目录

[一、工程说明 2](#_Toc34643615)

[1、文件夹说明 2](#_Toc34643616)

[（1）文件夹总览 2](#_Toc34643617)

[（2）Project文件夹 2](#_Toc34643618)

[（3）Doc略 2](#_Toc34643619)

[（4）Bootloader文件夹 2](#_Toc34643620)

[（5）APP文件夹 2](#_Toc34643621)

[2、MDK工程说明 3](#_Toc34643622)

[（1）工程切换 3](#_Toc34643623)

[（2）BootLoader工程说明 3](#_Toc34643624)

[（3）APP工程说明 3](#_Toc34643625)

[3、C文件介绍（外设） 3](#_Toc34643626)

[二、任务介绍（线程） 4](#_Toc34643627)

[1、BootLoader线程 4](#_Toc34643628)

[（1）空闲任务线程 4](#_Toc34643629)

[（2）APP任务线程 4](#_Toc34643630)

[（3）系统监视任务 4](#_Toc34643631)

[2、APP线程 4](#_Toc34643632)

[三、注意 4](#_Toc34643633)

[1、通用 4](#_Toc34643634)

[2、BootLoader端 5](#_Toc34643635)

[3、APP端 5](#_Toc34643636)

[附录1 5](#_Toc34643637)

[附录2 6](#_Toc34643638)

**BootLoader与APP说明**

# 一、工程说明

## 1、文件夹说明

### （1）文件夹总览



### （2）Project文件夹

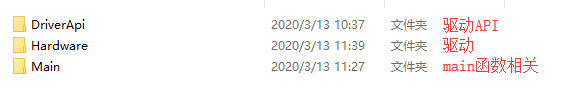
这是整个工程的项目管理文件，如下图所示



### （3）Doc略

### （4）Bootloader文件夹

存放Bootloader相关文件，如下图



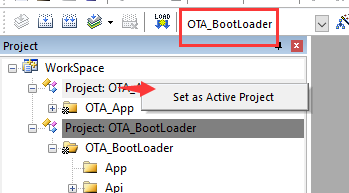
### （5）APP文件夹

存放APP代码，同上Bootloader文件夹， Library、freertos文件夹两者共用。

## 2、MDK工程说明

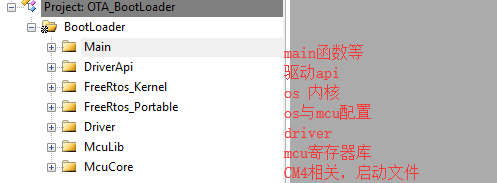
### （1）工程切换

如下图所示，右键Project，会出现set as active project选项，点击选择当前工程，当Project栏为灰色说明选中了，同时工具栏会出现选中的工程名称。



### （2）BootLoader工程说明

如下图所示

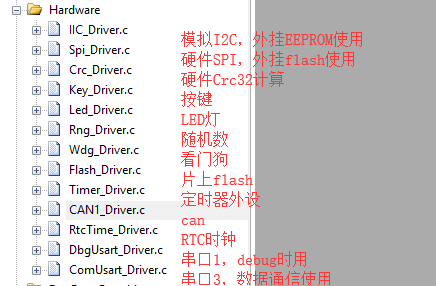


### （3）APP工程说明

同BootLoader工程，如上图所示。

## 3、C文件介绍（外设）

这里主要介绍了外设驱动相关的文件，如下图所示



# 二、任务介绍（线程）

## 1、BootLoader线程

准确的说，BootLoader中有四个任务，空闲任务、APP任务1、APP任务2、系统监视任务。

### （1）空闲任务线程

这是系统启动时最先建的任务（系统自建，必须含有），当系统中没有其他任务运行时，就会执行空闲任务。可以反应系统繁忙程度，系统执行空闲任务越多，系统压力越小。

### （2）APP任务线程

使用者自定义每个任务的功能。

### （3）系统监视任务

目前除了喂狗操作外，没有更多东西，当程序在某个任务卡死而无法正常执行喂狗操作，系统将重启。当然也可以获取各个任务状态，但太消耗资源，不建议。

## 2、APP线程

同BootLoader线程，具体略

# 三、注意

## 1、通用

1）系统时间片设置为1000Hz，也就是1ms1个时间片；

2）FreeRtos的每个任务在任务运行完都必须调用任务切换函数，工程中采用的是延时，延时函数（毫秒级延时）里面会有任务切换，**特别注意：工程中定义了两个毫秒级延时，一个会产生任务切换（delay\_ms），一个不会产生任务切换（delay\_xms），在OS起来之后禁止使用delay\_xms；**

3）因为OS的存在，每个线程分配的栈大小是固定的，**所以申请大内存时很容易造成栈溢出，所以任务栈大小要分配合理**，OS总共管理的内存（RAM）大小为64KB，在任务函数中定义的局部变量、数组等都将消耗任务栈空间；

4）若要打印调试信息，必须先初始化Log，调用Logging\_Init()；

5）设备驱动相关的API还需不同程度的完善；

6）在板子上跑的时候**注意外接晶振大小，在system\_stm32f4xx.c中与程序适配**；

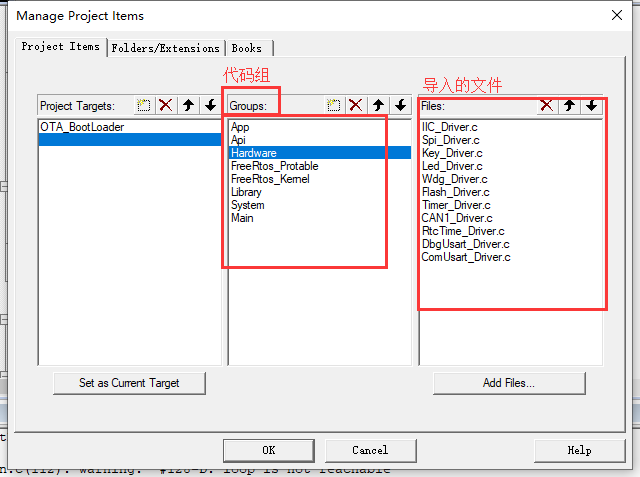
6）目前的基础工程，串口收到数据后，会原封不动发回去，每次传输不超过500字节；

7）flash备份区在0x080C0000至0x080FFFFF，共256KB

8）中断函数的优先级数字越小越高，而OS任务函数的优先级数字越大越高，OS任务函数的优先级取值为0至5，不包含5其中空闲任务优先级为0

9）MCU的硬件CRC仅支持32位数据，若传入8位或16位数据，也当做32位处理（高位补0）；

10）在添加自己的代码时，请模块化放入，工程中新建代码组后导入，如下图所示。



## 2、BootLoader端

1）BootLoader里定义了APP存放地址，不要随意更改；

2）BootLoader地址范围为0x08000000至0x08020000共128KB；

3）BootLoader中的跳转函数不要随便调用，只有BootLoader的事做完才能调用；

4）BootLoader的调试可以直接在线调试，如果有必要的话可以直接改为RAM调试，目前是在flash上。

## 3、APP端

1）APP端有重定向中断向量表到0x08020000地址，不能更改，否则APP运行入中断时程序跑飞；

2）APP中有个\_\_enable\_irq()函数是开启所有中断，在OS没有运行起来时禁用，任务创建完毕后调用，一开始就调用的话APP将进入硬件错误中断而死机；

3）APP地址范围为0x08020000至0x08060000，共256KB

4）APP中定义的所有变量必须初始化，因为BootLoader与APP共用RAM空间，当APP运行时可能有BootLoader残留的数据。

# 附录1

MCU工程是基于keil MDK的，这里附上下载地址：

链接：https://pan.baidu.com/s/1KWzHBvwmiLLgSJ4hl7keRg

提取码：lxkq

# 附录2