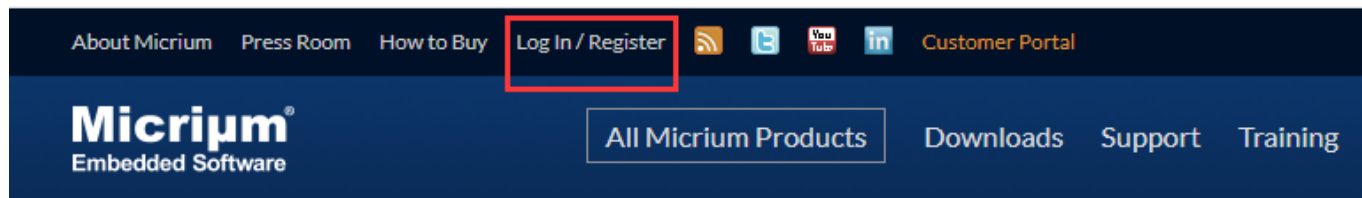


UCOS II 移植到 STM32F407

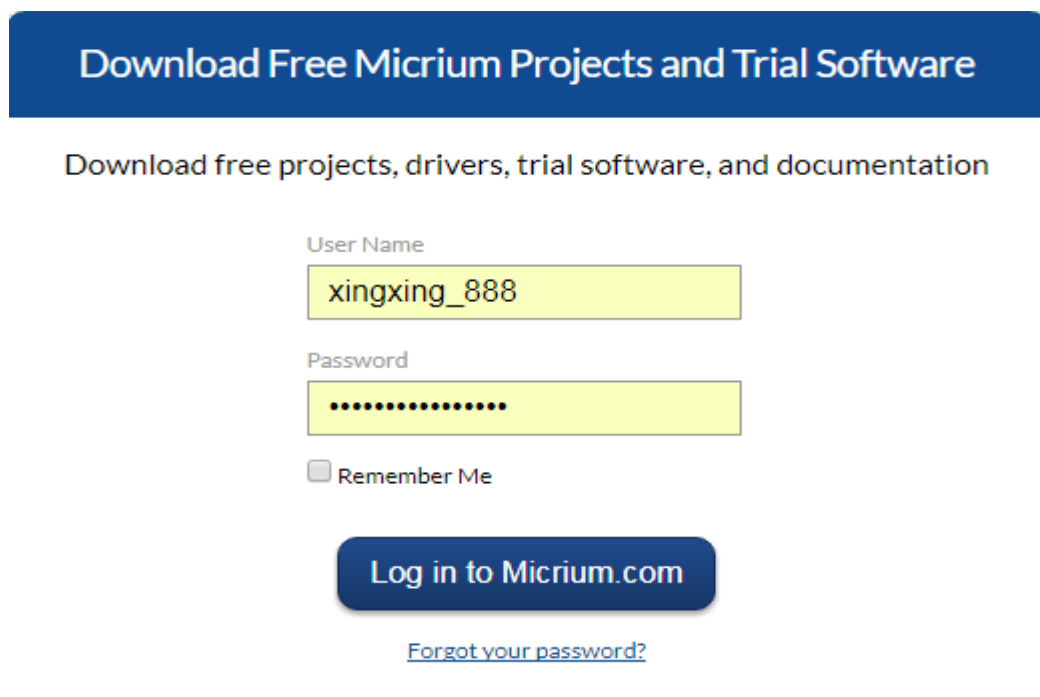
一。UCOS II 源码的获取

<https://www.silabs.com/developers/micrium>

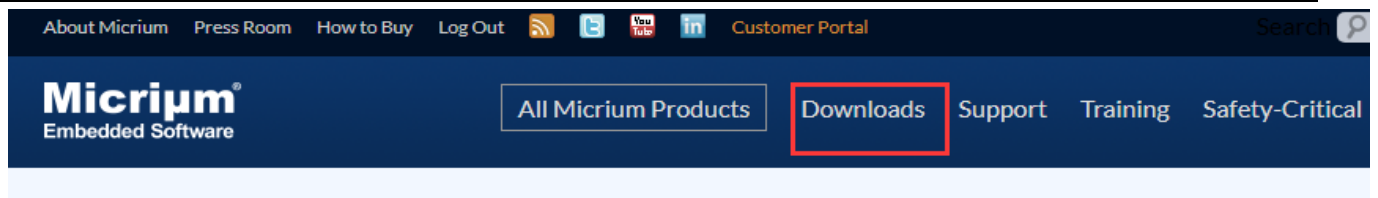
1. 登陆 μ C/OS-官方网站 <http://micrium.com/>，如下所示：



2. 点击网站上方的 Login/Registration，进行登陆或注册一个新帐号。如果没有帐号要先注册一个，注册过程不再描述。我已经有帐号，直接登陆。
3. 点击后转入登陆界面，网址是 <http://micrium.com/wp-login.php>，直接输入这个进行登陆也可以，界面如下：



4. 在这个界面中输入用户名和密码，点击 Log In,即可以进入，登陆后可以下载网站上的免费资源。



Welcome to Micrium

Logged In

You have successfully logged in to the Micrium web site.

You now have access to any file in the [Download Center](#) and you can post messages in the [Micrium Discussion Forums](#).

5. 点击上面的“Downloads Center”，进入下载页面。

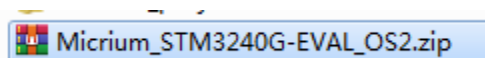


6. 点击下载，然后选择 uc0s ii 版本下载

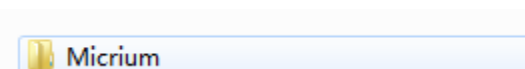
STMicroelectronics STM32F4xx	µC/OS-II µC/OS-II V2.92.11 µC/Probe V3.5.15.400 STM32CubeF4 Library V1.5.0	STM3240G-Eval	Atollic TrueSTUDIO V5.x IAR (EWARM) V7.x Keil MDK V5.x STM32CubeFx	2015/04/27
------------------------------	---	---------------	---	------------

7. 下载得到一个压缩包，并解压。

解压前：



解压后：



二。UCOS II 源码分析

1. 打开 Micrium 文件

Examples	2015/4/27 14:00	文件夹
Software	2015/4/27 14:00	文件夹
README_STM3240G-EVAL_OS2.pdf	2015/4/27 14:11	Foxit Reader Plu... 373 KB

Examples: 官方提供的例程

Software: 官方提供的源码

2. 打开 Software

uC-CPU	2015/4/27 14:00	文件夹
uC-LIB	2015/4/27 14:00	文件夹
uCOS-II	2015/4/27 14:00	文件夹
uC-Serial	2015/4/27 14:00	文件夹

μC-CPU: 这是和 CPU 紧密相关的文件, 我们不需要使用。

μC-LIB: Micrium 公司提供的官方库文件, 如字符串操作、内存操作等函数接口, 可用可不用。

μC-COS-II: 这是关键目录文件, 我们接下来要详细分析的文件, 跟移植、使用密切相关的。

μC-Serial: μC/COS-II 接口文件文件。

3. 打开 uCOS-LIB

Ports	2015/4/27 14:00	文件夹	
lib_ascii.c	2014/3/13 10:57	C Source file	24 KB
lib_ascii.h	2014/3/13 10:57	C++ Header file	48 KB
lib_def.h	2014/3/13 10:57	C++ Header file	62 KB
lib_math.c	2014/3/13 10:57	C Source file	12 KB
lib_math.h	2014/3/13 10:57	C++ Header file	14 KB
lib_mem.c	2014/3/13 10:57	C Source file	119 KB
lib_mem.h	2014/5/15 14:10	C++ Header file	85 KB
lib_str.c	2014/3/13 10:57	C Source file	201 KB
lib_str.h	2014/3/13 10:57	C++ Header file	21 KB

字符处理函数

数据结构定义

数学

内存相关

字符串处理函数

4. 打开 uCOS-II/ Source

os.h	2014/4/17 16:25	C++ Header file	2 KB
os_core.c	2014/12/11 16:59	C Source file	88 KB
os_flag.c	2014/4/17 16:25	C Source file	56 KB
os_mbox.c	2014/4/17 16:25	C Source file	31 KB
os_mem.c	2014/4/17 16:25	C Source file	20 KB
os_mutex.c	2014/4/17 16:25	C Source file	38 KB
os_q.c	2014/4/17 16:25	C Source file	42 KB
os_sem.c	2014/4/17 16:25	C Source file	29 KB
os_task.c	2014/4/17 16:25	C Source file	60 KB
os_time.c	2014/4/17 16:25	C Source file	11 KB
os_tmr.c	2014/4/17 16:25	C Source file	44 KB
ucos_ii.h	2014/4/17 16:25	C++ Header file	81 KB

头文件

内核文件

标志性事件组

消息邮箱

内存管理

互斥信号量

消息队列

信号量

任务相关

时间相关

软件定时器

头文件

三. UCOS II 移植

3.1 裸机工程的准备

找一个之前的裸机工程，最好带串口功能，方便调试。

CORE2017/6/5 20:46文件夹

FWLIB2019/3/31 17:29文件夹

HARDWARE2019/3/31 17:29文件夹

OBJ2019/3/31 17:29文件夹

STM32F4xx_StdPeriph_Driver2019/3/31 17:29文件夹

SYSTEM2019/3/31 17:29文件夹

USER2019/3/31 17:31文件夹

keilkill.bat2011/4/23 10:24Windows 批处理...

Project

Project: Template

Template

USER

main.c

HARDWARE

delay.c

sys.c

usart.c

spi_flash.c

bsp_lcd.c

gpio.c

touch.c

sd.c

rtc.c

SYSTEM

stm32f4xx_it.c

main.c

startup_stm32f40_41xxx.s

stm32f4xx_spi.c

1#include "sys.h"

2#include "delay.h"

3#include "usart.h"

4#include "bsp_lcd.h"

5#include "gpio.h"

6#include "rtc.h"

7#include "timer.h"

8#include "touch.h"

9#include "sd.h"

10int main(void)

11

12delay_init(168); //延时初始化

13NVIC_SetPriorityGrouping(7-2); //设置为分组2

14USART3_Init(115200); //串口波特率设置

15Lcd_Init();

16TFTLCD_Init();

17sFLASH_Init();

18SD_Init();

19while(1)

20{

21

22}

23

24

3.2 复制相关 μC/OS- II 的移植文件

1. 复制 ucOS-II 到工程中

CORE2017/6/5 20:46文件夹

FWLIB2019/3/31 17:29文件夹

HARDWARE2019/3/31 17:29文件夹

OBJ2019/3/31 17:29文件夹

STM32F4xx_StdPeriph_Driver2019/3/31 17:29文件夹














SYSTEM2019/3/31 17:29文件夹

uCOS-II2019/3/31 17:32文件夹

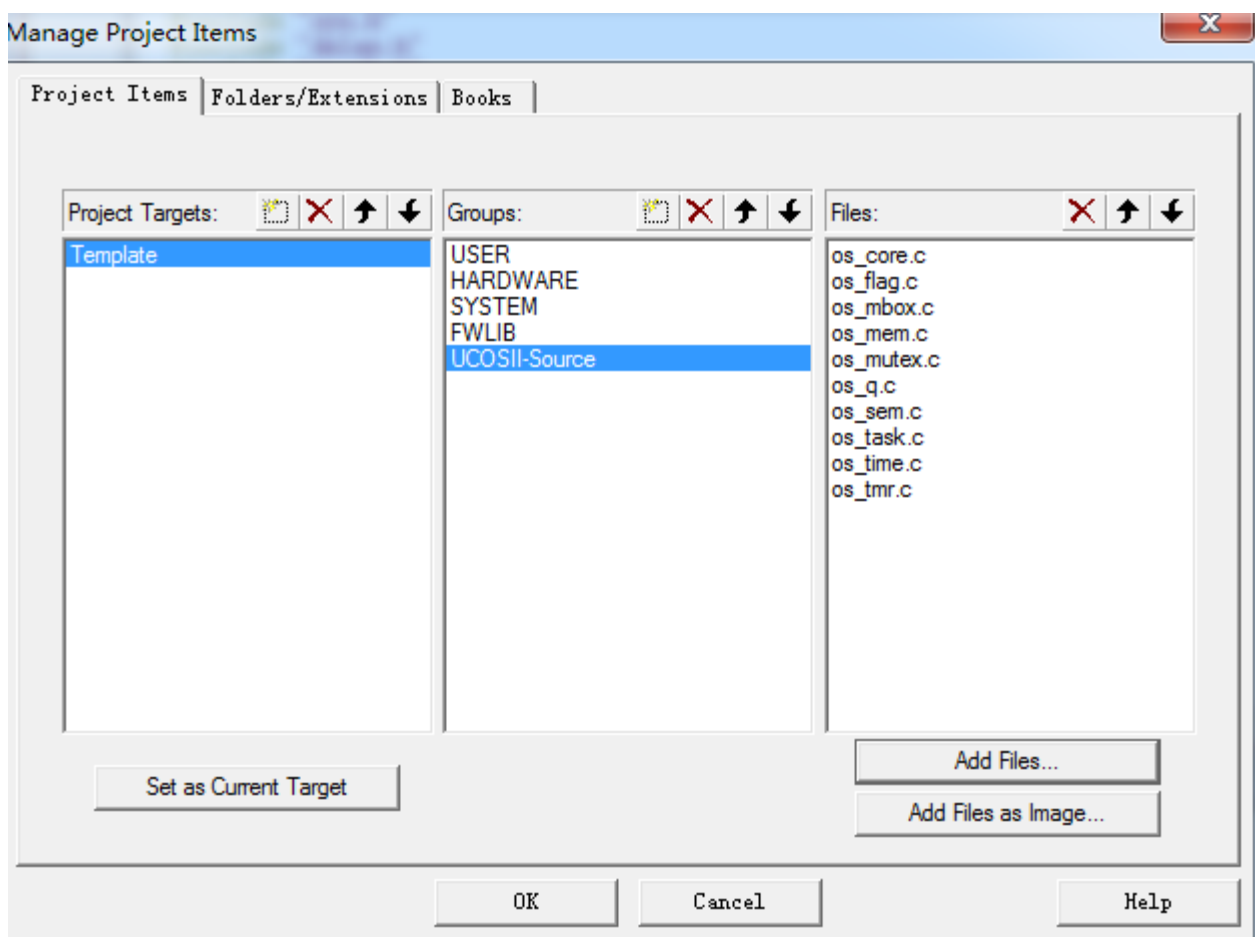
USER2019/3/31 17:31文件夹

keilkill.bat2011/4/23 10:24Windows 批处理...1 KB

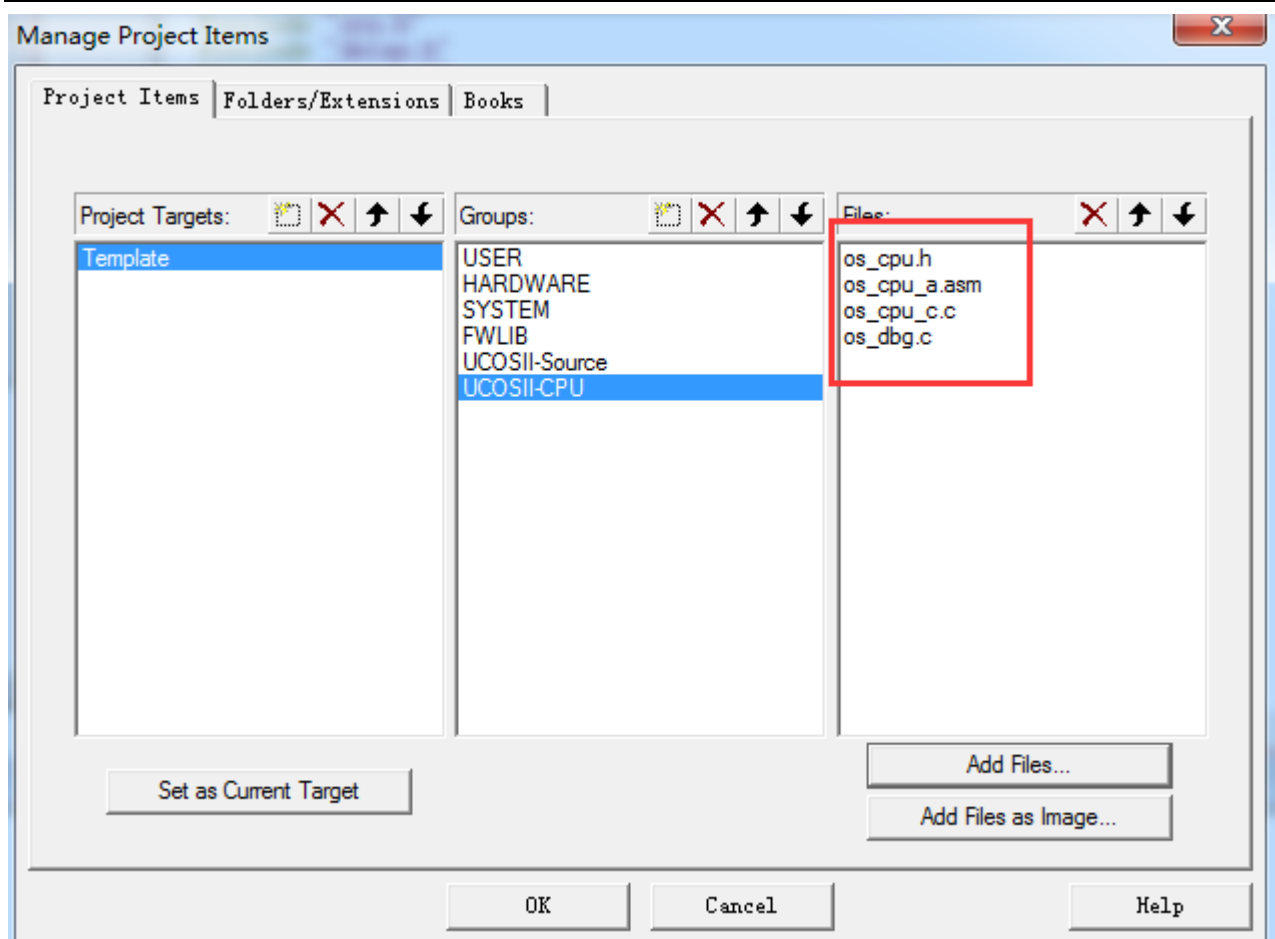
2. 复制\Micrium\Examples\ST\STM3240G-EVAL\OS2\ os_cfg.h 到 uCOS-II\Source 中

 os.h	2014/4/17 16:25	C++ Header file	2 KB
 os_cfg.h	2015/4/21 15:05	C++ Header file	11 KB
 os_core.c	2014/12/11 16:59	C Source file	88 KB
 os_flag.c	2014/4/17 16:25	C Source file	56 KB
 os_mbox.c	2014/4/17 16:25	C Source file	31 KB
 os_mem.c	2014/4/17 16:25	C Source file	20 KB
 os_mutex.c	2014/4/17 16:25	C Source file	38 KB
 os_q.c	2014/4/17 16:25	C Source file	42 KB
 os_sem.c	2014/4/17 16:25	C Source file	29 KB
 os_task.c	2014/4/17 16:25	C Source file	60 KB
 os_time.c	2014/4/17 16:25	C Source file	11 KB
 os_tmr.c	2014/4/17 16:25	C Source file	44 KB
 ucos_ii.h	2019/3/31 17:58	C++ Header file	81 KB

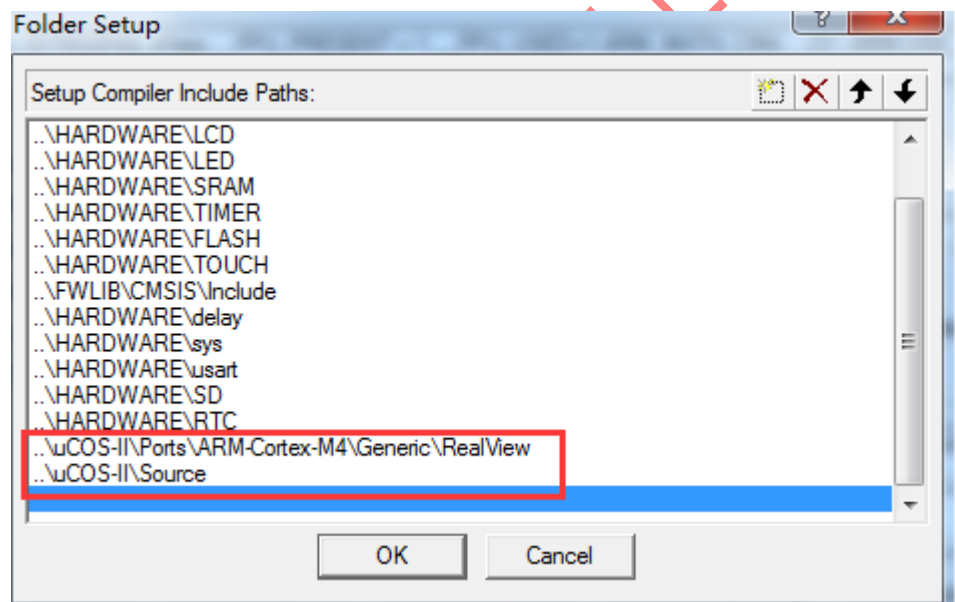
3. 添加 ucos-ii 源码到工程中



4. 添加 uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView 中的 CPU 相关文件到工程中



5. 添加头文件路径

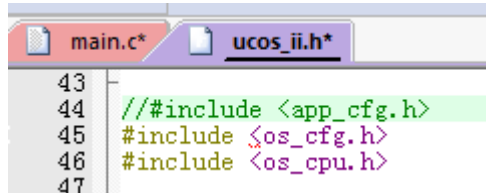


3.3 编译纠错

1. 编译

```
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_cpu_c.c: 0 warnings, 1 error
compiling os_dbg.c...
..\uCOS-II\Source\ucos_ii.h(44): error: #5: cannot open source input file "app_cfg.h": No such file or directory
#include <app_cfg.h>
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_dbg.c: 0 warnings, 1 error
"..\OBJ\Template.axf" - 12 Error(s), 0 Warning(s).
Target not created.
Build Time Elapsed: 00:01:10
```

找不到此头文件，此头文件在\Micrium\Examples\ST\STM3240G-EVAL\OS2\目录下，我们并没有使用到此头文件的功能，所以把包含此头文件屏蔽即可。

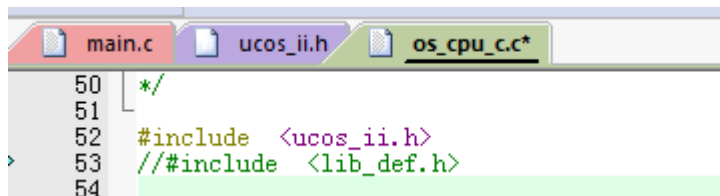


```
43
44 // #include <app_cfg.h>
45 #include <os_cfg.h>
46 #include <os_cpu.h>
47
```

2. 编译

```
compiling os_cpu_c.c...
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_cpu_c.c(53): error: #5: cannot open source input file "lib_def.h": No such file or directory
#include <lib_def.h>
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_cpu_c.c: 0 warnings, 1 error
compiling os_dbg.c...
"..\OBJ\Template.axf" - 1 Error(s), 0 Warning(s).
Target not created.
Build Time Elapsed: 00:00:10
```

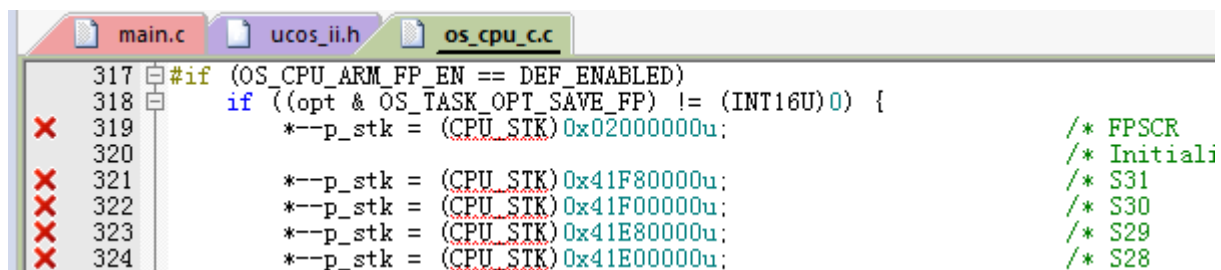
此关文件在\Micrium\Software\uC-LIB 目录下，不需要它，再次屏蔽。



```
50 /*
51
52 #include <ucoss_ii.h>
53 // #include <lib_def.h>
54
```

3. 编译

```
Build Output
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_cpu_c.c(347): error: #65: expected a ";"
*--p_stk = (CPU_STK) 0x40A00000u; /* S5 */
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_cpu_c.c(348): error: #65: expected a ";"
*--p_stk = (CPU_STK) 0x40800000u; /* S4 */
..\uCOS-II\Ports\ARM-Cortex-M4\Generic\RealView\os_cpu_c.c: 0 warnings, 30 errors
"..\OBJ\Template.axf" - 30 Error(s), 0 Warning(s).
Target not created.
Build Time Elapsed: 00:00:02
```



```
317 #if (OS_CPU_ARM_FP_EN == DEF_ENABLED)
318     if ((opt & OS_TASK_OPT_SAVE_FP) != (INT16U)0) {
319         *--p_stk = (CPU_STK) 0x02000000u; /* FPSCR
320                                           /* Initiali
321         *--p_stk = (CPU_STK) 0x41F80000u; /* S31
322         *--p_stk = (CPU_STK) 0x41F00000u; /* S30
323         *--p_stk = (CPU_STK) 0x41E80000u; /* S29
324         *--p_stk = (CPU_STK) 0x41E00000u; /* S28
```

把#if 部分内容屏蔽：


```

main.c  ucos_ii.h  os_cpu_c.c*
314  *(&p_stk) = (OS_STK)0x06080808uL; /* F5 */
315  *(&p_stk) = (OS_STK)0x04040404uL; /* F4 */
316
317  //if (OS_CPU_ARM_FP_EN == DEF_ENABLED)
318  if ((opt & OS_TASK_OPT_SAVE_FP) != (INT16U)0) {
319      *p_stk = (CPU_STK)0x02000000u; /* FPSCR */
320      // Initialize S0-S31 floating point registers
321      *p_stk = (CPU_STK)0x41F80000u; /* S31 */
322      *p_stk = (CPU_STK)0x41F00000u; /* S30 */
323      *p_stk = (CPU_STK)0x41E80000u; /* S29 */
324      *p_stk = (CPU_STK)0x41E00000u; /* S28 */
325      *p_stk = (CPU_STK)0x41D80000u; /* S27 */
326      *p_stk = (CPU_STK)0x41D00000u; /* S26 */
327      *p_stk = (CPU_STK)0x41C80000u; /* S25 */
328      *p_stk = (CPU_STK)0x41C00000u; /* S24 */
329      *p_stk = (CPU_STK)0x41B80000u; /* S23 */
330      *p_stk = (CPU_STK)0x41B00000u; /* S22 */
331      *p_stk = (CPU_STK)0x41A80000u; /* S21 */
332      *p_stk = (CPU_STK)0x41A00000u; /* S20 */
333      *p_stk = (CPU_STK)0x41980000u; /* S19 */
334      *p_stk = (CPU_STK)0x41900000u; /* S18 */
335      *p_stk = (CPU_STK)0x41880000u; /* S17 */
336      *p_stk = (CPU_STK)0x41800000u; /* S16 */
337      *p_stk = (CPU_STK)0x41700000u; /* S15 */
338      *p_stk = (CPU_STK)0x41600000u; /* S14 */
339      *p_stk = (CPU_STK)0x41500000u; /* S13 */
340      *p_stk = (CPU_STK)0x41400000u; /* S12 */
341      *p_stk = (CPU_STK)0x41300000u; /* S11 */
342      *p_stk = (CPU_STK)0x41200000u; /* S10 */
343      *p_stk = (CPU_STK)0x41100000u; /* S9 */
344      *p_stk = (CPU_STK)0x41000000u; /* S8 */
345      *p_stk = (CPU_STK)0x40E00000u; /* S7 */
346      *p_stk = (CPU_STK)0x40C00000u; /* S6 */
347      *p_stk = (CPU_STK)0x40A00000u; /* S5 */
348      *p_stk = (CPU_STK)0x40800000u; /* S4 */
349      *p_stk = (CPU_STK)0x40400000u; /* S3 */
350      *p_stk = (CPU_STK)0x40000000u; /* S2 */
351      *p_stk = (CPU_STK)0x3F800000u; /* S1 */
352      *p_stk = (CPU_STK)0x00000000u; /* S0 */
353  }
354  #endif

```

4. 编译

```

linking...
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TCBInitHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TaskCreateHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TaskDelHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TaskIdleHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TaskReturnHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TaskStatHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TaskSwHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol App_TimeTickHook (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol DEF_BIT_FIELD (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol DEF_BIT_MASK (referred from os_cpu_c.o).
Not enough information to list image symbols.
Finished: 1 information, 0 warning and 10 error messages.
"..\OBJ\Template.axf" - 10 Error(s), 2 Warning(s).
Target not created.
Build Time Elapsed: 00:00:04

```

函数没定义。

```

#if OS_APP_HOOKS_EN > 0u
    App_TaskSwHook();
#endif
}

```

把宏定义改为0，让条件不成立。

```

main.c  ucos_ii.h  os_cpu_c.c  os_cfg.h*
30  #define OS_APP_HOOKS_EN 0u /*

```

5. 编译


```

..\OBJ\Template.axf: Error: L6218E: Undefined symbol DEF_BIT_FIELD (referred from os_cpu_c.o).
..\OBJ\Template.axf: Error: L6218E: Undefined symbol DEF_BIT_MASK (referred from os_cpu_c.o).
Not enough information to list image symbols.
Finished: 1 information, 0 warning and 2 error messages.
"..\OBJ\Template.axf" - 2 Error(s), 2 Warning(s).

```

没定义，搜索在哪。

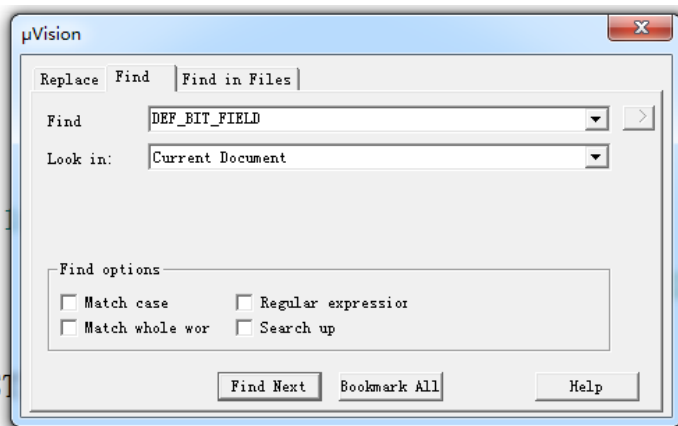
```

void OS_CPU_SysTickInit (INT32U cnts)
{
    INT32U prio;

    OS_CPU_CM4_NVIC_ST_RELOAD = cnts - 1;

    prio = OS_CPU_CM4_NVIC_SHPRI3;
    prio &= DEF_BIT_FIELD(24, 0);
    prio |= DEF_BIT_MASK(OS_CPU_CFG_SYSTICK_PRIO, 24);
}

```



此函数为系统滴答初始化，把内容屏蔽，换成我们之前写的。

```

core_cm4.h  main.c  ucos_ii.h  os_cpu_c.c  os_cfg.h
1676  */
1677  STATIC_INLINE uint32_t SysTick_Config(uint32_t ticks)
1678  {
1679      if ((ticks - 1) > SysTick_LOAD_RELOAD_Msk) return (1); /* Reload value impossible */
1680
1681      SysTick->LOAD = ticks - 1; /* set reload register */
1682      NVIC_SetPriority (SysTick_IRQn, (1<<__NVIC_PRIO_BITS) - 1); /* set Priority for SysTick Interrupt */
1683      SysTick->VAL = 0; /* Load the SysTick Counter Value */
1684      SysTick->CTRL = SysTick_CTRL_CLKSOURCE_Msk |
1685                     SysTick_CTRL_TICKINT_Msk |
1686                     SysTick_CTRL_ENABLE_Msk; /* Enable SysTick IRQ and SysTick Timer */
1687      return (0); /* Function successful */
1688  }

void OS_CPU_SysTickInit (INT32U cnts)
{
    // INT32U prio;
    // OS_CPU_CM4_NVIC_ST_RELOAD = cnts - 1u;
    // prio = OS_CPU_CM4_NVIC_SHPRI3;
    // prio &= DEF_BIT_FIELD(24, 0);
    // prio |= DEF_BIT_MASK(OS_CPU_CFG_SYSTICK_PRIO, 24);
    // OS_CPU_CM4_NVIC_SHPRI3 = prio;
    // OS_CPU_CM4_NVIC_ST_CTRL |= OS_CPU_CM4_NVIC_ST_CTRL_CLK_SRC |
    //                             OS_CPU_CM4_NVIC_ST_CTRL_ENABLE;
    // OS_CPU_CM4_NVIC_ST_CTRL |= OS_CPU_CM4_NVIC_ST_CTRL_INTEN;
}

```

改为：

```
#include "stm32f4xx.h"
void OS_CPU_SysTickInit (INT32U cnts)
{
    if ((cnts - 1) > SysTick_LOAD_RELOAD_Msk) return ;
    SysTick->LOAD = cnts - 1;
    NVIC_SetPriority (SysTick_IRQn, (1<<__NVIC_PRIO_BITS) - 1)
    SysTick->VAL = 0;
    SysTick->CTRL = SysTick_CTRL_CLKSOURCE_Msk |
                    SysTick_CTRL_TICKINT_Msk |
                    SysTick_CTRL_ENABLE_Msk;

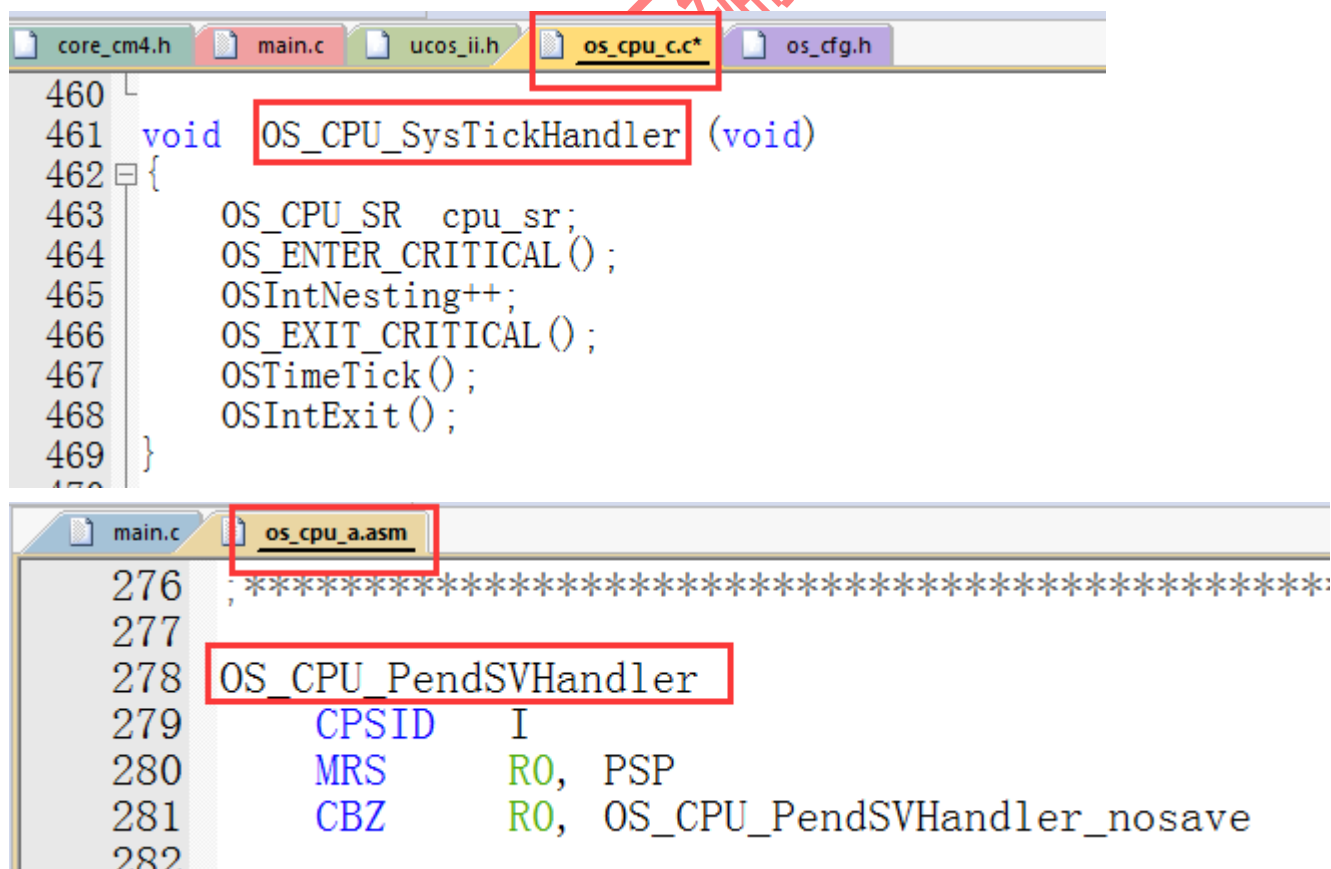
    return ;
}
```

6. 编译

```
compiling os_cpu_c.c...
linking...
Program Size: Code=32668 RO-data=6532 RW-data=64 ZI-data=12600
FromELF: creating hex file...
"..\\OBJ\\Template.axf" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:04
```

3.4 修改文件

1. 修改中断服务函数



The screenshot shows two windows from an IDE. The top window, titled 'os_cpu_c.c', contains the following code:

```
460
461 void OS_CPU_SysTickHandler (void)
462 {
463     OS_CPU_SR cpu_sr;
464     OS_ENTER_CRITICAL();
465     OSIntNesting++;
466     OS_EXIT_CRITICAL();
467     OSTimeTick();
468     OSIntExit();
469 }
```

The bottom window, titled 'os_cpu_a.asm', contains the following assembly code:

```
276 ;*****
277
278 OS_CPU_PendSVHandler
279     CPSID     I
280     MRS       R0, PSP
281     CBZ       R0, OS_CPU_PendSVHandler_nosave
282
```

可以看出此中断服务函数跟之前学习的不一样，所以需要修改。
修改启动文件的中断服务函数名。

```
startup_stm32f40_41xxx.s*  main.c  os_cpu_c.c
82 DCD 0 ; Reserved
83 ;DCD PendSV_Handler ; PendSV Handler
84 ;DCD SysTick_Handler ; SysTick Handler
85 ;修改为
86 DCD OS_CPU_PendSVHandler ; PendSV Handler modifi by chenxingxing
87 DCD OS_CPU_SysTickHandler ; SysTick Handler modifi by chenxingxing
88
```

加上声明

```
startup_stm32f40_41xxx.s*  main.c  os_cpu_a.asm
65 EXPORT __Vectors
66 EXPORT __Vectors_End
67 EXPORT __Vectors_Size
68
69 IMPORT OS_CPU_SysTickHandler ;by chenxingxing
70 IMPORT OS_CPU_PendSVHandler ;by chenxingxing
71
```

2. 修改 main.c

1) 添加 ucOS 头文件

```
#include "ucos_ii.h"
```

2) 添加系统滴答初始化

```
OS_CPU_SysTickInit(168000000 / OS_TICKS_PER_SEC);
```

3) 添加 ucOS 初始化

```
OSInit();
```

4) 创建 ucOS 任务

```
#include "stm32f4xx.h"
#include "led.h"
#include "key.h"
#include "usart.h"
#include "string.h"
#include "ucos_ii.h"

#define OS_TASK_START_PRIO 10
#define OS_TASK_START_STK_SIZE 128
OS_STK OSTaskStartStk[OS_TASK_START_STK_SIZE];
void OS_TaskStart (void *p_arg);
int main ()
{

    LED_Init(); //LED 初始化
    KEY_Init(); //按键初始化
    usart1_init(9600); //串口初始化，波特率为 9600
    OS_CPU_SysTickInit(168000000 / OS_TICKS_PER_SEC); //系统滴答初始化
    OSInit(); //OS 初始化
    //创建任务
```

```
OSTaskCreate(OS_TaskStart,
              (void *)0,
              &OSTaskStartStk[OS_TASK_START_STK_SIZE - 1u],
              OS_TASK_START_PRIO);
OSStart();//开启 os,开始调度,跳到已经就绪优先级最高的任务中
}
void OS_TaskStart (void *p_arg)
{
    p_arg = p_arg; /* Prevent compiler warning for not using 'p_arg' */
    for (;;) {
        printf("start task\r\n");
        // OStimeDly(100);//释放 CPU 使用权，释放了多长时间？ 10* 心跳节拍时间。
        OStimeDlyHMSM(0,0,1,0);//释放 CPU1 秒时间
    }
}
```

陈工编制