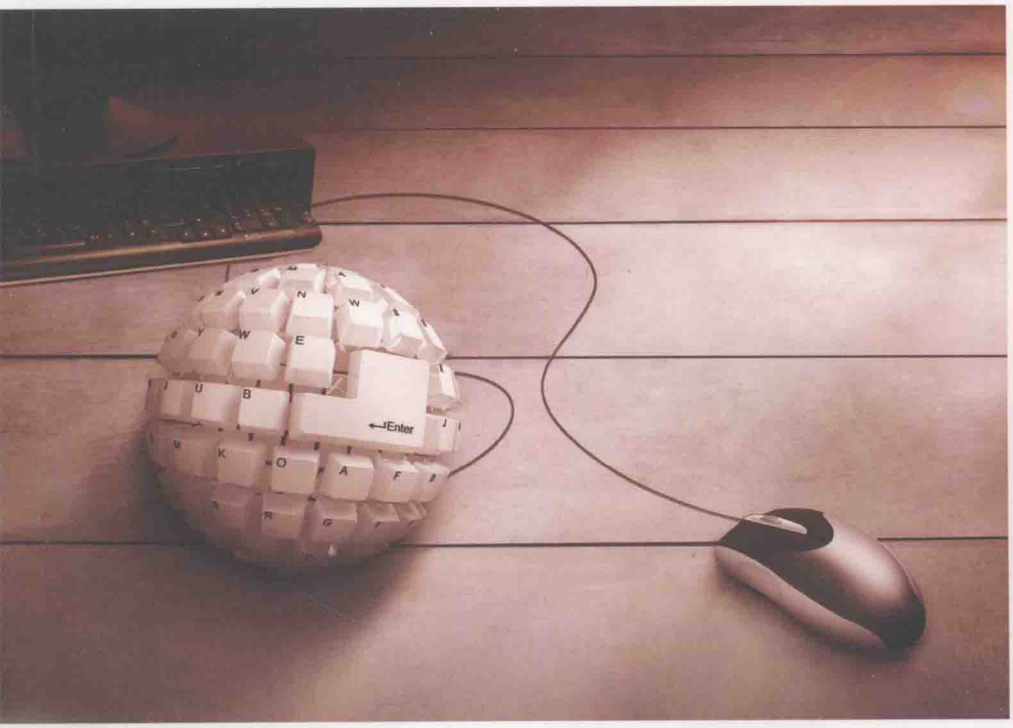
TEACHING MATERIALS
FOR COLLEGE STUDENTS
高等学校教材



计算机操作系统原理

张琼声 编著

中国石油大学出版社

TEACHING MATERIALS
FOR COLLEGE STUDENTS
高等学校教材

计算机操作系统原理

张琼声　编著

中国石油大学出版社

图书在版编目(CIP)数据

计算机操作系统原理/张琼声编著. —东营:中国石油大学出版社,2012.9

ISBN 978-7-5636-3799-7

Ⅰ. ①计… Ⅱ. ①张… Ⅲ. ①操作系统 Ⅳ. ①TP316

中国版本图书馆 CIP 数据核字(2012)第 195828 号

中国石油大学(华东)规划教材

书　　名:计算机操作系统原理

作　　者:张琼声

责任编辑:高　颖(电话 0532—86981531)

封面设计:青岛友一广告传媒有限公司

出 版 者:中国石油大学出版社(山东 东营　邮编 257061)

网　　址:http://www.uppbook.com.cn

电子信箱:shiyoujiaoyu@126.com

印 刷 者:青岛锦华信包装有限公司

发 行 者:中国石油大学出版社(电话 0532—86981532,0546—8392563)

开　　本:180 mm×235 mm　印张:19.25　字数:383 千字

版　　次:2012 年 9 月第 1 版第 1 次印刷

定　　价:31.00 元

内容提要

本书内容涵盖了实现操作系统内核的基本原理，其中第1章介绍了计算机操作系统的功能、发展、特征、应用领域等；第2章详细说明了什么是进程、线程，以及操作系统内核中断处理、时钟管理、系统调用功能实现的基本原理；第3章介绍了同步机制的必要性和实现技术；第4章阐述了进程调度的策略、算法、性能评价等方面的内容；第5章介绍了死锁及解决死锁的途径；第6章和第7章详细介绍了内存管理实现的原理；第8章结合实例对文件系统要解决的主要问题及实现的基本思路、安全性问题进行了介绍；第9章阐述了设备管理软件的构成、设备的控制方式、设备管理软件的功能及实现原理。本书尽量将原理与数据结构、算法、硬件特征相联系，用实例弱化操作系统原理的抽象感。

本书适用于计算机相关专业的本科生、大专生使用，也可作为计算机相关领域工作人员的参考书。

前言 PREFACE

“计算机操作系统原理”是计算机类专业的必修课程。学生通过本书的学习，能够掌握计算机操作系统的基本原理。

本书主要介绍实现操作系统内核的基本原理，其中第 1 章介绍了计算机操作系统的作用、功能、发展、特征、应用领域；第 2 章详细说明了什么是进程、线程，以及操作系统内核具有的中断处理、时钟管理、系统调用功能实现的基本原理；第 3 章介绍了同步机制的必要性和实现技术；第 4 章阐述了进程调度的策略、算法、性能评价方面的内容；第 5 章介绍了关于死锁及解决死锁的途径；第 6 章和第 7 章详细介绍了内存管理实现的原理；第 8 章结合实例对文件系统要解决的主要问题及实现的基本思路、系统安全性问题做了介绍；第 9 章介绍了设备管理软件的构成、设备控制方式、设备管理软件的功能及实现原理。本书的内容主要来自作者多年的授课经验和相关知识的积累，以及国内外相关教材和技术资料，已在书后列出，在此对这些文献的作者表示感谢。本书内容尽量使原理与数据结构、算法、硬件特征相联系，试图用实例削弱操作系统原理的抽象感，而且书中的实例采用 Tinix，Linux2.4 和 Linux2.6.11 源代码或简化的源代码编写。

由于作者能力有限，书中错误和不妥之处在所难免，敬请读者批评指正。

作　者

2012 年 3 月

目录 CONTENTS

第1章　操作系统概述

1.1　什么是操作系统?

操作系统(Operating System, OS)是一种复杂的系统软件,它提供计算机用户与计算机硬件之间的接口,并管理计算机软件和硬件资源。操作系统本身并不向用户提供功能,但是它为应用程序的运行提供平台,并使应用程序的编程变得简单、容易。操作系统如常用的 Windows XP,Unix,Linux 等,是由很多程序代码组成的复杂软件,可执行。我们常说的安装"系统",一般就是指安装"操作系统"。

操作系统是覆盖在裸机上的第一层软件,其他的支撑软件(编译程序、数据库管理系统)、应用程序都运行在操作系统之上。操作系统为这些软件提供运行环境。操作系统在计算机系统中的位置如图 1-1 所示。

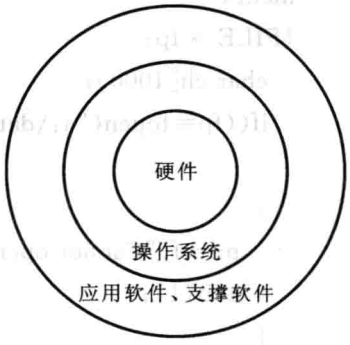


图 1-1　操作系统在计算机体系中的位置

下面从不同的角度说明什么是操作系统。

1.1.1　操作系统是计算机用户与计算机硬件之间的接口

为什么说操作系统是计算机用户与计算机硬件之间的接口?接口是两个不同组成部分的交接面。在计算机专业领域,接口分硬件接口(如 USB 接口、串口、并口)和软件接口(如 C 语言中的函数调用 printf ( ))。

计算机的所有功能最终都是由硬件的运转来实现的,但是通常使用的诸如 C 语言、Basic 语言、Fortran 语言所编写的高级语言程序中并没有直接操作硬件的语句、指令。那么为什么这些程序依然能够操纵显示器、打印机、绘图仪等硬件,并通过硬件对程序的执行得到执行结果,完成特定的功能呢?这是因为大多数应用程序是在操作系统上运行的,而对硬件的操纵过程都封装在操作系统的核心程序中。

例如,在用户程序中执行 printf ( ),并不需要在用户程序中写出直接操作打印机或显卡的指令,而是在操作系统上运行的用户程序通过接口 printf ( )调用操作系统中实现 printf ( )功能的、包括驱动硬件的指令的程序,从而实现对硬件的操作。

因为操作系统对硬件的抽象为应用程序提供了运行的环境，编写在操作系统之上运行的程序就简单多了。例如，在 Windows 下用 C 语言编写从软盘读文件的程序，只需要 fopen ( )，fread ( )和 fclose ( )几个函数调用，但是在没有操作系统的裸机上要写一个可以读文件的程序就要复杂得多，必须清楚要读的文件在软盘上的物理位置，即文件在哪个磁道，哪个盘面，第几个扇区，还要知道读入的数据放在内存的什么位置，然后调用相应的指令序列完成读操作。

下面看两个例子。

例 1：用 C 语言编写在操作系统上运行读软盘 A 上的文件 data. dat 的程序。

```
#include <stdio. h>
#include <stdlib. h>
  main ( )
  {FILE * fp;
    char ch[1000];
    if((fp=fopen("a:\data. dat","r"))==NULL)
                                          /* 打开要读的文件 data. dat */
    {
      printf("Cannot open file strike any key exit!");
      exit(1);
    }
    fread(ch, sizeof(int), 30, fp);          /* 从文件中读 */
    fclose(fp);
  }
```

例 2：在无操作系统的裸机上运行读软盘的程序。该程序省略了根据文件名 data. dat 查找根目录获取该文件所在扇区号部分的程序，所列程序清单仅包括初始化软驱和从文件所在的逻辑扇区号计算出文件扇区所在的物理位置、定位磁头、启动读操作部分的程序代码。类似功能的完整程序代码将在附录 1 中给出。

```
xor ah, ah              ;对 ah,dl 清零
xor dl, dl              ;软驱复位(当 ah=00h,dl=驱动器号时软驱复位。
                        ;其中,0 表示 A 盘)
int 13h                 ;BIOS 功能调用 int 13h 中断
mov ax, [wSectorNo]     ;将扇区号送 ax 寄存器
mov cl, 1               ;cl 中放入 1,读 1 个扇区
call ReadSector         ;调用 ReadSector,读出扇区内容
```

```
...
;怎样根据逻辑扇区号求物理扇区(包括柱面号、磁头号、扇区号)在磁盘中的位置
;设逻辑扇区号为 x,y 是 x/每磁道扇区数的商,z 是 x/每磁道扇区数的余数
;  x/每磁道扇区数 => { 商 y => { 柱面号=y>>1
;                                磁头号=y&1
;                    { 余 z => 起始扇区号=z+1
push bp
mov bp, sp
sub esp, 2                  ;辟出两个字节的堆栈区域保存要读的扇区数:
                            ;byte [bp-2]
mov byte [bp-2], cl         ;将 cl 中的值放入 bp-2 中,cl 为要读的扇区数
push bx                     ;保存 bx
mov bl, [BPB_SecPerTrk]     ;将每个磁道的扇区数值放入 bl 中
div bl                      ;y 在 al 中,z 在 ah 中(ax/bl 商送 al,余数送 ah)
inc ah                      ;z++(ah 中第一次存放为起始扇区号,然后
                            ;进行扇区号递加)
mov cl, ah                  ;起始扇区号送 cl 寄存器(存放当前扇区号)
mov dh, al                  ;dh<-y(将 y 的值送入 dh)
shr al, 1                   ;y>>1 (逻辑右移 1 位)
mov ch, al                  ;柱面号送 ch
and dh, 1                   ;dh&1=磁头号
pop bx                      ;恢复 bx
;至此,"柱面号,起始扇区,磁头号"全部得到
mov dl, [BS_DrvNum]         ;驱动器号(0 表示 A 盘)
GoOnReading:
mov ah, 2                   ;int 13h 功能调用 ah=2,表示功能号为 2,进行
                            ;读操作
mov al, byte [bp-2]         ;读 al 个扇区
int 13h                     ;BIOS 功能调用 int 13h 中断
```

上面这段程序,先执行了一次 int 13h 中断,对软盘进行初始化,通过对 ah 和 dl 清零实现软驱复位;然后根据逻辑扇区号计算出文件 data.dat 的第一个扇区在软盘的绝对物理地址,包括 data.dat 的第一个逻辑扇区在软盘上的柱面号、磁头号和起

始扇区号;最后将这些地址信息和功能号写入指定的寄存器,再次执行 int 13h 中断,执行一次读操作。

大多数应用软件的开发者可以在不了解计算机硬件结构和工作原理的情况下编写大量的应用程序,就是因为有了操作系统。通过以上两个实例的对比可以看出,操作系统屏蔽了对硬件操作的细节,提供了计算机用户与计算机硬件之间的接口,通过这个接口可使应用程序的开发变得简单。

操作系统必须完成两个主要目标:

(1) 与硬件部分相互作用,为包含在硬件平台上的所有低层可编程部件提供服务。

例如,要使用显示器,必须给驱动显示器工作的一些寄存器赋值,以便让显示器"知道自己要以什么样的背景颜色、前景颜色,显示什么内容"。

(2) 为运行在计算机系统上的应用程序(即用户程序)提供执行环境。

大多数高级语言的应用程序是不能在裸机上运行的,必须运行在操作系统上,由操作系统为应用软件的执行分配必要的软件和硬件资源,并对这些资源进行有效的管理。

1.1.2 操作系统是计算机资源的管理者

现代计算机系统的一个重要特点就是支持多任务,即允许在同一个系统内同时运行多个用户程序。当多个用户程序共同使用计算机硬件和软件资源时,需要操作系统对这些资源进行有效的管理,一方面保证用户程序的顺利执行,另一方面使计算机系统资源得到尽可能高效的利用,保证计算机系统的高性能。操作系统所管理的资源主要包括处理机、内存、设备、文件,在网络操作系统中还包括带宽。

1) 处理机管理

本书主要介绍多任务操作系统的实现原理。由于程序的执行必须依靠处理机,而任何时刻处理机都只能执行一个程序流,因此,在单处理机系统中执行多个程序流时,必须由操作系统的处理机调度程序来管理处理机的分配,以使多个程序共享处理机。从宏观上看,多个程序能同时顺利执行。在多处理机系统中,也需要操作系统对多个处理机在多任务的情况下进行有效的管理。

2) 内存管理

在多任务系统中,内存不再独占资源,而是被多个用户程序共同占用,如何为多个用户程序分配内存并使不同用户程序的地址空间互不干扰,如何在程序执行完毕回收其所占内存,以及如何完成逻辑地址到物理地址的转换都是操作系统内存管理程序所要完成的功能。

3) 设备管理

设备管理主要完成用户的 I/O 请求,为用户分配 I/O 设备、管理 I/O 缓存、驱动

I/O设备。

4）文件管理

计算机系统把大量的需要长时间保存的数据信息以文件的形式存放在外存储设备(如硬盘、光盘、磁带、U盘等)中，操作系统通过自己的文件管理程序完成外存空间的分配、回收，文件的按名存取，文件的组织、共享与保护等功能。

1.2 操作系统的发展

操作系统是在解决计算机系统所面临的问题的过程中诞生并自然发展的。操作系统的发展依次经历了无操作系统、单道批处理系统、多道程序系统(多道批处理系统、分时系统)等过程。随着计算机应用领域的扩大、计算机体系结构的多样化，又出现了微机操作系统、网络操作系统、实时操作系统。本节以计算机发展的时间为线索，说明操作系统是如何在解决计算机系统面临的问题的过程中自然发展的。

1.2.1 无操作系统

第一代计算机(1945—1955年)使用电子管作为主要的电子器件，用插件板上的硬连线或穿孔卡片表示程序，没有存储程序的内存，无操作系统。

以1946年诞生于宾夕法尼亚大学的第一台实用电子计算机“埃尼克”(ENIAC)为例，它没有真正的内存，只用20个字节的寄存器来存储数字，每个字节10位，也就是只有200位的存储容量，无法支持存储程序。操作系统需要常驻内存，而当时的情况是不仅在硬件技术上无法支持操作系统的实现，而且在理论研究领域也没有操作系统的概念。使用“埃尼克”的方式是：程序员用电路的连接方式表示算法，一旦算法发生改变，必须变更电路的连接方式。程序员在使用“埃尼克”进行计算工作之前，先用插件板上的电路连接表示出计算的算法程序，同时预约好一段上机时间，然后到机房中将插件板接到计算机中，开启计算机开关，开始一次运算。显然，一个用户程序进入计算机和退出计算机系统都需要人工干预，计算机无法自动完成程序的加载和卸载，因此整个计算机系统处于“运行—因等待人工操作暂停—运行”这样一种不能连续自动工作的状态。

1.2.2 单道批处理系统

第二代计算机(1955—1965年)使用的主要电子器件是晶体管，开始使用磁性存储设备，内、外存容量增加，计算机运算速度提高，出现了早期的单道批处理系统。

在单道批处理系统出现以前，使用计算机的方式是：操作员把作业挂到输入设备上，启动机器，运算结束取出结果。典型的情况是：程序员用穿孔卡片表示程序和数据，当需要运行这些穿孔卡片表示的作业时，将卡片盒交给输入室的计算机操作员，