

零死角玩转STM32



电源管理—实现低功耗

淘宝：firestm32.taobao.com

论坛：www.firebbs.cn



扫描进入淘宝店铺

主讲内容



01

STM32的电源管理简介

02

低功耗模式

03

电源管理相关的库函数及命令

04

电源管理实验

参考资料:《零死角玩转STM32》

“电源管理—实现低功耗” 章节

电源管理—实现低功耗



STM32的功耗模式

按功耗由高到低排列，STM32具有运行、睡眠、停止和待机四种工作模式。上电复位后STM32处于运行状态时，当内核不需要继续运行，就可以选择进入后面的三种低功耗模式降低功耗，这三种模式中，电源消耗不同、唤醒时间不同、唤醒源不同，用户需要根据应用需求，选择最佳的低功耗模式。三种低功耗的模式说明见下表。

这三种低功耗模式层层递进，运行的时钟或芯片功能越来越少，因而功耗越来越低。

电源管理—实现低功耗



模式	说明	进入方式	唤醒方式	对1.8V区域时钟的影响	对VDD区域时钟的影响	调压器
睡眠	内核停止，所有外设包括M3核心的外设，如NVIC、系统时钟(SysTick)等仍在运行	调用WFI命令	任一中断	内核时钟关，对其他时钟和ADC时钟无影响	无	开
		调用WFE命令	唤醒事件			
停止	所有的时钟都已停止	配置PWR_CR寄存器的PDDS +LPDS位+SLEEPDEEP位+WFI或WFE命令	任一外部中断(在外部中断寄存器中设置)	关闭所有1.8V区域的时钟	HSI 和 HSE 的振荡器关闭	开启或处于低功耗模式(依据电源控制寄存器的设定)
待机	1.8V 电源关闭	配置PWR_CR寄存器的PDDS +SLEEPDEEP位+WFI或WFE命令	WKUP 引脚的上升沿、RTC 闹钟事件、NRST 引脚上的外部复位、IWDG 复位			关

电源管理—实现低功耗



1.睡眠模式

在睡眠模式中，仅关闭了内核时钟，内核停止运行，但其片上外设，CM3核心的外设全都还照常运行。

有两种方式进入睡眠模式，它的进入方式决定了从睡眠唤醒的方式，分别是WFI(wait for interrupt)和WFE(wait for event)，即由等待“中断”唤醒和由“事件”唤醒。睡眠模式的各种特性见下表：

特性	说明
立即睡眠	在执行 WFI 或 WFE 指令时立即进入睡眠模式。
退出时睡眠	在退出优先级最低的中断服务程序后才进入睡眠模式。
进入方式	内核寄存器的SLEEPDEEP = 0，然后调用WFI或WFE指令即可进入睡眠模式； 另外若内核寄存器的SLEEPONEXIT=0时，进入“立即睡眠”模式，SLEEPONEXIT=1时，进入“退出时睡眠”模式。
唤醒方式	如果是使用WFI指令睡眠的，则可使用任意中断唤醒； 如果是使用WFE指令睡眠的，则由事件唤醒。
睡眠时	关闭内核时钟，内核停止，而外设正常运行，在软件上表现为不再执行新的代码。这个状态会保留睡眠前的内核寄存器、内存的数据。
唤醒延迟	无延迟。
唤醒后	若由中断唤醒，先进入中断，退出中断服务程序后，接着执行WFI指令后的程序；若由事件唤醒，直接接着执行WFE后的程序。

电源管理—实现低功耗



2.停止模式

在停止模式中，进一步关闭了其它所有的时钟，于是所有的外设都停止了工作，但由于其1.8V区域的部分电源没有关闭，还保留了内核的寄存器、内存的信息，所以从停止模式唤醒，并重新开启时钟后，还可以从上次停止处继续执行代码。停止模式可以由任意一个外部中断(EXTI)唤醒，在停止模式中选择电压调节器为开模式或低功耗模式。停止模式的特性见下表：

电源管理—实现低功耗



2.停止模式

特性	说明
调压器低功耗模式	在停止模式下调压器可工作在正常模式或低功耗模式，可进一步降低功耗
进入方式	<p>内核寄存器的SLEEPDEEP =1，PWR_CR寄存器中的PDDS=0，然后调用WFI或WFE指令即可进入停止模式；</p> <p>PWR_CR寄存器的LPDS=0时，调压器工作在正常模式，LPDS=1时工作在低功耗模式；</p>
唤醒方式	<p>如果是使用WFI指令睡眠的，可使用任意EXTI线的中断唤醒；</p> <p>如果是使用WFE指令睡眠的，可使用任意配置为事件模式的EXTI线事件唤醒。</p>
停止时	内核停止，片上外设也停止。这个状态会保留停止前的内核寄存器、内存的数据。
唤醒延迟	基础延迟为HSI振荡器的启动时间，若调压器工作在低功耗模式，还需要加上调压器从低功耗切换至正常模式下的时间。
唤醒后	若由中断唤醒，先进入中断，退出中断服务程序后，接着执行WFI指令后的程序；若由事件唤醒，直接接着执行WFE后的程序。唤醒后，STM32会使用HIS作为系统时钟。

电源管理—实现低功耗



3.待机模式

待机模式，它除了关闭所有的时钟，还把1.8V区域的电源也完全关闭了，也就是说，从待机模式唤醒后，由于没有之前代码的运行记录，只能对芯片复位，重新检测boot条件，从头开始执行程序。它有四种唤醒方式，分别是WKUP(PA0)引脚的上升沿，RTC闹钟事件，NRST引脚的复位和IWDG(独立看门狗)复位。

特性	说明
进入方式	内核寄存器的SLEEPDEEP =1，PWR_CR寄存器中的PDDS=1，PWR_CR寄存器中的唤醒状态位WUF=0，然后调用WFI或WFE指令即可进入待机模式；
唤醒方式	通过WKUP引脚的上升沿，RTC闹钟、唤醒、入侵、时间戳事件或NRST引脚外部复位及IWDG复位唤醒。
待机时	内核停止，片上外设也停止；内核寄存器、内存的数据会丢失；除复位引脚、RTC_AF1引脚及WKUP引脚，其它I/O口均工作在高阻态。
唤醒延迟	芯片复位的时间
唤醒后	相当于芯片复位，在程序表现为从头开始执行代码。

电源管理—实现低功耗



在以上讲解的睡眠模式、停止模式及待机模式中，若备份域电源正常供电，备份域内的**RTC**都可以正常运行，备份域内的寄存器的数据会被保存，不受功耗模式影响。

零死角玩转STM32



THANKS

论坛：www.firebbs.cn

淘宝：firestm32.taobao.com



扫描进入淘宝店铺