

HY0022

Software Development Guide

- The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by FDK CORPORATION is under license. Other trademarks and trade names are those of their respective owners.
- FDK Corporation is granted a license to use Slot Antenna on Shielded Package Technology patented by Toshiba Corporation.

Version	Date	Description
1.0	Dec.13, 2024	First edition issued

1 Overview

HY0022 is an evaluation board for software (firmware) development and evaluation with the ultra-compact Bluetooth® Low Energy module HY0020 with integrated nRF52832 from Nordic Semiconductor Corporation and Toshiba Corporation's Slot Antenna on Shielded Package technology. This document describes how to set up the software development environment using HY0022, and how to build, write, and debug sample source code.

2 Hardware Requirement

Hardware requirements (recommended environment) are listed below. Devices other than those listed below that meet equivalent performance requirements can also be used.

- PC with Windows® 10 or 11
- J-Link debugger (J-Link Base, etc.) manufactured by SEGGER Corporation and USB cable or flat cable for connection
- HY0022 evaluation board and USB cable for connection
- Smartphone (for Bluetooth® Low Energy operation check)

3 Development Environment

As mentioned above, the HY0020 mounted on the HY0022 has a built-in Bluetooth® IC (nRF52832) manufactured by Nordic Semiconductor, so the following two software development environments are available (free of charge).

- nRF5 SDK
- nRF Connect SDK

Please review the license of each SDK and choose one based on your current holding environment and usage history.

3.1 Building the nRF5 SDK Development Environment

The development of the nRF5 SDK by Nordic Semiconductor has been stopped since Bluetooth® Ver.5.2 and will not be maintained or updated in the future. Therefore, the SoftDevice, the Bluetooth® Low Energy core stack used in the nRF5 SDK, is Bluetooth® Ver.5.2 compliant.

1. Installing the nRF5 SDK
Choice "v17.1.0 nRF5 SDK", which is the final release version, from the following URL, download it and extract it to the desired location.
(The SoftDevice is included in the downloaded SDK.)
<https://www.nordicsemi.com/Products/Development-software/nRF5-SDK/Download>
2. Installing the SEGGER Embedded Studio
Download and install v5.66 from the Windows 64bit "Embedded Studio for ARM (Legacy)" installer at the following URL.
<https://www.segger.com/downloads/embedded-studio/>

3.2 Building the nRF Connect SDK Development Environment

The nRF Connect SDK environment also supports Bluetooth® Ver.5.4 and new features such as Mesh. If you are planning to build a new development environment, we recommend that you use this environment.

1. nRF Command Line Tools

Download and install the latest version of nRF Command Line Tools from the following URL.

[nRF Command Line Tools - Downloads - nordicsemi.com](https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-SDK/Downloads)

2. Visual Studio Code

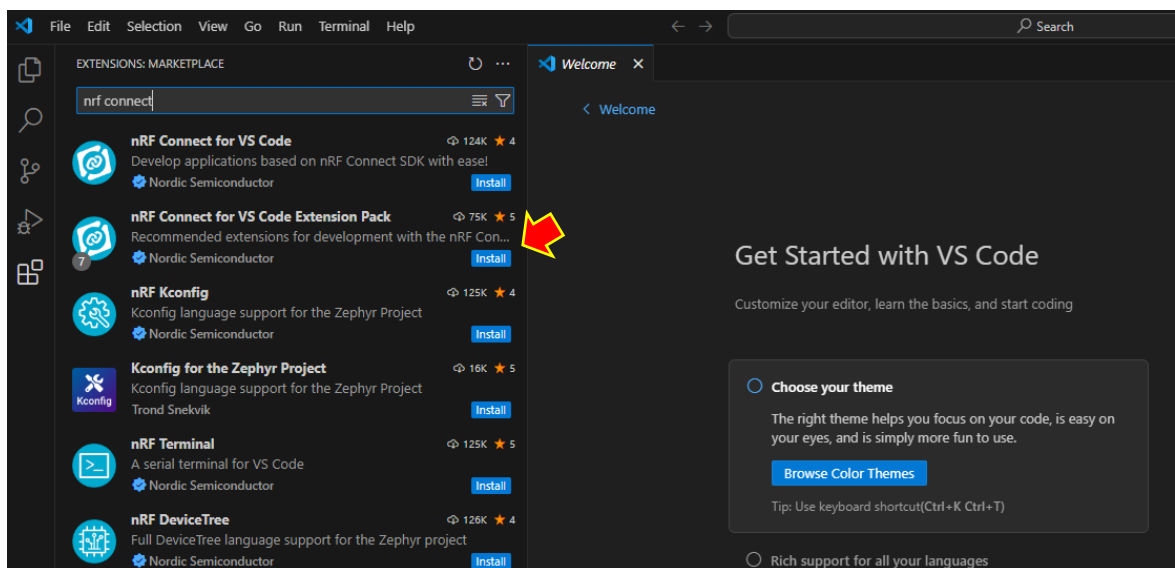
Download and install the latest version of Visual Studio Code (VS Code) from the following URL.

[Download Visual Studio Code - Mac, Linux, Windows](https://code.visualstudio.com/download)

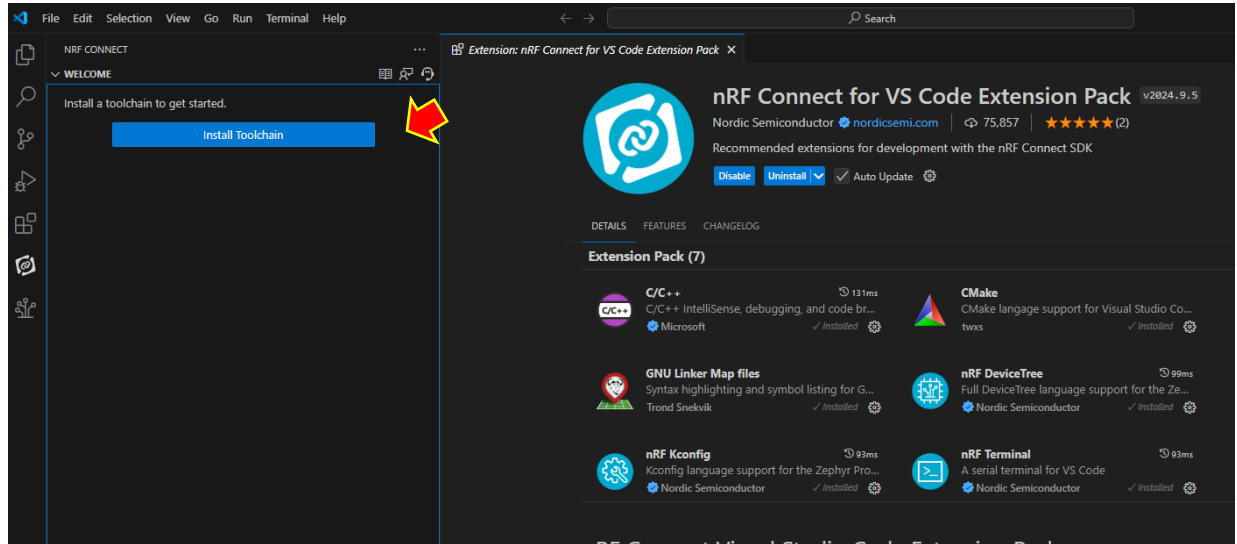
3. nRF Connect SDK

Note: The following installation procedure can be referred to the installation video (Installing nRF Connect SDK and nRF Connect for VS Code : [Get Started with nRF Connect SDK - nordicsemi.com](https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-SDK/Getting-started)) by Nordic.

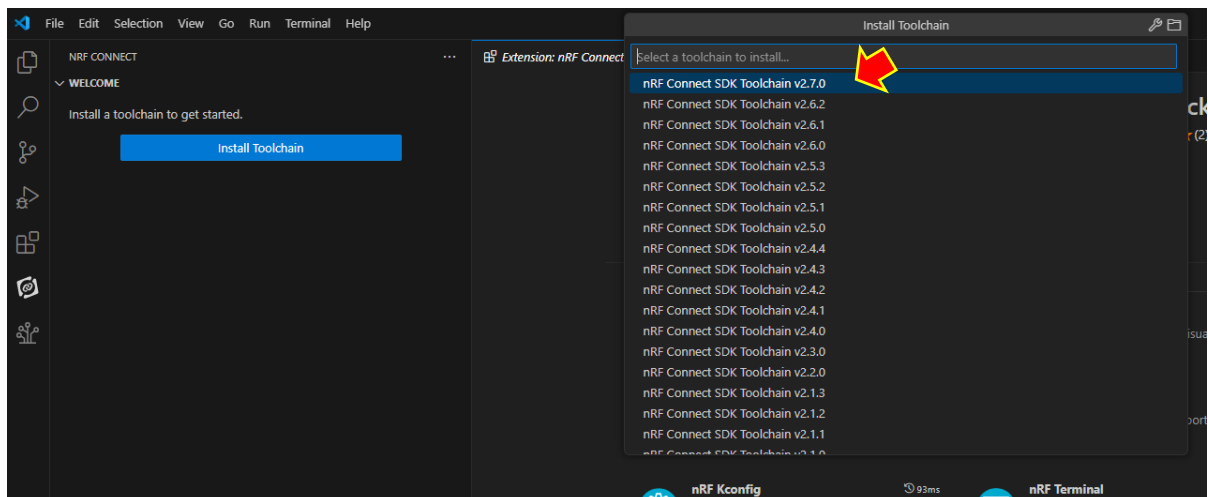
- (1) Launch VS Code and click on Extension in the left pane. Type “nrf connect” in the search box and click the Install button for the nRF Connect for VS Code Extension Pack.



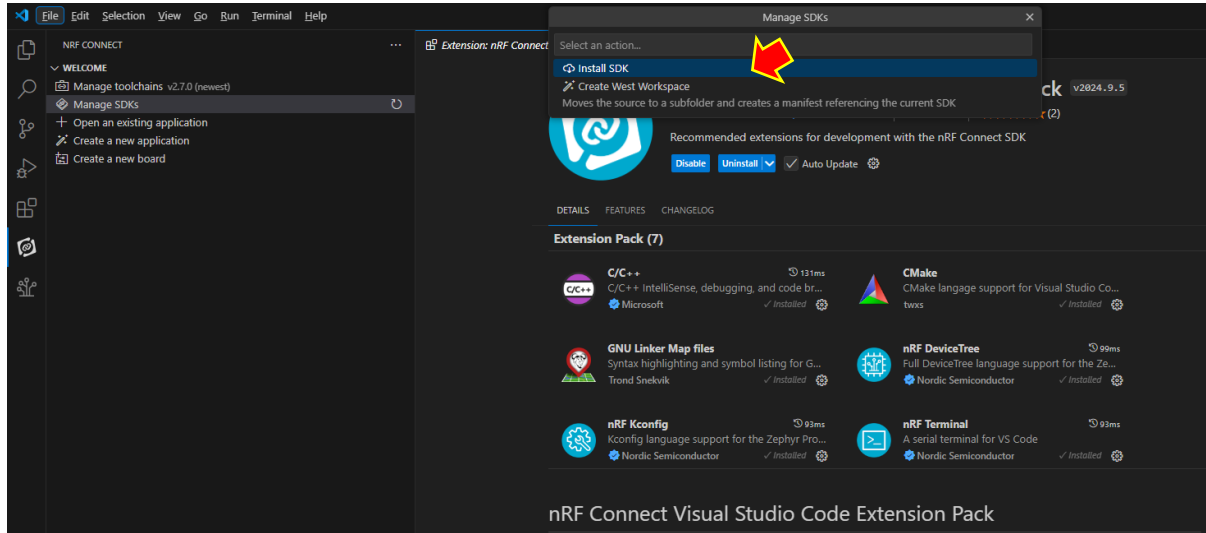
- (2) After the installation is complete, click the nRF Connect icon in the left pane, and then click the Install Toolchain button.



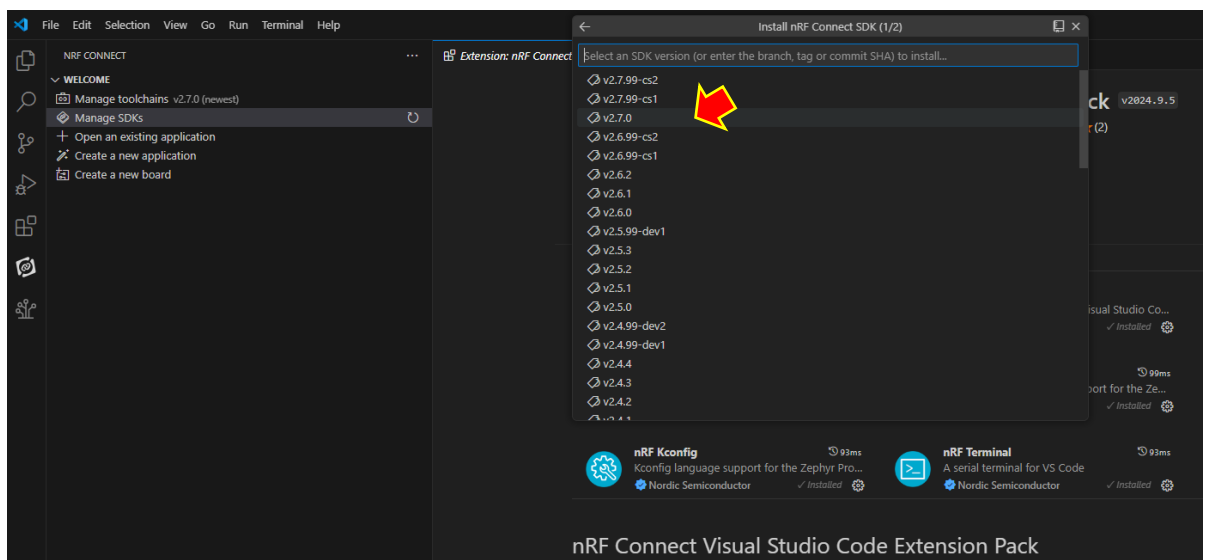
- (3) Click the latest version (e.g., nRF Connect SDK Toolchain v2.7.0) when the menu appears.



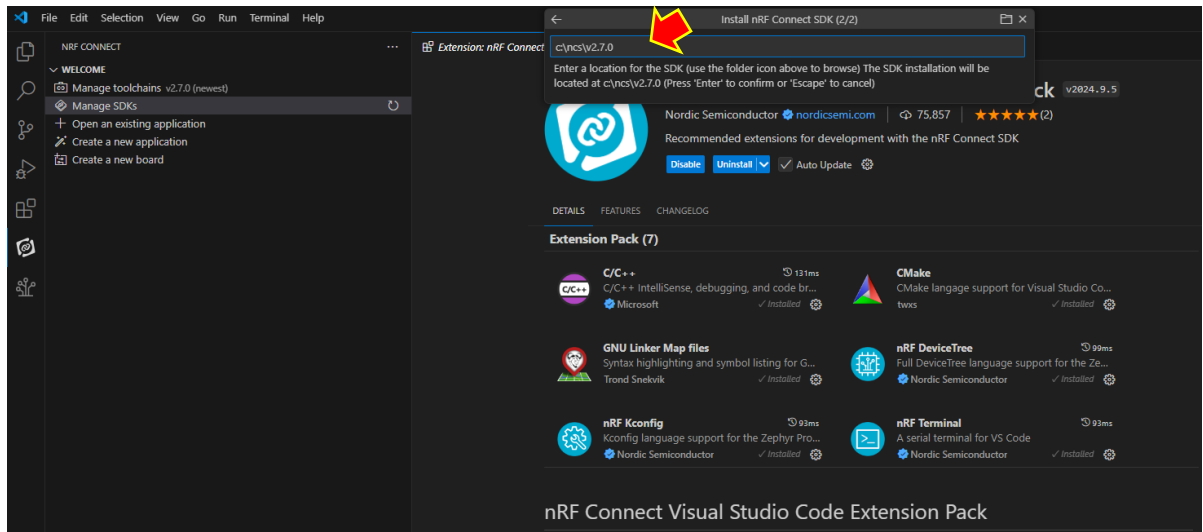
- (4) After the installation is complete, click “Manage SDKs”, and then click “Install SDK”, which is displayed in the center of the screen.



- (5) When the version is displayed, select the latest tag release version. In this case, select “v2.7.0”.



(6) Select the installation location. The default installation location is left as it is here.



3.3 Other necessary software

HY0022 is equipped with a USB-UART converter IC (FT232RNQ) manufactured by FTDI for UART communication, so the following USB driver must be installed beforehand.

- Driver for FT232R

<https://ftdichip.com/drivers/d2xx-drivers>

Please download the driver from the setup executable link in the URL above.

4 Project Example

4.1 Implementation – nRF5 SDK

Note: The following description assumes that the SDK installation folder has been extracted and placed in “c:\”.

4.1.1 Sample code

This document describes the sample code “ble_app_uart” for checking UART communication via Bluetooth® Low Energy as an example.

◆ Open IDE-Project

Start SEGGER Embedded Studio, select the following file from the menu “File” -> “Open Solution” and open it.

C:\nRF5_SDK_17.1.0_ddde560\examples\ble_peripheral\ble_app_uart\pca10040\s132\ses\ble_app_uart_pca10040_s132.emProject

4.1.2 SoftDevice

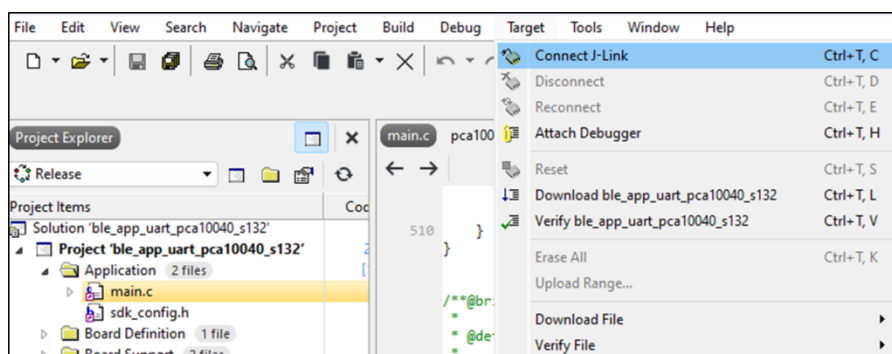
When using SoftDevice as Bluetooth® protocol stack, “S132 SoftDevice v7.2.0” for HY0020 (nRF52832) is stored in the SDK path below.

C:\nRF5_SDK_17.1.0_ddde560\components\softdevice\s132\hex

The SoftDevice other than S132 such as “S112 SoftDevice v7.2.0” are also included in the SDK, so please use them according to your FW specifications/applications. (S132 SoftDevice v7.2.0 is used in this document).

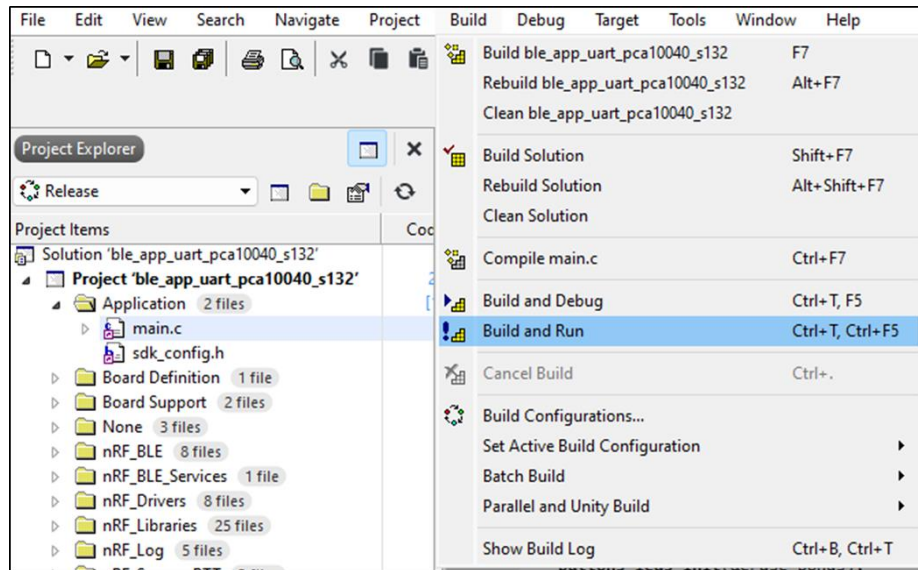
4.1.3 Connect Debugger

Connect the evaluation board to the J-Link debugger and perform “Connect J-Link” in the IDE.



4.1.4 Build and Run

Correct/update the files up to the previous section, confirm the connection, and execute “Build and Run” from the Build menu. After that, you can confirm the operation of this example (Bluetooth® Low Energy signal transmission) by plugging and unplugging the USB cable from the evaluation board or by rebooting the evaluation board with the reset switch on the board. For the reset switch on the evaluation board, refer to the manual that is provided with the evaluation board.



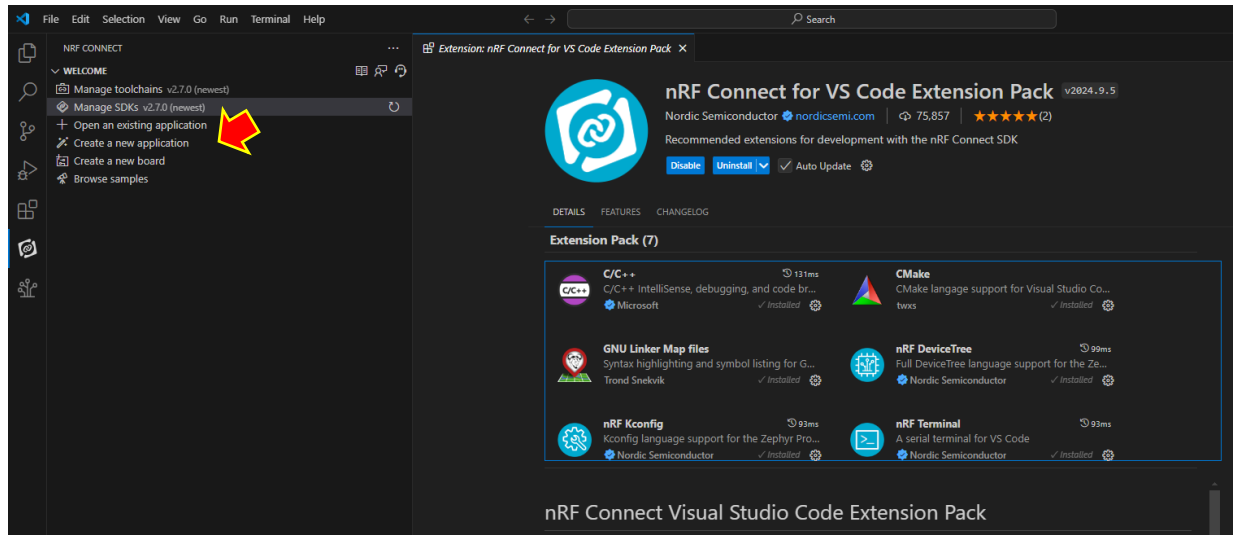
It is also possible to write the FW (.hex file) generated by the above build to the evaluation board by using the “Programmer” in the “nRF Connect for Desktop Tool”, which can be downloaded from [nRF Connect for Desktop - Downloads - nordicsemi.com](https://nordicsemi.com/nrf-connect-for-desktop-downloads).

4.2 Implementation – nRF Connect SDK

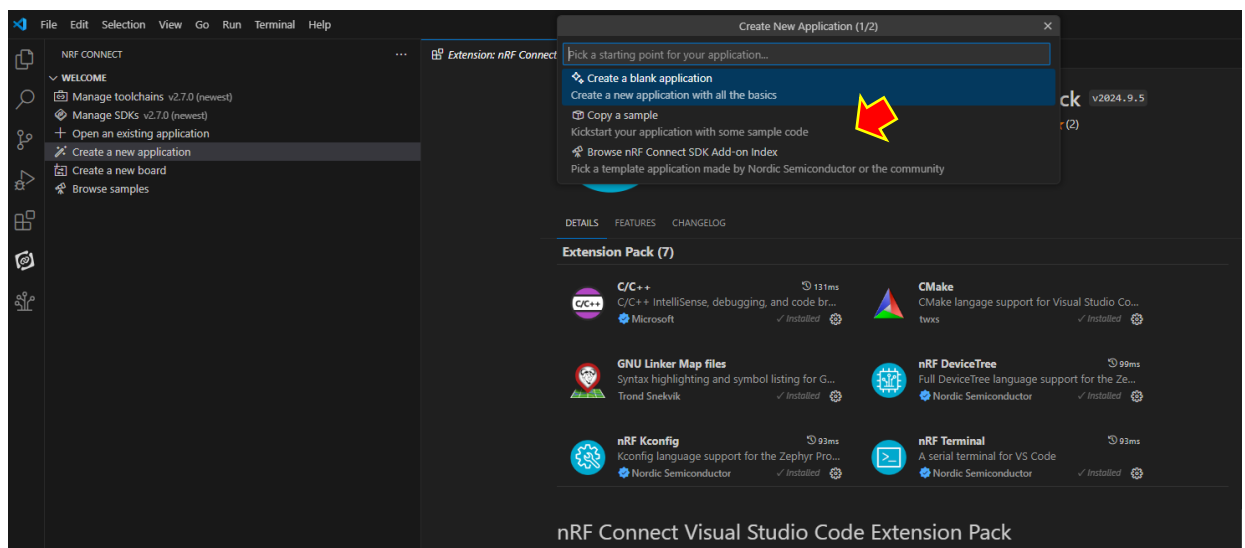
4.2.1 Sample code Build

The following shows the build to write procedure using “Bluetooth Peripheral UART” as an example to test the Bluetooth® Low Energy UART communication provided by the nRF Connect SDK.

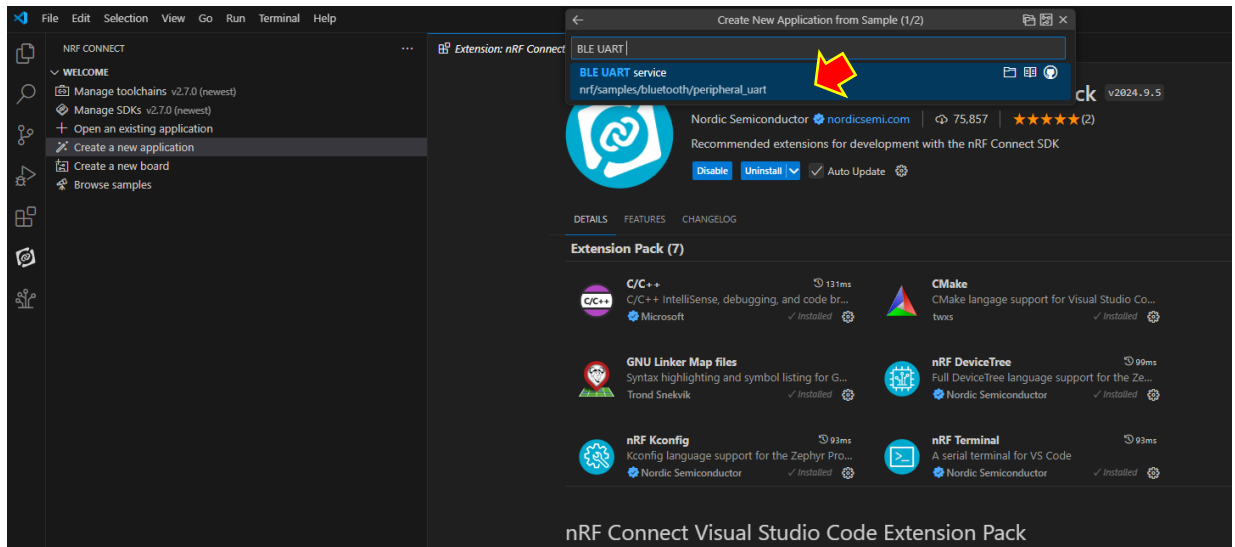
(1) Click “Create a new application” in the left pane of VS Code.



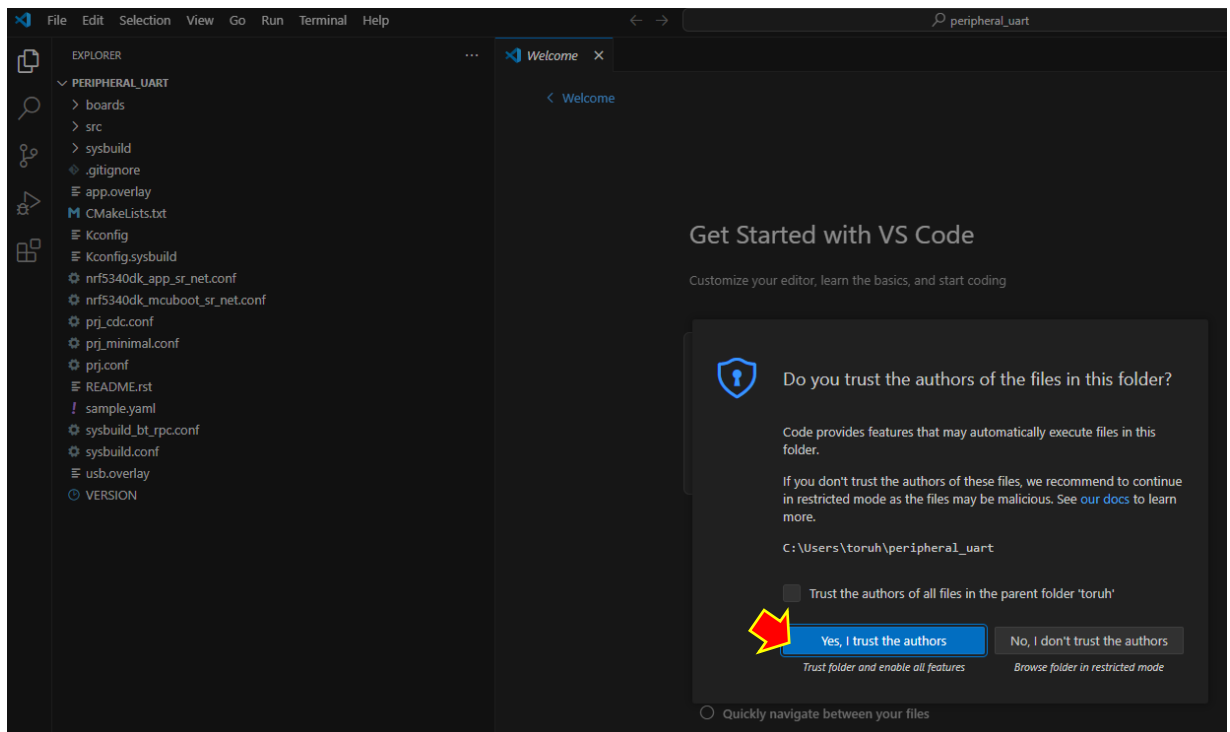
(2) Select “Copy a sample”.



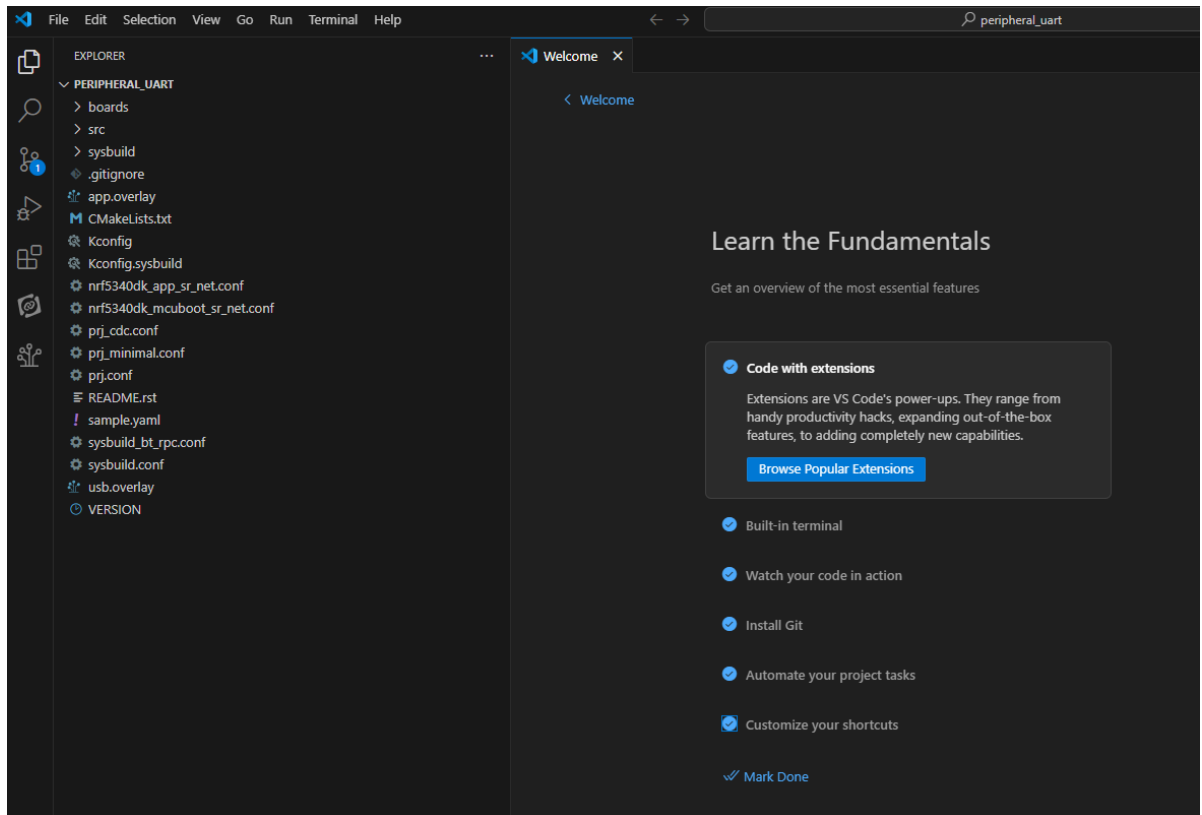
(3) Type "BLE UART" in the input text box and select "BLE UART service" as a candidate.



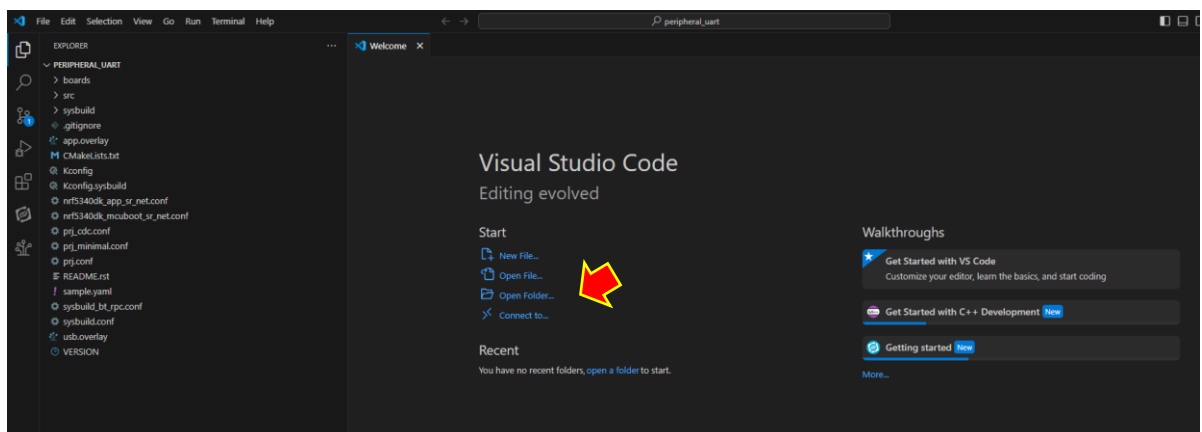
(4) Select "Yes" when asked "Do you trust the authors of the files in this folder?".



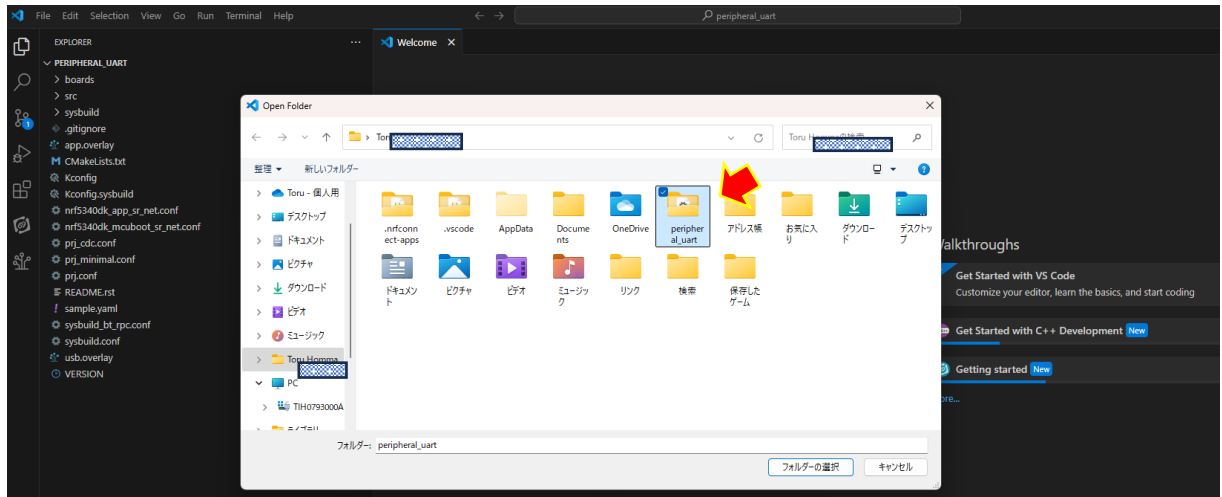
(5) Mark as appropriate and proceed.



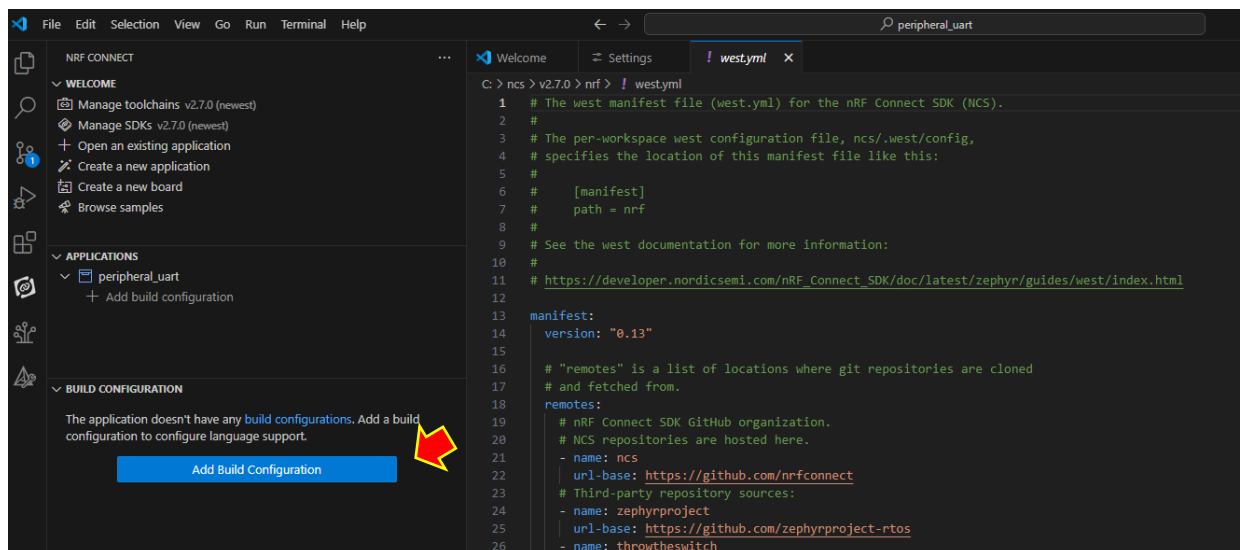
(6) Select “Open Folder...”.



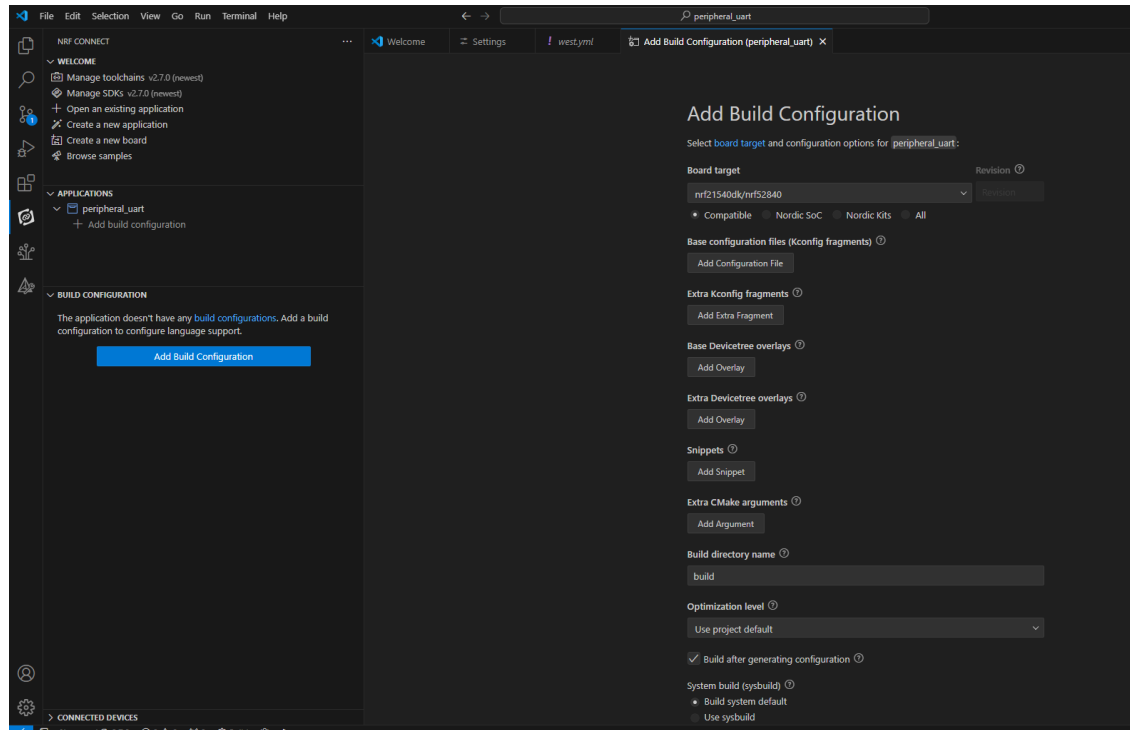
(7) Select the “Peripheral_uart” folder.



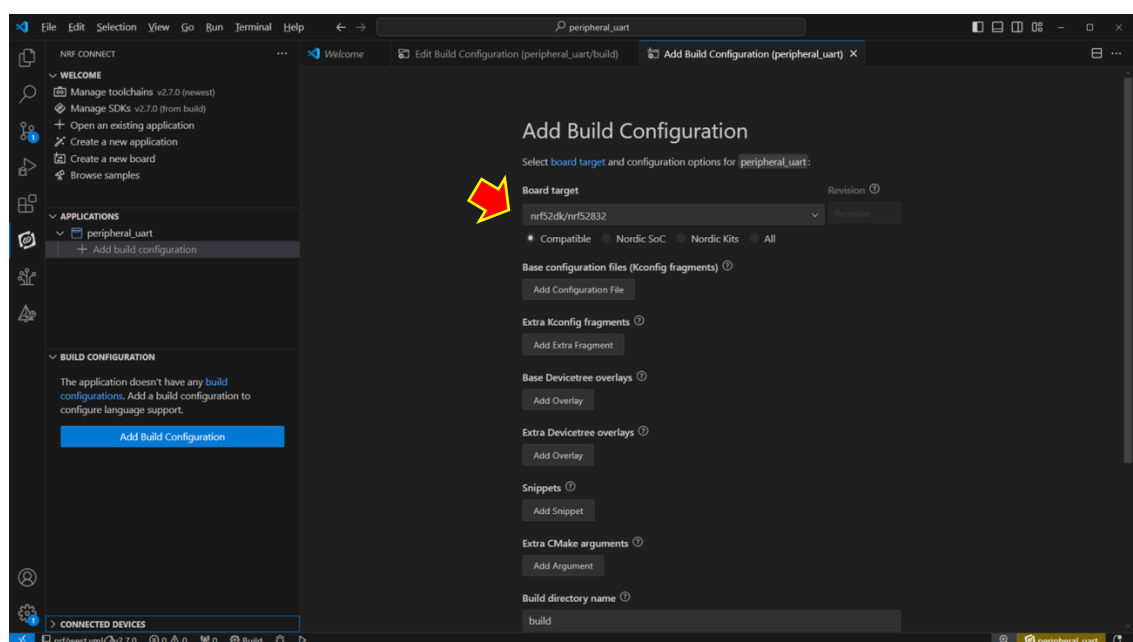
(8) Click “Add Build Configuration” in the left pane.



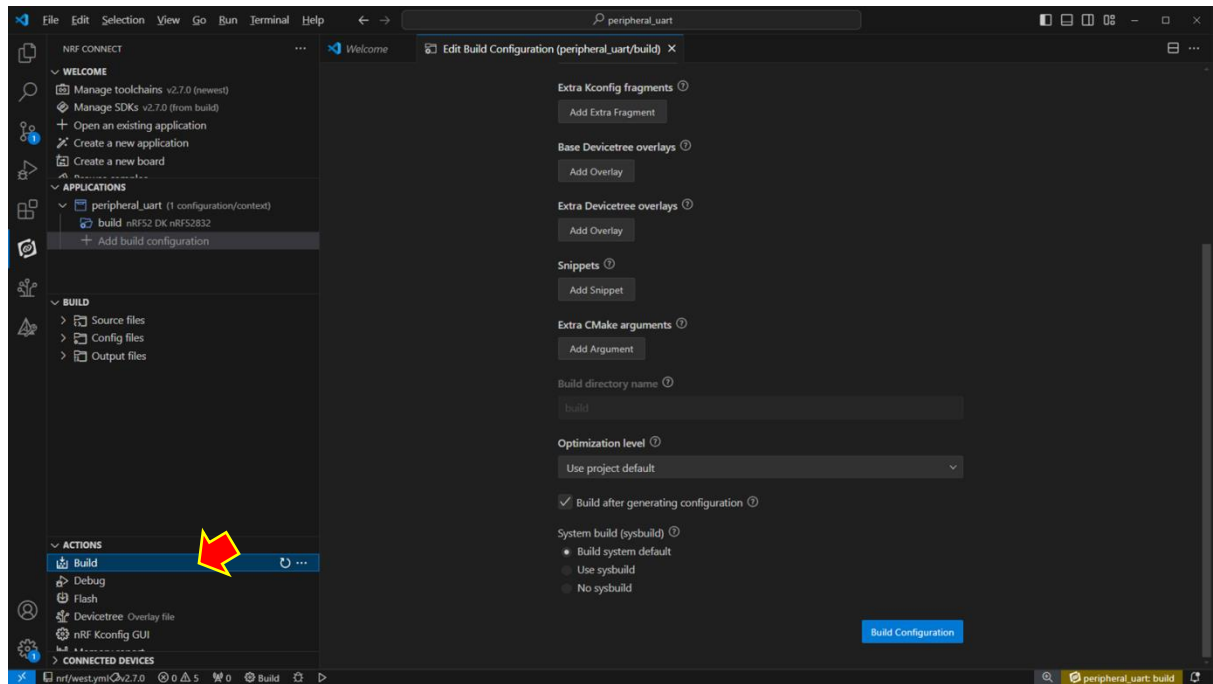
(9) The following will be displayed.



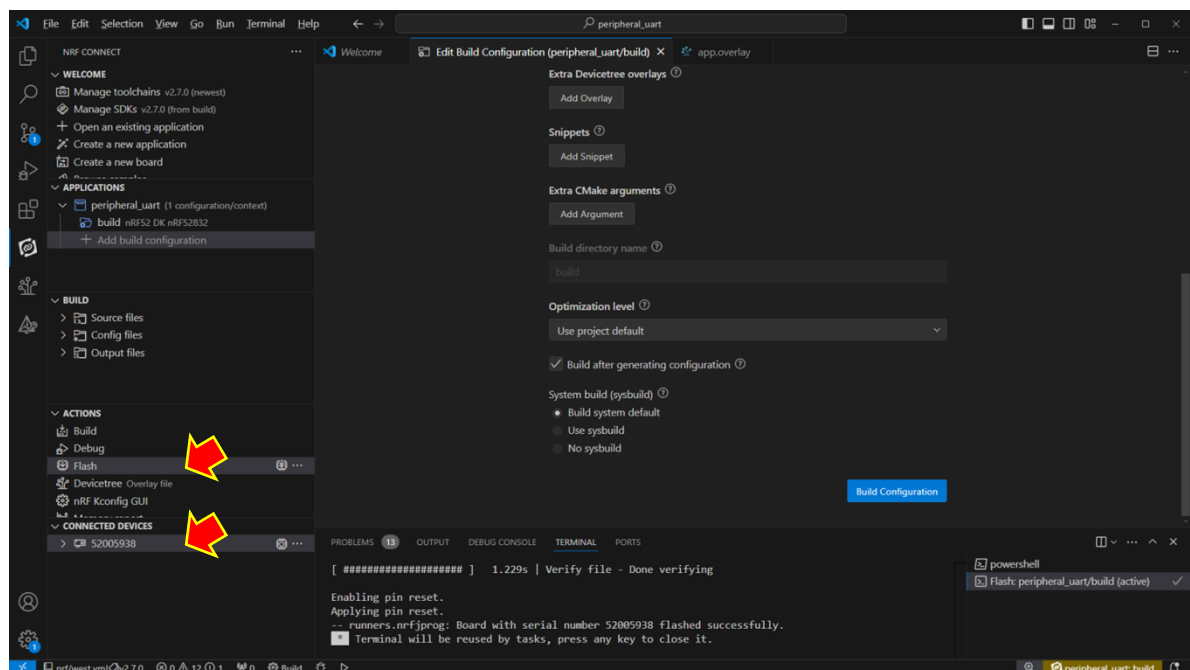
(10) Select Nordic SoC in the board target and choose “nRF52dk/nrf52832” from the pull-down menu. Scroll down and click “Build Configuration”.



(11) Click “Build” under Actions to build.



(12) Connect the J-Link Debugger and the HY0022 Evaluation Board to the PC via USB. Click on the node under CONNECTED DEVICES in the lower left corner to automatically detect the devices. Next, click on “Flash” under ACTIONS in the left pane and write FW to the evaluation board.



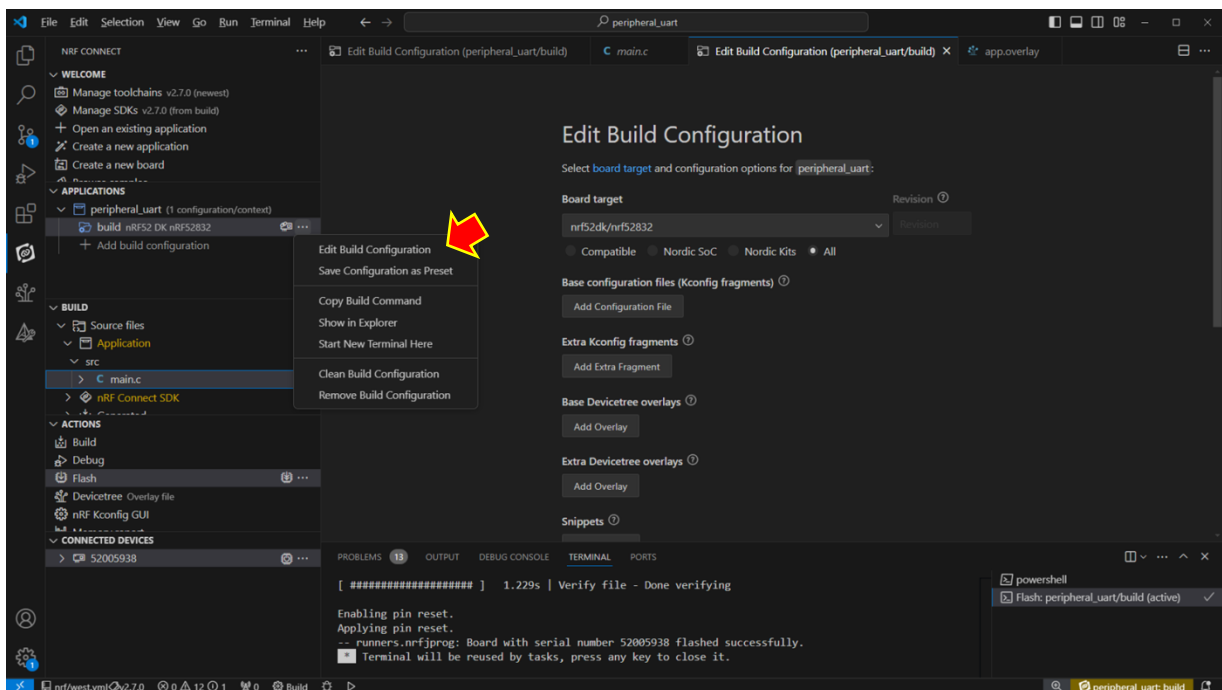
4.2.2 Debugging

Following the previous section, this section describes debugging operations using the “Bluetooth Peripheral UART” project to test the Bluetooth® Low Energy UART communication.

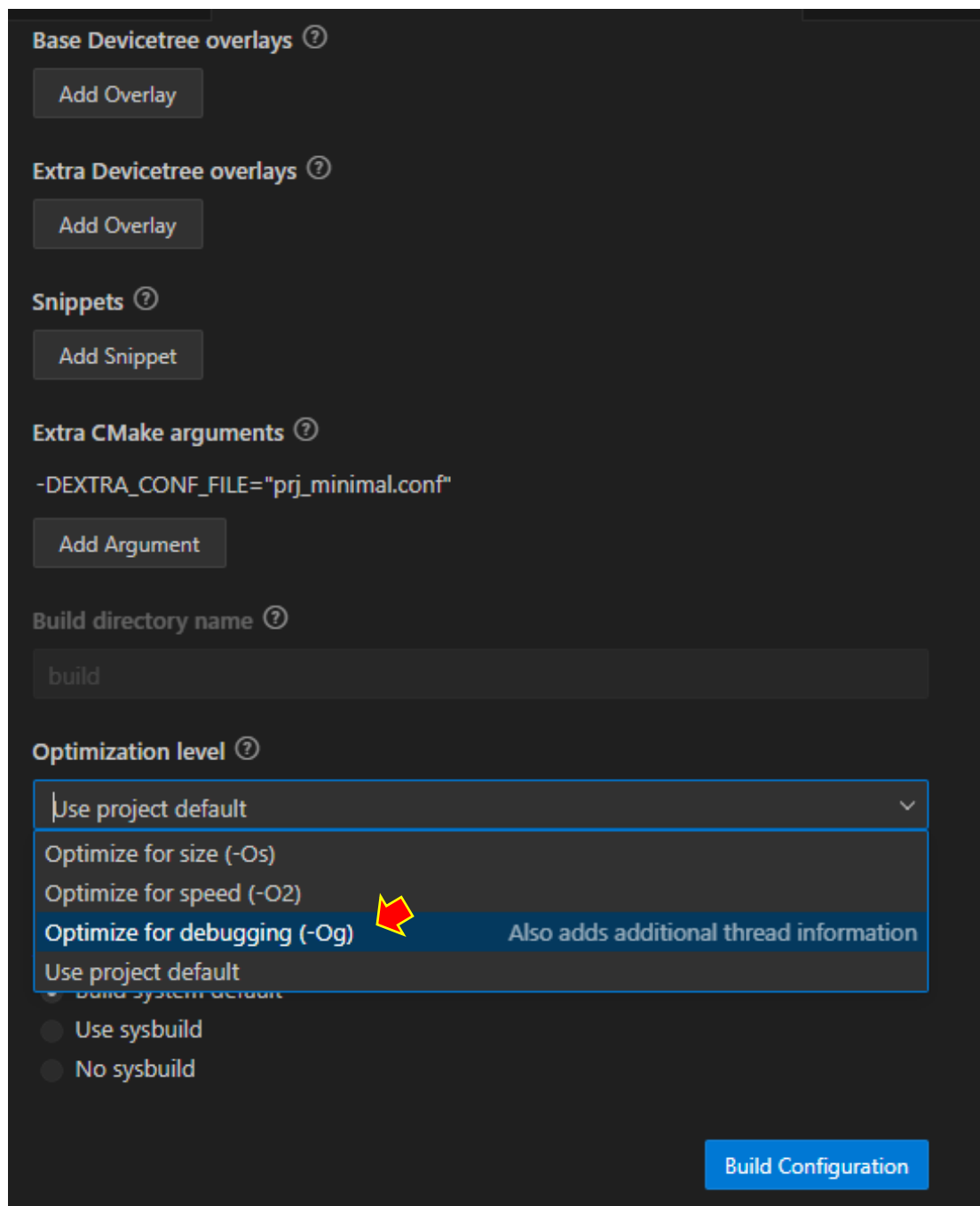
Note: The following debugging procedure is an example using a smartphone connection. Please install “nRF Connect for Mobile” from the following URL on your smartphone before starting the debugging procedure.

<https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-for-mobile>

- (1) Start VS Code, click “...” under build in the left pane under APPLICATIONS - peripheral_uart, and select “Edit Build Configuration”.



(2) Change the Optimization level to “Optimize for debugging (-Og)”.



The screenshot displays the FDK build configuration interface. It includes sections for Base Device tree overlays, Extra Device tree overlays, Snippets, Extra CMake arguments, Build directory name, and Optimization level. The Optimization level dropdown menu is open, showing options: Use project default, Optimize for size (-Os), Optimize for speed (-O2), Optimize for debugging (-Og), and Use project default. A red arrow points to the "Optimize for debugging (-Og)" option, which is highlighted in blue. To the right of this option, the text "Also adds additional thread information" is visible. Below the dropdown menu, there are radio buttons for "Build system default", "Use sysbuild", and "No sysbuild". A "Build Configuration" button is located at the bottom right.

Base Device tree overlays ?

Add Overlay

Extra Device tree overlays ?

Add Overlay

Snippets ?

Add Snippet

Extra CMake arguments ?

-DEXTRA_CONF_FILE="prj_minimal.conf"

Add Argument

Build directory name ?

build

Optimization level ?

Use project default

Optimize for size (-Os)

Optimize for speed (-O2)

Optimize for debugging (-Og) Also adds additional thread information

Use project default

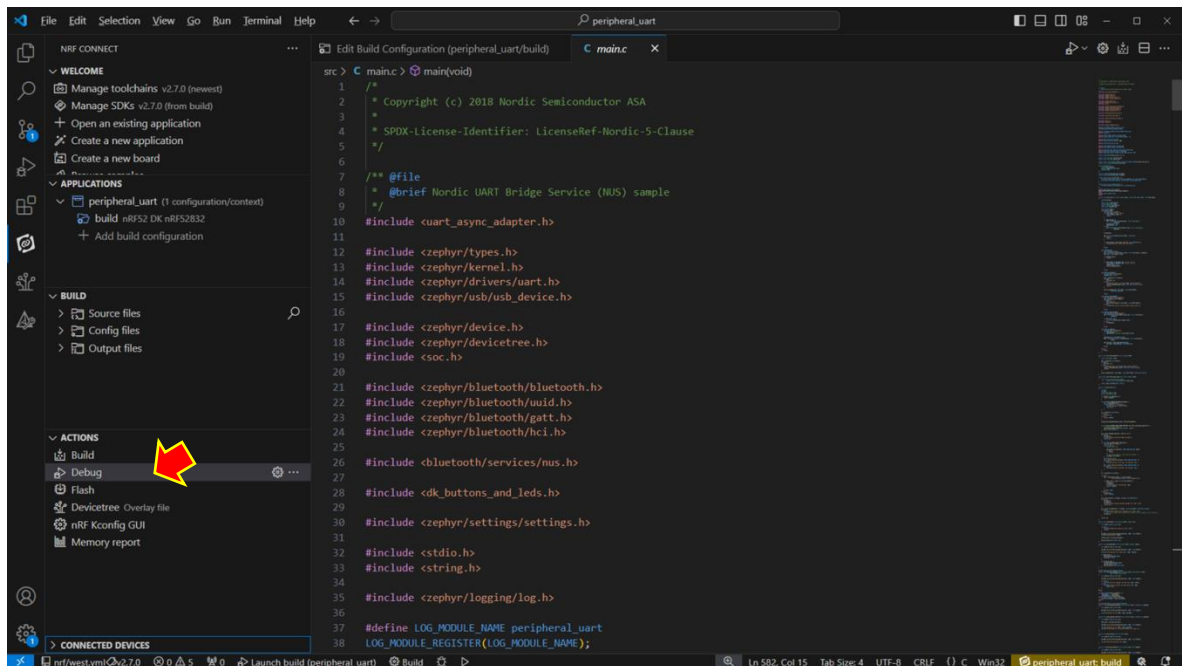
Build system default

Use sysbuild

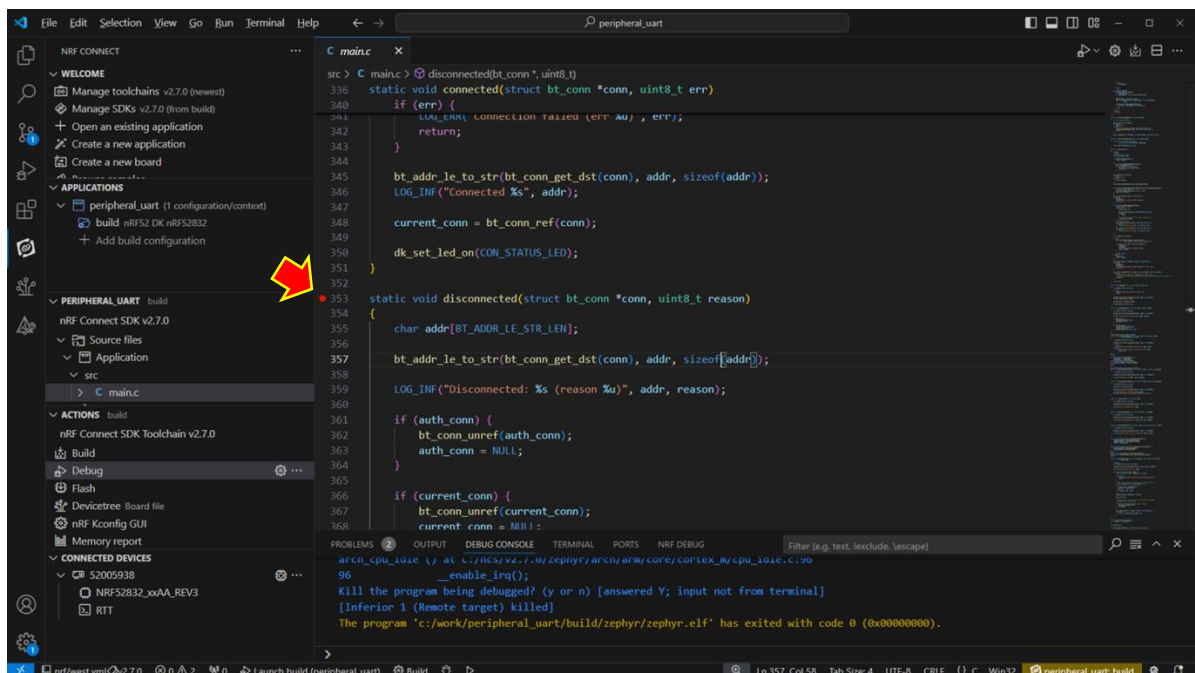
No sysbuild

Build Configuration

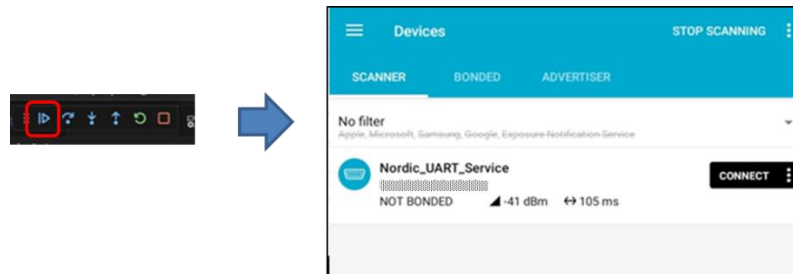
(3) Click “Debug” under ACTIONS in the left pane to start debugging.



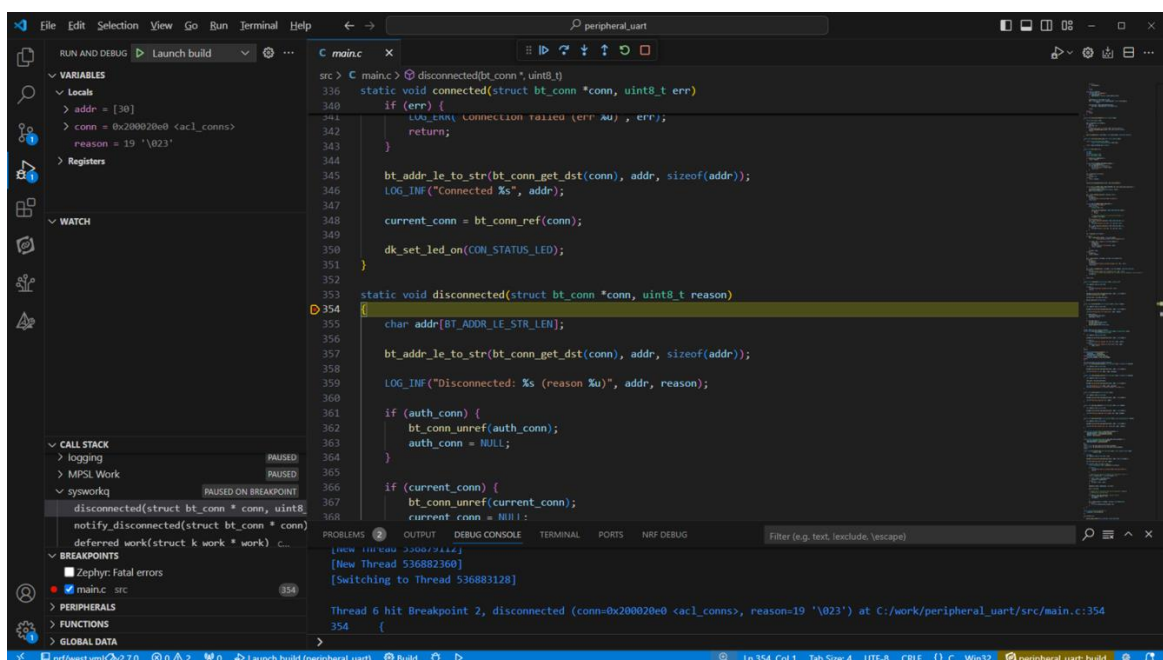
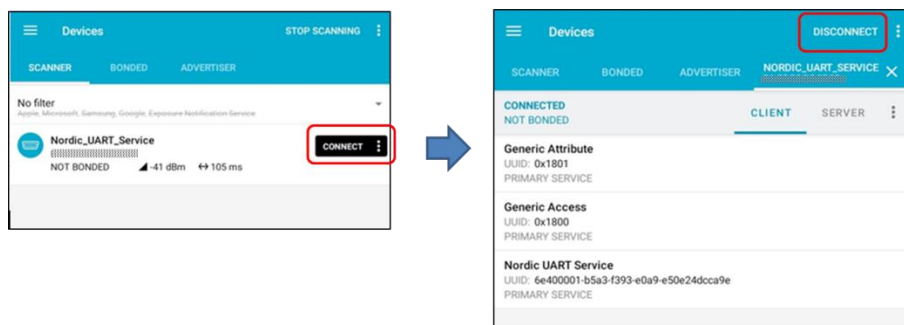
(4) Set a temporary breakpoint. (e.g., click `static void disconnected(struct bt_conn *conn, uint8_t reason)` in main.c to set a breakpoint)



- (5) Press the Play button in the upper center or the F5 key to execute. Launch Nordic's "nRF Connect for Mobile" application on your smartphone and you will see the "Nordic_UART_Service" device. This means that the device is transmitting a Bluetooth® Low Energy signal.



- (6) Tap the "CONNECT" button in the application to connect via Bluetooth®, then tap the "DISCONNECT" button to disconnect. You will see that it stops at the breakpoint (disconnect process) set in the previous section of the source code.



5 Verify action

5.1 Sample Verify

The following is a confirmation of the FW operation created in each of the above nRF5 SDK/nRF Connect SDK development procedures, confirming that UART communication via Bluetooth® Low Energy can be performed between a PC and a smartphone.

5.1.1 Setup of PC

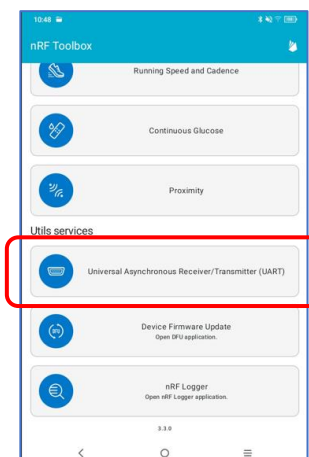
- (1) Install the software for UART terminal “TeraTerm” from below.
<https://teratermproject.github.io/index-en.html>
- (2) Start “TeraTerm” and set the COM port and baud rate of the evaluation board to 115200bps. Open the setup menu by clicking “Setup” --> “Terminal” and mark “Local Echo”.

5.1.2 Setup of Smartphone

- (1) Install “nRF Toolbox” application on smartphone.
<https://www.nordicsemi.com/Products/Development-tools/nRF-Toolbox>



- (2) Start the “nRF Toolbox” application and tap the following “UART service”.



- (3) Connect to the device named “Nordic_UART_Service”.

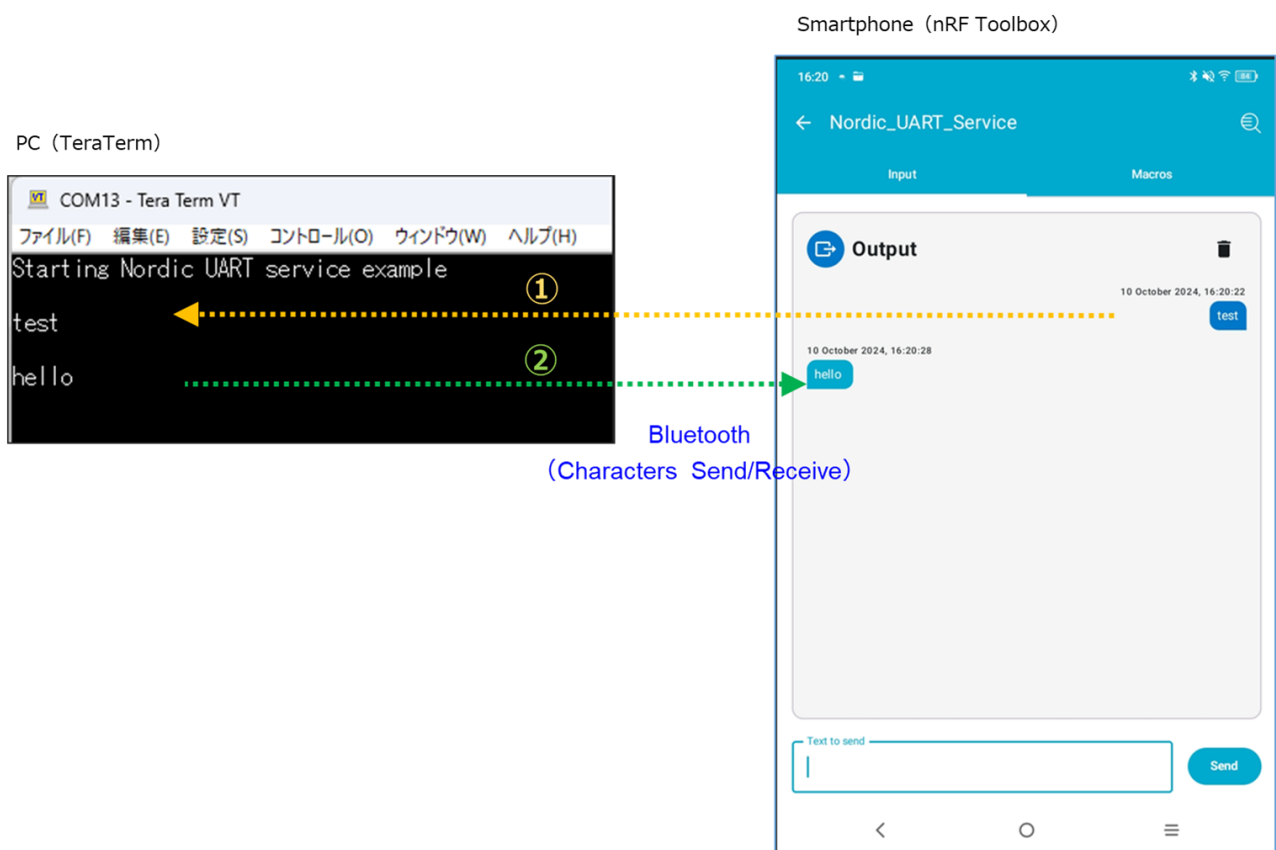
5.1.3 Bluetooth® Low Energy (UART) communication check

(1) Input text (e.g. "test") from smartphone and tap "Send".

→ The same characters will be displayed on the Tera Term.

(2) Enter any character in the Tera term and press Return key. (e.g. "hello")

→ The same characters will be displayed on smartphone.



Note: The following describes the flow up to the start of development using the sample code. For more detailed information, please refer to the technical information provided by Nordic Semiconductor.

6 Reference

- 1 Nordic Semiconductor ASA, “nRF Connect SDK”, [nRF Connect SDK - nordicsemi.com](https://nordicsemi.com)
- 2 Nordic Semiconductor ASA, “Bluetooth: Peripheral UART — nRF Connect SDK 2.7.0 documentation (nordicsemi.com)”,
https://docs.nordicsemi.com/bundle/ncs-2.7.0/page/nrf/samples/bluetooth/peripheral_uart/README.html
- 3 Nordic Semiconductor ASA, “Debugging in nRF Connect for VS Code - Nordic Developer Academy (nordicsemi.com)”,
<https://academy.nordicsemi.com/courses/nrf-connect-sdk-intermediate/lessons/lesson-2-debugging/topic/debugging-in-vs-code/>
- 4 Nordic Semiconductor ASA, “nRF52832 Product Specification”,
https://docs.nordicsemi.com/bundle/ps_nrf52832/page/nrf52832_ps.html

7 Precautions

- The information in this document is current at the time of writing and is subject to change.
- We assume no responsibility whatsoever for any damages, incidents, or accidents caused by this material.
- Please confirm the applicable license terms and conditions when using any of the software listed in this document.
- The names of companies, products, and services mentioned in this document may be used as trademarks of their respective owners.