第16章 μC/OS-III的启动流程

主讲内容

16.1 万事俱备，只欠东风

16.2 小心翼翼，十分谨慎

16.3 系统的启动

参考资料:《μCOS-III内核实现与应用开发实战指南》

万事俱备，只欠东风

这种方法是在main()函数中将硬件初始化，RTOS系统初始化，所有任务的创建这些都弄好，这个我称之为万事都已经准备好。最后只欠一道东风，即启动RTOS的调度器，开始多任务的调度

小心翼翼，十分谨慎

这种方法是在main()函数中将硬件和RTOS系统先初始化好，然后创建一个启动任务后就启动调度器，然后在启动任务里面创建各种应用任务，当所有任务都创建成功后，启动任务把自己删除

系统的启动

在系统上电的时候第一个执行的是启动文件里面由汇编编写的复位函数Reset_Handler，复位函数的最后会调用C库函数__main，__main()函数的主要工作是初始化系统的堆和栈，最后调用C中的main()函数，从而去到C的世界。

系统初始化

在调用创建任务函数之前，我们必须要对系统进行一次初始化，而系统的初始化是根据我们配置宏定义进行初始化的，有一些则是系统必要的初始化，如空闲任务，时钟节拍任务等

空闲任务

创建空闲任务：OS_IdleTaskInit()

CPU初始化

CPU的初始化，在μC/OS 中，有一个很重要的功能就是时间戳，它的精度高达ns级别，是CPU内核的一个资源，所以使用的时候要对CPU进行相关的初始化

SysTick初始化

时钟节拍的频率表示操作系统每1秒钟产生多少个tick，tick即是操作系统节拍的时钟周期，时钟节拍就是系统以固定的频率产生中断（时基中断），并在中断中处理与时间相关的事件，推动所有任务向前运行

内存初始化

```
Mem_Init();
```

OSStart()

在创建完任务的时候，我们需要开启调度器

THANKS

淘宝：fire-stm32.taobao.com

论坛： www.firebbs.cn

扫描进入淘宝店铺