Nordic Thingy:91

User Guide

v1.2

NORDIC® SEMICONDUCTOR

Contents

Revision history . . . iv

1 Introduction . . . 5

2 Kit content . . . 7

2.1 Downloadable content . . . 7
2.2 Related documentation . . . 8

3 Getting started . . . 9

3.1 Connecting LTE Link Monitor . . . 11

4 Firmware . . . 12

4.1 Getting the firmware . . . 12
4.1.1 Download pre-compiled firmware images . . . 12
4.1.2 Compile the source code . . . 12
4.2 Operating modes . . . 13
4.3 GPS . . . 14
4.4 LTE Band Lock . . . 14
4.5 LTE-M / NB-IoT switching . . . 14

5 Firmware update . . . 15

5.1 Updating modem firmware . . . 15
5.2 Updating application firmware . . . 16
5.2.1 Updating application firmware on nRF9160 . . . 16
5.2.2 Updating application firmware on nRF52840 . . . 20

6 Hardware description . . . 22

6.1 Block diagram . . . 22
6.2 Hardware figures . . . 22
6.3 nRF9160 . . . 23
6.3.1 Antenna tuning . . . 24
6.3.2 RF measurements . . . 25
6.3.3 GPS . . . 25
6.3.4 SIM card . . . 26
6.4 nRF52840 . . . 27
6.4.1 NFC passive tag . . . 27
6.4.2 USB . . . 28
6.5 Pin maps . . . 28
6.6 Motion sensors . . . 31
6.7 Environment sensors . . . 33
6.8 Buzzer . . . 33
6.9 LEDs and buttons . . . 34
6.9.1 RGB LED . . . 34
6.9.2 Buttons . . . 34
6.10 Power supply . . . 34
6.10.1 PMIC . . . 35
6.10.2 Current measurement . . . 35
6.11 Programming and debugging interface . . . 36

6.12 Interface 37
6.12.1 N-MOS transistors 37
6.12.2 Connectors 38
6.12.3 Test points 39

7 Regulatory notices 42
7.1 FCC regulatory notices 42

Glossary 43

Acronyms and abbreviations 45

Legal notices 46

Revision history

Date	Version	Description
January 2020	1.2	Updated Introduction on page 5
December 2019	1.1	Updated • Kit content diagram and added a short description • Getting started on page 9 • Connecting LTE Link Monitor on page 11 • Buttons on page 34 • Figures with callouts indicating functionality of components Added the different ways to obtain firmware images for updating firmware, and operating modes: • Firmware on page 12 Updated different firmware update methods and added new update technique using USB (MCUBoot) • Firmware update on page 15
August 2019	1.0	First release

1 Introduction

The Nordic Thingy:91™ is a battery-operated prototyping platform for cellular IoT, certified for global operation. It is ideal for rapid development of prototypes for cellular IoT systems and is especially suited for asset tracking applications and environmental monitoring.

Nordic Thingy:91 includes sensors that gather data about its own movements and the surrounding environment. Temperature, humidity, air quality, air pressure, color, and light data can easily be extracted for local or remote analysis. For input, the Nordic Thingy:91 offers a user-programmable button. Visual output is achieved with RGB indicator LEDs, while a buzzer can provide audible output. The standard application firmware on Nordic Thingy:91 extracts the data from the different sensors and relays it securely to the nRF Connect for Cloud, where it is displayed in a user-friendly interface.

The firmware supports concurrent operation with LTE Link Monitor, a tool providing an AT command interface, enabling link and network testing. LTE Link Monitor is an application, which is implemented as part of the nRF Connect for Desktop application. The firmware has been developed using the nRF Connect *Software Development Kit (SDK)*. It is open source and can be leveraged and modified to suit your specific needs. The firmware can be updated and debugged by using an external programmer/debug probe, for example nRF9160 *DK (Development Kit)* or J-Link device supporting Arm Cortex-M33.

Nordic Thingy:91 integrates the nRF9160 *System in Package (SiP)*[1], supporting LTE-M, NB-IoT and *Global Positioning System (GPS)*, and the nRF52840 *System on Chip (SoC)*, supporting *Bluetooth*® Low Energy and *Near Field Communication (NFC)* passive tag.

> **Note:** LTE-M or LTE NB-IoT can operate simultaneously with Bluetooth LE.

Source code for firmware, hardware layout, and schematics are all available on our web site www.nordicsemi.com.

Nordic Thingy:91 has an antenna supporting *GPS*, LTE-M, and NB-IoT that enables it to support a global range of LTE bands. It has two antennas connected to the nRF52840: a 2.4 GHz antenna for Bluetooth LE and an *NFC* passive tag antenna. To connect to cellular network out of the box, Nordic Thingy:91 has a nano/4FF SIM card slot and is bundled with a SIM card from iBasis that comes preloaded with 10 MB.

NFC in Nordic Thingy:91 operates as a passive tag (e.g. it does not feature a reader function). Nordic Thingy:91 may use this tag function for the Out of Band pairing feature as described in the Bluetooth Core Specification.

A 1400 mAh rechargeable Li-Po battery is also part of this prototyping platform giving a smooth transition into prototype field-testing.

Key features of Nordic Thingy:91

- 700-960 MHz + 1710-2200 MHz LTE band support[2]. The following bands, based on geographic regions, are used:
  - USA – 2, 4, 12, and 13
  - EU – 3, 8, 20, and 28
- Certifications: CE, FCC
- LTE-M/NB-IoT/*GPS*, Bluetooth LE and *NFC* passive tag antennas

[1] The nRF9160 *SiP* is certified for USA bands 2, 4, 5, 12, 13, 14, 17, 25, 26, and 66. However, Nordic Thingy:91 operates and is only certified for USA bands 2, 4, 12, and 13. The Nordic Thingy:91 firmware is written and documented to only use this subset of USA bands.

[2] The application currently enables the following frequency bands: 2, 3, 4, 8, 12, 13, 20, and 28.

- Nano/4FF *Subscriber Identity Module (SIM)* card slot
- User-programmable button and RGB LEDs
- Environmental sensor for temperature, humidity, air quality, and air pressure
- Color and light sensor
- Low-power and high-G accelerometer
- Buzzer
- 4 x N-MOS transistor for external DC motors or LEDs
- Rechargeable Li-Po battery with 1400 mAh capacity
- Charging through *Universal Serial Bus (USB)*
- PC connection through *USB*
- Normal operating temperature range: $5^{o}C$ ~ $35^{o}C$

nRF9160

- Multimode LTE-M/NB-IoT modem
  - GCF certified for global operation
  - 23 dBm output power
  - *GPS*
  - Power saving features: DRX, eDRX, PSM
  - Coverage enhancement modes
  - Single pin 50 Ω antenna interface
  - *Universal Integrated Circuit Card (UICC)* interface
- Application processor
  - 64 MHz Arm® Cortex®-M33 CPU
  - Arm TrustZone® for trusted execution
  - Arm CryptoCell 310 for application layer security
  - 1 MB flash and 256 kB RAM
  - 4 x SPI/UART/TWI, PDM, I2S, PWM, ADC

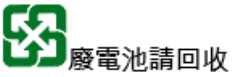
nRF52840 WLCSP

- Bluetooth LE and *NFC* passive tag support
- 64 MHz Arm Cortex-M4F CPU
- 1 MB flash and 256 kB RAM
- *USB*

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 battery in this product cannot be easily replaced by users themselves. Batteries should be removed only by qualified professionals due to safety concerns.

2 Kit content

The Nordic Thingy:91 kit consists of hardware and access to software components, hardware design files, applications, and documentation.

Figure 1: Nordic Thingy:91 hardware content

The Nordic Thingy:91 kit contains the following:

- Nordic Thingy:91 device with a rubber enclosure serving as a protective cover
- An eSIM (SIM card) from iBASIS supported by the nano/4FF SIM card slot of Nordic Thingy:91
- An information leaflet

WARNING - Power adapter is not included in the kit.[3]

2.1 Downloadable content

The Nordic Thingy:91 prototyping platform includes firmware source code, documentation, hardware schematics, and layout files.

Firmware

- Application firmware for Nordic Thingy:91

[3] Power supply adapter is not included in the safety certification test report, see separate test report according to IEC 62368. The power supply adapter you will use shall meet PS1 requirements.

  - Precompiled HEX files
  - nRF52840 *USB* to UART bridge
- nRF9160 modem firmware
- nRF Connect SDK

PC tools

- nRF Connect LTE Link Monitor
- Segger Embedded Studio
- nRF Connect Programmer

Web applications

- nRF Connect for Cloud

Hardware files

The hardware files can be downloaded from the Nordic Thingy:91 product page.

The zip file and its subdirectories contain the hardware design files for the Nordic Thingy:91. The hardware files for the circuit board are available in the following folder in the hardware files zip package:

```
\Thingy91 - Hardware files x_x_x\PCA20035-Thingy91 Board x_x_x
```

In this folder, you can find the following hardware design files:

- Altium Designer files
- Schematics and PCB layout files in PDF format
- Bill of materials
- Production files:
  - Drill files
  - Assembly drawings
  - Gerber files
  - Pick-and-place files

2.2 Related documentation

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

- nRF9160 Product Specification
- nRF52840 Product Specification
- nRF9160 DK
- nRF52840 Development Kit
- nRF9160 Errata
- nRF52840 Errata
- nRF Connect SDK documentation
- nRF Connect for Cloud
- nRF Connect LTE Link Monitor
- nRF Connect Programmer
- nRF91 AT Commands Reference Guide

3 Getting started

Setting up Nordic Thingy:91 requires completing the following steps.

Before you start:

- Unpack Nordic Thingy:91.
- Make sure to update the Nordic Thingy:91 firmware as explained in the section Firmware update on page 15.
- Make sure you have an nRF Connect for Cloud account and sign in to nRF Connect for Cloud.

Once you are signed in, perform the following steps.

1. Add new LTE device.

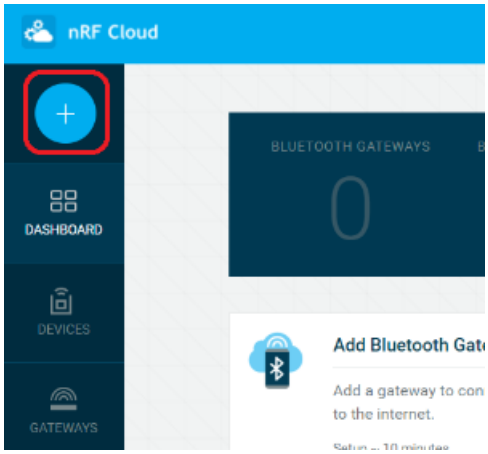


Figure 2: Adding new LTE device in nRF Cloud

2. Verify and activate the *SIM* card.

   Input *SIM* ICCID and *Personal Unblocking Key (PUK)* from the *SIM* card, and add personal information.

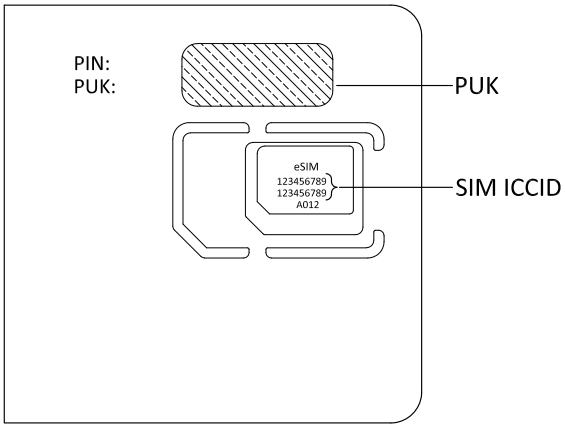


Figure 3: Placement of PUK and SIM ICCID on the SIM card

3. Gently remove the rubber overlay to reveal the power switch and the top of Nordic Thingy:91. Insert the *SIM* card.

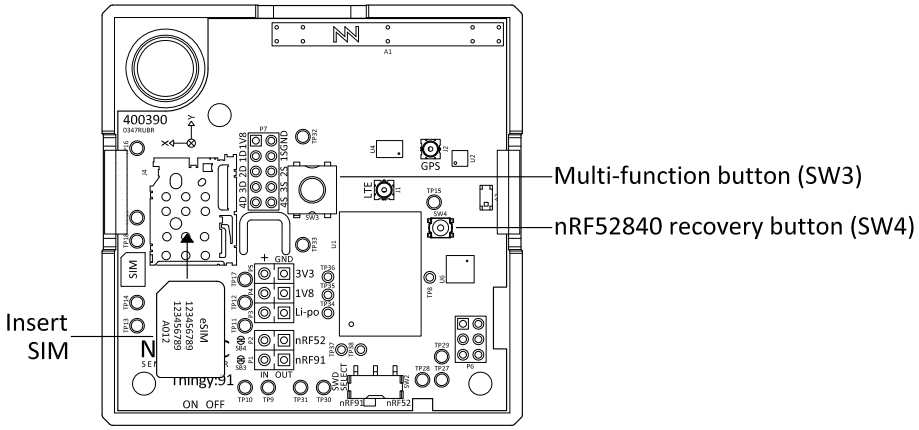


Figure 4: Inserting the SIM card

4. Once the *SIM* card is inserted, power on Nordic Thingy:91. You will find the power switch next to the micro-*USB* port.

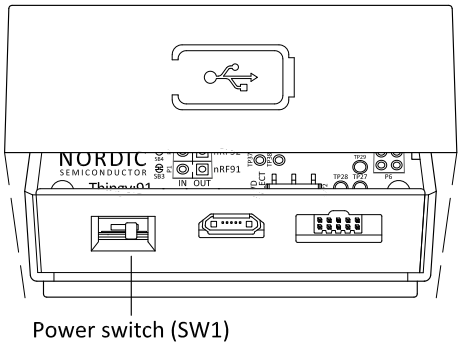


Figure 5: Nordic Thingy:91 Power switch

Wait for yellow breathing in the indicator LED. If the LED does not turn yellow, check Table 2: Nordic Thingy:91 Operating modes and LED indications on page 14 to determine the present state of Nordic Thingy:91.

5. Associate the Nordic Thingy:91 to your user account.

   Enter the IMEI and PIN information that is found on the sticker on the Nordic Thingy:91 PCB. Wait for blue breathing which means that your Nordic Thingy:91 is associated and connected.

Figure 6: Associating Nordic Thingy:91 to nRF Cloud

6. For optional activation of *GPS*, go outdoors and press the **SW3** button for a minimum of 10 seconds until the indicator LED begins breathing purple.

   - Purple breathing: *GPS* is active and searching
   - Green breathing: *GPS* has fix

   Wait for green breathing in the indicator LED which indicates that your Nordic Thingy:91 has *GPS* fix.

   Check that the position data and the environment sensor data are sent to nRF Connect for Cloud.

3.1 Connecting LTE Link Monitor

To get debug output and to send AT commands directly to the modem, you can connect to the Nordic Thingy:91 using nRF Connect LTE Link Monitor.

Before you begin, install and open nRF Connect for Desktop. To download the latest version, go to nRF Connect for Desktop. For instructions, see nRF Connect LTE Link Monitor.

To connect to the Nordic Thingy:91 using LTE Link Monitor, perform the following steps.

1. Install and launch LTE Link Monitor.
2. Connect your Nordic Thingy:91 to your computer with a *USB* cable.
3. Make sure that Nordic Thingy:91 is powered on.
4. Click **Select device** and select the device entry from the drop-down list in the LTE Link Monitor.
5. Verify the connection to the Nordic Thingy:91 modem by sending the command `AT` to the modem from the LTE Link Monitor terminal and observing that the modem responds with `OK`.

All asset tracker debug output shows up in the terminal view, and you can send AT commands to the modem to try out different settings. For further details on available AT commands, see nRF91 AT Commands Reference Guide.

4 Firmware

The firmware of Nordic Thingy:91 has been developed using the nRF Connect SDK. It is open source, and can be modified according to specific needs.

The asset tracker application firmware, which is pre-loaded in the Nordic Thingy:91, enables the device to use the environment sensors as described in Environment sensors on page 33, and provides an option of tracking the device using GPS. The data, along with information about the device, is transmitted to Nordic Semiconductor's cloud solution, nRF Cloud, where it can be visualized. For more information on the asset tracker application, see nRF9160 Asset Tracker.

4.1 Getting the firmware

Firmware on the Nordic Thingy:91 can be updated after obtaining the corresponding firmware images.

The different ways to obtain the firmware images are:

- Download pre-compiled firmware images
- Compile the source code

4.1.1 Download pre-compiled firmware images

You can update the firmware on Nordic Thingy:91 by direct download of the compiled versions of firmware images.

To obtain pre-compiled firmware images, perform the following steps:

1. Navigate to **Firmware for the Thingy:91** under the **Downloads** tab on the Nordic Thingy:91 product page.
2. Download the latest Nordic Thingy:91 firmware package.
3. Unzip the downloaded firmware package zip file.
4. Check the `CONTENTS.txt` file in the extracted folder for the location and names of the different firmware images.

4.1.2 Compile the source code

Another way to update the firmware on a Nordic Thingy:91 is by using the firmware images obtained by compiling the code in an nRF Connect SDK environment.

Nordic Thingy:91 ships with the nRF9160 Asset Tracker application which runs on the nRF9160 *SiP* and the USB-UART bridge sample which runs on the nRF52840 *SoC*.

Follow nRF Connect *SDK* Getting Started guide to set up your system to be able to compile a compatible firmware image. There are two compile targets of interest for Nordic Thingy:91 in nRF Connect *SDK*, namely:

- `nrf9160_pca20035ns`
- `nrf52840_pca20035`

The compile target `nrf9160_pca20035ns` should be used when compiling the application code for the nRF9160 *SiP* and the compile target `nrf52840_pca20035` should be used when compiling the application code for the onboard nRF52840 *SoC*.

To compile the source code, perform the following steps:

- Go to the desired application folder. For example, the folder path is `ncs/nrf/applications/asset_tracker` when compiling the source code for the asset tracker application on the `nrf9160_pca20035ns` target and `ncs/nrf/samples/usb/usb_uart_bridge` when compiling the source code for the USB-UART bridge sample on the `nrf52840_pca20035` target.
- Make sure that you have the latest version of nRF Connect *SDK* repository by pulling the nRF Connect SDK repository on GitHub repository using the command:

```
git pull
```

- Run the `west` command to get the rest of the dependencies:

```
west update
```

- Compile the code using `west` command as follows:
  - For the asset tracker on nRF9160:

```
west build -b nrf9160_pca20035ns
```

  - For the USB-UART bridge on nRF52840:

```
west build -b nrf52840_pca20035
```

The various firmware image files that can be used for the application firmware update can be located here: `build/zephyr/`.

File	File format and update scenario
merged.hex	Full image, HEX format. Used in the application firmware update using debug probe.
app_signed.hex	*MCUboot* compatible image, HEX format. Used in the application firmware update using nRF Connect Programmer.
app_update.bin	*MCUboot* compatible image, binary format. Used in the application firmware update using the `mcumgr` command line tool and for FOTA updates.

Table 1: Firmware image files

4.2 Operating modes

Nordic Thingy:91 contains RGB indicator LEDs which indicate the operating state of the device.

The following table shows the various operating modes and the LED indications in Nordic Thingy:91.

#	Mode		LED
1	Connecting		White breathing
2	Connected	Waiting for user association	Yellow breathing
3		Environment data only	Blue breathing
4		*GPS* Searching	Purple breathing
5		*GPS* fix	Green breathing
6	Low Battery		Red slow breathing
7	Error		Red fast breathing

Table 2: Nordic Thingy:91 Operating modes and LED indications

4.3 GPS

Nordic Thingy:91 has *GPS* which, if activated, allows the device to be located globally using GPS signals.

- To activate *GPS* long press button **SW3** (> 10 seconds) until the LED starts breathing purple. The LED starts breathing green when a GPS fix is achieved, and it starts breathing purple again when GPS is active and searching.
- To turn off *GPS* long press button **SW3** (> 10 seconds) until the LED starts breathing blue.

4.4 LTE Band Lock

The modem within Nordic Thingy:91 can be configured to use specific LTE bands by using the band lock AT command.

The pre-flashed firmware configures the modem to use the bands currently certified on the Nordic Thingy:91 hardware. The bands to enable can be configured when compiling the firmware. For more information see:

- Band lock AT command
- nRF Connect SDK Band Lock

4.5 LTE-M / NB-IoT switching

Nordic Thingy:91 has a multimode modem which enables it to support automatic switching between LTE-M and NB-IoT.

A built-in parameter in the Nordic Thingy:91 firmware determines whether the modem first attempts to connect in LTE-M or NB-IoT mode. If the modem fails to connect using this preferred mode within the default timeout period (10 minutes), the modem switches to the other mode.

5 Firmware update

You can update the modem firmware and the application firmware on Nordic Thingy:91 using multiple methods.

Modem firmware update can be done using an external debug probe such as nRF9160 DK. Application firmware update is supported through USB leveraging the built-in serial recovery mode of Nordic Thingy:91. The device also supports application firmware update using an external debug probe.

5.1 Updating modem firmware

You can update the modem firmware on Nordic Thingy:91 by using an external debug probe such as nRF9160 DK or J-Link device supporting Arm Cortex-M33.

To update the modem firmware, complete the steps in this section.

1. Download the latest modem firmware zip file.

   You can find it under the **Compatible Downloads** tab that is halfway down the nRF9160 product page.
2. Connect the Nordic Thingy:91 to the debug out port on a 10-pin external debug probe, for example, nRF9160 *DK*, using a 10-pin JTAG cable as shown in the following figure.

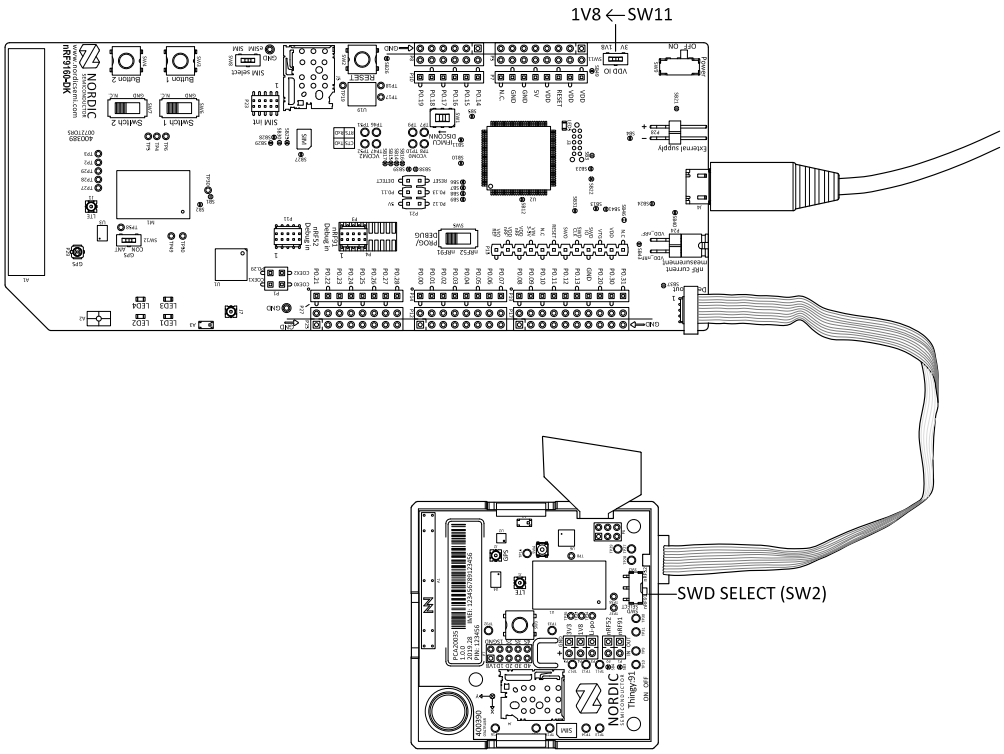


Figure 7: Connecting Nordic Thingy:91 to the external debug probe

> **Note:** If using nRF9160 *DK* as the debug probe, make sure VDD_IO (**SW11**) is set to 1.8 V.

3. Power on both the devices.
4. Start nRF Connect Programmer.
5. Connect the external debug probe to the PC with a *USB* cable.

6. Click **Select device** and select the appropriate debug probe entry from the drop-down list in the nRF Connect Programmer.
7. Click **Update Modem** in the **Cellular Modem** pane on the right and choose the zip file with the latest modem release.
8. Click **Write** in the **Device** pane on the right and wait for the update to complete.

> **Note:** If you have issues updating modem firmware, then do **Erase All** before trying to update the modem again.

5.2 Updating application firmware

Application firmware update on a Nordic Thingy:91 can be done either on the main nRF9160 *SiP* component or on the nRF52840 *SoC* component.

The selection of the target to be updated is based on the switch and the buttons in Nordic Thingy:91 which form part of its hardware technology.

5.2.1 Updating application firmware on nRF9160

Application firmware update on the nRF9160 *SiP* in Nordic Thingy:91 can be done in multiple ways.

- Using *USB* (*MCUboot*)
- Using an external debug probe

5.2.1.1 Updating application firmware on nRF9160 using USB (MCUboot)

Firmware updates can be done directly on the nRF9160 *SiP* of Nordic Thingy:91 by putting the device into serial recovery mode.

To update the application firmware on nRF9160 using *USB* (*MCUboot*) and the built-in serial recovery mode, complete the steps in this section.

1. Connect Nordic Thingy:91 to the PC using *USB*.
2. Put the Nordic Thingy:91 into application serial recovery mode by pushing the **SW3** button while powering on.
3. Start nRF Connect Programmer.
4. Click **Select device** and select the appropriate device entry from the drop-down list in the nRF Connect Programmer as shown in the following figure.

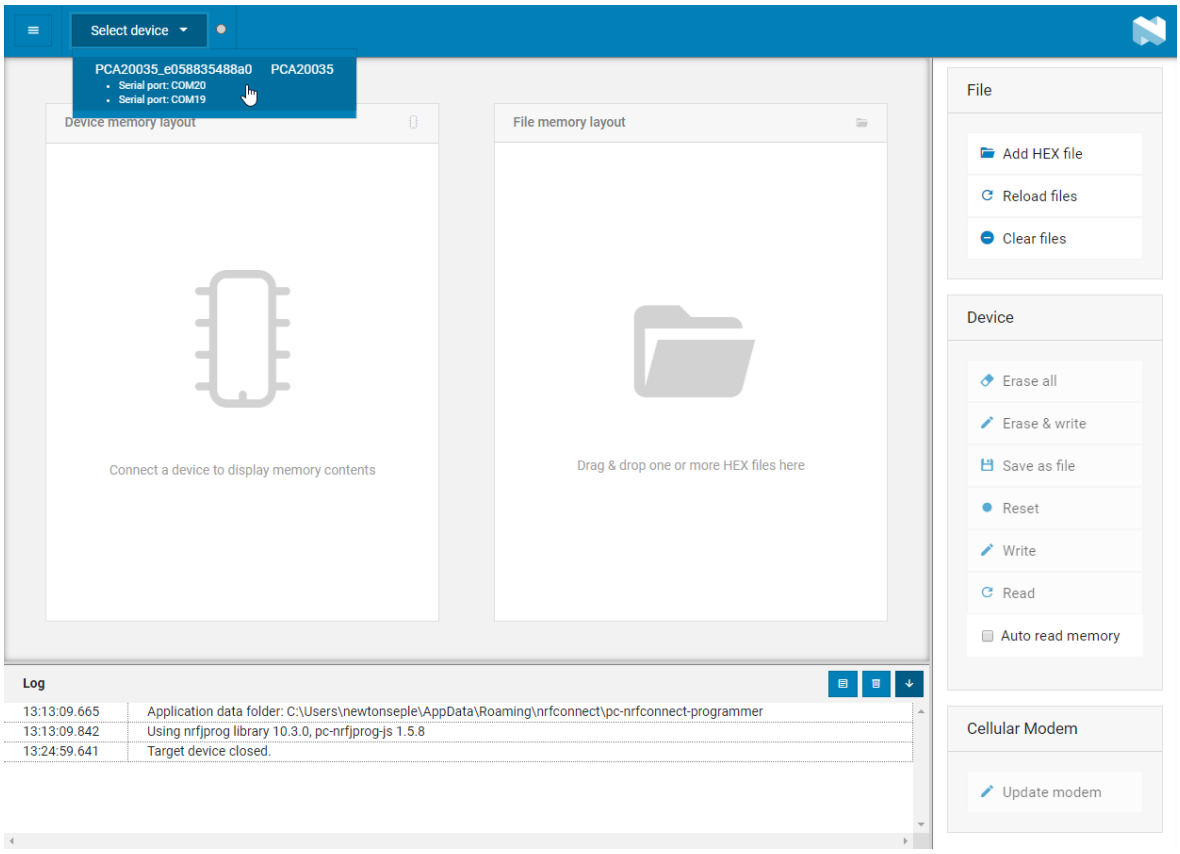


Figure 8: Connecting to Nordic Thingy:91 using nRF Connect Programmer

5. Click **Add HEX file** on the right pane (named as **File**).

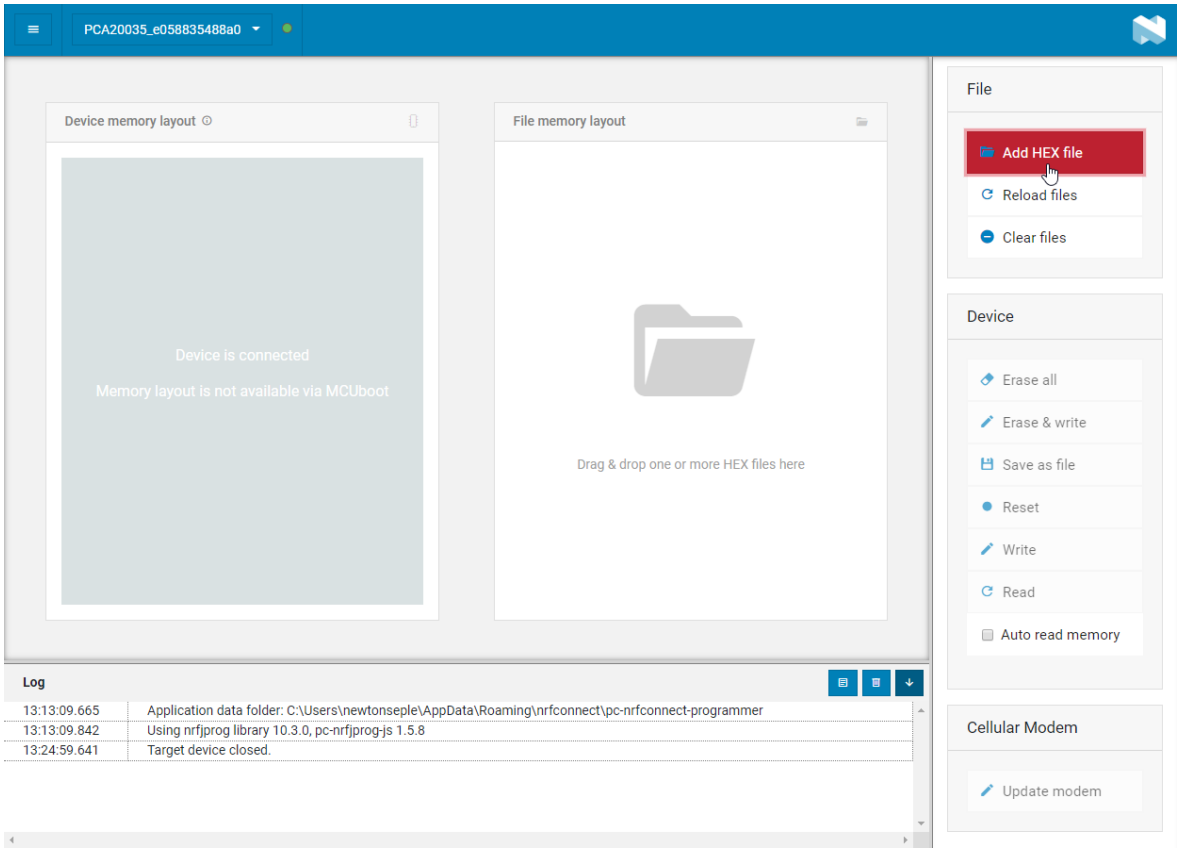


Figure 9: Selecting the firmware image file using Add HEX file option in the nRF Connect Programmer

6. Click **Browse** from the drop-down list that appears as a result of the previous step in the nRF Connect Programmer as shown in the following figure.

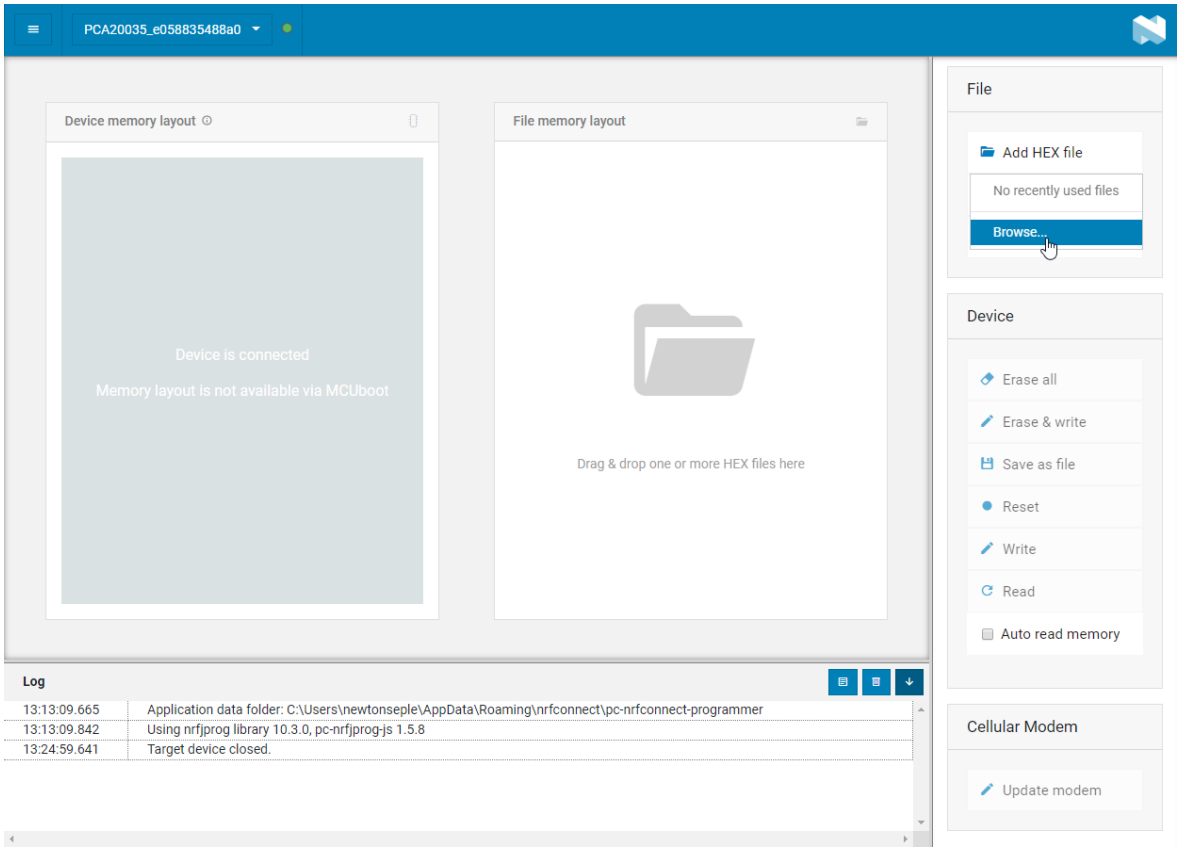


Figure 10: Selecting the firmware image file using Add HEX file (Browse) option in the nRF Connect Programmer

7. Select the *MCUboot* compatible HEX file from the file browser that opens up.
8. Click **Write** on the **Device** pane in the right as shown in the following figure.

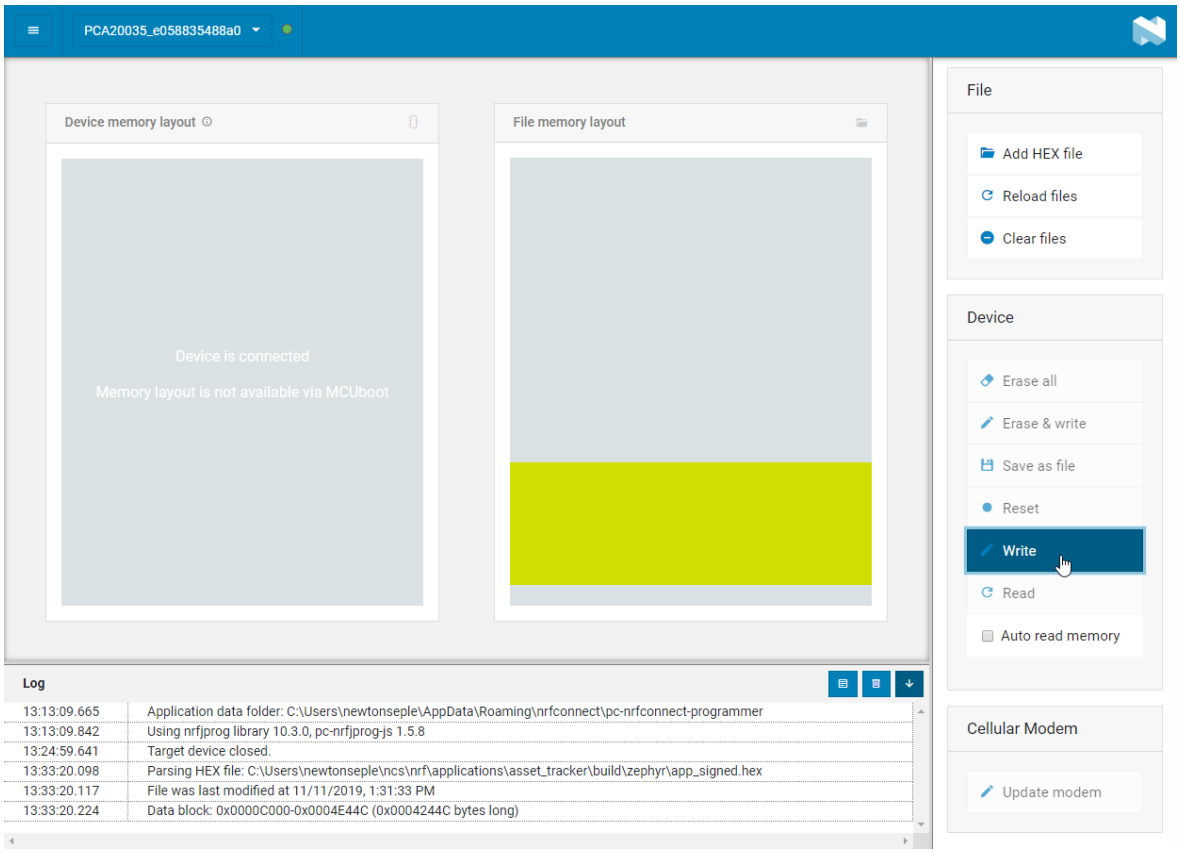


Figure 11: Updating the application firmware using Write option in nRF Connect Programmer

The **MCUboot DFU** dialogue box appears as shown in the following figure.

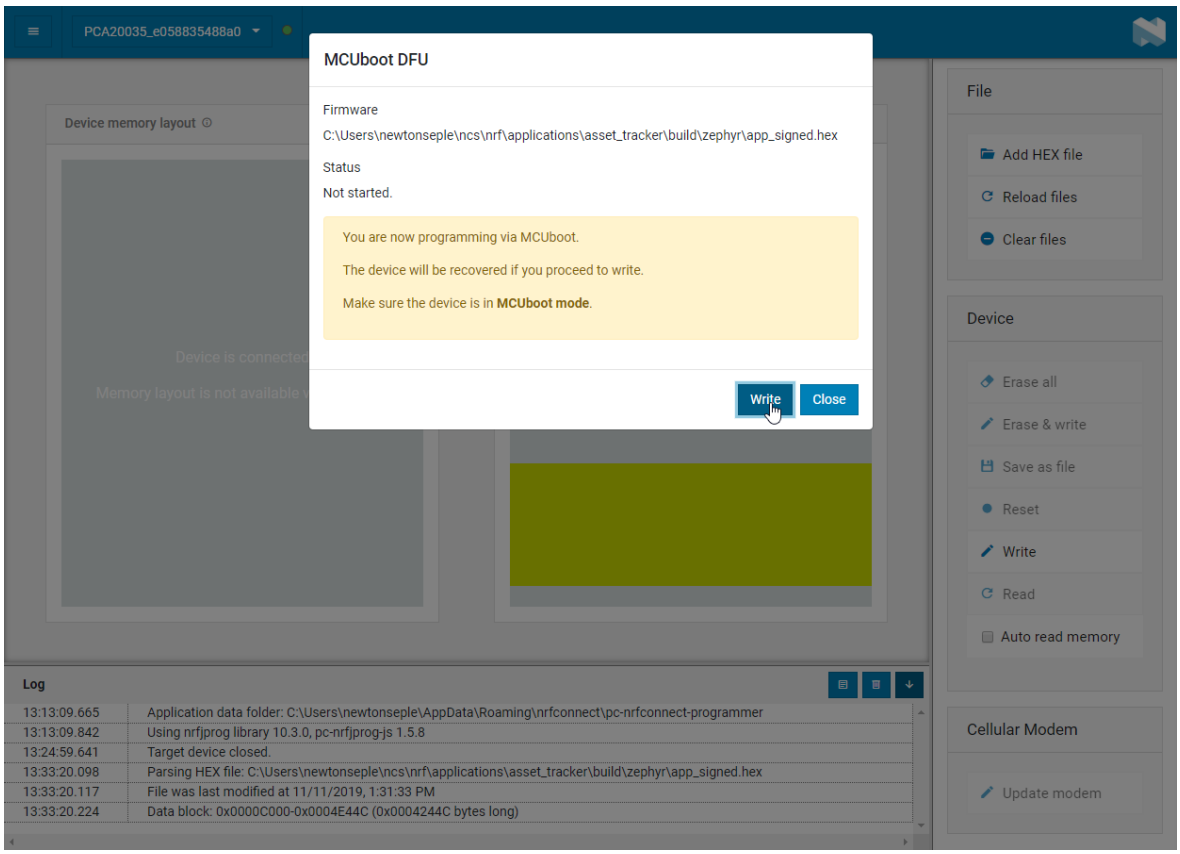


Figure 12: MCUboot DFU dialogue box

9. Click **Write** in the **MCUboot DFU** dialogue box and wait for the update to finish.

5.2.1.2 Updating application firmware on nRF9160 using an external debug probe

Firmware updates on the nRF9160 *SiP* of Nordic Thingy:91 can be performed by using an externally connected debug probe.

To update the application firmware on nRF9160 using an external debug probe, complete the steps in this section.

1. Set the Nordic Thingy:91 SWD selection switch (**SW2**) to nRF91. Refer Figure 35: SWD SELECT switch on page 37.
2. Connect the Nordic Thingy:91 to the debug out port on a 10-pin external debug probe, for example, nRF9160 *DK*, using a 10-pin JTAG cable. Refer Figure 7: Connecting Nordic Thingy:91 to the external debug probe on page 15.

   Note: If using nRF9160 *DK* as the debug probe, make sure that VDD_IO (**SW11**) is set to 1.8 V.
3. Make sure that the Nordic Thingy:91 and the external debug probe are powered on.
4. Update the application firmware using the nRF Connect Programmer by performing the following steps:
   a. Start nRF Connect Programmer.
   b. Connect the external debug probe to the PC with a *USB* cable.
   c. Click **Select device** and select the appropriate entry for the external debug probe from the drop-down list in the nRF Connect Programmer.
   d. Click **Add HEX file** on the right pane (named as **File**) and click **Browse** from the drop-down list that appears in the nRF Connect Programmer.
   e. Select the firmware image file in the file browser.
   f. Click **Erase & write** on the **Device** pane on the right and wait for the update to finish.

5.2.2 Updating application firmware on nRF52840

Application firmware update on the nRF52840 *SoC* in Nordic Thingy:91 can be done in multiple ways.

- Using *USB* (*MCUboot*)
- Using an external debug probe

5.2.2.1 Updating application firmware on nRF52840 using USB (MCUboot)

Firmware updates can be done directly on the nRF52840 *SoC* of Nordic Thingy:91 by putting the device into serial recovery mode.

To update the application firmware on nRF52840 using *USB* (*MCUboot*) and the built-in serial recovery mode, complete the steps in this section.

1. Connect Nordic Thingy:91 to the PC with a *USB* cable.
2. Put Nordic Thingy:91 into application serial recovery mode by pushing the **SW4** button while powering on.
3. Start nRF Connect Programmer.
4. Click **Select device** and select the appropriate device entry from the drop-down list in the nRF Connect Programmer as shown in Figure 8: Connecting to Nordic Thingy:91 using nRF Connect Programmer on page 17.
5. Click **Add HEX file** on the right pane (named as **File**) as shown in Figure 9: Selecting the firmware image file using Add HEX file option in the nRF Connect Programmer on page 17.

6. Click **Browse** from the drop-down list that appears as a result of the previous step in the nRF Connect Programmer as shown in Figure 10: Selecting the firmware image file using Add HEX file (Browse) option in the nRF Connect Programmer on page 18.
7. Select the *MCUboot* compatible HEX file from the file browser that opens up.
8. Click **Write** on the **Device** pane in the right as shown in Figure 11: Updating the application firmware using Write option in nRF Connect Programmer on page 19.
   The **MCUboot DFU** dialogue box appears as shown in Figure 12: MCUboot DFU dialogue box on page 19.
9. Click **Write** in the **MCUboot DFU** dialogue box and wait for the update to finish.

5.2.2.2 Updating application firmware on nRF52840 using an external debug probe

Firmware updates on the nRF52840 *SoC* of Nordic Thingy:91 can be performed by using an externally connected debug probe.

To update the application firmware on nRF52840 using an external debug probe, complete the steps in this section.

1. Set the Nordic Thingy:91 SWD selection switch (**SW2**) to nRF52. Refer Figure 35: SWD SELECT switch on page 37.
2. Connect the Nordic Thingy:91 to the debug out port on a 10-pin external debug probe, for example, nRF9160 *DK*. Refer Figure 7: Connecting Nordic Thingy:91 to the external debug probe on page 15.

   Note: If using nRF9160 *DK* as the debug probe, make sure that VDD_IO (**SW11**) is set to 1.8 V.
3. Make sure that the Nordic Thingy:91 and the external debug probe are powered on.
4. Update the application firmware using the nRF Connect Programmer by performing the following steps:
   a. Start nRF Connect Programmer.
   b. Connect the external debug probe to the PC with a *USB* cable.
   c. Click **Select device** and select the appropriate entry for the external debug probe from the drop-down list in the nRF Connect Programmer.
   d. Click **Add HEX file** on the right pane (named as **File**) and click **Browse** from the drop-down list that appears in the nRF Connect Programmer.
   e. Select the firmware image file in the file browser.
   f. Click **Erase & write** on the **Device** pane on the right and wait for the update to finish.

6 Hardware description

This chapter focuses on the hardware components of Nordic Thingy:91 with detailed descriptions of the various hardware blocks that are present on the device.

The sensors available in Nordic Thingy:91 are not calibrated in production. Nordic Semiconductor does not specify the accuracy of measurements. Users who want to reuse parts of this design to create measurement devices should conform to documentation of the specific sensors.

6.1 Block diagram

The block diagram represents interactions between hardware components on Nordic Thingy:91.

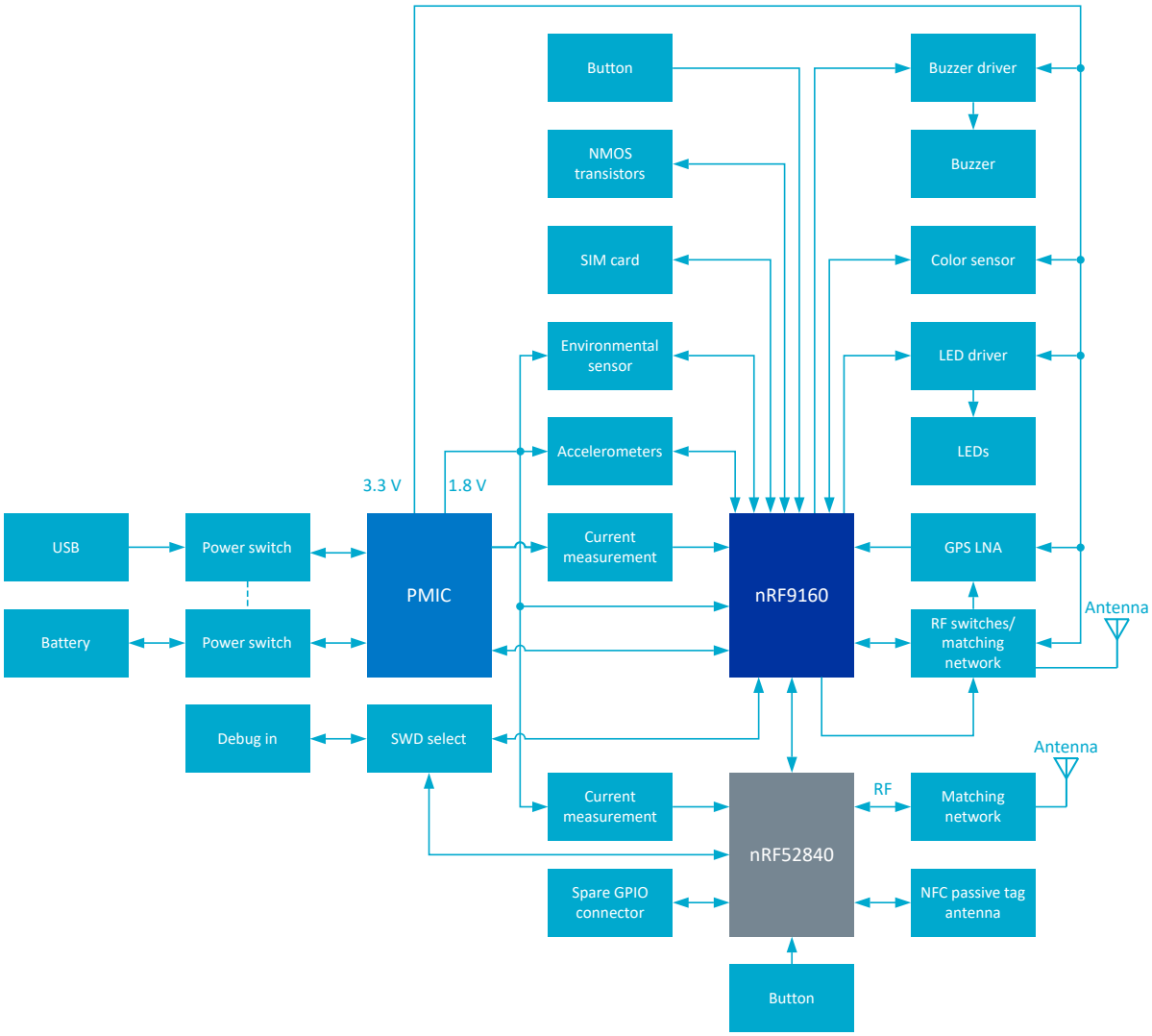


Figure 13: Nordic Thingy:91 hardware block diagram

6.2 Hardware figures

The hardware figures show elements on both sides of the Nordic Thingy:91 PCB.

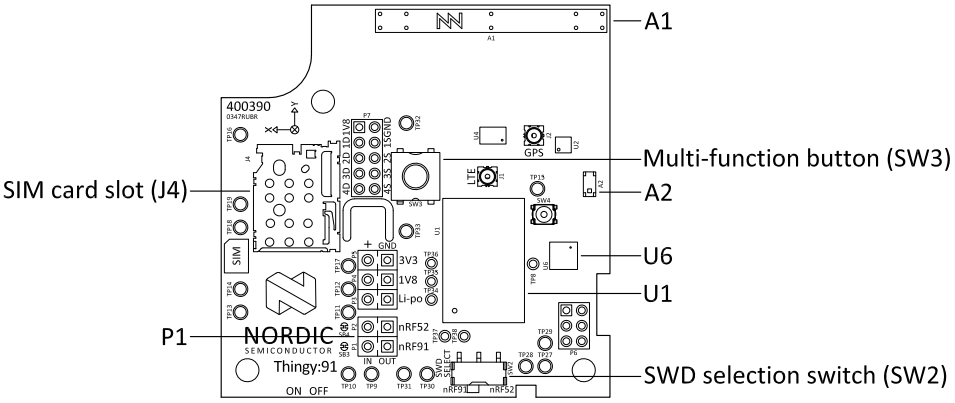


Figure 14: Nordic Thingy:91 PCB, top

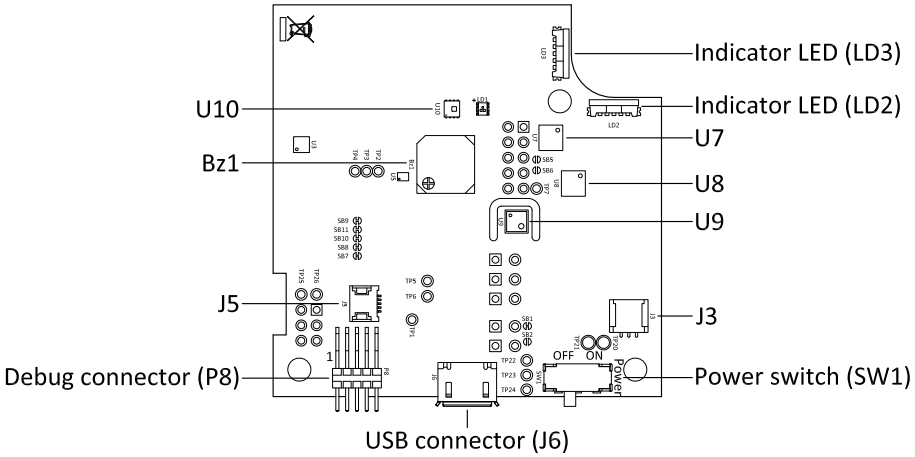


Figure 15: Nordic Thingy:91 PCB, bottom

6.3 nRF9160

The nRF9160 is the main device of Nordic Thingy:91. It is a compact, highly integrated *SiP* that makes use of the latest low-power LTE technology. It has advanced processing capabilities and security features. It also has the accessibility and flexibility to be used with a wide range of single-device low-power cellular IoT applications.

For more information, see nRF9160 Product Specification.

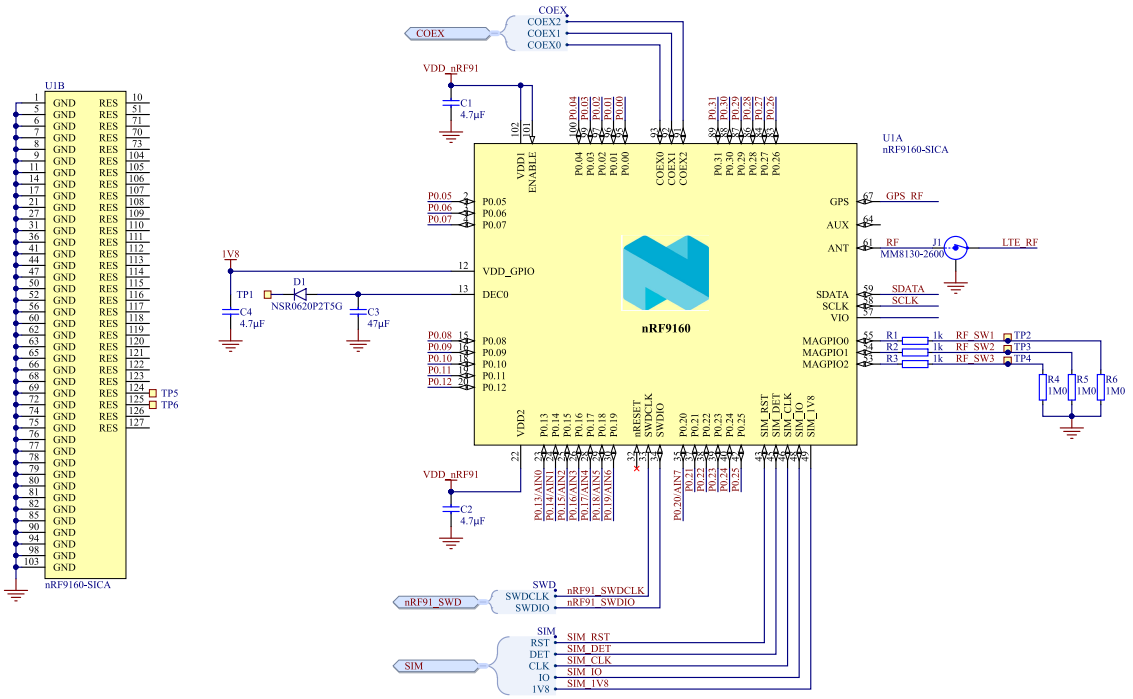


Figure 16: nRF9160 schematic

6.3.1 Antenna tuning

To improve antenna efficiency, Nordic Thingy:91 has dynamic antenna tuning.

Different tuning components are used for different frequencies. This is achieved by using tuning components between two SP8T RF switches. The switches are automatically controlled by the nRF9160 LTE modem and set to the correct state based on the frequency of operation. Six paths are used for LTE frequency, and one path is used for *GPS* frequency.

RF_SW3	RF_SW2	RF_SW1	State	Band	Frequency
0	0	0	RF2 - RFC	Not used	Not used
0	0	1	RF7 - RFC	13U/D, 28D	746 MHz - 803 MHz
0	1	0	RF5 - RFC	12U/D, 17U/D, 28U 1U/D, 2U/D, 3U/D, 4U/D, 25U/D	698 MHz - 748 MHz 1710 MHz - 2200 MHz
0	1	1	RF3 - RFC	5D, 20U, 26D	824 MHz - 894 MHz
1	0	0	RF1 - RFC	8U/D	880 MHz - 960 MHz
1	0	1	RF8 - RFC	5U, 20D, 26U	791 MHz - 849 MHz
1	1	0	RF6 - RFC	Not used	Not used
1	1	1	RF4 - RFC	GPS	1574 MHz - 1577 MHz

Table 3: Antenna tuning bands

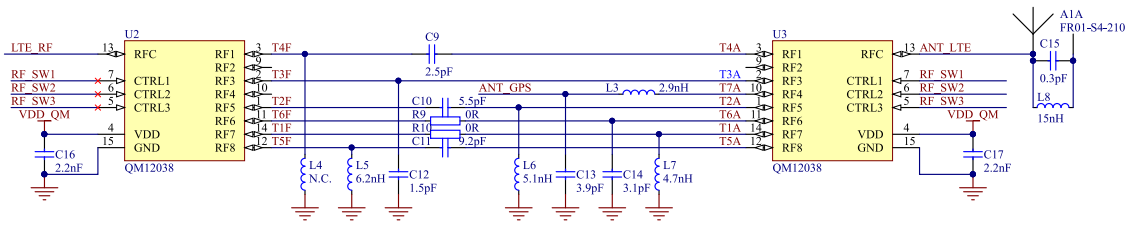


Figure 17: Antenna tuning circuitry schematic

6.3.2 RF measurements

The LTE signals are propagated through a coaxial connector. This makes it possible to perform conducted measurements or attach external antennas.

By default, when no cable is attached, the RF signal is routed to the onboard antenna. 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.

The connector is of SWF type (Murata part no. MM8130-2600) with an internal switch. 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. The insertion loss in the adapter cable is approximately 0.5–1 dB.

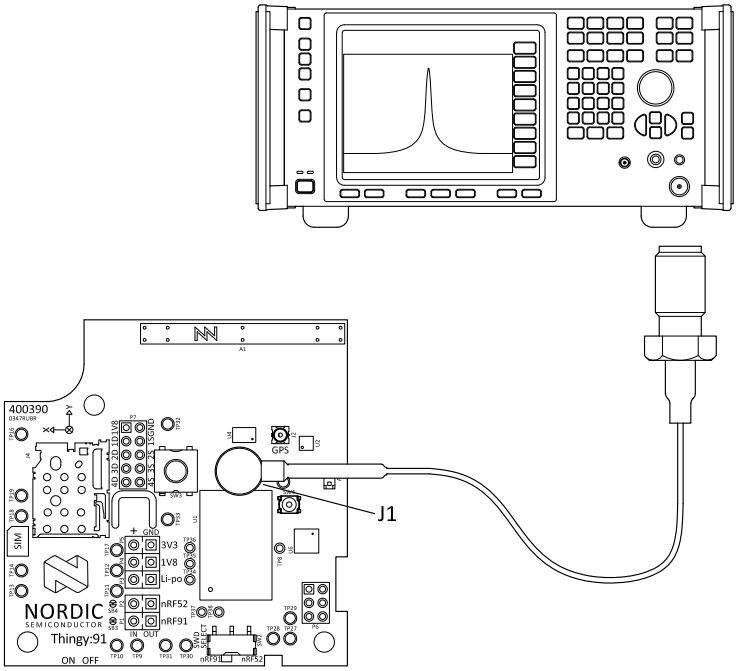


Figure 18: Connecting a spectrum analyzer

6.3.3 GPS

The nRF9160 has a dedicated *GPS* port to support global navigation, and the same antenna is used for both LTE and *GPS*. The *GPS* signal is RX only, and there is a low-noise amplifier (LNA) that amplifies the signal before it is fed to the *GPS* RF port on the nRF9160.

The *GPS* signals are propagated through a coaxial connector located between the antenna and the LNA. This makes it possible to attach external antennas. The connector is of SWF type (Murata part no. MM8130-2600) with an internal switch. 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. The insertion loss in the adapter cable is approximately 0.5–1 dB.

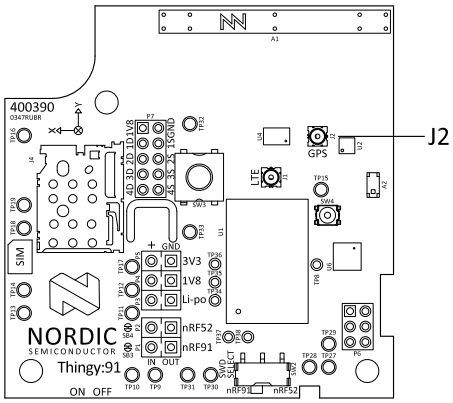


Figure 19: External GPS antenna connector

The LNA enable signal is controlled by the logic circuitry. It is enabled only when the antenna tuning circuitry is set to operate at the *GPS* frequency band. The LNA makes the *GPS* receiver more sensitive to *GPS* signals and less sensitive to interference from other sources nearby.

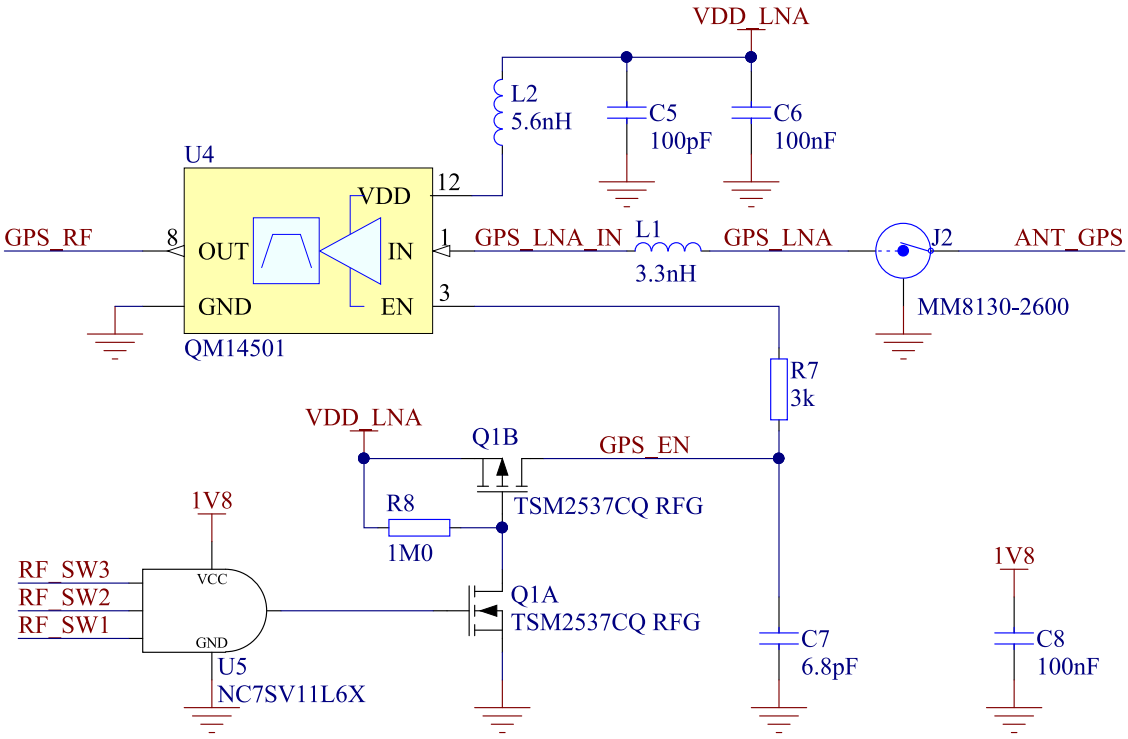


Figure 20: GPS circuit schematic

Note: *GPS* signals do not usually penetrate ceilings or other structures. For best *GPS* performance, Nordic Thingy:91 should be used outside in an open space, far from sources of interference and other structures that may block the signals.

6.3.4 SIM card

Nordic Thingy:91 is equipped with a nano-*SIM* (4FF) card slot.

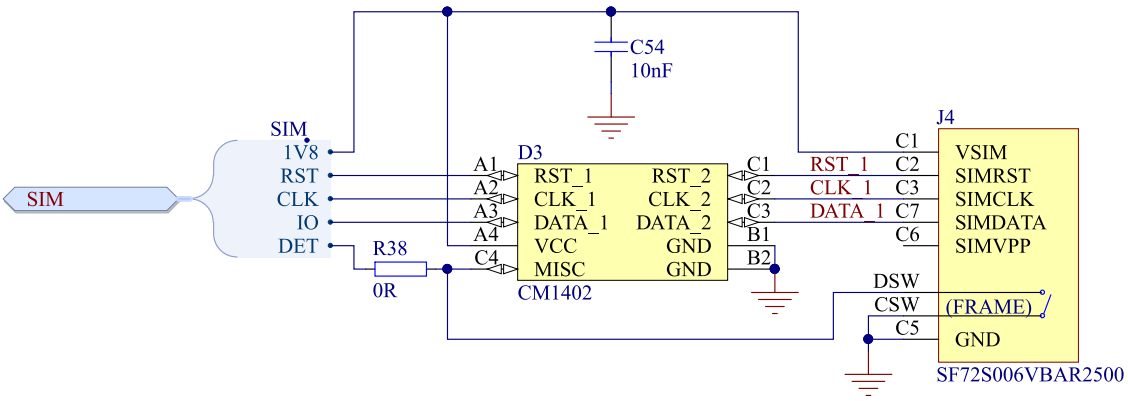


Figure 21: SIM card schematic

6.4 nRF52840

For *USB*, Bluetooth, and *NFC* passive tag connectivity, Nordic Thingy:91 uses a nRF52840 *SoC*. It is a powerful, highly flexible, ultra-low power *SoC* that incorporates a Bluetooth Low Energy radio and a 32-bit Arm Cortex-M4F CPU.

For more information on the *SoC*, see nRF52840 Product Specification.

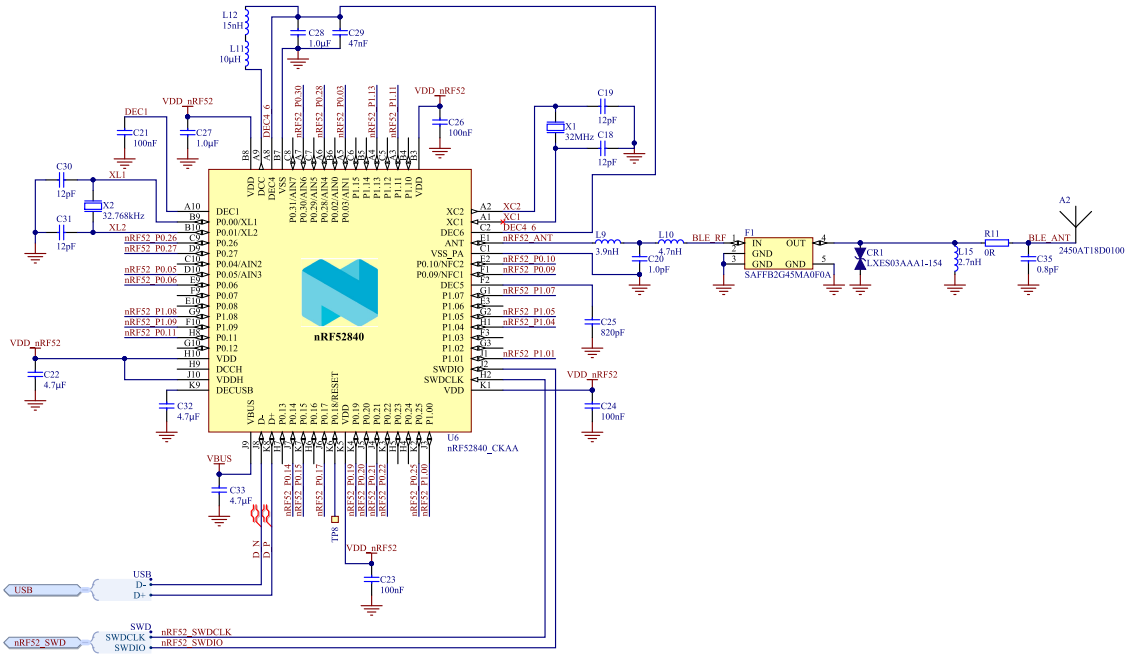


Figure 22: nRF52840 schematic

6.4.1 NFC passive tag

Nordic Thingy:91 supports an *NFC* passive tag. NFC-A listen mode operation is supported on the nRF52840.

The *NFC* passive tag antenna input is available on connector **J5**.

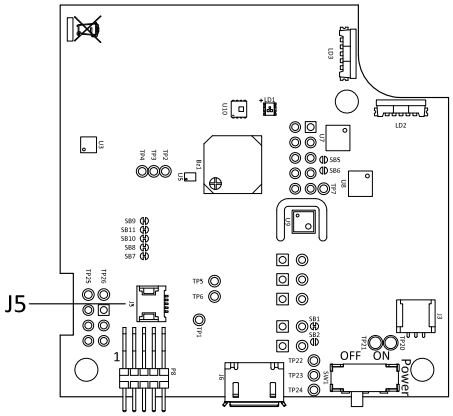


Figure 23: NFC passive tag antenna connector

The *NFC* passive tag uses two pins, **P0.02** (NFC1) and **P0.03** (NFC2), to connect the antenna. These pins are shared with *GPIO*s (**P0.09** and **P0.10**), 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* passive tag pins are enabled by default.

The *NFC* passive tag can be disabled and the *GPIO*s enabled by defining the *CONFIG_NFCT_PINS_AS_GPIOS* variable in the project settings. The way of doing this depends on the *Integrated Development Environment (IDE)* or toolchain in use.

- When using SEGGER Embedded Studio, go to **Project** > **Edit Options** > **Code** > **Preprocessor** > **Preprocessor Definitions** and add the *CONFIG_NFCT_PINS_AS_GPIOS* variable.
- When using Keil, go to **Project** > **Options for Target** > **C/C++** > **Preprocessor Symbols** > **Define** and add the *CONFIG_NFCT_PINS_AS_GPIOS* variable.

6.4.2 USB

The Nordic Thingy:91 *USB* connector is connected to the *USB* interface of the nRF52840 *SoC*. This enables PC communication and battery charging.

6.5 Pin maps

The pin assignments for the nRF9160 *SiP* and nRF52840 *SoC* are listed in the pin map tables.

I/O	Label	Description
P0.00	SENSE_LED_RED	Red color of the color sensor support LED
P0.01	SENSE_LED_GREEN	Green color of the color sensor support LED
P0.02	SENSE_LED_BLUE	Blue color of the color sensor support LED
P0.03	SCK	SPI clock line
P0.04	MOSI	SPI master output, slave input data line
P0.05	MISO	SPI master input, slave output data line
P0.06	ADXL372_INT1	High-G accelerometer interrupt line
P0.07	ADXL372_CS	High-G accelerometer chip select line

I/O	Label	Description
P0.08	ADXL362_CS	Low-power accelerometer chip select line
P0.09	ADXL362_INT1	Low-power accelerometer interrupt line
P0.10	ADXL362_INT2	Accelerometer interrupt line 2, selectable by solder bridge
P0.11	SDA	I^2C data line
P0.12	SCL	I^2C clock line
P0.13	N-MOS_1	Gate of N-MOS transistor externally available
P0.14	N-MOS_2	Gate of N-MOS transistor externally available
P0.15	N-MOS_3	Gate of N-MOS transistor externally available
P0.16	N-MOS_4	Gate of N-MOS transistor externally available
P0.17	ADP_INT	PMIC interrupt line
P0.18	MCU_IF0	nRF52840 interface
P0.19	MCU_IF1	nRF52840 interface
P0.20	MCU_IF2	nRF52840 interface
P0.21	MCU_IF3	nRF52840 interface
P0.22	MCU_IF4	nRF52840 interface
P0.23	MCU_IF5	nRF52840 interface
P0.24	MCU_IF6	nRF52840 interface
P0.25	MCU_IF7	nRF52840 interface
P0.26	BUTTON	Button input
P0.27	BH_INT	Color sensor interrupt line
P0.28	BUZZER	Buzzer PWM signal
P0.29	LIGHTWELL_RED	Red color of the lightwell LEDs
P0.30	LIGHTWELL_GREEN	Green color of the lightwell LEDs
P0.31	LIGHTWELL_BLUE	Blue color of the lightwell LEDs

Table 4: nRF9160 pin map

I/O	Label	Description
P0.00	XL1	Low frequency crystal
P0.01	XL2	Low frequency crystal
P0.02	N.A.	Not used
P0.03	SPARE7	Analog/digital *GPIO* externally available
P0.04	N.A.	Not used
P0.05	SPARE2	Analog/digital *GPIO* externally available
P0.06	SPARE1	Digital *GPIO* externally available
P0.07	N.A.	Not used
P0.08	N.A.	Not used
P0.09	NFC1	*NFC* passive tag antenna
P0.10	NFC2	*NFC* passive tag antenna
P0.11	MCU_IF0	nRF9160 interface
P0.12	N.A.	Not used
P0.13	N.A.	Not used
P0.14	IF_SWD_IO	nRF9160 SWD interface data line
P0.15	MCU_IF1	nRF9160 interface
P0.16	N.A.	Not used
P0.17	IF_SWD_CTRL	nRF9160 SWD interface control
P0.18	RESET	nRF52840 reset line, available on test point
P0.19	MCU_IF6	nRF9160 interface
P0.20	MCU_IF2	nRF9160 interface
P0.21	MCU_IF3	nRF9160 interface
P0.22	MCU_IF7	nRF9160 interface
P0.23	N.A.	Not used
P0.24	N.A.	Not used
P0.25	MCU_IF5	nRF9160 interface
P0.26	SPARE3	Digital *GPIO* externally available
P0.27	SPARE4	Digital *GPIO* externally available
P0.28	SPARE5	Digital *GPIO* externally available
P0.29	N.A.	Not used
P0.30	SPARE6	Analog/digital *GPIO* externally available
P0.31	N.A.	Not used
P1.00	MCU_IF4	nRF9160 interface
P1.01	COEX2	nRF9160 COEX interface

I/O	Label	Description
P1.02	N.A.	Not used
P1.03	N.A.	Not used
P1.04	COEX1	nRF9160 COEX interface
P1.05	IF_SWK_CLK	nRF9160 SWD interface clock line
P1.06	N.A.	Not used
P1.07	COEX0	nRF9160 COEX interface
P1.08	SDA	I^2C data line
P1.09	SCL	I^2C clock line
P1.10	N.A.	Not used
P1.11	SPARE8	Digital *GPIO* externally available
P1.12	N.A.	Not used
P1.13	BOOT	Boot button
P1.14	N.A.	Not used
P1.15	N.A.	Not used

Table 5: nRF52840 pin map

6.6 Motion sensors

Nordic Thingy:91 includes a low-power accelerometer and a high-G accelerometer.

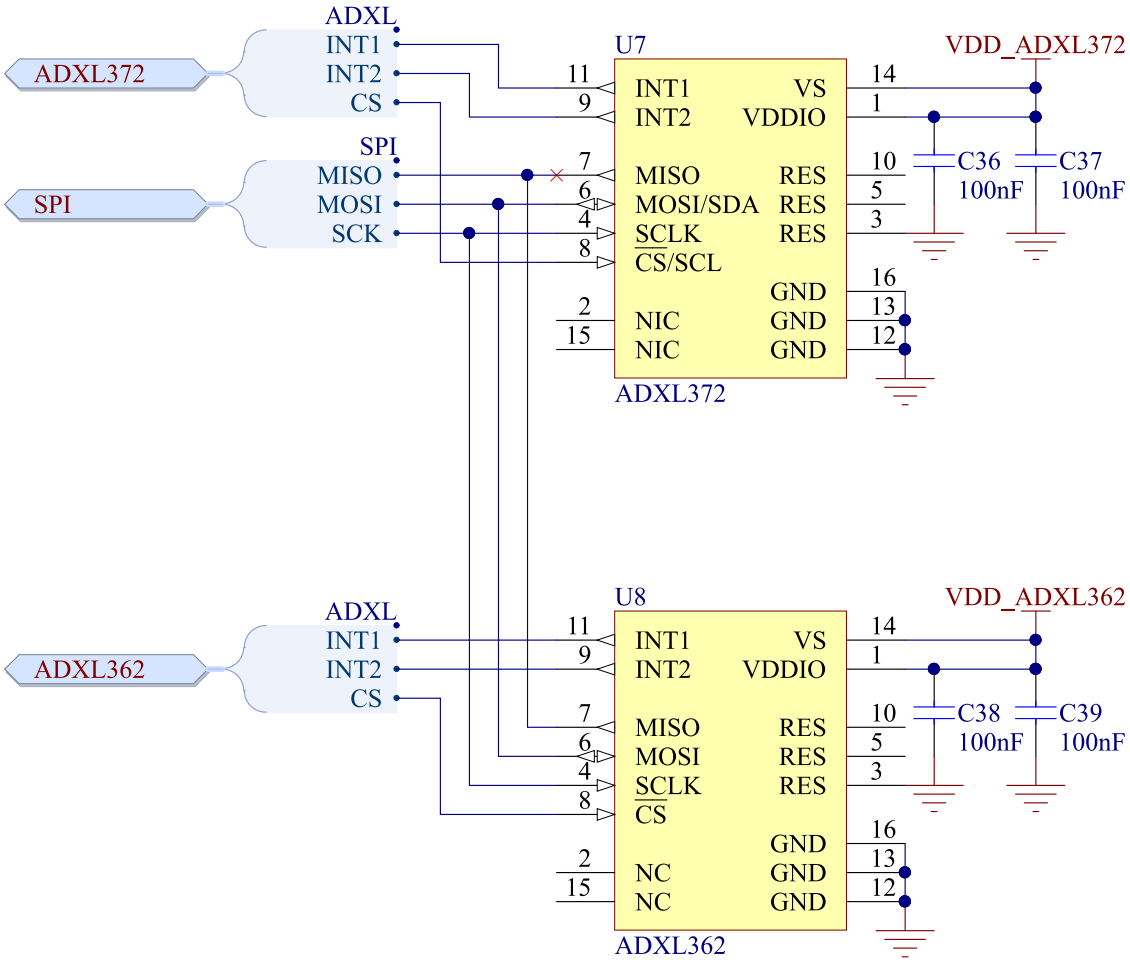


Figure 24: Low-power and high-G accelerometers schematic

When Nordic Thingy:91 is in low-power sleep mode, any user interaction will be detected by the low-power accelerometer. The accelerometer has an SPI interface and it can detect motion on three axes. By default, the INT2 line of the accelerometer is not connected to nRF9160. If you want to use the INT2 line, solder **SB6**.

For detecting shocks, Nordic Thingy:91 uses a high-G accelerometer. The accelerometer has an SPI interface, and it can detect motion on three axes. By default, the INT2 line of the accelerometer is not connected to nRF9160. If you want to use the INT2 line, solder **SB5**.

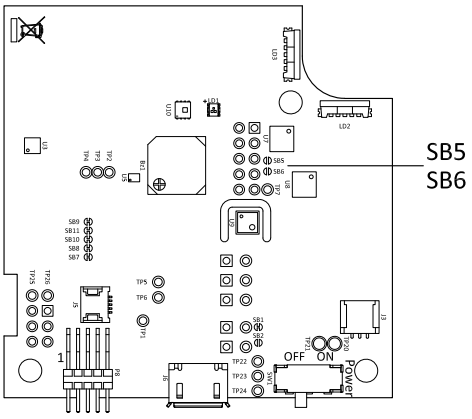


Figure 25: Low-power accelerometer and high-G accelerometer interrupt line 2 selection

6.7 Environment sensors

To monitor its surroundings, Nordic Thingy:91 has a multi-sensor chip that contains several sensors for detecting different environmental properties and a separate color and light sensor.

The multi-sensor chip contains sensors for temperature, humidity, air quality, and air pressure.

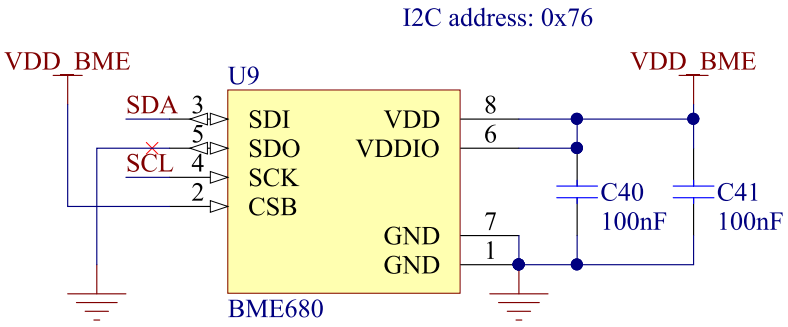


Figure 26: Environment sensor schematic

The color sensor onboard Nordic Thingy:91 senses red, green, blue, and infrared light. The sensor faces towards the blue transparent bottom case with light pipes guiding the light towards the sensor. To measure the color on a surface, the color sensor is accompanied with an RGB LED that can illuminate the surface enabling the color sensor to read the color of the reflected light. The color sensor is accessed through I^2C (slave address 0x38).

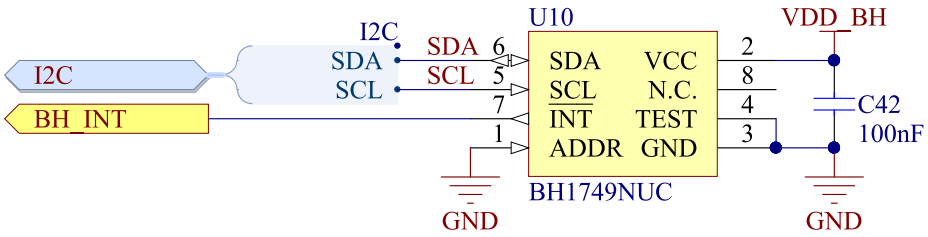


Figure 27: Color sensor schematic

6.8 Buzzer

For audio output, Nordic Thingy:91 has a magnetic buzzer. The buzzer is driven by a transistor using a PWM input.

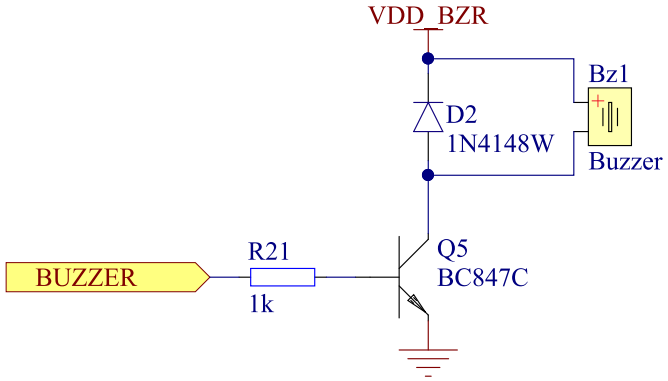


Figure 28: Buzzer schematic

6.9 LEDs and buttons

Nordic Thingy:91 user interface consists of RGB LEDs and two buttons.

6.9.1 RGB LED

Nordic Thingy:91 is equipped with three RGB LEDs.

Two of the LEDs are used to light up the light well and are controlled by the same signals using transistors as switches. The third LED is located near the color sensor and is used as auxiliary light for color measurements.

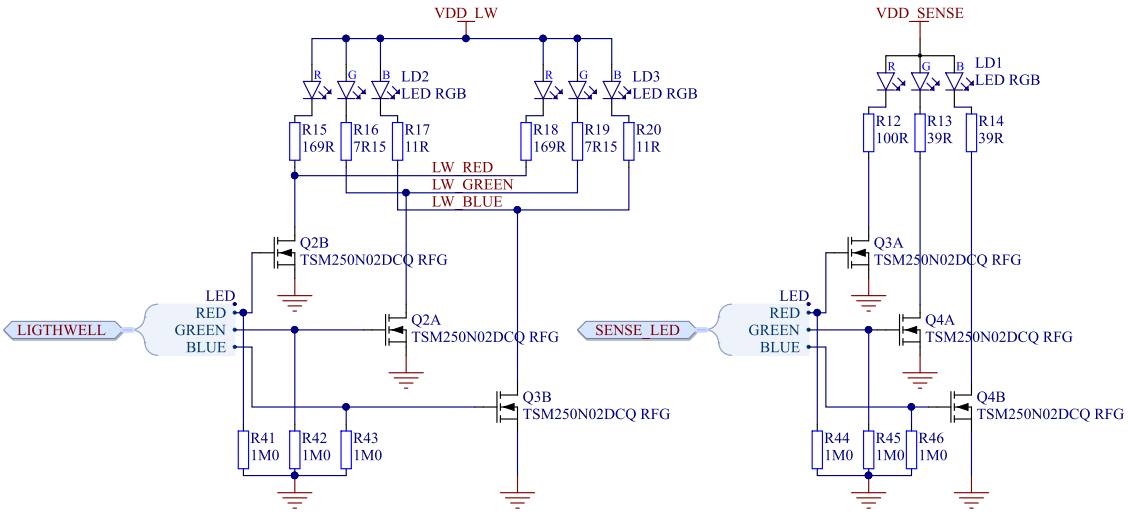


Figure 29: LED schematic

6.9.2 Buttons

Nordic Thingy:91 has two buttons.

The main button, located underneath the Nordic Semiconductor logo, is termed as the **SW3** button and it is used for user input. It is connected to the nRF9160 *SiP*. The second button, termed as **SW4**, is connected to the nRF52840 *SoC*. It is accessible only when the rubber cover on the device is removed. Refer the image Figure 4: Inserting the SIM card on page 10 to locate the buttons.

Either of the two buttons, **SW3** or **SW4**, can be used to activate the serial recovery mode of Nordic Thingy:91 to update the nRF9160 *SiP* or the nRF52840 *SoC* respectively.

For more information, see the sections Updating application firmware on nRF9160 using USB (MCUboot) on page 16 and Updating application firmware on nRF52840 using USB (MCUboot) on page 20.

6.10 Power supply

The main power source is a rechargeable lithium polymer (Li-Po) battery. The battery has a nominal capacity of 1400 mAh and can be recharged through *USB*.

Nordic Thingy:91 has a power switch that physically disconnects the battery and the *USB* power from the rest of the circuits. This switch must be on for Nordic Thingy:91 to work and charge the battery. When the power switch is in the OFF position, it activates a circuit that drains the 1.8 V power domain.

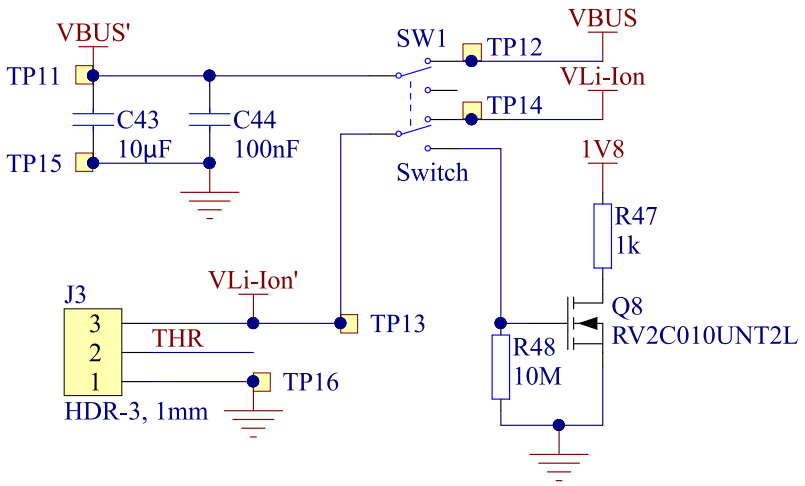


Figure 30: Schematics for battery connector, and the power switch

6.10.1 PMIC

Nordic Thingy:91 uses a power management IC (PMIC) as internal power management. The PMIC includes a battery charger, a fuel gauge and two regulator outputs.

The PMIC has three voltage domain outputs that are used on Nordic Thingy:91:

- VSYS used for the nRF9160 *SiP*
- One 1.8 V output used as *GPIO* voltage
- One 3.3 V output used for analog circuitry

The 1.8 V domain supplies the nRF52840, the accelerometers, the environment sensors, and the *GPIO*s of the nRF9160. This domain must always be on for Nordic Thingy:91 to work.

The 3.3 V domain supplies the LEDs, the color sensor, the buzzer, the RF switches and the *GPS* LNA. This power domain can be powered down to save power when Nordic Thingy:91 is in sleep mode.

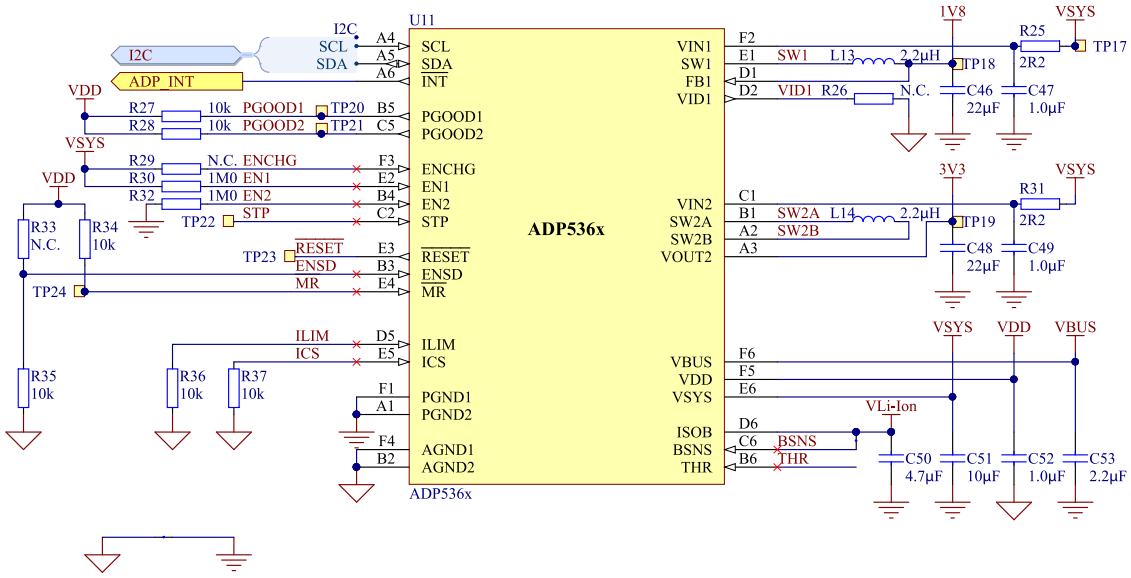


Figure 31: PMIC schematic

6.10.2 Current measurement

It is possible to measure the current flowing to nRF9160 and nRF52840 by cutting the short on SB3 (nRF9160) and/or SB4 (nRF52840) and placing an ampere meter between the terminals of **P1** (nRF9160) and terminals of **P2** (nRF52840).

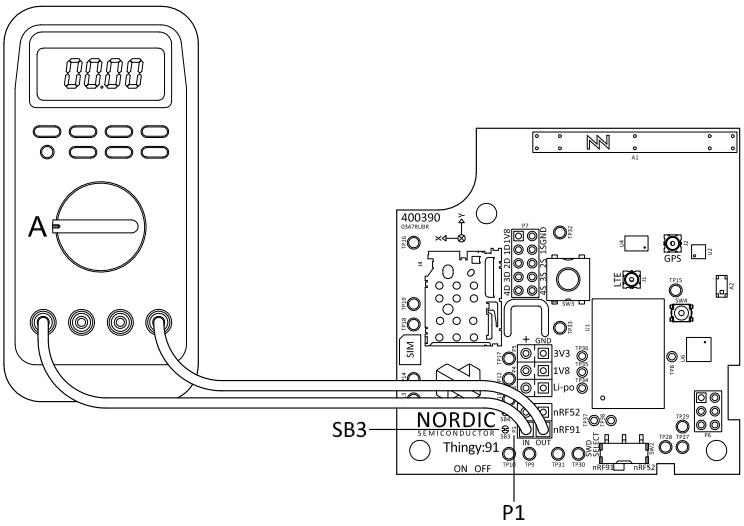


Figure 32: Measuring current to the nRF9160

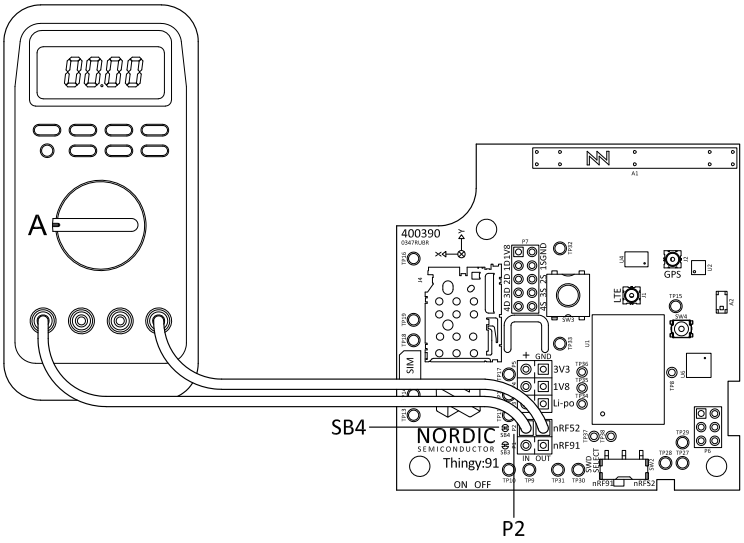


Figure 33: Measuring current to the nRF52840

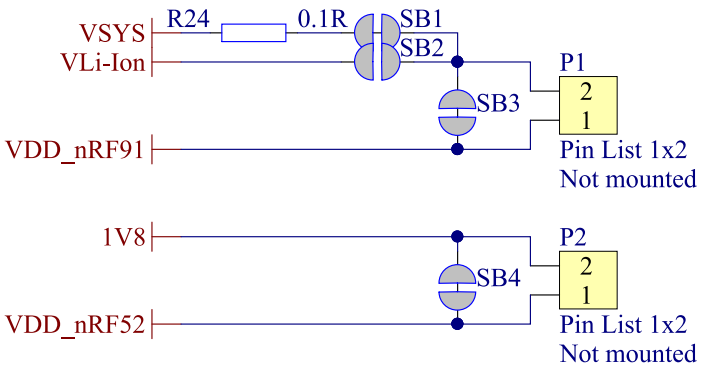


Figure 34: Current measurement schematic

6.11 Programming and debugging interface

Nordic Thingy:91 is equipped with one programming and debugging interface connector (P8) that is shared between the nRF9160 and nRF52840.

The device to be programmed is selected by the SWD SELECT switch (SW2). The selection of device can also be controlled by connecting TP28 to 1.8 V or ground.

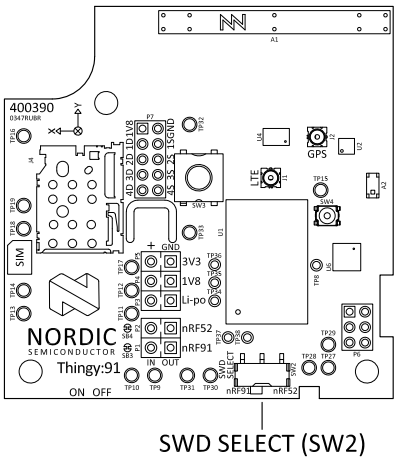


Figure 35: SWD SELECT switch

The SWD interface of the nRF9160 can also be connected to the nRF52840. The enabling of this connection is controlled by the nRF52840.

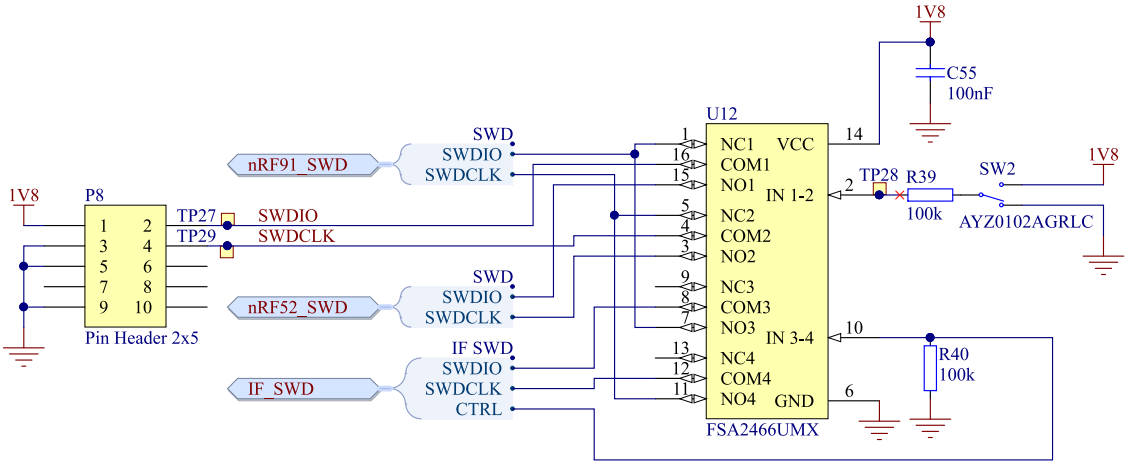


Figure 36: SWD interface and control schematic

6.12 Interface

To enable the user to connect external hardware, Nordic Thingy:91 routes some of the *GPIO*s to connectors or test points and transistors to drive higher currents.

6.12.1 N-MOS transistors

Nordic Thingy:91 is equipped with four N-MOS transistors that can be used to drive small DC motors or LEDs. The drain and source of the transistors are available on external connectors and the gate is connected directly to the nRF9160.

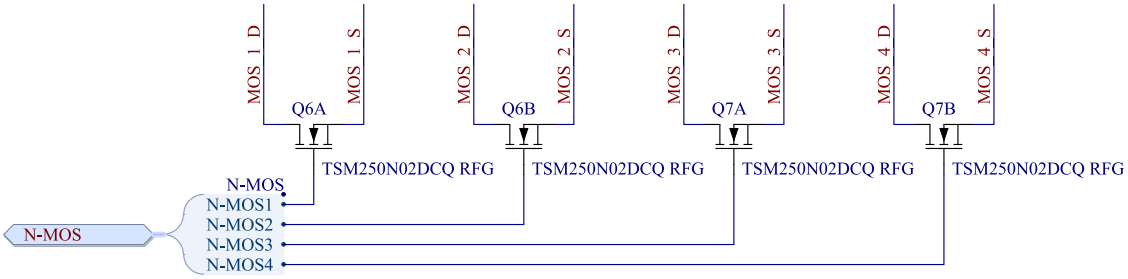


Figure 37: Schematics for the N-MOS transistors

6.12.2 Connectors

In addition to the N-MOS drain and source on **P7**, power domains and extra *GPIO*s can be found on connectors **P3**-**P6**.

For more information, see Connector pinouts on page 38.

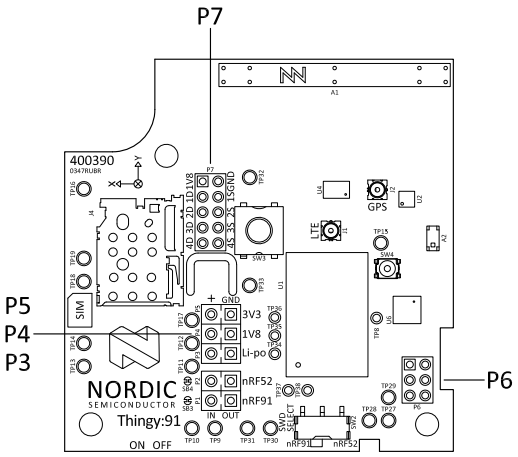


Figure 38: Nordic Thingy:91 external connectors

Figure 39: Interface connectors

6.12.2.1 Connector pinouts

Pin	Signal	Description
1	GND	Ground
2	$V_{Li\text{-}Ion}$	Battery voltage

Table 6: Pinout of connector P3

Pin	Signal	Description
1	GND	Ground
2	1V8	Regulated 1.8 V domain

Table 7: Pinout of connector P4

Pin	Signal	Description
1	GND	Ground
2	3V3	Regulated 3.3 V domain

Table 8: Pinout of connector P5

Pin	Signal	Description
1	SPARE1	*GPIO* of the nRF52840
2	SPARE2	*GPIO* of the nRF52840
3	SPARE3	*GPIO* of the nRF52840
4	SPARE4	*GPIO* of the nRF52840
5	SPARE5	*GPIO* of the nRF52840
6	SPARE6	*GPIO* of the nRF52840

Table 9: Pinout of connector P6

Pin	Signal	Description
1	1V8	Regulated 1.8 V domain
2	GND	Ground
3	MOS_1_D	Drain of n-channel MOSFET 1
4	MOS_1_S	Source of n-channel MOSFET 1
5	MOS_2_D	Drain of n-channel MOSFET 2
6	MOS_2_S	Source of n-channel MOSFET 2
7	MOS_3_D	Drain of n-channel MOSFET 3
8	MOS_3_S	Source of n-channel MOSFET 3
9	MOS_4_D	Drain of n-channel MOSFET 4
10	MOS_4_S	Source of n-channel MOSFET 4

Table 10: Pinout of connector P7

6.12.3 Test points

Test point	Location	Signal	Description
TP1	Bottom	N.A.	Reserved
TP2	Bottom	RF_SW1	Bit 0 of RF switch control signals
TP3	Bottom	RF_SW2	Bit 1 of RF switch control signals
TP4	Bottom	RF_SW3	Bit 2 of RF switch control signals
TP5	Bottom	N.A.	Reserved
TP6	Bottom	N.A.	Reserved
TP7	Bottom	nRF91-P0.10	*GPIO* of the nRF9160
TP8	Top	nRF52-P0.18/RESET	*GPIO*/RESET of the nRF52840
TP9	Top	SCL	I^2C clock line
TP10	Top	SDA	I^2C data line
TP11	Top	VBUS'	USB voltage before power switch
TP12	Top	VBUS	USB voltage after power switch
TP13	Top	$V_{Li\text{-}Ion'}$	Battery voltage before power switch
TP14	Top	$V_{Li\text{-}Ion}$	Battery voltage after power switch
TP15	Top	GND	Ground
TP16	Top	GND	Ground
TP17	Top	VSYS	Internal power domain of PMIC and default nRF9160 power supply
TP18	Top	1V8	Regulated 1.8 V domain
TP19	Top	3V3	Regulated 3.3 V domain
TP20	Bottom	ADP_PGOOD1	PMIC output status indication pin 1
TP21	Bottom	ADP_PGOOD2	PMIC output status indication pin 2
TP22	Bottom	ADP_STP	Stop the buck regulator switching of PMIC
TP23	Bottom	ADP_RESET	PMIC reset output
TP24	Bottom	ADP_MR	PMIC manual reset input
TP25	Bottom	SPARE7	*GPIO* of the nRF52840
TP26	Bottom	SPARE8	*GPIO* of the nRF52840
TP27	Top	SWDIO	Programming interface data line
TP28	Top	SWDSEL	Programming interface target select
TP29	Top	SWDCLK	Programming interface clock line
TP30	Top	D-	USB data line
TP31	Top	D+	USB data line
TP32	Top	nRF91-P0.13/AIN0	Analog/digital *GPIO* of the nRF9160, combined with N-MOS1

Test point	Location	Signal	Description
TP33	Top	nRF91-P0.16/AIN3	Analog/digital *GPIO* of the nRF9160, combined with N-MOS4
TP34	Top	SCK	SPI clock line
TP35	Top	MOSI	SPI master output, slave input data line
TP36	Top	MISO	SPI master input, slave output data line
TP37	Top	ADXL372_CS	High-G accelerometer chip select line
TP38	Top	ADXL362_CS	Low-power accelerometer chip select line

Table 11: Pinout of connector P3

7 Regulatory notices

The following regulatory notices apply to Nordic Thingy:91.

7.1 FCC regulatory notices

Modification statement

Nordic Semiconductor ASA has not approved any changes or modifications to this device by the user. Any changes or modifications could void the user's authority to operate the equipment.

Interference statement

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Wireless notice

This device complies with FCC radiation exposure limits set forth for an uncontrolled environment and meets the FCC radio frequency (RF) Exposure Guidelines. This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter. The antenna should be installed and operated with minimum distance of 20 cm between the radiator and your body.

FCC Class B digital device notice

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Glossary

DK (Development Kit)

A development platform used for application development.

GPIO

General-Purpose Input/Output

Global Positioning System (GPS)

A satellite-based radio navigation system that provides its users with accurate location and time information over the globe.

Integrated Development Environment (IDE)

A software application that provides facilities for software development.

MCUboot

A secure bootloader for 32-bit microcontroller units, which is independent of hardware and operating system.

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.

Personal Unblocking Key (PUK)

A digit sequence required in 3GPP mobile phones to unlock a *SIM* that has disabled itself after an in correct personal identification number has been entered multiple times.

Software Development Kit (SDK)

A set of tools used for developing applications for a specific device or operating system.

SEGGER Embedded Studio (SES)

A cross-platform *IDE* for embedded C/C++ programming with support for Nordic Semiconductor devices, produced by SEGGER Microcontroller.

Subscriber Identity Module (SIM)

A card used in *User Equipment (UE)* containing data for subscriber identification.

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.

User Equipment (UE)

Any device used by an end-user to communicate. The UE consists of the Mobile Equipment (ME) and the Universal Integrated Circuit Card (UICC).

Universal Integrated Circuit Card (UICC)

A new generation *SIM* used in *UE* for ensuring the integrity and security of personal data.

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.

Acronyms and abbreviations

These acronyms and abbreviations are used in this document.

DK
Development Kit

GPIO
General-Purpose Input/Output

GPS
Global Positioning System

IDE
Integrated Development Environment

NFC
Near Field Communication

PUK
Personal Unblocking Key

SDK
Software Development Kit

SES
SEGGER Embedded Studio

SIM
Subscriber Identity Module

SiP
System in Package

SoC
System on Chip

USB
Universal Serial Bus

UICC
Universal Integrated Circuit Card

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

© 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.

COMPANY WITH
QUALITY SYSTEM
CERTIFIED BY DNV GL
= ISO 9001 =