

Computer

计算机系统软件系列教材

操作系统原理与实践



高等教育出版社

教师信息

- **教师：贾静平**
- **电话：61772520 (O)**
- **Email：**
jingping.jia@ncepu.edu.cn



课程概述

- 课程的地位和任务
- 教学要求
- 课程内容
- 学习要求



课程的地位和任务

“操作系统原理”这门课是计算机科学与技术专业与软件工程专业本科教学计划中的必修专业课程。**操作系统**是计算机系统中最重要的系统软件。本课程主要从资源管理观点出发介绍操作系统的基本概念、基本原理、基本设计与实现技术，使学生对操作系统有较系统的了解。初步具有分析、开发、维护已有操作系统和设计实现操作系统的基本能力。

专业核心课程

考研核心课程

技术开发的理论基础



教学要求

本课程要求学生熟练掌握并理解计算机操作系统中的基本概念、基本原理、基本设计与实现技术。了解当代操作系统的新技术与趋势，熟悉一种实际操作系统的组成及工作原理。



课程主要内容介绍

- **操作系统概述**
- **进程管理**
- **存储器管理**
- **设备管理**
- **文件管理**
- **操作系统安全**



学习要求

- **课前预习教材**
- **按时上课，认真听讲，独立完成作业**
- **整理笔记，认真思考，积极讨论，善于发现问题、提出问题并努力寻求问题的答案**
- **结合操作系统专题学习网站丰富的教学资源,努力寻求问题的答案，掌握基本原理，拓展知识，延伸视野，提高分析问题和解决问题的能力。**



教材选用

计算机系统软件系列教材

《操作系统原理与实践》

邹鹏等编

高等教育社出版

2008年



Computer

操作系统原理与实践

第一章 概述



高等教育出版社

第一章 概述

- **目的与要求：**掌握操作系统的定义及操作系统的地位与作用。理解操作系统的结构及操作系统的发展过程。了解一些常见操作系统的发展历史和结构特点。
- **重点与难点：**操作系统的概念、操作系统的地位与作用、操作系统结构。
- **作业：**1， 3， 7



第一章 概述

- 1.1 什么是操作系统
- 1.2 操作系统硬件基础
- 1.3 操作系统发展历史
- 1.4 操作系统结构
- 1.5 常见操作系统介绍



1.1 什么是操作系统

- 1.1.1 计算机软件体系
- 1.1.2 操作系统的作用
- 1.1.3 操作系统的概念



1.1.1 计算机软件体系

- **计算机软件**是指计算机系统程序及其相应的文档。
- 计算机软件可以划分为**系统软件**、**应用软件**和**介于这两者之间的支撑软件**。
- 计算机系统由硬件和软件组成。（基本构成关系如图1.1所示）



1.1.1 计算机软件体系

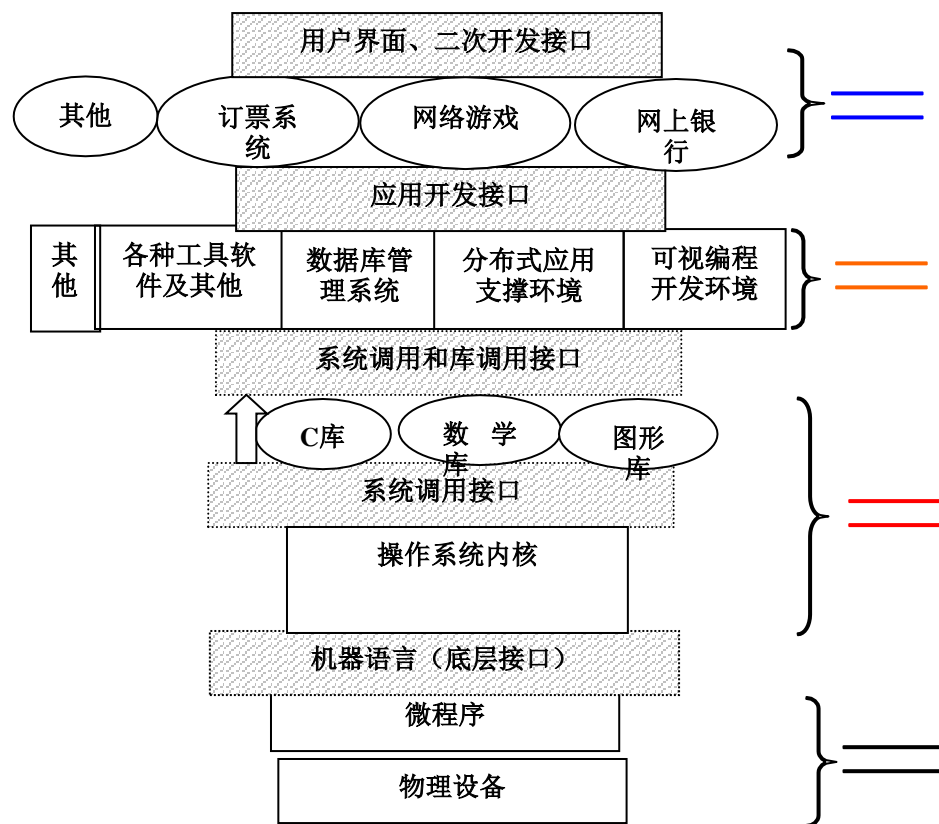


图1.1 计算机系统构成



1.1.2操作系统的作用

1. 操作系统使系统编程接口更简单
 - 操作系统为上层环境提供了系统调用和各种库函数，起到了将物理裸机进行抽象的作用。
 - 内核为用户层程序提供系统调用接口。系统调用可以看成是特殊的公共子程序。



1.1.2操作系统的作用

2. 操作系统使系统资源利用更充分

- 操作系统的介入使进程的并发和共享更易实现。
- 资源共享的两种方法：“时分”和“空分”。
 - **时分**就是由多个用户分时地使用资源。
 - **空分**是针对存储资源而言的，存储资源的空间可以被多个用户共同以分割的方式占用。



1.1.2操作系统的作用

3. 操作系统使程序运行环境更友好

- 操作系统既可以用命令解释程序与用户进行交互，也可以通过对进程的控制与调度来运行用户的程序，为用户提供使用计算机和编程的友好环境。



1.1.3操作系统的概念

- 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,合理地各类作业进行调度,以及方便用户使用的程序集合。是核心系统软件。



1.2 操作系统硬件基础

- **1.2.1 计算机基本组成结构**
- **1.2.2 总线结构**
- **1.2.3 I/O端口寻址和访问控制**
- **1.2.4 I/O控制器和控制卡**
- **1.2.5 引导过程及硬件支持**



1.2.1 计算机基本组成结构

- 计算机硬件系统主要由运算器、存储器、控制器、输入设备和输出设备等主要功能部件组成。结构示意图如图1.2所示。

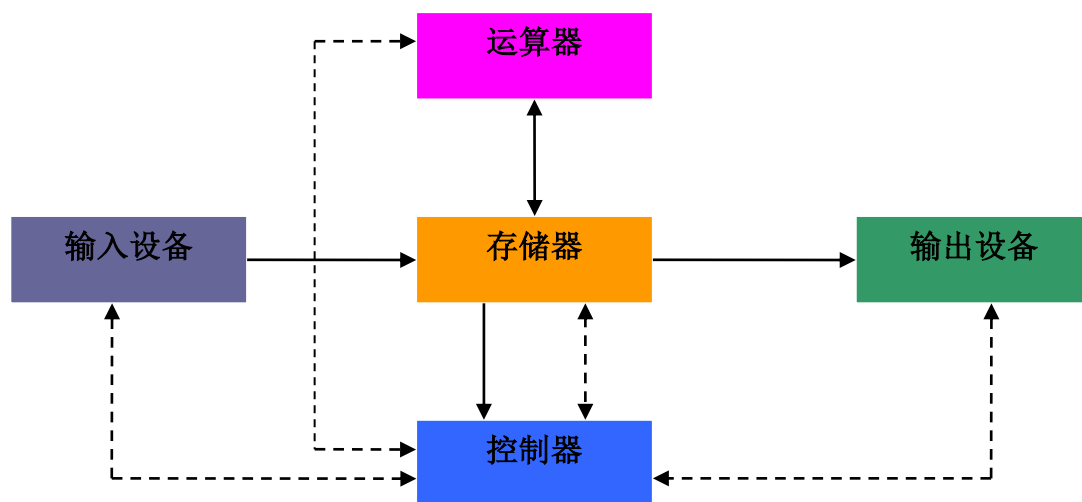


图1.2 计算机基本结构示意图



1.2.1 计算机基本组成结构

- 冯·诺伊曼计算机的五大部件
 - 运算器
 - 控制器
 - 存储器
 - 输入设备
 - 输出设备



1.2.1 计算机基本组成结构

- 网络设备
- 中断结构
- 总线结构



1.2.2 总线结构

- 总线的技术规范通常包括：
 - 机械结构规范：确定模块尺寸、总线插座插头、边沿连接器等规格及位置。
 - 功能规范：确定每条信号线（引脚）的名称、功能、时序及相互作用。
 - 电气规范：规定信号线的有效电平、动态转换时间、负载能力及最大额定值等。



1.2.2 总线结构

- 总线（**bus**）是计算机系统的互连结构之一，是一组信号线的集合，是计算机各部件间传输数据和命令的公共通道。
- 自**IBM PC**机问世以来，出现过**PC/XT**、**ISA**、**MCA**、**EISA**、**VESA**、**PCI**、**AGP**、**IEEE 1394**、**USB**等总线类型。



1.2.2 总线结构

- 计算机系统中采用总线来连接各功能部件，但不同的连接方式和总线数量可以形成多种总线连接结构。
 - 单总线
 - 双总线
 - 三总线（图1.3所示一种三总线结构）



1.2.2 总线结构

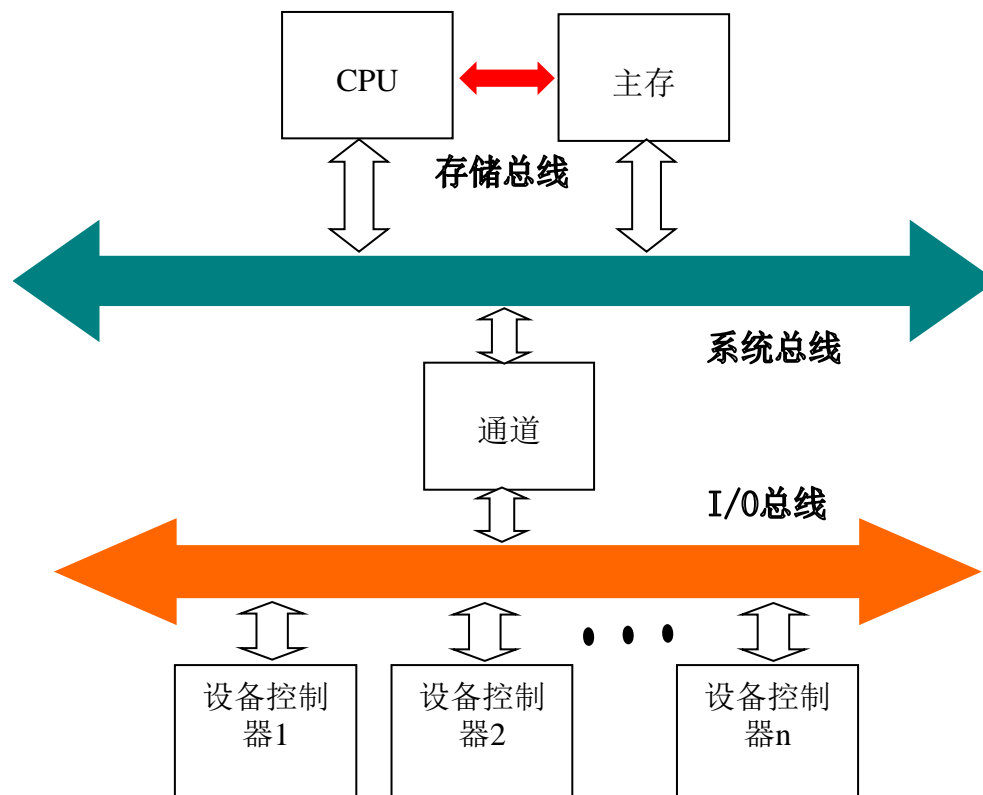
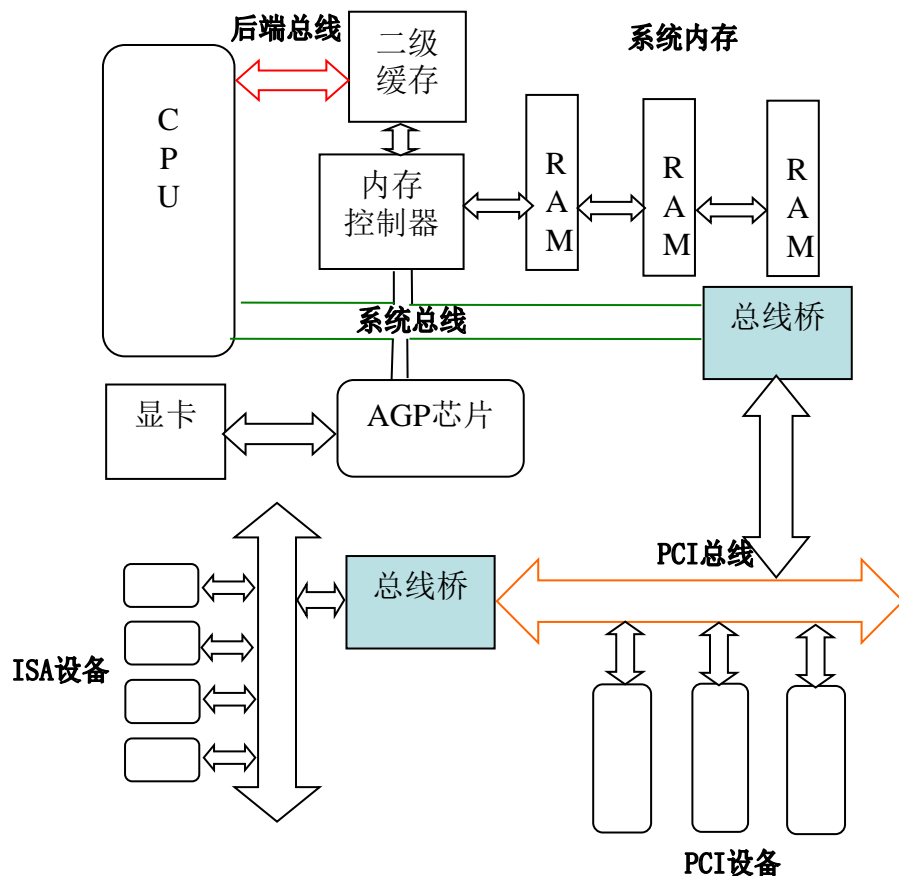


图1.3 一种三总线结构计算机系统



1.2.2 总线结构



- 在PC计算机中，由于成本和通用性的考虑，更常见的总线结构和连接方式如图1.4。

图1.4 总线和CPU之间的连接方式



1.2.2 总线结构

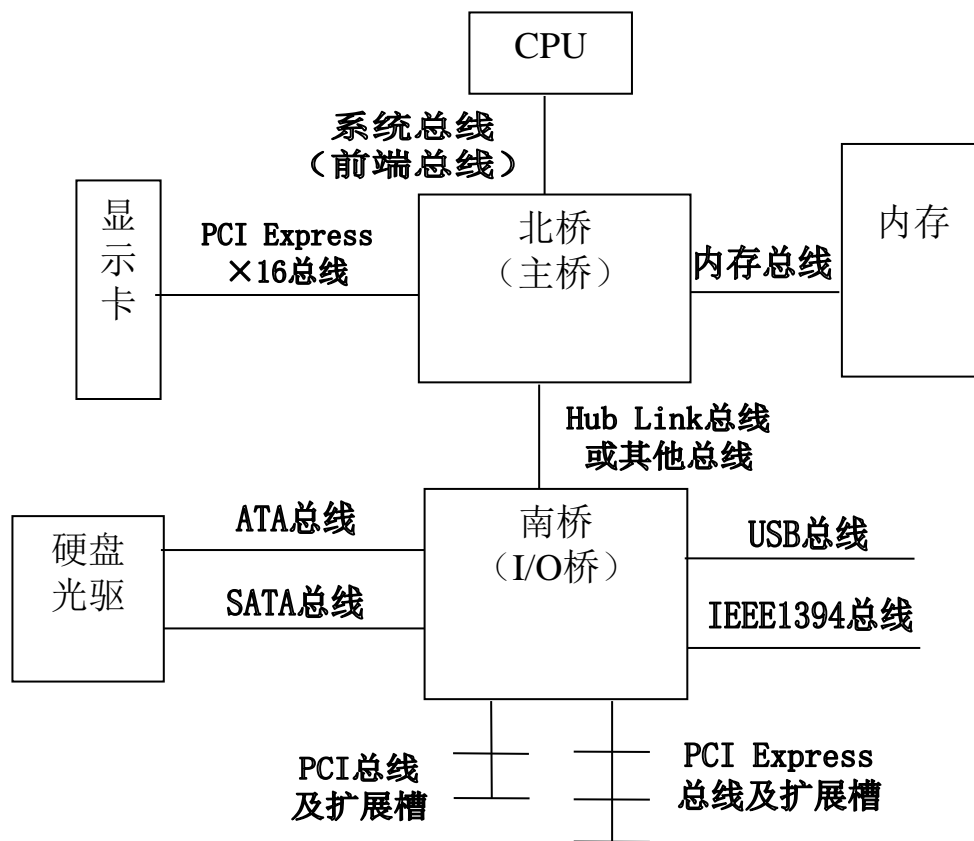


图1.5 以PCI Express为主I/O总线的奔腾PC总线体系



1.2.3 I/O端口寻址和访问控制

- I/O端口编址方式
 - 统一编址
 - 独立编址
- I/O控制方式
 - 轮询
 - 中断
 - **DMA**



1.2.4 I/O控制器和控制卡

- 中断控制器
- DMA控制器
- 定时/计数器
- 键盘控制器
- 串行控制卡
- 其他



*Ubuntu

Advanced options for Ubuntu
Memory test (memtest86+)
Memory test (memtest86+, serial console 115200)
Ubuntu 16.04.3 LTS (16.04) (on /dev/sda1)
Advanced options for Ubuntu 16.04.3 LTS (16.04) (on /dev/sda1)

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the commands before booting or 'c' for a command-line.
The highlighted entry will be executed automatically in 9s.

1.3 操作系统发展历史

- 1.3.1 无操作系统阶段
- 1.3.2 单道批处理系统
- 1.3.3 多道批处理系统
- 1.3.4 分时操作系统
- 1.3.5 实时操作系统
- 1.3.6 嵌入式操作系统
- 1.3.7 网络操作系统
- 1.3.8 分布式操作系统



1.3.1 无操作系统阶段



无操作系统阶段—第一代计算机



1.3.2 单道批处理系统

- 20世纪50年代末期，出现了脱机输入输出技术。图1.6演示了脱机输入输出的过程。

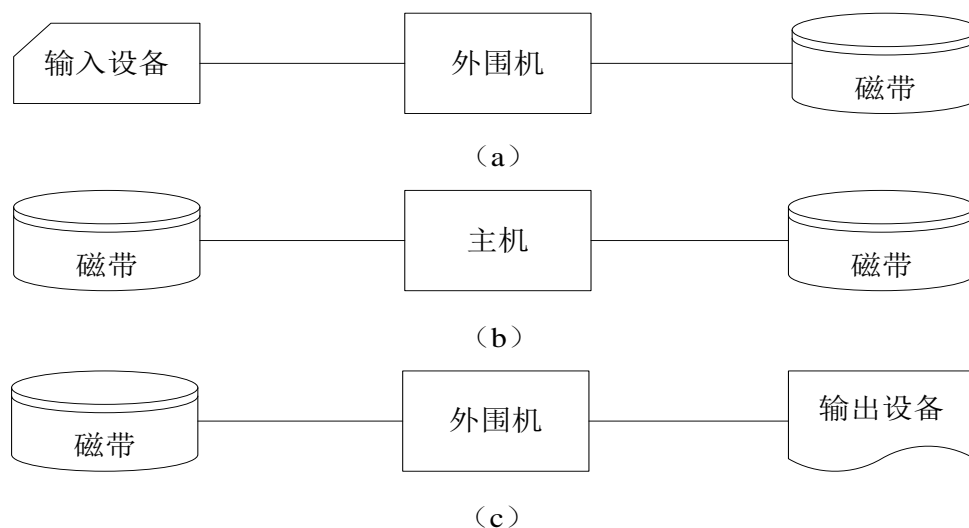


图1.6 脱机I/O示意图



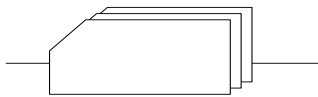
1.3.2 单道批处理系统

- 在**20世纪50年代后期到60年代中期**，随着第二代计算机的出现，在脱机输入输出系统的基础之上，单道批处理系统被开发出来。单道批处理系统的核心思想是使用一个被称为**监督程序(monitor)**的软件控制批量作业的调度，每次只调度一个作业进入内存运行（单道）。



1.3.2 单道批处理系统

单道批处理系统的操作如图1-7所示



打印输出



1.3.2 单道批处理系统

- 典型的输入作业的结构如图1.8所示。

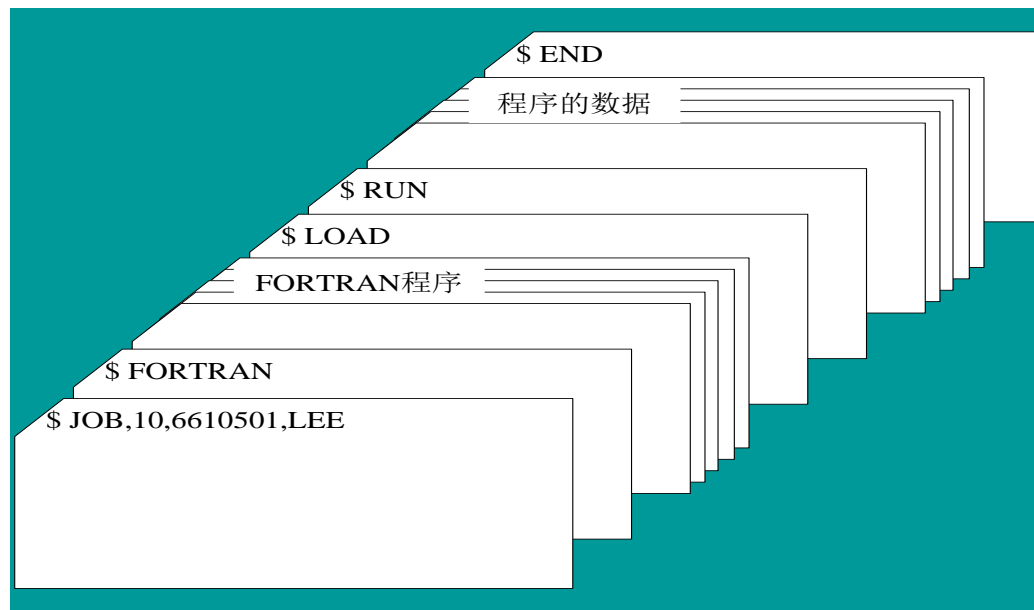


图1.8典型的FMS作业的结构

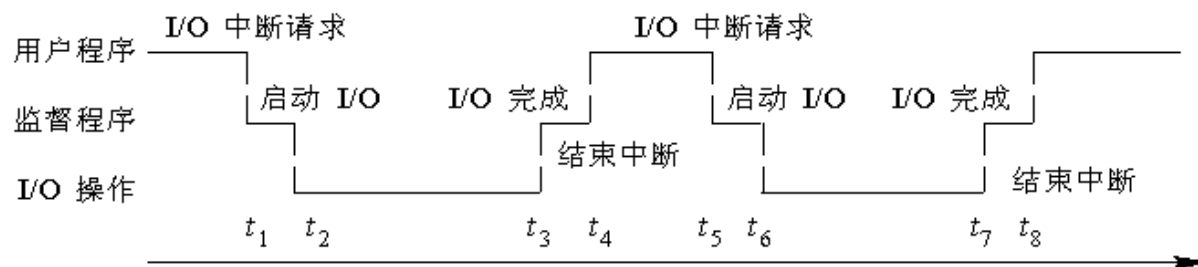


1.3.3 多道批处理系统

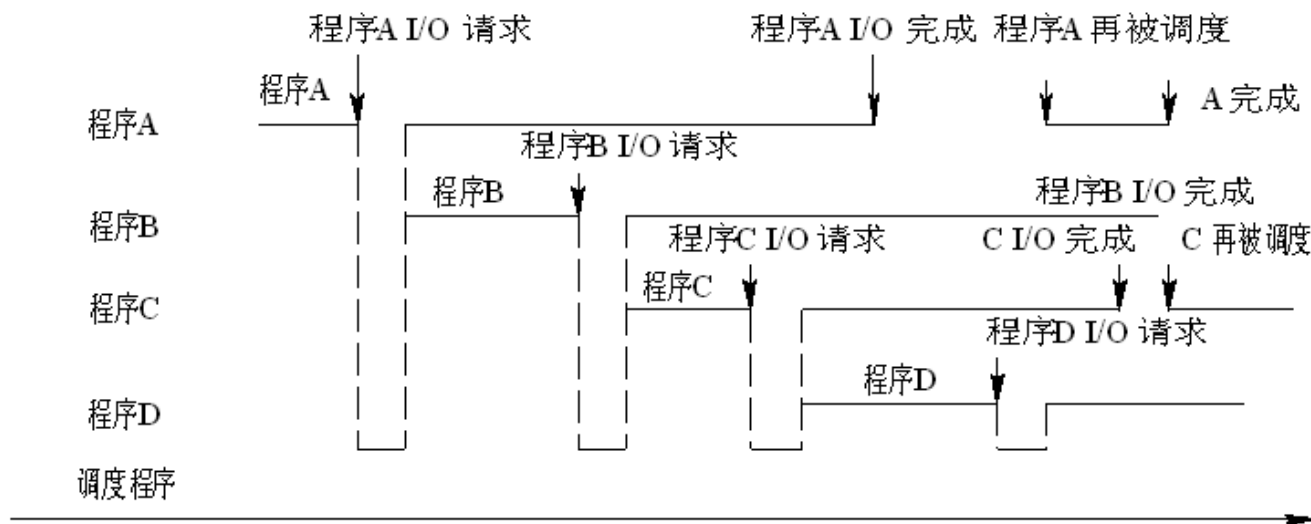
- 在单道批处理的计算机上，如果当前作业因为等待磁带或其他I/O操作而暂停时，**CPU**只能原地踏步直到该I/O操作完成。对于**CPU**密集操作的科学计算问题，由于I/O而浪费的**CPU**时间很少。但是对于I/O密集操作的商业数据处理来说，**CPU**等待I/O的时间往往达到**80%—90%**，这种浪费显然太昂贵了。因此同时具有科学计算和事务处理需求的用户往往同时需要一台较大的计算机和一台较小的计算机。



1.3.3 多道批处理系统



(a) 单道程序运行情况



(b) 四道程序运行情况



1.3.3 多道批处理系统

- 20世纪60年代，IBM System/360 系统为科学计算提供了包括大、中、小型计算机在内的指令系统。操作系统。操作系统设计



1.3.3 多道批处理系统

- 用户提交的作业被存放在外存上并排成队列，称为“后备队列”。然后作业调度程序根据调度算法将多个作业调入内存。内存被划分为几个分区，每个分区存放不同的作业，如图1.9所示。

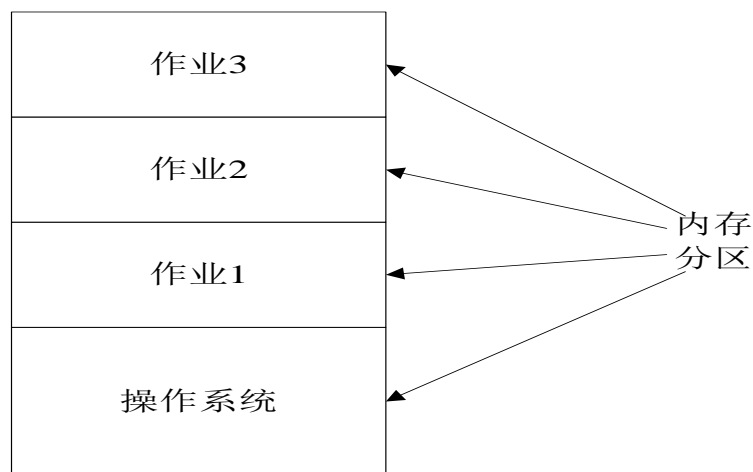


图1.9一个内存中有3道作业的多道程序系统



1.3.3 多道批处理系统

- 例子：某计算机系统，有**256KB**的主存(不包含操作系统)，一个磁盘，一个终端和一台打印机。同时提交的三个作业分别命名为**JOB1**、**JOB2**、**JOB3**。各作业运行时间分别为**5min**、**15min**和**10min**。它们对资源的使用情况如表1.1所示：



1.3.3 多道批处理系统

表1.1 三个作业对资源的使用情况

作业名	JOB1	JOB2	JOB3
作业类型	CPU	I/O	I/O
占用内存	50K	100K	80K
需要磁盘情况	NO	NO	YES
需要终端情况	NO	YES	NO
需要打印机情况	NO	NO	YES
运行时间	5分钟	15分钟	10分钟



1.3.3 多道批处理系统

- 三个作业单道和多道的各项技术指标计算结果如表1.2所示：

表1.2单道运行与多道运行平均资源利用率情况对比

	单道	多道
CPU利用率	$17\% = 5 / (5 + 15 + 10)$	$33\% = (5 / 15)$
主存利用率	$30\% = (50 + 100 + 80) / 256 / 3$	$90\% = (50 + 100 + 80) / 256$
磁盘利用率	$33\% = 10 / 30$	$67\% = 10 / 15$
打印机利用率	$33\% = 10 / 30$	$67\% = 10 / 15$
吞吐量/(作业·h-1)	$6 = 3 / (30 / 60)$	$12 = 3 / (15 / 60)$
平均周转时间(min)	$18 = (5 + 20 + 30) / 3$	$10 = (5 + 15 + 10) / 3$



1.3.3 多道批处理系统

- 多道程序设计提高系统性能的同时，由于对计算机资源的共享与竞争也增加了系统的复杂性。于是在多道批处理系统中增加了处理机管理、存储管理、I/O设备管理、文件管理、作业管理等功能。多道批处理系统的出现标志着真正意义的操作系统的诞生。
- **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,合理地各类作业进行调度,以及方便用户使用的程序集合。



1.3.4 分时操作系统

- 分时系统是指在一台主机上连接了多个带有显示器和键盘的终端，同时允许多个用户通过自己的终端，以交互的方式使用计算机，共享主机中的资源。
- 这里的分时是指多个用户分时使用**CPU**的时间。将**CPU**的单位时间划分成若干时间段，每个时间段称为一个时间片（**Time Slice**）。并按时间片把**CPU**分配各联机用户使用。



1.3.4 分时操作系统

- 分时系统较好的解决了批处理系统交互性不好的问题，使多用户能方便廉价的使用计算机。它的主要特点有如下四点：
 - 1. 多路性。
 - 2. 独立性。
 - 3. 交互性。
 - 4. 及时性。



1.3.5 实时操作系统

- 实时系统要求及时响应外部事件的请求并及时进行处理。这种应用可以被分成两类
 - 实时控制系统
 - 实时信息处理系统。



1.3.5 实时操作系统

- 实时系统与分时系统相比有一些不同的特点。
 - 1. 分时系统属于一种通用的操作系统。而实时系统是一种专用的操作系统，本身含有实时控制或实时信息处理的专用程序。
 - 2. 分时系统与实时系统在实时性上不一样。
 - 3. 分时系统与实时系统在交互性上不一样。
 - 4. 分时系统与实时系统在可靠性上不一样。



1.3.7 网络操作系统

- 网络操作系统是指能使网络上的计算机方便而有效的共享网络资源，为用户提供所需的各种服务的操作系统软件。应该具有下列功能：
 - (1) 提供高效可靠的网络通讯能力；
 - (2) 提供多项网络服务功能如远程管理、文件传输、电子邮件、远程打印等。
- 网络操作系统主要运行在各种服务器上，目前主要有**UNIX**、**Linux**、**Windows**以及**Netware**系统等。



1.3.6 嵌入式操作系统

- 嵌入式操作系统是一种实时的、支持嵌入式系统应用的操作系统软件，通常包括与硬件相关的底层驱动软件、系统内核、设备驱动接口、通信协议、图形界面、标准化浏览器 **Browser** 等。
- 目前，嵌入式操作系统的品种较多，仅用于信息电器的嵌入式操作系统就有**40**种左右，其中较为流行的主要有：**Windows CE、Palm OS、Real-Time Linux、VxWorks、pSOS、PowerTV**等。



1.3.8 分布式操作系统

- 多处理机协同工作的方式可以被分为两类：
 - 紧密耦合方式
 - 松散耦合方式。
- 分布式操作系统就是运行在松散耦合的方式下的。在分布式操作系统的作用下，整个联网的计算机群的行为像是一台单处理机的计算机一样。



1.3.8 分布式操作系统

- 分布式操作系统具有以下特征：
 - (1) 它是一个统一的操作系统；
 - (2) 所有的分布式系统中的资源深度共享；
 - (3) 系统对用户透明，对用户来说，系统像是一台单一计算机；
 - (4) 处于分布式系统的多个主机都是平等的，没有主从的概念。



1.3.9操作系统的特征

- 并发性
- 共享性
- 虚拟性
- 异步性



1.并行性

- 并行性是指两个或多个事件在同一时间发生。
- 并行性是指两个或多个事件在同一时间间隔内发生。（与并行相似，但多指硬件支持）
- 程序的并发执行，有效地改善了系统资源的利用率和提高了系统的吞吐量，但它使系统复杂化，操作系统必须具有控制和管理各种并发活动的能力。



2.共享性

- 操作系统与多个用户的程序共同使用计算机系统
中的资源(互斥共享、同时访问)
- 资源共享是指系统中的硬件和软件资源不再为某
个程序所独占，而是供多个用户共同使用。
- 并发和共享是操作系统两个最基本的特征，这两
者之间又是互为存在条件的。
- 资源共享是以程序的并发为条件的，若系统不允
许程序并发执行，自然不存在资源共享问题。
- 若系统不能对资源共享实施有效的管理，也必将
影响到程序的并发执行，甚至根本无法并发执行。



3.虚拟性

- 所谓虚拟，是指把一个物理上的实体，变为若干个逻辑上的对应物。物理实体(前者)是实的，而后者是虚的，相应地，用于实现虚拟的技术，称为虚拟技术。在**OS**中利用了多种虚拟技术，分别用来实现虚拟处理机、虚拟内存、虚拟外部设备和虚拟信道等。



4.异步性(Asynchronism)

- 在多道程序环境下，允许多个进程并发执行，但由于竞争资源等因素的限制，使进程的执行不是“一气呵成”，而是以“走走停停”的方式运行。
- 多道程序环境下程序的执行，是以异步方式进行的；每个程序在何时执行，多个程序间的执行顺序以及完成每道程序所需的时间都是不确定和不可预知的。进程是以人们不可预知的速度向前推进，此即进程的异步性。



1.3.10 操作系统的主要功能

从资源管理观点看，操作系统具有五大功能：

- **处理机管理**：是创建和撤销进程（线程），对诸进程（线程）的运行进行协调，实现进程之间的信息交换，以及按一定的算法把处理机分配给进程。
- **存储器管理**：内存分配、内存保护、地址映射和内存扩充等功能。



1.3.10 操作系统的主要功能

- **设备管理：**管理计算机系统中所有的外围设备。
- **文件管理：**对用户文件进行管理，以方便用户，并保证文件的安全性。
- **用户接口：**为方便用户使用，向用户的“用户和操作系统的接口”。接口包括命令接口、程序接口及图形接口。



1.4 操作系统结构

- 1.4.1 整体式结构
- 1.4.2 分层式结构
- 1.4.3 虚拟机结构
- 1.4.4 微内核结构



1.4.1 整体式结构

- 程序设计方法是将总体功能划分为若干子功能，子功能再往下细分，直至最基本的功能为止。实现每个子功能的程序称为模块。整个系统就是由接口将所有模块连接起来的一个整体。



1.4.1 整体式结构

- 结构模型（如图1.10所示）：
 - 1) 有一个主程序，用来调用请求的服务过程。
 - 2) 有一套服务过程，用来执行系统调用。
 - 3) 有一套实用过程，用来辅助服务过程。



1.4.1 整体式结构

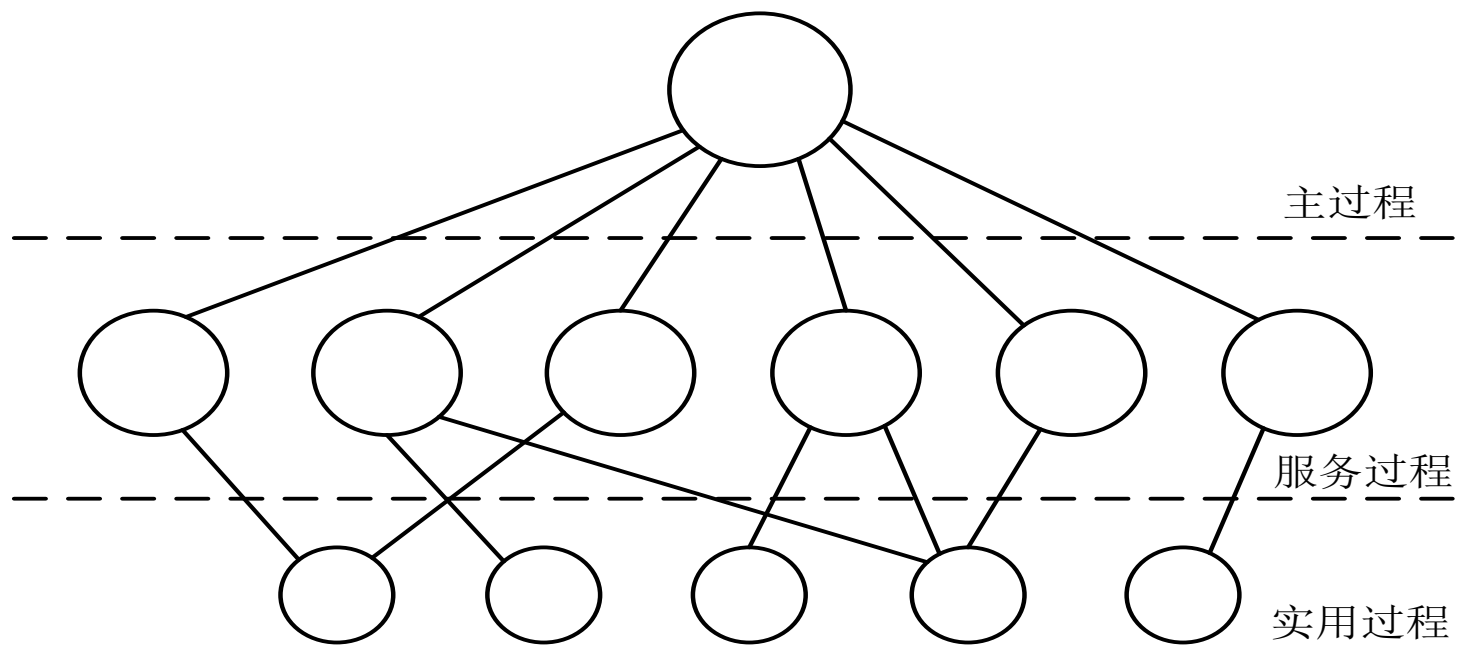


图1.10整体式结构的结构模型



1.4.2 分层式结构

- 在硬件裸机的基础上一层一层向外扩充软件，其中低层软件为高层软件提供服务，高层软件通过调用底层软件实现其功能。
- 第一个分层式系统是**E.W.Dijkstra**和他的学生开发的**THE**系统（**1968**年）。它共分为**6**层，如图**1.11**所示。后来的**MULTICS** 系统也采用了分层式结构。



1.4.2 分层式结构

层号	功能
5	操作员
4	用户程序
3	输入/输出管理
2	操作员进程通信
1	存储器和磁鼓管理
0	处理机分配和多道程序设计

图1.11 THE操作系统的结构



1.4.3 虚拟机结构

- **VM/370:** 系统的核心被称为虚拟机监控器(**virtual machine monitor**)。它在裸机上运行并且具备多道程序设计功能。该系统向上层提供多台虚拟机，每台虚拟机是对裸机的精确复制。每台虚拟机上都可以运行一台裸机所能够运行的任何类型的操作系统。**VM/370**的结构如图**1.12**所示。



1.4.3 虚拟机结构

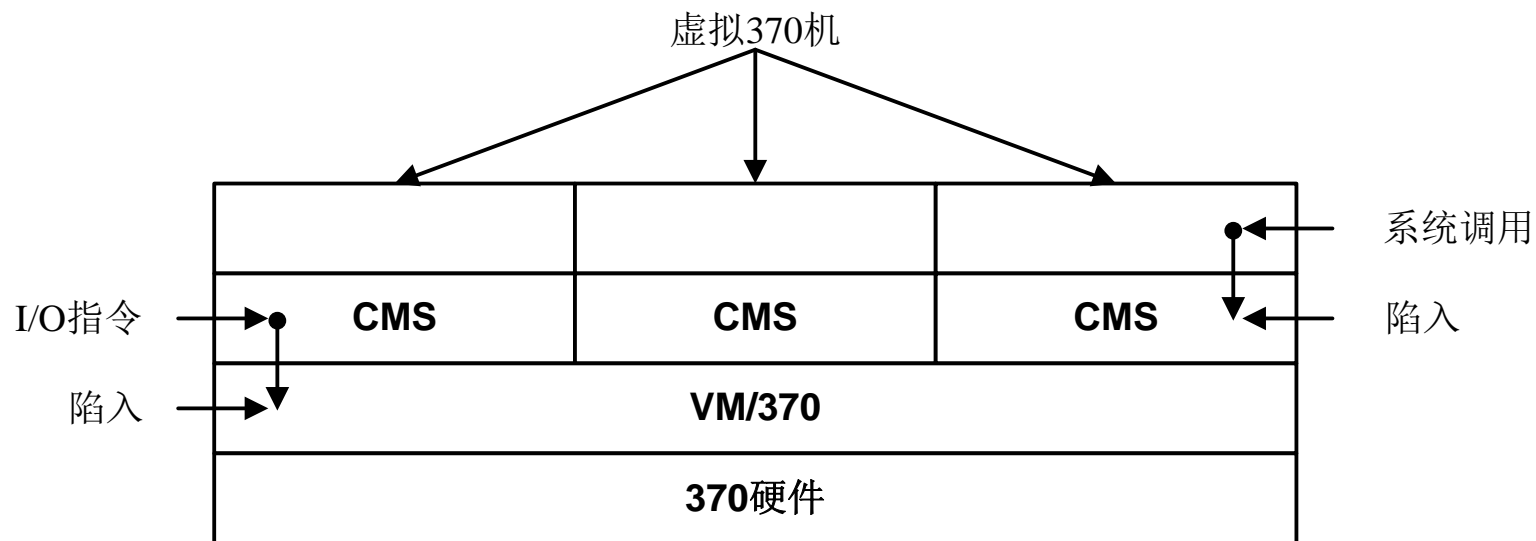


图1.12 配有CMS的VM/370的结构



1.4.3 虚拟机结构

- 虚拟机技术新的发展
 - **VMware**虚拟机技术
 - **JAVA**虚拟机技术



1.4.4 微内核结构

- 现代操作系统的趋势是尽可能的将代码移出核心，只保留一个很小的微内核。微内核的目标是将系统服务与系统的最基本操作分离开来。按照这种目标，**OS**被分成服务器和**OS**核心两部分。所有的服务器都工作于用户态，内核由一个非常简单的硬件抽象层和一组比较关键的原语或系统调用组成，用来处理客户程序与服务器之间的通信过程。结构模型如图1.13所示。



1.4.4 微内核结构

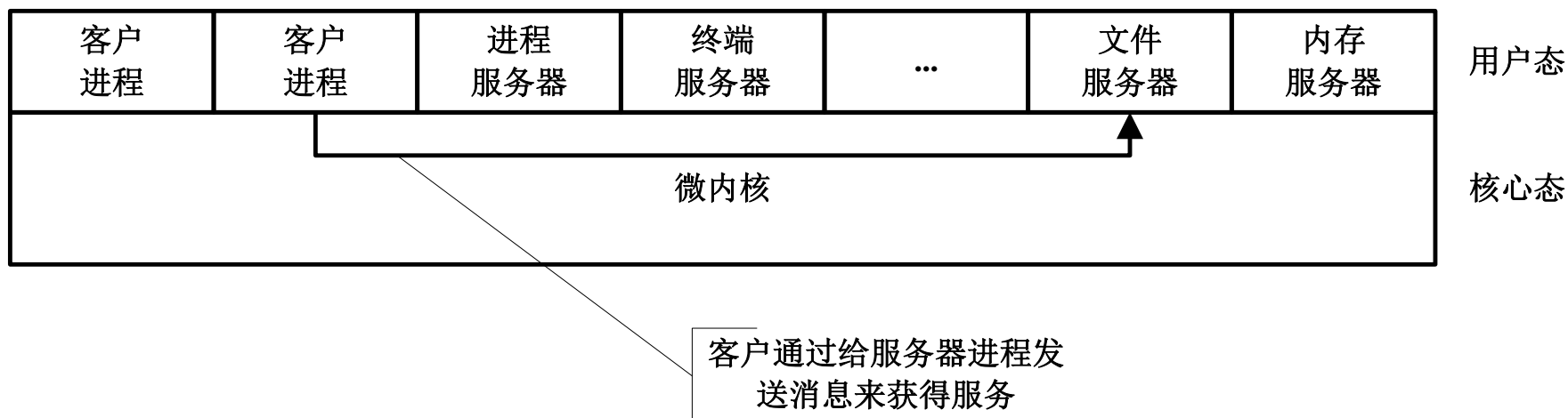


图1.13微内核结构模型



1.4.4 微内核结构

- 微内核结构提高了系统的简单性，灵活性和可扩充性。
- 微内核结构提高了系统的可靠性。
- 微内核结构支持多处理机运行，适用于分布式系统（如图1.14所示）。



1.4.4 微内核结构

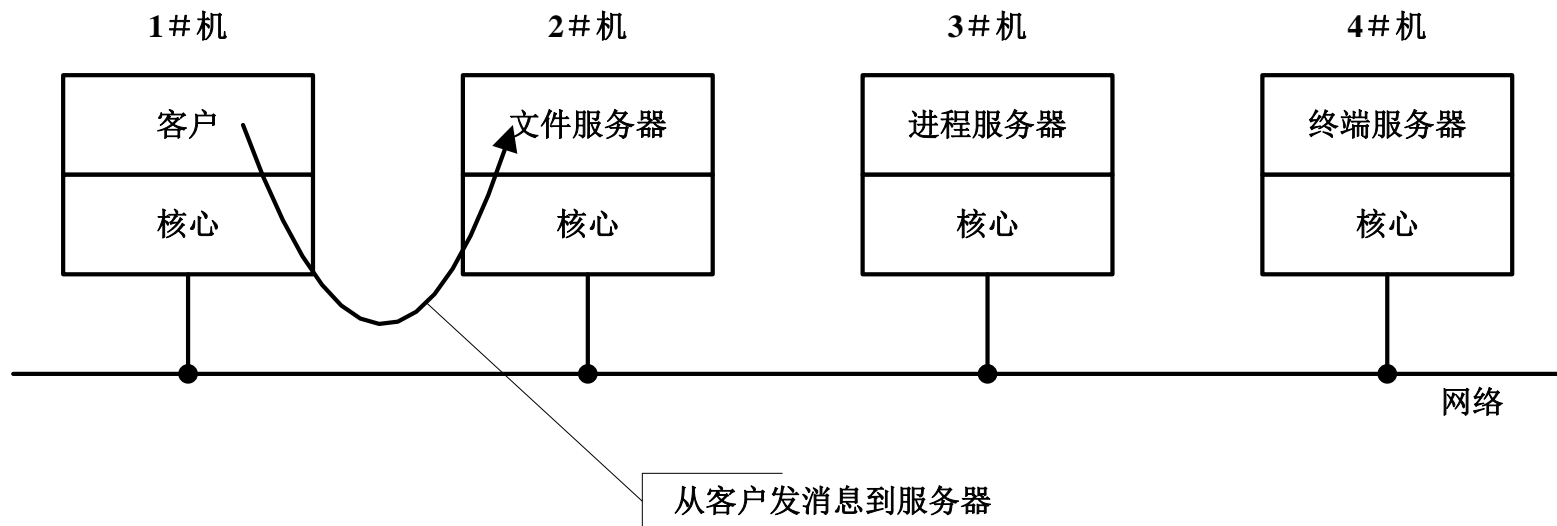


图1.14 分布式系统的客户/服务器模型



1.5 常见操作系统介绍

- 1.5.1 UNIX操作系统
- 1.5.2 Windows操作系统
- 1.5.3 Linux操作系统
- 1.5.4 Kylin操作系统



1.5.1 UNIX操作系统

- **UNIX最早是由Ken Thompson、Dennis Ritchie在AT&T的贝尔实验室开发出来的。**
- **最初的UNIX是用汇编语言编写的，不具备可移植性。1973年Thompson和Ritchie等人用C语言重写了UNIX，使UNIX具有了可移植性。**
- **UNIX系统的各种版本比较多，从流派上可以分为BSD UNIX和AT&T UNIX系统，一个来自加州大学伯克利（Berkeley）分校的研究型版本，另一个来自AT&T的商业型版本。**



1.5.1 UNIX操作系统

- UNIX系统分为内核和外壳两部分。它的系统结构如图1.15所示。

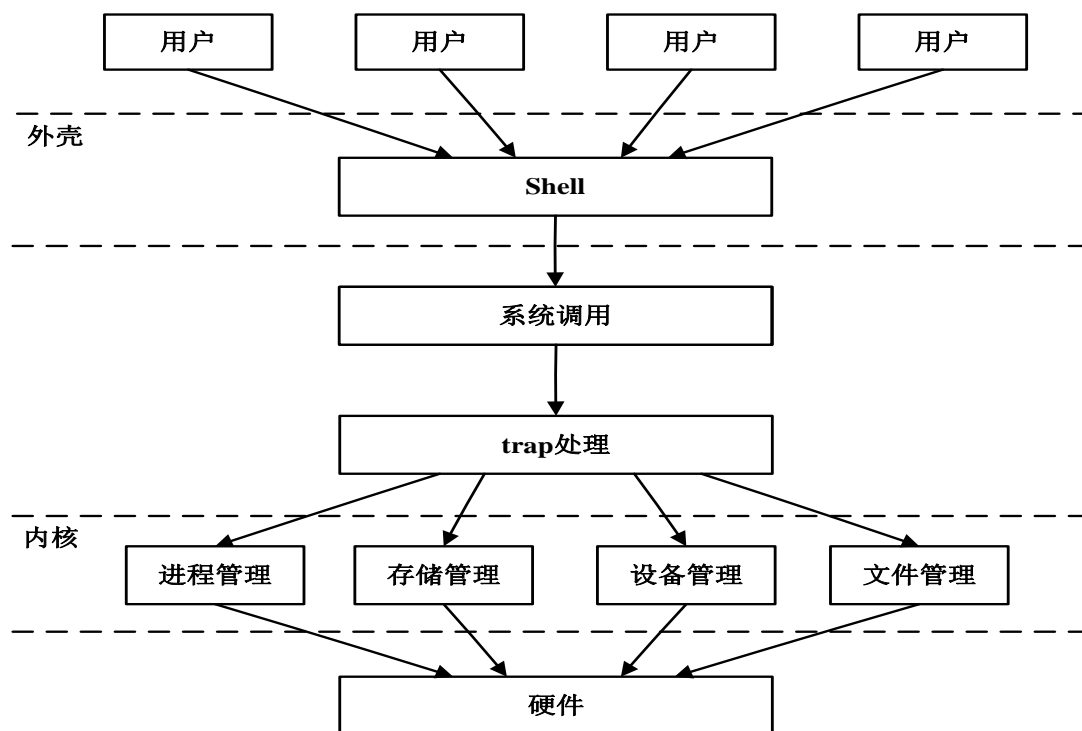


图1.15 UNIX操作系统的层次结构关系



1.5.1 UNIX操作系统

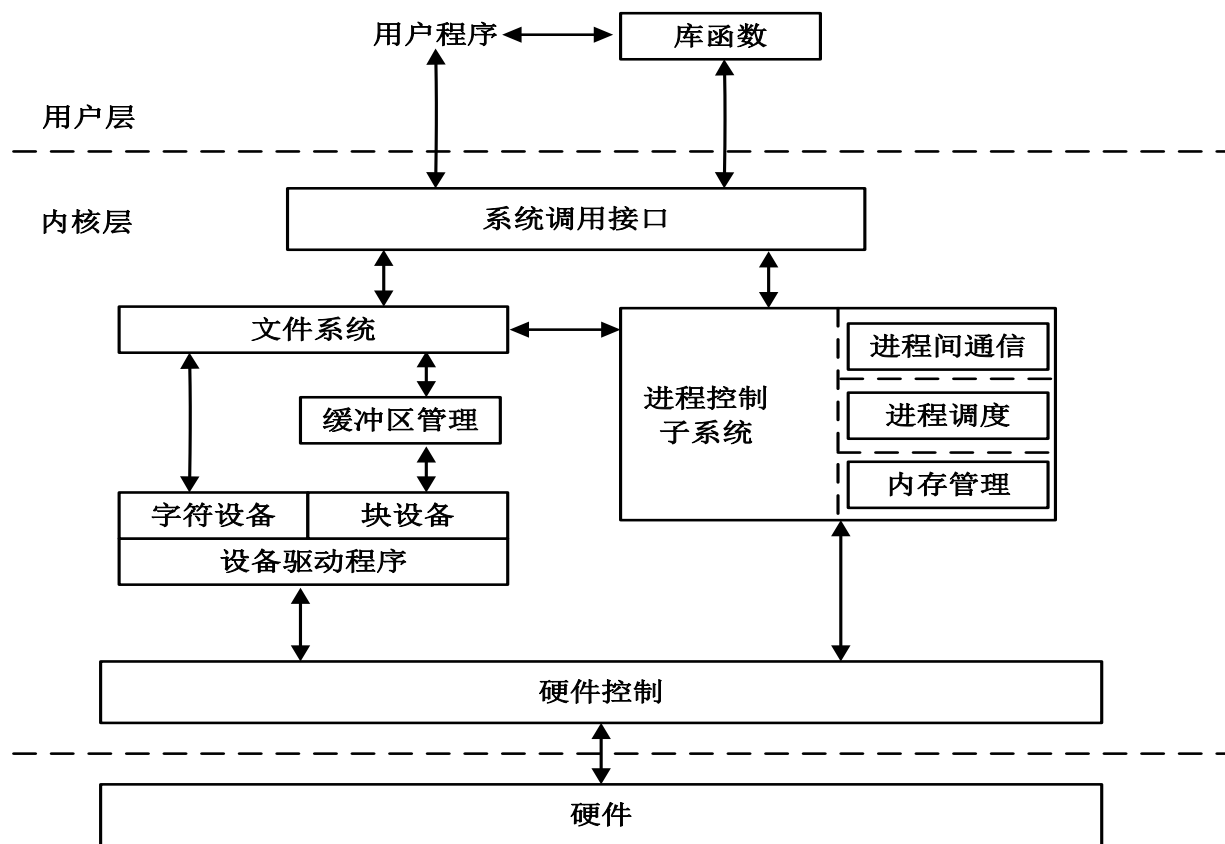


图1.16 UNIX系统内核分块图



1.5.2 Windows操作系统

- 由**Micorsoft**开发，演变过程经历了两条主线。如图1.17所示



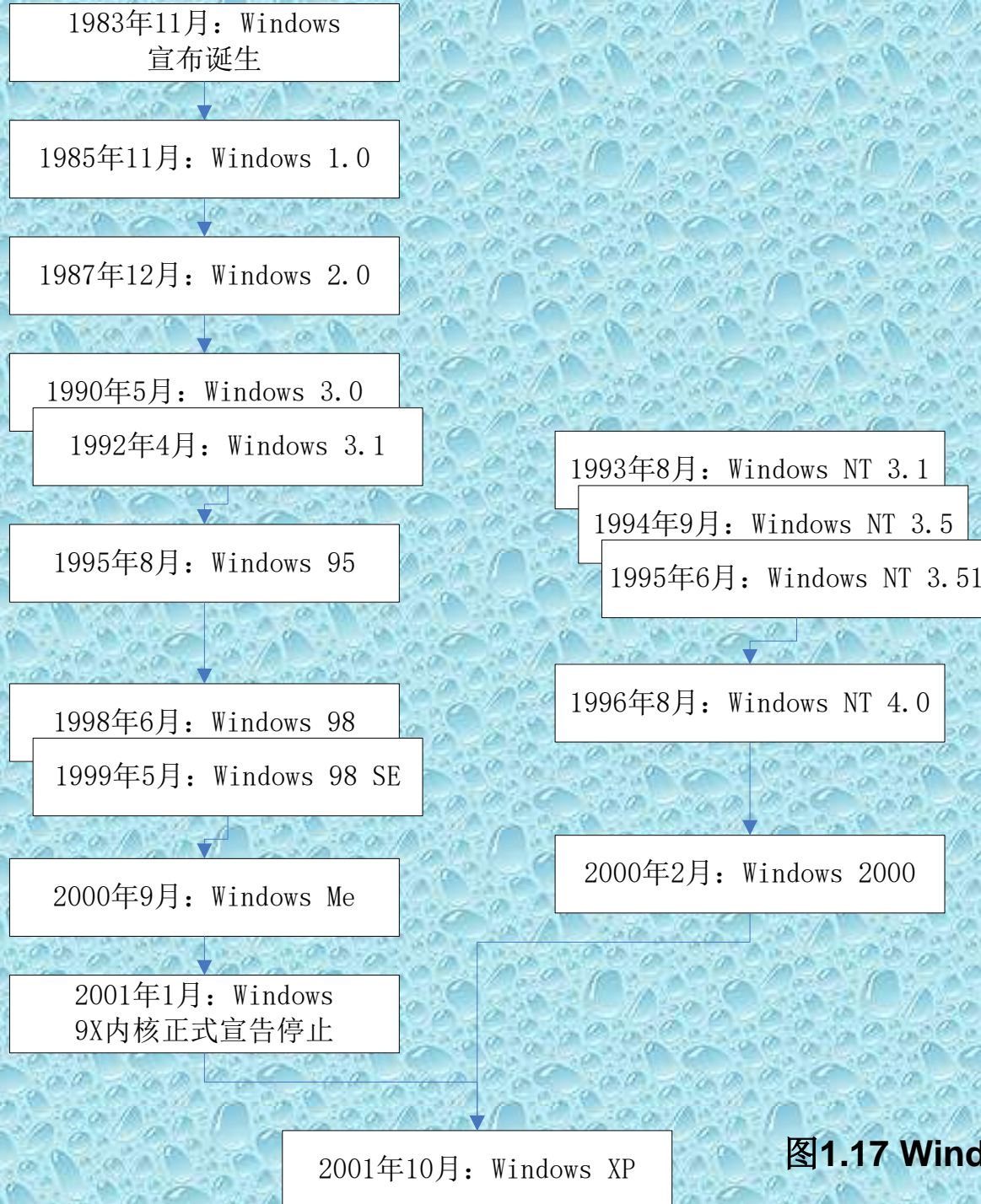


图1.17 Windows发展演变

1.5.2 Windows操作系统

- Windows 2000系统体系结构特点。见图1.18所示。

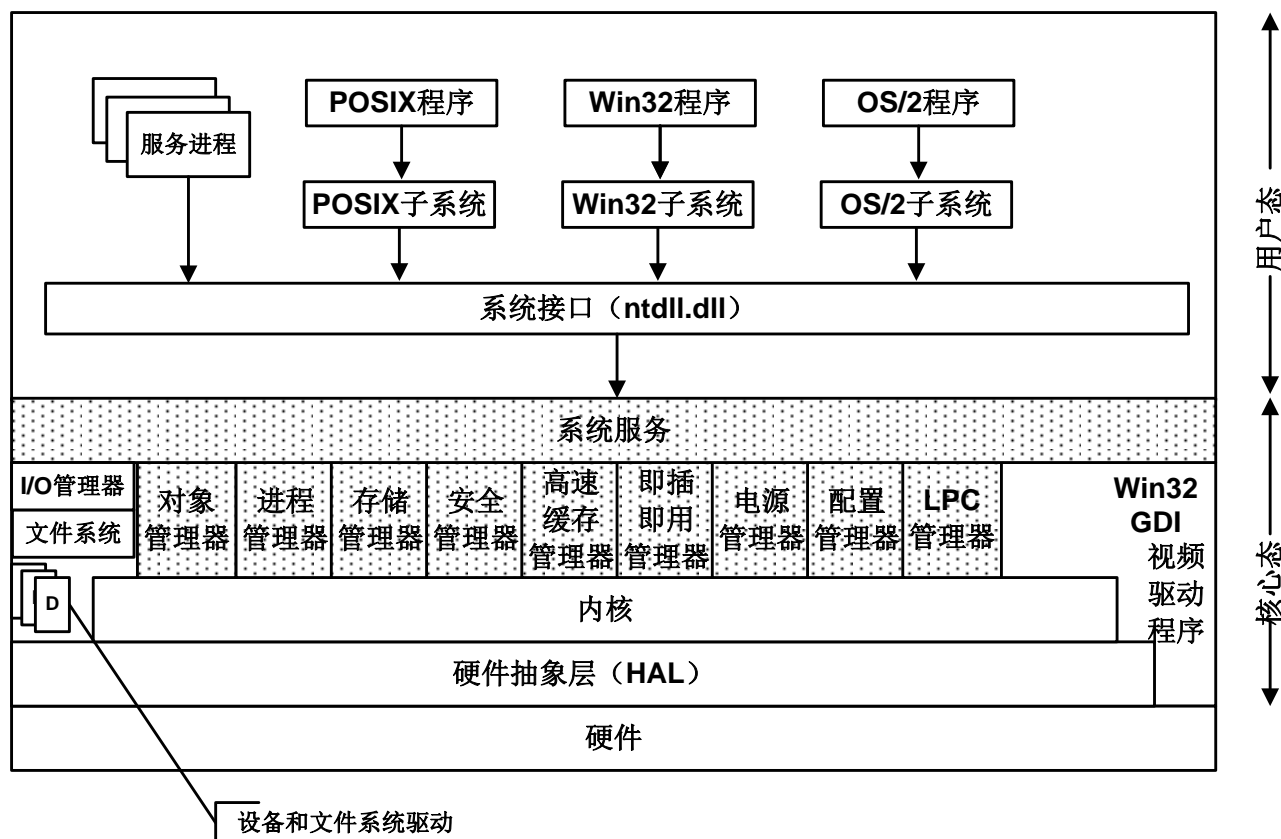


图1.18 Windows 2000操作系统的体系结构图



1.5.3 Linux操作系统

- **Linux操作系统是UNIX操作系统的一种克隆系统。**
- **Linus Torvalds 最早写出了Linux系统。**
- **Linux操作系统的诞生、发展和成长过程始终依赖着以下五个重要支柱：UNIX操作系统、MINIX 操作系统、GNU 计划、POSIX标准和Internet网络。**



1.5.3 Linux操作系统

- Linux系统体系结构特点。如图1.19所示。

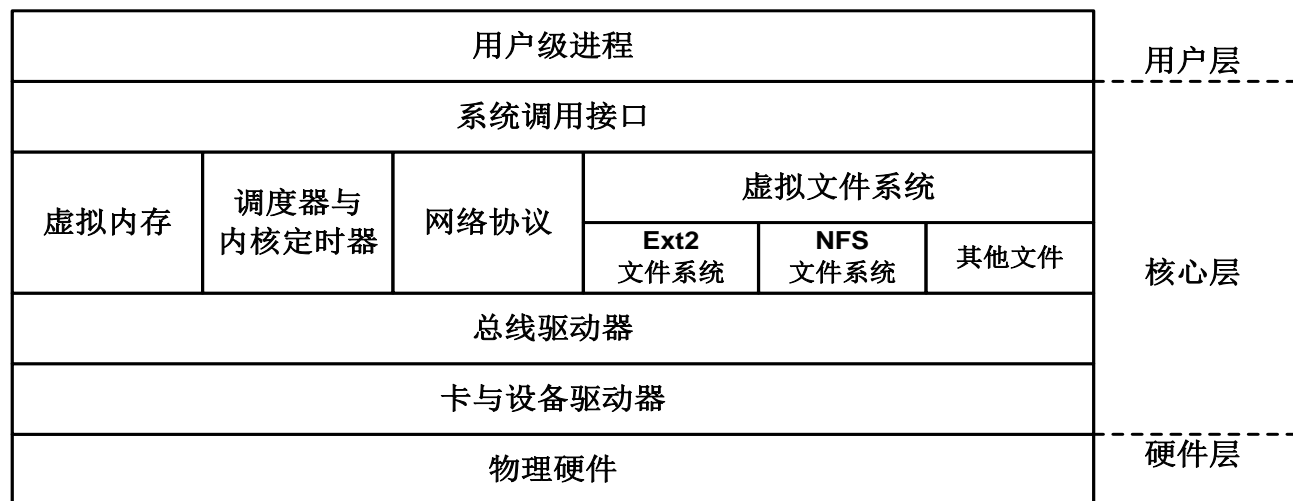


图1.19 Linux系统内核框图



1.5.4 Kylin操作系统

- 采用了层次式内核结构
- 安全等级达到结构化保护级
- 能支持多种微处理器和多种计算机体系结构并
- **Linux**二进制代码兼容
- 在模块化设计、高可信计算框架和服务服务器自治技术等方面均具有鲜明特色



1.5.4 Kylin操作系统

- 麒麟操作系统是在国家十五**863**重大软件专项的支持下，以国防科技大学为主、联合中软网络股份有限公司、联想控股有限公司、浪潮电子信息股份有限公司和民族恒星科技有限公司共同自主研发的服务器操作系统。



1.5.4 Kylin操作系统

- 主要技术特点
 - 采用层次式结构
 - 硬件支持出色
 - 支持多种文件系统
 - 具有系统级容错、高可靠性
 - 友好的桌面环境和开发环境

