新建工程文件夹



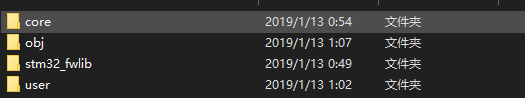
在工程文件夹中新建四个文件夹，分别存放工程文件

core存放

obj存放工程编译过程文件和结果文件

stm32\_fwlib存放

user存放

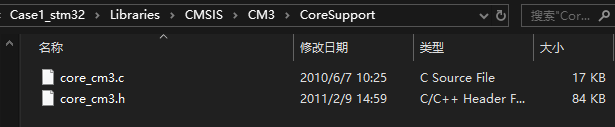


打开示例工程文件夹，Case1\_stm32\Libraries\STM32F10x\_StdPeriph\_Driver



这两个文件夹存放固件库版本的源文件和头文件，将这两个文件夹复制到新建的stm32\_fwlib中。

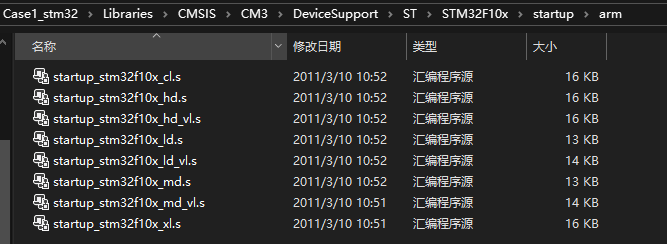
打开示例工程文件夹，Case1\_stm32\Libraries\CMSIS\CM3\CoreSupport



这两个文件是核心文件，将这两个文件复制到新建的core文件夹中。

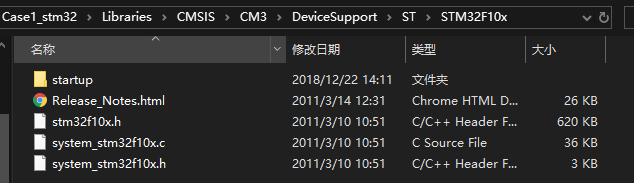
打开示例工程文件夹，

Case1\_stm32\Libraries\CMSIS\CM3\DeviceSupport\ST\STM32F10x\startup\arm



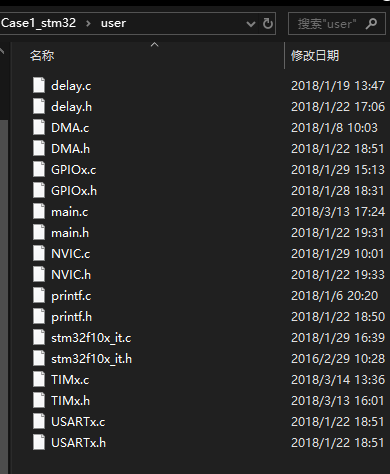
这些启动文件对应着不同容量开发板所选用不同的文件。将这些启动文件复制到新建的core文件夹中。

打开示例工程文件夹，Case1\_stm32\Libraries\CMSIS\CM3\DeviceSupport\ST\STM32F10x



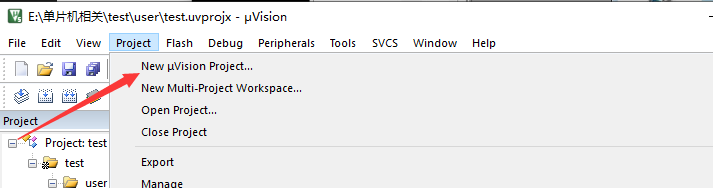
将下面的三个文件复制到新建的user中。

打开示例工程文件夹，Case1\_stm32\user

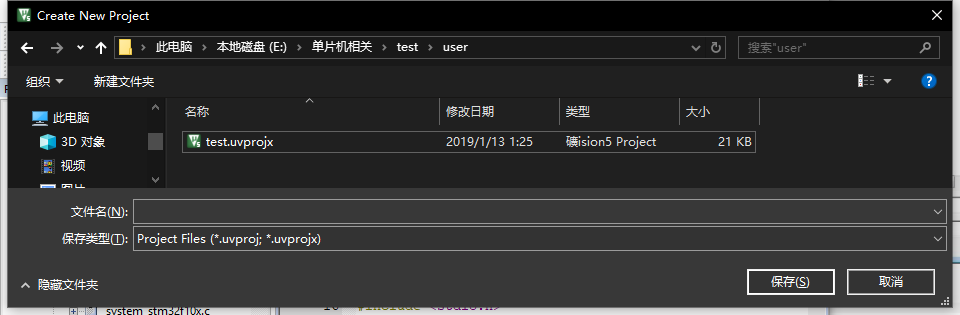


将目录下的文件全部复制到user中。

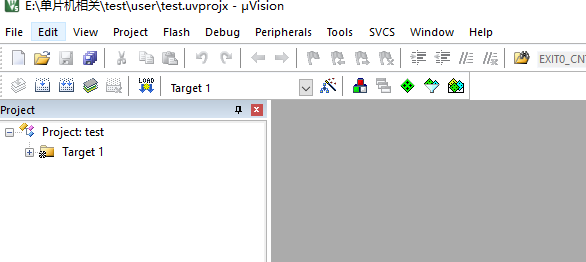
以上新建工程的文件就全部准备好了，开始新建工程。



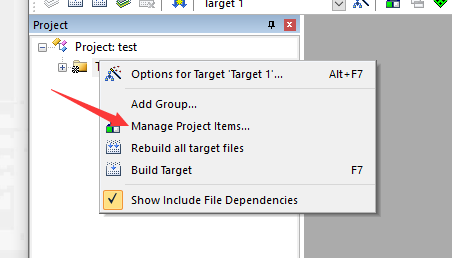
将工程目录定位到你新建的user文件夹



因为我之前新建过，所以已经有了一个工程文件，点击确定，下一步选择开发板对应芯片的型号，出现此界面点击关闭Cancel得到了一个空的工程。

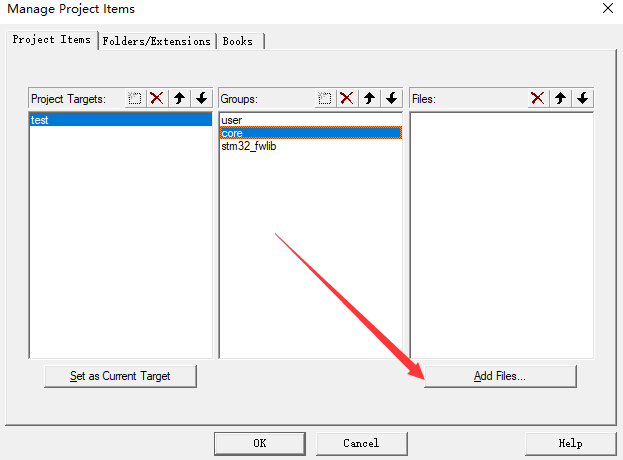


右击Target1，点击manage project ltems

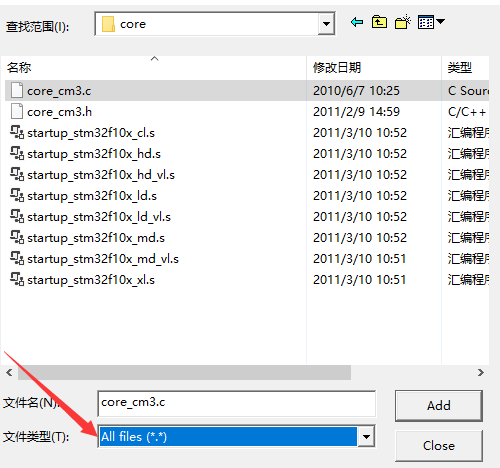


修改你喜欢的工程名称，并新建Groups组

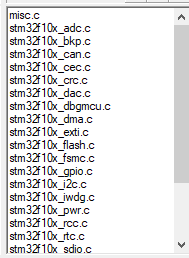
点击AddFiles在新建的相应文件夹中找到对应的文件



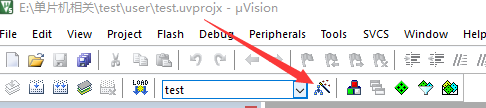
添加core文件夹内容时，记得选择所有文件，并添加core\_cm3.c和芯片型号对应的启动文件。



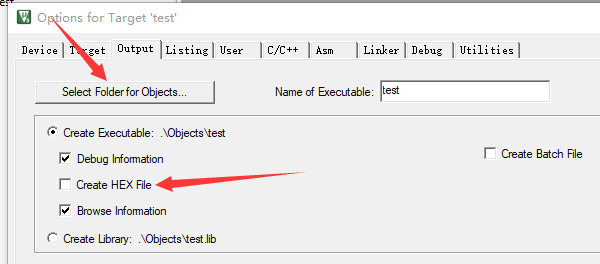
添加的stm32\_fwlib文件存在与src中。



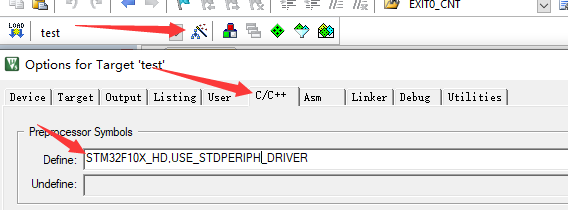
点击ok，回到默认界面，点击魔法棒



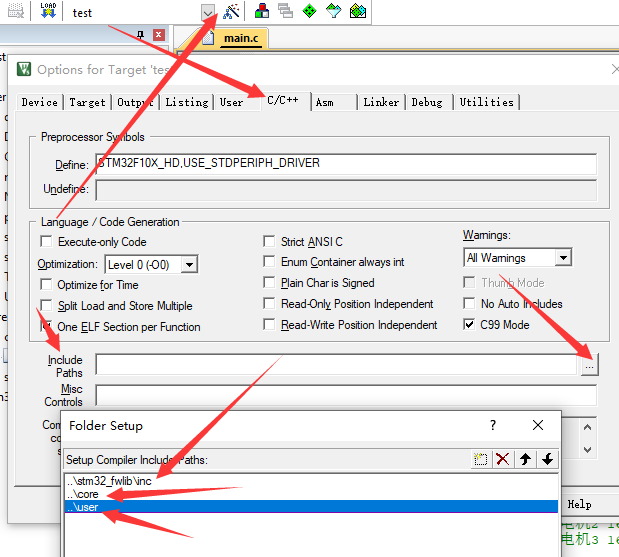
在output中设置工程编译文件存放的位置，选择刚新建的obj文件夹，并勾选生成hex文件。



第一次编译出错，问题在于芯片选择的宏定义，点击魔法棒，选到C/C++添加全局宏定义。STM32F10X\_HD,USE\_STDPERIPH\_DRIVER，前面的STM32F10X\_HD为你选择芯片型号对应的启动文件。



第二次编译出错是因为没有引用头文件，点击魔法棒， 选择include Paths后面的三个点，选择头文件所在的文件夹，需要注意的是，软件扫描头文件只扫描一级目录。点击确定添加



再次编译，成功。