



成都翌创微电子有限公司
Chengdu ET Microelectronics Co., Ltd.

ET6001

32 位 Arm[®] Cortex[®]-M7 微控制器 (MCU/DSP)

专注实时控制，铸造安全可靠中国芯

SDK 开发入门指南

文档版本 1.0.2

发布日期 2024-07-15

版权所有 © 成都翌创微电子有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



和其他翌创微商标均为成都翌创微电子有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受成都翌创微电子有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，成都翌创微电子有限公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

成都翌创微电子有限公司

地址：四川省成都市高新区高朋大道 3 号东方希望大厦 9 楼 901 号

电话：028-83350520

网址：<https://www.etmcu.com>

邮箱：support@etmcu.com

目录

目录.....	2
1 开发环境与工程配置 (Keil).....	3
1.1 Keil 集成开发环境安装.....	3
1.1.1 MDK-ARM 软件包安装.....	3
1.1.2 调试器驱动安装.....	3
1.1.3 USB 转串口驱动安装.....	4
1.2 DFP 软件包安装.....	4
1.3 工程配置.....	5
1.3.1 新建工程.....	5
1.3.2 仿真器连接.....	6
1.4 系统启动.....	8
1.4.1 OTA 模式.....	8
1.4.2 Boot 模式.....	8
2 程序下载与调试.....	9
2.1 使用 MDK 调试和下载程序.....	9
2.1.1 准备工作.....	9
2.1.2 Options 选项配置.....	9
2.1.3 程序下载.....	13
2.1.4 程序调试.....	14
2.2 程序脱机下载.....	14
2.2.1 准备工作.....	14
2.2.2 编译下载.....	15
修订记录.....	18

1 开发环境与工程配置 (Keil)

1.1 Keil 集成开发环境安装

MDK-ARM 是一款专为基于 ARM 架构微控制器设计的集成开发环境 (IDE)，由德国 Keil 公司开发，目前该公司已被 ARM 公司收购。MDK-ARM 是一个全面的软件开发解决方案，主要针对 ARM7、ARM9、Cortex-M、Cortex-R4 等系列微处理器及相应的微控制器产品。

ET6001 目前的 SDK 软件包也是基于 MDK-ARM 平台开发和提供，在搭建基于 ET6001 SDK 软件包的工程前，需要先安装 KEIL 软件和必要的驱动。

- 安装 MDK-ARM 软件包
- 安装调试器驱动
- 安装 USB 转串口驱动
- 安装 ET6001 SDK Keil DFP package

1.1.1 MDK-ARM 软件包安装

1. 下载安装包 <https://www.keil.com/demo/eval/arm.htm> (版本 v5.23 以上)
2. 按照指示安装软件包(建议最好安装在 C 盘，不要有中文路径)
3. 安装完成后提示是否选择安装 ULINK 驱动，选择是

1.1.2 调试器驱动安装

可用于 ARM Cortex-M 架构芯片的调试器种类繁多，大致有二种：

1. 兼容 CMSIS-DAP 的仿真器 (安装 MDK-ARM 后驱动已安装)
2. J-Link

在 SEGGER 官 网 (<https://www.segger.com/downloads/jlink/#J-LinkSoftwareAndDocumentationPack>) 下载，下载完成后直接安装，安装完成驱动也相应的安装完成。如安装成功，接入 J-link 调试器，打开设备管理器会显示如下：



1.1.3 USB 转串口驱动安装

Win10 以上系统，板载 USB 转串口系统自带驱动，无需额外安装。

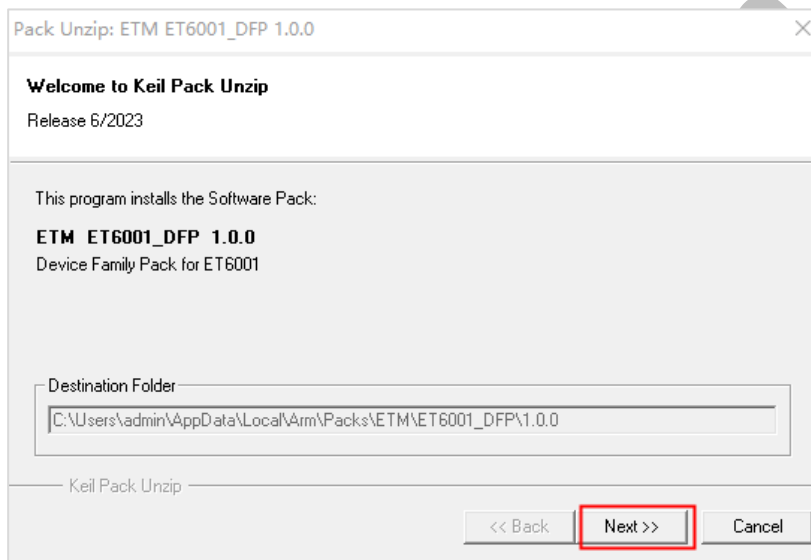
1.2 DFP 软件包安装

ETMCU 的 ET6001 软件包名称为 **ETMCU.ET600x_DFP.1.x.x.pack**（可在 **SDK 软件包 Package 目录下找到或通过官网下载**）。软件包的使用环境为 MDK 开发环境。需要在 MDK 环境下离线安装该 ETM 软件包。

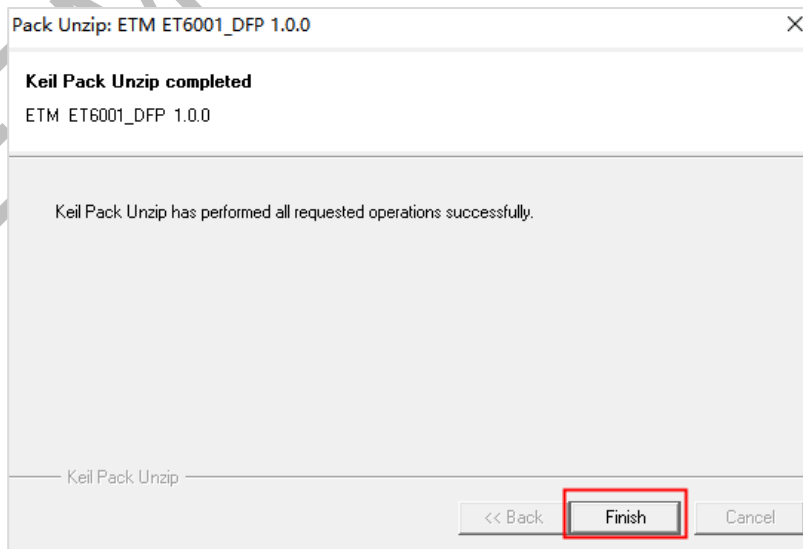
双击文件 **ETMCU.ET600x_DFP.1.x.x.pack**，一直点击 Next 即可。

安装步骤如下：

1. 双击软件包 **ETMCU.ET600x_DFP.1.0.0.pack**，点击 **Next**。



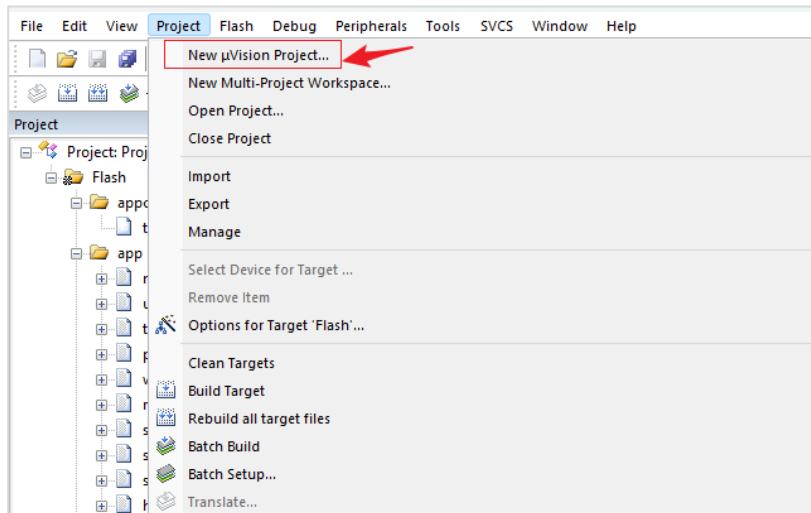
2. 等待安装完成，点击 **Finish**。



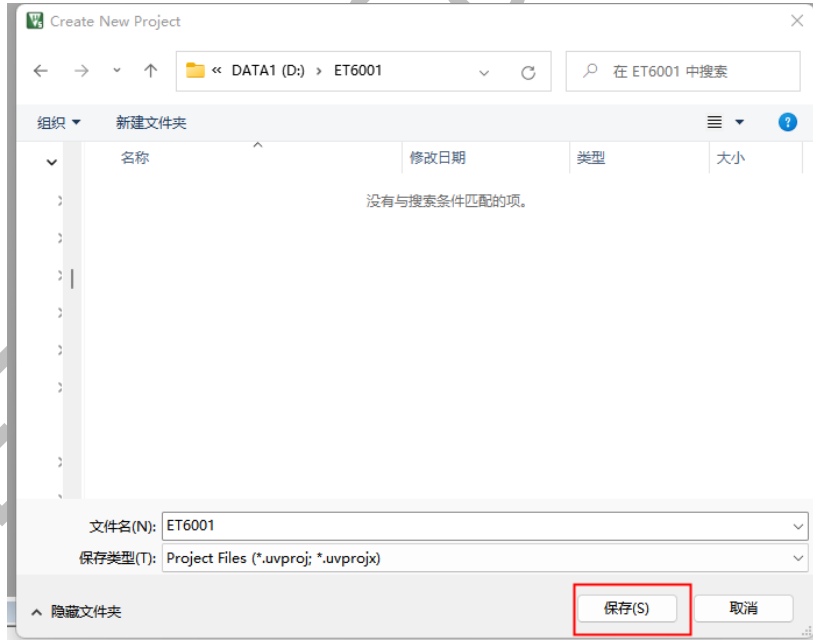
1.3 工程配置

1.3.1 新建工程

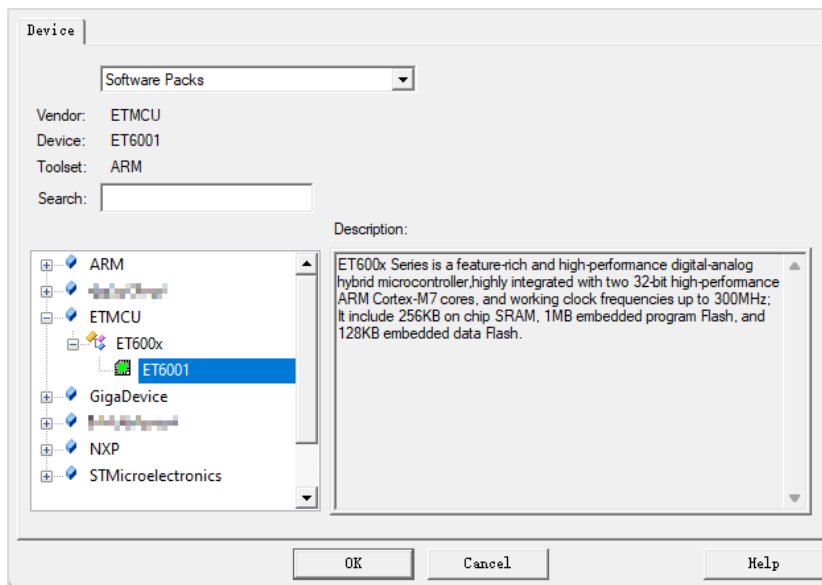
1. 打开 MDK 软件。
2. 选择 **Project > New uVision Project**。



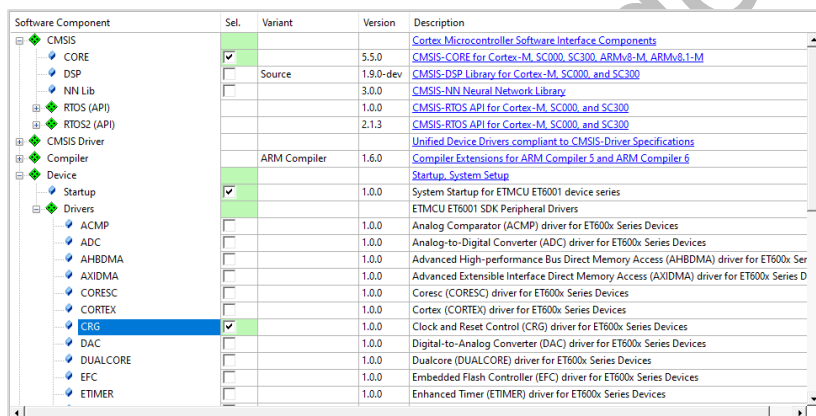
3. 选择创建工程的文件夹，并为工程文件命名；然后点击**保存**。



4. 创建工程项目后，自动弹出提示框，选择芯片 **ET6001**。



5. 根据需求，选择相关配置文件，其中 **CORE**, **Startup**, **CRG** 是必选的。



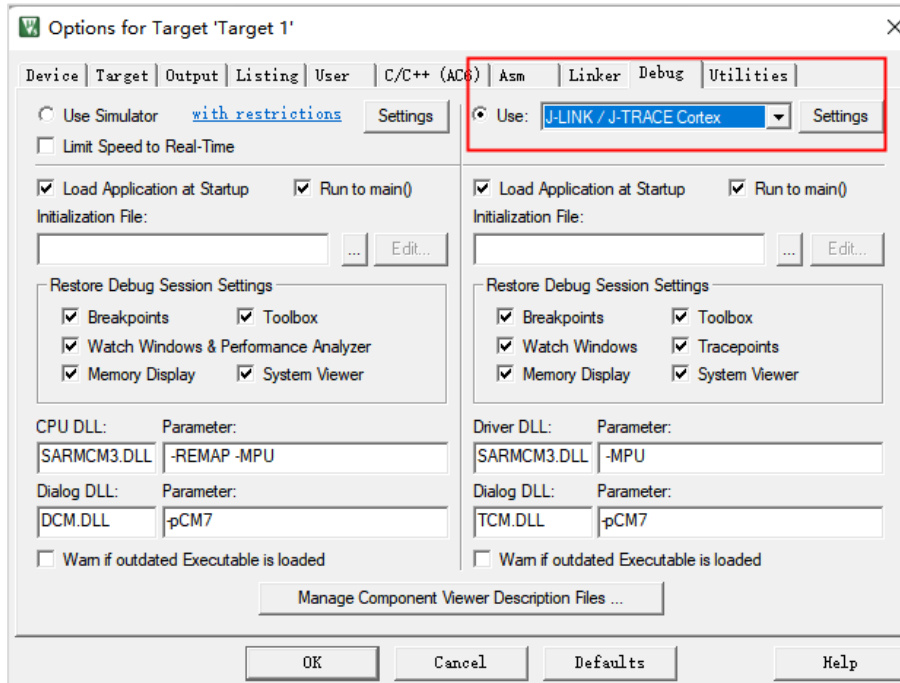
1.3.2 仿真器连接

在连接 J-Link 仿真器前，需要先修改 SEGGER 安装目录下的文件 **JLinkDevices.xml**。
如安装目录：**C:\Program Files (x86)\SEGGER\JLink**。在文件中添加如下标签：

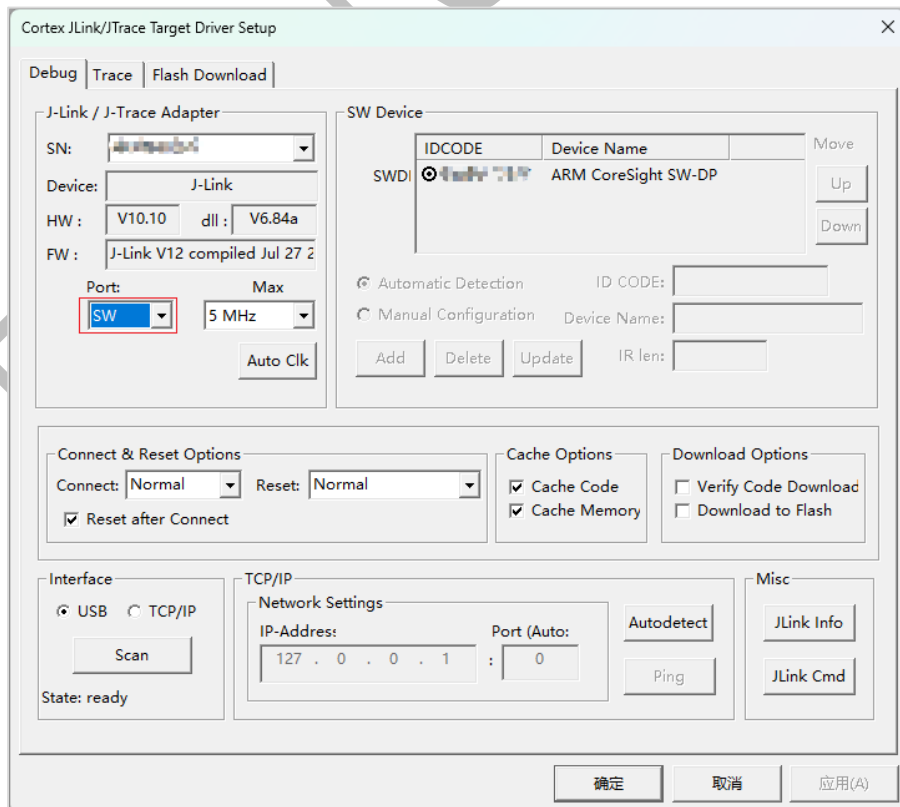
```
<!-- -->
<!-- ETMCU Devices (Cortex-7 devices)-->
<!-- -->
<Device>
  <ChipInfo Vendor="ETMCU" Name="ET6001" Core="JLINK_CORE_CORTEX_M7"
WorkRAMAddr="0x00000000" WorkRAMSize="0x10000" />
  <FlashBankInfo Name="Internal Flash" BaseAddr="0x08000000" MaxSize="0x80000"
Loader="Devices/ETMCU/ET6001.FLM" LoaderType="FLASH_ALGO_TYPE_OPEN"
AlwaysPresent="1"/>
</Device>
```

在 J-Link 的安装目录 **C:\Program Files (x86)\SEGGER\JLink\Device** 下新建 ETMCU 目录，并将 Flash 的下载算法文件 ET6001.FLM (可在 SDK 软件包 Flash 目录下找到或通过官网下载) 拷贝一份到该目录。

添加上述标签后，即可在 MDK 中通过 J-Link 调试芯片了。



注意，必须要选择 SW 接口。



1.4 系统启动

1.4.1 OTA 模式

在芯片的 NVR User Sector 保存有和固件升级及 APP 启动相关的 3 个选项字，他们分别是：CRC_EN, OTA_MODE, BANKSWAP。出厂时，三个字段默认全为 0。

CRC_EN 用于在启动时，判断是否需要校验 App 固件的 CRC32 值。

OTA_MODE 用于控制双 Bank Flash 的使用：

- 当 OTA_MODE=0 时，PFLASH0 和 PFLASH1 组合为 2x512 KB 的 Flash。
- 当 OTA_MODE=1 时，PLASH0 和 PFLASH1 互为备份。

BANKSWAP 用于对双 Bank 的映射物理地址互切：

- 当 OTA_MODE = 1 且 BANKSWAP = 1 时，PFLASH0 和 PFLASH1 之间的地址空间进行互换。

1.4.2 Boot 模式

在 ET6001 内部固化了一个引导加载程序，它会在每次上电时执行，通过 Bootmode 管脚电平的不同组合，实现不同引导模式的切换。

BOOT_MODE1 Pin	BOOT_MODE0 Pin	启动模式
0	0	Flash 启动
0	1	ROM 自举启动
1	0	SRAM 启动
1	1	预留

在系统复位后，Bootmode 配置引脚的值会被锁定。用户根据需要配置引脚来选择启动模式，在引脚锁定前，需要确保引脚处于稳定状态。

Flash 启动模式下：

- 当 CRC_EN=0 时，直接从 0x08000400 地址启动。
- 当 CRC_EN=1，OTA_MODE=0 时，如果 APP 镜像 CRC 校验成功，直接从 0x08000400 地址启动。
- 当 CRC_EN=1，OTA_MODE=1 时，先检测 0x08000000 地址起始的 APP 镜像 CRC 校验，若成功直接从 0x08000400 地址启动，否则继续检测 0x08080000 地址起始的 APP 镜像 CRC 校验，若成功，PFLASH0 和 PFLASH1 地址空间内容立即互换，并从 0x08000400 地址启动。

ROM 自举启动主要用于固件升级，进入到 ROM 自举模式后，自举程序初始化 UART0 和 I2C0 接口，等待上位机软件与其握手。握手成功后，可通过上位机软件对芯片进行 Flash 擦除及读写。

2 程序下载与调试

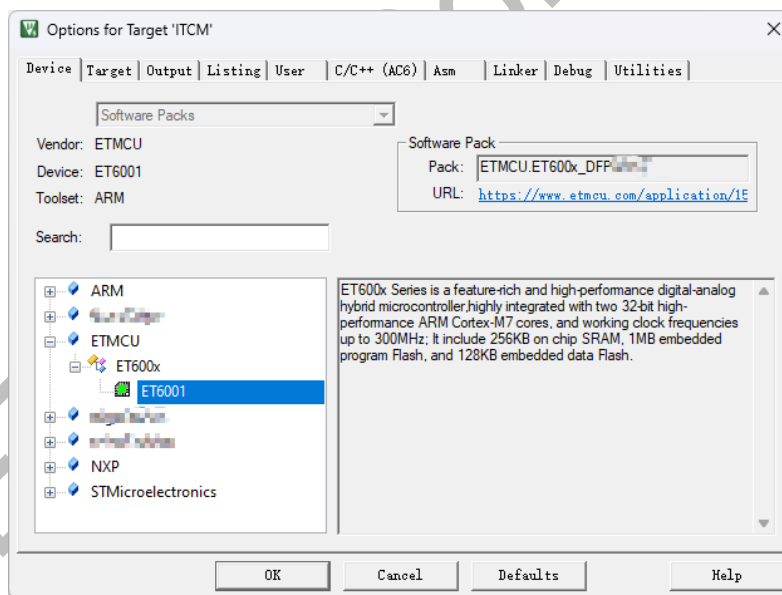
2.1 使用 MDK 调试和下载程序

2.1.1 准备工作

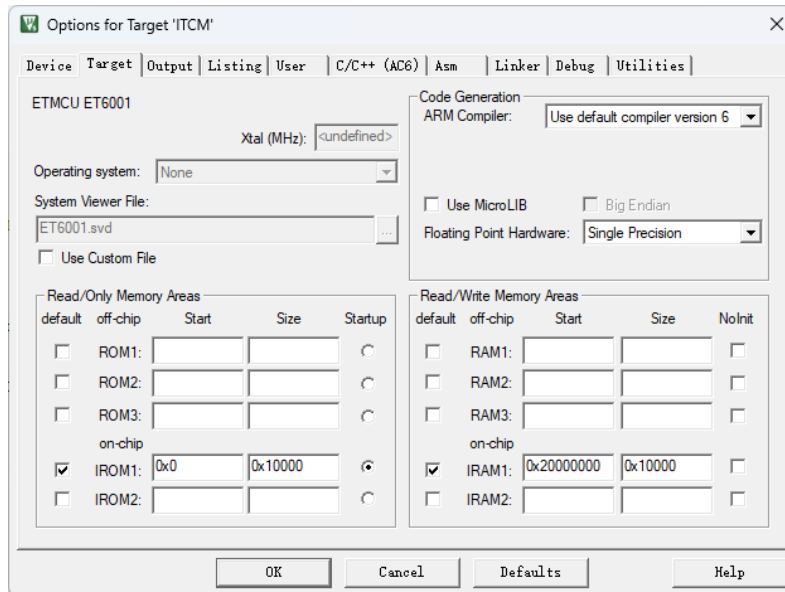
1. 连接调试器 (如 J-Link) 与芯片 SWJ 接口的管脚 CPU0_SWJ_SWCLKTCK 和 CPU0_SWJ_TMS 及 GND (如果使用 CMSIS-DAP 调试器, 需要额外连接 VREF)。
2. 已安装 MDK 软件 V5.3x 以上版本。
3. 已安装 **ETMCU.ET600x_DFP.1.0.1.pack** 以上版本 DFP package。
4. 安装 J-Link 版本为 V6.8x。

2.1.2 Options 选项配置

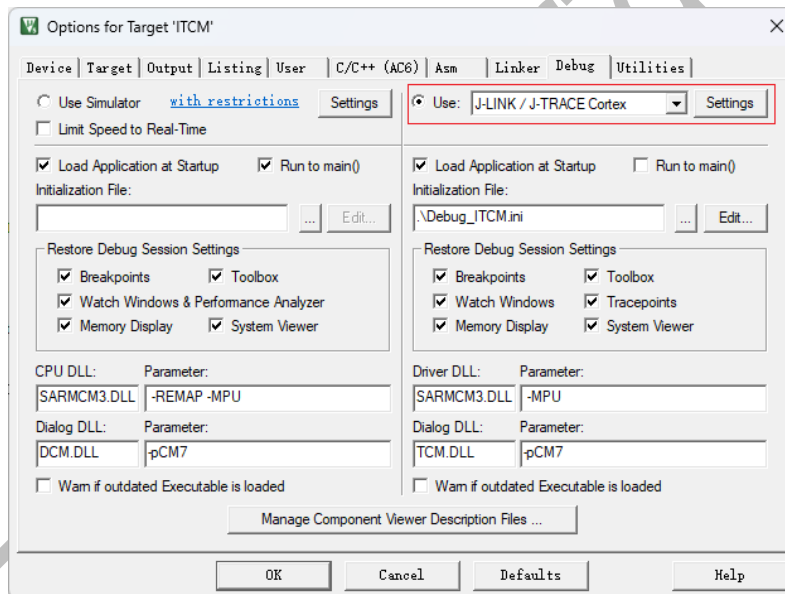
1. 安装 DFP pack 后, Device 属性如下:

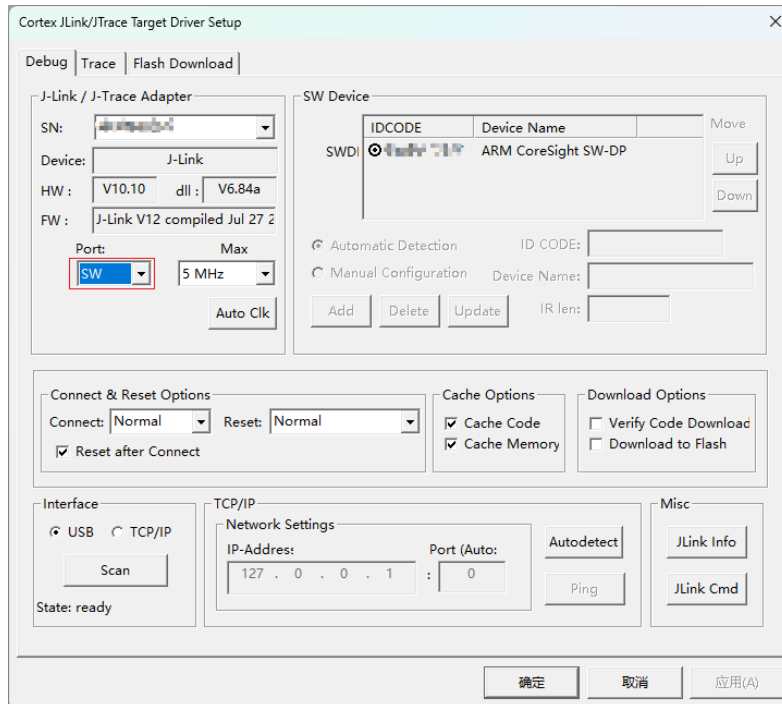


2. 示例工程默认运行在 TCM 中, 如果需要运行在 SRAM 或 Flash, 需要修改 IROM 和 IRAM 的地址范围。

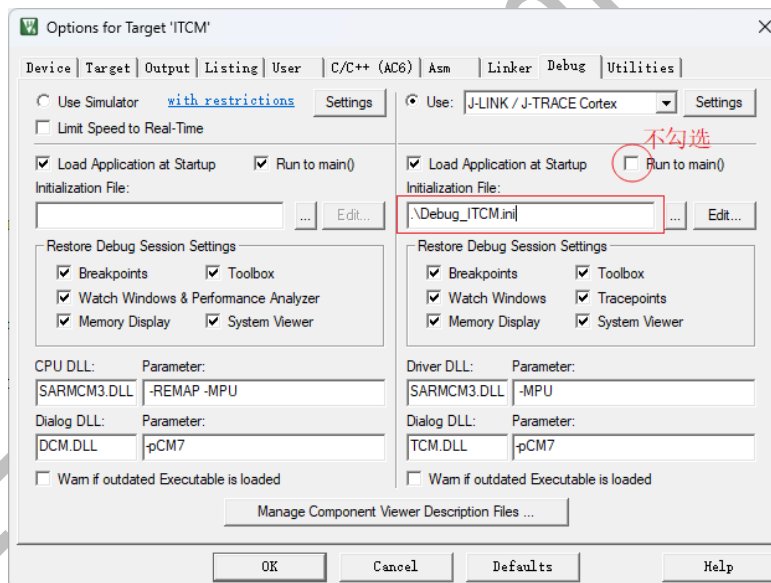


3. 示例工程默认使用 J-Link 作为调试工具，接口为 SW。目前 ET6001 支持的调试器有 CMSIS-DAP 和 J-Link 两种。

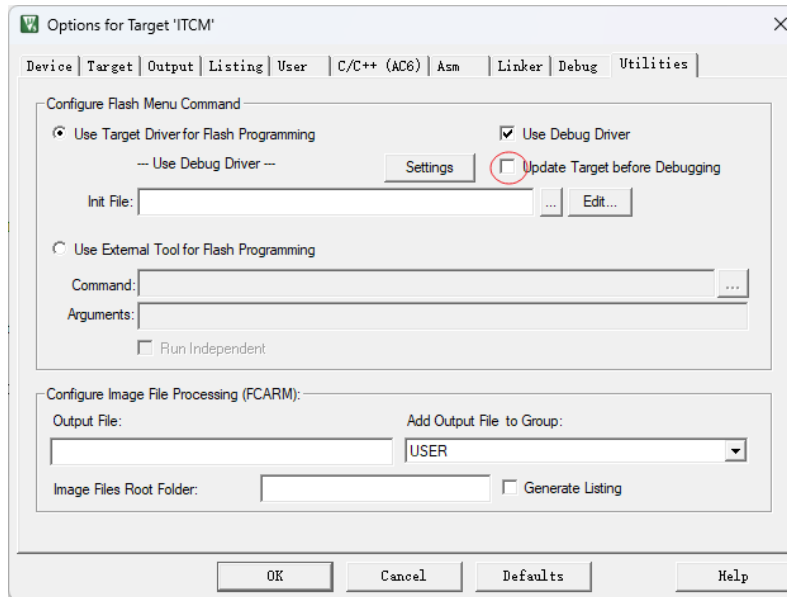




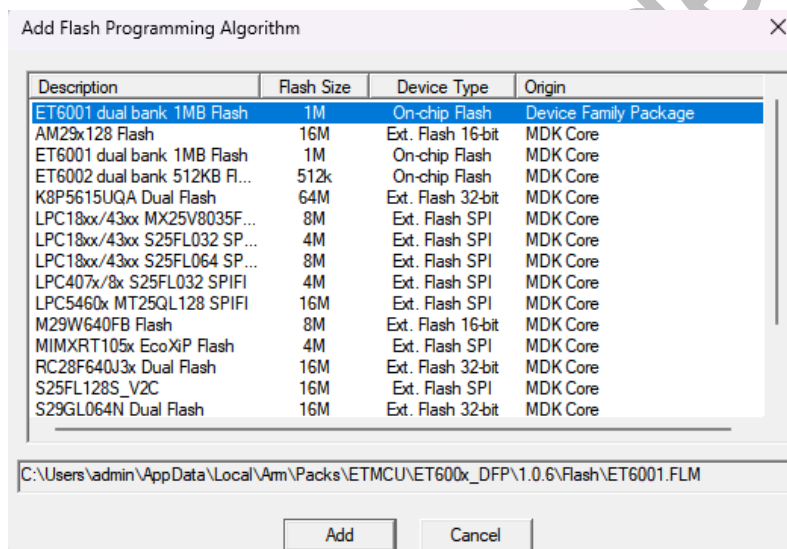
4. 调试前，还需要选择 **Debug Initialization File**。该文件在示例工程下包含。如下图为运行在 ITCM 所需要的 **Debug Initialization File**。

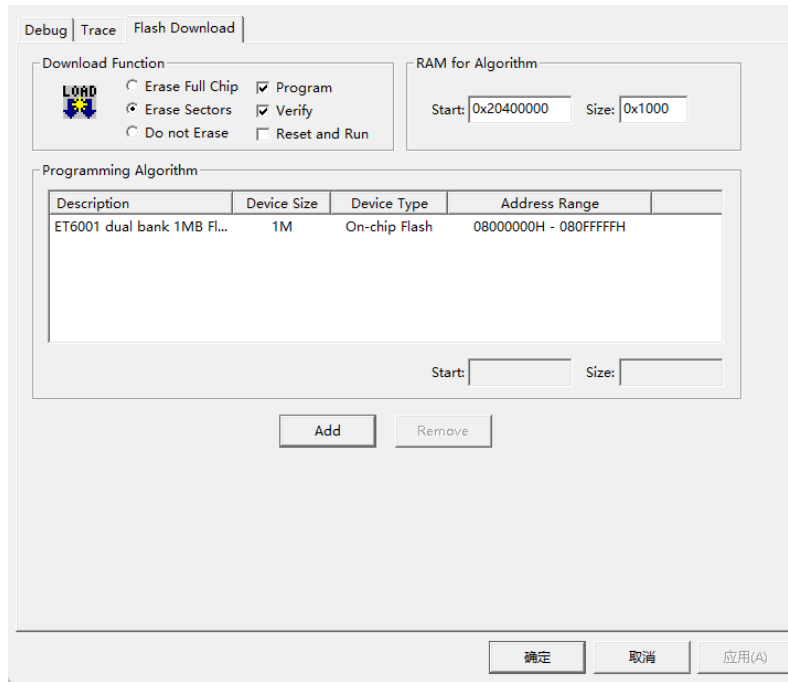


5. 如果程序运作在非 Flash 的 Target，不能勾选 **Update Target before Debugging**。运行在 Flash Target 下，则需要勾选。



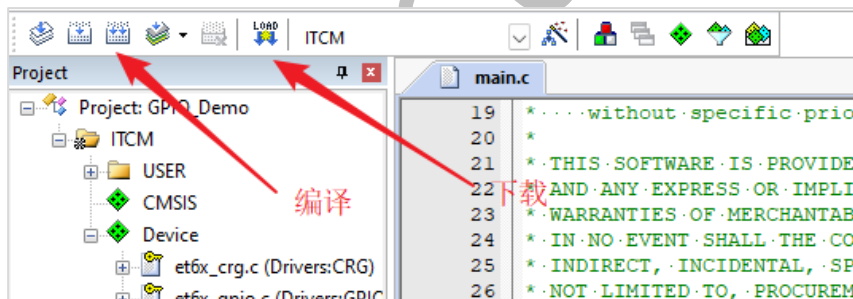
6. 最后，还需要添加 Flash 下载算法如下：





2.1.3 程序下载

完成前 2 步工作后，点击编译按钮编译工程，编译成功后点击下载按钮下载程序 (Flash Target)。



下载成功后，Build Output 窗口显示如下：

```
Target info:
-----
Device: ET6001
VTarget = 3.251V
State of Pins:
TCK: 0, TDI: 0, TDO: 0, TMS: 1, TRES: 1, TRST: 1
Hardware-Breakpoints: 8
Software-Breakpoints: 8192
Watchpoints: 4
JTAG speed: 5000 kHz

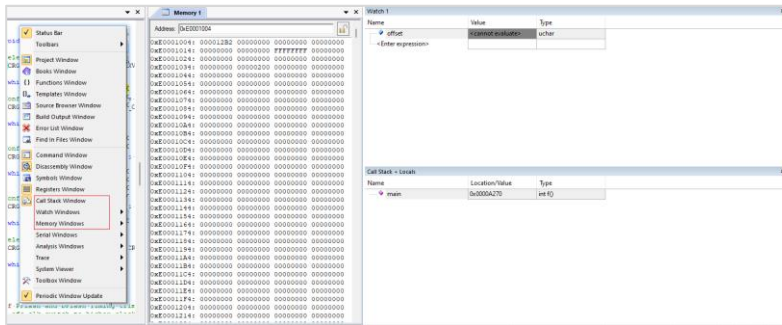
Erase Done.
Programming Done.
Verify OK.
Flash Load finished at 18:28:04
```

2.1.4 程序调试

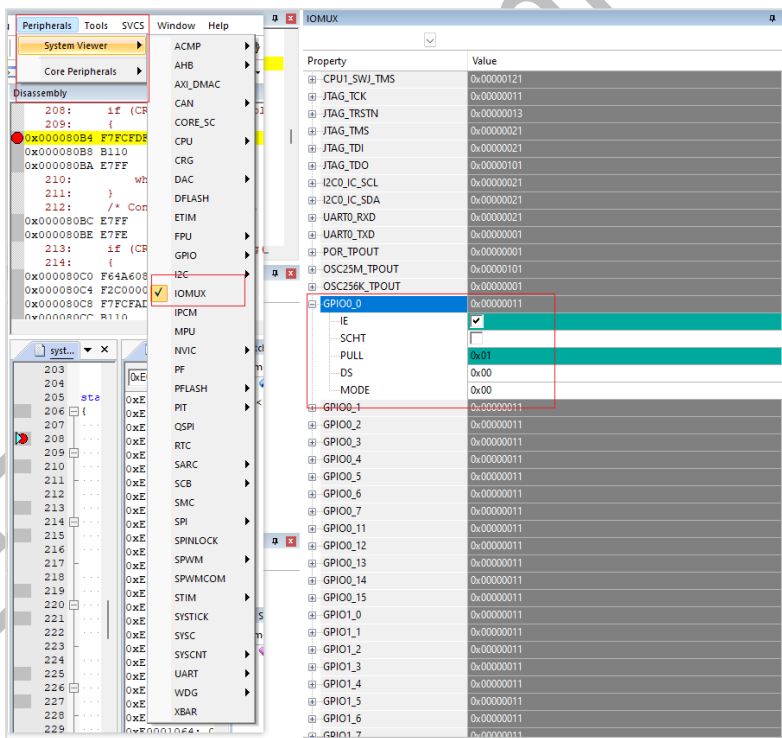
1. 按下调试按钮后即可开始进入调试模式。



2. 调试过程中需要查看变量或者指定内存数据可通过 MDK View 菜单选择 **Watch Windows** 和 **Memory Windows** 查看。



3. 调试过程中如果需要查看寄存器状态或者修改寄存器值可通过 **Peripherals** 菜单选择相应的外设进行查看与设置。



2.2 程序脱机下载

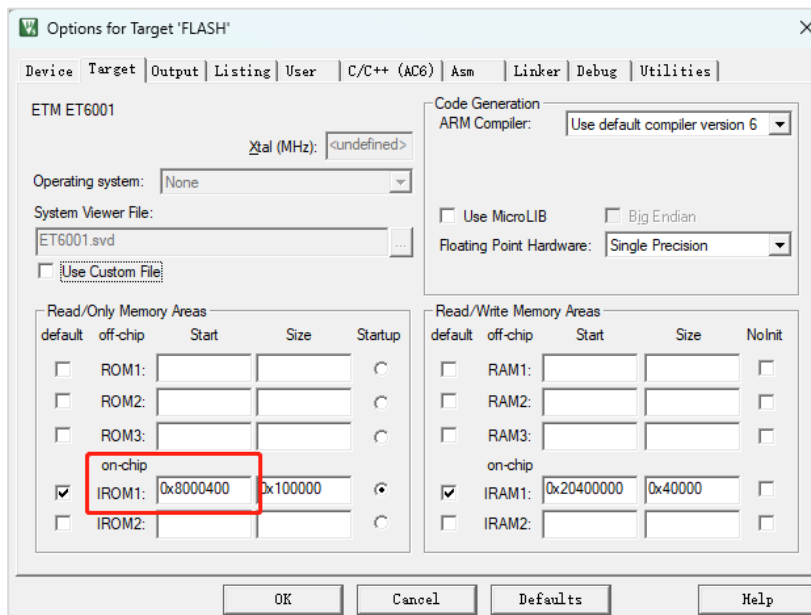
2.2.1 准备工作

1. 使用 USB-TTL 串口连接 ET6001 UART0 的 TX, RX. USB 口接 PC。

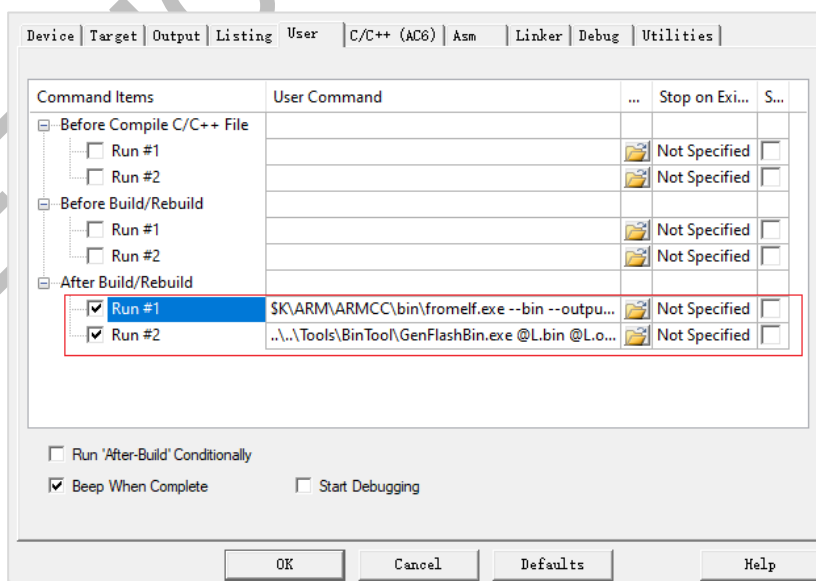
2. 下载 FlyMCU 软件 (<http://www.mcuisp.com/software/FlyMcu.rar>)。
3. 通过跳线设置芯片 Bootmode 管脚为 BOOT_MODE0 和 BOOT_MODE1 分别为 0 和 1。
4. 已安装 MDK 软件 V5.3x 以上版本。
5. 已安装 **ETMCU.ET600x_DFP.1.x.x.pack** 以上版本 DFP package。

2.2.2 编译下载

1. 确认工程配置为 Flash Target，且 IROM 起始地址为 0x08000400。



2. 添加自定义命令如下(示例工程已包含), 并勾选。编译成功后会自动生成添加了 App Header 的固件。

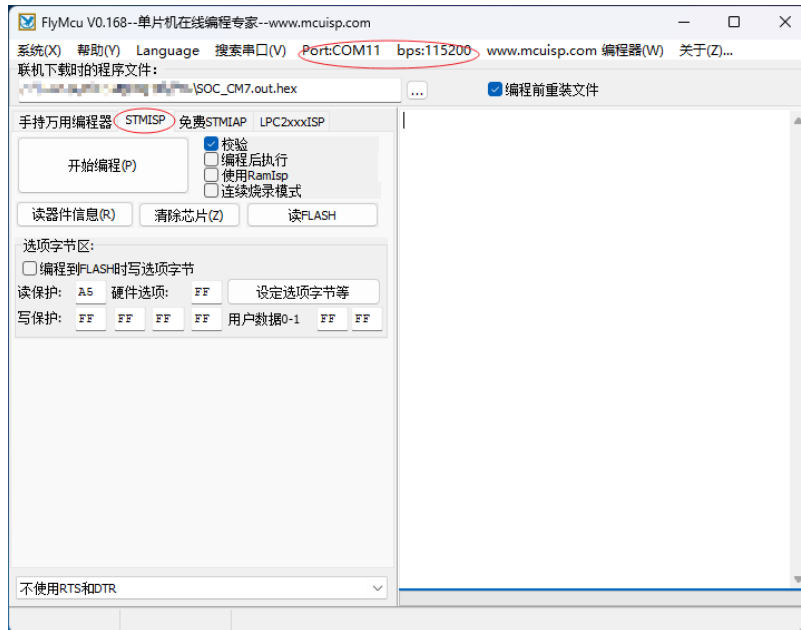


```
linking...
Program Size: Code=66076 RO-data=8116 RW-data=2000 ZI-data=4328
FromELF: creating hex file...
After Build - User command #1: C:\Keil_v5\ARM\ARMCC\bin\fromelf.exe --bin --output=SOC_CM7.bin .\Objects\SOC_CM7.axf
After Build - User command #2: ..\..\Tools\BinTool\GenFlashBin.exe SOC_CM7.bin SOC_CM7.out.bin
input bin file length: 74984
header_checksum: 0x810e29db, image_checksum: 0x14825aff
-----convert complete!
*.Objects\SOC_CM7.axf - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:12
```

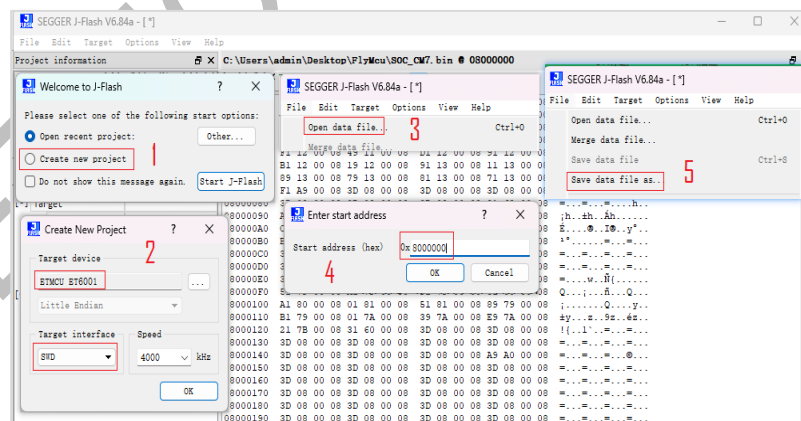
说明

芯片出厂默认没有使能 CRC_EN 固件启动 CRC 校验。若用户已开启，则 BootROM 只会引导加载经过上图 GenFlashBin 工具转换后的固件。

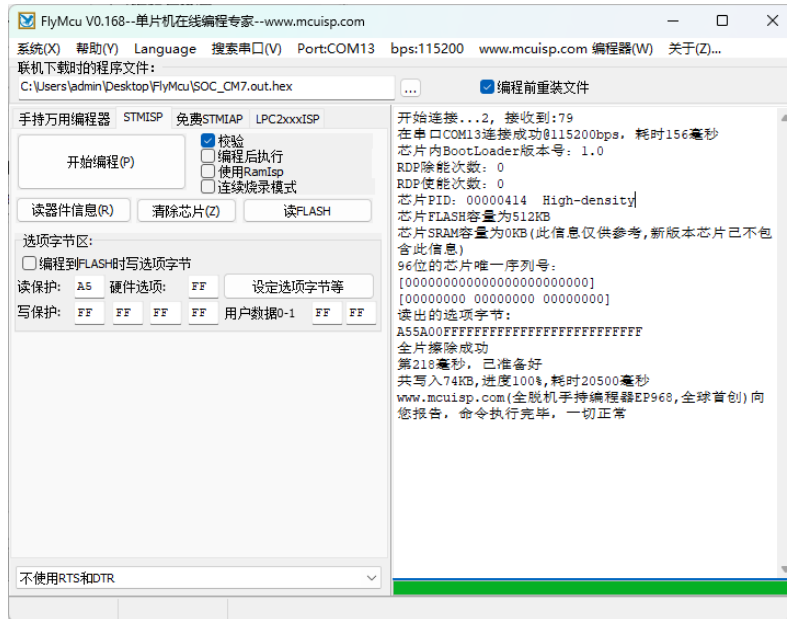
3. 打开 FlyMCU 软件，选择正确的串口端口和波特率 (115200)。



4. 利用 J-Flash 工具将 bin 文件转换为 hex 格式，起始地址 0x08000000 (另存为时，文件格式选择 Intel Hex File)。



5. 设置 BOOTMODE pin 为 01 自举模式。
6. 芯片上电，FlyMCU 选择要下载的文件，点击开始编程，下载完成后如下图所示：



7. 设置 BOOTMODE pin 为 00 Flash 启动模式, 重新上电。

修订记录

修订记录累积了本文档每次修订的具体信息。

序号	版本	修改日期	修改说明
1	1.0.0	2023/08/25	初始发布
2	1.0.1	2024/04/23	新增章节 1.1.1 "MDK-ARM 软件包安装"~2
3	1.0.2	2024/07/15	章节 1.1 "Keil 集成开发环境安装" 和 2.1 "使用 MDK 调试和下载程序" 增加补充描述。