

WS8100

快速入门指南

修订记录

修订版本	描述	作者
V1.0	初稿完成	zhlj
V1.1	修复几处不严谨的说明	zhlj

目录

1 keil环境搭建	5
1.1 温馨提示	5
1.2 安装keil	5
1.3 添加芯片flash烧写算法文件	5
2 工程配置	5
2.1 打开Demo工程	5
2.2 Keil配置	6
2.2.1 Device配置	6
2.2.2 Target配置	6
2.2.3 C/C++配置	7
2.2.4 Linker配置	8
2.2.5 仿真器Debug配置	8
2.2.6 Flash Dowload配置	9
2.2.7 Utilities配置	10
3 编译和下载	10
3.1 编译	10
3.2 下载方法一（Jlink仿真器）	11
3.3 下载方法二（BLE42_Download_Tool.exe）	11
3.4 debug仿真	14

图索引

1 keil环境搭建

1.1 温馨提示

- 安装路径名中不能带中文，必须是英文路径，如：D:/Keil。
- 安装目录不能与其它版本 keil 冲突，两者目录必须分开。
- 建议使用 MDK4.7，下载地址见开发包“TOOLS/Keil/Keil 工具说明.txt”

1.2 安装keil

1. 双击 keil 安装包，在弹出的对话框中点击“next”开始安装。
2. 勾选同意软件使用条款复选框，点击“next”按钮。
3. 选择安装路径，路径名不要带中文，点击“next”按钮。
4. 填写用户信息，全部填空格即可或自行填写，点击“next”按钮。
5. 安装完成后点击“finish”。
6. 使用电脑管理员的方式打开安装好的 keil，点击“File”，选择“License Management”，进入安装密钥。
7. 打开解压文件中的“keygen.exe”注册机，将注册界面的 CID 复制到注册机中，然后点击“Target”修改为“ARM”，点击“Generate”，生成注册码。
8. 将注册码复制到注册界面，点击“AddLIC”，成功后安装结束。

1.3 添加芯片flash烧写算法文件

将WS810x_FLASH_V1.1.FLM（文件位于开发包TOOLS/Keil目录下）文件放到keil安装目录，如：“D:\Keil\ARM\Flash”目录下。

2 工程配置

2.1 打开Demo工程

建议客户根据方案选择SDK中功能相近的example目录下的Keil工程直接修改。

Demo工程提示：

- 选择功能相近的示例代码

	at_demo	2019/8/20 9:21	文件夹
	findme_demo	2019/8/20 9:21	文件夹
	remote_control_demo	2019/8/20 19:15	文件夹
	transpond_demo	2019/8/20 19:15	文件夹

图 2.1-1 示例代码

- 打开Project.uvproj工程

	src	2019/8/20 19:15	文件夹	
	ble42_extflash.ini	2019/8/20 19:15	配置设置	1 KB
	ble42_extFlash.sct	2019/8/20 19:15	Windows Script ...	1 KB
	JLinkLog.txt	2019/8/20 19:15	文本文档	256 KB
	JLinkSettings.ini	2019/8/20 19:15	配置设置	1 KB
	Project.uvopt	2019/8/20 19:15	UVOPT 文件	20 KB
	Project.uvproj	2019/8/20 19:15	Microvision4 Project	21 KB
	Project_BLE42_SDK.dep	2019/8/20 19:15	DEP 文件	46 KB

图 2.1-2 打开工程

2.2 Keil配置

在keil工程界面中点击  选项卡进入工程Keil配置，此配置不可忽略，在任意配置完成后，点击"确定"键退出配置选项。

2.2.1 Device配置

根据图 2.2-1 配置Device，所示步骤如下：

1. 工程 Device 选择"ARM 下"的"Cortex-M3"。

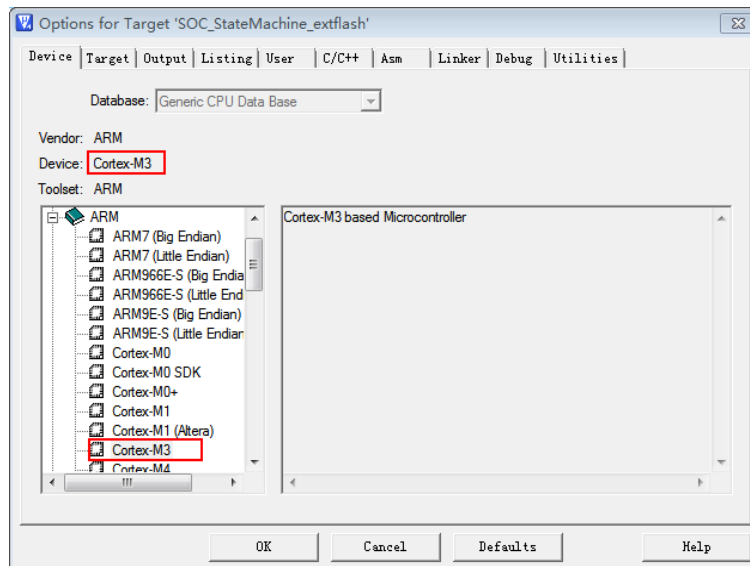


图 2.2-1 Device选择

2.2.2 Target配置

根据图 2.2-2配置Target，所示步骤如下：

1. "Code Generation"中勾选"UsbMicroLIB"选项。
2. 勾选"IROM1", "Start"地址为 0x1016000, "Size"大小为 0x32000, 需参考分区文件如图 3.3-1 中的"AppAddr"进行修改。
3. 勾选"IRAM1"和"IRAM2", "Start"地址和"Size"大小按图 2.2-2 中数据填写, 不可更改。

注意: 如果在系统中启动了 ble42_extFlash.sctw 文件作为系统的 ROM 和 RAM 的分区, 此处配置可忽略。

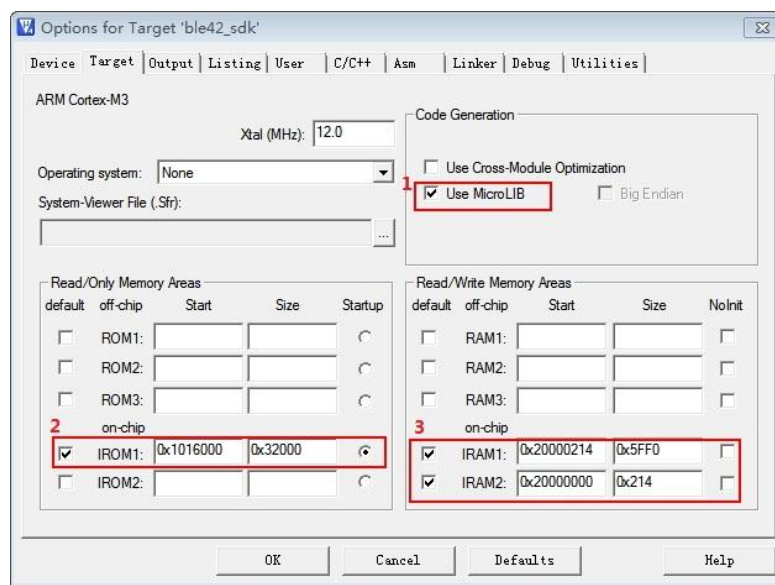


图 2.2-2 Target配置

2.2.3 C/C++配置

根据图2.2-3 配置C/C++, 所示步骤如下:

1. "Optimization"优化等级建议选择 Level 2 (-O2)。
2. "include Paths"可以添加工程中相应的头文件路径, 如有新的头文件在工程中创建和添加, 编译之前需要在此处添加头文件路径, 点击"..."进行操作。
3. Misc 控制"Misc Control" 建议填写-c99。

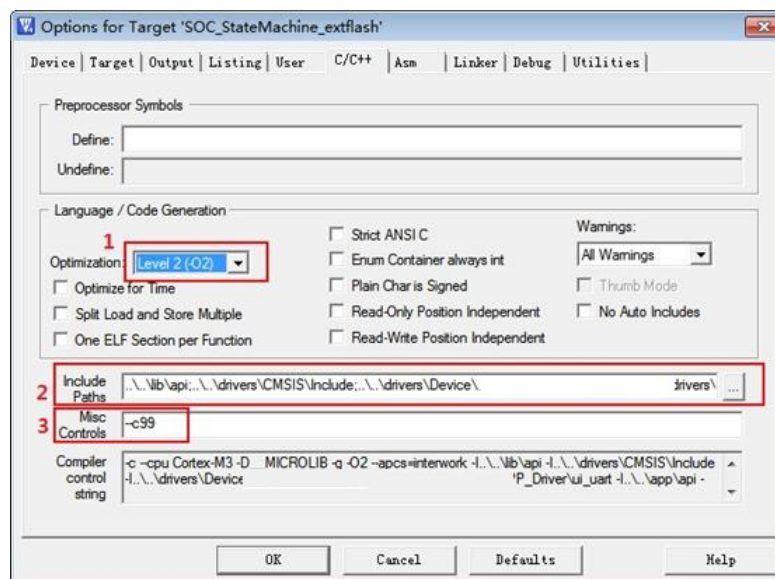


图 2.2-3 C/C++配置

2.2.4 Linker配置

根据图2.2-4 配置Linker，所示步骤如下：

1. 分散文件"Scatter File"，这里添加工程下的 Project_extFlash.sct 文件。点击"..."进行加载文件；点击"Edit"查看和编辑对于文件。
2. Linker 中，不要勾选 Use Memory Layout from Target Dialog。

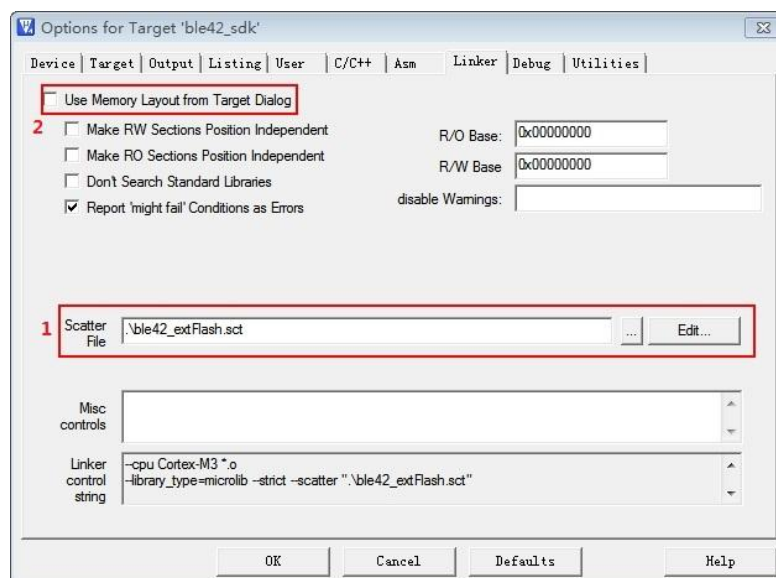


图 2.2-4 linker配置

2.2.5 仿真器Debug配置

根据图2.2-6 配置Debug，所示步骤如下：

1. 配置 Debug 选项，"Use"标准型号选择"J-LINK/J-Trace Cortex"。

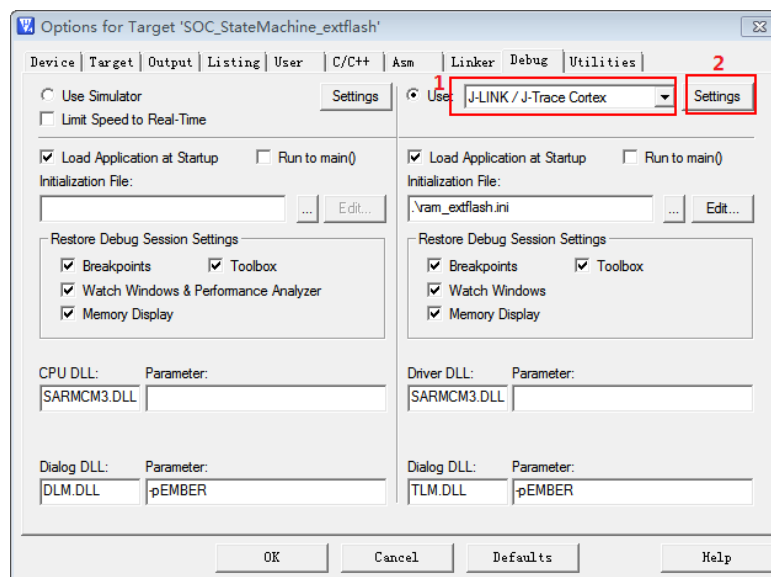


图 2.2-5 仿真器选择

2. 配置 Dubug Setting 选项, 在图 2.2-5 的基础上点击"Setting"进入 Setting 配置窗口, 在窗口中找到"ort:"选择 SW 接口, 如果链路连接没问题且板卡已上电, 则仿真器会识别调试板的芯片, 并在 "SWDIO"窗口显示出, 如图 2.2-6。

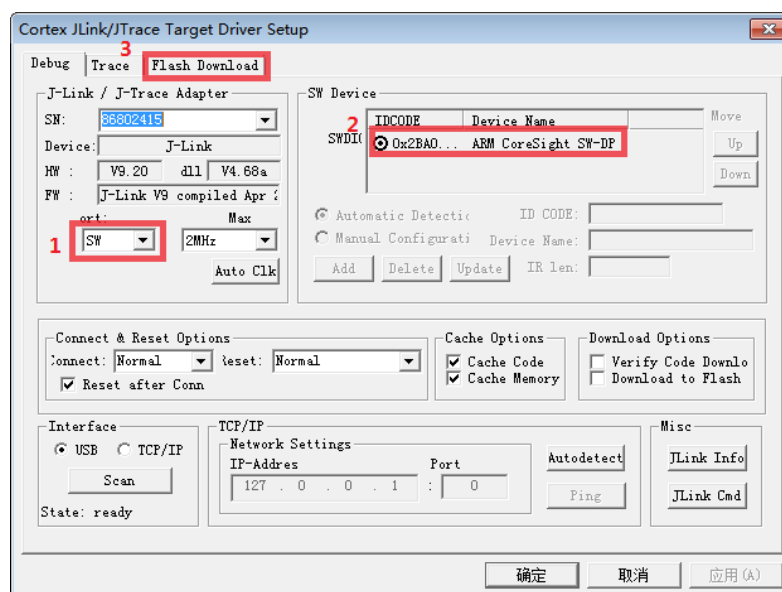


图 2.2-6Dubug Setting

2.2.6 Flash Download配置

确认先将"WS810x_FLASH_V1.1.FLM"文件放到"D:\Keil\ARM\Flash"目录下, 然后如图2.2-6 点击 "Flash Download"选项, 进入后根据图2.2-7配置Flash Download, 所示步骤如下:

1. 下载功能"Download Function", **重点注意勾选"Erase Sector"**, 防止下载程序时会将分区文件擦除。(注意: 一定不要选择 "Erase Full Chip".)
2. RAM 算法"RAM for Algorithm"中的":start:"填写 0x20000000, "size:"填写 0x2800。

3. 点击"Add"添加编程算法文件，选择章节 1.3 步骤添加的文件"WS810x_FLASH_V1.1.FLM"，配置成功后点击"确定"。

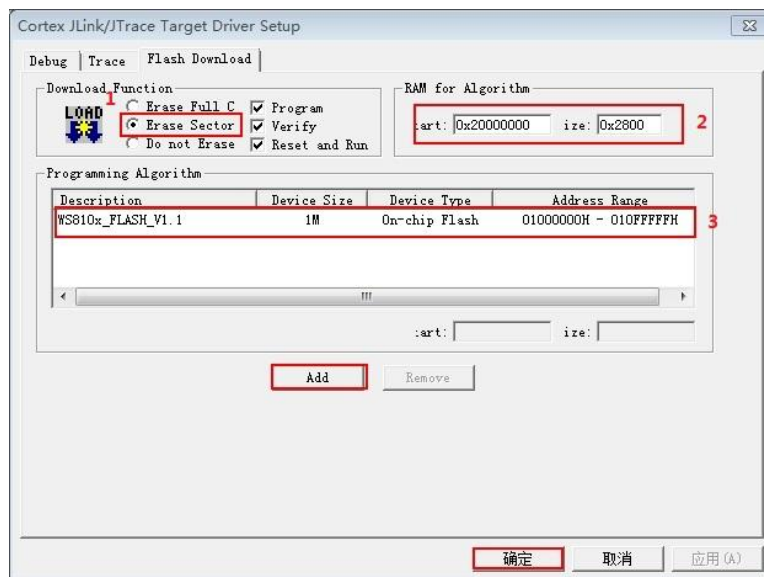


图 2.2-7 Flash配置

2.2.7 Utilities配置

防止下载仿真时会出现提示仿真器错误，需要此处和上图2.2-5的的仿真器类型做同步配置。

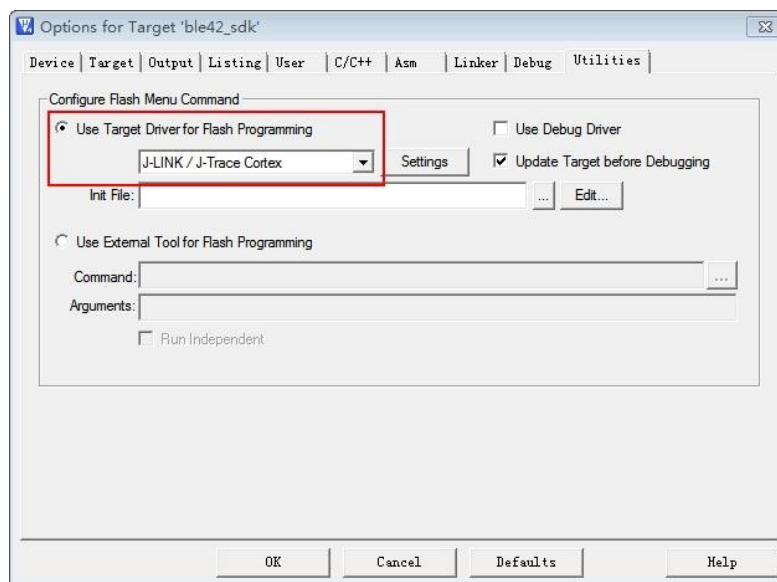



图 2.2-3 Utilities配置

3 编译和下载

3.1 编译

创建好的工程后，在左边工程组中右击选择创建文件.c和.h文件等，在文件中编写相应的工程代码，

完成后要对工程进行编译，点击" "rebuild图标，进行工程编译，如下方"Build Output"窗口中显示"0 errors(s)"则表示编译通过，生产了hex文件。

3.2 下载方法一（Jlink仿真器）

编译成功后，接下来就可以把编译好的工程下载到芯片中运行，连接好Jlink仿真器后，直接点击keil中的"LOAD"图标即可，图 3.2-1

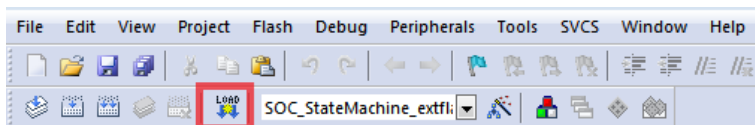


图 3.2-1 LOAD按钮

程序下好后，"Build Output"窗格中如果显示"Application running...",则表示程序下载成功，见图 3.2-2。

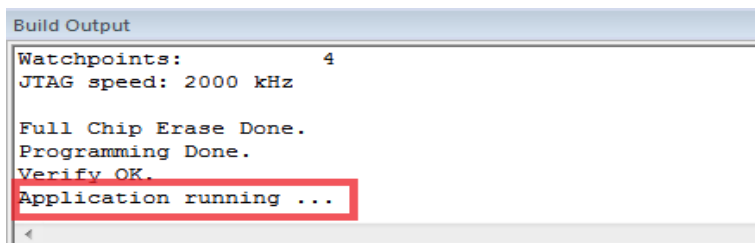


图 3.2-2 下载成功提示

3.3 下载方法二（BLE42_Download_Tool.exe）

boot下载是对工程编译后的hex文件进行操作，不需要使用Jlink接口下载，可以直接通过串口对芯片进行程序下载，为有效的在线升级功能。

提示：本文操作步骤只作为下载使用，详细操作见开发包"DOC/BOOT使用说明"。

所示操作如下：

1. 首先在编译工程之前，确认 2.2.2 章节"Target 配置"操作，如图 3.3-2 选中"IROM1"，"Start"和"Size"按照默认分区如图 3.3-1 中偏移地址进行填写后再编译工程，生成 boot 下载所需的 hex 文件，文件会自动生成在工程目录下。

```

/*BLE-PART-CONFIG*/
D4 2D 91 D3 /*PartInfoValid*/
00 60 01 00 /*appAddr*/
00 20 03 00 /*appSize*/
00 20 00 00 /*StackAddr*/
00 40 01 00 /*StackSize*/
00 10 00 00 /*SblAddr*/
00 10 00 00 /*SblSize*/
00 F0 07 00 /*UserData2Addr*/
00 10 00 00 /*UserData2Size*/
00 A0 07 00 /*UserData1Addr*/
00 50 00 00 /*UserData1Size*/
00 80 04 00 /*FreeAddr*/
00 20 03 00 /*FreeSize*/
  
```

图 3.3-1默认分区

2. 找到 boot 下载文件目录，双击"BLE42_Download_Tool.exe"打开引用程序，此操作说明如下：

1) 串口连接，如图 3.3-2：

- 端口选择 app 自动读取有效端口。
- 波特率为默认 115200。
- 点击“打开”连接串口。

2) 握手连接，如图 3.3-3：

- 用户密码：默认 0xFFFFFFFF，如密码变动，需根据实际改动。
- 串口成功连接后，区域中的"连接"按钮会变为有效按键，点击此按钮。
- 按照提示，对底板进行"复位"操作。
- 日志提示连接成功。

3) 信息获取，如图 3.3-4：

- 读芯片信息。
- 读出厂信息。
- 读分区信息。
- 读控制信息。
- 重启芯片。

4) "hex 文件"下载，如图 3.3-4：

- boot 与芯片连接成功后，选择适当的文件"下载提速"，最快支持 1Mbps。
- "应用固件"添加要下载的 hex 文件。
- 添加完成后点击区域中的"download"按钮。
- 当进度条跑满后且日志有成功提示，表示下载完成。



图 3.3-2 boot 打开界面



图 3.3-3 boot 连接

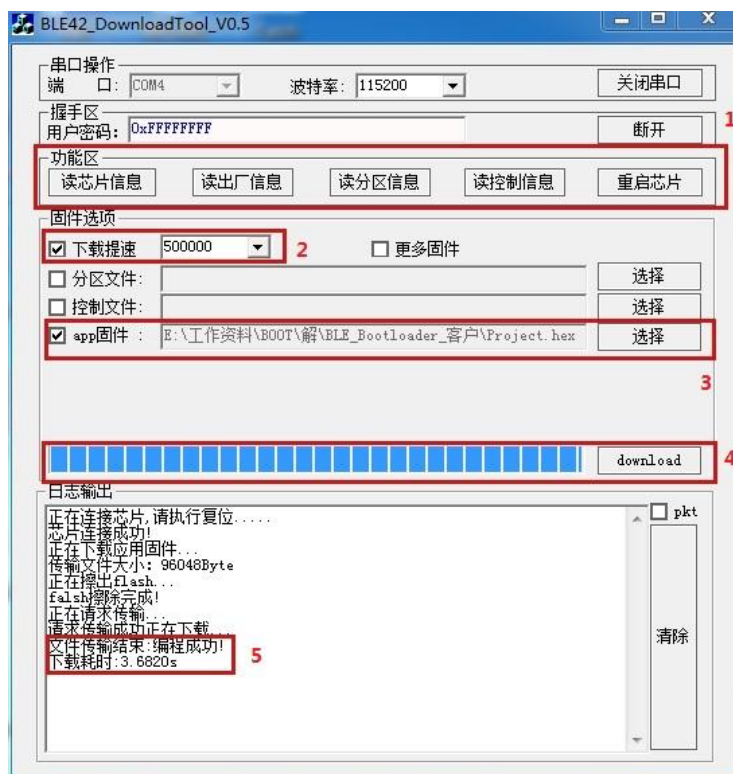



图 3.3-4 boot 成功下载

3.4 debug仿真

用J-LINK仿真器连接调试板进行程序debug仿真调试。点击，开始仿真（如果开发板的代码没被更新过，则会先更新代码，再仿真。