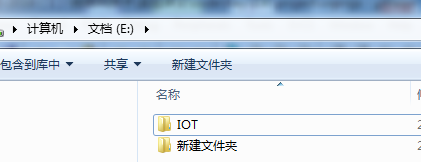
Keil C开发环境使用

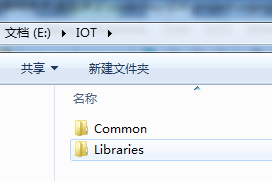
## （一）新建基于STM32F40x固件库的MDK5工程模板

说明：

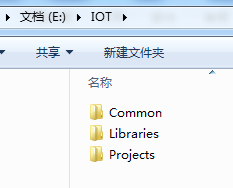
1. ST（意法半导体）为方便用户开发程序，提供了一套丰富的STM32F4固件库。固件库就是函数的集合，固件库函数的作用是向下负责与寄存器直接打交道，向上提供用户函数调用的接口（API）。
2. 本课程实验用的STM32F401RE所需要的固件库，在文件夹Libraries以及Common下，在实验的每个工程中都会使用这两个文件夹。
3. 选择一个盘符，如E盘下，建立文件夹IOT，作为所有本课程实验的总文件夹。



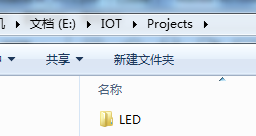
1. 把Common及Libraries拷贝到IOT下。



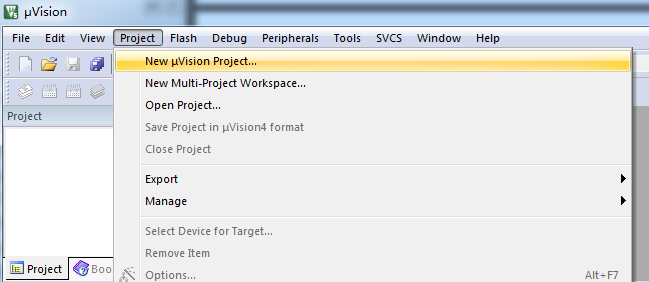
1. 在IOT下新建Projects文件夹，用于存放本学期所做的所有实验工程。



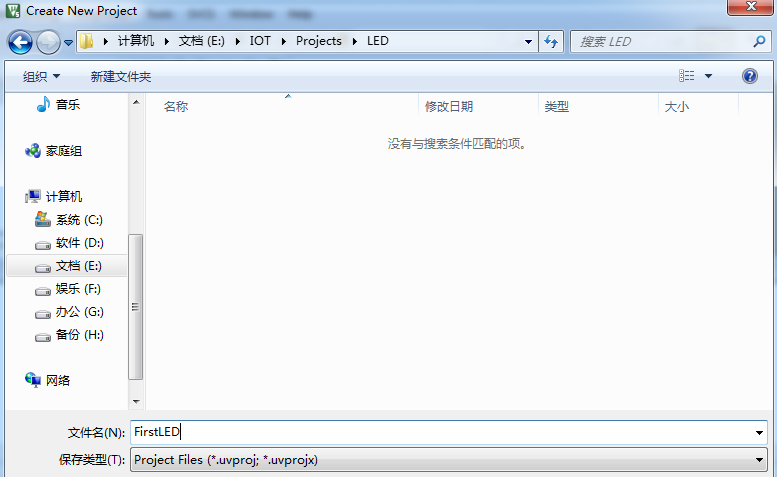
4、在Projects文件夹下新建第一个实验LED的目录。



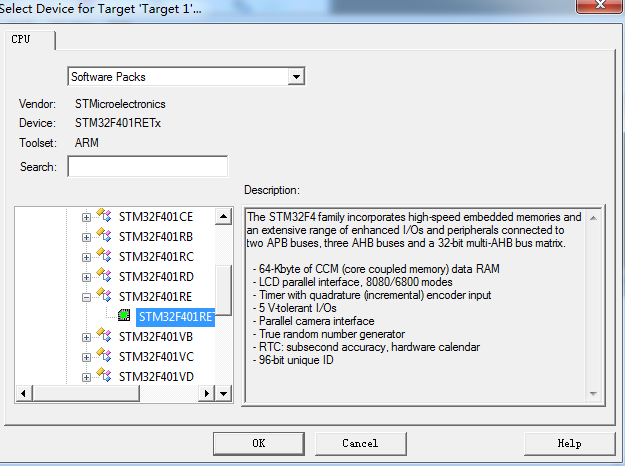
5、新建工程



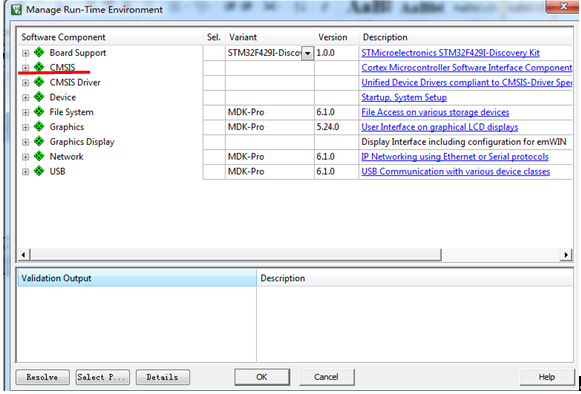
6、工程命名：FirstLED，放在LED文件夹下



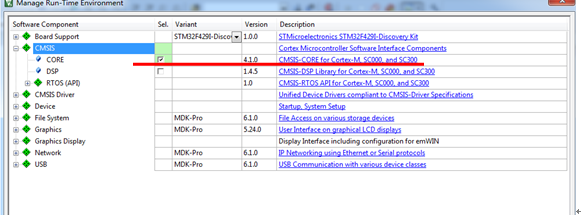
7、选择器件型号



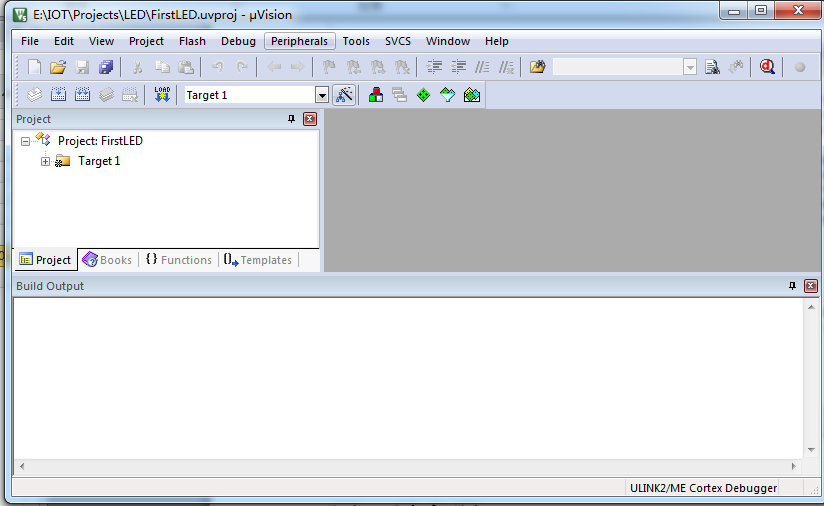
8、上一步点击OK之后，跳出管理运行环境（Manage Run Time Environment）：这是MDK5新增的一个功能，在这个界面，我们可以添加自己需要的组件，从而方便构建开发环境。选择CMISS



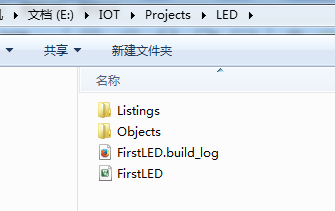
选中CMISS-Core 4.1.0版本。



9、工程界面如下图所示。

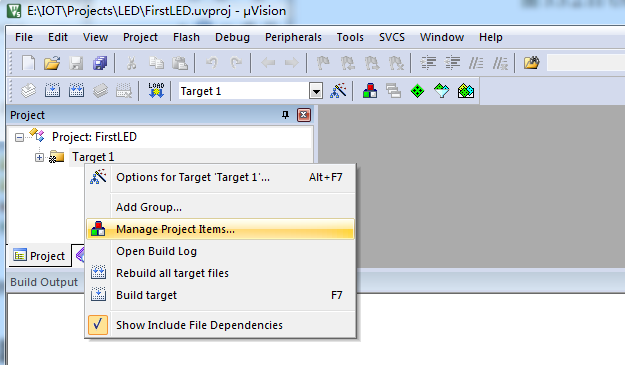


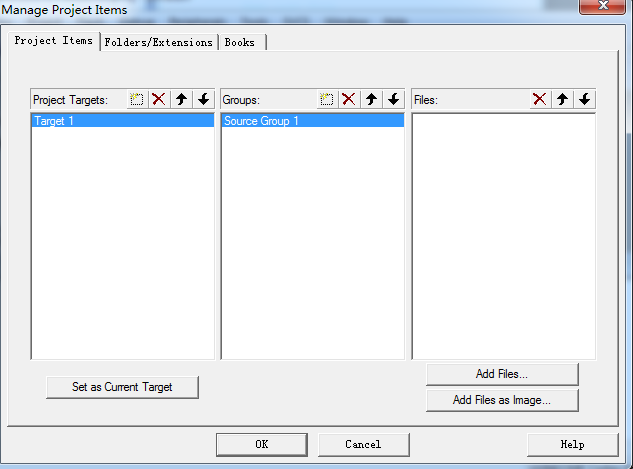
LED文件夹下包含如下图：



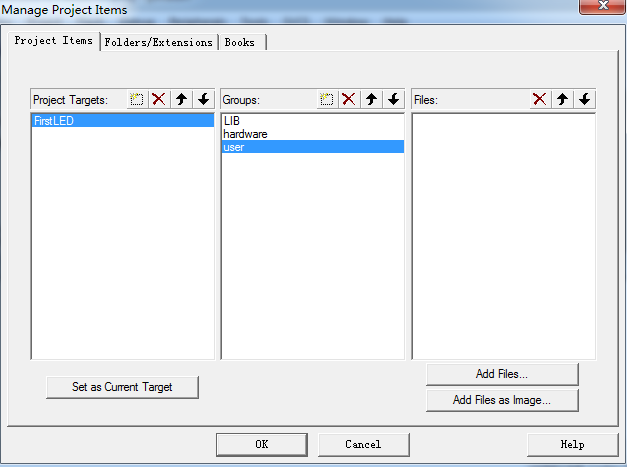
10、将需要的固件库相关文件加入到工程中去。

右键点击Target1，选择Manage Progect Items。

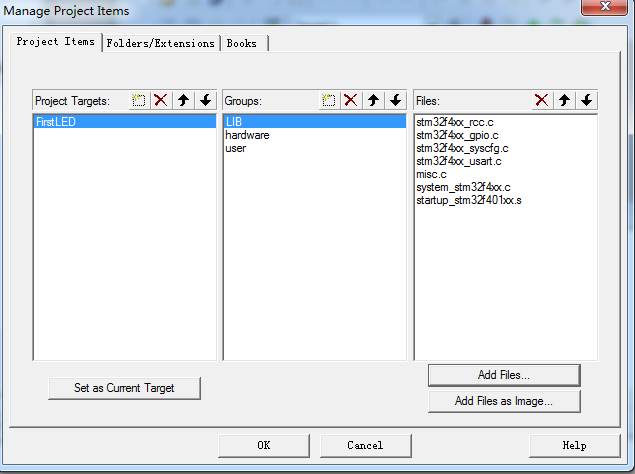


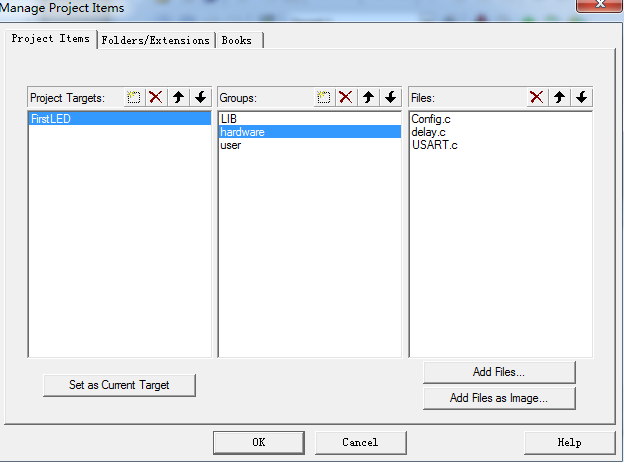


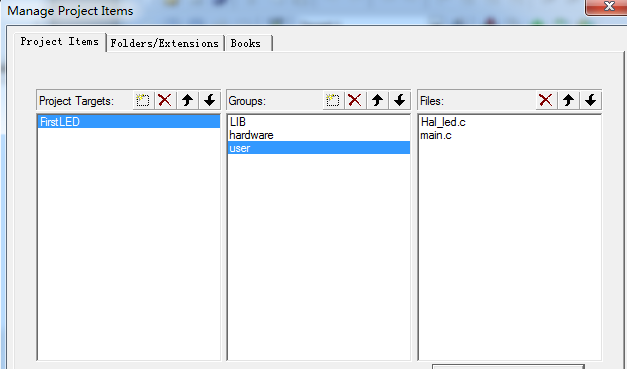
将Target1改名为FirstLEd，删除Source Group1，添加Groups: LIB,hardware,user文件夹



往Group里添加文件。

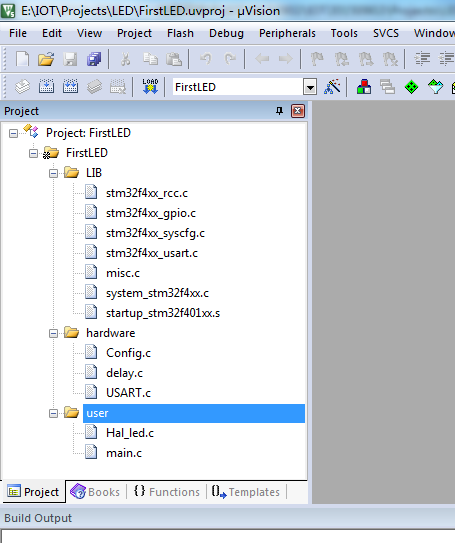




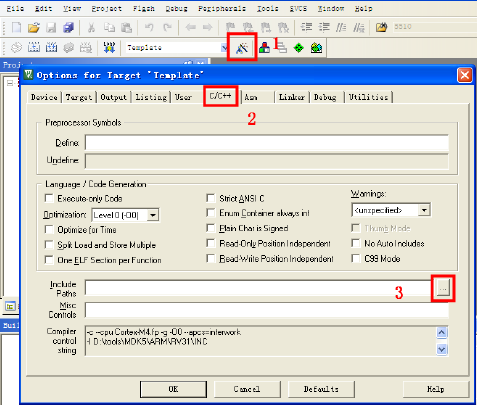


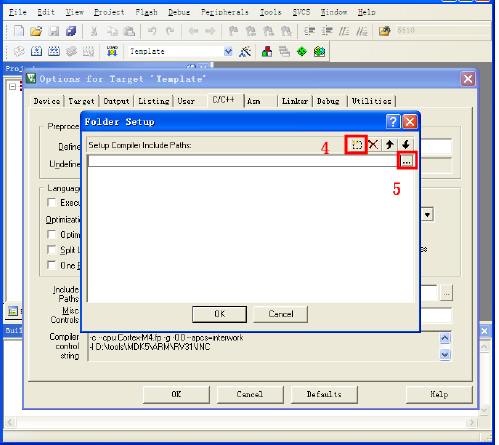
所有添加的固件库文件都在我们提供的Common以及Libraries中，user下的文件是用户自己编写的应用程序文件，LED实验的文件我们提供在SRC文件中。

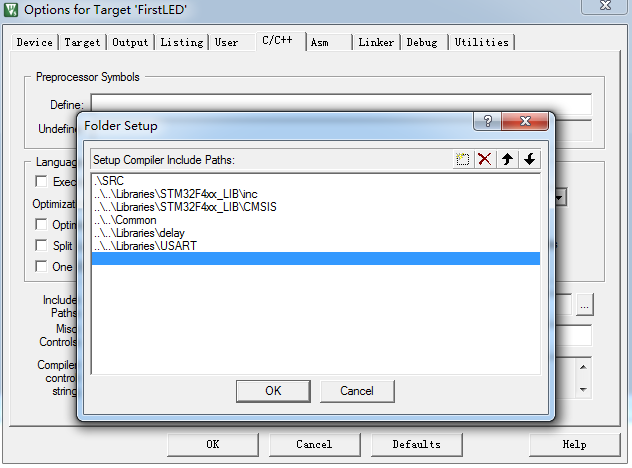
工程目录如下：



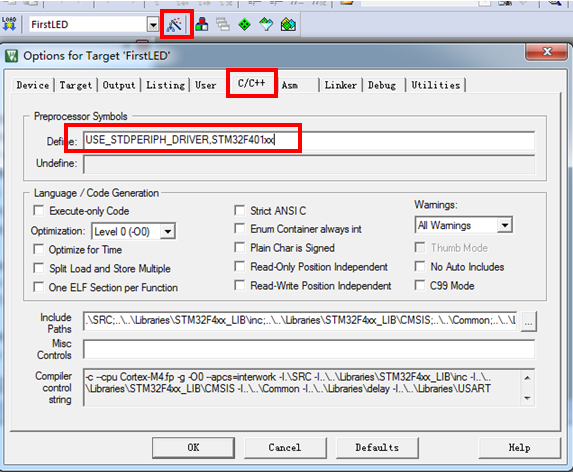
11、接下来设置工程的头文件存放路径。



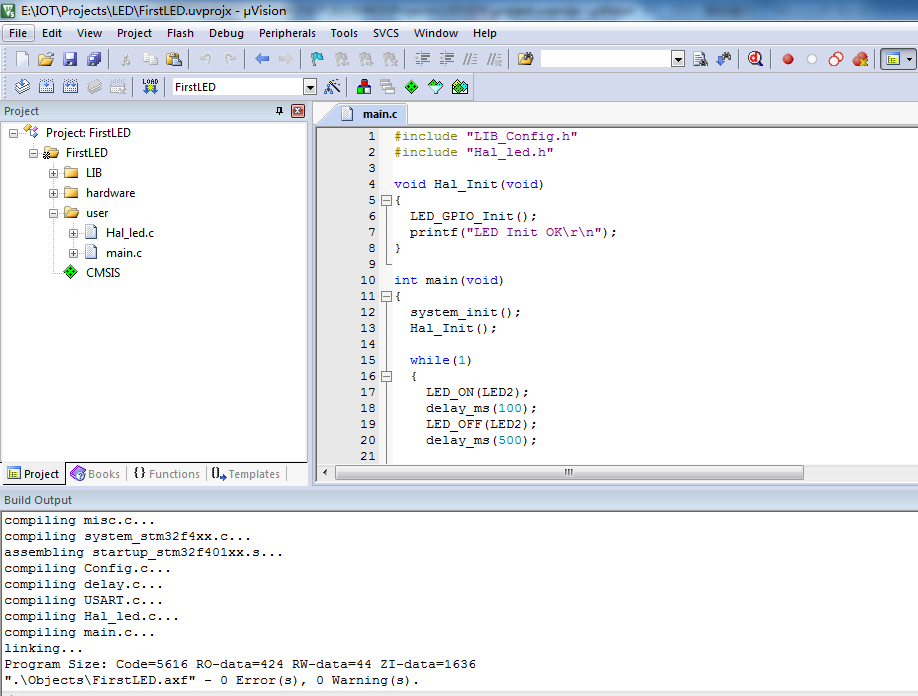




12、接下来对STM32F40系列的工程，还需要添加一个全局宏定义标识符。添加方法：点击魔术棒，选择C/C++选项卡，然后在Define输入框输入USE\_STDPERIPH\_DRIVER,STM32F401xx，这里有两个标识符USE\_STDPERIPH\_DRIVER和STM32F401xx，它们之间是半角逗号隔开，请注意。



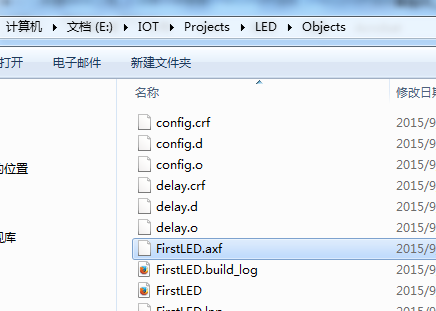
13、接下来点击编译工程。可以看到编译通过没有任何错误和警告。



14、到这里，工程创建完毕，在工程的Objects文件夹下生成了FirstLED.axf。将FirstLED.axf下载到开发板，可以看到LED灯的闪烁。

编译输出窗口看到：

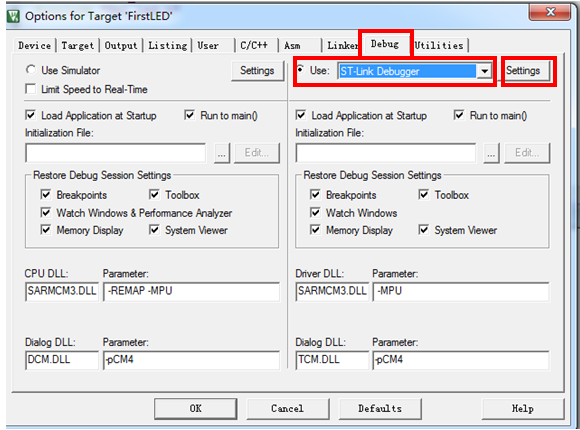




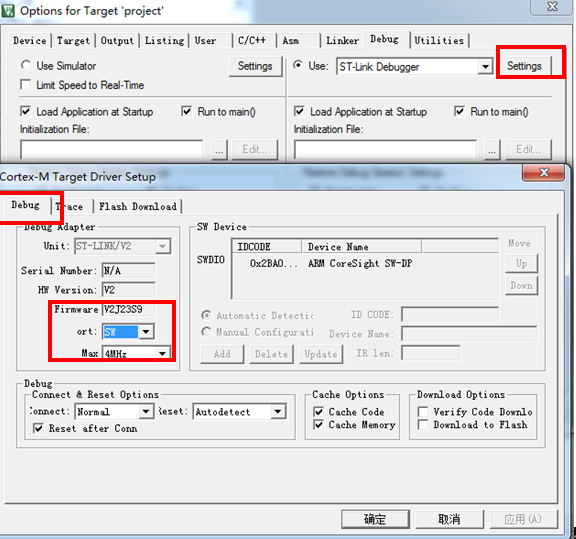
## （二）程序下载

STM32F4的程序下载有多种方法：USB、串口、JTAG、SWD等。我们这里采用板子自带ST-LINK下载与调试。

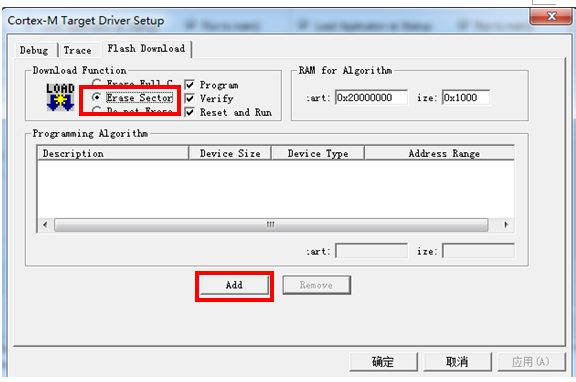
1. 点击魔术棒，选择Debug选项卡。在use的位子下来选择ST-Link Debugger。

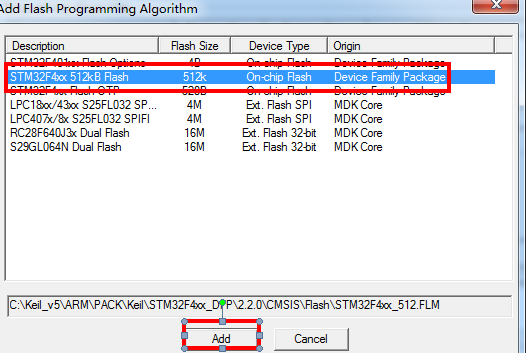


1. 然后点击后面的Settings，在跳出的Cortex-M Target Driver Setup中，选择Debug，然后在ort位置下拉选择SW。如下图所示。

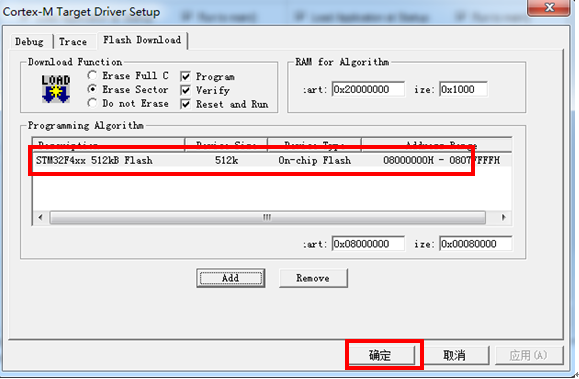


1. 继续在Cortex-M Target Driver Setup中，选择Flash Download选项卡。



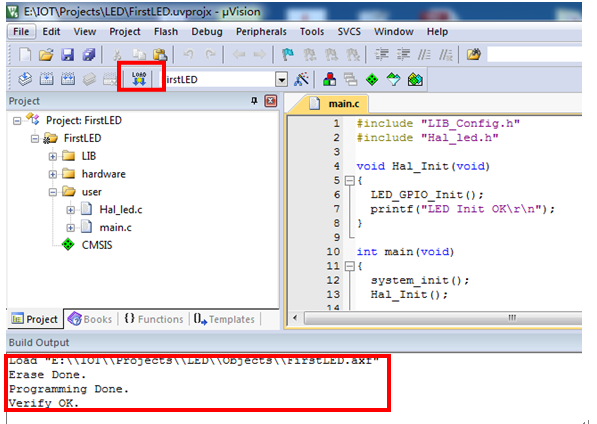


添加成功：



至此，下载调试环境配置完毕。

1. 点击LOAD，将程序下载到开发板，程序在开发板运行。



LED2间隔闪烁500ms和100ms，主程序功能：

