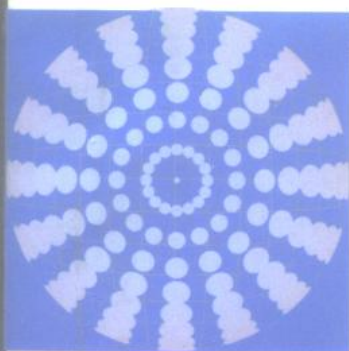
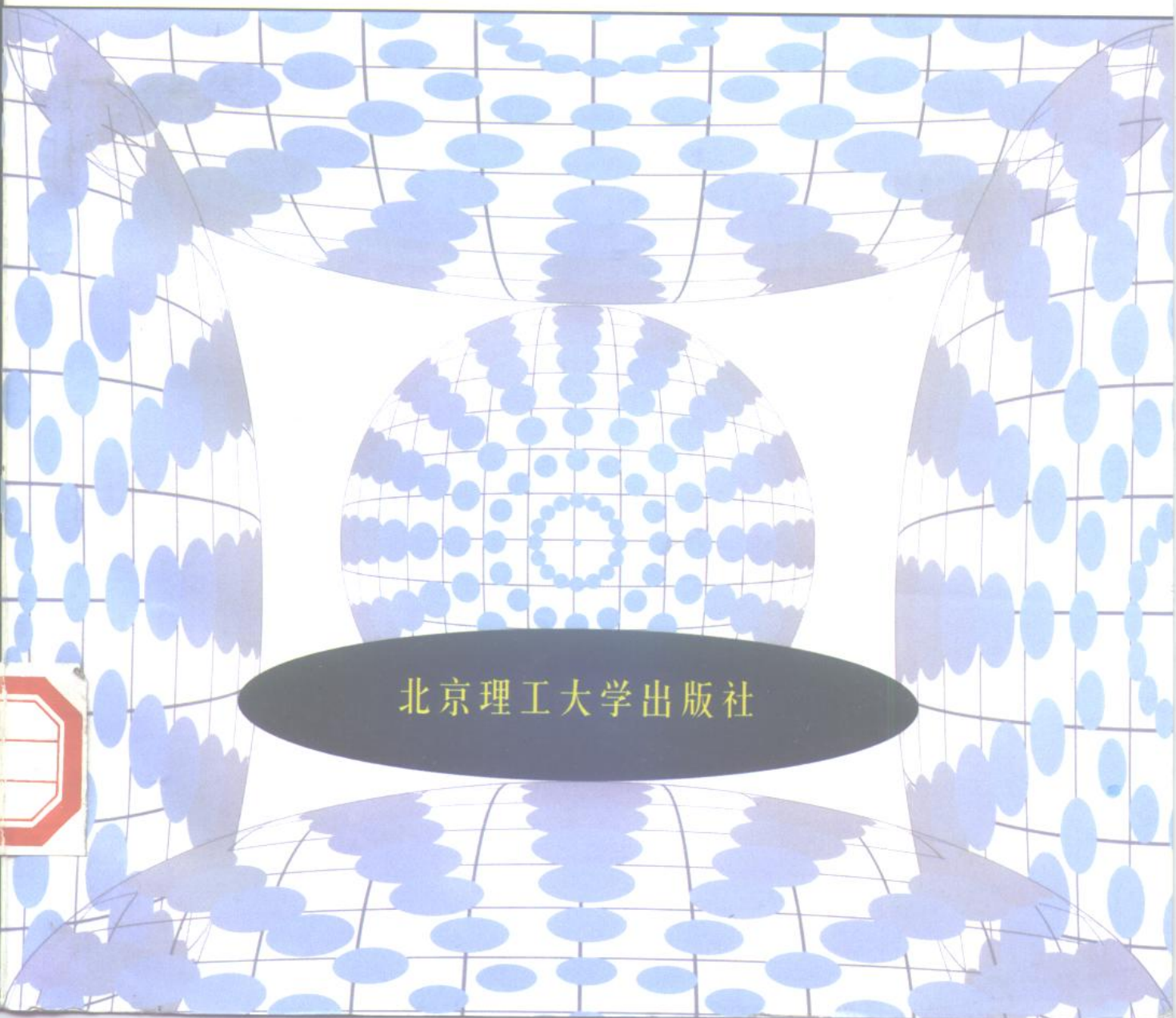


张丽芬 编著



操作系统原理与设计



北京理工大学出版社



操作系统原理与设计

张丽芬 编著

北京理工大学出版社

内 容 简 介

计算机“操作系统”是计算机系统的必不可少的基本系统软件,是计算机专业学生的必修课。

本教材分四篇共十五章。第一篇分为六章,介绍操作系统的基本概念;操作系统五大组成:进程管理、作业管理、存贮器管理、文件系统和设备管理的实现原理及各种管理算法。第二篇分为七章,以现在国际上流行的 UNIX 系统 V 的操作系统为实例,较详细地介绍了操作系统各部分的具体实现技术。包括进程管理、存贮器管理、文件管理、设备管理、中断和异常处理、进程之间的高级通信等。第三篇介绍了微型机单用户单任务 MS-DOS 操作系统。第四篇介绍了网络和分布式处理的通信结构、分布式进程管理及算法。

本书可作为计算机专业及相关的计算机应用专业(电子工程和自动控制类)的教科书,也可作为从事计算机工作的科技人员学习计算机操作系统的参考书。

图书在版编目(CIP)数据

操作系统原理与设计/张丽芬编著. —北京:北京理工大学出版社,1997. 12

ISBN 7-81045-329-7

I. 操作… II. 张… III. 操作系统(软件)-概论 IV. TP316

中国版本图书馆 CIP 数据核字(97)第 22774 号

责任印制:刘季昌 责任校对:陈玉梅

北京理工大学出版社出版发行

(北京市海淀区白石桥路 7 号)

邮政编码 100081 电话 (010)68912824

各地新华书店经售

北京房山先锋印刷厂印刷

*

787×1092 毫米 16 开本 22.25 印张 535 千字

1997 年 12 月第 1 版 1997 年 12 月第 1 次印刷

印数:1-3000 册 定价:25.50 元

※图书印装有误,可随时与我社退换※

前 言

操作系统是计算机系统中不可缺少的基本系统软件,主要用来管理和控制计算机系统的软、硬件资源,提高其利用率,并为用户提供一个方便、灵活、安全、可靠的使用计算机的工作环境。

计算机操作系统不仅是计算机有关专业的核心骨干课程,也是从事计算机应用开发人员的必不可少的知识。

本书作者从1979年以来一直从事操作系统教学和科研工作,对多种机器的操作系统,如国产的实时操作系统(130~140机的RTOS)、200系列的操作系统、微机上的CP/M、MS-DOS及世界流行的UNIX系统、系统V都有较系统、全面的了解,特别是对DOS和UNIX进行了深入的剖析和专题研究,积累了大量的知识素材和实际经验。为填补我校教材空白,在前面工作的基础上,又查阅了大批国内外有关资料,编写了此书。

本书除对操作系统原理作详细叙述外,针对学生学习操作系统感到抽象、难懂的情况,在原理介绍的基础上,插入了大量机器的实例,特别是对UNIX系统V进行了深入的介绍。有些关键技术还给出了源程序,以帮助学生理解和掌握。

本书共分四篇15章,第一篇第六章介绍计算机操作系统的基本原理,包括进程管理、作业管理、存贮器管理、文件管理和设备管理;第二篇七章较详细地介绍了UNIX系统V操作系统的原理和设计技术。包括进程管理、存贮器管理、文件系统、输入输出子系统、进程之间的通信、中断和异常处理、UNIX系统V的系统初始化;第三篇介绍了MS-DOS操作系统的实现,包括MS-DOS引导、进程、内存、文件和设备的管理技术;第四篇介绍了80年代以来发展起来的网络和分布式处理系统。主要包括网络的通信结构、服务器管理、进程迁移、进程的同步和通信,以及分布式处理中解决死锁等的一些调度算法。

本书在编写完后,又请我校的彭一苇、高永峰、林国璋教授分别对有关章节进行审阅和指导,在此基础上又进行了修改。

本书的主审是北京工业大学计算机学院,高永慧教授。高教授对本书认真细致逐章逐节地审阅了全部书稿,并提出了不少宝贵建议。在此对高永慧、彭一苇、高永峰、林国璋教授表示诚挚的感谢。

由于作者水平所限,书中一定还有许多不妥之处,恳切希望读者予以指正。

编 著 者

1997年1月于北京理工大学

目 录

第一篇 操作系统基本原理

第一章 操作系统概论	(1)
1.1 计算机系统的组成	(1)
一、硬件	(1)
二、实用程序	(1)
三、应用程序	(2)
四、操作系统	(2)
1.2 操作系统的形成与发展	(2)
一、顺序处理	(2)
二、简单的批处理	(3)
三、多道成批处理系统	(5)
四、分时系统	(8)
五、实时系统	(10)
六、操作系统的功能和特性	(11)
七、操作系统的进一步发展	(12)
1.3 小结	(14)
习题	(14)
第二章 进程管理	(15)
2.1 进程的引入和概念	(15)
一、程序的顺序执行	(15)
二、程序的并行执行	(15)
三、进程的概念	(17)
2.2 进程的描述	(18)
一、进程控制块	(18)
二、进程的状态	(18)
三、进程队列	(19)
2.3 进程的控制	(20)
一、创建原语	(20)
二、撤消原语	(21)
三、阻塞原语	(21)
四、唤醒原语	(21)
五、挂起原语	(21)
六、解挂原语	(22)
2.4 进程低级通信	(22)
一、进程之间的互斥	(22)
二、解决进程之间互斥的方法	(23)

三、进程之间的同步	(25)
四、信号量	(25)
2.5 进程的高级通信	(28)
2.6 进程调度	(30)
一、进程调度的功能	(30)
二、进程调度的方式和调度时机	(31)
三、进程调度算法	(31)
2.7 死锁	(32)
一、资源的特性	(33)
二、死锁的定义	(33)
三、死锁产生的必要条件	(33)
四、解决死锁的方法	(34)
2.8 小结	(42)
习题	(43)
第三章 作业管理	(45)
3.1 作业管理	(45)
一、作业和作业步	(45)
二、用户与操作系统的接口	(45)
三、作业管理的功能	(45)
3.2 作业输入	(46)
一、作业提交形式	(46)
二、作业输入	(46)
3.3 作业调度	(47)
一、作业的状态及其转换	(47)
二、作业调度	(47)
3.4 作业控制	(49)
一、脱机作业控制	(49)
二、联机作业控制	(50)
3.5 系统调用	(50)
一、系统调用的定义	(50)
二、系统调用的执行	(51)
3.6 小结	(51)
习题	(52)
第四章 存贮器管理	(54)
4.1 引言	(54)
一、存贮器管理功能	(54)
二、存贮器管理中使用的几个概念	(54)
4.2 单用户单道程序的存贮分配	(56)
4.3 多用户多道程序的存贮分配——分区分配	(57)
一、固定式分区	(57)
二、可变式分区	(58)
三、分区管理的存贮器保护	(63)
四、分区管理的优缺点	(64)

五、多重分区管理	(64)
4.4 复盖与交换技术	(65)
一、复盖	(65)
二、交换	(66)
4.5 页式存贮管理	(67)
一、页式管理的实现原理	(67)
二、页式动态地址变换	(68)
三、快表和联想存贮器	(68)
四、页式管理的主存分配与回收	(70)
4.6 段式存贮管理	(70)
一、段式管理的实现原理	(70)
二、段式动态地址变换	(72)
三、段式管理的存贮保护	(72)
四、段式与页式管理的比较	(72)
4.7 虚拟存贮器	(73)
一、页式虚拟存贮管理	(74)
二、段式虚拟存贮管理	(82)
三、段页式虚拟存贮管理	(86)
4.8 小结	(87)
习题	(87)
第五章 文件系统	(90)
5.1 文件和文件系统	(90)
一、文件	(90)
二、文件系统	(91)
5.2 文件的结构和存取方法	(91)
一、文件的逻辑结构	(92)
二、文件的存取方法	(92)
5.3 文件的物理结构和存贮介质	(93)
一、文件的物理结构	(93)
二、文件存贮设备	(96)
5.4 文件目录	(97)
一、一级目录结构	(97)
二、二级目录结构	(100)
三、多级目录结构	(100)
5.5 文件存贮器存贮空间的管理	(102)
5.6 文件的共享、保护与保密	(103)
一、文件的共享	(103)
二、文件的保护	(104)
三、文件的保密	(105)
5.7 文件的操作命令	(107)
5.8 小结	(108)
小结	(108)
第六章 设备管理	(110)

6.1 I/O 硬件组成	(110)
一、I/O 设备	(110)
二、设备控制器	(110)
三、直接存取器访问(DMA)	(112)
四、通道方式	(113)
6.2 I/O 软件的组成	(114)
一、I/O 软件的目标	(114)
二、中断处理程序	(115)
三、设备驱动程序	(115)
四、设备独立的软件	(116)
五、用户空间的 I/O 软件	(117)
6.3 磁盘管理	(118)
一、磁盘的物理性能	(118)
二、磁盘调度算法	(119)
三、磁盘调度算法的比较	(122)
四、磁盘的错误处理	(122)
6.4 终端管理	(123)
一、终端硬件	(123)
二、终端软件	(126)
6.5 时钟管理	(129)
一、时钟硬件	(130)
二、时钟软件	(130)
6.6 小结	(132)
习题	(132)

第二篇 UNIX 系统 V 的实例研究

第七章 UNIX 系统 V 的进程管理	(134)
7.1 UNIX 系统的进程组成	(135)
一、UNIX 系统的进程映象	(135)
二、UNIX 系统的进程控制块	(135)
7.2 进程的状态及其转换	(141)
7.3 UNIX 系统进程的上下文	(142)
7.4 UNIX 系统的进程控制	(144)
一、进程的创建 fork()	(144)
二、进程的终止 exit()	(150)
三、进程的等待 wait()	(152)
四、进程的睡眠和唤醒(sleep()和 wakeup())	(153)
五、进程执行其它程序 exec()	(155)
六、进程控制的实例分析	(157)
7.5 UNIX 系统的进程调度	(158)
一、进程优先级	(158)
二、进程调度中的三个标志	(159)
三、进程调度的时机	(160)

四、进程调度的功能	(160)
7.6 进程的换入与换出	(161)
第八章 UNIX 系统 V 存储器管理	(164)
8.1 虚存空间的划分	(164)
一、系统空间布局	(164)
二、VAX-11 机的地址结构	(165)
8.2 存储器管理中的几个数据结构	(165)
一、页表	(165)
二、系统空闲页表(SPt)区的管理	(166)
三、主存位示图页表	(166)
四、交换区空闲空间的管理	(167)
8.3 主存分配与回收	(167)
一、主存的分配与释放函数	(167)
二、进程页表区及主存的分配与释放	(172)
8.4 地址转换	(175)
一、系统空间的地址转换	(175)
二、进程空间的地址转换	(175)
8.5 请求调页处理	(177)
一、请求调页使用的数据结构	(177)
二、请求调页处理	(178)
第九章 UNIX 系统 V 的文件系统	(181)
9.1 UNIX 文件系统的逻辑结构	(181)
9.2 UNIX 文件系统中所用的数据结构	(181)
一、文件卷的结构	(181)
二、专用块结构	(182)
三、UNIX 系统文件目录与索引节点	(183)
四、磁盘 i 节点的分配与回收	(184)
五、空闲盘块的分配与回收	(186)
六、UNIX 系统使用的打开文件数据结构	(187)
9.3 UNIX 文件系统的系统调用命令	(189)
一、文件的创建、打开、删除命令	(189)
二、文件的读写命令	(191)
三、文件关闭的系统调用命令(close)	(192)
四、文件链接系统调用命令(link)	(193)
五、改变文件读写指针的系统调用命令(seek)	(193)
六、解除链接的系统调用命令(unlink)	(194)
9.4 UNIX 文件卷的安装与拆卸	(194)
一、安装表与安装过程	(194)
二、文件卷的拆卸	(195)
第十章 UNIX 系统 V 的输入/输出子系统	(197)
10.1 与设备驱动程序的接口	(197)
10.2 块设备管理	(197)
一、系统缓冲池及各种队列的管理	(198)

二、缓冲区的分配与释放	(201)
三、块设备管理中的几个数据结构	(203)
四、块设备的驱动程序	(209)
10.3 字符设备管理	(213)
一、字符设备中使用的数据结构	(213)
二、字符设备的缓冲技术	(217)
三、终端设备的输入输出字符处理及设备驱动	(223)
第十一章 UNIX 系统进程之间的通信	(226)
11.1 管道(pipes)	(226)
一、创建无名管道的系统调用	(226)
二、创建一个有名管道	(227)
11.2 消息缓冲机制	(230)
一、消息缓冲使用的数据结构	(230)
二、消息缓冲的系统调用	(232)
三、消息缓冲的通信过程	(234)
11.3 共享内存段机制	(235)
一、共享内存段使用的数据结构	(235)
二、共享内存段的系统调用	(236)
11.4 信号量机制	(239)
一、信号量机制中的数据结构	(239)
二、信号量机制的系统调用	(241)
第十二章 中断和异常处理	(244)
12.1 中断和异常处理的硬件基础	(244)
一、中断源	(244)
二、中断装置	(244)
三、中断优先级	(245)
四、处理机的访问方式	(245)
五、处理机的中断优先级	(246)
六、异常和中断向量	(247)
12.2 中断和异常处理	(249)
一、中断处理	(249)
二、异常处理	(252)
12.3 软中断	(255)
一、软中断的处理方式	(256)
二、软中断中使用的几个函数的分析	(256)
三、软中断信号机制的应用	(258)
第十三章 UNIX 系统 V 的初始化	(260)
一、初启代码 start.c	(260)
二、系统空间的初始化程序 mlsetup.c	(261)
三、主程序 main.c	(262)

第三篇 MS-DOS 操作系统

第十四章 MS-DOS 操作系统	(263)
-------------------------------	-------

14.1	MS-DOS 操作系统发展历史	(263)
14.2	MS-DOS 的基本组成和功能	(264)
	一、DOS 的基本结构	(264)
	二、DOS 的启动	(265)
14.3	DOS 的功能服务	(267)
	一、DOS 的中断类型	(267)
	二、中断向量表和中断处理	(268)
	三、系统调用功能	(270)
14.4	DOS 的主存管理	(275)
	一、DOS 主存管理的特点	(276)
	二、DOS 主存空间划分	(277)
	三、DOS 常规主存管理	(278)
	四、高位主存区(HMA)和扩展主存管理	(280)
	五、上位主存区(UMB)管理	(280)
14.5	DOS 的进程管理	(281)
	一、程序段前缀(PSP)	(281)
	二、环境块(EVB)	(283)
	三、进程终止处理	(284)
	四、严重错误处理	(285)
14.6	DOS 的磁盘管理	(286)
	一、磁盘驱动器	(286)
	二、驱动器参数块	(289)
	三、设备驱动程序	(291)
	四、磁盘缓冲区管理	(295)
14.7	DOS 的文件系统	(297)
	一、DOS 的文件目录	(297)
	二、DOS 的文件分配表(FAT)	(300)
	三、DOS 的文件操作	(302)
	四、DOS 的文件共享	(306)
14.8	DOS 的网络重定向	(309)
	习题	(309)

第四篇 网络与分布式处理

第十五章	网络与分布式处理	(311)
15.1	分布式系统概述	(311)
	一、通信结构	(312)
	二、开放式系统互连通信结构	(315)
15.2	网络服务器	(318)
	一、服务器的结构	(319)
	二、磁盘和文件服务器	(320)
	三、文件高速缓冲存储器的一致性问题的	(320)
	四、打印机服务器	(321)
	五、调制解调器服务器	(322)

15.3 进程迁移	(322)
一、进程迁移机制	(322)
二、迁移处理	(324)
15.4 分布式进程通信	(325)
一、信息传送机制	(325)
二、远程过程调用	(326)
三、确定分布式系统的全局状态	(328)
15.5 分布式进程管理	(331)
一、分布式互斥	(331)
二、分布式算法	(334)
15.6 分布式进程死锁问题	(337)
一、资源分配中的死锁	(337)
二、消息通信中的死锁	(338)
15.7 小结	(341)
参考文献	(342)

第一篇 操作系统基本原理

第一章 操作系统概论

1.1 计算机系统的组成

随着计算机技术的迅速发展,计算机系统的硬件和软件资源愈来愈丰富。从功能上,可把整个计算机系统划分为四个层次:硬件、操作系统、实用程序和应用程序,如图 1.1 所示。这四层的关系表现为一种单向调用关系,即外层的软件必须以事先约定的方式调用内层软件或硬件提供的服务。通常把这种约定称为界面或接口(interface)。下面简单介绍一下各个层次的特点。

一、硬件

硬件层是由计算机的硬件资源组成,它包括中央处理机(简称 CPU 也称处理器)、存贮器(包括主存和辅助存贮器:磁盘、磁带等)和输入输出设备。其中输入设备有卡片输入机、键盘、鼠标、图形扫描仪等;输出设备有卡片输出机、显示器、打印机、绘图机等。这种不加任何扩充的计算机称为裸机。一个裸机的功能即使很强,用户也很难使用,因为这些裸机的部件是通过执行机器指令来实现计算和输入、输出功能的。

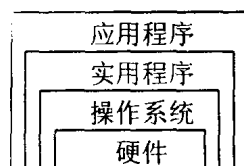


图 1.1 计算机系统的组成

二、实用程序

实用程序层包括下面一些程序,且它们通常是驻留在磁盘上的。

1. 各种语言编译程序

语言编译程序包括高级语言和汇编语言编译程序。高级语言程序是独立于机器硬件的,具有较好的可移植性。汇编语言是与机器硬件密切相关的,因此,不可移植。

2. 文本编辑程序

文本编辑程序用来建立和修改用户的源程序或其它文本数据的,这类编辑程序种类繁多,功能各异。有面向行的和面向全屏幕的。

3. 调试程序

调试程序又称排错程序,它用来帮助用户调试程序,从而方便地找出程序中的逻辑错误。

4. 连接装配程序

连接装配程序把用户分别编译好的各目标程序连接装配成一个可执行的程序。

还有一些实用程序,如标准过程和函数、系统诊断程序、分类合并程序、文件加密、解密程序等。总之,这些实用程序为用户使用计算机系统提供了一个良好环境。

三、应用程序

应用程序是计算机系统的最外层软件,它主要负责解决用户的实际问题。这些程序通常由用户或专门的软件公司编制。这类软件比较丰富,如各种数据库管理软件、计算机辅助设计软件,各种事务处理软件等。

四、操作系统

从宏观上讲,我们使用计算机的整个系统。而从微观上讲,我们使用计算机的各种硬件资源。用存储器保存程序和数据,用 CPU 完成各种操作运算,通过外围设备实现计算机与外部世界的信息交换,即完成输入输出工作,用辅助存储器保存大量的临时和永久性的信息文件。另外,还要用到各种软件资源,如上述的各种实用程序和应用程序,完成用户程序的计算工作。由谁来组织和管理这些软硬件资源,使之协调一致并高效地完成各种用户的计算工作呢?这就是操作系统。

操作系统究竟是什么,又如何定义它?为了深入全面理解操作系统的含义,我们从以下不同方面来阐述:

① 从计算机系统设计者的角度看,操作系统是一个负责管理和控制计算机软硬件资源的控制机构,它由一系列程序模块组成。它依据设计者设计的各种调度和管理策略,对计算机的软硬资源进行分配、管理和调度,合理地组织计算机的工作流程,从而提高系统资源的利用效率。由此可认为,操作系统是计算机资源的管理系统。

② 从用户角度看,配上操作系统的计算机是一台比裸机功能更强、使用方便简单的虚拟机。也即,它是用户与计算机裸机之间的一个接口界面,用户是通过它来使用计算机的。

由此可见,操作系统是现代计算机系统不可缺少的关键部分。计算机系统越复杂,操作系统越显得重要。为此,有必要了解操作系统的组成和功能,以便更好地利用计算机系统进行系统和应用开发。这正是我们这本书要讨论的主题。

1.2 操作系统的形成与发展

一、顺序处理(手工操作阶段)

早期的计算机(从 1940 年~50 年代中期),程序员直接与计算机硬件打交道,没有操作系统。计算机由 CPU、主存、某种类型的输入设备(卡片输入机)、一台打印机,以及装有显示灯、乒乓开关的操作控制台组成。用机器代码编写的程序通过输入设备装入计算机,再由程序员从控制台上通过设置乒乓开关启动程序运行。程序处理过程中,若出现错误则停止运行,并通过控制台上的指示灯指示错误产生的条件。这时,程序员检查寄存器和主存单元,找出引起错误的原因。经修改后可再次运行。如果程序正常完成,再将输出结果送打印机打印。

这种早期的计算机存在两个问题:

一个问题是人工负责计算机的调度。

计算机各资源的使用是通过一张纸登记的。一般的做法是,各用户说明他大约使用计算机多长时间,由机房负责人为他安排上机时间。如果一个用户预约 1 小时,结果用 45 分钟完成

了,这样浪费了计算机的剩余时间。另一方面,用户程序运行过程中若出现问题,由于要检查原因解决问题,在预定时间内没有完成作业,也被迫暂停,要等待下一次运行。

另一个问题是人工负责作业编排顺序。

一个程序,又叫作业。一次运行可能只是编译一个程序。为此要将编译程序和一个用高级语言编写的源程序装入主存,保存被编译好的目标程序。然后装入和连接目标程序和公共函数库。这每一步都涉及安装和卸下磁带或卡片叠。如果在这次处理中出现错误,用户不得不返回到编排序列的开始重新运行。因此,大量时间用在编排程序运行上。这种操作方式叫顺序处理。

二、简单的批处理

早期计算机造价昂贵,因此要使计算机得到充分利用,上述的人工调度和作业编排方式浪费了大量的计算机时间,这是不能令人满意的。为了改进使用,操作系统的概念产生了。简单的批处理系统正是在 50 年代后期到 60 年代中期,伴随第二代计算机的出现而研制成功的。

简单的批处理模式的中心思想是使用一个监控程序软件。在这种软件控制下,用户不再直接与计算机打交道,而是将自己的一个作业卡片叠或纸带交给机房的操作员。再由操作员将多个作业的卡片或纸带按序成批地放在一个输入设备上,由监控程序自动开始装入一个个作业进行处理。

为了理解这个模式如何工作,让我们从两个方面进行观察:监控程序和处理机。

从监控程序方面看,监控程序控制事件的执行序列,它必须常驻主存。它负责从输入设备(卡片机或磁带机)以作业为单位读入一个进主存,当读入完成时,将处理机的控制权返回作业。当作业完成(正常完成或故障中断)时,控制又返回监控程序。它立刻读入下一个作业。当每个作业运行结束时,再由打印机打印结果后交于用户。

从处理机观点看,在某一时间点上,处理机开始执行存贮器中监控程序的指令,这些指令将启动输入机将下一作业读入主存中,一旦一个作业的读输入完成,处理机遇到监控程序中的一个转移指令,指示处理机转移到用户程序的开始位置继续执行。此后处理机执行用户程序中的指令,直到程序结束,或出现错误。其中任何一个事件(即程序正常结束或程序错误而故障终止)都将导致处理机返回监控程序执行。很清楚,手工操作阶段的两个问题已得到圆满解决:一方面由监控程序处理调度问题。一批作业排队,各作业以尽可能快的速度执行,从而不存在空闲的机器时间;另一方面由监控程序处理作业编排问题。为了使监控程序能代替用户完成对作业的编排控制,系统向用户提供了一套作业控制命令。每当用户提交作业时,将对作业的控制意图用作业控制卡或作业说明书的形式提交给监控程序。图 1.2 给出了以卡片形式提交作业的一个简单例子。在这个例子中,用户提交一个用 FORTRAN 语言编写的程序及程序所用数据。除此之外,还包括作业控制指令卡。它们以“\$”开始。下面是一个典型的监控程序使用的几个控制卡:

控制卡	功能
\$ JOB	启动一个程序
\$ FORTRAN	调用 FORTRAN 编译程序
\$ LOAD	调用装入程序
\$ RUN	运行程序
\$ END	作业结束卡

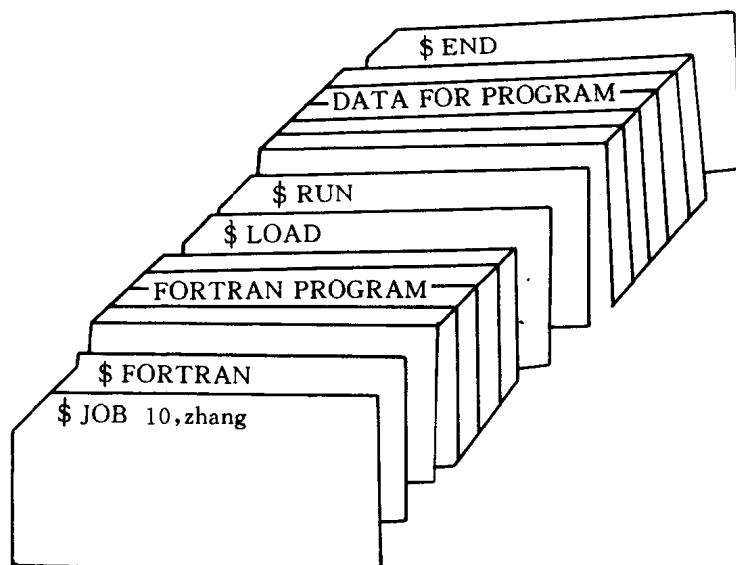


图 1.2 一个典型的输入作业的结构

为了执行这个作业,监控程序读入 \$ FORTRAN 控制卡,从外存磁带上装入 FORTRAN 编译。编译程序翻译用户程序为目标代码,并存入主存贮器或外存磁带上。如果它存入主存,这个操作叫“编译、装入和执行”。如果它存入磁带上(即只编译不装入执行),那么当要装入时还需要读入一个 \$ LOAD 卡。监控程序读入该卡片后,通过执行装入程序将目标程序及其调用的标准函数装入主存。接着读 \$ RUN 卡之后启动执行用户程序。在用户程序执行期间,用户程序的每个输入指令都引起读一张数据卡。由于用户程序的输入指令是通过调用监控程序的输入例程实现的,每当读入卡片时,输入例程检查确保程序不偶然地读一个控制卡,如果出现控制卡则产生错误,控制立即返回监控程序。一个用户作业的成功或错误完成,都引起监控程序扫描输入卡,直到遇到下一个作业的控制卡为止。

有了监控程序以后,用户不必亲自上机操作,由操作员代替。操作员只要通过控制台打字机敲入控制命令,监控程序就识别和执行。当计算机运行中发生错误或意外时,监控程序通过控制台打字机输出信息向操作员报告。这种输出信息不仅比亮灯显示所表达的更为丰富,而且便于操作员理解。总之,用这种半自动方式控制计算机不仅提高了效率,而且方便了使用。

这种简单的批处理在硬件结构上有两种不同的控制方式。

1. 早期联机批处理

早期的联机批处理的硬件控制方式是:作业的输入、计算和输出都是在 CPU 控制下进行的,图 1.3 给出了联机批处理的模型。

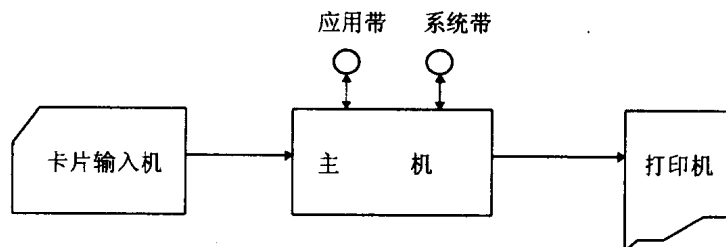


图 1.3 早期的联机批处理

由于输入或输出是在 CPU 直接控制下进行的,这样,主机的速度在输入或输出过程中,降

低为慢速的外设的水平。为此,引入早期的脱机批处理方式。

2. 早期的脱机批处理

为了使 CPU 从慢速的外设控制中解脱开来,增设一个小型的卫星机专门用来控制外部设备的输入、输出。其模型如图 1.4 所示。

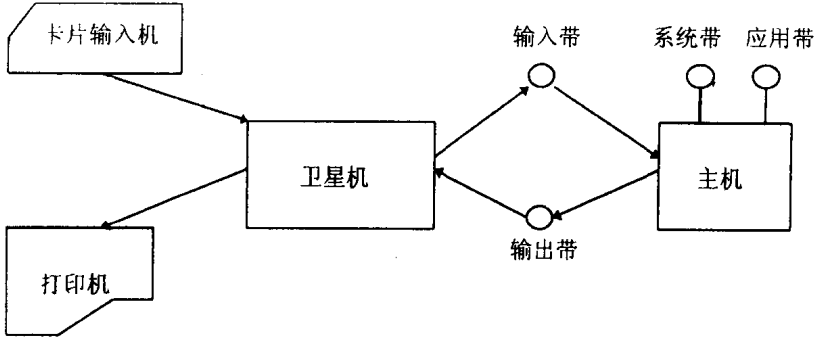


图 1.4 早期脱机批处理模型

小型卫星机的作用为

- ① 把卡片输入机上的作业逐个地记到输入磁带上,以便主机执行。
- ② 把主机送入输出带上的作业执行结果,控制从打印机上打印输出。由此可见,采用这种脱机技术后,主机的所有输入输出都是通过磁带进行,而且由主机与慢速外设的串行工作,变为并行,从而提高了主机运行效率。

从 50 年代后期到 60 年代中期,脱机批处理运行得相当成功。脱机技术的实质是用快速的输入输出设备代替慢速的设备。

三、多道成批处理系统(multiprogrammed batch system)

1. 多道程序设计

进入 60 年代中、后期,计算机的硬件有了突飞猛进的发展,产生了硬件通道、中断和缓冲技术。通道、中断的出现,使得计算机在组织结构上发生了重大变革。使原先以 CPU 为中心的体系结构,转变为以主存为中心,其结构模型如图 1.5 所示。

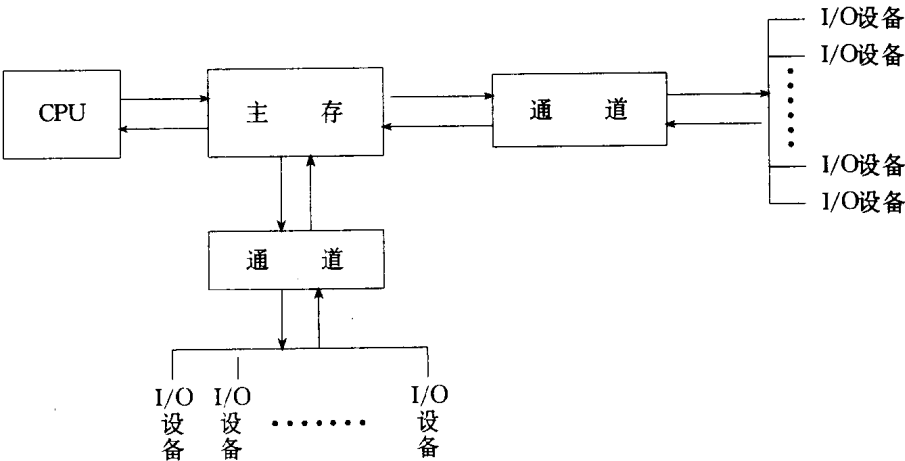


图 1.5 多道程序系统的计算机模型