AN11664

QN902x Quick Start Guide

Rev. 3 — 17 April 2018 Application note

Document information

Info	Content
Keywords	mini DK, SDK, installation, working mode, download, NVDS configuration, connect, GPIO configuration
Abstract	This application note discusses the use of QN9020 mini Development Kit (DK) for evaluating and developing Bluetooth Low Energy (BLE) solutions based on QN902x.

Revision history

Rev	Date	Description
3	20180417	• Updated figures.
2	20160615	• Added “IAR” in Section 1.2.2. • Updated Section 2.2, Section 4.2.4, Section 5.1.1.3, Section 5.4.2, • In Section 3.3.1 added Figure 5. • In Section 4.2 updated title “QBlue SDK GUI”. • In Section 5.1.1.2 updated Figure 13. • In Section 6.2.1.1 added Figure 34. • In Section 6.2.1.2 updated Figure 35 and in Section 6.2.1.3 updated Figure 36. • Added figure title for Figure 35, Figure 36, and Figure 37.
1	20150605	Initial release

Contact information

For more information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

1. Introduction

1.1 Overview

The QN9020 mini Development Kit (DK) is designed for evaluating and developing Bluetooth Low Energy (BLE) solutions based on QN902x. The purpose of this document is to give an overview of hardware and software included in the QN9020 mini DK. For more detailed information on BLE technology and software platform for QN9020, refer to *"QN9020 software developers guide"*.

1.2 System requirements

1.2.1 System requirements

The QN9020 mini DK has the following system requirements:

- a computer running with Windows XP, Windows 7 or Windows 8
- a computer with a minimum of 2 USB ports

Note: OS X (Mac) is not supported.

1.2.2 External resources

- Keil MDK-ARM
- J-Link software
- IAR

2. Contents of kit

2.1 Hardware

The QN9020 mini DK contains the following hardware components:

- mini DK board
- BLE dongle
- USB cable

The hardware components are shown in Figure 1.

Fig 1. Hardware components of DK

Note: mini DK can also be used for software development for QN9021. There is a difference between the GPIO for QN9021 and QN9020. For details, refer to the data sheet at https://www.nxp.com/documents/data_sheet/QN902X.pdf.

2.2 Software

The Software Development Kit (SDK) is named as QN902x_SDK_xxx.zip. Latest SDK version can be downloaded from NXP web http://www.nxp.com/products/microcontrollers-and-processors/more-processors/application-specific-mcus-mpus/bluetooth-low-energy-ble/ulta-low-power-ble-system-on-chip-solution:QN9020?fpsp=1&tab=Design_Tools_Tab.

3. Getting started

3.1 System overview

The system consists of three parts; see Figure 2:

- mini DK board: It is used for developing Bluetooth Low Energy (BLE) applications based on QN9020. It comprises of onboard RF matching circuit and antenna, power supply, GPIO connector, buttons, and LEDs. The J-Link On-Board (OB) debugger is used to bridge QN9020 SWD and UART interface to PC, download program and debug from PC.
- BLE dongle: It is a Bluetooth Low Energy (BLE) device controlled by the QTool running on a PC. It works with the mini DK as a pair for evaluation and debugging. It is connected to a PC via USB.
- QN902xStudio: It is a software tool chain to work with the mini DK and BLE dongle.

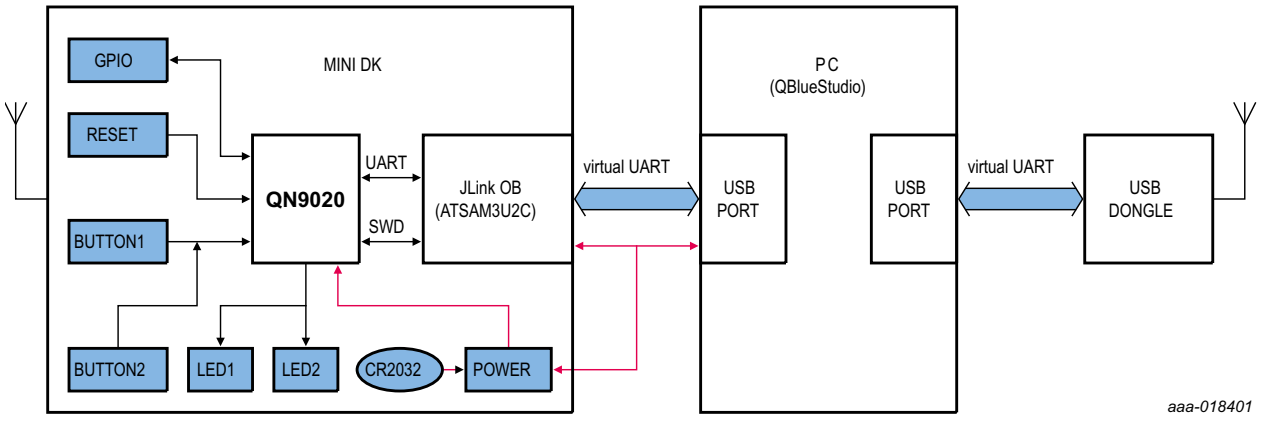

Fig 2. System model

3.2 Software installation

Before connecting the mini DK board to the computer, install Keil MDK-ARM or IAR, QN902x_SDK_xxx.exe and Setup_JLink_Vxxx.exe. Keil is used in this document to show the instructions.

3.2.1 Keil MDK-ARM

Keil MDK-ARM can be downloaded and installed from the URL: https://www.keil.com/download/product/; see Figure 3. Version 4.5 or newer is recommended.

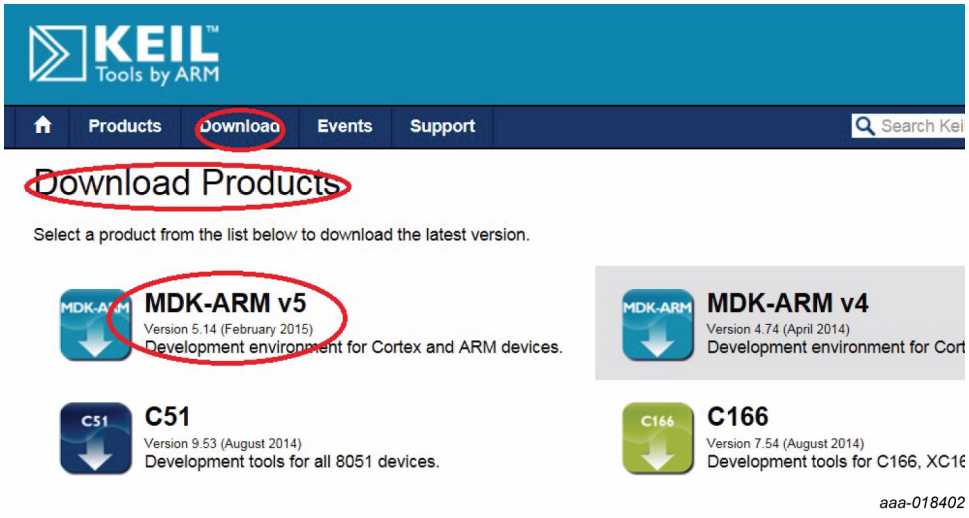

Fig 3. Download Keil MDK-ARM

3.2.2 QN902x_SDK_xxx.exe

Install released SDK package QN902x_SDK_xxx.exe.

3.2.3 J-Link software

Download and install Setup_JLink_Vxxx.exe (Version 4.6.6 or newer is recommended) from www.segger.com based on the development environment; see Figure 4). All mini DK boards have their individual SEGGER ID (serial number) labeled at the back.

Fig 4. Location for J-Link software

Note: The J-Link software also contains a USB driver for J-Link-OB.

3.3 Hardware setup

3.3.1 Hardware connection

Refer "*QN9020 mini DK User Guide*" for instructions to use the mini DK board.

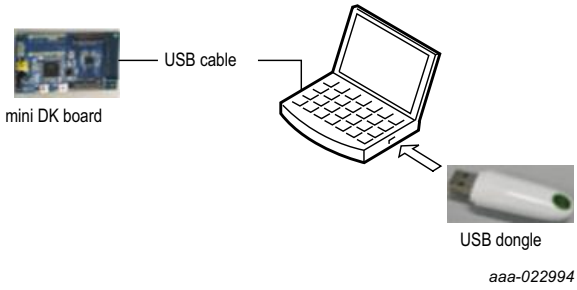

Fig 5. Hardware connection

3.3.2 Driver installation

Virtual COM Port (VCP) drivers make the USB device appear as an additional COM port on the PC. Application software accesses the USB device in the same way as it would access a standard COM port.

3.3.2.1 Driver for J-Link-OB

Driver for J-Link-OB is included in J-Link software; see Section 3.2.3. It is installed automatically while installing the J-Link software.

3.3.2.2 Driver for BLE dongle

Refer to "*Application Note AN-104*" for detailed installation. It can be downloaded from FTDI website https://www.ftdichip.com/Support/Documents/AppNotes.htm

This guide helps with the installation of the Combined Driver Model (CDM) driver from FTDI for the Microsoft windows operating system.

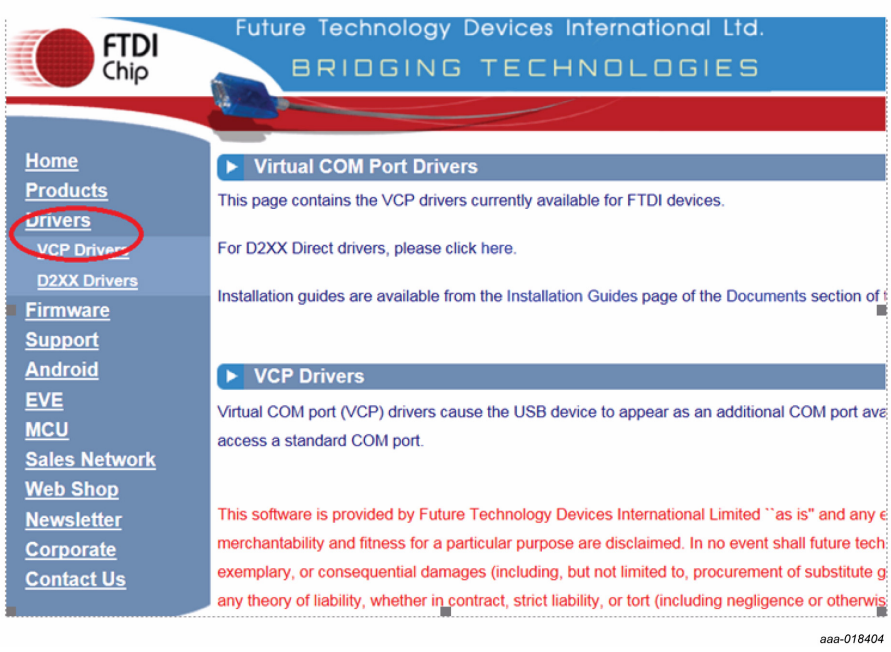

Fig 6. Driver location for BLE dongle

Installed drivers can be verified by checking the device manager in windows OS; see Figure 7. Two COM ports are added to the list. One is the USB serial port (COM5) indicating the BLE dongle. The other is a J-Link CDC UART port (COM4) for the J-Link-OB on mini DK board. The COM port number can be different for each computer. Therefore, it is always a good practice to check the ports with regards to the hardware connected.

Note: Ensure that correct COM port number is used when downloading the code and during evaluation using QTool. COM port assignments can be checked in the device manager.

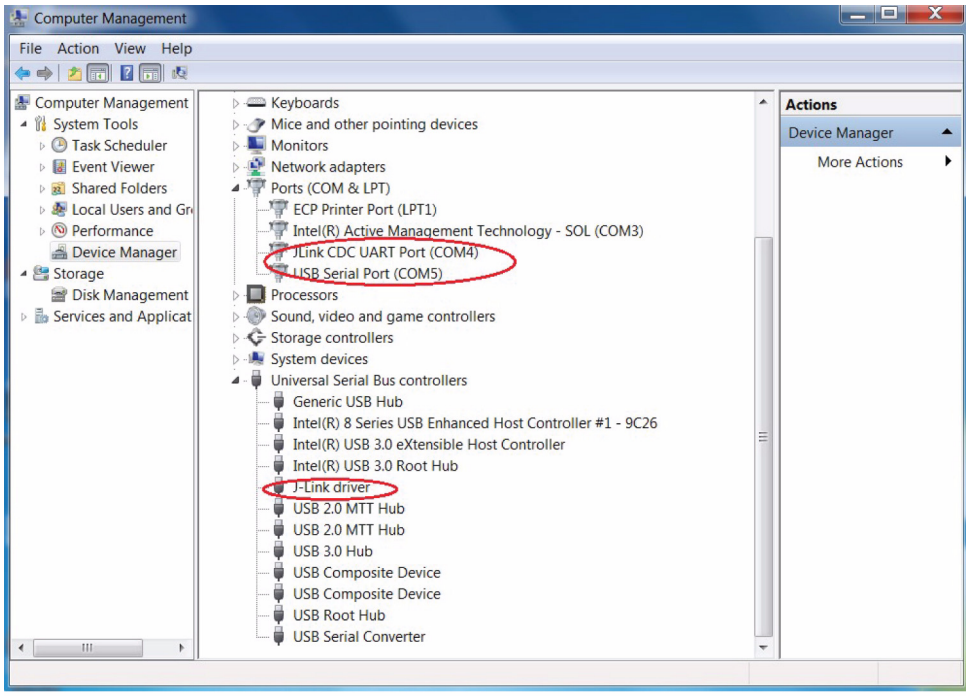

Fig 7. COM port of BLE dongle and J-Link-OB

3.4 Working mode

QN9020 provides a flexible platform for wireless applications. It supports two working modes, wireless SoC mode and network processor mode. For detailed information related to the working modes, refer to *"QN9020 software developers guide"*.

3.4.1 Wireless SoC mode

In wireless SoC mode, link layer, host protocol, profiles and application run on QN9020. Figure 8 shows how QN9020 is used in SoC mode. Application examples use SoC mode.

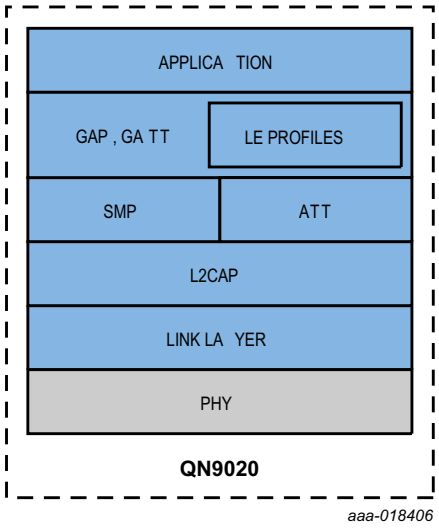

Fig 8. Wireless SoC mode

3.4.2 Network processor mode

In network processor mode, the link layer, host protocols and profiles run on QN9020 whereas the application runs on external processor. The two components communicate via Easy ACI (Easy Application Controller Interface) over UART, provided in QN9020 SDK. For details, refer to the *"QN9020 easy ACI programming guide"*.

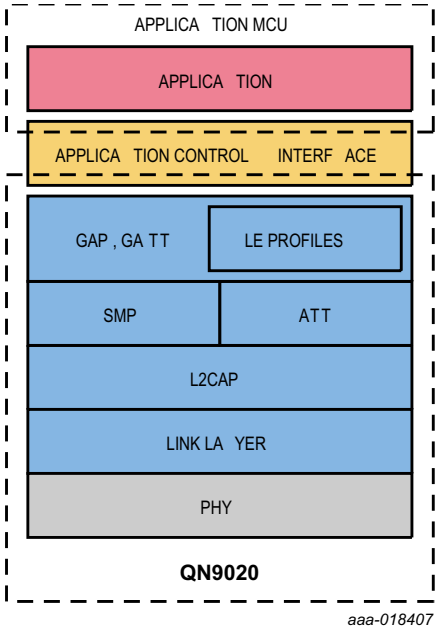

Fig 9. Network processor mode

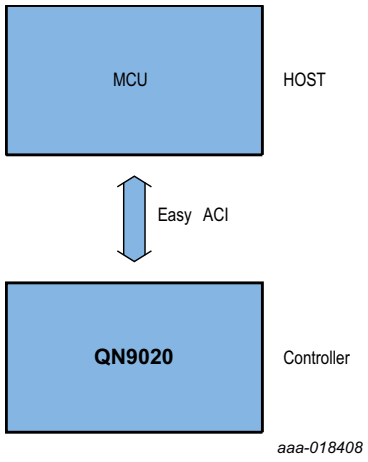

Fig 10. Easy ACI interface

4. Components of QN902x SDK

When QN902x SDK is installed and executed, it opens a GUI referred to as QN902xStudio; see Figure 12. The GUI has links to various tools, BLE example codes, driver codes and detailed software/hardware guides.

4.1 SDK folder structure

Once installed, a folder is created in the main directory. The installation folder contains various subfolders; see Figure 11.

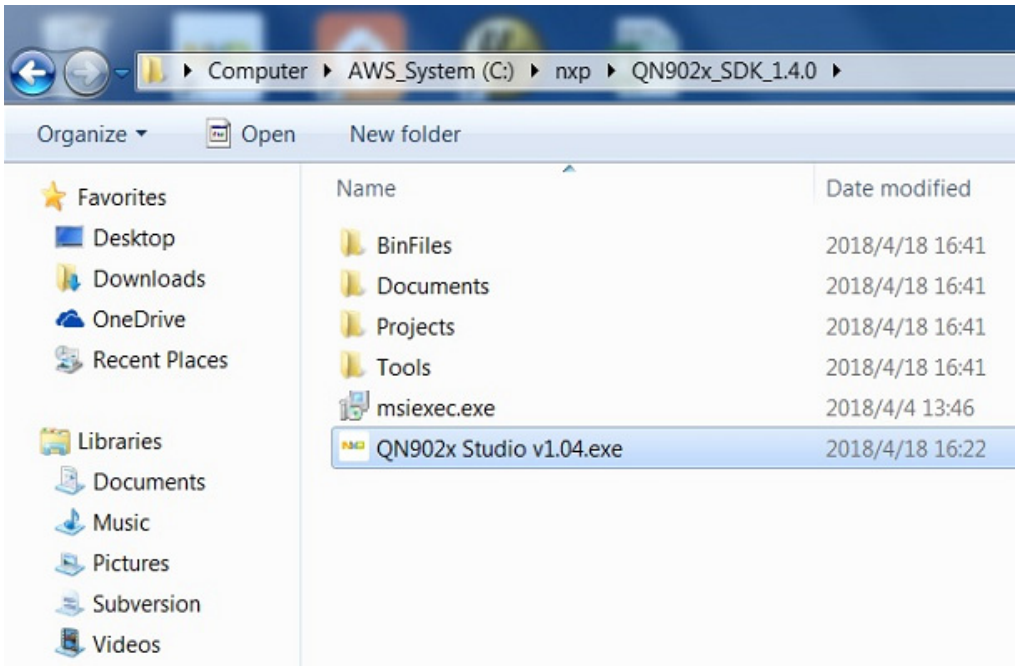

Fig 11. QN902xStudio installation folder

4.1.1 BinFiles

This folder contains all binary files for sample applications provided in SDK.

4.1.2 Documents

Documents related to QN9020 SDK are found here.

4.1.3 Projects

This folder contains source code and project file in Keil and IAR for various BLE example applications.

4.1.4 Tools

This folder contains all tools with GUI in SDK to support application development.

4.2 QN902x SDK GUI

QN902x SDK provides a GUI for users. Figure 12 shows the start page after launching QN902x Studio.

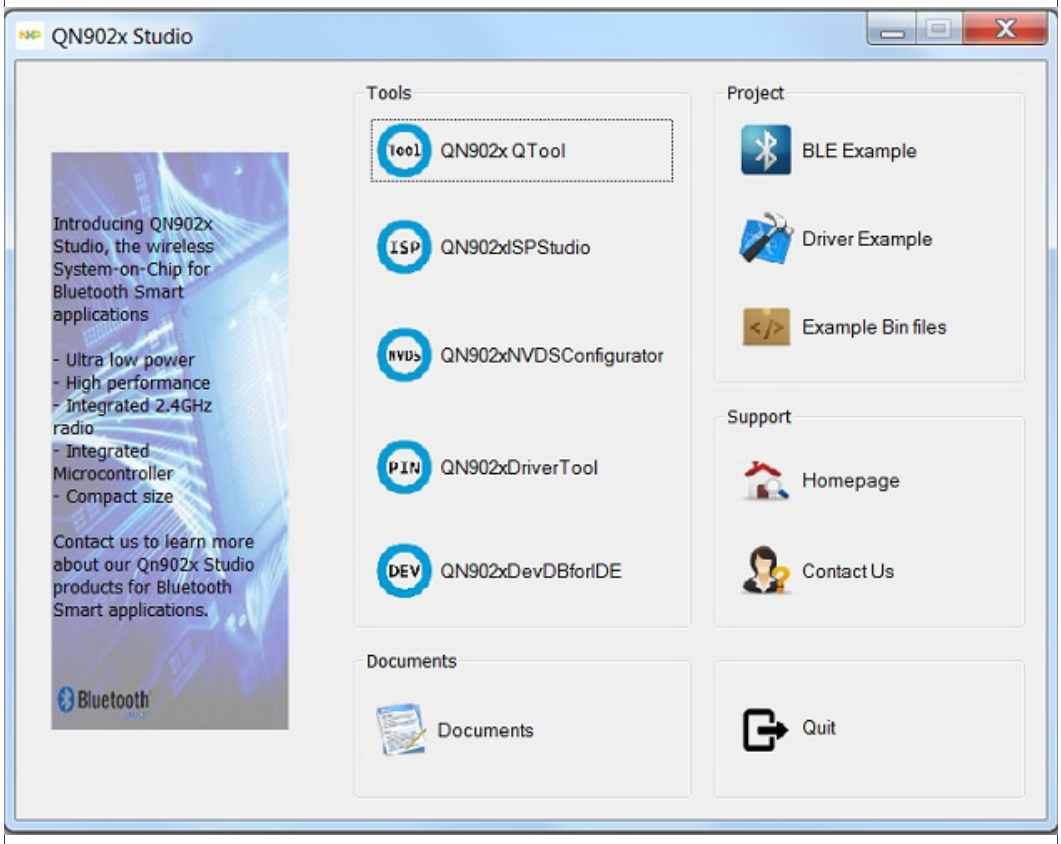

Fig 12. QN902x Studio start page

4.2.1 Tools

- QTool: QTool is a PC tool that controls the BLE dongle. It allows the user to set up a connection between the BLE USB dongle and the other BLE device. A detailed user manual for the QTool can be found in the software documentation (*"QTool User Manual"*).
- QN902xISPStudio: QN902xISPStudio is a tool to download application binary file, data file or NVDS configuration file into QN90xx series of BLE devices. Refer to *"QN902x ISP Studio Manual"* for more information.
- QN902xNVDSConfigurator: It is a tool to add, edit and delete NVDS configuration data for BLE application development based on QN9020 platform. Refer to "*QN902x NVDS Configurator Manual"* for details.
- QN902xDriverTool: It is a PC tool to make it easier to write peripheral driver code for BLE application development based on the QN9020 platform. Refer to "*QN902x Driver Tools Manual*" for further details.
- Qn9020DevDBforIDE: This tool is used to add device database for QN9020 series chip into IDE, including ARM Keil MDK-ARM and IAR. Refer to "*QN9020DevDBforIDE User Manual*".

4.2.2 Documents

Software Doc in the documents area leads directly to a software document folder described in Section 4.1.2.

4.2.3 Project

Project area consists of three parts namely:

- BLE example: linked to the BLE folder under the projects folder; see Section 4.1.3.
- Driver example: linked to the driver folder under the projects folder; see Section 4.1.3.
- Example bin files: linked to the BinFiles folder; see Section 4.1.1.

4.2.4 Support

It consists of links to get more information about QN902x and support page.

- Homepage: http://www.nxp.com/
- Contact us: http://www.nxp.com/support/communities:COMMUNITIES

5. Quick start with demo

The QN9020 mini DK board can be shipped with some pre-loaded profile. A proximity profile is used here as an example to demonstrate how to start with QN902x Studio quickly. The proximity demo enables an interactive connection with the BLE dongle acting as a BLE master and mini DK board as a slave.

Note: All mini DK boards are in a deep sleep mode when powered-up. It means that the debug ports do not respond. To debug or start advertising and eventually connect, press BUTTON1 on the mini DK.

5.1 Download file

5.1.1 Steps to download file with QN902xISPStudio

5.1.1.1 Step 1:

Start QN902x Studio.

5.1.1.2 Step 2:

Click ‘QN902xISPStudio’ on QN902x Studio start page; see menu in Figure 13.

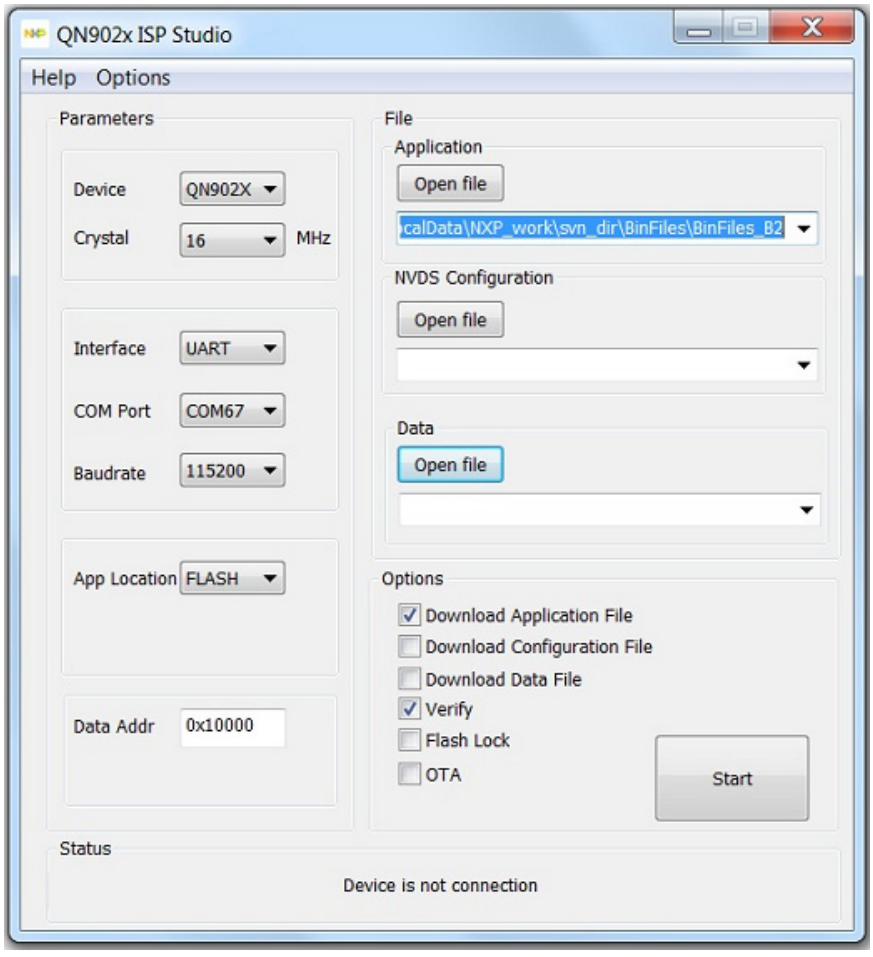

Fig 13. Setting ISP studio

5.1.1.3 Step 3:

Select UART interface, correct COM port and baudrate. Generally, the baud rate is 115200. Use the COM port for J-Link CDC UART port.

5.1.1.4 Step 4:

To locate the bin file, click “Open file” under application, and choose “proxr.bin”. The demo bin files provided in QN902x SDK are located in C:\NXP\QN902x_SDK_1.4.0\Projects\BLE\prj_proxr\keil\bin.

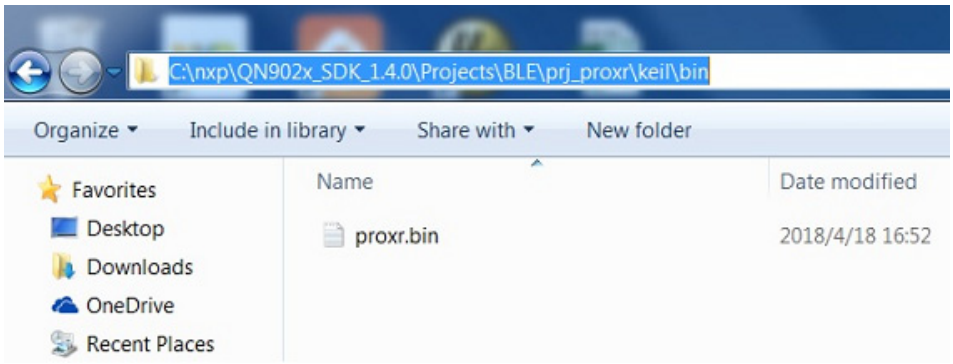

Fig 14. Proximity bin file

5.1.1.5 Step 5:

Click the “Start” button on QN902xISPStudio and then press the “QN_RST” button on the board; see Figure 15. The bin file is downloaded to board automatically, once the “QN_RST” button is released. If the download is successful, a message pops up; see Figure 16.

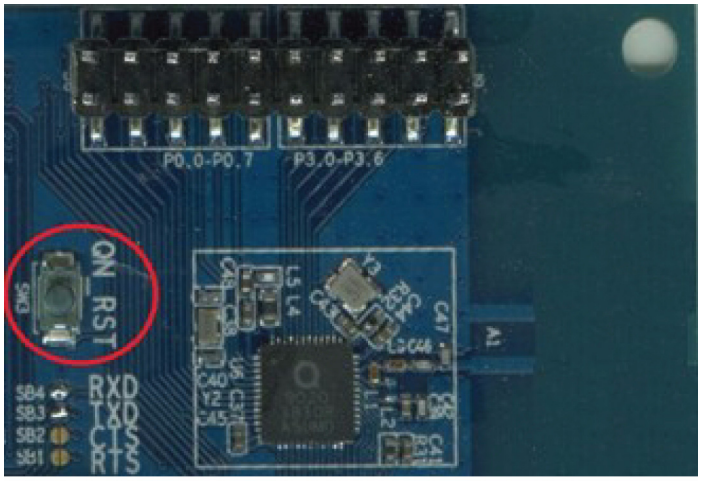

Fig 15. QN_RST button

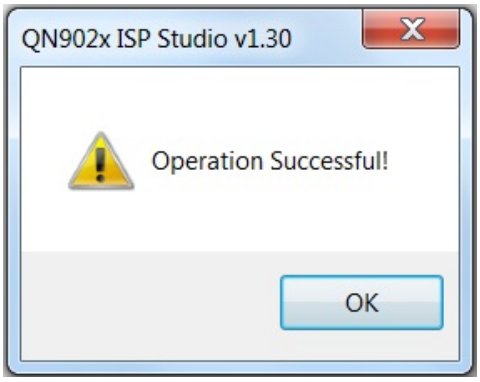

Fig 16. Download successful

If the bin file download fails, a message pops up; see Figure 17. If it occurs, perform the following:

1. Check the USB connection and whether LED1 and LED2 are ON.
2. Whenever the Start button is used to download files, press and release the QN_RST button on mini DK board to switch the board into boot mode.
3. Make sure that the COM port setting is correct on the system. The setting can be found in device manager; see Figure 7.

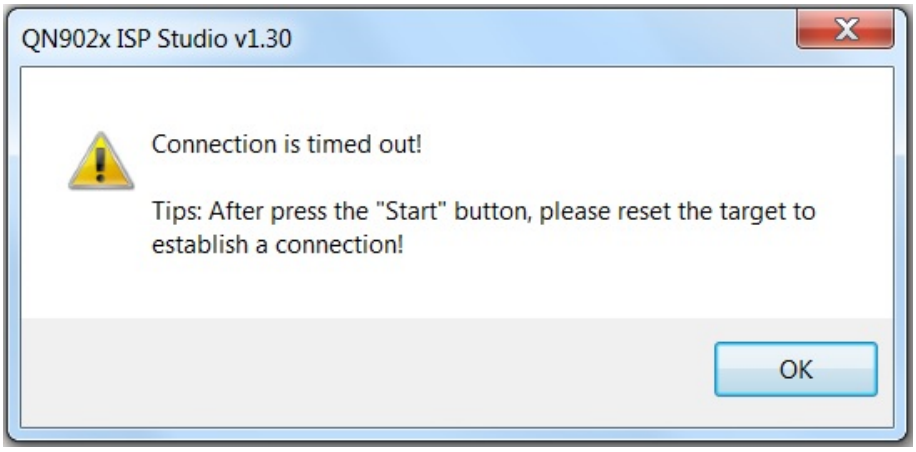

Fig 17. Failure message

For QN920E, bootloader support Fast boot mode which may be needed in some special application. If you want to enable fast boot mode, please config the ISP Studio before downloading. See Figure 18.

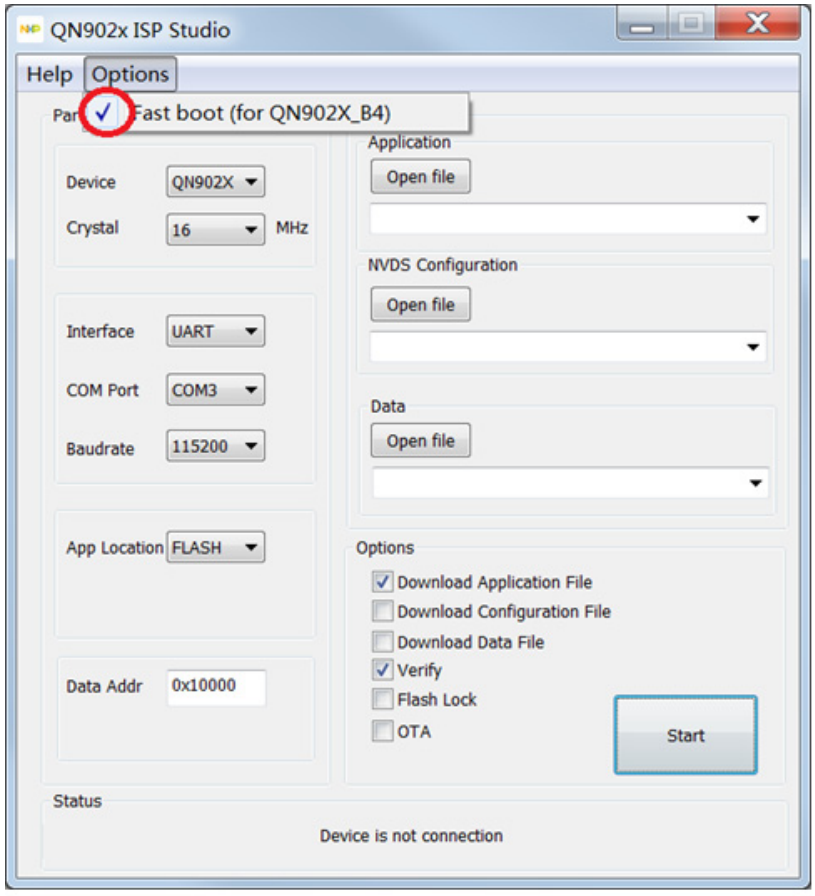

Fig 18. Failure message

For more information about the QN902xISPStudio tool, refer to the *"QN902x ISP Studio Manual*".

5.1.2 Download file in Keil

Keil can also be used to download the compiled bin file. If QN9020 is in sleep mode while programming, press BUTTON1 or BUTTON2 to wake up the mini DK. Figure 19 and Figure 20 show the error messages that pop up when QN9020 is in sleep mode and the user is trying to program through Keil.

Fig 19. Failure note

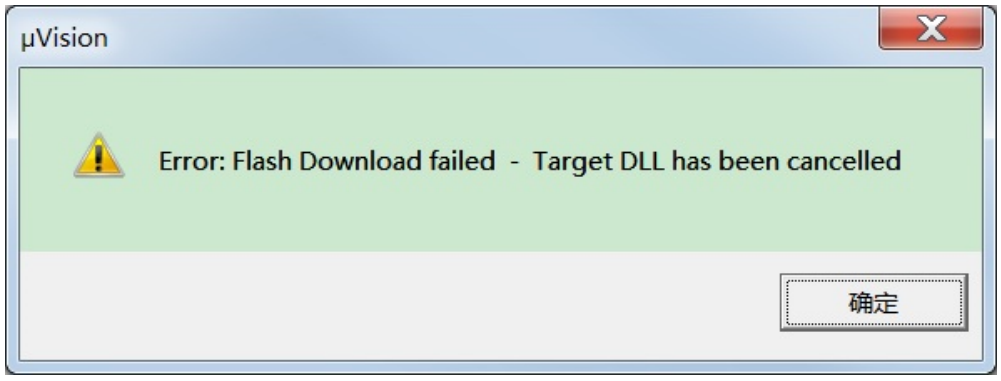

Fig 20. Failure note

5.2 Configure BLE device

Use NVDS configurator tool and configure the BLE device address, name, etc. This configuration data is stored in the NVDS area of flash. The NVDS configurator makes it easier to add, edit and delete the configuration data. The tool is also used to burn the configuration data to target chip, or dump it from the chip. For more information on this tool, refer to the *"QN902x NVDS Configurator Manual*".

5.2.1 Steps to configure BLE device

5.2.1.1 Step 1:

To start NVDS configurator, Click QN902xNVDSConfigurator button on QN902x Studio start page; see Figure 21.

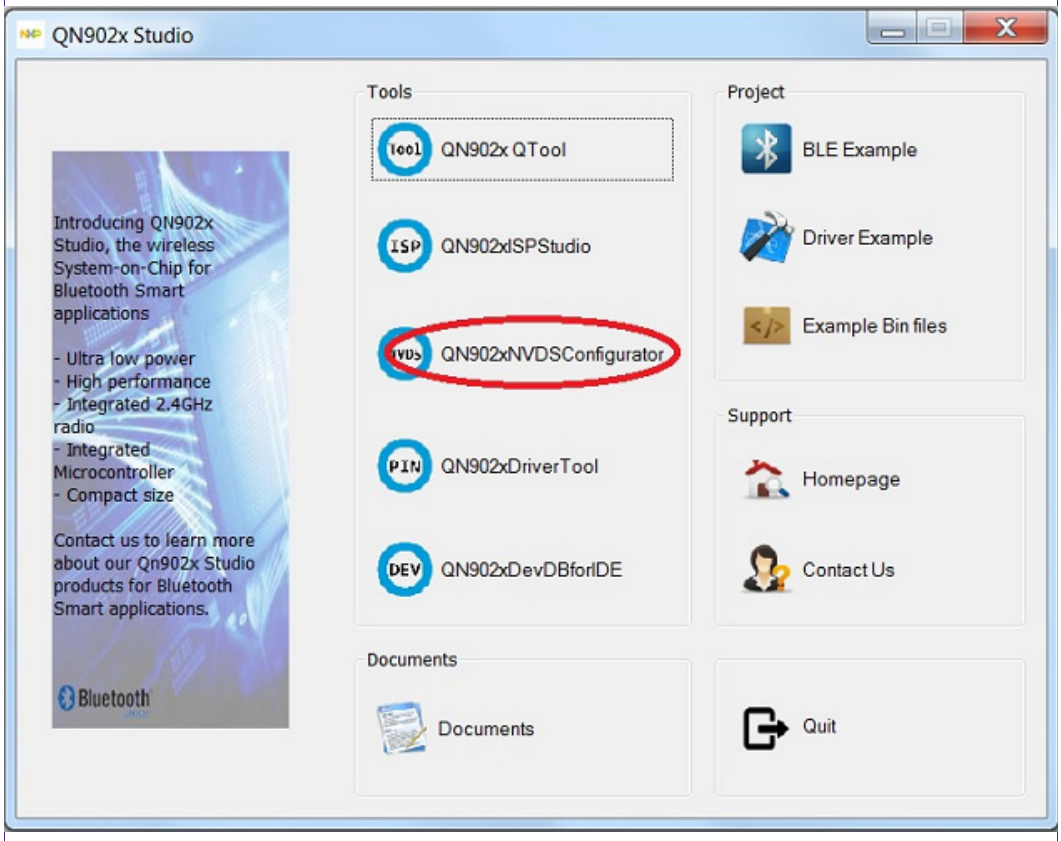

Fig 21. Start NVDS configurator

5.2.1.2 Step 2:

Choose the corresponding UART COM port and click connect button on the “Connect” dialog; see Figure 22.

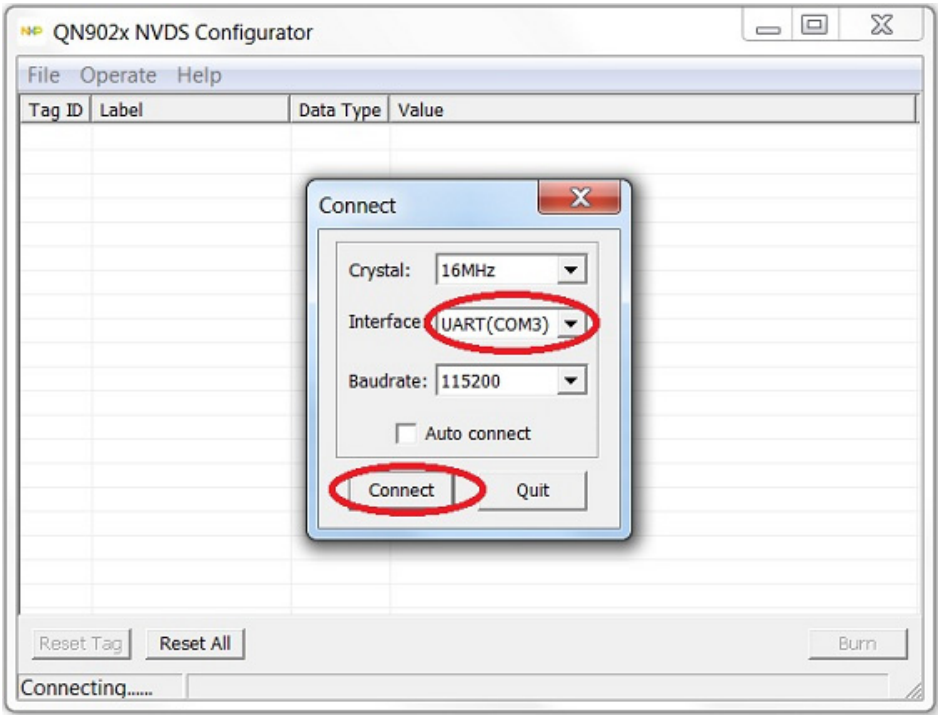

Fig 22. Set connection in NVDS

5.2.1.3 Step 3:

Press “QN_RST” button on the mini DK board and then release it to connect the board; see Figure 15. Once connected, it shows configurable items; see Figure 23.

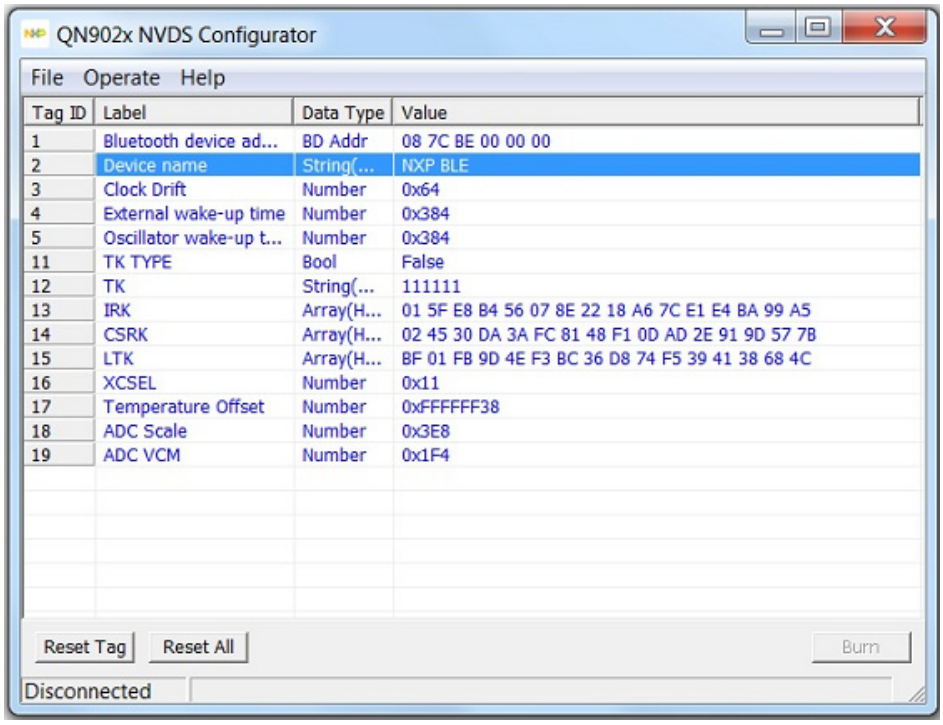

Fig 23. NVDS configurator main window

The values can be changed according to specific requirement. However, the default value is recommended.

Table 1. Important labels

Labels	Description
Bluetooth device address	48-bit Bluetooth device address.
Device name	name of the BLE device to differentiate from other Bluetooth devices.
Clock drift	while using 32.768 kHz crystal oscillator, set it according to the specification of crystal used; recommended value is 100 ppm, to have margin for load variation and temperature change; while using 32 kHz RC oscillator, set it to 500 ppm.
External wake-up time/Oscillator wake-up time	900 µs (0x384) is recommended by default; while using 32 K low-power mode, it should be set as 3000 µs (0xBB8).
XCSEL	cap load of 16 MHz crystal oscillator; default value is 0x11; adjust according to the specific PCB design and crystal used in real applications.
ADC scale/ADC VCM	both parameters are for ADC and their values may be different for each chip; values are tested and calibrated while manufacturing.

5.2.1.4 Step 4:

Click value field of device name record and change the device name; see Figure 24.

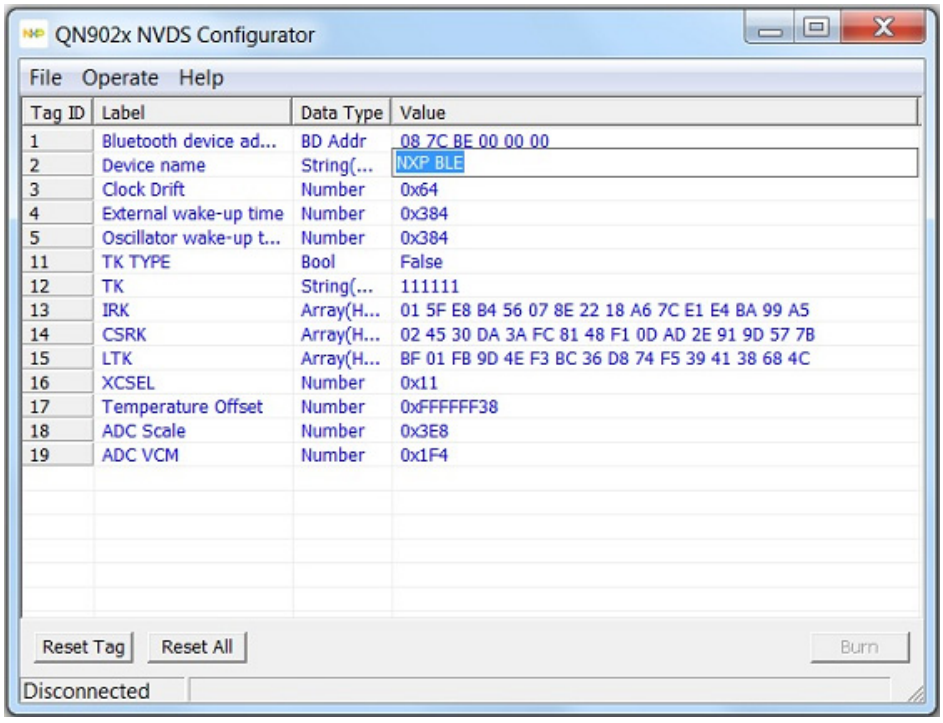

Fig 24. Change device name

5.2.1.5 Step 5:

Click Burn button and write parameters back to QN9020; see Figure 25.

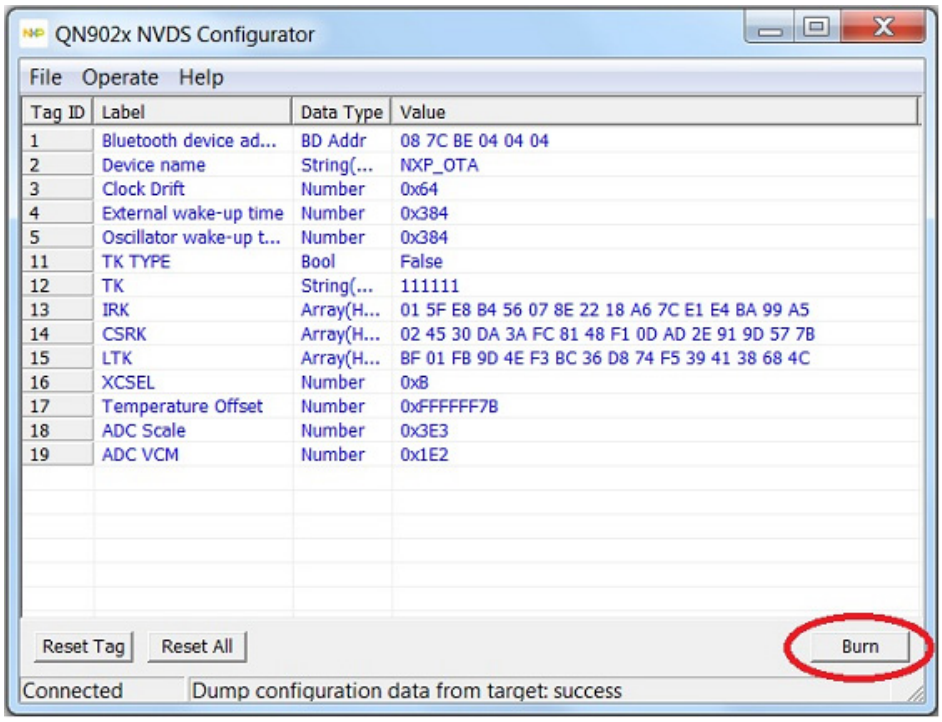

Fig 25. Burn configuration

5.3 Connection

When the mini DK board is powered-on or reset, it is in deep sleep mode. In this mode, LED2 is ON. The following are the LED indications:

- LED1 ON: It indicates that the application is connected. If it is flickering, it means that it is advertising.
- LED2 ON: It indicates that QN9020 is in the deep sleep state.

5.3.1 Steps to set up the connection between the mini DK board and BLE dongle

5.3.1.1 Step 1:

To switch the board into advertising mode and make it discoverable, press BUTTON1 as shown in Figure 26. The LED1 now starts flickering. The flashing frequency depends on the advertising interval. For the first 30 seconds, the application uses a short advertising interval. After 30 seconds, LED1 flashing frequency will be lower as the advertising interval has increased to around 1 sec. If BUTTON1 is pressed again, the application stops advertising and LED1 stops flashing. The application is back to deep sleep state.

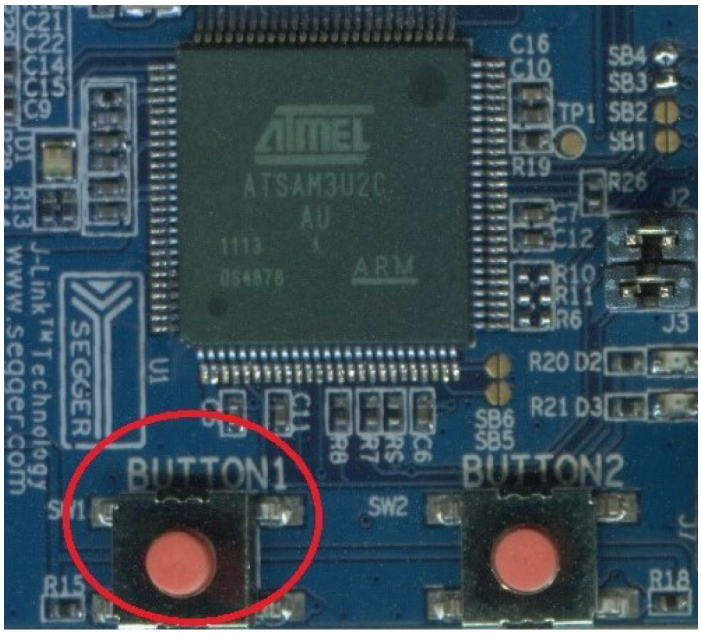

Fig 26. BUTTON1

5.3.1.2 Step 2:

Step 2: Start QTool from QN902x Studio start page. Figure 27 shows the message that pops up for communication setting.

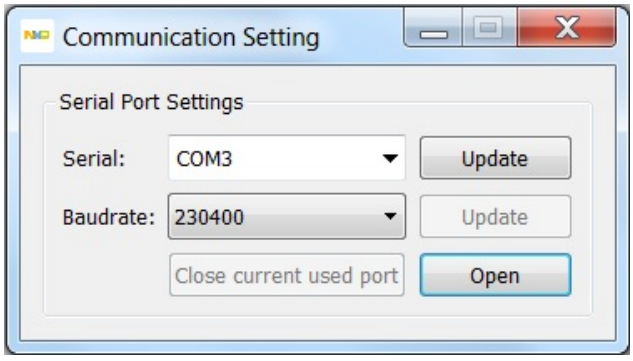

Fig 27. Connection setting in QTool

5.3.1.3 Step 3:

Select the correct COM port for BLE dongle. It can be checked with the device manager. Press “Open” button in the dialog box. Figure 28 shows the resultant window. The device representing BLE dongle would be in idle state by default.

5.3.1.4 Step 4:

Select the item in “Devices”. It shows the “Settings” area.

5.3.1.5 Step 5:

To switch to advertising mode, press BUTTON1 on mini DK board and press scan button shown in Figure 28. The discovered devices are now shown in “Devices” area.

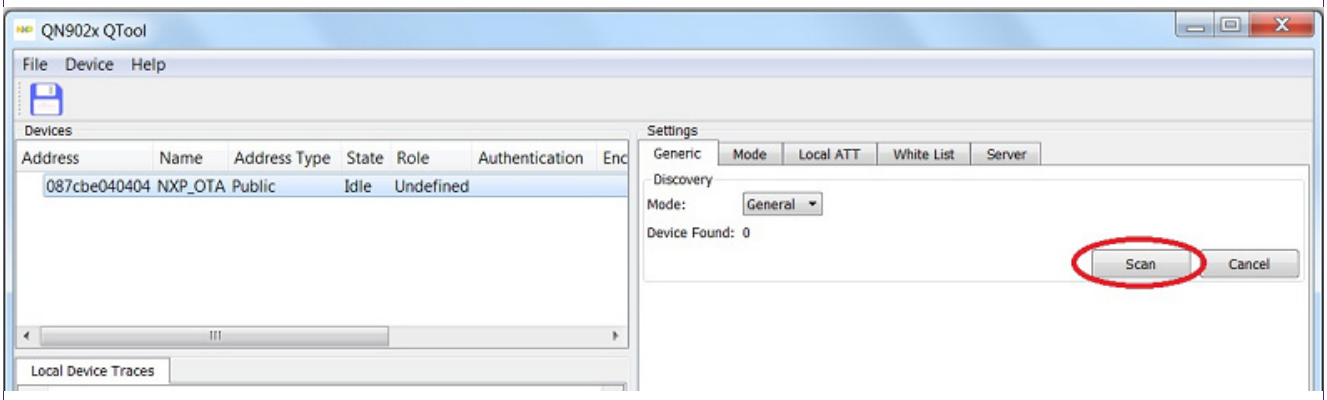

Fig 28. QTool opened

5.3.1.6 Step 6:

To initiate a connection, select the mini DK from the scan result and click the “Connect” button in settings block (see Figure 29).

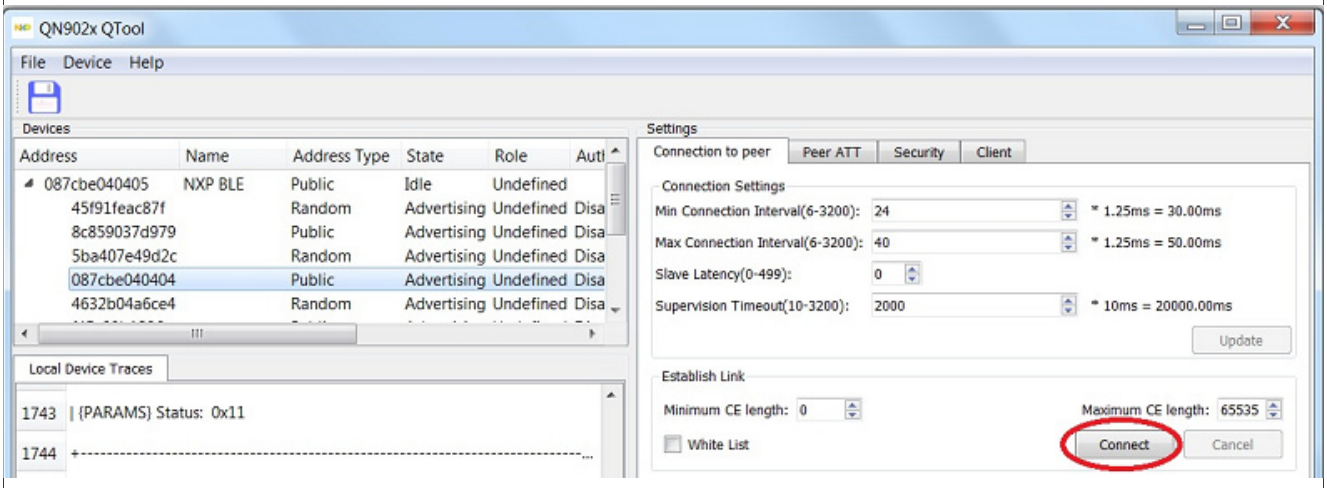

Fig 29. Connection in QTool

The role of BLE dongle changes to master and the mini DK board acts as slave. The state of both devices is connected as shown in Figure 30.

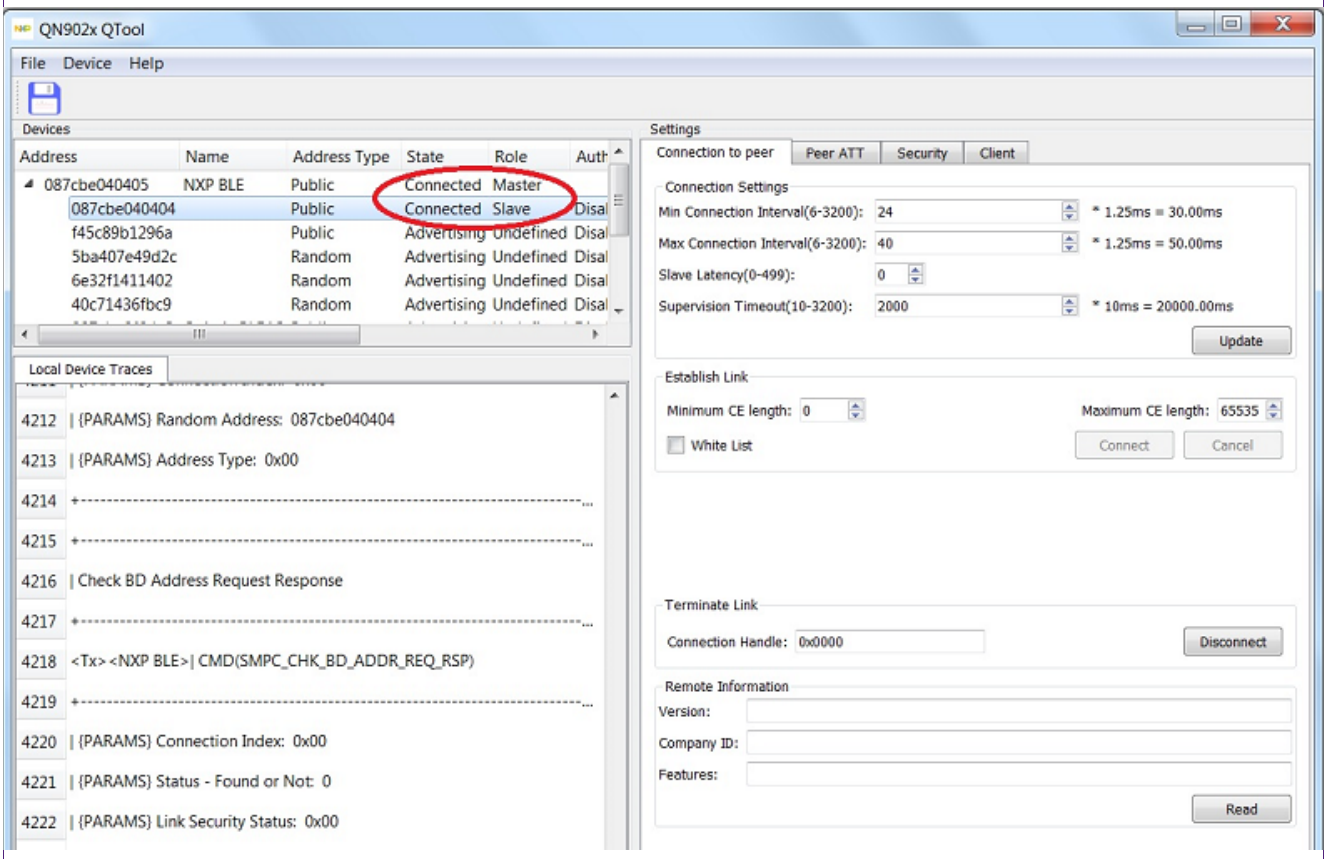

Fig 30. Devices connected

5.4 Function operation

Various interactions between mini DK board and BLE dongle can be performed when connected.

5.4.1 Find me

Steps to simulate “to find the mini DK board” are discussed below.

5.4.1.1 Step 1:

Click “Client” tab and “PROXM” tab in settings area (see Figure 31).

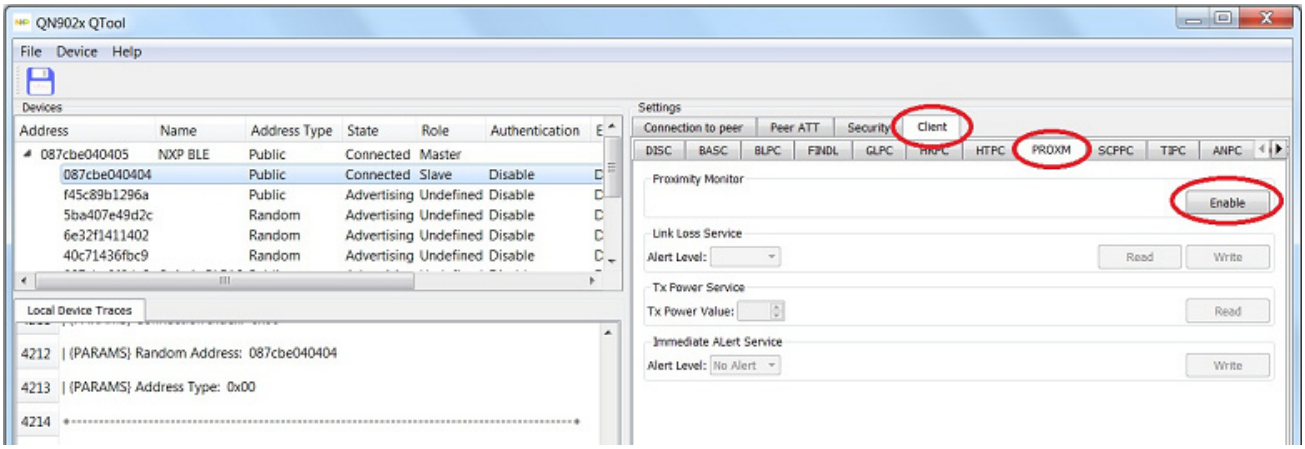

Fig 31. PROXM app discovered

5.4.1.2 Step 2:

To make discovered services editable, click the “Enable” button.

5.4.1.3 Step 3:

Set the alert level in the “Immediate Alert Service” area to medium or high.

5.4.1.4 Step 4:

Click “Write” button. The mini DK board sounds an alert with the help of a buzzer. Proximity alerts are indicated using the buzzer. The volume of the buzzer indicates the type of alert.

5.4.2 Disconnect

There are two options to disconnect the RF link between two devices.

1. Click “disconnect” button in “Connection to peer” tab; see Figure 30. The device disconnects with no response.
2. If the two devices are moved away from each other, the RF link may break. In such a case, the mini DK board responds with an alert.

5.4.3 Others

When buzzer is ON, press BUTTON2 to turn-off the buzzer.

6. Advanced application development

According to specific requirement, it may be necessary to add some peripheral functions such as button, LED, and buzzer. Useful tools are provided to help with implementation. It is easier to implement specific functions observing the sample codes in QN902x SDK.

6.1 Add driver file

Suppose that it is required to implement button control, which needs GPIO definition. It is easy to implement it by using the Keil development tool. By default, there is a folder named “driver” at *C:\NXP\QN902x_SDK_x.x.x\Projects\BLE\src*. All peripheral function source codes are included here. The relevant driver files can be added to the project in Keil, as shown in Figure 32.

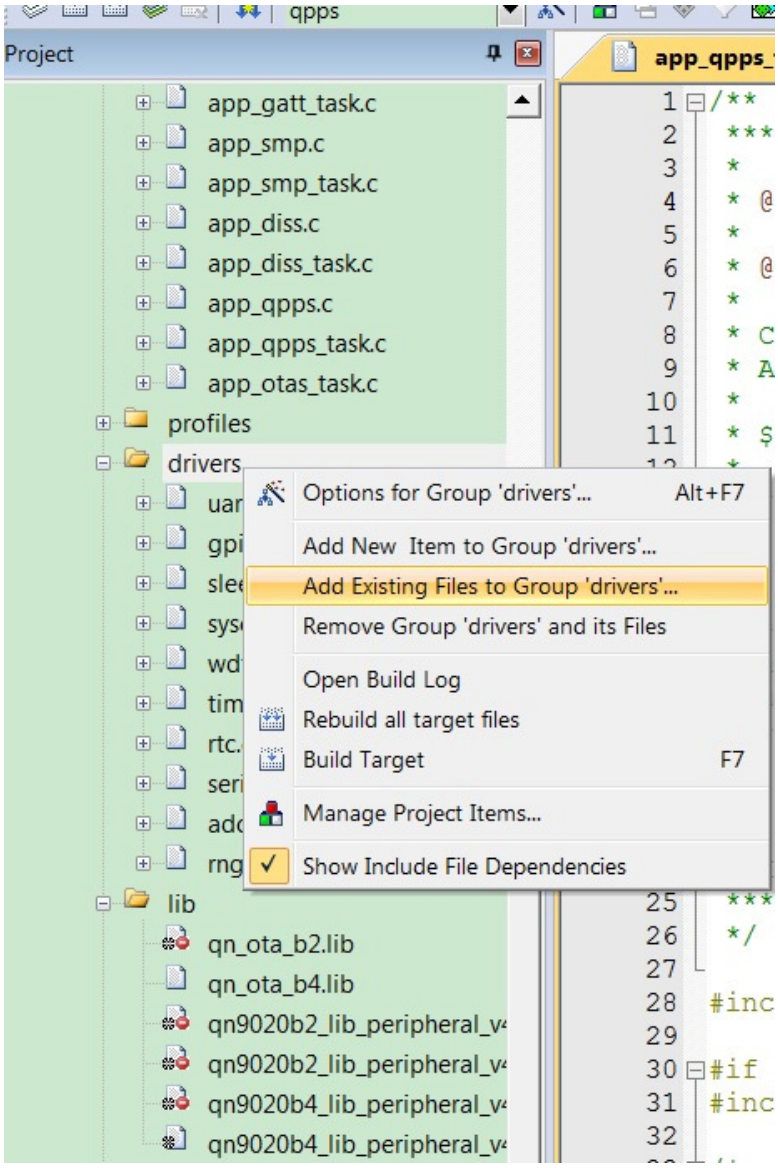

Fig 32. Adding a driver file

To study how to use the peripheral functions, open the dedicated peripheral driver files, typically located in “C:\NXP\QN902x_SDK_x.x.x\Projects\Driver”. Open the required project and directly download to the mini DK board. Modify it according to the specific requirements.

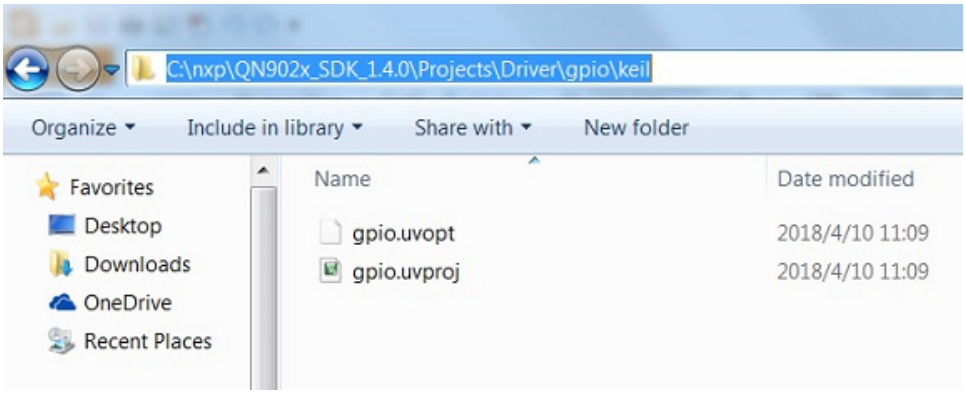

Fig 33. Adding a driver file

Double-click “gpio.uvproj” and open it using Keil; see Figure 34.

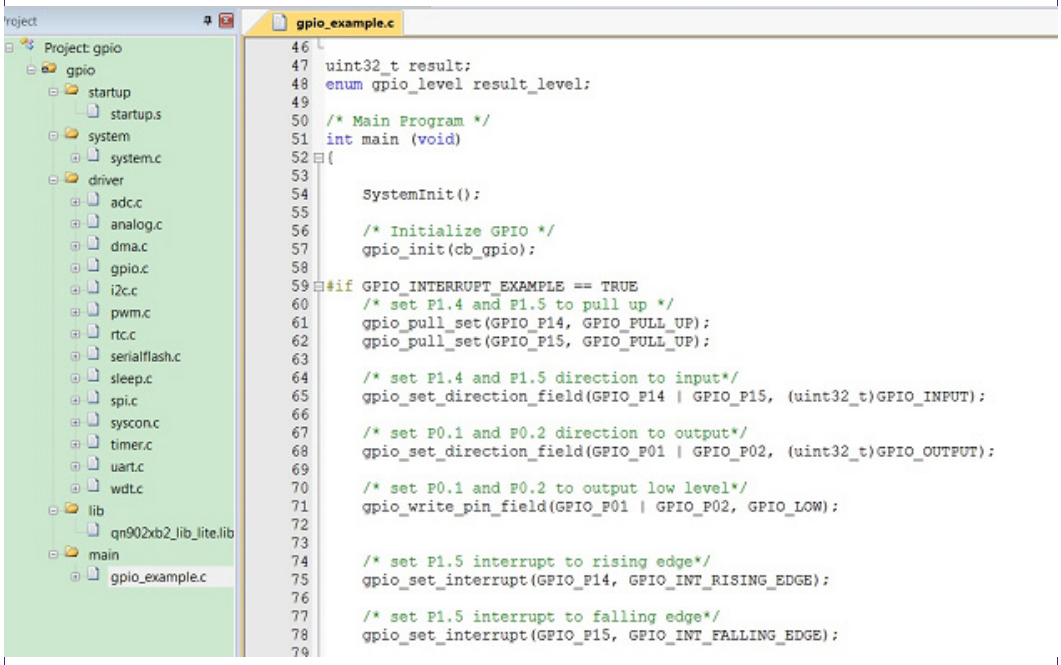

Fig 34. GPIO example

The main function is in gpio_example.c.

6.2 GPIO MUX configuration

The GPIO pins are shared with digital or analog peripherals such as SPI, UART, I^2C, and ADC. The function of IO can be configured in QN902xDriverTool, which generates the source code for the specific assignment of GPIO pins.

6.2.1 Change GPIO function

6.2.1.1 Step 1:

Open “QN902xDriverTool” and select the right device; see Figure 35.

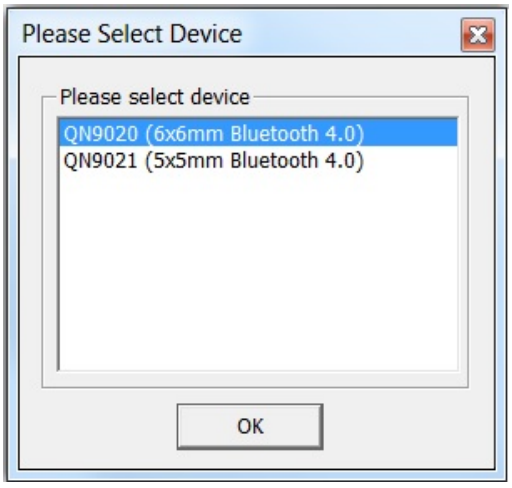

Fig 35. Device selection

6.2.1.2 Step 2:

Choose IO icon; see Figure 36.

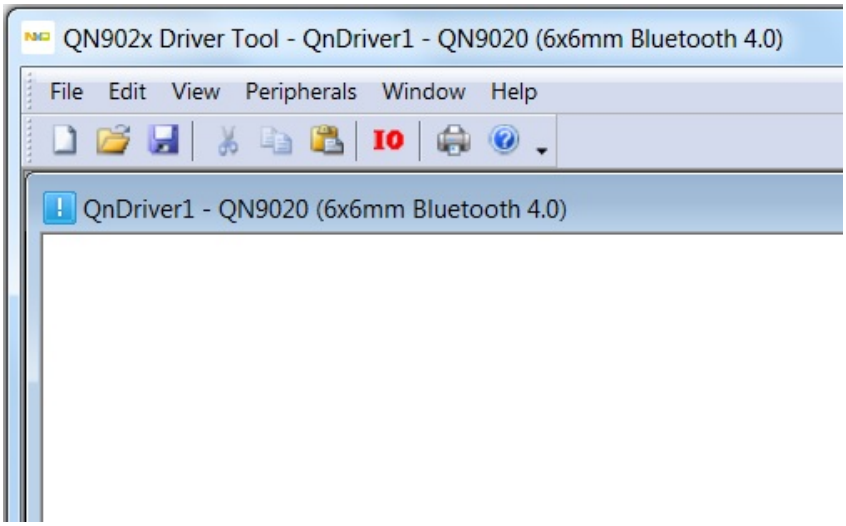

Fig 36. QN902xDriverTool

6.2.1.3 Step 3:

Select P2.6 and P2.6 as PWM function. To switch to P2.6, check the box at the left of pwml and click the refresh icon.

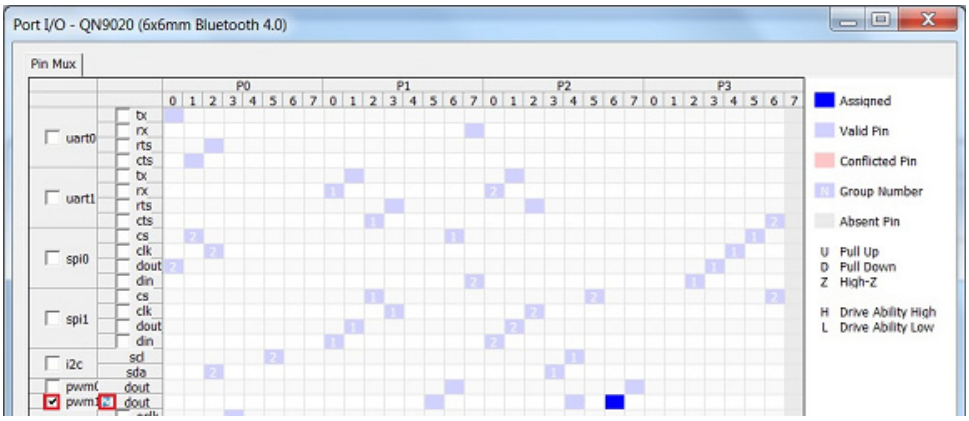

Fig 37. Setting IO

6.2.1.4 Step 4:

After the setting is completed, click OK button at the bottom right of the “QN902xDriverTools” window. The source code is generated automatically; see Figure 38.

```
//Pin Mux Control Register
syscon_SetPMCR0(QN_SYSCON, P00_GPIO_3_PIN_CTRL
                         | P03_GPIO_2_PIN_CTRL
                         | P05_GPIO_15_PIN_CTRL
                         | P06_SW_DAT_PIN_CTRL
                         | P07_SW_CLK_PIN_CTRL
                         | P10_GPIO_8_PIN_CTRL
                         | P11_GPIO_7_PIN_CTRL
                         | P12_GPIO_6_PIN_CTRL
                         | P13_GPIO_5_PIN_CTRL
                         | P17_GPIO_4_PIN_CTRL
                         | P23_GPIO_12_PIN_CTRL
                         | P24_GPIO_11_PIN_CTRL
                         | P26_PWM1_PIN_CTRL
                         | P27_GPIO_9_PIN_CTRL
                         | P31_GPIO_14_PIN_CTRL
                         | P32_GPIO_13_PIN_CTRL);

//Pin Mux Control Register
syscon_SetPMCR1(QN_SYSCON, P01_GPIO_18_PIN_CTRL
                         | P02_GPIO_17_PIN_CTRL
```

aaa-018439

Fig 38. Created code for IO configure

6.2.2 Add PWM function to project

Open “system.c” which is at “C:\NXP\QN902x_SDK_x.x.x\Projects\BLE\prj_proxr\src” by default.

Find “P26_GPIO_22_PIN_CTRL” in function “SystemIOCfg” and replace it with “P26_PWM1_PIN_CTRL”. Save this file.

After adding above code to current project, compile the project, and download it to mini DK board to confirm the functionality.

Note:

1. For more information about how to enable related module, refer to example code at “C:\NXP\QN902x_SDK_1.4.0\Projects\Driver\adc\keil” .

2. For more information on “QBlueDriverTools”, refer to *“QN902x Driver Tools Manual”*.

7. Trouble shooting

1. Bin file download failed while using QN902xISPStudio; see Section 5.1.1.5.
2. Bin file download failed while using Keil: Set the board in active mode and download again; see Section 5.1.2.
3. Error messages pop up while downloading application to mini DK board: To wake up the board and download again, press BUTTON1 or BUTTON2; see Section 5.1.2.

8. Abbreviations

Table 2. Abbreviations

Acronym	Description
ACI	Application Control Interface
BLE	Bluetooth Low Energy
DK	Development Kit
EACI	Easy Application Control Interface
GPIO	General Purpose Input Output
ISP	In System Programming
SDK	Software Development Kit
SMP	Security Management Protocol
SoC	System-on-Chip
SPI	Serial Port Interface
SWD	Serial Wire Debug
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus

9. References

[1] **QN9020 Easy ACI Programming Guide —** Programming guide

[2] **QN902x QTool User Manual —** User manual

[3] **QN902x ISP Studio Manual —** Manual

[4] **QN902x NVDS Configuration Manual —** Configuration manual

[5] **QN902x Driver Tools Manual —** Tools manual

[6] **QN902xDevDBforIDE User Manual —** User manual

[7] **QN902x Software Developers Guide —** Developers guide

[8] **QN9020_mini_DK_User_Guide —** User guide

[9] **http://www.keil.com/uvision/debug.asp —** Keil website

[10] **http://www.segger.com —** Segger website

[11] **http://www.keil.com —** Keil website

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 17 April 2018
Document identifier: AN11664

10. Legal information

10.1 Definitions

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

10.2 Disclaimers

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

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

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

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

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

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

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

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

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

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

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

10.3 Trademarks

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

I²C-bus — logo is a trademark of NXP B.V.

11. Tables

12. Figures

13. Contents