

高职高专规划教材



双高规划教材

计算机操作系统

本书编委会 编



西北工业大学出版社

高职高专规划教材

计算机操作系统

本书编委会 编

江苏工业学院图书馆
藏书章

西北工业大学出版社

【内容提要】操作系统是计算机的核心软件，是所有计算机专业的必修课程。本书分为十章，两个附录。第1章介绍了操作系统的概念；第2章至第6章详细的讲解了操作系统中常见的类型、基本结构、主要控制和管理部分；上述各章都附有小结与习题，使学生能更熟练地掌握所学知识。第7章讨论了研究与实用领域中，操作系统的各种概念和技术；第8章介绍面向对象操作系统的设计；第9章介绍了操作系统结构和程序设计；第10章介绍了操作系统的安全机制。附录中给出了相关的实验和附加内容。

本书可作为一本面向高职高专学习的教材，也可作为学生在职、业余自主学习的书籍，满足不同学习对象的多样需要。

图书在版编目(CIP)数据

计算机操作系统/《计算机操作系统》编委会编.—西安：西北工业大学出版社，2004.1
ISBN 7-5612-1737-4

I. 计… II. 计… III. 操作系统 IV. TP316

中国版本图书馆CIP数据核字(2004)第004300号

出版发行：西北工业大学出版社

通信地址：西安市友谊西路127号 邮编：710072 电话：029-88493844

网 址：www.nwpup.com

印 刷 者：陕西天元印务有限公司

开 本：787 mm×1 092 mm 1/16

印 张：13

字 数：349千字

版 次：2004年4月第1版

2004年4月第1次印刷

定 价：18.00元

前 言

随着计算机技术快速广泛的发展,为了深化职业教育的改革,推进高职高专教育的发展,培养新世纪高级职业技术全面实用型人才,强化实践能力和创新意识,编者通过对国内外操作系统方面有关教材和论著的深入研究,力求使内容由浅入深、通俗易懂、涵盖面广、理论知识详实,并力图反映操作系统技术的新发展和新成果。

操作系统作为现代计算机系统最重要的系统软件,起着“中枢神经”的作用,它也是计算机应用者关心和感兴趣的一个领域。用户使用计算机实际上就是熟悉使用操作系统所提供的用户界面环境。每台计算机必须安装操作系统,有的不止安装一套。普通用户只需了解操作系统的外部功能,而无须知道其内部实现细节,这时,操作系统就如同“黑盒子”,只需要了解它的外部接口。在教学的过程中,按照“由表及里、由浅入深”的原则,不仅要求会操作它,还要学会利用计算机去开发各种软件,解决复杂的应用问题。

本书分为十章,两个附录。第1章介绍了操作系统的概念;第2章至第6章详细的讲解了操作系统中常见的类型、基本结构、主要控制和管理部分;上述各章都附有小结与习题,使学生能更好地自学和练习,以便于掌握所学知识。第7章讨论了研究与实用领域中,操作系统所面对的各种概念和技术。第8章介绍面向对象操作系统的设计;第9章介绍了操作系统结构和程序设计;第10章介绍了操作系统的安全机制。附录中给出了相关的实验和附加内容。

根据高职高专的培养目标,本书的参考教学时数为45~70学时,第8章到第10章的内容可以根据需要有选择的讲解,或留给学生自学。附录中的内容作为学生上机实习和课外阅读,以适应学生在职、业余自主学习为目标,满足不同学习对象的多样需要。

由于编者水平所限,不足之处在所难免。恳请广大读者将本套教材的使用情况及意见、建议及时反馈给我们,以便进一步改进和完善。

本书编委会

目 录

第 1 章 操作系统概论	1
1.1 操作系统概述	1
1.2 操作系统的形成和发展	2
1.2.1 手工操作阶段	2
1.2.2 脱机输入/输出技术	2
1.2.3 批处理技术	3
1.2.4 多道程序设计技术	3
1.3 操作系统的基本类型	3
1.3.1 多道批处理系统	4
1.3.2 分时系统	5
1.3.3 实时系统	6
1.3.4 网络操作系统	7
1.3.5 分布式操作系统	7
1.3.6 五大类型操作系统特点的比较	7
1.4 操作系统的特征与功能	8
1.4.1 操作系统的特征	8
1.4.2 操作系统的功能	8
1.4.3 操作系统的性能评价	10
1.5 操作系统的结构设计模式	11
1.5.1 模块化结构	11
1.5.2 层次化结构	11
1.5.3 客户/服务器结构	12
1.5.4 对象模式	12
1.5.5 对称多处理模式	12
1.6 操作系统的安装及配置	12
1.7 小 结	13
习题一	14
第 2 章 进程管理	15
2.1 进程的引入	15
2.1.1 进程的引入	15
2.1.2 程序的执行	15
2.2 进程的描述	17
2.2.1 进程的定义及特征	17

2.2.2 进程的状态	18
2.2.3 进程控制块	20
2.3 线 程	21
2.3.1 线程的引入	21
2.3.2 线程的定义	21
2.3.3 进程与线程	22
2.4 进程的控制	22
2.4.1 原语	23
2.4.2 进程控制原语	23
2.4.3 Linux 中的进程控制	25
2.5 同步与互斥	26
2.5.1 同步与互斥	26
2.5.2 原语	27
2.5.3 经典同步、互斥问题例子	30
2.6 进程通信	32
2.6.1 电子邮件	32
2.6.2 对话	33
2.6.3 管道文件	34
2.7 进程实例	34
2.8 进程调度	35
2.8.1 调度的层次	35
2.8.2 进程调度	36
2.8.3 进程调度算法	36
2.9 死 锁	38
2.9.1 死锁的概念	38
2.9.2 产生死锁的必要条件	39
2.9.3 死锁的判定	40
2.9.4 死锁的预防	40
2.9.5 死锁的避免与解除	41
2.10 小 结	42
习题二	43

第 3 章 作业管理

3.1 作业的基本概念	44
3.1.1 作业的定义	44
3.1.2 作业组织	46
3.1.3 作业管理的基本功能	46

3.1.4 作业的状态及其转换	47
3.2 作业管理	49
3.2.1 作业管理的任务	50
3.2.2 作业与资源	51
3.3 作业调度及常用作业调度算法	52
3.3.1 作业调度	52
3.3.2 调度性能的衡量	53
3.3.3 常用作业调度算法	54
3.4 用户界面	56
3.4.1 操作系统是用户和计算机用户之间的接口	56
3.4.2 用户界面的发展	58
3.4.3 界面管理的任务	58
3.4.4 界面管理的功能	59
3.4.5 用户界面的设计特点	59
3.5 小 结	60
习题三	60

第 4 章 存储管理

62

4.1 存储管理概述	62
4.1.1 存储管理技术	62
4.1.2 Windows 98 的存储器	62
4.1.3 存储分配方式	63
4.1.4 地址重定位	64
4.2 连续分配存储管理	66
4.2.1 单一连续分配	66
4.2.2 分区分配	67
4.2.3 覆盖与交换技术	72
4.3 分页存储管理	73
4.3.1 分页存储管理	73
4.3.2 分段存储管理	74
4.3.3 虚拟存储器	75
4.3.4 请求式分页存储管理	76
4.3.5 请求式分段存储管理	78
4.3.6 段页式存储管理	78
4.4 常用系统的存储管理	79
4.4.1 Windows 98 的存储管理	79
4.4.2 Linux 系统的存储管理	80

4.5 小 结	80
习题四	80
第 5 章 设备管理	82
5.1 设备管理概述	82
5.1.1 设备分类	82
5.1.2 设备管理的任务和功能	83
5.2 外部设备的安装	86
5.2.1 系统配置	86
5.2.2 用户外接设备	86
5.2.3 用户外接特殊设备	86
5.2.4 外部设备的即插即用	87
5.2.5 设备驱动程序的编制	87
5.3 数据传送控制方式	87
5.3.1 程序直接控制方式	88
5.3.2 中断方式	89
5.3.3 DMA 方式	89
5.3.4 通道控制方式	92
5.3.5 Windows 98 中的数据传输控制方式	93
5.4 设备管理中常用技术	95
5.4.1 中断技术	95
5.4.2 缓冲技术	97
5.5 设备分配	99
5.5.1 设备分配中的数据结构	99
5.5.2 设备分配策略	100
5.6 设备驱动程序	101
5.6.1 设备驱动程序	101
5.6.2 设备驱动程序的功能	101
5.6.3 设备驱动程序的特点	102
5.6.4 设备处理方式	102
5.7 SPOOLing 技术	103
5.7.1 什么是 SPOOLing	103
5.7.2 SPOOLing 系统的组成	104
5.7.3 共享打印机	104
5.7.4 SPOOLing 系统的特点	105
5.8 Windows 98 的设备管理	105
5.9 小 结	106

习题五	107
第 6 章 文件系统	108
6.1 文件和文件系统	108
6.1.1 文件和文件系统	108
6.1.2 文件的分类	109
6.1.3 实用系统中文件的分类	110
6.1.4 文件管理的功能	111
6.1.5 Windows 98 文件管理工具的实例	112
6.2 文件的结构和存取方法	113
6.2.1 文件的逻辑结构	113
6.2.2 文件的物理结构	114
6.2.3 文件的存取方法	116
6.2.4 文件结构、文件存储设备和存取法的关系	117
6.3 文件存储设备及存储空间管理	118
6.3.1 空白块链	118
6.3.2 位示图	118
6.3.3 FAT 磁盘格式	119
6.3.4 FAT32 磁盘格式特点	120
6.4 文件目录结构	121
6.4.1 文件控制块	121
6.4.2 一级目录结构	122
6.4.3 二级目录结构	122
6.4.4 目录树结构	123
6.4.5 目录树的构造	124
6.5 文件共享与保护	127
6.5.1 存取控制表	127
6.5.2 口令	128
6.5.3 密码	129
6.5.4 计算机病毒简介	129
6.6 文件使用	130
6.6.1 文件的创建和删除	130
6.6.2 文件的打开和关闭	131
6.6.3 文件的读写	132
6.7 小 结	132
习题六	133

第 7 章 各种操作系统	134
7.1 操作系统的各种模型	134
7.1.1 网状结构与层次结构	134
7.1.2 面向过程与面向对象	136
7.2 分布式操作系统	139
7.2.1 分布式系统定义	139
7.2.2 分布式操作系统的设计目标	139
7.3 网络操作系统	141
7.3.1 什么是网络	141
7.3.2 网络的结构	141
7.4 Windows NT 网络操作系统	143
7.4.1 网络构成	143
7.4.2 Windows NT 的结构	144
7.4.3 Windows NT 的体系结构	145
7.4.4 Windows NT 的管理职能	146
7.4.5 Windows NT 的特点	147
7.5 Linux 评述	147
7.5.1 系统体系结构	147
7.5.2 Linux 的文件系统及其组织结构	148
7.5.3 Linux 的网络功能	149
7.5.4 Linux 的主要特点	150
第 8 章 面向对象操作系统的设计	152
8.1 面向对象的基本概念	152
8.2 面向对象和操作系统	153
8.2.1 面向对象技术的引入	154
8.2.2 面向对象操作系统的设计方法	155
8.3 面向对象操作系统的分类	160
第 9 章 操作系统结构和程序设计	162
9.1 操作系统的编程概述	162
9.1.1 编程语言简史	162
9.1.2 不同应用领域的计算机语言	163
9.2 结构设计的目标	163
9.2.1 设计目标	164

9.2.2 分层原则	164
9.2.3 分块原则	166
9.2.4 模块接口法的设计步骤	167
9.2.5 模块接口法的优缺点	167
9.3 层次结构设计	168
9.3.1 层次模块化结构设计的必要性	168
9.3.2 操作系统的结构模型	170
9.3.3 操作系统的结构设计	170
9.4 DOS 模块结构	173
9.5 Windows 的编程模式简介	174
9.6 微内核结构简介	174
9.7 Linux 的安装与程序模块举例	175
第 10 章 安 全	178
10.1 鉴 别	178
10.1.1 口 令	178
10.1.2 物理鉴别	179
10.2 阻 止	179
10.3 检 测	180
10.4 校 正	180
10.5 身份识别	180
10.6 威胁的种类	180
10.7 程序威胁	181
附录	183
附录 1 计算机操作系统实验指导	183
实验 1 UNIX 或 Linux 操作系统的实际使用	183
实验 2 命令解释程序	183
实验 3 作业调度设计	185
实验 4 存储管理设计	186
实验 5 进程管理设计	187
附录 2 Linux 简要使用说明 (配合实验用)	189

第1章 操作系统概论

计算机系统由硬件和软件两部分组成。操作系统 OS (Operating System) 作为用户与计算机硬件系统之间的接口, 是配置在计算机硬件上的第一层软件, 是对硬件系统的第一次扩充。它在计算机系统中占据了特殊重要的地位, 随着计算机技术的发展和计算机应用的日益广泛而逐渐发展和完善。其他所有的软件如汇编程序、编译程序、数据库管理系统软件以及大量的应用软件, 都依赖于操作系统的支持。提高系统资源的利用率和方便用户使用操作系统的宗旨, 也是推动操作系统由简单到复杂、由低级到高级、由功能简单到功能强大的逐步发展的不竭动力。

1.1 操作系统概述

计算机系统提供收集数据、执行计算、存储信息以及与其他计算机系统通信和产生输出报告的能力。这些能力有的用硬件实现, 有的用软件实现。操作系统是操纵硬件并为程序执行建立一个更实用的平台的软件。操作系统相当于计算机系统的“管家”。所谓“管家”就是方便主人管理家务的人。实际上, 每一个人在你的头脑里已形成有“操作系统”, 管理和控制你的形体(硬件)和思维(软件)。而计算机操作系统要管理的是计算机系统。

一个完整的计算机系统, 不论是大型机、小型机还是微型机, 都由两大部分组成: 计算机硬件和计算机软件。硬件是指计算机的机器部分, 即我们所见到的物理设备和器件的总称, 如中央处理器(CPU)、存储器(内存与外存)、输入/输出设备等。软件通常是指计算机系统中使用的程序和数据, 并按功能分为系统软件和应用软件两类。系统软件是指为进行计算机系统的管理和使用而必须配置的软件, 如: 操作系统、各种语言处理程序等; 应用软件是指针对某类专门应用而配置的软件, 如: 计算机辅助教学(CAI)软件、财务管理软件等。

操作系统直接构建在硬件接口上, 为硬件和用户程序之间提供了连接。没有任何软件支持的计算机称为裸机, 裸机仅构成计算机系统的物质基础, 不能直接使用, 需要在裸机上安装操作系统和应用软件。呈现在用户面前的计算机系统是经过多层软件改造过的计算机, 而操作系统正是计算机启动时运行的第一个程序。

硬件接口由程序计数器、寄存器、中断、硬盘和终端等部件组成。操作系统同时拥有软件和硬件的特性。操作系统是软件, 它是经过编译、连接、运行在计算机上的一个程序; 但它又像硬件, 只有一个操作系统拷贝运行在计算机上, 它扩展了计算机的性能。其他软件使用特殊的“自陷”(trap)或“系统调用”命令请求操作系统提供服务, 这类似于使用硬件的指令。

现代计算机系统中硬件与软件之间的关系可分成若干层次, 如图 1.1.1 所示。硬件(裸机)在最里层, 是计算机系统工作的物质基础, 它的外面是操作系统, 通过系统程序对计算机系统中各类资源(处理器、存储器、设备、数据等)进行管理并提供方便用户使用的多种服务功能, 隐蔽对硬件的复杂操作, 把裸机改造成功能更强、使用更方便的系统。

操作系统的外面是其他系统软件。操作系统是最基本的系统软件, 其他系统软件对用户的操作和开发软件提供支持。如文本编辑、汇编程序、编译程序、连接程序、命令解释程序等, 它们不是操作

系统的一部分，一般与操作系统一起由供应商提供。有时，人们也将系统实用软件、系统工具软件等视做系统软件。

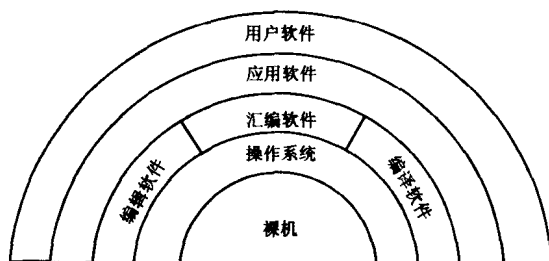


图 1.1.1 操作系统与硬件软件的关系

系统软件的外面是应用软件，它们是为各应用领域服务的软件，如数据库软件、财务软件等。用户可以直接经过系统软件操作计算机，也可以通过实用软件或工具软件操作计算机。因此，操作系统可以定义如下 3 个方面的程序集合：

- (1) 控制和管理计算机系统的硬件和软件资源；
- (2) 合理地组织计算机的工作流程；
- (3) 方便用户使用。

1.2 操作系统的形成和发展

1.2.1 手工操作阶段

早期的计算机系统上没有配置操作系统，人们使用计算机采用手工操作方式。在手工操作方式下，用户一个接一个地轮流使用计算机，每个用户的使用过程大致如下：先将程序纸带（或卡片）装入输入机，然后启动输入机把程序和数据送入计算机，接着通过控制台开关启动程序运行，当程序运行完毕，由用户取走纸带和计算结果。这种手工操作方式具有用户独占计算机资源、资源利用率低及 CPU 等待人工操作的缺点。

随着 CPU 速度的大幅度提高，手工操作的慢速与 CPU 运算的高速之间出现了矛盾，这就是所谓的人机矛盾。另一方面，CPU 与 I/O 设备之间速度不匹配的矛盾也日益突出。

1.2.2 脱机输入/输出技术

为解决低速输入设备与 CPU 速度不匹配的问题，将用户程序和数据在一台外围机（又称卫星机）的控制下，预先从低速输入设备输入到磁带上，当 CPU 需要这些程序和数据时，再直接从磁带机高速输入到内存，从而大大加快程序的输入过程，减少 CPU 等待输入的时间，这就是脱机输入技术。

类似地，当程序运行完毕或告一段落，CPU 需要输出时，无需直接把计算结果送至低速输出设备，而是高速地把结果送到磁带上，然后在外围机的控制下，把磁带上的计算结果由相应的输出设备输出，这就是脱机输出技术。输入/输出操作在主机控制下进行则称之为联机输入/输出。

采用脱机输入/输出技术后，低速 I/O 设备上数据的输入/输出都在外围机的控制下进行，而 CPU

只与高速的磁带机打交道，从而有效地减少了 CPU 等待慢速设备输入/输出的时间。

1.2.3 批处理技术

批处理技术是指计算机系统对一批作业自动进行处理的一种技术。

早期的计算机系统非常昂贵，为了能充分地利用它，应尽量让系统连续地运行，以减少空闲时间。为此通常是把一批作业以脱机输入方式输入到磁带上，并在系统中配置监督程序（它管理作业的运行，负责装入和运行各种系统程序来完成作业的自动过渡），在它的控制下，先把磁带上的第一个作业传送到内存，并把运行的控制权交给它，当第一个作业处理完后又把控制权交还给监督程序，由监督程序再把第二个作业调入内存。计算机系统按这种方式对磁带上的作业自动地、一个接一个地进行处理，直至把磁带上的所有作业全部处理完毕，这样便形成了早期的批处理系统。

1.2.4 多道程序设计技术

多道程序设计技术是将多个作业存放在主存中，使它们同时处于运行状态，这些作业共享处理机时间和外围设备以及其他资源。现代计算机系统一般都基于多道程序设计技术。

在单处理机系统中，多道程序运行的特点是：

（1）多道：计算机内存中同时存放几道相互独立的程序。

（2）宏观上并行：同时进入系统的几道程序都处于运行过程中，即它们先后开始了各自的运行，但都未运行完毕。

（3）微观上串行：实际上，内存中的多道程序轮流地占有 CPU，交替执行。

多道程序设计技术能有效地提高系统的吞吐量和改善资源利用率。但实现多道程序系统时，由于主存中总是同时存在几道作业，因而还需要妥善解决下述一系列问题：

（1）应如何分配处理机，以使处理机既能满足各程序运行的需要又有较高的利用率以及将处理机分配给某程序后，应何时收回等问题。

（2）如何为每道程序分配必要的内存空间，使它们各得其所又不致因相互重叠而丢失信息以及应如何防止因某道程序出现异常情况而破坏其他程序等问题。

（3）系统中可能有多种类型的 I/O 设备供多道程序共享，应如何分配这些 I/O 设备，如何做到既方便用户对设备的使用，又能提高设备的利用率。

（4）在现代计算机系统中，通常都存放着大量的程序和数据，应如何组织它们才能便于用户使用并能保证数据的安全性和一致性。

为解决上述问题，应在多道程序系统中增设一组软件，用以对上述问题进行妥善有效的处理，此外还应提供方便用户使用计算机的软件，这样便形成了操作系统。

操作系统是一组控制和管理计算机硬件和软件资源，合理地组织计算机工作流程以及方便用户的程序的集合。

1.3 操作系统的基本类型

当前计算机已广泛深入人类生活的各个领域，从办公室自动化到为人们看家、做饭；从工业控制

到科学计算。自然,在如此广泛的使用领域中,人们对计算机的要求是不同的。于是对计算机上的操作系统的性能要求,使用方式也是不同的。因此对操作系统的类型进行分类的方法也很多。例如,可以按照机器硬件的大小而分为大型机操作系统、小型机操作系统和微型机操作系统。由于大型机属于能力较强,附属设备较多,所以价格比较昂贵。机器主人关心的是如何使所有硬件设备得到充分使用,也就是说希望机器有较大的工作负荷,并要求机器能适应各种类型和性质的工作(如科学计算、数据处理、数据库管理和网络等)的通用性。所以很好地调度和管理系统资源就成为大型机操作系统的主要任务。而资源使用的有效性以及机器的吞吐量(指固定时间间隔中机器所完成的作业数),往往是大型机操作系统所追求的主要目标。

随着计算机操作系统日新月异的发展变化,现在一般将操作系统分成五大类:

- (1) 多道批处理系统;
- (2) 分时系统;
- (3) 实时系统;
- (4) 网络操作系统;
- (5) 分布式操作系统。

1.3.1 多道批处理系统

多道批处理系统按用户作业的类型不同分成若干批次,将不同批次的作业都存放于存储器中,每一批次作业顺序处理。如果需要输入/输出,就调用另一批次的作业运行,从而实现资源的充分利用。下面看看一些具体的概念:

1. 单道程序与多道程序

单道程序是指在主存储器中只存放着一批程序(或者一个程序),当 CPU 运行该程序发生某种条件等待时, CPU 暂停当前程序的运行,在等待的条件被满足以前, CPU 将一直处于闲置状态。多道程序是指在主存储器中存放着不止一批程序(或者多个程序),当 CPU 运行某一个程序发生条件等待时,可以转向执行另外的程序,因此,多道程序方案可以减少 CPU 的闲置时间。从操作系统方面来说,管理多道程序比管理单道程序更为复杂。

在单道程序环境下,操作系统不需要考虑对处理机、存储器、输入/输出设备的分配,它的主要工作是在适当的时候将需要执行的程序从辅助存储器调到主存储器中,安排编译或者汇编,安排目标代码的运行,接受输入信息及传送输出信息,管理起来相对简单。

但多道程序环境就不是这样。由于内存中存放了大量的程序,并且由于多道程序分享着 CPU 的时间,那么,多道系统就必须考虑 CPU 时间的分配、主存储器空间的分配及安全、输入/输出设备中断系统的实现等。

2. 作业

作业是用户交给计算机执行的具有独立功能的任务。例如,图 1.3.1 就是作业的组织与运行。在用户要求计算机执行任务时,首先用一种表达方式将任务进行描述,内容包括:作业的名称、作业的执行步骤、作业所涉及到的程序与数据等。作业的执行步骤往往代表着一个具体的子功能,它被称为作业步。作业步的执行顺序是:前一个作业步的输出是后一个作业步的输入,后一个作业步必须在前一个作业步执行完毕后执行。作业的描述方法有多种,可以通过专用的作业控制语言、高级语言、键盘命令等来对作业进行说明。

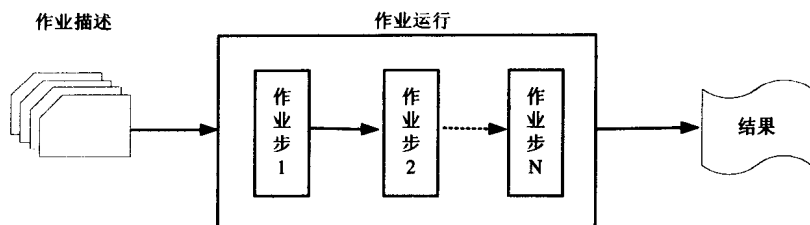


图 1.3.1 作业的组织与运行

3. 批处理

它是指将作业组织成批，并一次将该批作业的所有描述信息和作业内容输入计算机，计算机将按照作业和作业步进入的先后顺序依次自动执行，在一个批次范围内用户不得对程序的运行进行任何的干预。

批处理系统是一个脱机处理系统，由于没有用户的介入，它围绕着提高系统的效率而开展工作。具体方法将涉及到处理机时间的分配、存储器空间的划分、设备运行效率及均衡性以及计算机各部件之间速度的匹配。批处理系统适用于专门承接运算业务的计算中心，可帮助用户完成大型工程运算等工作。

由于批处理系统采用的是脱机工作方式，因此基本不考虑用户的联机要求；又由于系统的设计目标是尽可能提高系统的运行性能和效率，从作业委托到作业完成之间的时间可能比较长。这也是批处理系统不足的地方。

如果用户希望参与控制和选择程序的运行，批处理系统就不是一个好的方案。事实上计算机系统不再是专门人员的特殊装备，随着计算机硬件和软件的发展，已逐渐成为普通用户的日常工具。

1.3.2 分时系统

一台计算机能为多个终端用户服务吗？还能保证每个用户都能经常与计算机进行交互吗？分时操作系统这位“管家”便具有这方面的本领。那么它是如何做到这一点的呢？分时操作系统的秘诀是采用“转盘子”技术。你看过杂技中的“转盘子”表演吗？在一张很长的桌子上摆着一排正要旋转的盘子，演员的任务便是保证每个盘子都在旋转。他的方法是轮流地去转每一个盘子，并尽量争取在最短的时间内完成，这样才能保证在转到最后一个盘子后再回到下一轮开始前第一个盘子仍然保持旋转的状态。这位身手敏捷忙得团团转的杂技演员就是 CPU。为了能在计算机上实现“转盘子”表演，人们引入了分时技术，即把 CPU 的时间分成很短的时间片（例如，几十至几百毫秒）工作。采用这种分时技术的系统称为分时系统。

分时系统中的 CPU 有很强的“转盘子”本领。在每一个时间片内，CPU 专为一个用户工作。时间一到，无论工作结束与否 CPU 都会暂停服务，转向为下一个用户工作。不过，多亏 CPU 运转的速度非常快（实际上，在几十至几百毫秒的时间片内已能执行成百上千条指令），所以对付区区几个甚至成百上千用户（实质上是在轮流执行某个用户某一段程序段）根本不在话下。这样，每个用户的要求都能达到快速响应，给每个用户的印象是：好像他独占一台计算机。分时操作系统这位“管家”的特点是计算机规定或制约人（用户），工作的主动权掌握在计算机手里。虽然“管家”会尽力快速工作来满足每个用户的要求，但对对不起，只要时间片一到就立即说再见，无论眼下这位用户的工作多么重要，也得赶去为下一个用户服务。当然，要做好“记录”，以便下一次接着做时正确无误。在分时

系统中,要有给定的合理响应时间,以便保证用户的“忍受”程度,时间片的大小在一定程度上取决于时间片的切换速度。因为,随着时间片的时间减少,对换时间所占的比例随之增大。随着用户数目的不断增加,这种矛盾会越来越突出。CPU 速度再快,毕竟也有个极限,也有应付不过来的时候。所以在使用多用户系统时要注意操作系统厂商提供的最多用户数目。

1.3.3 实时系统

实时系统是为了满足特殊用户的需要,在响应时间上有着特殊要求,利用中断驱动执行专门的处理程序、具有高可靠性的系统。这类系统广泛地应用于军事、工业控制、金融证券、交通及运输等领域。

以证券交易系统为例,当一次交易行为发生,主机必须在极短的时间内进行反应,然后将交易结果输出到显示终端上。如果主机的反应时间有所拖延,将导致显示终端上的数据与正在发生的数据不一致,这会影​​响所有交易者的正确决策,也就必然影响到随之而来的所有交易行为的正确性。另外,这种系统还不允许错误的发生,如果某一个数据的处理发生错误,也将使整个系统的正确性得不到很好保障。

在军事上,实时系统的立即响应和高可靠性表现得更为突出,例如:有一种红外制导导弹,它需要对导弹发出的红外光的反射光进行分辨,来确定导弹紧跟着的运行轨迹。如果分辨并做出决定的时间太长,飞速运行的导弹可能早已偏离正确的轨迹,这种后果是不堪设想的。

下面是与实时系统相关的一些概念:

1. 专门系统

实时系统一般说来都是定制系统,它针对某一个特殊的需要,由设计者设计相应的硬件并配合编制出对应的管理系统。各领域之间的实时系统不能通用,甚至同一领域内由于用途的细微差别也不可能照搬同一个实时系统。因此,系统的设计费用无法均摊,专门系统比普通的分时系统价格要高得多。

2. 立即响应

立即响应要求从事件发生到计算机做出反应之间的时间非常短,这种短不在于人的感觉而在于机器时钟的度量,通常在微秒数量级范围。不同的系统其反应时间的要求也不同,这种反应时间必须保证被控制设备能够做出正确的动作,任何时间延迟都会导致系统的错误。

3. 事件驱动

实时系统是针对某一种特殊需要而设计的,因此,它为每一种可能发生的情况都编制好了对应的处理程序,这些程序被称为事件处理程序或者中断处理程序,并且在系统启动时就被存放在存储器上。只有当事件发生了,事件处理程序才会被运行,因此说事件处理程序是靠事件来驱动的。在事件没有发生的情况下,实时系统一般处于等待状态。

4. 高可靠性

由于实时系统发生错误所导致的结果都非常严重,因此,不能允许实时系统产生失误,这就需要保证系统每一个部件的正确和稳定运行。保证系统高可靠性的方法可以有:多存储器系统或者存储器镜像系统可将同样的数据重复保存在不同的存储位置上,以保证存储的数据在意外情况发生时还能够被恢复;多处理机系统可采用主处理机和后备处理机处理同样的事件,如果主处理机发生意外,则启用后备处理机的处理结果;多主机系统,多套处理机及存储器组合,以此来避免任何意外所导致的不