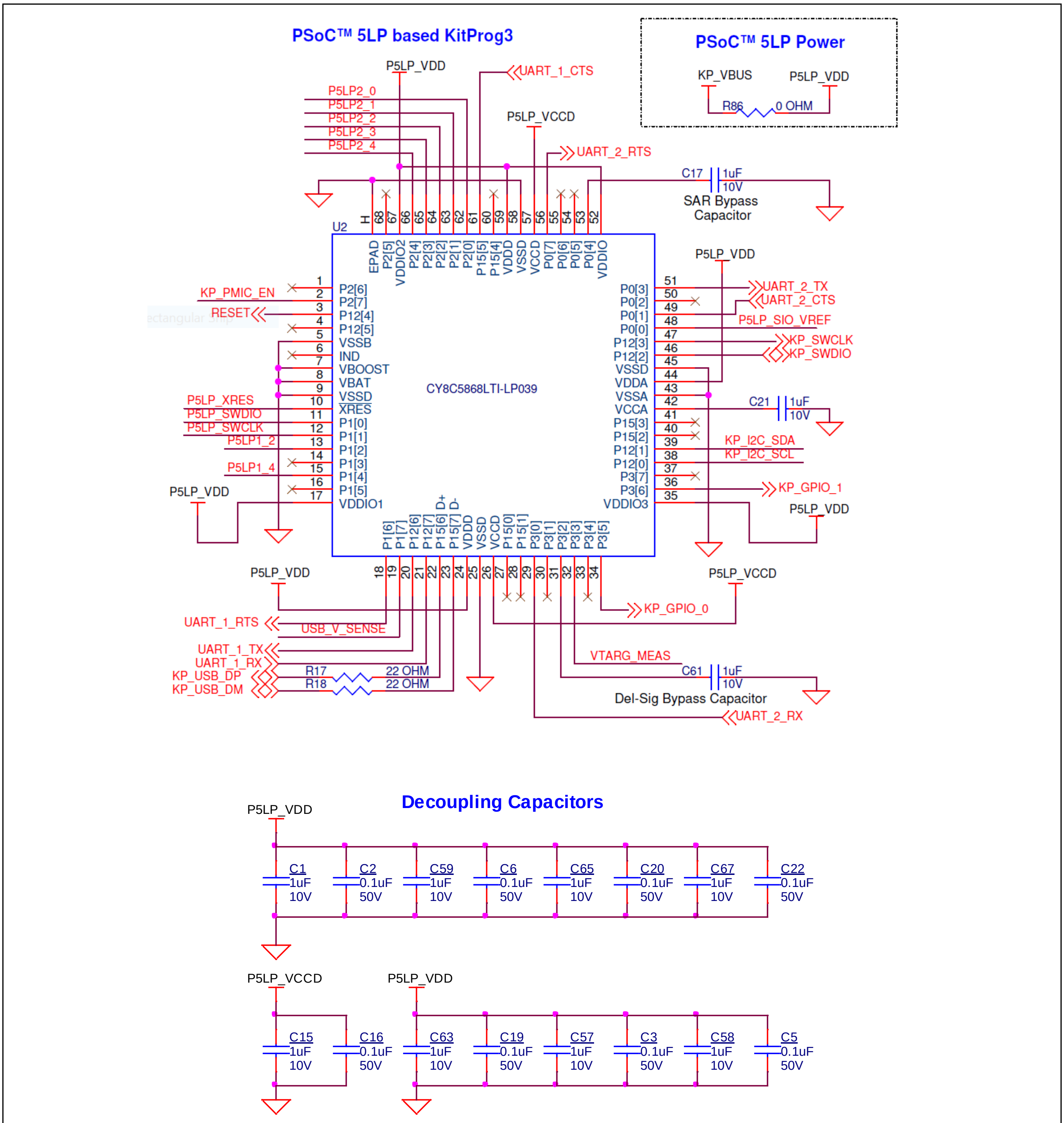


Figure 22 PSoC™ 5LP-based KitProg3 (U2)

3.2.4 Serial interconnection between PSoC™ 5LP and PSoC™ 6 MCU

In addition of use as an on-board programmer, the PSoC™ 5LP device functions as an interface for USB-UART and USB-I2C bridges, as shown in **Figure 23**. The USB-Serial pins of the PSoC™ 5LP device are hard-wired to the I2C/UART pins of the PSoC™ 6 MCU device. These pins are also available on the I/O headers compatible with Arduino Uno R3.

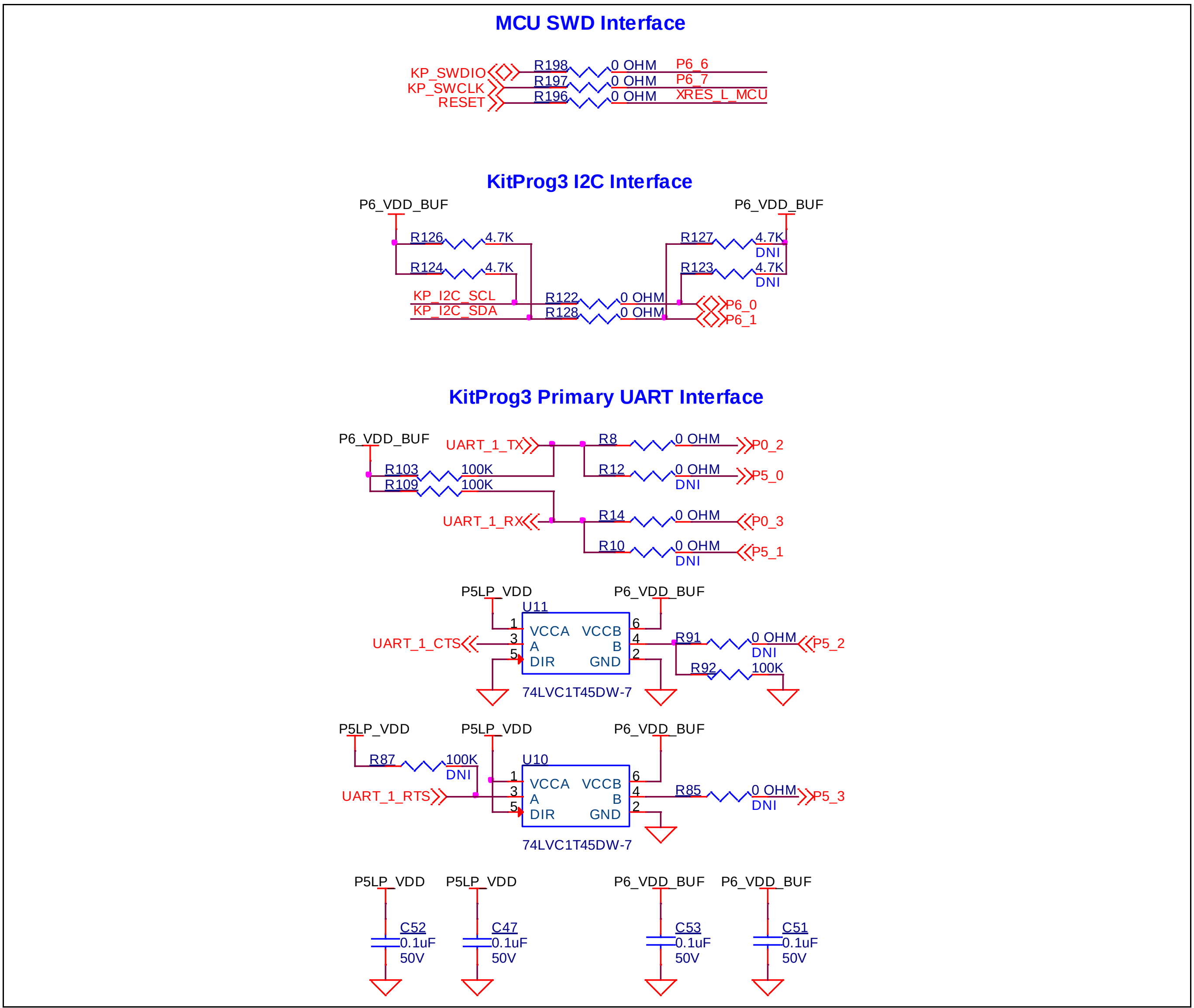


Figure 23 Programming and serial interface connections

3.2.5 Serial interconnection between PSoC™ 5LP and M.2 interface

The PSoC™ 5LP device also has a secondary UART that is connected to BT_UART of M.2 interface.

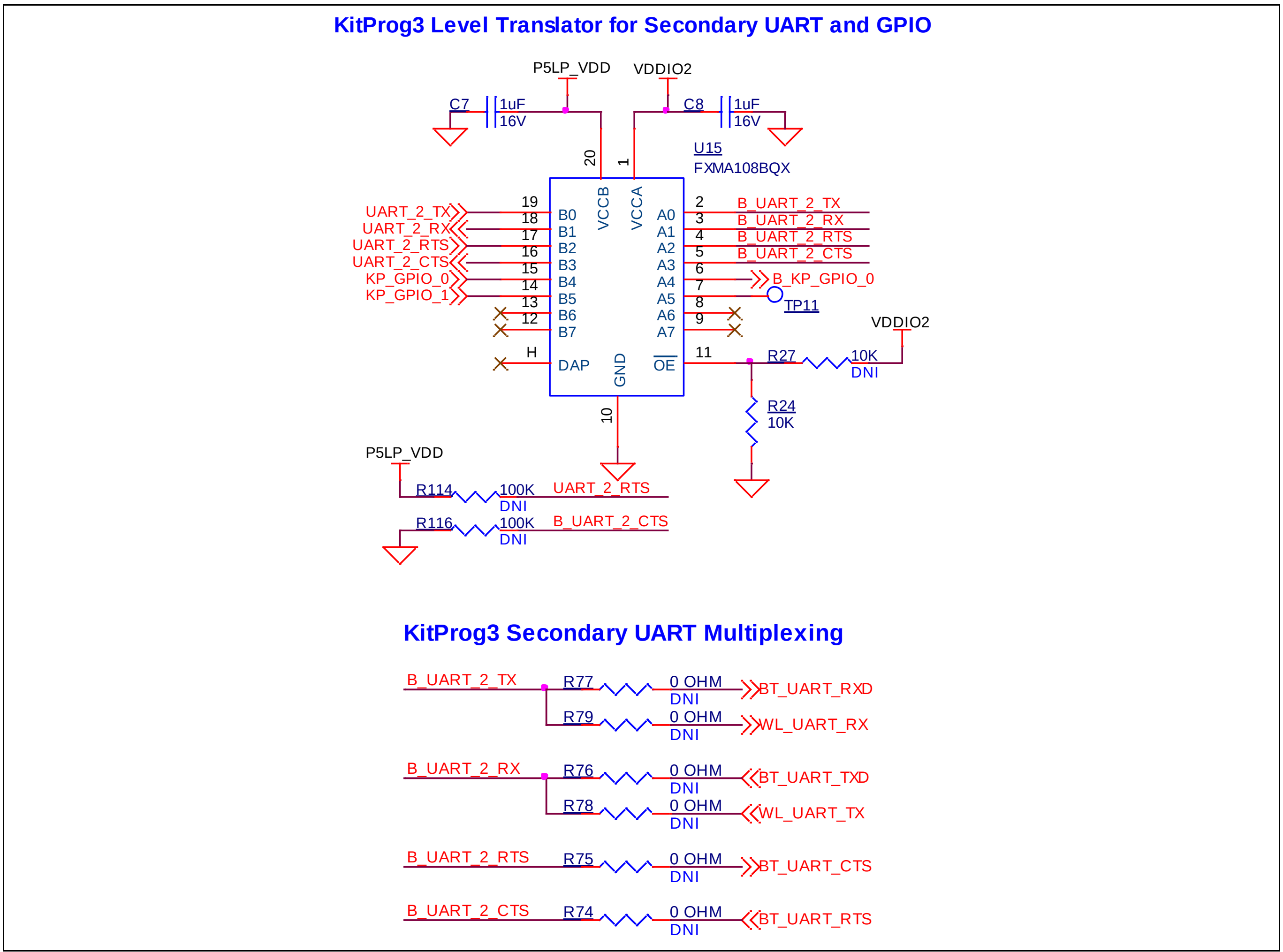


Figure 24 Serial interconnection between PSoC™ 5LP and M.2 interface

3.2.6 Power supply system

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

- 5 V from the on-board USB Micro-B connectors **(J9** and **J14)**
- 5 V–12 V from external power supply through the VIN barrel jack (**J8**) or from a shield compatible with Arduino

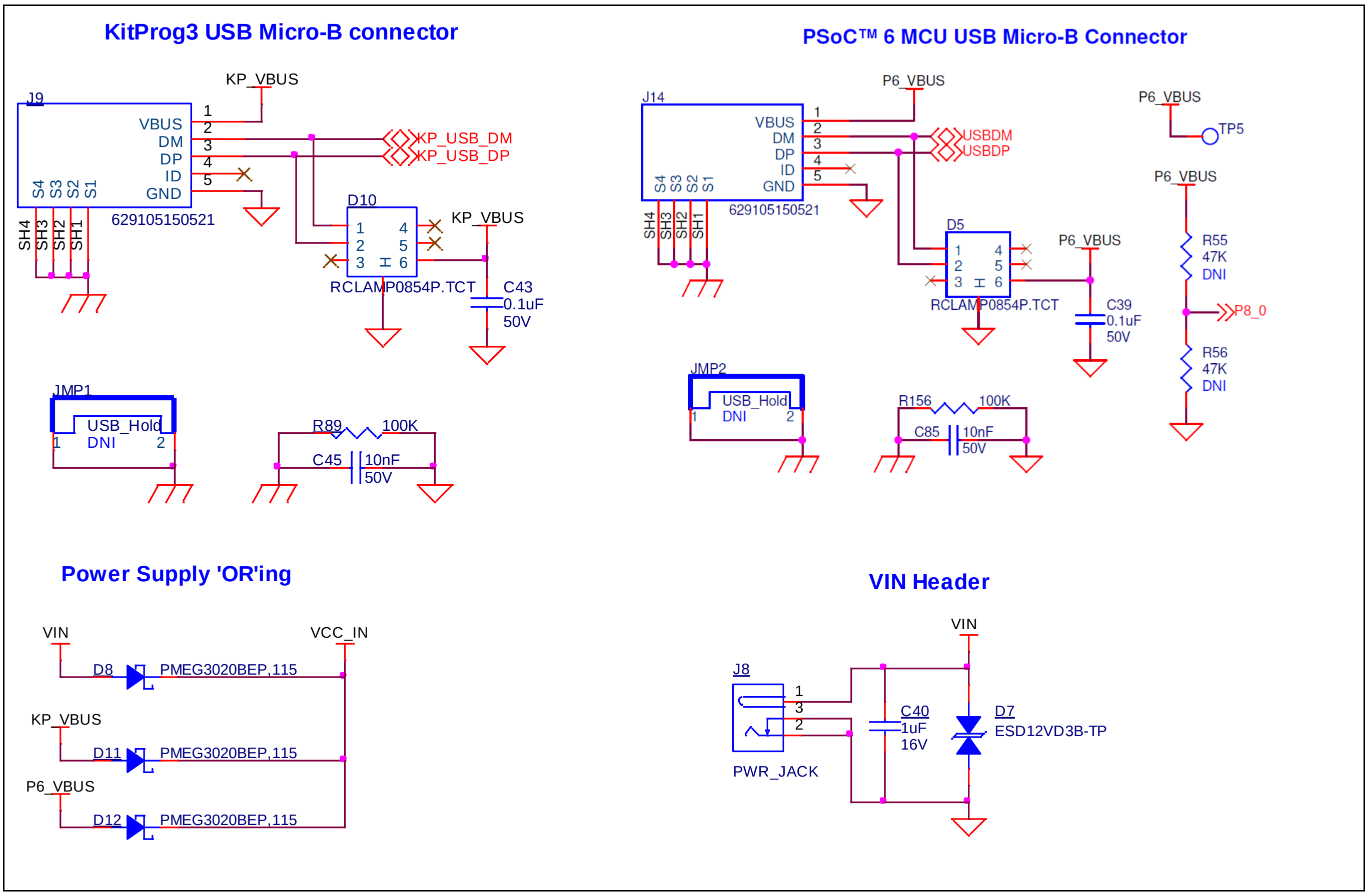


Figure 25 **Power supply input and ORing**

3.2.6.1 Voltage regulators

The power supply system is designed for the following voltage configurations:

VBAT (M.2 Interface) – 3.6 V, 3.3 V and 1.8 V

VDDIO2 (PSoC™ 6 MCU) – 1.8 V and 3.3 V

VDDD, VDDIO1, VDDA, VDD_NS, VBACKUP (PSoC™ 6 MCU) – 1.8 V, 2.5 V and 3.3 V

Some configurations may not be possible by changing jumper positions but rather requires rework of the respective 0-ohm resistors.

VCC_VDDIO2 must be at the same voltage as the M.2 radio module I/O voltage because it powers the SDIO and UART interfaces between PSoC™ 6 MCU and the radio. If VCC_VDDIO2 voltage is not the same as the M.2 radio I/O voltage, it can cause improper function or may even damage the radio module.

There are four buck regulators **U20, U19, U21**, and **U9** that are used to generate 1.8 V, 2.5 V, 3.3 V and 3.6 V (1.8 V/2.5 V/3.3 V/3.6 V) outputs respectively.

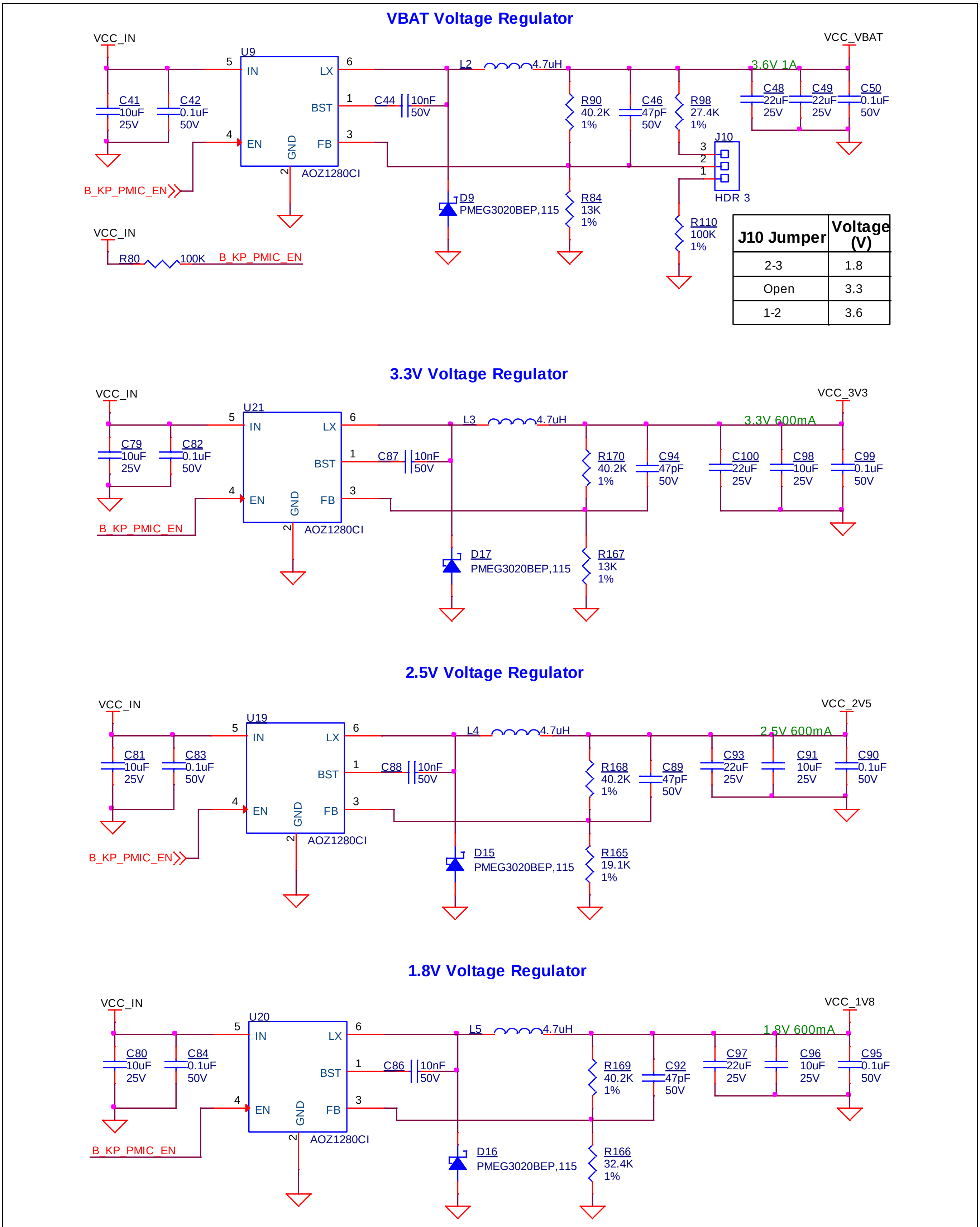


Figure 26 **Voltage regulators**

3.2.6.2 Voltage selection

VBAT has a dedicated regulator that changes voltage by varying the feedback voltage through the resistor network at **J10**. VTARG has a 4-pin voltage selection header **J18** to select between VCC_3V3, VCC_2V5 or VCC_1V8 voltages. VCC_VDDIO2_IN has a dedicated 3-pin voltage selection header **J16** that select between VCC_3V3 or VCC_1V8 voltages.

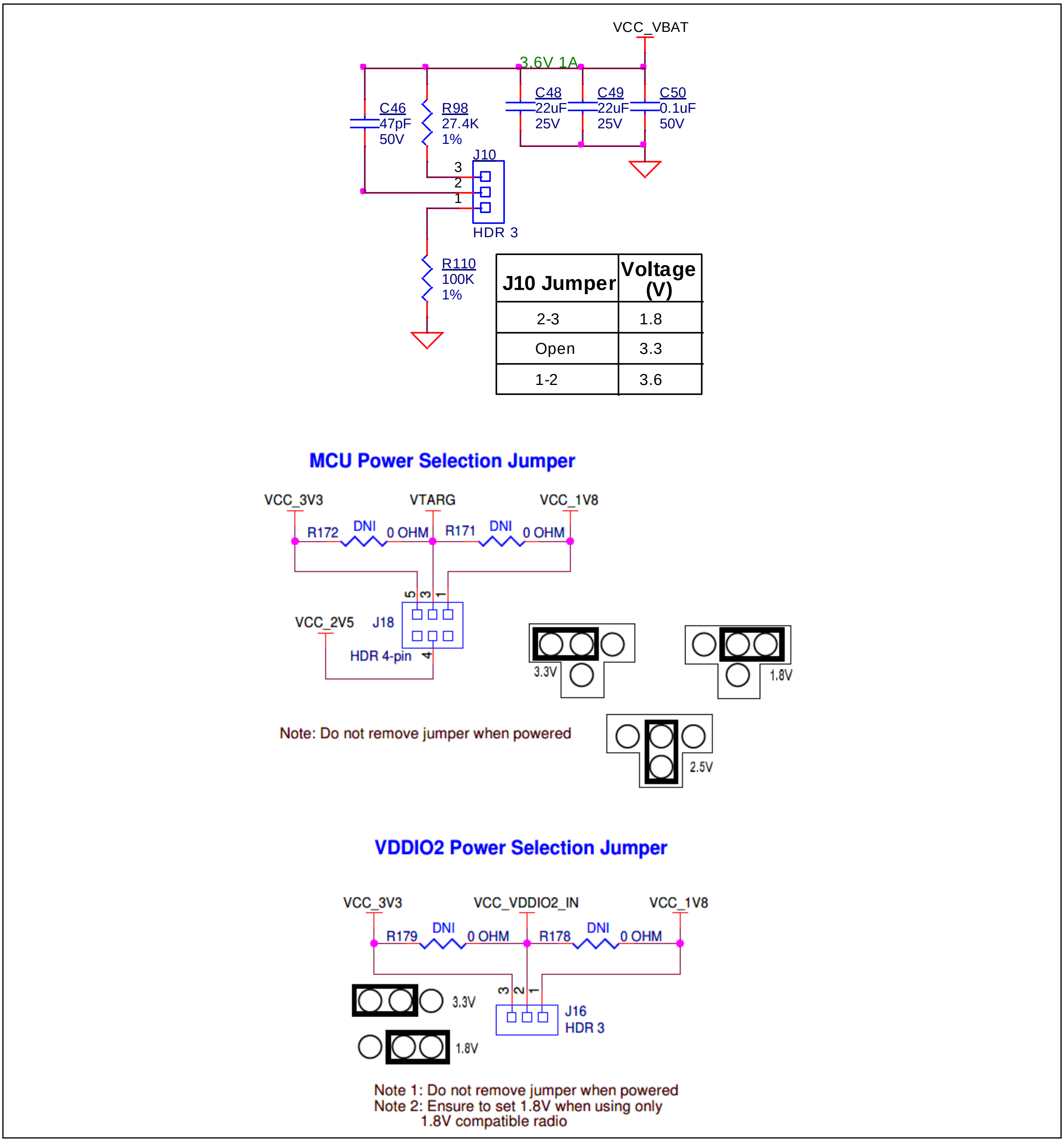


Figure 27 Voltage selection

The VCC_VDDIO0 voltage can also be selected between VTARG, VCC_3V3 and VCC_1V8 but using 0-Ω resistors only. It is connected to VTARG by default.

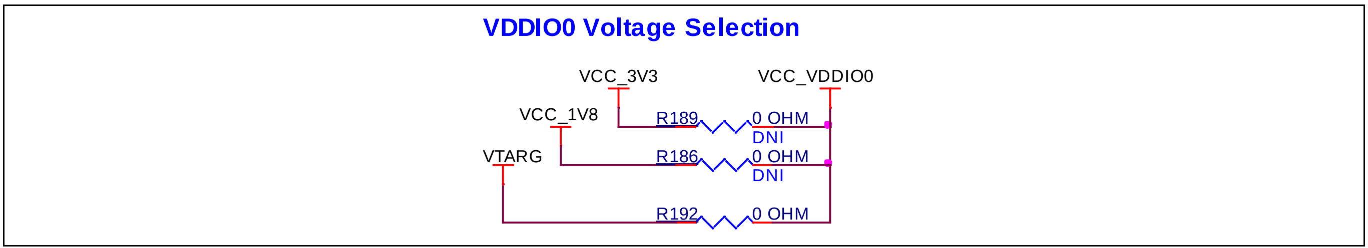


Figure 28 Voltage selection

3.2.6.3 Current measurement headers

The current of the following domains have dedicated 2-pin headers to facilitate easy current measurement using an ammeter across the pins.

Table 8 Current measurement headers

Domain name	Header reference designator	Loaded by default
VBAT	J11	Y
P6_VDD	J15	Y
VDDIO2	J17	Y
VDDIO0	J20	N

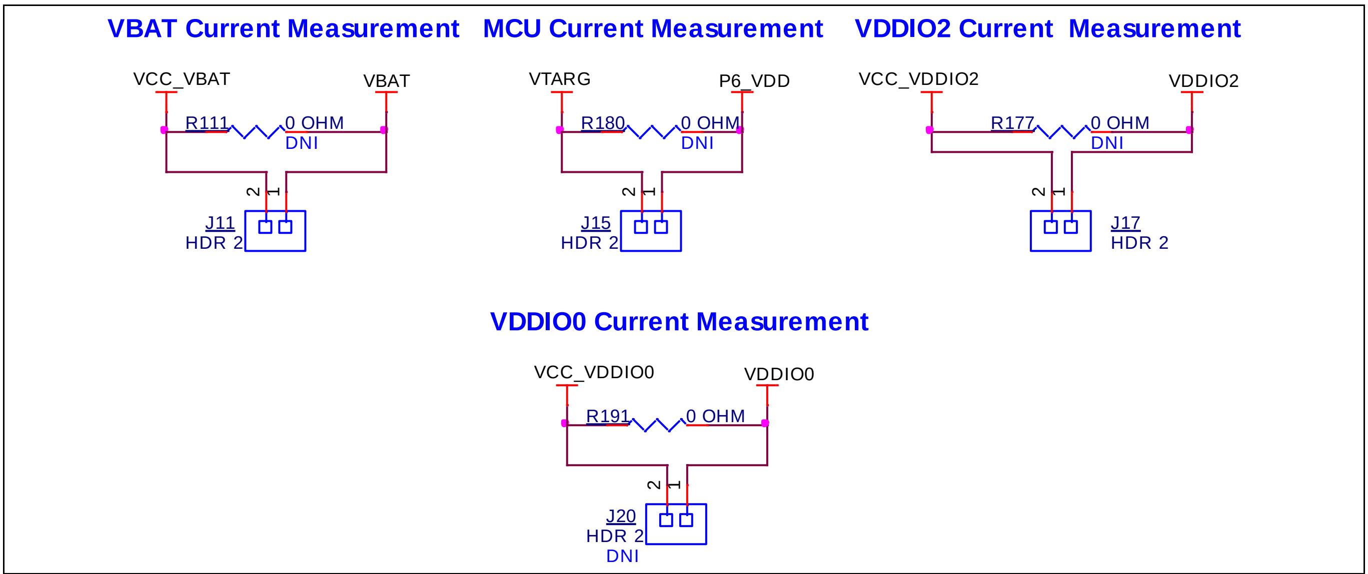


Figure 29 Current measurement headers

Note: When measuring the P6_VDD current, make sure that jumper J21 is removed. This will disconnect the potentiometer from VDDA and remove the leakage caused by it.

3.2.7 I/O headers

3.2.7.1 Headers compatible with Arduino Uno R3 (J1, J2, J3, J4)

The board has four headers compatible with Arduino Uno R3: J1, J2, J3, and J4. You can connect 1.8 V or 3.3 V shields compatible with Arduino Uno R3 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, the maximum voltage supported by PSoC™ 6 MCU is 3.6 V.

3.2.7.2 PSoC™ 6 MCU I/O headers (J6 and J7)

These headers provide connectivity to PSoC™ 6 MCU GPI/Os that are not connected to the headers compatible with Arduino. A majority of these pins are multiplexed with on-board peripherals and are not connected to the PSoC™ 6 MCU by default. These connectors are not populated by default.

3.2.7.3 Headers compatible with mikroBUS by Mikroelektronika (J19 and J23)

The board has two headers compatible with mikroBUS by Mikroelektronika (J19 and J23) to support different add-on boards compatible with mikroBUS by Mikroelektronika. You can connect 3.3 V add-on boards to develop applications. PSoC™ 6 MCU supports a maximum voltage of 3.6 V, connecting a 5 V add-on board may permanently damage the board.

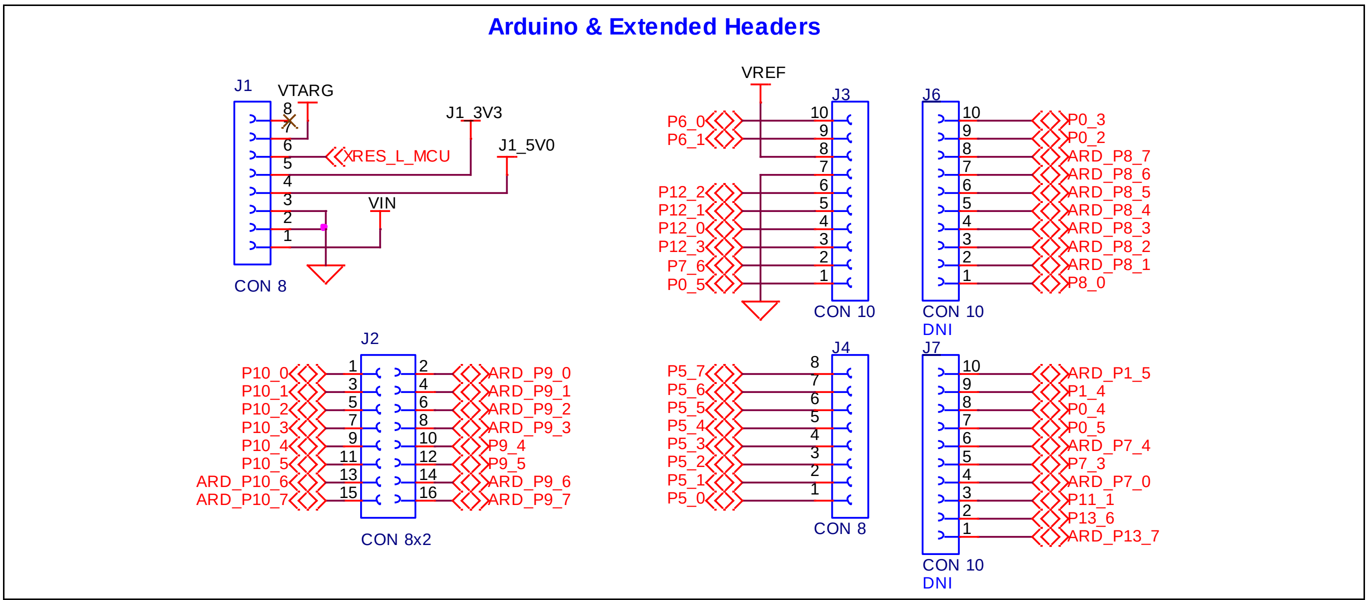


Figure 30 Headers compatible with Arduino Uno R3

3.2.7.4 WLAN/BT GPIO header (J5)

This header provides connectivity to a few of the M.2 Interface radio interface signals. The GPI/Os that are available will depend on the M.2 radio module connected. This connector is not populated by default.

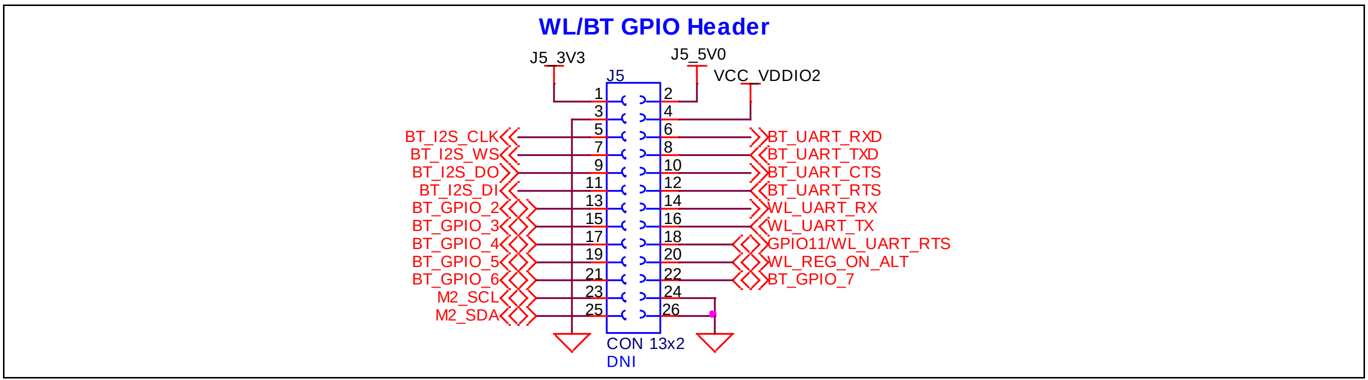


Figure 31 WLAN/BT GPIO header

3.2.8 CAPSENSE™ circuit

A CAPSENSE™ slider and two buttons, all supporting both self-capacitance (CSD) and mutual-capacitance (CSX) sensing are connected to the PSoC™ 6 MCU. Three external capacitors - CMOD for CSD, CINTA and CINTB for CSX are present on CY8CEVAL-062S2. Note that CINTB can be re-used as CSH (shield capacitor). For details on using CAPSENSE™ including design guidelines, see ***Getting started with CAPSENSE™***.

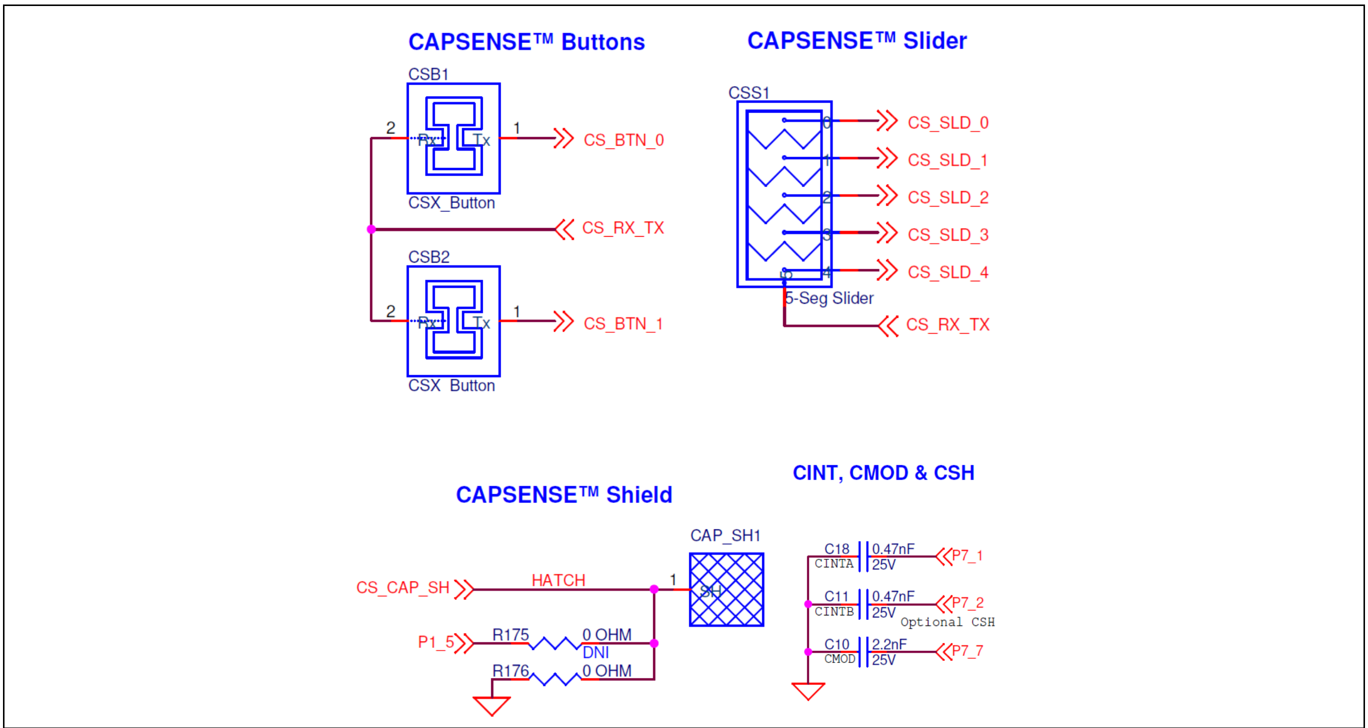


Figure 32 CAPSENSE™ circuit

Simultaneous GPIO switching with unrestricted drive strengths and frequency can affect CAPSENSE™ and ADC performance. For more details, see the “Errata” section of the corresponding device datasheet.

3.2.9 LEDs

The **D4** (amber) LED indicates the status of the KitProg3 (See the **KitProg3 user guide** for details). **D6** (amber) indicates the status of power supplied to the board.

The board also has two user-controllable LEDs (**D1** and **D2**) and an RGB LED (**D3**) connected to PSoC™ 6 MCU pins for user applications.

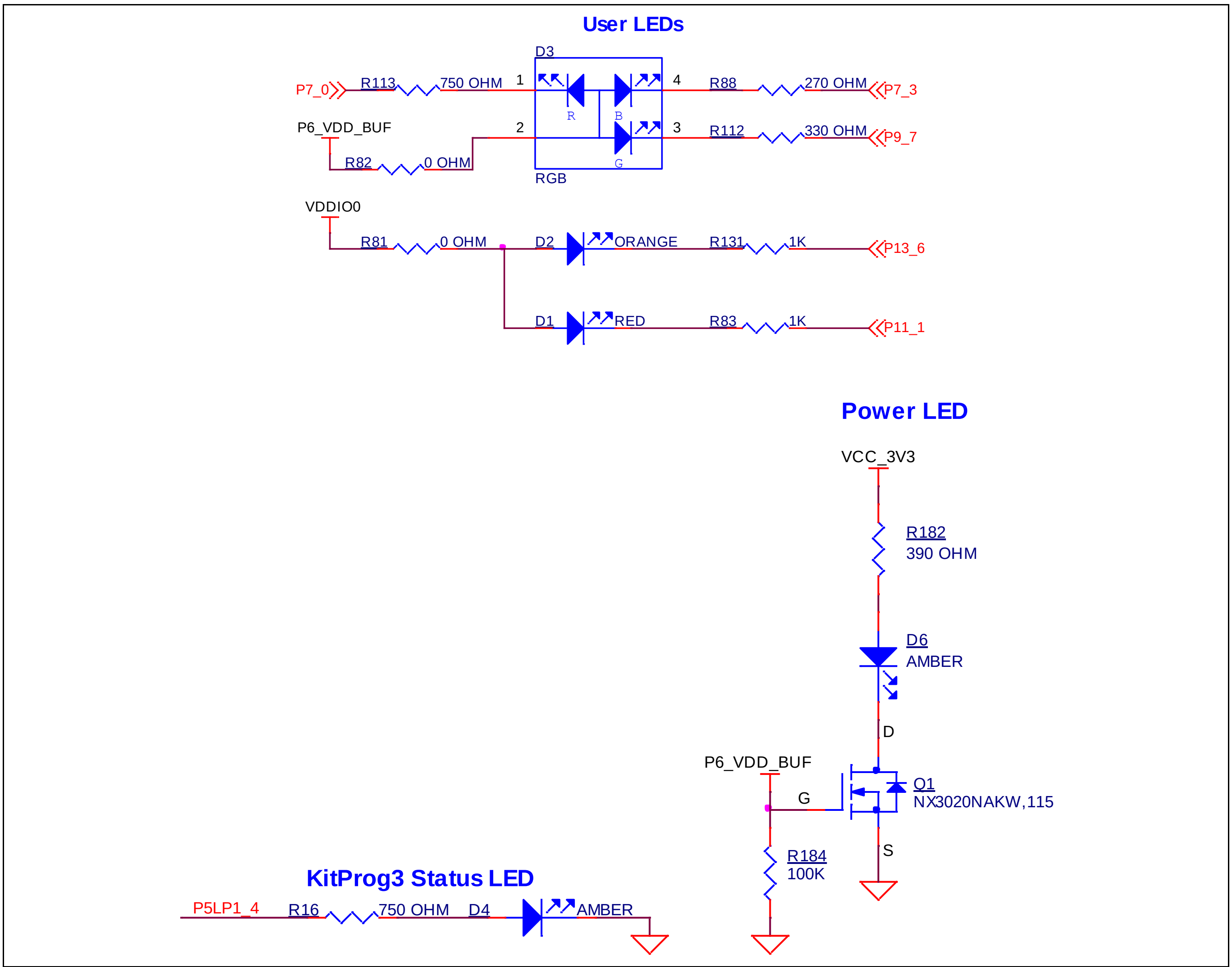


Figure 33 LEDs

3.2.10 Push buttons

The board has a reset button, two user-controllable buttons and a KitProg3 mode selection button. The reset button (SW1) is connected to the XRES pin of the PSoC™ 6 MCU device and is used to reset the device. Two user buttons (SW2 and SW4) are connected to pin P0[4] and P1[4] of the PSoC™ 6 MCU device respectively. In addition, the mode selection button (SW3) is connected to the PSoC™ 5LP device for programming mode selection. This button function is reserved for future use (see the **KitProg3 user guide** for details). All buttons are active LOW configuration and short to GND when pressed.

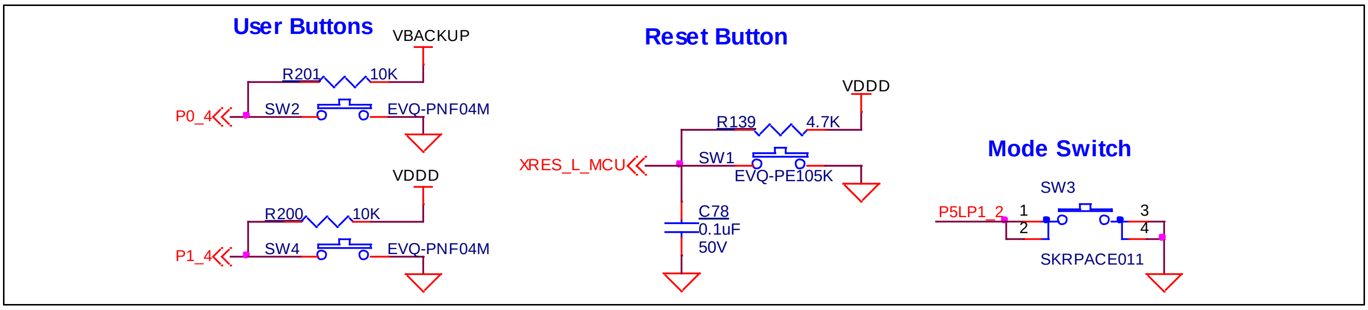


Figure 34 Push buttons

3.2.11 Crystal oscillators

The board has two crystal oscillators, a 32.768-kHz crystal connected to P0[0] and P0[1] and a 17.2032-MHz crystal connected to P12[6] and P12[7] of the PSoC™ 6 MCU.

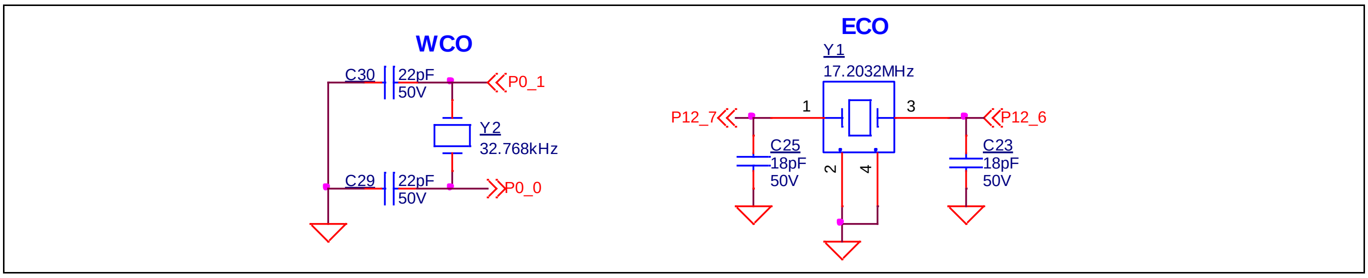


Figure 35 Crystal oscillators

3.2.12 Quad SPI NOR flash

The PSoC™ 62S2 evaluation board has a 512-Mbit Quad SPI NOR flash memory (S25FL512SAGMFIR10). The NOR flash is connected to the Quad SPI interface of the PSoC™ 6 MCU.

The NOR flash device can be used for both data and code with execute-in-place (XIP) support and encryption.

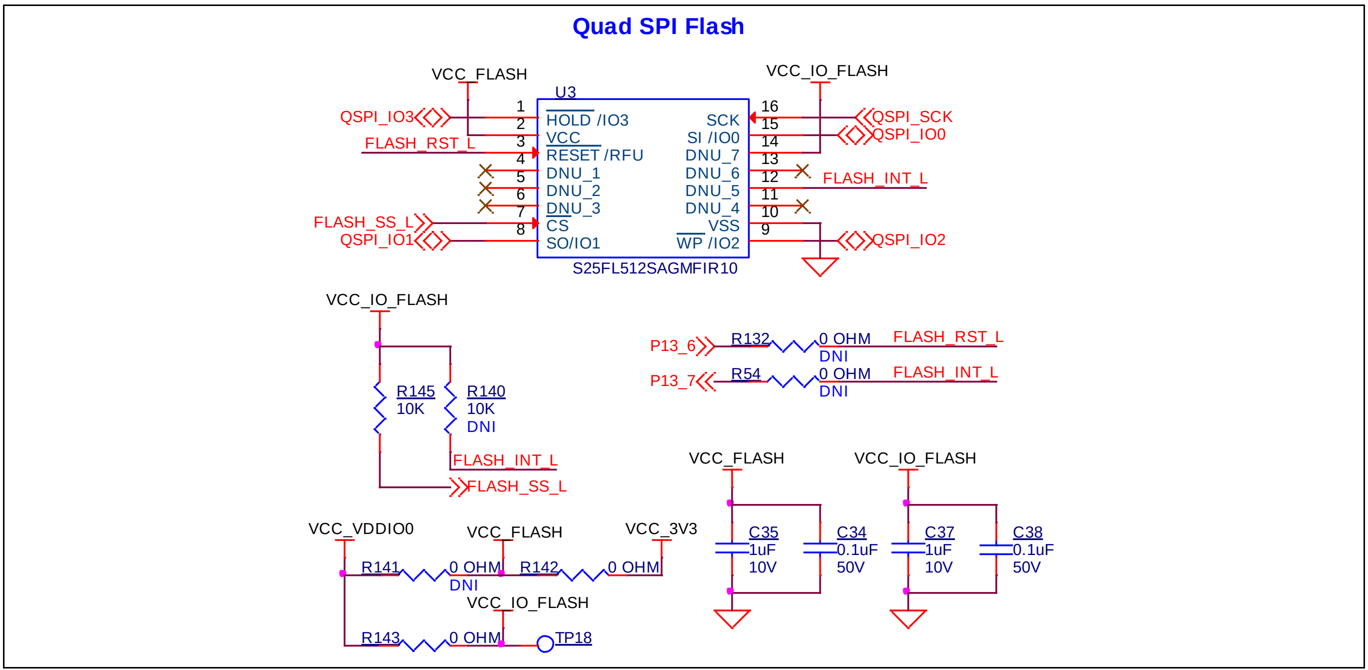


Figure 36 QSPI NOR flash

3.2.13 Quad SPI F-RAM

This kit contains the CY15B104QSN 4-Mbit (512K × 8) Excelon™ F-RAM device, which can be accessed through the Quad SPI interface, which is capable of Quad SPI speed up to 108 MHz but PSoC™ 6 MCU is limited to 80 MHz.

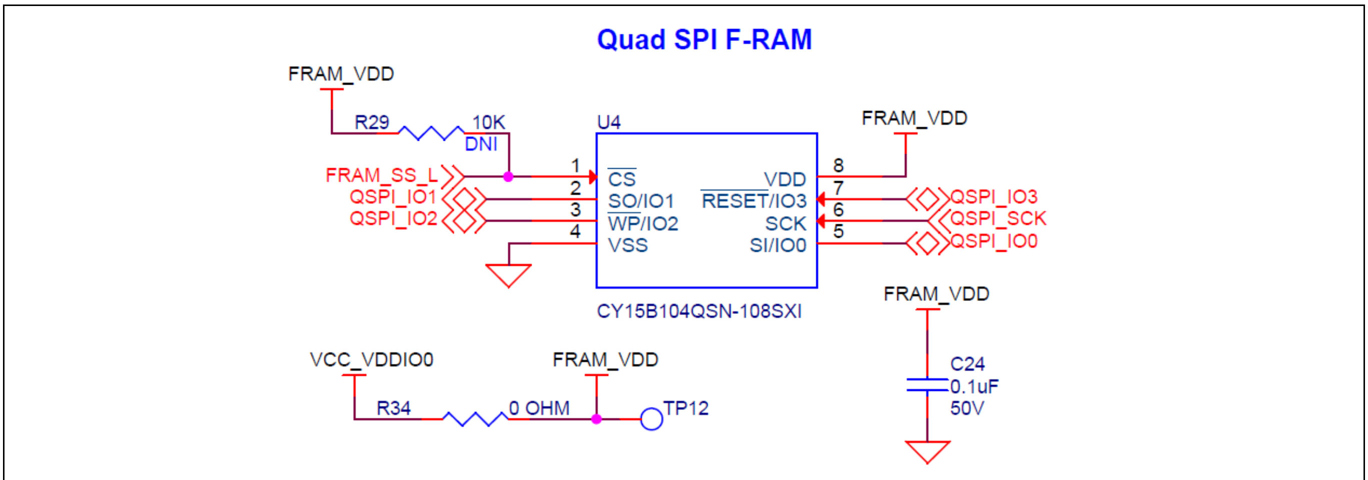


Figure 37 Quad SPI F-RAM

3.2.14 Infineon OPTIGA™ Trust M

This kit includes an OPTIGA™ Trust M high-end security controller. The OPTIGA™ Trust M is based on an advanced security controller with built-in tamper-proof NVM for secure storage and symmetric/asymmetric crypto engines to support ECC NIST curves up to P-521, ECC Brainpool curve up to P-512, RSA® up to 2048, AES key up to 256, HMAC up to SHA-512, HKDF up to SHA-512 and SHA-256. It is connected to the I²C interface of the PSoC™ 6 MCU.

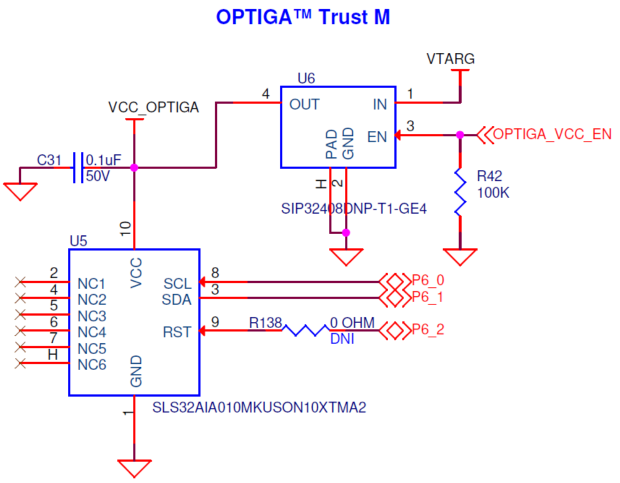


Figure 38 OPTIGA™ Trust M

3.2.15 microSD card

This kit contains a bottom-mounted microSD card holder with a card detect pin that is connected to the PSoC™ 6 MCU. It is powered by VDD_VDDIO0 (connected to VTARG by default). The PSoC™ 6 MCU is capable of UHS-I, but is limited to High-Speed mode (50-MHz clock) in this kit.

By default, the PSoC™ 6 MCU device is connected using an SDHC interface but optionally can be connected using SPI by re-working a few zero-ohm resistors.

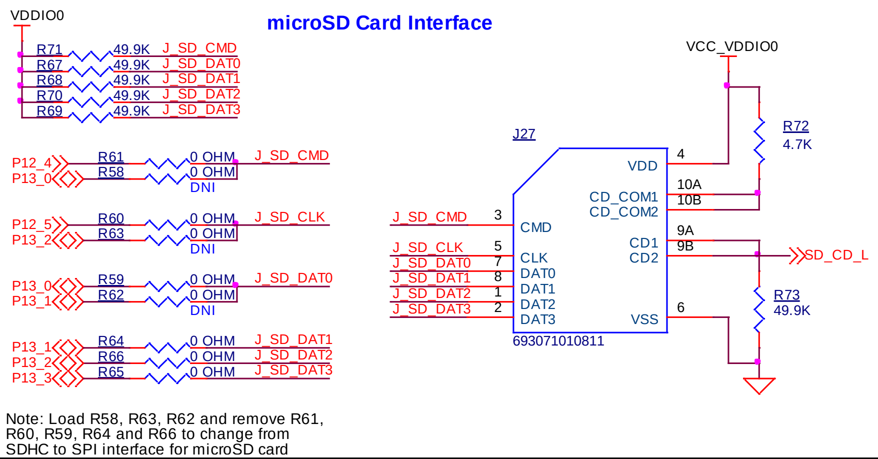


Figure 39 microSD card section

3.2.16 PSoC™ 6 MCU USB Micro-B connector

The board contains a micro-B USB connector for the PSoC™ 6 MCU. The board supports only USB device functionality, even though PSoC™ 6 MCU supports both device and host functionalities.

The resistors R55 and R56 are not loaded by default. If you want to use the PSoC™ 6 MCU USB detect feature, then use the following resistor values for different operating voltages.

Table 9 Resistor values for different operating voltages

VTARG	R55	R56	Voltage at P8[0]
2.5V/3.3V	47K	47K	2.5V
1.8V	68K	36K	1.73V

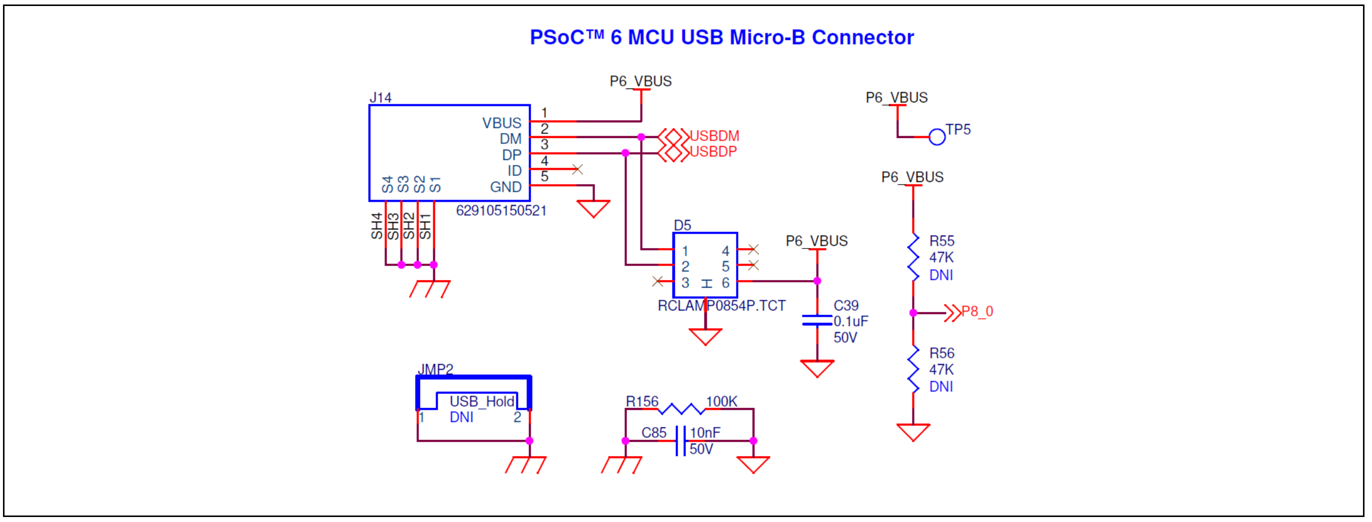


Figure 40 PSoC™ 6 MCU USB Micro-B connector

3.2.17 Potentiometer

The board contains a 10k potentiometer connected to pin A6 (P10[6]) of the header compatible with Arduino Uno R3 (J2). The fixed ends are connected to VDDA (VDD_POT through J21) and GND and therefore may contribute to leakage current on the P6_VDD supply. Remove jumper J21 to disconnect power from the potentiometer when measuring the P6_VDD current.

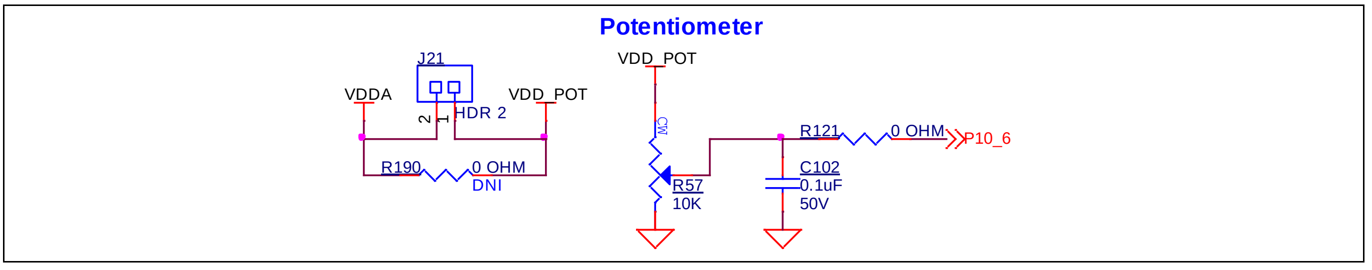


Figure 41 Potentiometer

3.3 Kit rework

3.3.1 CAPSENSE™ shield

The hatched pattern around the CAPSENSE™ buttons and slider are connected to ground. If liquid tolerance is required, this pattern must be connected to the shield pin P7[4] by populating resistor R174 and removing resistor R176. Pin P7[4] must be configured as a shield pin in the firmware. CINTB (C11) connected to P7[2] also must be configured as CSH (shield capacitor) in the firmware when using the CAPSENSE™ shield.

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

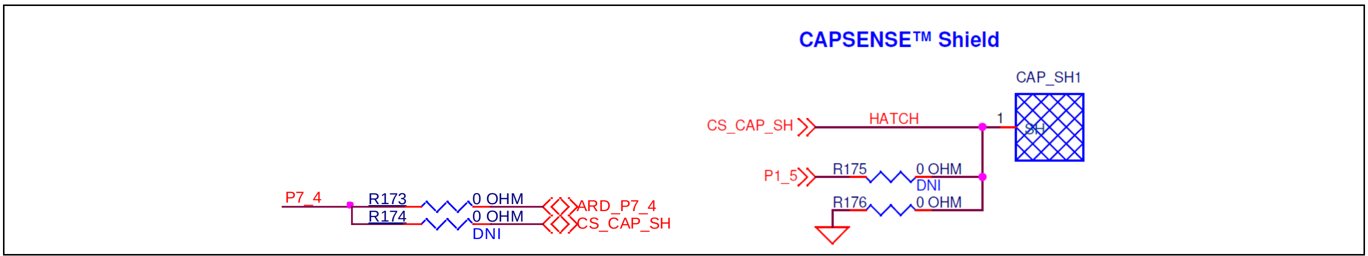


Figure 42 CAPSENSE™ shield

3.3.2 ETM trace header

The 20-pin ETM trace header is not loaded by default; the lines to the header are used as I/Os on header J2. To connect the PSoC™ 6 MCU to the trace header, populate resistors R11, R13, R19, R101, and R102 and remove resistors R9, R15, R20, R105, and R106.

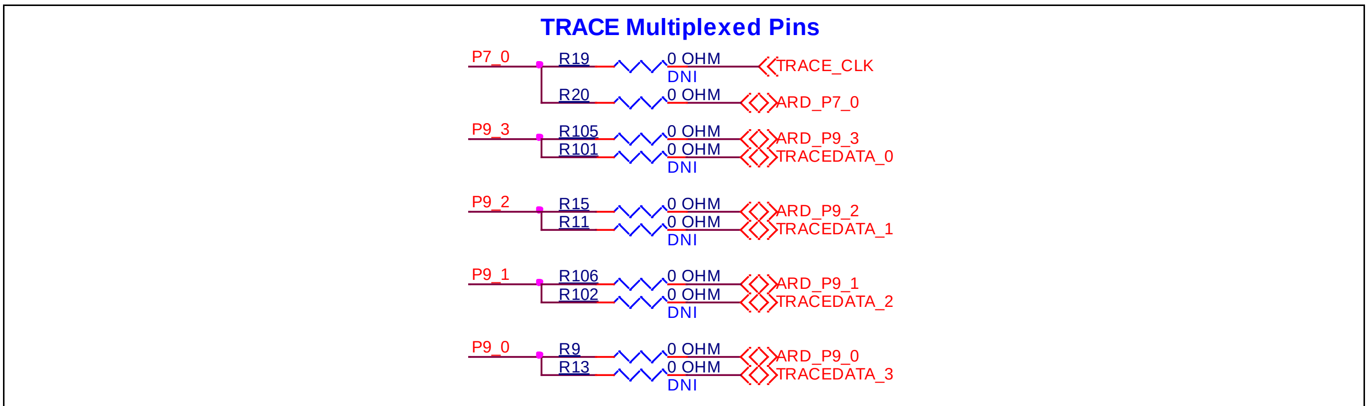


Figure 43 ETM trace header

3.3.3 microSD card detect multiplexing

On the PSoC™ 6 MCU, the default card detect pin for the SHDC block is P12[1]. However, on this kit, P13[7] is connected to the card detect pin on the microSD card slot. Therefore, the firmware must be modified to use P13[7] as the card detect pin. The default card detect pin (P12[1]) can work without any firmware modification. In order to use the default card detect pin, remove R136 and load R129. Optionally, P13[7] can be used as GPIO by removing R136 and loading R134 which connects it to an I/O header.

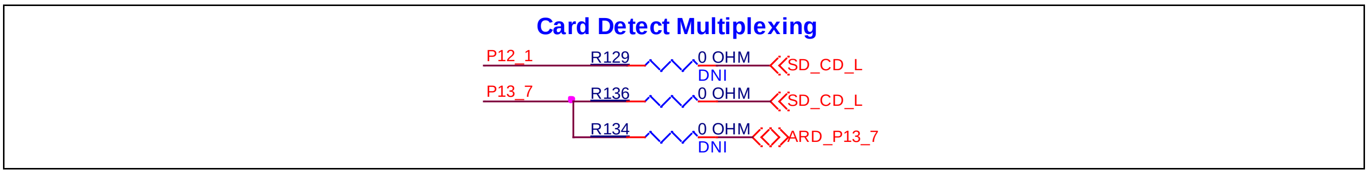


Figure 44 microSD card detect multiplexing

3.3.4 microSD card SPI multiplexing

The microSD card is connected by a 6-pin SDHC interface by default i.e., CLK, CMD, and DAT[0:3]. There is an optional provision to connect it over a 4-pin SPI interface i.e., CLK, MOSI, MISO, and SSEL. To do this, load R58, R62, and R63 and remove R59, R60, R61, R64, and R66.

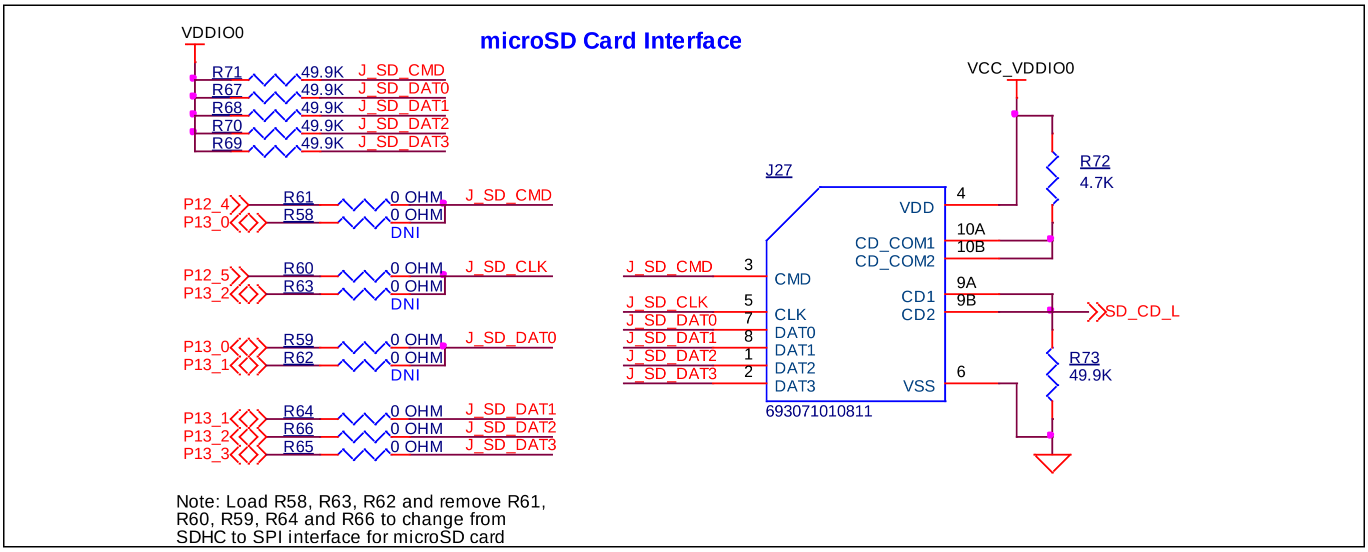


Figure 45 microSD card SPI multiplexing

3.3.5 WLAN interface SPI multiplexing

The WLAN interface to the M.2 connector is connected by a 6-pin SDHC interface by default i.e., CLK, CMD, and DAT[0:3]. There is an optional provision to connect it over a 4-pin SPI interface i.e., CLK, MOSI, MISO, and SSEL. To do this, load R37, R39, R43, R46, R50, and R130 and remove R35, R38, R40, R41, R45, and R48.

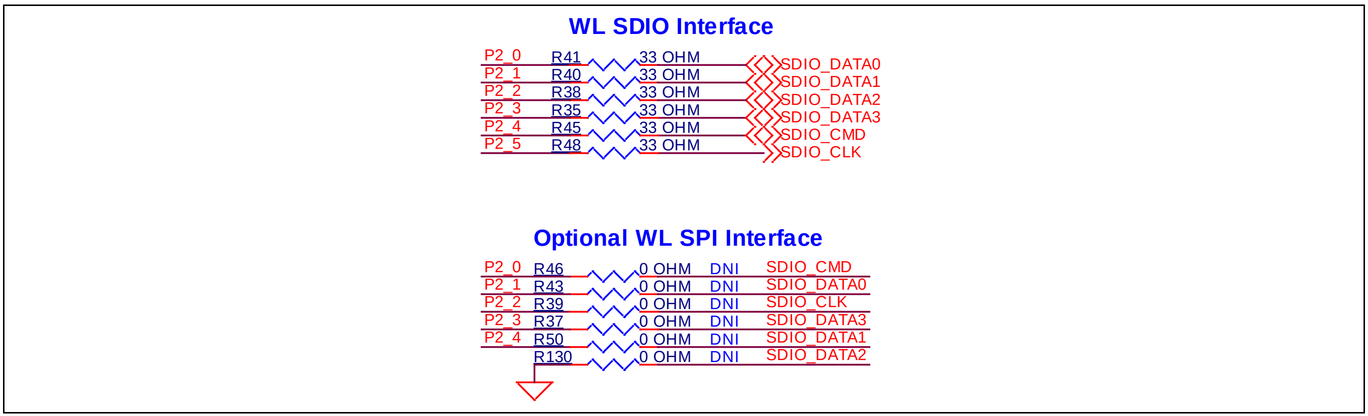


Figure 46 WLAN interface SPI multiplexing

3.3.6 UART interface between PSoC™ 5LP and PSoC™ 6 MCU with flow control

If any application requires the UART interface between PSoC™ 5LP and PSoC™ 6 MCU with flow control, populate R10, R12, R85, and R91 and remove R8 and R14.

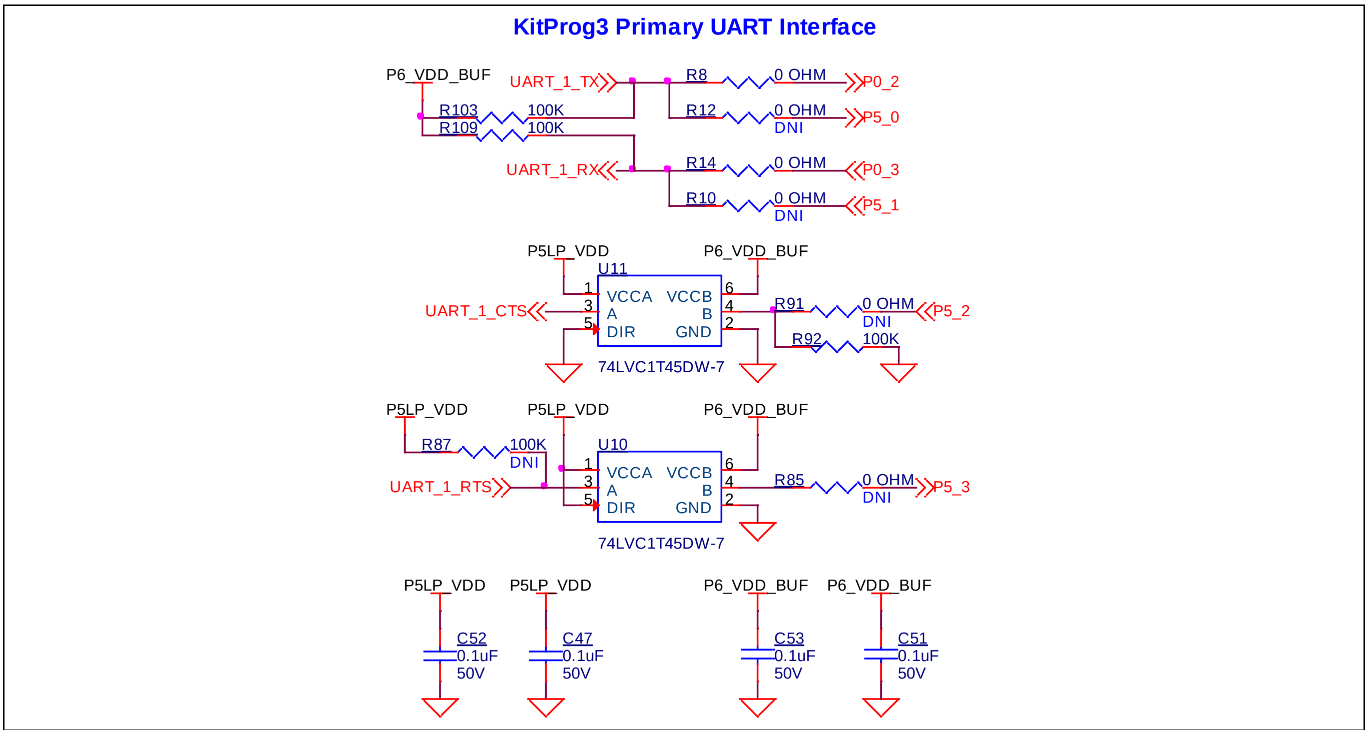


Figure 47 UART interface between PSoC™ 5LP and PSoC™ 6 MCU with flow control

3.3.7 Secondary UART interface between PSoC™ 5LP and M.2 module

There is a secondary UART interface between the PSoC™ 5LP device and the M.2 module which is disabled by default. In order to enable the Bluetooth® UART interface, populate resistors R77, R76, R75, and R74. On the other hand, the WL UART interface can be enabled by populating R79 and R78 depending on the communication options of the M.2 module.

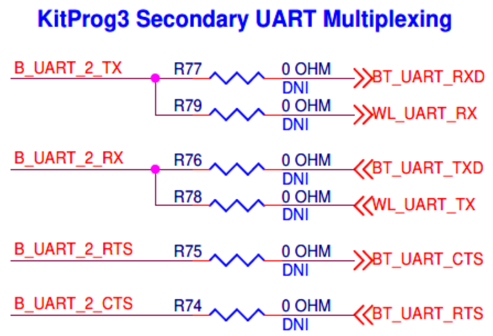


Figure 48 Secondary UART interface between PSoC™ 5LP and M.2 module

3.4 Bill of materials

Refer to the BOM file available on the **kit webpage**.

3.5 Frequently asked questions

1. How does CY8CEVAL-062S2 handle voltage connection when multiple power sources are plugged in?

   There are three different options to power the kit:

   - KitProg3 Micro-B USB connector (**J9**)
   - PSoC™ 6 MCU Micro-B USB connector (**J14**)
   - External DC supply via VIN connector (**J8**)

   The voltage from each of the sources in passed through ORing diodes that supply VCC_IN.

2. What are the input voltage tolerances? Is there any overvoltage protection on this kit?

   Input voltage levels are shown in **Table 10**:

   There is no overvoltage protection on this Kit.

Table 10 Input voltage levels

Supply	Typical input voltage	Absolute max
USB Micro-B connector (**J9, J14**)	4.5 V to 5.5 V	5.5 V
VIN connector (**J8**)	5 V to 12 V	18 V

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

   PSoC™ 6 MCU is not meant to be operated at voltages greater than 3.6 V. Powering PSoC™ 6 MCU to more than 4 V will damage the chip. It is recommended to power PSoC™ 6 MCU at 3.3 V.

4. I am unable to program the target device.
   - Check **J18** to ensure that the jumper shunt is placed.
   - Make sure that no external devices are connected to the external programming header J24.
   - Update your KitProg3 version to the latest one using the steps mentioned in the **KitProg3 user guide**.

5. What additional overlays can be used with CAPSENSE™?

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

6. Can I power the kit using external program/debug headers J22 and J24?

   No, this is not possible in this board by default. The target MCU is powered by on-board regulators only and therefore, one of the three main sources (J8, J9, and J14) must be present. There is a protection circuit that prevents reverse voltage from VTARG_REF to VTARG. Therefore, the board cannot be powered through J22 and J24. However, this can be by-passed by loading R185.

   Note: This modification is not recommended as the target MCU will have no protection and will be permanently damaged if 5 V is supplied.

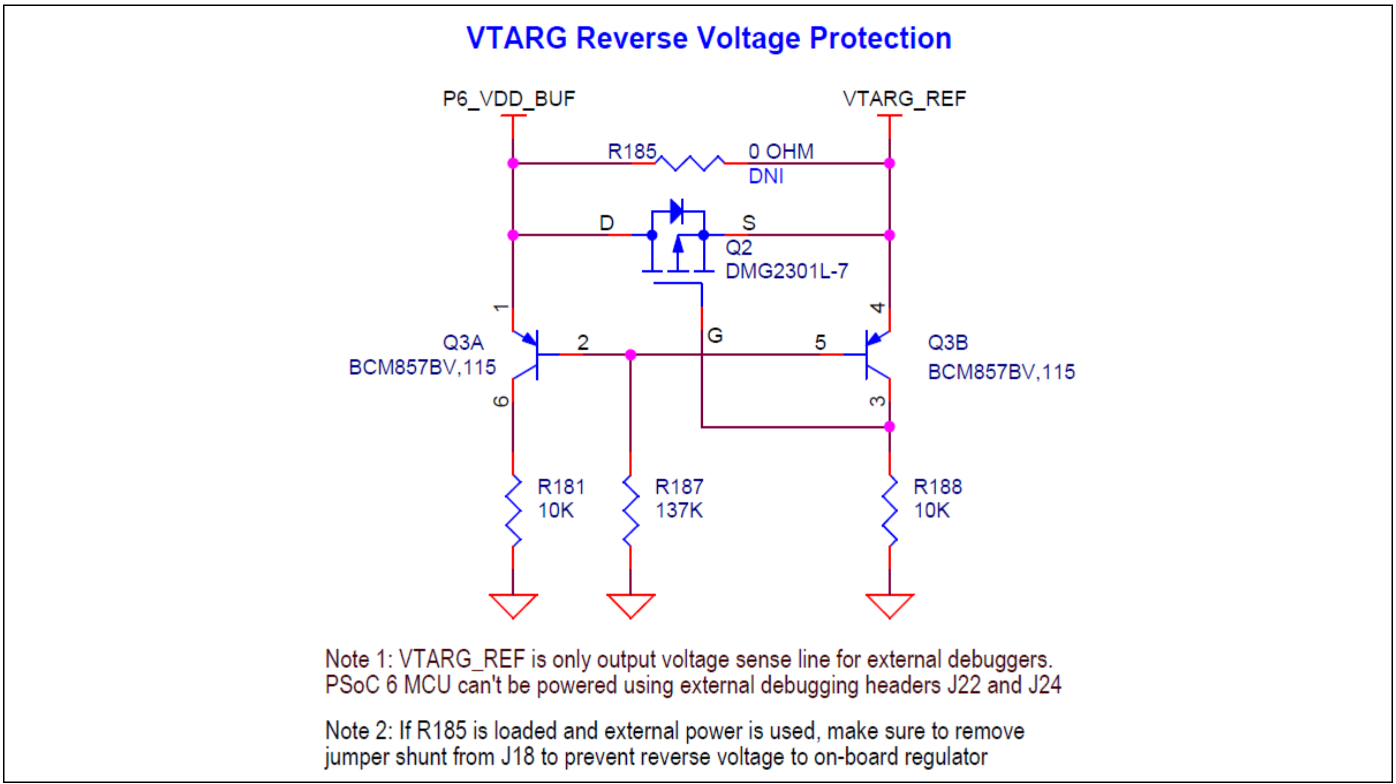


Figure 49 **VTARG reverse voltage protection**

Revision history

Major changes since the last revision

Date	Version	Description
2021-06-08	**	Initial release.
2021-07-28	*A	Internal release.
2021-11-22	*B	Updated **Figure 5**. Updated **Figure 6**. Updated **Figure 12**. Updated **Figure 13**. Updated **Figure 16**. Updated **Figure 20**. Updated **Figure 22**. Updated **Figure 25**. Updated **Figure 32**. Updated **Figure 38**. Added **Table 9**. Updated **Figure 40**. Updated **Figure 42**.
2022-01-24	*C	Updated **Figure 1**. Updated **Figure 2**. Updated **Figure 3**. Updated **Figure 4**. Updated **Figure 7**. Updated **Figure 8**. Updated **Table 4**. Updated **Figure 9**. Updated **Figure 10**.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition <2022-01>
Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Go to **www.cypress.com/support**

Document reference
002-33177 Rev. *C