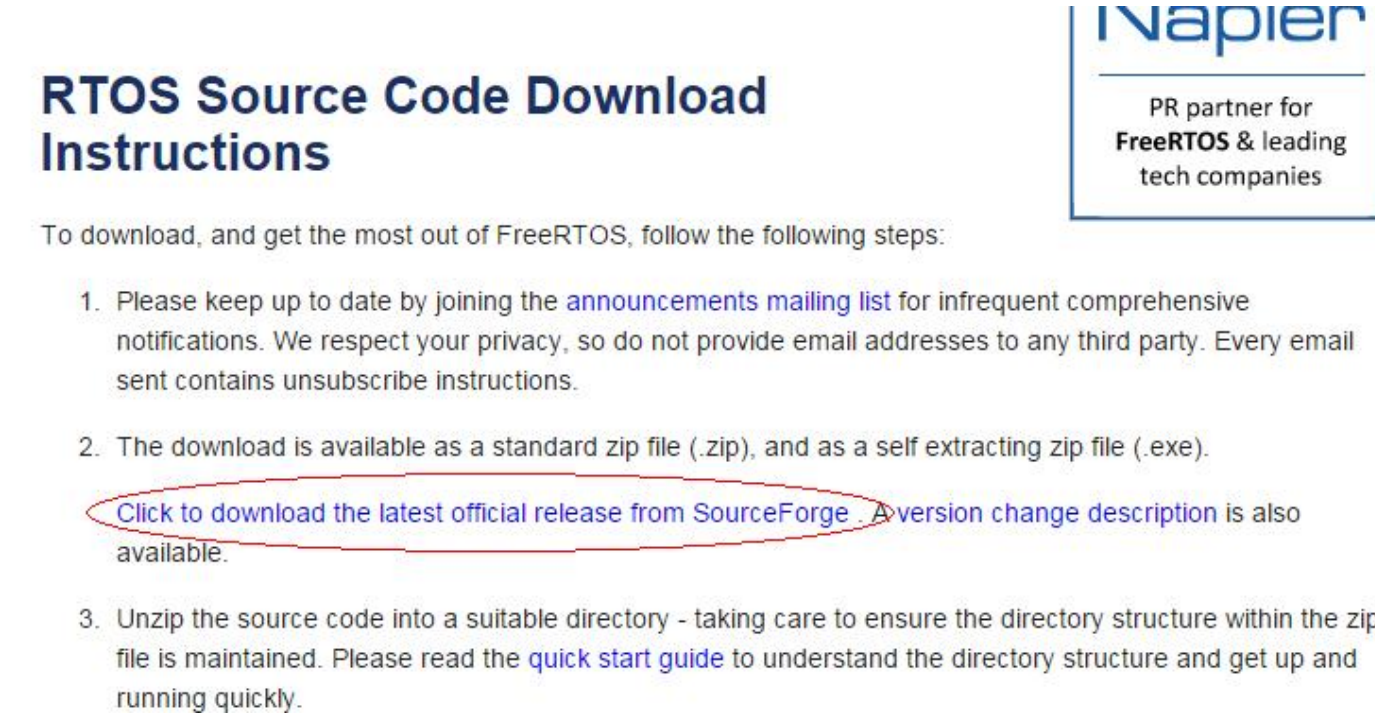


STM32F103移植FreeRtos笔记

RTOS版本：FreeRTOS_V8.2.2

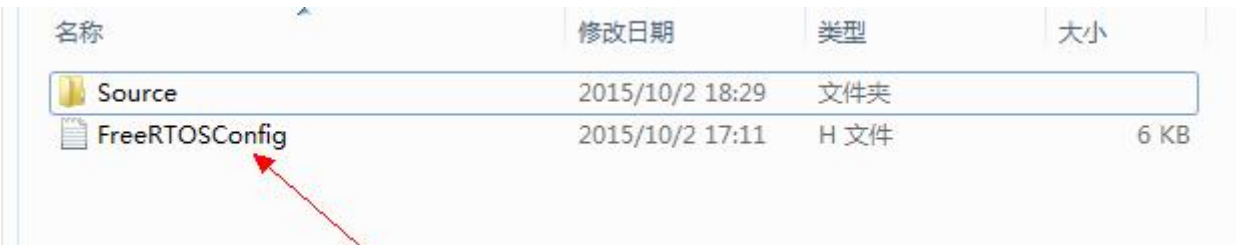
- 一、下载FreeRTOS源文件
- 这个可以在百度上下载，或者在官网上面下载<http://www.freertos.org/a00104.html>
- 下图位置即可下载了



- 目前最新版本是8.2.2
- 二、下载源码下来，解压，将源码添加到工程
- 2.1 新建一个移植工程（拷贝原子哥的也行），在工程中新建一个文件夹FreeRTOS存放源码
- 2.2 打开解压到的源码“..\FreeRTOSv8.2.2”，里边有几个文件夹和文件等东西

| 名称 | 修改日期 | 类型 | 大小 |
|------------------------------------|-----------------|---------------|------|
| FreeRTOS | 2015/9/30 23:39 | 文件夹 | |
| FreeRTOS-Plus | 2015/9/30 23:39 | 文件夹 | |
| New - Direct to Task Notifications | 2015/1/16 18:56 | Internet 快捷方式 | 1 KB |
| New - FreeRTOS+TCP | 2014/12/22 3:03 | Internet 快捷方式 | 1 KB |
| Quick_Start_Guide | 2013/9/17 16:17 | Internet 快捷方式 | 1 KB |
| readme | 2013/9/17 16:20 | 文本文档 | 2 KB |
| Upgrading-to-FreeRTOS-8 | 2014/1/1 3:14 | Internet 快捷方式 | 1 KB |

- 看到上图中红色圈圈的就是源码了，整个文件夹拷贝到新建工程的“FreeRTOS”文件夹中
- 2.4 在源码文件夹中“..\FreeRTOSv8.2.2\FreeRTOS\Demo\CORTEX_STM32F103_Keil”这个路径下，找到文件“FreeRTOSConfig.h”，拷贝到工程文件夹“FreeRTOS”中



- 三、打开MDK软件，新建工程，添加文件

公告

昵称：狗尾巴草word
园龄：2年2个月
粉丝：10
关注：1
[+加关注](#)

| 2019年11月 | | | | | | |
|----------|----|----|----|----|----|----|
| 日 | 一 | 二 | 三 | 四 | 五 | 六 |
| 27 | 28 | 29 | 30 | 31 | 1 | 2 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |

搜索

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

我的标签

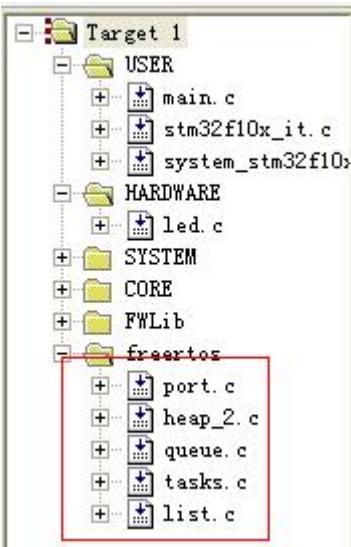
DPS(1)

随笔分类

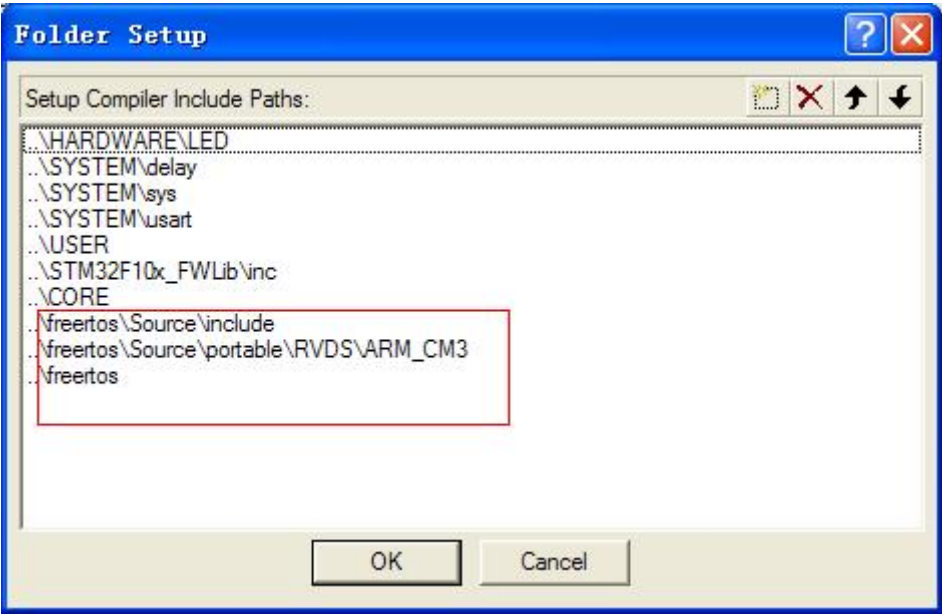
ARM
C++(4)
DSP(1)
Java(10)
java web(8)
Mysql(2)
STM32(33)
励志修行(2)
通讯(2)

随笔档案

2019年10月(3)
2019年8月(1)
2019年3月(9)
2018年11月(1)
2018年10月(14)
2018年9月(34)
2017年8月(1)



添加文件如上如所示，红色框框就是添加的FreeRTOS文件了；
“port.c”文件路径.\FreeRTOS\Source\portable\RVDS\ARM_CM3
“heap_2.c”文件路径.\FreeRTOS\Source\portable\MemMang
“list.c”、“queue.c”、“tasks.c”文件路径.\FreeRTOS\Source



五、完成以上工作之后，编写main函数了

```
#include "FreeRTOS.h"
#include "task.h"
#include "queue.h"
#include "misc.h"
#include "led.h"

void LED0_Task(void *pvParameters);
void LED1_Task(void *pvParameters);

int main()
{
    LED_Init();          //初始化与LED连接的硬件接口
    xTaskCreate(LED0_Task,(const char
*)"LED0",configMINIMAL_STACK_SIZE,NULL,tskIDLE_PRIORITY+3,NULL);
    xTaskCreate(LED1_Task,(const char
*)"LED1",configMINIMAL_STACK_SIZE,NULL,tskIDLE_PRIORITY+4,NULL);
    vTaskStartScheduler();
}

void LED0_Task(void *pvParameters)
{
    while(1)
    {
        LED0=!LED0;
        vTaskDelay(1000/portTICK_RATE_MS);
    }
}

void LED1_Task(void *pvParameters)
{
    while(1)
    {
        LED1=!LED1;
        vTaskDelay(300/portTICK_RATE_MS);
    }
}
```

最新评论

- 1. Re:Java 13 在win10 安装及配置
1.5以后就不用配置CLASSPATH，11以后也不需要装jre了，jdk里面已经集成了jre，只需要配置path就好。
--Code_Rookie
- 2. Re: (4) STM32使用HAL库实现串口通讯——理论讲解
这不是掰碎了喂给我吃么！！！真香
--xiaoxiaofeiz
- 3. Re: (5) STM32使用HAL库实现串口通讯——实战操作
写的真好o(^v^)o，博主大才
--努力嗯~o(*￣▽￣*)o
- 4. Re: (4) STM32使用HAL库实现串口通讯——理论讲解
写的太好了，博主你他娘的可真sky个人才。~o(*￣▽￣*)o
--努力嗯~o(*￣▽￣*)o

阅读排行榜

- 1. (4) STM32使用HAL库实现串口通讯——理论讲解(12698)
- 2. (3) STM32使用HAL库操作外部中断——实战操作(5667)
- 3. STM32学习笔记：读写内部Flash（介绍+附代码）(2975)
- 4. STM32定时器配置（TIM1-TIM8）高级定时器+普通定时器，定时计数模式下总结(2032)
- 5. DSP28035的编程初步--GPIO操作(1562)

评论排行榜

- 1. (4) STM32使用HAL库实现串口通讯——理论讲解(2)
- 2. (5) STM32使用HAL库实现串口通讯——实战操作(1)
- 3. Java 13 在win10 安装及配置(1)

推荐排行榜

- 1. (4) STM32使用HAL库实现串口通讯——理论讲解(2)
- 2. (5) STM32使用HAL库实现串口通讯——实战操作(1)

```
}  
}
```

只设置了两个LED，按照不同的频率闪烁

六、修改启动文件，首先见启动文件的只读属性去掉，否则修改不了（已经去掉的就略过了）

```
45 Heap_Size      EQU      0x00000200  
46  
47  
48 __heap_base     AREA     HEAP, NOINIT, READWRITE, ALIGN=3  
49 Heap_Mem        SPACE   Heap_Size  
50 __heap_limit  
51  
52 PRESERVE8  
53 THUMB  
54  
55 ;修改相关中断服务函数由FreeRTOS管理  
56 IMPORT xPortPendSVHandler  
57 IMPORT xPortSysTickHandler  
58 IMPORT vPortSVCHandler  
59  
60  
61 ; Vector Table Mapped to Address 0 at Reset  
62 AREA RESET, DATA, READONLY  
63 EXPORT __Vectors  
64 EXPORT __Vectors_End  
65 EXPORT __Vectors_Size  
66  
67 __Vectors        DCD      __initial_sp          ; Top of Stack  
68 DCD      Reset_Handler                        ; Reset Handler  
69 DCD      NMI_Handler                          ; NMI Handler  
70 DCD      HardFault_Handler                    ; Hard Fault Handler  
71 DCD      MemManage_Handler                    ; MPU Fault Handler  
72 DCD      BusFault_Handler                     ; Bus Fault Handler  
73 DCD      UsageFault_Handler                   ; Usage Fault Handler  
74 DCD      0                                    ; Reserved  
75 DCD      0                                    ; Reserved  
76 DCD      0                                    ; Reserved  
77 DCD      0                                    ; Reserved  
78 DCD      SVC_Handler                          ; SVC Call Handler  
79 DCD      vPortSVCHandler;由FreeRTOS管理  
80 DCD      DebugMon_Handler                     ; Debug Monitor Handler  
81 DCD      0                                    ; Reserved  
82 DCD      PendSV_Handler                       ; PendSV Handler  
83 DCD      SysTick_Handler                      ; SysTick Handler  
84 DCD      xPortPendSVHandler;  
85 DCD      xPortSysTickHandler;  
86  
87 | ; External Interrupts  
88 DCD      WWDOG_IRQHandler                      ; Window Watchdog  
89 DCD      PVD_IRQHandler                       ; PVD through EXTI Line detect  
90 DCD      TAMPER_IRQHandler                     ; Tamper  
91 DCD      RTC_IRQHandler                       ; RTC
```

为什么这么改，我也不明白，IMPORT和EXPORT一个是调用内部引用文件，一个是调用外部引用文件，就和C语音写extern意思差不多。修改这几个地方将函数名和port.c文件中的相同就性了，好调用freertos的函数

七、搞定之后，就开始编译改错了，完成后就下载，看到两个灯按照不同的频率在闪烁，呵呵

八、上面差不多也算大功告成了，另外在中断的时候怎么办，需要保护机制。

//外部中断2服务程序

```
void EXTI2_IRQHandler(void)  
{  
    taskENTER_CRITICAL();  
  
    delay_us(1000);  
    if(KEY2==0) //按键KEY2  
    {  
        BEEP=!BEEP;  
    }  
  
    EXTI_ClearITPendingBit(EXTI_Line2); //清除LINE2上的中断标志位  
    taskEXIT_CRITICAL();  
}
```

测试了下，在两个灯的基础上，写了一个KEY（用中断方式）触发蜂鸣器，也是可行的。

上面理解了，基本上算入门了，下面主要还是研究UCOS-III，虽然freertos是完全免费的，UCOS需要收费，但是在国内的公司，只要不出口，还是用UCOS的多，混口饭吃还是随大流了，思维方式差不多

我总结了攻城狮在MCU时间利用率上有三个阶段

第一个阶段，只要有延时就采用delay方法，死等，不碰个头破血流不回头

第二个阶段，采用标志位，之前我一直是这样的，我要10ms检测一个东西，我会设置一个主定时，不断的查询到了没有，到了才执行，不到就执行其他事情

第三个阶段，采用操作系统，把上面查询的时间也省了，一个任务完成了，会自动进入到下一个优先级高的地方去，虽然有空闲模式，但是在任务集中的时候，效率还是很高的。

分类: STM32

狗尾巴草word
关注 - 1
粉丝 - 10
+加关注

« 上一篇： [sprintf用法](#)
» 下一篇： [按键扫描程序，仅三行程序\(转\)](#)

posted @ 2018-10-10 17:03 狗尾巴草word 阅读(261) 评论(0) 编辑 收藏

00

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)， [访问](#) 网站首页。

【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库
【活动】京东云服务器_云主机低于1折，低价高性能产品备战双11
【推荐】天翼云双十一提前开抢，1核1G云主机3个月仅需59元
【推荐】阿里云双11冰点钜惠，热门产品低至一折等你来抢！
【福利】个推四大热门移动开发SDK全部免费用一年，限时抢！

相关博文：
· [FREERTOS移植（MDK、STM32F103）](#)
· [FreeRTOS移植](#)
· [FreeRTOS学习及移植笔记之一：开始FreeRTOS之旅](#)
· [STM32下移植FreeRTOS操作系统](#)
· [【零基础学习FreeRTOS嵌入式系统】之一：FreeRTOS环境搭建](#)
» [更多推荐...](#)

最新 IT 新闻:
· [以小吃大蛇吞象？施乐计划砸锅卖铁收购惠普](#)
· [TIOBE 11月榜单：C、Swift、Go、D与Rust起起伏伏](#)
· [马丁斯科塞斯解释为什么他认为漫威电影不是真正的电影](#)
· [南开大学团队研获高性能柔性有机太阳能电池](#)
· [另辟蹊径！等离子体喷枪“瞄准”可控核聚变](#)
» [更多新闻...](#)

历史上的今天：
[2018-10-10 STM32_NVIC寄存器详解](#)
[2018-10-10 protel99se 问题汇总（不定期更新）](#)
[2018-10-10 STM32串口IAP实验笔记](#)