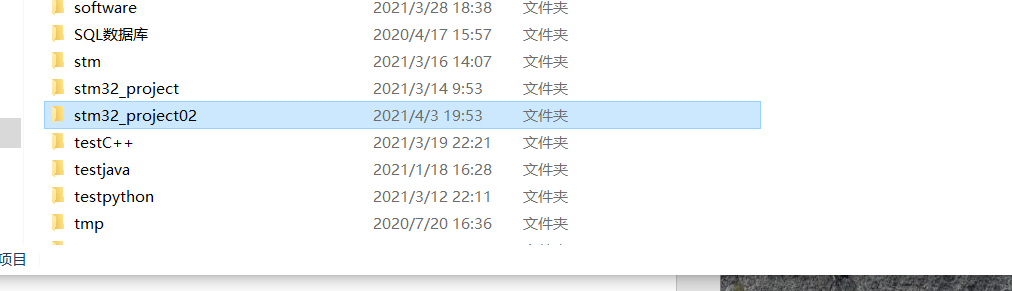
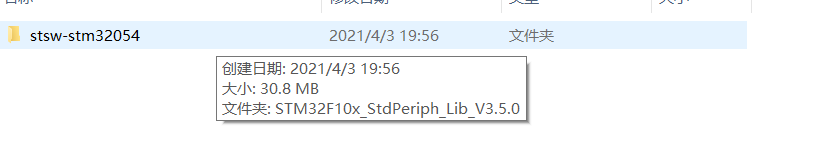
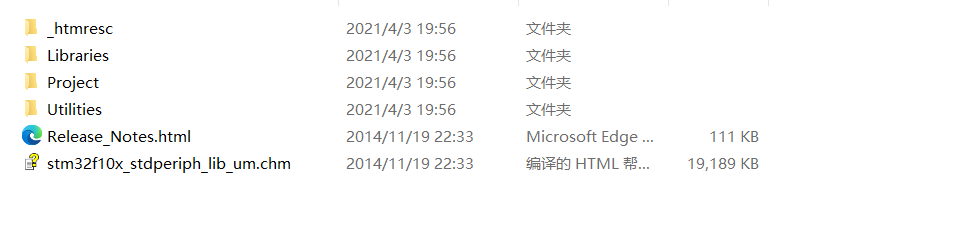
Keilu工程项目的建立

先建立一个文件夹

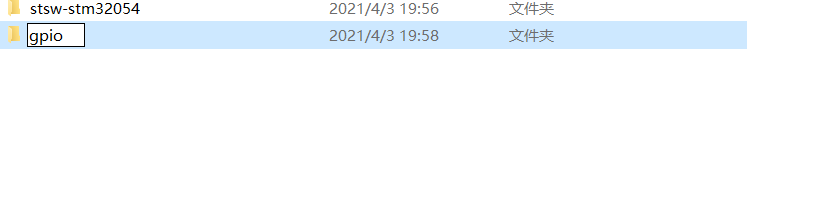


然后需要库

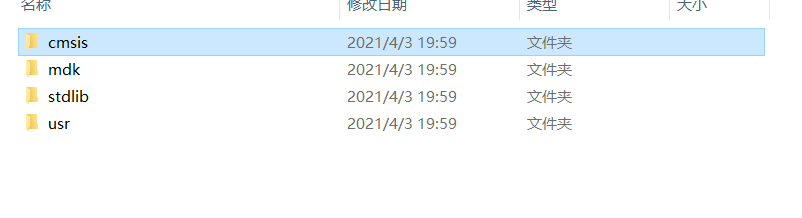




然后新建一个gpio的历程

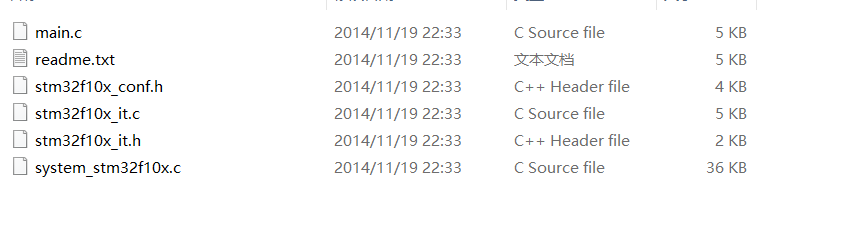


然后在里面新建四个文件夹



在下面路径下

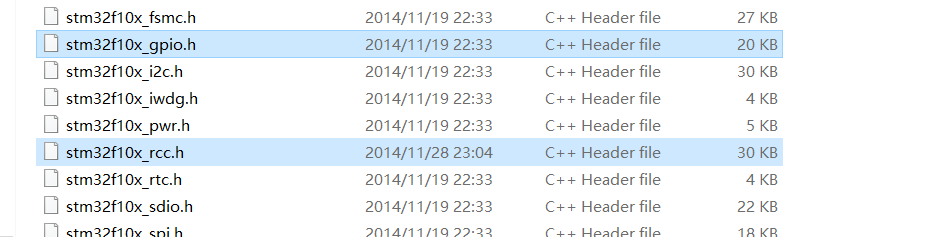
D:\stm32\_project02\stsw-stm32054\STM32F10x\_StdPeriph\_Lib\_V3.5.0\Project\STM32F10x\_StdPeriph\_Examples\GPIO\IOToggle



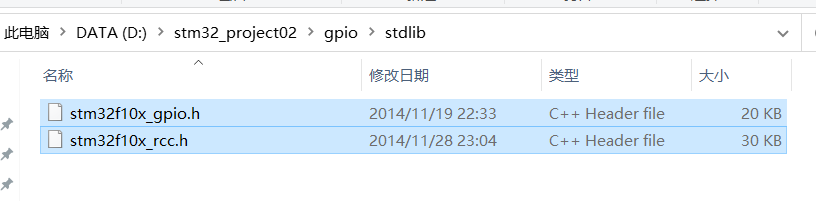
拷贝到



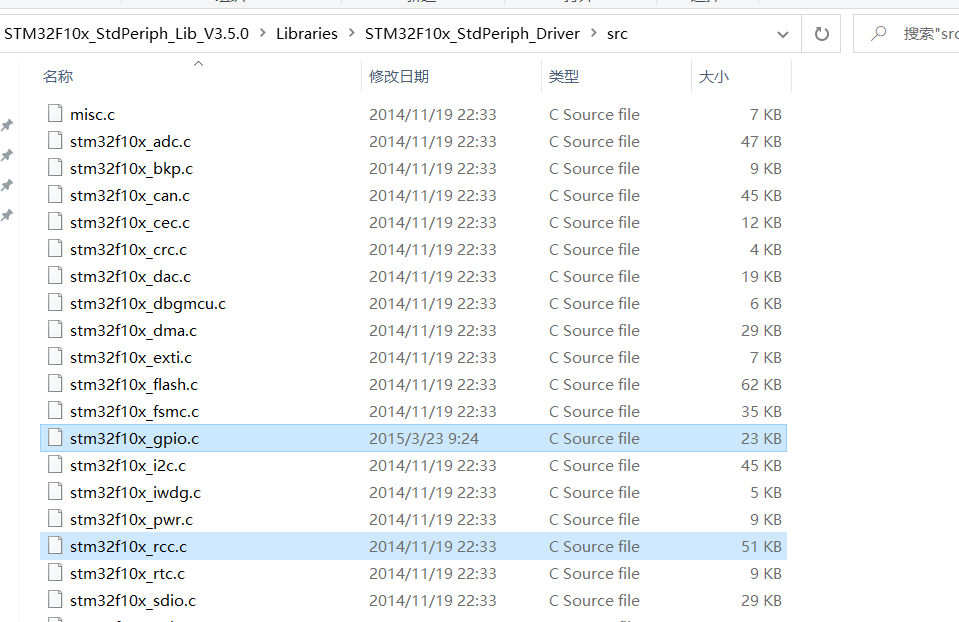
选中这两个



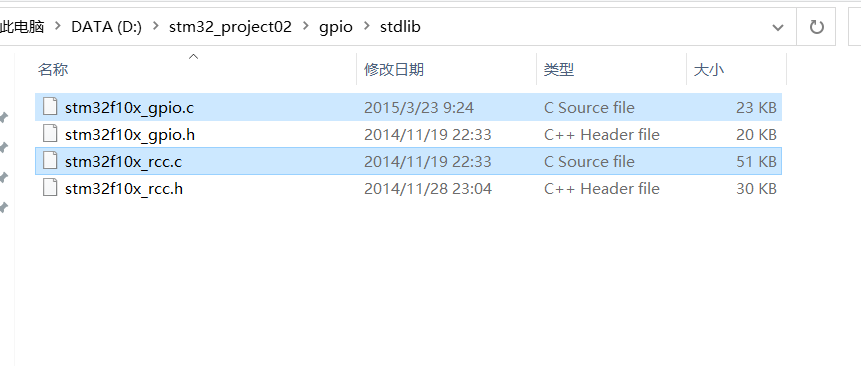
拷贝到



将下面这两个拷贝到

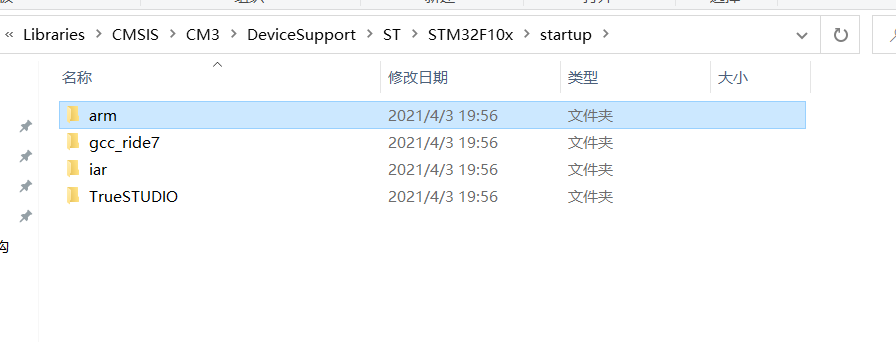


拷贝到

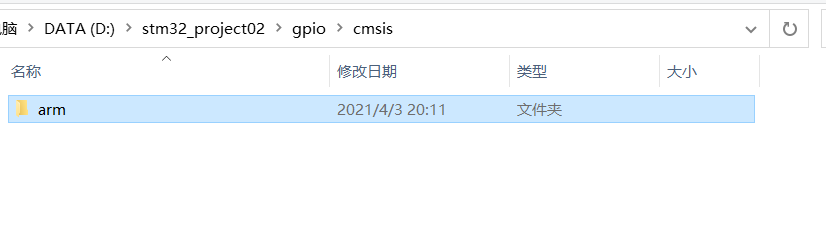


D:\stm32\_project02\stsw-stm32054\STM32F10x\_StdPeriph\_Lib\_V3.5.0\Libraries\CMSIS\CM3\DeviceSupport\ST\STM32F10x\startup

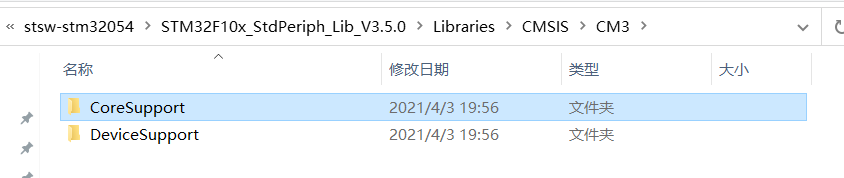
找到arm



拷贝到



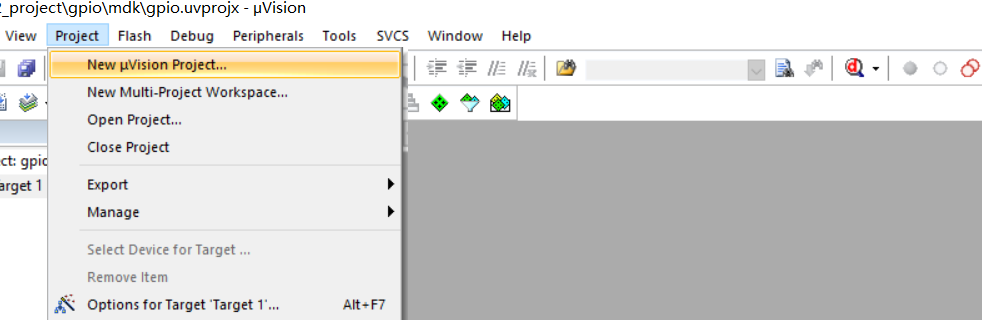
把下面这个



拷贝到

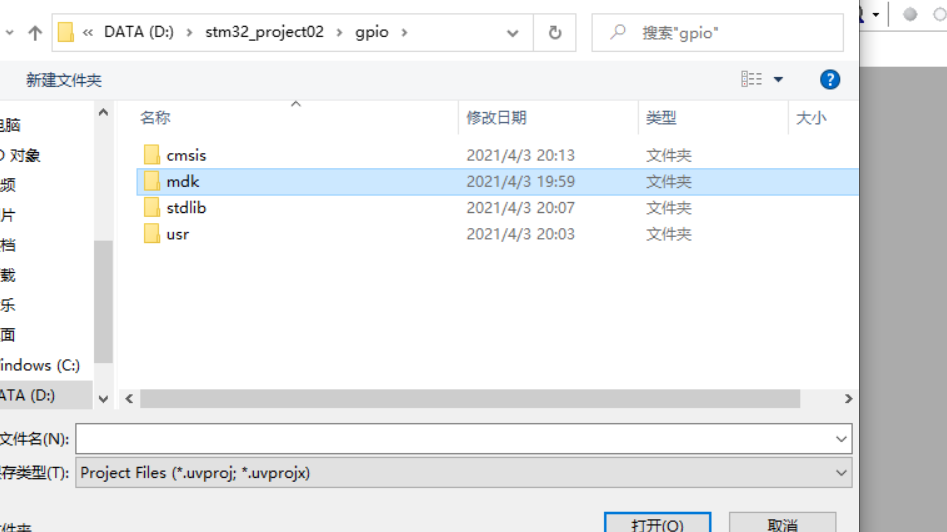


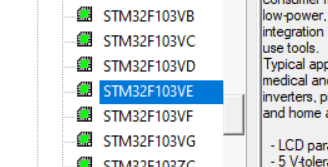
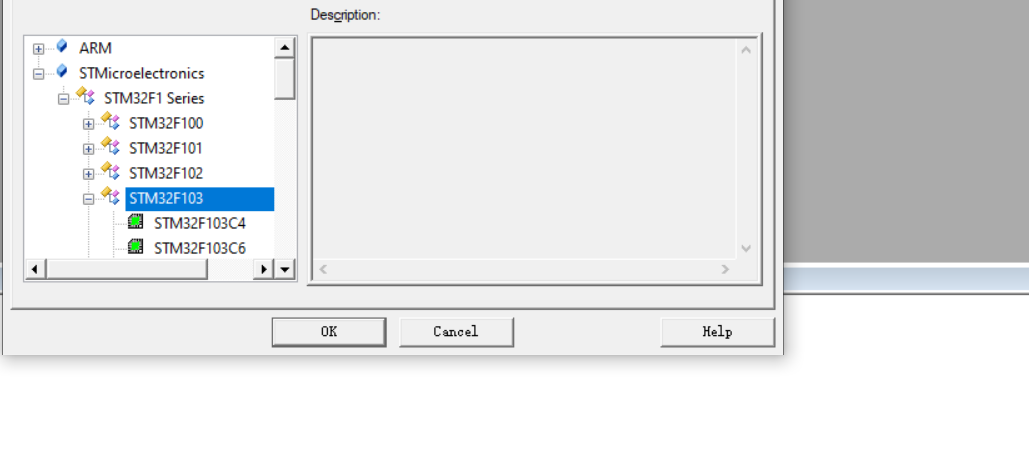
下面开始建工程



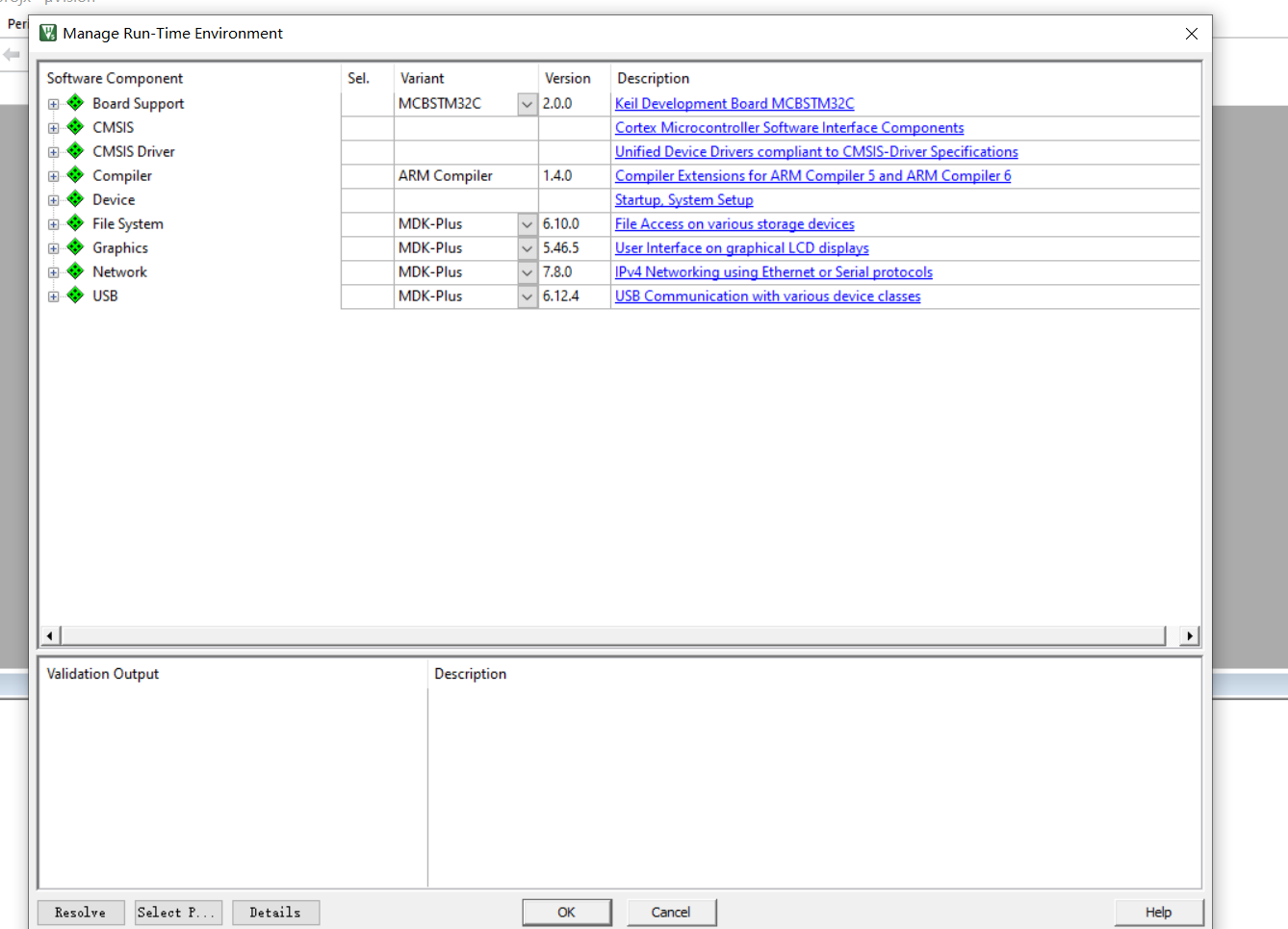
选择mdk

输入工程名

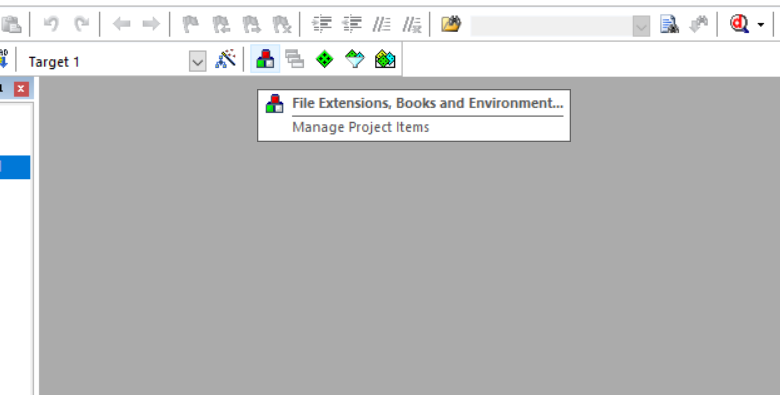


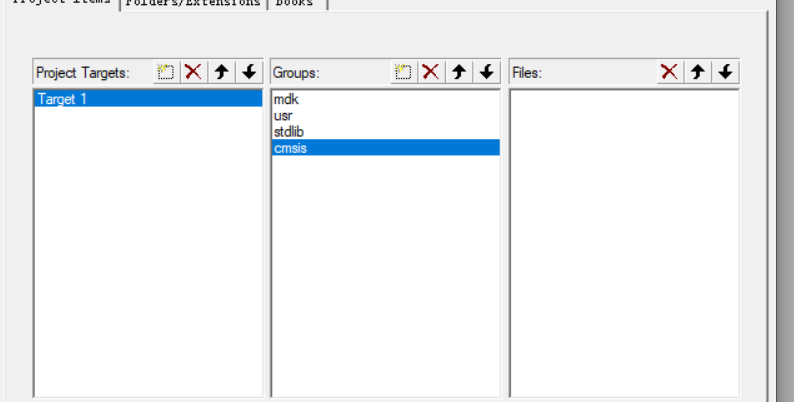


然后点×就好了

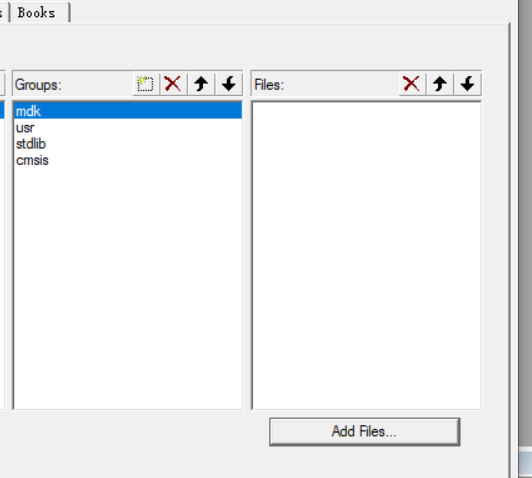


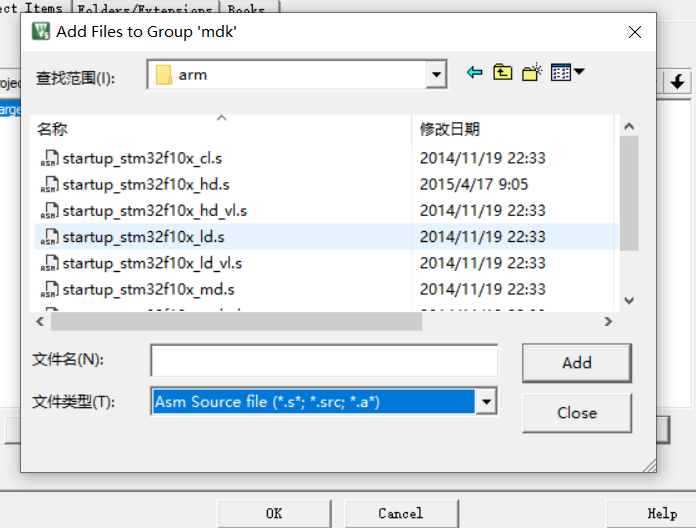
点下面的图标，就是文件环境配置

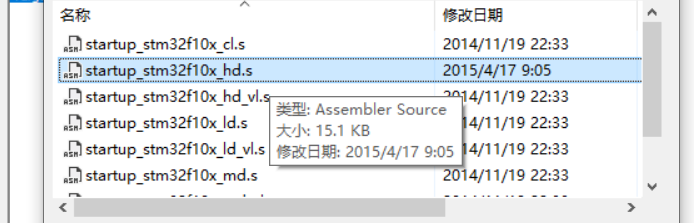




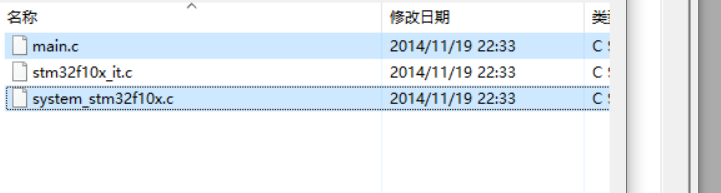
点击Add Files



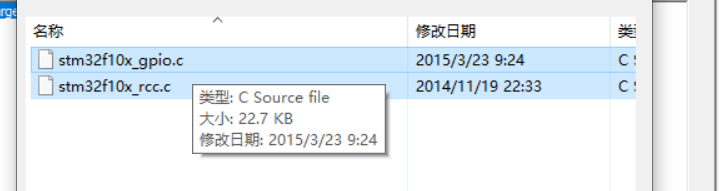




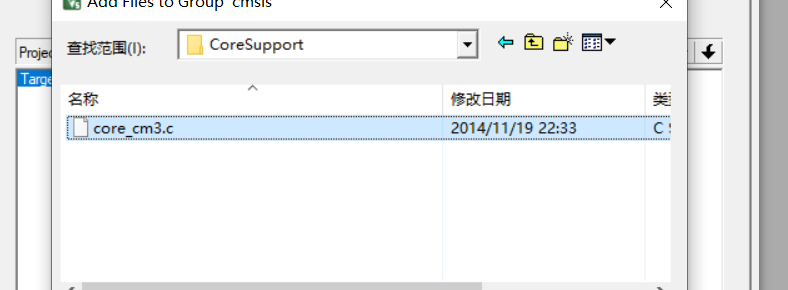
在usr中添加



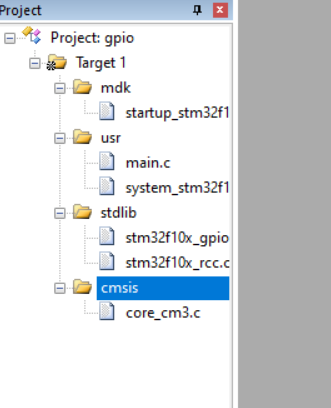
在stdlib中加



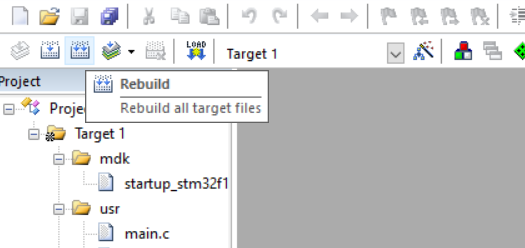
在cmsis中添加



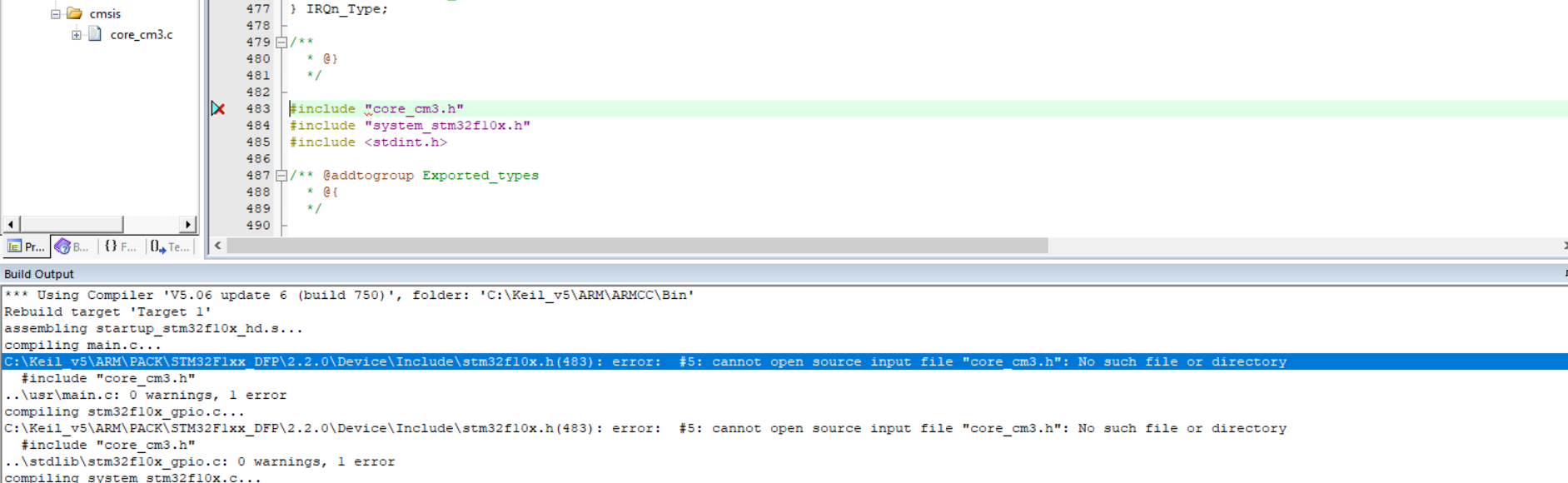
这是所有添加完了之后的



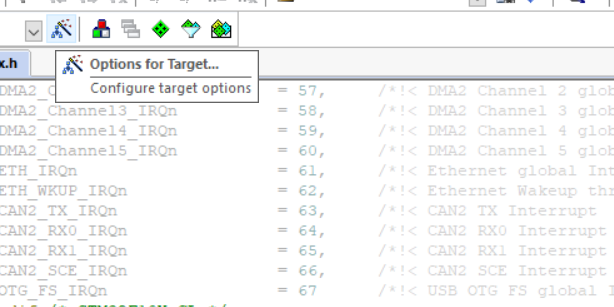
点击编译

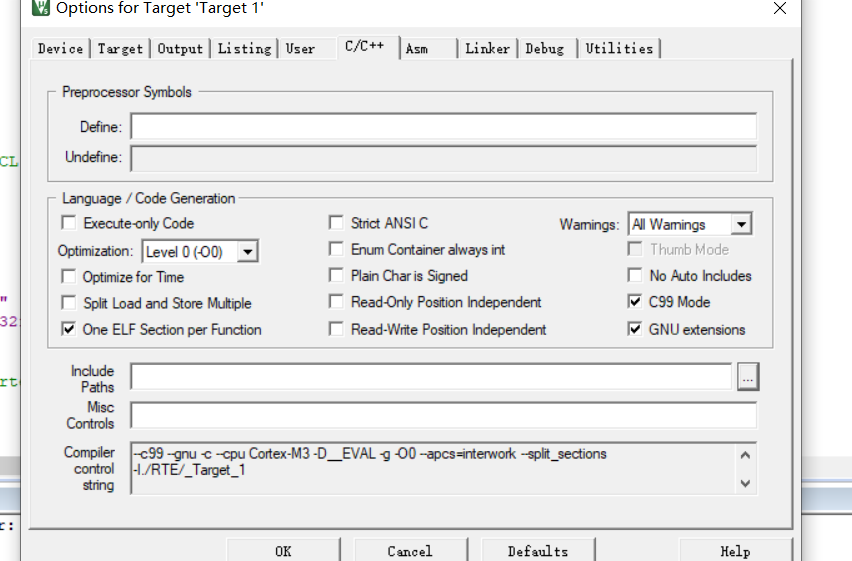


有错误，慢慢消除

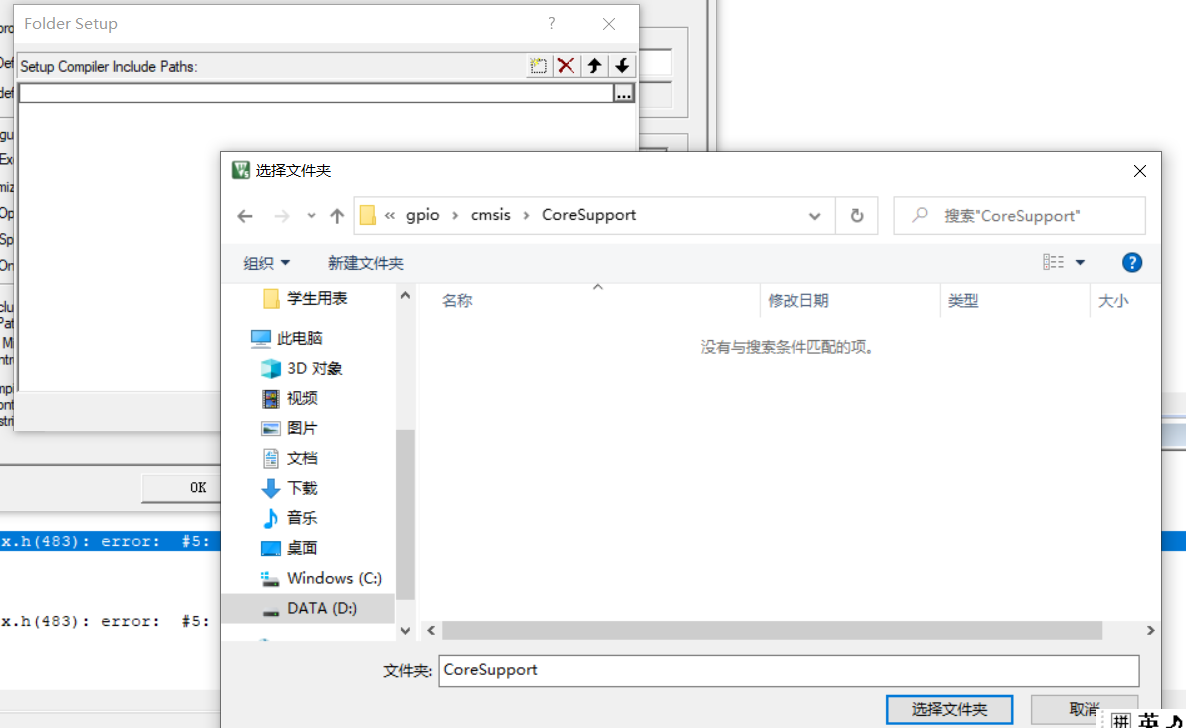


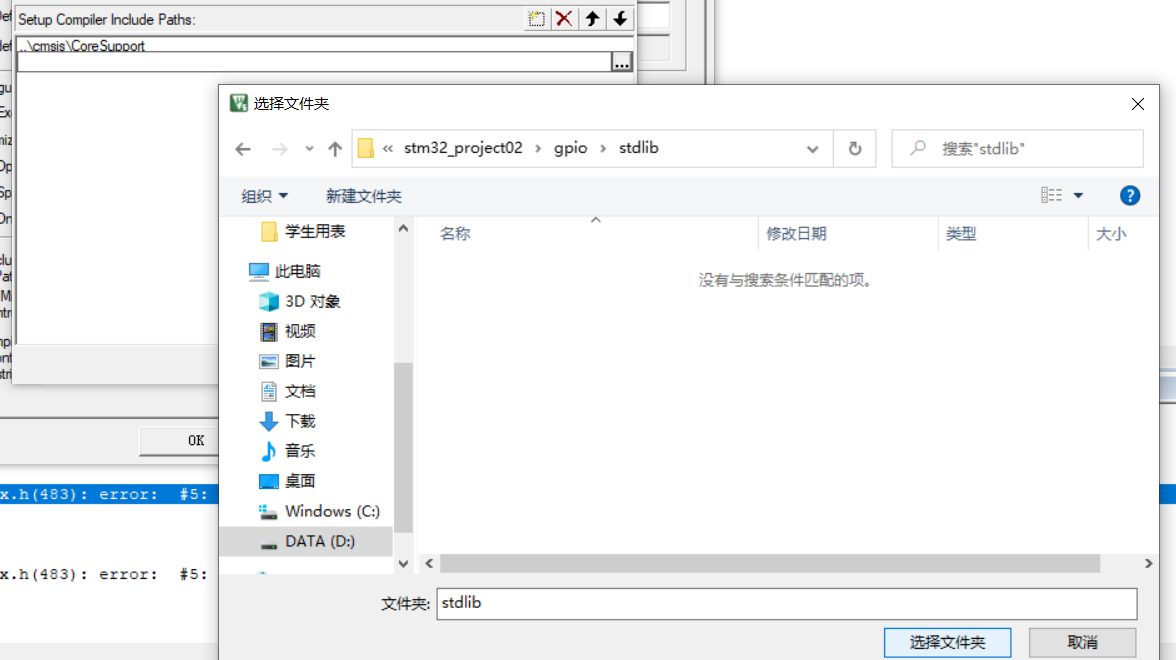
点击魔法棒

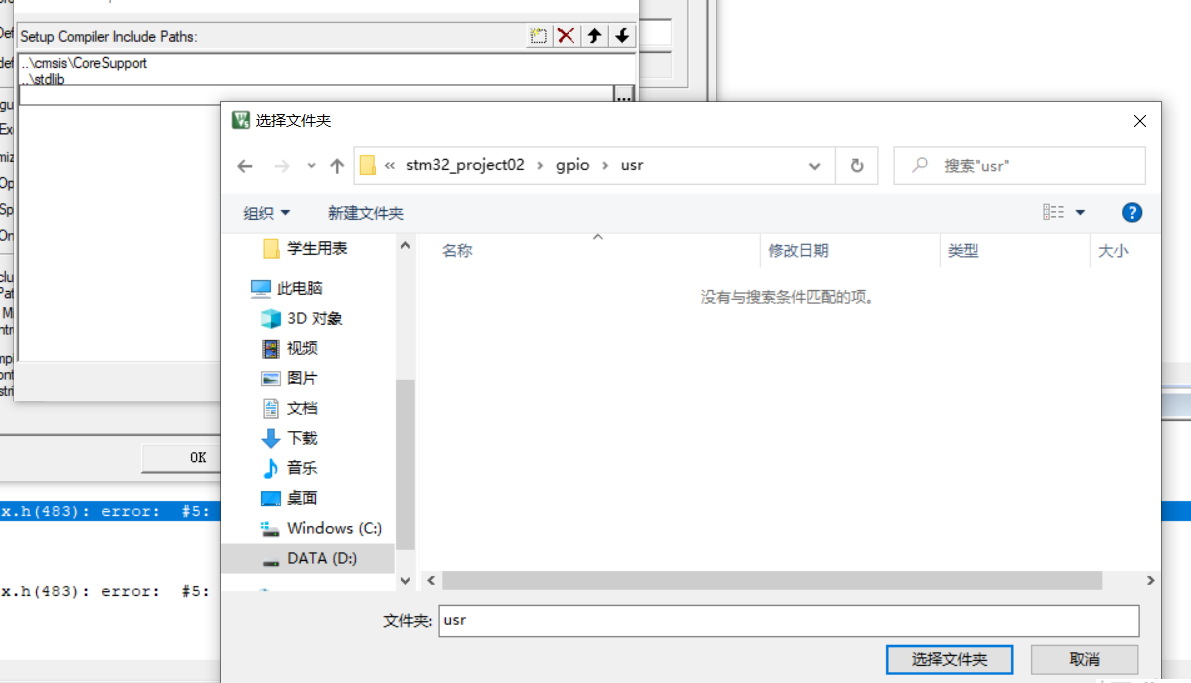




点击Include paths

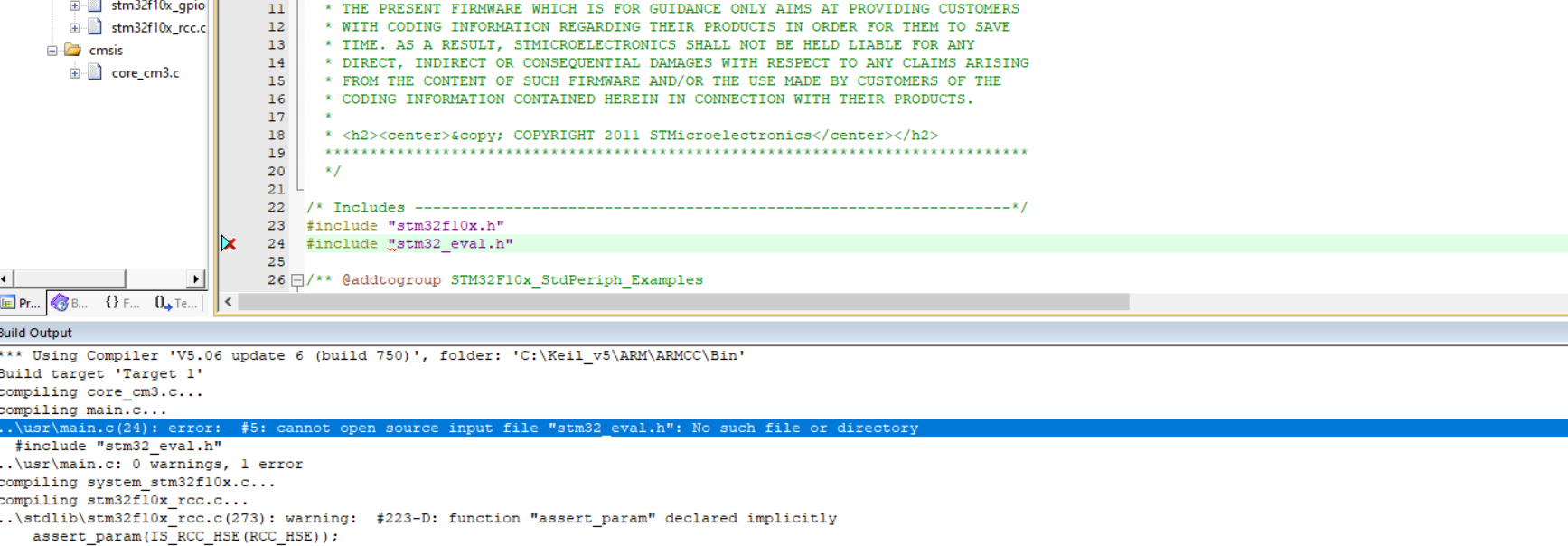




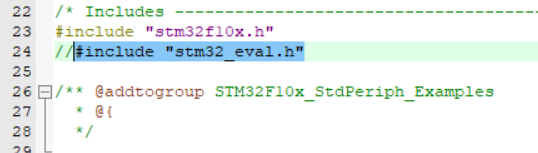


一路OK

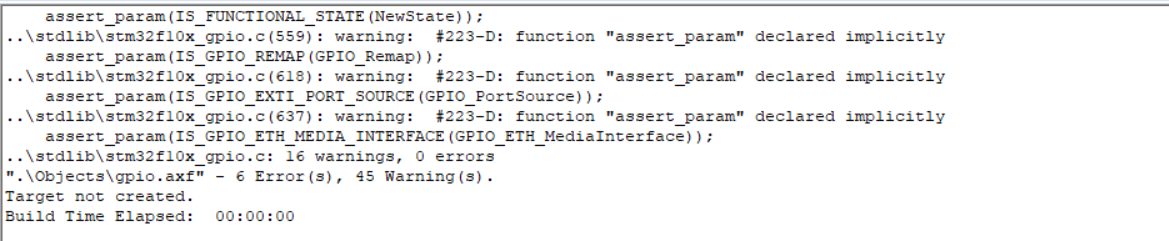
再次编译



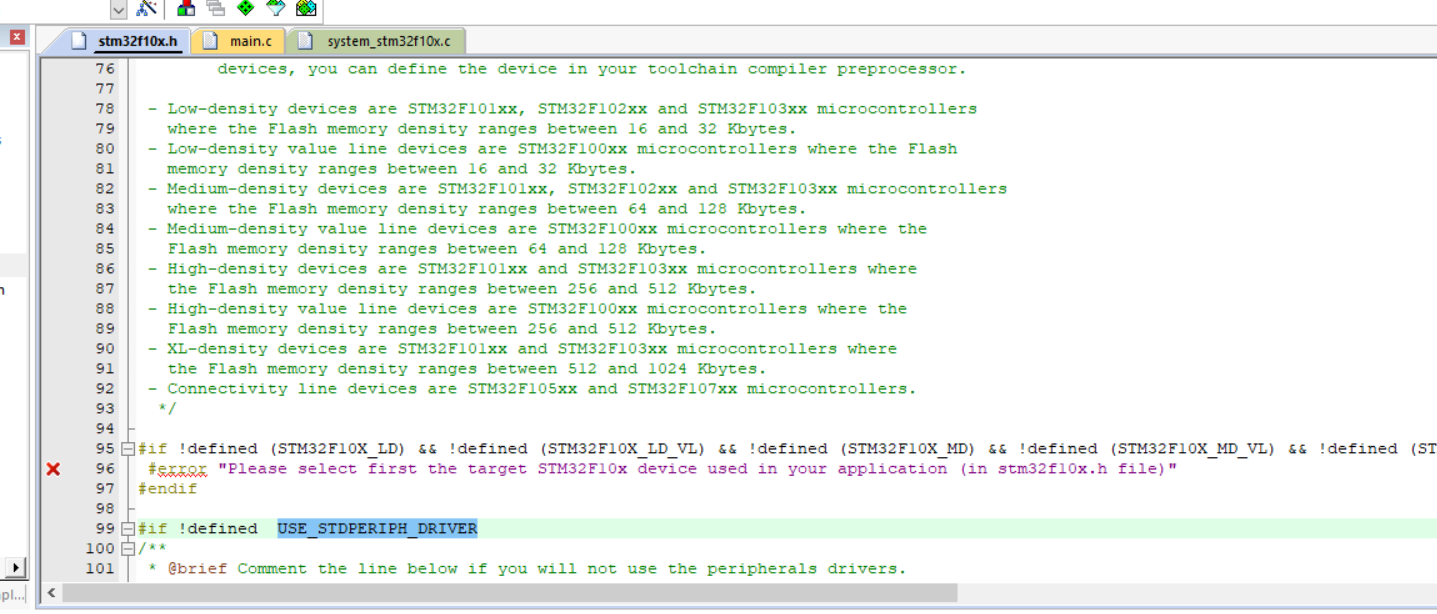
注销掉



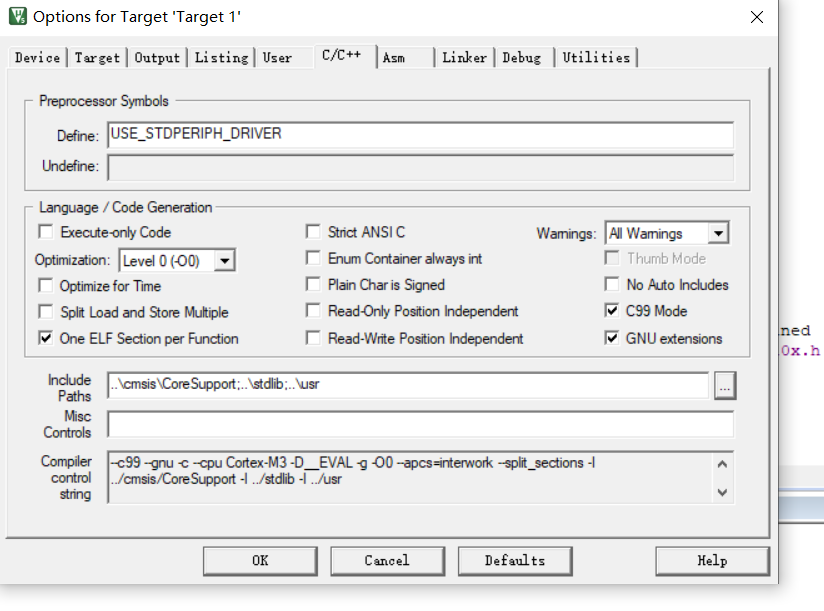
再编译还有错误



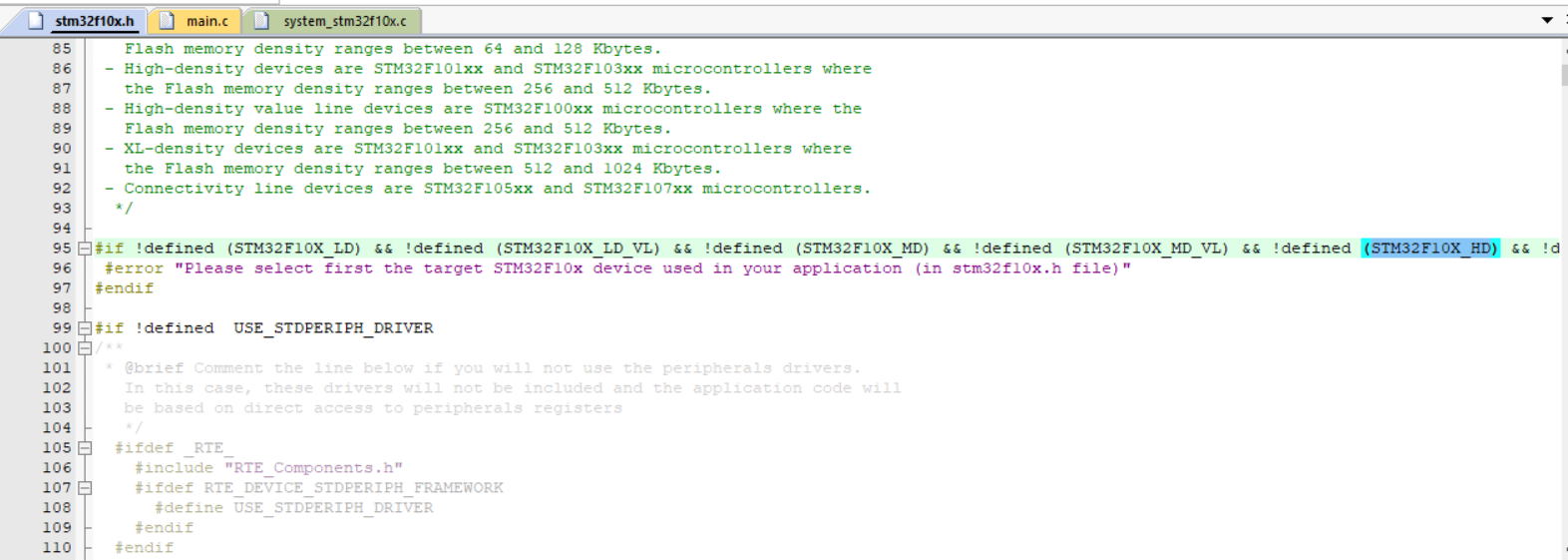
拷贝下面高亮

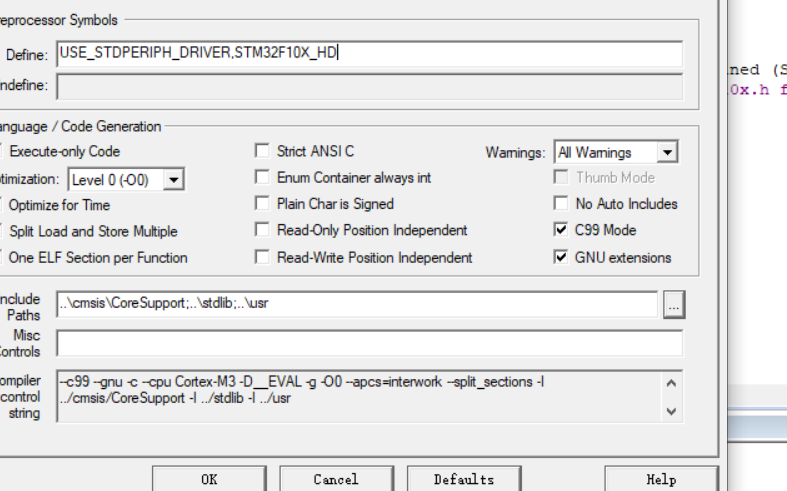


放入如图

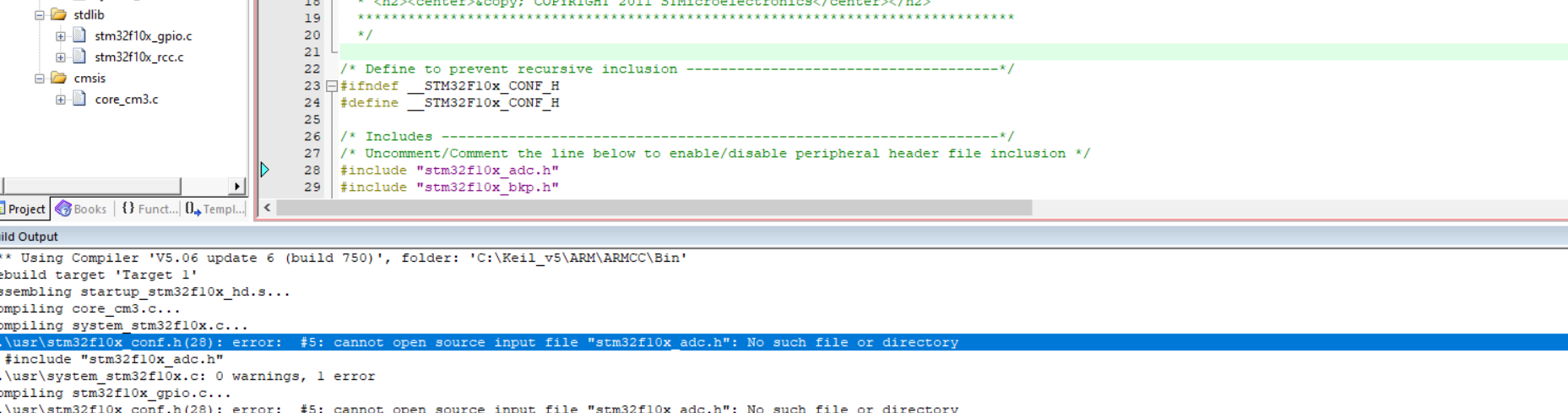


继续拷贝高亮

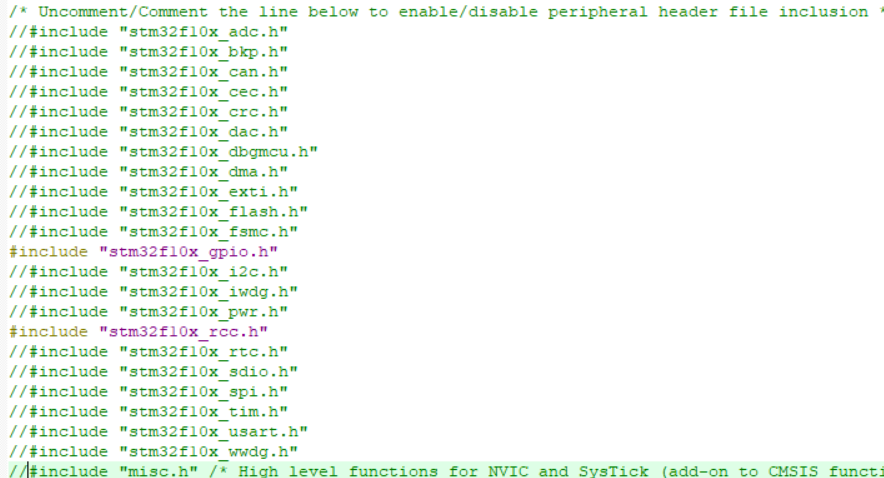




再编译。还有错

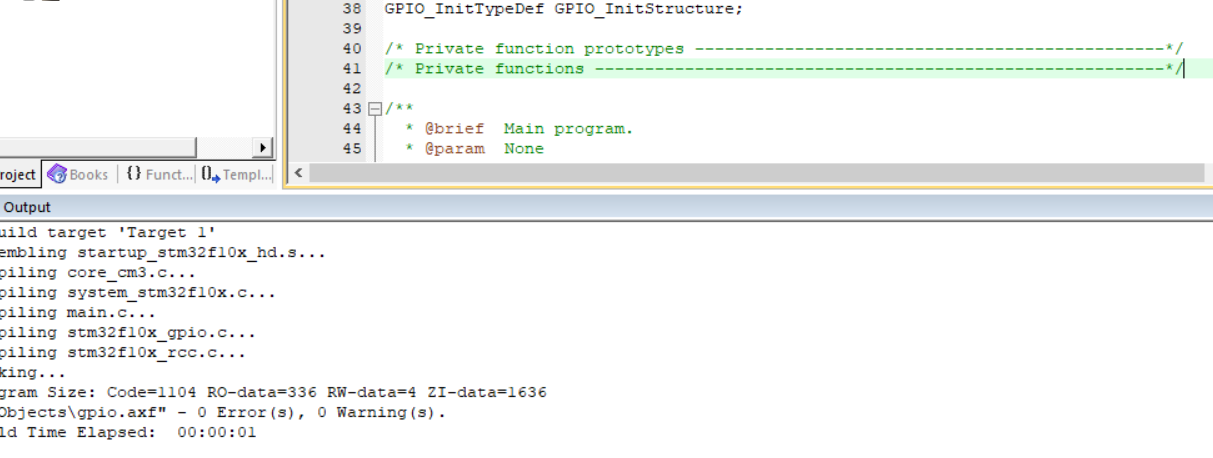


只保留gpio 和rcc,其他注销掉



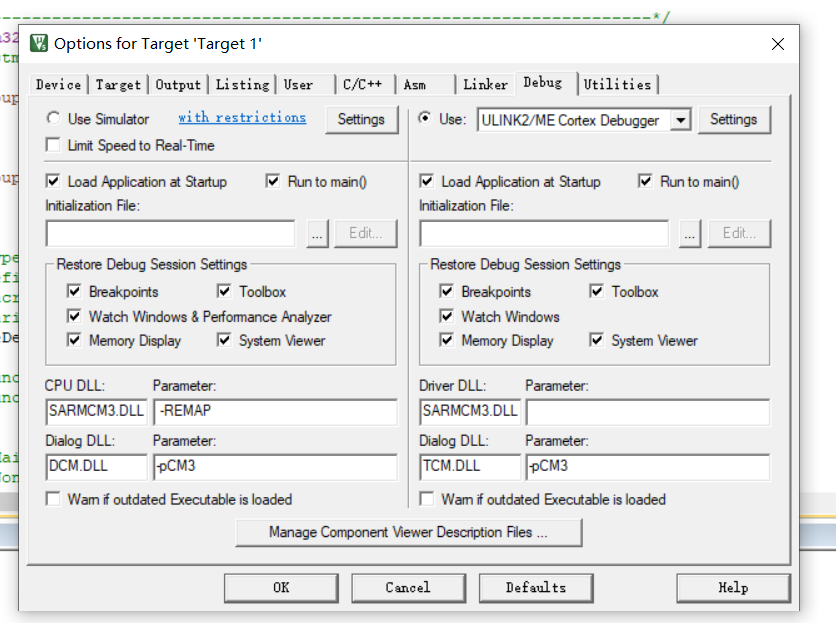
再编译

就没有错误了

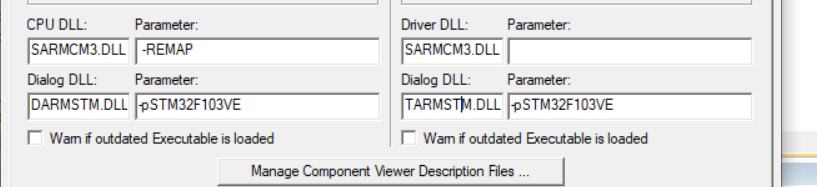


下来我们软仿真，看下效果

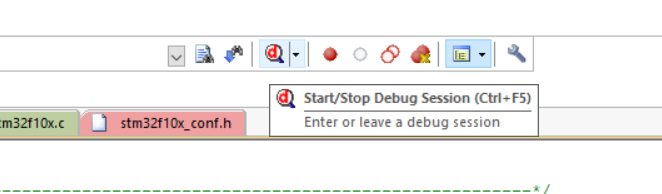
选择Debug



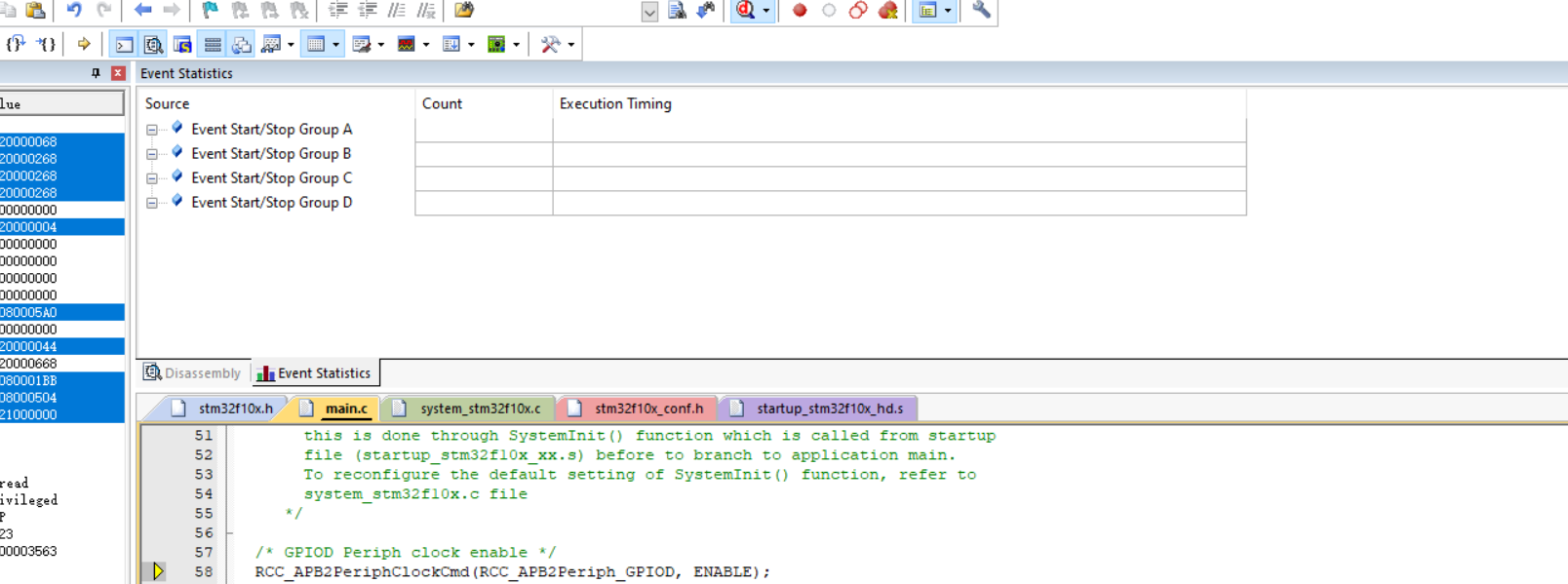
改变下面的值



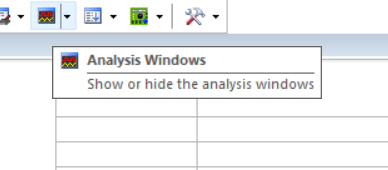
点击下面的图标



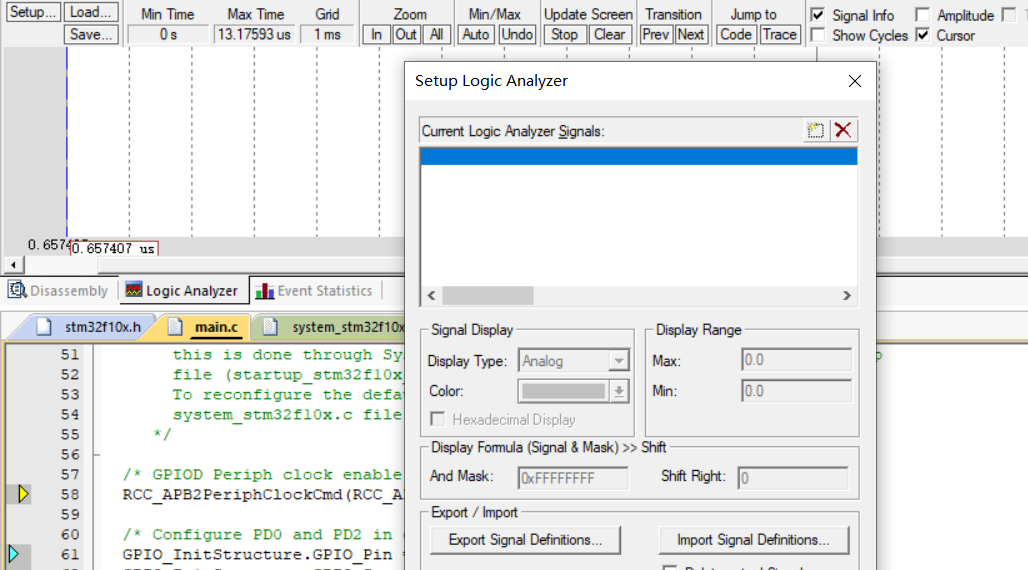
到下面界面



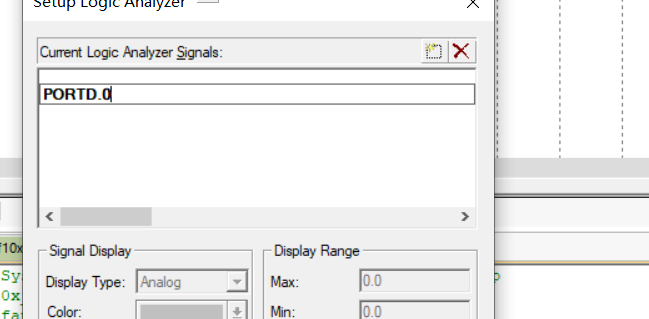
点击这个图标

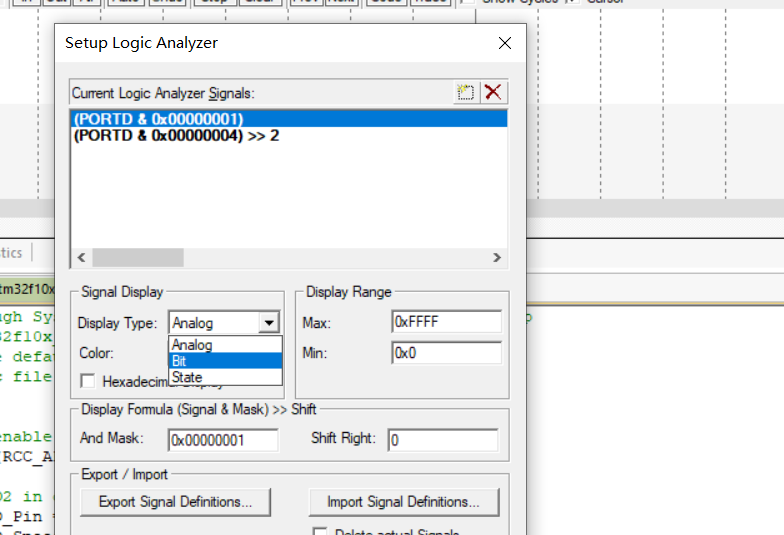


点击Setup



添加一个管口





然后点击zoom in放大

