第 1 章　用 KEIL5 新建工程模版

版本说明：MDK5.15

1.1 新建工程

1.1.1 新建本地工程文件夹

为了工程目录更加清晰，我们在本地电脑上新建 6 个文件夹，具体如下：

表格 1 工程目录文件夹清单

名称	作用
Doc	用来放对程序说明的文件，由写程序的人添加
Libraries	存放是库文件
Listing	存放编译器编译时候产生的 c/汇编/链接的列表清单
Output	存放编译产生的调试信息、hex 文件、预览信息、封装库等
Project	用来存放工程
User	用户编写的驱动文件

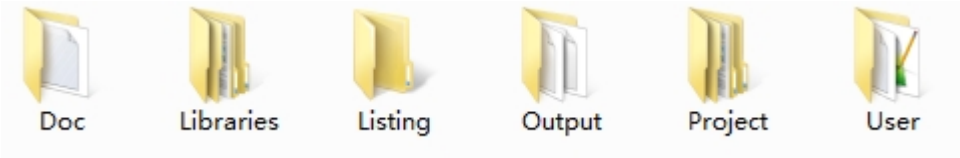


图 10 工程文件夹目录

在本地新建好文件夹后，把准备好的库文件添加到相应的文件夹下：

表格 2 工程目录文件夹内容清单

名称	作用
Doc	readme.txt
Libraries	CMSIS：里面放着跟 CM3 内核有关的库文件

	FWlib：STM32 外设库文件
Listing	暂时为空
Output	暂时为空
Project	暂时为空
User	stm32f10x_conf.h：用来配置库的头文件
	stm32f10x_it.h stm32f10x_it.c：中断相关的函数都在这个文件编写，暂时为空
	main.c：main 函数文件

1.1.2 新建工程

打开 KEIL5，新建一个工程，工程名根据喜好命名，我这里取 LED-LIB，保存在 Project\RVMDK（uv4）文件夹下。

1. 选择 CPU 型号

这个根据你开发板使用的 CPU 具体的型号来选择，MINI 选 STM32F103VE，ISO 选 STM32F103ZE。如果这里没有出现你想要的 CPU 型号，或者一个型号都没有，那么肯定是你的 KEIL5 没有添加 device 库，KEI5 不像 KEIL4 那样自带了很多 MCU 的型号，KEIL5 需要自己添加，关于如何添加请参考《如何安装 KEIL5》这一章。

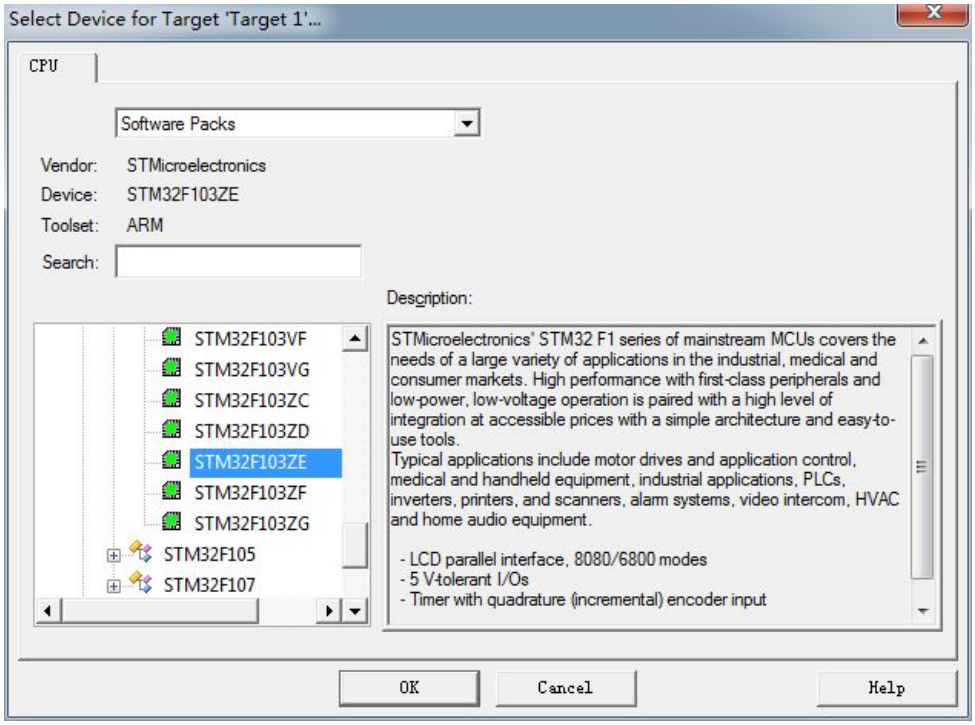


图 11 选择具体的 CPU 型号

2. 在线添加库文件

等下我们手动添加库文件，这里我们点击关掉。

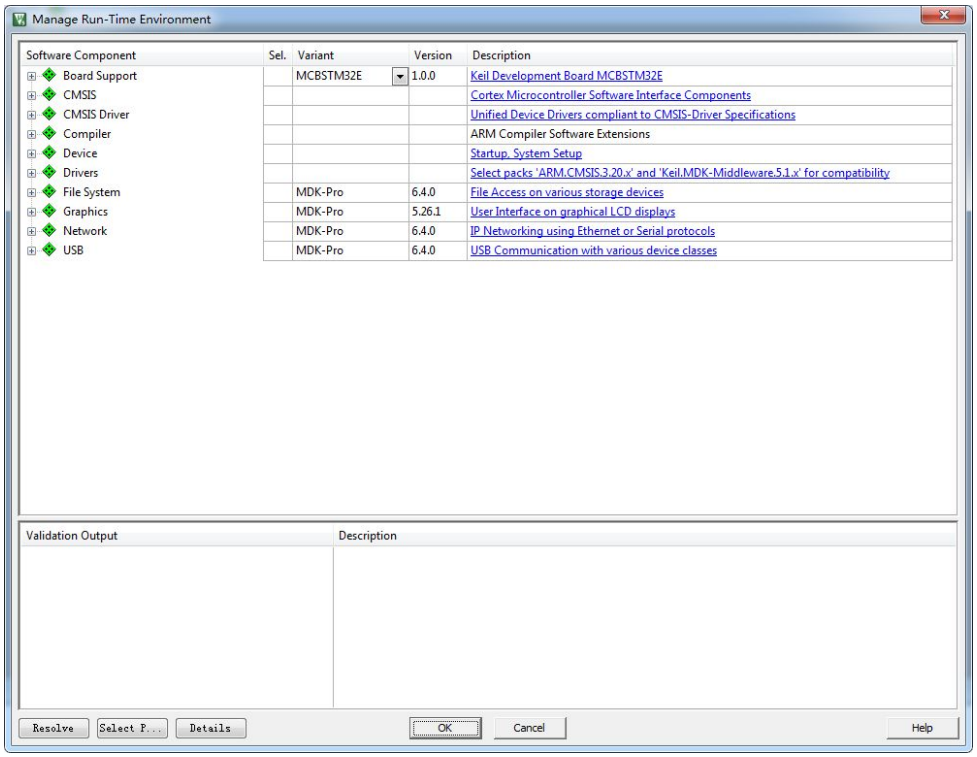


图 12 库文件管理

3. 添加组文件夹

在新建的工程中添加 5 个组文件夹，用来存放各种不同的文件，文件从本地建好的工程文件夹下获取，双击组文件夹就会出现添加文件的路径，然后选择文件即可。

表格 3 工程内组文件夹内容清掉

名称	作用
STARTUP	存放汇编的启动文件：startup_stm32f10x_hd.s
CMSIS	与 CM3 内核有关的库文件： core_cm3.c system_stm32f10x.c
FWLB	与 STM32 外设相关的库文件 misc.c stm32f10x_xxx.c（总共 22 个，xxx 代表外设名称）
USER	用户编写的文件： main.c：main 函数文件，暂时为空 Stm32f10x_it.c：跟中断有关的函数都放这个文件，暂时为空
DOC	readme.txt：程序说明文件，用于说明程序的功能和注意事项等

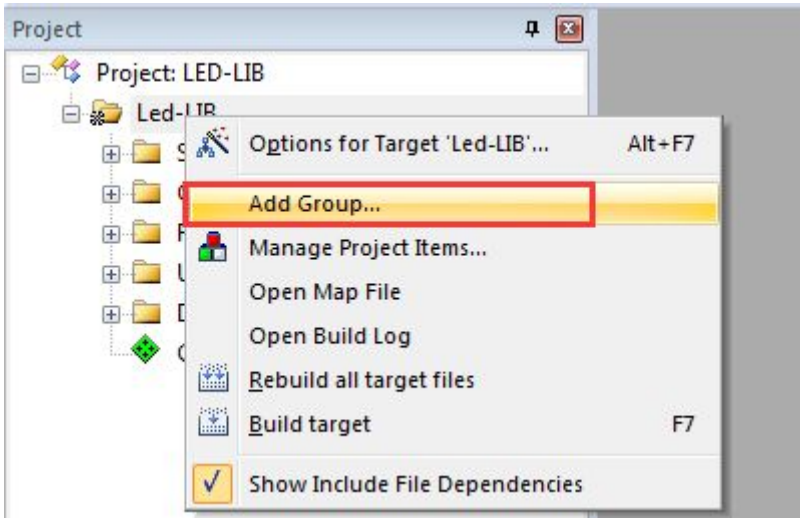


图 13 如何在工程中添加文件夹

4. 配置魔术棒选项卡

这一步的配置工作很重要，很多人串口用不了 printf 函数，编译有问题，下载有问题，都是这个步骤的配置出了错。

①在 Target 中选中微库，为的是在日后编写串口驱动的时候可以使用 printf 函数

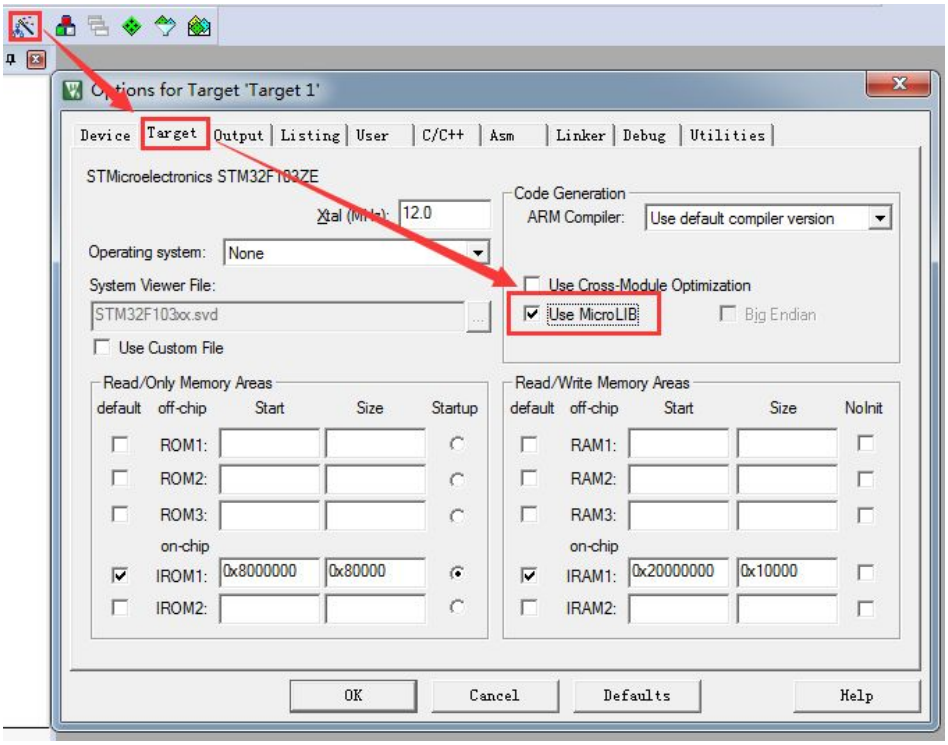


图 14 添加微库

②在 Output 选项卡中把输出文件夹定位到我们工程目录下的 output 文件夹，如果想在编译的过程中生成 hex 文件，那么那 Create HEX File 选项勾上。

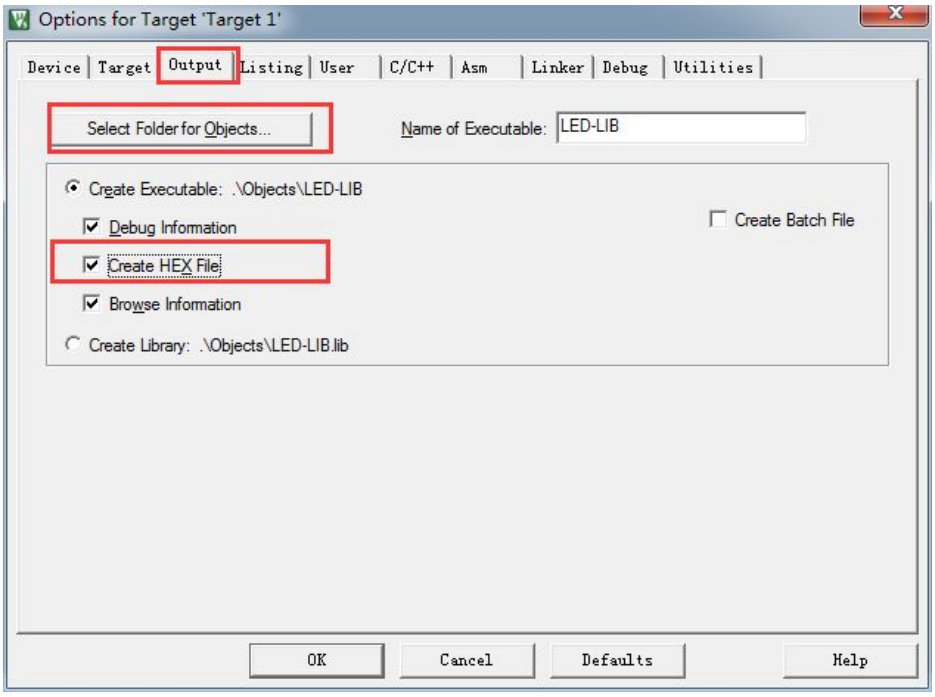


图 15 配置 Output 选项卡

③在 Listing 选项卡中把输出文件夹定位到我们工程目录下的 Listing 文件夹。

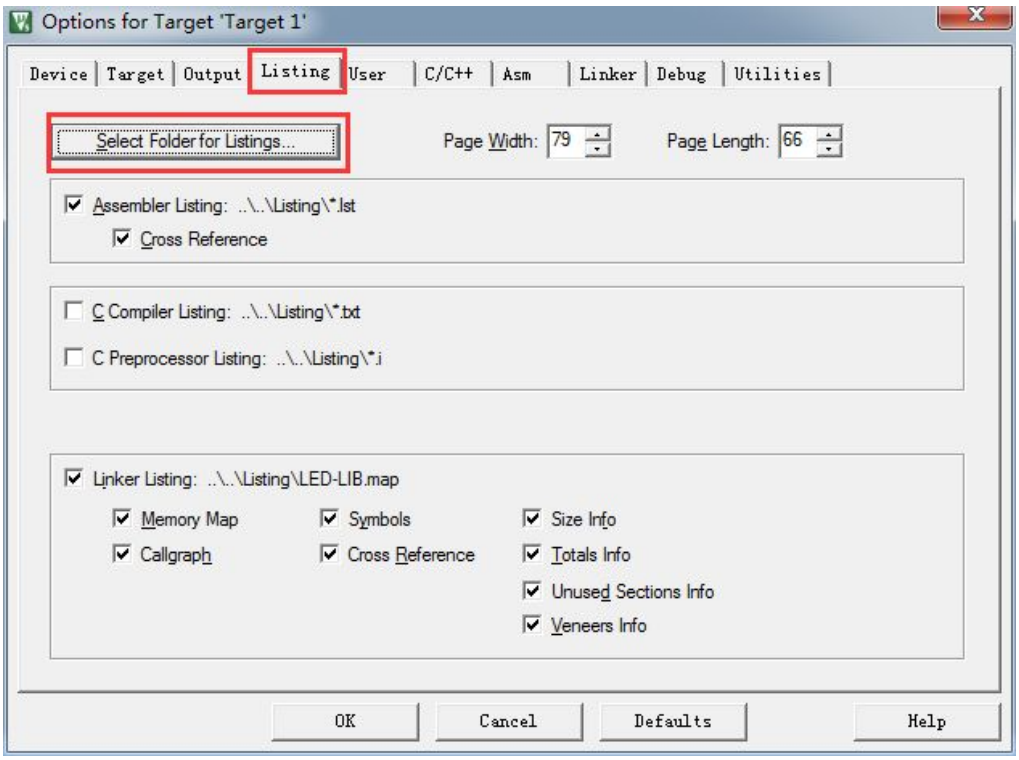


图 16 配置 Listing 选项卡

④在 C/C++选项卡中添加处理宏，和编译器编译的时候查找的头文件路径。

STM32F10X_HD：这个宏是为了区分使用 STM32F103 系列中不同容量型号的单片机库。我们用的单片机的 FLASH 的容量都是 512K，属于大容量

STM32F10X_HD：FLASH 大小在 256K~512K 之间的 STM32F101xx 和 STM32F103xx 控制器。STM32F10X_MD：FLASH 大小在 64K~128K 之间的 STM32F101xx 和 STM32F103xx 控制器。STM32F10X_LD：FLASH 大小在 16K~32K 之间的 STM32F101xx 和 STM32F103xx 控制器。

USE_STDPERIPH_DRIVER：为了包含 stm32f10x_conf.h 这个头文件。

在编译器中添加宏的好处就是，只要用了这个模版，就不用源文件中修改代码或者添加头文件。

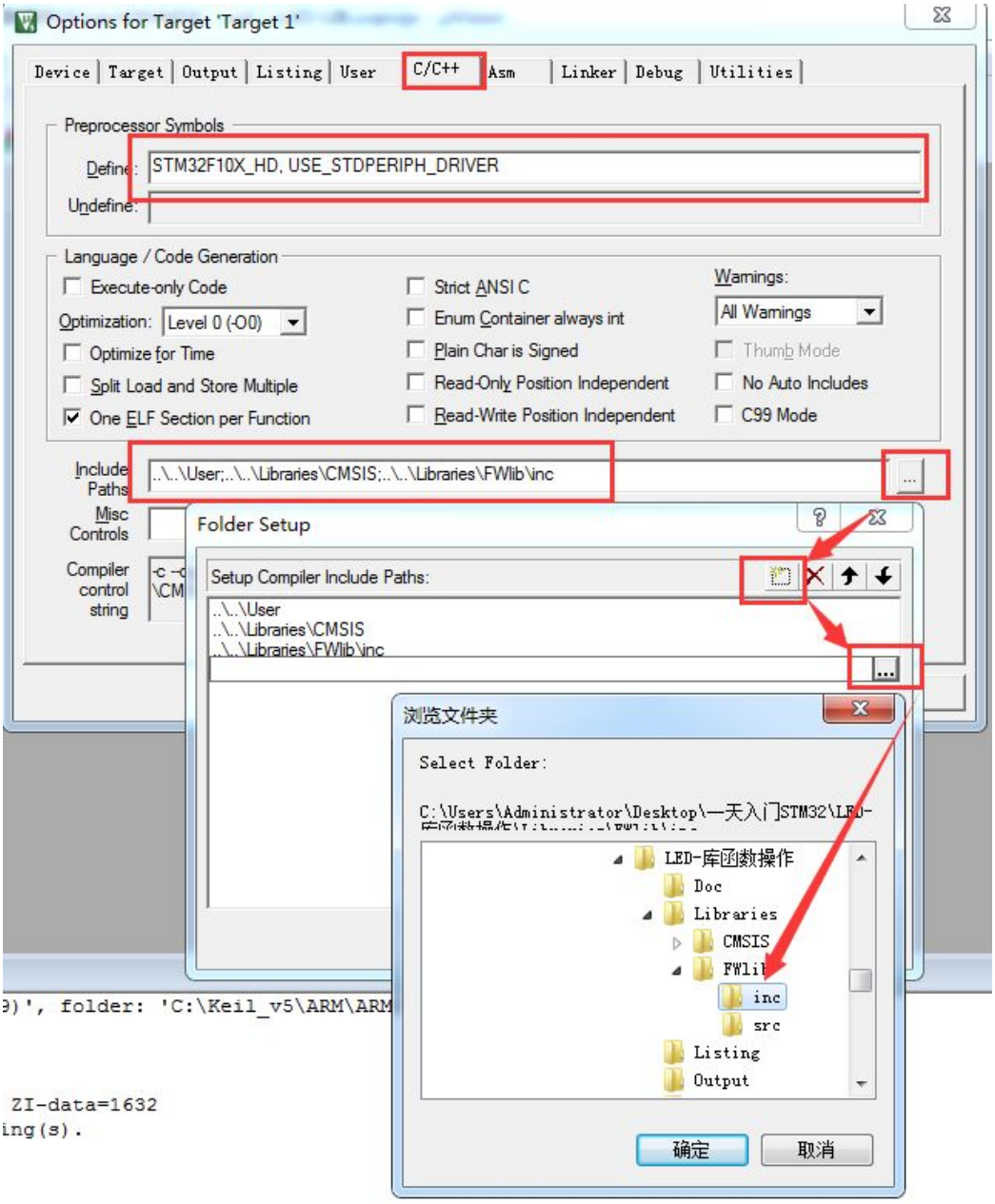


图 17 配置 C/C++ 选项卡

Include Paths 这里添加的是头文件的路径，如果编译的时候提示说找不到头文件，一般就是这里配置出了问题。你把头文件放到了哪个文件夹，就把该文件夹添加到这里即可。

5. 下载器配置

这部分的配置最好是在安装好下载器驱动，下载器连接了电脑和开发板，且开发板上电后来配置。

这里面需要根据你使用了什么仿真器来配置，常用的有三种仿真器：JLINK/ARM-OB，ST-LINK，ULINK2，而且这个配置不是配置完一次之后以后就不会改变，当你换了芯片型号，或者其他操作（具体原因不明）都会改变下载器的配置。

①JLINK/ARM-OB 配置

要先安装了 JLINK 驱动之后，该配置才能下载，两者缺一不可。

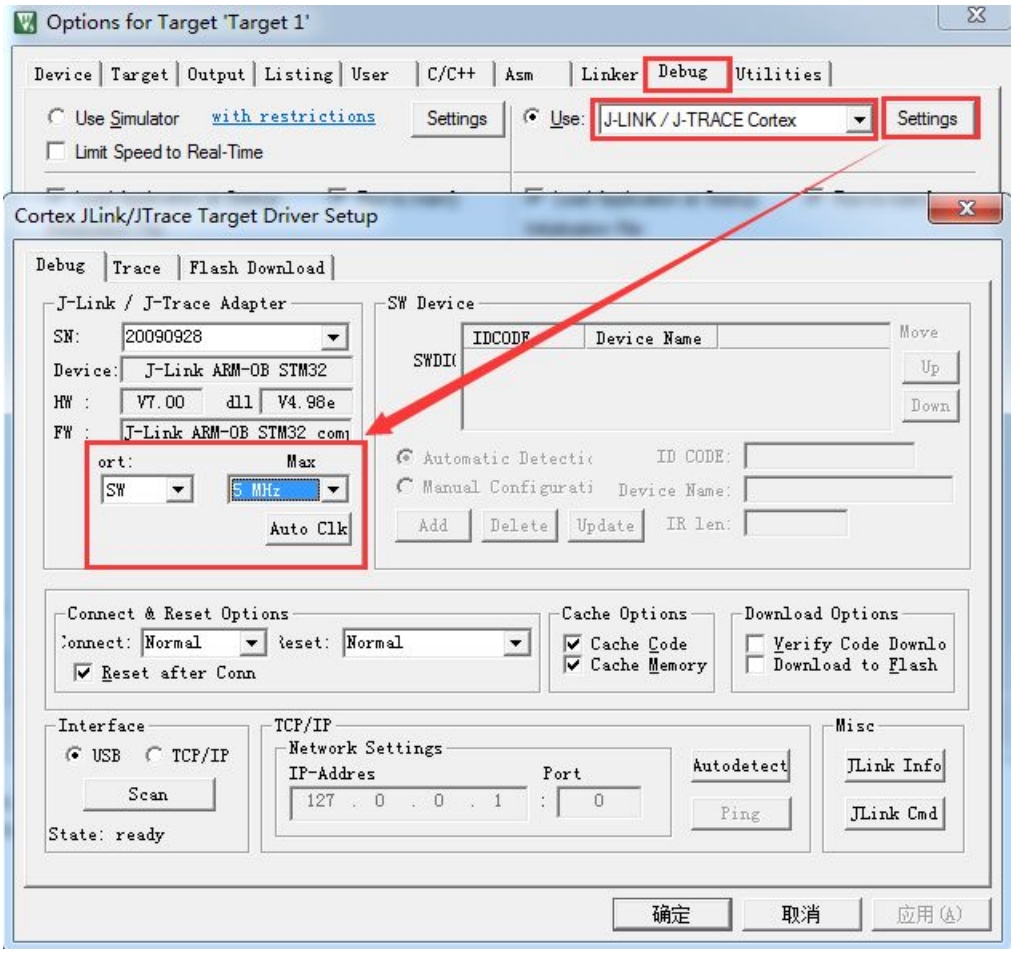


图 18 JLINK/ARM-OB 下载配置

②ST-LINK 配置

要先安装了 ST-LINK 驱动之后，该配置才能下载，两者缺一不可。

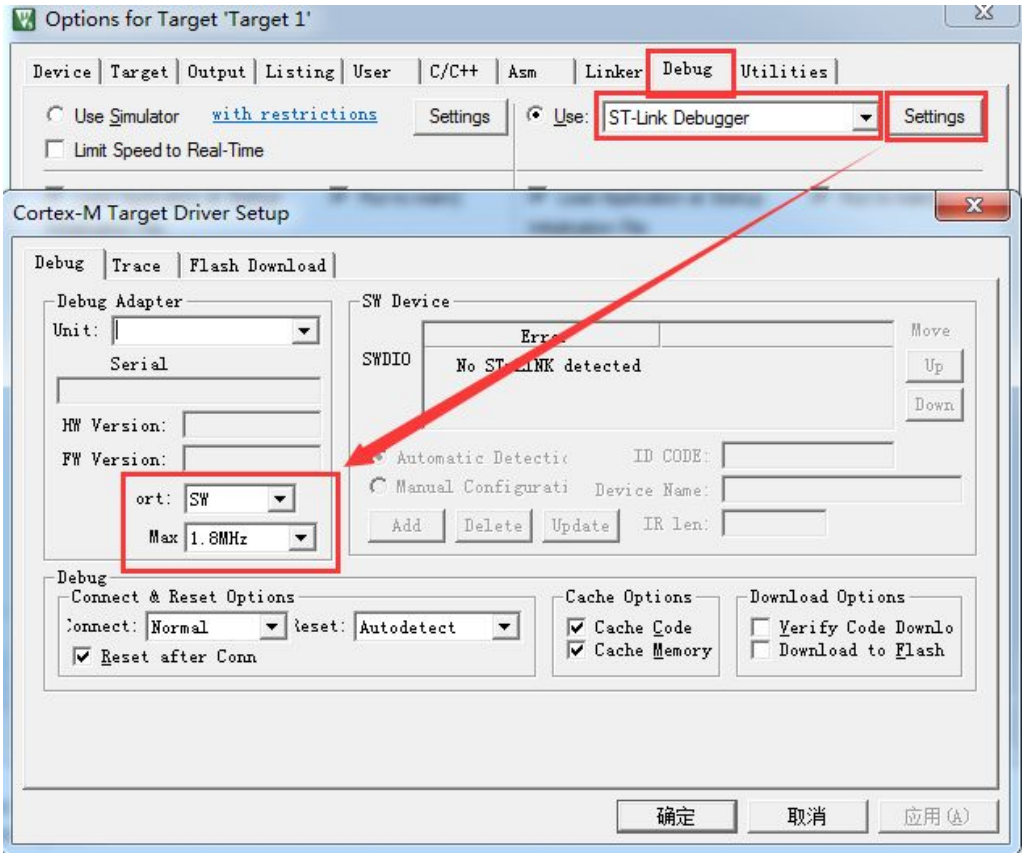


图 19 ST-LINK 下载配置

③ULINK2 配置

要先安装了 ULINK2 驱动之后，该配置才能下载，两者缺一不可。要注意的是设置成 ULINK2，而不是 ULINK。

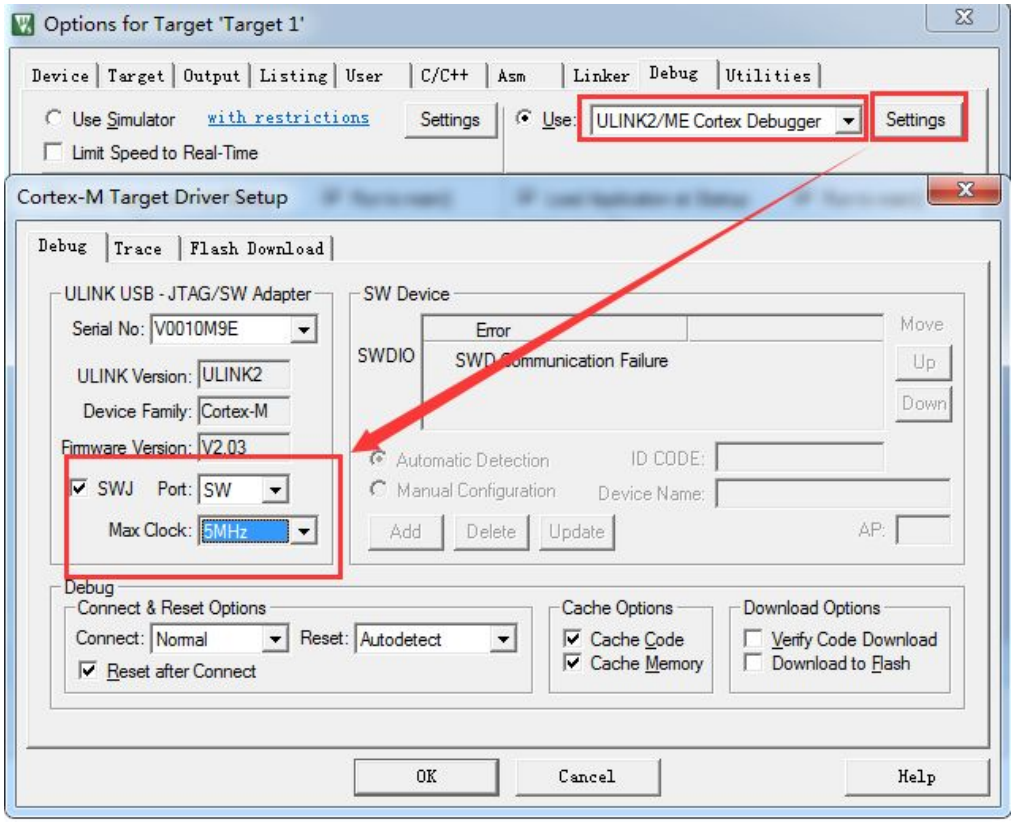
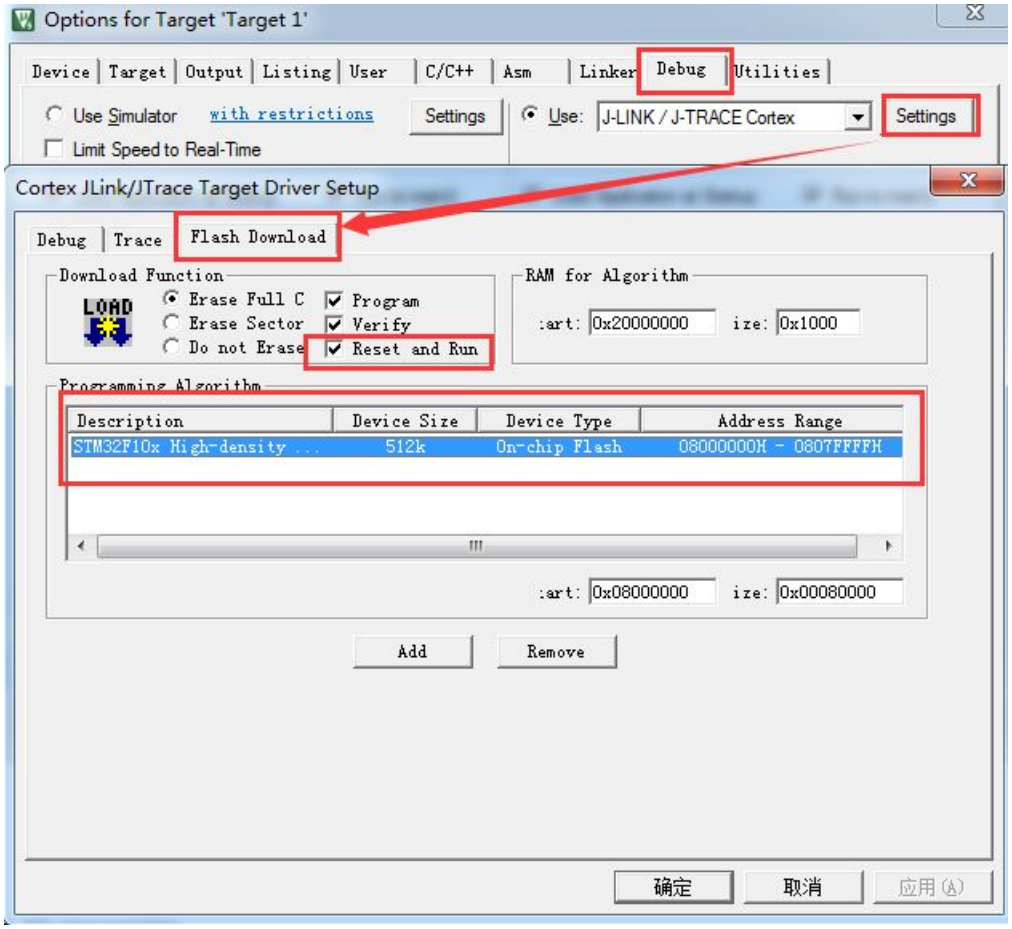


图 20 ULINK2 下载配置

6. 选择 CPU 型号

这一步的配置也不是配置一次之后完事，常常会因为各种原因需要重新选择，当你下载的时候，提示说找不到 Device 的时候，请确保该配置是否正确。有时候下载程序之后，不会自动运行，要手动复位的时候，也回来看看这里的 Reset and Run 配置是否失效。MINI 和 ISO 用的 STM32 的 FLASH 都是 512K，所以选择 512K 大容量，如果使用的是其他型号的，要根据实际情况选择。



一个新的工程模版新建完毕。