ARM体系结构对中断的支持

ARM指定了一些CPU模式来支持各种中断的处理。一般情况下CPU处于用户模式，当中断处理器接收到了中断信号时，CPU会进入IRQ模式；当接收到一个快速中断请求（Fast Interrupt Request, FIQ）时，CPU会进入FIQ模式。也就是说，ARM体系根据中断的优先级定义了两种中断，IRQ是普通的中断，FIQ是对处理时间要求较高的中断。之所以要将中断分成两级，是为了在处理大量中断和减少中断处理时间之间寻求平衡。

当ARM机器接收到IRQ信号时，会从内存空间0x18开始执行中断处理程序。而FIQ的中断处理程序的入口在0x1c。因此，在处理IRQ的情况下，其中断处理程序的内存空间只能存储一条指令，所以这条指令是一条分支指令，会转移到完整的中断处理程序。

ARM在当前程序状态寄存器（Current Program Status Register, cpsr）中设置了第7位和第6位分别用来表示是否开FIQ中断和IRQ中断。默认情况下，这两位都被设置为1，相关中断被屏蔽。可以用改变程序状态指令（Change Program State, cps）来改变这些状态位。

栈结构在中断处理程序中被广泛使用，因为切换状态需要恢复之前状态的信息，利用栈可以很好地实现状态信息的保存和恢复。每一种模式都有一个特定的寄存器保存栈指针，因此切换状态时可以自动切换栈。

ARM体系的机器对外设提供了若干种方式：把外设的物理寄存器映射到ARM的内存空间、映射到协同处理器的空间、或者是通过总线等设备连接到处理器。很多中断处理器提供了两种方式：内存空间和协同处理器。

具体地，当某个中断或异常导致CPU模式改变时，ARM会自动执行以下步骤：把cpsr保存到相应模式的程序状态保存寄存器（Saved Program Status Register, spsr）下，然后把pc保存到相应模式的链接寄存器（Link Register, lr）中。然后就可以把cpsr设置成相应的中断模式，并且把pc设置成相应异常处理程序的入口地址。在进入中断处理时，会相应地关FIQ或者IRQ中断，禁止同类型的其他中断被触发。因为在正常情况下，ARM里的中断是不可嵌套的。而当从中断处理程序退出时，ARM不会自动恢复环境。因此需要在程序中用软件实现以下操作：从相应模式的spsr和lr分别恢复数据到cpsr和pc中，这样程序可以继续正常进行。