

nRF7002 DK Hardware

v0.7.2

User Guide

Contents

Revision history	iv
1 Introduction	5
2 Kit content	7
3 Interface MCU	8
3.1 Interface Boot/Reset button	8
3.2 Virtual COM ports	8
3.2.1 Dynamic hardware flow control	9
3.3 Mass storage device	9
4 Hardware description	11
4.1 Hardware drawings	11
4.2 Block diagram	11
4.3 nRF7002 companion IC	12
4.4 Power supply	13
4.4.1 Power sources	14
4.4.2 VDD power sources	14
4.4.3 Interface MCU power	15
4.5 Operating modes	15
4.5.1 USB detect	16
4.5.2 IMCU disable mode	16
4.5.3 Signal switches	16
4.6 External memory	18
4.7 Connector interface	19
4.7.1 Mapping of analog pins	21
4.8 Buttons and LEDs	22
4.9 32.768 kHz crystal	22
4.10 Debug input and trace	23
4.10.1 Tracing instructions	24
4.11 Debug out for programming external boards	24
4.11.1 Programming an external board	25
4.11.2 Programming a board with custom connections	26
4.12 Antennas	29
4.13 NFC antenna interface	29
4.14 Extra op-amp	30
4.15 Solder bridge configuration	31
5 Measuring current	34
5.1 Preparing the nRF7002 DK	34
5.2 Using an oscilloscope for current profile measurement	35
5.3 Using an ampere meter for current measurement	36
6 RF measurements	38
Glossary	40
Recommended reading	43

FCC regulatory notice. 44

Legal notices. 45

Revision history

Date	Description
2023-01-31	First release

1 Introduction

The nRF7002 DK is a hardware development platform to facilitate the design and development of Wi-Fi® 6 applications.

The *Development Kit (DK)* combines the Wi-Fi 6 capabilities of the nRF7002 companion *Integrated Circuit (IC)* with the nRF5340 *System on Chip (SoC)*.

Key features

- nRF7002 Wi-Fi companion IC
 - Dual-band 2.4 GHz and 5 GHz Wi-Fi 6
 - Compatible with IEEE 802.11ax (also known as Wi-Fi 6) and earlier standards IEEE 802.11a/b/g/n/ac.
 - 20 MHz wide channels, 1x1 (SISO) operation and up to 86 MHz 802.11 PHY rate, QSPI for fast host interface support
 - Open-source Wi-Fi driver - L2 Network Technologies layer-compatible
 - 3 or 4-wire COEX interface configurations
 - Secure, 64-word *One Time Programmable (OTP) memory* with logical and voltage-level based protection mechanisms
- nRF5340 SoC, with support for the following additional wireless protocols:
 - *Bluetooth®* Low Energy
 - *Near Field Communication (NFC)*
 - 802.15.4
 - Thread®
 - Zigbee®
 - ANT™
 - 2.4 GHz proprietary
- Onboard 2.4 GHz and 2.4/5 GHz antennas
- NFC antenna
- An optional 32.768 kHz crystal (X2) for higher accuracy and lower average power
- *Microwave coaxial connector with switch (SWF)* RF connector for direct RF measurements
- User-programmable LEDs (2) and buttons (2)
- SEGGER J-Link on board programmer/debugger
- Pins for measuring power consumption
- 1.8 V power supply from *Universal Serial Bus (USB)* or external *Lithium-polymer (Li-Poly)* battery
- 3.6 V power supply from USB or external Li-Poly battery to the **VBAT** of the nRF7002 companion IC

For access to firmware source code, hardware schematics, and layout files, see www.nordicsemi.com.

Note: The nRF7002 DK must be powered by a PS1 class (IEC 62368-1) power supply with maximum power less than 15 W.

**Environmental Protection**

Waste electrical products should not be disposed of with household waste.

Please recycle where facilities exist. Check with your local authority or retailer for recycling advice.

The nRF7002 DK is intended for use only by skilled persons.

A skilled person is someone with relevant education or experience that enables them to identify potential hazards and takes appropriate action to reduce the risk of injury to themselves and others.



2 Kit content

The nRF7002 DK includes hardware, preprogrammed firmware, documentation, hardware schematics, and layout files.

The nRF7002 DK (PCA10143) comes with an NFC antenna.

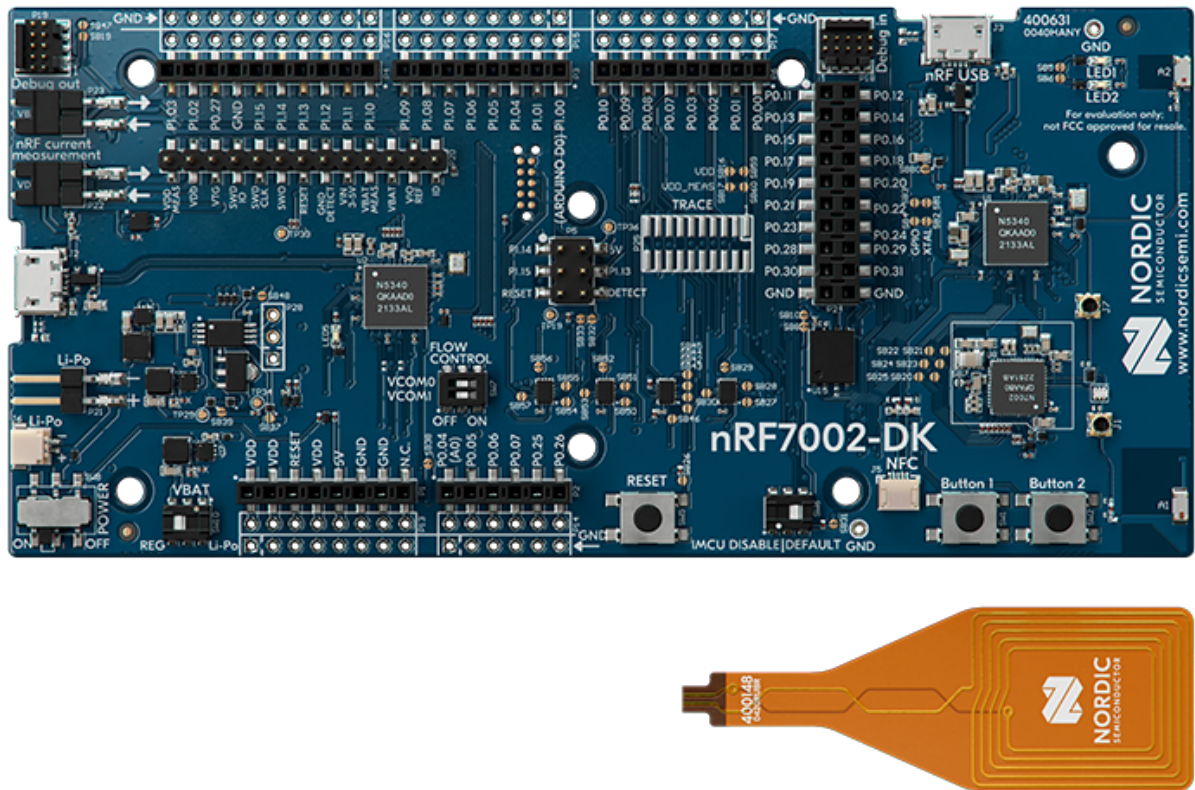


Figure 1: nRF7002 DK (PCA10143) front view with NFC antenna

Hardware files

The hardware design files including schematics, *Printed Circuit Board (PCB)* layout files, bill of materials, and Gerber files for the nRF7002 DK are available on the [nRF7002 product page](#).

The nRF7002 DK is susceptible to *Electrostatic Discharge (ESD)*.

To avoid damage to your DK, it should be used in an electrostatic free environment, such as a laboratory.



3 Interface MCU

The interface MCU on the nRF7002 DK runs SEGGER J-Link onboard interface firmware and is used to program and debug the firmware of the nRF5340 SoC.

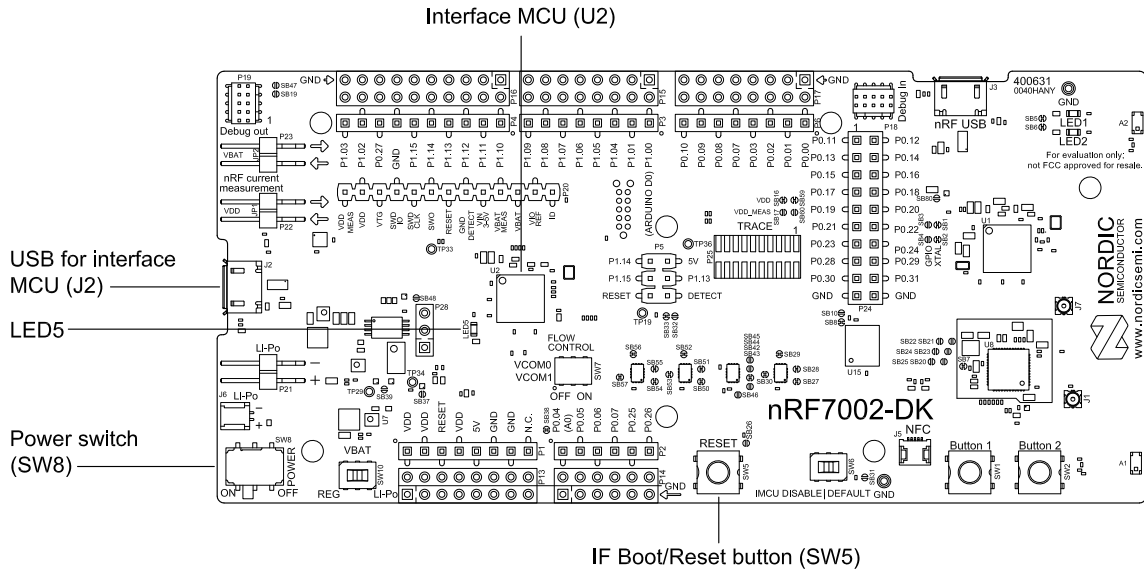


Figure 2: Interface MCU

3.1 Interface Boot/Reset button

The nRF7002 DK has a reset button (**SW5**).

This button is connected to the interface MCU on the DK and is used to reset the nRF5340 SoC. If the IMCU is disabled, the reset button is connected directly to the nRF5340 SoC.

3.2 Virtual COM ports

The onboard interface MCU features *Universal Asynchronous Receiver/Transmitter (UART)* interface through virtual COM ports.

The virtual COM ports have the following features:

- Flexible baud rate setting up to 1 Mbps (baud rate 921,600 bps is not supported)
- Dynamic *Hardware Flow Control (HWFC)* handling
- Tri-stated UART lines when no terminal is connected

The following table shows an overview of the UART connections on nRF5340 SoC and the interface MCU.

GPIO nRF5340	nRF5340 UART_2	nRF5340 UART_1
P0.19	RTS	
P0.20	TXD	
P0.21	CTS	
P0.22	RXD	
P1.05/TRACEDATA0		RTS
P1.01		TXD
P1.04/TRACEDATA1		CTS
P1.00		RXD

Table 1: Relationships of UART connections on nRF5340 and interface MCU

The UART signals are routed directly to the interface MCU. The UART pins connected to the interface MCU are tri-stated when no terminal is connected to the virtual COM port on the computer.

Note: The terminal software used must send a *Data Terminal Ready (DTR)* signal to configure the UART interface MCU pins.

The **P0.19** (*Request to Send (RTS)*) and **P0.21** (*Clear to Send (CTS)*) can be used freely when HWFC is disabled on the SoC.

3.2.1 Dynamic hardware flow control

When the interface MCU receives a *DTR* signal from a terminal, it performs automatic *HWFC* detection.

Automatic *HWFC* detection is done by driving **P0.21** (*CTS*) from the interface MCU and evaluating the state of **P0.19** (*RTS*) when the first data is sent or received. If the state of **P0.19** (*RTS*) is high, *HWFC* is assumed not to be used. If *HWFC* is not detected, both *CTS* and *RTS* can be used freely by the nRF application.

After a power-on reset of the interface MCU, all *UART* lines are tri-stated when no terminal is connected to the virtual COM port. Due to the dynamic *HWFC* handling, if *HWFC* has been used and detected, **P0.21** (*CTS*) is driven by the interface MCU until a power-on reset has been performed or until a new *DTR* signal is received and the detection is redone.

To ensure that the *UART* lines are not affected by the interface MCU, the solder bridges for these signals can be cut and later resoldered if needed. This might be necessary if *UART* without *HWFC* is needed while **P0.19** (*RTS*) and **P0.21** (*CTS*) are used for other purposes.

If you need *RTS* and *CTS* for another task, use **SW7** to disable *RTS* and *CTS* on both *UARTs*.

3.3 Mass storage device

The interface MCU features an *Mass Storage Device (MSD)*. This makes the *DK* appear as an external drive on your computer.

This drive can be used for drag-and-drop programming. By copying a *HEX* file to the drive, the interface MCU programs the file to the device. However, files cannot be stored on this drive.

Note:

- If Windows tries to defragment the MSD part of the interface MCU, the interface MCU disconnects and becomes unresponsive.
- Your antivirus software might try to scan the MSD part of the interface MCU. Some antivirus programs trigger a false positive alert in one of the files and quarantine the unit. If this happens, the interface MCU becomes unresponsive.
- If the computer is set up to boot from USB, it can try to boot from the DK if the DK is connected during boot. This can be avoided by unplugging the DK before a computer restart or changing the boot sequence of the computer.

You can also disable the MSD of the DK by using the **msddisable** command in J-Link Commander. To enable, use the **msdenable** command. These commands take effect after a power cycle of the DK and stay this way until changed again.

4 Hardware description

The nRF7002 DK (PCA10143) features an onboard programming and debugging solution.

In addition to radio communication, the nRF5340 SoC can communicate with a computer through *USB* and a virtual COM port provided by the interface MCU.

4.1 Hardware drawings

nRF7002 DK hardware drawings show both sides of the PCA10143.

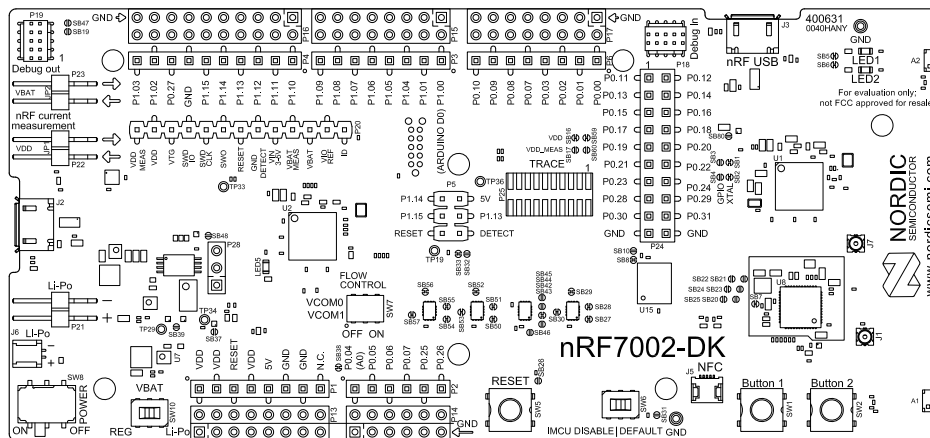


Figure 3: nRF7002 DK (PCA10143) front view

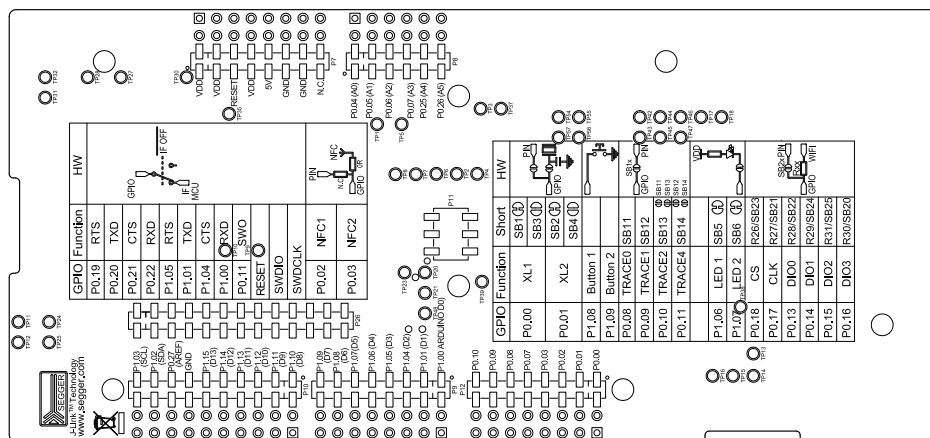


Figure 4: nRF7002 DK (PCA10143) back view

4.2 Block diagram

The block diagram illustrates the nRF7002 DK functional architecture.

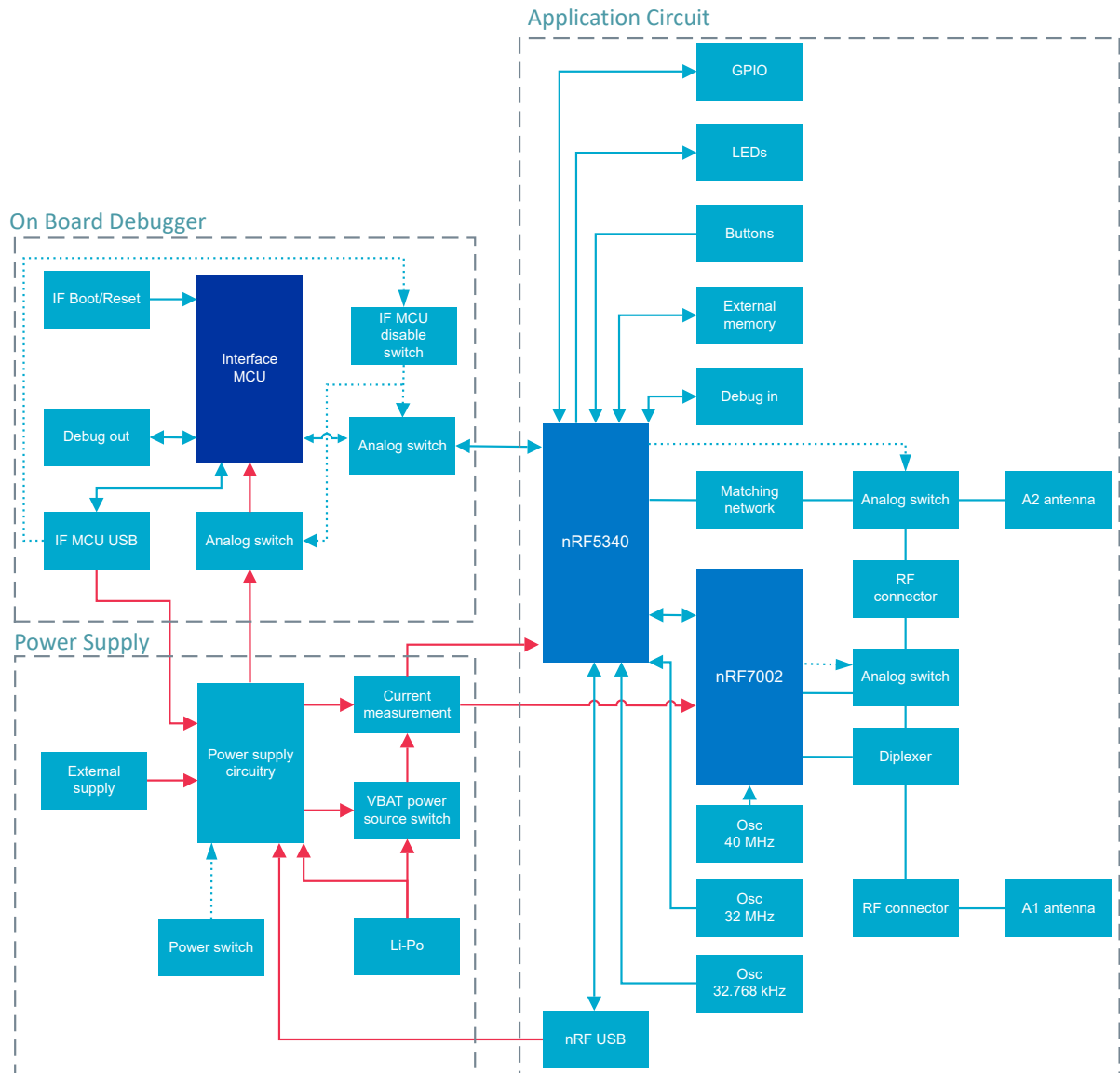


Figure 5: Block diagram

4.3 nRF7002 companion IC

The nRF7002 companion IC is connected to the nRF5340 SoC using *Quad Serial Peripheral Interface (QSPI)* and a set of control signals.

The control signals are also available on header **P24**, while the QSPI by default is not connected. These can be connected to **P24** by shorting the solder bridges **SB20—SB25**.

nRF5340	nRF7002	Default use
P0.12	BUCKEN	Enable power to nRF7002
P0.13	QSPI_DATA0	Data line 0
P0.14	QSPI_DATA1	Data line 1
P0.15	QSPI_DATA2	Data line 3
P0.16	QSPI_DATA3	Data line 3
P0.17	QSPI_CLK	Clock
P0.18	QSPI_SS	Slave select
P0.23	HOST_IRQ	Interrupt request to host
P0.24	COEX_GRANT	Coexistence grant to host
P0.28	COEX_REQ	Coexistence request from host
P0.29	SW_CTRL1	Switch control 1
P0.30	COEX_STATUS0	Coexistence status
P0.31	IOVDD	Enable power to I/O interface

Table 2: nRF7002 interface

4.4 Power supply

The nRF7002 DK has multiple power options.

The power options are:

- USB connector **J2** for the interface MCU (5 V)
- USB connector **J3** for the nRF5340 SoC (5 V) for the application MCU
- *Li-Poly* battery connectors **J6** or **P21** (2.9 V to 4.5 V)
- **VIN 3-5V** pin on **P20** (3.0 V to 5.0 V)

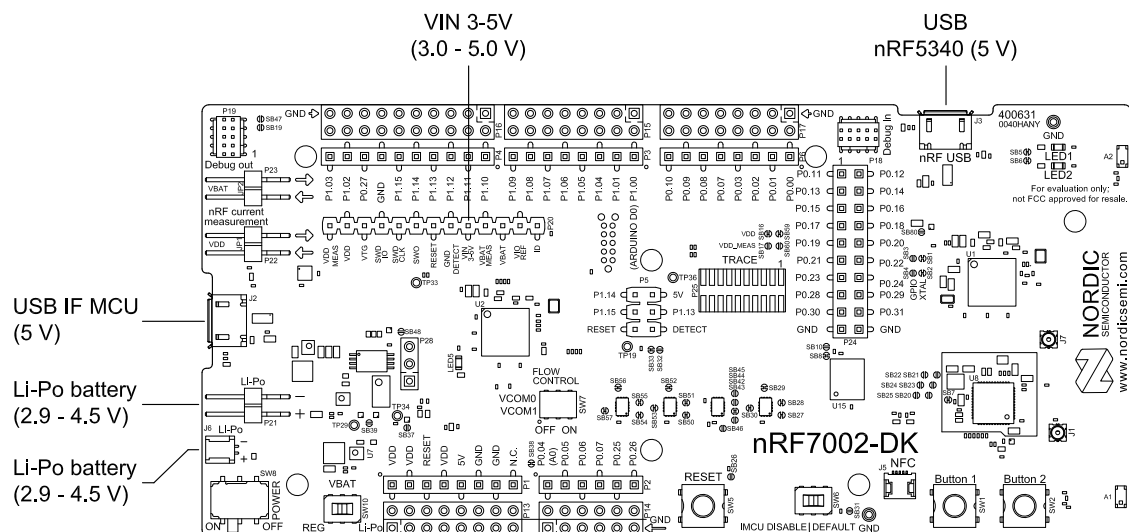


Figure 6: Power supply options (front)

4.4.1 Power sources

The nRF7002 DK has a 5 V boost regulator.

It gives a stable 5 V output from the following four possible sources:

- USB connector **J2** for the interface MCU
- USB connector **J3** for the nRF5340 application MCU
- *Li-Poly* battery connectors (**J6** or **P21**)
- **VIN 3–5V** pin on **P20**

Each source has a reverse protection diode to prevent current flowing in the wrong direction if multiple sources are connected at the same time.

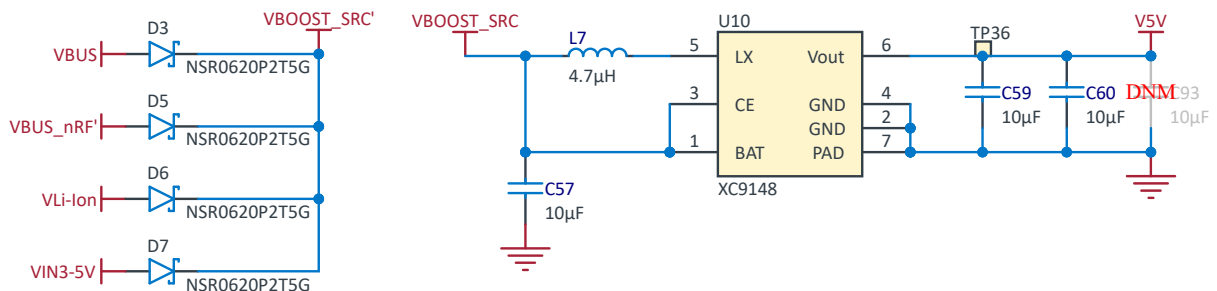


Figure 7: 5 V regulator and protecting diodes

CAUTION: Reverse protection is not applied if the **SW10** switch is moved to Li-Poly. In this case, the power is routed directly to the VBAT of the nRF7002 DK. Care must be taken not to apply an external voltage in reverse, as this might damage the DK circuitry.

4.4.2 VDD power sources

The main supply (VDD) can be sourced from the 5 V domain.

For the 5 V domain, there is one fixed 1.8 V buck regulator and one voltage follower regulator that follows the VIO_REF voltage

The power sources are routed through a set of load switches, which is controlled by logic to prioritize the power sources in the correct manner.

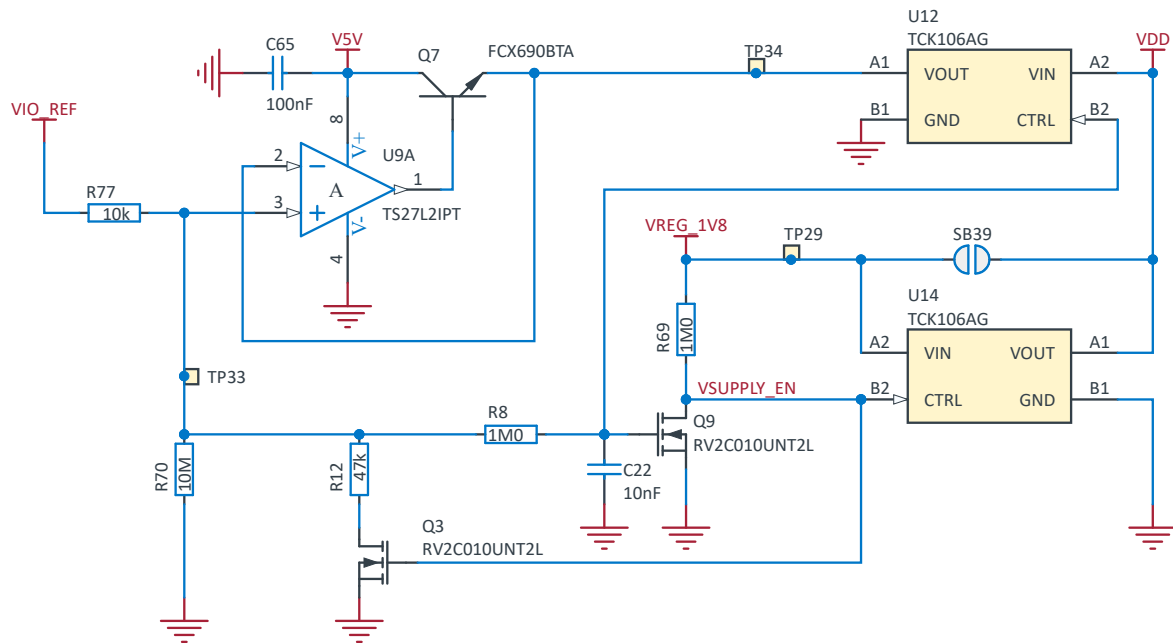


Figure 8: Power supply circuitry

4.4.3 Interface MCU power

The power for the interface MCU is routed through two load switches, one for the VDD supply and one for the USB supply. This makes it possible to disconnect the interface MCU from the power domain when it is not in use.

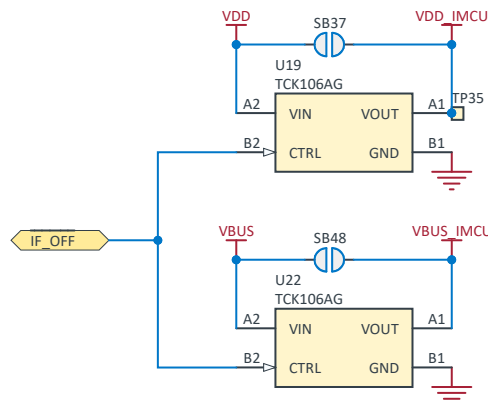


Figure 9: Interface MCU power switch

These switches are controlled by the presence of a USB connected to the interface MCU USB connector (**J2**), and the state of the IMCU disable switch (**SW6**). See [Operating modes](#) on page 15 for more information.

4.5 Operating modes

USB detect and IMCU disable, the two modes of operation of the nRF7002 DK and signal switches are described in the following sections.

4.5.1 USB detect

To detect when *USB* for the interface MCU is connected, there is a circuit sensing the VBUS of USB connector **J2**.

When the USB cable is connected, the VDD is propagated to the USB_DETECT signal.

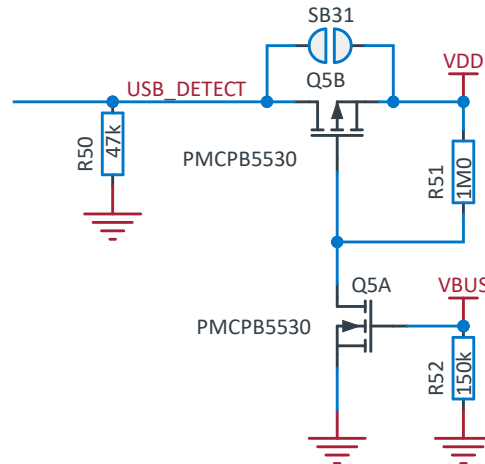


Figure 10: USB detect

4.5.2 IMCU disable mode

The IMCU disable switch disconnects the power supply and LEDs of the interface MCU. It also disconnects the signal lines between the nRF5340 SoC and the interface MCU, using analog switches.

This is done to isolate the SoC on the *DK* as much as possible and is useful when measuring currents on low-power applications.

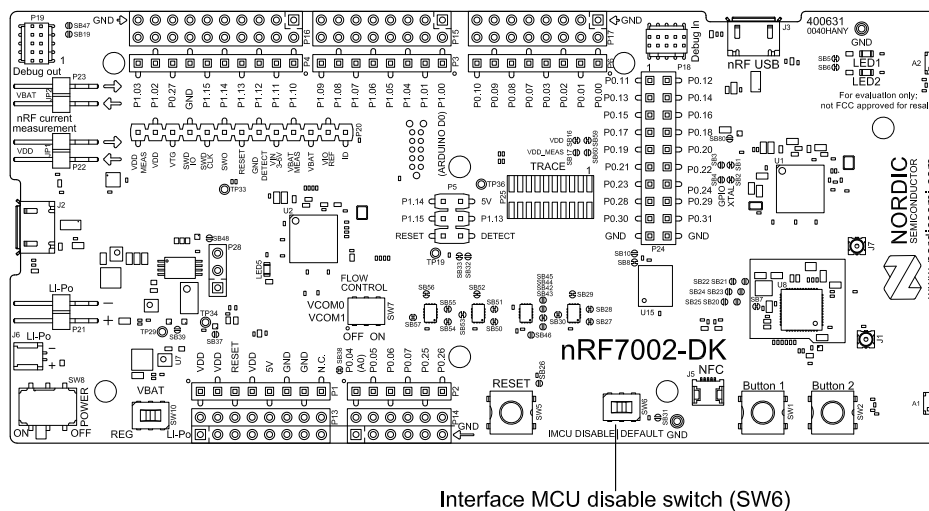


Figure 11: IMCU disable switch (SW6)

4.5.3 Signal switches

On the nRF7002 DK, multiple analog switches are used to connect and disconnect signals based on different scenarios.

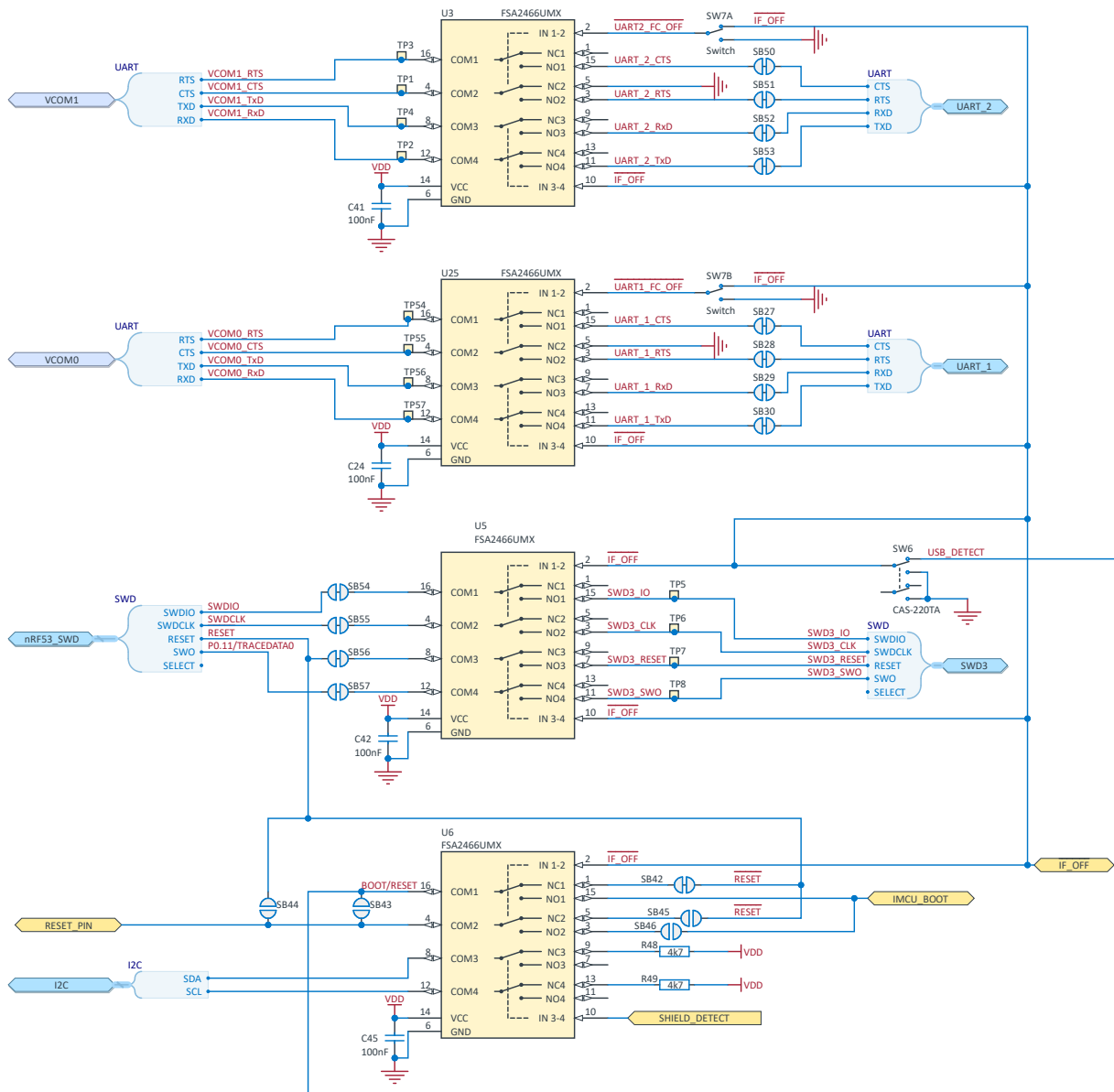


Figure 12: Signal switches

The USB and **SW6** control the signal switches by using **USB_DETECT** as an input to **SW6**. Therefore, the interface MCU can be disconnected either by unplugging the USB cable in **J2** or by toggling **SW6**.

The signal controls a set of switches (**U3**, **U5**, **U6**, **U25**) that break the connection between the nRF5340 SoC and the interface MCU and control the power for the interface MCU. See [Interface MCU power](#) on page 15 for more information.

Switches **U3**, **U5**, and **U25** break the connection of the UART lines and SWD/RESET lines. In addition, the signal controls the routing of the RESET signal depending on user preference when the interface MCU is connected or disconnected.

- When the interface MCU is connected, shorting **SB46** connects the RESET pin in the Arduino interface to the BOOT input of the interface MCU.
- Shorting **SB43** connects the RESET pin in the Arduino interface to the IF Boot/Reset button.

When a shield is connected, there are two analog switches connecting the pull-up resistors to the *Inter-integrated Circuit* (I^2C) bus lines (SDA and SCL). This function uses a ground pin on the Arduino shield to control the switch. This feature can be disabled by cutting **SB33**. To permanently enable pull-up resistors, short **SB32**.

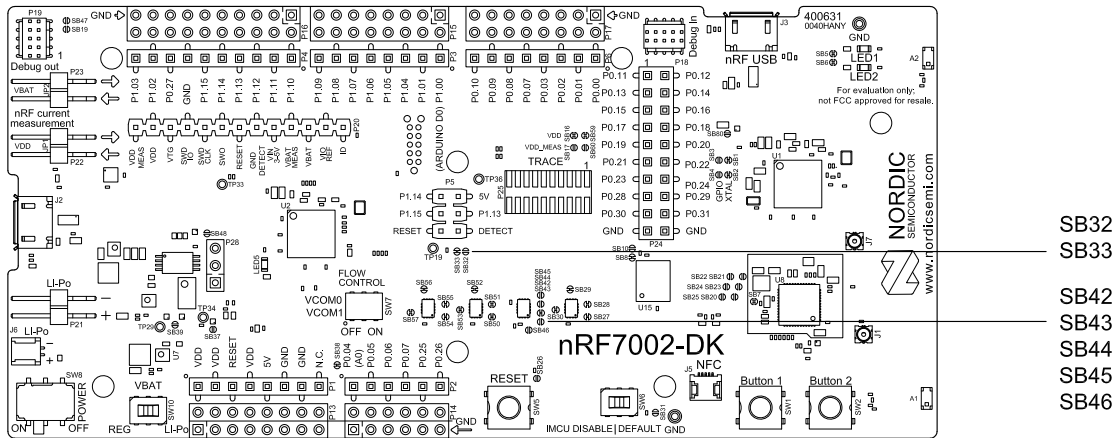


Figure 13: Solder bridges: Shield detect and reset behavior

4.6 External memory

The nRF7002 DK has a 64 Mb external flash memory. The memory is connected through a regular high-speed *Serial Peripheral Interface (SPI)*.

The memory is connected to the chip using the following *General-Purpose Input/Output (GPIO)*s:

nRF5340 GPIO	Flash memory pin
P0.11	CS
P0.08	SCLK
P0.09	MOSI
P0.10	MISO

Table 3: Flash memory GPIO usage and connecting solder bridges

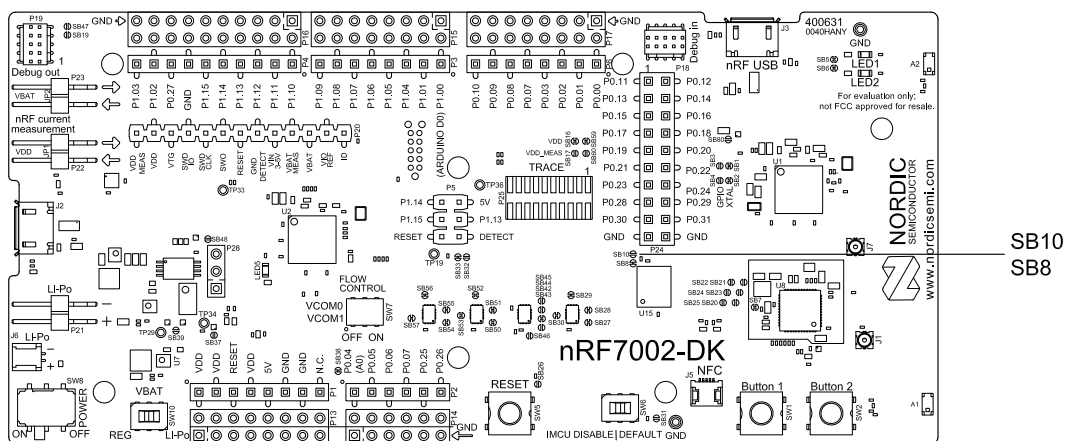


Figure 14: Configuring GPIOs for external memory

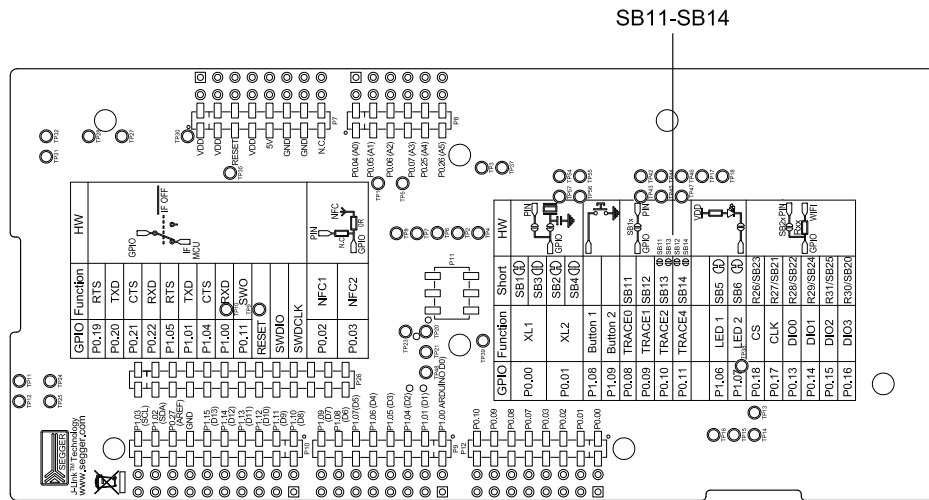


Figure 15: Configuring GPIOs for external memory

By default, the power supply of the external memory comes from the VDD domain. There are two optional power sources for keeping the external memory powered: VDD and VDD_MEAS. If VDD_MEAS is selected, the power consumption of the external memory is added to the nRF7002 current measured on **P22**. See the following table for configuration:

Power source	Solder bridge	Default state
VDD	SB8	Shorted
VDD_MEAS	SB10	Open

Table 4: Flash memory power source configuration

4.7 Connector interface

Access to the nRF5340 GPIOs is available from connectors **P2**, **P3**, **P4**, **P5**, **P6**, and **P24**.

The **P1** connector provides access to ground and power on the nRF7002 DK.

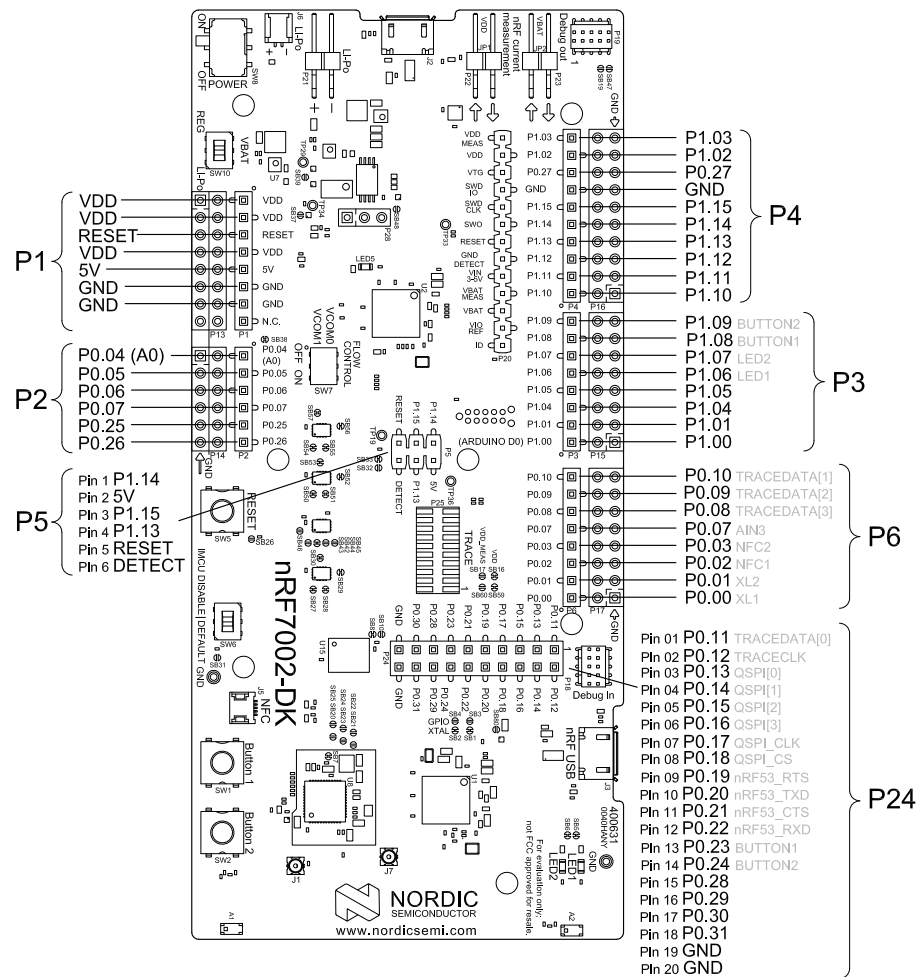


Figure 16: nRF7002 DK connectors

Some of the signals are also available on connectors **P7**, **P8**, **P9**, **P10**, **P11**, and **P12**, which are on the back of the DK. By mounting pin lists on the connector footprints, the nRF7002 DK can be used as a shield for 3.3 V Arduino motherboards or other boards that follow the Arduino standard.

For easy access to GPIO, power, and ground, the signals can also be found on the through-hole connectors **P13–P17**.

Some pins have default settings:

- **P0.00** and **P0.01** are used for the 32.768 kHz crystal and are not available on the connectors. See [32.768 kHz crystal](#) on page 22 for more information.
- **P0.19**, **P0.20**, **P0.21**, and **P0.22** are used by the UART connected to the interface MCU. See [Virtual COM ports](#) on page 8 for more information.
- **P0.02** and **P0.03** are by default used by NFC1 and NFC2. See [NFC antenna interface](#) on page 29 for more information.
- **P1.08–P1.09** are by default connected to the buttons and **P1.06 - P1.07** are connected to the LEDs. See [Buttons and LEDs](#) on page 22 for more information.
- **P0.13–P0.18** are by default connected to nRF7002. See [Solder bridge configuration](#) on page 31 for more information.

When the nRF7002 DK is used as a shield together with an Arduino standard motherboard, the Arduino signals are routed as shown in the following figure.

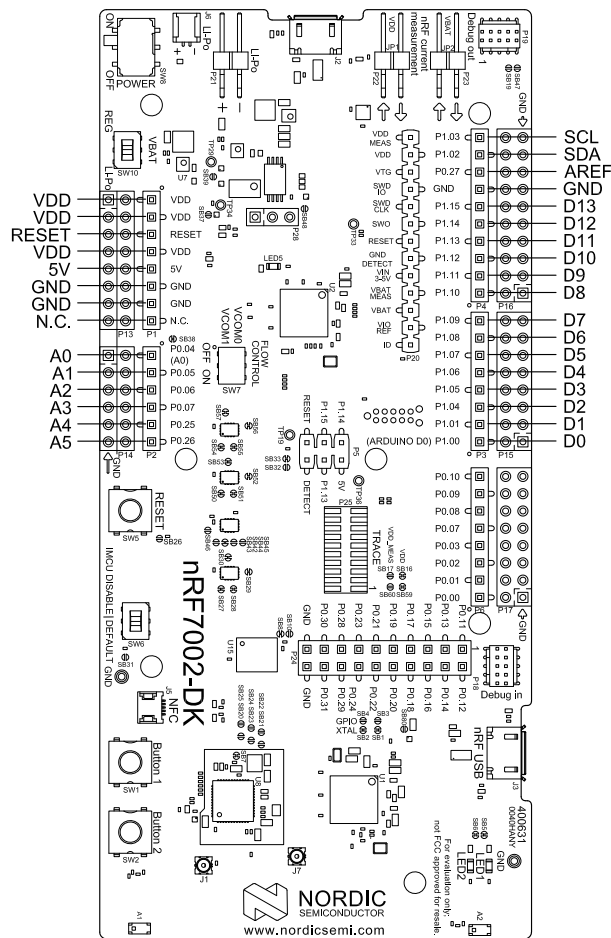


Figure 17: Arduino signals routing on the nRF7002 DK

Note: The nRF7002 DK runs at 1.8 V and needs modification to work with standard Arduino boards.

4.7.1 Mapping of analog pins

The following table shows the mapping between GPIO pins, analog inputs, and the corresponding Arduino analog input naming.

GPIO	Analog input	Arduino naming
P0.04	AIN0	A0
P0.05	AIN1	A1
P0.06	AIN2	A2
P0.07	AIN3	A3
P0.25	AIN4	A4
P0.26	AIN5	A5

Table 5: Mapping of analog pins

4.8 Buttons and LEDs

The two buttons and two LEDs on the nRF7002 DK are connected to dedicated GPIOs on the nRF5340 SoC.

Part	GPIO	Solder bridge
Button 1	P1.08	-
Button 2	P1.09	-
LED 1	P1.06	SB5
LED 2	P1.07	SB6

Table 6: Button and LED connection

The buttons are active low, meaning that the input is connected to ground when the button is activated. The buttons have no external pull-up resistor, and therefore, to use the buttons, the **P1.08–P1.09** pins must be configured as an input with an internal pull-up resistor.

The LEDs are active high, meaning that writing a logical one (1) to the output pin causes the LED to light up.

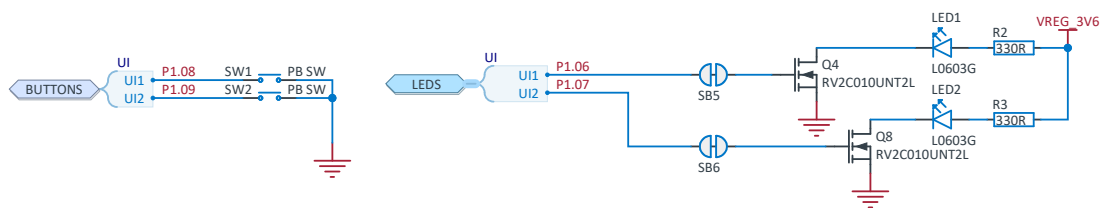


Figure 18: Button and LED configuration

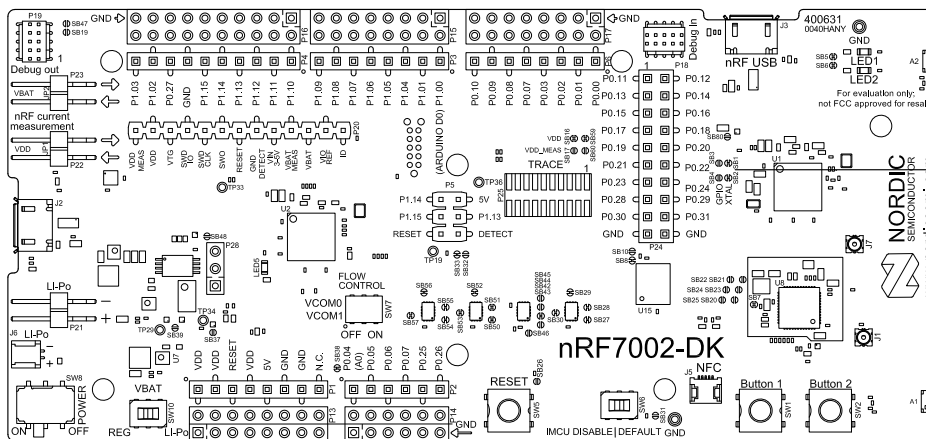
4.9 32.768 kHz crystal

The nRF5340 SoC can use an optional 32.768 kHz crystal (**X2**) for higher accuracy and lower average power consumption.

On the nRF7002 DK, **P0.00** and **P0.01** are used for the 32.768 kHz crystal by default and are not available as GPIO on the connectors.

Note: When using ANT/ANT+™, the 32.768 kHz crystal (**X2**) is required for correct operation.

If **P0.00** and **P0.01** are needed as normal I/Os, the 32.768 kHz crystal can be disconnected and the GPIO routed to the connectors. Cut the shorting track on **SB1** and **SB2**, and solder **SB3** and **SB4**. See the following figure for reference.



Cut SB1
Cut SB2
Solder SB3
Solder SB4

Figure 19: Configuring P0.00 and P0.01

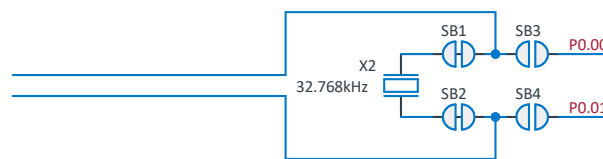


Figure 20: 32.768 kHz crystal and SB1–SB4

4.10 Debug input and trace

The Debug in connector (**P18**) makes it possible to connect external debuggers for debugging when the interface MCU USB cable is not connected or the *DK* is in MCU disable mode.

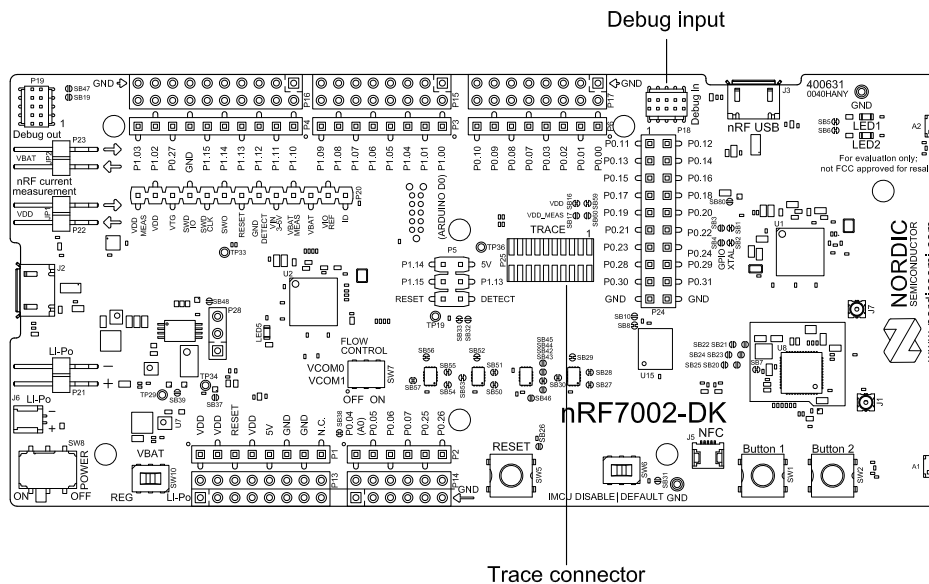


Figure 21: Debug input and trace connectors

For trace, a footprint for a 20-pin connector is available (**P25**). If trace functionality is required, it is possible to mount a 2x10 pin 1.27 mm pitch surface-mount pin header.

GPIO	Trace	Default use
P0.12	TRACECLK	nRF7002 Buck Enable (SB7 disconnects buck enable)
P0.11	TRACEDATA[0]	External memory, default disconnected
P0.10	TRACEDATA[1]	External memory, default disconnected
P0.09	TRACEDATA[2]	External memory, default disconnected
P0.08	TRACEDATA[3]	External memory, default disconnected

Table 7: Default and Trace GPIOs

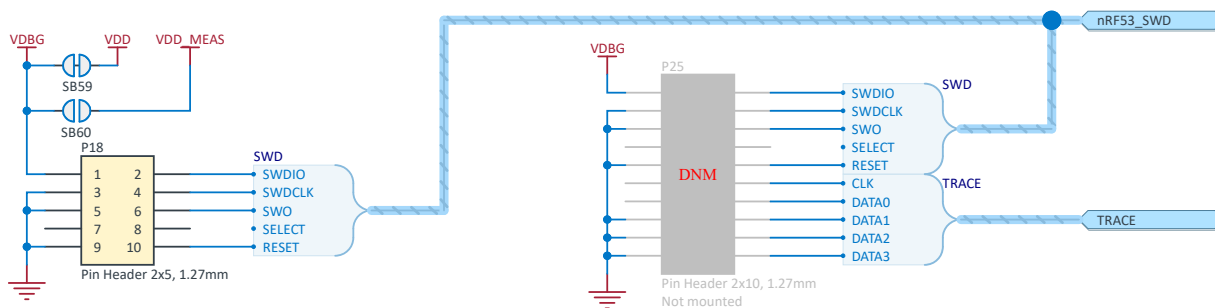


Figure 22: nRF7002 DK debug and trace headers

The reference voltage for the debug input and trace is by default connected to VDD. This can be connected to VDD_MEAS by cutting **SB59** and soldering **SB60**.

4.10.1 Tracing instructions

nRF7002 DK supports *Embedded Trace Macrocell (ETM)* and *Instrumentation Trace Macrocell (ITM)* trace.

Refer to [Figure 22: nRF7002 DK debug and trace headers](#) on page 24 for more information.

1. Solder the **P25** connector on the nRF7002 DK.
2. Cut **SB7** and connect a wire from the **SB7** pad on nRF7002 side to VDD_MEAS (always high) or any GPIO.
For example: **P0.05** to control BUCKEN from the firmware.
3. Modify the firmware to remove **P0.12** from BUCKEN and enable it as TRACECLK.
4. Short the **SB11**, **SB12**, **SB13**, **SB14** solder bridges.
These are the trace pins out of the nRF5340 SoC.
5. Cut **SB8**.
This disables the onboard flash memory which shares the trace data pins.

4.11 Debug out for programming external boards

The nRF7002 DK supports programming and debugging external boards with nRF51, nRF52, or nRF53 Series SoCs.

The interface MCU on the nRF7002 DK runs SEGGER J-Link onboard interface firmware and is used to program and debug the firmware of the nRF5340 SoC by default.

To program or debug an external board instead, connect to the Debug out connector (**P19**) using a 10-pin cable or use **P20** for custom connection.

Note: It is recommended to power the external board separately from the DK. The voltage on the external board must match that of the DK, which is 1.8V when the DK is powered through the USB connector.

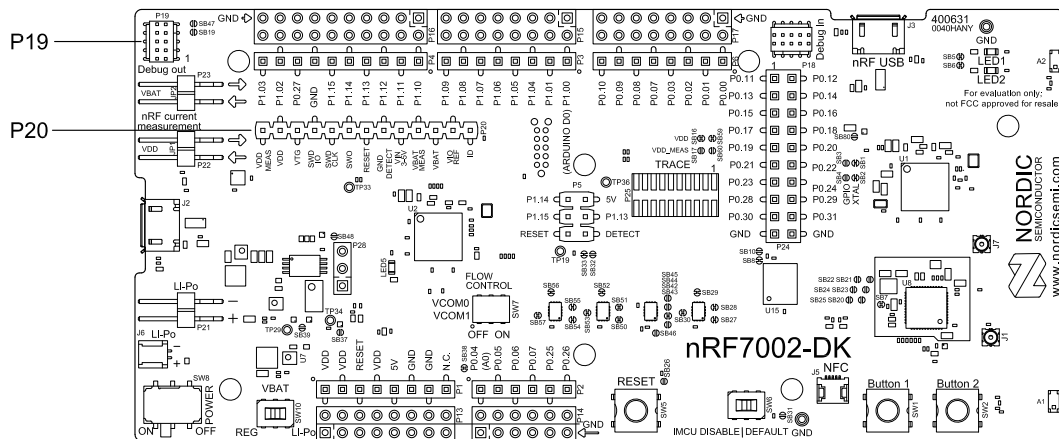


Figure 23: Debug output connectors

4.11.1 Programming an external board

If your custom board has a 10-pin Arm® Cortex® Debug Connector, connection to **P19** is recommended.

Connect the boards as shown in the following figure.

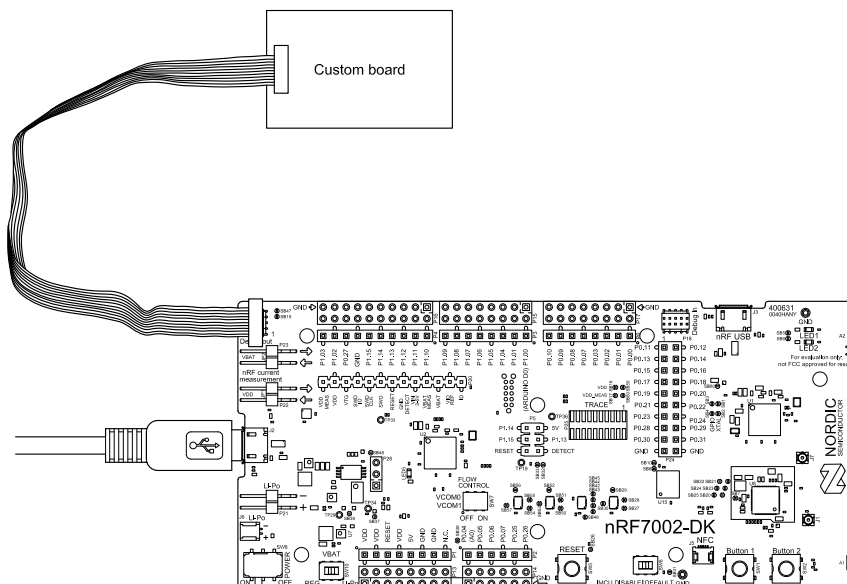


Figure 24: Connecting an external board to P19

It is recommended to power the external board separately from the DK. The voltage on the external board must match that of the DK, which is 1.8V when the DK is powered through the USB connector.

When the interface MCU detects that the ground pin 3 (**GND**) of **P19** is pulled low, it programs or debugs the target chip on the external board instead of the onboard nRF5340 SoC.

If it is inconvenient to have a separate power supply on the external board, the nRF7002 DK can supply power through the Debug out connector **P19**. To enable this, short solder bridge **SB47**.

CAUTION: To avoid damaging your board, when **SB47** is shorted, do not connect a separate power supply to the external board.

The following image shows **P19** pinout schematic with a description table.

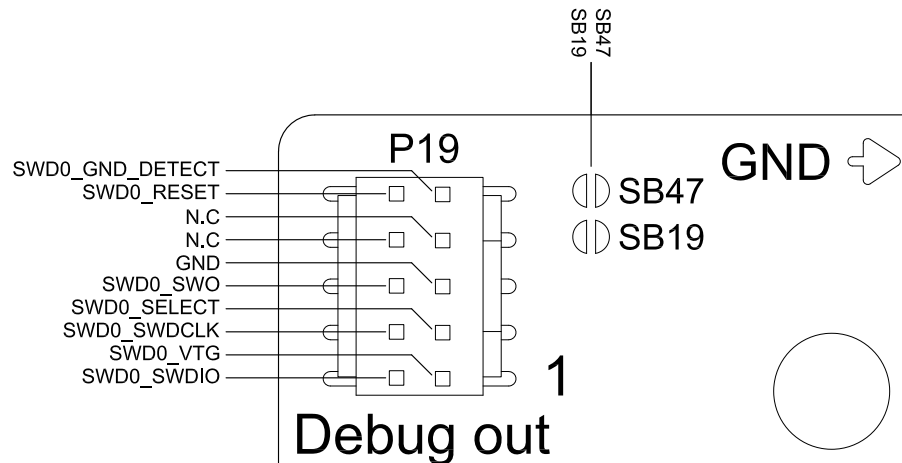


Figure 25: Debug output connector **P19**

Pin number	Signal	Description
1	SWD0_VTG	Voltage supply from the external target
2	SWD0_SWDIO	Serial Wire debug Data Input/Output
3	SWD0_SELECT	Ground
4	SWD0_SWDCLK	Serial Wire Clock line
5	GND	Ground
6	SWD0_SWO	The <i>Serial Wire Output (SWO)</i> line is not needed for programming and debugging over SWD
7	N.C.	Not used
8	N.C.	Not used
9	SWD0_GND_DETECT	Ground detect
10	SWD0_RESET	Reset line

Table 8: Pinout of connector **P19** for programming external targets

4.11.2 Programming a board with custom connections

If your external board has custom connections to programming and debugging pins, you can use the debug output on pin list **P20**.

Connect the boards as shown in the following figure.

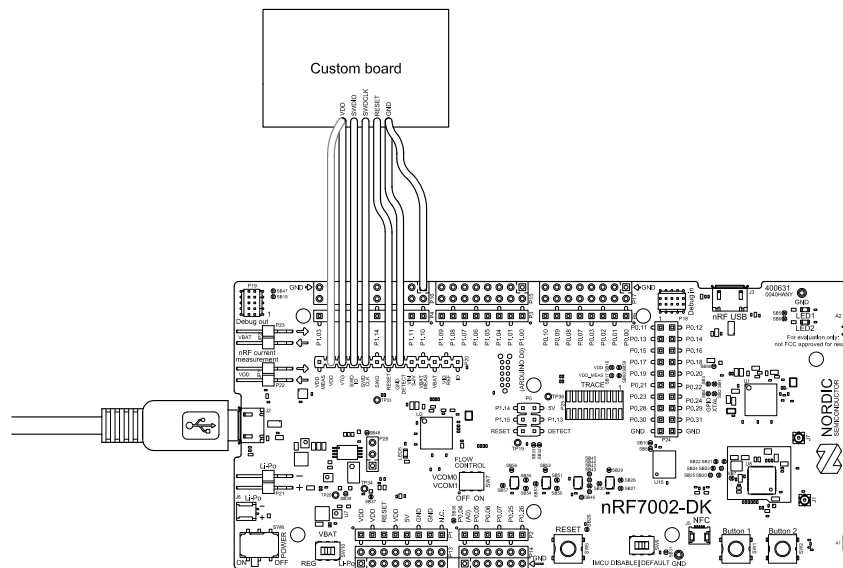


Figure 26: Connecting an external board to P20

It is recommended to power the external board separately from the DK. The voltage on the external board must match that of the DK, which is 1.8V when the DK is powered through the USB connector.

When the interface MCU detects the voltage of the external board on pin 3 (VTG) of **P20** it programs or debugs the target chip on the external board instead of the onboard nRF5340 SoC.

If it is inconvenient to have a separate power supply on the external board, the nRF7002 DK can supply power through pin 2 (**VDD**) of **P20**. The connection is shown with a grey outline in [Figure 26: Connecting an external board to P20](#) on page 27.

Note: If the interface MCU detects ground on **P19** and power on **P20**, it programs or debugs the target connected to **P19** by default.

CAUTION: To avoid damaging your board, when **VDD** of nRF7002 DK is connected to the external board, do not connect a separate power supply to the external board.

The following figure shows **P20** connector pinouts with a description table.

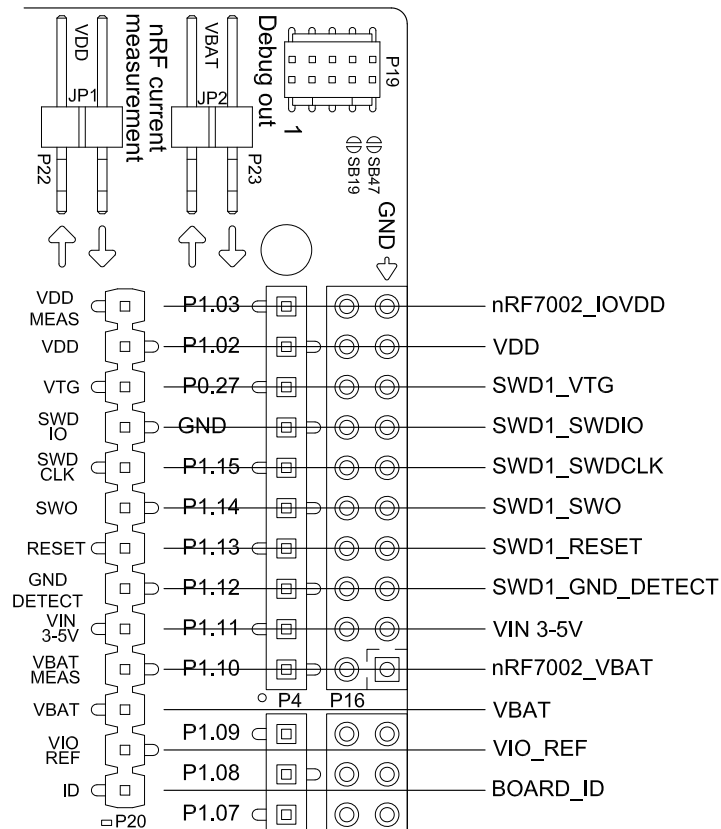


Figure 27: Debug output connector P20

Pin number	Signal	Description
1	nRF7002_IOVDD	nRF7002 DK companion IC power domain for current measurement
2	VDD	Main nRF7002 DK power domain
3	SWD1_VTG	Voltage supply from the external target
4	SWD1_SWDIO	Serial Wire debug Data Input/Output
5	SH_SWCLK	Serial Wire Clock
6	SWD1_SWO	The SWO line is not needed for programming and debugging over SWD
7	SH_RESET	Reset line
8	SH_GND_DETECT	Ground detect
9	VIN3-5V	Voltage supply
10	nRF7002_VBAT	nRF7002 DK VBAT power domain for current measurement
11	VBAT	Main VBAT power domain
12	VIO_REF	GPIO voltage reference input
13	BOARD_ID	DK ID resistor

Table 9: Pinout of connector P20 for programming external targets

4.12 Antennas

The nRF7002 DK has two antennas, a 2.4 GHz antenna and a dual-band 2.4GHz/5GHz antenna.

It is possible to configure the board to use only the dual-band antenna for both Wi-Fi and Bluetooth Low Energy by controlling the two RF switches or to use separate antennas for Bluetooth and Wi-Fi.

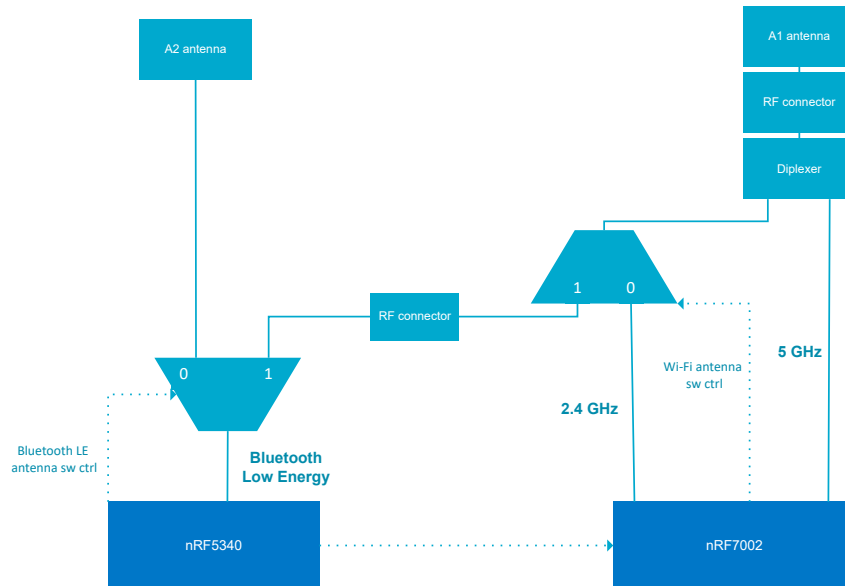


Figure 28: nRF7002 DK antenna configuration

Note: When using only the dual-band antenna, the Bluetooth Low Energy and 2.4GHz Wi-Fi share input to the diplexer and the Wi-Fi antenna switch needs to be controlled to select which signal to route.

4.13 NFC antenna interface

The nRF7002 DK supports an *NFC* tag.

NFC-A listen mode operation is supported on the nRF5340 SoC. The *NFC* antenna input is available on connector **J5** on the nRF7002 DK.

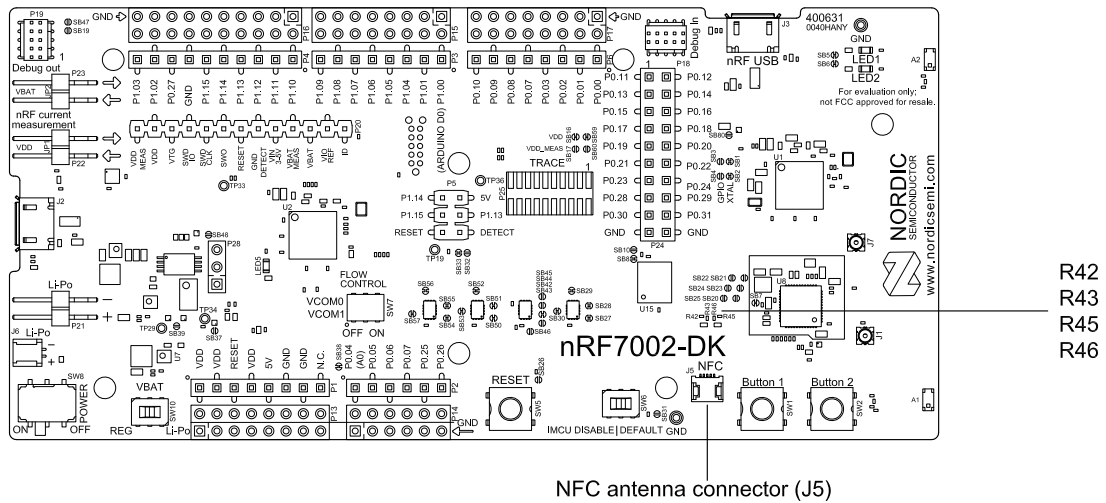


Figure 29: NFC antenna connector

NFC uses the **W1** (NFC1) and **AA1** (NFC2) pins to connect the antenna. These pins are shared with GPIOs (**P0.02** and **P0.03**), and the PROTECT field in the NFCPINS register in UICR defines the usage of these pins and their protection level against abnormal voltages. The content of the NFCPINS register is reloaded at every reset.

Note: The NFC pins are enabled by default. NFC can be disabled and GPIOs enabled by setting the CONFIG_NFCT_PINS_AS_GPIOs to `y`. See [Configuring your application](#) for instructions.

Pins **W1** and **AA1** are by default configured to use the NFC antenna, but if they are needed as normal GPIOs, **R43** and **R46** must be NC and **R42** and **R45** must be shorted by **OR**.

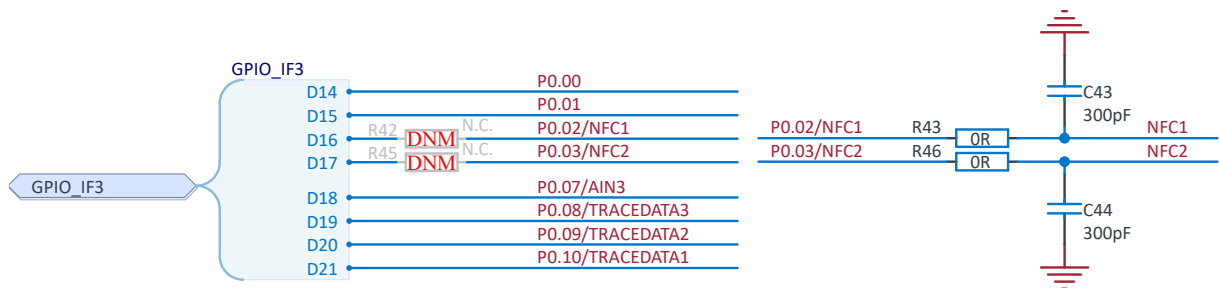


Figure 30: NFC input

4.14 Extra op-amp

The voltage follower for the power supply uses a dual package *Operational Amplifier (op-amp)*.

The extra op-amp is routed to a connector (**P28**, not mounted) so that it is accessible for the user.

For more information on the power supply, see [Power supply](#) on page 13.

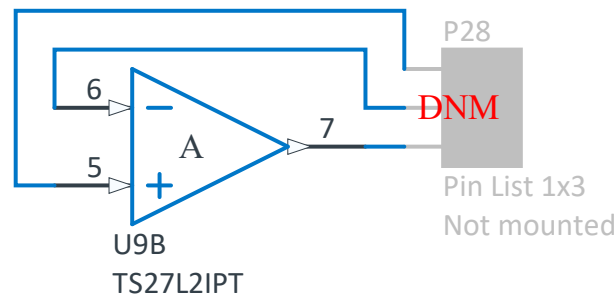


Figure 31: Extra op-amp

4.15 Solder bridge configuration

The nRF7002 DK has a range of solder bridges for enabling or disabling functionality on the DK. Changes to these are not needed for normal use of the DK.

The following table is a complete overview of the solder bridges on the nRF7002 DK.

Solder bridge	Default	Function
SB1	Closed	Cut to disconnect the 32.768 kHz on P0.01
SB2	Closed	Cut to disconnect the 32.768 kHz on P0.00
SB3	Open	Short to enable P0.01 as normal GPIO
SB4	Open	Short to enable P0.00 as normal GPIO
SB5	Closed	Cut to disconnect LED1
SB6	Closed	Cut to disconnect LED2
SB7	Closed	Cut to disconnect BUCK_EN. Refer to Tracing instructions on page 24
SB8	Closed	Cut to disconnect external memory from VDD
SB10	Open	Short to connect External Memory to VDD_MEAS
SB11	Open	Short to connect P0.08 to headers
SB12	Open	Short to connect P0.09 to headers
SB13	Open	Short to connect P0.10 to headers
SB14	Open	Short to connect P0.11 to headers
SB16	Closed	Cut to disconnect nRF5340 SoC from VDD
SB17	Open	Short to connect nRF5340 SoC to VDD_MEAS
SB19	Open	Short to enable voltage follower of the external device when using the debug out connector
SB20	Open	Short to connect P0.16 to the P24 header
SB21	Open	Short to connect P0.17 to the P24 header
SB22	Open	Short to connect P0.13 to the P24 header
SB23	Open	Short to connect P0.18 to the P24 header
SB24	Open	Short to connect P0.14 to the P24 header

Solder bridge	Default	Function
SB25	Open	Short to connect P0.15 to the P24 header
SB26	Open	Short to enable the pull-up resistor of the BOOT/RESET line
SB27	Closed	Cut to disconnect the nRF5340 UART1 line from the signal switch and interface MCU
SB28	Closed	Cut to disconnect the nRF5340 UART1 line from the signal switch and interface MCU
SB29	Closed	Cut to disconnect the nRF5340 UART1 line from the signal switch and the interface MCU
SB30	Closed	Cut to disconnect the nRF5340 UART1 line from the signal switch and interface MCU
SB31	Open	Short to bypass the USB detect switch
SB32	Open	Short to permanently enable the I ² C pull-up resistors
SB33	Closed	Cut to permanently disable the I ² C pull-up resistors
SB37	Open	Short to bypass the interface MCU power switch
SB38	Closed	Cut to disable VDD power to the Arduino interface
SB39	Open	Short to bypass the power switch for regulator or external supply
SB42	Closed	Cut to disconnect IF Boot/Reset button from nRF5340 reset pin when the interface MCU is disconnected
SB43	Open	Short to connect IF Boot/Reset button to RESET pin on the Arduino interface
SB44	Open	Short to connect the RESET pin on the Arduino interface to the nRF5340 reset pin
SB45	Open	Short to connect the RESET pin on the Arduino interface to the interface nRF5340 reset pin when the interface MCU is disconnected
SB46	Open	Short to connect the RESET pin on the Arduino interface to the interface MCU Boot when the interface MCU is disconnected
SB47	Open	Short to enable power supply of the external device when using the debug out connector
SB48	Open	Short to bypass the interface MCU USB power switch
SB50	Closed	Cut to disconnect the nRF5340 UART2 line from the signal switch and interface MCU
SB51	Closed	Cut to disconnect the nRF5340 UART2 line from the signal switch and interface MCU
SB52	Closed	Cut to disconnect the nRF5340 UART2 line from the signal switch and the interface MCU
SB53	Closed	Cut to disconnect the nRF5340 UART2 line from the signal switch and interface MCU
SB54	Closed	Cut to disconnect the nRF5340 SWDIO line from the signal switch and interface MCU

Solder bridge	Default	Function
SB55	Closed	Cut to disconnect the nRF5340 SWDCLK line from the signal switch and interface MCU
SB56	Closed	Cut to disconnect the nRF5340 RESET line from the signal switch and interface MCU
SB57	Closed	Cut to disconnect the nRF5340 SWO line from the signal switch and the interface MCU
SB59	Closed	Cut to disconnect debug in and trace reference voltage to VDD
SB60	Open	Solder to connect debug in and trace reference voltage to VDD_MEAS
SB80	Open	Short to bypass the power switch for the VBUS of nRF5340

Table 10: Solder bridge configuration

5 Measuring current

The current drawn by the nRF7002 companion IC can be monitored on the nRF7002 DK.

Current can be measured using any of the following test instruments.

- Power analyzer
- Oscilloscope
- Ampere meter
- Power Profiler Kit II

Power analyzer and Power Profiler Kit II (PPK2) measurements are not described in this document. See the [PPK2](#) product page for more information.

For measurement instructions, see sections [Using an oscilloscope for current profile measurement](#) on page 35 and [Using an ampere meter for current measurement](#) on page 36.

The nRF7002 companion IC has two available power supplies, VDD (1.8 V) and VBAT (2.9-4.5 V). The nRF7002 DK is prepared for measuring current on both domains. Only the VBAT domain current measurement is described here, but the approach is the same with the VDD supply. See the following table for the corresponding components.

Component	VDD	VBAT
Measurement connector	P22	P23

Table 11: Components for current measurement on VDD and VBAT

When measuring the current consumption:

- It is not recommended to use a USB connector to power the DK during current measurements. However, when measuring current on an application using the USB interface of the nRF5340 SoC, the USB must be connected. It is recommended to power the DK from an external power supply on connector **P21** (3.6 V) or through the *Li-Poly* connector **J6** (2.9-4.5 V).
- The current measurements are unreliable when a serial terminal is connected to the virtual COM port.
- After programming the nRF5340 SoC, disconnect the USB for the interface MCU.

For more information on current measurement, see the tutorial [Current measurement guide: Introduction](#).

5.1 Preparing the nRF7002 DK

To measure current, you must first prepare the DK.

The suggested configurations split the power domains for the nRF7002 DK and the rest of the DK.

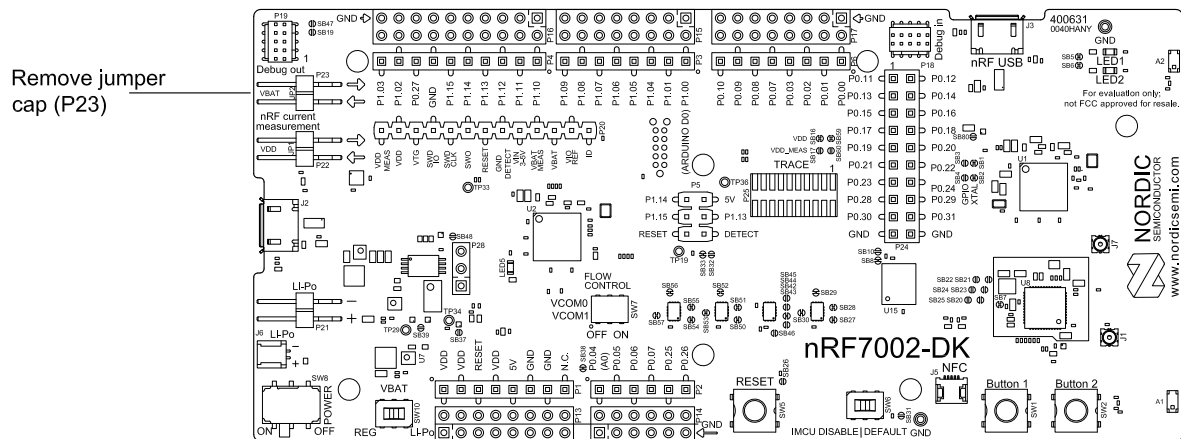


Figure 32: Preparing the DK for current measurements

- To put **P23** in series with the load, remove the jumper cap.
- To restore normal DK function after measurement, apply a jumper on **P23**.
- To reprogram the nRF5340 SoC while the DK is prepared for current measurements, remove measurement devices from **P23** and then connect the USB cable.

5.2 Using an oscilloscope for current profile measurement

An oscilloscope can be used to measure both the average current over a given time interval and capture the current profile.

Make sure you have followed the instructions in [Preparing the nRF7002 DK](#) on page 34.

1. Mount a 10 Ω resistor between the pins on **P23**.
2. Connect an oscilloscope in differential mode or similar with two probes on the pins of the **P23** connector, as shown in the following figure.
3. Calculate or plot the instantaneous current from the voltage drop across the 10 Ω resistor by taking the difference of the voltages measured on the two probes. The voltage drop is proportional to the current. The 10 Ω resistor causes a 10 mV drop for each 1 mA drawn by the circuit being measured.

The plotted voltage drop can be used to calculate the current at a given point in time. The current can then be averaged or integrated to analyze current and energy consumption over a period.

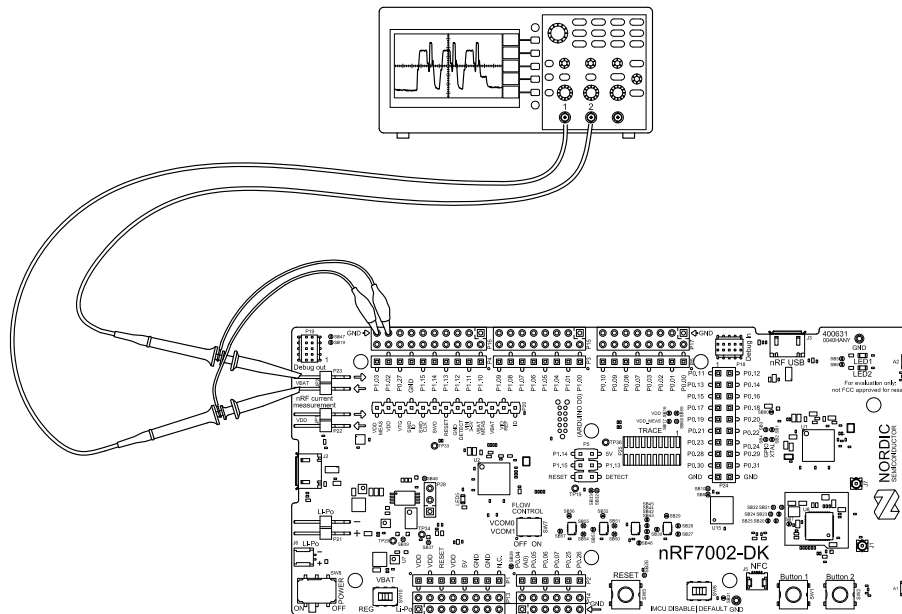


Figure 33: Current measurement with an oscilloscope

The following actions can be taken to reduce noise:

- Use probes with 1x attenuation.
- Enable averaging mode to reduce random noise.
- Enable high resolution function if available.

A minimum of one sample every 5 μ s is needed to accurately measure the average current.

5.3 Using an ampere meter for current measurement

The average current drawn by the nRF7002 DK can be measured using an ampere meter. This method monitors the current in series with the nRF device.

Make sure you have prepared the DK as described in [Preparing the nRF7002 DK](#) on page 34.

Connect an ampere meter between the pins of connector **P23** as shown in the following figure.

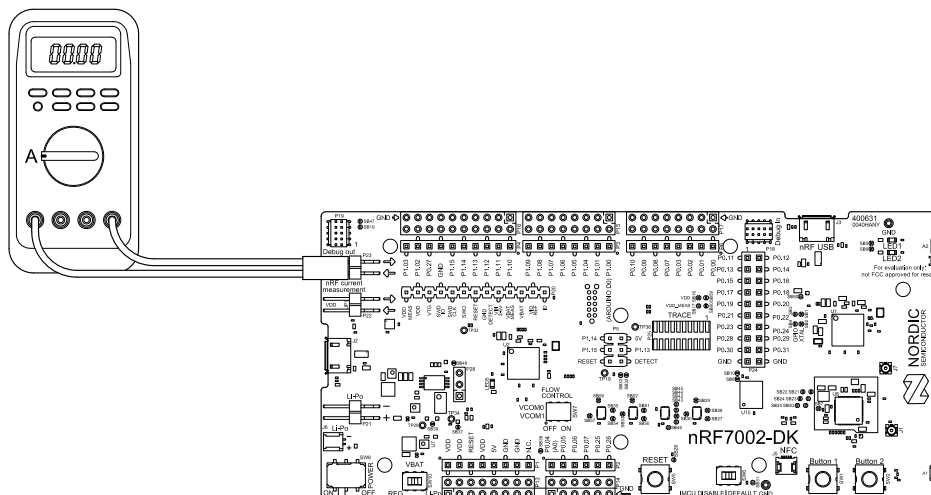


Figure 34: Current measurement with an ampere meter

An ampere meter will measure the average current drawn by the nRF7002 DK if:

- The DK is in a state where it draws a constant current for the activity on the device changing load current, like Wi-Fi connection events, is repeated continuously and has a short cycle time (less than 100 ms) so that the ampere meter averages whole load cycles and not parts of the cycle.
- The dynamic range of the ampere meter is wide enough to give accurate measurements from 1 μA to 15 mA.

It is recommend to use a true *Root Mean Square (RMS)* ampere meter.

6 RF measurements

The nRF7002 DK is equipped with two small coaxial connectors to measure RF signals from the nRF7002 (J1) or nRF5340 SoC (J7) using a spectrum analyzer.

The connectors are of SWF type (Murata part no. MM8130-2600) with an internal switch. By default, when no cable is attached, the RF signal is routed to the onboard chip antenna.

In this example, a test probe (Murata part no. MXHS83QE3000) is used with a standard *SubMiniature Version A (SMA)* connection for instruments (the test probe is not included with the kit). When connecting the test probe, the internal switch in the SWF connector disconnects the onboard antenna and connects the RF signal from the nRF5340 SoC to the test probe.

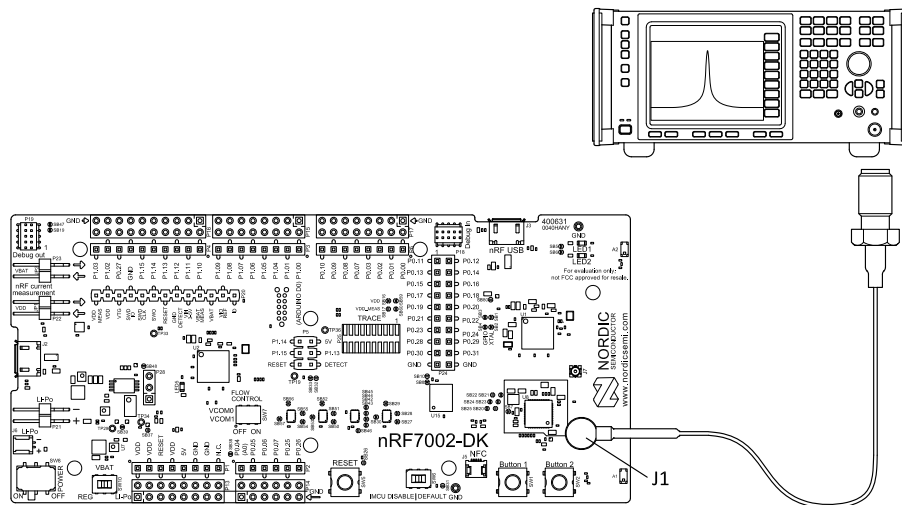


Figure 35: Connecting a spectrum analyzer to J1 on the nRF7002 DK

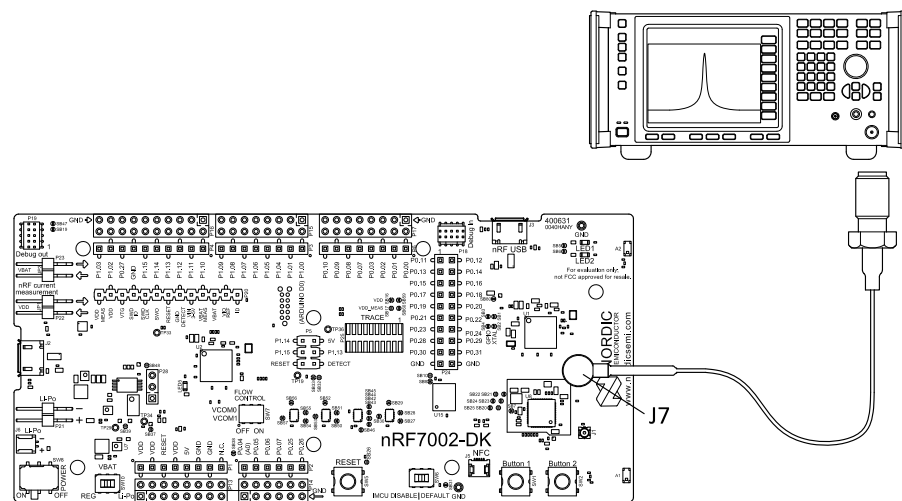


Figure 36: Connecting a spectrum analyzer to J7 on the nRF7002 DK

The connector and test probe add loss to the RF signal, which should be taken into account when measuring. See the following table for more information or consult the test probe user guide if you are using another model.

Frequency (MHz)	Loss (dB)
2440	1.0
4880	1.7
7320	2.6

Table 12: Typical loss in connector and test probe (Murata part no. MXHS83QE3000)

Glossary

Clear to Send (CTS)

In flow control, the receiving end is ready and telling the far end to start sending.

Development Kit (DK)

A hardware development platform used for application development.

Data Terminal Ready (DTR)

A control signal in RS-232 serial communications transmitted from data terminal equipment, such as a computer, to data communications equipment.

Electrostatic Discharge (ESD)

A sudden discharge of electric current between two electrically charged objects.

Embedded Trace Macrocell (ETM)

A real-time trace module providing instruction and data tracing of a processor.

General-Purpose Input/Output (GPIO)

A digital signal pin that can be used as input, output, or both. It is uncommitted and can be controlled by the user at runtime.

Hardware Flow Control (HWFC)

A handshaking mechanism used to prevent an overflow of bytes in modems. It uses two dedicated pins on the RS-232 connector, Request to Send and Clear to Send.

Inter-integrated Circuit (I²C)

A multi-master, multi-slave, packet-switched, single-ended, serial computer bus.

Integrated Circuit (IC)

A semiconductor chip consisting of fabricated transistors, resistors, and capacitors.

Integrated Development Environment (IDE)

A software application that provides facilities for software development.

Instrumentation Trace Macrocell (ITM)

An application-driven trace source that supports printf() style debugging to trace operating system and application events and generates diagnostic system information.

Lithium-polymer (Li-Poly)

A rechargeable battery of lithium-ion technology using a polymer electrolyte instead of a liquid electrolyte.

Mass Storage Device (MSD)

Any storage device that makes it possible to store and port large amounts of data in a permanent and machine-readable fashion.

Near Field Communication (NFC)

A standards-based short-range wireless connectivity technology that enables two electronic devices to establish communication by bringing them close to each other.

NFC-A Listen Mode

Initial mode of an NFC Forum Device when it does not generate a carrier. The device listens for the remote field of another device. See [Near Field Communication \(NFC\)](#) on page 40.

Operational Amplifier (op-amp)

A high-gain voltage amplifier that has a differential input and, usually, a single output.

One Time Programmable (OTP) memory

A type of non-volatile memory that permits data to be written to memory only once.

Printed Circuit Board (PCB)

A board that connects electronic components.

Receive Data (RXD)

A signal line in a serial interface that receives data from another device.

Quad Serial Peripheral Interface (QSPI)

A Serial Peripheral Interface (SPI) controller that allows the use of multiple data lines.

Request to Send (RTS)

In flow control, the transmitting end is ready and requesting the far end for a permission to transfer data.

Root Mean Square (RMS)

An RMS meter calculates the equivalent Direct Current (DC) value of an Alternating Current (AC) waveform. A true RMS meter can accurately measure both pure waves and the more complex nonsinusoidal waves.

Serial Clock (SCL)

A pin used by the I²C module to control the I²C bus lines.

Serial Data (SDA)

A pin used by the I²C module to control the I²C bus lines.

SubMiniature Version A (SMA)

A semi-precision coaxial RF connector for coaxial cables with a screw-type coupling mechanism.

System on Chip (SoC)

A microchip that integrates all the necessary electronic circuits and components of a computer or other electronic systems on a single integrated circuit.

Serial Peripheral Interface (SPI)

Synchronous serial communication interface specification used for short-distance communication.

Serial Wire Debug (SWD)

A standard two-wire interface for programming and debugging Arm CPUs.

Microwave coaxial connector with switch (SWF)

A small, RF surface-mount switch connector series for wireless applications.

Serial Wire Output (SWO)

A data line for tracing and logging.

Transmit Data (TXD)

A signal line in a serial interface that transmits data to another device.

Universal Asynchronous Receiver/Transmitter (UART)

A hardware device for asynchronous serial communication between devices.

Universal Serial Bus (USB)

An industry standard that establishes specifications for cables and connectors and protocols for connection, communication, and power supply between computers, peripheral devices, and other computers.

Recommended reading

In addition to the information in this document, you may need to consult other documents.

Nordic documentation

- [nRF7002 Product Specification](#)
- [nRF5340 Product Specification](#)
- [nRF Connect SDK](#)

FCC regulatory notice

The following regulatory notices apply to the nRF7002 DK.

This kit has not been authorized under the rules of the FCC and is designed to allow:

- Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine whether to incorporate such items in a finished product.
- Software developers to write software applications for use with the end product.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of 47 CFR Chapter I - FCC, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of the latter chapter.

Legal notices

By using this documentation you agree to our terms and conditions of use. Nordic Semiconductor may change these terms and conditions at any time without notice.

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function, or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Nordic Semiconductor ASA 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. If there are any discrepancies, ambiguities or conflicts in Nordic Semiconductor's documentation, the Product Specification prevails.

Nordic Semiconductor ASA reserves the right to make corrections, enhancements, and other changes to this document without notice.

Life support applications

Nordic Semiconductor products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury.

Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

RoHS and REACH statement

Complete hazardous substance reports, material composition reports and latest version of Nordic's REACH statement can be found on our website www.nordicsemi.com.

Trademarks

All trademarks, service marks, trade names, product names, and logos appearing in this documentation are the property of their respective owners.

Copyright notice

© 2023 Nordic Semiconductor ASA. All rights are reserved. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.

