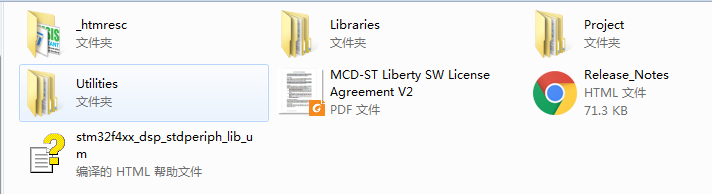
使用STM32标准固件库源码建立开发工程

一：下载stm32固件库源码，解压后目录分布如下



目录解析：

二：创建工程目录结构

1. 新建文件夹，命令为first\_project,内部结构如下



CMSIS：Cortex微控制器软件接口标准文件，该目录下文件适用所有Cortex系列（启动文件、配置文件）

DEVICE\_LIB：M4对应外设的模块代码。

Inc+src

HARDWARE：用户实现的模块功能函数

LIST：链接相关

OBJ：编译产生的中间文件+hex/bin文件

SYSTEM：系统相关代码

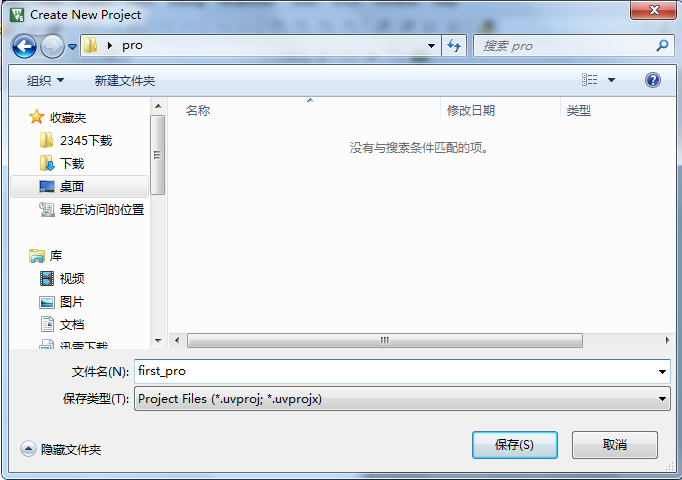
USER：自定义代码（main.c）

1. 将stm32固件库的代码拷贝到各个文件夹内，拷贝位置如下

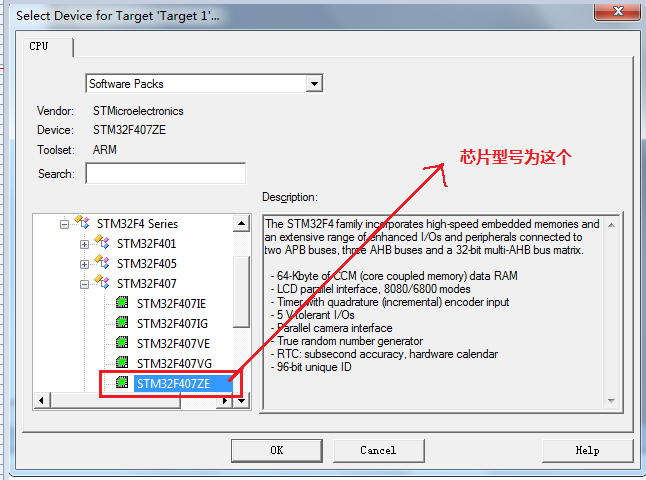
|  |  |
| --- | --- |
| stm32F4xx的官方库 | 工程 |
| Libraries\STM32F4xx\_StdPeriph\_Driver\inc 库头文件夹  Libraries\STM32F4xx\_StdPeriph\_Driver\src 库源文件夹 | DEVICE\_LIB |
| Project\STM32F4xx\_StdPeriph\_Templates\main.c  Project\STM32F4xx\_StdPeriph\_Templates\stm32f4xx\_it.h 中断函数头文件  Project\STM32F4xx\_StdPeriph\_Templates\stm32f4xx\_conf.h 配置文件  Project\STM32F4xx\_StdPeriph\_Templates\stm32f4xx\_it.c 中断函数文件 | USER |
| Libraries\CMSIS\Device\ST\STM32F4xx\Include\stm32f4xx.h  Libraries\CMSIS\Device\ST\STM32F4xx\Source\Templates\ system\_stm32f4xx.c  Libraries\CMSIS\Device\ST\STM32F4xx\Include\system\_stm32f4xx.h | SYSTEM |
| Libraries\CMSIS\Include\core\_cm4\_simd.h  Libraries\CMSIS\Include\core\_cm4.h Cortex-M4系统文件  Libraries\CMSIS\Include\core\_cmFunc.h  Libraries\CMSIS\Include\core\_cmInstr.h  Libraries\CMSIS\Device\ST\STM32F4xx\Source\Templates\arm\startup\_stm32f40\_41xxx.s | CMSIS |

三：keil5创建工程 + 配置工程（项目保存位置：自己新建的工程文件夹first\_project/USER）

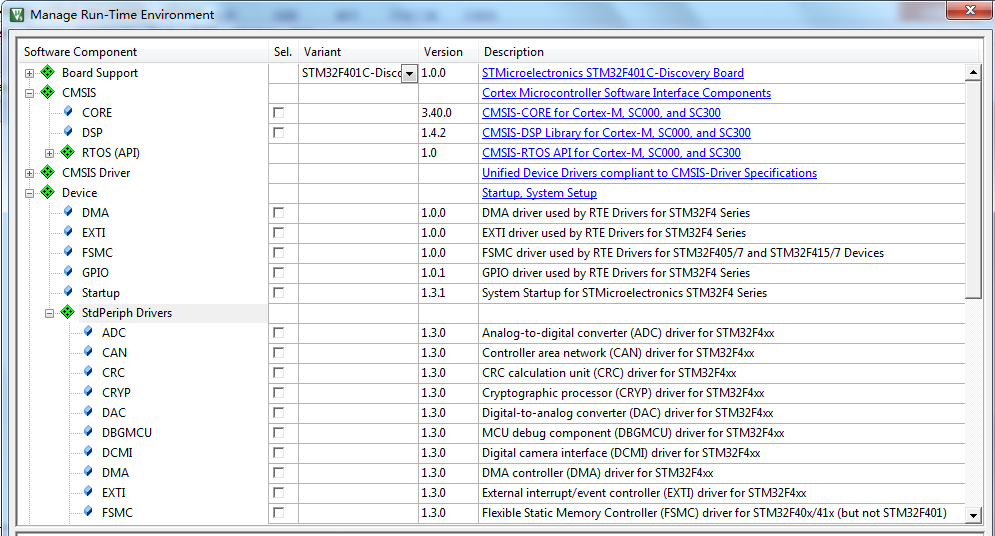
1. 打开keil5，选中project-->new project-->在弹出的对话框中，选择项目保存位置。



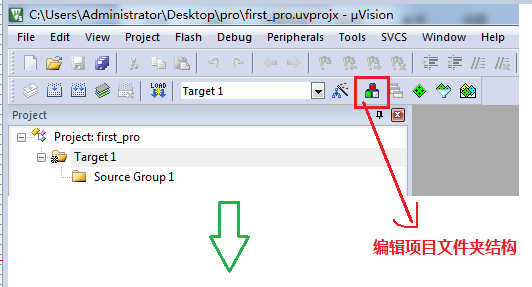
1. 保存后，选择该工程的芯片型号，如下图。（前提是安装了芯片型号对应的pack）

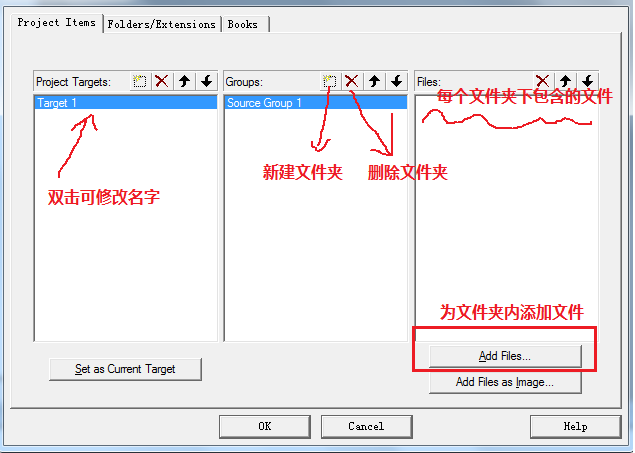


1. 选定型号后，出现下图，为该项目添加要使用的模块代码，需要就勾选。（使用固件库源码复制，不需要此步骤，直接关闭）。

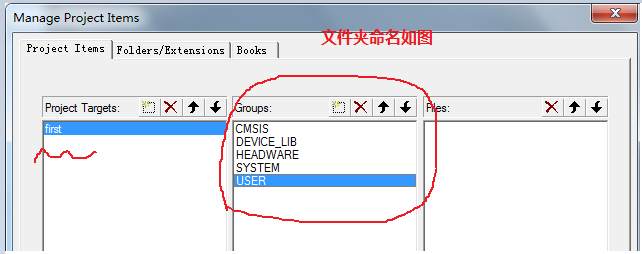


1. 创建成功后，出现下图结构。（必须对项目内文件夹分布作规划）

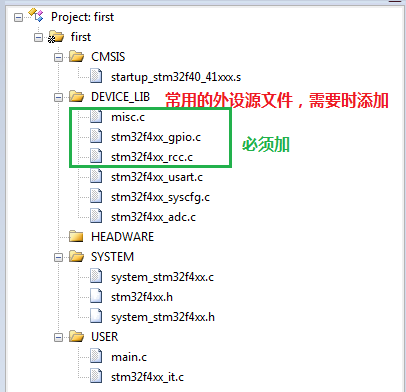




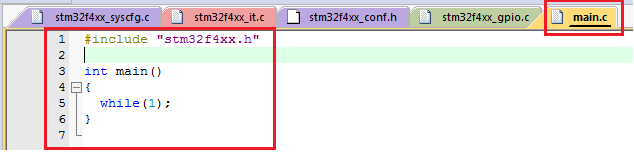
每个文件夹命名与工程代码框架类似，见下图



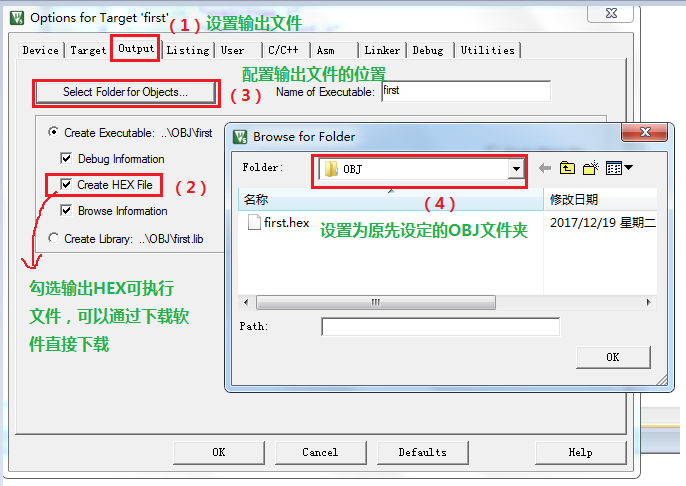
5. 每个文件夹内文件分布如下

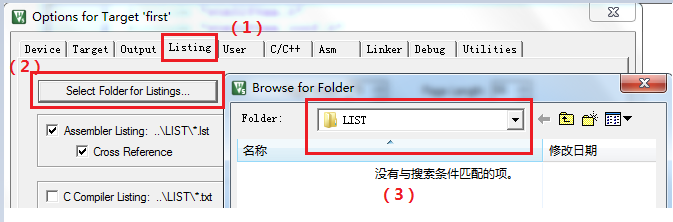


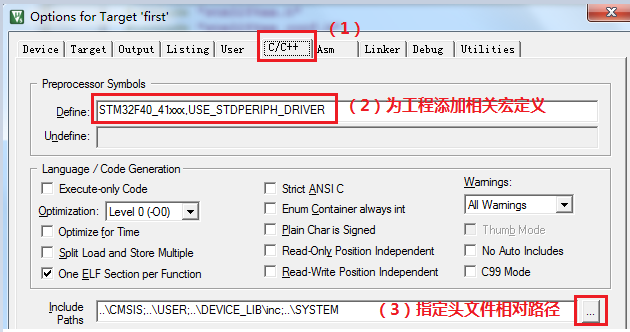
6. 打开main.c，删除所有代码，作以下简单代码编写。



1. 对工程进行配置，具体步骤如下。Project-->options for target

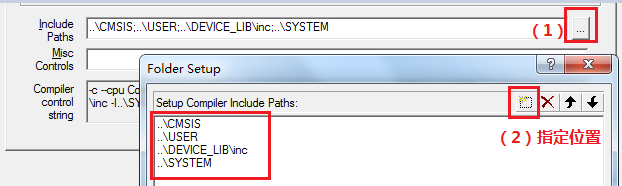




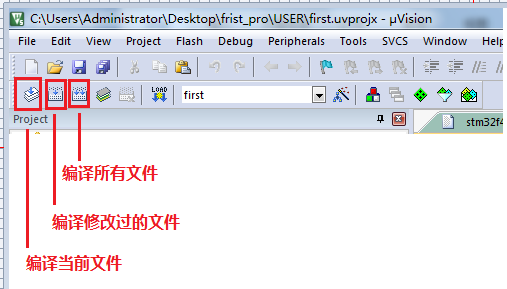


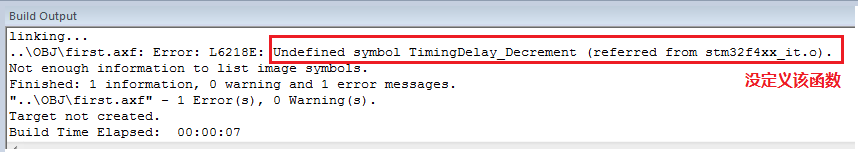
**STM32F40\_41xxx：该宏指定芯片的型号，不同型号对应的硬件代码有不同，必须定义。**

**USE\_STDPERIPH\_DRIVER：该宏指定是否启用外设，定义则为启用外设，必须定义。**

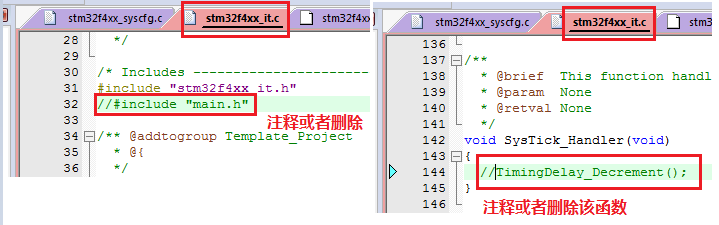


1. 点击按钮编译。

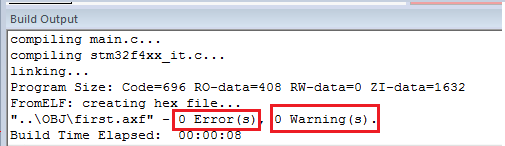




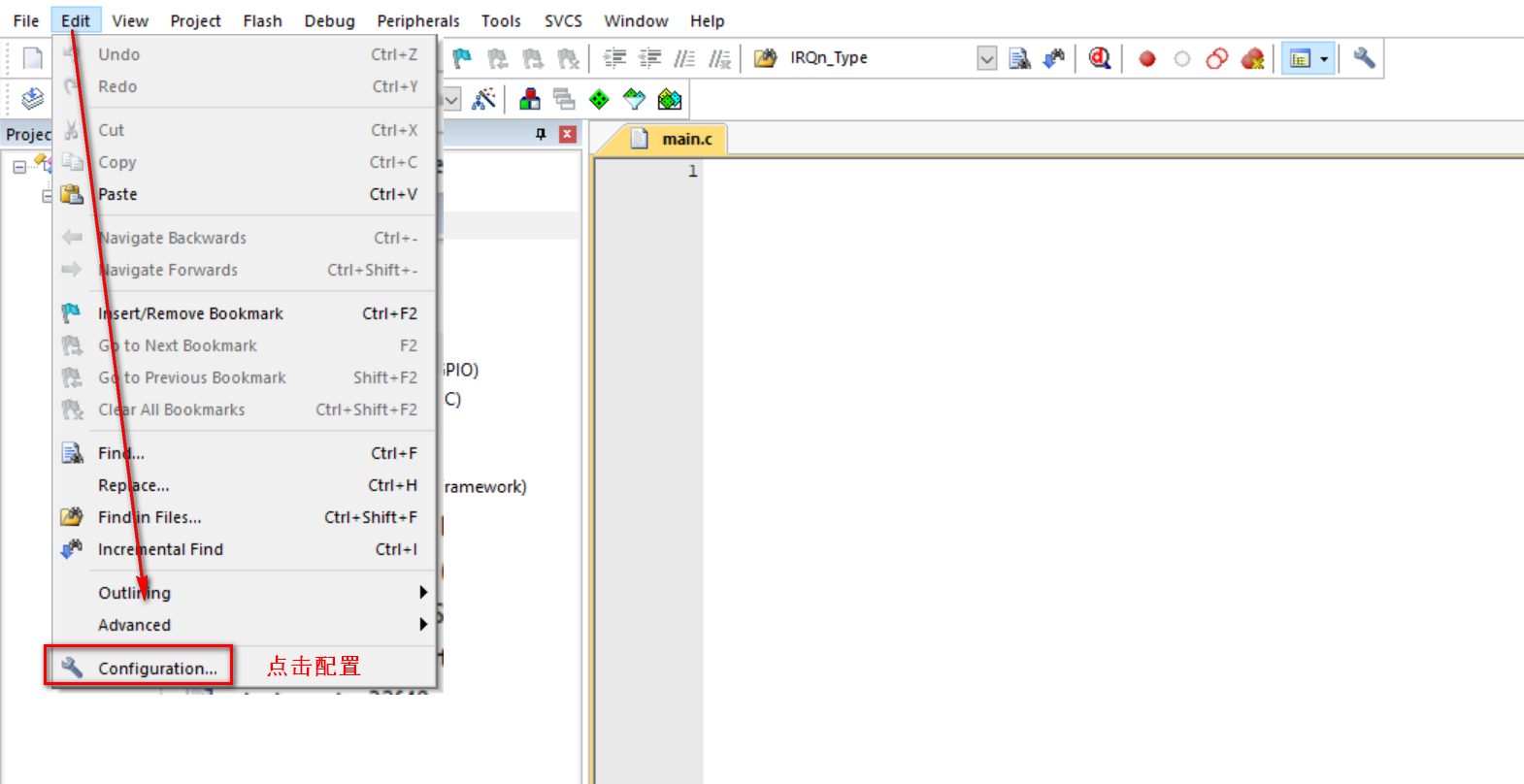
1. 编译出错，因为拷贝的中断源文件中，添加了多余代码，作以下修改。

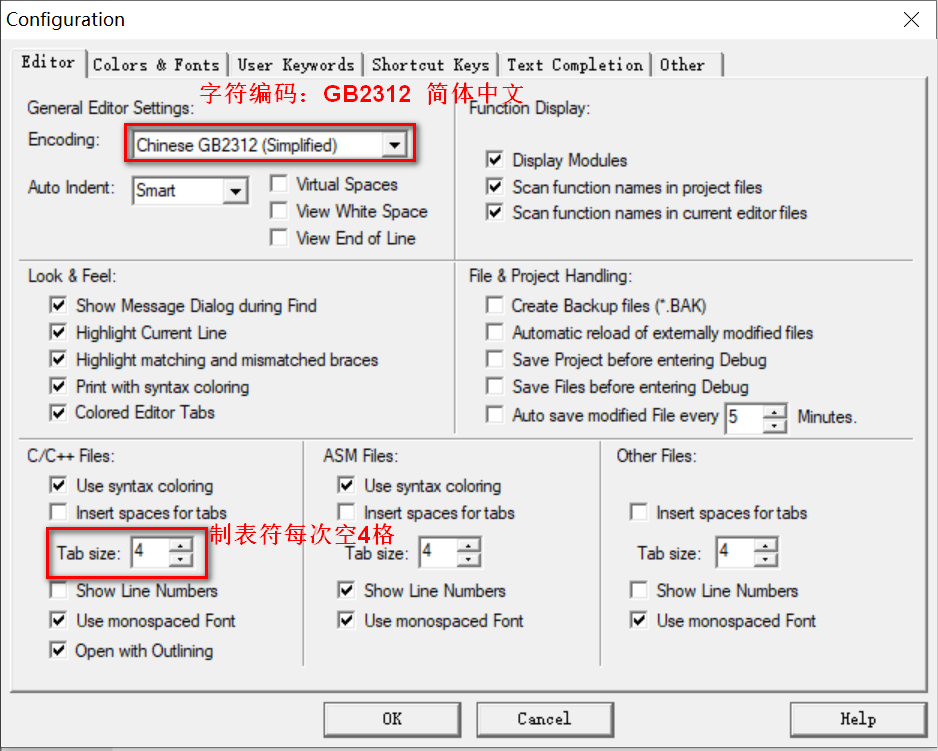


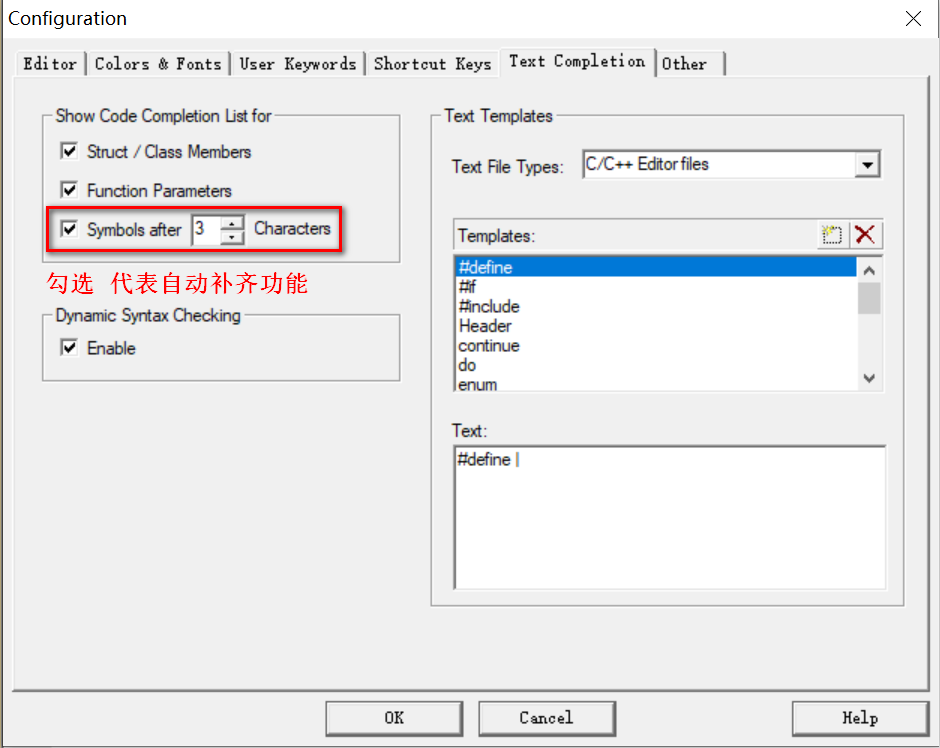
10.成功编译通过，可以往里面添加模块功能。



11.点击MKD5菜单栏的Edit选项，配置MKD5，方便以后开发







12.整个工程配置结束，可以正常使用