KIT_XMC72_EVK XMC7200 evaluation kit guide

About this document

Scope and purpose

This document explains about the KIT_XMC72_EVK XMC7200 evaluation kit: kit operation, out-of-the-box example and its operation, and the hardware details of the board.

Intended audience

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

Evaluation Board

This board is to be used during the design-in process for evaluating and measuring characteristic curves, and for checking datasheet specifications.

Note: *PCB and auxiliary circuits are NOT optimized for final customer design.*

Reference Board/Kit

Product(s) embedded on a PCB with a focus on specific applications and defined use cases that may include software. PCB and auxiliary circuits are optimized for the requirements of the target application.

Note: *Boards do not necessarily meet safety, EMI, quality standards (for example UL, CE) requirements*

Important notice

“Evaluation Boards and Reference Boards” shall mean products embedded on a printed circuit board (PCB) for demonstration and/or evaluation purposes, which include, without limitation, demonstration, reference and evaluation boards, kits and design (collectively referred to as “Reference Board”).

Environmental conditions have been considered in the design of the Evaluation Boards and Reference Boards provided by Infineon Technologies. The design of the Evaluation Boards and Reference Boards has been tested by Infineon Technologies only as described in this document. The design is not qualified in terms of safety requirements, manufacturing and operation over the entire operating temperature range or lifetime.

The Evaluation Boards and Reference Boards provided by Infineon Technologies are subject to functional testing only under typical load conditions. Evaluation Boards and Reference Boards are not subject to the same procedures as regular products regarding returned material analysis (RMA), process change notification (PCN) and product discontinuation (PD).

Evaluation Boards and Reference Boards are not commercialized products, and are solely intended for evaluation and testing purposes. In particular, they shall not be used for reliability testing or production. The Evaluation Boards and Reference Boards may therefore not comply with CE or similar standards (including but not limited to the EMC Directive 2004/EC/108 and the EMC Act) and may not fulfill other requirements of the country in which they are operated by the customer. The customer shall ensure that all Evaluation Boards and Reference Boards will be handled in a way which is compliant with the relevant requirements and standards of the country in which they are operated.

The Evaluation Boards and Reference Boards as well as the information provided in this document are addressed only to qualified and skilled technical staff, for laboratory usage, and shall be used and managed according to the terms and conditions set forth in this document and in other related documentation supplied with the respective Evaluation Board or Reference Board.

It is the responsibility of the customer’s technical departments to evaluate the suitability of the Evaluation Boards and Reference Boards for the intended application, and to evaluate the completeness and correctness of the information provided in this document with respect to such application.

The customer is obliged to ensure that the use of the Evaluation Boards and Reference Boards does not cause any harm to persons or third party property.
The Evaluation Boards and Reference Boards and any information in this document is provided "as is" and Infineon Technologies disclaims any warranties, express or implied, including but not limited to warranties of non-infringement of third party rights and implied warranties of fitness for any purpose, or for merchantability.

Infineon Technologies shall not be responsible for any damages resulting from the use of the Evaluation Boards and Reference Boards and/or from any information provided in this document. The customer is obliged to defend, indemnify and hold Infineon Technologies harmless from and against any claims or damages arising out of or resulting from any use thereof.

Infineon Technologies reserves the right to modify this document and/or any information provided herein at any time without further notice.

Safety precautions

Note: *Please note the following warnings regarding the hazards associated with development systems*

Table 1 Safety precautions

	**Caution:** *The evaluation or reference board contains parts and assemblies sensitive to electrostatic discharge (ESD). Electrostatic control precautions are required when installing, testing, servicing or repairing the assembly. Component damage may result if ESD control procedures are not followed. If you are not familiar with electrostatic control procedures, refer to the applicable ESD protection handbooks and guidelines.*
	**Caution:** *The evaluation or reference board is shipped with packing materials that need to be removed prior to installation. Failure to remove all packing materials that are unnecessary for system installation may result in overheating or abnormal operating conditions.*

Table of contents

About this document **1**
Important notice **2**
Safety precautions **3**
Table of contents **4**
1 Introduction **6**
1.1 Getting started 6
1.2 Additional learning resources 7
1.3 Technical support 7
1.4 Documentation conventions 7
1.5 Abbreviations and definitions 7
2 Kit details **9**
2.1 Kit contents 9
2.2 Board details 9
3 Kit operation **20**
3.1 Theory of operation 20
3.2 M.2 wireless connectivity module 26
3.3 Board support package (BSP) selection 26
3.4 KitProg3: onboard programmer/debugger 26
3.4.1 Programming and debugging using ModusToolbox™ software 26
3.4.1.1 Using the OOB example – XMC7000 MCU: OOB demo 31
3.4.2 USB-UART bridge 32
3.4.3 USB-I2C bridge 33
4 Hardware **34**
4.1 Schematics 34
4.2 Hardware functional description 34
4.2.1 XMC7200D (U1) 34
4.2.2 M.2 interface connector (J25) 36
4.2.3 PSoC™ 5LP-based KitProg3 (U2) 38
4.2.4 Serial interconnection between PSoC™ 5LP and XMC7200D 39
4.2.5 Serial interconnection between PSoC™ 5LP and M.2 interface 40
4.2.6 Power supply system 41
4.2.6.1 Voltage regulators 41
4.2.6.2 Voltage selection 44
4.2.6.3 Current measurement headers 45
4.2.7 I/O headers 46
4.2.7.1 Headers compatible with Arduino Uno R3 (J1, J2, J3, J4) 46
4.2.7.2 XMC7200D I/O headers (J29, J30, J31 and J32) 46
4.2.8 CAN FD Transceiver 47
4.2.9 Quad SPI NOR flash 47
4.2.10 LEDs 48
4.2.11 Push buttons 48
4.2.12 Crystals and oscillators 49
4.2.13 Ethernet transceiver and RJ45 connector 49
4.2.14 SMIF dual header compatible with Digilent Pmod 50
4.2.15 Potentiometer 51
4.2.16 Reverse voltage protection 52
4.3 Kit rework 53

Table of contents

4.3.1 SMIF dual header compatible with Digilent Pmod interface....53
4.3.2 VDDIO2 supply voltage selection....54
4.3.3 UART interface on headers compatible with Arduino....54
4.3.4 WLAN UART interface....55
4.3.5 ETM trace and debug header....55
4.3.6 125-MHz oscillator....55
4.3.7 WLAN SDIO interface....56
4.3.8 UART interface between PSoC™ 5LP and XMC7200D with flow control....56
4.4 Bill of materials....56
4.5 Frequently asked questions....57
References....59
Revision history....60
Disclaimer....61

1 Introduction

The XMC7200 evaluation kit enables you to evaluate and develop your applications using the **XMC7200D microcontroller** (hereafter called "XMC7200D").

XMC7200D is designed for industrial applications. XMC7200D is a true programmable embedded system-on-chip, integrating up to two 350-MHz Arm® Cortex®-M7 as the primary application processors, 100-MHz Arm® Cortex®-M0+ that supports low-power operations, up to 8-MB flash and 1-MB SRAM, Gigabit Ethernet, Controller Area Network Flexible Data-Rate (CAN FD), Secure Digital Host Controller (SDHC) supporting SD/SDIO/eMMC interfaces, programmable analog and digital peripherals that allow faster time-to-market.

The evaluation boardhas a M.2 interface connector for interfacing radio modules-based on AIROC™ Wi-Fi & Bluetooth® combos[1], SMIF dual header compatible with Digilent Pmod for interfacing HYPERBUS™ memories[1], and headers compatible with Arduino for interfacing Arduino shields. In addition, the board features an onboard programmer/debugger (KitProg3), a 512-Mbit QSPI NOR flash, CAN FD transceiver, Gigabit Ethernet PHY transceiver with RJ45 connector interface, a micro-B connector for USB device interface, three user LEDs, one potentiometer, and two push buttons. The board supports operating voltages from 3.3 V to 5.0 V for XMC7200D.

ModusToolbox™ software is used to develop and debug the XMC7200D projects. **ModusToolbox™ software** is a set of tools that enable you to integrate these devices into your existing development methodology.

For more details on XMC7200D, refer to the application note **AN234334 - Getting started with XMC7000 MCU on ModusToolbox™ software** which aids in creating a customized design using the Eclipse IDE for ModusToolbox™ software.

1.1 Getting started

The guide has the following sections:

- The **Kit details** chapter provides the kit .package,and board details.
- The **Kit operation** chapter describes the major features of the XMC7200 evaluation board and their 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 XMC7200 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 XMC7000 SDK with XMC7200D. ModusToolbox™ software, enables and configures device resource, middleware libraries; write C/assembly source code; and program and debug the device. The software can be downloaded from the **ModusToolbox™ home page**. For more details, refer to the ModusToolbox™ software installation guide.
- There are wide range of code examples to evaluate the XMC7200 board. These examples help to familiarize with XMC7200D, and help to create a customized design. These examples can be accessed through ModusToolbox™ Project Creator tool. To acess the code examples, visit Infineon's Code Example page.

[1] Currently there is no software support.

1.2 Additional learning resources

Infineon provides a wealth of data at **www.infineon.com/cms/en/product/microcontroller/32-bit-industrial-microcontroller-based-on-arm-cortex-m,** to select the right XMC™ MCU for the design and to to quickly and effectively integrate the device into the design.

1.3 Technical support

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

1.4 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.

1.5 Abbreviations and definitions

Table 3 Abbreviations

Abbreviation	Definition
ADC	analog-to-digital converter
BOM	bill of materials
CAN	controller area network
CPU	Central Processing Unit
DC	direct current
ECO	external crystal oscillator
ESD	electrostatic discharge
ETM	Embedded Trace Macrocell
FPC	Flexible Printed Circuit
GPIO	general purpose input/output
IC	Integrated Circuit
IDE	integrated development environment
IoT	Internet of Things

Downloaded from Arrow.com.

Introduction

Abbreviation	Definition
I2C	Inter-Integrated Circuit
I2S	inter-IC sound
JTAG	Joint Test Action Group
LED	light-emitting diode
LPO	low-power oscillator
MAC	medium access control
PC	personal computer
PDL	Peripheral Driver Library
POT	potentiometer
QSPI	Quad Serial Peripheral Interface
RGMII	Reduced Gigabit Media Independent 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 details

2.1 Kit contents

- XMC7200 evaluation board
- USB Type-A to Micro-B cable
- 12 V/3A DC power adapter with additional blades
- Six jumper wires (five inches each)
- Quick start guide



Figure 1 Kit contents

For information on missing parts, go to **www.infineon.com/support**.

2.2 Board details

The XMC7200 evaluation board has the following features:

- XMC7200D – XMC7200D-E272K8384. See the device **datasheet**.
- M.2 interface connector to connect radio modules based on AIROC™ Wi-Fi & Bluetooth® combos (currently not supported)
- 512-Mbit external Quad SPI NOR flash that provides a fast, expandable memory for data and code
- KitProg3 onboard SWD programmer/debugger, USB-UART and USB-I2C bridge functionality
- A micro-B connector for USB device interface
- Selectable input supply voltages of 3.3 V and 5.0 V for the XMC7200D
- Gigabit Ethernet interface

Downloaded from Arrow.com.

Kit details

- CAN FD interface
- Three user LEDs, two user buttons, and a reset button for the XMC7200D
- A potentiometer which can be used to simulate analog sensor output
- A mode button and a mode LED for KitProg3

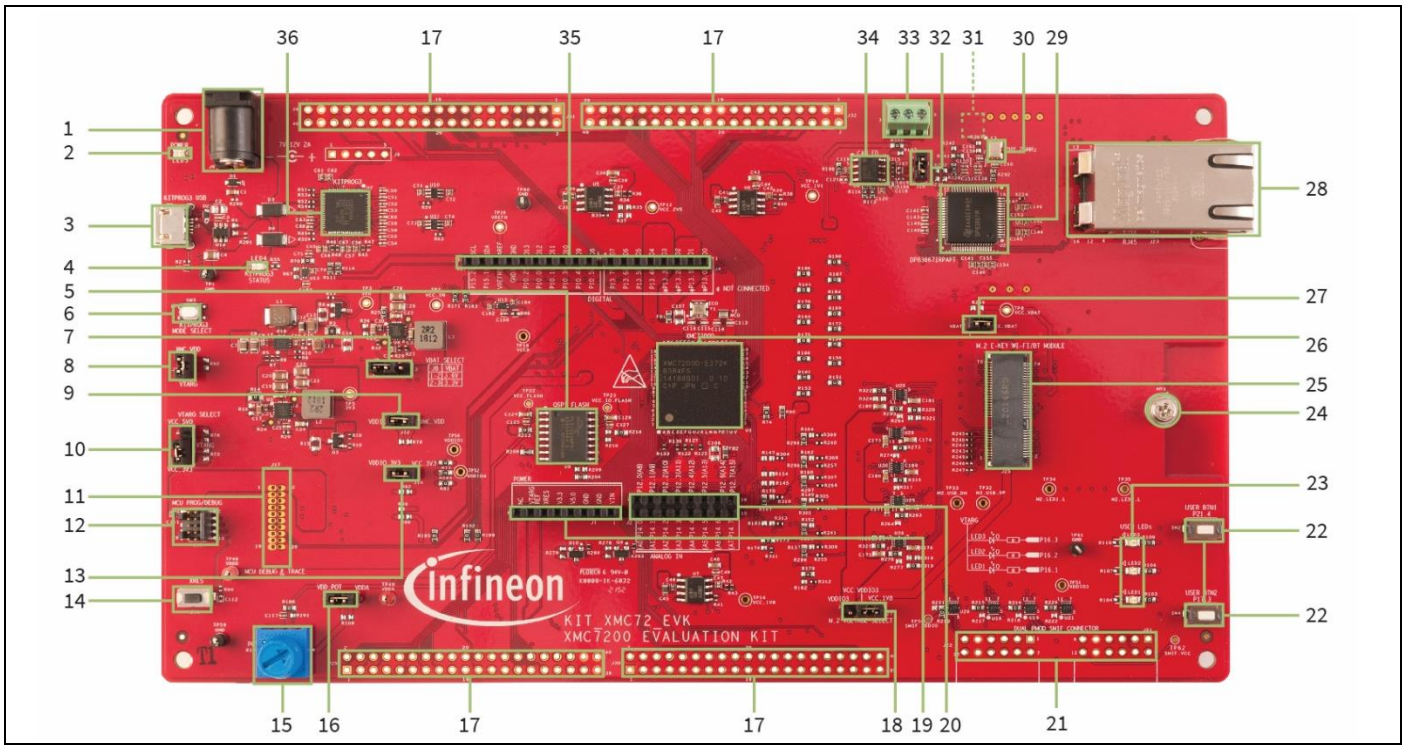


Figure 2 XMC7200 evaluation board – top view

The XMC7200 evaluation board has the following interfaces as shown in Figure 2

1 External power supply VIN connector (J6)

2 Power LED (LED5)

3 KitProg3 USB connector (J7)

4 KitProg3 status LED (LED4)

5 512-Mbit serial NOR flash memory (S25FL512S, U9)

6 KitProg3 programming mode selection button (SW3)

7 VBAT power selection jumper (J8)

8 XMC7200D VTARG current measurement jumper (J15)

9 XMC7200D VDDIO2 current measurement jumper (J12)

10 System power (VTARG) selection jumper (J10)

11 XMC7200D 20-pin debug and trace header (J17)*

12 XMC7200D 10-pin SWD/JTAG program and debug header (J16)

13 XMC7200D target I/O current measurement jumper (J14)

Kit details

14 XMC7200D reset button (SW1)

15 Potentiometer (R105)

16 Potentiometer connection jumper (J18)

17 XMC7200D extended I/O headers (J29, J30, J31, J32)*

18 M.2 I/O power selection jumper (J27)

19 Power header compatible with Arduino Uno R3 (J1)

20 Analog-IN header compatible with Arduino Uno R3 (J2)

21 SMIF dual header compatible with Digilent Pmod (J21, J22)*

22 XMC7200D user buttons (SW2, SW4)

23 XMC7200D user LEDs (LED1, LED2, LED3)

24 M.2 stand-off (MT1)

25 M.2 interface connector (J25)

26 XMC7200D microcontroller (XMC7200D-E272K8384 – U1)

27 VBAT current measurement jumper (J28)

28 RJ45 Gigabit Ethernet connector (J23)

29 Ethernet physical layer (PHY) transceiver (U22)

30 25-MHz crystal for Ethernet transceiver (Y3)

31 125-MHz crystal for Ethernet transceiver (Y4)**

32 CAN FD resistor termination jumper (J20)

33 CAN FD interface connector (J19)

34 CAN FD transceiver (TLE9251VSJ – U8)

35 Digital I/O headers compatible with Arduino Uno R3 (J3, J4)

36 KitProg3 programmer and debugger based on PSoC™ 5LP (CY8C5868LTI-LP039 – U2)

* Footprint only, not populated on the board.

** Component is located at the bottom side of the board.

Kit details

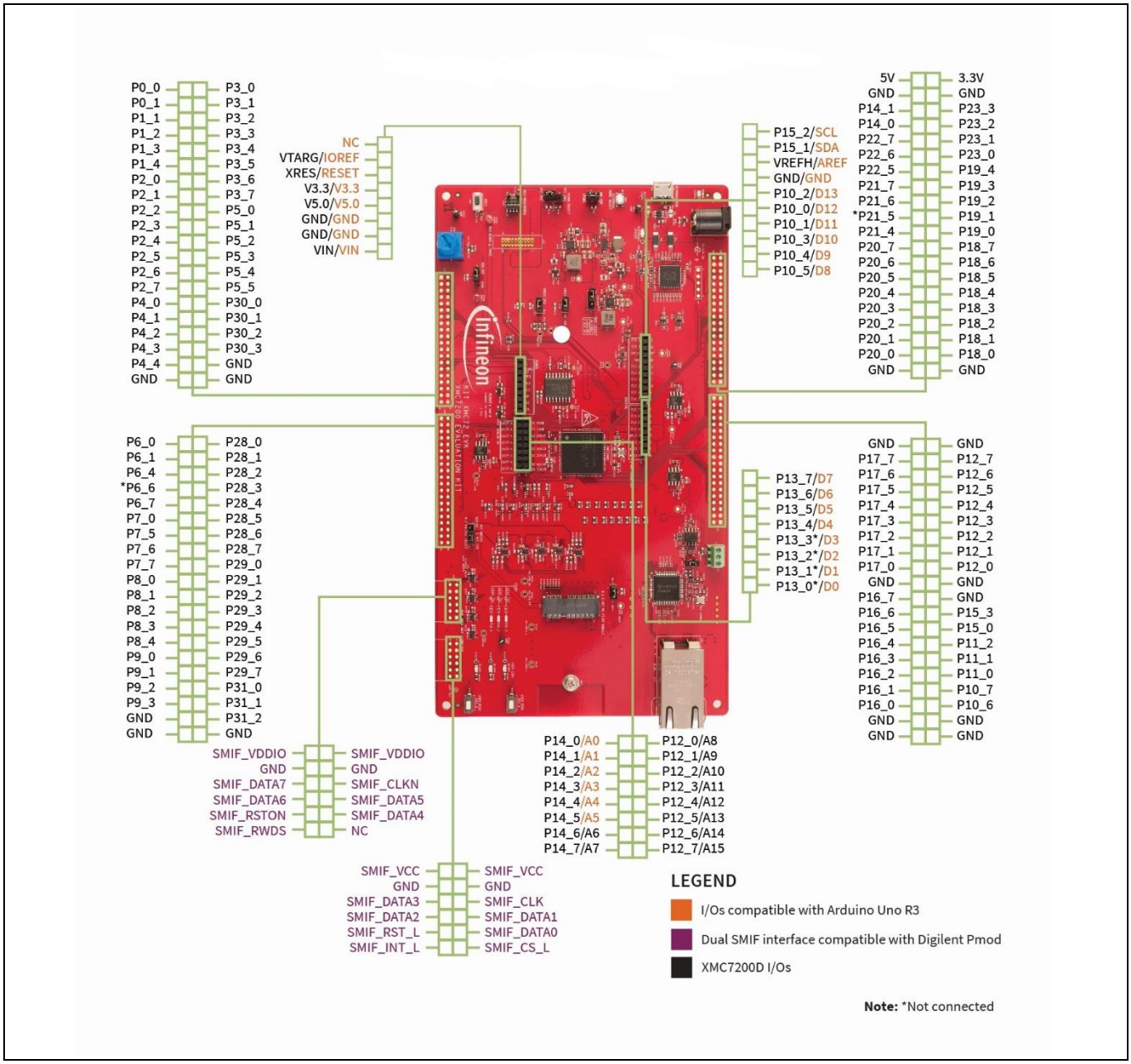


Figure 3 XMC7200 evaluation board pinout

Kit details

Table 4 XMC7200 evaluation board pinout

Pin	Primary onboard function	Secondary onboard function	Connection details
XMC7200D pins			
XRES	Hardware reset	–	Remove R183 to disconnect it from KitProg3 reset.
P0[2]	CAN transmit CAN_TX	–	Connected to TXD of CAN FD transceiver TLE9251VSJ.
P0[3]	CAN receive CAN_RX	–	Connected to RXD of CAN FD transceiver TLE9251VSJ.
P1[0]	CAN transceiver standby	–	Connected to STB of CAN FD transceiver TLE9251VSJ.
P6[2]	FLASH_INT_L	–	Not connected
P6[3]	QSPI_SCK	–	Connected to SCK of QSPI flash S25FL512SAGMFMR10.
P6[5]	FLASH_SS_L	–	Connected to CS of QSPI flash S25FL512SAGMFMR10.
P6[6]	Potentiometer (POT) output POT_AOUT	Generic GPIO expansion header (J30.7)	Remove R188 to disconnect from the potentiometer. Populate R191 to connect to expansion header J30.7.
P6[7]	QSPI flash reset	Generic GPIO expansion header (J30.9)	Connected to RESET/RFU of QSPI flash S25FL512SAGMFMR10. Remove R140 to use as GPIO.
P7[1]	QSPI_IO0	–	Connected to IO0 of QSPI flash S25FL512SAGMFMR10.
P7[2]	QSPI_IO1	–	Connected to IO1 of QSPI flash S25FL512SAGMFMR10.
P7[3]	QSPI_IO2	–	Connected to IO2 of QSPI flash S25FL512SAGMFMR10.
P7[4]	QSPI_IO3	–	Connected to IO3 of QSPI flash S25FL512SAGMFMR10.
P10[0]	D12 – header compatible with Arduino (J3.5)	–	–
P10[1]	D11 – header compatible with Arduino (J3.4)	–	–
P10[2]	D13 – header compatible with Arduino (J3.6)	–	–
P10[3]	D10 – header compatible with Arduino (J3.3)	–	–

Kit details

Pin	Primary onboard function	Secondary onboard function	Connection details
P10[4]	D9 – header compatible with Arduino (J3.2)	–	–
P10[5]	D8 – header compatible with Arduino (J3.1)	–	–
P12[0]	A8 – header compatible with Arduino (J2.2)	–	–
P12[1]	A9 – header compatible with Arduino (J2.4)	–	–
P12[2]	A10 – header compatible with Arduino (J2.6)	–	–
P12[3]	A11 – header compatible with Arduino (J2.8)	–	–
P12[4]	A12 – header compatible with Arduino (J2.10)	–	–
P12[5]	A13 – header compatible with Arduino (J2.12)	–	–
P12[6]	A14 – header compatible with Arduino (J2.14)	–	–
P12[7]	A15 – header compatible with Arduino (J2.16)	–	–
P13[0]	Debug UART_RX	Pin D0 of connector J4.1 compatible with Arduino	This pin is connected to KitProg3 UART TX pin. Remove R129, R139 and install R134 to connect to Pin D0 of connector J4.1 compatible with Arduino.
P13[1]	Debug UART_TX	Pin D1 of connector J4.2 compatible with Arduino	This pin is connected to KitProg3 UART RX pin. Remove R132, R142 and install R136 to connect to Pin D1 of connector J4.2 compatible with Arduino.
P13[2]	Debug UART_RTS	Pin D2 of connector J4.3 compatible with Arduino	This pin is connected to KitProg3 UART CTS pin. Remove R146 and install R150 to connect to Pin D2 of connector J4.3 compatible with Arduino.

Kit details

Pin	Primary onboard function	Secondary onboard function	Connection details
P13[3]	Debug UART_CTS	Pin D3 of connector J4.4 compatible with Arduino	This pin is connected to KitProg3 UART RTS pin. Remove R144 and install R148 to connect to Pin D3 of connector J4.4 compatible with Arduino.
P13[4]	Pin D4 of connector J4.5 compatible with Arduino	–	–
P13[5]	Pin D5 of connector J4.6 compatible with Arduino	–	–
P13[6]	Pin D6 of connector J4.7 compatible with Arduino	–	–
P13[7]	Pin D7 of connector J4.8 compatible with Arduino	–	–
P14[0]	A0 – header compatible with Arduino (J2.1)	–	–
P14[1]	A1 – header compatible with Arduino (J2.3)	–	–
P14[2]	A2 – header compatible with Arduino (J2.5)	–	–
P14[3]	A3 – header compatible with Arduino (J2.7)	–	–
P14[4]	A4 – header compatible with Arduino (J2.9)	–	–
P14[5]	A5 – header compatible with Arduino (J2.11)	–	–
P14[6]	A6 – header compatible with Arduino (J2.13)	–	–
P14[7]	A7 – header compatible with Arduino (J2.15)	–	–
P15[1]	I2C SDA - Common I2C SDA pin for KitProg3 USB-I2C bridge	I2C SDA pin on Arduino header (J3.9)	Remove R114 to disconnect from KitProg3 I2C SDA pin.
P15[2]	I2C SCL - Common I2C SCL pin for KitProg3 USB-I2C bridge	I2C SCL pin on Arduino header (J3.10)	Remove R111 to disconnect from KitProg3 I2C SCL pin.
P16[1]	Orange user LED (LED1)	Generic GPIO header (J32.8)	Remove R104 to disconnect from the orange user LED (LED1).
P16[2]	Orange user LED (LED2)	Generic GPIO header (J32.10)	Remove R107 to disconnect from the orange user LED (LED2).

Kit details

Pin	Primary onboard function	Secondary onboard function	Connection details
P16[3]	Orange user LED (LED3)	Generic GPIO header (J32.12)	Remove R110 to disconnect from the orange user LED (LED3).
P17[3]	User button (USER_BTN2)	Generic GPIO header (J32.30)	Remove R141 to disconnect USER_BTN2.
P21[0]	WCO IN (Y2)	–	32.768-kHz watch crystal oscillator input
P21[1]	WCO OUT (Y2)	–	32.768-kHz watch crystal oscillator output
P21[2]	ECO IN (Y1)	–	16-MHz external crystal oscillator input
P21[3]	ECO OUT (Y1)	–	16-MHz external crystal oscillator output
P21[4]	User button (USER_BTN1) (Also used for hibernate wakeup function)	–	–
P21[5]	TRACEDATA [0]	Generic GPIO header (J31.22)	XMC™ MCU 20-pin ETM header J17. Remove R192 and install R314 to connect to J31.22.
P22[1]	TRACEDATA [1]	–	XMC™ MCU 20-pin ETM header J17
P22[2]	TRACEDATA [2]	–	XMC™ MCU 20-pin ETM header J17
P22[3]	TRACEDATA [3]	–	XMC™ MCU 20-pin ETM header J17
P22[4]	TRACE_CLOCK	–	XMC™ MCU 20-pin ETM header J17
P23[4]	TDO_SWO	–	XMC™ MCU 10-pin SWD/JTAG interface J16 / 20-pin ETM header J17
P23[5]	TCK_ SWCLK	–	XMC™ MCU 10-pin SWD/JTAG interface at J16 / 20-pin ETM header J17
P23[6]	TMS_ SWDIO	–	XMC™ MCU 10-pin SWD/JTAG interface at J16 / 20-pin ETM header J17
P23[7]	TDI	–	XMC™ MCU 10-pin SWD/JTAG interface at J16 / 20-pin ETM header J17
P24[0]	SMIF_INT_L_LT	–	Install R145 resistor to connect to Serial Memory Interface (SMIF) INT signal.
P24[1]	SMIF_CLK_LT	–	Install R147 resistor to connect to Serial Memory Interface (SMIF) CLK signal.
P24[2]	SDIO_CLK_LT	Serial Memory Interface SMIF_RWDS_LT signal	Connected to M.2 WL SDIO_CLK signal via U25 level translator. Remove R301 resistor and install R149 resistor to connect to Serial Memory Interface (SMIF) RWDS signal.
P24[3]	SDIO_CMD_LT	Serial Memory Interface SMIF_CS_L_LT signal	Connected to M.2 WL SDIO_CMD signal via U25 level translator.

Kit details

Pin	Primary onboard function	Secondary onboard function	Connection details
			Remove R300 resistor and install R152 resistor to connect to Serial Memory Interface (SMIF) CS signal.
P24[4]	SMIF_RST_L_LT	–	Install R154 resistor to connect to Serial Memory Interface (SMIF) RST signal.
P25[0]	SDIO_DATA0_LT	Serial Memory Interface(SMIF) DATA0 signal SMIF_DATA0_LT	Connected to M.2 WL SDIO_DATA0 signal via U25 level translator. Remove R296 and install R157 resistor to connect to (SMIF DATA0 signal.
P25[1]	SDIO_DATA1_LT	Serial Memory Interface DATA1 signal SMIF_DATA1_LT	Connected to M.2 WL SDIO_DATA1 signal via U25 level translator. Remove R297 and install R160 resistor to connect to SMIF DATA1 signal.
P25[2]	SDIO_DATA2_LT	Serial Memory Interface DATA2 signal SMIF_DATA2_LT	Connected to M.2 WL SDIO_DATA2 signal via U26 level translator. Remove R298 and install R162 resistor to connect to Serial Memory Interface (SMIF) DATA2 signal.
P25[3]	SDIO_DATA3_LT	Serial Memory Interface DATA3 signal SMIF_DATA3_LT	Connected to M.2 WL SDIO_DATA3 signal via U26 level translator. Remove R299 and install R164 resistor to connect to Serial Memory Interface (SMIF) DATA3 signal.
P25[4]	WL_HOST_WAKE_LT	Serial Memory Interface DATA4 signal SMIF_DATA4_LT	Connected to M.2 WL_HOST_WAKE signal via U28 level translator. Remove R167 resistor and install R170 to connect to Serial Memory Interface (SMIF) DATA4 signal.
P25[5]	WL_DEV_WAKE_LT	Serial Memory Interface DATA5 signal SMIF_DATA5_LT	Connected to M.2 WL_DEV_WAKE signal via U28 level translator. Remove R173 resistor and install R176 to connect to Serial Memory Interface (SMIF) DATA5 signal.
P25[6]	WL_REG_ON_LT	Serial Memory Interface DATA6 signal SMIF_DATA6_LT	Connected to M.2 WL_REG_ON_3V3 signal via U29 level translator. Remove R179 resistor and install R182 to connect to Serial Memory Interface (SMIF) DATA6 signal.
P25[7]	SMIF_DATA7_LT	–	Install R185 resistor to connect to Serial Memory Interface (SMIF) DATA7 signal.
P26[0]	ETH_REF_CLK	–	The Ethernet signals are connected to U22A Ethernet PHY transceiver in RGMII configuration mode.
P26[1]	ETH_TX_EN_CTL	–	
P26[2]	ETH_TX_CLK	–	

Kit details

Pin	Primary onboard function	Secondary onboard function	Connection details
P26[3]	ETH_TX_D0	–	
P26[4]	ETH_TX_D1	–	
P26[5]	ETH_TX_D2	–	
P26[6]	ETX_TX_D3	–	
P26[7]	ETH_RX_D0	–	
P27[0]	ETH_RX_D1	–	
P27[1]	ETH_RX_D2	–	
P27[2]	ETH_RX_D3	–	
P27[3]	ETX_RX_DV_CTL	–	
P27[4]	ETH_RX_CLK	–	
P27[5]	ETH_MDIO	–	
P27[6]	ETH_MDC	–	
P27[7]	ETH_RST	–	
P32[0]	BT_UART_RXD_LT	–	Connected to M.2 radio module connector pin BT_UART_TXD via level translator U30.
P32[1]	BT_UART_TXD_LT	–	Connected to M.2 radio module connector pin BT_UART_RXD via level translator U30.
P32[2]	BT_UART_RTS_LT	–	Connected to M.2 radio module connector pin BT_UART_CTS via level translator U30.
P32[3]	BT_UART_CTS_LT	–	Connected to M.2 radio module connector pin BT_UART_RTS via level translator U30.
P32[4]	BT_HOST_WAKE_LT	–	Connected to M.2 radio module connector pin BT_HOST_WAKE_3V3 via level translator U28.
P32[5]	BT_DEV_WAKE_LT	–	Connected to M.2 radio module connector pin BT_DEV_WAKE via level translator U28.
P32[6]	BT_REG_ON_LT	–	Connected to M.2 radio module connector pin BT_REG_ON_3V3 via level translator U29.
P32[7]	BT_DEV_WAKE_LAI	–	Not connected. Populate R303 when using BT_DEV_WAKE signal on Laird M.2 radio module.
M.2 radio module pins			
BT_UART_TXD	UART interface with the host MCU (XMC7200D)	UART interface with KitProg3	Remove R265 to disconnect from XMC7200D UART Rx. Remove R69 to disconnect WL_UART_TX.

Kit details

Pin	Primary onboard function	Secondary onboard function	Connection details
			Populate R66 to connect to KitProg3 Secondary UART Rx.
BT_UART_RXD	UART interface with the host MCU (XMC7200D)	UART interface with KitProg3	Remove R266 to disconnect from XMC7200D UART Tx. Remove R65 to disconnect WL_UART_RX. Populate R64 to connect to KitProg3 Secondary UART Tx.
BT_UART_CTS	UART interface with the host MCU (XMC7200D)	UART interface with KitProg3	Remove R267 to disconnect from XMC7200D UART RTS. Populate R71 to connect to KitProg3 Secondary UART RTS.
BT_UART_RTS	UART interface with the host MCU (XMC7200D)	UART interface with KitProg3	Remove R268 to disconnect from XMC7200D UART CTS. Populate R73 to connect to KitProg3 Secondary UART CTS.
BT_I2S_SCK	I2S serial clock	–	–
BT_I2S_WS	I2S serial word select	–	–
BT_I2S_OUT	I2S serial data out	–	–
BT_I2S_IN	I2S serial data in	–	–
BT_IO_2	Bluetooth® GPIO	–	–
BT_IO_3	Bluetooth® GPIO	–	–
BT_IO_4	Bluetooth® GPIO	–	–
BT_IO_5	Bluetooth® GPIO	–	–
BT_IO_6	Bluetooth® GPIO	–	–
BT_IO_7	Bluetooth® GPIO	–	–
WL_UART_RX	Wi-Fi debug UART Rx pin	–	–
WL_UART_TX	Wi-Fi debug UART Tx Pin	–	–
WL_GPIO11	Programable GPIO Pin	–	–
WL_REG_ON _ALT	Alternate WL Regulator ON pin	–	–
M2_SCL	I2C SCL pin	–	–
M2_SDA	I2C SDA pin	–	–

3 Kit operation

3.1 Theory of operation

The XMC7200 evaluation board is built around XMC7200D. For more information on device features, see the device **datasheet**.

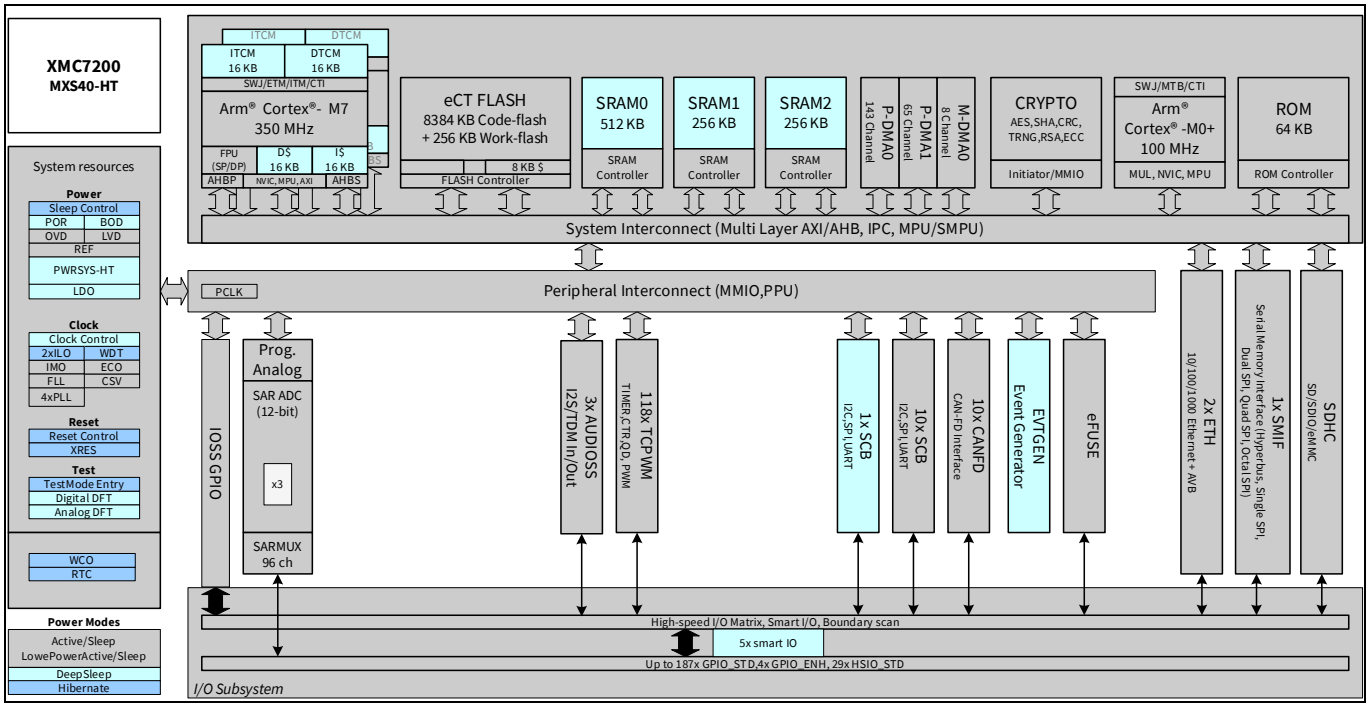


Figure 4 XMC7200D block diagram

Kit operation

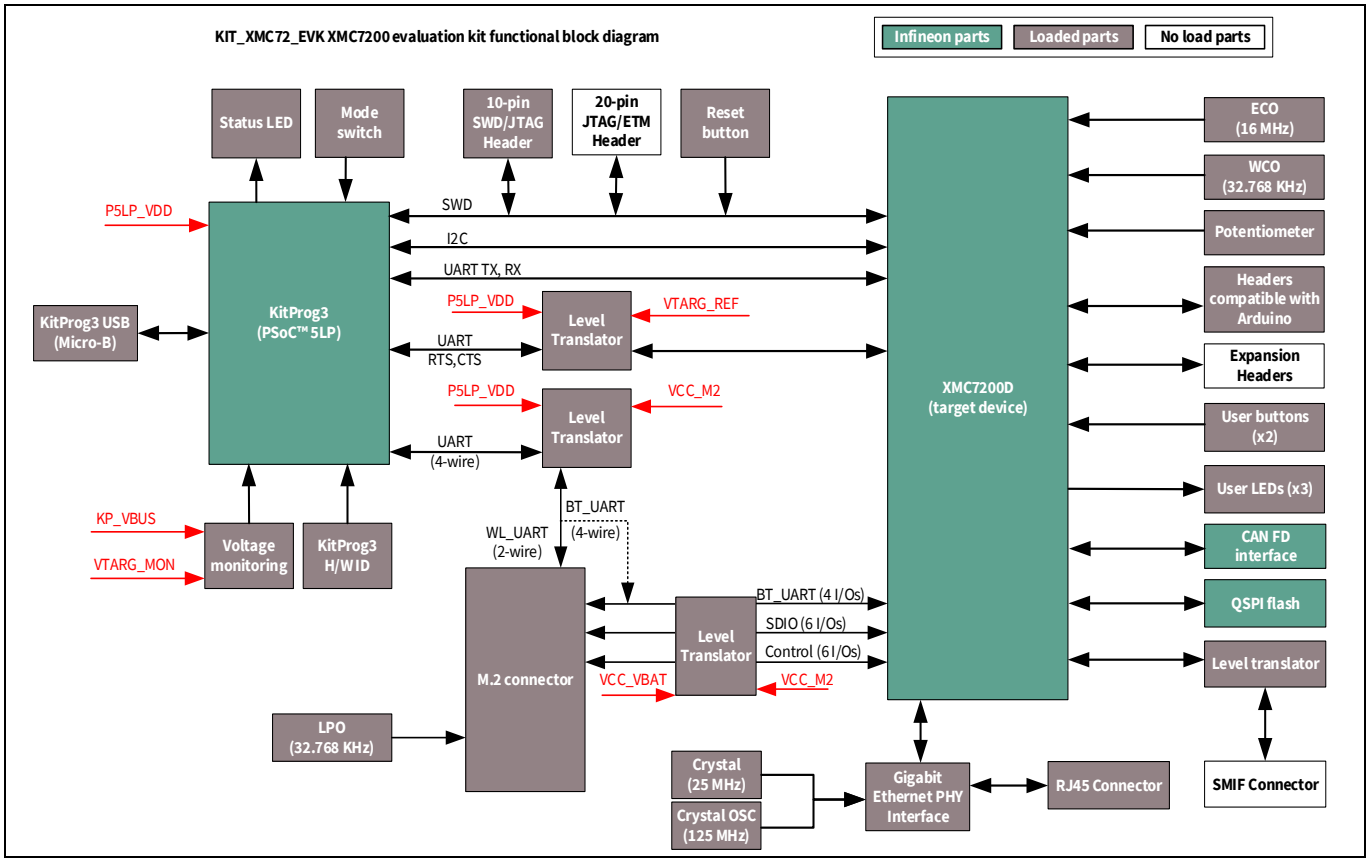


Figure 5 Block diagram of XMC7200 evaluation board

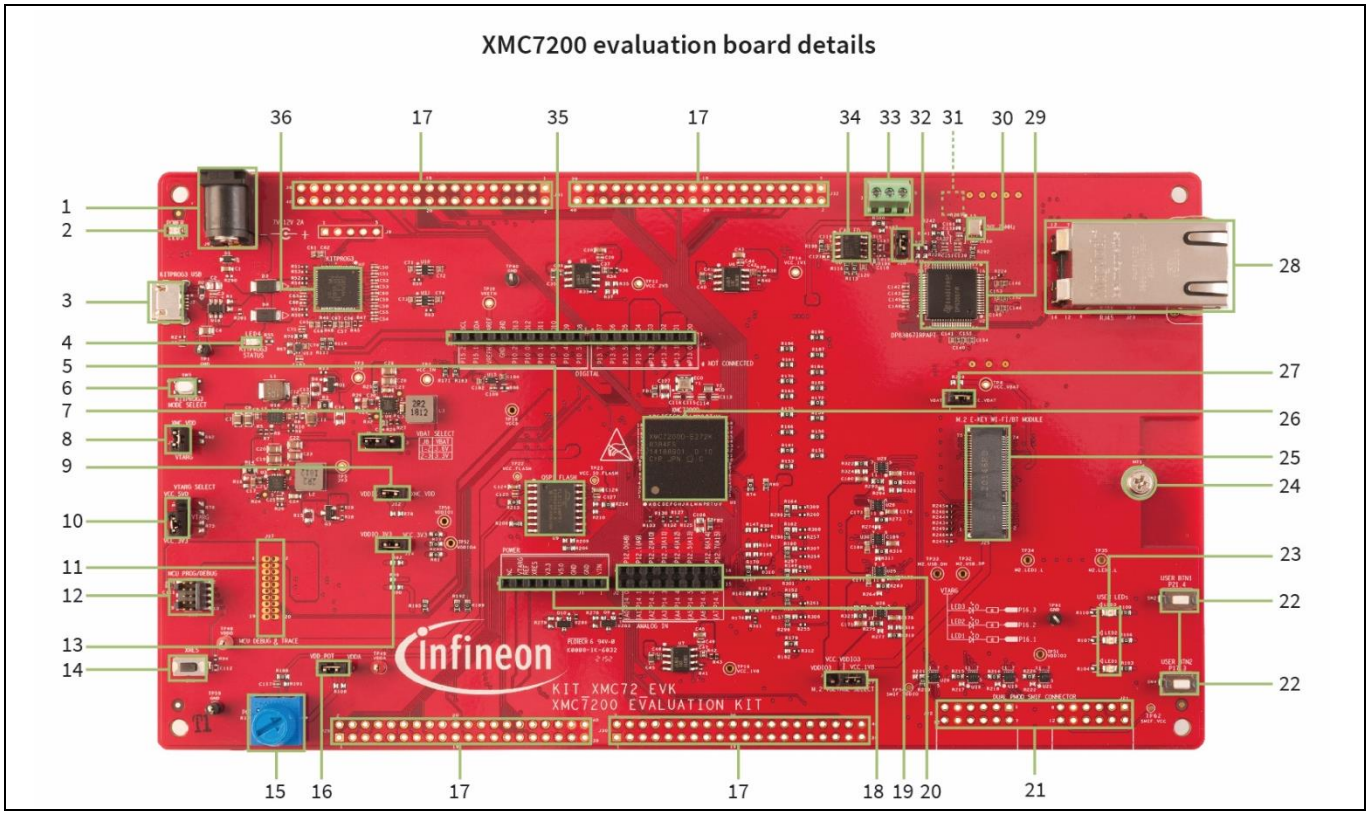


Figure 6 XMC7200 evaluation board – top view

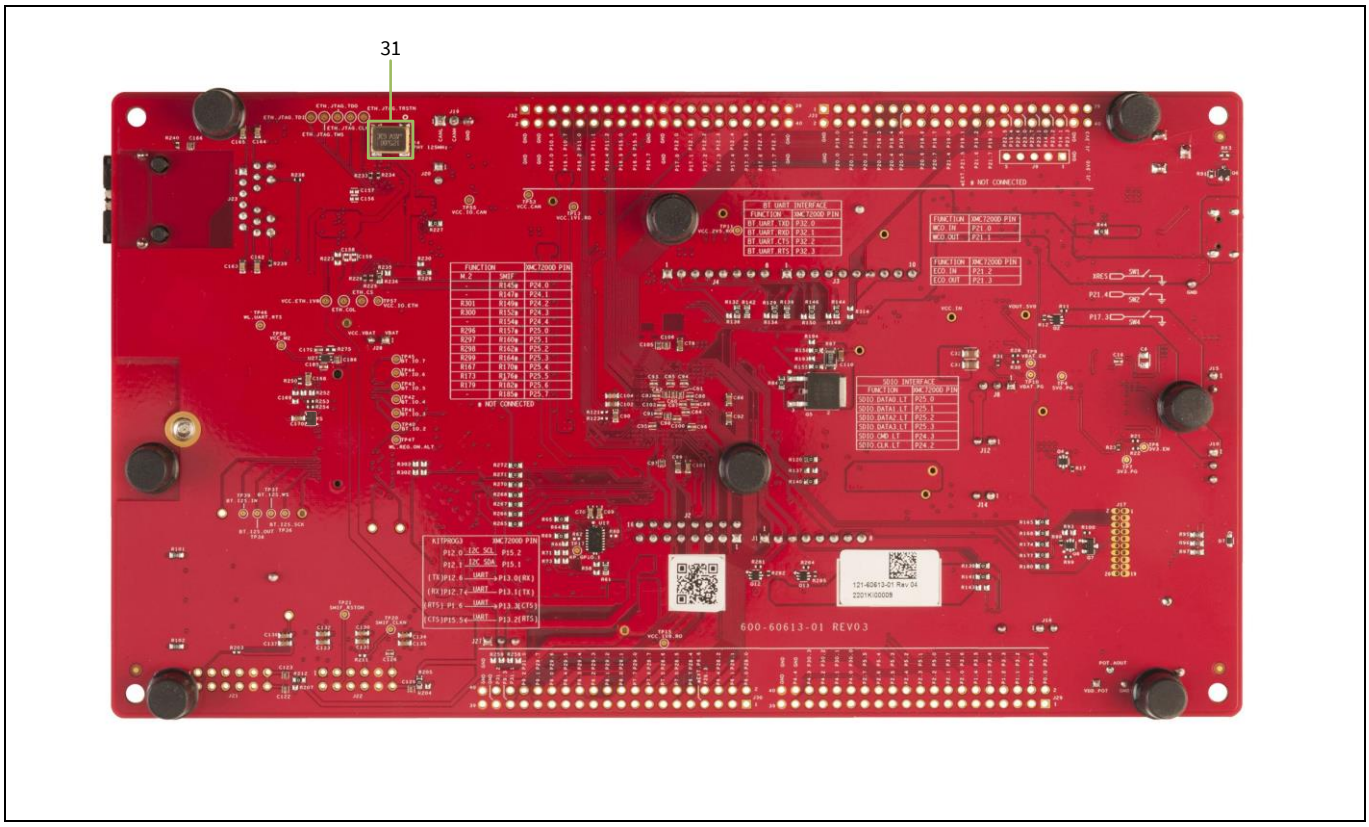


Figure 7 XMC7200 evaluation board – bottom view

Kit operation

The XMC7200 evaluation board has the following peripherals.

Table 5 Peripheral details

Sl. No.	Peripheral	Description
1.	External power supply VIN connector (J6)	Connect to an external 12V/3A DC power adapter. For information about the DC adapter (SMI36-12-V-P5 from CUI, Inc.) included with the kit, see the manufacturer's **website**.
2.	Power LED (LED5)	Power supply ON/OFF LED status indicator.
3.	KitProg3 USB connector (J7)	Connect to a PC to use the KitProg3 onboard programmer and debugger and to provide power to the board.
4.	KitProg3 status LED (LED4)	Indicates the status of KitProg3. For details, see the **KitProg3 user guide**.
5.	512-Mbit serial NOR flash memory (S25FL512S - U9)	Connected to the serial memory interface (SMIF) of the XMC7200D. The NOR flash device can be used for both data and code memory with execute-in-place (XIP) support and encryption.
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.	VBAT power selection jumper (J8)	Use this jumper to select the VBAT supply for the module is provided from the XMC7200 evaluation kit, which is 3.3 V/3.6 V selectable depending on the M.2 radio module being used.
8.	XMC7200D VTARG current measurement jumper (J15)	Connect an ammeter to this jumper to measure the current consumed by the XMC_VDD power domain of the MCU.
9.	XMC7200D VDDIO2 current measurement jumper (J12)	J12 connects MCU VDDIO2 to XMC_VDD which is outcome from J10 (VCC_3V3 or VCC_5V0) power supply selection.
10.	System power (VTARG) selection jumper (J10)	J10 is a three-pin voltage selection header to select between VCC_3V3 or VCC_5V0 voltages.
11.	XMC7200D 20-pin debug and trace header (J17)	Connect to an Embedded Trace Macrocell (ETM)-compatible programmer/debugger. This is not loaded by default.
12.	XMC7200D 10-pin SWD/JTAG program and debug header (J16)	This 10-pin header allows you to program and debug the XMC7200D using an external programmer such as MiniProg4.
13.	XMC7200D target I/O current measurement jumper (J14)	Connect an ammeter to this jumper to measure the current consumed by the XMC™ MCU I/O (VDDIO1, VDDIO3, and VDDIO4).
14.	XMC7200D reset button (SW1)	Resets XMC7200D. It connects the XMC7200D reset (XRES) pin to ground.

Kit operation

Sl. No.	Peripheral	Description
15.	Potentiometer (R105)	10-kΩ potentiometer connected to XMC7200D pin P6[6]. It can be used to simulate a sensor output to XMC7200D.
16.	Potentiometer connection jumper (J18)	Connects the XMC7200D VDDA supply to the potentiometer.
17.	XMC7200D extended I/O headers (J29, J30, J31, J32)	These headers provide connectivity to XMC7200D GPIOs that are not connected to the headers compatible with Arduino. Few of these pins are multiplexed with onboard peripherals and are not connected to the XMC7200D by default. These connectors are not populated by default.
18.	M.2 I/O power selection jumper (J27)	Use this jumper to select the M.2 I/O supply for the module is provided from the XMC7200 evaluation kit, which is 1.8 V/3.3 V selectable.
19.	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.
20.	Analog-IN header compatible with Arduino Uno R3 (J2)	Bring out pins from XMC7200D to interface with shields compatible with Arduino. Some of these pins are multiplexed with onboard peripherals and are not connected to XMC7200D by default. For detailed information on how to rework the kit to access these pins, see **Table 4**.
21.	SMIF dual header compatible with Digilent Pmod (J21, J22)	The connectors designed as per SMIF dual header compatible with Digilent Pmod to validate HYPERFLASH™ memory interface. These connectors are not populated by default.
22.	XMC7200D user buttons (SW2, SW4)	Provide an input to XMC7200D. Note that by default the button connects the XMC7200D pin to ground when pressed, so you need to configure the XMC7200D pin as a digital input with resistive pull-up for detecting the button press. SW2 button also provide a wakeup source from hibernate mode of the device.
23.	XMC7200D user LEDs (LED1, LED2, LED3)	The user LEDs can operate at the entire operating voltage range of the XMC7200D. The LEDs are active LOW, so the pins must be driven to ground to turn ON the LEDs.
24.	M.2 stand-off (MT1)	Stand-off interface to M.2 module
25.	M.2 interface connector (J25)	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.
26.	XMC7200D Microcontroller (XMC7200D-E272K8384 – U1)	XMC™ MCU device used on the kit.
27.	VBAT current measurement jumper (J28)	Connect an ammeter to this jumper to measure the current consumed by M.2 module.
28.	RJ45 Gigabit Ethernet connector (J23)	RJ45 ethernet connector port to connect kit to the Ethernet network.

Kit operation

Sl. No.	Peripheral	Description
29.	Ethernet physical layer (PHY) transceiver (U22)	Ethernet PHY
30.	25-MHz crystal for Ethernet transceiver (Y3)	Oscillator for Ethernet PHY
31.	125-MHz crystal for Ethernet transceiver (Y4)	Oscillator for Ethernet PHY
32.	CAN FD resistor termination jumper (J20)	Install a jumper to enable the termination resistor for another CAN FD transceiver IFX1051SJ in place of U8. Along with the jumper, C187 and R199 components must be installed and two resistors R196 and R315 removed.
33.	CAN FD interface connector (J19)	Connector to connect kit to the CAN / CAN FD network.
34.	CAN FD transceiver (TLE9251VSJ - U8)	CAN FD transceiver
35.	Digital I/O headers compatible with Arduino Uno R3 (J3, J4)	Bring out pins from the XMC7200D to interface with shields compatible with Arduino. Some of these pins are multiplexed with onboard peripherals and are not connected to the XMC7200D by default. For detailed information on how to rework the kit to access these pins, see **Table 4**.
36.	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**.

See “**Hardware functional description**” on page **Error! Bookmark not defined.** for details on various hardware blocks.

3.2 M.2 wireless connectivity module

Note: Supporting software and the code examples will be available in future.

The XMC7200 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 XMC7000 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 requires an external antenna.

This kit has been verified for Wi-Fi functionality with the following M.2 modules from Embedded Artits:

- 1YN M.2 module (with CYW43439 chipset)
- 1LV M.2 module (with CYW43012 chipset)

Note: Currently, support for the Bluetooth® functionality of these modules is not verified.

Note: WL Host Wake and Bluetooth® Host Wake functionality are not available in this kit.

3.3 Board support package (BSP) selection

The XMC7200 evaluation kit comes with a KIT_XMC72_EVK board support package to build and run code examples on the kit by following the instructions given in the **Board Support Packages** section of the **ModusToolbox™ user guide**.

3.4 KitProg3: onboard programmer/debugger

The XMC7200 evaluation board can be programmed and debugged using the onboard KitProg3. KitProg3 is an onboard 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. The PSoC™ 5LP device is used to implement the KitProg3 functionality. For more details on the KitProg3 functionality, see the **KitProg3 user guide**.

3.4.1 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 8**. 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.

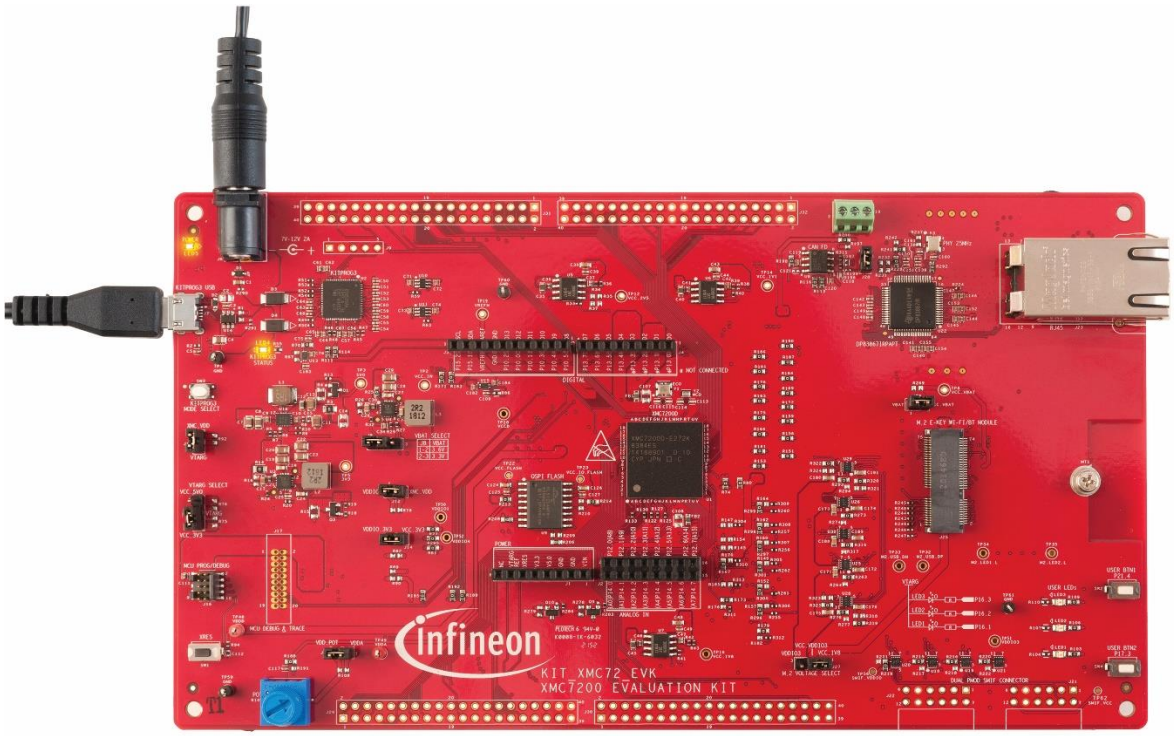


Figure 8 Connect USB cable to USB connector on the board

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**.

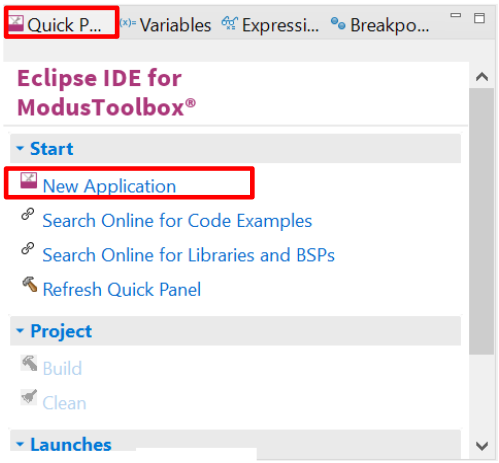


Figure 9 Create new application

Kit operation

b) Select the BSP in the “Choose Board Support Package” window and click **Next**.

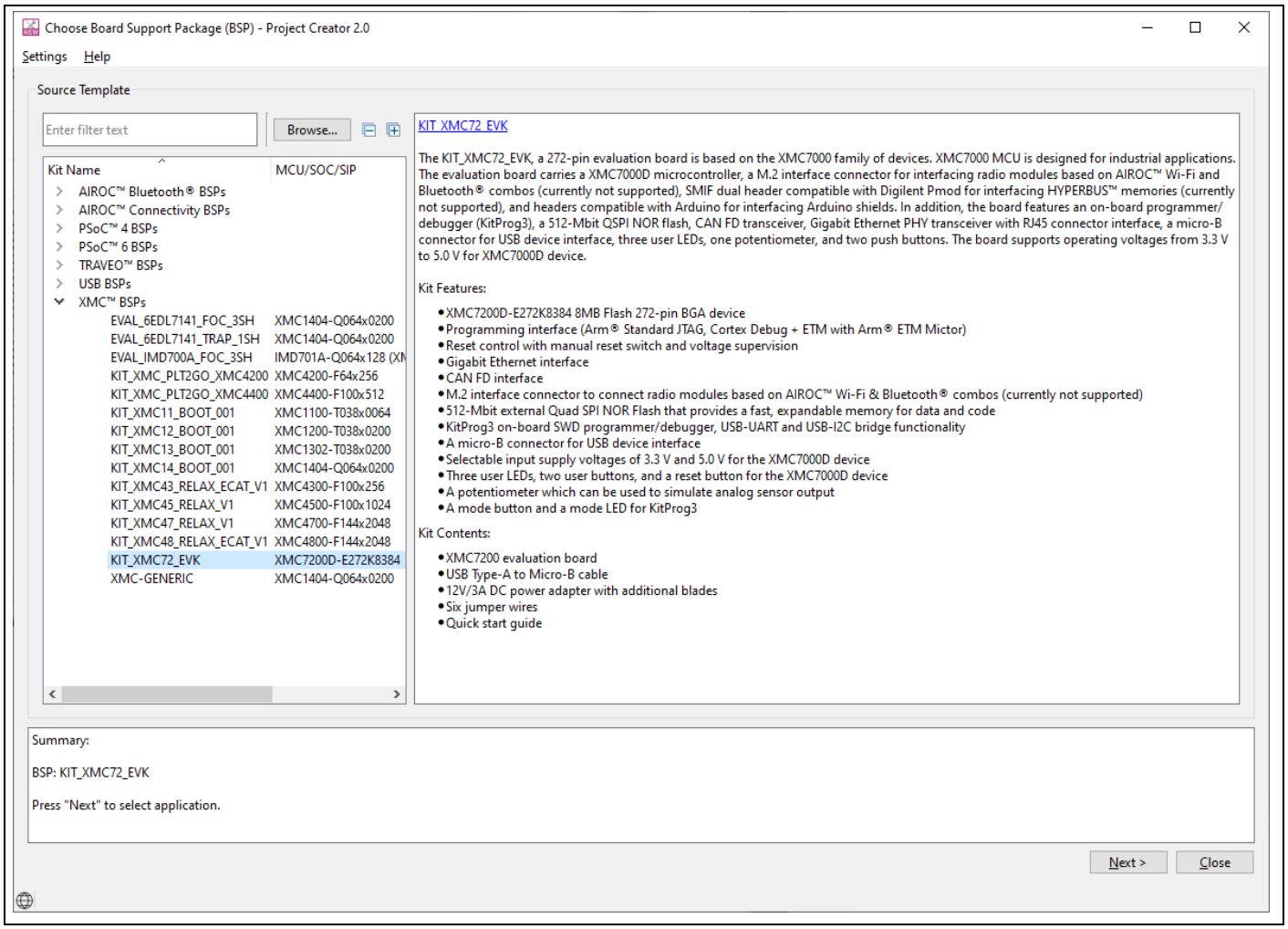


Figure 10 Board Support Package in the new application window

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

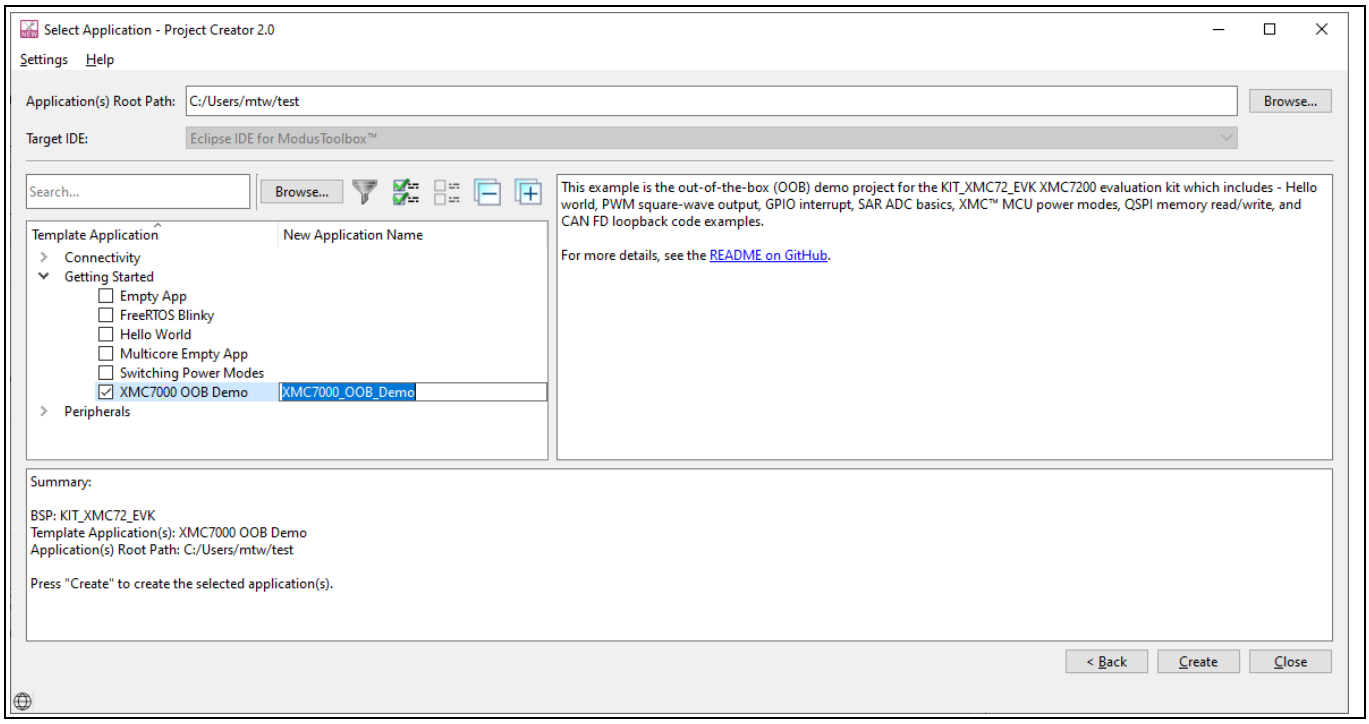


Figure 11 Select application window

4. To build and program a XMC7200D 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 12**.

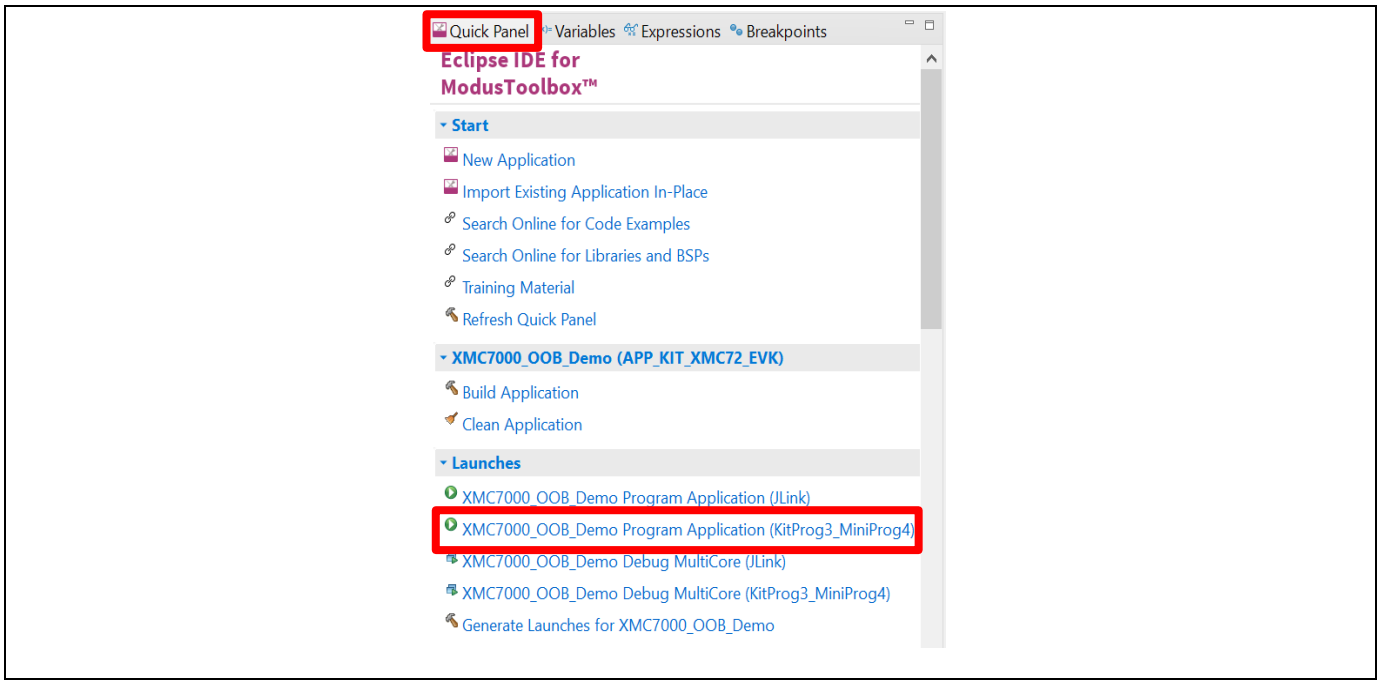


Figure 12 Programming in the ModusToolbox™ software

5. ModusToolbox™ software has an integrated debugger. To debug a XMC7200D 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 13**. For more details, see the "Program and debug" section in the **Eclipse IDE for ModusToolbox™ user guide**.

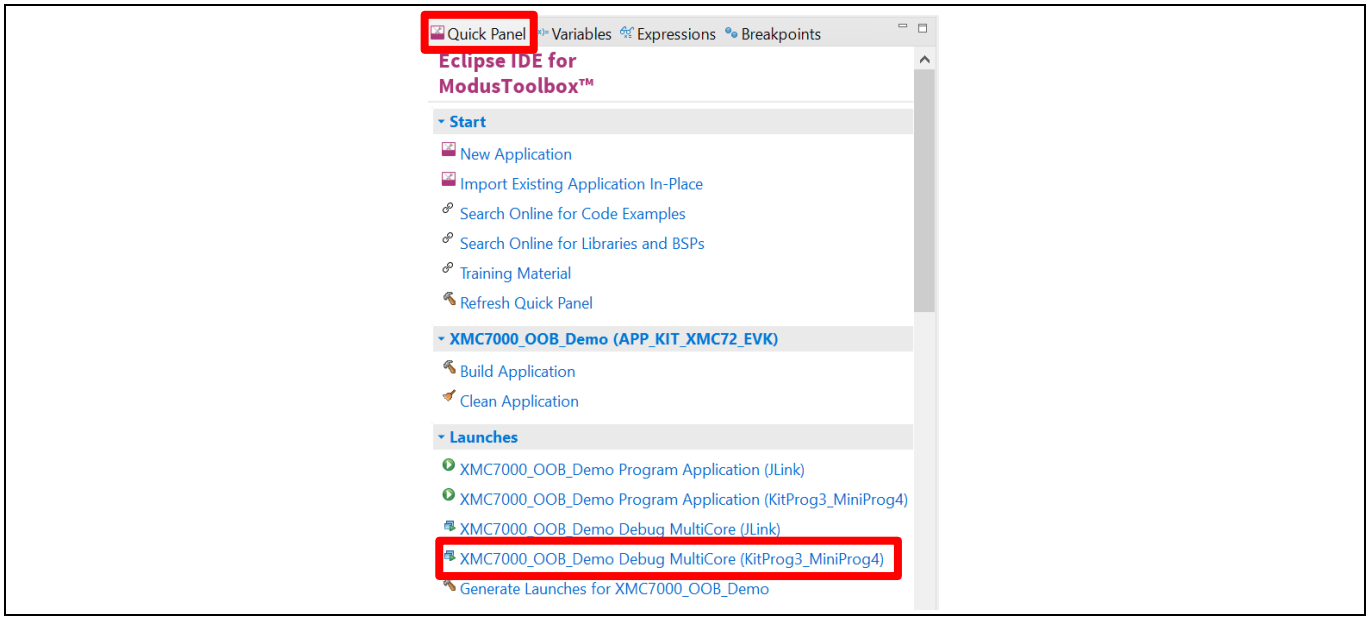


Figure 13 Debugging in the ModusToolbox™ software

3.4.1.1 Using the OOB example – XMC7000 MCU: OOB demo

The XMC7200 evaluation board is by default programmed with the code example: *XMC7000 MCU: OOB demo*. The steps explain how to use the example. For a detailed description of the project, see the example's README.md file in the GitHub repository. The README.md file is in the application directory once the application is created.

Note: At any point of time, if you overwrite the OOB demo example, you can restore it by programming the XMC7000 MCU: OOB demo.

6. Connect the board to the PC using the provided USB cable through the KitProg3 USB connector.
7. Open a terminal program and select the KitProg3 COM port. Set the serial port parameters to 8N1 and 115200 baud.
8. Press the reset button (SW1) on the board,andconfirm that terminal application displays the code example title and other text as shown in **Figure 14**.

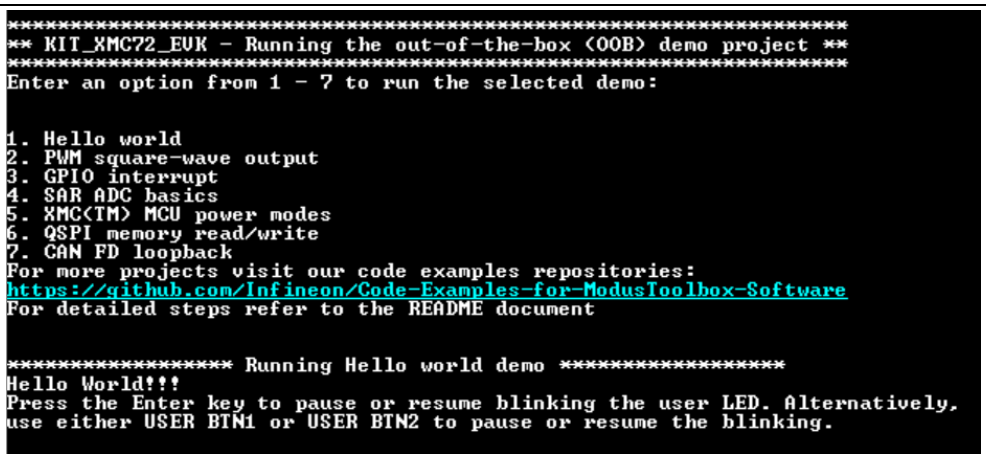


Figure 14 OOB demo project message in the terminal

9. Confirm that the kit LEDs blinks at 1 Hz in a sequence.
10. If the **Enter** keyis pressed, the LEDs stops blinking,and the terminal displays the message "LED blinking paused".
11. If the **Enter** key is pressed again, the LEDs resumes blinking at 1 Hz and the message displayed on the terminal is updated to "LED blinking resumed". Alternatively, **USER BTN1** or **USER BTN2** can be used to pause or resume the blinking.
12. Follow instructions on the serial terminal to run other demos.

3.4.2 USB-UART bridge

KitProg3 on the XMC7200 evaluation board can act as a USB-UART bridge.

The UART RX and TX pins of KitProg3 are connected to the XMC7200D UART pins as follows:

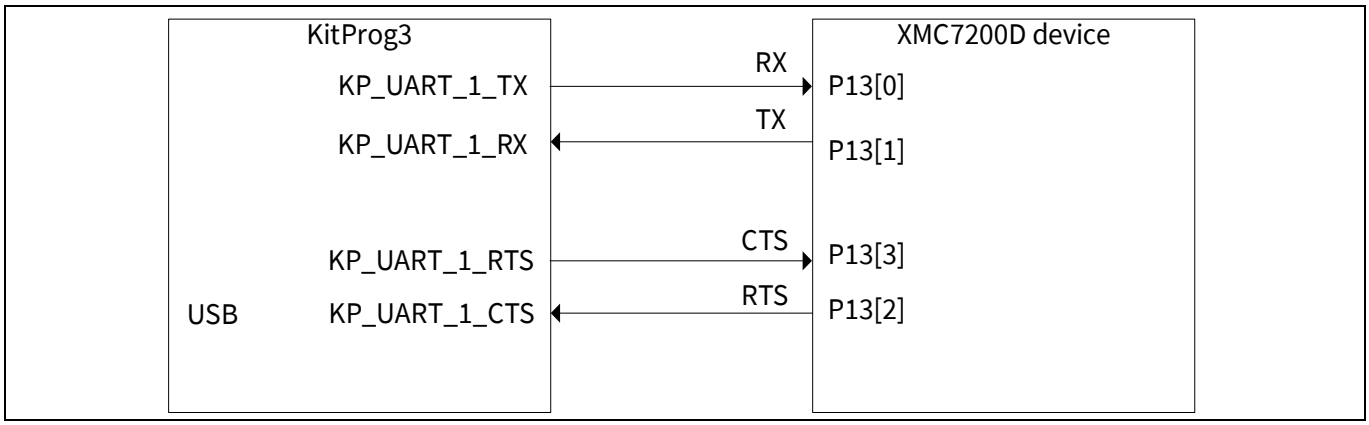


Figure 15 UART connection between KitProg3 and XMC7200D

Figure 16 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 **4.2.5 Serial interconnection between PSoC™ 5LP and M.2 interface** to see how to re-work the board to connect KitProg3 USB-UART bridge with AIROC™ Wi-Fi & Bluetooth® combo-based M.2 module.

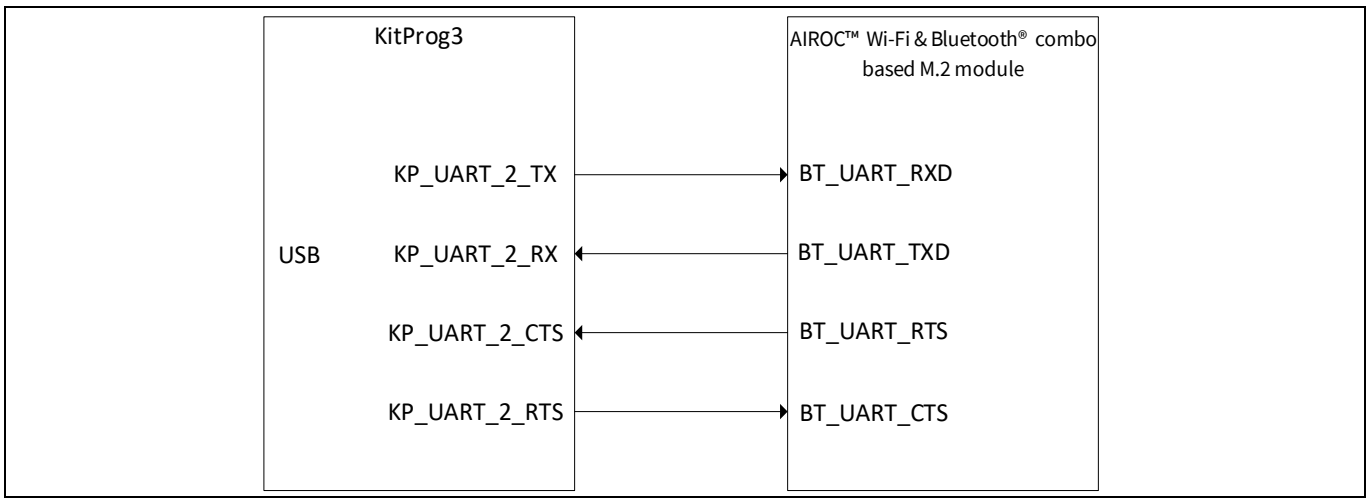


Figure 16 UART connection between KitProg3 and AIROC™ Wi-Fi & Bluetooth® combo-based M.2 module

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

3.4.3 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 XMC7200D are hard-wired on the board to the I2C lines of the KitProg3, with onboard 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**.

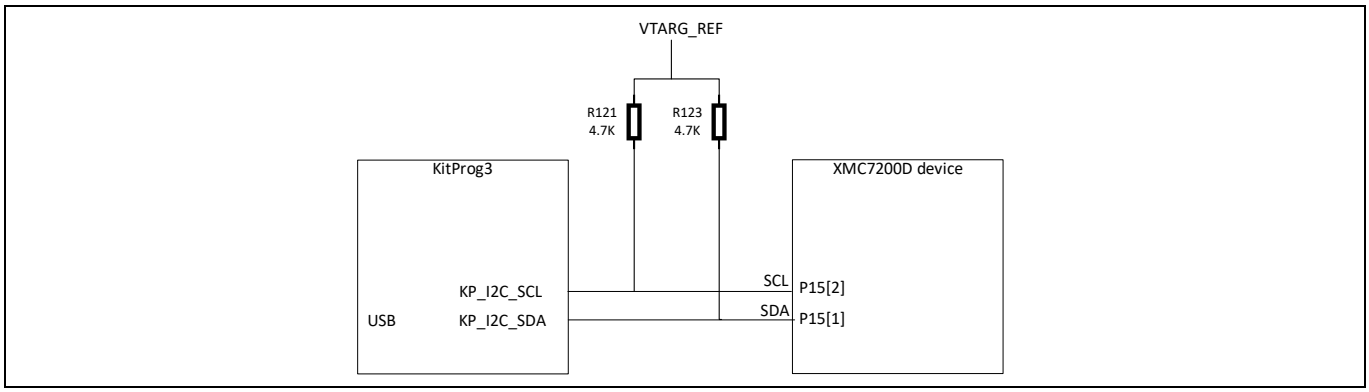


Figure 17 I2C connection between KitProg3 and XMC7200D

4 Hardware

4.1 Schematics

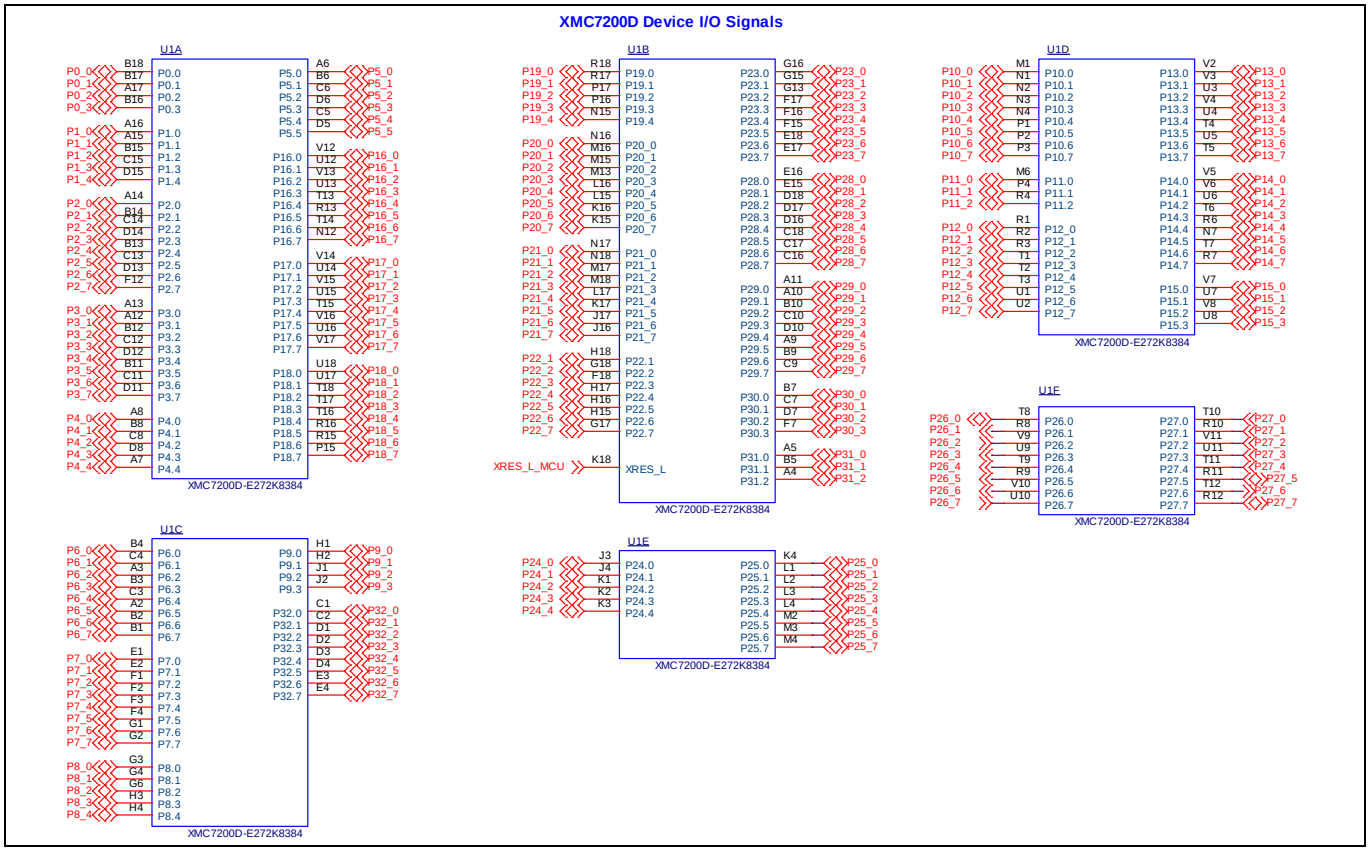
See the schematic files available in the **kit webpage**.

4.2 Hardware functional description

4.2.1 XMC7200D (U1)

XMC7200 is a family of XMC7000 MCUs with industrial applications. XMC7200 has two Arm® Cortex®-M7 CPUs for primary processing, and an Arm® Cortex®-M0+ CPU for peripheral and security processing. These devices contain embedded peripherals supporting CAN FD and Gigabit Ethernet. XMC7200 devices are manufactured on an advanced 40-nm process. XMC7200 incorporates Infineon's low-power flash memory, multiple high-performance analog and digital peripherals, and enables the creation of a secured computing platform.

For more information, see the **XMC7200D datasheet**.



Hardware

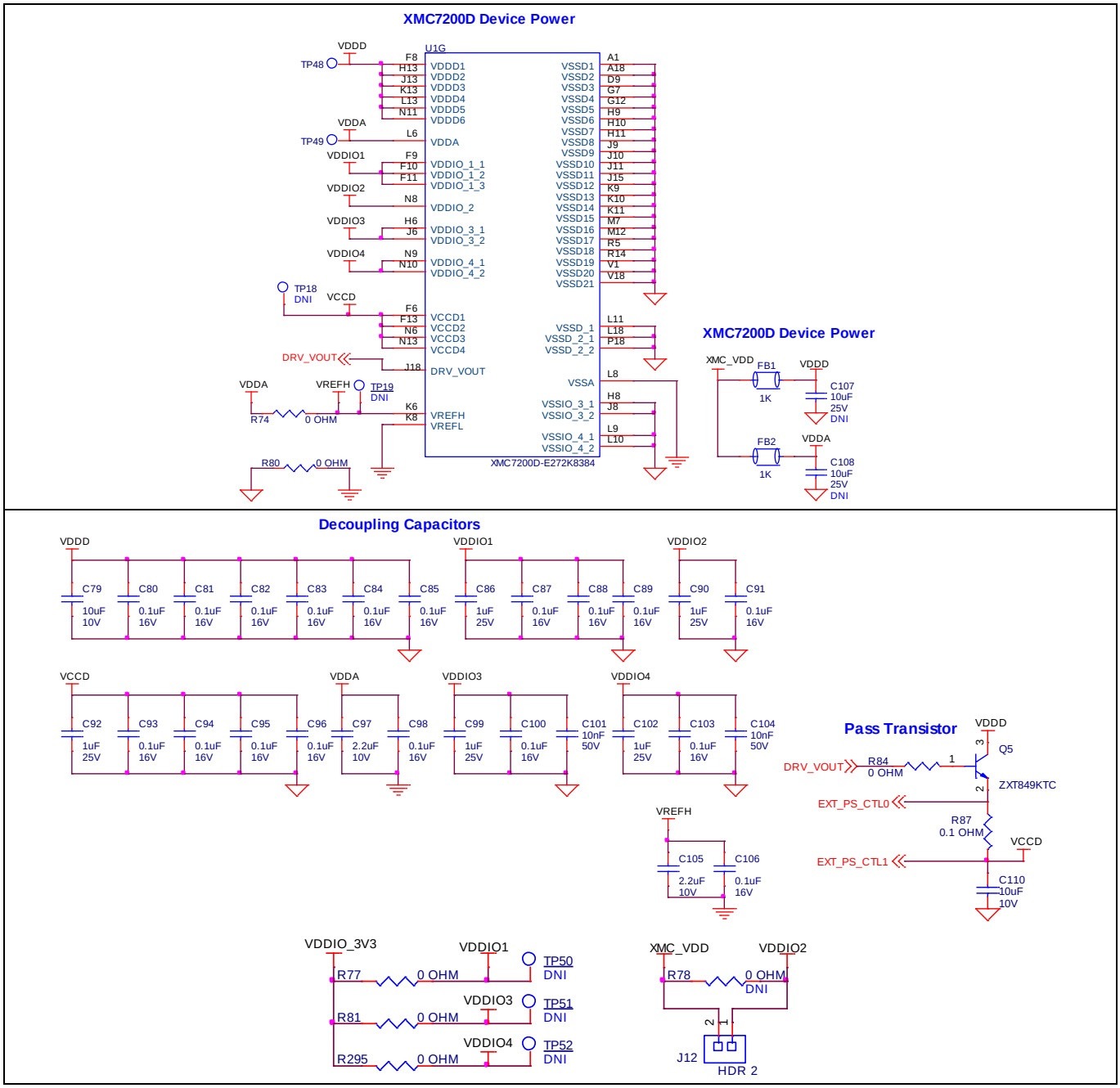


Figure 18 XMC7200D

4.2.2 M.2 interface connector (J25)

The M.2 E-key connector adds radio connectivity to XMC7200D as required. The M.2 E-key radio module, compatible with Infineon M.2 connector pin mapping, can be interfaced to the XMC7200 evaluation kit using SDIO for WLAN and UART for Bluetooth®. The VBAT supply for the module is provided from the XMC7200 evaluation kit (3.3 V/3.6 V, selectable based on the M.2 radio module). Refer to the appropriate M.2 radio module datasheet for the valid operating voltage.

Suitable level translators have been added in between the XMC7200D device and the M.2 connector to match the module I/O voltage level between VBAT and VCC_M2.

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+ do not follow the Infineon M.2 interface standard. For such modules, use the resistor 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	Laird M.2 connector pin (populate R303)
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.

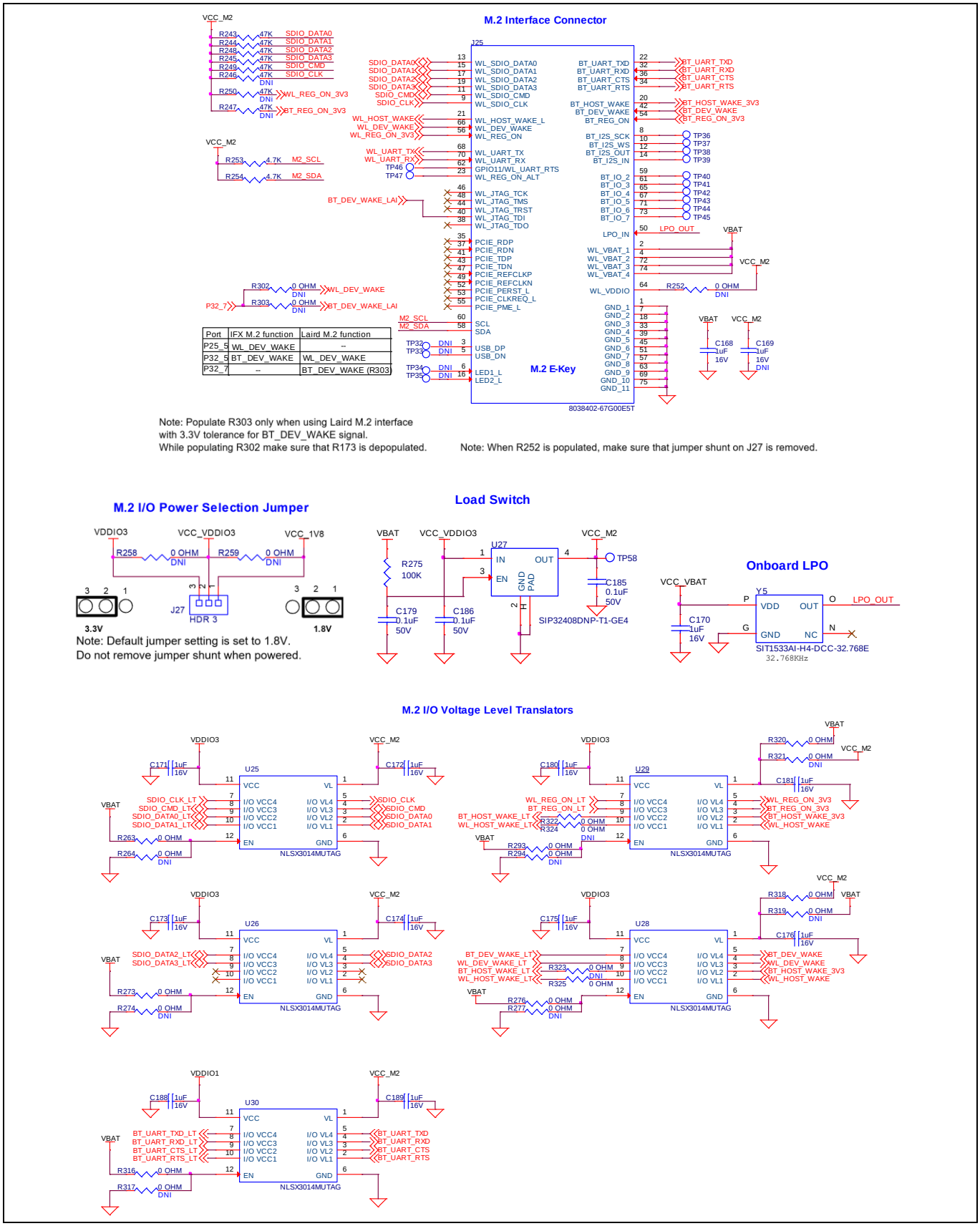


Figure 19 M.2 interface connector

4.2.3 PSoC™ 5LP-based KitProg3 (U2)

An onboard PSoC™ 5LP (CY8C5868LTI-LP039) is used as KitProg3 to program and debug XMC7200D. 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 the XMC7200D.

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

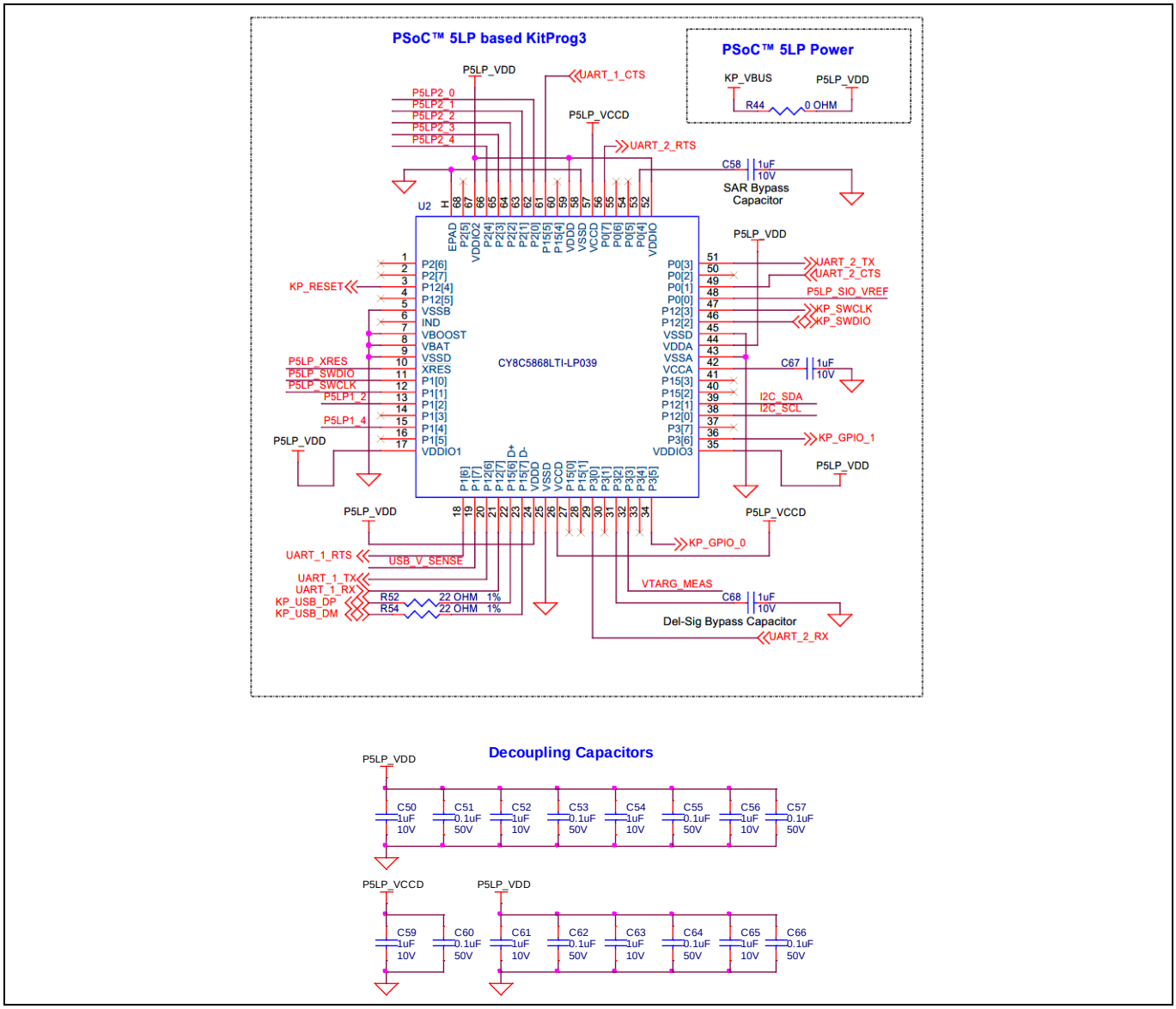


Figure 20 PSoC™ 5LP-based KitProg3 (U2)

4.2.4 Serial interconnection between PSoC™ 5LP and XMC7200D

The PSoC™ 5LP device functions as an interface for USB-UART and USB-I2C bridges in addition to be used as an onboard programmer , as shown in **Figure 21**. The USB-serial pins of the PSoC™ 5LP device are hard-wired to the I2C/UART pins of the XMC7200D device. These pins are also available on the I/O headers compatible with Arduino Uno R3.

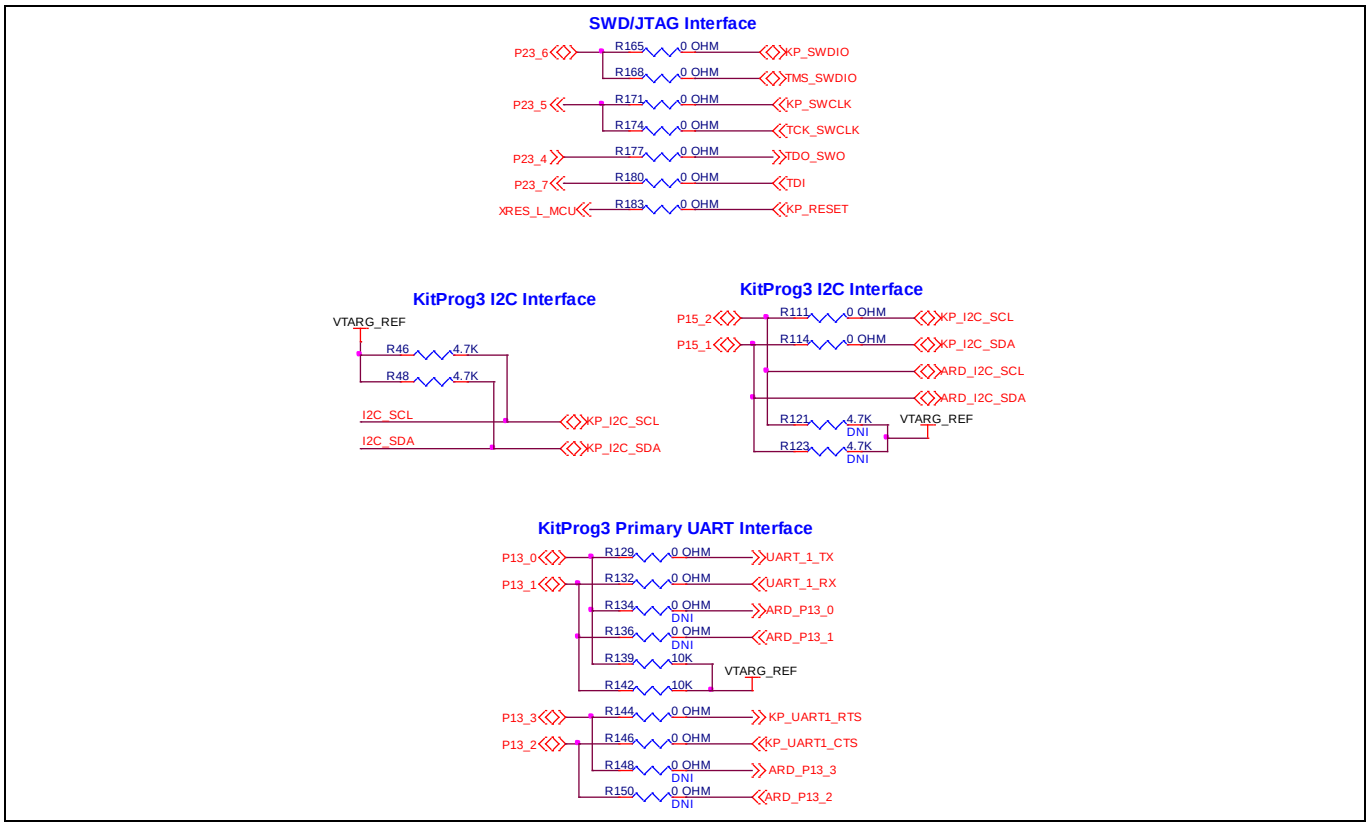


Figure 21 Programming and serial interface connections

4.2.5 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 the M.2 interface.

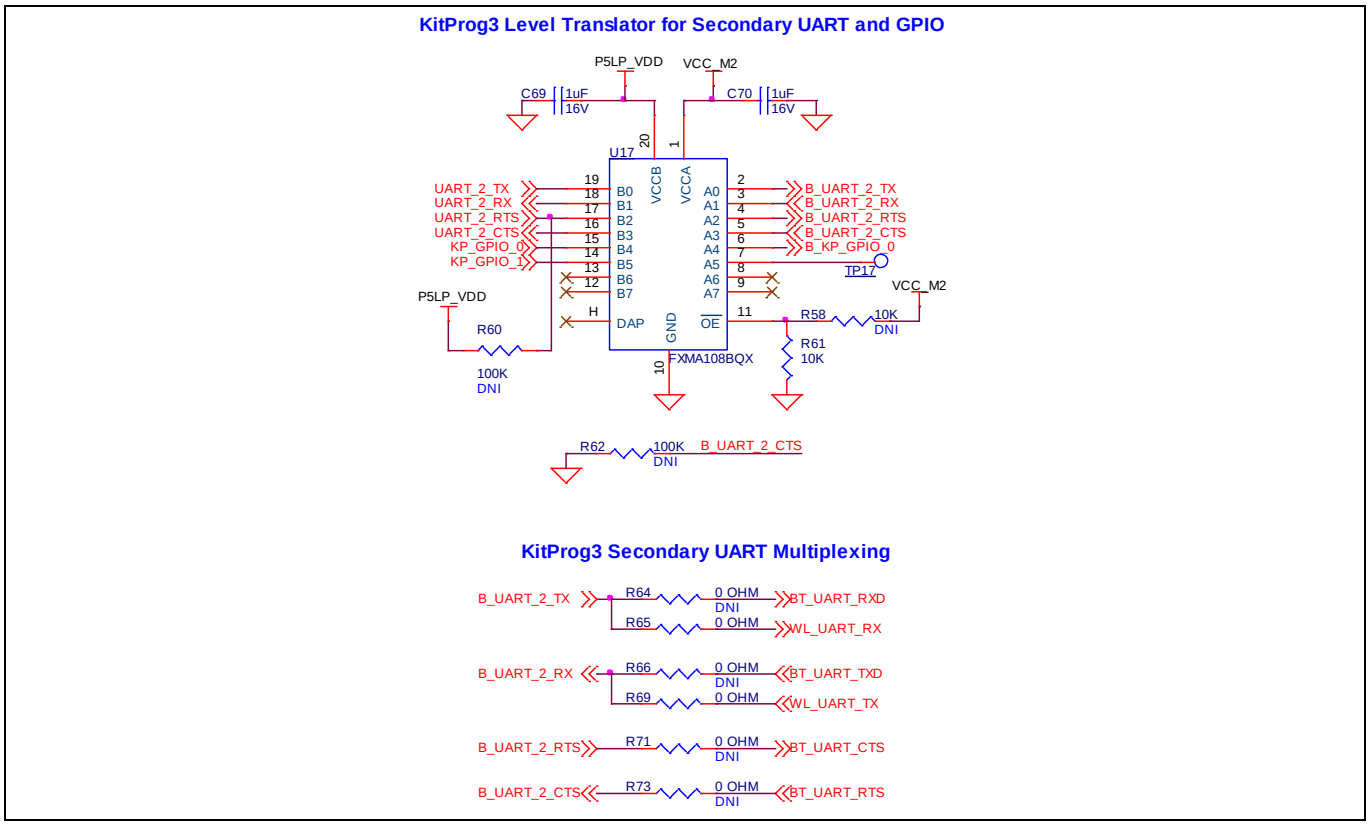


Figure 22 Serial interconnection between PSoC™ 5LP and M.2 interface

4.2.6 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 onboard USB Micro-B connectors **(J7)**
- 7 V–12 V from the external power supply through the VIN barrel jack (**J6**) or from a shield compatible with Arduino

Note: Ensure to power the board with external DC supply via VIN connector for the applications which require more than 400 mA current.

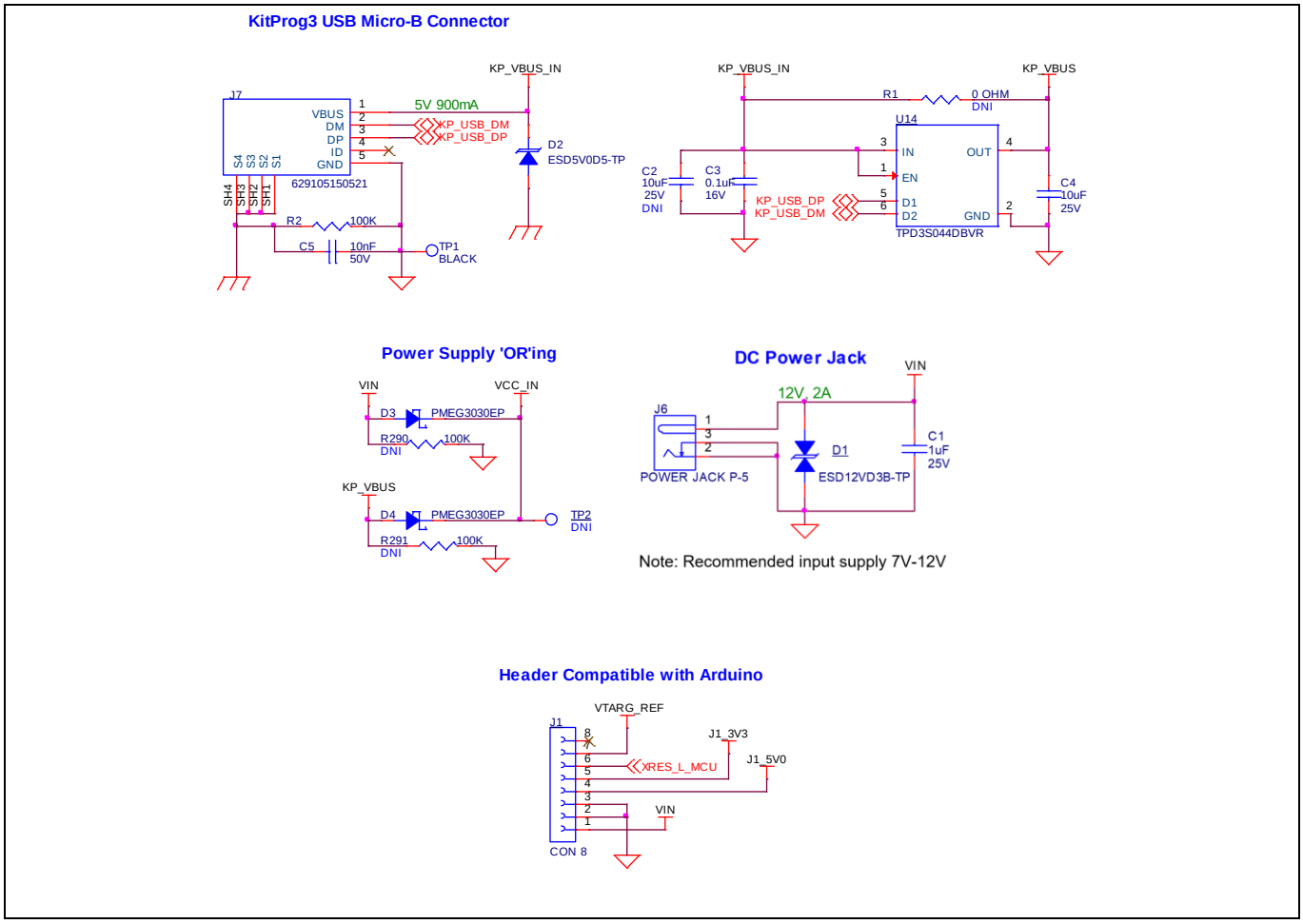


Figure 23 Power supply input and ORing

4.2.6.1 Voltage regulators

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

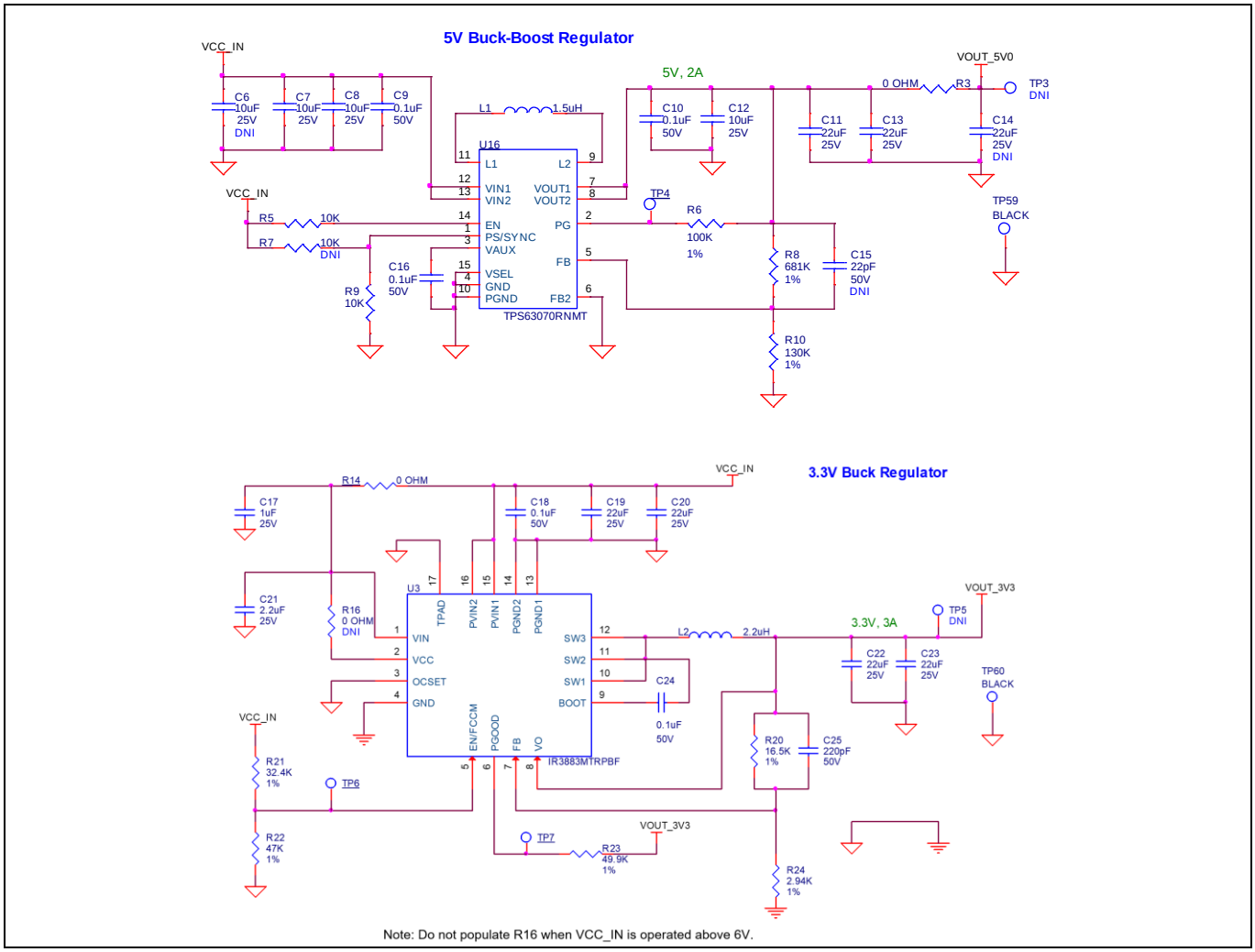
- VBAT, VCC_M2 (M.2 Interface) – 3.6 V, 3.3 V and 1.8 V
- VDDIO1, VDDIO3 and VDDIO4 (XMC7200D) – 3.3 V
- VDDD, VDDA, VDDIO2, VREFH (XMC7200D) – 3.3 V and 5 V
- VCC_1V1, VCC_1V8, VCC_2V5 (Ethernet PHY) – 1.1 V, 1.8 V and 2.5 V

Hardware

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

VCC_M2 must be at the same voltage as the M.2 radio module I/O voltage because it powers the SDIO and UART interfaces between XMC7200D and the radio. If VCC_M2 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.

A buck-boost regulator (U16) is used on the kit to generate a stable 5-V output. There are two buck regulators U3 and U4 that are used to generate 3.3 V and 3.6 V. Three LDOs (U5, U6, U7) are designed to generate 2.5 V, 1.1 V, and 1.8 V outputs respectively.



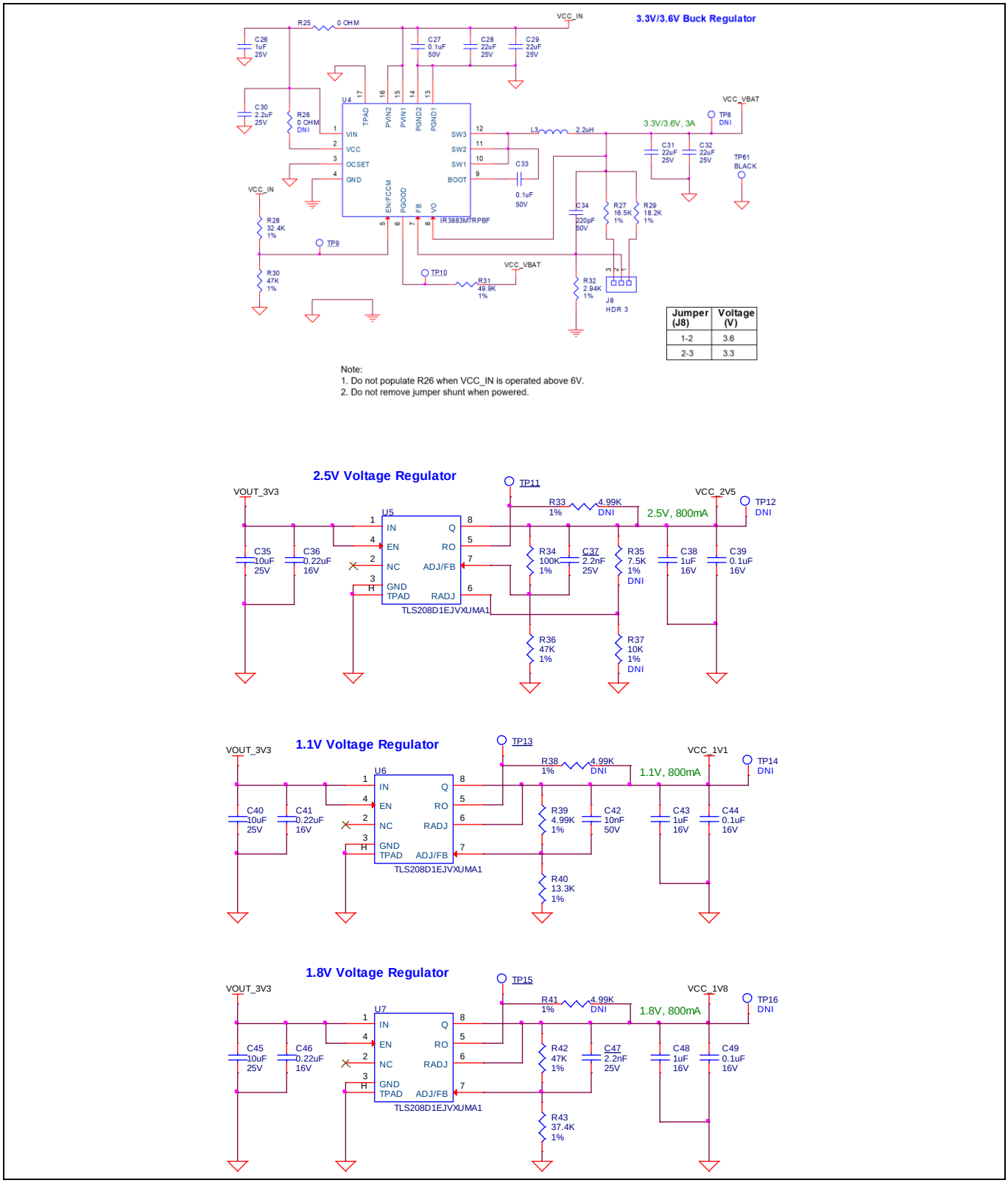


Figure 24 Voltage regulators

4.2.6.2 Voltage selection

VBAT has a dedicated regulator that changes voltage by varying the feedback voltage through the resistor network at **J8**. VTARG has a 3-pin voltage selection header **J10** to select between VCC_3V3 or VCC_5V0 voltages. VDDIO2 has a dedicated 2-pin voltage selection header **J12** that connects to XMC_VDD which is an outcome from J10 (VCC_3V3 or VCC_5V0).

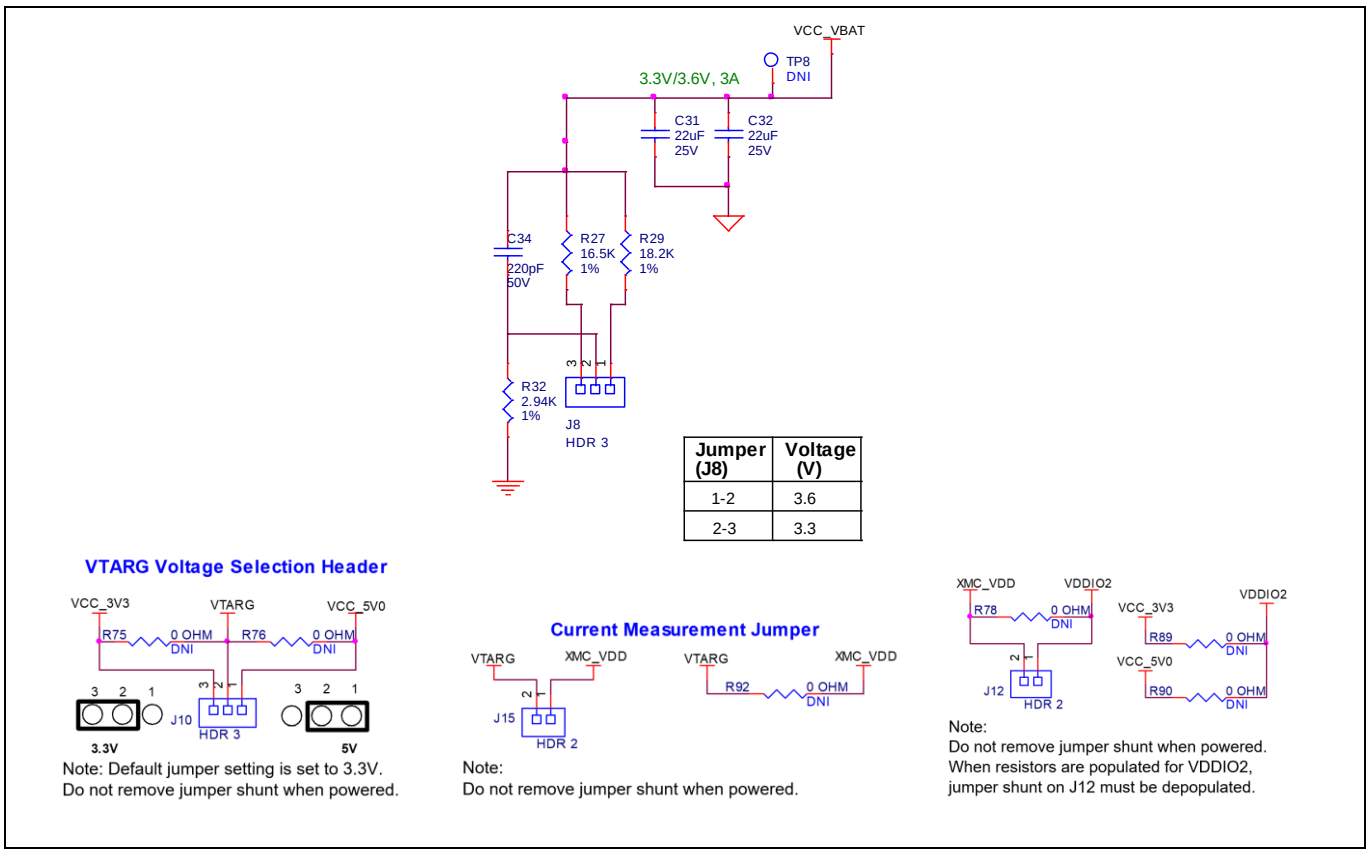


Figure 25 Voltage selection

PSoC™ 5LP uses its programmable SIO (special I/O) pins which can set the logic levels as per the reference voltage for the interface with the target device. The P5LP_SIO_VREF input is used to sense the target voltage and set the logic levels accordingly. The internal ADC is used to monitor the target device voltage and USB supply voltage.

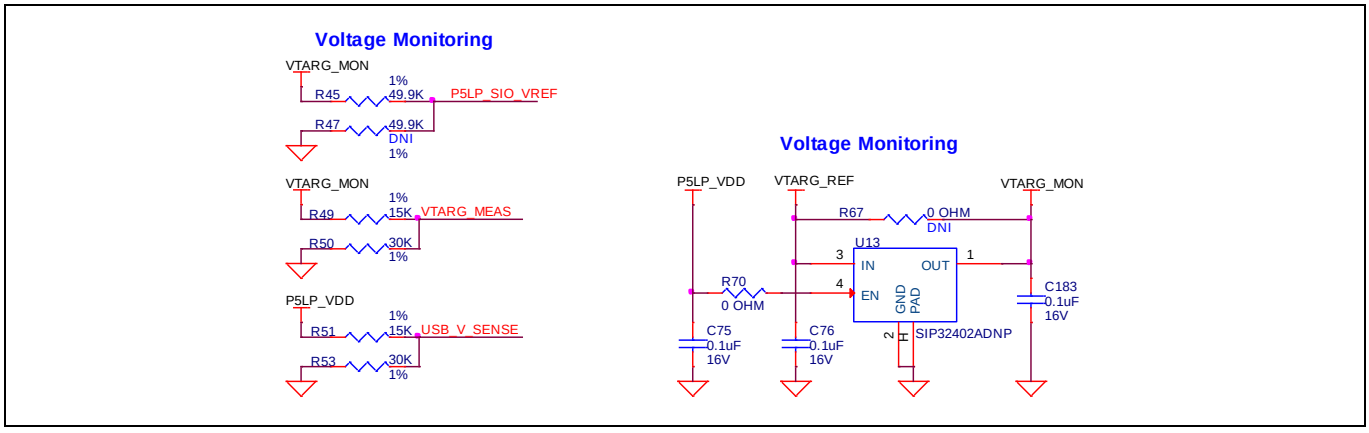


Figure 26 Voltage monitoring

The I/O voltages for XMC7200 can be selected for VDDD, VDDA, and VDDIO2 I/O domains; can be switched between VCC_3V3 or VCC_5V0 voltage levels. The voltage levels for VDDIO1, VDDIO3, and VDDIO4 are dedicated to work on 3.3-V supply voltage. To disable each individual, I/O supply voltage, rework is required by removing the 0-ohm resistor connected to the respective I/O supply voltage.

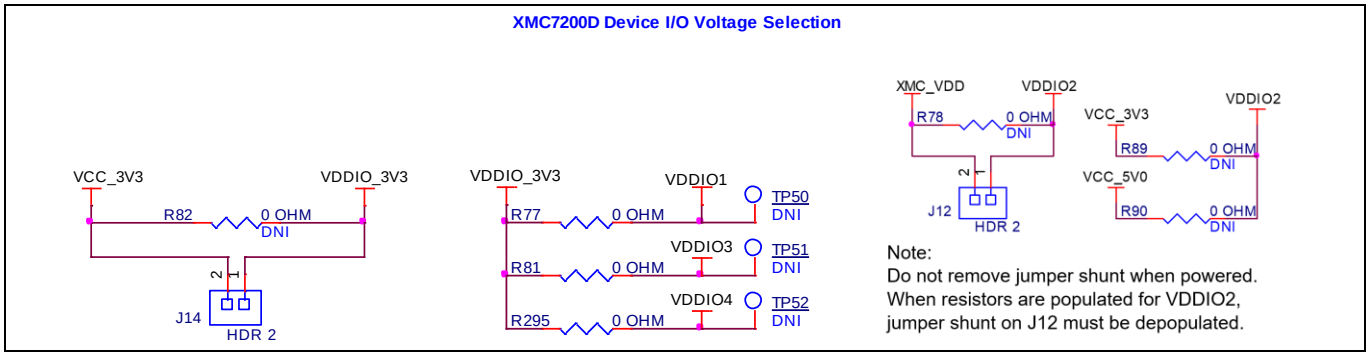


Figure 27 Voltage selection

4.2.6.3 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	Load by default
VBAT	J28	Y
XMC_VDD	J15	Y
VDDIO2	J12	Y
VDDIO1, VDDIO3, VDDIO4	J14	Y
Resistor configuration to measure current on respective I/O domains		
Domain name	**Resistor reference designator**	**Loaded by default**
VDDIO1	R77	Y
VDDIO3	R81	Y
VDDIO4	R295	Y

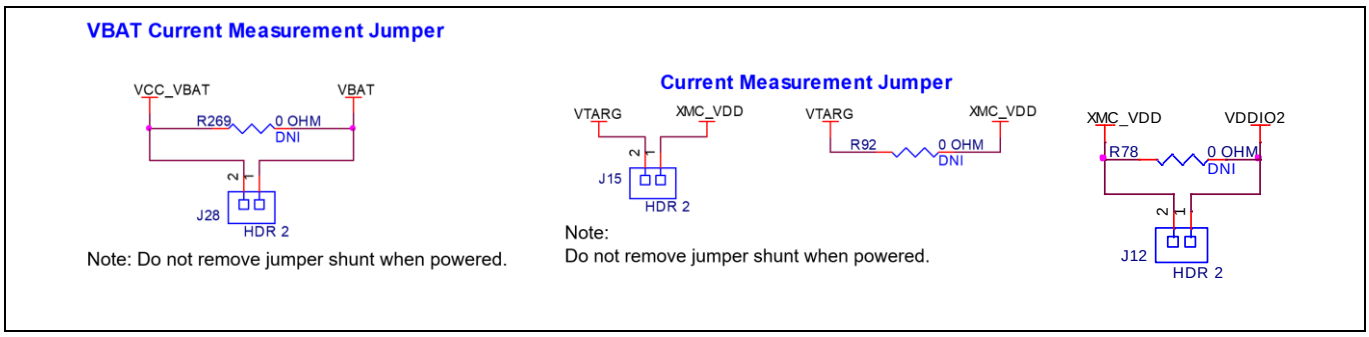


Figure 28 Current measurement headers

Note: When measuring the XMC_VDD current, make sure that jumper shunt from J18 is removed. This will disconnect the potentiometer from VDDA and remove the leakage.

4.2.7 I/O headers

4.2.7.1 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. Connect 3.3 V or 5 V shields compatible with Arduino Uno R3 to develop applications-based on the shield's hardware. Note that 1.8-V shields are not supported by the kit and XMC7200 operating voltage is between 2.7 V and 5 V.

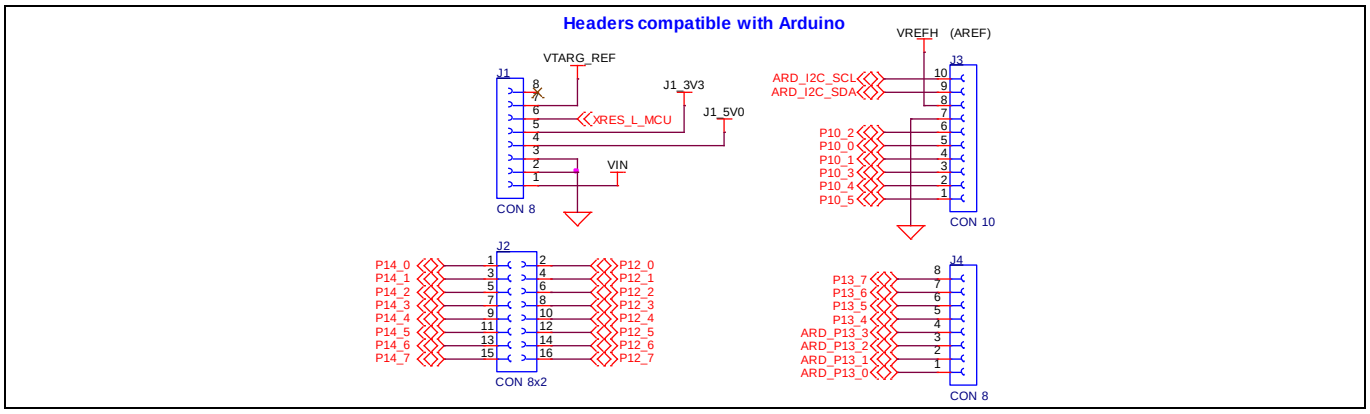


Figure 29 Headers compatible with Arduino

4.2.7.2 XMC7200D I/O headers (J29, J30, J31 and J32)

These headers provide connectivity to XMC7200D GPIOs that are not connected to the headers compatible with Arduino. A few of these pins are multiplexed with onboard peripherals and are not connected to the XMC7200D by default. These connectors are not populated by default.

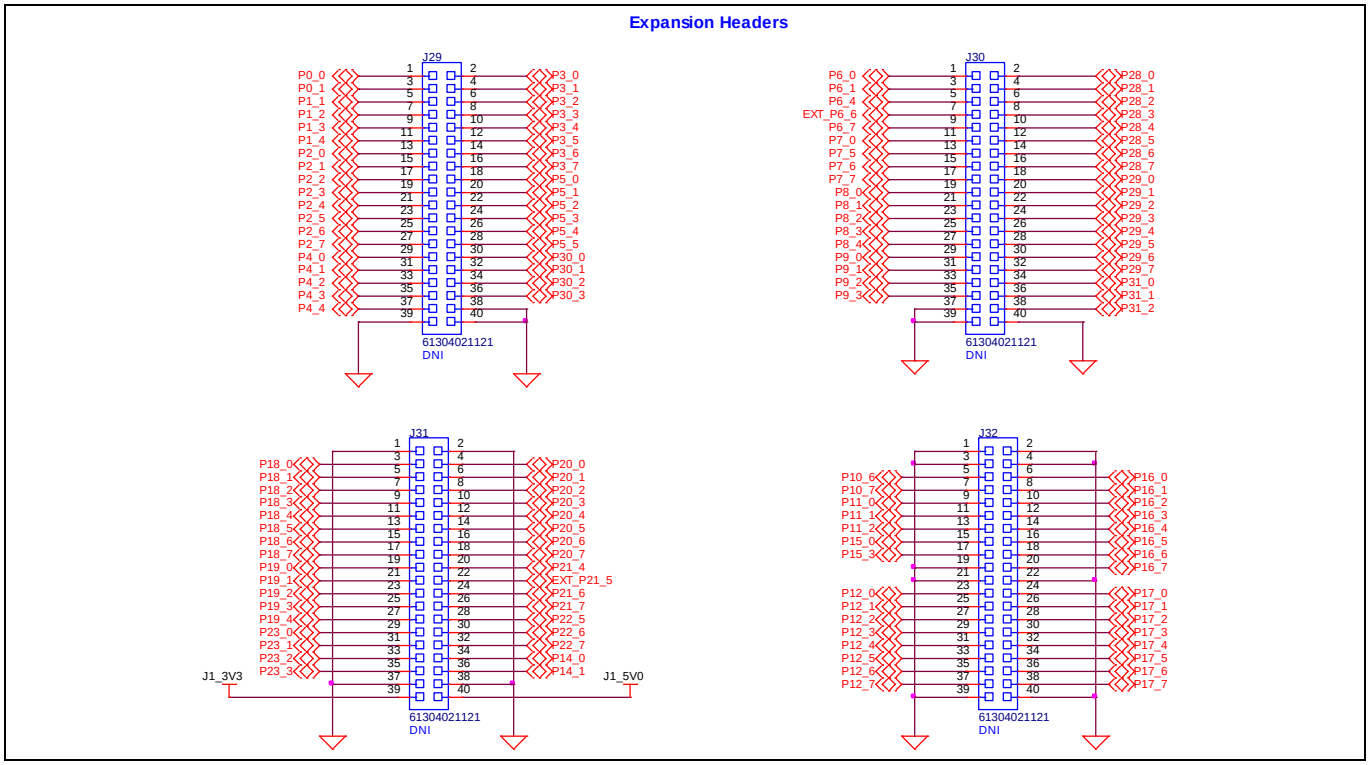


Figure 30 XMC7200 MCU GPIO headers

4.2.8 CAN FD Transceiver

The TLE9251V is a high-speed CAN transceiver, operating as an interface between the CAN controller and the physical bus medium. A high-speed CAN network is a two wire, differential network which allows data transmission rates up to 5 Mb/s. The CANH and CANL pins are the interfaces to the CAN bus; both pins operate as an input and an output. The RxD and TxD pins are the interfaces to the microcontroller. The TxD pin is the serial data input from the CAN controller; the RxD pin is the serial data output to the CAN controller. The voltage level on the digital input TxD and the digital output RxD is determined by the power supply level at the VIO pin. The signal levels on the logic pins (STB, TxD, and RxD) are compatible with microcontrollers having a 5 V or 3.3 V I/O supply. Usually, the digital power supply VIO of the transceiver is connected to the I/O power supply of the microcontroller.

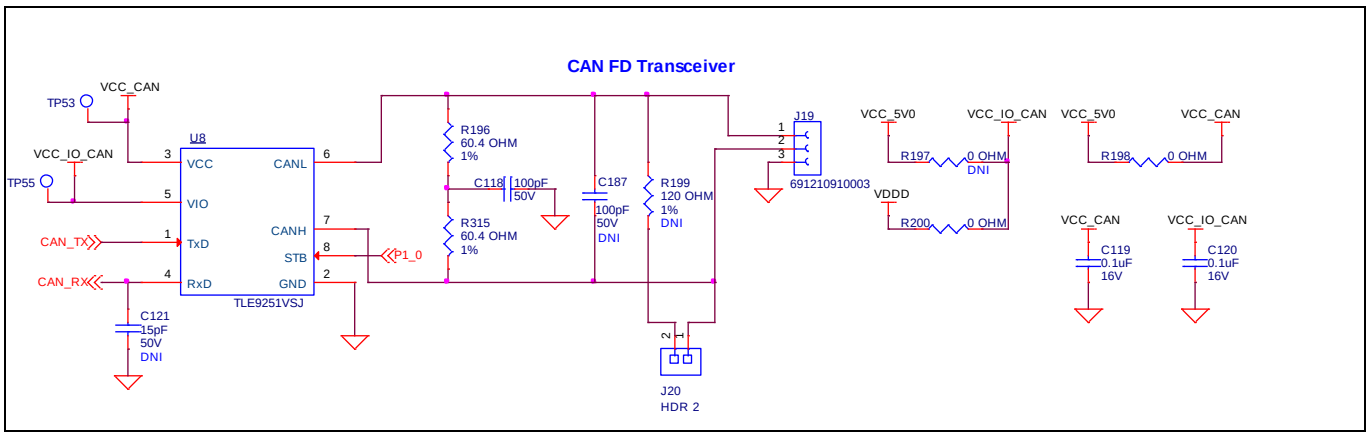


Figure 31 CAN FD Transceiver

Note: The STB pin of CAN transceiver internal pull-up to VIO, the VIO connected to VDDD and therefore contributes to leakage current on the XMC_VDD supply, set the P1[0] (STB pin) to high when measuring the XMC_VDD current for low-power mode.

4.2.9 Quad SPI NOR flash

The XMC7200 evaluation board has a 512-Mbit Quad SPI NOR flash memory (S25FL512SAGMFMR10). The NOR flash is connected to the Quad SPI interface of the XMC7200D.

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

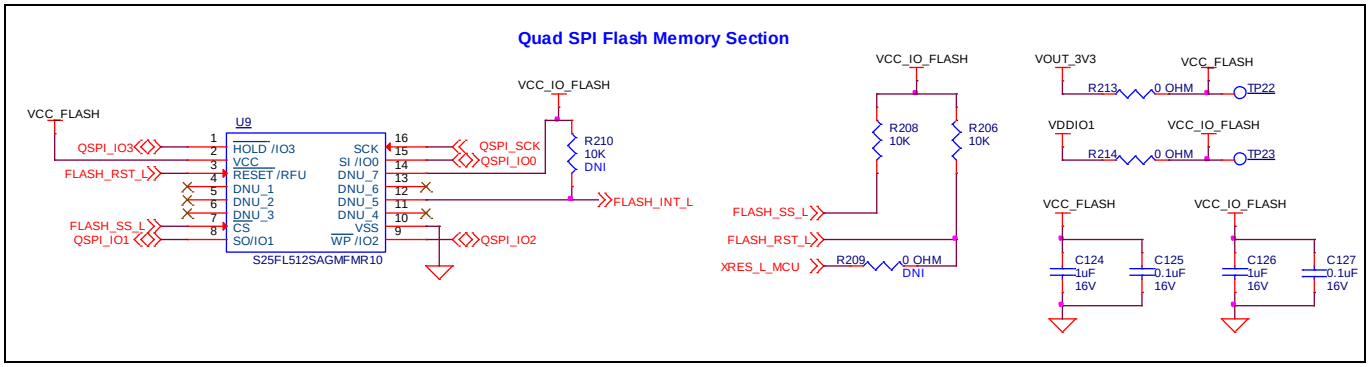


Figure 32 Quad SPI NOR flash

4.2.10 LEDs

LED4 (amber) indicates the status of KitProg3 (See the **KitProg3 user guide** for details). LED5 (amber) power LED indicates the status of power supplied to the board.

The board also has three user-controllable LEDs (LED1, LED2, and LED3) connected to XMC7200D pins for user applications.

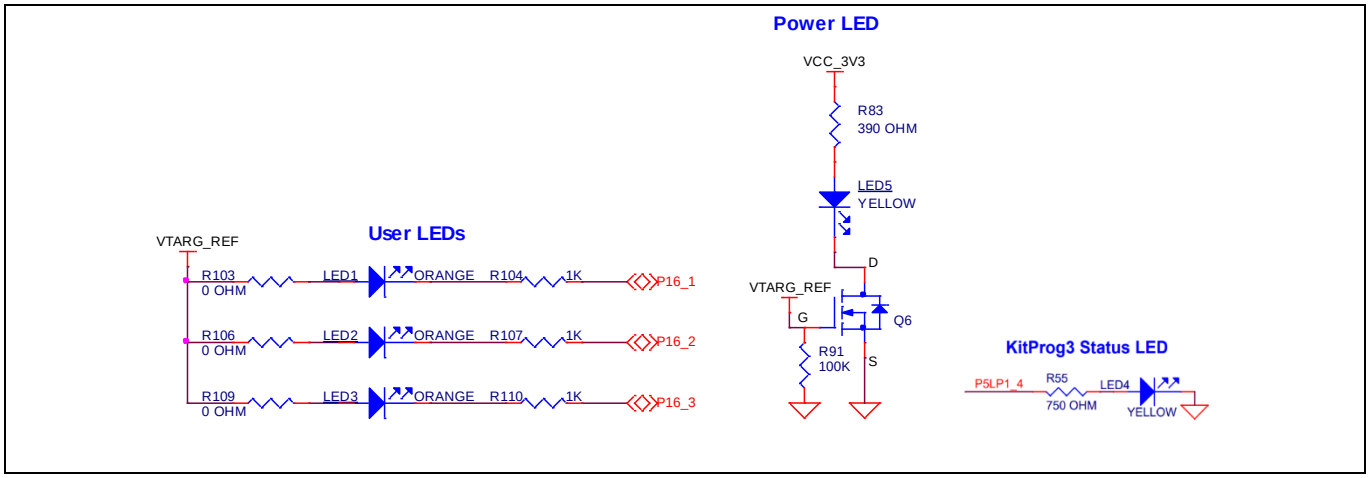


Figure 33 LEDs

4.2.11 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 XMC7200D device and is used to reset the device. Two user buttons (SW2 and SW4) are connected to pin P21[4] and P17[3] of the XMC7200D 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.

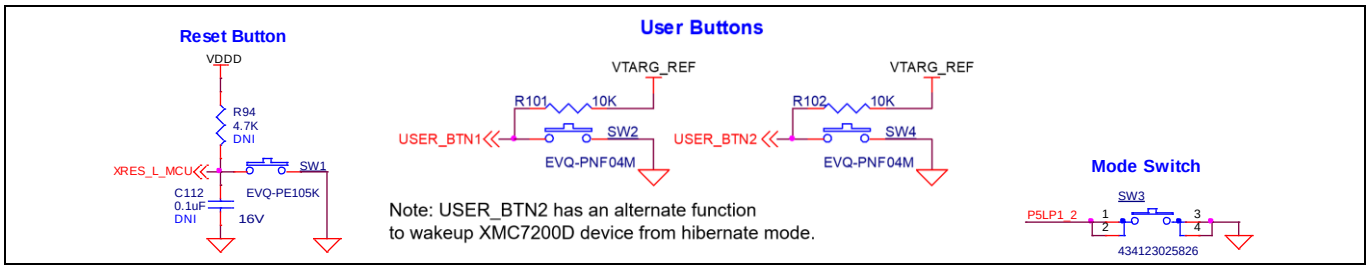


Figure 34 Push buttons

4.2.12 Crystals and oscillators

The board has four crystals/oscillators:

- A 32.768-kHz crystal connected to P21[0] and P21[1] as a watch crystal oscillator
- A 16-MHz crystal connected to P21[2] and P21[3] of the XMC7200D as the external crystal oscillator
- A 25-MHz crystal and 125-MHz oscillator used for Ethernet application
- A 32.768-kHz low-power oscillator for M.2 radio application

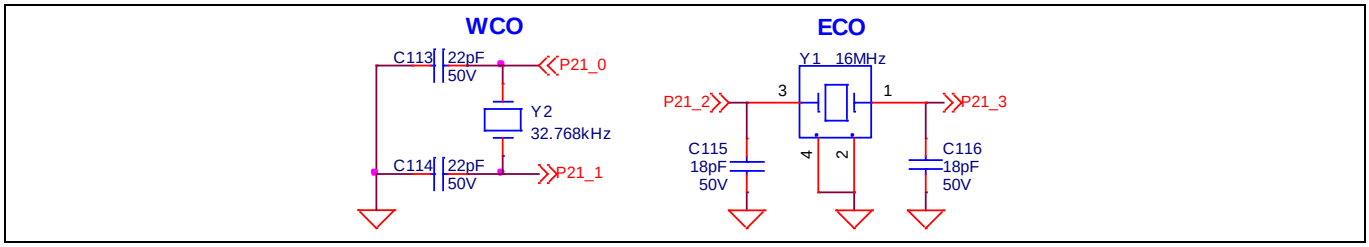
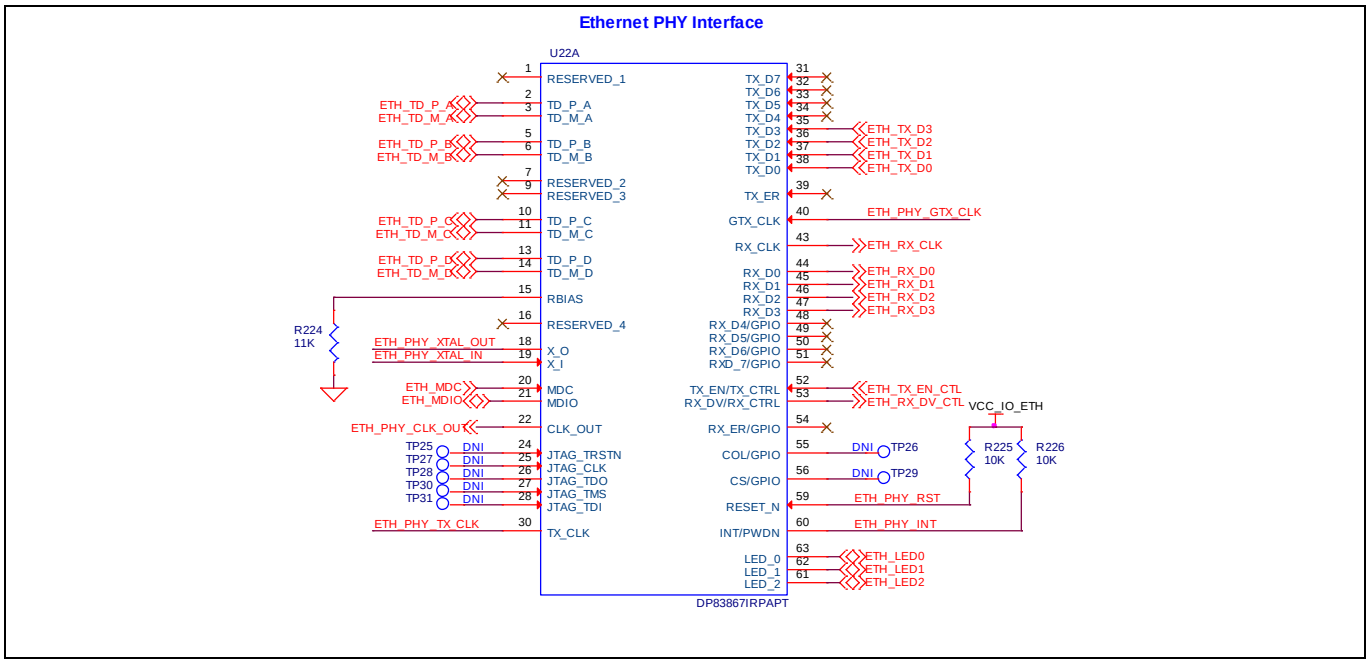


Figure 35 Crystal oscillators

4.2.13 Ethernet transceiver and RJ45 connector

Ethernet interface is provided on the board for RGMII mode operation. A physical layer controller must be used to operate at 10/100/1000 Mbps speeds. The ETH1 port from XMC7200 is interfaced with the Ethernet transceiver with signals necessary to operate in RGMII mode. A RJ45 connector must be used which supports 10/100/1000 Mbps communication speeds. The RJ45 connector must be included with the necessary magnetics inbuilt to protect from external world. There must be two clock references provided on the board for the Ethernet transceiver.

By default, a 25-MHz clock is connected and the 125-MHz crystal is not connected. PHY is capable of working with a 1.1 V core voltage supply and 2.5 V or 3.3 V I/O supply. The device is intended to operate at 3.3 V I/O operation as XMC7200 supports 2.7 V to 5.5 V applications.



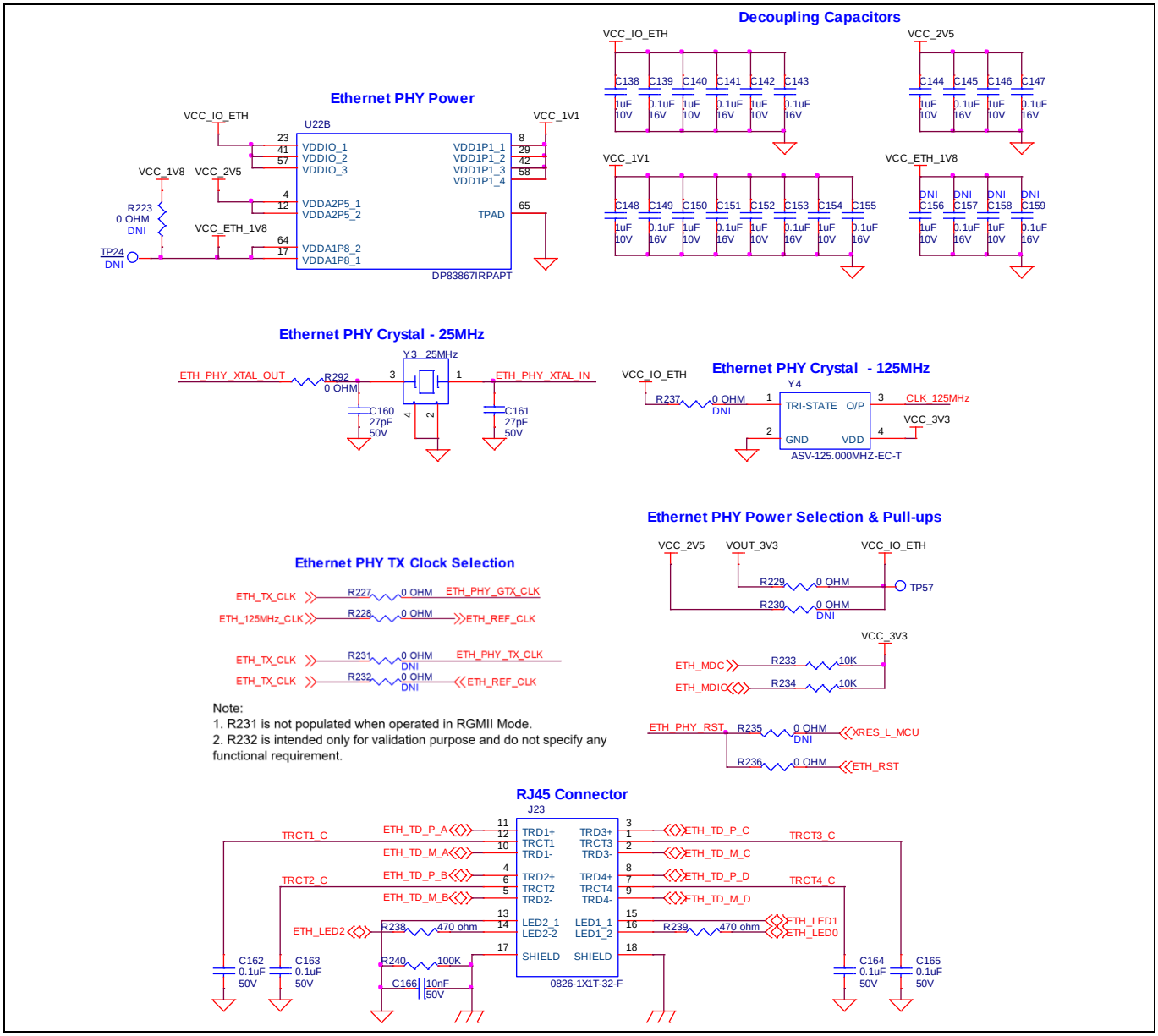


Figure 36 Ethernet transceiver and RJ45 connector

4.2.14 SMIF dual header compatible with Digilent Pmod

Note: Supporting software and code examples will be available in future.

The kit is designed with SMIF dual header compatible with Digilent Pmod (J21 and J22) to validate the HYPERFLASH™ memory interface. Infineon made S26KS series devices with 128-Mb, 256-Mb, and 512-Mb density are expected to be supported with the kit. External memory modules must be interfaced for validating the HYPERFLASH™. The connectors are not populated on the board by default.HYPERFLASH™ interface with S26KL512SDABHI020 is tested up to 64 MHz.

Level translators are used on the board to match the voltage levels of memory modules. Module supply and I/O voltages can be selected with resistor option to switch between 1.8 V and 3.3 V operation. By default, the voltage levels are set to 1.8 V.

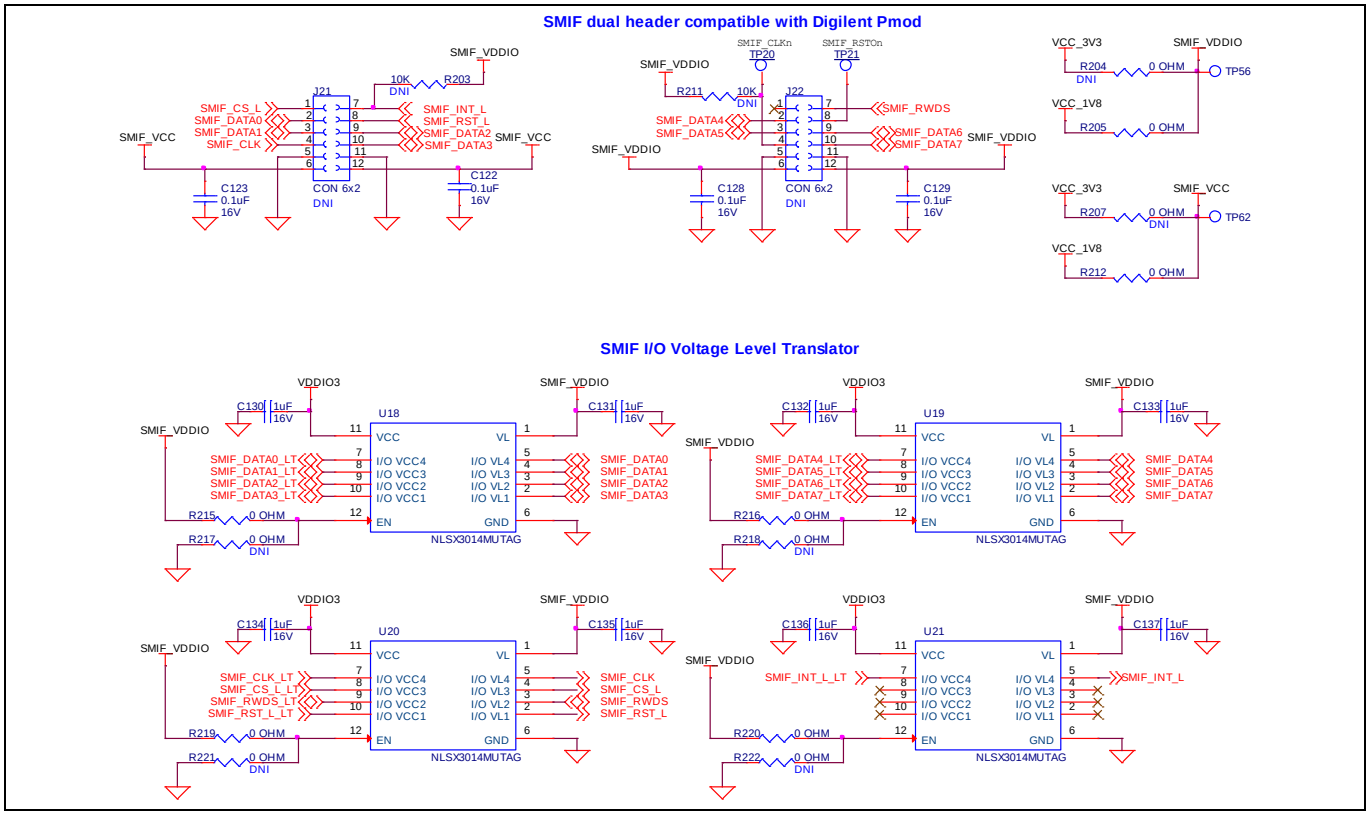


Figure 37 SMIF dual header compatible with Digilent Pmod with voltage level translators

4.2.15 Potentiometer

The board contains a 10k potentiometer connected to P6[6] of the GPIO header J30.7, which is not connected by default. The fixed ends are connected to VDDA (VDD_POT through J18) and GND and therefore may contribute to leakage current on the XMC_VDD supply. Remove the jumper J18 to disconnect power from the potentiometer when measuring the XMC_VDD current.

POT_AOUT signal is connected to VDDIO1 supply block which is limited to work with 3.3 V voltage domain. Consequently, according to the target voltage at J10 header, ADC output readings from the XMC7200D are fairly accurate at 3.3 V. When VTARG is set to 5 V, jumper on J18 header must be depopulated. As a workaround, while working at 5 V target voltage, depopulate the series resistor R188 from connecting P6[6] GPIO to potentiometer and use an external reference voltage by connecting any ADC active GPIO from VDDD or VDDIO2 domain in the test firmware and the specific signal needs to be connected to POT_AOUT.

Note: Clockwise rotation of potentiometer gives the user decremental ADC readings where as counterclockwise provides incremental ADC output.

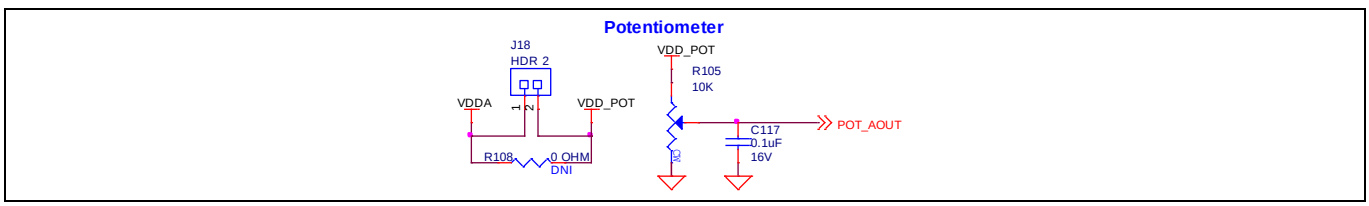


Figure 38 Potentiometer

4.2.16 Reverse voltage protection

Necessary reverse voltage protection circuit has been provided on the kit to avoid the damage due to reverse supply to the board. Any accidental supply coming as input to the board will be prevented by the circuit protecting the specific regulators and XMC7200 MCU. A FET-based diode circuit is used as a reverse voltage protection on the board.

Any reverse supply voltage coming at the 3.3-V regulator U3 and 5-V regulator U16 are terminated before being injected to the regulator output. 3.3 V and 5 V voltage output at Arduino power header J1 and programming header supply are protected by the corresponding reverse voltage protection circuit.

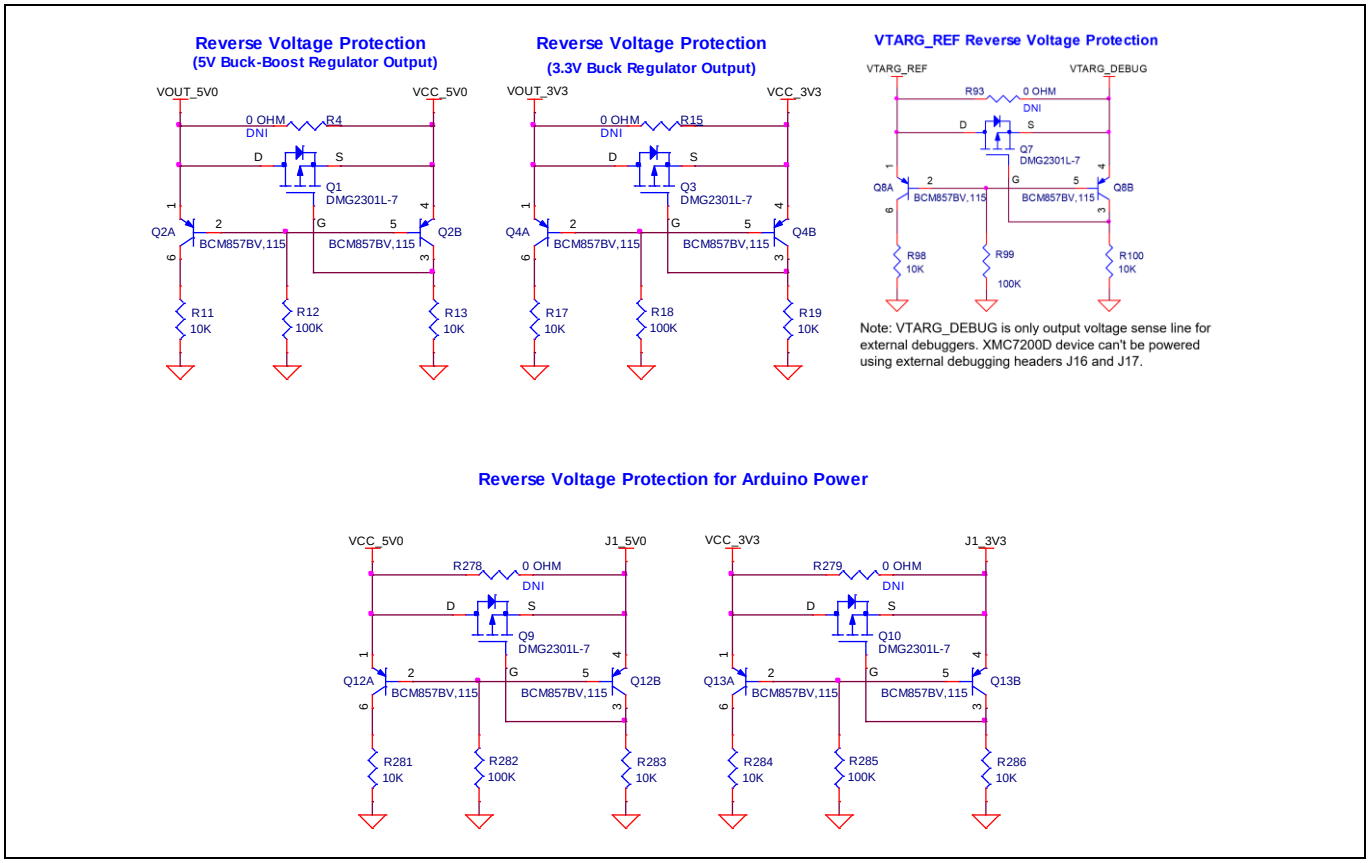


Figure 39 Reverse voltage protection circuit

4.3 Kit rework

4.3.1 SMIF dual header compatible with Digilent Pmod interface

The kit is provisioned with dual Pmod SMIF header (J21, J22) which are by default not loaded on the kit. SMIF memory interface I/Os are shared between SMIF and M.2 (WLAN/Bluetooth®) connector interface.

User should perform the necessary rework on the board by removing or adding corresponding connectors and 0-ohm resistors to work with external SMIF memory modules. Refer **Table 9** to resistor rework for SMIF

Table 9 Dual Pmod header and M.2 connector interface multiplexing

Kit function		XMC7200D pin
M.2 connector	**SMIF (PMOD) module**	**---**
---	R145*	P24[0]
---	R147*	P24[1]
R301	R149*	P24[2]
R300	R152*	P24[3]
---	R154*	P24[4]
R296	R157*	P25[0]
R297	R160*	P25[1]
R298	R162*	P25[2]
R299	R164*	P25[3]
R167	R170*	P25[4]
R173	R176*	P25[5]
R179	R182*	P25[6]
---	R185*	P25[7]

***Not connected by default**

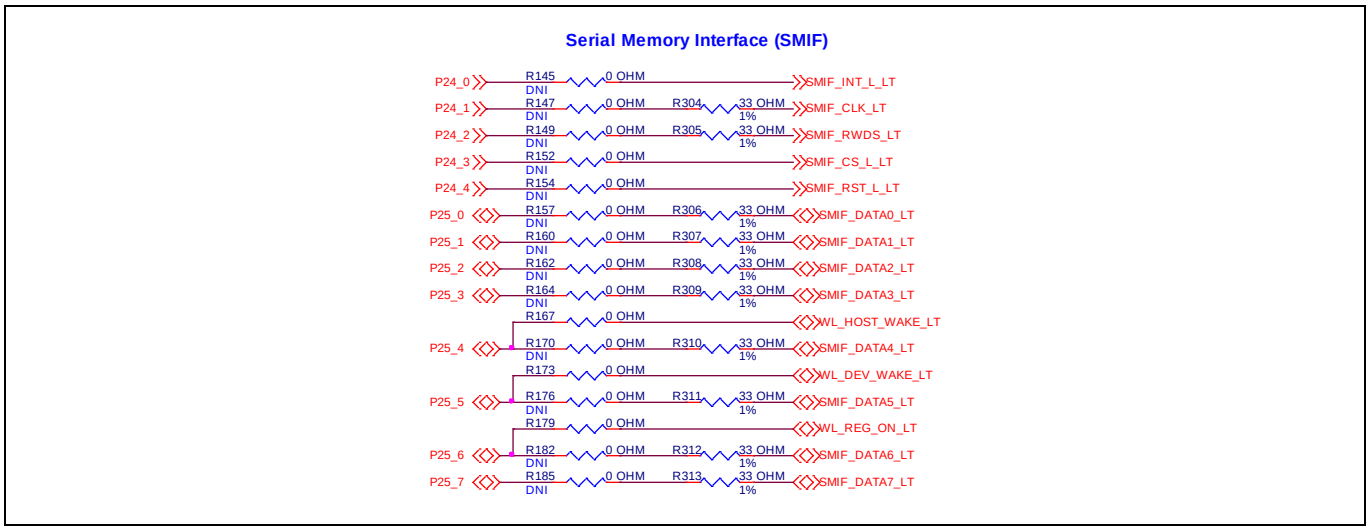


Figure 40 SMIF memory interface

4.3.2 VDDIO2 supply voltage selection

VDDIO2 is by default connected to the XMC_VDD (VTARG) interface; voltage selection depends on the VTARG selection between 3.3 V and 5 V. To select separate voltage domains for VDDIO2, use the resistor rework option provisioned on the board. Populate resistor R89 for 3.3 V and R90 for 5 V. Ensure that you remove jumper J12 whenever using the resistor configuration for VDDIO2.

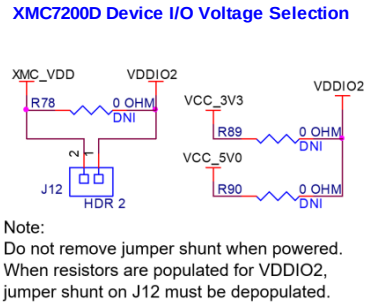


Figure 41 XMC7200 VDDIO2 voltage selection

4.3.3 UART interface on headers compatible with Arduino

UART interface signals are connected between PSoC™ 5LP device and XMC7200 MCU. The same signals have been shared across digital interface connector compatible with Arduino J4.1–J4.4. For using UART lines on connector compatible with Arduino, perform the board rework with the specific resistor assembly.

R134, R136, R148 and R150 all four 0-ohm resistors to be populated to work with UART on a connector compatible with Arduino. You can use either the interface for PSoC™ 5LP and XMC7200 or XMC7200 and the interface specific to Arduino.

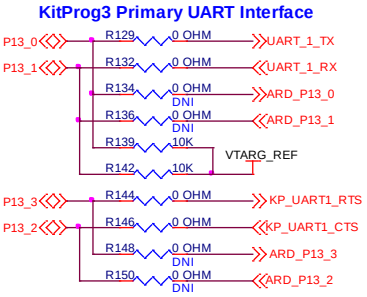


Figure 42 Primary UART interface

4.3.4 WLAN UART interface

The secondary UART from PSoC™ 5LP device is connected to WLAN UART lines of the M.2 connector. The board rework is performed to use WLAN UART lines for Bluetooth® UART.

Remove 0-ohm resistors R65, R69 and populate R64, R66, R71, R73 for Bluetooth® UART interface to be active.

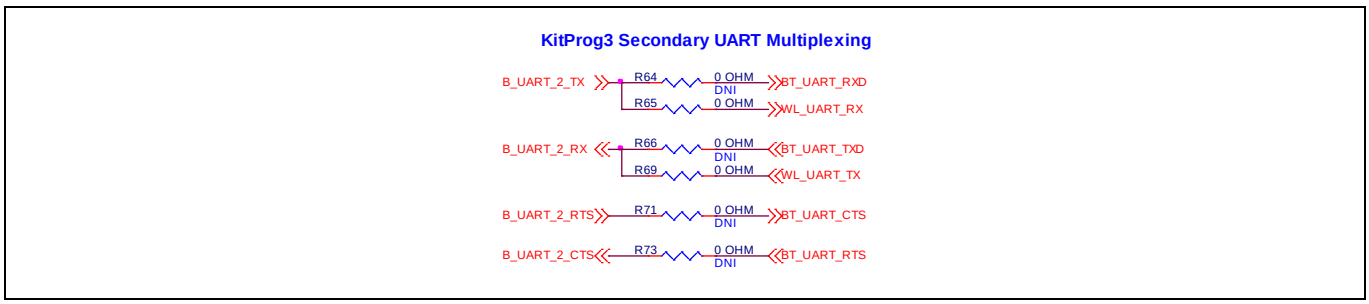


Figure 43 Secondary UART interface

4.3.5 ETM trace and debug header

By default, the ETM trace and debug header is not populated on the kit. Populate the 20-pin ETM standard connector before using a third-party programmer for trace programming.

Remove resistors R155 and R158, and populate R193 and R194 to connect the XMC7200D to the trace header.

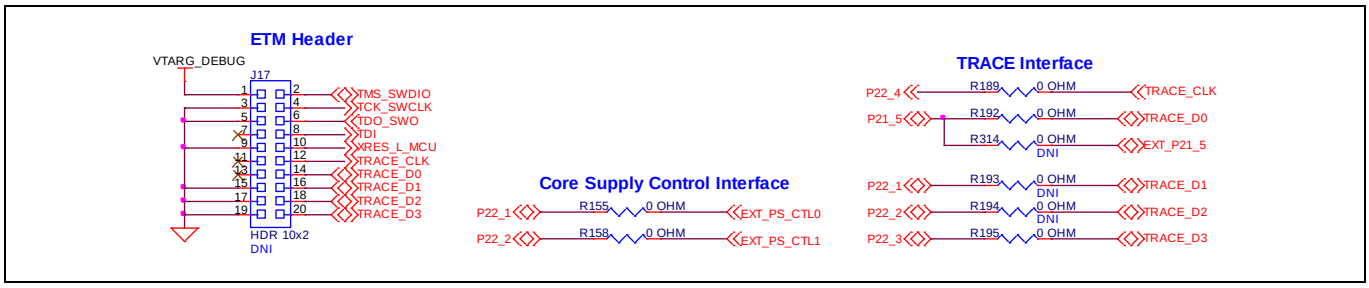


Figure 44 ETM trace and debug header

4.3.6 125-MHz oscillator

By default, the kit is set to work with 10/100 Mbps Ethernet operation on RGMII mode. You need to perform board rework to operate at 1000 Mbps (Gigabit): Remove R241 and populate R242 to remove the 25-MHz crystal interface (R292) and connect to the 125-MHz oscillator output.

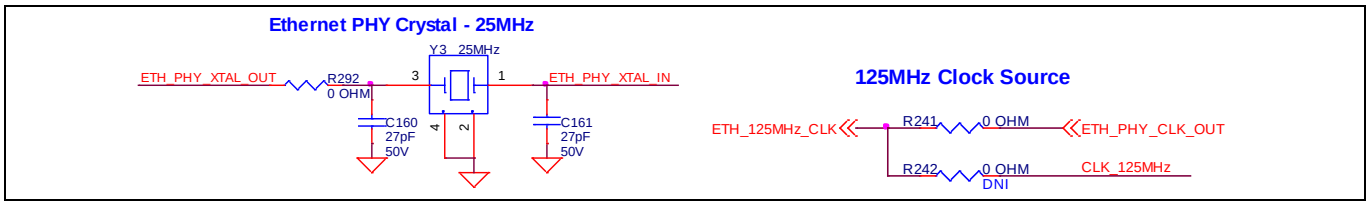


Figure 45 125-MHz oscillator

4.3.7 WLAN SDIO interface

On the kit, default connection is made for WLAN SDIO interface which is connected to an M.2 connector. Whenever SMIF (Pmod) memory module is functionally enabled by specific rework, the series resistors connected to the M.2 connector must be removedas shown in **Table 9**.

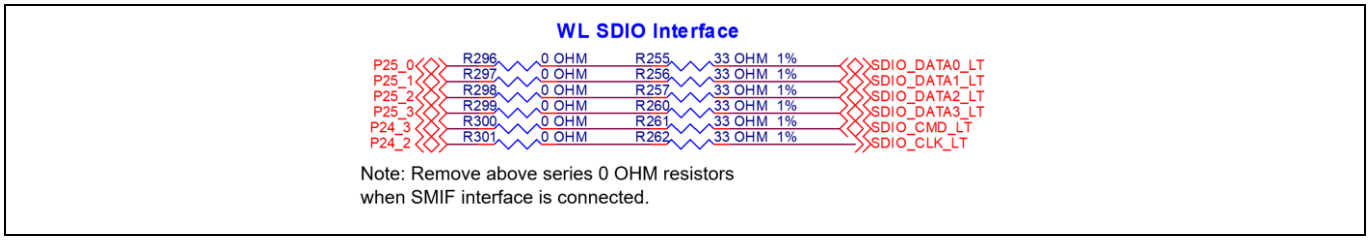


Figure 46 WLAN SDIO interface

4.3.8 UART interface between PSoC™ 5LP and XMC7200D with flow control

If any application requires the UART interface between PSoC™ 5LP and XMC7200D with flow control, below the indicated voltage level translators U10 and U11 is used. The user needs to have rework done on the board by removing series 0-ohm resistors R129, R132, R144 and R146 by populating R134, R136, R148 and R150 when required to use these flow control pins on J4 header compatible to Arduino Uno.

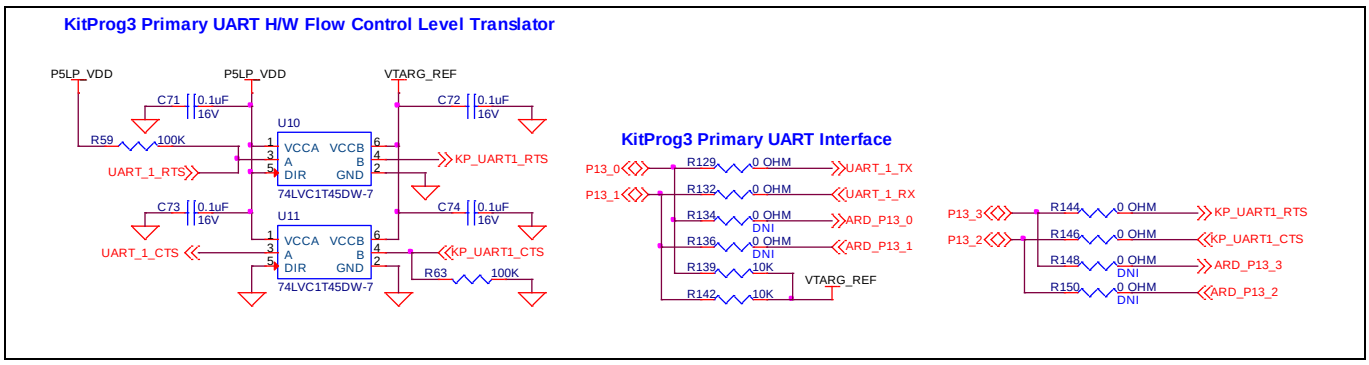


Figure 47 UART interface between PSoC™ 5LP and XMC7200D with flow control

4.4 Bill of materials

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

4.5 Frequently asked questions

13. Is the core external regulator capable to handle load currents up to 600mA?

The core external regulator is implemented with an external NPN pass transistor in KIT_XMC72_EVK.

The pass transistor (Q5) in the Rev** has a schematic error. The collector and emitter pins of the transistor are swapped as in image below.

This error on transistor does not impact functionality of the code examples provided with kit and, works as expected. The circuit can still provide load currents up to 600mA

This issue will be resolved in future revisions of this EVK.

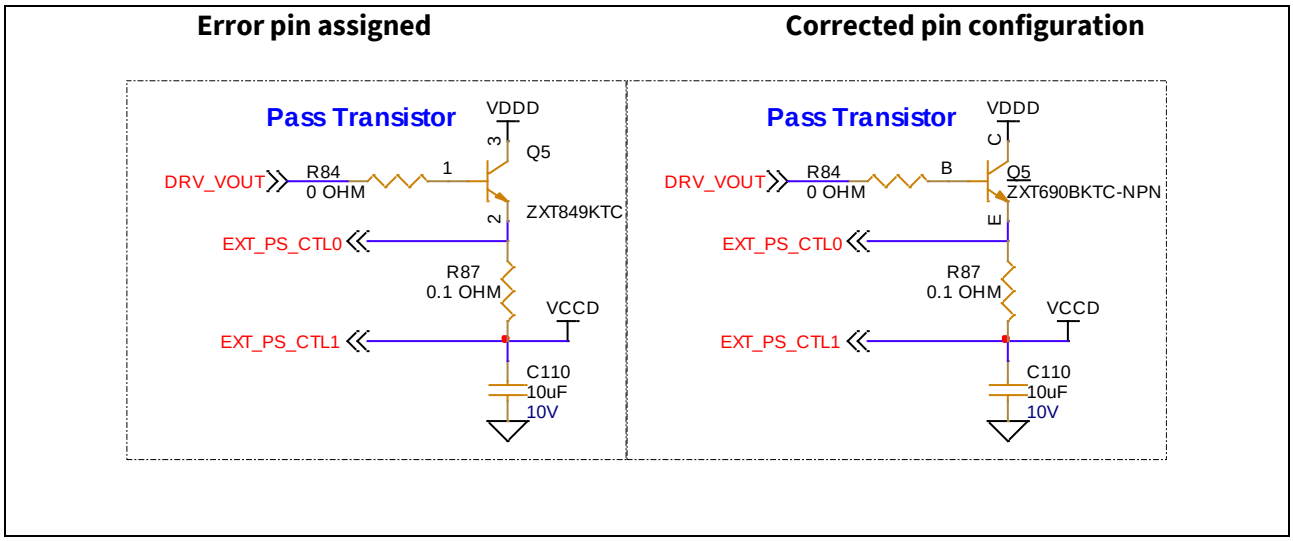


Figure 48 Pass transistor configuration

14. How does XMC7200 evaluation kit handle voltage connection when multiple power sources are plugged in?

There are two different options to power the kit:

- KitProg3 Micro-B USB connector (**J7**)
- External DC supply via VIN connector (**J6**)

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

15. 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 for this kit.

Table 10 Input voltage levels

Supply	Typical input voltage	Absolute max
USB Micro-B connector (**J7**)	4.5 V to 5.5 V	5.5 V
VIN connector (**J6**)	5 V to 12 V	18 V

16. Why is the voltage of the kit restricted to 3.3 V? Does it drive external 1.8 V interfaces?

    XMC7200D is not meant to be operated at voltages lower than 2.7 V. Powering the XMC7200D to lower than 2.7 V will stop the chip to be powered ON. It is recommended to power the XMC7200D at 3.3 V or 5 V.

17. HJow to program the target devices?
    - Check **J10** to ensure that the jumper shunt is placed.
    - Make sure that no external devices are connected to the external programming header J16 or J17.
    - Update your KitProg3 version to the latest as mentioned in the **KitProg3 user guide**.

18. Can the kit be powered using external program/debug headers J16 and J17?

    No, this is not possible in this board by default. The target MCU is powered by onboard regulators only and therefore, one of the two main sources (J7 and J6) must be present. There is a protection circuit that prevents reverse voltage from VTARG_Debug to VTARG. Therefore, the board cannot be powered through J16 and J17.

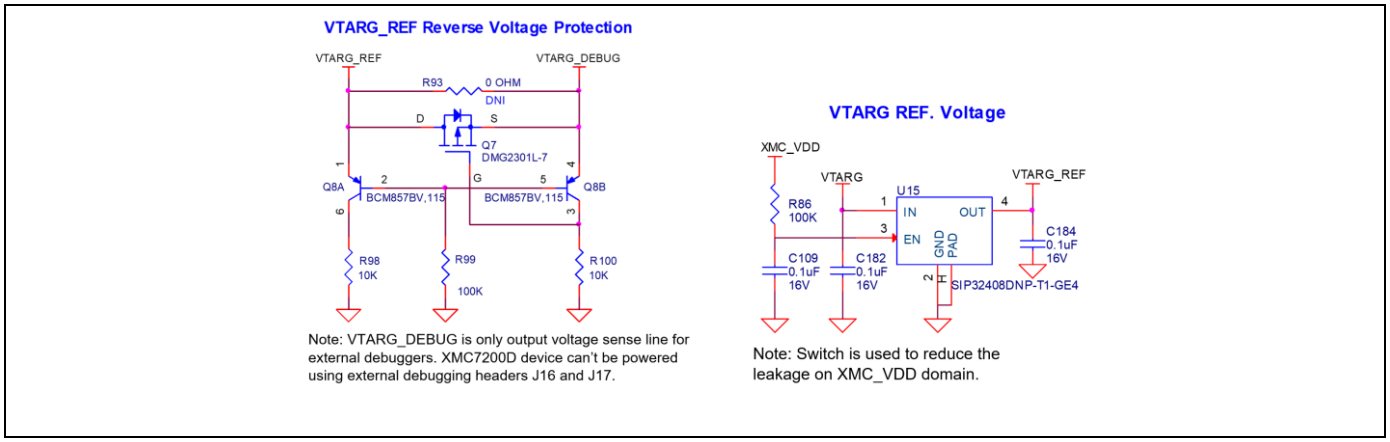


Figure 49 VTARG reverse voltage protection

References

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

[1] **AN234334 - Getting started with XMC7000 MCU on ModusToolbox™ software**

[2] **XMC7100, XMC7200 datasheet**

Revision history

Document revision	Date	Description of changes
**	2022-06-23	Initial release
*A	2022-10-07	Updated Kit details: Updated Board details: Updated Figure 2. Updated Figure 3. Updated Table 4. Updated Kit operation: Updated Theory of operation: Updated Figure 6. Updated Figure 7. Updated KitProg3: onboard programmer/debugger: Updated Programming and debugging using ModusToolbox™ software: Updated Figure 8. Updated Figure 10. Updated Figure 11. Updated Figure 12. Updated Figure 13. Updated Hardware: Updated Hardware functional description: Updated I/O headers: Updated Headers compatible with Arduino Uno R3 (J1, J2, J3, J4): Updated Figure 29. Updated Crystals and oscillators: Updated description. Updated Ethernet transceiver and RJ45 connector: Updated description. Updated Potentiometer: Updated description.
*B	2023-02-02	Updated hyperlinks across the document. Updated to new template.
*C	2023-04-21	Updated the Pass transistor configuration in Frequently asked questions section.

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

Edition 2023-04-21

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2023 Infineon Technologies AG. All Rights Reserved.

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

002-34389 Rev. *C

Warnings

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.