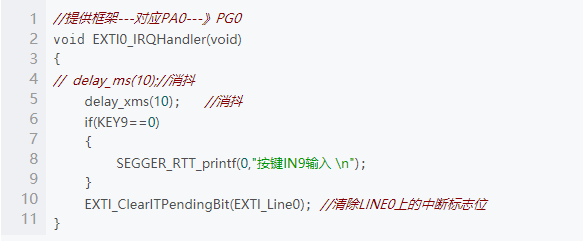
STM32 FreeRTOS 外部中断服务函数注意事项

移植正点原子的FreeRTOS操作系统到我们自己的工程，发现的一个注意事项；

看下面代码，带FreeRTOS的外部中断服务函数，实现按键消抖，不能用delay\_ms()或者和vTaskDelay()，会导致系统奔溃；

原因是delay\_ms()其实就是对 FreeRTOS 中的延时函数 vTaskDelay()的简单封装，所以在使用 delay\_ms()的时候就会导致任务切换，即不会导致这个外部中断延时阻塞，而切换到其他任务，导致外部中断出错；

这里我们可以使用delay\_xms()，其不会引起任务调度，是真的延时；（仅针对外部中断函数，因为我发现在按键扫描函数里面，可以使用delay\_ms();）



号外：

（1）delay\_ms()和vTaskDelay()一样使用，delay\_ms()其实就是对 FreeRTOS 中的延时函数 vTaskDelay()的简单封装

但vTaskDelay()程序更简洁，ms级延时更精准；

为了编程方便，delay\_ms()一般使用在外设模块初始化时候的ms级延时，例如RS485、LCD、以太网模块等，实测没有问题；

（2）注意NVIC\_IRQChannelPreemptionPriority优先级配置，正点原子例程的优先级0~4不会被FreeRTOS禁止，一般用于定时器；优先级5~15受FreeRTOS控制，一般用于外部中断、USART等；

（3）注意区分中断优先级和任务优先级，任务优先级数字越低表示任务的优先级越低， 0 的优先级最低， configMAX\_PRIORITIES-1 的优先级最高。空闲任务的优先级最低，为 0。

（4）任务函数一般不允许跳出循环，如果一定要跳出循环的话在跳出循环以后一定要调用函数 vTaskDelete(NULL)删除此任务！