

蓝牙

由 | 宋廷宇 | 最近更新于 2024年4月11日在GMT+8 20:38

芯片: nRF52840

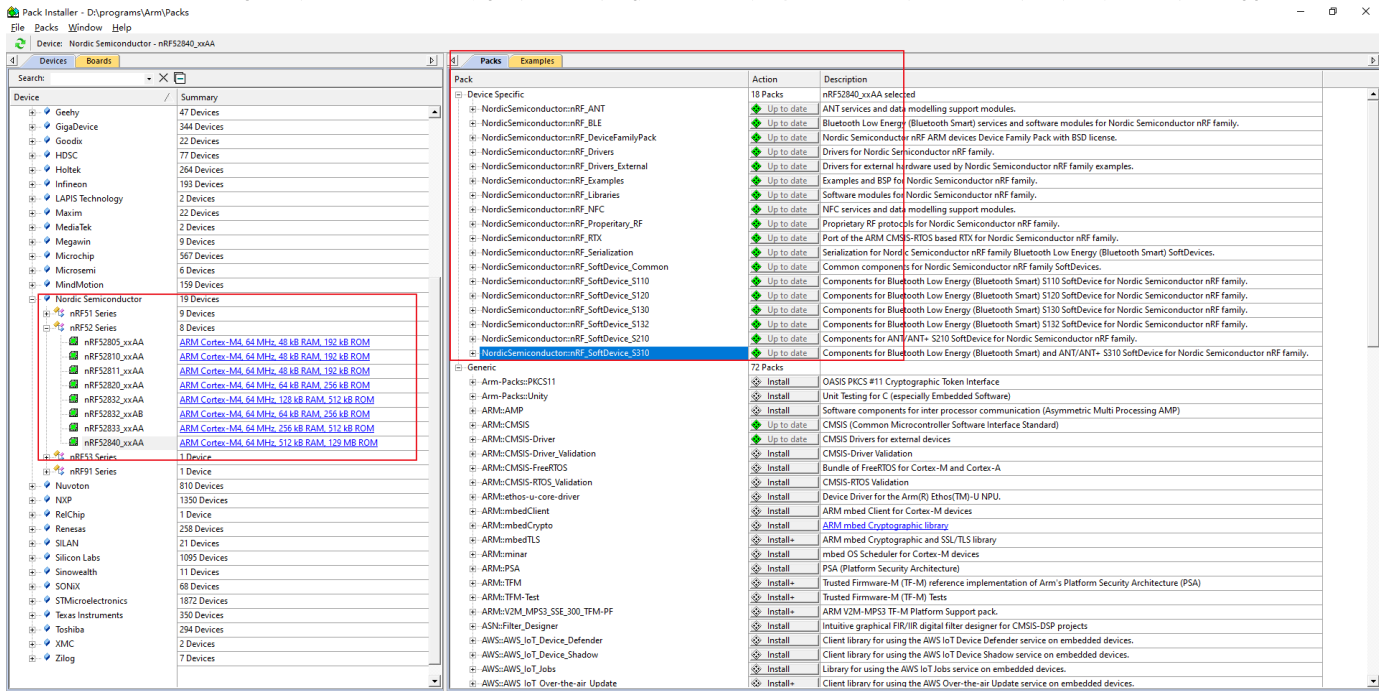
官网: https://infocenter.nordicsemi.com/index.jsp?topic=%2Fug_nrf52840_dk%2FUG%2Fdk%2Fintro.html

硬件连接

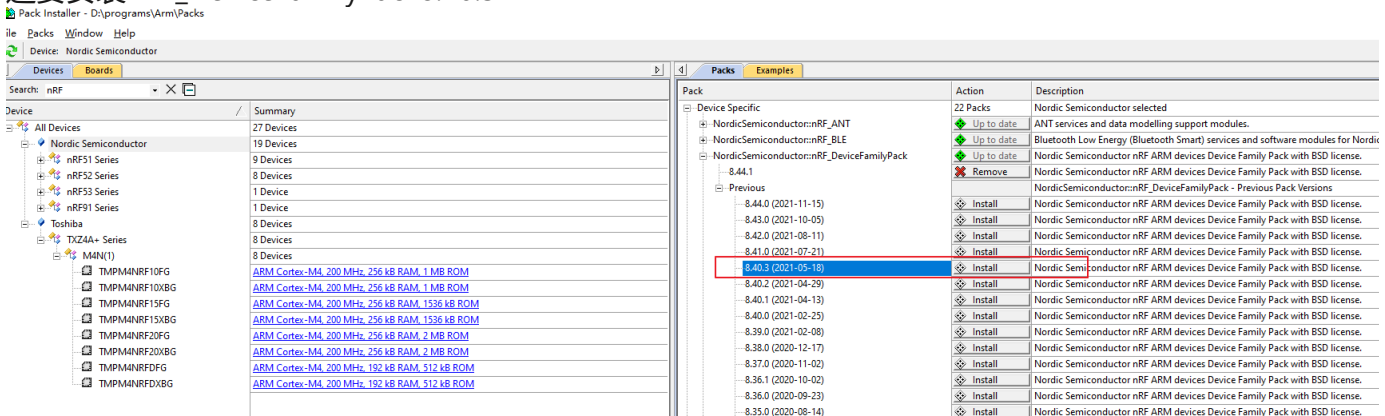
1. 开发板都使用USB供电，通过USB连接线连接开发板USB1接口到PC，不需要安装电池
2. ECU模拟板在通过另一根USB连接线连接开发板USB设备接口
3. Jlink连接需要进行调试的开发板

开发环境

1. 安装Keil(安装好后进行注册激活，激活工作在工作电脑失效，可以复制CID码，到另一台windows系统电脑下进行激活，Target选择ARM)
2. Keil Pack installer中安装Nordic器件库 以管理身份进行打开，出现安装包下载失败尝试切换网络。

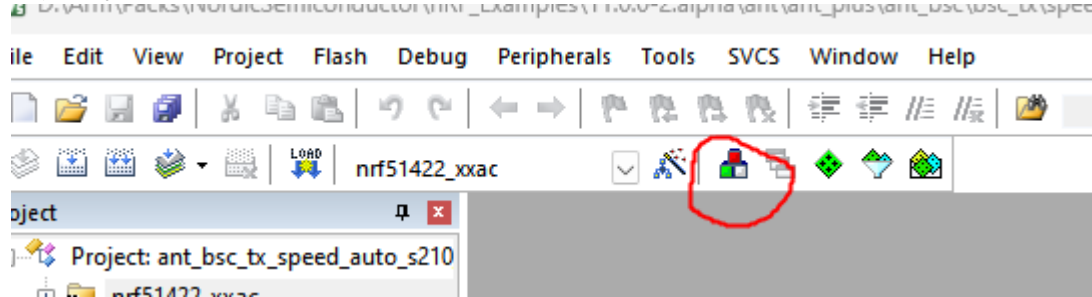


还要安装nRF_DeviceFamilyPack8.40.3

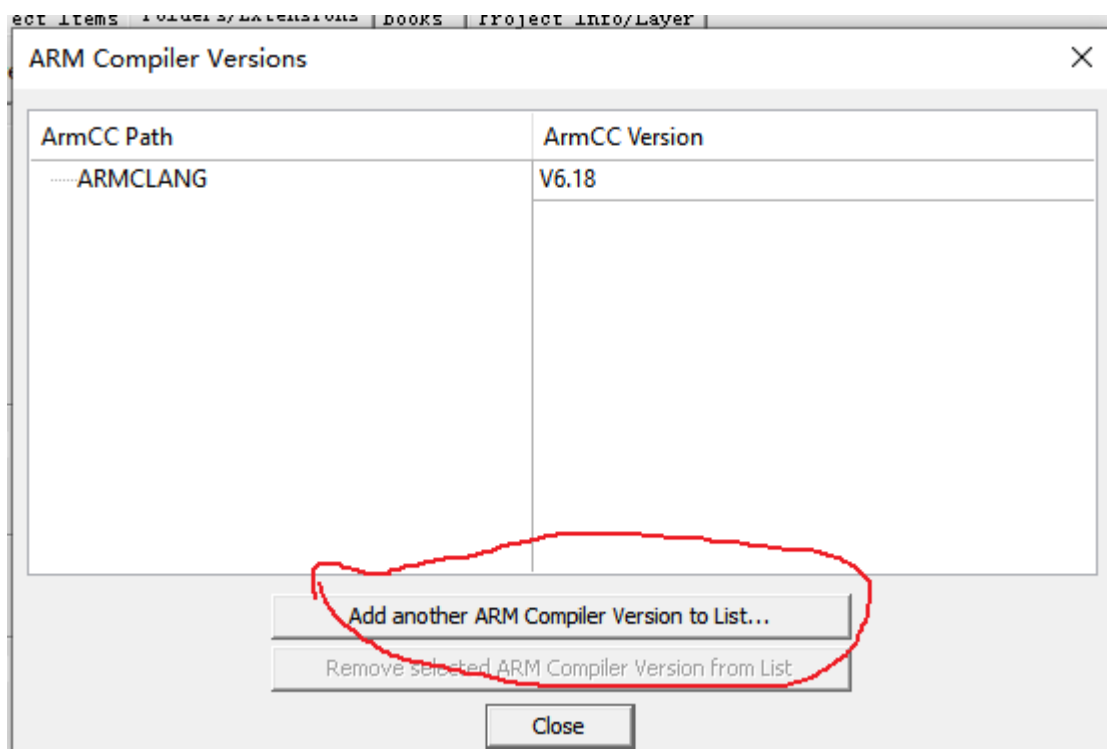
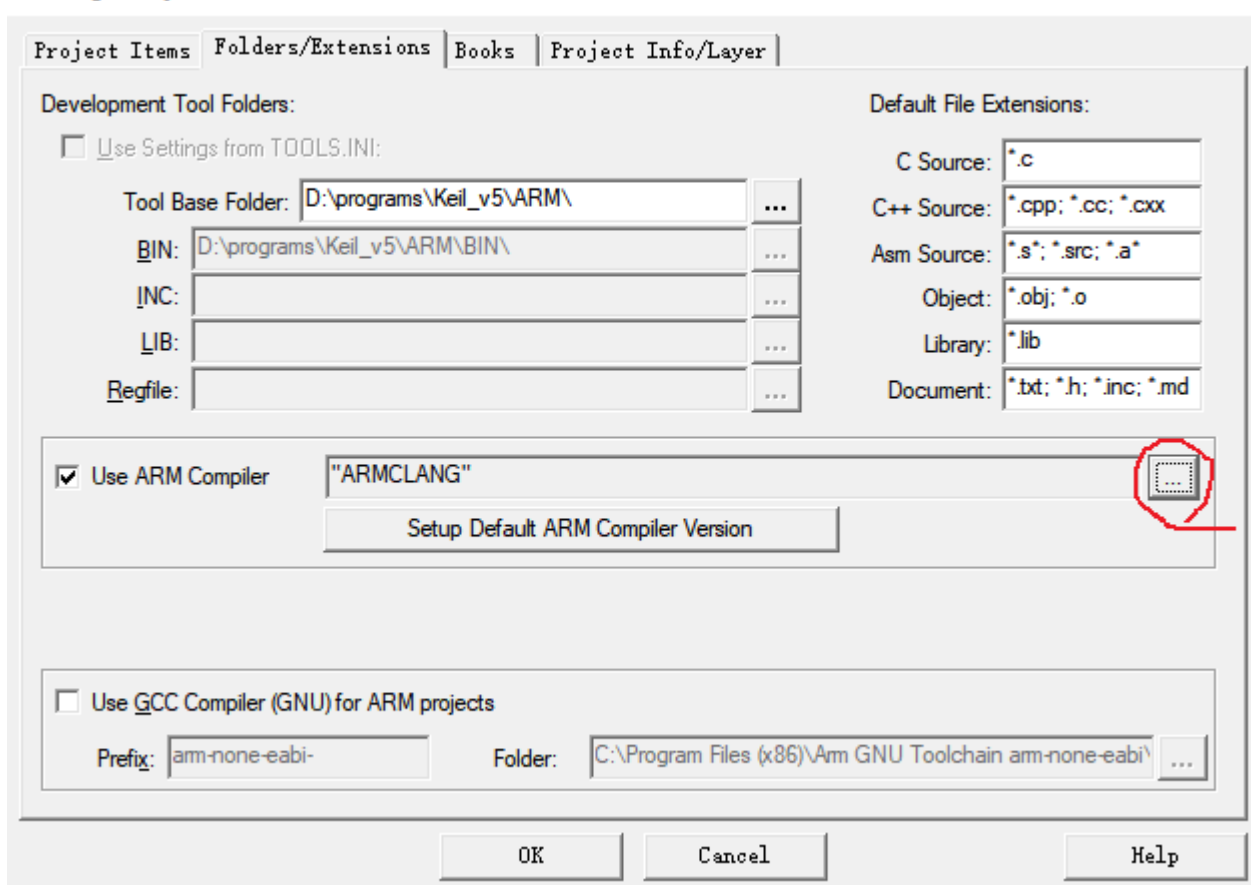


3. 安装AC Compiler 5
ARMCompiler_506_Windows，安装到Keil的安装路径的Arm路径下，目标路径起名为ARMCC

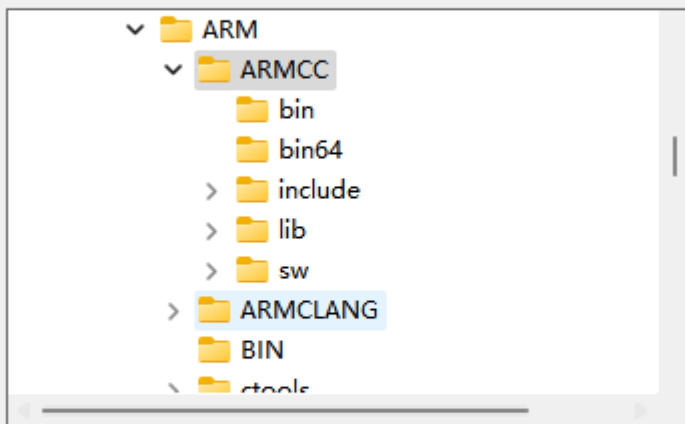
打开工程点击配置



Manage Project Items



Select a ARMCC Base Folder:



文件夹(F): ARMCC

新建文件夹(M)

确定

取消

ARM Compiler Versions

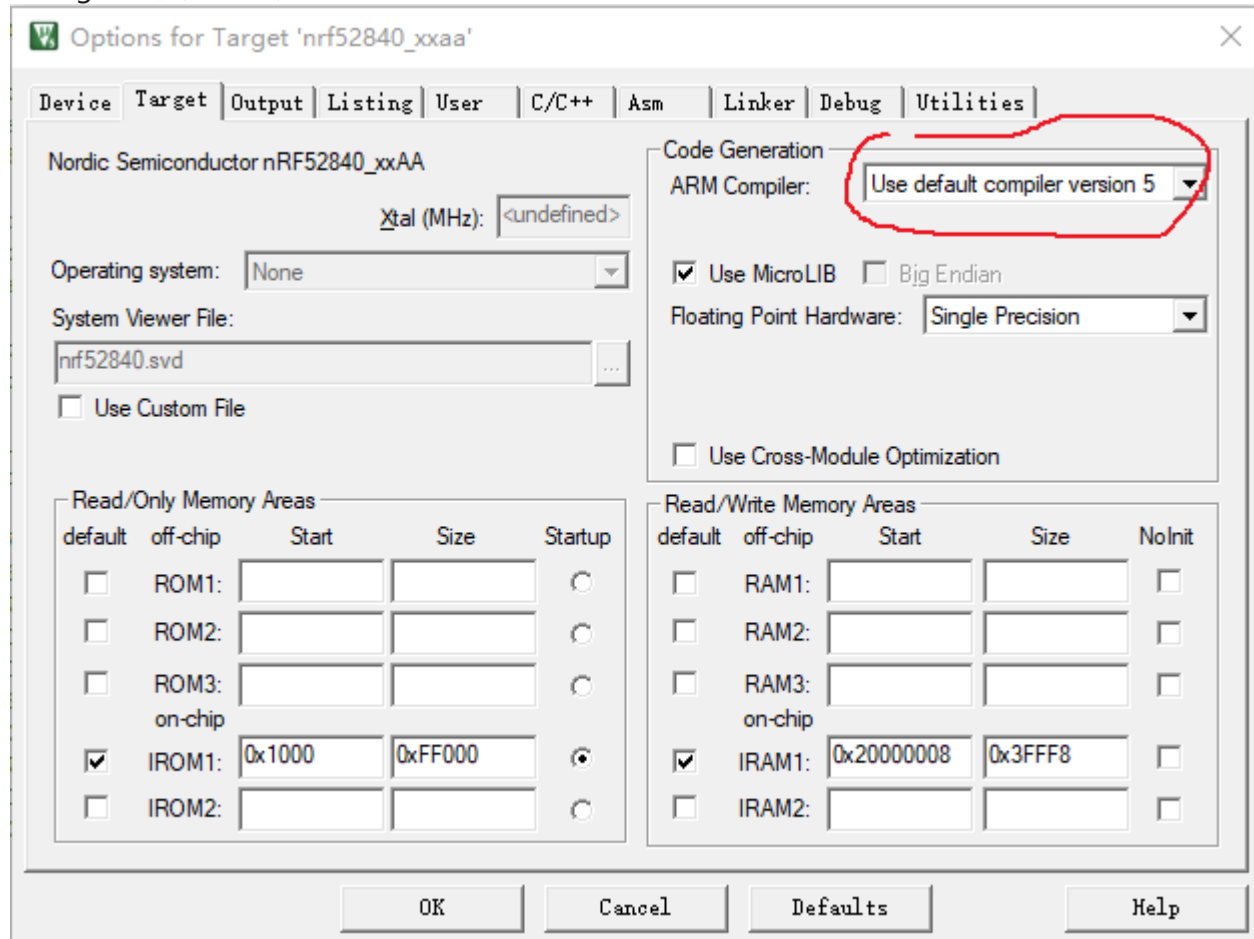
ArmCC Path	ArmCC Version
ARMCLANG	V6.18
.\ARMCC	V5.06 update 7 (build 960)

Add another ARM Compiler Version to List...

Remove selected ARM Compiler Version from List

Close

到target选项，选择设置5.0编译器



4. 安装Jlink仿真器驱动

JLink_Windows_V786f_x86_64.exe

4. 安装ST_Link驱动（需要在烧录器文件夹内执行）

stlink_winusb_install.bat

5. 开发板与model的对应关系

USB dongle nRF52840对应的model是10056

仿真器nRF52832 对应的model是10040

6. JLINK配置

Debug | Trace | Flash Download

J-Link / J-Trace Adapter

SN: 601012352

Device: J-Link

HW : V11.00 dll : V7.88j

FW : J-Link V11 compiled Jun 20

Port: SW

Max 2 MHz

Auto Clk

SW Device

IDCODE	Device Name	Move
SWDI 0x2BA01477	ARM CoreSight SW-DP	<div>Up</div> <div>Down</div>

☒ Automatic Detection

ID CODE:

☐ Manual Configuration

Device Name:

Add

Delete

Update

IR len:

Connect & Reset Options

Connect: Normal

Reset: Normal

☒ Reset after Connect

Cache Options

☒ Cache Code

☒ Cache Memory

Download Options

☐ Verify Code Download

☒ Download to Flash

Interface

☒ USB

☐ TCP/IP

Scan

State: ready

TCP/IP

Network Settings

IP-Address: 0 . 0 . 0 . 0

Port (Auto: 0

Autodetect

Ping

Misc

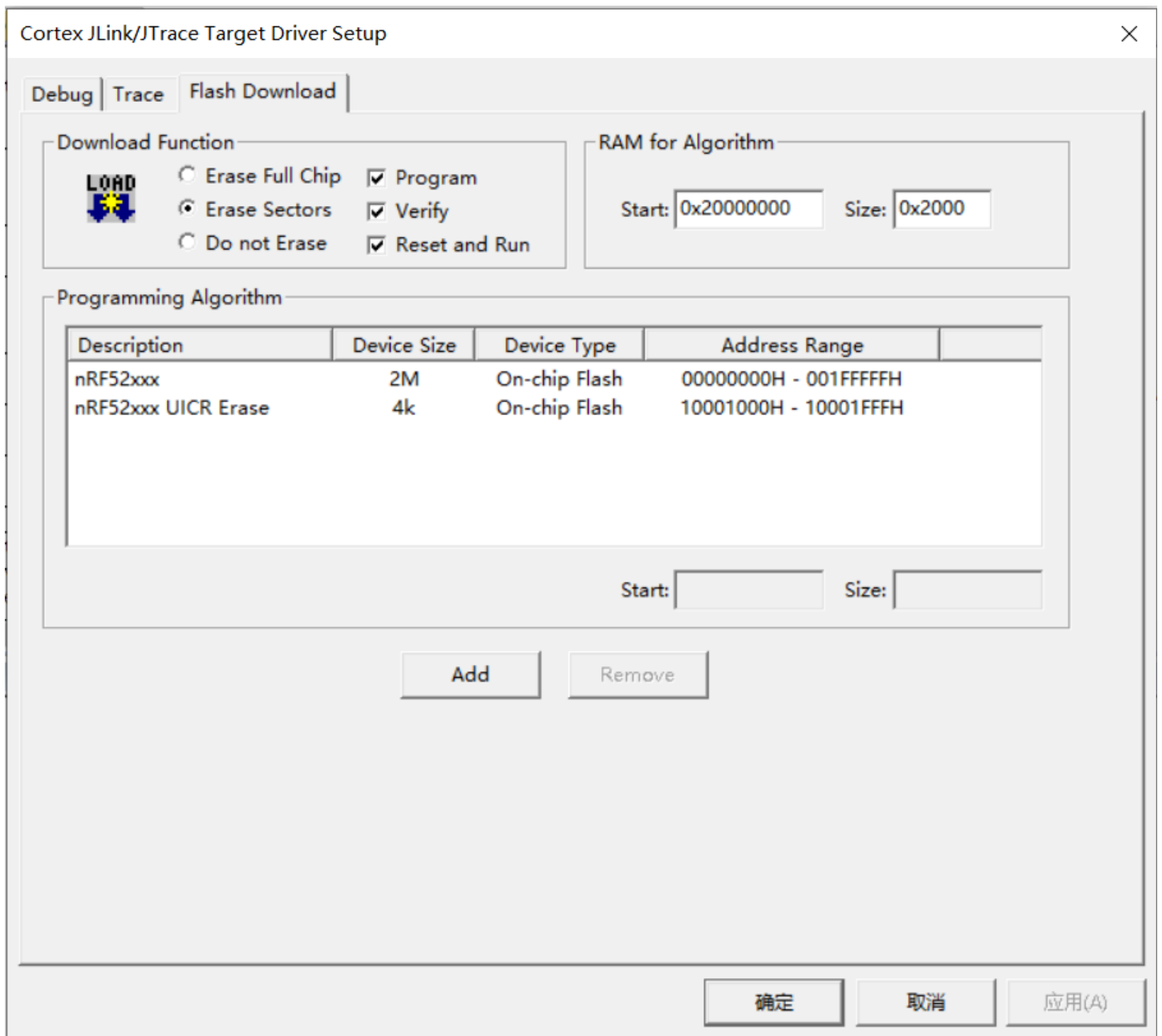
JLink Info

JLink Cmd

确定

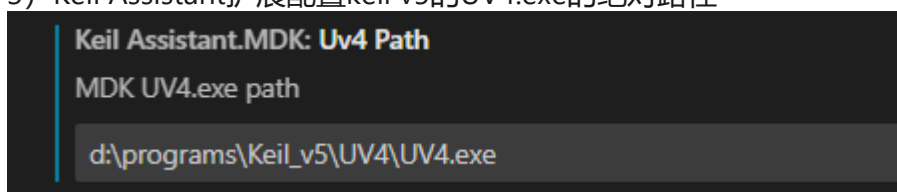
取消

应用(A)



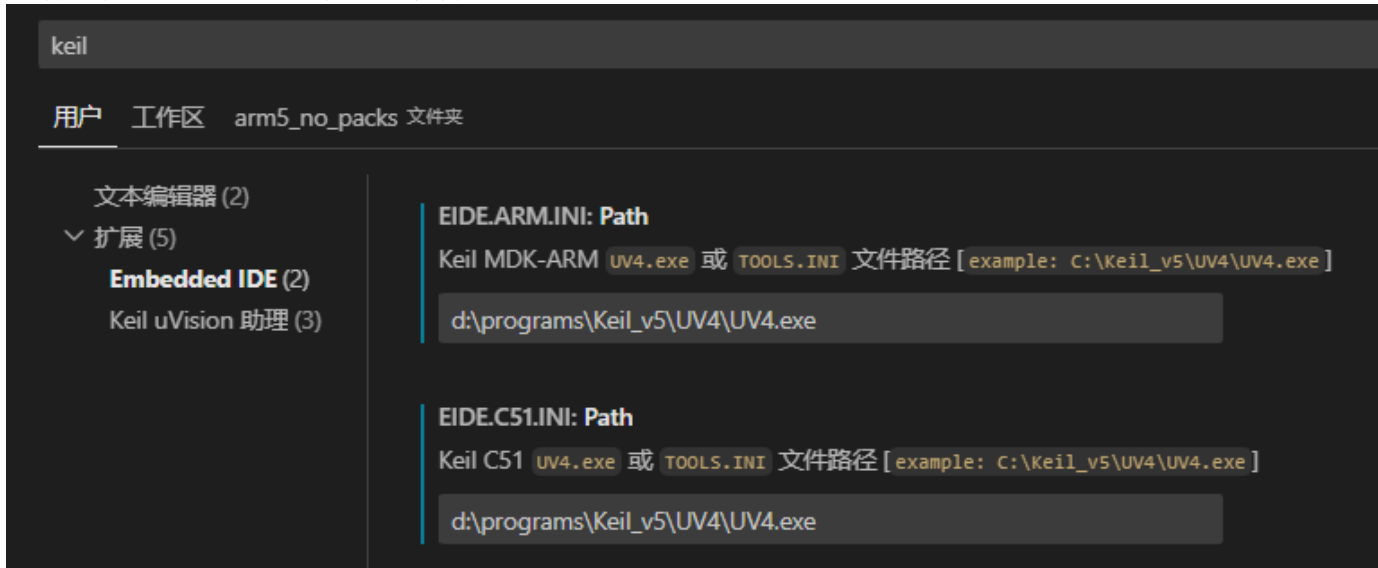
7. vs code打开keil工程

- 1) 安装C/C++扩展
- 2) 安装 Keil Assistant扩展
- 3) Keil Assistant扩展配置keil v5的UV4.exe的绝对路径



- 4) 通过vscode的打开文件夹功能打开keil工程文件所在的文件夹
- 5) 工作模式是通过keil新增删除文件，通过vs code编码、编译、下载、调试通过keil进行

6. 继续安装Embedded IDE，配置参数：



7) 通过Embedded IDE再次导入项目，导入后可有效解决头文件找不到报错的问题

基本概念

SoftDevice

参考资料：[Nordic nRF5 SDK和softdevice介绍](#) [🔗](#)

Nordic的Softdevice是蓝牙的软件协议栈

1. SoftDevice命名规则

Softdevice命名规则一。Softdevice包括两种底层协议栈：BLE和ANT，BLE包括两种角色：central（又称master）和peripheral（又称slave），为此需要给这些不同类型的协议栈进行命名区分。协议栈命名格式为Sxyz，其中

- x – 表示协议栈的类型，1表示BLE协议栈，2表示ANT协议栈，3表示同时支持BLE和ANT
- y – 表示BLE角色，1表示从设备，2表示主设备，3表示同时支持主设备和从设备
- z – 表示芯片类型，0表示nRF51系列，2表示nRF52系列

比如S110，表示只支持从设备模式的nRF51 BLE协议栈

比如S130，表示既支持从设备模式又支持主设备模式的nRF51 BLE协议栈

比如S132，表示既支持从设备模式又支持主设备模式的nRF52 BLE协议栈

比如S212，表示nRF52 ANT协议栈

比如S332，表示nRF52既支持BLE协议栈又支持ANT协议栈，而且BLE协议栈既支持从设备模式又支持主设备模式

Softdevice命名规则二。大体上跟命名规则1相同，但是协议栈编号最后2位跟芯片型号一样，比如S140，代表这个协议栈专门用于nRF52840。由于52840 Flash空间很大，没有必要做各种细分的协议栈，S140协议栈是一个大而全的协议栈，包含蓝牙所有功能。

低功耗蓝牙基本概念

[GATT协议](#) [🔗](#)

[NRF的UUID](#) [🔗](#)

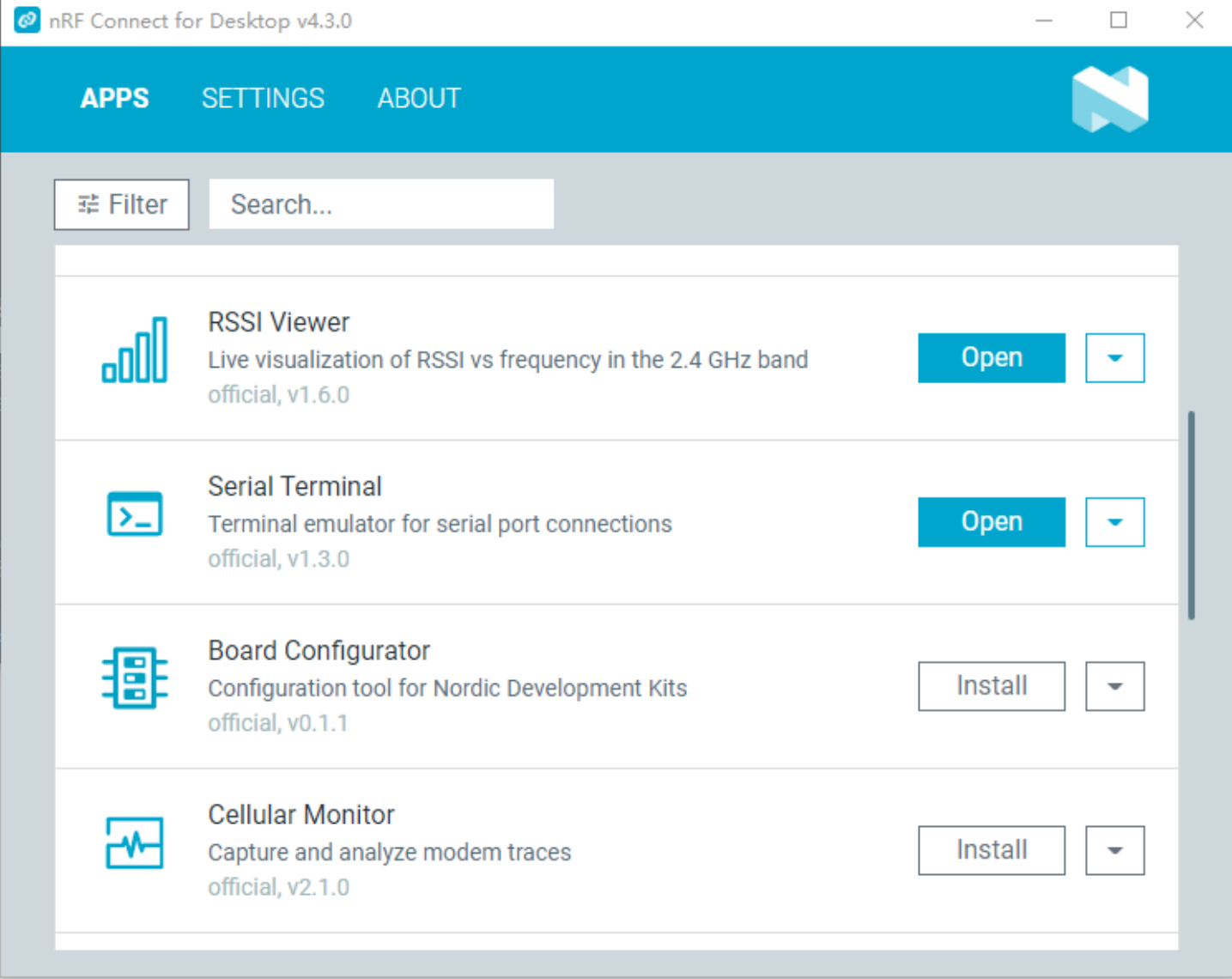
QSPI

[QSPI协议](#) [🔗](#)

USB ADC

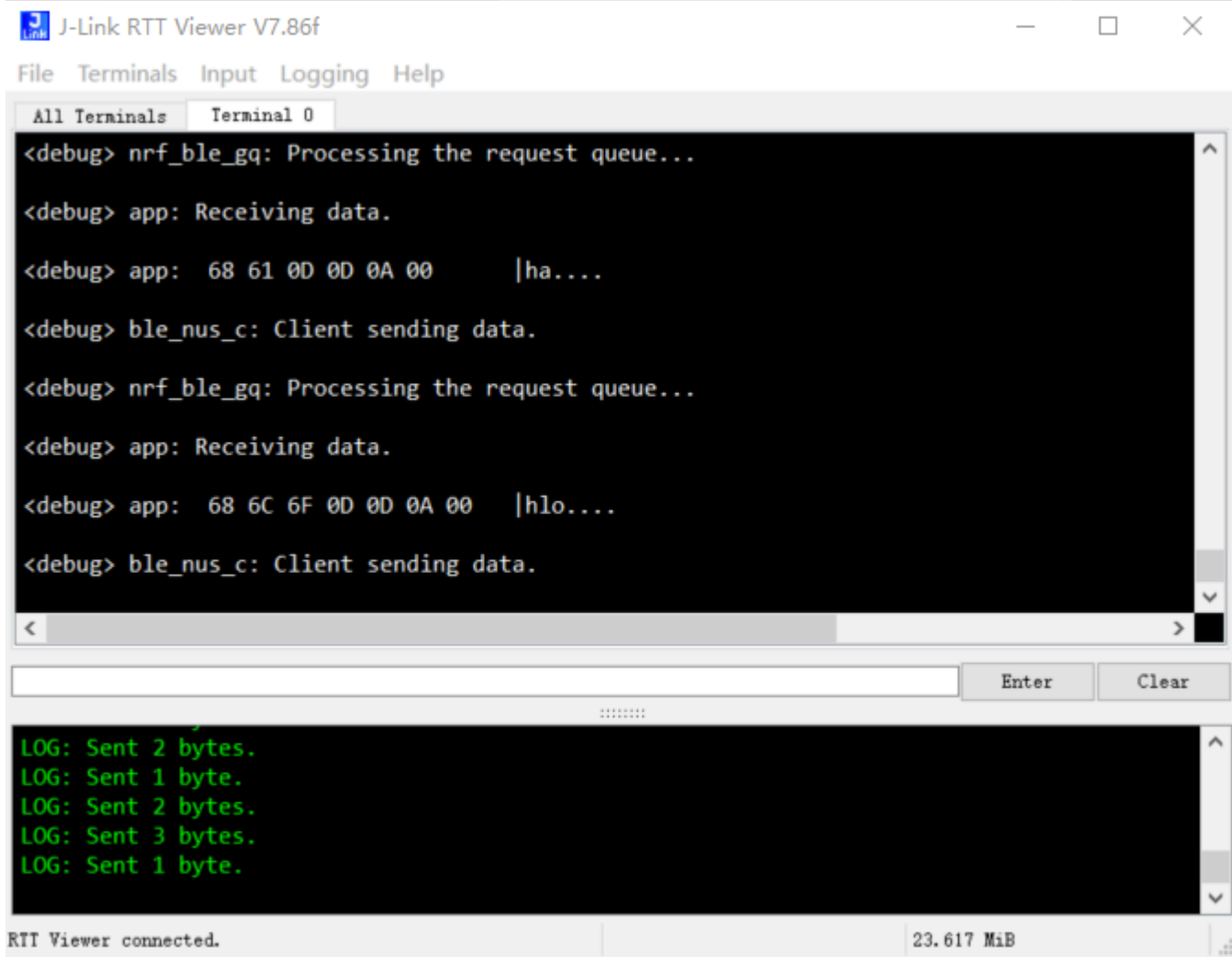
USB ADC发送数据

使用nRF connect for desktop的Serial Terminal工具



RTT日志

需要安装J-Link的工具套装，使用J-Link RTT Viewer



The screenshot shows the J-Link RTT Viewer V7.86f application window. The title bar includes the J-Link logo and the text 'J-Link RTT Viewer V7.86f'. The menu bar contains 'File', 'Terminals', 'Input', 'Logging', and 'Help'. Below the menu bar, there are two tabs: 'All Terminals' and 'Terminal 0'. The 'Terminal 0' tab is active, displaying a black background with white text representing debug logs. The logs show a sequence of events: 'nrf_ble_gq: Processing the request queue...', 'app: Receiving data.', 'app: 68 61 0D 0D 0A 00 |ha....', 'ble_nus_c: Client sending data.', 'nrf_ble_gq: Processing the request queue...', 'app: Receiving data.', 'app: 68 6C 6F 0D 0D 0A 00 |hlo....', and 'ble_nus_c: Client sending data.'. Below the terminal window, there is a text input field with 'Enter' and 'Clear' buttons. At the bottom of the window, a status bar shows 'RTT Viewer connected.' on the left and '23.617 MiB' on the right.

```
<debug> nrf_ble_gq: Processing the request queue...
<debug> app: Receiving data.
<debug> app: 68 61 0D 0D 0A 00 |ha....
<debug> ble_nus_c: Client sending data.
<debug> nrf_ble_gq: Processing the request queue...
<debug> app: Receiving data.
<debug> app: 68 6C 6F 0D 0D 0A 00 |hlo....
<debug> ble_nus_c: Client sending data.

LOG: Sent 2 bytes.
LOG: Sent 1 byte.
LOG: Sent 2 bytes.
LOG: Sent 3 bytes.
LOG: Sent 1 byte.

RTT Viewer connected. 23.617 MiB
```

问题及解决：

1、J—link仿真器接板子，擦除烧写flash失败

dell电脑USB口供电不足，蓝牙板需要装电池。

2、烧写softdevice失败

需要先全芯片擦除flash，然后再烧写softdevice

3、52480在Jlink不能识别，按复位键都不行

拔掉Jlink线，去掉电池，去掉USB连接线，使板子彻底掉电。

电池电量不足也会造成此现象

如果能够识别，但是烧写就报设备找不到，可能是sw max时钟太高，降低该时钟直到可稳定烧写

4、不能进入调试模式，烧写完程序，RTT日志没有输出，板子没起来，windows设备管理器看不到外设模式的USB ACM端口

需要擦除整个flash芯片，并重新烧写softdevice，app程序

或者切换到外设模式的工程，重新烧写softdevice和APP，再切换会主机模式工程，重新烧写softdevice和APP。

5、如果不能进入调试，RTT也没有启动消息，大概率是代码有问题，降低C/C++编译优化级别到O0

6、如果调试模式总是跳转到app_error_weak.c的NRF_BREAKPOINT_COND，需要kill掉所有断点，只留最后一个需要的断点，重启启动调试。

7、调试时台式电脑尽量使用后面板连接，连接前面板可能会出现供电不足，导致数据接收失败。

8、RTT日志有输出，板子起来，运行一段时间后，windows设备管理器看不到外设模式的USB ACM端口，大概率代码崩溃，检查代码，修改后要先编译，然后烧录。