# 任务书

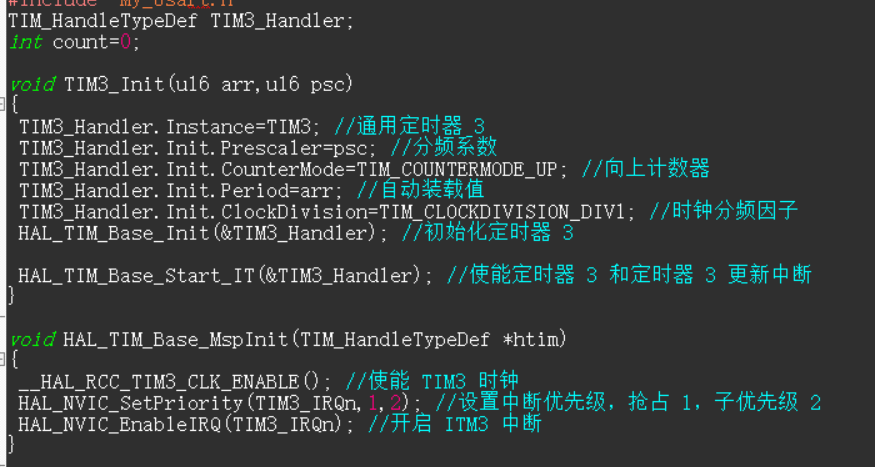
# 功能：

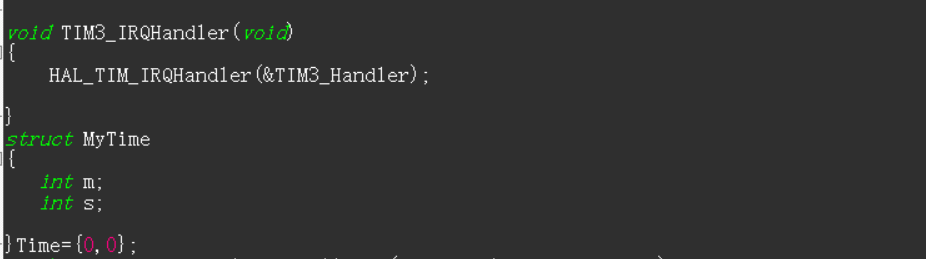
# 开发板与电脑连接，开发板将上电时间以 “已上电 xx分xx秒 按键事件：x次”形式向电脑上的串口调试助手发送数据。同时接收串口调试助手下发的数据，发送0x01点亮LED1,发送0x02点亮LED2，发送0x03则LED1、LED2以可见频率交替闪烁。其他数据不处理。

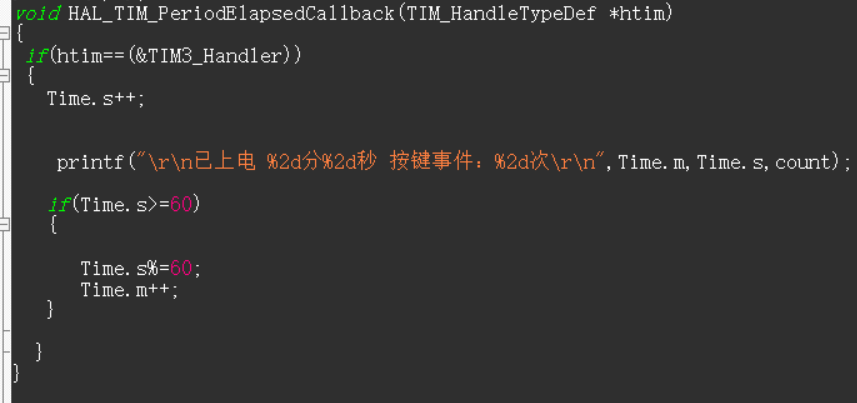
# 功能解析：上电计时部分用到的是定时器的计数功能，需定义两个变量代表分和秒，合适的分频使定时器以1s的间隔进入中断，进入60次中断，秒清零，分加一。按键事件的检测用到的是GPIO的输入功能，可用按键中断也可直接在循环中不断检测IO口电平状态，电平下降沿对应计数加一（消抖）。使用USART将以上信息发送到串口调试助手中。同时接收串口缓冲器中的数据，并加以判断，执行对应的动作。建议以上每个部分/功能 以子函数的形式书写。

定时器：选用TIM3中断，配置TIM3，使1s中断一次，秒+1，到了60次分+1，秒清0。

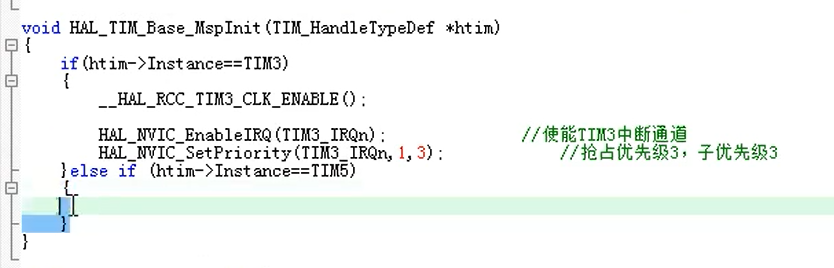
在timer.c写，这里只用了一个定时器的配置，如果要用到两个定时器，看b站f429 hal库的定时器中断这一节，它（https://www.bilibili.com/video/av28194216/?p=54）讲。



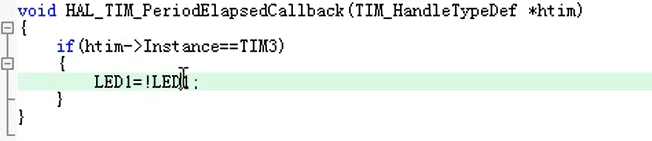




//多个定时器时这样写，



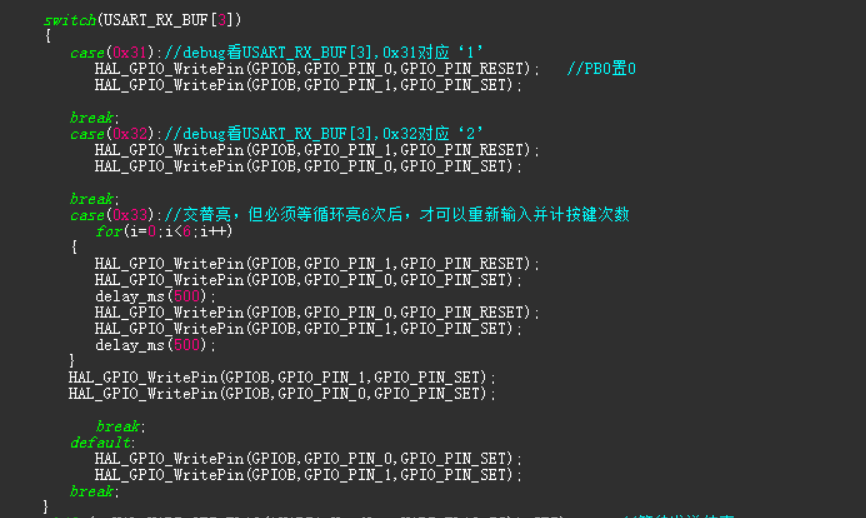
//还有中断回调函数



Main.c



LED控制：读取USART\_RX\_BUF[3]，串口发送数据是一个一个的发送的，如果发“0x01”则USART\_RX\_BUF[0]=’0’, USART\_RX\_BUF[1]=’x’, USART\_RX\_BUF[2]=’0’, USART\_RX\_BUF[3]=’1’,因此我们只需要读出USART\_RX\_BUF[3]的ASCII并判断给出响应的指示。



次数检测：如果USART\_RX\_BUF[3]有以下的值，说明串口控制led变化，次数+1。

