## Contents

Revision history ........................................ iv

1 Introduction ........................................... 5

2 Kit content ........................................... 6
   2.1 Hardware content .................................. 6
   2.2 Related documentation .............................. 6

3 Getting started ....................................... 7

4 Operating modes ...................................... 8
   4.1 Default mode: Interface MCU ....................... 8
      4.1.1 Device programming ........................... 8
      4.1.2 Virtual COM port ............................... 8
      4.1.3 MSD ............................................ 9
      4.1.4 Reset ........................................... 9
   4.2 nRF ONLY mode .................................... 9
      4.2.1 USB detect .................................... 10

5 Hardware description ................................ 11
   5.1 Block diagram ...................................... 11
   5.2 Hardware figures ................................... 11
   5.3 Power supply ....................................... 12
      5.3.1 nRF9160 supply ................................ 14
      5.3.2 VDD supply rail ................................ 15
      5.3.3 Other power domains ......................... 15
   5.4 Antenna interfaces ................................ 15
   5.5 GPS .................................................. 16
   5.6 GPIO interfaces ..................................... 17
   5.7 nRF52840 ........................................... 18
      5.7.1 nRF9160 DK board control .................... 19
      5.7.2 Bluetooth/IEEE 802.15.4 network processor .. 21
   5.8 Buttons, slide switches, and LEDs ................ 22
   5.9 Debug input and trace options .................... 23
      5.9.1 Debug output .................................. 24
   5.10 Signal routing switches ........................... 25
      5.10.1 Switches for buttons and LEDs ............. 25
   5.11 SIM and eSIM ...................................... 27
   5.12 Additional nRF9160 interfaces .................. 28
   5.13 SiP enable ......................................... 29
   5.14 Solder bridge configuration ...................... 29

6 Measuring current .................................. 31
   6.1 Preparing the development kit for current measurements ... 31
   6.2 Using an oscilloscope for current profile measurement ... 32
   6.3 Using a current meter for current measurement .......... 32

7 RF measurements ..................................... 33
Revised history

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 2019</td>
<td>0.7.2</td>
<td>Updated:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>nRF9160 supply</strong> on page 14</td>
</tr>
<tr>
<td>March 2019</td>
<td>0.7.1</td>
<td>Updated to match DK v0.8.5:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Hardware content</strong> on page 6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>nRF9160 supply</strong> on page 14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Antenna interfaces</strong> on page 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>GPS</strong> on page 16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>nRF9160 DK board control</strong> on page 19</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Solder bridge configuration</strong> on page 29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Preparing the development kit for current measurements on page 31</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>RF measurements</strong> on page 33</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Radiated performance of <strong>nRF9160 DK</strong> on page 34</td>
</tr>
<tr>
<td>December 2018</td>
<td>0.7</td>
<td>Updated to match DK v0.8.2</td>
</tr>
<tr>
<td>November 2018</td>
<td>0.5.1</td>
<td>Preview DK changed into DK</td>
</tr>
<tr>
<td>October 2018</td>
<td>0.5</td>
<td>First release</td>
</tr>
</tbody>
</table>
1 Introduction

The nRF9160 DK (Development Kit) is a hardware development platform to be used for the design and development of application firmware on the nRF9160 LTE Cat-M1 and Cat-NB1 System in Package (SiP).

The board includes all necessary external circuitry like a SIM card holder and an antenna and it provides developers access to all I/O pins and relevant module interfaces.

**Note:** The DK can be connected to a base station by the onboard LTE antenna or to an LTE emulator by an RF cable. See our nRF9160 Certifications webpage to find the bands supported by the LTE modem. If no SIM card is inserted, the DK cannot initiate communication with the LTE network.

The key features of the development kit are:

- nRF9160 SiP
- LTE antenna that supports all bands supported by the SiP
- Global Positioning System (GPS) antenna
- nRF52840 as a board controller and network processor for Bluetooth® and IEEE 802.15.4 protocols
- Buttons, switches, and LEDs for user interaction
- I/O interface for Arduino form factor plug-in modules
- SEGGER J-Link OB Debugger with debug out functionality
- UART interface through virtual COM port
- USB connection for debug/programming and power
- SIM card socket for nano-SIM (4FF SIM)
- Interfaces for nRF9160 current consumption measurements

**Note:** nRF9160 DK is compliant with the PS1 classification according to the IEC 62368-1 standard.

---

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.

---

Skilled person: Person with relevant education or experience to enable him or her to identify hazards and to take appropriate actions to reduce the risks of injury to themselves and others.
Kit content

The nRF9160 DK includes hardware, preprogrammed firmware, documentation, hardware schematics, and layout files.

2.1 Hardware content

The nRF9160 DK contains the development kit board PCA10090 and a SIM card.

![nRF9160 DK board (PCA10090)](image)

![SIM card](image)

Figure 1: nRF9160 DK kit content

**Hardware files**

The hardware design files including schematics, PCB layout files, bill of materials, and Gerber files for the nRF9160 DK will be available on the product page, nRF91 SiP Series.

2.2 Related documentation

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

**Nordic documentation**

- nRF9160 Product Specification
- nRF91 Modem Firmware Release Notes
- nRF Connect SDK documentation
- nRF91 AT Commands Reference Guide
To get started with nRF9160 DK, go to nRF9160 DK Getting Started and follow the instructions there.
Operating modes

The nRF9160 DK has various modes of operation.

4.1 Default mode: Interface MCU

The primary interface for programming and debugging the nRF9160 DK is the USB port (J4). The USB port is connected to an interface MCU which embeds a SEGGER J-Link-OB (On Board) debug probe.

![Figure 2: Interface MCU](image)

4.1.1 Device programming

The nRF9160 DK supports SWD programming interfaces for both onboard and off-board nRF targets.

The primary target for programming and debugging in the DK is the nRF9160. The interface MCU also supports programming the onboard nRF52840 as well as external nRF devices fitted on a shield or through a connector to external boards such as the user's own prototypes.

The interface MCU will automatically detect if external targets are plugged in. The SW5 switch is used to select nRF9160 or nRF52840 for programming or debugging on board. Note that there are significant limitations on using the nRF52840 on the nRF9160 DK. For more information, see nRF9160 DK board control on page 19.

4.1.2 Virtual COM port

The interface MCU also features three UART interfaces through three virtual COM ports.

The virtual COM ports are the following:

- **VCOM0** – Connected to nRF9160 (default)
- **VCOM1** – Connected to nRF52840 (nonconfigurable)
- **VCOM2** – Not connected to nRF9160 (default)

For details on routing VCOM0 and VCOM2, see nRF9160 DK board control on page 19.

The virtual COM ports have the following features:

- Flexible baud rate settings up to 1 Mbps
- RTS/CTS-style **Hardware Flow Control (HWFC)** handling

The table below shows an overview of the GPIOs used for the UART connections on the nRF9160 and nRF52840:
### Operating modes

<table>
<thead>
<tr>
<th></th>
<th>nRF91 APP1</th>
<th>nRF91 APP2</th>
<th>nRF52840</th>
</tr>
</thead>
<tbody>
<tr>
<td>TXD</td>
<td>P0.29</td>
<td>P0.01</td>
<td>P0.03</td>
</tr>
<tr>
<td>RXD</td>
<td>P0.28</td>
<td>P0.00</td>
<td>P0.05</td>
</tr>
<tr>
<td>CTS</td>
<td>P0.26</td>
<td>P0.15</td>
<td>P0.07</td>
</tr>
<tr>
<td>RTS</td>
<td>P0.27</td>
<td>P0.14</td>
<td>P1.08</td>
</tr>
</tbody>
</table>

*Table 1: GPIOs used for virtual COM ports on nRF9160 and nRF52840*

**Note:**
- Baud rate 921 600 is not supported through the virtual COM port.

#### 4.1.3 MSD

The interface MCU features a mass storage device (MSD). This makes the development kit appear as an external drive on your computer.

This drive can be used for drag-and-drop programming. However, files cannot be stored on this drive. By copying a HEX file to the drive, the interface MCU will program the file to the device. The J-Link OB will program the target that is selected with the nRF52/nRF91 switch (SW5). For limitations on nRF52840 use, see [nRF9160 DK board control](#) on page 19.

**Note:**
- Windows might try to defragment the MSD part of the interface MCU. If this happens, the interface MCU will disconnect and become unresponsive. To return to normal operation, the development kit must be power cycled.
- 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 will become unresponsive.
- If your computer is set up to boot from USB, it can try to boot from the development kit if the development kit is connected during boot. This can be avoided by unplugging the development kit before a computer restart or changing the boot sequence of the computer.

You can also disable the MSD of the kit 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 development kit and stay this way until changed again.

#### 4.1.4 Reset

The nRF9160 DK board is equipped with a RESET button (SW2).

By default, the RESET button is connected to the interface MCU that will forward the reset signal to the nRF9160 or nRF52840, depending on the state of the nRF52/nRF91 switch. If nRF ONLY is activated, the RESET button will be connected to the nRF9160 directly.

#### 4.2 nRF ONLY mode

The nRF ONLY mode disconnects the interface MCU from the nRF9160 using analog switches.

This is done to isolate the chip on the board as much as possible, and can be of use when measuring currents on low-power applications.
The development kit detects if there is a USB cable plugged in (see USB detect on page 10) and will disconnect and power down the interface MCU when another supply than the USB is used. If the USB connector is used for power supply only (USB battery back) you can also disable the interface MCU through SW1.

USB detect and SW1 also control the routing of the RESET signal from the RESET button (SW2). Normally, it is routed through the interface MCU, but if that is disabled, the reset button will be routed directly to the nRF9160.

There are also a number of other reset routing options available through the use of solder bridges:

- When the interface MCU is disconnected, the RESET button is connected to pin 32 (nRESET) of the nRF9160. The RESET button can be disconnected from the nRF9160 by cutting SB25.
- When the interface MCU is disconnected, shorting SB29 will connect the RESET pin of the Arduino interface to the reset pin of the nRF9160.
- When the interface MCU is connected, shorting SB30 will connect the RESET pin of the Arduino interface to the BOOT input of the interface MCU.
- Shorting SB27 will connect the RESET pin of the Arduino interface to the reset button.
- Shorting SB28 will connect the RESET pin of the Arduino interface to the reset pin P0.29 of the nRF9160.

4.2.1 USB detect

To detect when the USB for the interface MCU is connected, there is a circuit sensing the VBUS of USB connector J4.

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

The nRF9160 DK board PCA10090 can be used as a development platform for the nRF9160. It features an onboard programming and debugging solution.

5.1 Block diagram

The block diagram shows the main functionality of the nRF9160 DK.

5.2 Hardware figures

The nRF9160 DK hardware drawings show both sides of the PCA10090 board.
5.3 Power supply

nRF9160 DK has a flexible and configurable power supply system to allow software development and testing using different power sources and to facilitate accurate power measurements.

The power source options are:

- USB connector J4 (5 V)
- External supply on P28 (3.3 V–5.5 V)
- VIN 3–5 V on P15 (3.3 V–5.5 V)
Figure 9: Power sources and switches
To ensure that only one of the power sources are used on the board at a time, power switches are implemented on each of them as shown in Figure 9: Power sources and switches on page 13. These switches prioritize the supply sources in the following manner:

1. USB
2. P15
3. P28

This means that if power is connected to more than one the interfaces, the higher priority interface will be chosen to supply the board.

The supply voltage is then routed through the ON/OFF switch (SW9) to the common rail VSUPPLY, which acts as the source for the supply voltage regulators for the circuitry on the board.

The supply flows from VOUT to VIN, which is correct. The body diode of the internal transistor powers the VSUPPLY net, which supplies the gates controlling the enable signal of the switches.

The power switches will introduce a small voltage drop between the power source connected to the board and the VSUPPLY. To avoid this, the power switches can be bypassed by shorting one of the solder bridges:

<table>
<thead>
<tr>
<th>Power source</th>
<th>Power switch bypass</th>
<th>Voltage level</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB connector (J4)</td>
<td>SB24</td>
<td>5 V</td>
</tr>
<tr>
<td>External supply (P28)</td>
<td>SB36</td>
<td>3.3 V–5.5 V</td>
</tr>
<tr>
<td>VIN 3–5 V (P15)</td>
<td>SB4</td>
<td>3.3 V–5.5 V</td>
</tr>
</tbody>
</table>

*Table 2: Bypassing power switch*

**Note:** Connect only one power source at a time in this case. Shorting the solder bridges removes the reverse voltage protection.

### 5.3.1 nRF9160 supply

The nRF9160 has a supply range of 3.0–5.5 V and is therefore powered by the VSUPPLY rail directly.

Alternatively, the nRF9160 SiP can be supplied directly from an external source. To enable this, do the following:

1. Cut SB43.
2. Connect the positive terminal of the power supply to pin 1 of P24.
3. Connect the negative (ground) terminal of the power supply to pin 3 of P24.

*Figure 10: nRF9160 supply from external source*
5.3.2 VDD supply rail

VDD is the main supply for the rest of the circuitry on the board. It is regulated down from VSUPPLY by a buck regulator (U22).

You can set the VDD voltage to 1.8 V (default) or 3 V with SW11. Running 3 V GPIO with heavy load may degrade the LTE RF performance. A third option for customizing VDD voltage level is applying a custom voltage level to VIO_REF (P15). For more information, see GPIO - General purpose input/output in the nRF9160 Objective Product Specification.

VDD powers most of the other circuits and will set the GPIO signal amplitude between nRF9160 and other circuits on the board including connectors and PIN headers.

5.3.3 Other power domains

The interface MCU needs a 3.3 V for its USB interface supply, a low-dropout voltage regulator (U32) is used for this. This regulator also supplies the LEDs on the board, giving these a fixed supply.

5.4 Antenna interfaces

nRF9160 DK has three antenna interfaces mounted representing LTE, GPS, and the 2.4 GHz radio.

The LTE and 2.4 GHz RF signals are propagated through two coaxial connectors with switches that will disconnect the corresponding antenna from the radio if adapter cables are connected. This makes it possible to perform conducted measurements or attach external antennas to the radio.

The GPS signal is RX only. There is a Low-Noise Amplifier (LNA) and a Band-Pass Filter (BPF) that amplify and filter the signal before it is fed to the GPS RF port on the nRF9160. Switch SW12 controls if the input signal to the LNA is connected to RF connector P29 or if the signal is routed from the GPS antenna A2 through switch U34. The LNA is connected to the GPS antenna by default.

The relation between the switches, radios, and antennas are the following:

- J1 – Switch between the nRF9160 and the LTE antenna (A1)
- J7 – Switch between the nRF52840 and the 2.4 GHz antenna (A3)
- **P29** – GPS connector for an external GPS antenna

For more details of the GPS antenna interface, see GPS on page 16.

### 5.5 GPS

The nRF9160 has a dedicated GPS port to support global navigation.

The GPS signal is received either from the onboard or an external GPS antenna. The onboard antenna (A2) is connected to the combined LNA and BPF U3 through switches U33 and U34.

![Figure 13: Onboard GPS antenna (A2), LNA/BPF (U3), connector P29, and switch SW12](image)

The onboard antenna can be disconnected and an external GPS antenna enabled by attaching the external GPS antenna to connector **P29** and sliding switch **SW12** to CON position. 3.3 V DC is fed through **P29** to supply an external LNA, if desired. The external GPS antenna can be connected to connector **P29** (Hirose part no. U.FL-R-SMT-1). A compatible adapter cable is required to connect a GPS antenna to the DK.

Switch **U34** and the combined LNA and BPF are enabled by the GPS-enable signal from the nRF9160.

The signal is next amplified and filtered in the LNA and BPF U3 before it is fed to the nRF9160. This makes the GPS receiver more sensitive to GPS signals and less sensitive to interference from other sources on the DK or nearby.

**Note:**

- GPS signals do not usually penetrate ceilings or other structures that well. Therefore, for best GPS performance, the DK should be placed on a flat surface in an open space outside, far from sources of interference and other structures that may block the signals from space.
- This functionality is only available if the modem firmware used in the nRF9160 supports GPS.
5.6 GPIO interfaces

Access to the nRF9160 GPIOs is available from connectors P7, P10, P14, P19, and P27. The nRF9160 DK supports the Arduino UNO interface.

GPIO signals are also available on connectors P5, P6, P12, P17, and P25, which are on the bottom side of the board. By mounting pin lists on the connector footprints, the nRF9160 DK board can be used as a shield for Arduino motherboards.

For easy access to GPIO, power, and ground, the signals can also be found on the through-hole connectors P8, P12, P17, and P25.

Note: GPIO P0.29 is not available on any through-hole connector.
Hardware description

<table>
<thead>
<tr>
<th>GPIO nRF9160 DK</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0.00, P0.01, P0.14, and P0.15</td>
<td>Used as a second UART connection to the interface MCU. For more information, see Virtual COM port on page 8.</td>
</tr>
<tr>
<td>P0.02, P0.03, P0.04, P0.05, P0.06, P0.07, P0.08, and P0.09</td>
<td>Connected by default to buttons, slide switches, and LEDs. For more information, see Buttons, slide switches, and LEDs on page 22.</td>
</tr>
<tr>
<td>P0.17, P0.18, P0.19, P0.21, P0.22, P0.23, COEX0, COEX1, and COEX2</td>
<td>Used to connect the nRF9160 to the nRF9160 DK board control on page 19.</td>
</tr>
<tr>
<td>P0.26, P0.27, P0.28, and P0.29</td>
<td>Used as the primary UART connection to the interface MCU. For more information, see Virtual COM port on page 8.</td>
</tr>
</tbody>
</table>

Table 3: Default pin settings

| Figure 16: nRF9160 DK pins |

5.7 nRF52840

An nRF52840 Bluetooth/IEEE 802.15.4 System on Chip (SoC) is included on the nRF9160 DK board.

This device has two functions:
- nRF9160 DK board control
- Bluetooth/IEEE 802.15.4 network processor
5.7.1 nRF9160 DK board control

The nRF52840 controls analog switches on the nRF9160 DK, enabling routing of some of the nRF9160 GPIO pins to onboard functionality, for example LEDs, or the regular GPIO interfaces.

For details on which GPIOs on the nRF9160 can be routed by these analog switches, see the table below.
### Hardware description

#### Table 4: Board control routing

<table>
<thead>
<tr>
<th>Name</th>
<th>nRF52 switch control</th>
<th>Source nRF9160</th>
<th>Destination</th>
<th>Default¹</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>nRF91_APP1</td>
<td>P1.14</td>
<td>P0.26</td>
<td>Interface MCU VCOM0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.28</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.29</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>nRF91_APP2</td>
<td>P1.12 (P0.12; see footnote)</td>
<td>P0.01</td>
<td>P14</td>
<td>2</td>
<td>Interface MCU VCOM2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.00</td>
<td></td>
<td></td>
<td>TXD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.15</td>
<td></td>
<td></td>
<td>RTS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.14</td>
<td></td>
<td></td>
<td>CTS</td>
</tr>
<tr>
<td>nRF91_LED1</td>
<td>P1.05</td>
<td>P0.02</td>
<td></td>
<td></td>
<td>LED1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.04</td>
<td></td>
<td></td>
<td>LED3</td>
</tr>
<tr>
<td>nRF91_LED2</td>
<td>P1.07</td>
<td>P0.03</td>
<td></td>
<td></td>
<td>LED2</td>
</tr>
<tr>
<td>nRF91_LED3</td>
<td>P1.01</td>
<td>P0.06</td>
<td></td>
<td></td>
<td>LED3</td>
</tr>
<tr>
<td>nRF91_LED4</td>
<td>P1.03</td>
<td>P0.13</td>
<td></td>
<td></td>
<td>SWITCH1</td>
</tr>
<tr>
<td>nRF91_SWITCH1</td>
<td>P1.09</td>
<td>P0.14</td>
<td></td>
<td></td>
<td>SWITCH2</td>
</tr>
<tr>
<td>nRF91_SWITCH2</td>
<td>P0.08</td>
<td>P0.15</td>
<td></td>
<td></td>
<td>SWITCH2</td>
</tr>
<tr>
<td>nRF91_BUTTON1</td>
<td>P0.06</td>
<td>P0.17</td>
<td></td>
<td></td>
<td>BUTTON1</td>
</tr>
<tr>
<td>nRF91_BUTTON2</td>
<td>P0.26</td>
<td>P0.18</td>
<td></td>
<td></td>
<td>BUTTON2</td>
</tr>
<tr>
<td>nRF91_GPIO</td>
<td>P0.13</td>
<td>P0.19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>nRF91_TRACE</td>
<td>P0.24</td>
<td>P0.21</td>
<td>TRACECLK</td>
<td>nRF52</td>
<td>P0.17</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.22</td>
<td>TRACEDATA0</td>
<td>nRF52</td>
<td>P0.22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P0.23</td>
<td>TRACEDATA1</td>
<td></td>
<td>P1.04</td>
</tr>
<tr>
<td>nRF91_COEX</td>
<td>P1.10</td>
<td>COEX0</td>
<td></td>
<td></td>
<td>COEX0_PH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>COEX1</td>
<td></td>
<td></td>
<td>COEX1_PH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>COEX2</td>
<td></td>
<td></td>
<td>COEX2_PH</td>
</tr>
</tbody>
</table>

**Name**

The name to be used when referring to this signal path and the configuration needed to control it.

**nRF52 switch control**

The GPIOs on nRF52840 that are used to control the analog switches selecting between default and optional routing.

¹ For nRF91_APP2, two GPIOs need to be set high on nRF52840 to get the optional destination for the signals.
Source nRF9160
The GPIOs on nRF9160 that can be configured to have an alternative routing.

Destination "Default"
The default destination for the "Source nRF9160" if the "nRF52 switch control" pins are set as output and pin driver is low.

Destination "Optional"
The optional destination for the "Source nRF9160" if the "nRF52 switch control" pins are set as output and pin driver is high.

The setup described as "Destination Default" in the table above is active if all the GPIOs on nRF52840 listed as "nRF52 switch control" are set as output and driven low. There is a pull-down resistor on the switch control lines, and therefore, "Destination Default" can also be received if the control lines from the nRF52840 are not driven. If the optional routing is desired, the "nRF52840 switch control" pins have to be set as outputs and driven high.

The nRF52840 is preprogrammed with firmware that provides the default destination setting. Configurable code examples for the nRF52840 allowing to change the board routing can be found in the nRF Connect SDK. After the changes are done, the new program must be compiled and programmed to the nRF52840.

Note: To program and debug the nRF52840, nRF52 needs to be selected on the PROG/DEBUG switch (SW5).

Since this firmware in the nRF52840 decides the nRF9160 DK behavior, it is vital that it is always present in the nRF52840. If it is accidentally erased or firmware affecting the use of the key nRF52840 GPIOs is programmed in, nRF9160 DK functionality is not guaranteed. In such cases, nRF52840 needs to be preprogrammed with the default board control firmware from the nRF Connect SDK.

5.7.2 Bluetooth/IEEE 802.15.4 network processor
A second function of the onboard nRF52840 is to act as a Bluetooth or IEEE 802.15.4 network processor. This makes it possible to develop products that utilize both LTE and other RF protocols supported by Nordic on the nRF9160 DK.

Since most of the GPIOs of the nRF52840 are used for board control (see the schematic below), it is not intended to be a full programming and development target on the nRF9160 DK.

However, if combined with board control firmware, it can run code that requires limited access to GPIO, such as wireless controllers or network processor implementations of Bluetooth and IEEE 802.15.4 protocols from Nordic Semiconductor.

This enables the nRF9160 DK to implement a cellular to Bluetooth (and/or IEEE 802.15.4) bridge or gateway. To enable this, nRF52840 has a 2.4 GHz antenna fitted along with nine GPIOs that can be connected to nRF9160 through the board control routing. For details, see Table 4: Board control routing on page 20.

Firmware examples for such bridge or gateway implementations can be found in the nRF Connect SDK.
5.8 Buttons, slide switches, and LEDs

The nRF9160 DK board has four LEDs, two buttons, and two switches for simple user interaction. By default, they are connected to nRF9160 GPIOs as shown in the following table.

To change default nRF9160 GPIO connections, see nRF9160 DK board control on page 19

<table>
<thead>
<tr>
<th>Part</th>
<th>GPIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>LED 1</td>
<td>P0.02</td>
</tr>
<tr>
<td>LED 2</td>
<td>P0.03</td>
</tr>
<tr>
<td>LED 3</td>
<td>P0.04</td>
</tr>
<tr>
<td>LED 4</td>
<td>P0.05</td>
</tr>
<tr>
<td>Button1</td>
<td>P0.06</td>
</tr>
<tr>
<td>Button2</td>
<td>P0.07</td>
</tr>
<tr>
<td>Switch1</td>
<td>P0.08</td>
</tr>
<tr>
<td>Switch2</td>
<td>P0.09</td>
</tr>
</tbody>
</table>

Table 5: Button and LED connection

The nRF52840 can be used to disconnect any of the LEDs, slide switches, or buttons.

The buttons and switches are active low, meaning that the input will be connected to ground when the buttons are pushed or switches slid to the GND position. The buttons and switches have no external pull-up resistor, and therefore the P0.06, P0.07, P0.08, and P0.09 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 will illuminate the LED. The nRF9160 GPIOs control power transistors and LEDs are fed from a separate 3.3 V domain. Therefore, LED current will not be drawn from nRF9160 GPIOs or the nRF9160 supply.
5.9 Debug input and trace options

The primary debug interface on the nRF9160 DK is the Segger OB debugger available through the USB port. However, if other than USB supply is used on the board, this functionality will be disabled.

The Debug in connectors P3 (nRF9160) and P11 (nRF52840) make it possible to connect external debuggers for debugging when the interface USB cable is not connected or if the board is in nRF ONLY mode.
To utilize the SW trace feature on nRF9160, a footprint for a 20-pin connector is available (P4). If trace functionality is required, a 2×10 pin 1.27 mm pitch surface mount connector can be mounted. nRF9160 GPIOs used for the trace interface will not be available for application firmware use during trace.

**Note:** Connectors P3 and P4 overlap and share the same footprint.

<table>
<thead>
<tr>
<th>GPIO</th>
<th>Trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0.21</td>
<td>TRACECLK</td>
</tr>
<tr>
<td>P0.22</td>
<td>TRACEDATA[0]</td>
</tr>
<tr>
<td>P0.23</td>
<td>TRACEDATA[1]</td>
</tr>
<tr>
<td>P0.24</td>
<td>TRACEDATA[2]</td>
</tr>
<tr>
<td>P0.25</td>
<td>TRACEDATA[3]</td>
</tr>
</tbody>
</table>

**Table 6: nRF9160 trace interfaces**

### 5.9.1 Debug output

The nRF9160 DK board supports programming and debugging external boards with Nordic SoCs and SiPs. To debug an external board with SEGGER J-Link OB IF, connect to the Debug out connector P22 to your target board with a 10-pin flat cable.

When the external board is powered, the interface MCU will detect the supply voltage of the board and program/debug the target chip on the external board instead of the onboard nRF9160 and/or nRF52840.

**Note:** The voltage supported by external debugging/programming is the VDD voltage. This voltage can be selected to 1.8 V or 3 V using slide switch SW11. For optimal performance of the nRF9160 radio, it is recommended that only 1.8 V is used. Make sure the voltage level of the external board matches the VDD of the nRF9160 DK.

P15 can also be used as a debug out connector to program shield-mounted targets. For both P22 and P15, the interface MCU will detect the supply voltage on the mounted shield and program/debug the target.

If the interface MCU detects target power on both P22 and P15, it will program/debug the target connected to P22 by default.

If it is inconvenient to have a separate power supply on the external board, the nRF9160 DK can supply power through the Debug out connector P22. To enable this, short solder bridge SB43.
5.10 Signal routing switches

A number of the GPIO signals of the nRF9160 are routed through analog switches for use for onboard functionality or having them available on the pin headers for external circuitry or Arduino type shields.

The switches in the following figure enable the nRF ONLY mode. For more information, see nRF ONLY mode on page 9.

5.10.1 Switches for buttons and LEDs

On the nRF9160 DK board, there are a few analog switches that are used to connect and disconnect signals to control buttons, switches, and LEDs.

The switches are controlled by the nRF52840, which contains firmware from Nordic Semiconductor.
Figure 24: LED switch control
5.11 SIM and eSIM

The nRF9160 DK is designed to support both regular and embedded SIM (eSIM). For this purpose, it has a pluggable SIM card socket that takes a nano-sized SIM (4FF) and a non-populated footprint for an eSIM (MFF2).

Using the SIM socket is the default. If an eSIM is soldered on to the DK, switch SW8 can be used to select the eSIM. Connector P23 can be used to connect and monitor the traffic on the SIM interface.
5.12 Additional nRF9160 interfaces

The nRF9160 supports a few dedicated interfaces.

Coexistence pins

On the nRF9160 DK, there are three COEX pins enabling coexistence handling between multiple wireless devices. These pins are routed to the nRF52840 for use when nRF52840 is used as a network processor. These signals are also available on connector P1.

MAGPIO

The nRF9160 has an interface allowing the modem to control external circuitry through the MAGPIO pins. On the nRF9160 DK, MAGPIO0 is used to enable the GPS LNA, while the remaining two MAGPIOs are not in use in this DK. The user can access MAGPIO1 and MAGPIO2 through test points TP2 and TP3.
5.13 SiP enable

The SiP can be enabled or disabled by pulling pin 101 high and low.

By default, the enable signal is pulled high by resistor R3. To disable the SiP, connect to test point TP1 (on the back of the board) and pull the pin low.

5.14 Solder bridge configuration

The nRF9160 DK has a range of solder bridges for enabling or disabling functionality on the board. Changes to these are not needed for normal use of the development kit.

The following table is a complete overview of the solder bridges on the nRF9160 DK.

<table>
<thead>
<tr>
<th>Solder bridge</th>
<th>Default</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB2</td>
<td>Closed</td>
<td>Cut to disconnect VDD_GPIO supply to the SiP</td>
</tr>
<tr>
<td>SB3</td>
<td>Open</td>
<td>Short to bypass switch U26</td>
</tr>
<tr>
<td>SB4</td>
<td>Open</td>
<td>Short to bypass switch U31</td>
</tr>
<tr>
<td>SB5</td>
<td>Closed</td>
<td>Cut to disconnect the RESET signal from nRF52840</td>
</tr>
<tr>
<td>SB6</td>
<td>Closed</td>
<td>Cut to disconnect RTS signal from nRF52840</td>
</tr>
<tr>
<td>SB7</td>
<td>Closed</td>
<td>Cut to disconnect CTS signal from nRF52840</td>
</tr>
<tr>
<td>SB8</td>
<td>Closed</td>
<td>Cut to disconnect RXD signal from nRF52840</td>
</tr>
<tr>
<td>SB9</td>
<td>Closed</td>
<td>Cut to disconnect TXD signal from nRF52840</td>
</tr>
<tr>
<td>SB10</td>
<td>Closed</td>
<td>Cut to disconnect SWDIO from nRF52840</td>
</tr>
<tr>
<td>SB11</td>
<td>Closed</td>
<td>Cut to disconnect SWDCLK from nRF52840</td>
</tr>
<tr>
<td>SB12</td>
<td>Closed</td>
<td>Cut to disconnect the 3.3 V USB supply from the interface MCU (U2)</td>
</tr>
<tr>
<td>SB13</td>
<td>Closed</td>
<td>Cut to disconnect supply to LDO (U32)</td>
</tr>
<tr>
<td>SB14</td>
<td>Closed</td>
<td>Cut to disconnect RTS from the SiP</td>
</tr>
<tr>
<td>SB15</td>
<td>Closed</td>
<td>Cut to disconnect RTS from the SiP</td>
</tr>
<tr>
<td>SB16</td>
<td>Closed</td>
<td>Cut to disconnect RXD from the SiP</td>
</tr>
<tr>
<td>SB17</td>
<td>Closed</td>
<td>Cut to disconnect TXD from the SiP</td>
</tr>
<tr>
<td>SB21</td>
<td>Open</td>
<td>Short to bypass switch U27</td>
</tr>
<tr>
<td>SB22</td>
<td>Closed</td>
<td>Cut to disconnect protection diode D3</td>
</tr>
<tr>
<td>SB23</td>
<td>Open</td>
<td>Short to bypass protection diode D3</td>
</tr>
<tr>
<td>SB24</td>
<td>Open</td>
<td>Short to bypass the USB detect switch (Q3)</td>
</tr>
<tr>
<td>SB25</td>
<td>Closed</td>
<td>Cut to disconnect the RESET button from the nRF9160 reset pin when the interface MCU is disconnected</td>
</tr>
<tr>
<td>SB26</td>
<td>Closed</td>
<td>Cut to disable the pull-up resistor of the RESET button (SW2)</td>
</tr>
<tr>
<td>SB27</td>
<td>Open</td>
<td>Short to connect the RESET button to the RESET pin on the Arduino interface</td>
</tr>
<tr>
<td>Solder bridge</td>
<td>Default</td>
<td>Function</td>
</tr>
<tr>
<td>--------------</td>
<td>----------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SB28</td>
<td>Open</td>
<td>Short to connect the RESET pin on the Arduino interface to the nRF9160 reset pin</td>
</tr>
<tr>
<td>SB29</td>
<td>Open</td>
<td>Short to connect the RESET pin on the Arduino interface to the nRF9160 reset pin when the interface MCU is disconnected</td>
</tr>
<tr>
<td>SB30</td>
<td>Open</td>
<td>Short to connect the RESET pin on the Arduino interface to the interface MCU Boot when the interface MCU is disconnected</td>
</tr>
<tr>
<td>SB31</td>
<td>Open</td>
<td>Short to bypass switch U18</td>
</tr>
<tr>
<td>SB36</td>
<td>Open</td>
<td>Short to connect P0.12 to the SD card reader P2 (not populated by default)</td>
</tr>
<tr>
<td>SB37</td>
<td>Open</td>
<td>Short to enable power supply of the external device when using the Debug out connector (P22)</td>
</tr>
<tr>
<td>SB38</td>
<td>Open</td>
<td>Short to permanently enable the I²C pull-up resistors</td>
</tr>
<tr>
<td>SB39</td>
<td>Closed</td>
<td>Cut to permanently disable the I²C pull-up resistors</td>
</tr>
<tr>
<td>SB40</td>
<td>Open</td>
<td>Short to bypass switch U24</td>
</tr>
<tr>
<td>SB41</td>
<td>Open</td>
<td>Short to bypass switch U30</td>
</tr>
<tr>
<td>SB43</td>
<td>Closed</td>
<td>Cut to disconnect VDD_nRF and VDD_nRF'</td>
</tr>
<tr>
<td>SB46</td>
<td>Closed</td>
<td>Cut to disconnect the VSUPPLY and VDD_nRF'</td>
</tr>
<tr>
<td>SB49</td>
<td>Closed</td>
<td>Cut to disconnect VDD and VIO</td>
</tr>
</tbody>
</table>

*Table 7: Solder bridge configuration*
Measuring current

The current drawn by the nRF9160 can be monitored on the nRF9160 DK board.

Current can be measured using various test instruments. Examples of test equipment are the following:

- Power analyzer
- Oscilloscope
- Ampere-meter

Connector **P24** can be used for measuring current consumption or monitoring voltage levels to the nRF9160.

The use of a USB connector is not recommended for powering the board during current measurements due to potential noise from the USB power supply. Instead, the board should be powered by connecting a power supply to connector **P28**.

For more information on measuring, see sections **Using an oscilloscope for current profile measurement** on page 32 and **Using a current meter for current measurement** on page 32.

6.1 Preparing the development kit for current measurements

To measure the current consumption of the **SiP**, you must first prepare the board.

1. Cut solder bridge **SB43**.
   
   This cut disconnects the nRF9160 from the power management circuitries on the board.

2. To restore normal kit functionality after measurement, short **SB43** or apply a jumper on **P24** by shorting the two uppermost pins as shown in the figure above.

---

Figure 28: Solder bridge SB43 and P24 on the nRF9160 DK board

Figure 29: nRF power source
6.2 Using an oscilloscope for current profile measurement

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

Before you start, make sure you have prepared the board as described in section Preparing the development kit for current measurements on page 31.

Follow the steps below to measure the current profile of the nRF9160:

1. Solder a 0.5 Ω resistor to R60.

![Figure 30: Current measurement with an oscilloscope](image)

2. Connect probes to the two uppermost pins on connector P24.
3. Measure the current power profile by measuring the voltage drop over the 0.5 Ω resistor.

6.3 Using a current meter for current measurement

The current drawn by the nRF9160 can be measured using a current meter.

Before you start, make sure you have prepared the development kit board as described in Preparing the development kit for current measurements on page 31.

To connect the current meter in series with the nRF9160, connect the current meter to the two uppermost pins on connector P24.

![Figure 31: Current measurement with a current meter](image)

**Note:** Use a high-speed, high-dynamic-range ampere meter for the best and most reliable measurements. Current range switching in the current meter may affect the power supply to the nRF9160. High speed and bandwidth is required to detect rapid changes in the current consumption in the nRF9160.
RF measurements

The nRF9160 DK board is equipped with a small size coaxial connectors J1 and J7 for measuring the RF signal for LTE and 2.4 GHz. 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 antenna. The insertion loss in the adapter cable is approximately 0.5–1 dB.

An adapter is available (Murata part no. MXHS83QE3000) with a standard SMA connection on the other end for connecting instruments (the adapter is not included in the kit). When connecting the adapter, the internal switch in the SWF connector will disconnect the onboard antenna and connect the RF signal from the nRF9160 to the adapter.

![Figure 32: Connecting a spectrum analyzer](image-url)
Radiated performance of nRF9160 DK

The LTE antenna on the DK is optimized for global operation, supporting all LTE frequency bands in the region of 698–960 MHz and 1710–2200 MHz.

All antennas on the nRF9160 DK have fixed matching networks, which means that no matching configuration is needed to switch between the frequency bands. The average performance of the LTE antenna mounted on the nRF9160 DK is shown in the table below. The LTE antenna also supports other frequency ranges but it is not optimized for operating on these frequency bands.

<table>
<thead>
<tr>
<th>Band</th>
<th>Average return loss [dB]</th>
<th>Average efficiency [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-13.5</td>
<td>62.0</td>
</tr>
<tr>
<td>4</td>
<td>-7.1</td>
<td>47.1</td>
</tr>
<tr>
<td>13</td>
<td>-8.5</td>
<td>58.5</td>
</tr>
<tr>
<td>20</td>
<td>-10.5</td>
<td>73.8</td>
</tr>
</tbody>
</table>

*Table 8: Antenna performance versus radio band*

The 2.4 GHz antenna on the nRF9160 DK is optimized for the frequency range of 2.4–2.48 GHz. It has an average return loss of -15 dB, average efficiency being 80%.
Glossary

Band-Pass Filter (BPF)
An electronic device or circuit that passes frequencies within a certain range and rejects frequencies outside that range.

Cat-M1

Cat-NB1
Narrowband Internet of Things (NB-IoT) User Equipment (UE) category with 200 kHz UE bandwidth and a single RX antenna, specified in 3GPP Release 13.

Clear to Send (CTS)
In flow control, the receiving end is ready and telling the far end to start sending.

DK (Development Kit)
A development platform used for application development.

Fast Identity Online (FIDO)
A bundle of hardware and software modules serving as a tracing interface between traced device(s) and user application.

Global Positioning System (GPS)
A satellite-based radio navigation system that provides its users with accurate location and time information over the globe.

Hardware Flow Control (HWFC)
A handshaking mechanism used to prevent an overflow of bytes in modems. It is utilizing 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.

Low-Dropout Regulator (LDO)
A linear voltage regulator that can operate even when the supply voltage is very close to the desired output voltage.

Low-Noise Amplifier (LNA)
In a radio receiving system, an electronic amplifier that amplifies a very low-power signal without significantly degrading its signal-to-noise ratio.

nRF Cloud
A cloud solution for visualizing data from Nordic reference designs. See nRF Cloud.

Operational Amplifier (op-amp)
A high-gain voltage amplifier that has a differential input and, usually, a single output.
Receive Data (RXD)
A signal line in a serial interface that receives data from another device.

Request to Send (RTS)
In flow control, the transmitting end is ready and requesting the far end for a permission to transfer data.

SAW filter
A high-performing filter using Surface Acoustic Wave (SAW) technology. This technology employs piezoelectric transducers, which, when excited, produce waves that are used to filter out desired frequencies.

Surface Acoustic Wave (SAW)
An acoustic wave traveling along the surface of a material exhibiting elasticity, with an amplitude that typically decays exponentially with depth into the substrate.

System in Package (SiP)
A number of integrated circuits, often from different technologies, enclosed in a single module that performs as a system or subsystem.

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.

Transmit Data (TXD)
A signal line in a serial interface that transmits data to another device.
Acronyms and abbreviations

These acronyms and abbreviations are used in this document.

BPF
Band-Pass Filter

CTS
Clear to Send

DK
Development Kit

FIDO
Fast Identity Online

GPS
Global Positioning System

HWFC
Hardware Flow Control

I²C
Inter-integrated Circuit

LDO
Low-Dropout (Regulator)

LNA
Low-Noise Amplifier

RTS
Request to Send

RXD
Receive Data

SAW
Surface Acoustic Wave

SiP
System in Package

SoC
System on Chip

TXD
Transmit Data
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


The SVHC (Substances of Very High Concern) candidate list is continually being updated. 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

© 2019 Nordic Semiconductor ASA. All rights are reserved. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.