

操作系统原理与设计

第一章绪论

陈香兰

中国科学技术大学计算机学院

2009年09月01日

提纲

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

Outline

What is OS?

操作系统的定义和目标

操作系统的作用

操作系统的层次模型

操作系统的发展动力

操作系统的种类

操作系统的发展过程

1945~1955, 无操作系统

1955~1965, 批处理系统

1965~1980, 引入分时

1980~now, 百花齐放

其他操作系统

Personal-Computer Systems

Parallel Systems

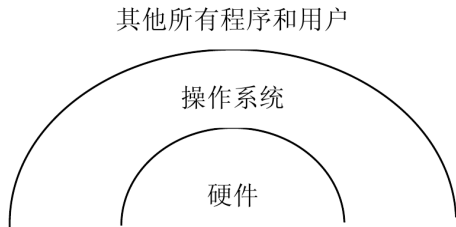
Distributed Systems

Embedded System

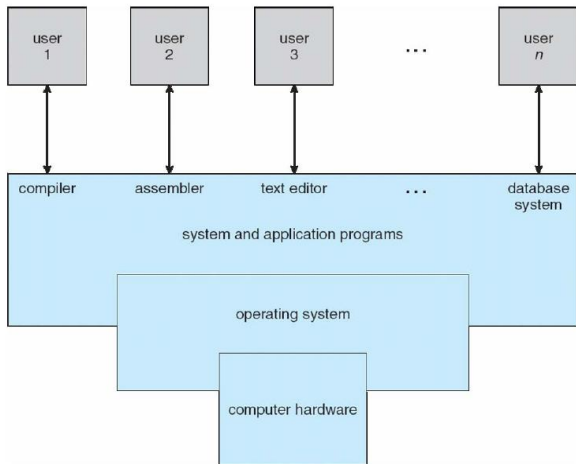
作业、实验和小结

Computer System Organization (1)

- ▶ 计算机系统由硬件和软件两部分组成。
- ▶ 操作系统是硬件之上的第一层软件，可以看成是对计算机硬件系统的第一次扩充。
- ▶ 所有运行在操作系统上层的应用软件都会或多或少、或直接或间接地调用操作系统提供的功能



计算机系统的4个组成成分

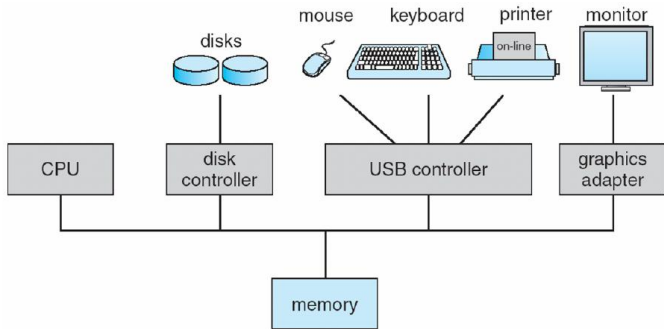


提问：

计算机系统的硬件组成

有哪些？

一个简单示意图：



提问:

你所知道的操作系统

有哪些?

- ▶ Windows系列、Unix系列、SUN Solaris、FreeBSD、苹果、Linux系列，等等
- ▶ 各种实时、非实时的嵌入式操作系统
 - ▶ 例如UC/OS-II、RTEMS、VxWorks、QNX、PalmOS等等
- ▶ 各种网络操作系统、分布式操作系统、集群操作系统、并行操作系统
- ▶ 各种研究型操作系统，等等

提问:

你所知道的操作系统

有哪些?

- ▶ Windows系列、Unix系列、SUN Solaris、FreeBSD、苹果、Linux系列，等等
- ▶ 各种实时、非实时的嵌入式操作系统
 - ▶ 例如UC/OS-II、RTEMS、VxWorks、QNX、PalmOS等等
- ▶ 各种网络操作系统、分布式操作系统、集群操作系统、并行操作系统
- ▶ 各种研究型操作系统，等等

OS definition

- ▶ 没有一个统一的、适用的定义!
- ▶ An Operating System is a **program** that
 - ▶ manages the computer hardware
 - ▶ provides a basis for application programs
 - ▶ acts as an intermediary between the computer **user** and the computer **hardware**
- ▶ OS is a **resource allocator**
 - ▶ Manages all resources
 - ▶ Decides between conflicting requests for efficient and fair resource use
- ▶ OS is a **control program**
 - ▶ Controls execution of programs to prevent errors and improper use of the computer
- ▶ 操作系统是一组控制和管理计算机软硬件资源、合理地各类作业进行调度以及方便用户的程序的集合。

操作系统的（设计）目标

- ▶ 在计算机硬件上配置的OS的（设计）目标有以下几点：
 - ▶ **方便性**
 - ▶ Execute user programs and make solving user problems easier
 - ▶ Make the computer system convenient to use
 - ▶ **有效性**
 - ▶ Use the computer hardware in an efficient manner（提高软硬件资源的利用率）
 - ▶ 可扩充性
 - ▶ 适应软硬件的发展需求
 - ▶ 开放性
 - ▶ 可移植性、互操作性
- ▶ **方便性和有效性是操作系统最重要的两个目标。**

Outline

What is OS?

操作系统的定义和目标

操作系统的作用

操作系统的层次模型

操作系统的发展动力

操作系统的种类

操作系统的发展过程

1945~1955, 无操作系统

1955~1965, 批处理系统

1965~1980, 引入分时

1980~now, 百花齐放

其他操作系统

Personal-Computer Systems

Parallel Systems

Distributed Systems

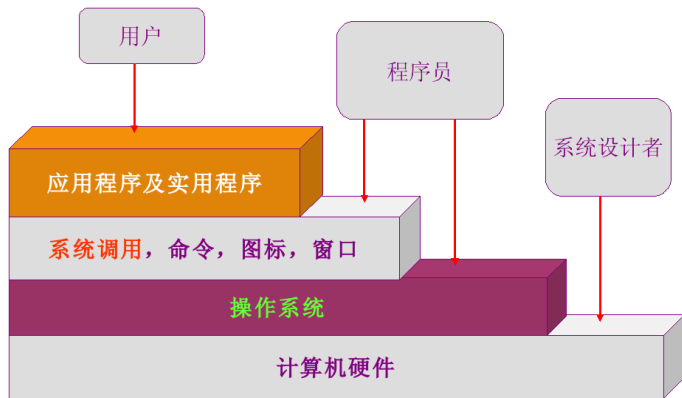
Embedded System

作业、实验和小结

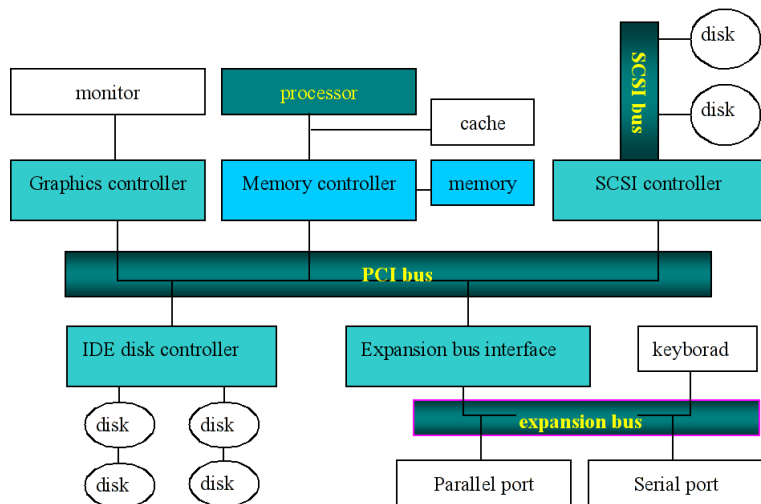
操作系统的作用

- ▶ 对操作系统作用的理解，有**不同的观点**：
 - ▶ **用户与计算机硬件系统之间的接口**（图）
 - ▶ 命令接口、图形用户接口
 - ▶ 编程接口（系统调用接口）
 - ▶ **计算机资源的管理者**（图）
 - ▶ 四类资源：处理机、存储器、I/O设备、文件
 - ▶ **扩充机器（或虚拟机）**
 - ▶ 虚拟机：覆盖了软件的机器
 - ▶ 层次性

作为用户与计算机硬件系统之间的接口



作为计算机资源的管理者



Outline

What is OS?

操作系统的定义和目标

操作系统的作用

操作系统的层次模型

操作系统的发展动力

操作系统的种类

操作系统的发展过程

1945~1955, 无操作系统

1955~1965, 批处理系统

1965~1980, 引入分时

1980~now, 百花齐放

其他操作系统

Personal-Computer Systems

Parallel Systems

Distributed Systems

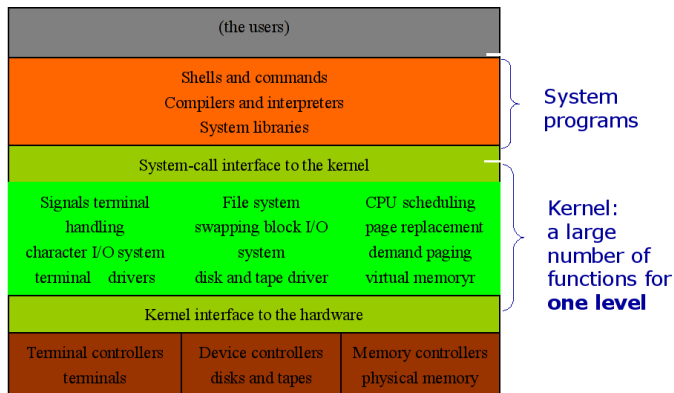
Embedded System

作业、实验和小结

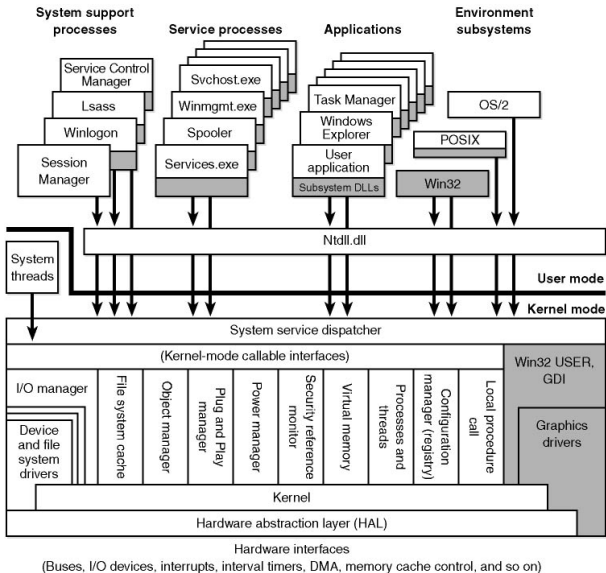
操作系统的层次模型

- ▶ 什么是层次模型？
- ▶ 一种经典的操作系统的结构模型
 - ▶ 最高层：接口
 - ▶ 中间层：对对象进行操纵和管理的软件集合
 - ▶ 最底层：OS操纵和管理的对象，包括各类软硬件资源
- ▶ 以Linux，Windows和嵌入式操作系统RTEMS为例

举例：类UNIX操作系统的经典体系结构图

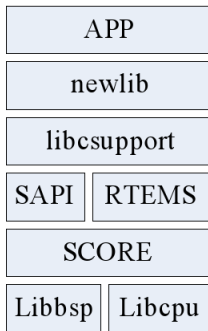


举例：Windows操作系统的经典体系结构图



举例：RTEMS体系结构

- ▶ RTEMS: 一种微内核抢占式实时操作系统
 - ▶ 现在:
Real Time Executive for Multiprocessor Systems;
 - ▶ 最早: Real Time Executive for Missile Systems
 - ▶ 后来: Real Time Executive for Military Systems
- ▶ 4.0.0核心代码约9万行
- ▶ 维护网站: <http://www.rtems.com/>
- ▶ 中文网站: <http://www.rtems.net/>



Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

操作系统的发展动力

- ▶ 操作系统形成至今将近60年，出现了上百种操作系统
 - ▶ 大型机、小型机、微机、嵌入式、实时、分布等等
- ▶ 推动操作系统发展的主要动力有4个方面：
 - ▶ 不断提高计算机资源利用率的需要
 - ▶ 方便用户
 - ▶ 器件的不断更新换代
 - ▶ 计算机体系结构的不断发展
- ▶ 历程：
 - ▶ 无OS时代→批处理系统→分时系统→实时系统→PC→分布式和并行系统→嵌入→移动系统 ...

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

操作系统的种类

- ▶ FMS (FORTRAN Monitor System, FORTRAN监控系统)
- ▶ OS/360 (IBM为系列机360配备的操作系统)
- ▶ CTSS (Compatible Time Sharing System)
- ▶ MULTICS (MULTiplexed Information and Computer Service)
- ▶ UNIX类、Linux
- ▶ CP/M
- ▶ Windows、Macintosh
- ▶ Mach
- ▶ VxWorks、嵌入式Linux系列、uC/OS-II、RTEMS
- ▶

本实验室操作系统研究

- ▶ 基于服务体/执行流模型的操作系统：MiniCore
- ▶ 安全操作系统
- ▶ 分布式操作系统
- ▶ 可重构混成操作系统

- ▶ 车控操作系统
- ▶ 机器狗操作系统
- ▶

操作系统的发展过程

- ▶ 无操作系统时期
- ▶ **单道批处理**系统
- ▶ **多道批处理**系统
- ▶ **分时**系统
- ▶ **实时**系统
- ▶ 其他操作系统
 - ▶ 分布式、并行、安全

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

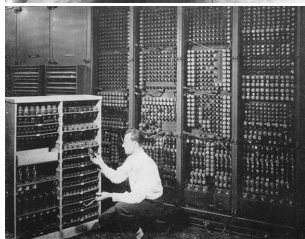
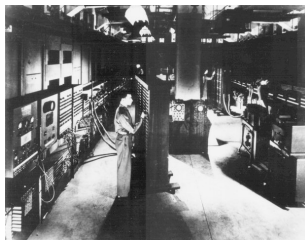
其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

History of the OS(1945~1955, 无操作系统)

- ▶ Tube-based (电子管)
 - ▶ ENIAC
 - ▶ 17,468 vacuum tubes
 - ▶ 5,000 additions/sec,
 - ▶ 1800 square feet, 30 tons
 - ▶ Program are hardwired on plug boards
 - ▶ One program at a time, Need professional operator
 - ▶ User VS. operator
 - ▶ Only useful to Numerical calculations
 - ▶ No OS at all !
 - ▶ Manual system (人工操作)



Replacing a bad tube meant checking among ENIAC's 19,000 possibilities.

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统**
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

History of the OS(1955~1965, 批处理系统)

- ▶ Transistor based (晶体管)
- ▶ **Jobs** on cards or tapes
 - ▶ Job (作业)
 - ▶ Control cards
- ▶ **Language:**
 - ▶ ASM
 - ▶ High level language
 - ▶ FORTRAN, ALGOL, COBOL
- ▶ Applications
 - ▶ Scientific APPs & Engineering APPs
- ▶ **Batch** system

专题1: **Batch system**, 批处理系统

- ▶ 批处理系统概述
- ▶ 专题1.1: 单道批处理系统
- ▶ 专题1.2: 脱机IO和SPOOLING技术
- ▶ 专题1.3: 多道批处理系统

批处理系统的工作方式

- ▶ **用户**（user）将**作业**（job）交给**系统操作员**（operater）
- ▶ 系统操作员将许多用户的作业组成一批作业，输入到计算机系统中，在系统中形成一个自动转接的连续的作业流
 - ▶ 作业是**成批的**
- ▶ 启动操作系统
- ▶ **系统自动、依次执行每个作业**
- ▶ 由操作员将作业结果交给用户

▶ 批处理系统中**作业的组成**:

- ▶ 用户程序
- ▶ 数据
- ▶ 作业说明书（作业控制语言）

▶ **批**的含义:

- ▶ 供一次加载的磁带或磁盘，通常由若干个作业组装而成，在处理中使用一组相同的系统软件

▶ 批作业处理：对批作业中的每个作业进行相同的处理

- ▶ 从磁带读入用户作业和编译链接程序，编译链接用户作业，生成可执行程序；启动执行；执行结果输出

批处理系统

经历了两个阶段

- ▶ Simple batch systems, 单道批处理系统
- ▶ Multiprogramming systems, 多道批处理系统

专题1.1：单道批处理系统

- ▶ 单道批处理系统简介
- ▶ 单道批处理系统的工作过程
- ▶ 单道批处理系统的分析

单道批处理系统

- ▶ 输入设备：
 - ▶ Card readers, tape drives
- ▶ 输出设备：
 - ▶ Line printers, card punches and tape drives
- ▶ Operator **BATCH** similar jobs to speed up processing
 - ▶ User VS. operator
 - ▶ Compare to : Manual system
- ▶ **Monitor** (OS) , load program and execute
 - ▶ Always resident in memory
 - ▶ **FIFO**: Transfer control automatically from one job to the next
- ▶ Only One Job in Memory at a time

单道批处理系统的工作过程

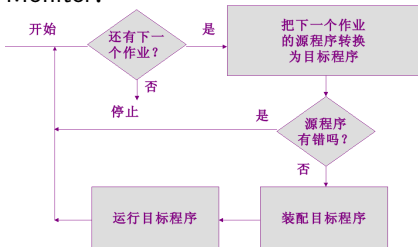
▶ 处理过程

- ▶ 监督程序 (monitor)
- ▶ 系统对作业的处理都是成批进行的、且内存中始终只保持一道作业。
- ▶ 批处理系统的引入是为了**提高系统资源的利用率和吞吐量**

▶ 特征

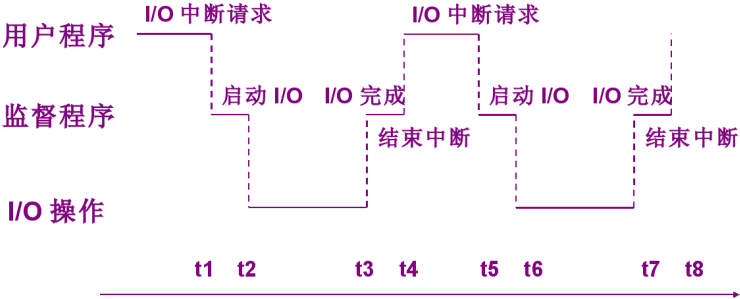
- ▶ **自动性、顺序性、单道性**

Monitor:



单道程序运行情况

用户进程的行为序列：执行→I/O→执行→I/O→...→执行→I/O



单道批处理系统分析

Analysis

- ▶ Serial Card reader: **Jobs execute one by one**
- ▶ Mechanical I/O device: **poor speed**
 - ▶ CPU速度与I/O速度之间的矛盾
 - ▶ CPU: **thousands of instructions/sec** VS. Card reader: **20 cards/sec**
 - ▶ CPU is often **idle**→CPU **utilization is LOW**

解决问题的办法：引入的新技术和成果

- ▶ **Off-line I/O (脱机I/O)**
 - ▶ a cheaper system reads from cards into tapes
- ▶ **磁盘**
 - ▶ Allowed OS to keep all jobs on a disk
 - ▶ With direct access to several jobs
Could do **Job scheduling** to use resources and perform task efficiently
 - ▶ Multiprogramming (多道程序)
CPU utilization(利用率)↑

单道批处理系统分析

Analysis

- ▶ Serial Card reader: **Jobs execute one by one**
- ▶ Mechanical I/O device: **poor speed**
 - ▶ CPU速度与I/O速度之间的矛盾
 - ▶ CPU: **thousands of instructions/sec** VS. Card reader: **20 cards/sec**
 - ▶ CPU is often **idle**→CPU **utilization is LOW**

解决问题的办法：引入的新技术和成果

- ▶ **Off-line I/O (脱机I/O)**
 - ▶ a cheaper system reads from cards into tapes
- ▶ **磁盘**
 - ▶ Allowed OS to keep all jobs on a disk
 - ▶ With direct access to several jobs
Could do **Job scheduling** to use resources and perform task efficiently
 - ▶ Multiprogramming (多道程序)
CPU utilization(利用率)↑

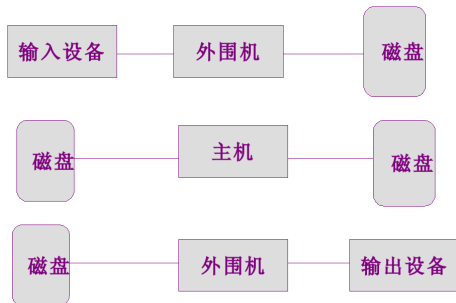
专题1.2：脱机I/O和SPOOLing技术

- ▶ 脱机IO
- ▶ SPOOLing

脱机I/O和SPOOLing技术

脱机I/O（时间：50年代末）

- ▶ 目的：解决人机矛盾和CPU与I/O设备之间速度不匹配的矛盾
- ▶ 方法：利用低速的外围机进行，纸带（卡片）→磁带（磁盘）



- ▶ 脱机的内涵：程序和**数据**都在脱离主机控制下，由**外围机**控制完成。

SPOOLing技术

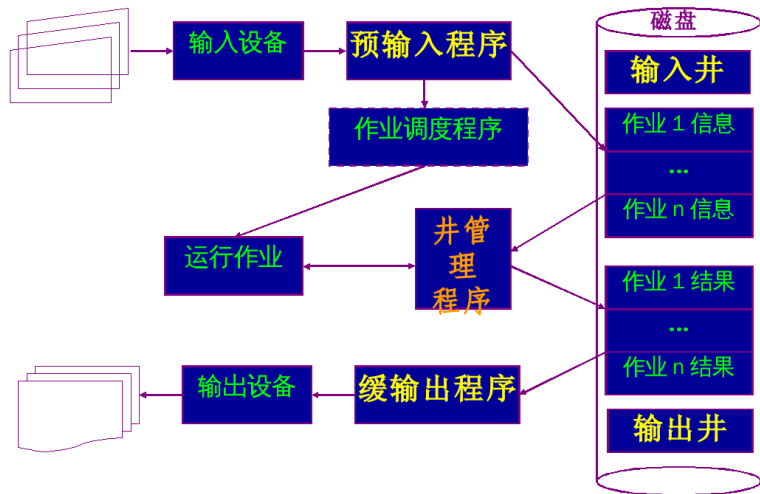
- ▶ 1961年，英国曼彻斯特大学，Atalas机
- ▶ **Simultaneous Peripheral Operation On-Line**
(同时的外围设备联机操作——假脱机技术)
- ▶ **基本思想：**
利用磁盘作缓冲，将输入、计算、输出分别组织成独立的任务流，使I/O和计算真正并行

SPOOLing系统工作原理

- ▶ 作业进入到磁盘上的**输入井**
- ▶ 按某种调度策略选择几种搭配得当的作业，并调入内存
- ▶ 作业运行的结果输出到磁盘上的**输出井**
- ▶ 结果从磁盘上的输出井送到打印机

- ▶ 使用**进程**代替外围机

SPOOLing系统的组成示意图

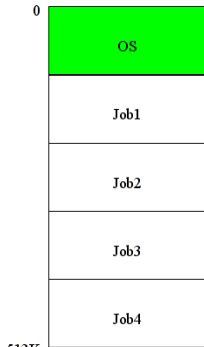


专题1.3：多道批处理系统

- ▶ 多道批处理系统的概念和工作过程
- ▶ 多道程序对操作系统的功能需求
- ▶ 多道批处理系统的分析

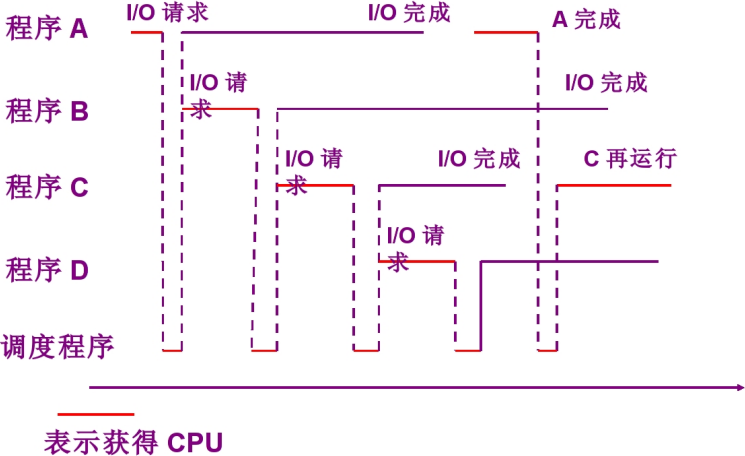
多道批处理系统

- ▶ **多道**：系统中同时驻留多个作业
 - ▶ **共享内存**
 - ▶ **复用CPU**：当一个作业因某个原因暂停运行时，切换到另一个作业上运行
- ▶ 多道引入的优点：
 - ▶ 提高**CPU**利用率
 - ▶ 提高**内存和I/O**设备利用率
 - ▶ 提高了**系统吞吐量**
- ▶ **特征**
 - ▶ **多道性、无序性、调度性**（作业调度、进程调度）



Memory layout for a multiprogramming system

多道程序运行情况（四道时）



多道程序对操作系统的功能需求

- ▶ **Job Scheduling**

- ▶ How many & which jobs entered memory

- ▶ **Memory management**

- ▶ where & how much memory: the system must allocate the memory to several jobs.
- ▶ Memory **protection** for each job

- ▶ **CPU scheduling**

- ▶ Which job in memory(job pool) would get the CPU
- ▶ Job and CPU scheduling makes up **2-phrase of scheduling**

- ▶ **I/O routine** supplied by the system.

- ▶ **Allocation of devices.**

多道批处理系统分析

- ▶ When the job have to wait for some task, such as an I/O op. to complete
 - ▶ Single Batch System: CPU→idle
 - ▶ Multiprogramming system: CPU→switch to another job and execute(CPU is never idle)
- ▶ **Advantages**
 - ▶ Higher CPU, I/O, Memory Utilization
 - ▶ Higher system throughput
- ▶ **Disadvantages**
 - ▶ No User interaction with computer
 - ▶ Job time too long (why?)
 - ▶ Simple batch system VS. Multiprogramming system

多道批处理系统分析

- ▶ When the job have to wait for some task, such as an I/O op. to complete
 - ▶ Single Batch System: CPU→idle
 - ▶ Multiprogramming system: CPU→switch to another job and execute(CPU is never idle)
- ▶ **Advantages**
 - ▶ Higher CPU, I/O, Memory Utilization
 - ▶ Higher system throughput
- ▶ **Disadvantages**
 - ▶ No User interaction with computer
 - ▶ Job time too long (why?)
 - ▶ Simple batch system VS. Multiprogramming system

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时**
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

History of the OS(1965~1980, 引入分时)

- ▶ IC circuits (集成电路)
 - ▶ LSI,VLSI
- ▶ UNIX
- ▶ More Applications
- ▶ OS
 - ▶ Multiprogramming batch systems↑
 - ▶ Time-sharing systems (分时系统)

专题2：分时系统

- ▶ 分时系统的需求及其解决
- ▶ 经典案例
- ▶ 分时系统下的工作方式
- ▶ 关键技术、设计目标和实现

Time-sharing system 分时系统

- ▶ 需求: User need interaction with computer
 - ▶ Response time < 1 sec
- ▶ 解决方法:
 - ▶ Share CPU by time pieces (时间片)
 - ▶ Time-sharing (multitasking)
- ▶ Users share Main frame
 - ▶ One main frame VS. Multi users & Multi terminal
- ▶ Time-sharing system is a logical extension of multiprogramming.
- ▶ 经典操作系统: MULTICS、UNIX

经典案例：UNIX

- ▶ 一群计算机迷在贝尔实验室开发出UNIX
- ▶ 初衷：可以在一台无人使用的DEC PDP-7 小型计算机上玩星际探险游戏
- ▶ **Ken Thompson, Dennis Ritchie** 1983年图灵奖获得者 1999年4月美国国家技术金奖

分时系统下的工作方式

- ▶ **一台主机连接了若干个终端，**
每个终端有一个用户在使用
 - ▶ 交互式的向系统提出命令请求
 - ▶ 系统接受每个用户的命令
 - ▶ 采用时间片轮转方式处理服务请求
 - ▶ 通过交互方式在终端上向用户显示结果
 - ▶ 用户根据上步结果发出下道命令

分时系统的关键技术

- ▶ Receive input in time (及时接收)
- ▶ Process in time (及时处理)
- ▶ 解决思路:
 - ▶ Mutual job (交互作业) always in memory
 - ▶ Time pieces
- ▶ 分时系统的**特征**
 - ▶ **多路性、独立性、及时性、交互性**

分时系统的设计目标

- ▶ 分时操作系统所追求的设计目标：
 - ▶ **及时响应**，其依据是响应时间
- ▶ 响应时间：
 - ▶ **从终端发出命令到系统给予回答所经历的时间**
- ▶ **影响响应时间的因素：**
 - ▶ 机器处理能力
 - ▶ 请求服务的时间长短
 - ▶ 系统中连接的终端数目
 - ▶ 服务请求的分布
 - ▶ 调度算法（时间片的选取）

分时系统的实现

- ▶ 单道与分时的结合：
 - ▶ 单道分时
- ▶ 分时与批处理相结合：
 - ▶ 原则：分时优先，批处理在后
 - ▶ 具有前后台的分时：
 - ▶ “前台”：需频繁交互的作业
 - ▶ “后台”：时间性要求不强的作业
- ▶ 分时与多道相结合
 - ▶ 多道分时

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

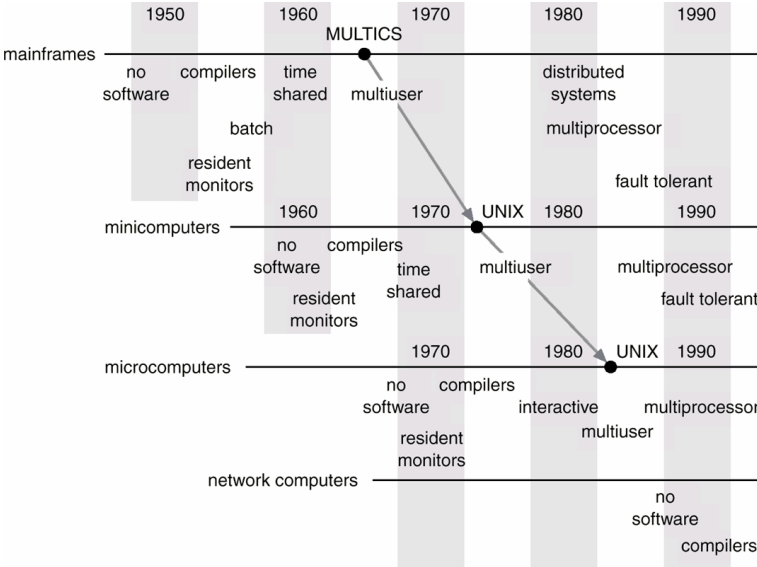
- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

History of the OS(1980~now, 百花齐放)

- ▶ Development of Computer architecture (图)
 - ▶ 32bits→64bits
 - ▶ Workstations and PCs
 - ▶ Parallel processors
 - ▶ Computer networks
 - ▶ Cluster
- ▶ Special purpose computer system Types
 - ▶ Parallel systems
 - ▶ Real-time systems
 - ▶ Embedded systems
 - ▶ Distributed systems

Migration of OS Concepts and Features



专题3：实时系统

- ▶ 实时系统
- ▶ 实时任务的分类
- ▶ 实时系统的实现和应用
- ▶ 实时系统与批处理系统、分时系统的比较

实时系统

- ▶ 通常在一些专门的应用中，用来控制设备
 - ▶ 这种系统往往具有及时响应的时间限制
 - ▶ 严格 vs. 不严格
- ▶ 定义：

实时系统是指使计算机能及时响应外部事件的请求，在规定的严格时间内完成对该事件的处理，并控制所有实时设备和实时任务协调一致地工作的操作系统
- ▶ 按领域分类：
 - ▶ 第一类：实时过程控制
 - ▶ 第二类：实时通信（信息）处理

实时任务的分类

- ▶ 按任务执行是否呈现周期性来划分
 - ▶ 周期性的，有规律；
 - ▶ 非周期性的，无规律，但有截止时间
 - ▶ 开始截止时间 vs. 完成截止时间
- ▶ 根据对截止时间的要求来划分
 - ▶ **硬实时**任务 vs. **软实时**任务
- ▶ 实时操作系统追求的设计目标：
 - ▶ 满足实时性要求：
对外部请求在严格时间范围内作出反应
 - ▶ 高可靠性

实时系统的实现和应用

- ▶ 硬实时系统: Hard real-time system
 - ▶ Secondary storage limited or absent, data stored in short-term memory, or read-only memory (ROM)
 - ▶ Conflicts with time-sharing systems, not supported by general-purpose operating systems
- ▶ 软实时系统: Soft real-time system
 - ▶ Limited utility in industrial control or robotics
 - ▶ Useful in applications (multimedia, virtual reality) requiring advanced operating-system features
- ▶ 典型: VxWorks、QNX、RTEMS

实时系统与批处理系统和分时系统的区别

- ▶ **专用系统：**许多实时系统是专用系统，而批处理与分时系统通常是通用系统
- ▶ **实时控制：**实时系统用于控制实时过程，要求对外部事件的迅速响应，具有较强的中断处理机构
- ▶ **高可靠性：**实时系统用于控制重要过程，要求高度可靠，具有较高冗余（如双机系统）
- ▶ **事件驱动和队列驱动：**实时系统的工作方式：接受外部消息，分析消息，调用相应处理程序进行处理。

实时、分时的比较

- ▶ 多路性：相同
- ▶ 独立性：相同

- ▶ 及时性：实时系统要求更高
- ▶ 交互性：分时系统交互性更强
- ▶ 可靠性：实时系统要求更高

专题小结

- ▶ 到目前为止，介绍了三种最基本的操作系统类型
 - ▶ 批处理系统
 - ▶ 分时系统
 - ▶ 实时系统
- ▶ 一个实际的操作系统，往往兼有上述三种基本操作系统类型的功能
- ▶ 下面简单介绍其他类型的操作系统

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems**
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

Personal-Computer Systems, 个人计算机系统

- ▶ Personal computers (PCs)
 - ▶ 计算机为单用户服务
- ▶ I/O devices 键盘、鼠标、显示器、打印机
- ▶ PC系统所追求的设计目标是：
 - ▶ 界面友好，使用方便 (User convenience & responsiveness)，有丰富的应用软件
 - ▶ 不必过于追求CPU利用率

常见的PC system用的操作系统

- ▶ OS

- ▶ MS-DOS

- ▶ OS/2

- ▶ Microsoft windows ... NT, 95, 98, 2000, xp, windows me, windows vista

- ▶ Apple Macintosh

- ▶ Linux (...)

- ▶ ...

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems**
- Distributed Systems
- Embedded System

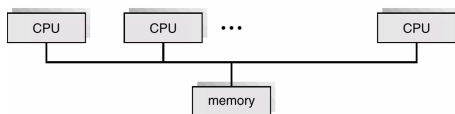
作业、实验和小结

并行系统 Parallel Systems

- ▶ Multiprocessor systems with more than one CPU in close communication
- ▶ Tightly coupled system 紧耦合系统
 - ▶ processors share memory and a clock; communication usually takes place through the shared memory
- ▶ 优点:
 - ▶ Increased throughput
 - ▶ Economical
 - ▶ Increased reliability
 - ▶ graceful degradation
 - ▶ fail-soft systems

并行系统(Cont.)

- ▶ Symmetric multiprocessing (SMP, 对称多处理器)
 - ▶ Each processor runs an identical copy of the operating system.
 - ▶ Many processes can run at once without performance deterioration.



- ▶ 现在的大多数通用操作系统都支持SMP，例如Linux、UNIX、Windows

并行系统(Cont.)

- ▶ Asymmetric multiprocessing 非对称多处理 ASMP
 - ▶ Each processor is assigned a specific task; master processor schedules and allocates work to slave processors.
 - ▶ More common in extremely large systems

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems**
- Embedded System

作业、实验和小结

Distributed Systems 分布式系统

- ▶ 分布式系统：处理和控制的分散
- ▶ Loosely coupled system 松耦合系统
 - ▶ each processor has its own local memory; processors communicate with one another through various communications lines, such as high-speed buses or telephone lines
- ▶ Advantages of distributed systems
 - ▶ Resources Sharing 资源共享
 - ▶ Computation speed up – load sharing 负载均衡
 - ▶ Reliability 可靠
 - ▶ Communications 通信

分布式系统上的操作系统

- ▶ Network Operating System 网络操作系统
 - ▶ provides file sharing
 - ▶ provides communication scheme
 - ▶ runs independently from other computers on the network
- ▶ Distributed Operating System 分布式操作系统
 - ▶ less autonomy between computers
 - ▶ gives the impression there is a single operating system controlling the network 单一映像

Outline

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System**

作业、实验和小结

Embedded System

- ▶ 嵌入式系统是在各种设备、装置或系统中，完成特定功能的软硬件系统
 - ▶ 它们是一个大设备、装置或系统中的一部分，这个大设备、装置或系统可以不是“计算机”
 - ▶ 通常工作在反应式或对处理时间有较严格要求环境中
- ▶ 由于它们被嵌入在各种设备、装置或系统中，因此称为嵌入式系统
- ▶ 嵌入式系统具有最广泛的应用

- ▶ 嵌入式操作系统与通用操作系统有很大不同
 - ▶ Small size、Low power
 - ▶ Special environment, special function
 - ▶ 开发方式也不同
 - ▶ 交叉开发
 - ▶ Host, simulator VS. target
- ▶ 经典：VxWorks、嵌入式Linux系列、RTEMS、WindowsCE、PalmOS

作业一

▶ 作业:

- ▶ 阅读至少2本操作系统相关书籍，给出这些书中关于操作系统的定义，要列出出处。
- ▶ 阅读至少2本操作系统相关书籍，阐明操作系统的公共设计目标和某些操作系统特有的设计目标，要列出出处。
- ▶ 阅读至少2本操作系统相关书籍，阐明操作系统的作用，要列出出处。
- ▶ 操作系统的基本类型是哪三种？他们的关键技术有哪些？

实验一

▶ 安装虚拟机

- ▶ 在虚拟机(VirtualBox)上安装一款Linux发行版操作系统(推荐KUbuntu, 或按助教指定)
- ▶ 熟悉Linux下的软件安装方法
- ▶ 安装一个可用的集成开发环境(推荐: eclipse CDT)和GCC编译工具链(大多自带)
- ▶ 学会编写简单的shell脚本
- ▶ 注意: 为防止数据因重装系统丢失, 建议将数据与系统分开管理

▶ 1人1组

- ▶ 检查方法和要求: 提交详细的安装说明书, 要求列出遇到的困难和解决的方法
- ▶ 提交报告的时间: 以邮件发送时间为准
- ▶ 截止时间: 待定

小结

What is OS?

- 操作系统的定义和目标
- 操作系统的作用
- 操作系统的层次模型
- 操作系统的发展动力
- 操作系统的种类

操作系统的发展过程

- 1945~1955, 无操作系统
- 1955~1965, 批处理系统
- 1965~1980, 引入分时
- 1980~now, 百花齐放

其他操作系统

- Personal-Computer Systems
- Parallel Systems
- Distributed Systems
- Embedded System

作业、实验和小结

谢谢!