

第九章 习题及解答

9-5 Linux 系统的中断处理为什么要分为上半部和下半部？

答：操作系统的中断机制实现了 I/O 设备与 CPU 以及多进程之间的同时执行，大大提高了系统效率。

操作系统的中断处理程序是比较复杂的，而且在系统一级处理时是不允许打断的。如何提高中断处理的效率呢？如何解决处理时间短的要求和处理事务复杂性的矛盾？Linux 提出了一个很好的解决办法。Linux 系统将中断处理程序分为两部分，将中断响应后必须立即处理的工作即刻执行（而且其执行时必须关中断），而将更多的处理工作向后推迟执行。即将中断处理程序分为上半部（tophalf）和下半部（bottom half）。Linux 系统将中断处理程序分为上半部和下半部的目的是为了缩短关中断的时间，提高系统的处理能力。

9-7 在 Linux 系统中，增加一个新的系统调用需要做哪些工作？

答：在 Linux 系统中，增加一个新的系统调用需要做的工作包括如下几个方面。

(1) 编写一个新增加的功能的服务例程。

编写新增的服务例程，并加到内核中去，即在 `/usr/src/linux/kernel/sys.c` 文件中增加一个新的函数。

(2) 增加一个新的系统调用号。

在 linux 中，每个系统调用被赋予一个唯一的系统调用号。找到 linux 中定义系统调用号定义的文件（在 `include/asm-i386/unistd.h` 头文件中）。在此文件中按其规定的格式添加一项。

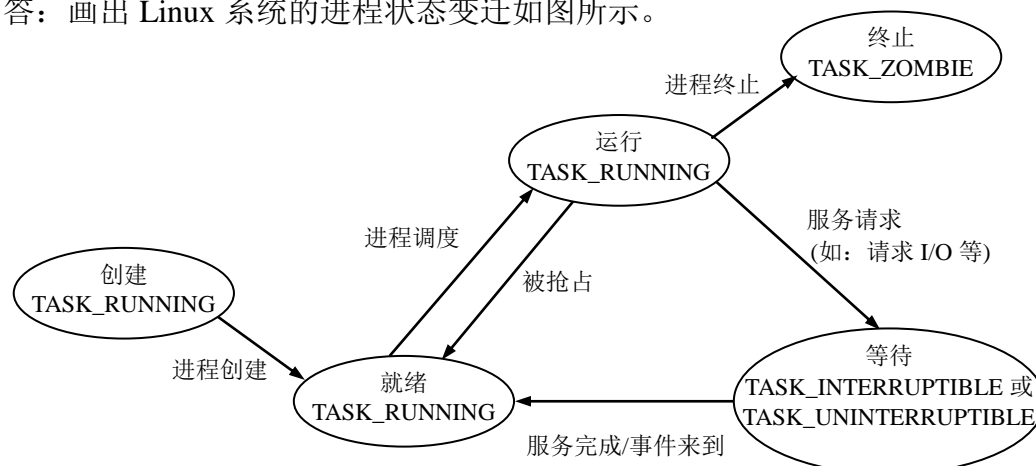
(3) 在系统调用表中登记新的系统调用号以及对应的服务例程。

系统调用表记录了内核中所有已注册过的系统调用，它是系统调用的跳转表，实际上是一个函数指针数组，表中依次保存所有系统调用的函数指针。找到 linux 中的系统调用表（Linux 系统调用表保存在 `arch/i386/kernel/` 下的 `entry.S` 中）。在此文件中按其规定的格式增加一个新的系统调用号以及对应的服务例程。

(4) 新增加的服务例程要为 Linux 系统接受，必须重新编译内核，生成新的包含新增服务例程的内核。

9-9 试画出 Linux 系统的进程状态变迁，并说明这些变迁可能的原因。

答：画出 Linux 系统的进程状态变迁如图所示。



(1) 进程创建

当系统或用户需要创建一个新进程时，调用 `fork()` 系统调用，被创建的新进程被置为就绪状态 `TASK_RUNNING`。

(2) 进程调度

当调度时机来到时，进程调度程序从进程运行队列中选择优先级最高的进程，将其投入运行，设置状态为运行状态。

(3) 被抢占

正在 CPU 上运行的进程，当其优先级低于处于就绪状态的某一个进程的优先级时，它被抢占而被迫让出 CPU 的控制权，此时，该进程的状态转为就绪状态。

(4) 进程等待

若正在运行的进程因等待某一事件而暂时不能运行下去时，进入相应的等待队列，设置为等待状态。

(5) 进程唤醒

当某个进程等待的原因撤销时，该进程被唤醒，将其从等待队列中移出，进入就绪队列。

(6) 进程终止

当正在运行的进程完成其任务时，通过 `exit()` 系统调用终止自己而进入终止状态。

9-12 Linux2.6 版本为了实现 $O(1)$ 级算法复杂度，采用了什么措施 ？

答：Linux 系统进程调度用的数据结构最重要的是运行队列结构，该结构给出了处理机上可运行进程的链表。该结构中包含一个称为优先级数组的结构数组。每个数组都表示一个可运行进程集合，包括两个重要信息：① 一个优先级位图；② 140 个双向链表头，每个链表对应一个可能的进程优先级队列。

Linux 系统采用优先调度策略。在 Linux2.6 版本的进程调度程序中，基于上述进程调度用数据结构，查找系统中优先级最高的进程这一问题转化为查找优先级位图中第一个置为 1 的位。找到这一位就是找到了最高优先级链表，即可确定优先级最高的、可运行的进程。由于优先级个数是定值，所以查找时间恒定。许多体系结构提供 `find_first_bit` 指令（字操作指令），找到第一个设置为 1 的位所花费的时间微不足道。这是保证 Linux 系统进程调度具有 $O(1)$ 级算法复杂度的关键所在。

9-29 打开文件系统调用 `open()` 返回的文件描述符 `fd` 的物理意义是什么 ？

答：打开文件系统调用 `open()` 返回的是文件描述符 `fd`，是一个打开文件在文件对象指针数组 `fd` 中的索引，该数组的索引就是文件描述符。用户程序对打开文件进行读、写时，直接使用其文件描述符。通常，数组的第一个元素（索引为 0）是进程的标准输入文件，数组的第二个元素（索引为 1）是进程的标准输出文件，数组的第三个元素（索引为 2）是进程的标准错误文件。