

# 操作系统 教程

● 王素华 编著



6  
1

人民邮电出版社



TP316  
WSH/1

计算机技术丛书

# 操作系统教程

王素华 编著

人民邮电出版社

037448

## 内 容 提 要

本书讨论计算机操作系统的基本原理、基本概念和基本算法,立意新颖,取材先进,并附有全部习题的参考答案,是一本适用范围较广的教学参考书。

全书共分九章,第一章讨论了操作系统的功能、类型及其发展。以后六章,按资源管理的观点,分别讨论了单处理机条件下的处理机管理、存储管理、设备管理和文件管理。第八章专门讨论操作系统的安全问题。第九章则讨论了多处理机条件下操作系统的特点,以及现代多机操作系统 Mach 3.0 引入的新思想、新概念和新技术。本书附录给出了各章习题的参考答案。

本书适用于大学计算机本科生,对于计算机应用领域从事科研、开发、教学、管理的人员,大专院校相关专业的教师,以及报考研究生和参加自学考试的学生,亦有一定的参考价值。

计算机技术丛书

操作系统教程

王素华 编著

责任编辑 王晓丹

\*

人民邮电出版社出版发行

北京朝阳门内南竹杆胡同 111 号

北京顺义向阳胶印厂印刷

新华书店总店科技发行所经销

\*

开本:787×1092 1/16 1995 年 10 月第 一 版

印张:19.5 1996 年 6 月北京第 2 次印刷

字数:482 千字 印数:4 001—9 000 册

ISBN7-115-05815-6/TP·234

定价:26.00 元

J5393/27

# 丛 书 前 言

世界上发达国家普遍重视发展以计算机和通信为核心的信息技术、信息产业和信息技术的应用,一些经济发达国家信息产业发展迅速。

当前,我国处于国民经济高速发展时期。与此相伴随,必将有信息技术、信息产业和信息技术应用的高速发展。各行各业将面临信息技术应用研究与发展的大课题以及信息化技术改造的大任务、大工程。

为了适应信息技术应用大众化的趋势,提高应用水平,我们组织编写、出版了这套“计算机技术丛书”。这套丛书以实用化、系列化、大众化为特点,介绍实用计算机技术。

这套丛书采取开放式选题框架,即选题面向我国不断发展着的计算机技术应用的实际需要和国际上的实用新技术,选题不断增添又保持前后有序。

这套丛书中的著作还拟配合出版软件版本,用软盘形式向读者提供著作中介绍的软件,以使读者方便地使用软件。

我们希望广大读者为这套丛书的出版多提意见和建议。

# 前 言

计算机系统是一个软件和硬件紧密结合的统一整体,而操作系统是对于硬件功能的首次扩充,是其它系统软件和应用软件建立的基础和平台。因此,操作系统在整个计算机系统中处于承上启下的关键地位,在系统软件中,则占据着核心地位。

该课程主要涉及对于各种资源(软件和硬件资源)的有效管理,以及为用户程序的运行提供良好的支撑环境与界面。故而,对于操作系统概念、理论、算法和实现技术的研究,以及对于操作系统的分析、设计、使用和开发,历来是计算机学科领域的重要课题与任务。而对于该课程的教学,则一直在计算机学科的教学计划中占据重要位置。美国 ACM 和 IEEE/CS 联合专题组发表的“Computing curricula 1991”报告中,确定了九个主科目领域,操作系统便是其中的一个重要领域。中国计算机学会教育委员会,全国高等院校计算机教育研究会所提出的“计算机教学计划 1993”,对此也进一步做了肯定。

本教程是在多年教学实践的基础上,汲取国内、外较新理论和技术,参考国内、外比较成熟的教材,并参照“计算机教学计划 1993”对于操作系统的要求而进行编写的。而且,编者试图使本教程具有某些特色:

第一,结合性。指的是操作系统(OS)和计算机体系结构(CA)的结合。从微观上,对于 OS 相关内容的论述,紧紧扣住“软/硬件交界面”这个基本点,站在 CA 这个生长点上,讨论 OS 的功能与结构。从宏观上,则以揭示冯·诺依曼结构的局限性到开拓并行结构为线索,研讨 OS 从传统到并行发展的内在联系,以揭示 OS 的本质问题并展望其发展。

第二,先进性。指的是在一定程度上反映操作系统的最新发展,特别是 90 年代出现的新思想、新概念、新技术。为此,在讨论单机操作系统的基础上,结合讨论多机操作系统 Mach 3.0,介绍了微内核技术、线程、端口等新思想、新概念、新技术。再如,随着计算机应用领域的发展与深化,计算机系统的安全问题变得越来越重要,而计算机系统安全的关键是操作系统的安全。因此,本教程专门增加了一章,讨论操作系统的安全问题。

第三,实用性。指的是要提供一定的手段,帮助学员理解和掌握操作系统的基本理论、基本概念和基本算法。为此,本教程精心为每章都安排了一定数量的习题,并且在附录中给出了参考答案,与正文相配合,它们是正文内容的扩充与深化。这样,对于计算机自学者,无疑也等于提供了一位求之不得的辅导教师。

本教程共分九章,第一章为引论,讨论了操作系统的功能、类型及其发展。以后六章,按资源管理观点,分别讨论了单处理机条件下的处理机管理(第二到第四章),存储管理(第五章),设备管理(第六章)和文件管理(第七章)。第九章则讨论了多处理机条件下操作系统的特点,以及现代多机操作系统 Mach 3.0 引入的新思想、新概念、新技术。而第八章则专门讨论了操作系统的安全问题。

本教程适用于大学计算机学科本科生。对于计算机应用领域从事科研、开发、教学和管理

的人员,大专院校相关专业的教师,以及报考研究生和参加自学考试的考生,亦有一定的参考价值。

由于作者水平及时间所限,本教程缺点和错误在所难免,前面所言之“特色”也未必能得到充分体现,还望同行专家、学者,以及广大读者批评、指正。同时,作者向本教程中引用的有关著作的作者表示谢意。

作者

1995年3月31日

# 目 录

<b>第一章 引论</b> .....	1
1.1 操作系统的历史与演变 .....	1
1.1.1 计算机体系结构的演变 .....	1
1.1.2 操作系统的产生与发展 .....	3
1.2 操作系统的功能与定义 .....	5
1.2.1 机器硬件的延伸 .....	5
1.2.2 资源管理者 .....	6
1.2.3 提供服务 .....	6
1.2.4 操作系统的功能与定义 .....	6
1.3 操作系统的结构模型 .....	7
1.3.1 单体结构 .....	7
1.3.2 层次结构 .....	7
1.3.3 微内核结构(顾客-服务员模型) .....	8
习题 .....	10
<b>第二章 进程管理</b> .....	11
2.1 进程的概念 .....	11
2.1.1 程序顺序执行与其特征 .....	11
2.1.2 程序的并发执行与特征 .....	12
2.1.3 前趋图和 Bernstein 条件 .....	13
2.1.4 进程的定义 .....	15
2.1.5 进程的基本状态与状态的变迁 .....	16
2.1.6 进程控制块 .....	16
2.1.7 进程的上下文 .....	17
2.2 进程的控制 .....	19
2.2.1 进程的家族树 .....	19
2.2.2 进程控制原语 .....	19
2.3 进程同步与互斥 .....	22
2.3.1 进程间的制约关系 .....	22
2.3.2 临界资源与互斥 .....	23
2.3.3 解决互斥的锁机制 .....	23
2.3.4 信号量同步机制 .....	24
2.3.5 信号量机制的应用 .....	26
2.3.6 信号量集同步机制 .....	28
2.3.7 管程 .....	31

2.4	经典同步问题 .....	33
2.4.1	生产者-消费者问题 .....	33
2.4.2	读者-写者问题 .....	34
2.4.3	五个哲学家就餐问题 .....	36
2.4.4	用其它机制解决经典同步问题 .....	36
2.5	进程通信 .....	39
2.5.1	直接通信 .....	39
2.5.2	间接通信 .....	42
2.5.3	管道通信 .....	44
2.6	死锁 .....	44
2.6.1	死锁与死锁产生的必要条件 .....	45
2.6.2	死锁的描述——资源分配图 .....	46
2.6.3	死锁预防 .....	47
2.6.4	死锁避免 .....	48
2.6.5	死锁检测 .....	51
2.6.6	死锁解除 .....	54
	习题 .....	56
<b>第三章</b>	<b>作业管理 .....</b>	<b>61</b>
3.1	概述 .....	61
3.1.1	作业和作业步 .....	61
3.1.2	作业状态与处理流程 .....	61
3.1.3	作业管理的任务 .....	62
3.1.4	作业输入/输出方式的演变 .....	62
3.2	批处理系统的作业控制 .....	63
3.2.1	脱机作业控制 .....	64
3.2.2	联机作业控制 .....	64
3.3	分时系统作业(交互作业)控制 .....	64
3.4	用户接口 .....	64
3.5	系统调用(广义指令) .....	65
	习题 .....	67
<b>第四章</b>	<b>处理机调度 .....</b>	<b>68</b>
4.1	多级调度 .....	68
4.2	作业调度 .....	69
4.2.1	调度算法的评价 .....	69
4.2.2	调度性能的衡量 .....	70
4.2.3	单道批处理系统的调度算法 .....	70
4.2.4	多道批处理系统的调度算法 .....	72
4.3	进程调度 .....	74
4.3.1	概述 .....	74
4.3.2	进程调度算法 .....	74
4.4	进程调度实现 .....	78
4.4.1	就绪队列组织 .....	78



4.4.2 分派程序(dispatch).....	78
习题 .....	80
<b>第五章 存储管理</b> .....	<b>85</b>
5.1 概述 .....	85
5.1.1 存储器的层次结构 .....	85
5.1.2 存储管理的目的和功能 .....	86
5.1.3 存储分配方式与时机 .....	87
5.1.4 重定位 .....	88
5.1.5 虚拟存储器 .....	90
5.2 覆盖与交换 .....	91
5.3 单一连续区存储分配 .....	92
5.4 分区存储管理 .....	92
5.4.1 固定分区 .....	93
5.4.2 可变分区 .....	94
5.4.3 可变分区的分配与回收算法 .....	95
5.4.4 可重定位分区 .....	96
5.4.5 多重分区 .....	96
5.4.6 分区的保护 .....	97
5.5 分页存储管理 .....	98
5.5.1 页表 .....	99
5.5.2 地址转换 .....	99
5.5.3 快表 .....	100
5.6 请求式分页存储管理——页式虚拟存储系统 .....	101
5.6.1 页描述子扩充 .....	101
5.6.2 地址转换与缺页中断 .....	101
5.6.3 页面淘汰算法 .....	101
5.6.4 性能分析 .....	104
5.6.5 页面共享与保护 .....	105
5.7 分段存储管理 .....	106
5.7.1 段表与地址转换 .....	107
5.7.2 段式虚拟存储系统 .....	108
5.7.3 段的动态连接 .....	109
5.7.4 段的共享 .....	111
5.7.5 段的保护 .....	111
5.8 段页式存储管理 .....	112
5.8.1 实现原理 .....	113
5.8.2 段页式存储管理算法 .....	116
习题 .....	117
<b>第六章 设备管理</b> .....	<b>122</b>
6.1 输入/输出硬件原理 .....	122
6.1.1 外部设备 .....	122
6.1.2 中断技术 .....	123
6.1.3 DMA .....	125

6.1.4	通道	127
6.1.5	外围处理机	129
6.2	设备管理的任务与功能	130
6.3	设备分配	131
6.3.1	独占、共享与虚拟设备	131
6.3.2	分配方式与分配算法	132
6.3.3	与设备无关性	133
6.3.4	设备分配	133
6.4	驱动调度	136
6.4.1	物理特性	137
6.4.2	驱动调度算法	137
6.5	设备驱动与中断处理	138
6.6	缓冲技术	142
6.6.1	缓冲技术的引入	142
6.6.2	缓冲区的设置	143
习题		144
<b>第七章</b>	<b>文件系统</b>	<b>146</b>
7.1	概述	146
7.1.1	文件与文件系统	146
7.1.2	文件分类	147
7.1.3	文件说明	147
7.2	文件的逻辑结构和物理结构	148
7.2.1	文件的逻辑结构	148
7.2.2	文件的物理结构	149
7.2.3	组块与解块	151
7.2.4	多级索引	153
7.3	存取方法	156
7.4	文件目录	158
7.4.1	单级目录	158
7.4.2	二级目录	159
7.4.3	多级目录	159
7.4.4	符号文件目录与基本文件目录	160
7.4.5	文件目录的管理	162
7.5	文件存储空间的管理	163
7.5.1	空白文件目录法	163
7.5.2	位示图法	163
7.5.3	空闲块链接法	164
7.6	文件存取控制	166
7.7	文件系统的层次模型	168
7.7.1	用户接口及初始化	169
7.7.2	符号文件系统(SFS)	169
7.7.3	基本文件系统(BFS)	169

7.7.4	存取控制验证(ACV)	170
7.7.5	逻辑文件系统(LFS)	170
7.7.6	物理文件系统(PFS)	170
7.7.7	设备策略模块(DSM)	171
7.7.8	I/O 调度和控制系统(IOCS)	171
7.7.9	分配策略模块(ASM)	171
7.8	文件的主要操作	172
习题		175
<b>第八章</b>	<b>系统安全</b>	<b>178</b>
8.1	系统安全的重要性	178
8.1.1	计算机系统的脆弱性	178
8.1.2	系统安全的主要内容	179
8.1.3	系统安全的评价与标准	180
8.1.4	数据信息的安全与加密	182
8.1.5	计算机病毒的危害	182
8.1.6	操作系统对系统安全的作用	183
8.2	用户身份鉴别	184
8.2.1	口令鉴别	184
8.2.2	信物鉴别	186
8.2.3	利用人类特征鉴别	186
8.3	访问控制	187
8.3.1	访问控制矩阵	187
8.3.2	访问控制类型	188
8.3.3	自主访问控制	188
8.3.4	强制访问控制	193
8.3.5	隐蔽信道	197
8.4	操作系统安全与保护	199
8.4.1	进程的支持	199
8.4.2	主存保护	199
8.4.3	保护环	201
8.4.4	I/O 访问控制	202
8.4.5	安全核	206
习题		210
<b>第九章</b>	<b>多机操作系统</b>	<b>211</b>
9.1	由单机操作系统到多机操作系统	211
9.1.1	单机操作系统	211
9.1.2	多机操作系统的特征	211
9.1.3	多机操作系统的功能	213
9.2	进程同步	214
9.2.1	集中式同步机构	214
9.2.2	利用事件计数实现同步示例	216
9.2.3	分布式进程同步	217
9.3	任务分配	220

9.3.1 任务分配应考虑的问题.....	220
9.3.2 最早调度算法.....	221
9.3.3 子集调度算法.....	222
9.3.4 搜索树.....	223
9.4 Mach 多机操作系统 .....	225
9.4.1 Mach 引入的新概念 .....	225
9.4.2 Mach 的处理机调度 .....	228
9.4.3 Mach 的存储对象概念 .....	229
9.4.4 Mach 的 UNIX 服务员 .....	231
习题 .....	235
参考文献 .....	236
附录:习题参考答案 .....	237

# 第一章 引 论

## 1.1 操作系统的历史与演变

### 1.1.1 计算机体系结构的演变

现代计算机系统是由硬件和软件组成的统一整体。无论从设计、研究、应用、开发,乃至性能评价角度,人们面临的都是这样一个包括硬件和软件的整体。为了便于分析和认识这样一个复杂的系统,可以从使用的角度,按功能把计算机系统划分成多级的层次结构(图 1.1)。

实际上,机器硬件只能识别和执行二进制机器指令(通常称之为机器语言),而直接使用机器语言编程既不方便,也根本无法满足计算机应用日益发展的需要。因此,人们就在硬件的基础上,用软件构造了不同层次的软件机器(通常称之为虚拟机),这样用户就可以利用越来越方便的语言(从汇编语言、高级语言,乃至应用语言,今后还将进一步发展为自然语言)来使用机器。其中,每一层次虚拟机之间语言的转换,则是由相应的翻译软件(解释程序或编译程序)来完成的。

以上这种计算机系统的多级层次结构概念,有助于我们对于操作系统地位、功能、作用的理解。实际上,操作系统是对于机器硬件功能的第一次扩充,而且也构成了以上各层次软件继续扩充系统功能的基础。

从计算机系统的层次结构上看,操作系统正处于硬、软件交界面的另外一侧。而所谓软、硬件交界面,正如 G. M. Amclahl 定义的,指的是计算机体系结构(Computer Architecture)(图 1.2)。他在 1964 年,介绍 IBM 360 系统时指出:“体系结构这个词,表示从程序员角度看的机器属性。也就是脱离开数据的流动、控制方法、逻辑设计、物质实现等方面,而从概念上论述其结构,从功能上说明其动作”。从这个意义上讲,操作系统扩充功能的基础与对象正是计算机体系结构。

因此,计算机体系结构的变化与发展,自然也就影响到操作系统的变化与发展。故而,在讨论操作系统之前,首先了解一下计算机体系结构的发展过程,也就显得很自然了。

说到计算机体系结构的发展过程,冯·诺依曼可以说是经典体系结构的奠基人。他于 1946 年 6 月发表的题为“关于电子计算装置逻辑设计的初步探讨”的报告,提出了构造电子计算机的基本理论,奠定了几乎今天所有计算机的理论基础,而可以统称之为冯·诺依曼型计算机。

冯·诺依曼经典结构的最重要的特点是,由单一的计数器(PC)控制,顺序执行单一

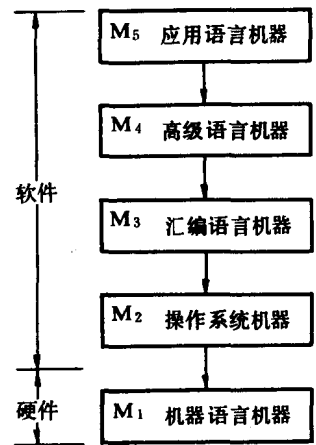


图 1.1 计算机系统的层次结构

的指令流(IS),处理单一的数据流(DS)(图 1.3)。这种完全串行执行指令的工作方式,一直为今天所有程序设计语言所遵循。而在单处理机系统条件下,对于冯·诺依曼经典结构与程序设计语言的完全依赖,已经把计算机的发展引入了死胡同(虽然尚未到尽头)。

当然,即使在单处理机条件下,由于采取了一系列并行处理技术,也使计算机系统的性能得到了很大改善。例如,现代计算机系统一改冯·诺依曼经典结构以运算器为中心的结构原理,以内存为中心(图 1.4),设置通道,使入/出操作与 CPU 操作并行执行。可以使一道程序由运算转去入/出时,调上另外一道程序继续运算,这就为多道程序运行提供了物质基础。

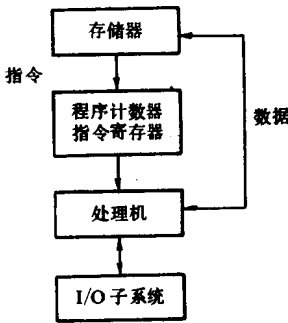


图 1.3 冯·诺依曼结构原理

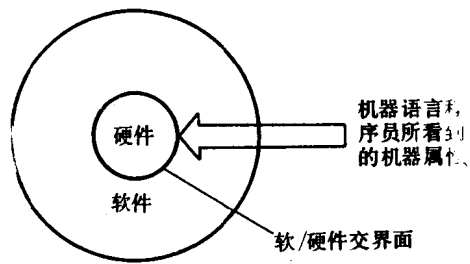


图 1.2 计算机系统的软硬件界面

再如,为了适应计算机应用的深入发展,计算机网络的建立,也势必在单机操作系统的基础上,促进网络操作系统的发展。

但是,正如前面讲到的,在单处理机条件下建立起来的计算机系统,其性能的大幅度提高正面临冯·诺依曼经典结构造成的限制。好在 VLSI 技术的发展,以及计算机应用领域的进一步扩大与深入对于超高性能计算机系统的需求,已经为计算机体系结构的发展提供了雄厚的物质基础,并注入了强大的动力。当前,采用多处理机并行结构构成超高性能计算机已成为主流,并行处理技术已成为近代计算机技术发展的热点。

为了描述计算机体系结构由经典到并行的结构特点,我们可以按照 Flynn 分类法,将体系结构进行分类。1966 年, M. J. Flynn 提出了按指令流和数据流的多倍性状况进行分析:指令流是指机器执行的指令序列;数据流是指由指令流调用的数据序列(包括输入数据和中间结果);多倍性则是指在系统性能瓶颈部件上,处于同一执行阶段的指令流或数据流的最大可能个数。

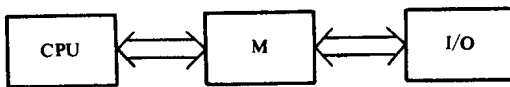


图 1.4 以内存为中心的现代计算机基本结构

根据以上定义,可以将计算机系统分成四类:它们是单指令流单数据流(Single Instruction Stream Single Data Stream, SISD),单指令流多数据流(Single Instruction Stream Multiple Data Stream, SIMD),多指令流单数据流(Multiple Instruction Stream Single Data Stream, MISD),多指令流多数据流(Multiple Instruction Stream Multiple Data Stream, MIMD)。

其中,SISD 实际上就是传统的单处理机结构,尽管在单处理机条件下,也可以采取多种并行处理技术,但其工作方式基本上是在一个指令流的控制下,对于一个数据流的处理。而其它三种,则均为并行结构。特别是 MIMD 结构,则能够实现在作业、任务、指令、数组等各个级别全面并行处理的多处理机系统(图 1.5)。对于这种系统,根据多个处理机是否共享主存空间,

可以进一步分成紧密耦合和松散耦合两种系统。

并行结构的出现与发展,无可置疑地会影响操作系统相应的发展与变化。这就是面向分布式计算环境和并行计算环境的操作系统出现的背景。

### 1.1.2 操作系统的产生与发展

操作系统的产生与发展,源自两方面的动力:其一,作为操作系统的下层基础体系结构的不断发展与进步;其二,操作系统支撑的上层日益复杂多样的应用软件系统的变化,以及使用者的不断增长的要求。特别是前者,更是与之休戚与共。因此,我们先简单观察一下操作系统的发展历史来证明这一点。

#### 1. 手工操作阶段

大约在 40 年代中期,第一台真空管计算机出现,它是个庞然大物,但是速度却比当今世界上最低档的个人计算机还慢许多。人们通过手工操作方式去启动设备和主机运行。用机器语言编制程序。在一个程序员上机期间,整台计算机连同附属设备全被其占用。程序员兼职操作员,效率低下。其特点是手工操作,独占方式。后来人们开发了汇编语言及其汇编编译程序,以及其它一些控制外设的程序等等。但仍属于这一阶段。

#### 2. 监督程序(早期批处理)阶段

相应于硬件上属于第二代计算机系统阶段。

由于处理机速度提高,使得手工操作设备输入/输出信息与计算机计算速度不匹配。例如,第一代计算机用一小时计算一道题目,其中人工干预三分钟。那么当计算机速度提高十倍以后,计算所需时间可能只要六分钟,而手工干预期间不变,仍为三分钟,这就相当于占据 50%的时间用于手工操作。因此,人们设计了监督程序(或管理程序),来实现作业的自动转换处理。这期间,每道作业由程序提供一组在某种介质上准备好的作业信息(文件)。它们是:用作业控制语言书写的作业说明书,相应的程序和数据。由程序员提交给系统操作员。而操作员将作业“成批”地输入到计算机中,由监督程序识别一个作业,进行处理后再取下一个作业。这种自动定序的处理方式称为“批处理”方式。而且,由于是串行执行作业,因此称为单道批处理。

#### 3. 多道批处理

在第二代计算机后期,特别是进入第三代以后,系统软件有了很大发展,它的作用也日益显著。与此同时硬件也有了很大发展,特别是主存容量增大,又出现了大容量的辅助存储器——磁盘以及代理 CPU 来管理设备的通道,使得计算机体系结构发生了很大变化。由以中央处理机为中心的结构改变为以主存为中心,通道使得输入/输出操作与 CPU 操作并行处理成为可能。软件系统也随之相应变化,实现了在硬件提供并行处理之上的多道程序设计。

所谓多道是指它允许多个程序同时存在于主存之中,由中央处理机以切换方式为之服务,使得多个程序可以同时执行。计算机资源不再是“串行”地被一个个用户独占,而可以同时为几

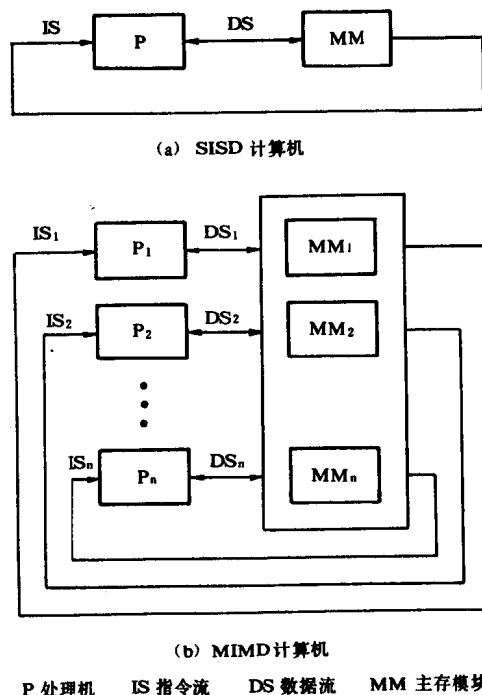


图 1.5 SISD 与 MIMD 计算机

个用户共享,从而极大地提高了系统在单位时间内处理作业的能力。这时,管理程序已迅速地发展成为一个重要的软件分支——操作系统。

#### 4. 分时与实时系统出现

人们不满足于把作业提交给系统操作员后就不能再控制自己作业运行的方式。特别是在编制与调试程序的过程中更需要能与计算机及时交互作用,或称会话。因此,产生了分时系统。所谓分时系统是指多个用户通过终端设备与计算机交互作用来运行自己的作业,并且共享一个计算机系统而互不干扰,就好像自己有一台计算机。

另外,还有些计算机需要对特定的输入做出足够快的反应以便控制发出实时信号的对象。换言之,计算机能及时响应外部事件的请求,在规定的时间内完成对该事件的处理。由这样的需求而产生了实时操作系统。实时系统一般又可分为两种类型:实时控制系统,实时信息处理系统。

至此而知,批处理系统、分时系统和实时系统便是操作系统的三个基本类型。而一个实用的操作系统可以是分别独立的一种系统,亦可是两两结合的或是三者兼而有之的通用操作系统。

#### 5. 网络操作系统

随着数据通讯技术的进步,人们通过数据通信系统,把地理上分散的计算机群和 workstation 设备联结起来,以达到数据通信和资源共享的目的,这便是网络系统。在网络操作系统的支持下用户能够向远方的计算机登录,可以把文件从一台机器复制到另一台。用户感觉到有多台计算机存在,本地机器上运行着自己的操作系统,并且拥有自己的多个或一个用户。

另外,由于硬件的进步,近年来,各种各样的功能强大的 workstation 和高档次的个人微型机的出现,使得局域网络系统更加普及。

网络操作系统的研制与开发是在原来计算机操作系统的基础上进行的,并按照网络系统的体系结构的各种协议标准进行开发。网络操作系统是包括网络管理、通信、资源共享、系统安全和多种网络应用服务在内的诸方面功能的一个系统软件。

#### 6. 操作系统进一步发展

计算机从诞生到现在已经经历了四代。现在,我们再从人们使用计算机的方法,来划分计算机的历史。

50年代,程序员必须事先预约上机时间。当他们上机时,独占全部计算机资源。60年代出现了批处理系统,人们提交作业并等候处理,最后晚些时候拿到计算结果。70年代人们主要使用分时系统,多个用户觉得好像他们在单独使用整个计算机,或者通过终端、本地计算机使用网络,与远地计算机通信,共享资源。80年代是个人计算机时代,每个用户可在自己的办公室或家里有一台属于他自己专用的计算机。

操作系统的发展使得新技术与经济因素方面的结合成为可能。批处理系统的出现是由于主存可以做得充分大,以至可以装下整个操作系统和待处理的多道程序,从而共享昂贵的处理机。分时系统的出现是它可以使程序员获得更快的响应时间。同时,也是由于硬件变得更为便宜而且功能更强大。超大规模集成电路和局域网络技术的发展使 workstation 形成的网络代替了分时系统。这不仅在经济上是可以接受的,而且会获得更大的计算能力。个人机的发展使得人们对操作系统的设计从资源充分利用,特别是更有效地利用处理机的设计观点,转到如何使计算机更有效地为人服务上来。

90年代的工作站功能更加强大,它将具有高分辨率的彩色显示器;并拥有多媒体的输入



输出设备。网络接口支持较高的通信速率,允许若干个通道进行实时视频传输。但是网络操作系统要求用户知道本地对象和远程对象的区别,以及对象放在哪个机器上。如果系统很庞大,这一问题将变得很严重。另外,系统的管理也是个大问题。这主要是因为传统的操作系统从来都不是为具有多个处理机和很多文件系统的环境的设计的。这种环境要求分布式操作系统。90年代将是分布式系统的时代。在分布式系统中,本地操作和远程操作对用户来说都是一样的。在一个工作站上打入的命令所要运行的程序可以在其它工作站上运行。整个系统逻辑上只有一个文件系统,它为所有用户共享,同时用户也可以共享外部设备。处理机可以动态地分配给最需要的地方。分布式系统比通常的集中式系统具有更好的容错性和操作并行性。

分布式系统是由多个相互连接的处理资源组成的计算系统,它们在整个系统的控制下可合作执行一个共同任务,最少依赖于集中的程序、数据或硬件。这些资源可以是物理上相邻的,也可以是在地理上分散的。

对此定义还需进一步说明:

(1) 系统是由多个处理器或计算机系统组成的。

(2) 这些计算资源可以是物理上相邻的,使用机器内部总线或开关网连接的处理器,通过共享主存进行通信;也可以是在地理上分开的,使用计算机通信网络连接的计算机系统,通过报文(message)进行通信。

(3) 这些计算资源组成一个整体,对用户是透明的,即用户使用任何资源时不必知道这些资源在哪里。

(4) 一个程序可分散到各计算机上运行。

(5) 各计算机地位平等,除了受全系统的操作系统控制外,不存在主从控制和集中控制环节。

这种计算机系统属于多指令流多数据流(MIMD)体系结构。目前的分布式系统尚未完全成熟,只有一定程度上的分布特征,但是,已显露出其强大的生命力。

## 1.2 操作系统的功能与定义

在给操作系统下定义之前,我们还是从几个视角察看操作系统,看看操作系统是做什么的。这样,将有助于我们更好地理解什么是操作系统。

### 1.2.1 机器硬件的延伸

可以把一个计算机系统看成是由硬件和软件按层次构成的,如图 1.6 所示。

计算机在机器语言一级的体系结构,大多数是很原始的。对于程序设计,尤其是为 I/O 进行程序设计,显得十分笨拙。这里,不必列举硬件细节就会明白,一般的程序人员不愿意陷入这一泥潭中去。相反,他们希望处理一种简单而又高级的抽象。例如,对磁盘的抽象:磁盘是一个文件卷,它有一批命名的文件,文件可以打开供存取之用,然后可以读写,最后关闭之。而无需关心磁盘上数据的物理位置、各磁道的区段号码、扇区之间的间隙、控制器返回的状态和错误字段,甚至驱动器的电机是否启动,启动延迟时间大小等等硬件细节。能够向程序员隐蔽硬件的真象,对可供读写的文件实现“按名存取”,并作巧妙而简捷的处理,自然是操作系统。另外它还能隐蔽起关于中断、计时和存储器管理等令人生厌的杂务。因此,操作系统是向用户提供一