K32W148-EVKUM

K32W148-EVK Board User Manual

Rev. 1 — 13 February 2023 User manual

Document information

Information	Content
Keywords	K32W148-EVKUM, K32W148-EVK, K32W148, Bluetooth LE 5.3, IEEE 802.15.4
Abstract	The K32W148-EVK board is a highly configurable, low-power, and cost-effective, evaluation and development platform for NXP K32W148 MCU, which is a low-power, highly secure, single-chip wireless MCU supporting Arm Cortex-M33 core

1 K32W148-EVK overview

The K32W148-EVK board is a highly configurable, low-power, and cost-effective, evaluation and development platform for NXP K32W148 microcontroller (MCU), which is a low-power, highly secure, single-chip wireless MCU supporting Arm Cortex-M33 core. It offers an easy-to-use user interface with a virtual serial port and standard programming and run-control capabilities.

The K32W148 MCU includes a 96 MHz Arm Cortex-M33 CPU, a radio transceiver operating in the 2.4 GHz frequency band (supporting GFSK modulations for Bluetooth Low Energy (LE) 5.3 and OQPSK modulations for IEEE 802.15.4), Bluetooth LE controller stack, IEEE 802.15.4 packet processor, transceiver drivers, up to 1 MB flash, up to 128 KB SRAM, and peripherals optimized to meet the requirements of target applications. The K32W148 MCU supports multiprotocol use cases with Matter, Thread, Zigbee, and Bluetooth LE for IoT and industrial applications.

The K32W148-EVK board provides RF circuitry (including antenna), 16 Mbit QSPI NOR flash memory, accelerometer, and 32 MHz crystal. It is compatible with the Arduino UNO R3 and Mikroe click boards. The board is a standalone PCB that supports application development with GFSK and OQPSK libraries from NXP. It can be used with a wide range of development tools, including NXP MCUXpresso IDE and IAR Embedded Workbench. The board is lead-free and RoHS-compliant.

For debugging the K32W148 MCU, the K32W148-EVK board uses an onboard debug probe, known as MCU-Link OB (OB stands for "onboard"), which is based on another MCU, LPC55S69. For simplicity, MCU-Link OB is referred to as "MCU-Link debug probe" or just "MCU-Link" and the K32W148 MCU is referred to as "target MCU" in this document.

This document provides details about the K32W148-EVK board interfaces, power supplies, clocks, push buttons, jumpers, and LEDs.

Note: *For information on the operating temperature range for the board, see Section 2.13.*

1.1 Radio Equipment Directive compliance information

The following information is provided per Article 10.8 of the Radio Equipment Directive 2014/53/EU:

- Frequency bands in which the equipment operates
- The maximum RF power transmitted

Part number	RF technology	Frequency ranges (EU)	Maximum transmitted power
K32W148-EVK	Bluetooth LE	2360 MHz – 2483 MHz	+10 dBm
	IEEE 802.15.4	2360 MHz – 2483 MHz	+10 dBm

EUROPEAN DECLARATION OF CONFORMITY (Simplified DoC per Article 10.9 of the Radio Equipment Directive 2014/53/EU)

This apparatus, namely K32W148-EVK Development Platform, conforms to the Radio Equipment Directive 2014/53/EU. The full EU Declaration of Conformity for this apparatus is available at European Union Declaration of Conformity for K32W148-EVK Evaluation Kit.

1.2 Acronyms and abbreviations

Table 1 lists and explains the acronyms and abbreviations used in this document.

Table 1. Acronyms and abbreviations

Term	Description
4PDT	Four pole double throw

Table 1. Acronyms and abbreviations...*continued*

Term	Description
ADC	Analog-to-digital converter
DNP	Do not populate
DoC	Declaration of Conformity
GFSK	Gaussian frequency shift keying
GND	Ground
GPO	General-purpose output
GPIO	General-purpose input/output
HID	Human interface device
HS	High-speed
I2C	Inter-integrated circuit
IDE	Integrated development environment
IoT	Internet of things
ISP	In-System Programming
LE	Low Energy
LED	Light-emitting diode
LPI2C	Low-Power Inter-Integrated Circuit
LPSPI	Low-Power Serial Peripheral Interface
LPUART	Low-Power Universal Asynchronous Receiver/Transmitter
MCU	Microcontroller unit
NC	Not connected
NMI	Non-maskable interrupt
OQPSK	Offset quadrature phase-shift keying
OS	Operating system
OTA	Over-the-air
PCS	Peripheral chip select
PER	Packet error rate
PMIC	Power management integrated circuit
ppm	Parts per million
PWM	Pulse width modulation
QSPI	Quad serial peripheral interface
RED	Radio Equipment Directive
RTC	Real-time clock
SMA	Subminiature version A
SPI	Serial peripheral interface
SWD	Serial wire debug
SWO	Serial wire debug trace output

Table 1. Acronyms and abbreviations...*continued*

Term	Description
TCXO	Temperature compensated crystal oscillator
UART	Universal asynchronous receiver/transmitter
USB	Universal serial bus
USBSIO	USB serial input/output
UWB	Ultra-wideband
VCOM	Virtual communication
WUU	Wake-up unit

1.3 Related documentation

Table 2 lists and explains the additional documents and resources that you can refer to for more information on the K32W148-EVK board. Some of the documents listed below may be available only under a non-disclosure agreement (NDA). To request access to these documents, contact your local field applications engineer (FAE) or sales representative.

Table 2. Related documentation

Document	Description	Link / how to access
K32W1480 Reference Manual	Intended for system software and hardware developers and application programmers who want to develop products with K32W148 MCU	K32W1480RM.pdf
K32W14x Product Family Data Sheet	Provides information about electrical characteristics, hardware design considerations, and ordering information	K32W1480.pdf
Hardware Design Considerations for KW45B41Z and K32W148 Bluetooth LE Devices (AN13227)	Provides PCB design considerations for new designs based on NXP KW45B41Z and K32W148 MCUs	AN13227.pdf
KW45 and K32W148 RF System Evaluation Report for Bluetooth LE and K32W148 for 802.15.4 Applications only (AN13728)	Describes the K32W148-EVK board RF system evaluation report for Bluetooth LE and IEEE 802.15.4 applications	AN13728.pdf
K32W148 Connectivity Test for 802.15.4 Application (AN13687)	Describes how to use the connectivity test tool to perform the K32W148 802.15.4 RF performance	AN13687.pdf
LPC55S6x/LPC55S2x/LPC552x User manual (UM11126)	Intended for system software and hardware developers and application programmers who want to develop products with LPC55S6x/LPC55S2x/LPC552x MCU	UM11126.pdf

1.4 Board kit contents

The K32W148-EVK board kit contains the following items:

- K32W148-EVK board hardware assembly
- A USB cable
- Spare jumpers

1.5 Block diagram

Figure 1 shows the K32W148-EVK block diagram.

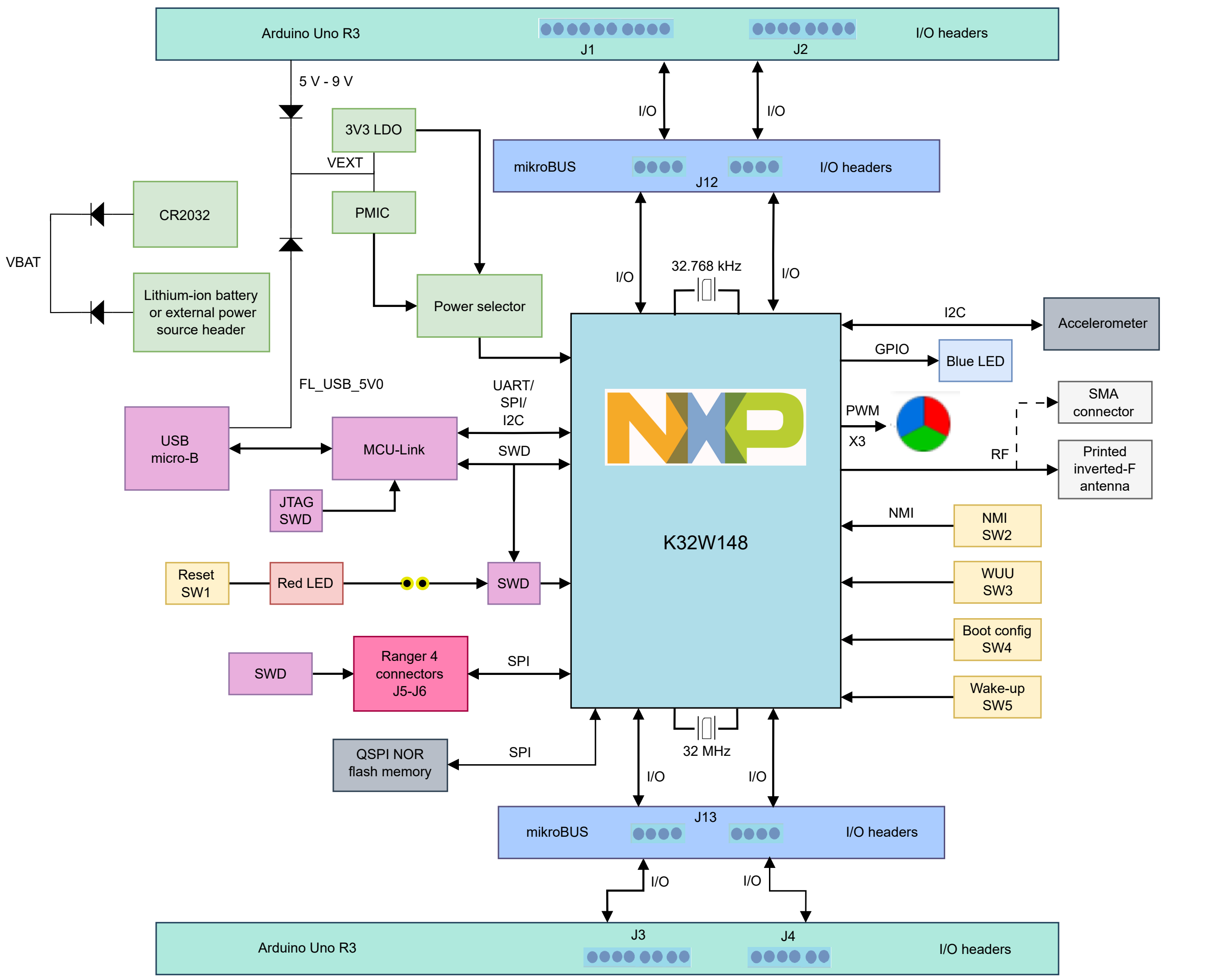

Figure 1. K32W148-EVK block diagram

1.6 Board features

Table 3 describes the features of the K32W148-EVK board.

Table 3. K32W148-EVK features

Board feature	Target MCU feature used	Description
MCU (target MCU)		K32W148 MCU with Arm Cortex-M33 core running at a frequency of up to 96 MHz ***Note:*** *For more details on the K32W148 MCU, see K32W14x Product Family Data Sheet and K32W1480 Reference Manual.*

Table 3. K32W148-EVK features*...continued*

Board feature	Target MCU feature used	Description
LPUART	Two LPUART modules (LPUART0 and LPUART1)	• LPUART0: – Connects to one of the two mikroBUS socket connectors for an external UART connection – Connects to one of the four Arduino socket connectors for an external UART connection – Can be used for MCU-Link connection in place of LPUART1 • LPUART1: – Supports a USB-to-UART bridge for connecting MCU-Link to the target MCU – Connects to another Arduino socket connector for an external UART connection
LPSPI	One of the two LPSPI modules (LPSPI1) with four peripheral chip selects (PCS0, PCS1, PCS2, and PCS3)	• PCS0: Connects to 16 Mbit QSPI NOR flash memory, which supports over-the-air (OTA) programming • PCS1: Connects to one of the two Ranger 4 socket connectors • PCS2: Connects to one of the four Arduino socket connectors • PCS3: – Supports a USB-to-SPI bridge for connecting MCU-Link to the target MCU – Provides an alternative SPI connection to PCS0 connection – Provides an alternative SPI connection to PCS2 connection – Connects to one of the two mikroBUS socket connectors
LPI2C	One of the two LPI2C modules (LPI2 C1)	LPI2C1 module connects to: • MCU-Link through a USB-to-I2C bridge connecting MCU-Link to the target MCU • Power management integrated circuit (PMIC) • Accelerometer • One of the two mikroBUS socket connectors • One of the four Arduino socket connectors
RF	2.4 GHz radio	A PCB-integrated inverted-F antenna and an SMA connector for RF operation
Accelerometer		3-axis, 12-bit, compact digital accelerometer
mikroBUS		mikroBUS socket with two 1x8 position receptacles
Arduino		Arduino UNO board compatible socket with four receptacles (1x8, 1x10, 1x8, and 1x6 positions)
Ranger 4		Ranger 4 UWB board compatible socket with two receptacles (2x10 and 2x8 positions)
MCU-Link USB		USB 2.0 micro-B connector for creating MCU-Link high-speed USB connection to the host computer, and providing 5 V power to the board
Power supply		• The board can be powered up through: – Coincell battery – USB micro-B connector – VBAT header (DNP)

Table 3. K32W148-EVK features*...continued*

Board feature	Target MCU feature used	Description
		• Following power configuration modes are supported: – DC-DC Converter with Buck mode – DC-DC Converter with Bypass mode – PMIC mode – Smart Power Switch mode
Clock		The target MCU receives the following two clocks: • A 32 MHz clock from a crystal for Arm core and radio. Alternatively, the 32 MHz clock can be provided from a TCXO (not populated on the board). • A 32.768 kHz clock from another crystal as RTC and low-power RF clock
Debug		• Onboard MCU-Link debug probe with CMSIS-DAP and SEGGER J-Link protocol options. It can connect to the target MCU through a USB-to-UART, USB-to-SPI, or USB-to-I2C bridge. • 10-pin Arm JTAG/SWD connector for connecting an external debug probe

1.7 Board pictures

Figure 2 shows the top-side view of the K32W148-EVK board, with connectors, push buttons, and LEDs highlighted.

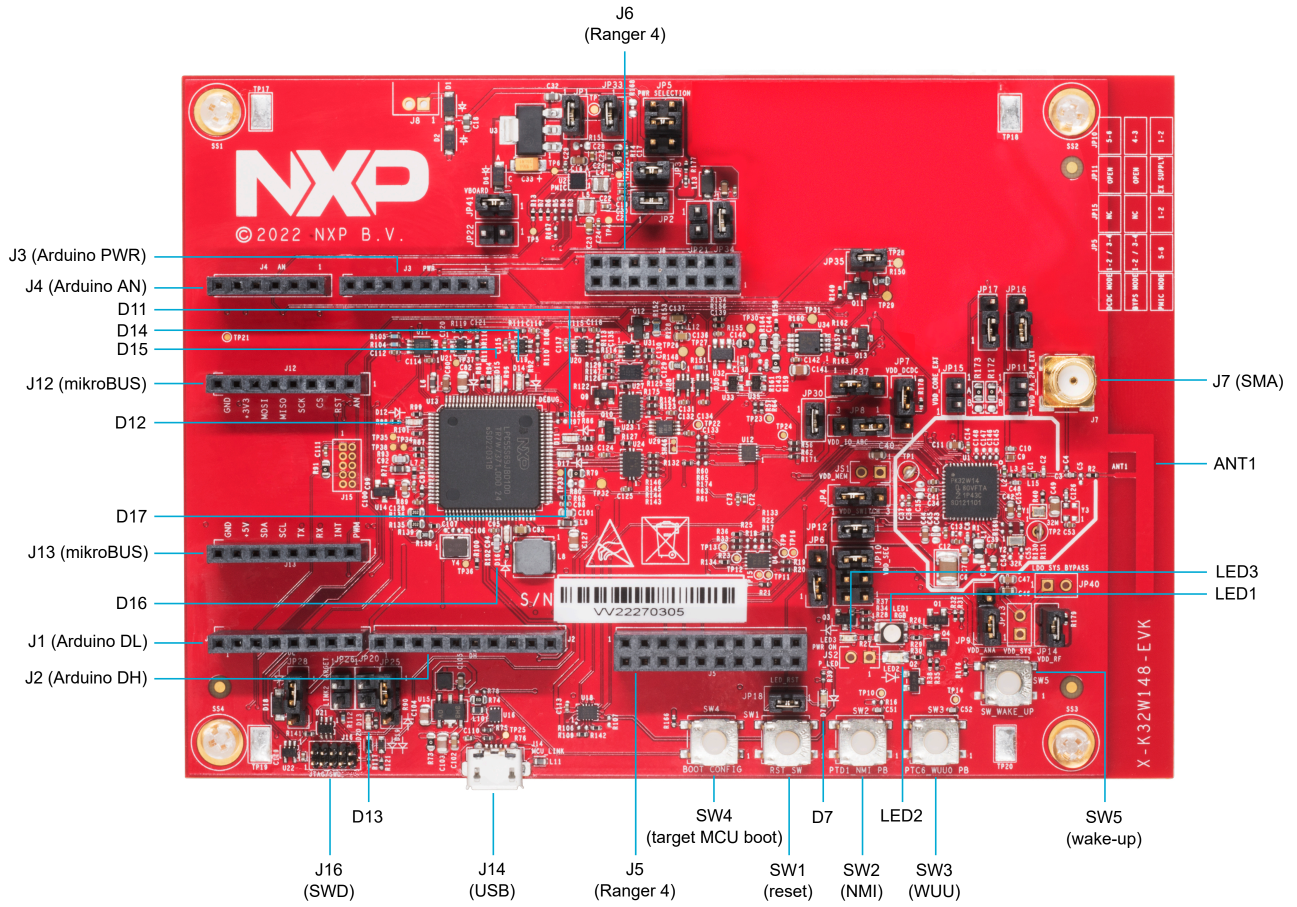

Figure 2. K32W148-EVK connectors, push buttons, and LEDs (top-side view)

Figure 3 shows the jumpers of the K32W148-EVK board.

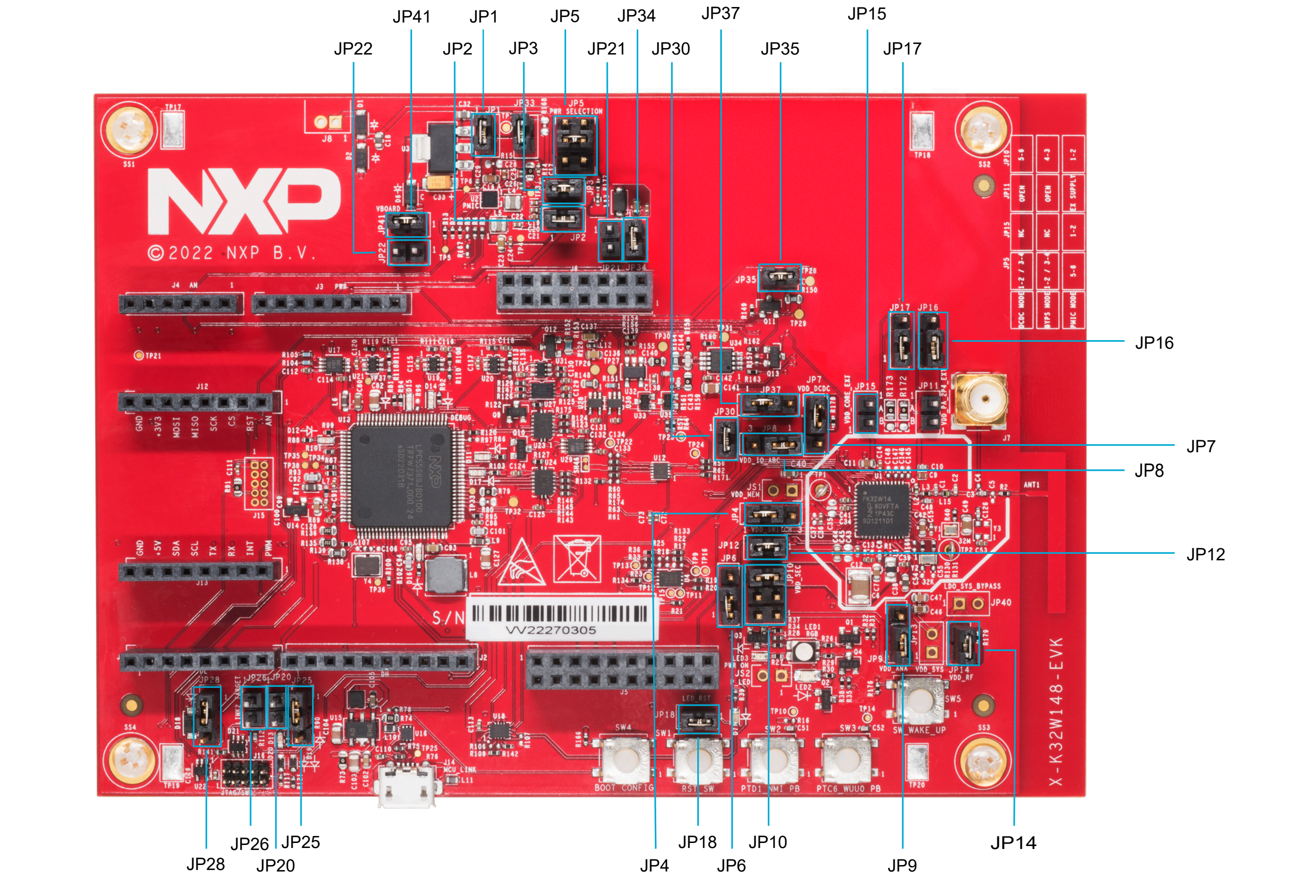

Figure 3. K32W148-EVK jumpers

Figure 4 shows the bottom-side view of the K32W148-EVK board, with battery holder BT2 highlighted.

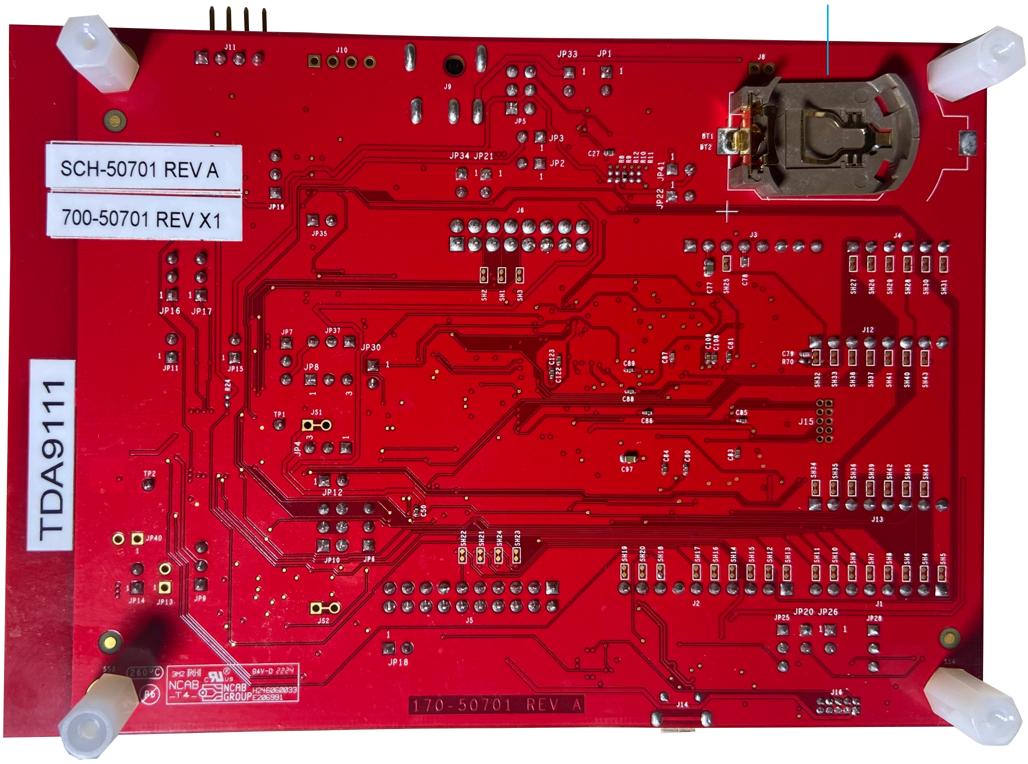

Figure 4. K32W148-EVK bottom-side view

1.8 Connectors

Table 4 describes the K32W148-EVK connectors. The connectors are shown in Figure 2 and Figure 4.

Table 4. K32W148-EVK connectors

Part identifier	Connector type	Description	Reference section
BT1 (DNP) (secondary option)	Coincell battery holders **Note:** *BT1 and BT2 have overlapping footprints (BT2 is used by default).*	Holder for 3 V CR2450N lithium coincell battery	Section 2.1
BT2 (primary option)		Holder for 3 V CR2032 lithium coincell battery	
J8 (DNP)	1x2 pin header	Lithium-ion/external battery connector	
J14	USB 2.0 micro-B receptacle	MCU-Link USB connector	Section 3.3
J7	SMA connector	SMA RF connector	Section 2.6
ANT1	Antenna	PCB-integrated RF antenna	
J12	1x8 position receptacles	mikroBUS socket connectors	Section 2.8
J13			
J1	1x8 position receptacle	Arduino socket connectors	Section 2.9
J2	1x10 position receptacle		
J3	1x8 position receptacle		

Table 4. K32W148-EVK connectors*...continued*

Part identifier	Connector type	Description	Reference section
J4	1x6 position receptacle		
J5	2x10 position receptacle	Ranger 4 UWB socket connectors	Section 2.10
J6	2x8 position receptacle		
J16	2x5 pin header	Arm JTAG (SWD) connector to connect an external debug probe or external debug target	Section 3.5
J15 (DNP)	2x5 pin header	MCU-Link SWD connector	For more information on this connector, see K32W148-EVK schematics

1.9 Jumpers

Table 5 describes the K32W148-EVK jumpers. The jumpers are shown in Figure 3.

Table 5. K32W148-EVK jumpers

Part identifier	Jumper type	Description	Reference section
JP18	1x2 pin header	Reset LED enable jumper: • Open: Reset LED is disabled • Shorted (default setting): Reset LED is enabled and it turns ON when Reset button (SW1) is pressed	Section 1.10
JP25	1x2 pin header	Target MCU boot configuration button (SW4) enable jumper: • Open (default setting): SW4 is disabled • Shorted: SW4 is enabled	
JP1	1x2 pin header	PMIC 5 V input power enable jumper: • Open: 5 V input power is disconnected from PMIC • Shorted (default setting): 5 V input power is supplied to PMIC	Section 2.1
JP2	1x2 pin header	P1V8_EXT supply enable jumper: • Open: P1V8_EXT supply is OFF • Shorted (default setting): P1V8_EXT supply is ON	
JP3	1x2 pin header	P1V1_EXT supply enable jumper: • Open: P1V1_EXT supply is OFF • Shorted (default setting): P1V1_EXT supply is ON	
JP4	1x3 pin header	VDD_SWITCH supply power source selection jumper: • 1-2 shorted (default setting): VDD_SWITCH supply is produced from P3V3 supply • 2-3 shorted: VDD_SWITCH supply is produced from MCU_VDD_REG supply	
JP42	1x2 pin header	Target MCU debugging enable jumper:	

Table 5. K32W148-EVK jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Open: Target MCU debugging is disabled. ***Note:*** *To use VBAT supply (coincell power) for producing P3V3 supply at jumper JP5, pins 1-2 of JP5 must be shorted with a shunt. With this setting, VBOARD supply is unpowered. Therefore, VCC_TGMCU supply, which is powered by VBOARD supply, is also unpowered. This causes voltage level issues on RST_TGTMCU_b signal to the target MCU.* *To avoid such issues, open JP42 before using VBAT as source supply at JP5.* • Shorted (default setting): Target MCU debugging is enabled	
JP5	2x3 pin header	P3V3 supply power source selection jumper: • 1-2 shorted: P3V3 supply is produced from VBAT supply. ***Note:*** *To use VBAT supply (coincell power) for producing P3V3 supply at JP5, pins 1-2 of JP5 must be shorted with a shunt. With this setting, VBOARD supply is unpowered. Therefore, VCC_TGMCU supply, which is powered by VBOARD supply, is also unpowered. This causes voltage level issues on RST_TGTMCU_b signal to the target MCU.* *To avoid such issues, open jumper JP42 before using VBAT as source supply at JP5. With JP42 open, target MCU debugging is disabled.* • 3-4 shorted (default setting): P3V3 supply is produced from VBOARD supply • 5-6 shorted: P3V3 supply is produced from PMIC_LDO2 supply	
JP6	1x3 pin header	VDD_REG supply power source selection jumper: • 1-2 shorted (default setting): VDD_REG supply is produced from P3V3 supply • 2-3 shorted: VDD_REG supply is produced from VOUT_SWITCH power of the target MCU	
JP7	1x3 pin header	VDD_DCDC supply power source selection jumper: • 1-2 shorted (default setting): VDD_DCDC supply is produced from VDD_REG supply • 2-3 shorted: VDD_DCDC supply is produced from MCU_VDD_REG supply	
JP8	1x3 pin header	VDD_IO_ABC supply power source selection jumper: • 1-2 shorted (default setting): VDD_IO_ABC supply is produced from VDD_REG supply	

Table 5. K32W148-EVK jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• 2-3 shorted: VDD_IO_ABC supply is produced from MCU_VDD_REG supply	
JP9	1x3 pin header	VDD_ANA supply power source selection jumper: • 1-2 shorted (default setting): VDD_ANA supply is produced from VDD_REG supply • 2-3 shorted: VDD_ANA supply is produced from MCU_VDD_REG supply	
JP10	2x3 pin header	V_SEC supply power source selection jumper: • 1-2 shorted: V_SEC supply is produced from P1V8_EXT supply • 3-4 shorted: V_SEC supply is produced from jumper JP37 output • 5-6 shorted (default setting): V_SEC supply is produced from 1V8_DCDC supply	
JP11 (DNP)	1x2 pin header	VDD_PA_2G4 supply enable jumper. JP11 is not populated on the board. If populated, it can be used to enable VDD_PA_2G4 supply for +10 dBm RF output.	
JP12	1x2 pin header	VDD_LDO_CORE supply enable jumper: • Open: VDD_LDO_CORE supply is OFF • Shorted (default setting): VDD_LDO_CORE supply is ON	
JP14	1x2 pin header	VDD_RF supply enable jumper: • Open: VDD_RF supply is OFF • Shorted (default setting): VDD_RF supply is ON	
JP15	1x2 pin header	VDD_CORE supply enable jumper: • Open (default setting): VDD_CORE supply is OFF • Shorted: VDD_CORE supply is ON	
JP35	1x2 pin header	MCU_VDD_REG supply enable jumper: • Open: MCU_VDD_REG supply is OFF. This jumper setting also enables onboard current measurement using MCU-Link. • Shorted (default setting): MCU_VDD_REG supply is ON	
JP37	1x3 pin header	Jumper JP10 pin 4 power source selection jumper: • 1-2 shorted (default setting): Jumper JP10 pin 4 input power is produced from VDD_REG supply • 2-3 shorted: Jumper JP10 pin 4 input power is produced from MCU_VDD_REG supply	
JP41	1x2 pin header	VBOARD supply enable jumper: • Open: VBOARD supply is OFF • Shorted (default setting): VBOARD supply is ON	

Table 5. K32W148-EVK jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
JP13 (DNP)	1x2 pin header	VOUT_SYS / VDD_SYS supply enable jumpers. JP13 and JP40 are not populated on the board. If populated, either can be used to enable VOUT_SYS / VDD_SYS supply. JP13 uses the P1V8_EXT supply as input whereas JP40 uses the VDD_DCDC supply as input.	
JP40 (DNP)	1x2 pin header		
JS1 (DNP)	1x2 pin header	VDD_MEM supply enable jumper. JS1 is not populated but pin 1 and pin 2 positions of the jumper are connected through a physical (trace) connection on the secondary (bottom) side of the PCB. Therefore, the VDD_MEM supply is enabled, by default. When populated, JS1 can be used to enable/ disable the VDD_MEM supply. If populating the jumper to allow manual connection/ disconnection, cut the trace connection between its pin 1 and pin 2 positions.	
JS2 (DNP)	1x2 pin header	P_LED supply enable jumper. JS2 is not populated but pin 1 and pin 2 positions of the jumper are connected through a physical (trace) connection on the secondary (bottom) side of the PCB. Therefore, the P_LED supply is enabled, by default. When populated, JS2 can be used to enable/ disable the P_LED supply. If populating the jumper to allow manual connection/ disconnection, cut the trace connection between its pin 1 and pin 2 positions.	
JP16	1x3 pin header	LPUART controller selection jumper for receiving UART signal from MCU-Link: • 1-2 shorted (default setting): LPUART1 controller of the target MCU receives UART signal from MCU-Link • 2-3 shorted: LPUART0 controller of the target MCU receives UART signal from MCU-Link	Section 2.3
JP17	1x3 pin header	LPUART controller selection jumper for sending UART signal to MCU-Link: • 1-2 shorted (default setting): LPUART1 controller of the target MCU sends UART signal to MCU-Link • 2-3 shorted: LPUART0 controller of the target MCU sends UART signal to MCU-Link	
JP21	1x2 pin header	USB-to-UART bridge disable jumper: • Open (default setting): MCU-Link is connected to the target MCU through a USB-to-UART bridge • Shorted: The USB-to-UART bridge between MCU-Link and the target MCU is disabled	
JP34	1x2 pin header	USB-to-SPI bridge disable jumper: • Open: MCU-Link is connected to the target MCU through a USB-to-SPI bridge	Section 2.4

Table 5. K32W148-EVK jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Shorted (default setting): The USB-to-SPI bridge between MCU-Link and the target MCU is disabled. With this jumper setting, the SPI PCS3 connection can be used to connect the target MCU to the SPI NOR flash memory (U4).	
JP30	1x2 pin header	USB-to-I2C bridge enable jumper: • Open: The USB-to-I2C bridge between MCU-Link and the target MCU is disabled • Shorted (default setting): MCU-Link is connected to the target MCU through a USB-to-I2C bridge	Section 2.5
JP43	1x3 pin header	Accelerometer I2C address configuration (when accelerometer acts as a secondary I2C device) jumper: • 1-2 shorted (default setting): Accelerometer 8-bit I2C addresses are as follows: – Read address: 0x33 – Write address: 0x32 • 2-3 shorted: Accelerometer 8-bit I2C addresses are as follows: – Read address: 0x31 – Write address: 0x30	Section 2.7
JP20	1x2 pin header	MCU-Link (LPC55S69) force ISP mode jumper: • Open (default setting): MCU-Link follows normal boot sequence (MCU-Link boots from internal flash if a boot image is found). With the internal flash erased, MCU-Link normal boot sequence falls through to ISP boot mode. • Shorted: MCU-Link is forced to ISP mode (USB1). Use this setting to reprogram MCU-Link internal flash with a new image or use the MCUXpresso IDE with CMSIS-DAP protocol. ***Note:*** *By default, MCU-Link flash is preprogrammed with a version of CMSIS-DAP firmware.*	Section 3.2
JP26	1x2 pin header	MCU-Link debug probe target selection jumper: • Open (default setting): The MCU-Link debug probe or an external debug probe can be connected to the onboard target MCU. Use this setting when the target MCU operates in Normal mode (its SWD (debug) interface is enabled). • Shorted: The MCU-Link debug probe can be connected to an external target MCU. Use this setting when using the MCU-Link debug probe for debugging an external target MCU.	Section 3.5
JP28	1x3 pin header	Target power selection jumper: • 1-2 shorted (default setting): Onboard target MCU is used as debug target or an external	

Table 5. K32W148-EVK jumpers...*continued*

Part identifier	Jumper type	Description	Reference section
		target MCU is used as debug target but it uses board power • 2-3 shorted: An external target MCU is used as debug target and it uses its own power	
JP22	1x2 pin header	MCU-Link serial wire debug (SWD) disable jumper: • Open (default setting): MCU-Link SWD feature is enabled. MCU-Link drives SWD of target MCU or external target MCU. • Shorted: MCU-Link SWD feature is disabled. This setting of JP22 can be used when connecting an external debug probe for debugging the target MCU.	For more information on this jumper, see K32W148-EVK schematics

1.10 Push buttons

Table 6 describes the K32W148-EVK push buttons. The push buttons are shown in Figure 2.

Table 6. K32W148-EVK push buttons

Part identifier	Switch name	Description
SW1	Reset button (RST_SW)	Pressing SW1 resets the target MCU (K32W148) which causes board peripherals to reset to their default states and executes the boot code. When SW1 is pressed, the reset LED D7 turns ON if jumper JP18 is shorted. If the target MCU is in Deep Power-Down mode, then it comes out of this mode.
SW2	Non-maskable interrupt (NMI) button (PTD1_NMI PB)	Interrupt push buttons
SW3	Wake-up unit (WUU) button (PTC6_WUU0 PB)	
SW4	Target MCU boot configuration button (BOOT CONFIG)	Helps the target MCU boot code to determine if the MCU should enter into In-System Programming (ISP) mode. The main purpose of this button is to force the target MCU into ISP mode at boot time; to do this, hold down this button while pressing and releasing the reset button, or while connecting power to the board. If an application in the target MCU internal flash is crashing or disabling the SWD port unintentionally, then ISP mode can be useful to recover control of the board. See *K32W1480 Reference Manual* for more information on ISP mode. By default, SW4 is disabled. It can be enabled by shorting jumper JP25.
SW5	Wake-up button (SW_ WAKE_UP)	Wakes up the target MCU (K32W148) from Deep Power-Down mode

Figure 5 shows the circuit diagrams of the K32W148-EVK push buttons.

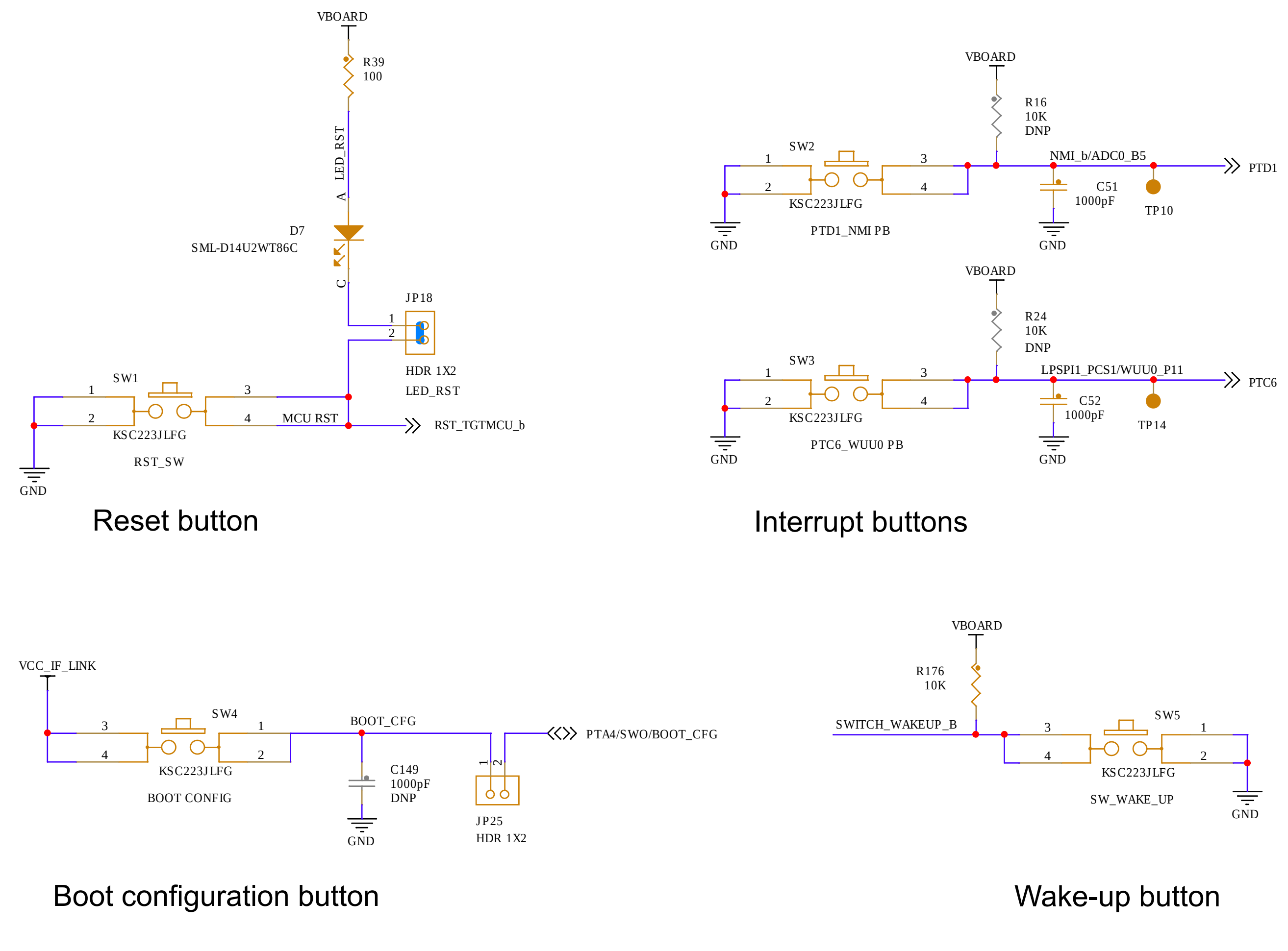

Figure 5. Push button circuit diagram

1.11 LEDs

Table 7 describes the K32W148-EVK light-emitting diodes (LEDs) that correspond to the target MCU. The board also has some MCU-Link specific LEDs, which are described in Section 3.9. The LEDs are shown in Figure 2.

Table 7. K32W148-EVK LEDs

Part identifier	LED color	LED name/function	Description
LED1	Red/green/blue	RGB LED	User application LEDs. Each of these LEDs can be controlled through a user application.
LED2	Blue	General-purpose LED	
LED3	Green	Power-on indicator LED	Indicates system power-on status. When the board is powered up, LED3 turns ON.
D7	Red	Reset LED	Indicates system reset activity. When board reset is initiated, for example, by pressing the reset button, D7 turns ON.

Figure 6 shows the circuit diagrams of the LEDs described in Table 7.

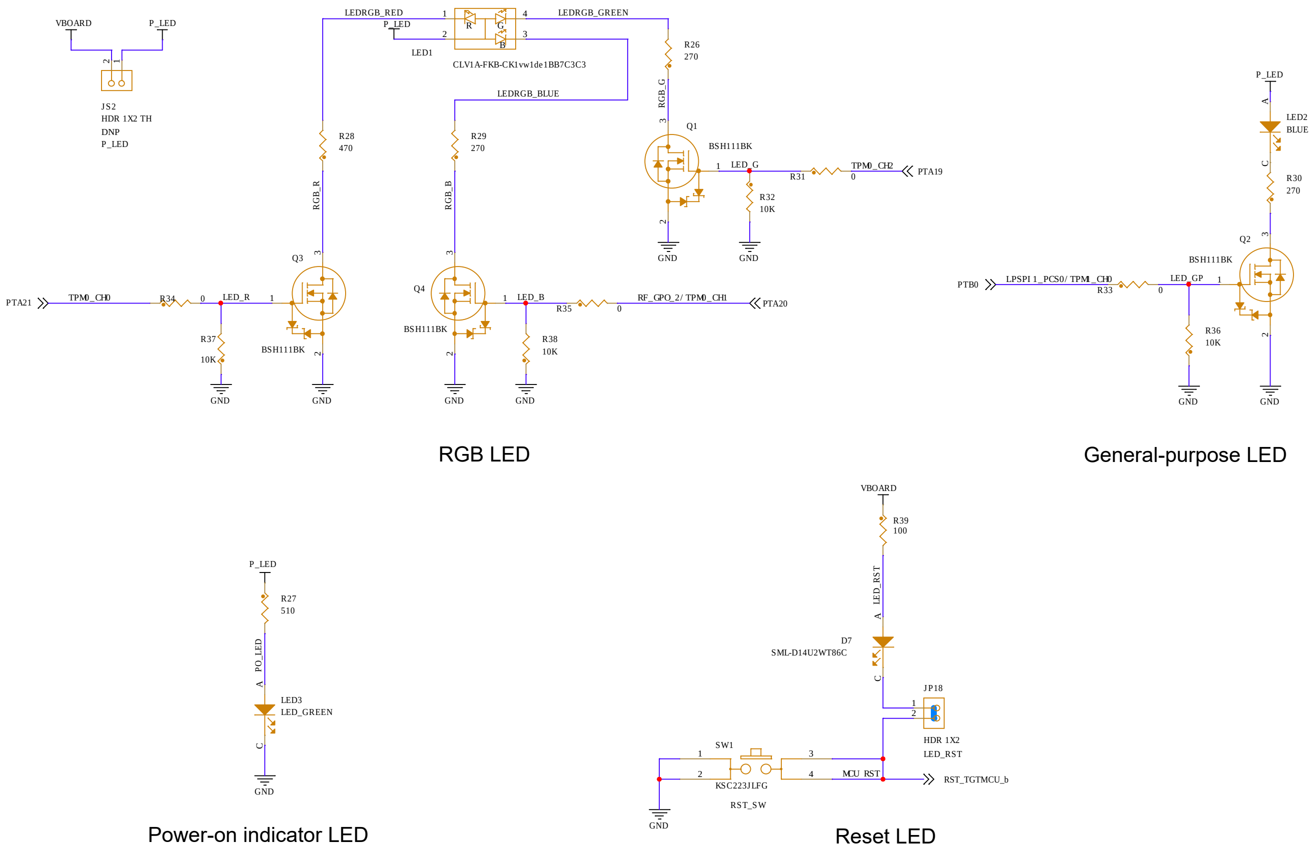

Figure 6. LED circuit diagrams

2 K32W148-EVK functional description

This section describes the features and functions of the K32W148-EVK board. You can use the functionality described in this section as a reference while designing your own target board.

Note: *For more details on the K32W148 MCU, see K32W14x Product Family Data Sheet and K32W1480 Reference Manual.*

2.1 Power supplies

The K32W148-EVK board can be powered up using the following means:

- From an external battery (coincell CR2032/CR2450). Two coincell battery options are provided for more battery lifetime.
- Through the USB micro-B connector (J14), which produces 5 V power (FL_USB_5V0 supply). This supply powers the MCU-Link 3.3 V regulator U15.
- From an external DC supply. Connect to pin 1 of header J8 (not populated on the board) an adapter that can supply 1.8+0.3 V to 3.6+0.3 V DC power (+0.3 V is required due to the voltage drop of diode D1).

The above primary power supplies are used to produce secondary power supplies for the board to power up other board components, including the target MCU, MCU-Link, accelerometer, QSPI NOR flash memory, mikroBUS socket, Arduino socket, Ranger 4 UWB socket, push buttons, and LEDs.

Many power supplies in the K32W148-EVK board are produced through jumpers, which can be used to configure their respective power supplies. This configuration includes enabling/disabling a power supply and changing input power source for a power supply.

Figure 7, Figure 8, and Figure 9 show the circuit diagrams of major K32W148-EVK power supplies.

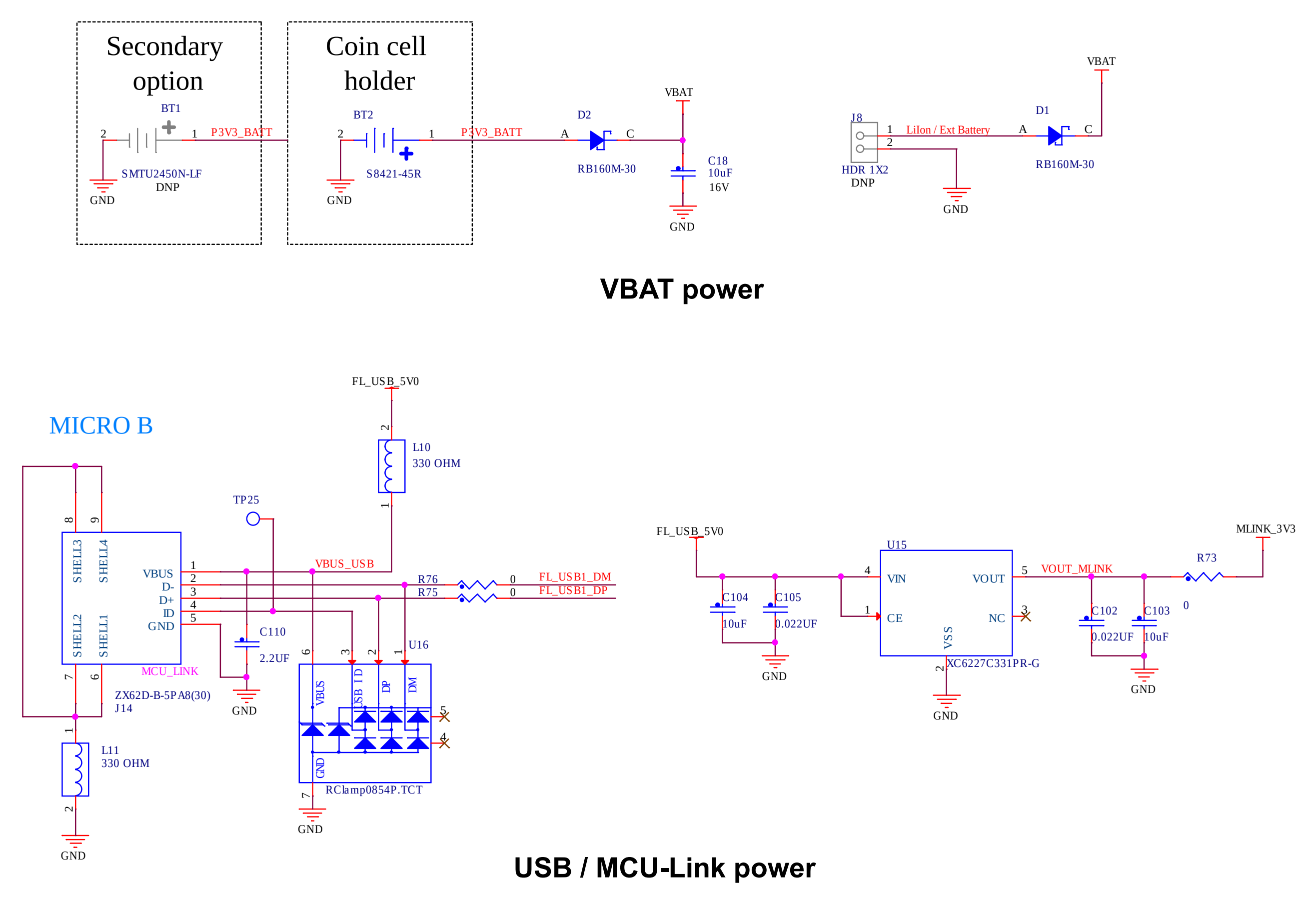

Figure 7. Power supply circuit diagram 1

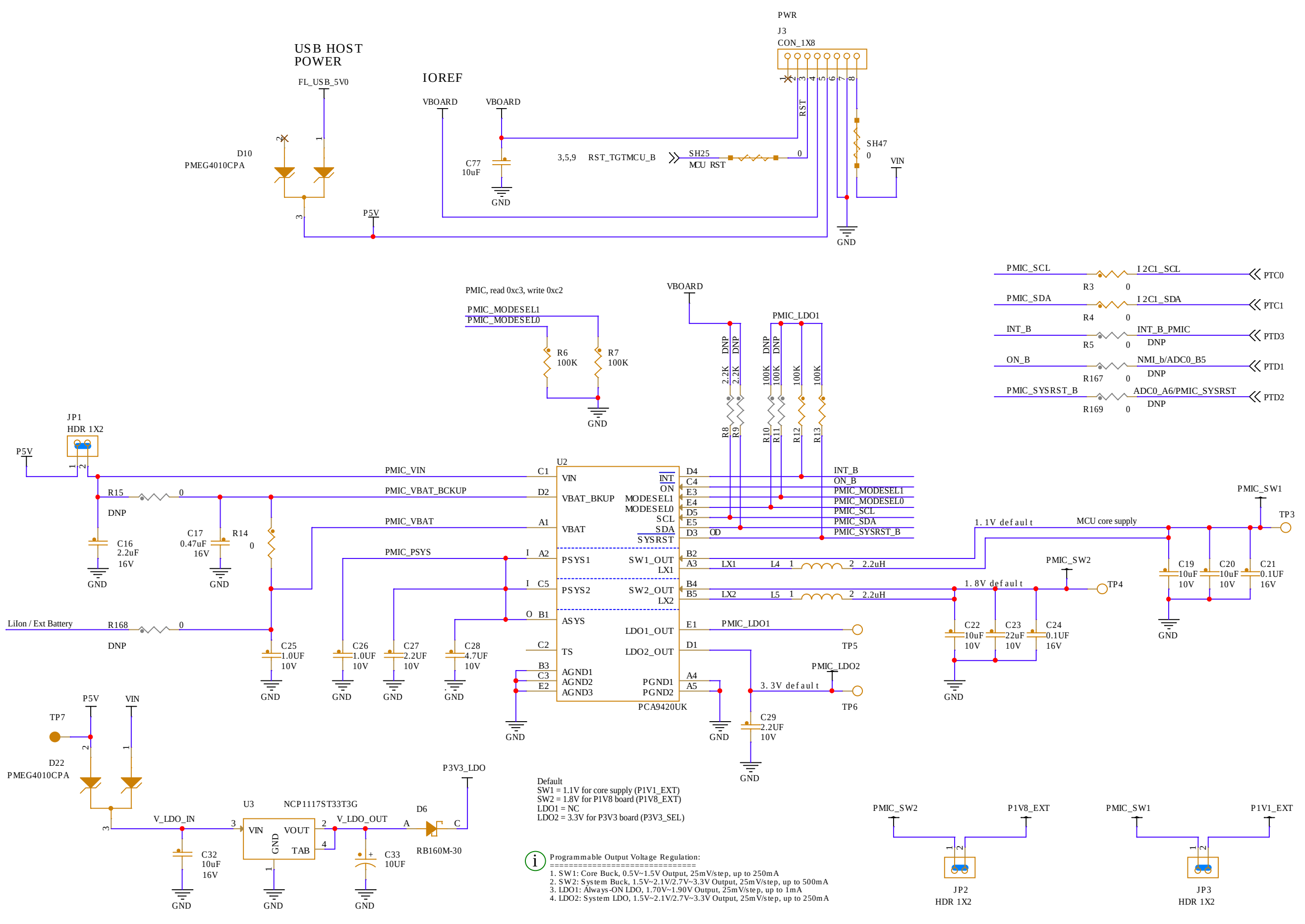

Figure 8. Power supply circuit diagram 2

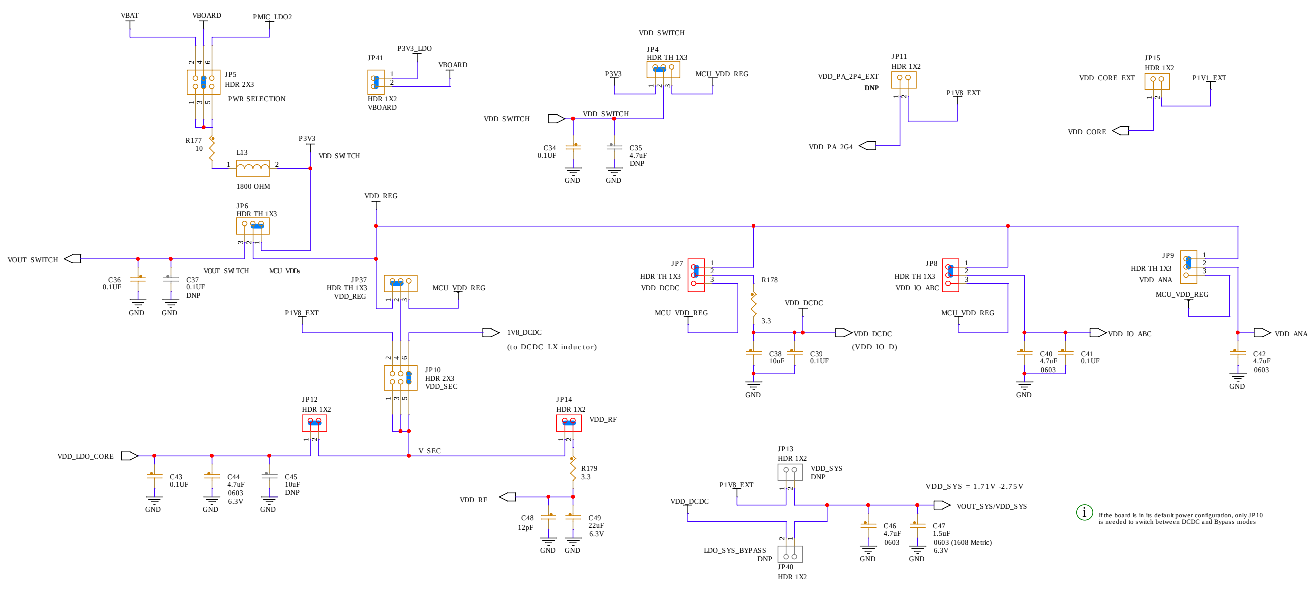

Figure 9. Power supply circuit diagram 3

Table 8 describes the K32W148-EVK power supplies.

Table 8. K32W148-EVK power supplies

Power source	Manufacturing part number	Power supply rail	Description
Coincell battery holder BT2 (primary option)	S8421-45R	VBAT (3.3 V) ***Note:*** *BT1 and BT2 have overlapping footprints (BT2 is used by default).*	• One of the three power supplies used to produce P3V3 supply • VBAT supply from connector J8 can also be used to power up PMIC U2 (PCA9420UK)
Coincell battery holder BT1 (DNP) (secondary option)	SMTU2450N-LF		
Lithium-ion/external battery, connected through header J8 (DNP)	S2B-PH-KL	VBAT (should not exceed 3.6 V)	
MCU-Link USB connector J14	ZX62D-B-5PA8(30) (Hirose Electric)	FL_USB_5V0 (5 V)	• Produces P5V supply • Provides USB1_VBUS power to MCU-Link • Supplies power to MCU-Link 3.3 V regulator U15
From FL_USB_5V0 supply		P5V (5 V)	• Produces P5VA supply, which is used in current measurement circuit • Provides 5 V power to power management integrated circuit (PMIC) U2 (PCA9420UK) that can be controlled through jumper JP1 • One of the two power supplies for linear voltage regulator U3 (NCP1117ST33T3G) • Supplies power to mikroBUS socket connector J13 • Provides 5 V power to Arduino socket power connector J3 • Provides 5 V power to Ranger 4 socket connector J6
U2	PCA9420UK (NXP Semiconductors)	SW1_OUT: PMIC_SW1 (default voltage: 1.1 V, maximum current: 250 mA)	Produces P1V1_EXT supply
		SW2_OUT: PMIC_SW2 (default voltage: 1.8 V, maximum current: 500 mA)	Produces P1V8_EXT supply
		LDO2_OUT: PMIC_LDO2 (default voltage: 3.3 V, maximum current: 250 mA)	Another power supply used to produce P3V3 supply
Arduino socket power connector J3		VIN (7 V - 12 V)	Another power supply for linear voltage regulator U3
U3	NCP1117ST33T3G (ON Semiconductor)	P3V3_LDO (3.3 V)	Produces VBOARD supply through jumper JP41
From VBAT / VBOARD / PMIC_LDO2 supply through jumper JP5 ***Note:*** *When using VBAT supply for producing*		P3V3 (3.3 V)	• Produces VDD_SWITCH supply through jumper JP4 • One of the two power supplies used to produce VDD_REG supply

Table 8. K32W148-EVK power supplies*...continued*

Power source	Manufacturing part number	Power supply rail	Description
P3V3 supply, open jumper JP42 to avoid any power supply issues.			
From P3V3_LDO supply through jumper JP41		VBOARD (3.3 V)	• Produces MCU_VDD_REG supply • Produces VCC_TGMCU supply • Produces VDD_SENSOR supply for powering up the accelerometer • Produces VDD_MEM supply for QSPI NOR flash memory through jumper JS1 • Produces P_LED supply for the target MCU LEDs through jumper JS2 • Third power supply used to produce P3V3 supply • Supplies power to mikroBUS socket connector J12, TCXO Y3 (not populated), and reset LED D7 • Provides 3.3 V powers to Arduino socket power connector J3 • Provides 3.3 V power to Ranger 4 socket connector J6 • Supplies power to the pullup resistor for SW5 push button • Can be used to supply power to the pullup resistors for SW2 and SW3 push buttons
From PMIC_SW1 output of PMIC through jumper JP3		P1V1_EXT (1.1 V)	Produces VDD_CORE supply through jumper JP15
From PMIC_SW2 output of PMIC through jumper JP2		P1V8_EXT (1.8 V)	• Produces VDD_PA_2G4 supply through jumper JP11 (DNP) • One of the three power supplies used to produce V_SEC supply • One of the two power supplies used to produce VOUT_SYS / VDD_SYS supply
From P3V3 supply through jumper JP4		VDD_SWITCH	Provides VDD_SWITCH power to the target MCU
From P1V1_EXT supply through jumper JP15		VDD_CORE	Provides external VDD_CORE power (0.84 V – 1.21 V) to the target MCU
From P1V8_EXT supply through jumper JP11 (DNP)		VDD_PA_2G4	Provides external VPA_2P4GHZ power to the target MCU
From P3V3 supply or VOUT_SWITCH power of the target MCU through jumper JP6		VDD_REG	• One of the two input power supplies for jumper JP37 • One of the two power supplies used to produce VDD_DCDC supply • One of the two power supplies used to produce VDD_IO_ABC supply • One of the two power supplies used to produce VDD_ANA supply

Table 8. K32W148-EVK power supplies...*continued*

Power source	Manufacturing part number	Power supply rail	Description
From VBOARD supply through jumper JP35		MCU_VDD_REG	• Another input power supply for jumper JP37 • Another power supply used to produce VDD_DCDC supply • Another power supply used to produce VDD_IO_ABC supply • Another power supply used to produce VDD_ANA supply • One of the power supplies used in current measurement circuit
From VDD_REG / MCU_VDD_REG supply through jumper JP7		VDD_DCDC	• Provides VDD_DCDC power to the target MCU • Another power supply used to produce VOUT_SYS / VDD_SYS supply
From VDD_REG / MCU_VDD_REG supply through jumper JP8		VDD_IO_ABC (default value: 3.3 V)	Provides VDD_IO_ABC power to the target MCU
From VDD_REG / MCU_VDD_REG supply through jumper JP9		VDD_ANA	Provides VDD_ANA power to the target MCU
From DCDC_LX power of the target MCU		1V8_DCDC	Another power supply used to produce V_SEC supply
From 1V8_DCDC supply / jumper JP37 output / P1V8_EXT supply through jumper JP10		V_SEC	• Produces VDD_LDO_CORE supply through jumper JP12 • Produces VDD_RF supply through jumper JP14
From V_SEC supply through jumper JP12		VDD_LDO_CORE	Provides VDD_LDO_CORE power to the target MCU
From V_SEC supply through jumper JP14		VDD_RF	Provides VDD_RF power to the target MCU
From P1V8_EXT supply through jumper JP13 (DNP)		VOUT_SYS / VDD_SYS	Provides VDD_SYS power (1.71 V – 2.75 V) to the target MCU. ***Note:*** *If populated, JP13 must be programmed to receive 2.5 V power to produce the VOUT_SYS / VDD_SYS supply.*
From VDD_DCDC supply through jumper JP40 (DNP)			
From VBOARD supply		VCC_TGMCU	• Produces VCC_IF_LINK supply • Supplies power to four pole double throw (4PDT) switch U18 • One of the two power supplies for voltage translators (U23 and U24) and I2C repeater / voltage translator U27
From VCC_TGMCU through jumper JP28		VCC_IF_LINK	• Supplies power to the target MCU JTAG / serial wire debug (SWD) connector (J16), boot configuration button (SW4), buffer U22, and PIO0_31 pin of MCU-Link • One of the two power supplies for voltage translators U17, U19, U20, U21, and U29

Table 8. K32W148-EVK power supplies...*continued*

Power source	Manufacturing part number	Power supply rail	Description
U15	XC6227C331 PR-G (Torex Semiconductor)	MLINK_3V3 (3.3 V)	• Produces MCU_LINK_3V3 supply • Supplies power to MCU-Link, MCU-Link SWD connector (J15) (DNP), MCU-Link LEDs, PIO1_8 pin of MCU-Link, and current measurement circuit • Another power supply for voltage translators (U23 and U24) and I2C repeater / voltage translator (U27) • Another power supply for voltage translators U17, U19, U20, U21, and U29
From MLINK_3V3 supply		MCU_LINK_3V3 (3.3 V)	Supplies VDD and USB 3.3 V powers to MCU-Link

The K32W148-EVK board supports current measurement for board power supplies through the onboard MCU-Link debug probe. For more details, see Section 3.8.

2.1.1 Power modes

The K32W148-EVK board supports the following power modes:

- DC-DC Buck mode: This is the default power mode of the board. In this mode, the DC-DC converter of the target MCU is enabled.
- DC-DC Bypass mode: In this mode, the target MCU DC-DC converter is disabled
- PMIC mode
- Smart Power Switch mode

Changing power mode of the board from default mode (DC-DC Buck mode) to another power mode (DC-DC Bypass mode, PMIC mode, or Smart Power Switch mode) involves reconfiguring some of the board jumpers. Changing power mode also involves cutting traces at the bottom side of the PCB.

Table 9 explains how to switch the K32W148-EVK board from one power mode to another power mode by changing jumper settings.

Table 9. Power mode switching

Current power mode	Desired power mode	Required jumper setting changes
DC-DC Buck mode	DC-DC Bypass mode	Change jumper JP10 setting from 5-6 shorted to 3-4 shorted
DC-DC Bypass mode	DC-DC Buck mode	Change jumper JP10 setting from 3-4 shorted to 5-6 shorted
DC-DC Buck mode	PMIC mode[1]	• Change jumper JP10 setting from 5-6 shorted to 1-2 shorted • Change jumper JP5 setting from 3-4 shorted to 5-6 shorted • Short jumper JP15 • Short jumper JP11 (if populated)
PMIC mode	DC-DC Buck mode	• Change jumper JP10 setting from 1-2 shorted to 5-6 shorted • Change jumper JP5 setting from 5-6 shorted to 3-4 shorted • Open jumper JP15

Table 9. Power mode switching*...continued*

Current power mode	Desired power mode	Required jumper setting changes
		• Open jumper JP11 (if populated)
DC-DC Bypass mode	PMIC mode[1]	• Change jumper JP10 setting from 3-4 shorted to 1-2 shorted • Change jumper JP5 setting from 3-4 shorted to 5-6 shorted • Short jumper JP15 • Short jumper JP11 (if populated)
PMIC mode	DC-DC Bypass mode	• Change jumper JP10 setting from 1-2 shorted to 3-4 shorted • Change jumper JP5 setting from 5-6 shorted to 3-4 shorted • Open jumper JP15 • Open jumper JP11 (if populated)
DC-DC Buck mode or DC-DC Bypass mode	Smart Power Switch mode	Change jumper JP6 setting from 1-2 shorted to 2-3 shorted
Smart Power Switch mode	DC-DC Buck mode	• Change jumper JP6 setting from 2-3 shorted to 1-2 shorted • Change jumper JP10 setting to 5-6 shorted
Smart Power Switch mode	DC-DC Bypass mode	• Change jumper JP6 setting from 2-3 shorted to 1-2 shorted • Change jumper JP10 setting to 3-4 shorted

[1] To enable Full PMIC mode, additionally disable Core LDO and DC-DC of the target MCU by configuring the CNTRL register of System Power Controller (SPC) in software as follows:
1. Set CNTRL[CORELDO_EN] = 0
2. Set CNTRL[DCDC_EN] = 0

2.1.2 DC-DC inductor

The K32W148-EVK board uses a 1 µH DC-DC inductor L1 (TDK MLZ2012A1R0WTD25). The inductor is enabled when the board is configured in DC-DC Buck mode. Figure 10 shows the DC-DC inductor circuit diagram of the K32W148-EVK board.

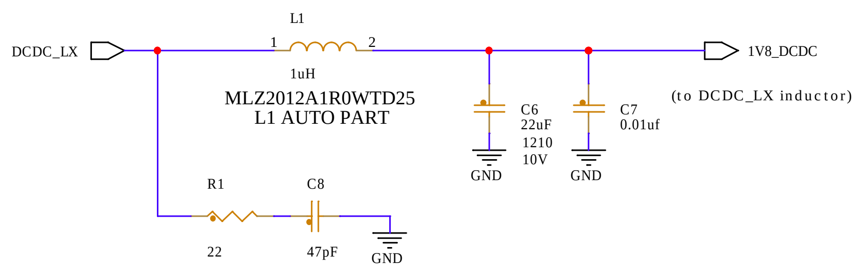

Figure 10. DC-DC inductor circuit diagram

Other recommended inductors are LQM18PH1R0MFRL from Murata and 78438336010 from Würth Elektronik. Figure 11 shows other possible inductor options.

Inductor type	Value	Manufacturer	Part number	Automotive qualified AEC-Q200	ESR (Ω)	Isat (mA)	Maximum DC current (mA)	Temperature range (°C)	Size (LxWxH) mm
SMD shielded Multilayer ferrite	1 µH	TDK	MLZ2012A1R0WTD25	Yes	0.13	280	900	-55 to +125	2.2x1.45x1.05
SMD shielded Multilayer ferrite	1 µH	TDK	MLZ1608A1R0WTD25	Yes	0.15	190	600	-55 to +125	1.6x0.8x0.8
SMD shielded Thin film Metal core	1 µH	TDK	TFM2016ALMA1R0MTAA	Yes	0.05	3400	3700	-55 to +125	2x1.6x1
Wound (Metal)	1 µH	Murata	DFE322512F-1R0M	Yes	0.032	4800		-40 to +125	3.2x2.5x1.2
SMD shielded Multilayer ferrite	1 µH	Murata	LQM18PH1R0MFRL	Yes	0.29	600	950	-40 to +125	1.6x0.8x0.5
SMD shielded Multilayer ferrite	1 µH	Murata	LQM21PH1R0NGCD	Yes	0.1–0.13	200		-40 to +125	2x1.25x0.9
Chip coil	1 µH	Murata	LHQ3NPH1R0MM	Yes	0.03	2350	3000	-40 to +125	3x3x1.4
Chip coil	1 µH	Murata	LQW21FT1R0M0H	Yes	0.13	700	700	-40 to +125	2x1.2x1.6
SMD shielded Multilayer ferrite	1 µH	Chilisin	AKPB001608DZ1R0MA2	Yes	0.2	200	700	-40 to +125	1.6x0.8x0.8
SMD shielded Multilayer ferrite	1 µH	Chilisin	AKPB002012101R0MA2	Yes	0.12	650	1200	-40 to +125	2x1.25x1
SMD shielded Multilayer ferrite	1 µH	Chilisin	AKPB002016101R0MA6	Yes	0.085	850	1300	-40 to +125	2x1.6x1
T-core	1 µH	Cyntec	VCTA20161B-1R0MS6	Yes	0.041–0.048	3100	3800	-55 to +165	2x1.6x1.2
Wound shielded	1 µH	Wurth	78 438 336 010	Yes	0.032	4000	5000	-40 to +125	3x3x2

Figure 11. DC-DC inductor options

Choosing right DC-DC inductor for your target board is very important. While selecting a DC-DC inductor, look for the following specifications:

- Inductor value: 1 µH or 1.5 µH
- ESR: < 0.3 Ω
- Saturation current (Isat): > 300 mA
- Self-resonant frequency: > 50 MHz

2.2 Clocks

Table 10 describes the clocks available on the K32W148-EVK board.

Table 10. K32W148-EVK clocks

Clock generator	Clock	Frequency	Destination
Y1: Crystal (NX1612SA 32MHZ EXS00A CS14160)	[XTAL, EXTAL]_32M	32 MHz	Target MCU (K32W148)
Y2: Crystal (NX2012SE 32.768kHz EXS00A-MU01517)	XTAL32K / EXTAL32K	32.768 kHz	Target MCU
Y3: Temperature compensated crystal oscillator (TCXO) (NT2016SB-32MHZ-NSA3561B) (DNP)	EXTAL_RF	32 MHz	Target MCU
Y4: Crystal	MCU_LINK_[P, N]_16MHz	16 MHz	MCU-Link (LPC55S69)

In the K32W148-EVK board, the target MCU requires the following two clocks:

- 32 MHz clock (with ±30 ppm accuracy): Provides clock inputs to the Arm Cortex-M33 core and 2.4 GHz radio
- 32.768 kHz clock: Provides an accurate low-power timebase and acts as real-time clock (RTC) and low-power RF clock

By default, the Y1 crystal provides a 32 MHz clock to the target MCU. Alternatively, Y3 TCXO (not populated on the board) can be used to provide a 32 MHz clock to the target MCU.

Figure 12 shows the circuit diagram of the Y1 crystal that provides 32 MHz clock. Internal load capacitors provide the crystal load capacitance, and they can be adjusted to tune the center frequency of the crystal. For information on how to tune the frequency, see *K32W148 Connectivity Test Tool User Guide for 802.15.4* (AN13687).

To measure the 32 MHz crystal frequency, program the XTAL (XTAL_RF) signal to provide buffered output clock signal.

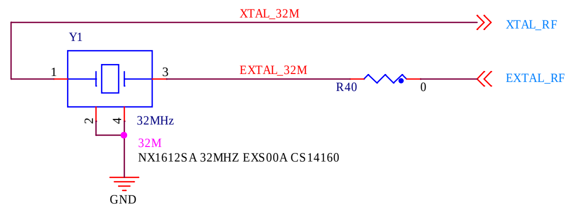

Figure 12. 32 MHz clock circuit diagram

Figure 13 shows the circuit diagram of the Y2 crystal that provides 32.768 kHz clock. Internal load capacitors provide the crystal load capacitance. 0 Ω resistors are provided to bypass the Y2 crystal which gives two additional GPIOs, PTD4 and PTD5, to the I/O headers.

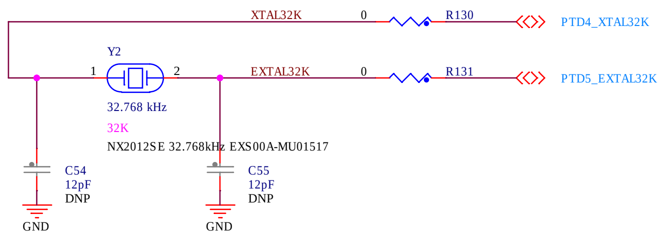

Figure 13. 32.768 kHz clock circuit diagram

2.3 LPUART interface

The target MCU (K32W148) has two Low-Power Universal Asynchronous Receiver/Transmitter (LPUART) modules: LPUART0 and LPUART1. The K32W148-EVK board supports connections to both LPUART modules. Figure 14 shows the LPUART diagram of the K32W148-EVK board.

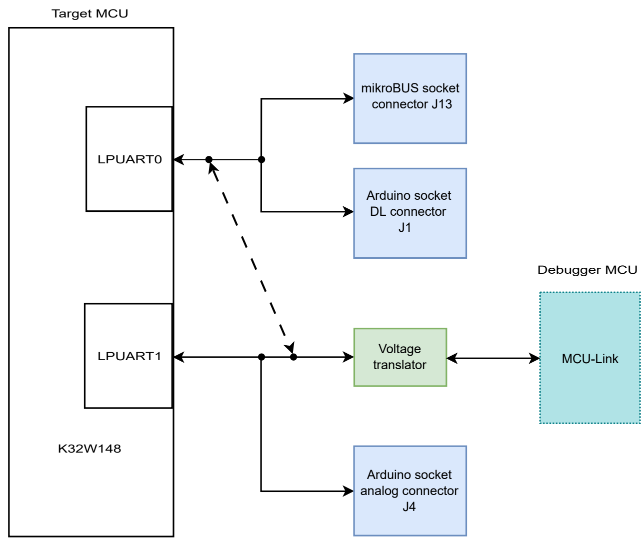

Figure 14. LPUART diagram

By default, MCU-Link is connected to the LPUART1 controller. It can be connected to the LPUART0 controller by changing the setting of each of the jumpers JP16 and JP17 from 1-2 shorted to 2-3 shorted.

Table 11 provides more details on LPUART connections in the K32W148-EVK board.

Table 11. K32W148-EVK LPUART connections

LPUART module	Voltage translators	Connector / MCU-Link
LPUART0		mikroBUS socket connector J13. It allows a UART connection between the target MCU and plugged-in click board.
		Arduino socket DL connector J1. It allows a UART connection between the target MCU and plugged-in Arduino board.
LPUART1	Bidirectional voltage translator U23 (74AVC4TD245 BQ). It supports a USB-to-UART bridge between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to VCC_TGMCU and vice versa. The USB-to-UART bridge can be disabled by shorting jumper JP21.	MCU-Link (LPC55S69). The USB-to-UART bridge between MCU-Link and the target MCU can be used for debugging the target MCU from MCU-Link.
		Arduino socket analog connector J4. It allows a UART connection between the target MCU and plugged-in Arduino board.

2.4 LPSPI interface

The target MCU has two Low-Power Serial Peripheral Interface (LPSPI) modules: LPSPI0 and LPSPI1. Each LPSPI module can act as a SPI master or slave, and supports four peripheral chip selects (PCSes): PCS0, PCS1, PCS2, and PCS3. The K32W148-EVK board only supports LPSPI1 module, with all its PCSes.

Figure 15 shows the K32W148-EVK LPSPI diagram.

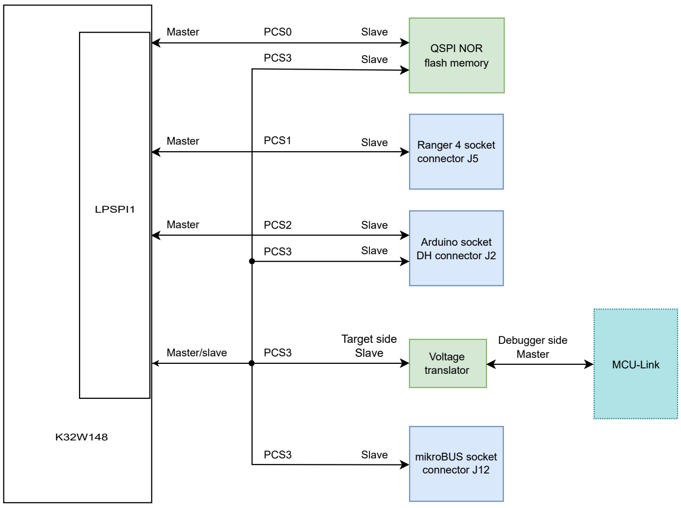

Figure 15. LPSPI diagram

Table 12 explains the K32W148-EVK LPSPI connections.

Table 12. K32W148-EVK LPSPI connections

PCS	Voltage translator	Slave device / MCU-Link
PCS0		16 Mbit (2 MB) QSPI NOR flash memory U4 (AT25XE161D-MAHN-T) for over-the-air (OTA) programming or for storing non-volatile system data or parameters
PCS1		Ranger 4 socket connector J5. It allows a SPI connection between the target MCU and plugged-in Ranger 4 UWB board.
PCS2		Arduino socket DH connector J2. It allows a SPI connection between the target MCU and plugged-in Arduino board.
PCS3	Bidirectional voltage translator U24 (74AVC4 TD245BQ). It supports a USB-to-SPI bridge between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to VCC_TGMCU and vice versa. By default, the USB-to-SPI bridge is disabled, as jumper JP34 is shorted. It can be enabled by opening JP34.	MCU-Link (LPC55S69). The USB-to-SPI bridge between MCU-Link and the target MCU can be used for debugging the target MCU from MCU-Link.
		QSPI NOR flash memory (U4)
		Arduino socket DH connector J2
		mikroBUS socket connector J12. It allows a SPI connection between the target MCU and plugged-in click board.

Figure 16 shows the circuit diagram of the QSPI NOR flash memory.

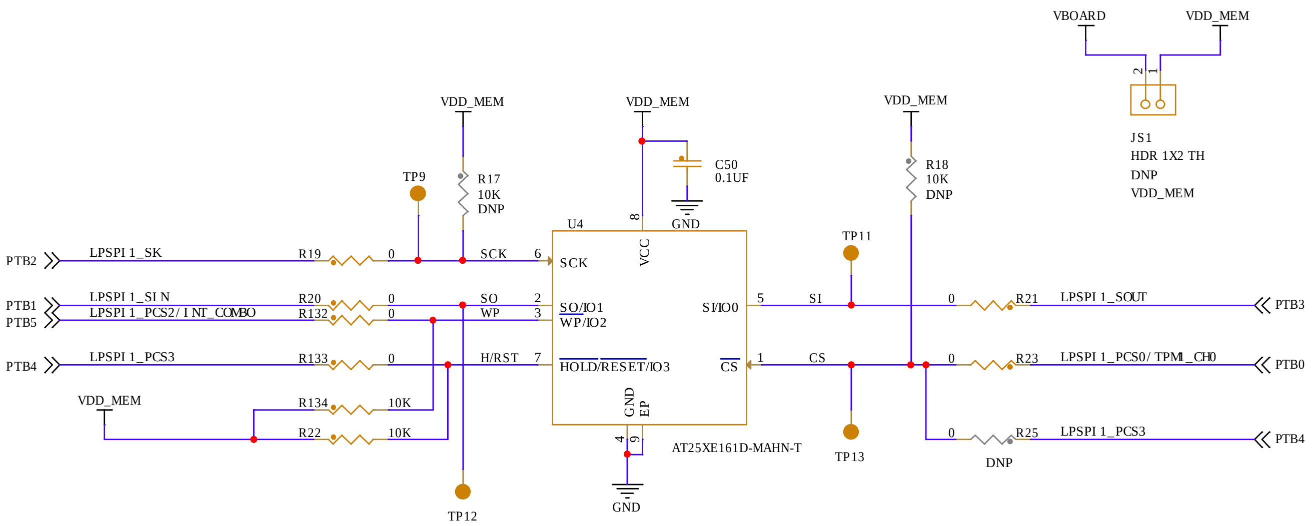

Figure 16. QSPI NOR flash memory circuit diagram

The above circuit diagram is explained below:

- The QSPI NOR flash memory (U4) is powered from the VDD_MEM supply, which is derived from the VBOARD supply through jumper JS1
- By default, JS1 is not populated but its pin 1 and pin 2 positions are connected through a physical (trace) connection on the secondary (bottom) side of the PCB. Therefore, QSPI NOR flash memory is connected to power, by default.
- When populated, JS1 can be used to enable/disable the VDD_MEM supply. If populating the jumper to allow manual connection/disconnection, cut the trace connection between its pin 1 and pin 2 positions.
- Current drawn by U4 can be measured via JS1 through an ammeter
- Discrete pullup resistors are provided for SPI signals
- The SPI signals can be shared with other peripherals through Arduino socket DH connector J2

2.5 LPI2C interface

The target MCU (K32W148) has two Low-Power Inter-Integrated Circuit (LPI2C) modules: LPI2C0 and LPI2C1. The K32W148-EVK board only supports LPI2C1 module. Figure 17 shows the K32W148-EVK LPI2C diagram.

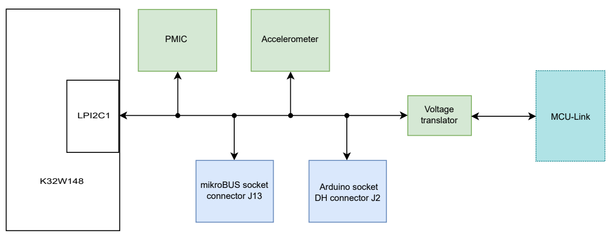

Figure 17. LPI2C diagram

Table 13 shows the K32W148-EVK LPI2C bus device map.

Table 13. K32W148-EVK LPI2C bus device map

LPI2C module	8-bit I2C address	Device	Description
		K32W148 (U1)	Target MCU. It acts as I2C master for all I2C connections on the board except for I2C connection with MCU-Link.

Table 13. K32W148-EVK LPI2C bus device map*...continued*

LPI2C module	8-bit I2C address	Device	Description
LPI2C1		LPC55S69JBD100 (U13)	MCU-Link. A USB-to-I2C bridge is supported between MCU-Link and the target MCU using an I2C repeater / voltage translator U27. This I2C connection can be used for debugging the target MCU (I2C slave) from MCU-Link (I2C master). Jumper JP30 can be used to enable/ disable the USB-to-I2C bridge.
	• Read address: 0xC3 • Write address: 0xC2	PCA9420UK (U2)	Power management integrated circuit (PMIC)
	• Read addresses: 0x33, 0x31 • Write addresses: 0x32, 0x30	FXLS8964AF (U12)	Accelerometer
	I2C address is defined by the plugged-in click board	mikroBUS socket connector J13	Allows an I2C connection between the target MCU and plugged-in click board
	I2C address is defined by the plugged-in Arduino board	Arduino socket DH connector J2	Allows an I2C connection between the target MCU and plugged-in Arduino board

2.6 RF interface

The target MCU integrates a high-performance radio transceiver, which includes a Bluetooth Low Energy (LE) version 5.3 radio and an IEEE 802.15.4 radio. The radio transceiver operates in the 2.4 GHz frequency band supporting the following types of modulations:

- Gaussian frequency shift keying (GFSK) modulations for Bluetooth LE 5.3
- Offset quadrature phase-shift keying (OQPSK) modulations for IEEE 802.15.4

The K32W148-EVK board provides a small-footprint, low-cost RF interface for users to begin application development. Table 14 describes the features of the K32W148-EVK RF interface.

Table 14. RF interface features

Feature	Description
Input/output port	Single-ended RF input/output port (one of the following ports is available at a time): • PCB-integrated inverted-F antenna ANT1 (enabled by default) • SMA connector J7 (SLE-100010072-F01)
External component requirement	Very few external components are required
Output power	Programmable output power from -30 dBm to +10 dBm at the SMA connector in DC-DC Bypass, DC-DC Buck, or Smart Power Switch mode
Typical receiver sensitivity	• -103 dBm for GFSK applications (250 kbit/s GFSK-BT = 0.5, h = 0.5) • -106 dBm for Bluetooth LE LR 125 kbit/s applications • -102 dBm for Bluetooth LE LR 500 kbit/s applications • -97.5 dBm for Bluetooth LE 1 Mbit/s applications • -95 dBm for Bluetooth LE 2 Mbit/s applications • -103 dBm for IEEE 802.15.4 radio applications (packet error rate (PER) = 1%)

Figure 18 shows the K32W148-EVK RF diagram.

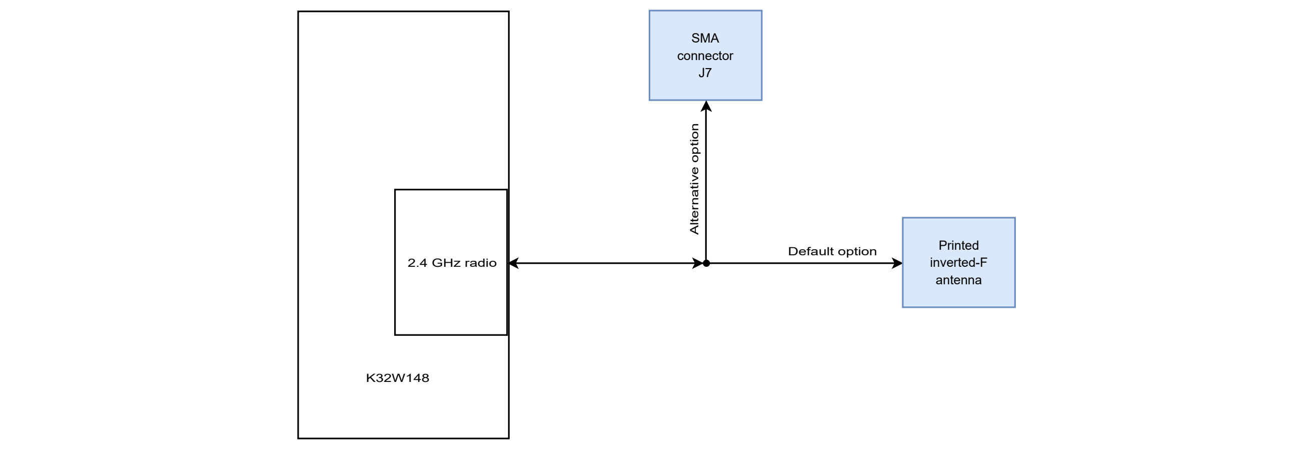

Figure 18. RF diagram

Figure 19 shows the K32W148-EVK RF circuit diagram.

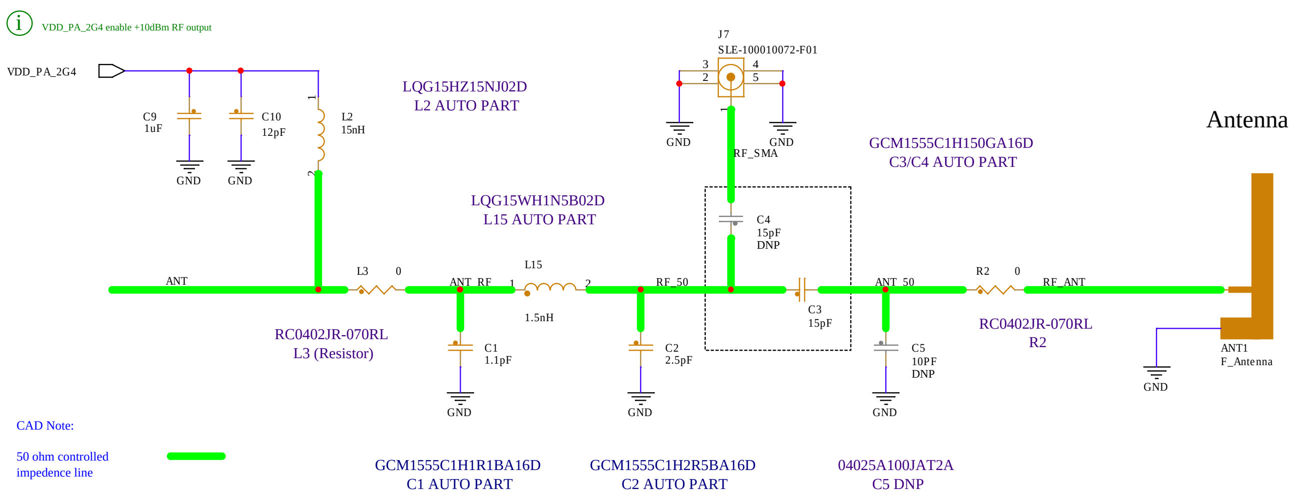

Figure 19. RF circuit diagram

The above circuit diagram is explained below:

- A 50 Ω controlled impedance line is used to connect various components on the RF circuit
- A minimum network required to match the antenna pin of the target MCU is provided through capacitors C1 and C2 and inductor L15
- Another capacitor C5 (DNP) can be added to match the inverted-F antenna to the 50 Ω controlled impedance line
- By default, the inverted-F antenna is enabled and the SMA connector is disabled. The inverted-F antenna can be disabled and the SMA connector can be enabled by moving the 15 pF capacitor from C3 to C4 (DNP).
- For +10 dBm RF output, the VDD_RF supply should be sourced with at least 2.5 V

2.7 Accelerometer

The K32W148-EVK board has a 3-axis, 12-bit, compact digital accelerometer U12 (NXP FXLS8964AF) with ±2g / ±4g / ±8g / ±16g acceleration support and selectable I2C. It is designed to be used for applications requiring ultra-low-power wake-up on motion. With AEC-Q100 qualification and an extended temperature range (–40 ℃ to +105 ℃), this device is an ideal choice for automotive and industrial IoT "motion sensing" applications.

Figure 20 shows the accelerometer circuit diagram.

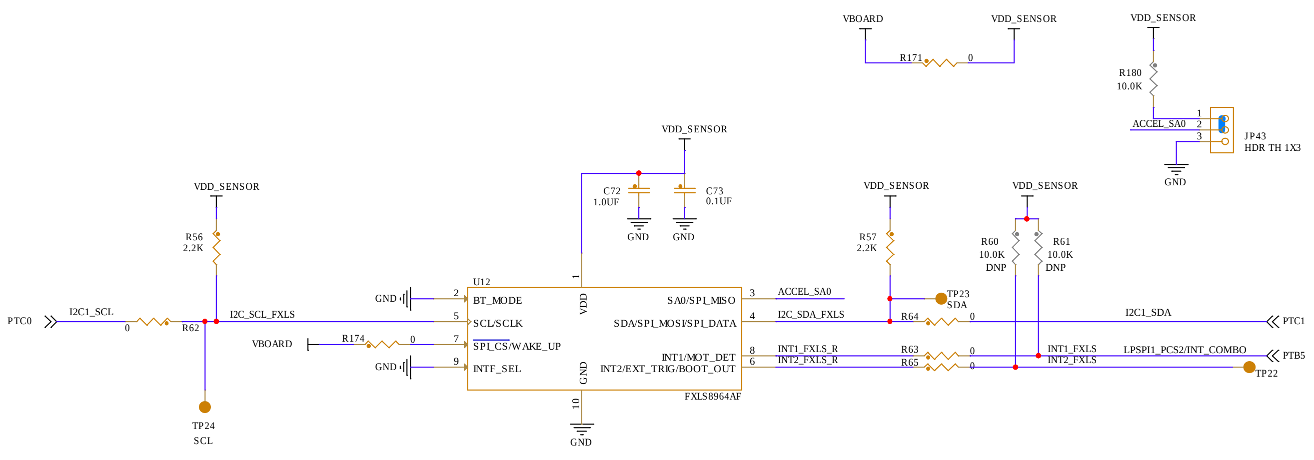

Figure 20. Accelerometer circuit diagram

The above circuit diagram is explained below:

- The accelerometer is powered by the VDD_SENSOR supply, which is tied to the VBOARD supply for voltage level compatibility with the target MCU inputs/outputs
- Discrete pullup resistors are provided for I2C bus lines
- The I2C data signal can be shared with other peripherals through Arduino socket DH connector J2
- One interrupt signal is routed to the accelerometer

Jumper JP43 can be used to configure I2C address of the accelerometer when the accelerometer acts a secondary I2C device, as explained in the table below.

Table 15. Accelerometer I2C address configuration

JP43 setting	8-bit I2C read address	8-bit I2C write address
1-2 shorted (default setting)	0x33	0x32
2-3 shorted	0x31	0x30

2.8 mikroBUS socket

A mikroBUS socket is a pair of 1x8 position receptacles (connectors) with a proprietary pin configuration. It allows maximum hardware expandability with smallest number of pins.

The K32W148-EVK board has a mikroBUS socket with two 1x8 position receptacles, J12 and J13. Figure 21 shows the pinouts of the mikroBUS socket connectors.

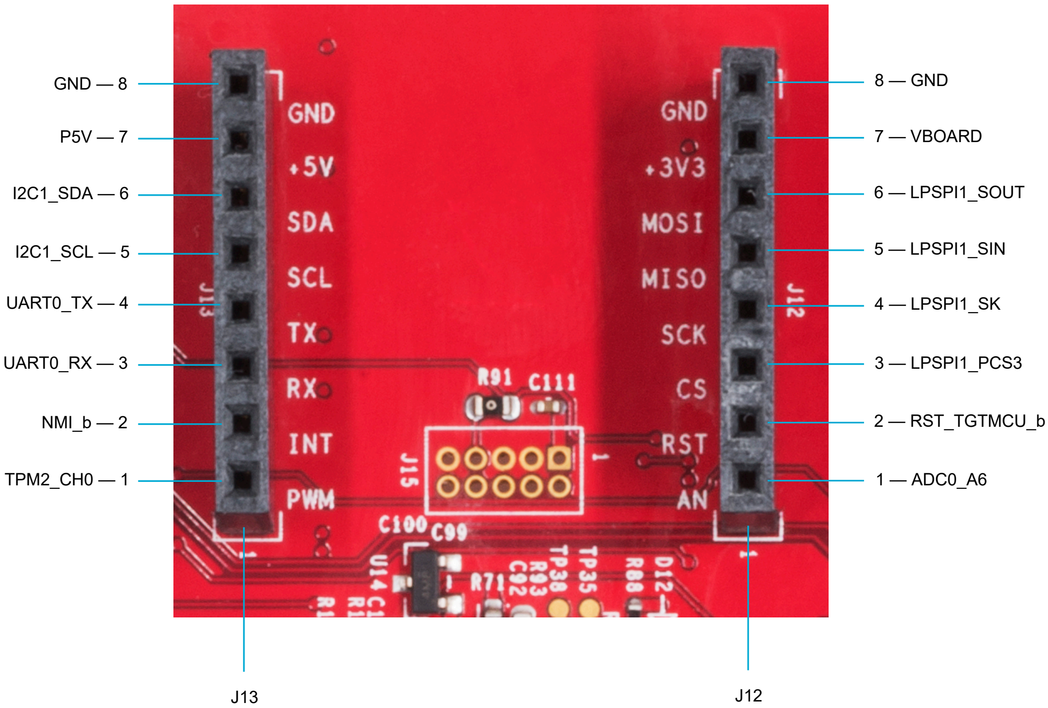

Figure 21. mikroBUS socket connector pinouts

The K32W148-EVK mikroBUS socket supports different types of add-on boards, called *click boards*, which are plug-and-play solutions to add new functionality to the board design. A click board has a pair of 1x8 pin headers that connect to the two receptacles of a mikroBUS socket.

MikroElektronika (MIKROE) is one of the manufactures of click boards. You can find details of some example click boards for the K32W148-EVK mikroBUS socket at MIKROE website.

2.9 Arduino socket

The K32W148-EVK board has an Arduino socket with four connectors (1x8, 1x10, 1x8, and 1x6 position receptacles). The two 1x8 position receptacles are placed diagonally opposite to each other. The Arduino socket is pin-compatible with an Arduino Uno revision 3 (R3) board.

Table 16 describes the connectors of the Arduino socket.

Table 16. Arduino socket connectors

Part identifier	Connector type	Description
J1	1x8 position receptacle	Destination address (low 32 bits) (DL)
J2	1x10 position receptacle	Destination address (high 32 bits) (DH)
J3	1x8 position receptacle	Power (PWR)
J4	1x6 position receptacle	Analog (AN)

Figure 22 shows the pinouts of the Arduino socket connectors.

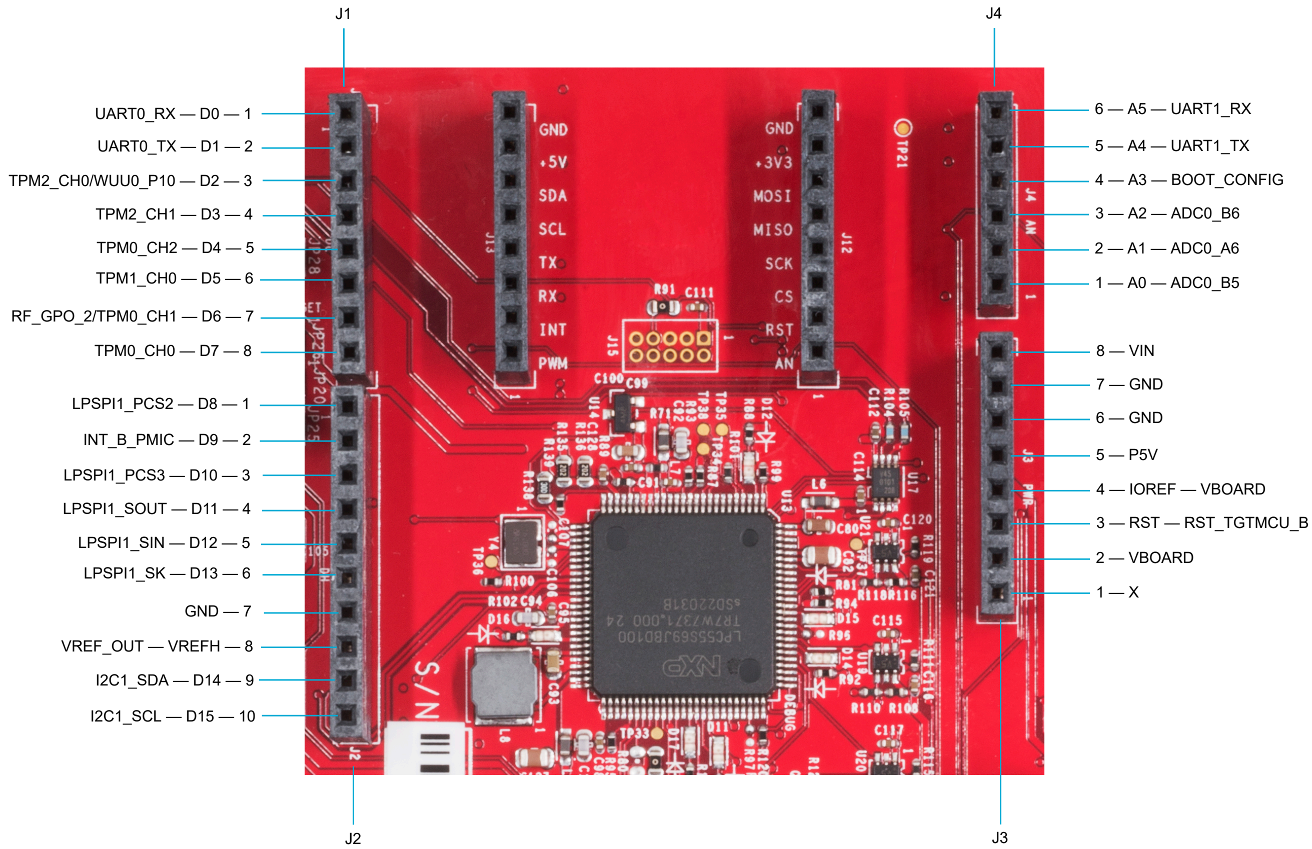

Figure 22. Arduino socket connector pinouts

2.10 Ranger 4 UWB socket

The K32W148-EVK board has a Ranger 4 ultra-wideband (UWB) socket with the following two connectors:

- J5: 2x10 position receptacle
- J6: 2x8 position receptacle

The Ranger 4 socket is pin-compatible with a Ranger 4 UWB board.

Figure 23 shows the pinouts of the Ranger 4 UWB socket connectors.

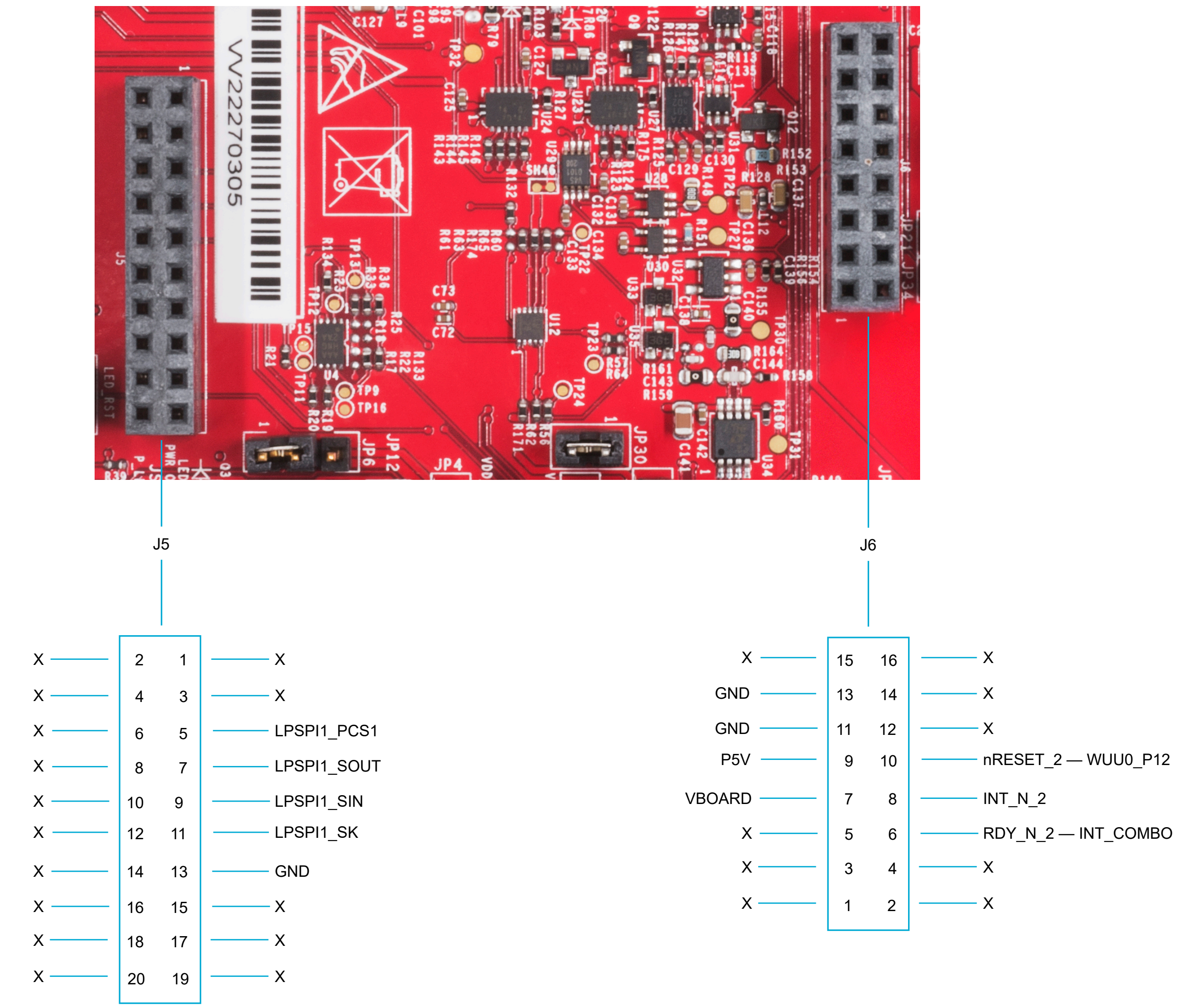

Figure 23. Ranger 4 UWB socket connector pinouts

2.11 GPOs

Table 17 lists the target MCU pin that is used as general-purpose output (GPO) on the K32W148-EVK board. This pin is used for GPO purpose when not used for its primary purpose.

Table 17. Target MCU GPO

Target MCU pin	Primary function (device)	GPO function (device)	GPO mode control
PTA20	TPM0_CH1 (RGB LED)	RF_GPO_2 (Arduino socket DL connector J1)	GPO mode is controlled through configuration resistor R35

2.12 Board errata

No known errata.

2.13 Board operating conditions

The operating temperature range for the K32W148-EVK board is -40 ℃ to +85 ℃. See *K32W14x Product Family Data Sheet* for more details on device operating conditions.

3 MCU-Link OB debug probe

This section describes the MCU-Link OB debug probe and explains how to connect it to the target MCU (K32W148).

3.1 MCU-Link overview

MCU-Link is a debug probe architecture jointly developed by NXP and Embedded Artists. The MCU-Link architecture is based on the LPC55S69 MCU, which is based on the Arm Cortex-M33 core.

NXP uses MCU-Link OB on its evaluation kits (EVK boards), for example, K32W148-EVK. MCU-Link is also available as a standalone debug probe in the following two types:

- MCU-Link (base model): For more details, see https://www.nxp.com/design/microcontrollers-developer-resources/mcu-link-debug-probe:MCU-LINK
- MCU-Link Pro (fully featured): For more details, see https://www.nxp.com/design/microcontrollers-developer-resources/mcu-link-pro-debug-probe:MCU-LINK-PRO

All these MCU-Link probes use the same firmware.

NXP EVK boards have some configuration strap pins that determine the type of MCU-Link used. The MCU-Link firmware reads these pins during boot process to determine the MCU-Link type.

In the K32W148-EVK board, some of the strap pins can be configured via jumpers; however, most of these configuration changes are made before powering up the board.

3.1.1 Supported firmware options

Supported firmware options for the K32W148-EVK MCU-Link debug probe are described below:

- By default, the K32W148-EVK MCU-Link debug probe is programmed with firmware based on CMSIS-DAP protocol from NXP. This firmware also supports all other features supported in hardware. For details on how to update MCU-Link firmware, see Section 3.2.
- A custom version of J-Link Lite is also available but this firmware is limited to debug and VCOM features
- Other firmware options available for MCU-Link OB may not provide the buffer enable / direction control support that is required to correctly configure the K32W148-EVK hardware. Therefore, care should be taken while using such firmware.

3.1.2 Using MCU-Link with development tools

The LPC55S69 MCU is fully supported by the MCUXpresso suite of tools, which provides device drivers, middleware, and examples to allow rapid development. MCUXpresso also provides configuration tools and an optional free IDE. The MCUXpresso software is compatible with the open source MCU operating system FreeRTOS and tools from vendors, such as IAR.

The LPC55S69 MCU can also be used with some debug probes from SEGGER.

3.2 Installing device drivers and updating MCU-Link firmware

MCU-Link is supported on host computers running on Windows 10, MacOS X, and Ubuntu Linux operating systems (OSs). For each of these OSs, an MCU-Link firmware package is available that includes the host device drivers, MCU-Link firmware, and scripts to program CMSIS-DAP and J-Link firmware. The host device drivers are included in the firmware package to configure the host so that it displays user-friendly device names.

To download and install the host device drivers and update the MCU-Link firmware, follow these steps:

1. Go to the MCU-Link Debug Probe page on the NXP website.
2. Click **Design Resources**, and then click **SOFTWARE**. Installation packages for all three OSs are displayed.
3. Download the firmware package applicable to your host OS.
4. Depending on the host OS, install the firmware package (Linux/MacOS) or execute the installer program (Windows). The firmware package is installed/unzipped to the MCU-LINK_installer_Vx_xxx directory (where Vx_xxx indicates the version number, for example, V2_220). The directory has a "Readme.txt" file and few subdirectories that contain other package components, including host device drivers, MCU-Link firmware, and scripts to program CMSIS-DAP and J-Link firmware.
   Host drivers are usually installed automatically during firmware package installation. In case you need to install the drivers manually, refer to the "Host Configuration" section of the "Readme.txt" file.
   Before powering up the board for updating MCU-Link firmware, you must switch MCU-Link to ISP mode (USB1).
5. To force MCU-Link to boot in ISP mode (USB1), short jumper JP20.
6. Connect the J14 connector on the board to the USB port of the host computer through a USB micro-B cable. MCU-Link gets powered up in ISP mode (USB1). The red MCU-Link status LED (D13) lights up (for more details on MCU-Link LEDs, see Section 3.9). The board gets enumerated as a human interface device (HID) class device.
7. Program the MCU-Link firmware into the MCU-Link internal flash using the instructions provided in the "Firmware Installation Guide" section of the "Readme.txt" file. Use the scripts provided to program the CMSIS-DAP or J-Link firmware.
8. Disconnect the board from the host computer, open jumper JP20, and reconnect the board.
   Now, the system behaves differently based on whether the CMSIS-DAP or J-Link firmware was programmed:
   - If the CMSIS-DAP firmware was programmed, then the red status LED (D13) fades in/out repeatedly ("breathing"), while the green USB status LED (D11) stays ON constantly. In this case, four devices (three HID devices and one VCOM port) get enumerated, as shown in Figure 24.

Figure 24. Board enumerated devices

   - If the J-Link firmware was programmed, then only the green VCOM LED (D14) is used to indicate status. The VCOM LED lights up when MCU-Link boots. Then, when you start a new debug session, the LED blinks to indicate that debug activity is in progress. In this case, two devices (one J-Link driver USB device and one J-Link CDC UART port VCOM port) get enumerated.

Now, your K32W148-EVK board is ready for use. If you use the board with MCUXpresso IDE version 11.3 or higher, you are notified in case a more recent firmware version is available for MCU-Link. If you use the board with a different IDE, ensure that latest MCU-Link firmware version is installed on the board.

3.3 MCU-Link USB connector

The K32W148-EVK board has a universal serial bus (USB) 2.0 micro-B connector J14 (Hirose Electric ZX62D-B-5PA8(30)). This USB connector is used to create MCU-Link high-speed USB connection to the host computer. It can also be used to supply 5 V power to the board.

3.4 Supported MCU-Link features

MCU-Link includes several mandatory and optional features. Table 18 summarizes the MCU-Link features supported on the K32W148-EVK board.

Table 18. Supported MCU-Link features

Feature	Description
Serial wire debug (SWD) / serial wire debug trace output (SWO)	Allows SWD-based debugging with SWO for profiling and/or low overhead debug standard I/O communication
Virtual communication (VCOM) serial port	Adds a serial COM port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-UART bridge
USB serial input/output (USBSIO[1]	Adds a USB serial I/O port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-SPI bridge or USB-to-I2C bridge
External debug probe support	Allows debugging the target MCU (K32W148) using an external debug probe, instead of MCU-Link. Support for external debug probe is enabled by disabling SWD feature. While using external debug probe, VCOM and USBSIO features can be used.
External target support[1]	Allows debugging an external target MCU using MCU-Link
Energy/power/current/voltage consumption measurement[1] (MCUXpresso IDE only)	Allows onboard measurement of current drawn by the target MCU

[1] J-Link firmware does not support this feature.)

3.5 Supported debug scenarios

In the K32W148-EVK board, the MCU-Link debug probe target can be either the K32W148 MCU or an external target compliant with MCU-Link. The board also allows you to use an external debugger for debugging the K32W148 MCU, in place of the MCU-Link debug probe.

Table 19 describes the debug scenarios supported on the K32W148-EVK board.

Table 19. Supported debug scenarios

Debug scenario	Feature support
Use MCU-Link as debugger for the target MCU (K32W148)	• Serial wire debug (SWD): – MCU-Link SWD feature is enabled (MCU-Link SWD disable jumper JP22 is open) – The target MCU SWD interface is connected to MCU-Link – The target MCU SWD connector J16 is not used for external connection • Virtual communication (VCOM): MCU-Link VCOM feature is enabled • USB serial input/output (USBSIO): MCU-Link USBSIO feature is enabled • Target selection: Target selection jumper JP26 is open • Target power selection: Target power selection jumper JP28 is set to 1-2 shorted
Use an external debugger to debug the target MCU (K32W148)	• SWD: – Disable MCU-Link SWD feature by shorting MCU-Link SWD disable jumper JP22

Table 19. Supported debug scenarios...*continued*

Debug scenario	Feature support
	– Connect the external debugger to the target MCU SWD connector J16. The target MCU SWD interface is connected to the external debugger. • VCOM: MCU-Link VCOM feature can be used • USBSIO: MCU-Link USBSIO feature can be used • Target selection: Target selection jumper JP26 is open • Target power selection: Target power selection jumper JP28 is set to 1-2 shorted
Use MCU-Link as debugger for an external target MCU	• SWD: – MCU-Link SWD feature is enabled (MCU-Link SWD disable jumper JP22 is open) – Connect the external target MCU to the target MCU SWD connector J16 • VCOM: MCU-Link VCOM feature is not supported • USBSIO: MCU-Link USBSIO feature is not supported • Target selection: Short target selection jumper JP26 • Target power selection: – If the external target MCU has to get power from the board, then set target power selection jumper JP28 to 1-2 shorted – If the external target MCU has to use its own power, then set jumper JP28 to 2-3 shorted

3.6 Connecting to a target through a USB-to-UART bridge

MCU-Link supports the VCOM serial port feature, which adds a serial COM port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-UART bridge.

In the K32W148-EVK board, MCU-Link is connected to the LPUART1 port of the target MCU through a voltage translator U23. The voltage translator enables communication between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to VCC_TGMCU and vice versa.

To use MCU-Link as a USB-to-UART bridge, verify the following jumper settings and connect the J14 connector on the board to the USB port of the host computer:

- Jumper JP20 is open (MCU-Link is in Normal mode)
- Jumper JP21 is open (USB-to-UART bridge is enabled)

When you boot the K32W148-EVK board, a VCOM port with the name MCU-Link Vcom Port (COMxx) is enumerated on the host computer, where “xx” may vary from one computer to another. Each MCU-Link based board has a unique VCOM number associated with it.

The USB-to-UART bridge (VCOM feature) can be disabled by disabling voltage translator U23 (setting it to high-impedance). To disable U23, unpower the K32W148-EVK board and short jumper JP21. Shorting/opening JP21 after powering up the board has no impact on the functions/features of the MCU-Link firmware.

3.7 Connecting to a target through a USB-to-SPI or USB-to-I2C bridge

MCU-Link supports the USB serial input/output (USBSIO) port feature, which adds a USB serial I/O port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-SPI bridge or USB-to-I2C bridge. Support for the USBSIO feature can be enabled on the host computer using the libusbsio library, which is a free host library from NXP for Windows/Linux/MacOS systems. For more details on the libusbsio library, see https://www.nxp.com/libusbsio.

In the K32W148-EVK board, MCU-Link is connected to the LPSPI1 port of the target MCU using peripheral chip select 3 (PCS3) connection, through a voltage translator U24, which enables communication between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to VCC_TGMCU and vice versa.

MCU-Link is also connected to the LPI2C1 port of the target MCU through an I2C repeater / a voltage translator U27. The voltage translator enables communication between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to VCC_TGMCU and vice versa.

To use MCU-Link as a USB-to-SPI or USB-to-I2C bridge, the board must be connected to the host computer through a USB cable from its J14 connector. A USB-to-SPI bridge can be used to emulate the host system. A USB-to-I2C bridge can be used to emulate the host system / board peripherals.

The USBSIO feature can be disabled for SPI or I2C so that the target MCU SPI/I2C port can be used for other purposes. Disabling this feature instructs the firmware not to enumerate the USB endpoint for USBSIO (which is called “MCU-Link LPCSIO” for backward compatibility reasons). Disabling the USBSIO feature also frees more USB bandwidth for the SWO profiling and energy measurement features of MCU-Link.

The USBSIO feature can be disabled for SPI by disabling voltage translator U24 (setting it to high-impedance). By default, U24 is disabled (jumper JP34 is shorted); therefore, the USBSIO feature is disabled for SPI. The feature can be enabled by opening jumper JP34. Shorting/opening JP34 after powering up the board has no impact on the functions/features of the MCU-Link firmware. When USBSIO feature is disabled for SPI, then the target MCU SPI port can be used for other purposes.

The USBSIO feature can be disabled for I2C by opening jumper JP30.

3.8 Measuring target MCU power consumption

The K32W148-EVK board includes circuitry to measure voltage, current, and power/energy consumption for the target MCU (K32W148) by the onboard MCU-Link debug probe. This measurement data can be analyzed and displayed using MCUXpresso IDE version 11.5.1 or later. For more details on the MCUXpresso IDE, see https://www.nxp.com/mcuxpresso/ide.

It is recommended to use the latest MCU-Link firmware. If the latest firmware version is not installed, then the MCUXpresso IDE shows a message indicating that the firmware version is not the latest one. For instructions on updating MCU-Link firmware, see Section 3.2.

The power measurement feature is intended for low power measurements, with the target MCU running at up to 3.6 V. The design uses the built-in 16-bit ADC of the MCU-Link, with power data sampled at up to 100 kS/s. At high sample rates, MCUXpresso IDE fails to capture all data; therefore, you must adjust the sample rate using the configuration options available in the energy measurement configuration settings of the tool.

When MCU-Link is powered on, the measurement circuit initiates self-calibration. No user intervention is needed for setting up calibration or adjusting measurement ranges.

Table 20 shows current measurement configuration settings. The maximum measurable current is 50 mA. Accuracy may vary with temperature and it is only mentioned for reference purposes.

Table 20. Current measurement configuration settings

Measurement range	Resolution	Accuracy (typical)
200 nA to 400 μA	200 nA	1%
> 400 μA to 50 mA	5 μA	1%

In the K32W148-EVK board, the onboard MCU-Link is used to measure the current drawn by the target MCU. For current measurement, the supported voltage range is 1.7 V – 3.6 V. By default, current measurement is disabled on the board.

Table 21 shows the power supplies and jumpers corresponding to various currents of the target MCU (K32W148).

Table 21. Target MCU currents

Target MCU current	Power supply	Jumper that drives power supply
VDD_SWITCH current	VDD_SWITCH	JP4 (1-2 shorted by default)
VDD_DCDC current	VDD_DCDC	JP7 (1-2 shorted by default)
VDD_IO_ABC current	VDD_IO_ABC	JP8 (1-2 shorted by default)
VDD_ANA current	VDD_ANA	JP9 (1-2 shorted by default)
VDD_LDO_CORE current	VDD_LDO_CORE	JP12 (1-2 shorted by default)
VDD_RF current	VDD_RF	JP14 (1-2 shorted by default)

For measuring the current drawn from a board power supply, the power supply is connected in line with the current measurement circuit by changing jumper settings.

To measure VDD_SWITCH, VDD_DCDC, VDD_IO_ABC, or VDD_ANA current, follow these steps:

1. Open jumper JP35. By default, the jumper is shorted.
2. Open the jumper corresponding to the power supply.
3. Connect pin 3 of the power supply jumper to pin 1 of JP35.
4. Connect pin 2 of the power supply jumper to pin 2 of JP35.

To measure VDD_LDO_CORE current, follow these steps:

1. Open jumper JP35, if it is shorted.
2. Open jumper JP12.
3. Change jumper JP37 setting to 2-3 shorted.
4. Change jumper JP10 setting to 3-4 shorted.
5. Connect pin 2 of JP12 to pin 1 of JP35.
6. Connect pin 1 of JP12 to pin 2 of JP35.

To measure VDD_RF current, follow these steps:

1. Open jumper JP35, if it is shorted.
2. Open jumper JP14.
3. Change jumper JP37 setting to 2-3 shorted.
4. Change jumper JP10 setting to 3-4 shorted.
5. Connect pin 1 of JP14 to pin 1 of JP35.
6. Connect pin 2 of JP14 to pin 2 of JP35.

3.9 MCU-Link status LEDs

The K32W148-EVK board has seven status indicator LEDs for MCU-Link. Table 22 lists these LEDs and describes how each LED behaves in different MCU-Link modes. These LEDs are shown in Figure 2.

Table 22. MCU-Link LEDs

Part identifier	LED name / color	MCU-Link mode		
		Normal mode (with CMSIS-DAP firmware)	Normal mode (with J-Link firmware)	Firmware Update (ISP) mode
D11	USB / green	Lights up after successful USB enumeration at startup. Afterward, the LED stays ON.	Remains OFF	Remains OFF
D12	SWO / green	Indicates if serial wire debug trace output (SWO) data is being received from the target MCU	Remains OFF	Remains OFF
D13	Status / red	Indicates heartbeat (fades in/out repeatedly), with SWD activity overlaid. The LED blinks rapidly at startup, if an error occurs.	Remains OFF	Lights up when MCU-Link target (LPC55S69) boots in ISP mode
D14	VCOM / green	Indicates if VCOM port is transmitting/receiving data	Lights up when MCU-Link boots, and then blinks when debug activity happens	Remains OFF
D15	FUNC / green	Reserved for future use	Remains OFF	Remains OFF
D16	SIO / green	Indicates if USBSIO bridge traffic is present	Remains OFF	Remains OFF
D17	NRG / green	Indicates if energy measurement communication is happening between the target MCU and MCU-Link	Remains OFF	Remains OFF

4 Revision history

Table 23 summarizes the revisions to this document.

Table 23. Revision history

Revision	Date	Topic cross-reference	Change description
1	13 February 2023		Marked jumper JP11 as DNP
		Section 1	Added a note regarding board operating temperature at the end of the section
		Section 1.1	Added as a new section
		Section 1.3	Added a reference of AN13687
		Section 1.9	In Table 5: • Updated details of jumpers JP5 and JP25 • Added two new jumpers JP42 and JP43
		Section 1.10	• Updated details of push buttons SW2, SW3, and SW4 in Table 6 • Updated SW4 circuit in Figure 5
		Section 2.1	• Updated Figure 8 and Figure 9 • Added a new power supply, VIN, in Table 8 • Added a note related to VBAT supply in P3V3 row

Table 23. Revision history*...continued*

Revision	Date	Topic cross-reference	Change description
		Section 2.2	Added a reference of AN13687
		Section 2.5	Updated accelerometer I2C addresses in Table 13
		Section 2.7	Updated the section
		Section 2.8	Replaced the "mikroBUS socket connector pinouts" table with Figure 21
		Section 2.9	Updated Table 16 and added Figure 22
		Section 2.10	Updated the section
0	20 August 2022		Initial NDA release

5 Legal information

5.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

5.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

5.3 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Contents

Date of release: 13 February 2023
Document identifier: K32W148-EVKUM