# 工程建立说明(配置工程)

说明: 这是一个基于 F103 建立一个标准库函数版本的工程模板的完整过程。

• 空白工程建立完成后,可以作为一个工程模板,之后写新工程时可以在这个基础上作一些配置上的微调,然后直接开始代码的编写。不必每次都如此麻烦。

## 建立一个空白工程

## 6.配置下载器选项、包含路径和公共宏

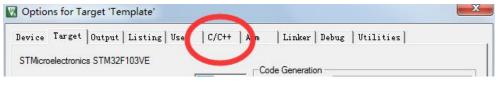
在工具栏找到魔术棒按钮,点击进入选项卡



#### (1)、配置包含路径

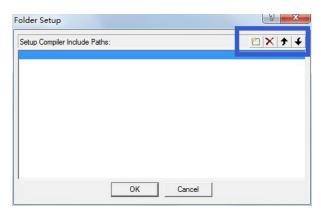
包含路径,就是告诉编译器编译时源文件所包含的头文件所在的位置。按照如下操作:

• 点击图示选项卡

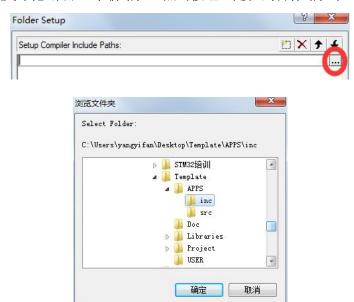


• 点击图示按钮

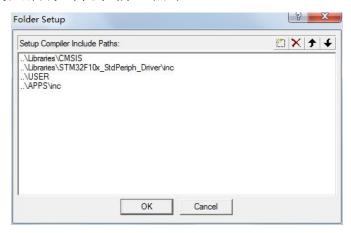
10	Output   Listing   Us	er C/C++ Asm Linker Debug	Utilities	
Preprocessor 9	Symbols			
Define:				
Undefine:				
	de Generation	22	Warnings:	
Execute-only Code		Strict ANSI C		1
Optimization: Level 0 (-00) ▼  Optimize for Time  Split Load and Store Multiple		Enum Container always int	All Warnings	
		Plain Char is Signed	☐ Thumb Mode	
		Read-Only Position Independent	No Auto Includes	
One ELF S	ection per Function	Read-Write Position Independent	C99 Mode	
Include				<b>f</b>
Paths Misc Controls				C
	cpu Cortex-M3 -g -00a C:\Users\yangyifan\Deskt	pcs=interworksplit_sections op\Template\RTE		^ +



- 这几个功能按钮与上面工程结构配置面板中的功能相同
- •【举例】添加一个新路径,点击按钮,定位到目标路径下:



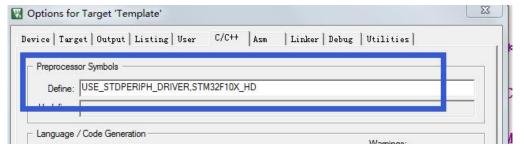
• 最终添加如下几个路径,点击 OK:



### (2)、配置公共宏

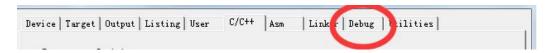
所谓公共宏,是编译器添加在每一个源文件中的宏,是对整个工程声明,我 们为了使用标准库,需要在此处声明一下。同时,还要声明我们使用的芯片型号,

#### 这样一些编译选项才可以正常地被配置。(上课的时候会讲解如何去查找这些宏)

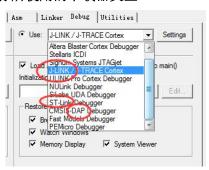


此处要声明的是两个: USE\_STDPERIPH\_DRIVER,STM32F10X\_HD 注意,二者之间要以英文逗号隔开,不然编译时会报许多个错(相当于每个源文件都出现了错误)。

- (3)、配置调试选项和下载选项
  - •调试选项,选中"Debug"选项卡



• 在这里选择你所使用的下载器类型



• 再点击旁边的 Settings 按钮,如果此时你的下载器连接着电脑且驱动安装完毕,则在如图框内出现你所使用的设备信息:



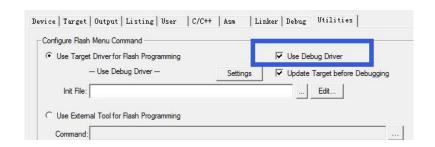
#### 如果不正常 (未识别设备), 会是这样的:

SN:	*		Error	Move
Device:		TDO	Cannot read JLink version number	Up
HW:	dll	TDI		Down
FW :   ort:	Max	6	TH CODE	

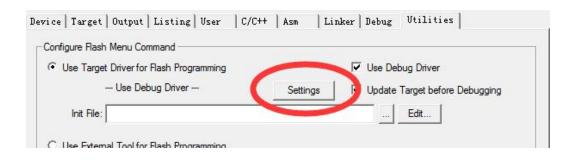
• 如果一切正常,配置下载选项。点击最后一个选项卡。



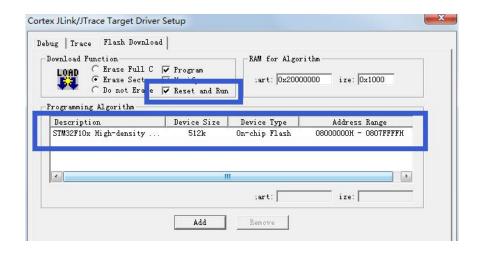
• 勾选图示选项(使用调试设备下载程序)



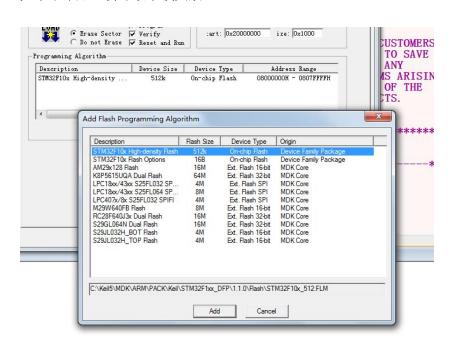
• 点击 "Settings"



• 注意下面框起来的这两个区域,上面那个需要打钩,(如果不勾选,那么下载程序后程序将不会自动运行,需要按下复位键才会运行)。下面那个地方是我们单片机 Flash 的规格。



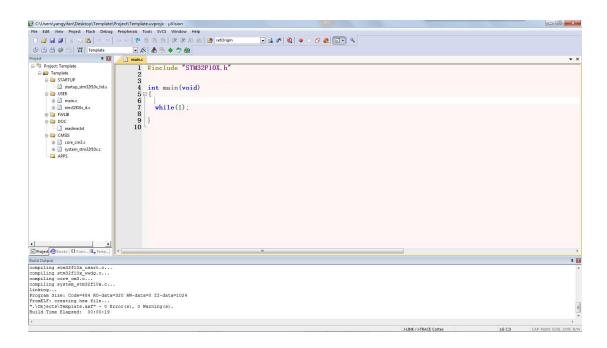
如果规格那里是空的,那么可以点击下面的"Add"按钮添加相应的型号即可(如果不添加,会导致下载失败)。



至此,我们所有的配置工作就全部结束啦!

把主函数中原有的的代码清除干净,加上头文件,写一个简单的 main 函数框架。(死循环是必须的!)

编译完成无误后,一个空白工程模板就全部完成啦!(不要在意为什么背景是粉色的...-\_-!!)



HIT-ROVER 团队 杨亦凡 2017.10.02