KIT_XMC71_EVK_LITE_V1 XMC7100 Lite Evaluation Board guide

About this document

Scope and purpose

This document explains about the KIT_XMC71_EVK_LITE_V1 XMC7100 Evaluation Board: kit operation, out-of-the-box example and its operation, and the hardware details of the board.

Intended audience

This document is intended for all embedded developers using the KIT_XMC71_LITE_EVK_V1 XMC7100 Lite Evaluation board.

Evaluation board

This board is to be used during the design-in process to evaluate and measure the characteristic curves, and to check 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

Table of contents

1 Introduction

The XMC7100 Evaluation Board enables you to evaluate and develop your applications using the XMC7100D microcontroller (hereafter called “XMC7100D”).

XMC7100D is designed for industrial applications. XMC7100D is a true programmable embedded system-on-chip, integrating up to two 250 MHz Arm® Cortex®-M7 as the primary application processors, 100 MHz Arm® Cortex®-M0+ that supports low-power operations, up to 4 MB flash and 768 kB SRAM, Ethernet, Controller Area Network Flexible Data-Rate (CAN FD), programmable analog and digital peripherals that allow faster time-to-market.

The evaluation board has two Infineon’s shield2Go standard headers (S2GO) and a mikroBus header for interfacing sensors, 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, a micro-B connector for USB device interface, two user LEDs, one potentiometer, and two push buttons. The board supports operating voltages from 3.3 V to 5.0 V for XMC7100D. Additionally, provision for standard 10/100 Base Tx Ethernet PHY transceiver with RJ45 connector interface is provided.

ModusToolbox™ software is used to develop and debug the XMC7100D projects. ModusToolbox™ is a set of tools that enable you to integrate these devices into your existing development methodology.

For more details on XMC7100D, see the AN234334 - Getting started with XMC7000 MCU on ModusToolbox™ software application note, which aids in creating a customized design using the Eclipse IDE for ModusToolbox™.

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 XMC7100 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 XMC7100 Evaluation Board is supported in ModusToolbox™. ModusToolbox™ is a free development eco-system that includes the Eclipse IDE for ModusToolbox™ and the XMC7100 SDK with XMC7100D. ModusToolbox™ enables and configures device resource and middleware libraries; writes C/assembly source code; and programs and debugs the device. The software can be downloaded from the ModusToolbox™ home page. For more details, see the ModusToolbox™ tools package installation guide.
- There are wide range of code examples to evaluate the XMC7100 board. These examples help to familiarize with XMC7100D, and help to create a customized design. These examples can be accessed through ModusToolbox™ Project Creator tool. To access the code examples, visit Infineon’s Code Example page.

1.2 Additional learning resources

Infineon provides a wealth of data on the 32-bit XMC™ Industrial Microcontroller Arm® Cortex®-M**Error! Hyperlink reference not valid.** webpage to select the right XMC™ MCU for the design and to quickly and effectively integrate the device into the design.

1.3 Technical support

For more assistance, go to Infineon support page. Ask questions about the kit on the Infineon Developer Community webpage.

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.

2 Kit details

2.1 Kit contents

- XMC7100 Evaluation Board
- USB Type-A to Micro-B cable
- Quick Start Guide (printed on the kit package)

Figure 1 Kit contents

For information on missing parts, go to Infineon support page.

2.2 Board details

The XMC7100 Evaluation Board has the following features:

- XMC7100D – XMC7100D-F176K4160AA. See the device datasheet.
- 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 XMC7100D
- CAN FD interface
- Two user LEDs, two user buttons, and a reset button for the XMC7100D
- A potentiometer that can be used to simulate analog sensor output
- A mode button and a mode LED for KitProg3
- Headers compatible with Arduino Uno R3
- Headers compatible with MIKROE's mikroBUS shields

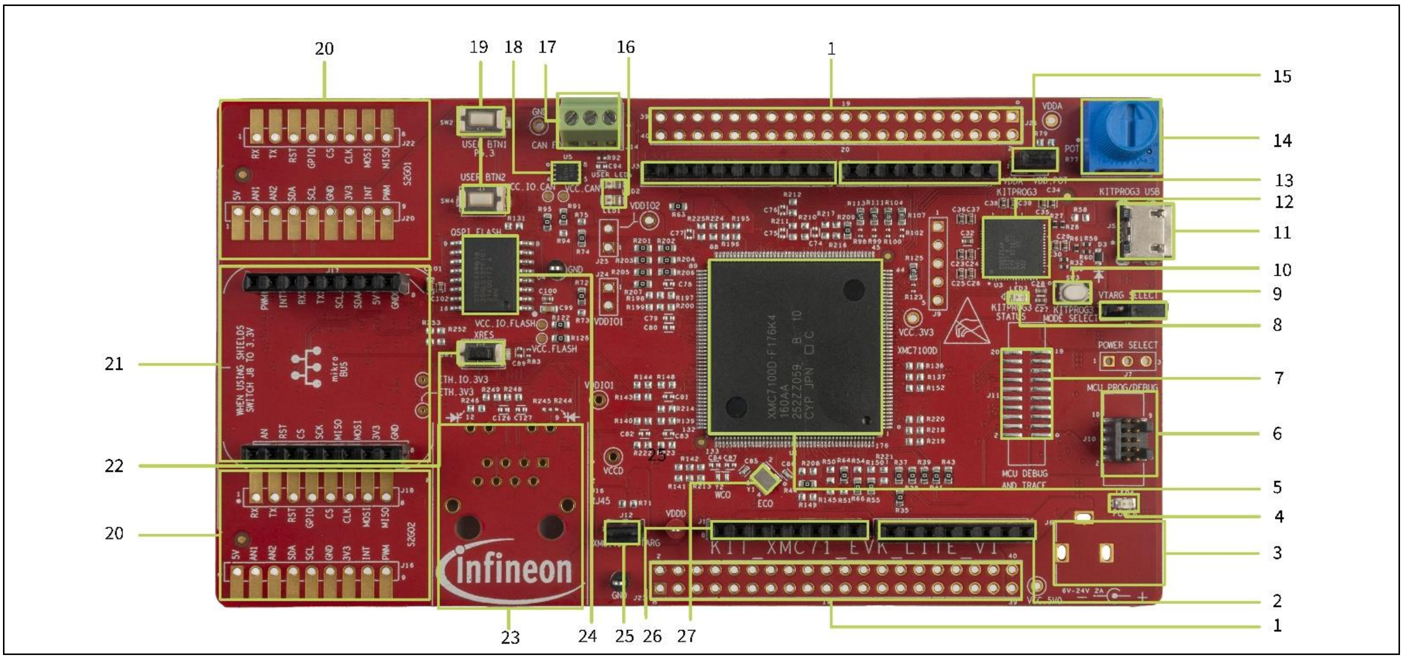


Figure 2 XMC7100 Evaluation Board – top view

The XMC7100 Evaluation Board contains the following (see Figure 2):

1. XMC7100 extended I/O headers (J21, J23)*
2. Analog-IN header compatible with Arduino Uno R3 (J2)
3. External power supply VIN connector (J6)*
4. Power LED (LED4)
5. XMC7100 microcontroller (XMC7100D-F176K4160AA - U1)
6. XMC7100 10-pin SWD/JTAG program and debug header (J10)
7. XMC7100 20-pin debug and trace header (J11)*
8. KitProg3 status LED (LED3)
9. System power (VTARG) selection jumper (J8)
10. KitProg3 programming mode selection button (SW3)
11. KitProg3 USB connector (J5)
12. KitProg3 programmer and debugger (CY8C5868LTI-LP039 - U3)
13. Digital I/O headers compatible with Arduino Uno R3 (J3, J4)
14. Potentiometer (R77)
15. Potentiometer power supply header (J13)
16. XMC7100 user LEDs (LED1, LED2)
17. CAN FD interface connector (J14)
18. CAN FD transceiver (TLE9251VLE - U5)
19. XMC7100 user buttons (SW2, SW4)
20. Shield2Go interface headers (S2GO1, S2GO2)*
21. mikroBUS shield compatible header (J17, J19)
22. XMC7100 reset button (SW1)
23. RJ45 Ethernet connector (U16)*
24. 512-Mbit serial NOR flash memory (U4)

25. XMC7100 VTARG current measurement jumper (J12)
26. Power header compatible with Arduino Uno R3 (J1)
27. External crystal oscillator for XMC7100 (Y1)

*Note: *Footprint only, not populated on the board.*

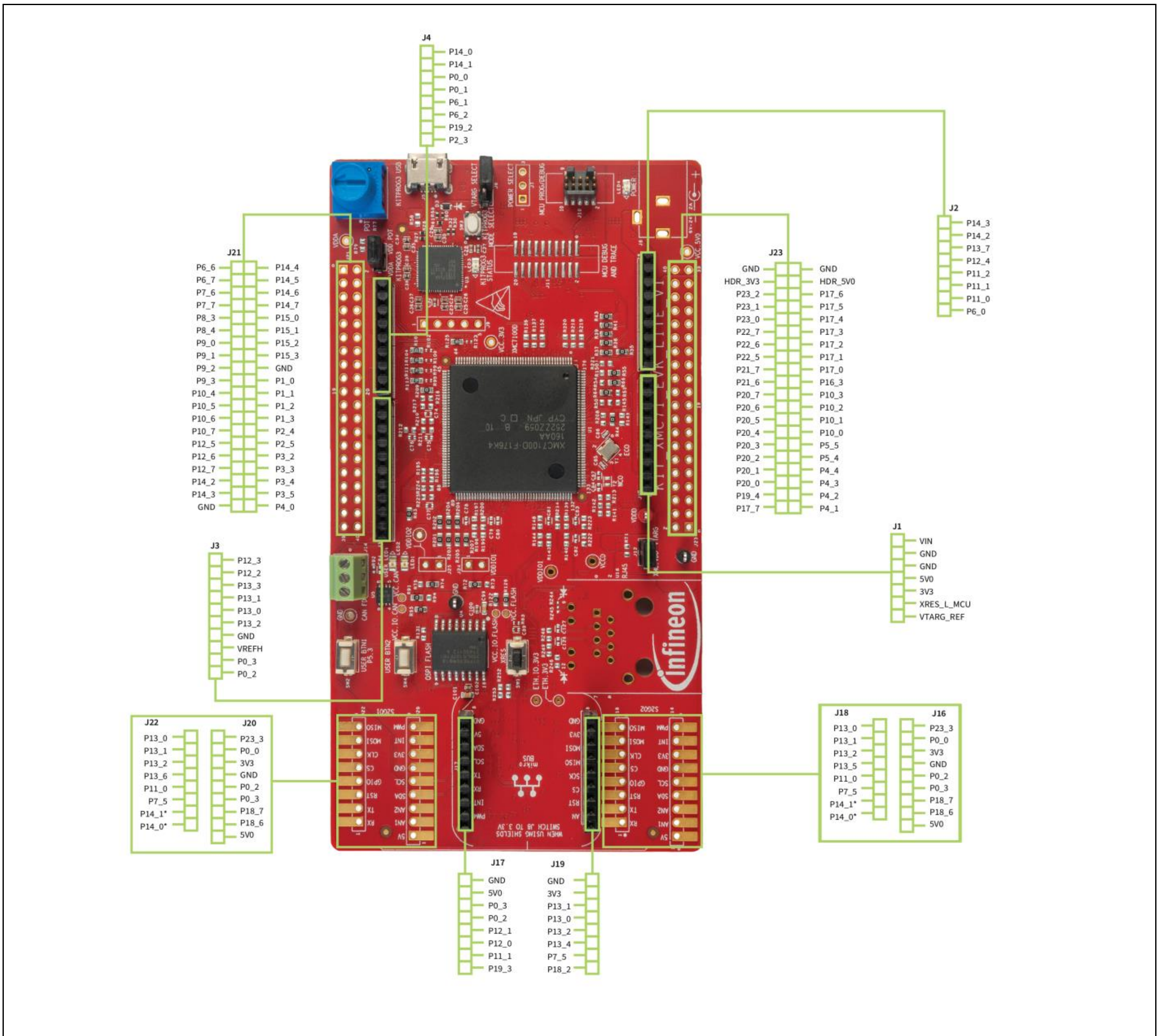


Figure 3 XMC7100 Evaluation Board pinout

*Note: *Not connected on the board by default. Rework required by populating zero ohm resistor for respective interface.*

Table 3 **XMC7100 Evaluation Board pinout**

Pin	Primary onboard function	Secondary onboard function	Connection details
XMC7100D pins			
XRES	Hardware reset	–	Remove R44 to disconnect it from KitProg3 reset.
P0[0]	D2 – header compatible with Arduino (J4.3)	–	–
P0[1]	D3 – header compatible with Arduino (J4.4)	–	–
P0[2]	I2C_SCL	Header compatible with Arduino (J3.10)	I2C_SCL communication interface between KitProg3 and XMC7100D
P0[3]	I2C_SDA	Header compatible with Arduino (J3.9)	I2C_SDA communication interface between KitProg3 and XMC7100D
P2[0]	UART RX	–	UART RX interface between KitProg3 and XMC7100D
P2[1]	UART TX	–	UART TX interface between KitProg3 and XMC7100D
P2[2]	ETH_RX_ER	–	ETH_RX_ER signal on standard RMII Ethernet interface
P2[3]	D7 – header compatible with Arduino (J4.8)	–	–
P3[0]	MDIO	–	MDIO interface between Ethernet PHY and XMC7100D
P3[1]	MDC	–	MDC interface between Ethernet PHY and XMC7100D
P5[0]	User LED1	–	Connected to USER_LED1 from XMC7100D
P5[1]	User LED2	–	Connected to USER_LED2 from XMC7100D
P5[2]	STB pin on CAN	–	Connected to Standby pin of CAN transceiver. Internal pull-up. Active low signal.
P5[3]	User Button1	–	Connected to USER_BTN1 using 10K pull-up resistor.
P6[0]	POT_AN_OUT	Connected to header compatible to Arduino (J2.1)	Remove R78 to disconnect from the potentiometer. Populate R76 to connect to header compatible to Arduino J2.1
P6[1]	D4 – header compatible with Arduino (J4.5)	–	–
P6[2]	D5 – header compatible with Arduino (J4.6)	–	–

Pin	Primary onboard function	Secondary onboard function	Connection details
P6[3]	QSPI_SCK	–	Connected to Clock signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P6[4]	FLASH_RST_L	–	Connected to reset signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P6[5]	FLASH_INT_L	–	Connected to interrupt signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P7[0]	FLASH_SS_L	–	Connected to slave-select signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P7[1]	QSPI_IO0	–	Connected to IO0 signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P7[2]	QSPI_IO1	–	Connected to IO1 signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P7[3]	QSPI_IO2	–	Connected to IO2 signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P7[4]	QSPI_IO3	–	Connected to IO3 signal of QSPI NOR flash U4-S25HL512T from XMC7100D
P7[5]	XRES_L_HDR	–	Connected to RESET/GPIO of S2GO and mikroBUS headers Remove R209 to disconnect from P7[5] Populate R208 to connect to XRES from XMC7100D
P8[0]	CAN_TX	–	Connected to RxD of CAN transceiver TLE9251VLE
P8[1]	CAN_RX	–	Connected to TxD of CAN transceiver TLE9251VLE
P8[2]	USER_BTN2	–	Populate R82 (1K) to connect user button2. By default, not populated. Used only when 100-TEQFP is populated as target MCU
P11[0]	A1 – header compatible with Arduino (J2.2)	Shield2Go GPIO S2GO2-J18.4 and S2GO1-J22.4	Shared pins between Header compatible to Arduino and Shield2Go header

Pin	Primary onboard function	Secondary onboard function	Connection details
P11[1]	A2 – header compatible with Arduino (J2.3)	INT_MIKRO_BUS connected to J17.2	Shared pins between Header compatible to Arduino and Shield2Go header
P11[2]	A3 – header compatible with Arduino (J2.4)	–	–
P12[0]	UART_RX_MIK_BUS	–	Connected to UART RX of mikroBUS header interface
P12[1]	UART_TX_MIK_BUS	–	Connected to UART TX of mikroBUS header interface
P12[2]	D9 – header compatible with Arduino (J10.2)	–	–
P12[3]	D8 – header compatible with Arduino (J10.1)	–	–
P12[4]	A4 – header compatible with Arduino (J2.5)	–	–
P13[0]	XMC_SPI_MISO	–	Multiplexed signal between XMC7100D, header compatible to Arduino, mikroBUS header and S2GO header interface
P13[1]	XMC_SPI_MOSI	–	Multiplexed signal between XMC7100D, header compatible to Arduino, mikroBUS header and S2GO header interface
P13[2]	XMC_SPI_CLK	–	Multiplexed signal between XMC7100D, header compatible to Arduino, mikroBUS header and S2GO header interface
P13[3]	XMC_SPI_CS0	–	Connected to SPI select signal between header compatible to Arduino (J3.3) and XMC7100D
P13[4]	XMC_SPI_CS1	–	Connected to SPI select signal between mikroBUS header (J19.3) and XMC7100D
P13[5]	XMC_SPI_CS2	–	Connected to SPI select signal between shield2Go header S2GO2- (J18.5) and XMC7100D
P13[6]	XMC_SPI_CS3	–	Connected to SPI select signal between shield2Go header S2GO1- (J22.5) and XMC7100D
P13[7]	A5 – header compatible with Arduino (J2.6)	–	–

Pin	Primary onboard function	Secondary onboard function	Connection details
P14[0]	UART_RX_ARD	UART_RX_S2GO connected to header S2GO2- J18.1 and S2GO1- J22.1	Connected to RX of header compatible to Arduino (J4.1)
P14[1]	UART_TX_ARD	UART_TX_S2GO connected to header S2GO2- J18.2 and S2GO1- J22.2	Connected to TX of header compatible to Arduino (J4.1)
P18[0]	ETH_PHY_XIN	ETH_PHY_50MHz_IN	Connected to 25 MHz clock input from crystal to Ethernet PHY Remove R154 to disconnect from 25 MHz crystal Populate R193 to connect 50 MHz clock from Oscillator output
P18[1]	XMC_ETH_TX_CTL	–	Connected to Tx enable of Ethernet transceiver U12
P18[2]	AN_MIKRO_BUS	–	Connected to mikroBUS header J19.1
P18[3]	XMC_TRACE_CLK	–	Populate R56 to connect with trace clock signal
P18[4]	XMC_ETH_TXD0	–	Data0 receive signal of Ethernet transceiver U12
P18[5]	XMC_ETH_TXD1	–	Data1 receive signal of Ethernet transceiver U12
P18[6]	AN1_S2G	–	Connected to Analog pin of shield2go header. S2GO2- J16.2 and S2GO1- J20.2
P18[7]	AN2_S2G	–	Connected to Analog pin of shield2go header. S2GO2- J16.3 and S2GO1- J20.3
P19[0]	XMC_ETH_RXD0	–	Data0 transmit signal of Ethernet transceiver U12
P19[1]	XMC_ETH_RXD1	–	Data1 transmit signal of Ethernet transceiver U12
P19[2]	D6 – header compatible with Arduino (J4.7)	–	–
P19[3]	PWM_MIKRO_BUS	–	Connected to PWM pin of mikroBUS header J17.1
P21[0]	WCO IN (Y2)	–	32.768 kHz watch crystal oscillator input. Not loaded by default.
P21[1]	WCO OUT (Y2)	–	32.768 kHz watch crystal oscillator output. Not loaded by default.

Pin	Primary onboard function	Secondary onboard function	Connection details
P21[2]	ECO IN (Y1)	–	20 MHz external crystal oscillator input
P21[3]	ECO OUT (Y1)	–	20 MHz external crystal oscillator output
P21[4]	User button (USER_BTN2) (Also used for hibernate wakeup function)	–	–
P21[5]	XMC_ETH_RX_CTL	TRACEDATA [0]	Connected to RX_CTL pin of Ethernet transceiver. Populate R50 to connect with XMC™ MCU 20-pin ETM header J11.
DRV_VOUT	DRV_VOUT	–	Gate drive signal on external pass transistor for core regulator feedback control.
P22[1]	EXT_PS_CTL0	TRACEDATA [1]	Connected to external pass transistor for core regulator feedback control. XMC™ MCU 20-pin ETM header J11 By default, connected to external pass transistor Q3. Need R50 to populate.
P22[2]	EXT_PS_CTL1	TRACEDATA [2]	Connected to external pass transistor for XMC7100D core regulator feedback control. Trace data on XMC™ MCU 20-pin ETM header J11 Not connected by default, populate R51 for trace operation.
P22[3]	XMC_ETH0_RST	TRACEDATA [3]	Connected to reset pin of Ethernet transceiver. Trace data on XMC™ MCU 20-pin ETM header J11. Populate R55 for trace operation.
P22[4]	TRACE_CLOCK	–	XMC™ MCU 20-pin ETM header J11 specific to 176-TEQFP device. By default, not connected, need R57 to populate.
P23[3]	PWM S2GO	–	Connected to PWM pin of standard Shield2Go interface.

Pin	Primary onboard function	Secondary onboard function	Connection details
P23[4]	TDO_SWO	–	XMC™ MCU 10-pin SWD/JTAG interface J10 / 20-pin ETM header J11
P23[5]	TCK_ SWCLK	–	XMC™ MCU 10-pin SWD/JTAG interface at J10 / 20-pin ETM header J11
P23[6]	TMS_ SWDIO	–	XMC™ MCU 10-pin SWD/JTAG interface at J10 / 20-pin ETM header J11
P23[7]	TDI	–	XMC™ MCU 10-pin SWD/JTAG interface at J10 / 20-pin ETM header J11

3 Kit operation

3.1 Theory of operation

The XMC7100 Evaluation Board is built around XMC7100D. For more information on device features, see the device datasheet.

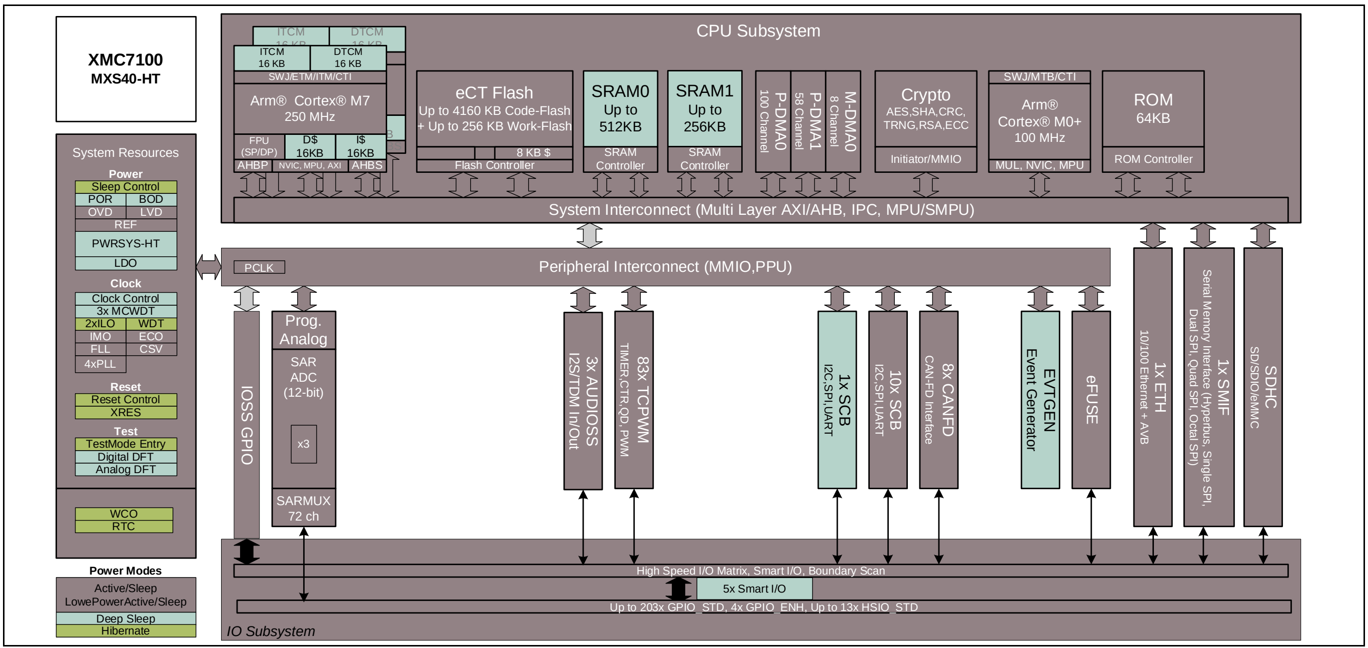


Figure 4 XMC7100D block diagram

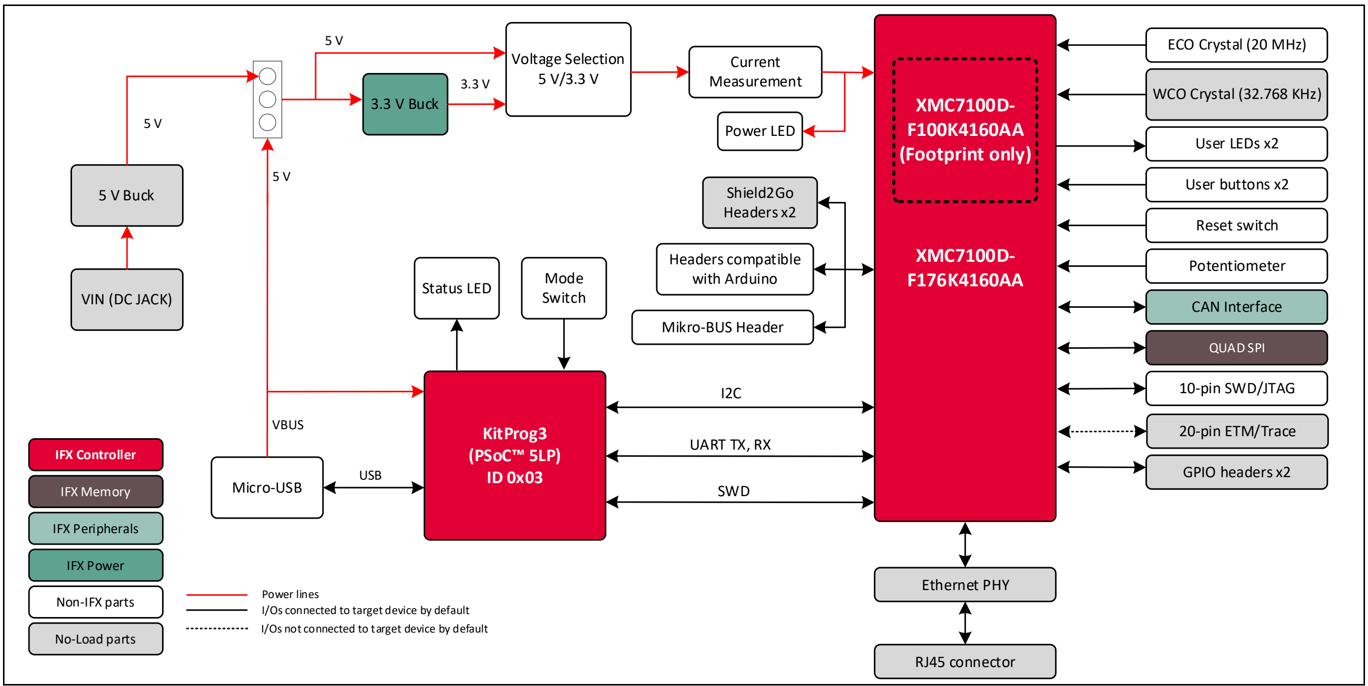


Figure 5 XMC7100 Evaluation Board block diagram

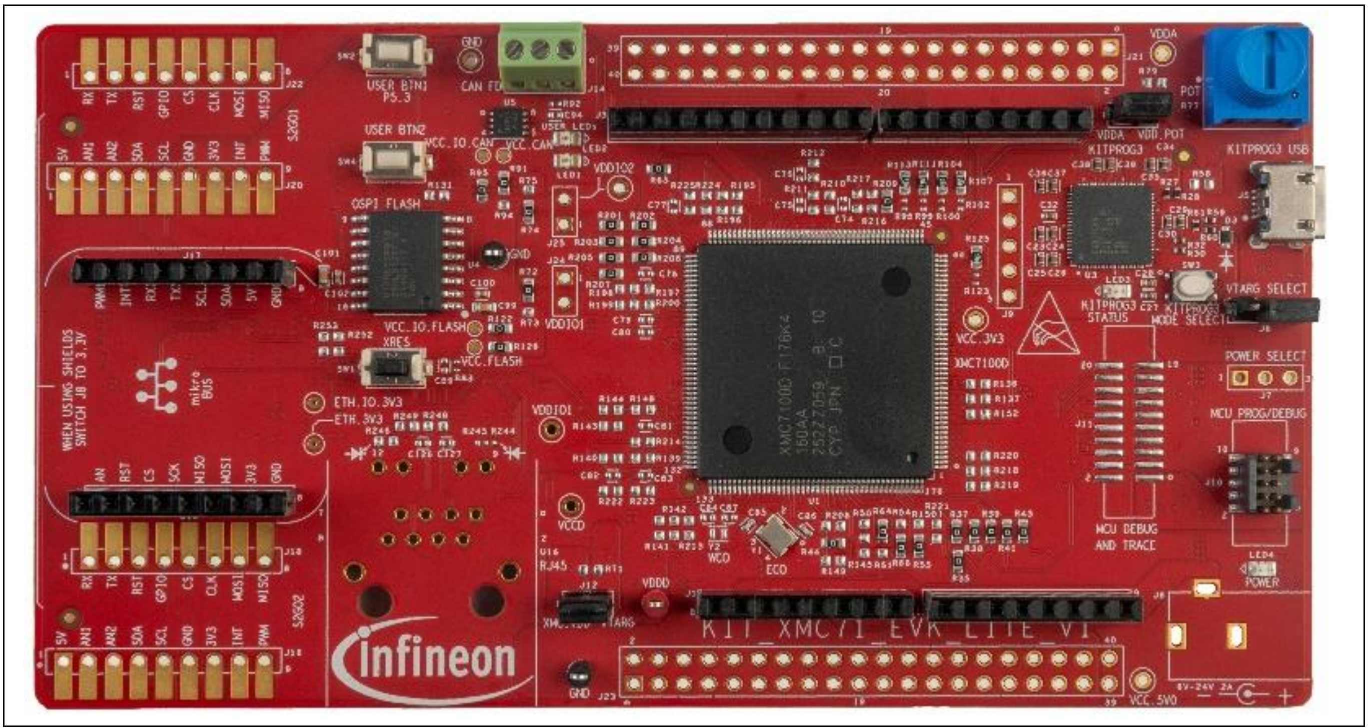


Figure 6 XMC7100 Evaluation Board – top view

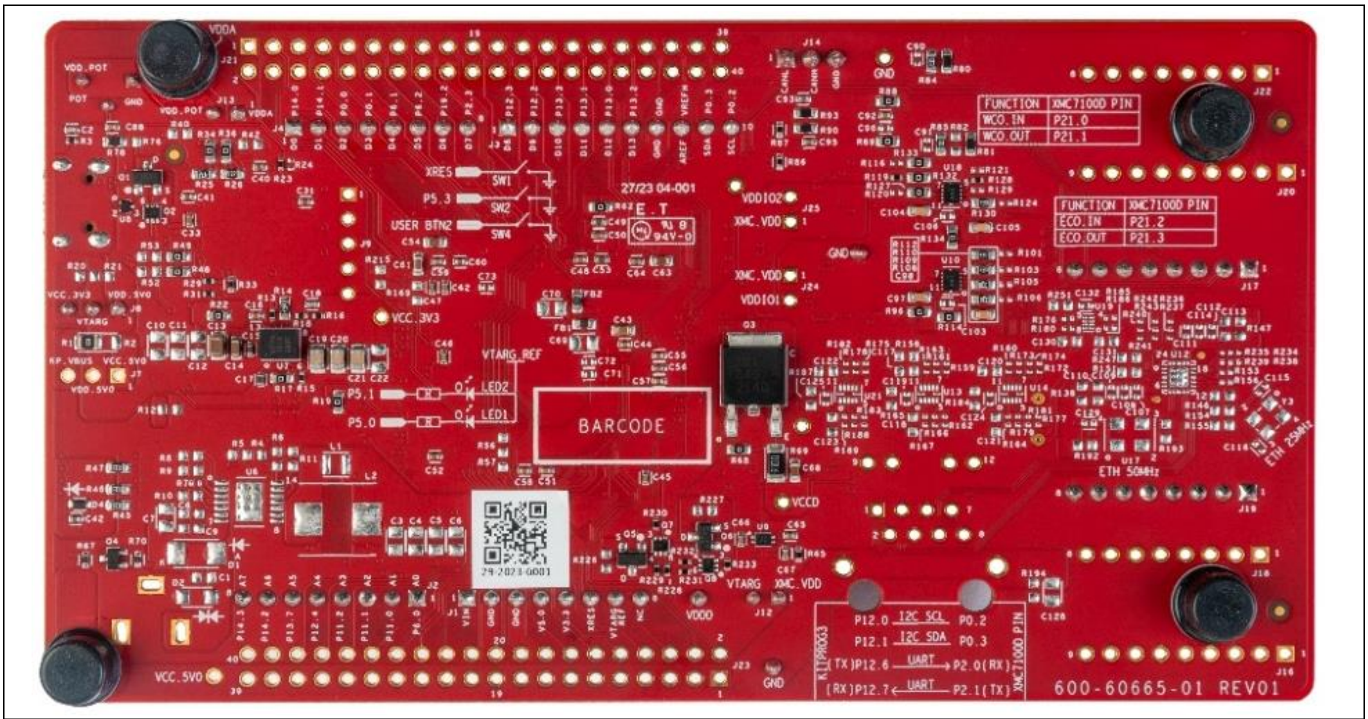


Figure 7 XMC7100 Evaluation Board – bottom view

The XMC7100 Evaluation Board has the following peripherals.

Table 4 **Peripheral details**

Sl. No.	Peripheral	Description
1.	External power supply VIN connector (J6)	External power supply from DC Jack. A DC adapter with 7 V – 24 V, 2.5 A can be used as a source of power to the kit. Not loaded by default.
2.	Power LED (LED4)	Power supply ON/OFF LED status indicator.
3.	KitProg3 USB connector (J5)	Connect to a PC to use the KitProg3 onboard programmer and debugger and to provide power to the board.
4.	KitProg3 status LED (LED3)	Indicates the status of KitProg3. For details, see the KitProg3 user guide.
5.	512 Mbit serial NOR flash memory (S25HL512T - U9)	Connected to the serial memory interface (SMIF) of the XMC7100D. 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.
8.	XMC7100D VTARG current measurement jumper (J12)	Connect an ammeter to this jumper to measure the current consumed by the XMC_VDD power domain of the MCU.
9.	XMC7100D VDDIO1 current measurement jumper (J24)	J24 connects MCU VDDIO1 to XMC_VDD which is outcome from J8 (VCC_3V3 or VDD_5V0) power supply selection. Not loaded by default.
10	XMC7100D VDDIO2 current measurement jumper (J25)	J25 connects MCU VDDIO2 to XMC_VDD which is outcome from J8 (VCC_3V3 or VDD_5V0) power supply selection. Not loaded by default.
11.	System power (VTARG) selection jumper (J8)	J8 is a three-pin voltage selection header to select between VCC_3V3 or VDD_5V0 voltages.
12.	XMC7100D 20-pin debug and trace header (J11)	Connect to an Embedded Trace Macrocell (ETM)-compatible programmer and debugger. This is not loaded by default.
13.	XMC7100D 10-pin SWD/JTAG program and debug header (J10)	This 10-pin header allows you to program and debug the XMC7100D using an external programmer, such as MiniProg4.
14.	XMC7100D reset button (SW1)	Resets XMC7100D. It connects the XMC7100D reset (XRES) pin to ground.
15.	Potentiometer (R77)	10 kΩ potentiometer connected to a XMC7100D pin P6[0]. This pin has shared connection to A0 of Arduino-compatible analog header J2 to simulate sensor output to XMC7100D.

Sl. No.	Peripheral	Description
16.	Potentiometer connection jumper (J13)	Connects the XMC7100D VDDA supply to the potentiometer.
17.	XMC7100D extended I/O headers (J21, J23)	These headers provide connectivity to XMC7100D 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 XMC7100D by default. These connectors are not populated by default.
18.	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.
19.	Analog-IN header compatible with Arduino Uno R3 (J2)	Bring out pins from XMC7100D to interface with the Arduino-compatible shields. Some of these pins are multiplexed with onboard peripherals and are not connected to XMC7100D by default. For detailed information on how to rework the kit to access these pins, see Table 3.
20.	XMC7100D user buttons (SW2, SW4)	Provide an input to XMC7100D. Note that by default the button connects the XMC7100D pin to ground when pressed, so you need to configure the XMC7100D pin as a digital input with resistive pull-up for detecting the button press. The SW4 button also provides a wakeup source from hibernate mode of the device.
21.	XMC7100D user LEDs (LED1 and LED2)	The user LEDs can operate at the entire operating voltage range of XMC7100D. The LEDs are active LOW; therefore, pins must be driven to ground to turn ON the LEDs.
22.	Shield2Go- S2GO2 (J16 & J18)	Infineon's shield2go-compatible header to support sensor functionality working with I2C, UART or SPI interface.
23.	Shield2Go- S2GO1 (J20 & J22)	Infineon's shield2go compatible header to support sensor functionality working with I2C, UART or SPI interface.
24.	XMC7100D Microcontroller (XMC7100D-F176K4160AA – U1)	XMC™ MCU device used on the kit.
25.	Mikro-Bus compatible header (J19 & J17)	MIKROE's mikroBUS-compatible header to support sensor functionality working with I2C, UART or SPI interface.
26.	RJ45 Ethernet connector (U16)	RJ45 ethernet connector port to connect kit to the Ethernet network. Not loaded by default.
27.	Ethernet physical layer (PHY) transceiver (U12)	Ethernet PHY. Not loaded by default.
28.	25 MHz crystal for Ethernet transceiver (Y3)	Crystal oscillator for Ethernet PHY. Not loaded by default.
29.	50 MHz crystal for Ethernet transceiver (U17)	Oscillator for Ethernet PHY. Not loaded by default.
30.	CAN FD interface connector (J14)	Connector to connect kit to the CAN / CAN FD network.

Sl. No.	Peripheral	Description
31.	CAN FD transceiver (TLE9251VLE – U5)	CAN FD transceiver
32.	Digital I/O headers compatible with Arduino Uno R3 (J3, J4)	Bring out pins from the XMC7100D to interface with shields compatible with Arduino. Some of these pins are multiplexed with onboard peripherals and are not connected to the XMC7100D by default. For detailed information on how to rework the kit to access these pins, see Table 3.
33.	KitProg3 (PSoC™ 5LP) programmer and debugger (CY8C5868LTI-LP039 – U3)	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.

For more details on various hardware blocks, see Hardware functional description.

3.2 Board support package (BSP) selection

The XMC7100 Evaluation Board comes with a KIT_XMC71_EVK_LITE_V1 board support package to build and run code examples on the kit by following the instructions provided in the “Create an application” section of the ModusToolbox™ user guide.

3.3 KitProg3: onboard programmer and debugger

The XMC7100 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.

3.3.1 Programming and debugging using ModusToolbox™

1. Connect the board to the PC using the provided USB cable through the KitProg3 USB connector, see Figure 8.If you are connecting it to the PC for the first time, it enumerates as a USB composite device.
2. KitProg3 on this kit operates in CMSIS-DAP Bulk mode. 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.

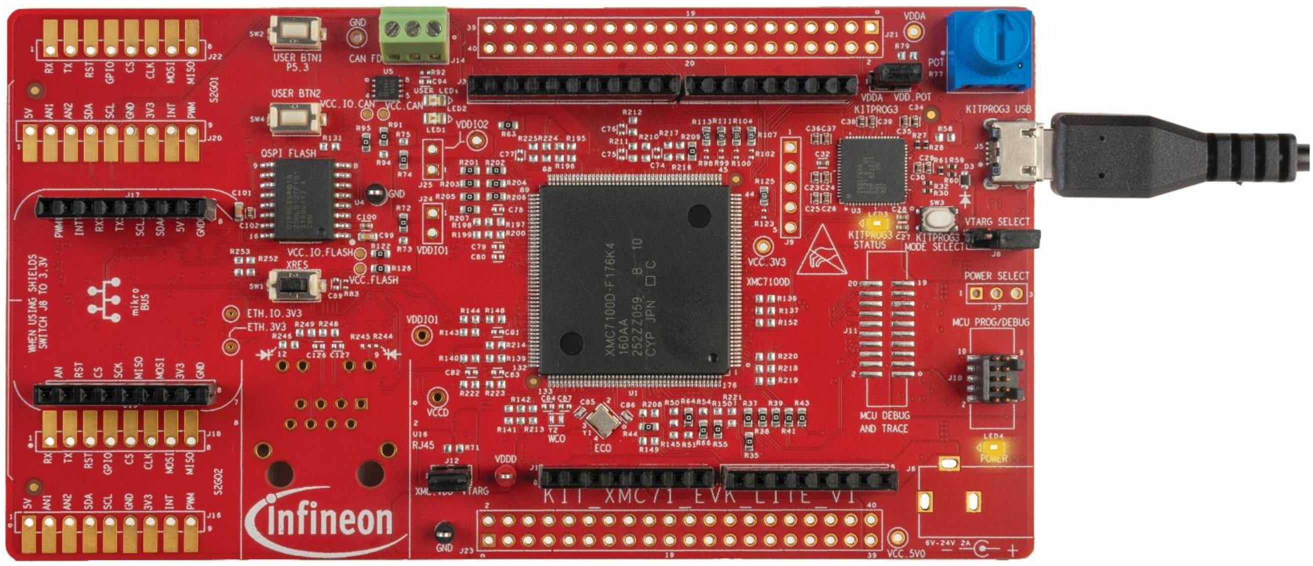


Figure 8 Connect USB cable to USB connector on the board

3. In the Eclipse IDE for ModusToolbox™, import the desired code example (application) into a new workspace.
   a) Click on **New Application** from **Quick Panel**.

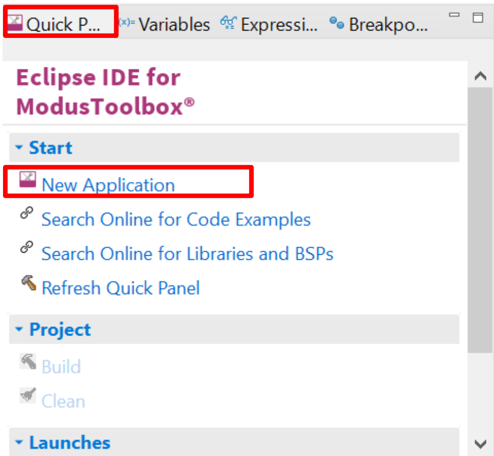


Figure 9 Create new application

b) Select the BSP in the “Choose Board Support Package” window

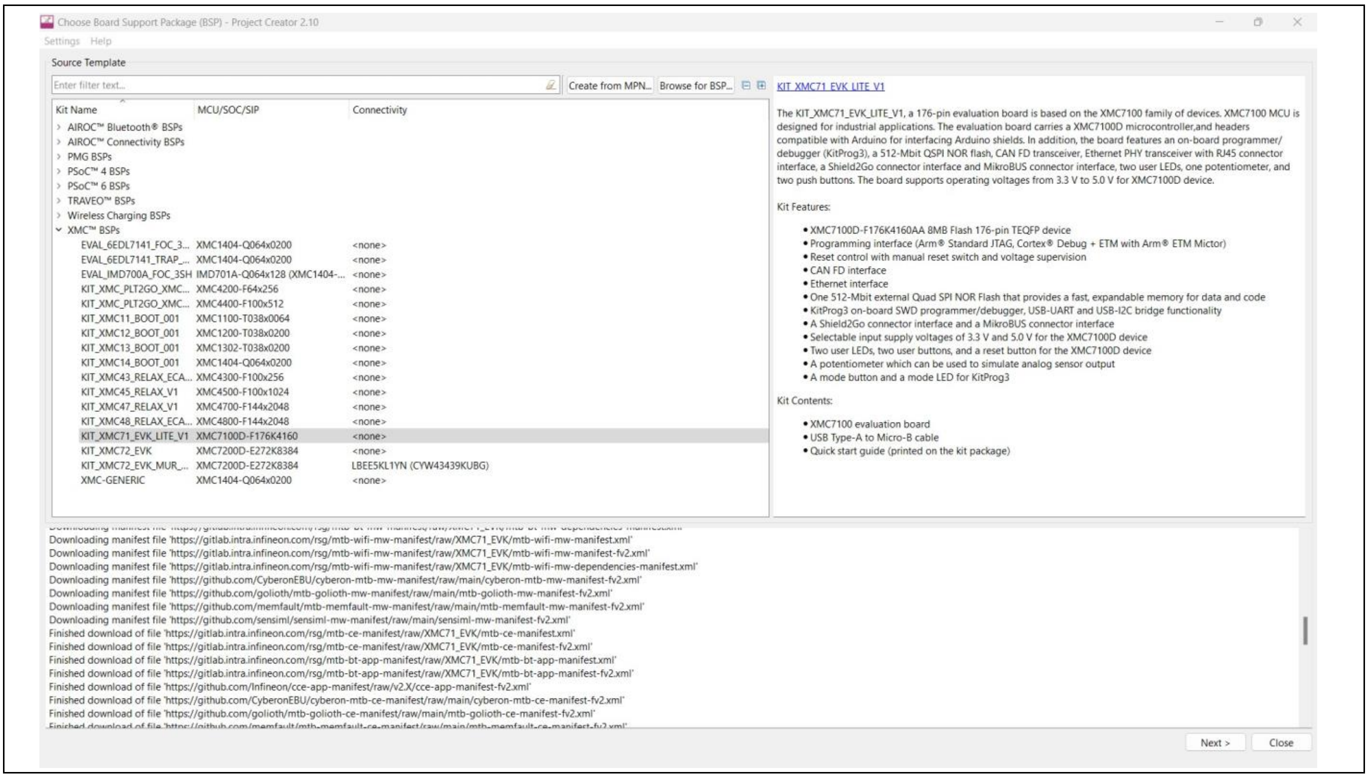


Figure 10 Board support package in the new application window

c) and click **Next**.

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

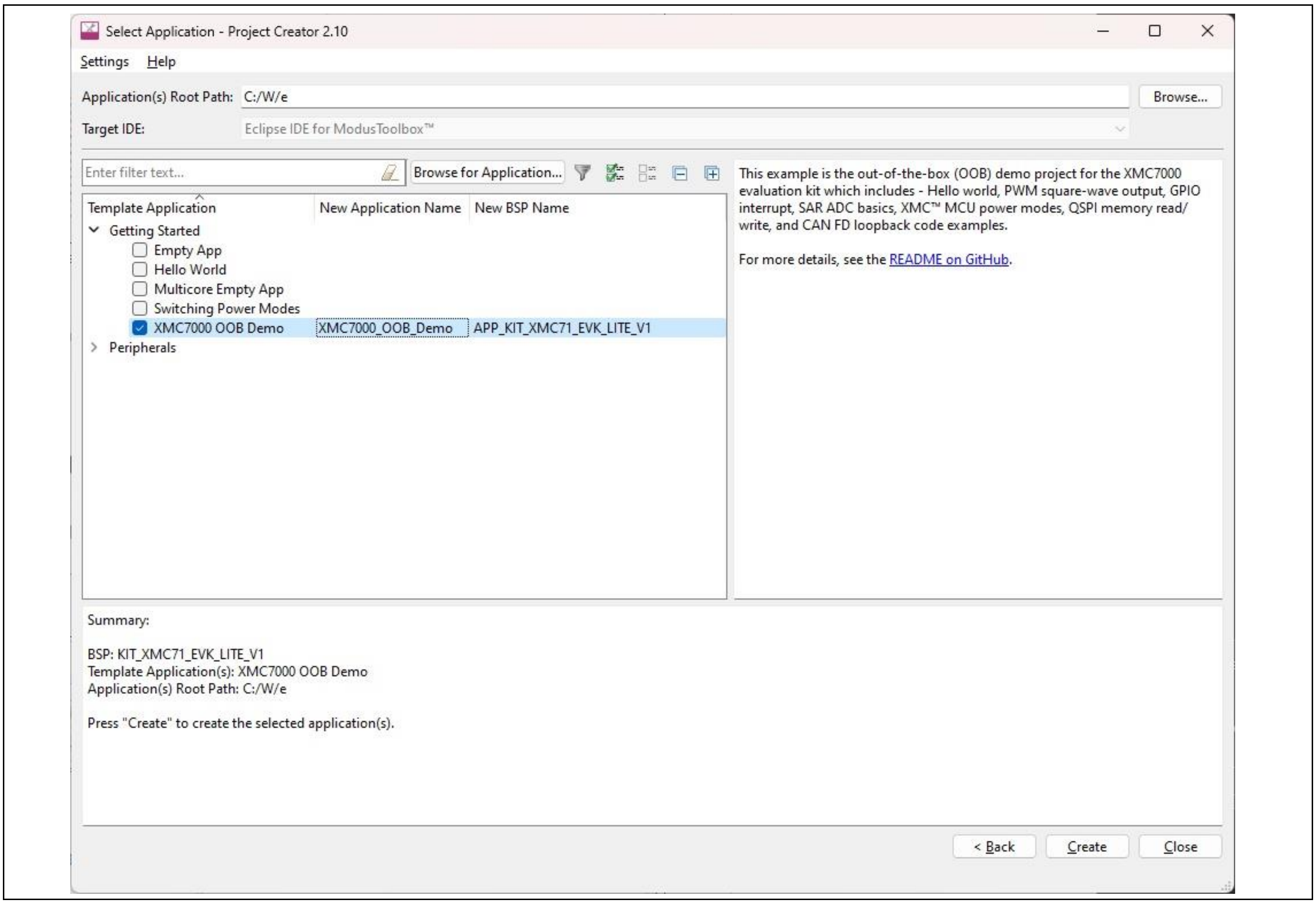


Figure 11 Select application window

4. To build and program a XMC7100D 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, see Figure 12.

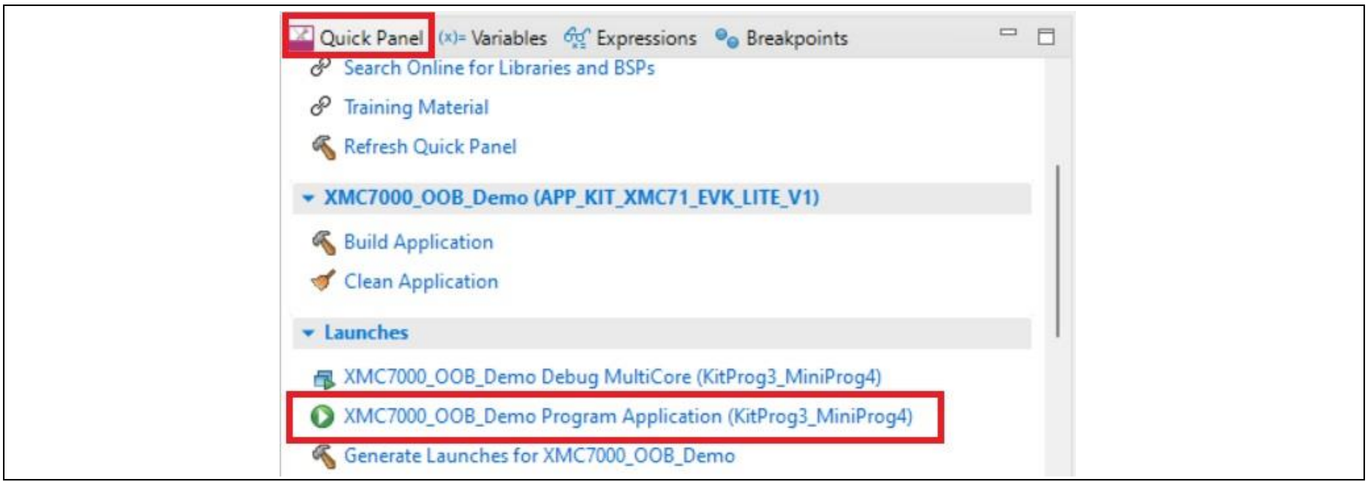


Figure 12 Programming in the ModusToolbox™ software

5. ModusToolbox™ software has an integrated debugger. To debug a XMC7100D 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.

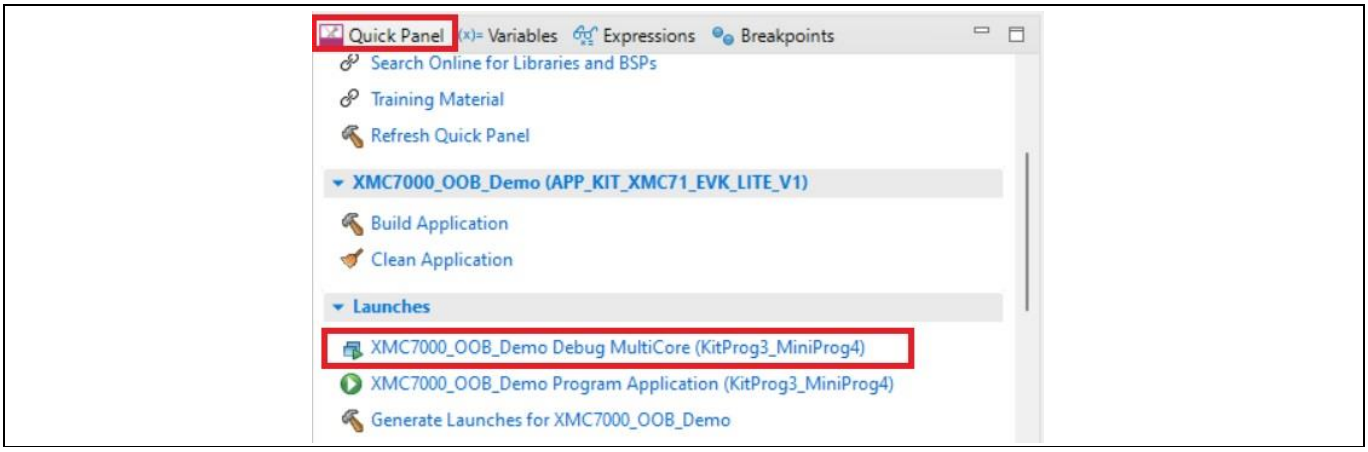


Figure 13 Debugging in ModusToolbox™

3.3.1.1 Using the OOB example – XMC7000 MCU: OOB demo

By default, the XMC7100 Evaluation Board is programmed with the *XMC7000 MCU: OOB demo* code example. The following 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.

1. Connect the board to the 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.
3. Press the reset button (SW1) on the board and confirm that terminal application displays the code example title and other text as shown in Figure 14.

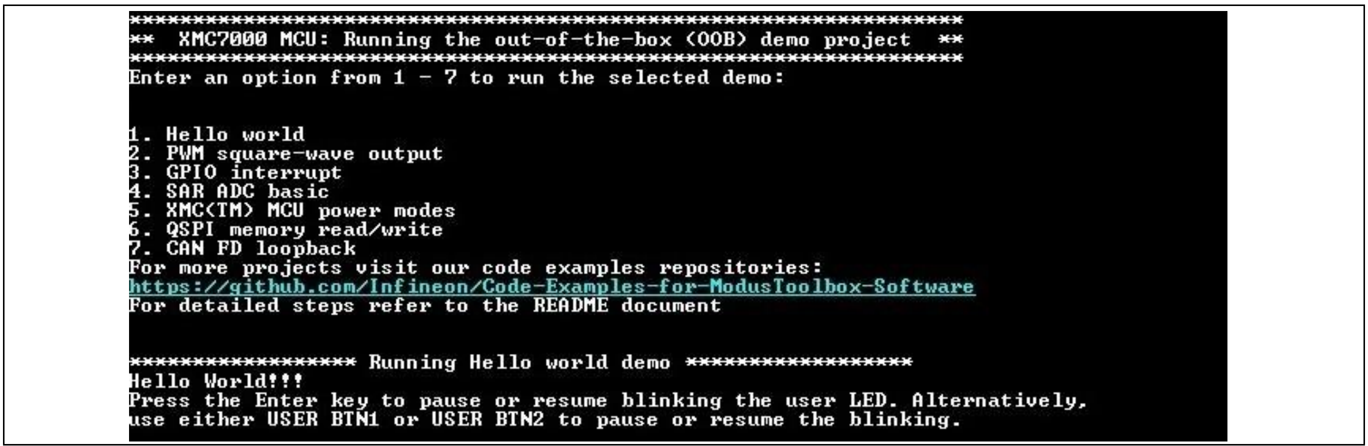


Figure 14 OOB demo project message in the terminal

4. Confirm that the kit LEDs blinks at 1 Hz in a sequence.
5. If you press the **Enter** key, the LEDs stop blinking; the terminal displays the message “LED blinking paused”.
6. If you repress the **Enter** key, the LEDs resume blinking at 1 Hz; 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.
7. Select the options from 1 to 7 as shown in Figure 14 to run the selected demos.

3.3.2 USB-UART bridge

KitProg3 on the XMC7100 Evaluation Board can act as a USB-UART bridge.

The UART RX and TX pins of KitProg3 are connected to the XMC7100D UART pins as follows:

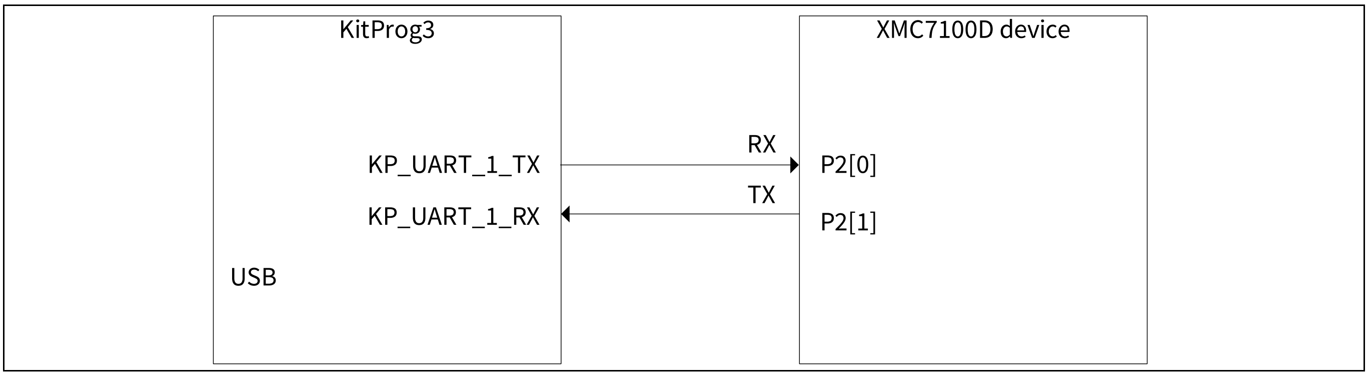


Figure 15 UART connection between KitProg3 and XMC7100D

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

3.3.3 USB-I2C bridge

KitProg3 can function as a USB-I2C bridge and communicate with an I2C master, such as the Bridge Control Panel (BCP). The I2C lines on the XMC7100D 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.

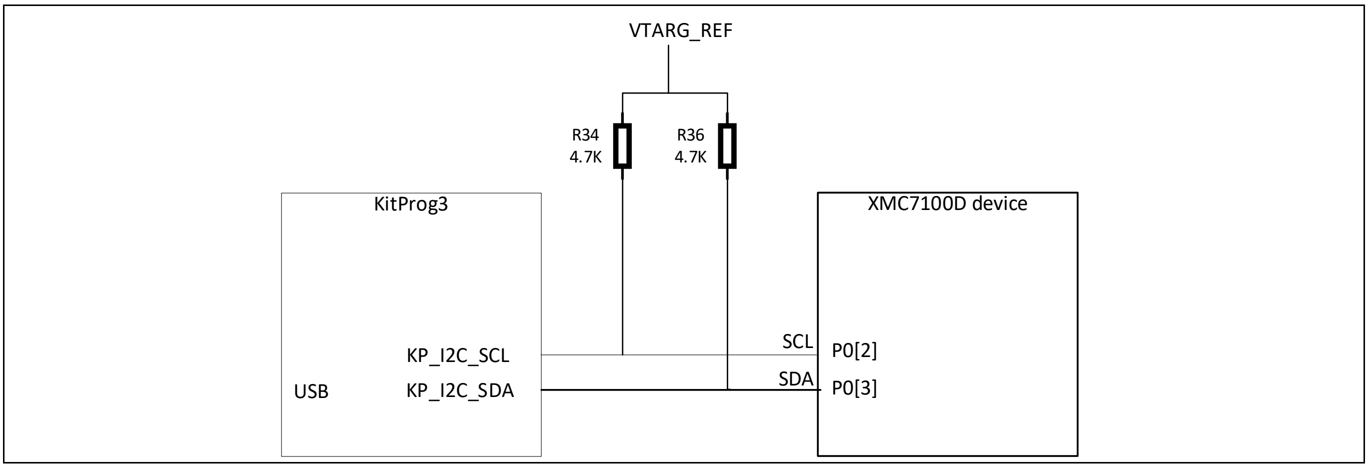


Figure 16 I2C connection between KitProg3 and XMC7100D

4 Hardware

4.1 Schematics

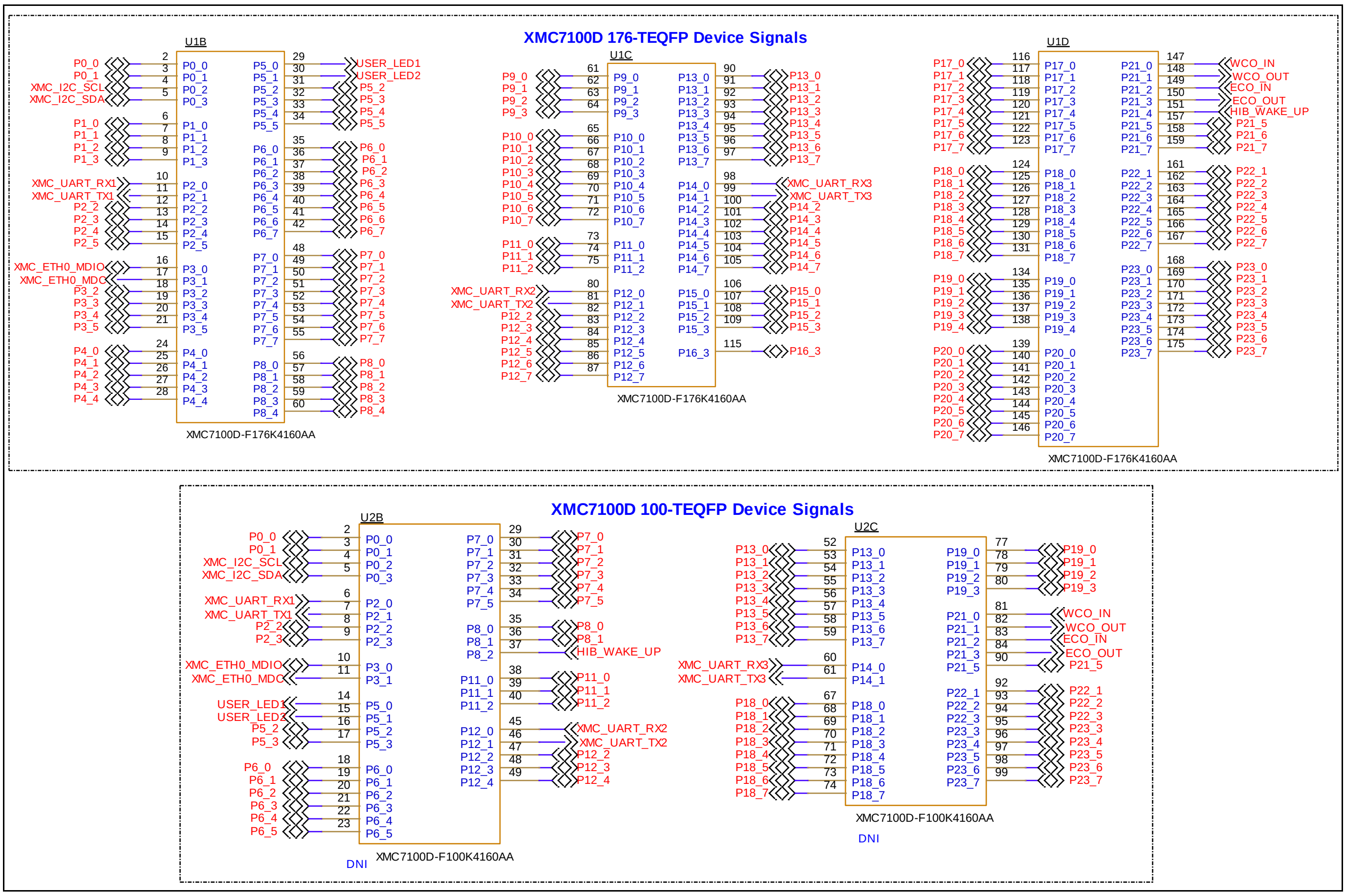
See the schematic files available in the kit webpage.

4.2 Hardware functional description

4.2.1 XMC7100D (U1)

XMC7100 is a family of XMC7000 MCUs with industrial applications. XMC7100 has two Arm® Cortex®-M7 CPUs for primary processing, and an Arm® Cortex®-M0+ CPU for peripheral and security processing. Single/Dual 250 MHz Arm® Cortex®-M7 and 100 MHz Cortex®-M0+. These devices contain embedded peripherals supporting CAN FD and Ethernet. XMC7100 devices are manufactured on an advanced 40-nm process. XMC7100 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 XMC7100D datasheet.



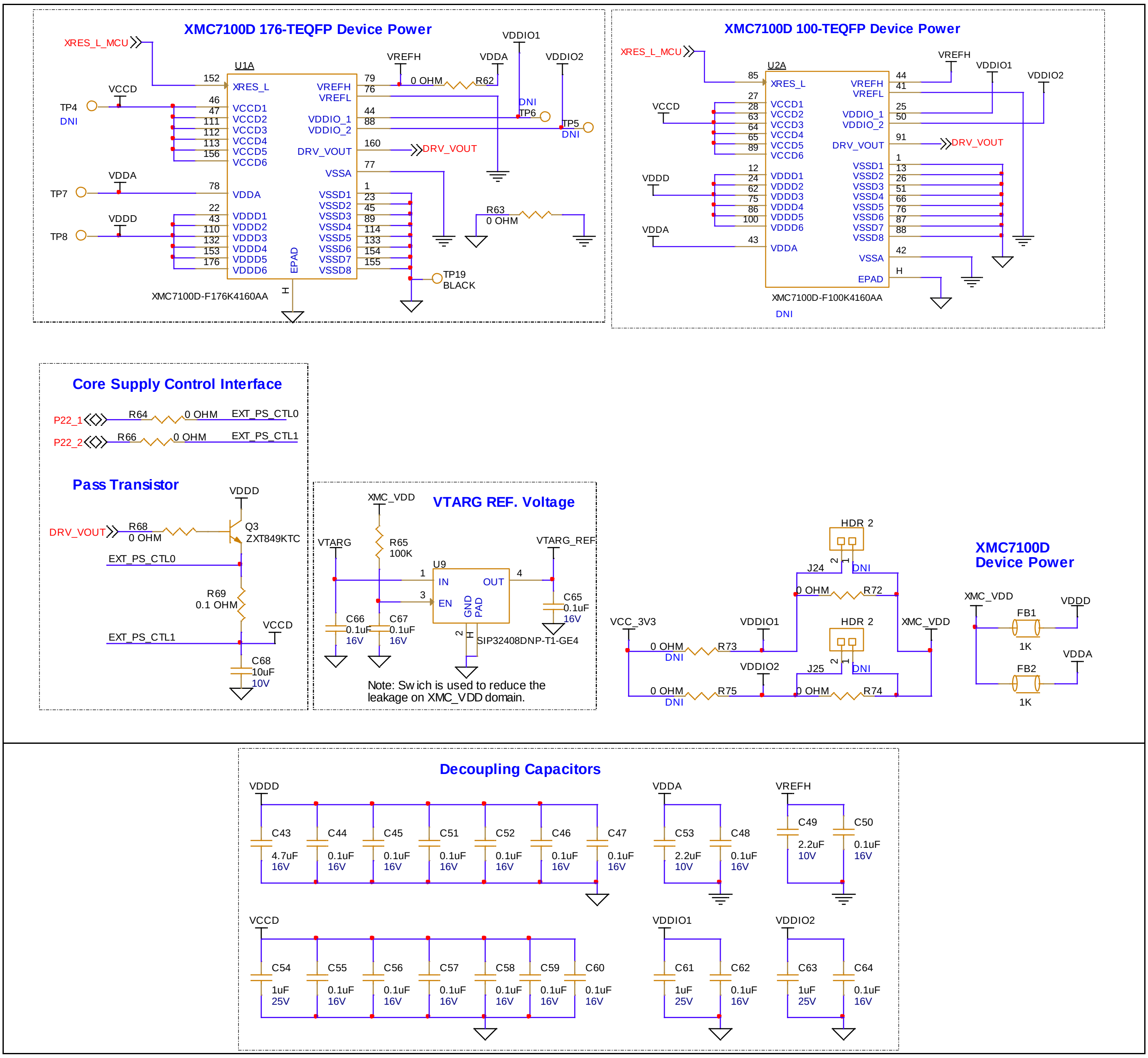


Figure 17 XMC7100D

4.2.2 PSoC™ 5LP-based KitProg3 (U3)

An onboard PSoC™ 5LP (CY8C5868LTI-LP039) is used as KitProg3 to program and debug XMC7100D. 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 XMC7100D.

Visit the PSoC™ 5LP webpage and for more information, see the CY8C58LPxx family datasheet .

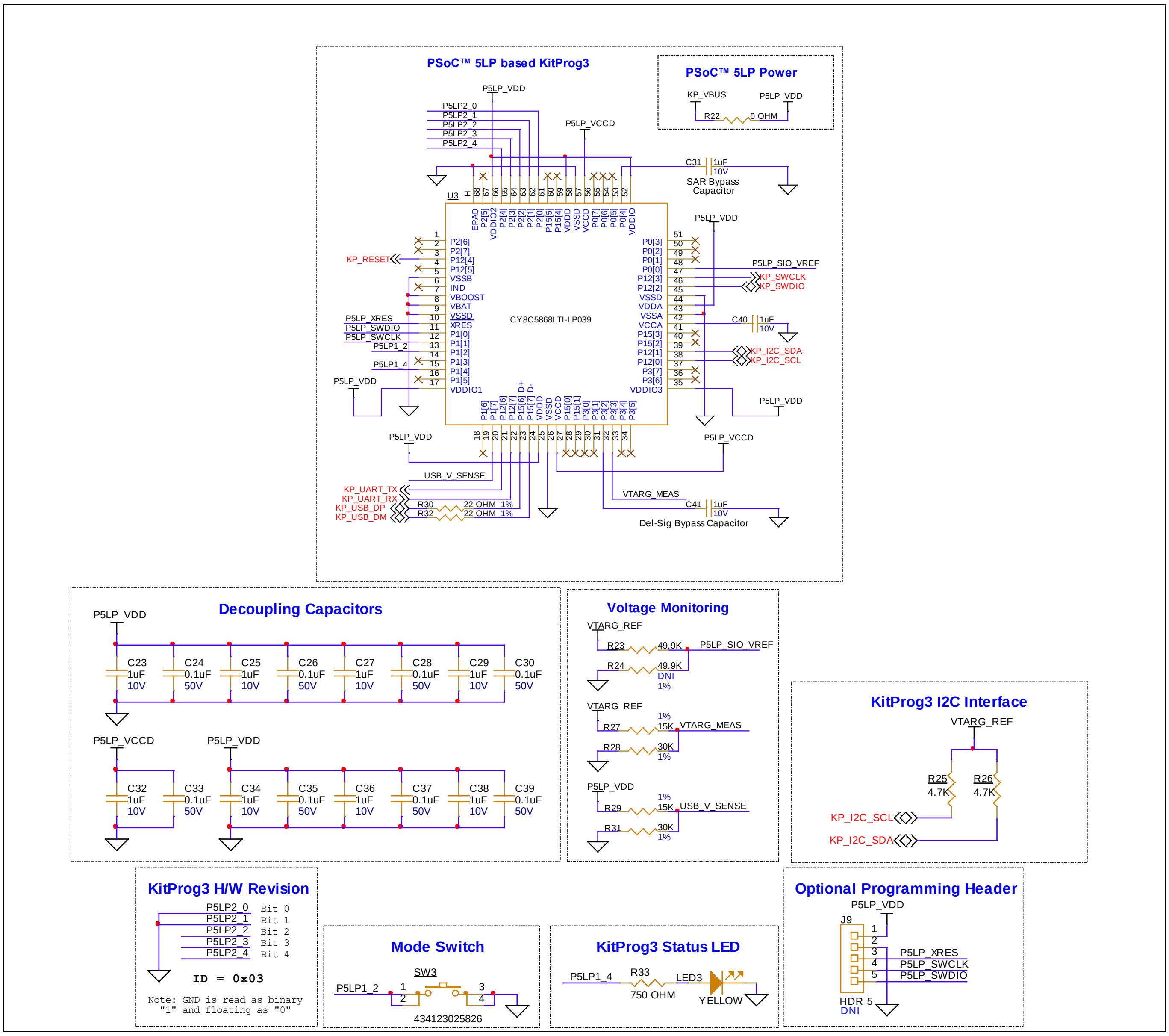


Figure 18 PSoC™ 5LP-based KitProg3 (U3)

4.2.3 Serial interconnection between PSoC™ 5LP and XMC7100D

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 19. The USB-serial pins of the PSoC™ 5LP device are hard-wired to the I2C/UART pins of the XMC7100D device. These pins are also available on the I/O headers compatible with Arduino Uno R3.

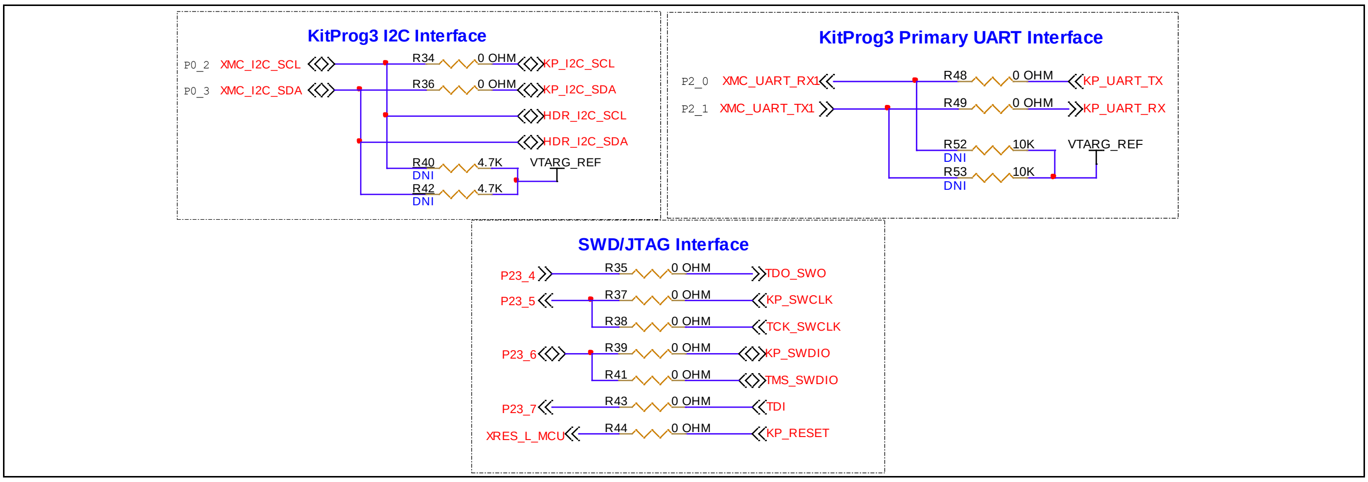


Figure 19 Programming and serial interface connections

4.2.4 Programming/debugging headers

The XMC7100 Lite Evaluation Board is designed with dedicated headers for programming the XMC7100 device. A 10-pin SWD/JTAG header allows you to program the target device using the SWD or JTAG interface. A 20-pin ETM standard header is provided for trace debugging. Trace output from the XMC7100 device can be used for debugging purposes. This 20-pin header is also connected to the SWD and JTAG signals, and this helps a create a single programming header concept for the programming and debugging of the XMC7100 device.

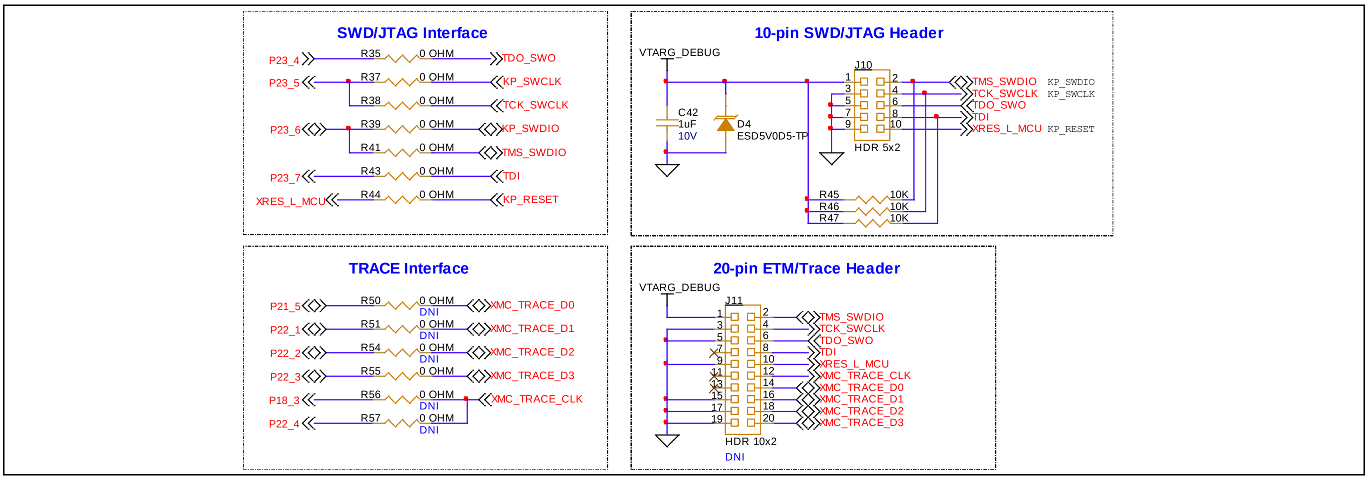


Figure 20 Programming/debugging headers

4.2.5 Power supply system

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

- 5 V from the onboard USB Micro-B connectors **(J5)**
- 7 V–24 V from the external power supply through the VIN barrel jack (**J6- not populated**) or from a J1 header compatible with Arduino

Note: Ensure to power the board with an external DC supply via a VIN connector for applications that require more than 500 mA of current. Modern systems with USB 3.0 should support up to 900 mA of source current through the USB interface.

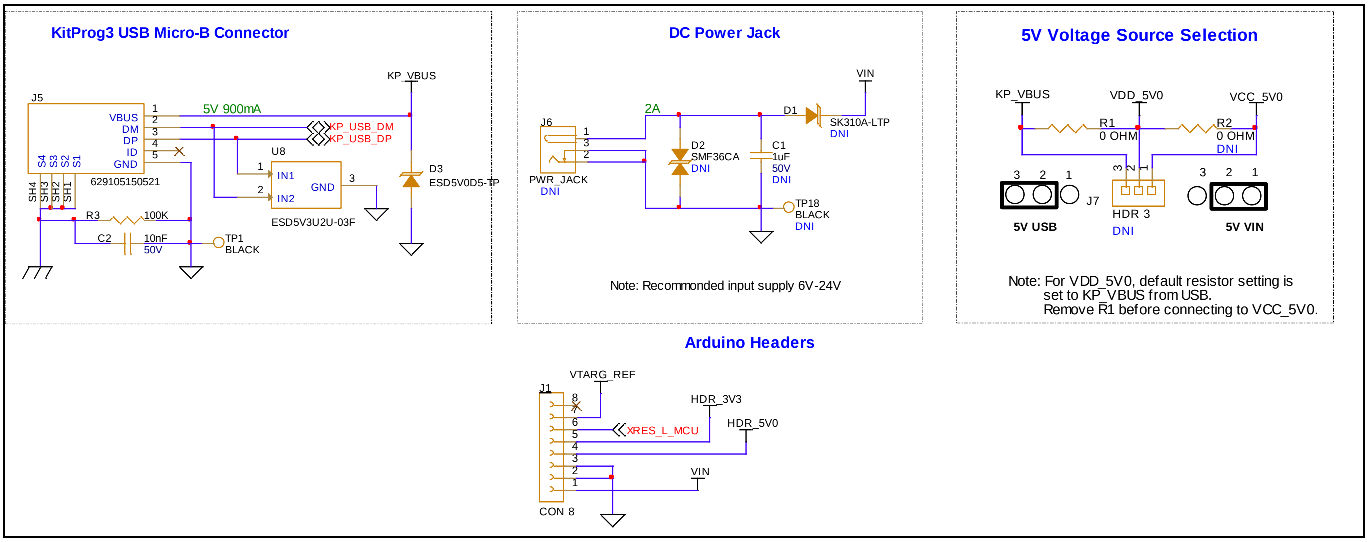


Figure 21 Power supply input and ORing

4.2.5.1 Voltage regulators

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

- VDDD, VDDA, VDDIO1, VDDIO2, VREFH (XMC7100D) – 3.3 V or 5 V

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

A buck regulator (U7) is used on the kit to generate a stable 3.3-V output. An optional power supply section is included in the design for generating stable 5 V from an external VIN supply. By default, the buck regulator for 5 V is not populated on the board.

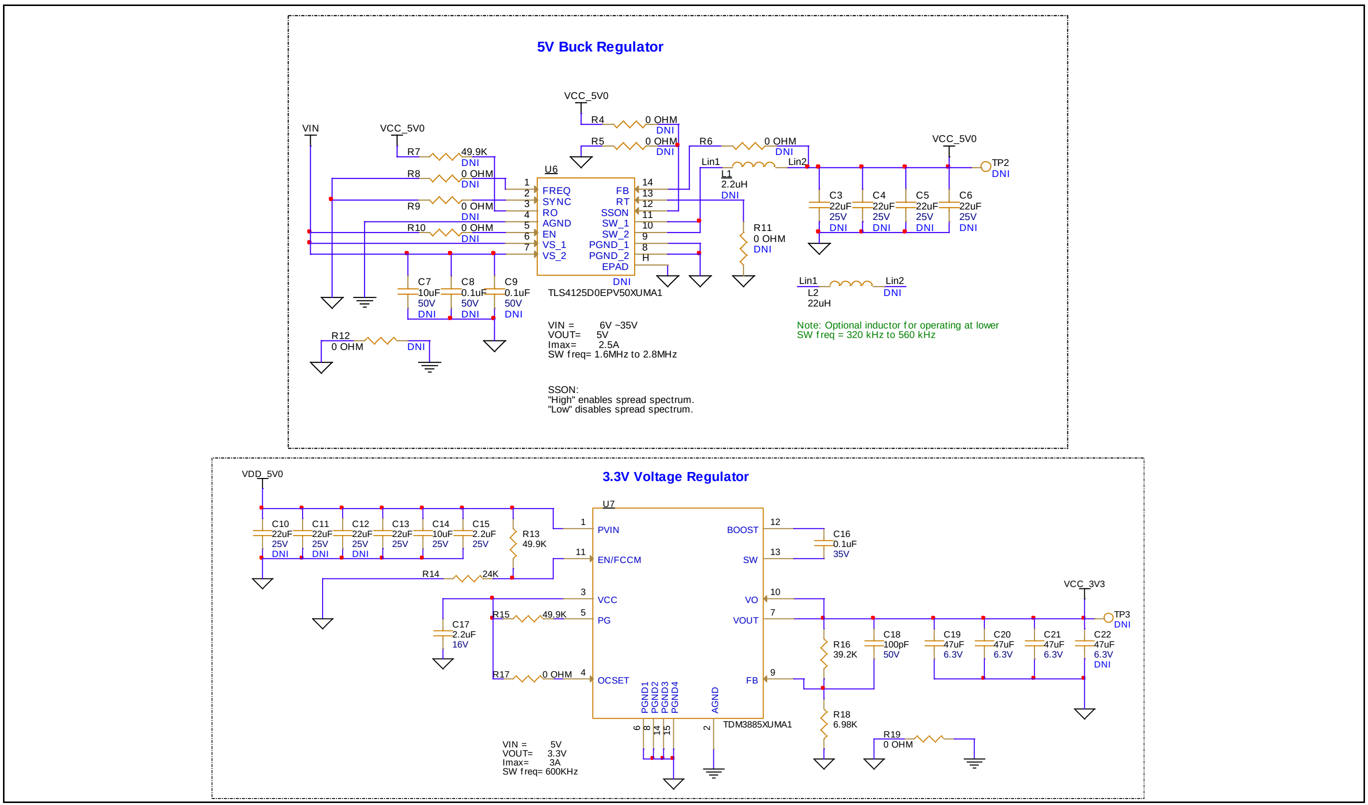


Figure 22 Voltage regulators

4.2.5.2 Voltage selection

Headers are provided on the board for selection source voltage for the target device. J8, a 3-pin header allows user to configure the target voltage to set either to 3.3 V or 5 V operation.

Note: Default jumper position (J8) for XMC7100D device is set to 3.3 V. While using sensor shields on S2GO or mikroBUS headers, keep the J8 position to 3.3 V. Configuring target voltage to 5 V when using shields may cause damage to sensor shields on S2GO or mikroBUS interface.

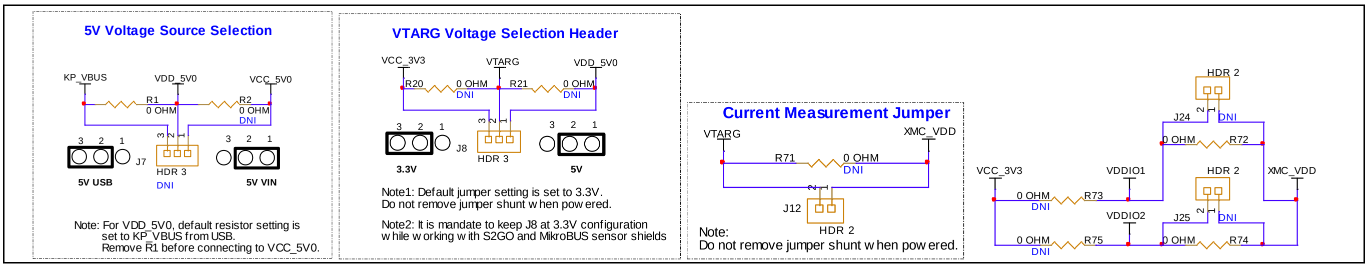


Figure 23 Voltage selection

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.

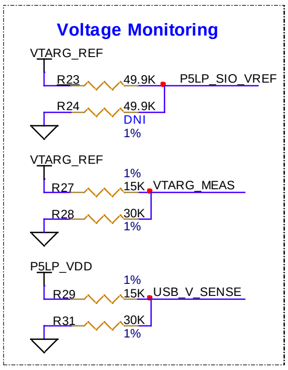


Figure 24 Voltage monitoring

The I/O voltages for XMC7100 can be selected for VDDD, VDDA, VDDIO1 and VDDIO2 I/O domains that can be switched between VCC_3V3 or VDD_5V0 voltage levels. VDDD and VDDA voltage can be switched between 3.3 V and 5 V by placing the jumper shunt in an appropriate position on header J8. For changing the voltage on I/O levels (VDDIO1 & VDDIO2) user need to have rework done on the board by populating respective 0-ohm resistors to switch between 3.3 V and 5 V target power. 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.

4.2.5.3 Current measurement headers

The current of the following domains (see Table 5) has dedicated 2-pin headers to facilitate an easy current measurement using an ammeter across the pins.

Table 5 Current measurement headers

Domain name	Header reference designator	Load by default
XMC_VDD	J12	Y
VDDIO1	J24	N
VDDIO2	J25	N
Resistor configuration to measure current on respective I/O domains		
Domain name	**Resistor reference designator**	**Loaded by default**
VDDIO1	R72	Y
VDDIO2	R74	Y

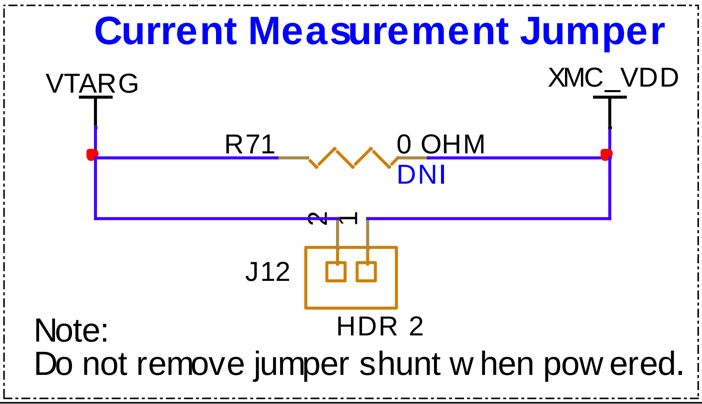


Figure 25 Current measurement headers

Note: When measuring the XMC_VDD current, make sure that the jumper shunt from J13 is removed. This will disconnect the potentiometer from VDDA and remove the leakage.

4.2.6 I/O headers

The EVK is provided with header interfaces to evaluate sensors and shields with the different form factors such as Arduino, mikroBUS from MikroE and Shield2Go from Infineon, which are supported on the EVK.

Some of the IO signals from the XMC7100 device are multiplexed by sharing between Arduino, mikroBUS and the shield2Go headers.

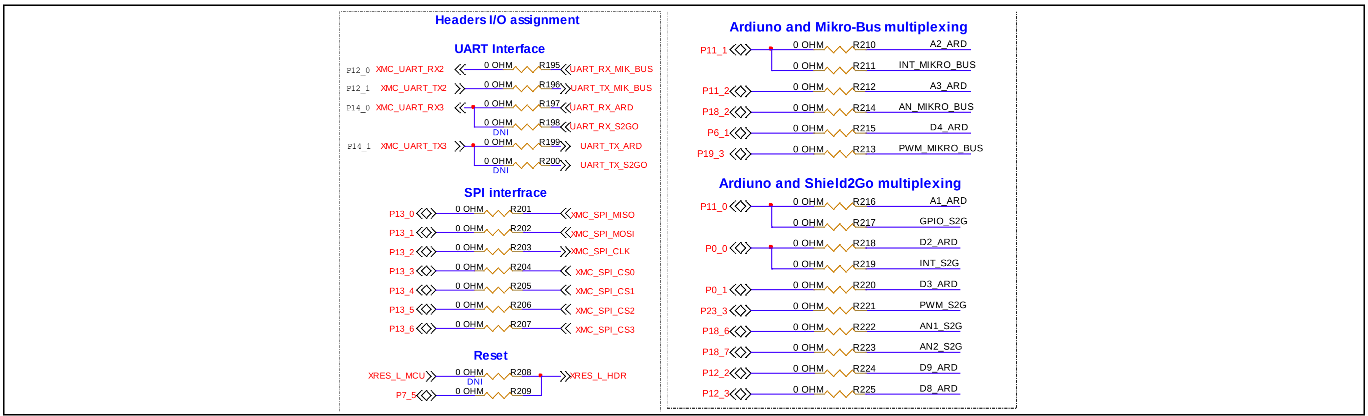


Figure 26 Headers I/O assignment

4.2.6.1 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 the XMC7100 operating voltage is between 2.7 V and 5 V.

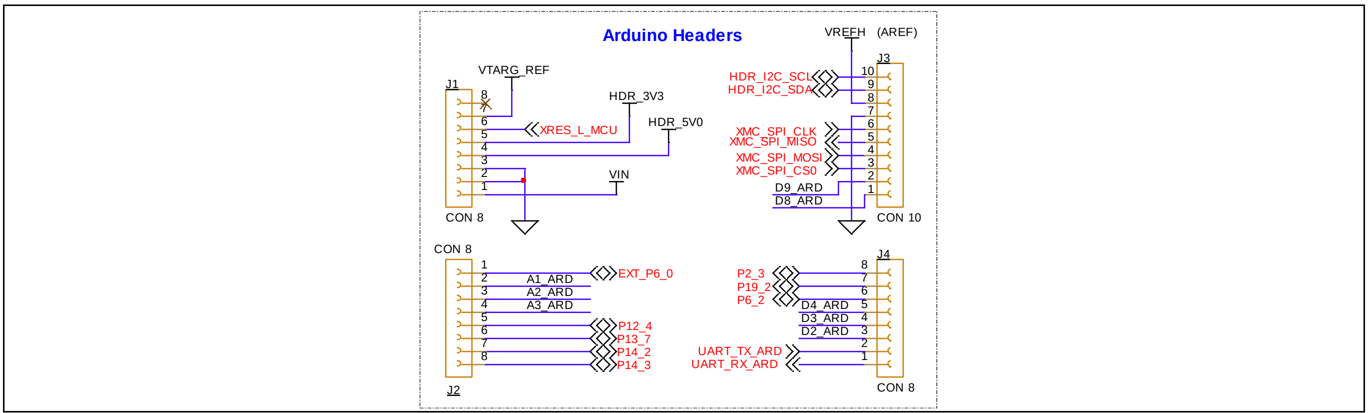


Figure 27 Headers compatible with Arduino

4.2.6.2 Infineon's ShieldtoGo sensor shields compatible headers

The XMC7100 Evaluation Board has two header footprints to interface with Infineon's standard ShieldtoGo sensor shields. You can either assemble the shield directly on the board or assemble suitable headers for connecting the shields. All the Infineon S2GO shields are equipped with one Infineon IC and come with a ready-to-use Arduino library. Infineon's Shield2Go boards are equipped with featured Infineon ICs and provide a standardized form factor and pin layout for fast orientation. All boards come with solderless connectors, allowing designers to stack the boards instead of soldering them.

Note: *Infineon's standard S2GO sensors are voltage tolerant up to 4V. User must have to configure the target voltage to 3.3V (J8) before working with sensor shields on S2GO headers (S2GO1 and S2GO2).*

Shield2go sensor specific to "S2GO MEMSMIC IM69D" is designed with bottom assembly of digital MEMS microphones. User need to have S2GO headers mounted on XMC7100 Evaluation Board for interfacing with this sensor shield.

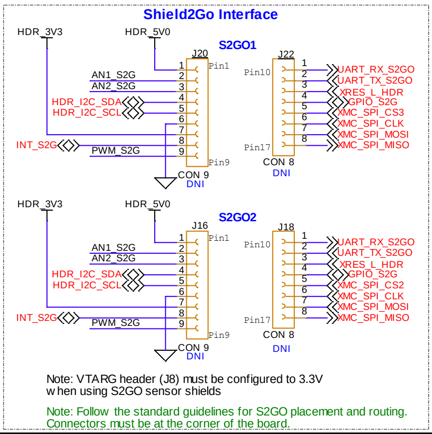
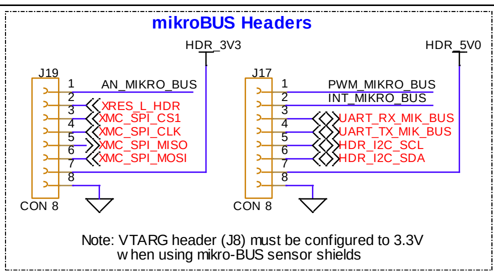


Figure 28 Headers compatible with S2GO

4.2.6.3 mikroBUS shield compatible headers

The XMC7100 Evaluation Board has headers populated to interface with MIKROE standard mikroBUS sensor shields. You can directly mount the respective mikroBUS sensor shields to evaluate the functionality. The mikroBUS shields are available with different working voltage levels such as 3.3 V and 5V. Based on operating voltage and tolerances of the shield sensors, the XMC7100 evaluation board must be configured (J8) for suitable operating voltage.

Note: *MIKROE's standard mikroBUS shields are available with different operating voltages. Configure the target voltage (J8) accordingly before working with sensor shields on mikroBUS header.*



4.2.6.4 XMC7100D I/O headers (J21, J23)

These headers provide connectivity to XMC7100D GPIOs that are not connected to any other interface. A few of these pins are multiplexed with Arduino-compatible headers. By default, these GPIO connectors are not populated. Signals on GPIO expansion headers are oriented by segregating the ADC-supported pins as one group.

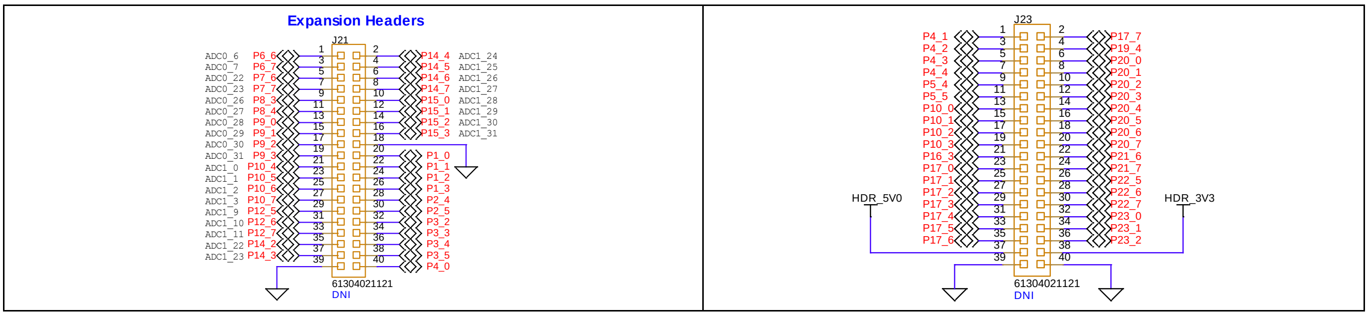


Figure 29 XMC7100 MCU GPIO headers

4.2.7 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 Mbps. 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. VIO power is connected to VTARG_REF power; therefore, depending on the jumper configuration (J8) on XMC7100 Evaluation Board, CAN transceiver will operate accordingly (3.3 V or 5 V).

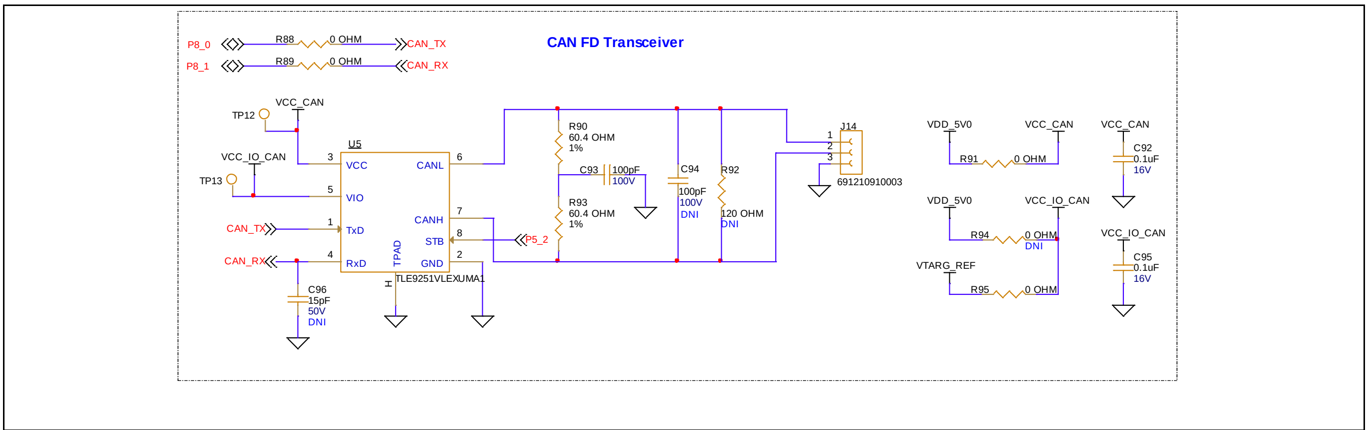


Figure 30 CAN FD Transceiver

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 P5[2] (STB pin) to high when measuring the XMC_VDD current for low-power mode.

4.2.8 Quad SPI NOR flash

The XMC7100 Evaluation Board has a 512-Mbit Quad SPI NOR flash memory (S25HL512T). The NOR flash is connected to the Quad SPI interface of the XMC7100D.

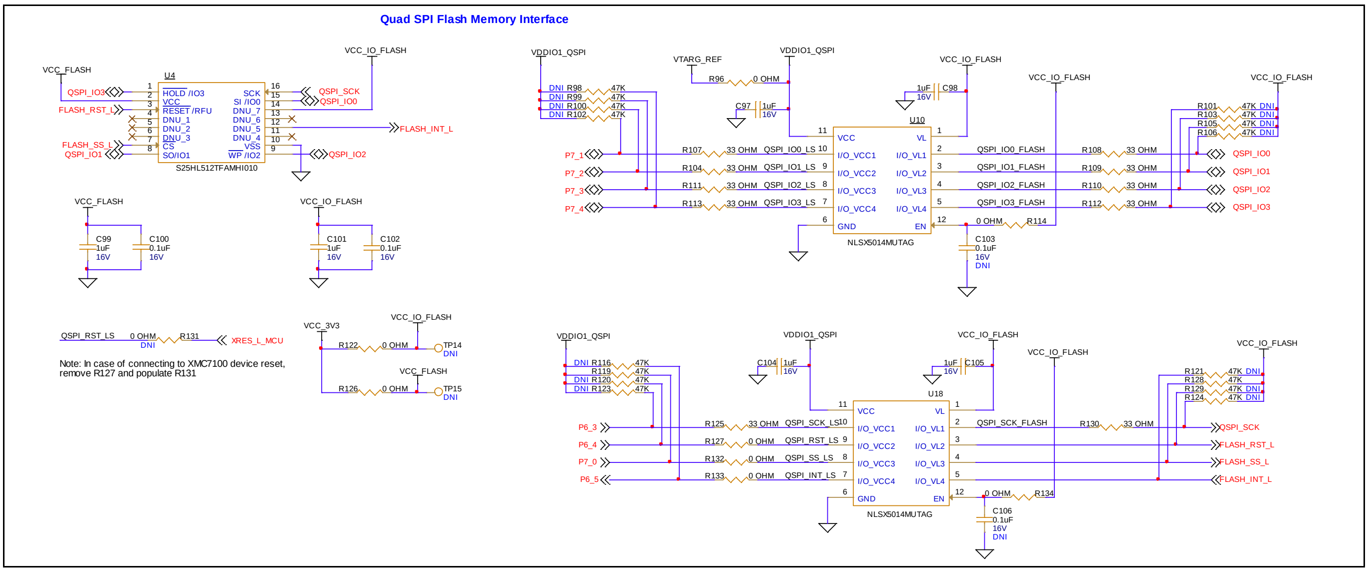


Figure 31 Quad SPI NOR flash

4.2.9 LEDs

LED3 (amber) indicates the status of KitProg3 (See the KitProg3 user guide for details). LED4 (amber) power LED indicates the status of power supplied to the board.

The board also has two user-controllable LEDs (LED1 and LED2) connected to XMC7100D pins for user applications.

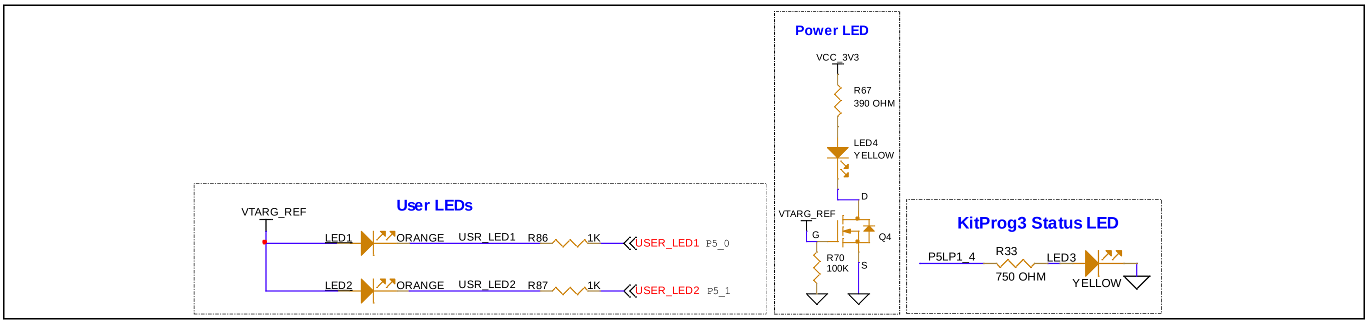


Figure 32 LEDs

4.2.10 Push buttons

The board has the following buttons:

- A reset button (SW1) is connected to the XRES pin of the XMC7100D device and is used to reset the device.
- Two user-controllable buttons (SW2 and SW4) are connected to pin P5[3] and P21[4] of the XMC7100D device respectively.
- A KitProg3 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.

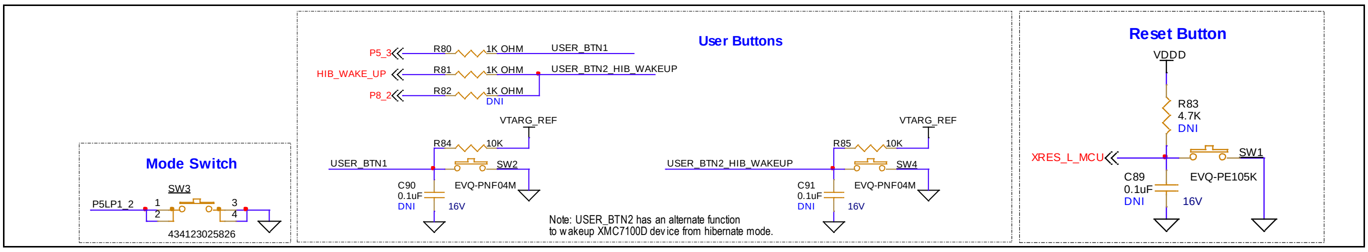


Figure 33 Push buttons

4.2.11 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 (not populated on the board by default).
- A 20 MHz crystal connected to P21[2] and P21[3] of the XMC7100D as the external crystal oscillator
- A 25 MHz crystal for standard Ethernet interface (not populated by default).
- A 50 MHz oscillator for Ethernet application (not populated by default).

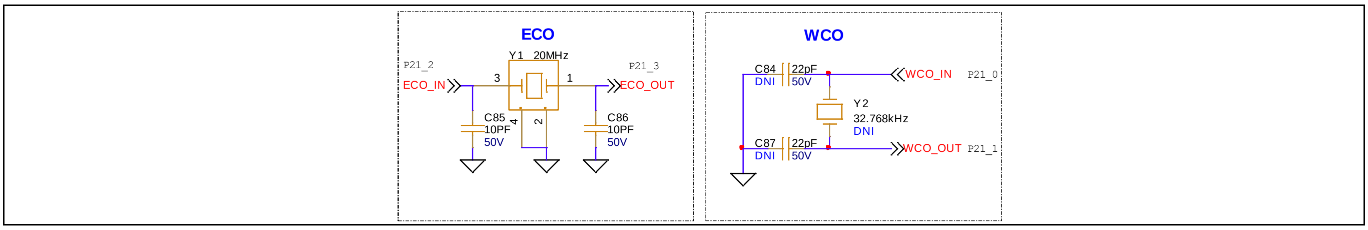


Figure 34 Crystal oscillators

4.2.12 Ethernet transceiver and RJ45 connector

TI-based DP83825IR is used on XMC7100 Evaluation Board for Ethernet validation. The whole Ethernet section including PHY, RJ45 connector and voltage level shifters are not loaded on the board by default. Ethernet interface is provided on the board for RMII mode operation. A physical layer controller must be used to operate at 10/100Mbps speeds. The ETH0 port from XMC7100 is interfaced with the Ethernet transceiver with signals necessary to operate in RMII mode. A RJ45 connector must be used which supports 10/100Mbps communication speeds. The RJ45 connector must be included with the necessary magnetics inbuilt to protect from external world. There are two clock references provided on the board for the Ethernet transceiver.

By default, a 25 MHz clock is connected and the 50 MHz crystal is not connected. PHY is capable of working with self generated 1.1 V core voltage along with 3.3 V supply and 3.3 V I/O voltage. XMC7100 device supports 2.7 V to 5 V operation due to which suitable voltage level shifters are introduced in the design to get desired voltage levels. With this configuration the PHY device operates in master mode and the XMC7100 device being in slave operating mode. The board has optional resistor strap options that allow you to manually change the configuration.

Ethernet to operate with XMC7100 device as master, a 50 MHz source is required between the PHY device and the MAC (XMC7100D).

Two options are provided on the EVK to generate the 50 MHz clock for the interface.

A 50 MHz generated from off-the-shelf oscillator device, or P18_0 from XMC7100 device is available to generate 50 MHz clock internally from the XMC7100 device.

The EVK is proved by testing Ethernet PHY as master and XMC7100 in the slave mode application.

For populating components required for Ethernet to work, see the Ethernet interface section.

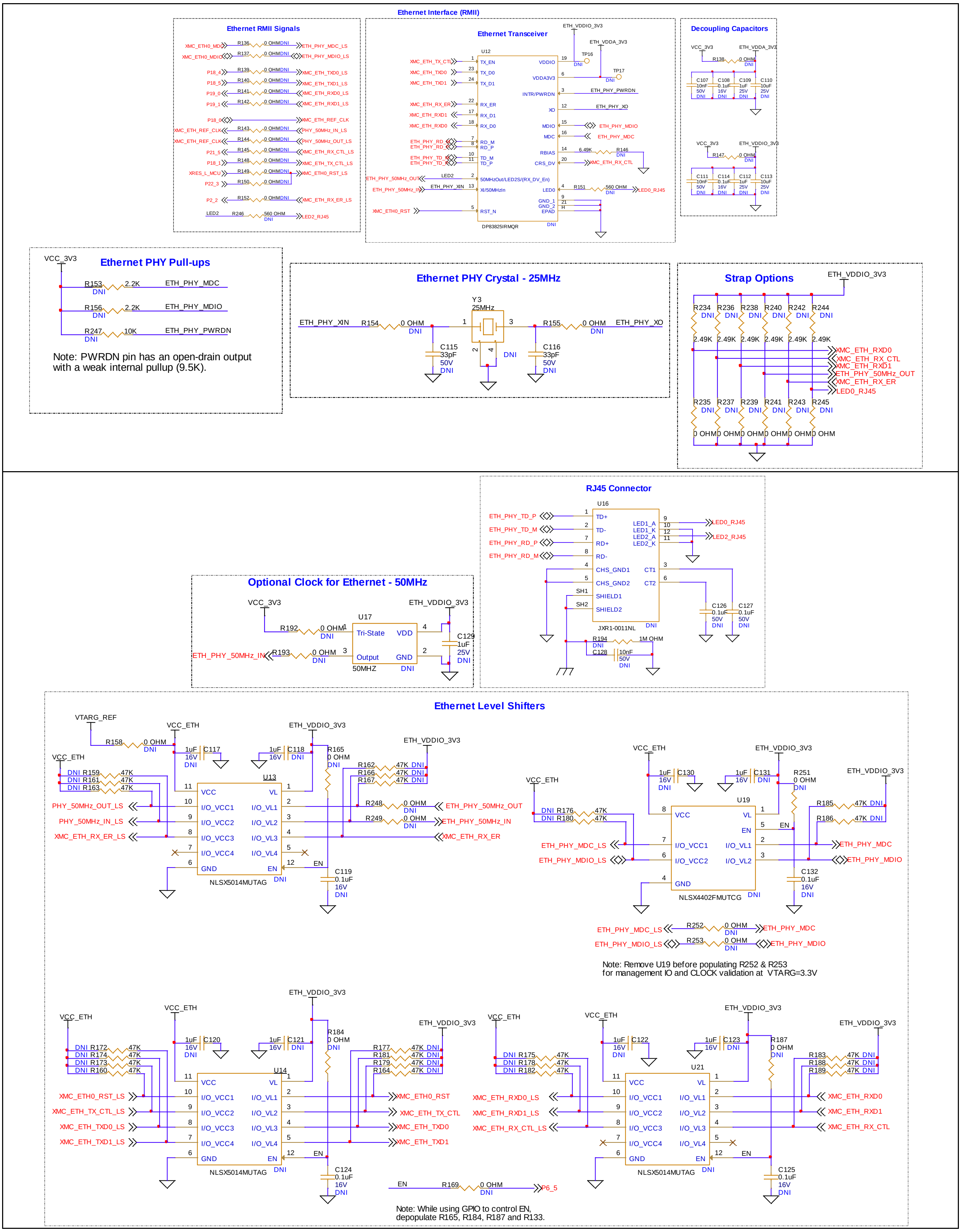


Figure 35 Ethernet transceiver and RJ45 connector

4.2.13 Potentiometer

By default, the board contains a 10k potentiometer connected to P6[0]. The same signal is connected to an Arduino analog header (J2.1). While using P6[0] on Arduino header for any external analog inputs, remove R78 resistor to isolate the signal from potentiometer. The fixed ends are connected to VDDA (VDD_POT through J13) and GND this may contribute to leakage current on the XMC_VDD supply. Remove the jumper J13 to disconnect power from the potentiometer when measuring the XMC_VDD current.

POT_AN_OUT signal is connected to VDDIO1 supply block that configurable for voltage levels between 3.3 V and 5 V. Consequently, ADC output readings from the XMC7100D vary according to the target voltage set at J8 header.

ADC output readings must increase when the potentiometer is rotated in clockwise direction and vice versa.

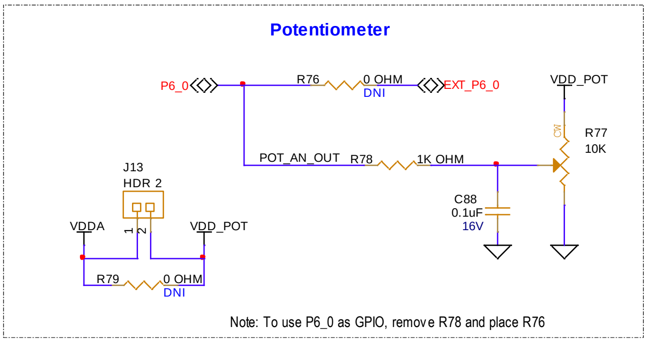


Figure 36 Potentiometer

4.2.14 Reverse voltage protection

A necessary reverse voltage protection circuit is provided on the kit to avoid damage caused by the reverse supply to the board. Any accidental supply coming in as input to the board will be prevented by the circuit protecting the XMC7100D device. A FET-based diode circuit is used as a reverse voltage protection on the board.

The corresponding reverse voltage protection circuit protects the power supply pins from all the headers, such as Arduino, Shield2Go, mikroBUS, and expansion headers. Programming headers are restricted to powering the XMC7100 Evaluation Board. A necessary reverse voltage protection circuit is added to the design to protect the target device in case of accidental power fluctuations from the external world.

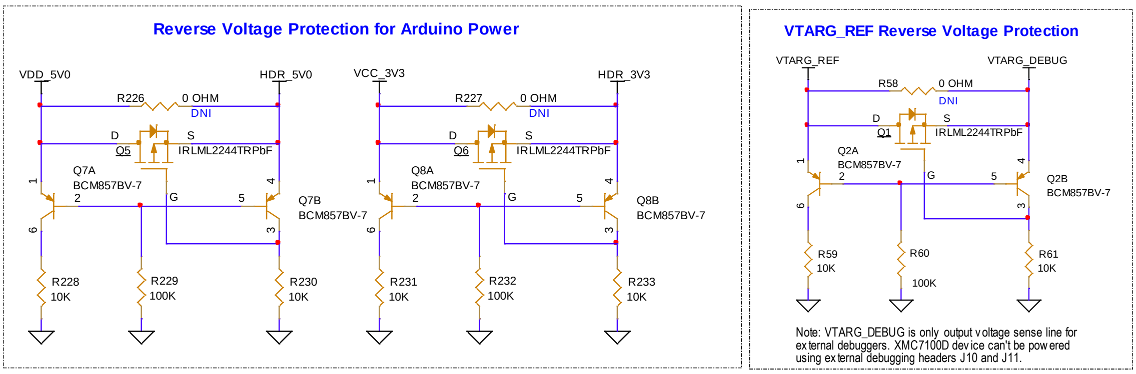


Figure 37 Reverse voltage protection circuit

4.3 Kit rework

4.3.1 XMC7100D-F100K4160AA Device Assembly

The XMC7100 Evaluation Board is designed for compatibility to have two devices testable in the XMC7000 series. The board has a dual-footprint option for 176-TEQFP and 100-TEQFP microcontrollers. 100-pin device footprint lies below the 176-TEQFP device. Either of the devices can be populated on the board for evaluation. On the existing XMC7100 evaluation boards, the 176-TEQFP device is populated by default. You need to have the respective rework done to evaluate the 100-TEQFP device.

Note: Assembly of micro controller is at its best when assembled with standard guidelines using machine assembly. Hand assembly/ soldering may have their own risk and criticalities and may intend to cause damage to the overall PCB.

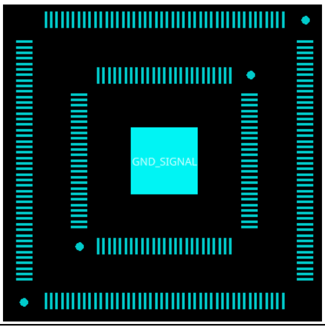


Figure 38 XMC™ device footprint

4.3.2 VIN Header

By default, the XMC7100 Evaluation Board comes with a powering object as a USB micro-B connector, which will be the source of power for the board when connected to the system, PC, or any standard USB powering object. A DC jack as an optional connector is provided for external VIN power using power supply adapters (6 V – 24 V). The VIN supply has to pass through a 5-V buck regulator to generate stable 5V output. By default, the overall interface to the VIN connector is not populated; you need to have respective components mounted on the board for VIN operation.

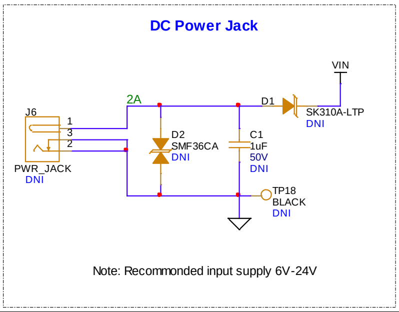


Figure 39 DC Power jack

4.3.3 5 V voltage source selection

The XMC7100 Evaluation Board has a source power selection header that directs which source needs to be used for powering up the board. Either from USB micro-B or from a VIN, a DC jack can be a source of power for the board. By default, J7 is not populated on the board, and resistor R1 is used to source the power by default from the USB micro-B connector. When using an external VIN supply, populate resistor R2 by removing R1 from the board.

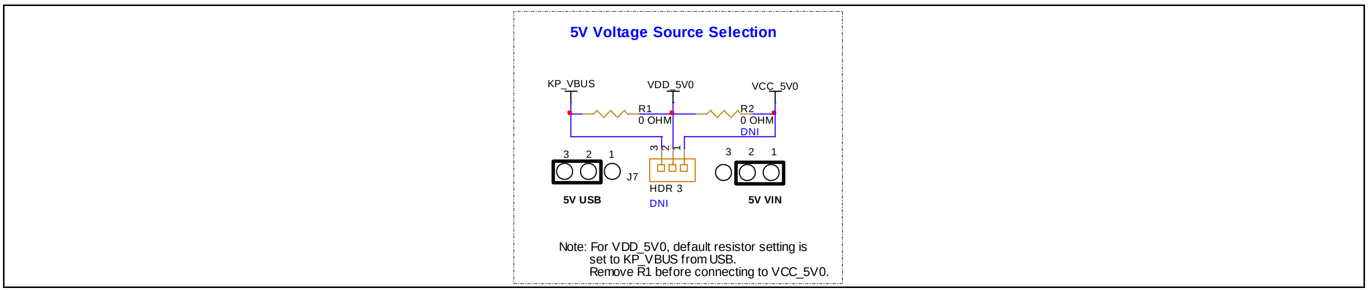


Figure 40 5 V voltage source selection

4.3.4 5 V buck regulator

Power supply section from VIN is directed to a 5 V buck regulator for stable supply of 5 V, 3 A. The whole section from VIN including 5 V regulator is not populated on the board by default. Overall on-board peripherals are functional with USB supply current. Only in case of validating external Arduino compatible shields with XMC7100 Lite Evaluation Board or if the application consumes more than 500 mA or 900 mA upon using USB2.0 or USB3.0 respectively, then for higher current capability, user may have to utilize VIN supply source from external adapter to source higher current as required. The on-board 5 V buck regulator supports up to 24 V of input with 3 A load capability. User need to have overall section assembled including VIN (DC Jack) for this to be operational.

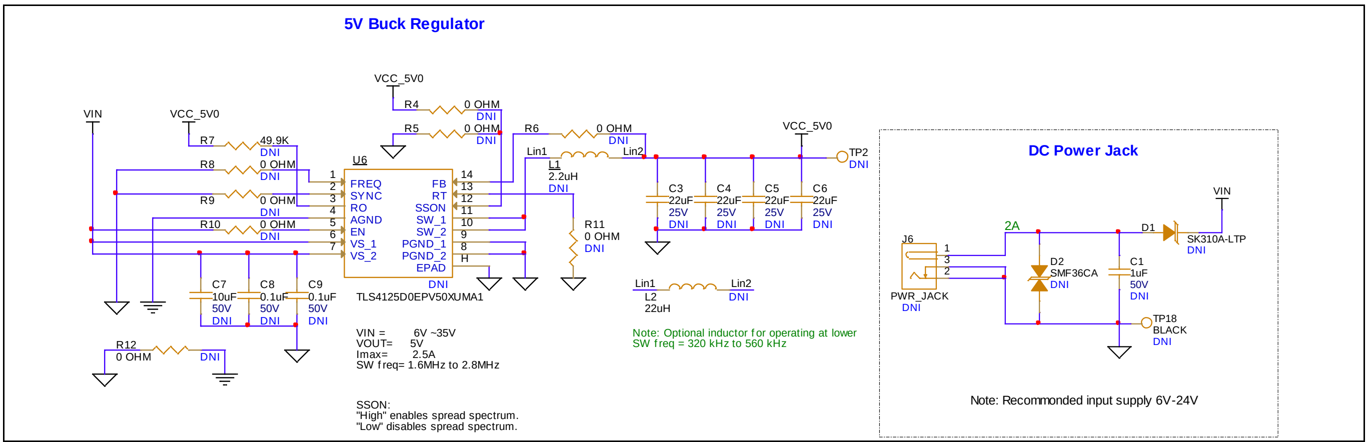


Figure 41 5 V buck regulator

4.3.5 KitProg3 UART and I2C interface

By default, UART and I2C communication interfaces are functional on the XMC7100 Lite Evaluation Board. If you find any unintended behavior in these lines, use optional pull-up options by populating 4.7K and 10K resistors for UART and I2C interfaces respectively.

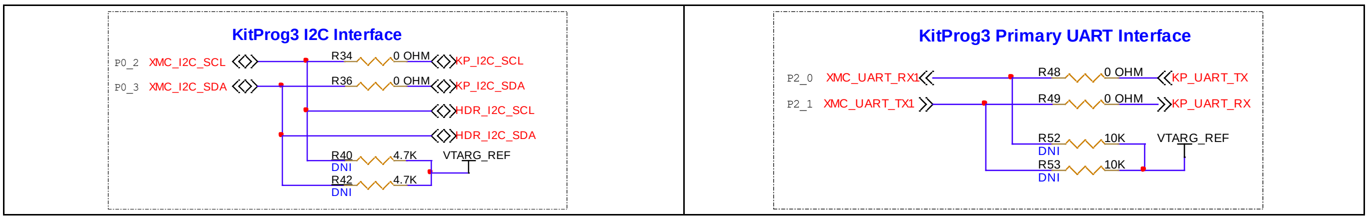


Figure 42 KitProg3 UART and I2C interface

4.3.6 VDDIO1 and VDDIO2 supply voltage selection

By default, the VDDIO1 and VDDIO2 I/O supplies are connected to XMC_VDD, which is directed from VTARG selection. Either 3.3 V or 5 V can be configured using the J8 header. For using fixed voltage at VDDIO1 or VDDIO2 I/O blocks, rework by assembling R73 for VDDIO1 and R75 for VDDIO2, which sets the I/O voltage level fixed to 3.3 V. Note that you must remove R72 and R74, respectively, for both I/O domains before fixing the I/O block voltages to VCC_3V3.

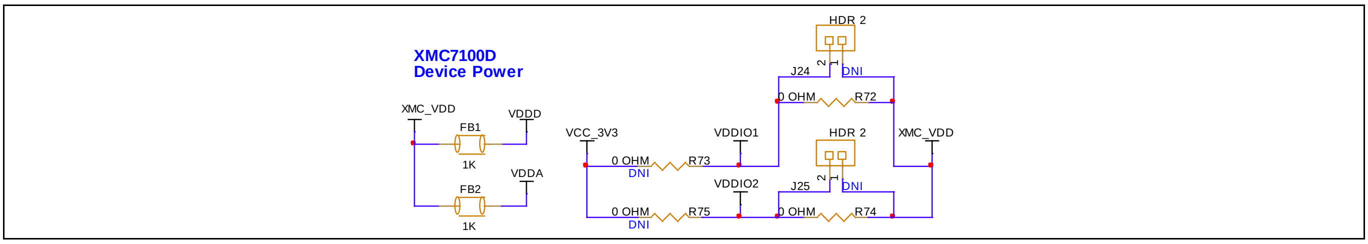


Figure 43 XMC7100 VDDIO1 and VDDIO2 voltage selection

4.3.7 Anti-aliasing filter provision for analog signals

To evaluate the ADC pin functionality on the XMC7100 Lite Evaluation Board, an anti-aliasing filter provision is given for analog signals that are connected to J2 of the Arduino-compatible header, J19 of the mikroBUS header, and J16 and J20 of the shield2go headers. All the analog-compatible pins on these headers are connected to the XMC7100 device GPIOs using a series resistor (0-ohm) and an optional capacitor acting as an anti-aliasing filter. You must have the required capacitors assembled for this operation. For optimizing the filtering option, you must have a respective 0-ohm resistor value to be modified for a suitable value specific to the user application.

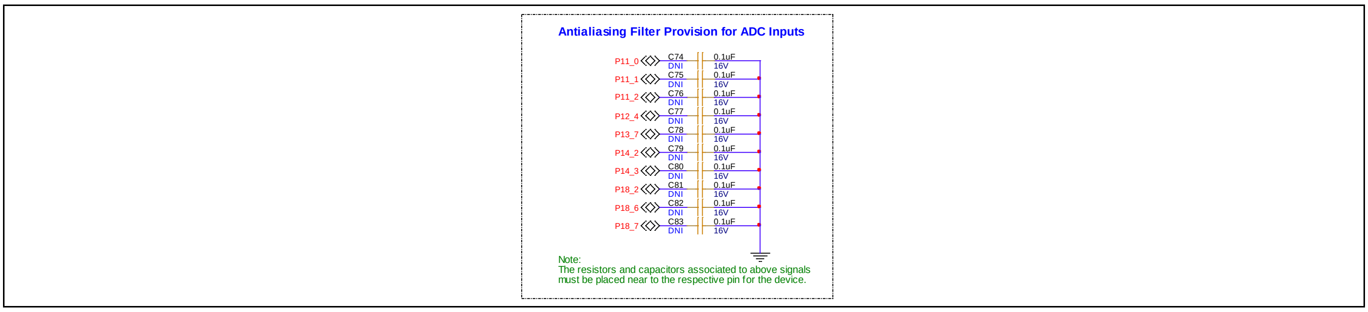


Figure 44 Anti-aliasing filter provision for analog signals

4.3.8 Watch crystal oscillator (WCO)

By default, WCO is not populated on the XMC7100 Lite Evaluation Board. You must have a compatible crystal of 32.768 KHz for evaluation of watch applications.

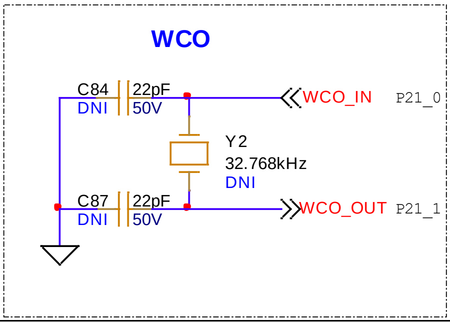


Figure 45 WCO

4.3.9 User button2

The XMC7100 Lite Evaluation Board is designed with dual-footprint compatibility, with the possibility of evaluating either a 176-TEQFP device or a 100-TEQFP device from the XMC7000 series. Pin assignment for user button2 is dedicatedly connected to P21.4, which is a multifunctional pin for awakening the device from Hibernate mode. The 100-pin device does not have a dedicated hibernate wakeup signals. When 100-pin device is used, the 0-ohm resistor R82 needs to be populated to connect P8.2 to the User button2.

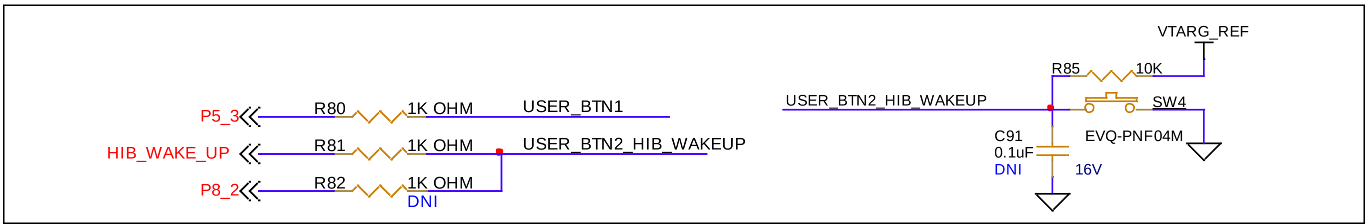


Figure 46 User buttons

4.3.10 CAN FD termination and I/O supply

The XMC7100 Lite Evaluation Board consists of an onboard CAN FD transceiver to evaluate classic CAN and CAN FD modes of operation. The transceiver is capable of operating at a 5 Mbps data rate. Hardware is designed for compatibility with similar transceivers from Infineon; therefore, 120-ohm parallel termination along with the C94 is provided on the board as an optional requirement. Bypass capacitor “C92” is also an optional requirement to filter the RxD line.

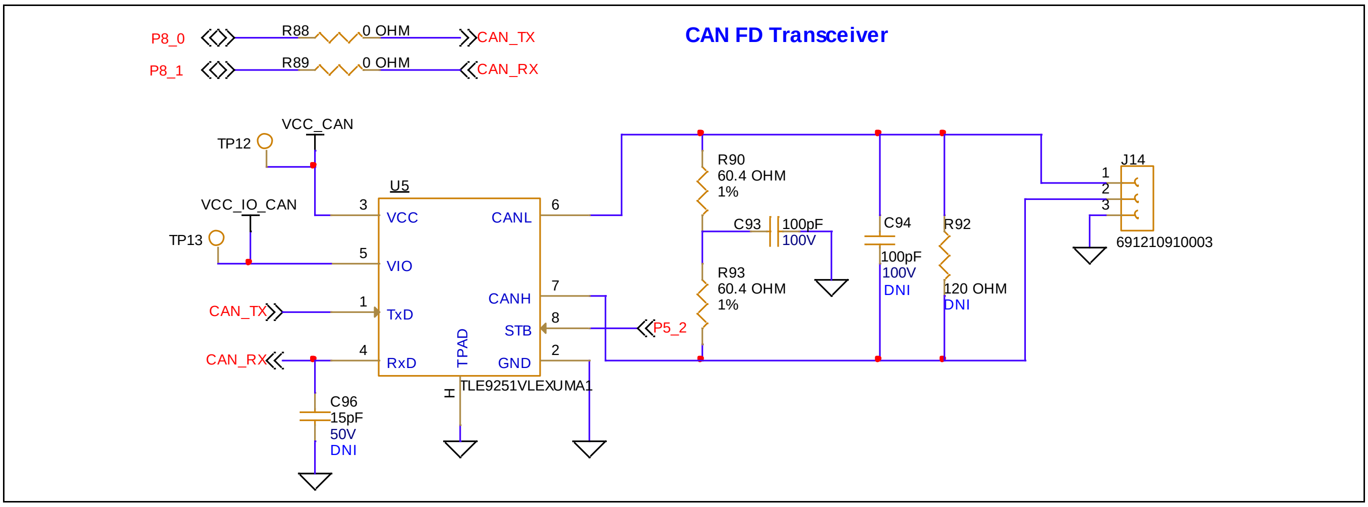


Figure 47 CAN FD transceiver

4.3.11 QSPI voltage-level translator pull-ups

The QSPI interface on the XMC7100 Lite Evaluation Board is connected to the XMC7100 device through voltage level translators. Optional resistor pull-ups are connected to the board for evaluation purposes. By default, all the pull-ups for data and clock are not populated on the board. In case of any unintended behavior with the QSPI interface, you can have this pull-up evaluated for enhanced operation.

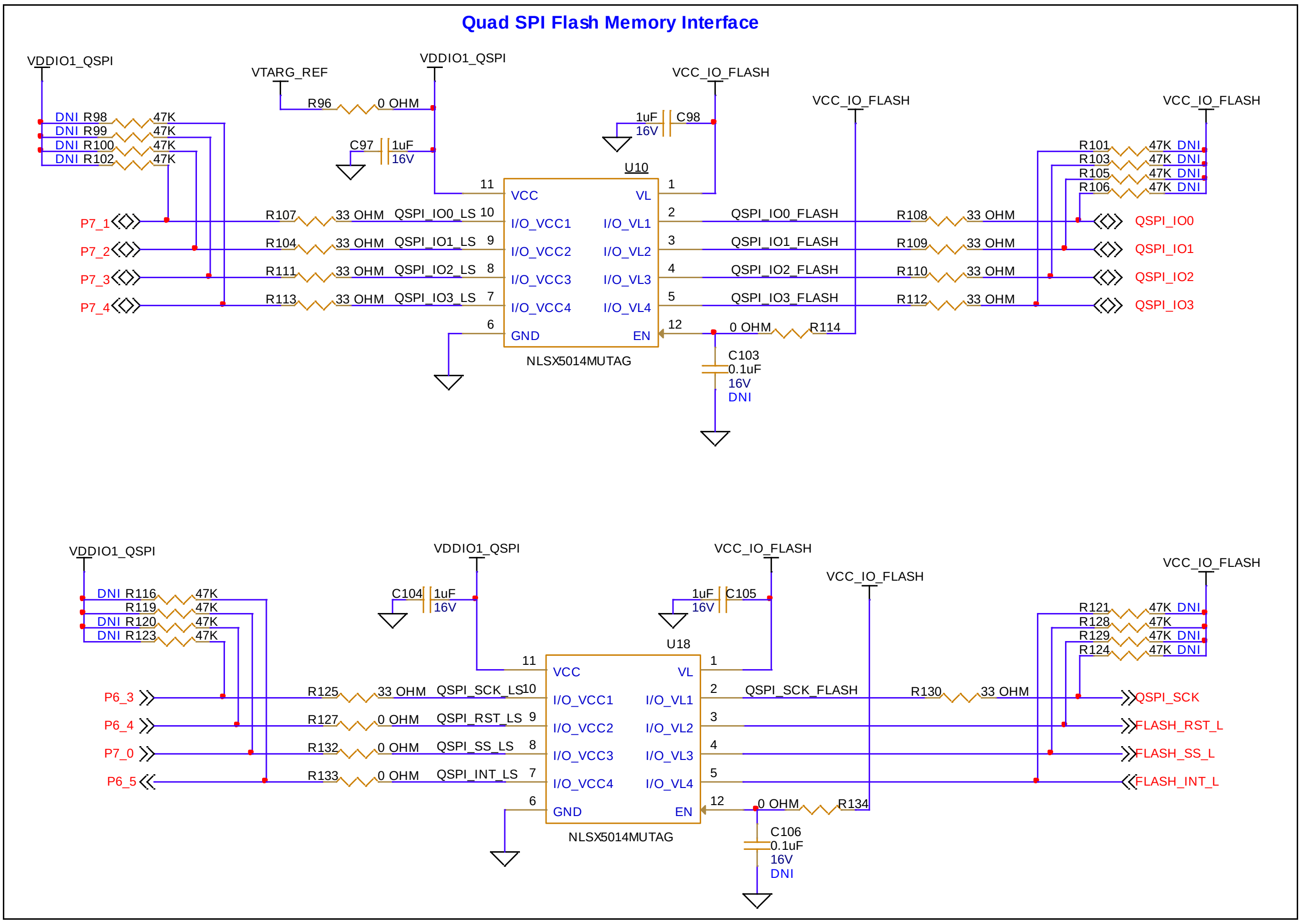


Figure 48 QSPI voltage level translator pull-ups

4.3.12 Ethernet interface

Overall Ethernet section on the XMC7100 Lite Evaluation Board is not populated. You may have to assemble the complete section of Ethernet for validation. For more details, see Ethernet transceiver and RJ45 connector.

Ethernet to work as PHY being master mode, do the following specific assembly or rework on the EVK:

- Populate Ethernet transceiver U12 with DP83825IRMQR
- Populate 25 MHz crystal as specified in the design or PCBA bill of material.
- Populate level shifter ICs U13, U14, U19, and U21
- Populate RJ45 connector specified in the BOM
- Populate other discrete resistors such as R136-R142, R144, R145, R147, R148, R156, R150, R152, R154, R155, R158, R165, R184, R187, R248, R251, R247, R194, R153, R156, R151, R246, R146
- Populate decoupling and bypass capacitors specified below as per the design requirement: C107-C118, C120-C123, C130, C126, C127, C128, C131

Note: For component information such as manufacturer, part number, and values, see the BOM file available on the kit webpage.

4.3.13 Shield2Go sensor shield assembly

The XMC7100 Lite Evaluation Board is featured with footprint compatible to Infineon's Shield2go sensor shields. User can have these shields mounted directly on the board or this can be mounted on headers by having suitable connectors assembled on the board.

Note: Infineon's standard S2GO sensors are voltage tolerant up to 4V. User must have to configure the target voltage to 3.3 V (J8) before working with sensor shields on S2GO headers (S2GO1 and S2GO2).

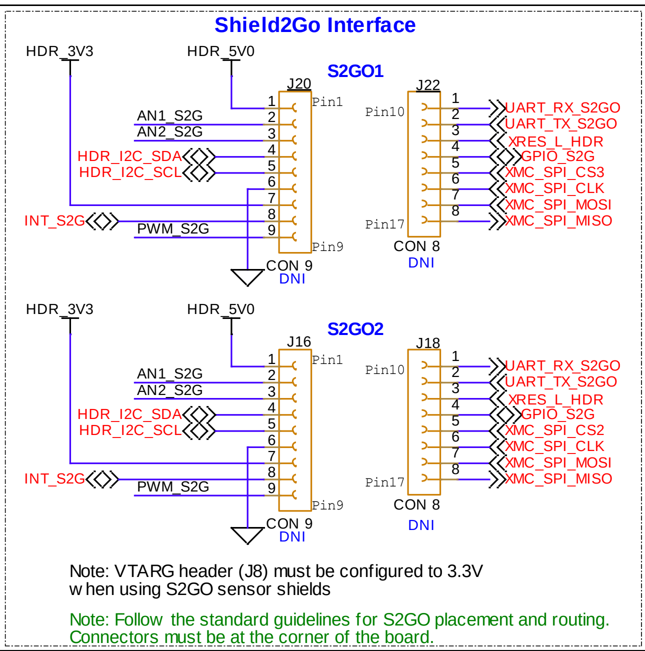


Figure 49 Shield2Go sensor shield assembly

4.3.14 GPIO expansion headers

All the free left GPIOs from 176-TEQFP device (other than connected to peripheral interfaces) are routed to 20x2 (80-pin) expansion headers. I/Os with different categories such as Analog, PWM, and multifunctional pins are connected to these expansion headers, which are by default not populated on the evaluation board.

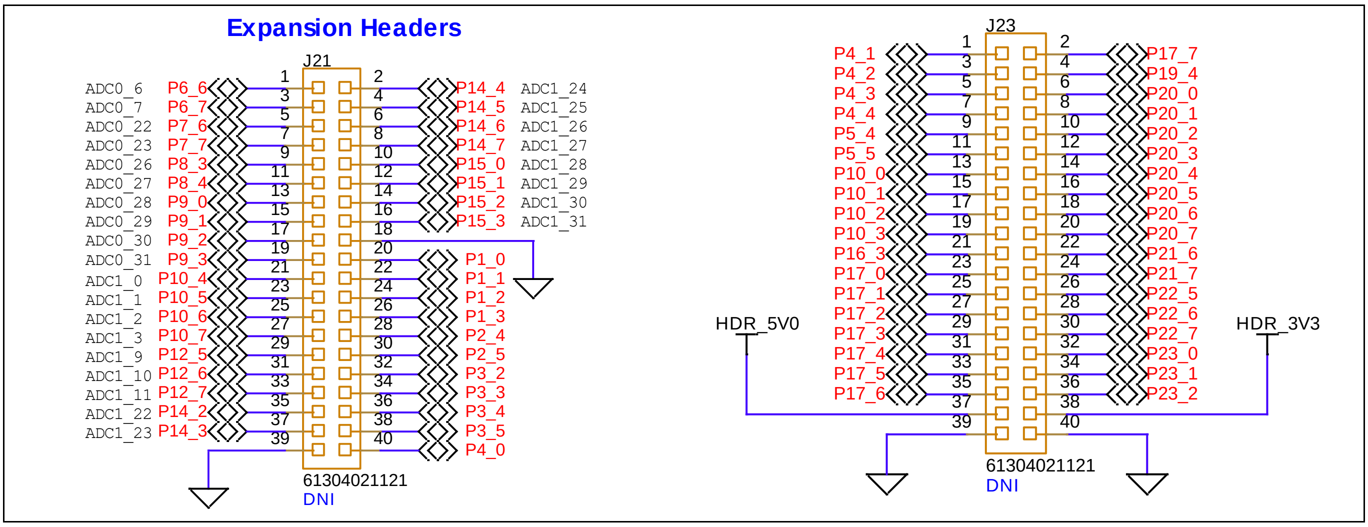


Figure 50 GPIO expansion headers

4.3.15 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.

Assemble the Trace, JTAG, and SWD respective resistor pins for functional validation of ETM trace header interface.

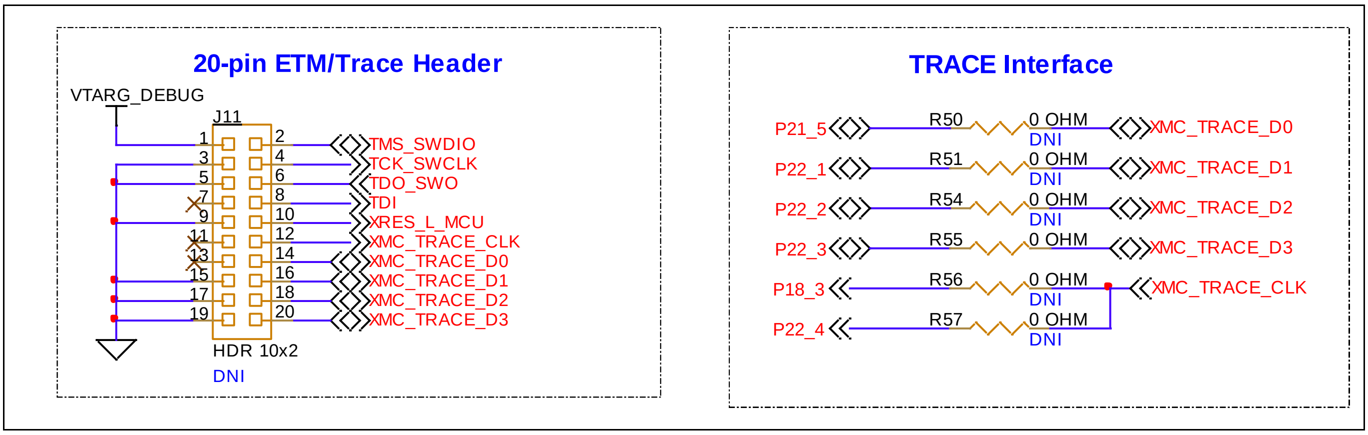


Figure 51 ETM trace and debug header

4.4 Bill of materials

See the BOM file available on the kit webpage.

4.5 Frequently asked questions

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

The core external regulator is implemented with an external NPN pass transistor in the XMC7100 Evaluation Board. The board can handle 300 mA with an internal core alone and 600 mA with an external regulator from the pass transistor.

2. How does XMC7100 Evaluation Board handle voltage connection when multiple power sources are plugged in?

There are two different options to power the kit:

- KitProg3 Micro-B USB connector (**J5**)
- External DC supply via VIN connector (**J6**) (by default, not populated on the board).

A header (J7) provision is given to switch the source power to the board. Either from USB micro-B or from VIN connector can be a source of power for the board.

Note: If using an external VIN supply, the communication interface between the system and KitProg3 will not be available unless a USB micro-B interface is connected between the system and the board.

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

Input voltage levels are shown in Table 6. There is no overvoltage protection for this kit.

Table 6 Input voltage levels

Supply	Typical input voltage	Absolute max
USB Micro-B connector (**J7**)	4.75 V to 5.25 V	5.5 V
VIN connector (**J6**)	6 V to 24 V	35 V

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

XMC7100D is not meant to be operated at voltages lower than 2.7 V. Powering the XMC7100D to lower than 2.7 V will stop the chip to be powered ON. It is recommended to power the XMC7100D at 3.3 V or 5 V.

The kit is capable of driving external 1.8 V interfaces provided a level shifter circuit is also used. Connecting a 1.8 V interface to 3.3 V will damage the interface.

5. How to program the target devices?

- Check **J8** to ensure that the jumper shunt is placed.
- Make sure that no external devices are connected to the external programming header J10 or J11.
- Update your KitProg3 version to the latest as mentioned in the KitProg3 user guide.
- To program the target, see the Programming and debugging using ModusToolbox™ section.

6. Can the kit be powered using the external program and debug headers J10 and J11?

No, this is not possible on this board by default. The target MCU is powered by onboard regulators only, and therefore, one of the two main sources (J5 and J6) must be present. There is a protection circuit that prevents a reverse voltage from VTARG_Debug to VTARG_REF. Therefore, the board cannot be powered through J10 and J11.

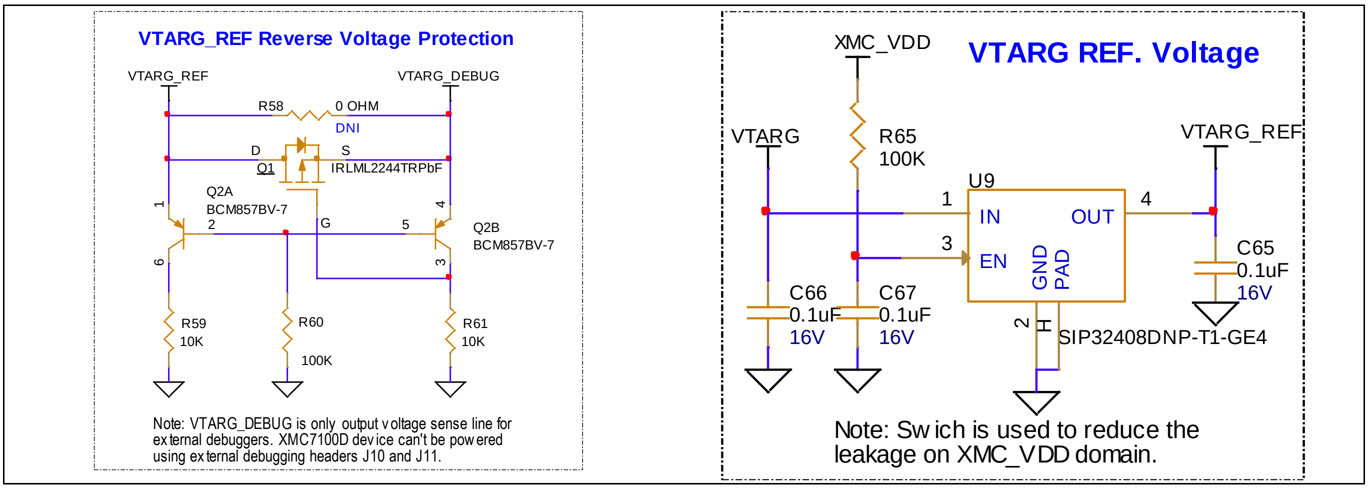


Figure 52 VTARG reverse voltage protection

References

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

Application note:

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

Datasheets:

[2] XMC7100, XMC7000 microcontroller 32-bit Arm® Cortex®-M7

[3] XMC7200, XMC7000 microcontroller 32-bit Arm® Cortex®-M7

Glossary

ADC
analog-to-digital converter

BOM
bill of materials

CAN FD
controller area network flexible data-rate

CPU
Central Processing Unit

DC
direct current

ECO
external crystal oscillator

ESD
electrostatic discharge

ETM
Embedded Trace Macrocell

GPIO
general purpose input/output

IC
Integrated Circuit

IDE
integrated development environment

I2C
Inter-Integrated Circuit

I2S
inter-IC sound

JTAG
Joint Test Action Group

LED
light-emitting diode

MAC
medium access control

PC
personal computer

PDL
Peripheral Driver Library

POT
Potentiometer

QSPI
Quad Serial Peripheral Interface

RMII
Reduced Media Independent Interface

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

S2GO
Shield2Go

UART
Universal Asynchronous Receiver Transmitter

USB
Universal Serial Bus

WCO
watch crystal oscillator

Revision history

Document revision	Date	Description of changes
**	2023-12-12	Initial release
*A	2024-04-17	Updated Introduction section.
*B	2024-08-27	Updated the section Ethernet interface.

Edition 2024-08-27

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

002-38520 Rev. *B