

在 MDK 中为 STM32 移植 FreeRTOS

/******

FreeRTOS 版本是 V7.0.1。STM32 本人用的是大容量的 stm32f103zgt6，故启动文件使用 startup_stm32f10x_hd.s。固件库版本 V3.5.0。编译环境用的是 MDK Version:4.22。

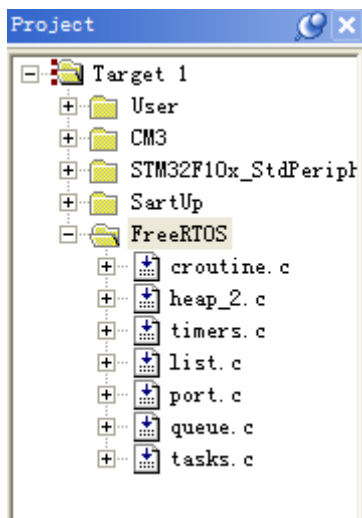
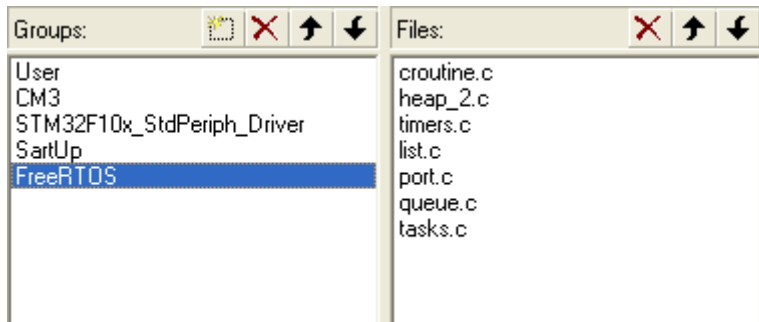
注：用户在移植 FreeRTOS 操作系统前最好先移植好 STM32 的固件库，调试好确保固件库能成功运行后，再进行操作系统的移植。

*****/

- 1、在 Project 里新建一个文件夹并命名为 FreeRTOS。复制 FreeRTOS 解压包里的 Source 文件夹里的 croutine.c, timers.c, list.c, queue.c, tasks.c 以及 Source->portable->RVDS->ARM_CM3->port.c 以及 Source->portable->MemMang->heap_2.c 到自建的 FreeRTOS 文件夹。

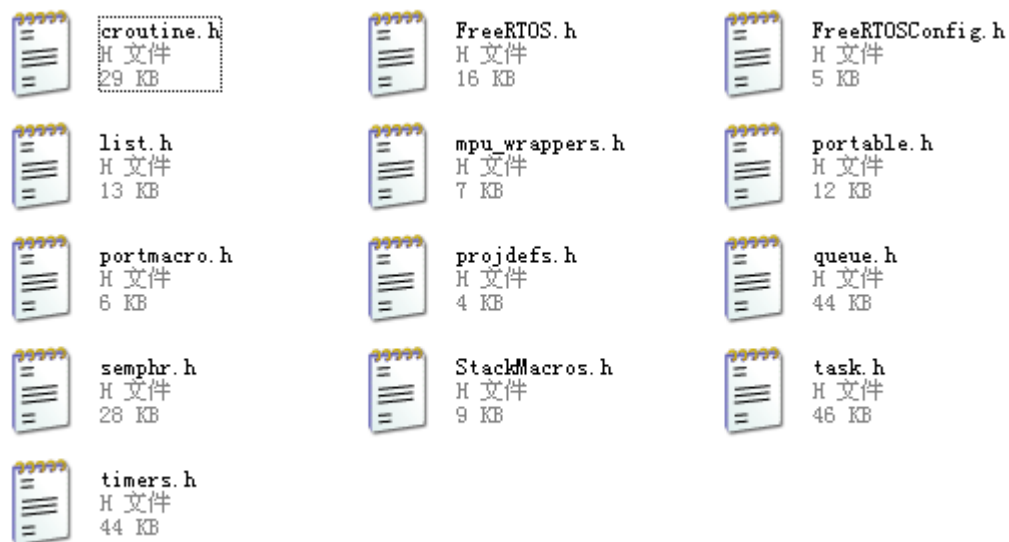
按  新建 FreeRTOS 组并添加相应文件。

如下图：

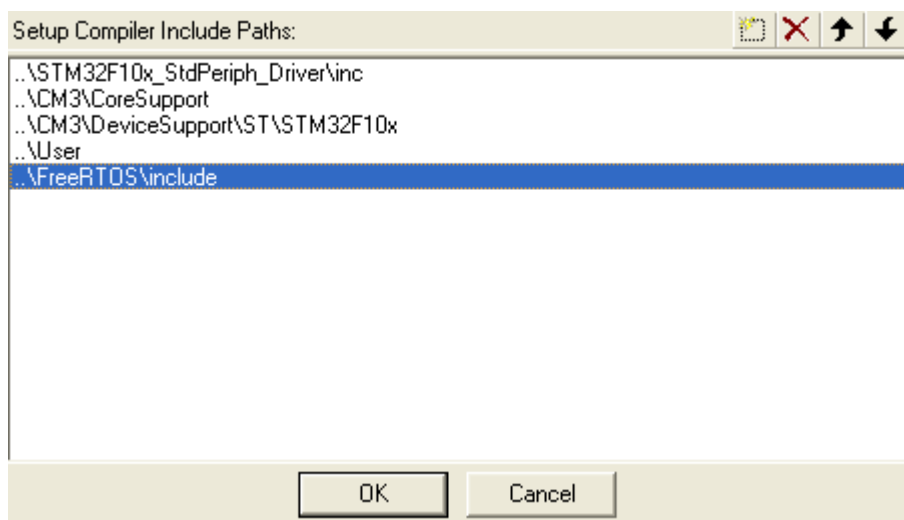


- 2、在 FreeRTOS 文件夹里再新建一个文件夹 include。在 include 里面加入如下文件：① FreeRTOS 解压包里的 Source 文件夹下的 include 文件夹中的所有 .h 文件；② Source->portable->RVDS->ARM_CM3->portmacro.h；③ FreeRTOS 解压缩包里的 Demo 文件夹下找到 CORTEX_STM32F103_Keil 文件夹(用户根据自己的开发平台选择相应文件夹)，找到 FreeRTOSConfig.h 文件加入到 include 中。

Include 文件夹里所有 .h 文件如下：



- 3、菜单栏找到 Project->options for target->C/C++->Include Paths 中加入上述的 include 文件夹。如下图：



- 4、在 main 文件头部#include 以下头文件：FreeRTOS.h、task.h、queue.h、list.h、portable.h、freertosconfig.h

- 5、打开启动文件 startup_stm32f10x_hd.s（用户请根据所用芯片容量选择相应的启动文件）。在__heap_limit 下面添加：

```
PRESERVE8
THUMB
IMPORT xPortPendSVHandler
IMPORT xPortSysTickHandler
```

```
IMPORT vPortSVCHandler
```

找到 DCD SVC_Handler 将其屏蔽掉，改成 DCD vPortSVCHandler。找到 DCD PendSV_Handler 将其屏蔽掉，改成 DCD xPortPendSVHandler。找到 DCD SysTick_Handler 将其屏蔽掉，改成 DCD xPortSysTickHandler。

至此 FreeRTOS 移植完毕。在 main 中创建任务后启动任务调度器就可以运行了。除了启动文件上的少许改动外，其他程序均不需要用户自己去修改。因为制作 FreeRTOS 的工程师们已经为我们作了这些工作。

另外网上广泛流传着一篇由广州伟研电子公司写的《FreeRTOS 在 STM32 的移植》的文章，该文章中介绍到的所有程序其实都不需要用户自己去改的。另外该文章中提到一个叫 PORTASM.S 的以汇编写成的文件，在 stm32 的移植中我们不需要用到此文件，此文件的内容已经用 __asm 语句内嵌到了 port.c 文件中了。