



OHTCOM

Technology Ltd.

nRF51822 Application Kit

nRF51822

User Guide v1.2

1 介绍

nRF51822 *Bluetooth®* low energy/2.4 GHz Application Kit (AK II) 提供了一整套的测试和应用 nRF51822 的解决方案。nRF51822 是 nRF51 系列中的一员，它是一个超低功耗（ultra-low power），单片系统 (SoC) 的 2.4 GHz 无线通信解决方案。

1.1 最小系统要求

- nRFgo Studio v1.14 或更高版本
- Windows XP or Windows 7

1.2 外部资源

- Keil MDK-ARM Lite v4.54 或更高版本 <https://www.keil.com/demo/eval/arm.htm>
- J-Link Software v4.52b 或更高版本 <http://www.segger.com/jlink-software.html>

1.3 Writing conventions

这篇用户指南遵从了一些排版规则，这样能够使文章更加连贯，更加易于阅读。以下是使用到的协作约定：

- 命令使用 **Luci da Consol e.**
- 芯片管脚使用 **Consolas.**
- 文件名和用户接口使用 **bold.**
- 内部关联采用斜体并使用 ***semi-bold.***

1.4 Application kit 发布说明

Date	Kit version	Description
2013年8月10日	0.1	Init.
2013年8月19日	0.2	加入蓝牙测试部分
2013年8月25日	0.3	整合 USB Dongle 介绍.
2013年8月28日	0.4	完成度 50%，增加 200%的内容。
2013年9月5日	0.8	修改一些排版错误，加入 AK II board 和 USB dongle 的特点介绍
2013年9月7日	0.9	增加文档目录树及说明
2013年10月17日	1.0	全新 AK II 发布，修改相关内容
2013年11月20日	1.1	加入 AK board 下载程序的硬件连接部分
2013年12月13日	1.2.1	更新 USB dongle2, bug fix

2 套件内容

nRF51822 Application Kit 的硬件资料以及相关软件和文档。可以从 https://gitcafe.com/overheat1984/nRF51822_STUDY 处下载。

2.1 nRF51822 Application Kit 硬件

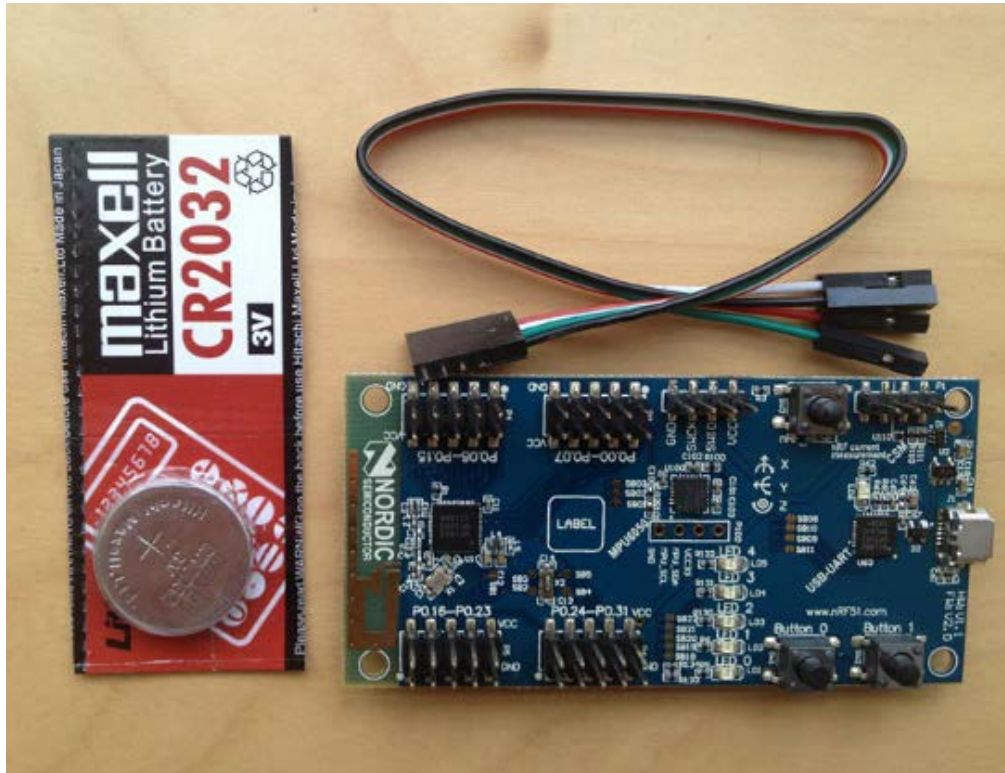


Figure 1 nRF51822 硬件

2.2 可下载的资源

nRF51822 Application Kit 包括固件程序，文档，硬件原理图等。需要索取资料者，请联系销售客服索取。

2.2.1 nRF51822 相关资料目录结构

1. 上手指南 (本文档)
2. 开发环境 MDK 4.72a
3. Jlink 驱动
4. 开发工具包 (SDK)
5. 芯片数据手册
6. iOS app 源码
7. 安卓 app 源码
8. 官方 EK 的参考设计
9. nRFgo Studio 官方烧写协议栈软件
10. S110 蓝牙协议栈
11. BLE 介绍视频

2.2.2 文件说明

资料中所包含的文件说明，请详见[附录 A](#)。

2.2.3 原理图

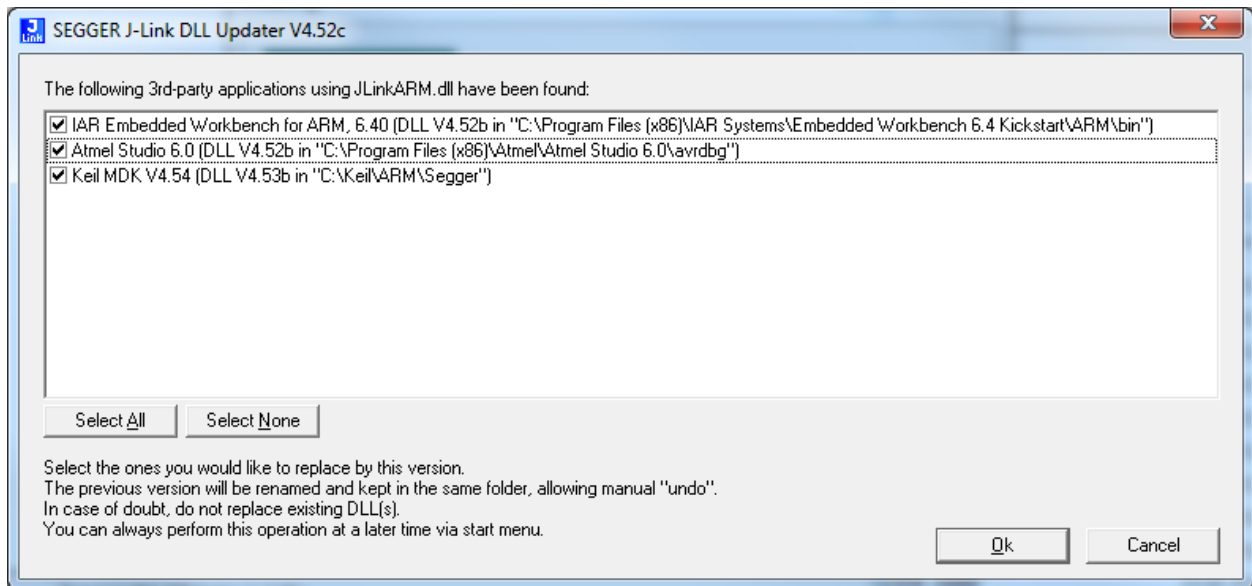
PDF 文件包括 Application Kit 的原理图，以及 PCB 的 PDF 文件。 .

3 开发环境配置

这一节演示了如何设置开发环境，并且提供了一个快速编程的实例。

注册, 下载, 和安装

1. 如果你已经安装了 Keil MDK-ARM Lite 请跳到第 2 步。否则，请下载并安装 Keil MDK-ARM 。
<https://www.keil.com/demo/eval/arm.htm>，网盘资料里提供该软件，以及安装方法：
地址为“nRF51822 AK 相关资料 > 软件 > KEIL ”目录下
2. 下载并安装 J-Link for Windows <http://www.segger.com/jlink-software.html> 可以使用 518002693 这个 serial number 来下载程序。(配套资料里也提供这个软件)
地址为“nRF51822 AK 相关资料 > 软件 > Jlink 驱动 ”目录下
在安装过程中，请准确选择需要支持的 IDE（我们使用 KEIL），之后才会正确的更新相应的 SEGGERDLLs。



3. 请使用 Keil v4.54 以上的版本。
4. 下载并安装 nRF51 SDK，并确保选择了 **Keil MDK-ARM** 安装选项。
SDK 地址为 nRF51822 AK 相关资料 > 软件 > nrf51_sdk_v4_4_xxxxx.msi

注意：如果 SDK 安装报错，请下载并安装 Microsoft .NET Framework 3.5。

3.1 nRF51822 II Application board

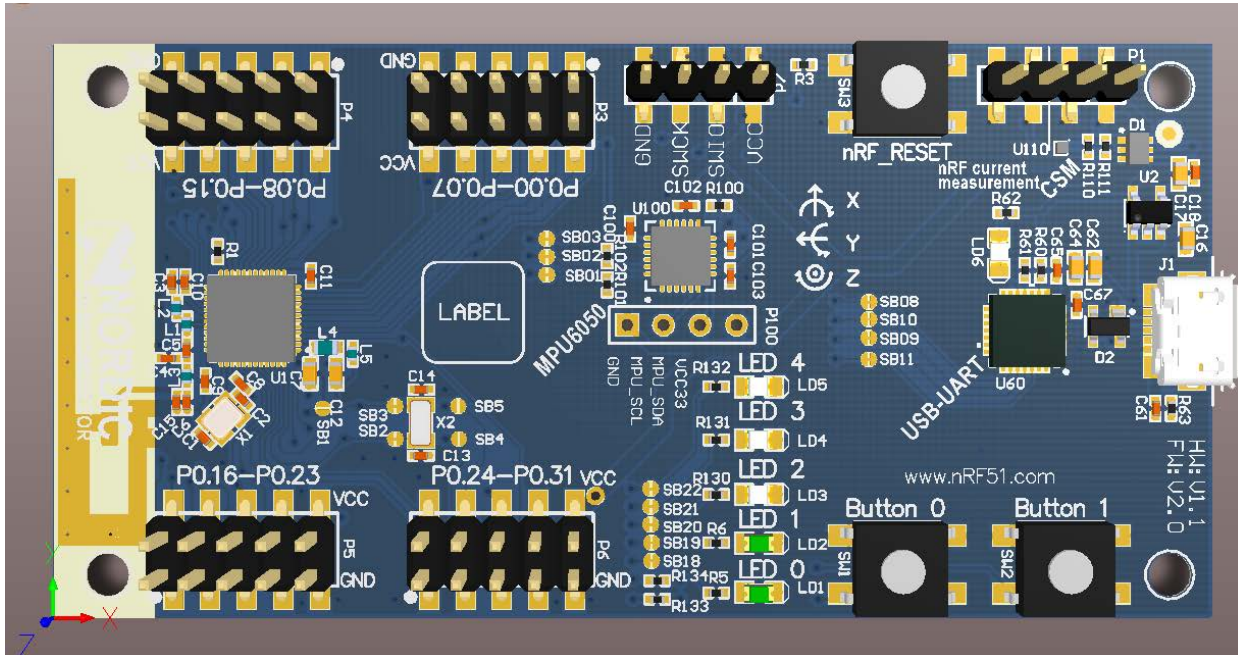


Figure 2 nRF51822 AK II 3D 效果图

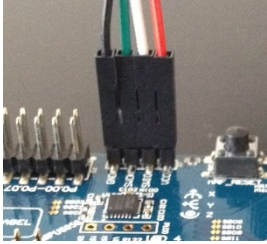
主要特性:

- 兼容官方 nRF51822-EK PCA10001 的外围接口。
- 兼容 2.4 GHz nRF24L 系列芯片 (on-the-air)
- 支持 Bluetooth low energy (低功耗蓝牙 4.0)
- 用于电流检测的 Current Shunt Monitor (CSM)
- PCB 印制天线 (Inverted F Antenna)
- 加速度计
- 陀螺仪
- 五个独立可编程 LED (共阴极)
- 两个用户按键
- CP2102 调试接口 (USB 转串口)
- 标准的 4pin SWD 调试接口
- 支持 iOS 应用, 提供源码

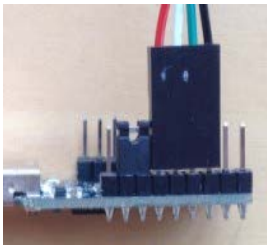
注意: 本开发板由于集成了六轴传感器 MPU6050c, USB 转串口 CP2102, 以及 5 个 LED 等。所以极为适合开发穿戴式设备, 如计步器、体感遥控器等。

3.1.1 硬件连接

1.将 4 pin (彩色) 连接线一端插在 AK board 的 “P7” 上



2.另一端连接 USB dongle (红色对应 VCC , 黑色对应 GND)



3. 或使用 Jlink (选购)

注意：由于 Jlink 默认并不对外供电，所以使用 Jlink 下载程序时需要将 Micro USB 连上



红色——pin1

白色——pin7

绿色——pin9

黑色——pin20

4.最后再将 USB 口接入电脑。

注：USB dongle 可以给 AK II board 供电，配套使用时不需要放电池。并注意 dongle 跳线位置，标记为 nRF 一侧为对 dongle 本身编程，OUT 一侧为对外编程 (AK board)。

3.1.2 开始 Blinky 工程 (闪灯)

注意：开发板出厂时已经预装程序，如需下载其他程序，需要使用 nRFgo Studio 左下的“nRF51 Programming”，之后点击“erase all”全部擦除后再次烧写。

1. 进入 Blinky project ，在你电脑上的
<你的 KEIL 安装路径下>\ARM\Device\Nordic\nRF51822\Board\PCA10001\blinky_example\arm 目录下。
2. 在 Keil μ Vision 中打开 Blinky 工程，双击 **blinky.uvproj** 文件。
3. 从 Select Target 下拉列表中选择 **nRF51822_xxaa(256K)**，然后点击 **Build** 图标，或者按键盘上的 **F7** 编译 Blinky 工程。
4. 点击 **Load** 图标，下载并运行 Blinky 演示程序。**LED 0** 和 **LED 1** 应该开始连续的闪烁了。

注意：

1.由于 Jlink 固件与 nordic SDK 配合的问题，部分 KEIL 无法正常下载。此时，请使用 nRFgo Studio 左下的“nRF51 Programming”，之后点击“Program Application”选项框下载。具体请参考 **3.3 蓝牙低功耗接近传感器演示程序**。使用 nRF51822 USB Dongle 下载程序则没有上述问题。

3.2 nRF51822 USB Dongle (PCA10001)

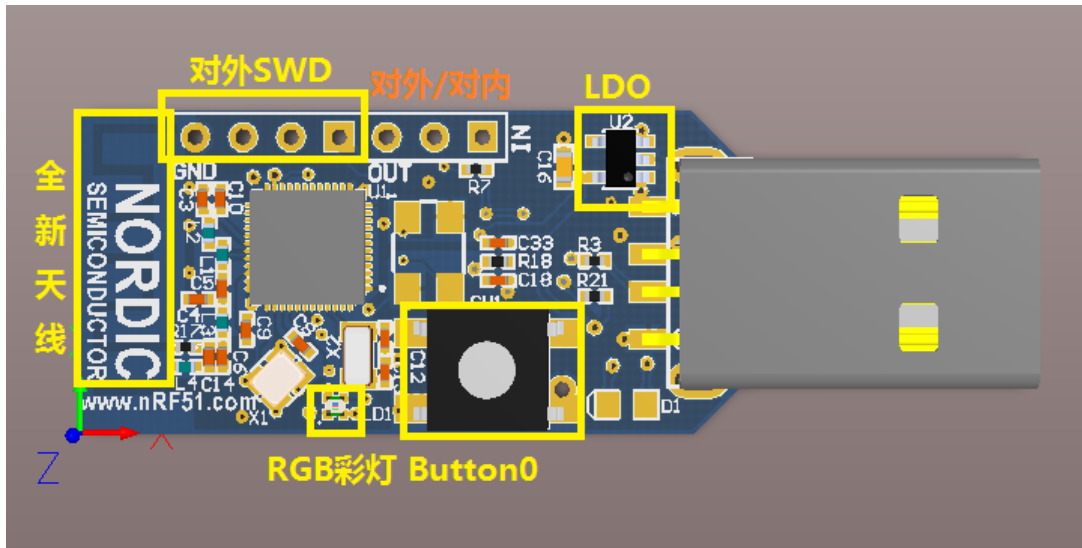


Figure 3 nRF51822 USB Dongle 3D 效果图

主要特性:

- 超小体积
- 兼容 2.4 GHz nRF24L 系列芯片
- 支持 Bluetooth low energy (低功耗蓝牙 4.0)
- PCB 印制天线
- RGB 用户可编程 LED (七彩灯, 共阳极)
- Button (用作演示官方例程)
- 标准的 4pin SWD 调试接口
- 兼容 Jlink V7, 并支持自动升级
- 支持 iOS 应用, 提供源码

注意: 本开发板是 nRF51822 的最小系统开发板, 并且集成了 JLINK 的下载功能 (无需外接 JLINK 下载器), 所以适合外出时随身携带。

另外, 与 AK II board 配合可以完成无线组网, 对 AK II board 编程、供电等任务。节省空间的同时节约您的资金。

特别注意!!! USB Dongle 上的跳线决定了对内还是对外编程。“IN”端 (最边上) 对 USB Dongle 内部编程, “OUT”端对外编程。请根据实际使用情况调整跳线帽位置。

3.2.1 USB Dongle 的 Blinky 工程（多彩 LED）

注意：开发板出厂时已经预装程序，如需下载其他程序，需要使用 nRFgo Studio 左下的“nRF51 Programming”，之后点击“erase all”全部擦除后再次烧写。

1. 进入 Blinky project，在你电脑上的<你的 KEIL 安装路径下> \ARM\Device\Nordic\nRF51822\Board\PCA10001\blinky_example\arm 目录下。
2. 在 Keil μ Vision 中打开 Blinky 工程，或者双击 **blinky.uvproj** 文件。
3. 修改 main 函数为：

```
int main(void)
{
    uint8_t output_state = 0;

    // Configure LED-pins as outputs
    nrf_gpio_range_cfg_output(21, 23);

    while(true)
    {
        nrf_gpio_port_write(NRF_GPIO_PORT_SELECT_PORT2, \
                           ~(1 << (output_state + 5)));
        output_state = (output_state + 1) & 0x03;
        nrf_delay_ms(100);
    }
}
```

4. 从 Select Target 下拉列表中选择 **nRF51822_xxaa(256K)**，然后点击 **Build** 或者按键盘上的 **F7** 编译 Blinky 工程。
5. 点击 **Load** 图标，下载并运行 Blinky 演示程序。多彩 **RGB LED** 应该开始连续的闪烁了。

3.3 蓝牙低功耗接近传感器演示程序

本节描述了如何在 SoftDevice（蓝牙协议栈）基础上，编写一个接近器（proximity）。它会通过 **BLE（Bluetooth low energy）4.0** 发送信号，手机应用程序可以通过判断信号强度来确定与设备之间的距离。

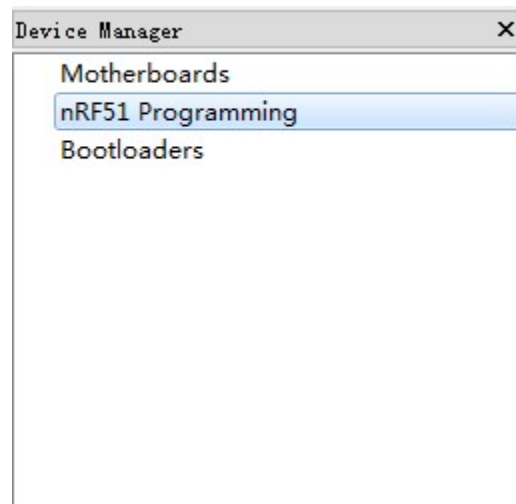
注意：该例程使用 AK II board 或者 USB dongle 均可

3.3.1 下载 SoftDevice:

下载 S110 nRF51822 SoftDevice（蓝牙协议栈），通常为 s110_nrf51822_xxxx.zip 压缩包。

并按照如下步骤操作:

1. 安装 nRFgo Studio（根据 windows 版本选择 nrfgostudio_win-32_xxxx_installer.msi 或者 nrfgostudio_win-64_xxxx_installer.msi），并打开。
2. 在 Device Manager 中选择 nRF51 Programming



3. 选择 **Program SoftDevice** 标签栏.

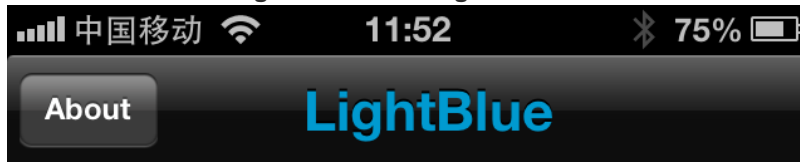


4. 点击 **Browse** 并找到需要下载的 SoftDevice 文件（在 s110_nrf51822_xxxxx.zip 内），本例中使用 **s110_nrf51822_5.2.1_softdevice.hex**.
5. 点击 **Program**.
6. 至此蓝牙协议栈下载完毕，下面演示下载上层应用程序。

注意：当下载非蓝牙程序时，如简单的“跑马灯”（上一节中的 **blink**）等，需要使用 **nRFgo Studio** 将芯片上的蓝牙协议栈擦除，才能正常下载。

3.3.2 编译, 下载, 并运行 ble_app_proximity 演示程序

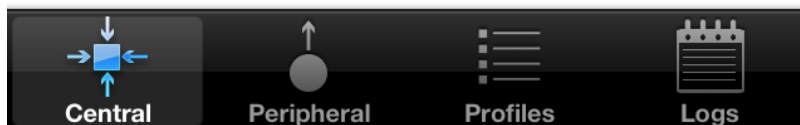
1. 找到 **ble_app_proximity** 工程, 路径为 <keil path> \ARM\Device\Nordic\nrf51822\board\PCA10001\ble\ble_app_proximity\arm
2. 双击 **ble_app_proximity.uvproj** 文件, 打开 Keil μ Vision
3. 单击 **Build** 图标, 或者按 **F7** 编译工程.
4. 打开 **Flash** 菜单, 并单击 **Download** 下载程序(或者点击 **Load** 图标).
5. **LED 0** 将会闪烁, 表示它正在广播.
6. iPhone 手机请安装 **Lightblue**, 打开 **Lightblue** 程序。见如下图片:



Scanning for Peripherals...



Central mode lets you connect to peripherals around you.





中国移动 11:52 75%
Peripherals **Services** RSSI -55

Peripheral: **Nordic_Prox**

[Clone Peripheral](#)

Services Found

0x1800 UUID: 1800	>
0x1801 UUID: 1801	>
Tx Power UUID: 1804	>
Immediate Alert UUID: 1802	>
Link Loss UUID: 1803	>
Battery Service UUID: 180F	>

Central Peripheral Profiles Logs

4 Application kit 开发环境配置

本节演示了如何下载第三方工具和如何设置开发环境。

4.1 开发环境

ARM compiler/IDE (本套件不包含)

所有的源代码都可以使用 Keil Microcontroller Development Kit (MDK)来编译。访问 <https://www.keil.com/demo/eval/arm.htm> 下载并安装 KEIL MDK-ARM Lite 。

J-Link OB 驱动 (本套件不包含)

访问 <http://www.segger.com/jlink-software.html> ，或者使用网盘提供的安装文件。请务必正确安装驱动，并确保能够被 Keil MDK 正确调用。请看 page 45 上的 *附录A*。

4.1.1 开发环境设置

nRF51822 能够通过几种方式来下载程序，本节将演示使用 Keil MDK-ARM 的方式。*Figure2* 演示了调试硬件和软件的环境。

注意：Keil μ Vision IDE 需要单独购买。

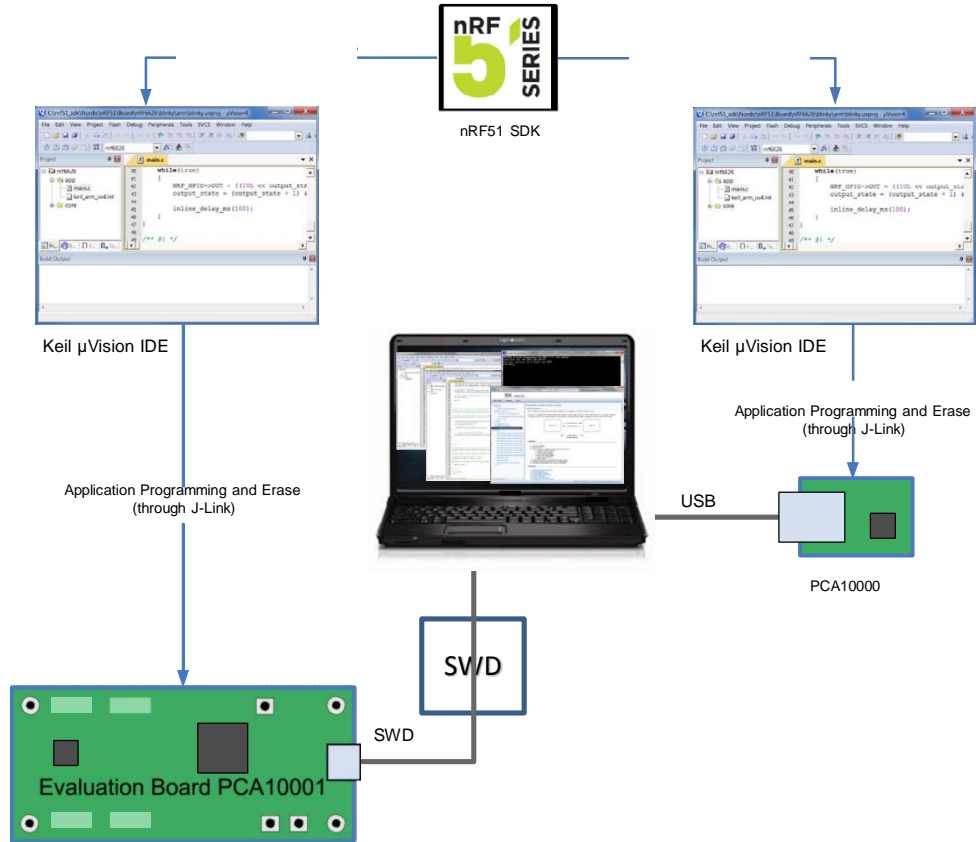


Figure 2 nRF51822 Application Kit 开发环境

5 硬件描述 Hardware description

本节描述了 nRF51822 Application Kit 的硬件部分。

5.1 nRF51822 Application Kit board (PCA10001)

nRF51822 Application Kit board (PCA10001) 是一个独立的应用开发板，包括加速度计、陀螺仪、USB 调试接口以及电流检测（CSM）。板内已经下载好了测试程序。具体测试方法请见说明书。

5.1.1 关键特性

nRF51822 AK II board (PCA10001) 拥有如下特性：

- nRF51822 带可擦写 Flash 的 SoC（单芯片）解决方案
- *Bluetooth low energy*（低功耗蓝牙）
- 2.4 GHz 与 nRF24L 系列芯片兼容
- 用户可编程的按键和 LED
- 全部引出的 I/O 接口，可外扩各种功能模块
- 带有加速度传感器、陀螺仪等体感元件
- 电流检测（CSM）
- USB 调试接口（CP2102）

5.1.2 硬件图片

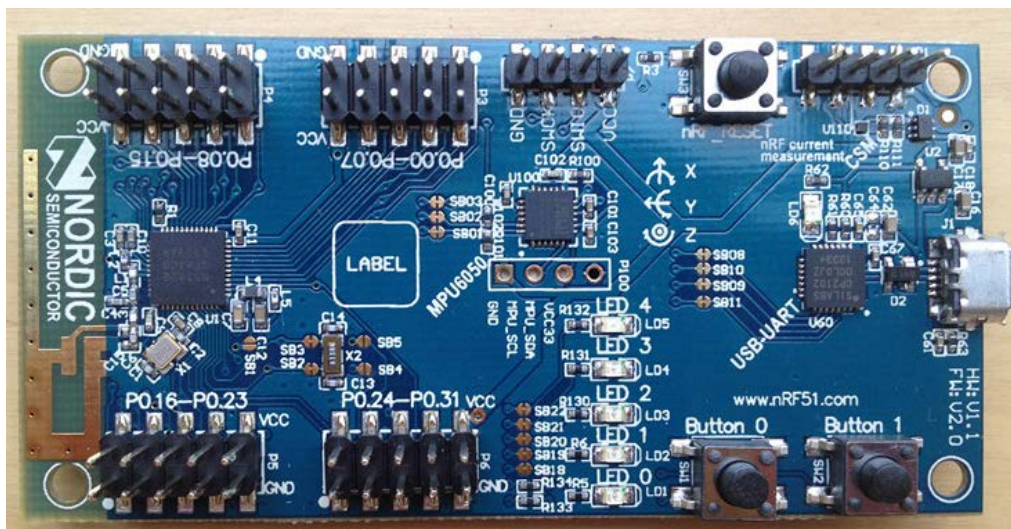


Figure 4 PCA10001 top

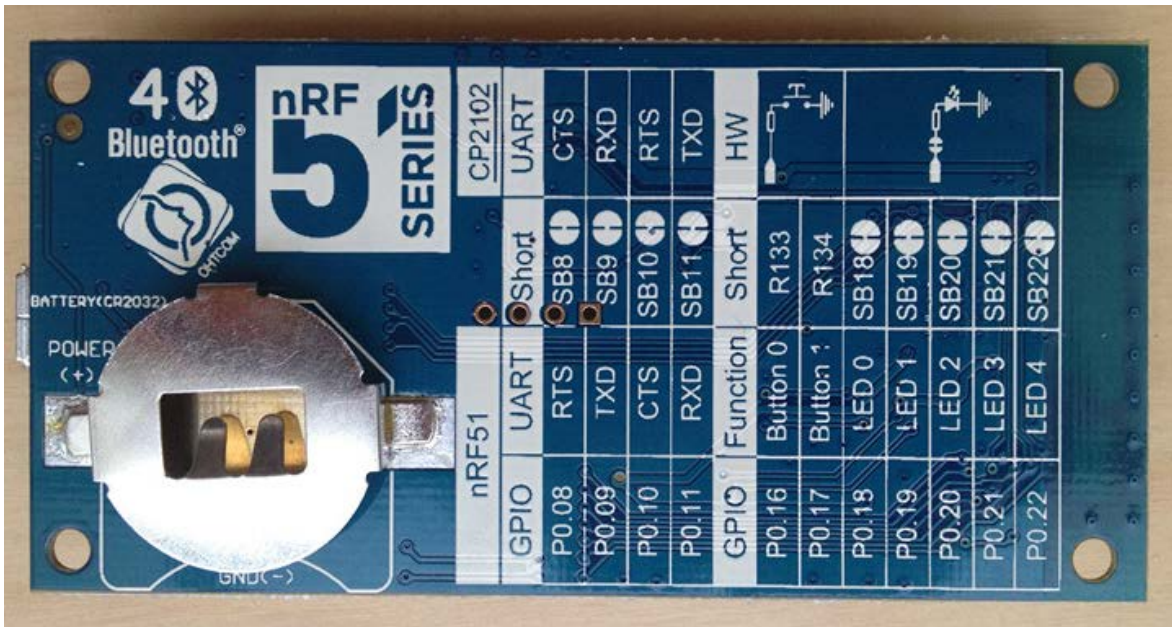


Figure 5 PCA10001 bottom

5.1.3 Block diagram

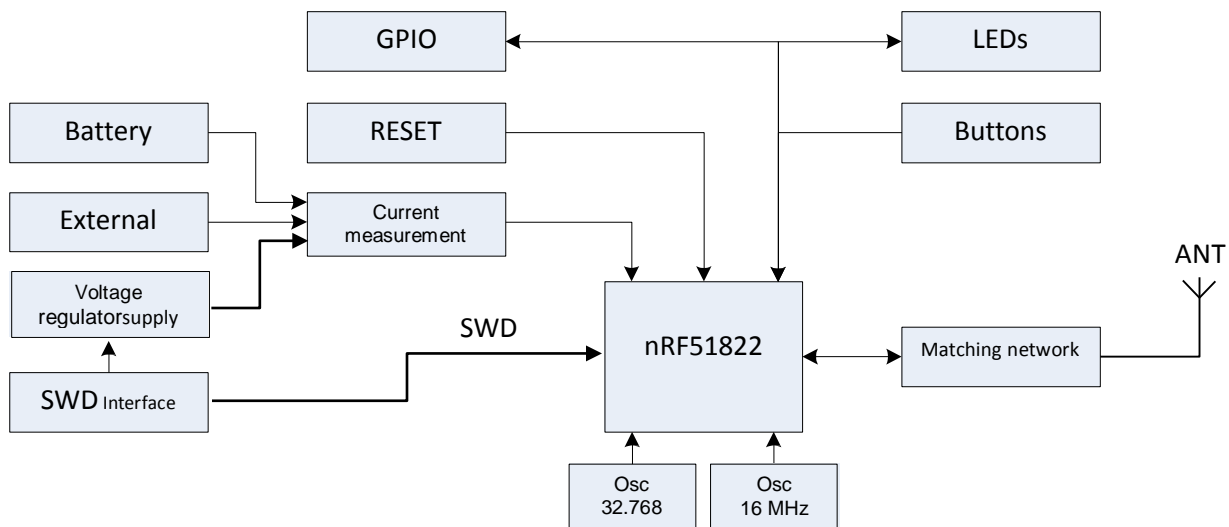


Figure 6 PCA10001 block diagram

5.1.4 复位按键

nRF51822 AK II board (PCA10001) 带有一个复位按键 (SW3)，该引脚与 SWD 调试接口的 SWDIO 复用，连接调试器时无法使用。

当使用 nRF51822 USB Dongle 时，软件程序内自带了复位功能，不需要手动复位。

5.1.5 供电

AK II board 有几种可选的供电方式:

- SWD(使用 USB dongle) **推荐**
- 外部电源, 通过 **USB**
- CR2032 coin cell battery (请看 **Figure 8**)

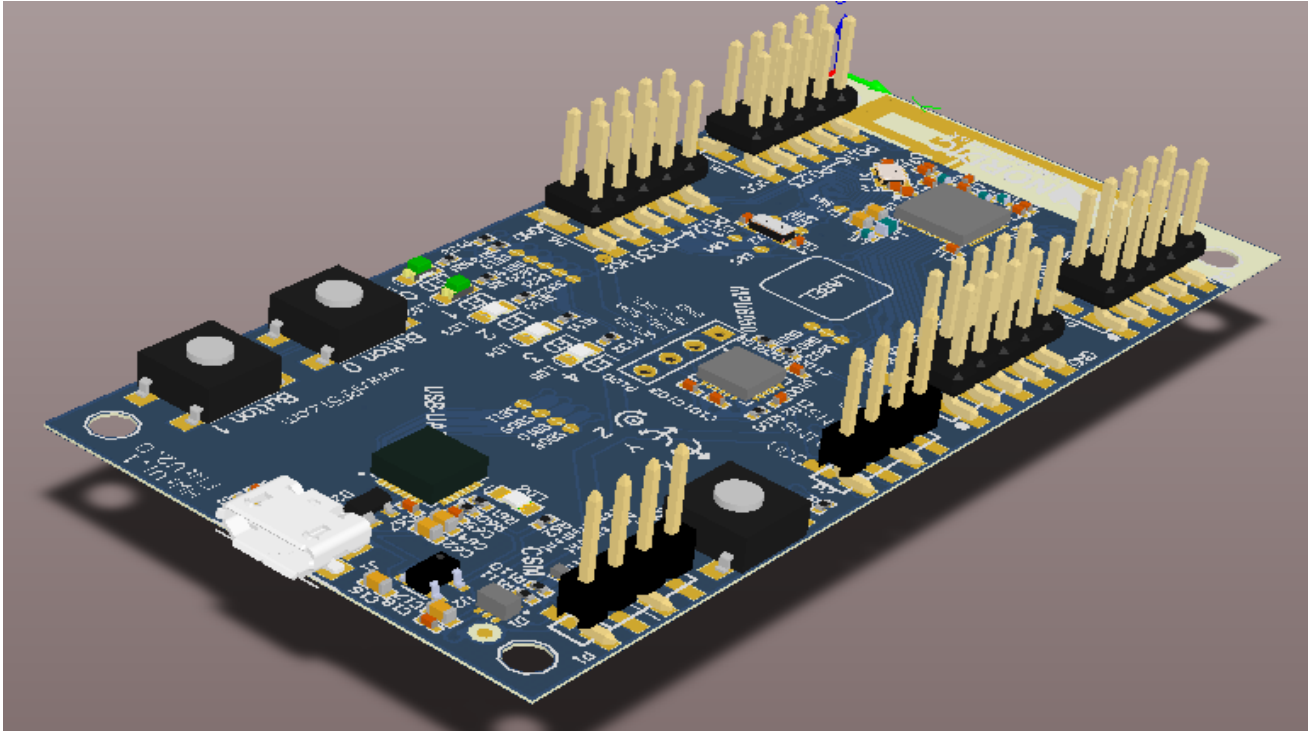


Figure 7 USB 和外部电源

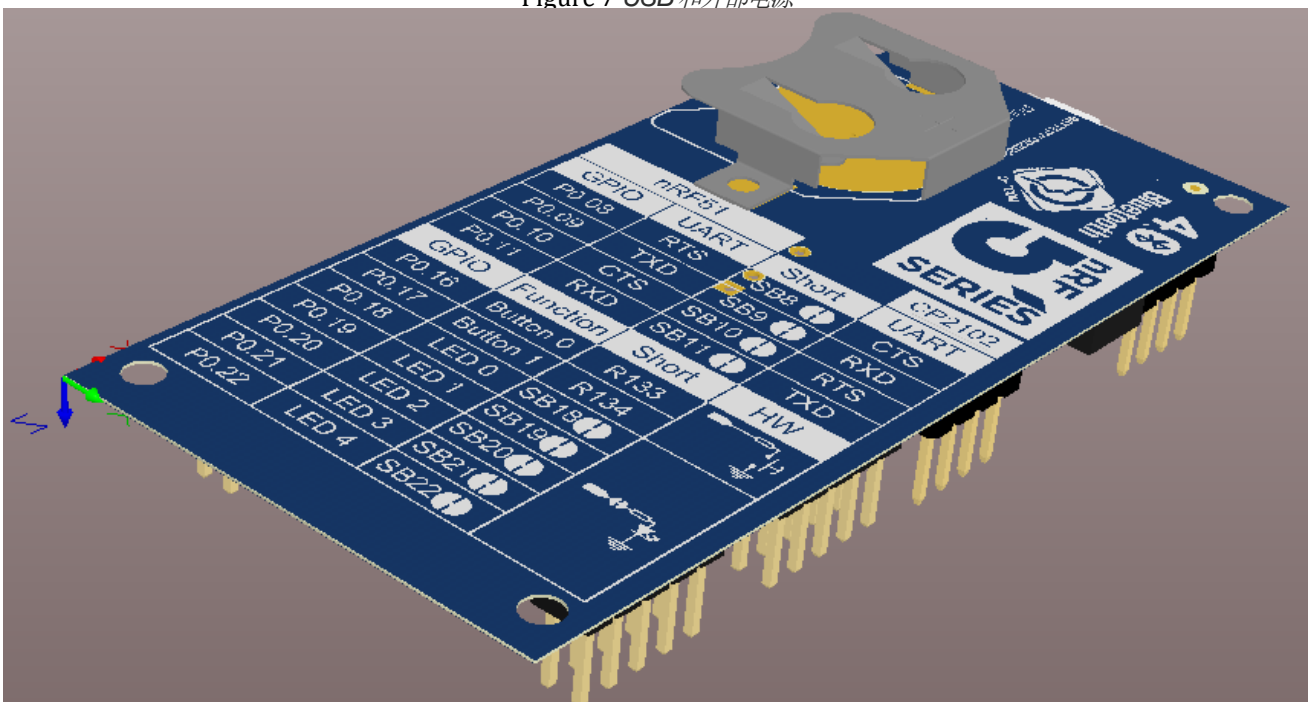


Figure 8 CR2032 纽扣电池

所有这些电源均通过一个防反二极管 (D1A, D1B, D1C), 所以系统电压将由电压最高的那个电源决定。

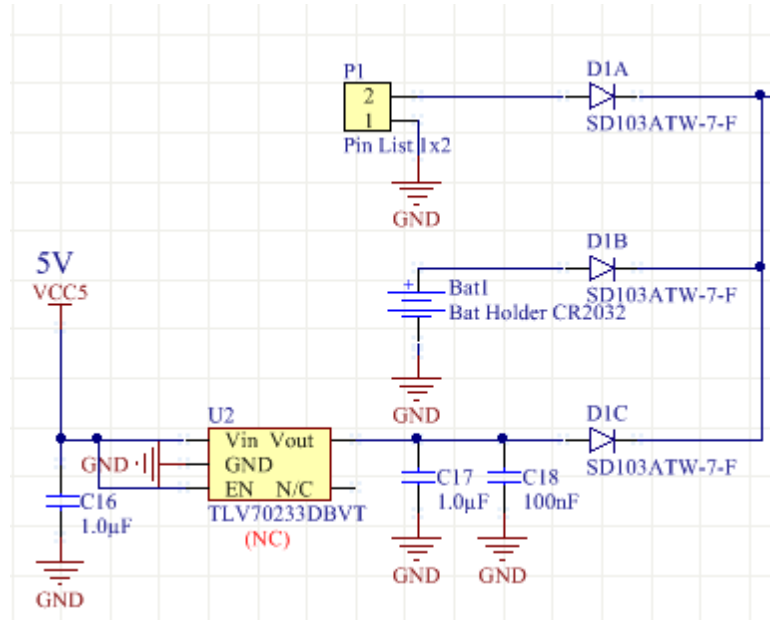


Figure 9 Power supply circuitry

5.1.6 GPIO 接口

P3, P4, P5, P6 为 AK II board PCA10001 的 GPIO 接口

P3			P6				
P0.00	1	2	P0.01	VCC	10	9	GND
P0.02	3	4	P0.03		8	7	P0.30
P0.04	5	6	P0.05	P0.29	6	5	P0.28
P0.06	7	8	P0.07	P0.27	4	3	P0.26
GND	9	10	VCC	P0.25	2	1	P0.24

P4			P5				
P0.08	1	2	P0.09	VCC	10	9	GND
P0.10	3	4	P0.11	P0.23	8	7	P0.22
P0.12	5	6	P0.13	P0.21	6	5	P0.20
P0.14	7	8	P0.15	P0.19	4	3	P0.18
GND	9	10	VCC	P0.17	2	1	P0.16

Figure 10 PCA10001 GPIO pin headers

注意:一些引脚已经在板上指定了功能, 可能需要断开相应的 Shortcut。



- P0.26 和 P0.27 默认是用来连接 32.768 kHz crystal，所以 **P6** 的这两个 GPIO 不可用。更多信息请看 **section 5.1.9 “32.768 kHz crystal”**
- P0.16----P0.22 默认是用来连接 LED 和按键。更多信息请看 **section 5.1.8 “按键和 LED”**
- P0.01, P0.02, P0.03 默认是用来连接陀螺仪、加速度计。更多信息请看 **section 5.1.10 “More configuration”**

5.1.7 按键和 LED

AK II board PCA10001 上有两个按键和五个用户可编程的 LED，它们被连接到 nRF51822 芯片的特定引脚上。请看 **Table 1**。

Part	GPIO	Short
Button 0	P0.16	
Button 1	P0.17	
LED 0	P0.18	SB6
LED 1	P0.19	SB7
LED 2	P0.20	SB8
LED 3	P0.21	SB9
LED 4	P0.22	SB10

Table 1 按键和 LED 连接

如果 GPIO P0.18 到 P0.22 需要用于别的功能，可以割断短接焊盘 **SB6, SB7, SB8, SB9, SB10**，请看 **Figure 12**。

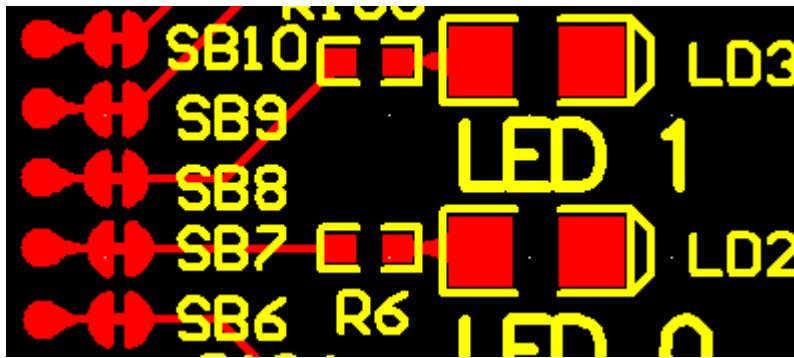


Figure 12 Disconnecting the LEDs and button

按键是低电平有效的，就是说按下按键后 IO 的输入端会被连接到地。按键没有外部的上拉电阻，所以需要使能引脚的内部上拉电阻。

LED 是高电平有效的，就是说写一个逻辑的 ('1') 到相应的输出引脚，将点亮 LED。

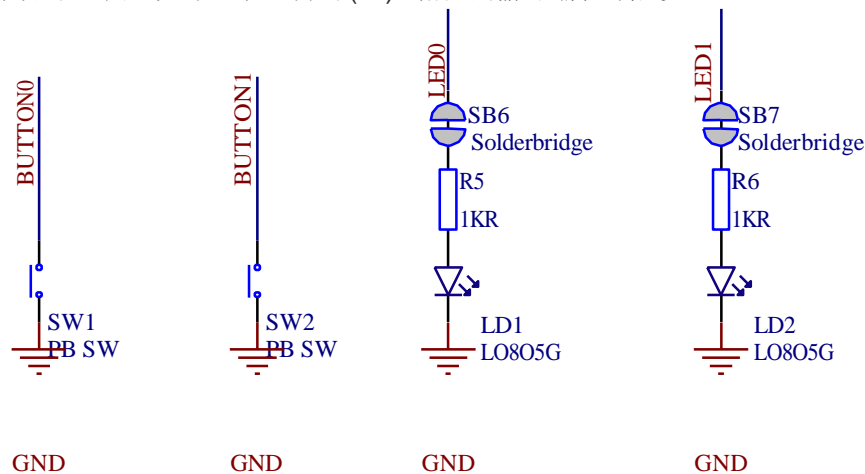


Figure 13 Button and LED configuration

5.1.8 32.768 kHz 晶振

nRF51822 可以选装一个 32.768 kHz 的晶振 (X2)，以便得到更高的精确性和更低的负载功耗。在 AK II board 上，P0.26 和 P0.27 默认用来连接 32.768 kHz 晶体，不能用作 GPIO（位于 P6 内）。

如果 P0.26 和 P0.27 想要当做普通 I/O 使用，可以割断短接焊盘 SB2 和 SB3，然后焊接 SB4 和 SB5。请看 Figure 14。

注意:使用 SDK 提供的 BLE 例程时，必须使用 32.768 kHz 晶体才能工作。

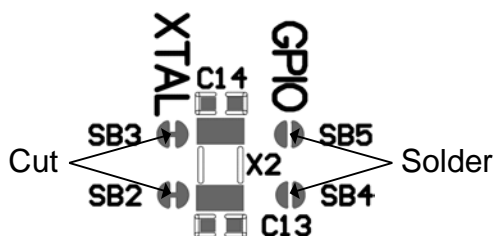


Figure 14 断开 32.768 kHz crystal 并连接 P0.26 和 P0.27 到 P6

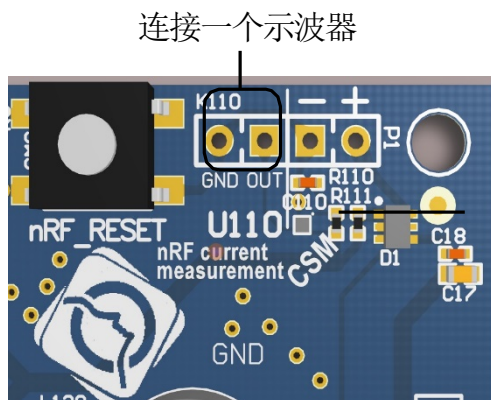
5.1.9 电流测量 (CSM)

PCA10001 板上的 nRF51822 消耗的电流可以通过 Current Shunt Monitor (CSM) 来检测。板载的 INA216 (U2) 的增益为 200V/V，所以保证了最小的压降。

在 K110 上连接一个示波器。

具体的计算公式：

电流值=电压值/200



5.2 USB Dongle II

nRF51822 USB Dongle II 是 nRF51822 的最小化开发平台。他板载了一个用于编程和调试的 Jlink V7 芯片，可以通过跳线选择对外部编程。

5.2.1 关键特性

PCA10000 拥有如下关键特性:

- nRF51822 SoC
- *Bluetooth low energy* 设备兼容
- 与 2.4 GHz 的 nRF24L 设备兼容
- Jlink V7, 支持固件更新
- 多彩 RGB LED and 两个 GPIO

5.2.2 硬件图片

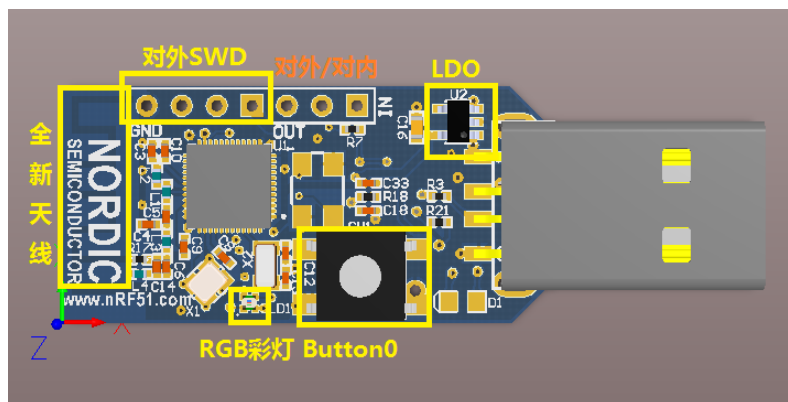


Figure 16 PCA10000 top side

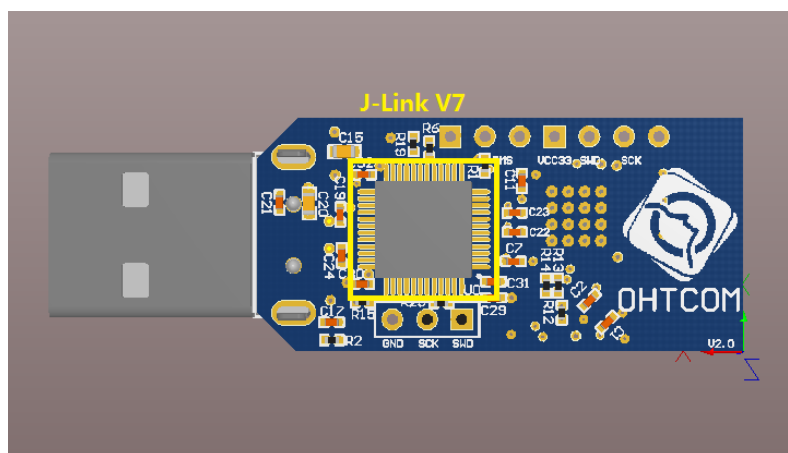


Figure 17 PCA10000 bottom side

5.2.3 方框图

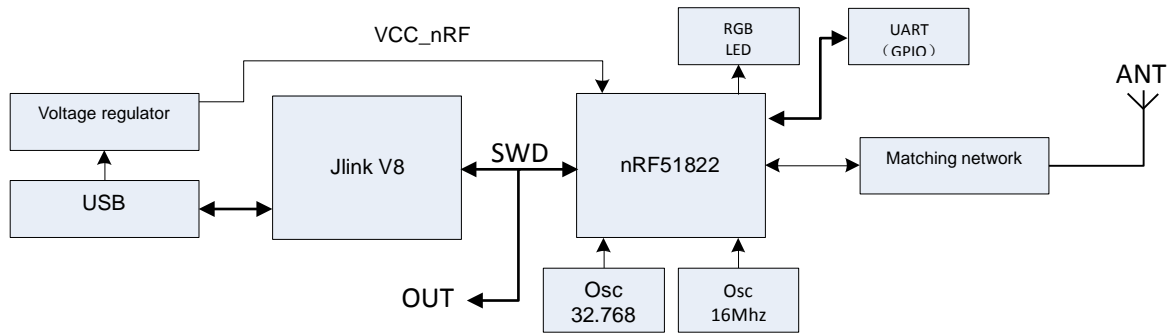


Figure 18 PCA10000 block diagram

5.2.4 多彩 LED

USB Dongle (PCA10000) 板载了一个多彩 RGB LED。LED 连接到了 nRF51822 芯片上特定的 I/O 端口。请看 **Table 3**。

Figure 18 USB dongle 接口说明

Color	GPIO
Red	P0.21
Green	P0.22
Blue	P0.23

Table 3 LED connection

LED 低电平有效，就是说写一个逻辑('0') 到相应的输出引脚，即可点亮 LED。

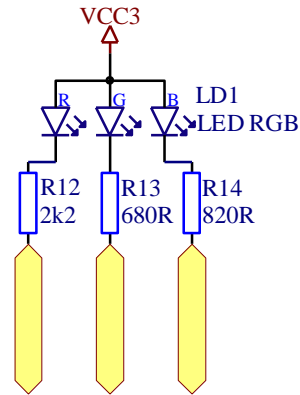


Figure 19 LED configuration

5.2.5 Button 配置

nRF51822 USB Dongle 的 Button 线连接到 pins P0.16，可以用于演示官方 BLE 例程。

6 Flash 下载和应用程序开发

nRF51822 芯片有两种开发方式：直接在上面编写应用程序（就像单片机一样），或者使用 nRF51822 的 SoftDevice，nRF51822 SoftDevice 是一个低功耗蓝牙（BLE）协议栈。更多信息请看 S110 SoftDevice Specification。

本章我们将描述如何下载（和擦除）nRF51822 上的 S110 nRF51822 SoftDevice 或其他应用程序的 HEX 文件。

如果你想开发一个不使用 S110 SoftDevice 的应用程序，请看 **section 6.1.5 “Programming an application”** on page 28.

6.1 使用 nRFgo Studio 下载

使用 nRFgo Studio 下载（和擦除）SoftDevice 或其他应用程序的 HEX 文件。

注意： 内存管理和保护的更多细节请看 *nRF51 Series Reference Manual*。

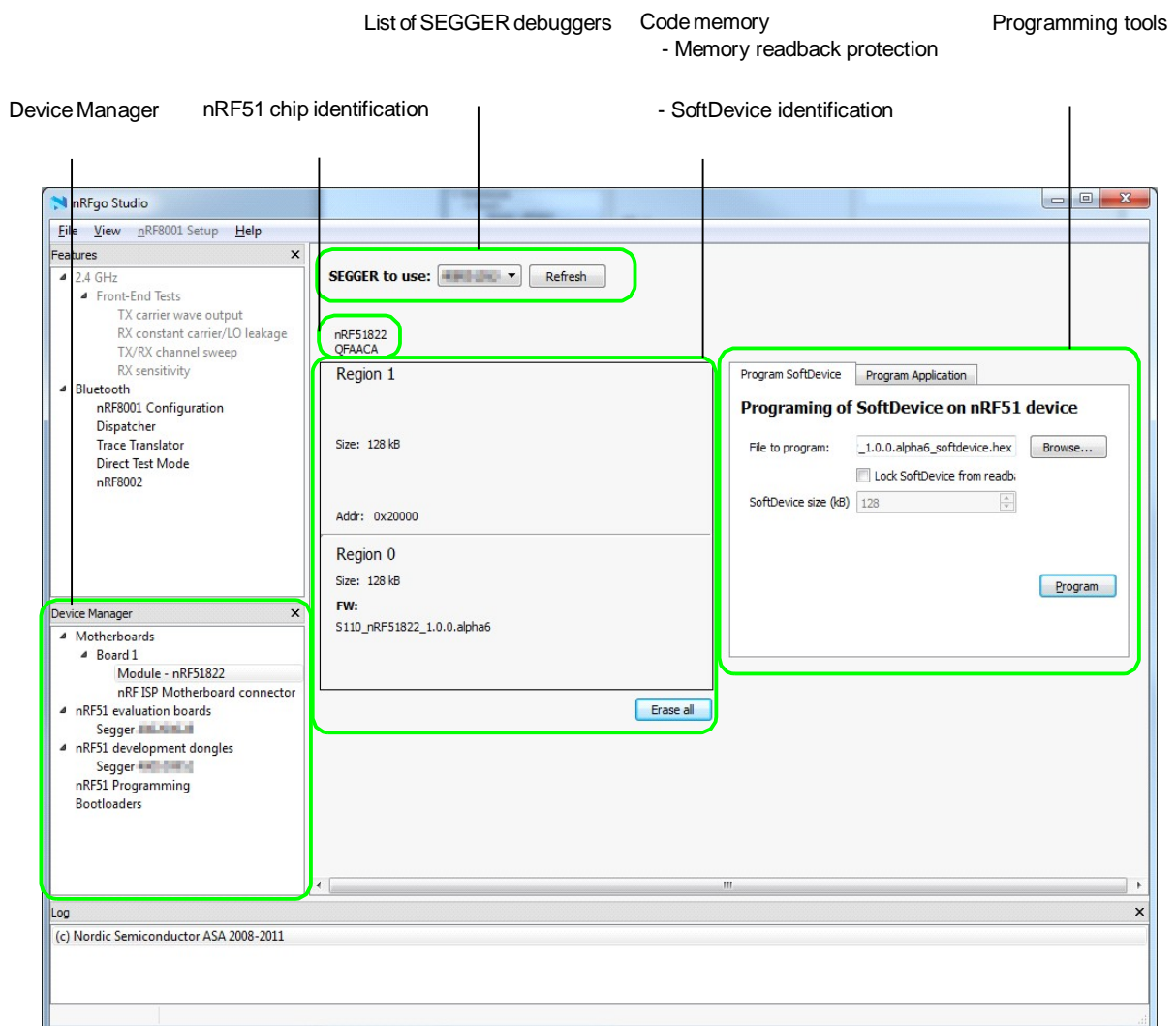


Figure 20 nRFgo Studio dashboard

6.1.1 进入下载界面

1. 打开 nRFgo Studio.
2. 在左下方的 Device Manager 栏中选择 nRF51 Programming。

6.1.2 识别 nRF51822 芯片和内存

进入下载界面后, nRFgo Studio 会识别 nRF51822 芯片并显示它的内存是如何分配的。下面是芯片和内存信息的含义:

- **nRF51 chip identification** – 用来识别芯片和编译代码 (例如, nRF51822 QFAACA) 如果调试器没有连接到芯片, 或者与芯片通讯异常, 将会显示如下信息: *“No device detected. Ensure that you have the SEGGER connected correctly to the board and that the board is powered and configured for debugging.”*
- **Code memory** – 显示代码空间 (code memory) 是如何分割的。显示一个或者两个区段和它们的大小。对于一个包含 SoftDevice (协议栈) 的芯片, 代码空间被分成了两个区段, SoftDevice 在 Region 0。这个工具会告诉你 SoftDevice 占用了多大的空间, 还剩下多少空间给应用程序使用。
 - **Memory readback protection** – 显示 readback protection 是否被勾选了。有两个可能的选项: Region 0 的 readback protection 或者整片 readback protection。如果只有一个 region 这个选项会变成 readback protection on (All) or off.
 - **SoftDevice identification** - nRFgo Studio 会试图识别 Region 0 上的固件 (firmware), 能够被识别的固件的 ID 将会显示出来; 无法识别的 firmware 将会显示 FWID number.

6.1.3 擦除

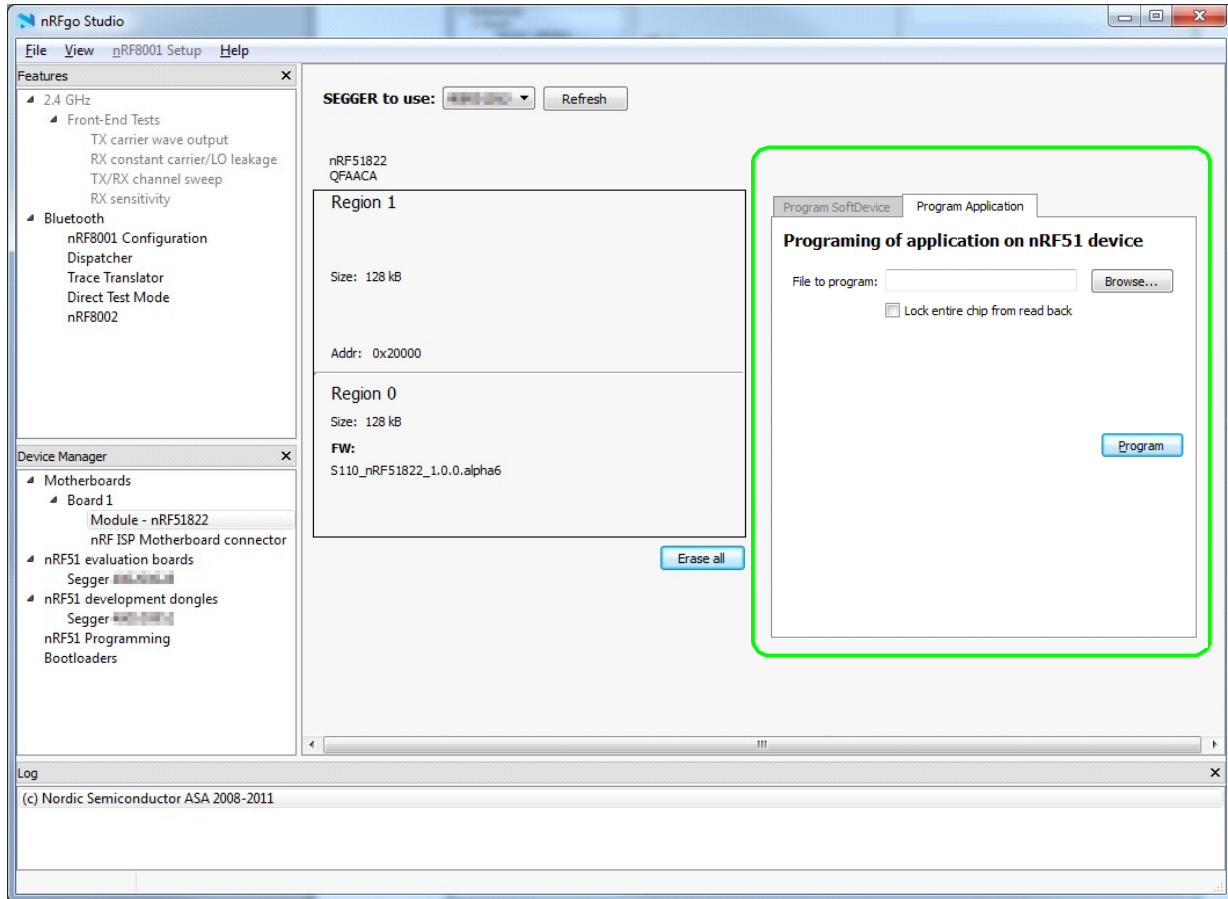
如下情况应该使用擦除功能:

- 芯片已经下载了 SoftDevice, 但是你想要重新下载其他不使用蓝牙的应用程序。
- 在 nRFgo Studio 中勾选了 “Lock entire chip from readback”

Erase All 方法说明, 完成 page 26 上的 **section 6.1.1 “进入下载界面”**, 然后点击 **Erase all**.

6.1.4 下载 SoftDevice

使用本方法可以下载 SoftDevice 到芯片上



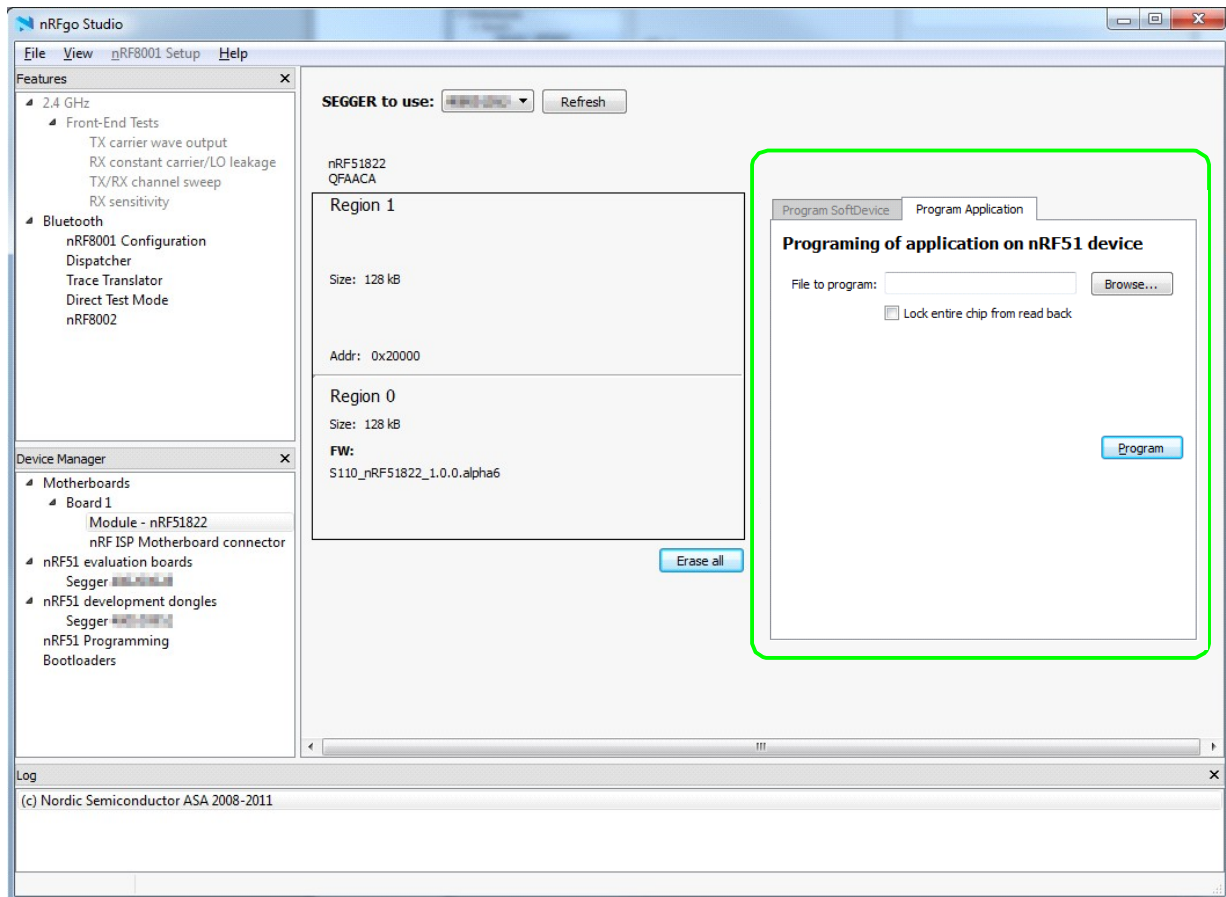
1. 完成 **section 6.1.1 “进入下载界面”**，然后选择 **Program SoftDevice** 标签.
2. 点击 **Browse** 并选择想要下载的 HEX 文件.
3. 决定是否勾选 Region 0 的 “Lock SoftDevice from chip”。用于锁定芯片，如果锁定只能全部擦除。
4. 设置 SoftDevice 大小（通常不需要）。这是设置 flash region 0 的大小,但是如果 HEX 文件已经确定大小就无法改变了。

注意: S110 nRF51822 SoftDevice 包含在 s110_nrf51822_xxxx.zip 文件中 (xxxx 替换为相应的版本号)

6.1.5 下载一个应用程序

在真正下载之前，nRFgo Studio 会检验 HEX 文件的起始内存地址是否与 SoftDevice 相匹配。如果没问题，nRFgo Studio 会开始下载，否则它将会停止并报错。例如：如果一个应用程序需要使用到 SoftDevice，点击 program 后，nRFgo Studio 会自动检测应用程序的 HEX 文件的内存设置是否与 SoftDevice 的设置相匹配。

注意：这个下载程序不会设置 memory Regions.



1. 完成 page 26 上的 **section 6.1.1 “进入下载界面”**，然后选择 **Program Application** 标签.
2. 点击 **Browse** 并选择要下载的 **HEX** 文件。
3. 决定是否勾选的“Lock entire chip from read back”。如果使能 readback protection，下次重新烧写之前就不得使用 Erase All。

下载时使能了“Lock entire chip from read back”后，**将无法进行开发调试**。解决方法是，再次 Erase all。Lock entire chip from read back 可以用来防止对芯片上程序的意外覆盖。

6.2 应用开发

应用程序的编译、链接、以及下载和 SoftDevices 是完全分离的。在已经下载好 SoftDevice 的芯片上开发和调试程序，就好像在一片完全空的芯片上操作一样。基本的区别只在于 memory layout 和 call stack size。

6.2.1 配置 memory layout

不同版本的 SoftDevice 可能会有不同的要求，请在操作前查看。

应用程序的向量表（vector table）地址，取决于芯片是否已下载了 SoftDevice。

SoftDevice 的下载地址从 0x0 开始，并且大小固定。应用程序的向量表（vector table）必须紧跟在 SoftDevice 之后。应用程序可以使用除去 SoftDevice 空间之外的内存空间。相应的，SoftDevice 数据区（data area）从最低的 RAM 地址开始，应用程序的数据区（data area）要跟在 SoftDevice data area 后面布置。

Table 6 演示了不同的 SoftDevice 对应用程序的代码空间和数据空间（code and data）的起始地址和大小的影响。这个演示基于 256 kB 代码空间（code memory）和 16 kB RAM 的芯片。

Device configuration	SoftDevice		App. code start address	Available code memory	App. data start address	Available RAM
	Code memory	RAM usage				
Blank chip	0 kB	0 kB	0x0	0x40000	0x20000000	0x4000
SoftDevice A	64 kB	2 kB	0x10000	0x30000	0x20000800	0x3800
SoftDevice B	128 kB	8 kB	0x20000	0x20000	0x20002000	0x2000

Table 6 SoftDevice memory layout

注意： 芯片详细的代码空间和可用 RAM 可以查看 *nRF51822 Product Specification*。SoftDevice 使用的代码空间 RAM 可以在 *S110 nRF51822 SoftDevice Specification* 中查询。

有两种方式可以配置内存布局（memory layout）：

- 使用 Keil IDE
- 使用 Scatter 文件

注意： Nordic Semiconductor 提供的例程，已经通过 Keil IDE 配置好了内存布局。当使用 Keil IDE 时不需要使用 Scatter 文件。

6.2.1.1 Keil 的内存布局（Memory layout）配置

进入Keil IDE 的memory layout:

1. 点击 **Project** 菜单，然后选择 **Options for Target**.
2. 选择 **Linker** 标签.
3. 点击 **Use memory layout from Target Dialog**.

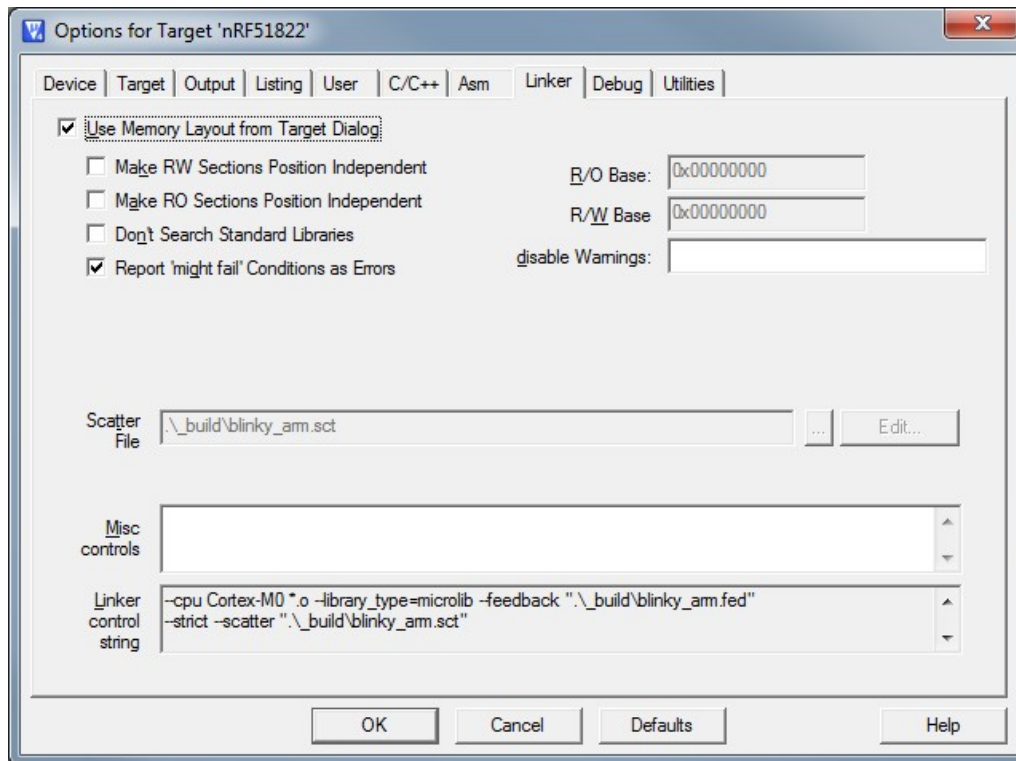


Figure 21 Keil linker settings

4. 选择 **Target** 标签.
5. 在 **Read/Only Memory Areas**, 定义 **Start** 和 **Size**.
6. 在 **Read/Write Memory Areas**, 定义 **Start** 和 **Size** , 见page31 的 **Figure 22**.
7. 点击 **OK**.

下面这个例子是一个使用 256 kB 的 code memory 和 16 kB 的 RAM 的应用，其中 SoftDevice 使用了 128 kB 的 code memory 和 8 kB 的 RAM (page 29 描述的 SoftDevice B **Table 6**).

- Base code memory address 0x20000 and available code memory size is 0x20000 (128 kB).
- Base RAM memory address 0x20002000 and available RAM size is 0x2000 (8 kB).

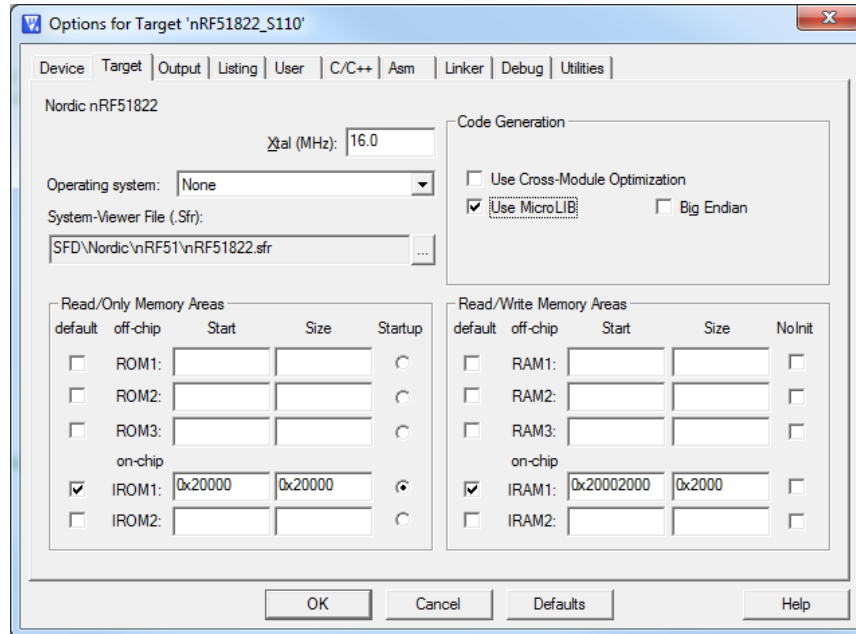


Figure 22 Memory layout with example SoftDevice

Memory		Description
IROM1	Start	指定应用程序 code memory 的 start address
	Size	指定应用程序的 code memory 大小
IRAM1	Start	指定应用程序 RAM 的 start address
	Size	指定应用程序的 RAM 大小

Table 7 内存布局 (Memory layout)

6.2.2 共享的 call stack

如果芯片烧写了 SoftDevice, 那么用户应用程序和 SoftDevice 共享同一个 call stack (堆栈)。应用程序必须保证 call stack 有足够的空间给自己和 SoftDevice。SoftDevice 的 call stack 空间需求跟不同的设备和不同的协议栈版本都有关系, 可以在 *S110 SoftDevice Specification* 中查找。

用户应用程序应该把 call stack 的大小设置为 它本身的需求加上 SoftDevice 的需求。然后设置堆栈指针 (stack pointer) 到应用程序的复位向量 (Reset Vector) 的首地址。

注意: 使用 Keil 和 ARMCC toolchain 时, call-stack 大小可以使用 `Stack_Size` 变量来定义, 它通常在工程的启动文件中, 比如 `arm_startup_nrf51.s`。(译者注: 通常可设置为芯片 RAM 的最大容量)

```
Stack_Size      EQU      0x400 ; The application call-stack size + protocol call-stack size
                AREA     STACK, NOINIT, READWRITE, ALIGN=3
Stack_Mem       SPACE   Stack_Size
__initial_sp
```

6.2.3 Debugger 配置

SDK 中提供的工程已经配置好了下载和 Debugger 选项。但是，如果使用新的工程的话，必须首先配置好 debugger 选项。如何配置请看：

1. 在 Keil 中，选择 **Project** 菜单下的 **Options for Target (ALT+F7)**。之后 会弹出 **Options for Target** 对话框。
2. 选择 **Debug** 标签。
3. 选中 **Use**，并在下拉列表中 选择 J-Link/J-Trace debugger。
4. 设置 **Driver DLL** 为 SARMCM3.DLL。
5. 设置 **Dialog DLL** 为 TARMCM1.DLL。

还有一些很有用的工具可以用来辅助调试程序。

- BreAK Ilpoints
- Load Application at Startup
- Memory Display
- Toolbox
- Watch Windows

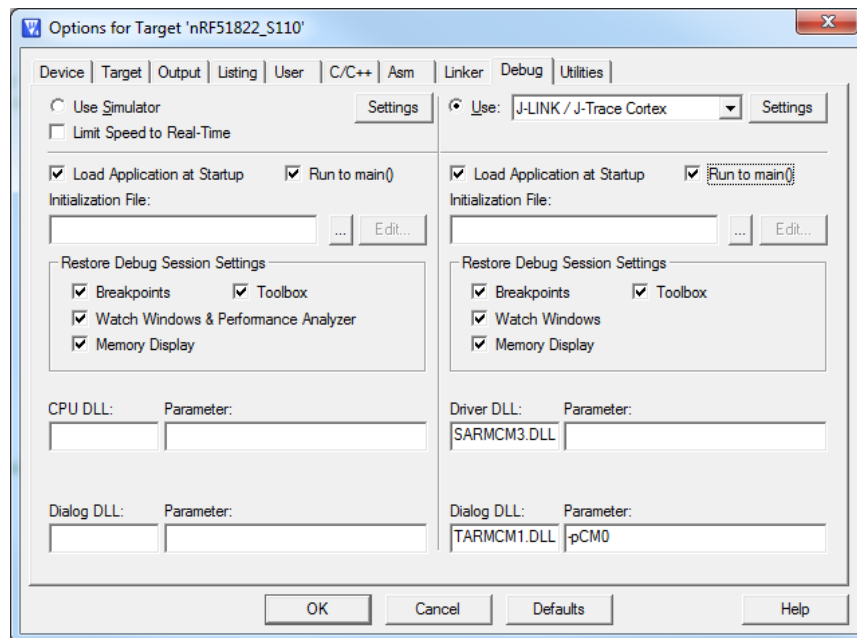


Figure 23 Debugger options

6. 点击右上角 **Use** 后面的 **Settings**。
7. 在 Target 的 driver setup 中，显示了 debugging protocol 和 maximum speed。在 **Port** 下拉列表中选择 **SW**。
8. 在 **Max Clock** 中的 调试接口的最高速度不要超过(1 MHz)。page 34 的 **Figure 24** 显示了一个正确的配置。

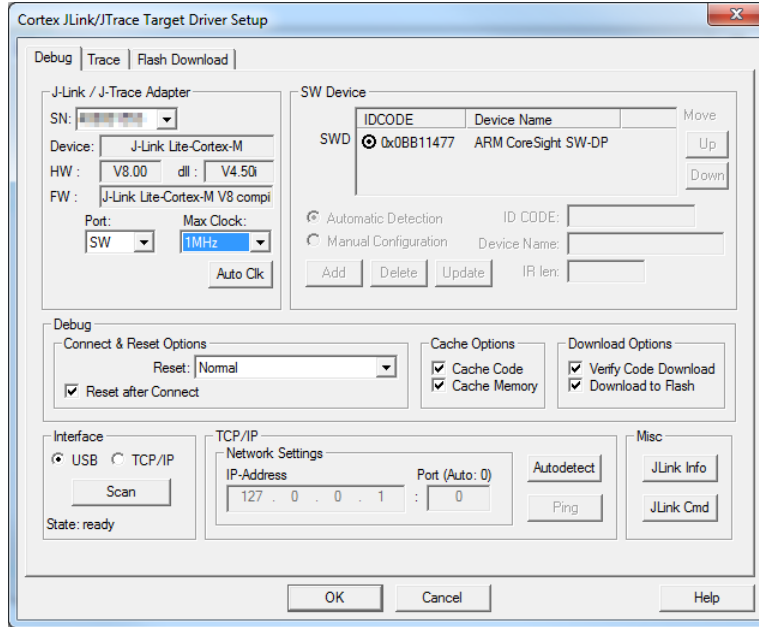


Figure 24 Target driver setup

9. 在 Keil IDE 中，点击 **Start/Stop Debug Session (CTRL+F5)** 就可以启动 debugging。

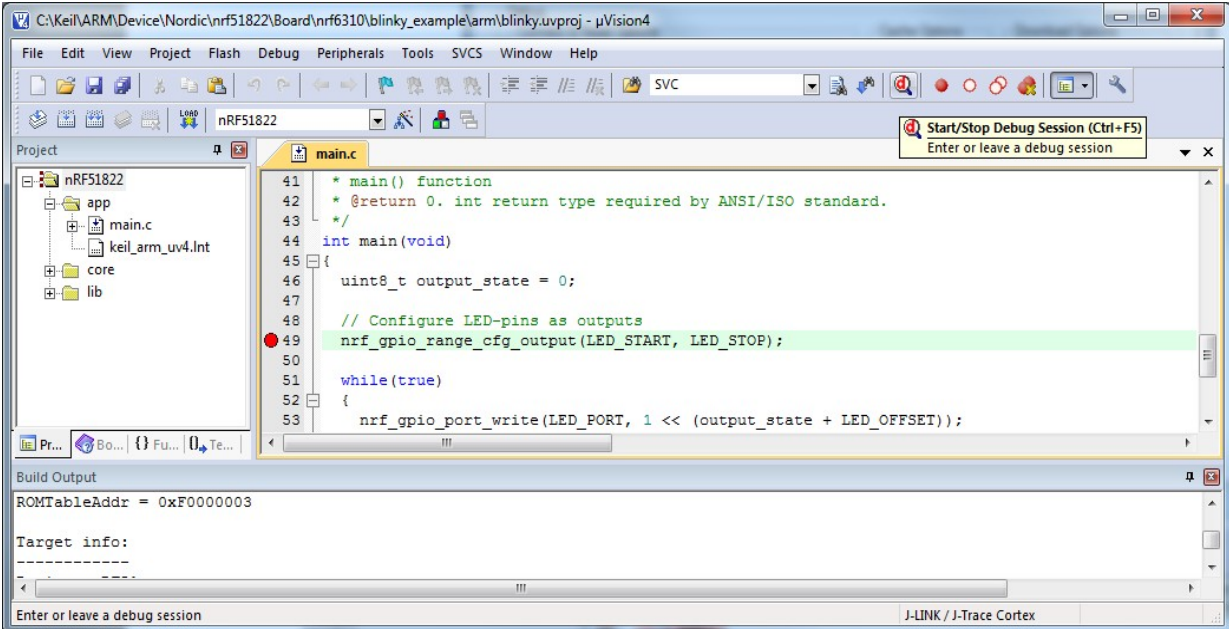


Figure 25 Debugger initiation

6.2.4 使用 SoftDevice 调试时的限制

如果芯片上下载了 SoftDevice，调试时将会有一些限制。

6.2.5 对芯片编程

为了保证 SoftDevice 的正确性，microcontroller 包含了一个 Memory Protection Unit (MPU)，它可以防止对特定区域的访问。debugger 读取这些区域将会返回 0x0000 (no operation instruction)。

被 SoftDevice 占用的空间将禁止 write 和 erase。使能 SoftDevice 后，MPU 将会对协议栈进行写保护。被保护的部分在 *S110nRF51822 SoftDevice Specification* 中有相应描述。配置好 debugger 和编译器 (compile) 还有连接器 (link) 之后，点击 Keil IDE 的 download 按钮就可以下载程序。下面演示了如何配置和启动下载：

1. 选择 **Project** 菜单下的 **Options for Target**
2. 选择 **Utilities** 标签
3. 点击 **Settings**
4. 选择 **Flash Download**
5. 选择 **Erase Full Chip**
6. 点击 **Add** 并选择列表中的 nRF51xxx
7. 点击 **Download**.

注意： nRF51xxx 的相关程序会随同 SDK 一起安装好。这些程序是 nRF51 系列芯片公用的，有了它们就能够下载所有代码空间 (code memory) 小于 2M 的 nRF51 系列芯片。

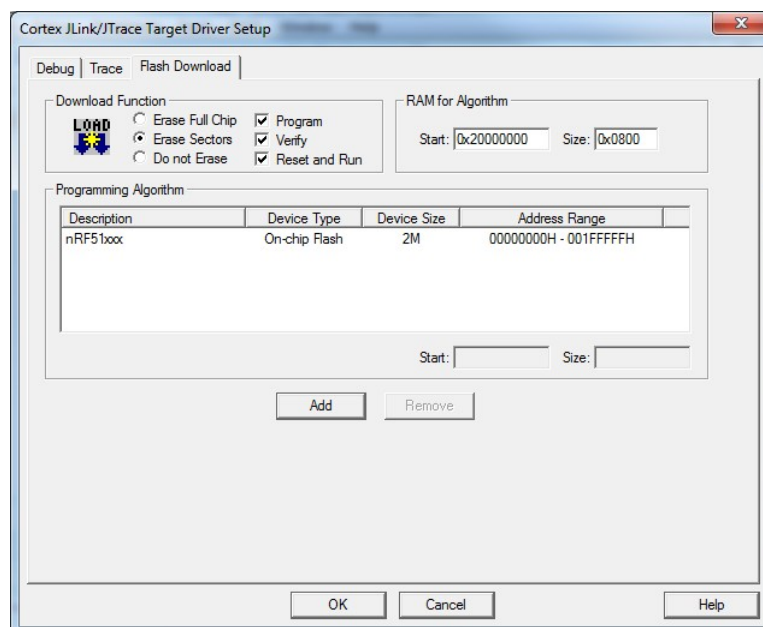


Figure 26 Selecting J-LINK/J-Trace Cortex

8. 进入 **Utilities** 标签, 选择 **Use Target Driver for Flash Programming**.
9. 在下拉菜单中选择适当的调试器 **Figure 27**.

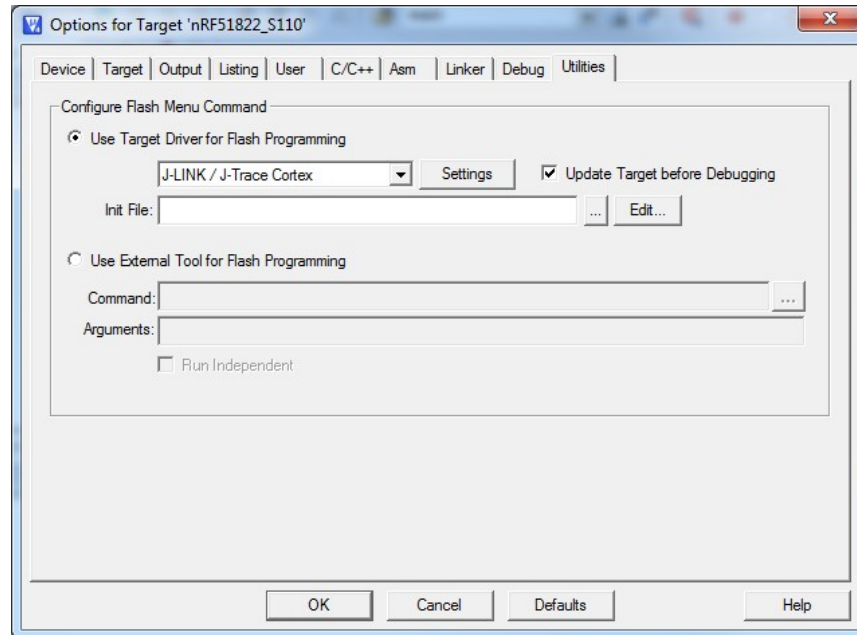


Figure 27 Debugger selection

7 Debugging nRF51822

使用 SEGGER J-Link Debugging 请看 page 45 的 [附录 A](#)。使用 Keil μ Vision IDE 的通用调试方法, 请看 <http://www.keil.com/uvision/debug.asp>.

下面的步骤描述了如何在 Keil 中调试 nRF51822

1. 打开 **Project** 菜单, 点击 **Options for Target (CTRL+F7)**.
2. 选择 **Debug** 标签.
3. 点击 **Start/Stop Debug Session (CTRL+F5)**

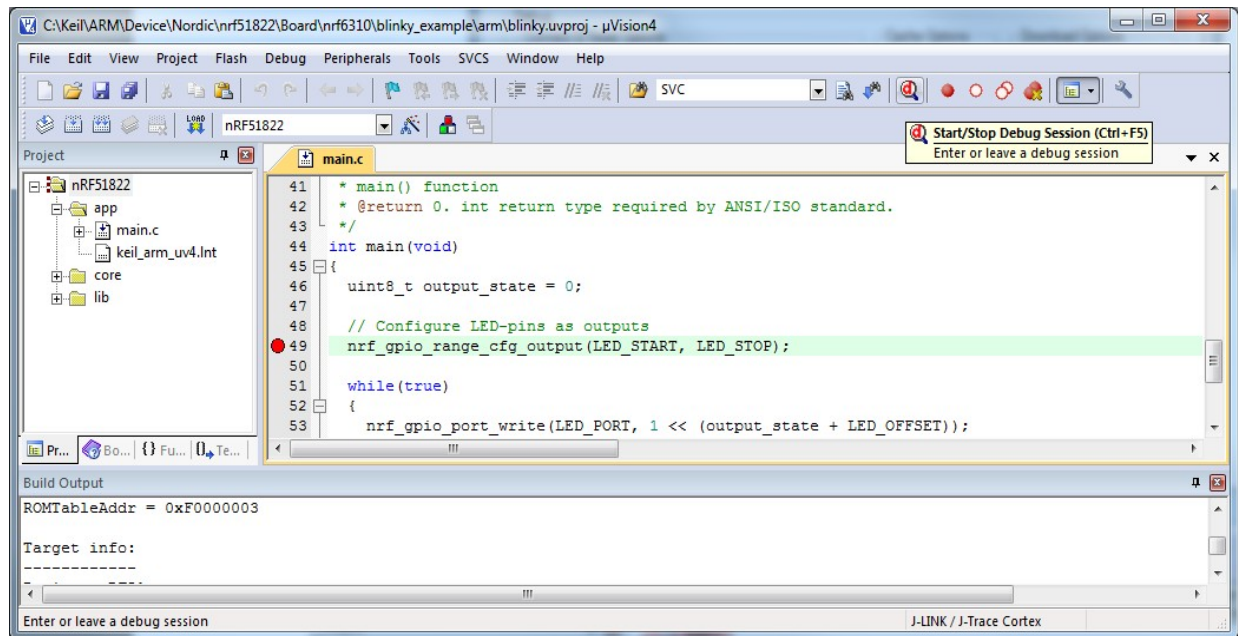


Figure 28 Start debugging mode

7.1 nRF51822 debug 特性及要点

本节包括几个内容：System Viewer Windows，如何在 SoftDevice readback protected 的情况下调试应用程序，以及如何使用 SEGGER J-Link debugger 设置断点。

7.1.1 System Viewer 窗口

System Viewer 可以通过一个独立的窗口查看各种外设的寄存器信息。请看 **Figure 29**。在 **View** 菜单中移动光标到 **System Viewer** 并选择你想要查看的外设。这个外设的寄存器信息将会显示在 Keil 右边的栏内。请看 **page 39** 的 **Figure 30**。更多关于 System Viewer 的信息可以点击以下网址查找：
http://www.keil.com/uvision/db_view_sysview.asp.

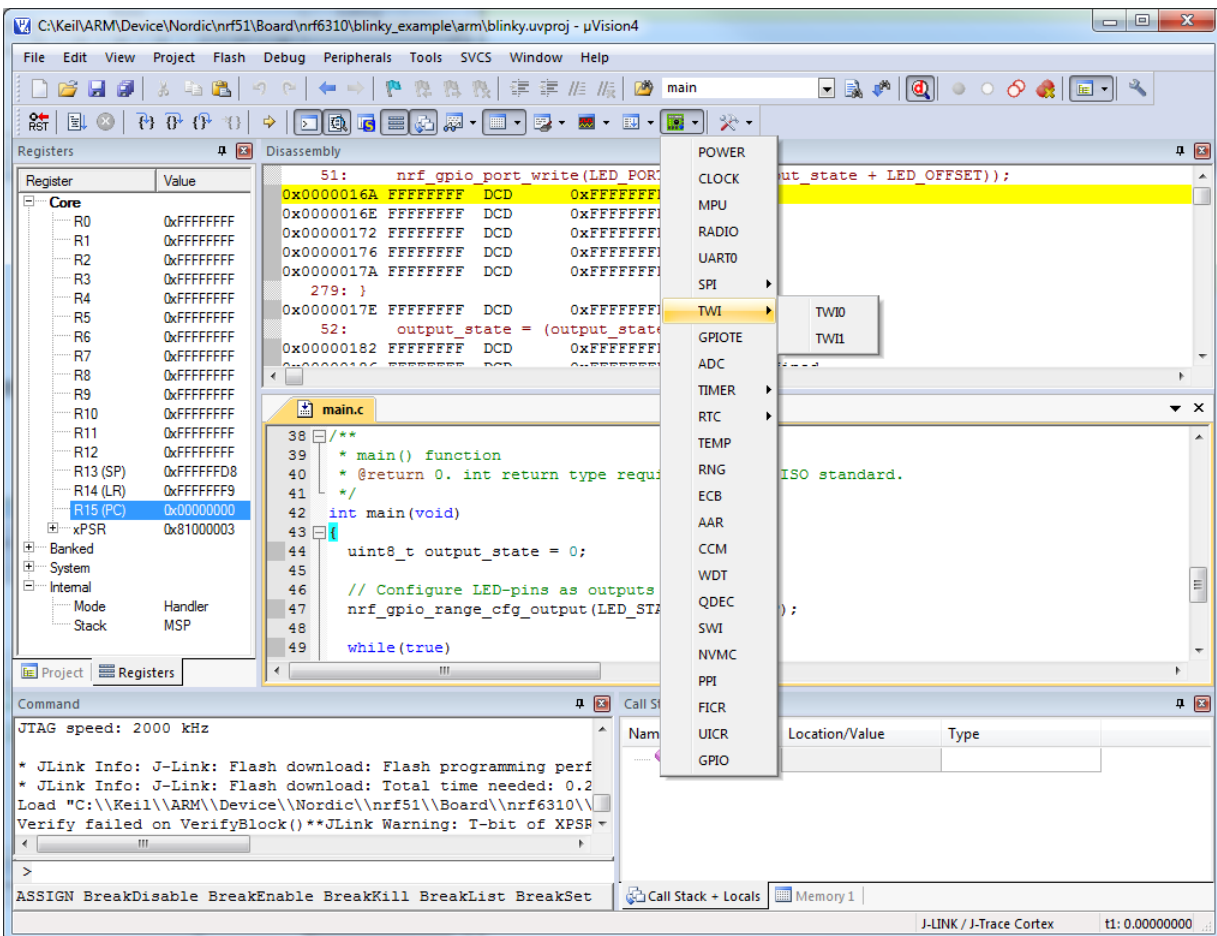


Figure 29 System Viewer Windows

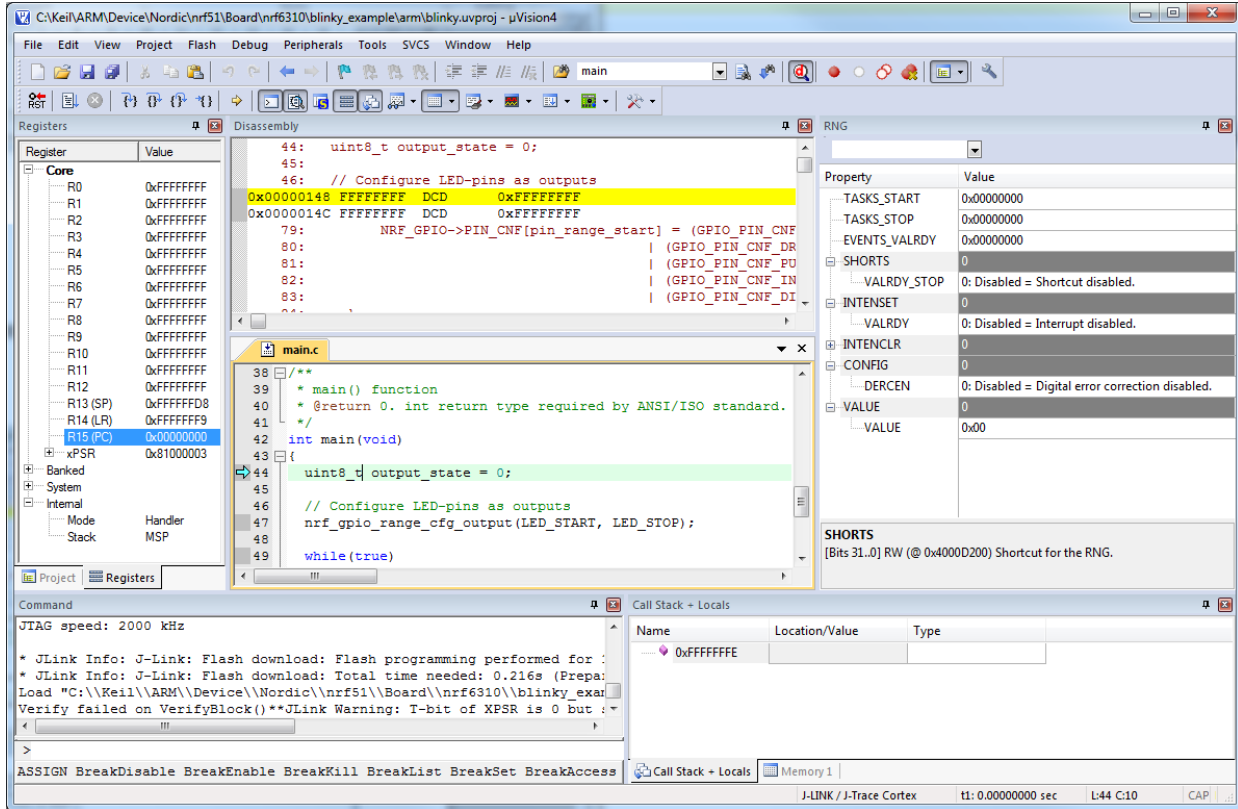


Figure 30 System viewer window of the RNG peripheral

7.1.2 调试程序，在 SoftDevice readback protected 的情况下

带有 SoftDevice 的应用程序调试可以参考 <http://www.keil.com/uvision/debug.asp>。但是，如果对 SoftDevice 所在的 Region0 使能了 readback protection，那么从这个保护区内读取的代码字均为 zero。

任何外设中受 SoftDevice 限制的寄存器也无法被 debugger 看到。SoftDevice 配置和内存资源分布图可以在 *S110 nRF51822 SoftDevice Specification* 中查找。

注意：请避免单步调试保护区域（protected area）。替代方法是，在 SVC calls 后面设置断点，然后直接运行到该断点，请看 page 41 上的 **Figure 32**。也可以使用“step over”方法 (**F10**) 跳过 SVC calls，避免进入 readback protected 区域。

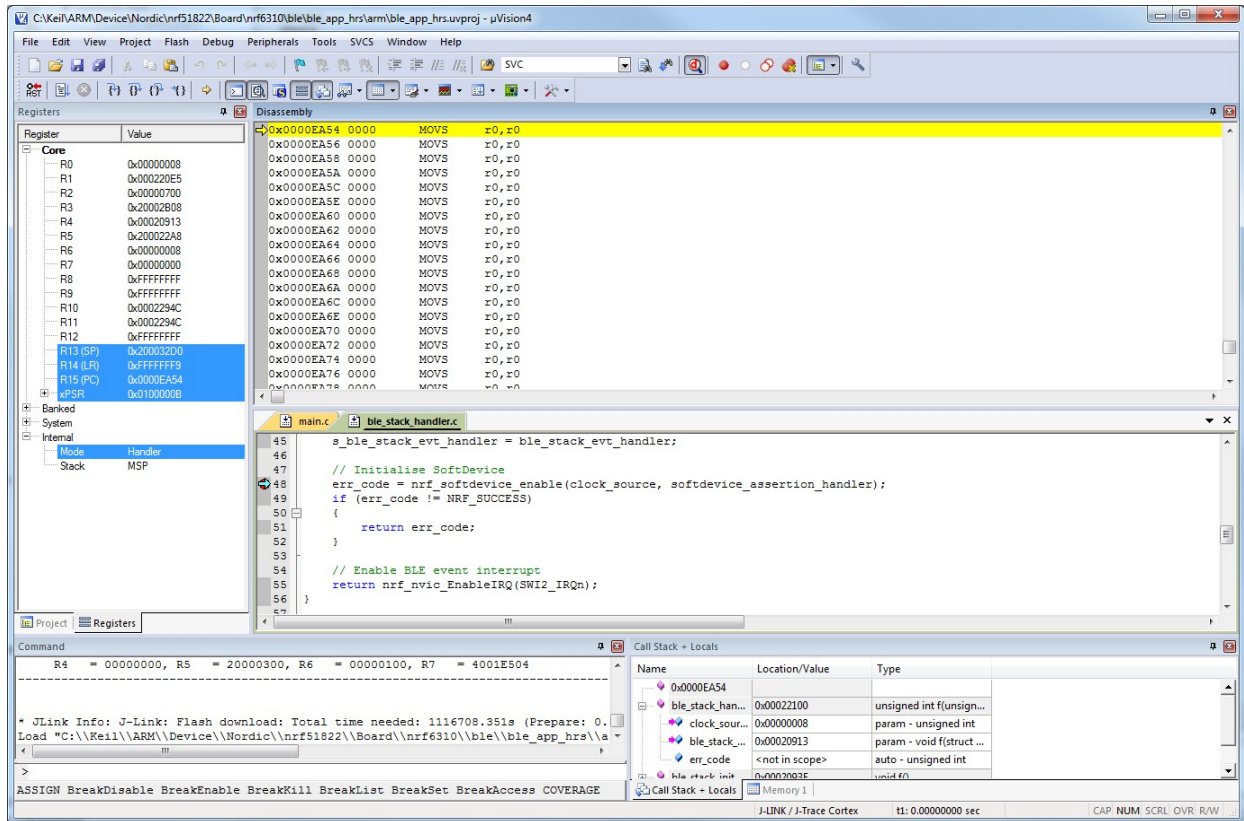
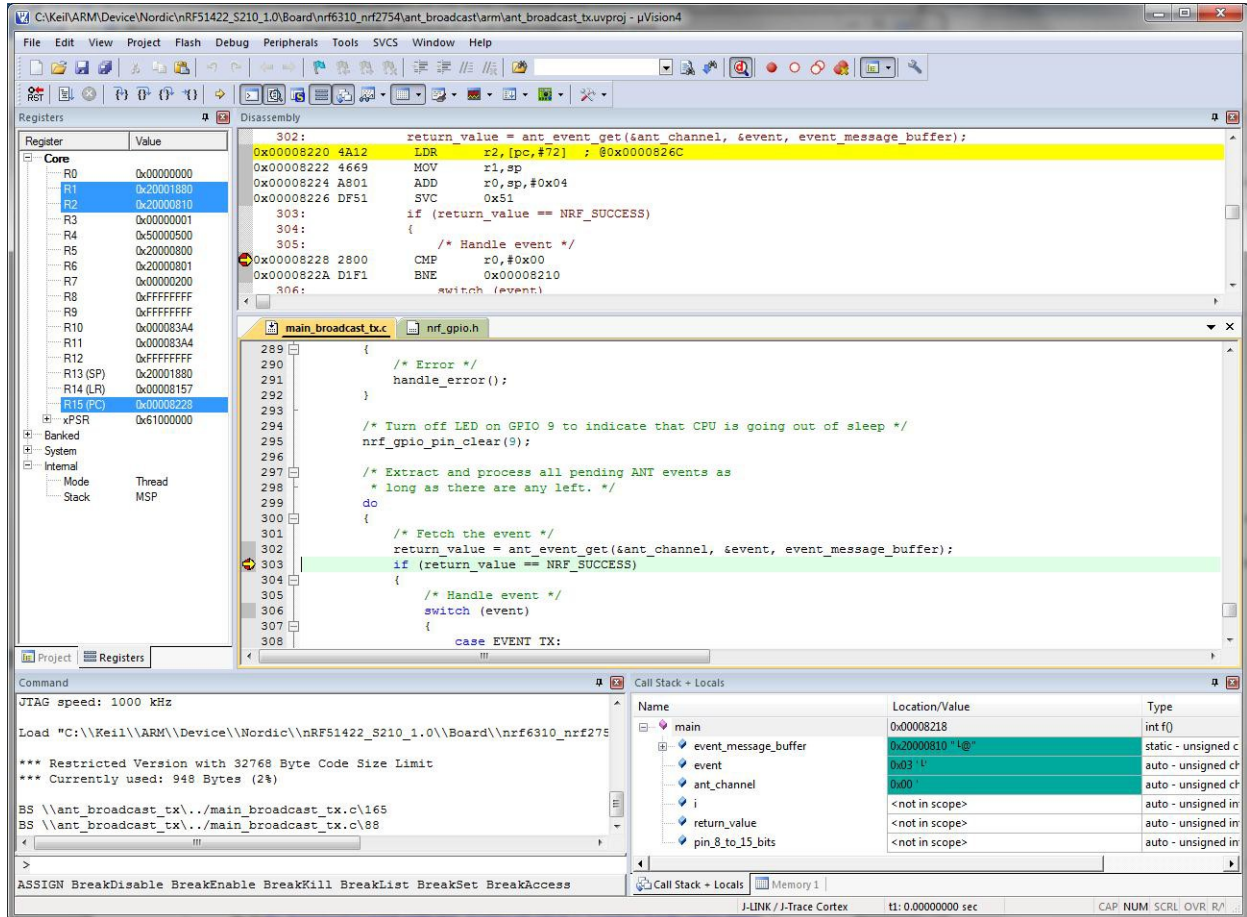


Figure 31 Debugger 信息 for a setup with a SoftDevice enabled in the protected area



The screenshot shows the Keil uVision4 IDE interface. The main window displays the disassembly of the `SVC` instruction at address `0x00008226`. The instruction is highlighted in yellow. The assembly code is as follows:

```

302:      return_value = ant_event_get(&ant_channel, &event, event_message_buffer);
0x00008220 4A12      LDR      r2,[pc,#72] ; @0x0000822C
0x00008222 4669      MOV     r1,sp
0x00008224 A801      ADD     r0,sp,#0x04
0x00008226 DF51      SVC     0x51
303:      if (return_value == NRF_SUCCESS)
304:      {
305:          /* Handle event */
0x00008228 2800      CMP     r0,#0x00
0x0000822A D1F1      BNE     0x00008210
306:          switch (event)
307:          {
308:              case EVENT_TX:
    
```

The C code in `main_broadcast_tx.c` at line 303 is highlighted in green, showing the condition `if (return_value == NRF_SUCCESS)`. The Call Stack + Locals window shows the following variables:

Name	Location/Value	Type
main	0x00008218	int f0
event_message_buffer	0x20000810 *l@	static - unsigned c
event	0x03 '1'	auto - unsigned ch
ant_channel	0x00	auto - unsigned ch
i	<not in scope>	auto - unsigned in
return_value	<not in scope>	auto - unsigned in
pin_8_to_15_bits	<not in scope>	auto - unsigned in

Figure 32 在 SVCcall 之后设置断点

8 Software Development Kit

nRF51 的软件开发工具包 (SDK) 可以帮助你开发使用如下协议的应用程序:

- *Bluetooth* low energy (使用 S110_nRF51822 SoftDevice)
- 专有的 2.4 GHz, 包括 Nordic's Gazell protocol
- Non-concurrent combinations of *Bluetooth* low energy and proprietary 2.4 GHz

8.1 安装 nRF51 SDK

nRF51 SDK 是一个 MSI 文件 (Windows 安装程序), 双击即可安装。安装 SDK 后你可以选择: Keil MDK Support, Master Control Panel, 等等。下面描述两个各个安装选项的含义:

- **Keil MDK support** – 安装 Keil μ Vision 例程, J-Link debugger 的下载规则, 以及 Nordic nRF51 系列设备的 database 文件。
- **Custom install** – 安装到指定位置。
- **nrfjprog** – 安装一个可以使用 SEGGER debuggers 的命令行 (command line) 的编程接口。

注意: Keil MDK 的支持选项只有在你已经安装了 Keil MDK 工具链以后才会出现。

9 Troubleshooting

nRF51822 USB Dongle 为什么无法对外编程？

请查看 9 针的单排插针上的跳线是否在 OUT 一侧。另外，请查看连线是否正确。

nRF51822 AK II 为什么 KEIL 无法下载程序？

首先检查系统是否正常供电，之后尝试用 nRFgo Studio 的 nRF51 Programming 验证一下是否 KEIL 工程设置问题。

在我的 32 位 Windows XP 电脑上，安装 nRF51 SDK 时出现 错误代码 2908 。

卸载 nRF51 SDK 后重新安装可能会引起 2908 错误。请使用任务管理器 (Ctrl+Shift+Esc) 来结束 nRF514/8 SDK Setup。此时 SDK 应该已经安装在电脑上了。

调试的时候 debugger 不动了。

如果烧写 SoftDevice 时选中了“Lock SoftDevice from Readback” (请看 page 25 的 section **section 6.1 “Programming and erasing flash using nRFgo Studio”**), 当跳到 SVC 指令时 debugger 就会停止。应该在 SVC 指令之后设置断点，然后运行应用程序到这个断点。或者，跳过任何的 SVC 指令。请看 page 40 上的 **section 7.1.2 “Debugging an application when a readback protected SoftDevice is present”**

debugger 在断点处不停止

SDK 内的有些 Keil 工程 选择了 **Optimization level 3 (-O3)** 和 **Optimize for time** 优化选项。此时调试这些应用，可能会有某些断点失效。

1. 按 **Alt+F7** 打开 Target options 对话框。
2. 选择 **C/C++**.
3. 在下拉列表中选择 **Optimization level 0**
4. 不选 **Optimize for time**.

调试过程中无法与软件同步。

当系统运行时使用 SEGGER debugger 设置或修改断点来暂定 CPU，可能会导致失去同步。应该避免在系统运行时设置断点。

我下载程序之后 debugger 无法连接到 nRF51 芯片

如果 nRF51 芯片复位之后刚刚进入 **SystemOff** 状态时，可能无法与 J-Link debugger 通讯。可以点击 nRFgo Studio 的 **Recover** 按钮来重新覆盖并查找芯片。

1. 开始 Recover 之前给 RF51 芯片重新上电

注意：Recover 操作将擦除芯片上的应用程序

为什么我的 MPU6050 连接不上？

请使用 0x69 这个地址读取 MPU6050 上的信息。

我的工程以前是正常的，但是当我打开别的使用 **SoftDevice** 的工程后及不行了。

请确保工程的 memory layout 和芯片相匹配。请看 page 29 上的 **section 6.2.1 “配置 memory layout”**

附录 A: 安装 SEGGER debugger 驱动并配置 KEIL 工程

本附录描述了如何在安装并使用 SEGGER J-Link Lite debugger，以及使用 Keil μ Vision 来编写 nRF51 系列芯片，基于 J-Link software version 4.52b or later.

前提条件

需要安装 Keil μ Vision (ARM-MDK)，然后查看一下 MDK 是否为 version 4.54，或更新版本。

注意：所有 nRF51 SDK 提供的工程都可以与 SEGGER debugger 配合工作。只需要按照下面 **"Download and install SEGGER drivers"** 的要求操作。

Download and install SEGGER drivers

1. 下载最新的 SEGGER J-Link 软件包，
<http://www.segger.com/jlink-software.html>.
2. 在安装过程中，请准确选择需要支持的 IDE，之后才会正确的更新相应的 SEGGER DLLs。
选择 **Keil MDK** 和其他你想要 SEGGER 支持的 IDE。

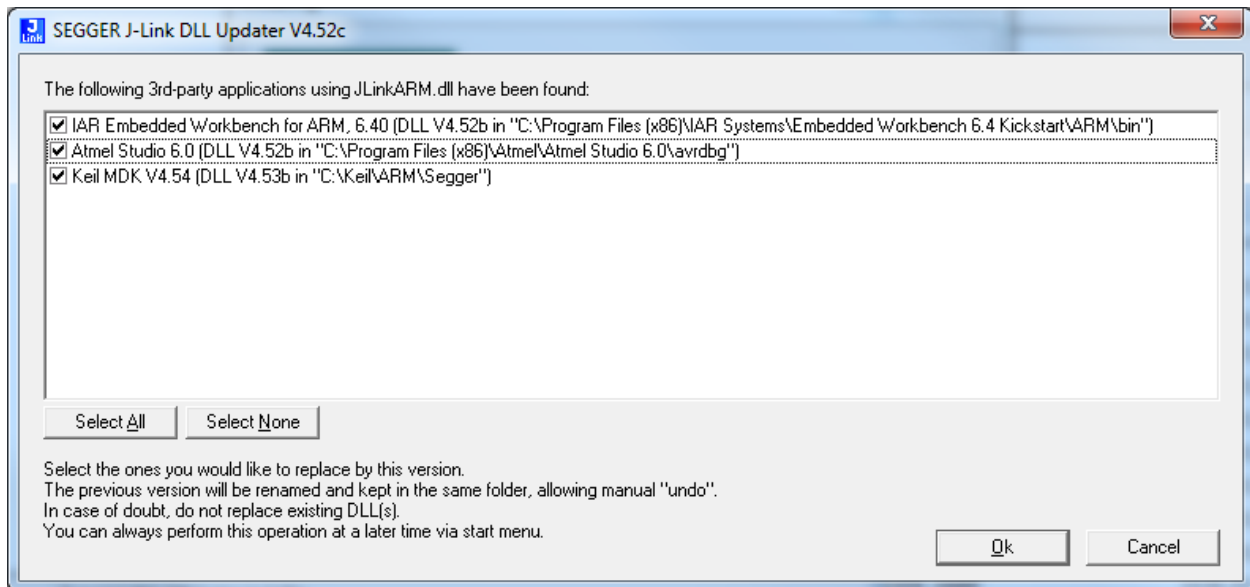


Figure 33 IDEs selected for updating to the latest SEGGER DLLs

- 到 http://www.segger.com/IDE_Integration_Keil.html#knownproblems for MDK v4.54. 下载 JL2CM3 并 copy 到 <keil>/ARM/Segger.
- 将 nRF51822 USB Dongle (PCA10000) 插入电脑的 USB 接口。如果驱动安装正确，D1 LED 将会闪烁，等待驱动安装完成 LED 将持续亮起。

初次使用 SEGGER debugger 的 KEIL 工程设置

建立一个 **JLinkSettings.ini** 文件，内容如 page 50 上的 **Figure 40** 所示。**JLinkSettings.ini** 要和 Keil μ Vision 工程 (uvproj) 文件保存到同一个目录下。

- 双击打开一个示例工程。
- 点击工具栏上的 **Target Options** 或者点击 **Project** 菜单再选择 **Options for Target**

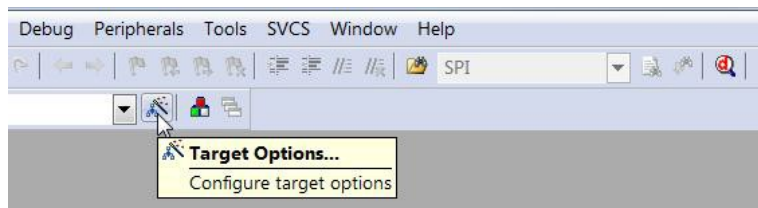


Figure 35 Keil Target configuration

- 在 **Debug** 标签下的 Use 列表中, 选择 **J-LINK / J-Trace Cortex** 选项, 如图 **Figure 36**.
- 点击 **Settings**, 如图 **Figure 36**. SEGGER Control Panel 和 Keil Target Driver Setup 两个界面将会打开。

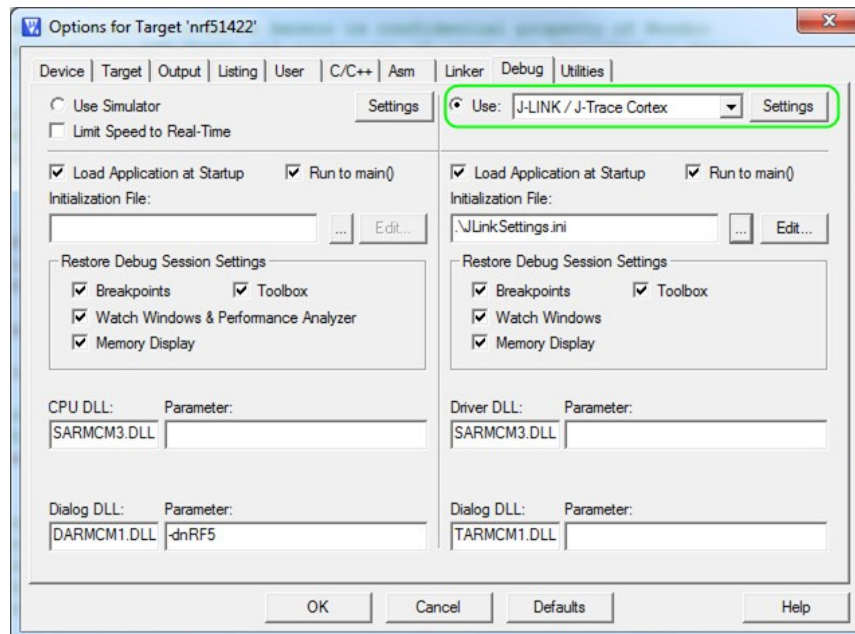


Figure 36 Selecting JLink debugger in Keil

注意: 如果 SEGGER J-Link Lite 固件需要升级, 将会弹出一个信息 “A new firmware version is available for the connected emulator”, 此时请点击 **OK**。

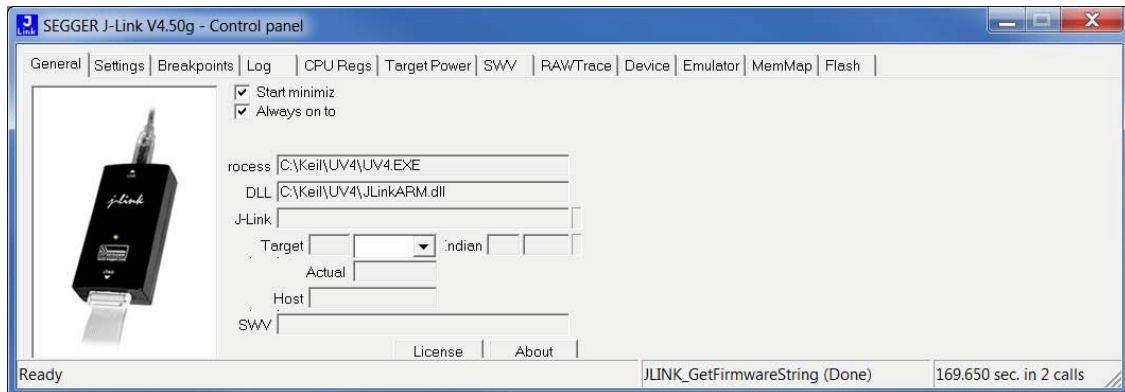


Figure 37 Segger control panel

5. 点击 **Debug** 标签. 将 Port 设置为 **SW**, Max Clock 为 **1 MHz**。请看 **Figure 38** 之后点击 **OK**。

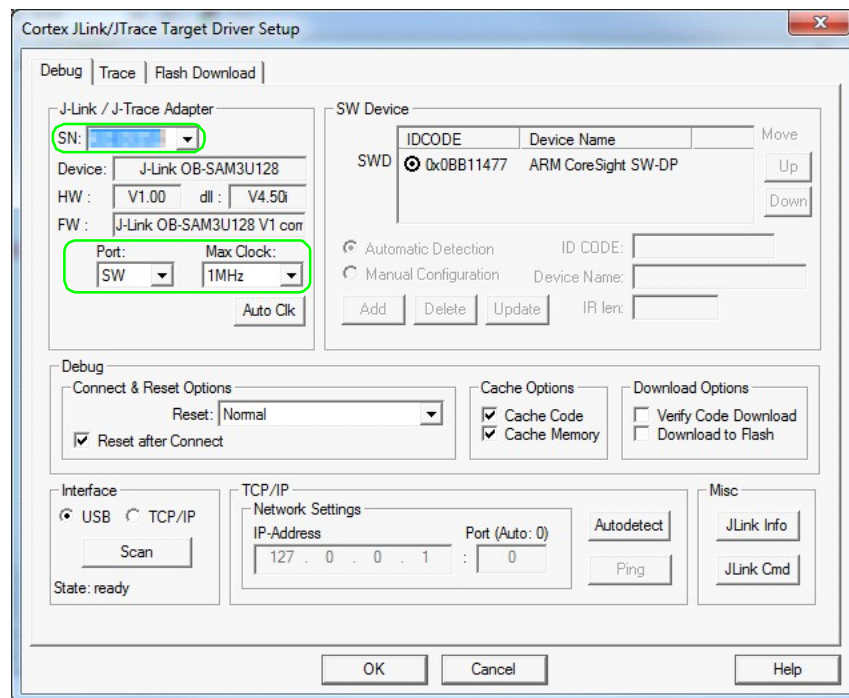


Figure 38 Debug settings

6. 在 Use target Driver for programming 选项下面选择 J-Link, 并提供恰当的 code memory 分布。

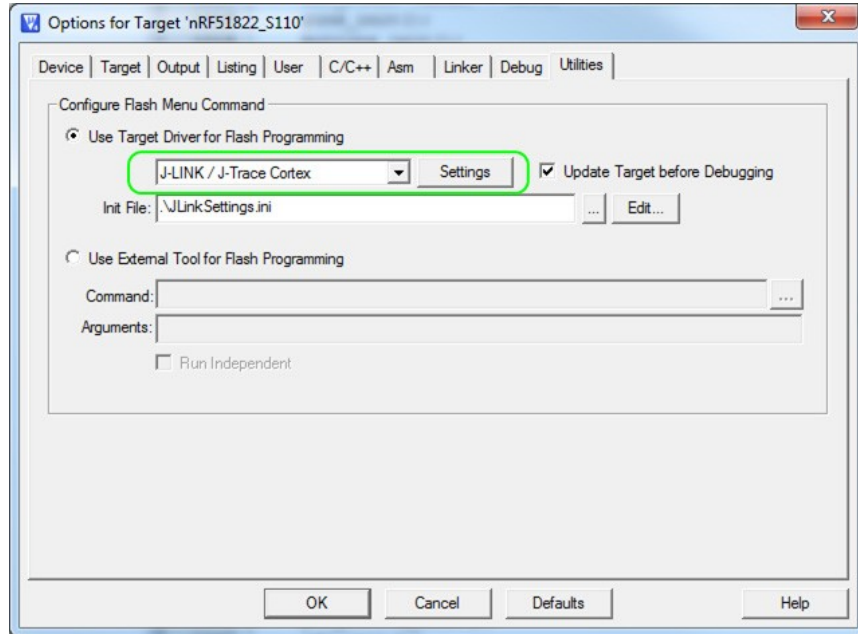


Figure 39 Flash settings

JLinkSettings.ini 文件

当调试一个带 SoftDevice 的软件时, 将 AllowSimulation = 1 改为 AllowSimulation = 0 , JLinkSettings.ini 文件就在工程文件夹内。

1. 进入 KEIL target options (**ALT+F7**) 并点击 ... 按钮来选择 JLinksettings.ini 文件。
2. 点击 **Edit** 并修改 AllowSimulation , 如上述修改。

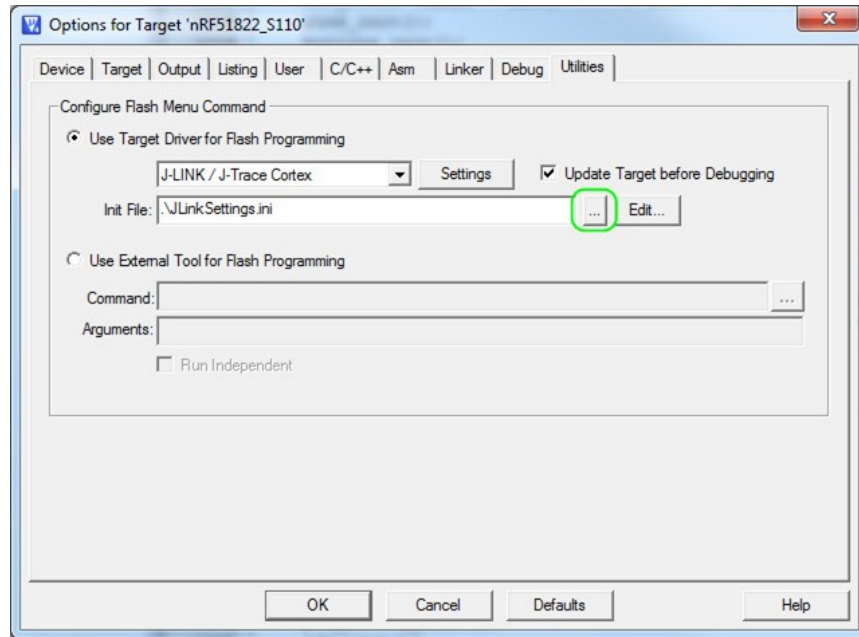


Figure 40 定位 JLinkSettings.ini

附录 A 文档及软件功能介绍

nRF51822 软件

- nRFgo Studio (烧写蓝牙协议栈的必备软件, 也可以下载程序)
- nRF51 Software Development Kit (SDK)
- 预编译好的 蓝牙协议栈 HEX 文件 (在 s110_nrf51822_xxxx.zip 压缩包内)
- 源代码 (安装 nrf51_sdk_v4_3_0_xxxxx.msi 后可见)
- Keil ARM 工程文件 (源码内)
- S110 nRF51822 SoftDevice (蓝牙协议栈)

nRF51822 文档

- *nRF51822 上手指南*
- *PROD BRIEF nRF51822 (nRF51822 介绍)*
- *nRF51822 PS (只针对 nRF51822 的特性说明)*
- *nRF51 Series Reference Manual (nRF51 系列芯片的参考手册)*
- *S110 nRF51822 SoftDevice Specification (蓝牙协议栈说明)*
- *Wearable Devices Primer (nRF51822 穿戴设备开发实务)*
- *nRF51822_Evaluation_Kit_User_Guide v1.1.pdf (官方 EK 开发板使用说明书)*