1，中断源： 按键、定时器、网络数据等等等

2，异常（中断属于异常的一种）源：指令不对、数据访问有问题、Reset

3，中断的处理流程

1）中断初始化

a：设置中断源，让其可以产生中断

b：设置中断控制器，屏蔽某些中断，设置优先级

c：CPU 打开/关闭中断总开关

2）执行正常的程序

3）产生中断

4）CPU没执行完一条指令都会检查有无中断/异常产生

5）发现有中断/异常产生便开始处理

a:针对不同的中断/异常 跳到不同的地址执行程序

b:这些地址上的程序只是跳到某个函数执行

函数需要 保护现场、处理中断/异常 恢复现场

4，CPU 7种模式

1）usr 正常模式

2）sys 特权

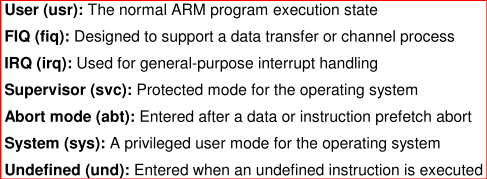
3）und 未定义

4）svc 管理

5）abt 中止 （指令预取指中止，访问数据中止

6）IRQ 中断

7）FIQ 快速中断

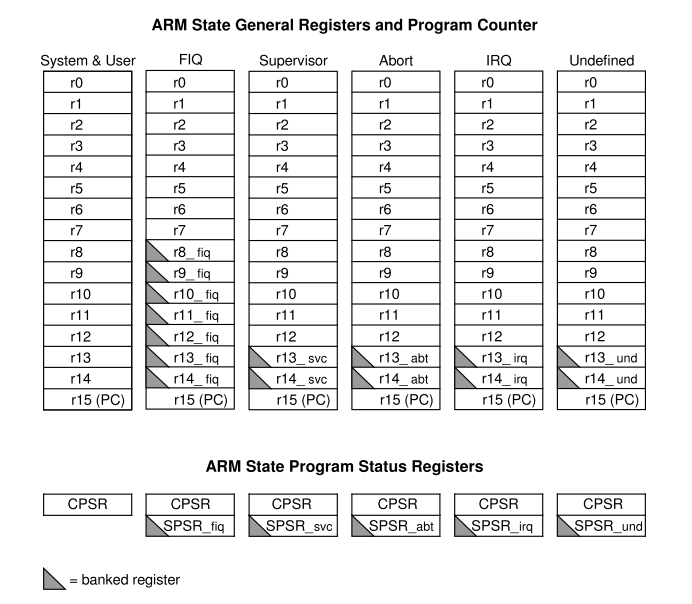


5,ARM 指令与Thumb指令

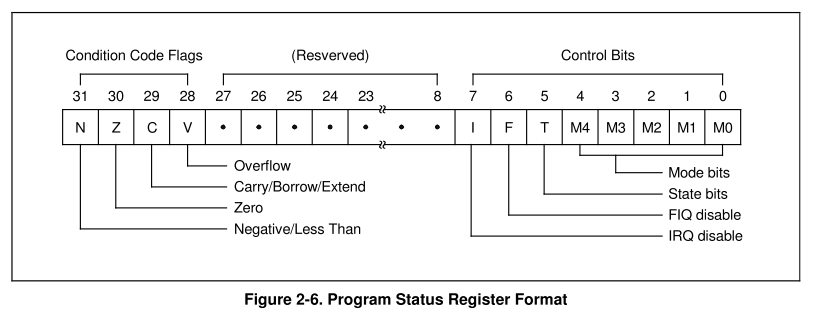
ARM占4Byte

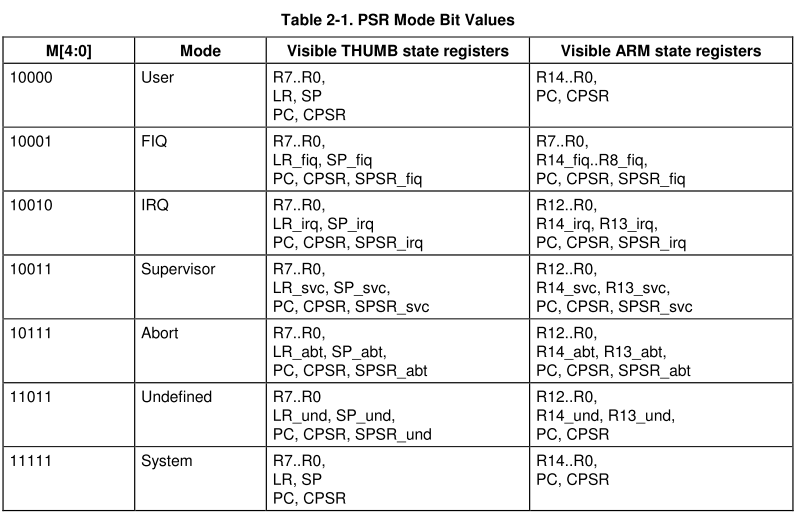
Thumb占2Byte

6 ARM State 通用寄存器



7，CPSR 寄存器





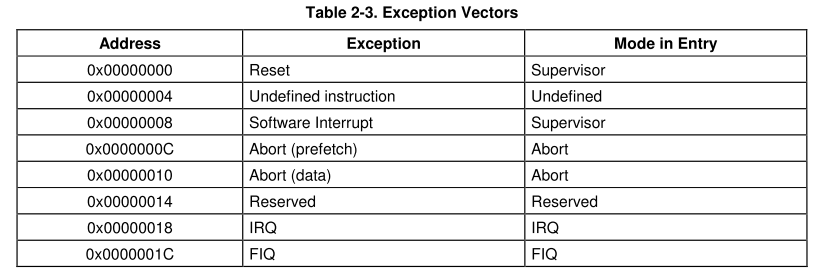
8, 进入中断/异常时ARM内核自动执行的动作

1）将返回地址保存在LR（R14）寄存器

2）将CPSR复制到SPSR

3）设置CPSR的模式位

4）设置PC值为对应的中断/异常向量表

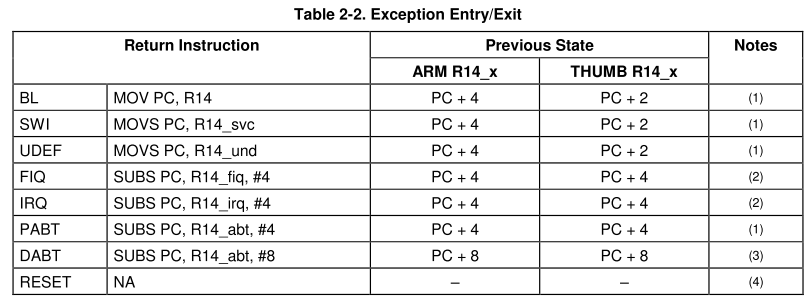


9，离开中断/异常时需要手动处理的步骤

1）将LR减去对应的值，再复制给PC

2）将SPSR复制到CPSR

3）清除中断标志位



10，执行Reset 中断/异常时ARM内核自动执行的步骤

1）复制PC到LR， 复制CPSR到SPSR， PC和CPSR的值是未定义的

2）将CPSR的 M[4:0]设置为10011(supervisor mode), I bit和 F bit置1，T bit清零

3）将PC设置为0

4）切到ARM状态