ARM的七种工作模式

一、ARM处理器7种工作模式（特权模式 特权模式异常模式）

1. 用户模式（USR）：正常程序执行模式，不能直接切换到其他模式
2. 系统模式（SYS）：运行操作系统的特权任务，与用户模式类似，但具有可以直接切换到其他模式等特权
3. 快中断模式（FIQ）：支持高速数据传输及通道处理，FIQ异常响应时进入此模式
4. 中断模式（IRQ）：用于通用中断处理，IRQ异常响应时进入此模式
5. 管理模式（SVC）：操作系统保护模式，系统复位和软件中断响应时进入此模式（由系统调用执行软中断SWI命令触发）
6. 中止模式（ABT）：用于支持虚拟内存和/或存储器保护，在ARM7TDMI没有大用处
7. 未定义模式（UND）：支持硬件协处理器的软件仿真，未定义指令异常响应时进入此模式
8. 表3-1 ARM处理器工作模式

<table>
<tr><th>处理器工作模式</th><th>特权模式</th><th>异常模式</th><th>说明</th></tr>
<tr><td>用户（user）模式</td><td></td><td></td><td>用户程序运行模式</td></tr>
<tr><td>系统（system）模式</td><td rowspan="7">该组模式下可以任意访问系统资源</td><td></td><td>运行特权级的操作系统任务</td></tr>
<tr><td>一般中断（IRQ）模式</td><td rowspan="6">通常由系统异常状态切换进该组模式</td><td>普通中断模式</td></tr>
<tr><td>快速中断（FIQ）模式</td><td>快速中断模式</td></tr>
<tr><td>管理（supervisor）模式</td><td>提供操作系统使用的一种保护模式，swi命令状态</td></tr>
<tr><td>中止（abort）模式</td><td>虚拟内存管理和内存数据访问保护</td></tr>
<tr><td>未定义指令终止（undefined）模式</td><td>支持通过软件仿真硬件的协处理</td></tr>
</table>

CPU的模式可以简单的理解为当前CPU的工作状态，比如：当前操作系统正在执行用户程序，那么当前CPU工作在用户模式，这时网卡上有数据到达，产生中断信号，CPU自动切换到一般中断模式下处理网卡数据（普通应用程序没有权限直接访问硬件），处理完网卡数据，返回到用户模式下继续执行用户程序。

特权模式

除用户模式外，其它模式均为特权模式（Privileged Modes）。ARM 内部寄存器 和一些 片内外设 在硬件设计上只允许（或者可选为只允许）特权模式下访问。此外，特权模式可以自由的切换处理器模式，而用户模式不能直接切换到别的模式。

异常模式

特权模式中除系统（system）模式之外的其他5种模式又统称为异常模式。它们除了可以通过在特权下的程序切换进入外，也可以由特定的异常进入。比如硬件产生中断信号进入中断异常模式，读取没有权限数据进入中止异常模式，执行未定义指令时进入未定义指令中止异常模式。其中管理模式也称为超级用户模式，是为操作系统提供软中断的特有模式，正是由于有了软中断，用户程序才可以通过系统调用切换到管理模式

7种工作模式介绍

（1）用户模式：

用户模式是用户程序的工作模式，它运行在操作系统的用户态，它没有权限去操作其它硬件资源，只能执行处理自己的数据，也不能切换到其它模式下，要想访问硬件资源或切换到其它模式只能通过软中断或产生异常。

（2）系统模式：

系统模式是特权模式，不受用户模式的限制。用户模式和系统模式共用一套寄存器，操作系统在该模式下可以方便的访问用户模式的寄存器，而且操作系统的一些特权任务可以使用这个模式访问一些受控的资源。

说明：用户模式与系统模式两者使用相同的寄存器，都没有SPSR（Saved Program Statement Register，已保存程序状态寄存器），但系统模式比用户模式有更高的权限，可以访问所有系统资源。

（3）一般中断模式：

一般中断模式也叫普通中断模式，用于处理一般的中断请求，通常在硬件产生中断信号之后自动进入该模式，该模式为特权模式，可以自由访问系统硬件资源。

（4）快速中断模式：

快速中断模式是相对一般中断模式而言的，它是用来处理对时间要求比较紧急的中断请求，主要用于高速数据传输及通道处理中。

（5）管理模式（Supervisor，SVC）：

管理模式是CPU上电后默认模式，因此在该模式下主要用来做系统的初始化，软中断处理也在该模式下。当用户模式下的用户程序请求使用硬件资源时，通过软件中断进入该模式。

说明：系统复位或开机、软中断时进入到SVC模式下。

（6）终止模式：

中止模式用于支持虚拟内存或存储器保护，当用户程序访问非法地址，没有权限读取的内存地址时，会进入该模式，linux下编程时经常出现的segment fault通常都是在该模式下抛出返回的。

（7）未定义模式：

未定义模式用于支持硬件协处理器的软件仿真，CPU在指令的译码阶段不能识别该指令操作时，会进入未定义模式。

说明：

1、用户模式外，其它6种模式称为特权模式。所谓特权模式，即具有如下权利：

a.MRS（把状态寄存器的内容放到通用寄存器）；

b.MSR（把通用寄存器的内容放到状态寄存器中）。

由于状态寄存器中的内容不能够改变，因此要先把内容复制到通用寄存器中，然后修改通用寄存器中的内容，再把通用寄存器中的内容复制给状态寄存器中，即可完成“修改状态寄存器”的任务。

2、剩下的六种模式中除去系统模式外，统称为异常模式。

除用户模式外，其余6种工作模式都属于特权模式
特权模式中除了系统模式以外的其余5种模式称为异常模式
大多数程序运行于用户模式
进入特权模式是为了处理中断、异常、或者访问被保护的系统资源

硬件权限级别：系统模式 > 异常模式 > 用户模式

快中断与慢中断区别：快中断处理时禁止中断

二、异常的优先级

异常类型	优先级	
复位	1（最高优先级）	优先级降低
数据中止	2	
FIQ	3	
IRQ	4	
预取指中止	5	
未定义指令	6	
SWI	7（最低优先级）	

三、存储器格式

1. 大端格式：高字节在低地址，低字节在高地址
2. 小端格式：高字节在高地址，低字节在低地址

四、ARM体系的CPU有两种工作状态

1. ARM状态
2. THumb状态

五、Linux操作系统与ARM工作模式

首先，ARM开发板在刚上电或复位后都会首先进入SVC即管理模式，此时、程序计数器R15-PC值会被赋为0x0000 0000；bootloader就是在此模式下，位于0x0000 0000的NOR FLASH或SRAM中装载的，因此、开机或重启后bootloader会被首先执行。

接着，bootloader引导Linux内核，此时、Linux内核一样运行在ARM的SVC即管理模式下；当内核启动完毕、准备进入用户态init进程时，内核将ARM的当前程序状态CPSR寄存器M[4:0]设置为10000、进而用户态程序只能运行在ARM的用户模式。

由于ARM用户模式下对资源的访问受限，因此、可以达到保护Linux操作系统内核的目的。

需要强调的是：Linux内核态是从ARM的SVC即管理模式下启动的，但在某些情况下、如：硬件中断、程序异常（被动）等情况下进入ARM的其他特权模式，这时仍然可以进入内核态（因为就是可以操作内核了）；同样，Linux用户态是从ARM用户模式启动的，但当进入ARM系统模式时、仍然可以操作Linux用户态程序（进入用户态，如init进程的启动过程）。

即：Linux内核从ARM的SVC模式下启动，但内核态不仅仅指ARM的SVC模式（还包括可以访问内核空间的所有ARM模式）；Linux用户程序从ARM的用户模式启动，但用户态不仅仅指ARM的用户模式