

信息与电子学科百本精品教材工程

新编计算机类本科规划教材

操作系统原理 实用教程

任满杰 刘树刚 李军红 等编著

<http://www.phei.com.cn>



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

新编计算机类本科规划教材

操作系统原理实用教程

任满杰 刘树刚 李军红 柯敏毅 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书详细介绍了计算机操作系统的基本概念、基本原理和典型实现技术,理论学习和实践应用相结合。全书共6章,分别介绍了操作系统的基本概念、功能和特征,进程的概念及进程管理的各种策略,存储管理方式及实现方法,文件的基本概念及实现过程,设备管理技术及驱动程序设计,网络操作系统的概念、网络通信技术及资源共享技术。每章都以目前流行的Linux为例,介绍具体的实现方案。

本书可作为高等学校计算机科学与技术及相关专业本科生教材,也可作为从事信息科学和计算机工作的科技人员学习操作系统的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

操作系统原理实用教程 / 任满杰, 刘树刚, 李军红等编著. —北京: 电子工业出版社, 2006. 1

新编计算机类本科规划教材

ISBN 7-121-02265-6

I. 操… II. ①任… ②刘… ③李… III. 操作系统—高等学校—教材 IV. TP316

中国版本图书馆CIP数据核字(2006)第007471号

责任编辑: 王羽佳

印 刷: 北京牛山世兴印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路173信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 17 字数: 436千字

印 次: 2006年1月第1次印刷

印 数: 5000册 定价: 21.60元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话:(010) 68279077。质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

信息与电子学科百本精品教材工程

《新编计算机类本科规划教材》编委会

主任委员：刘乃琦（电子科技大学）

副主任委员：徐建民（河北大学）

周 娅（桂林电子工业学院）

王 越（重庆工学院）

胡先福（电子工业出版社）

委 员：陈建铎 陈联诚 陈玉明 丁新民 焦占亚 雷景生 骆耀祖

马立权 任满杰 佟伟光 王 力 肖建华 杨 威 叶核亚

朱玉玺 朱战立 张孟玮 冉 哲 李 岩

编辑出版组

主 任：胡先福

成 员：张孟玮 冉 哲 李 岩 王 颖

凌 毅 韩同平 李维荣 张 昱

《新编计算机类本科规划教材》参编单位

(按拼音顺序排列)

- ▶ 北方工业大学
- ▶ 北京联合大学
- ▶ 长春大学
- ▶ 长春税务学院
- ▶ 重庆工学院
- ▶ 大理学院
- ▶ 大连海事大学
- ▶ 大连民族学院
- ▶ 大连轻工业学院
- ▶ 电子科技大学
- ▶ 佛山大学
- ▶ 广东嘉应学院
- ▶ 广东韶关学院
- ▶ 广西大学
- ▶ 广西工学院
- ▶ 桂林电子工业学院
- ▶ 贵州工业大学
- ▶ 贵州师范大学
- ▶ 哈尔滨工程大学
- ▶ 海南大学
- ▶ 杭州电子工业学院
- ▶ 河北大学
- ▶ 合肥学院
- ▶ 湖北工业大学
- ▶ 湖南工程学院
- ▶ 华北航天工业学院
- ▶ 华南农业大学
- ▶ 江汉大学
- ▶ 金陵科技学院
- ▶ 南京工程学院
- ▶ 南京师范大学
- ▶ 齐齐哈尔大学
- ▶ 青岛科技大学
- ▶ 山西农业大学
- ▶ 山西师范大学
- ▶ 陕西科技大学
- ▶ 上海第二工业大学
- ▶ 上海应用技术学院
- ▶ 邵阳学院
- ▶ 沈阳工程学院
- ▶ 首都经济贸易大学
- ▶ 太原理工大学阳泉学院
- ▶ 武汉工业学院
- ▶ 武汉科技大学
- ▶ 五邑大学
- ▶ 西安石油大学
- ▶ 西安邮电学院
- ▶ 孝感学院
- ▶ 烟台大学

前 言

操作系统是计算机软件系统的核心，是所有计算机系统的基础和支撑，它管理和控制着计算机系统软硬件资源，可以说操作系统是计算机系统的灵魂。由于操作系统原理过于抽象，要真正理解操作系统的概念，必须将原理与实践相结合。本教材将操作系统原理、概念和实例融为一体，使学生通过学习这门课程，对操作系统有一个明确清晰的认识。

本书作者根据教学经验，参考国内外大量最新教材和相关资料，注重基础性、系统性、实用性和新颖性，结合实际操作系统，深入浅出地阐述了操作系统的概念、原理与实现技术。高等学校教材的应用性和实践性尤为重要，本书本着有利于培养学生获取知识的能力、运用知识的能力和科学创新能力的原则安排教材的内容，不仅有操作系统原理，更有操作系统实现，把理论学习和实践应用融为一体，使学生在理解操作系统原理的基础上，能够进行操作系统的实验及设计。每章都以目前流行的 Linux 为例，介绍具体的实现方案，既体现了应用性又将新技术和新知识融入到各章之中。

本书由 6 章组成，建议课堂讲授 60 学时，实验 20 学时。

第 1 章 操作系统概述。主要阐述操作系统的定义、发展历史、分类、功能、特性及目前常用的操作系统。重点是操作系统的基本概念、功能和特征。建议课堂讲授 4 学时。

第 2 章 进程管理。主要介绍程序的并发执行、进程、线程的概念，进程的状态及其转换，进程同步与互斥的概念及进程同步与互斥的各种实现策略，进程通信，进程调度与死锁，以及 Linux 中进程的实现等内容。重点是进程和同步的概念、同步机制、通信方式、调度算法、死锁的概念及解决方法、线程的概念、线程与进程的关系；难点是同步问题的实现。建议课堂讲授 16 学时，实验 6 学时。

第 3 章 存储管理。主要介绍存储管理的概念，存储管理的目的，存储管理的四大基本功能——内存分配与回收、逻辑地址到物理地址的转换、存储保护和内存扩充，实存管理、虚存管理的各种策略，以及 Linux 中的存储管理。重点是各种存储管理方式及实现方法，难点是虚拟存储概念及实现。建议课堂讲授 14 学时，实验 4 学时。

第 4 章 文件管理。主要介绍文件系统的概念、文件的组织结构及存取方法、文件存储空间的管理、文件的安全与保护，以及 Linux 中的文件系统。重点是文件的基本概念及实现过程，难点是目录管理和文件空间管理。建议课堂讲授 12 学时，实验 6 学时。

第 5 章 设备管理。主要介绍设备管理的任务与功能、设备管理的硬件组织与软件组织、缓冲技术、驱动程序设计技术，以及 Linux 中的设备管理。重点是缓冲技术和设备处理过程，难点是驱动程序设计。建议课堂讲授 8 学时，实验 4 学时。

第 6 章 网络操作系统。主要介绍网络操作系统的概念、功能及实现，网络通信技术——原语通信与远程过程调用，网络资源共享的概念，网络文件系统的概念与实现，以及 Linux 中网络文件系统的体系结构。重点是网络操作系统的概念、网络通信技术及资源共享技术。建议课堂讲授 6 学时。

本书每章的开始有教学要点，最后有小结，并配有大量习题。

本书第1章和第2章由任满杰编写，第3章由柯敏毅、任满杰编写，第4章由刘树刚、任满杰编写，第5章和第6章由李军红编写，附录由翟一鸣编写，韩俊芳老师参与了第5章和第6章的部分编写工作，全书由任满杰统稿。

山东大学石冰教授审阅了全书，并提出了宝贵意见；潘卫平同学在本书的资料整理、编辑修订过程中做了大量工作；教材编写委员会和电子工业出版社在本书的出版过程中给予了大力支持。在此一并表示衷心的感谢！

由于作者水平有限，书中疏漏与错误在所难免，敬请读者批评指正。

联系方式：rmj@ytu.edu.cn。

作 者
2005年8月

目 录

第 1 章 操作系统概述	(1)
1.1 操作系统的概念	(1)
1.1.1 操作系统的地位	(1)
1.1.2 资源管理功能	(2)
1.1.3 接口功能	(2)
1.2 操作系统的发展历史	(3)
1.2.1 手工操作阶段	(3)
1.2.2 联机批处理系统	(3)
1.2.3 脱机批处理系统	(4)
1.2.4 执行系统	(4)
1.2.5 多道批处理系统	(5)
1.2.6 分时系统	(7)
1.2.7 实时系统	(8)
1.2.8 网络操作系统	(9)
1.2.9 分布式操作系统	(9)
1.2.10 PC 操作系统	(10)
1.3 操作系统分类	(10)
1.4 操作系统的功能和主要特征	(12)
1.4.1 操作系统的功能	(12)
1.4.2 操作系统的主要特征	(15)
1.5 常用操作系统介绍	(16)
1.5.1 MS-DOS 及 Windows 系列	(16)
1.5.2 UNIX 家族	(19)
1.5.3 自由软件: Linux 及其他	(22)
1.6 小结	(25)
习题 1	(26)
第 2 章 进程管理	(27)
2.1 进程的概念	(27)
2.1.1 程序的并发执行	(27)
2.1.2 进程的定义	(31)
2.1.3 进程的状态及其转换	(32)
2.1.4 进程控制块	(34)
2.1.5 进程的特性	(35)
2.2 进程控制	(36)

2.3	进程互斥与同步	(38)
2.3.1	临界资源与临界区	(38)
2.3.2	互斥工具	(39)
2.3.3	同步机制	(43)
2.3.4	经典同步问题	(47)
2.4	进程通信	(55)
2.4.1	共享存储	(55)
2.4.2	消息传递	(56)
2.4.3	共享文件	(58)
2.5	进程调度	(59)
2.5.1	进程调度的基本概念	(59)
2.5.2	进程调度算法	(60)
2.6	死锁	(63)
2.6.1	死锁的概念	(63)
2.6.2	死锁的预防	(65)
2.6.3	死锁的避免	(66)
2.6.4	死锁的检测	(67)
2.6.5	死锁的解除	(70)
2.7	线程的概念及其实现	(70)
2.7.1	线程的引入	(71)
2.7.2	进程与线程的关系	(71)
2.7.3	线程的状态及转换	(72)
2.7.4	线程调度	(73)
2.7.5	线程通信	(73)
2.7.6	多线程应用举例	(75)
2.8	Linux 中的进程	(76)
2.8.1	Linux 进程控制块 PCB	(76)
2.8.2	进程的创建	(82)
2.8.3	进程调度	(83)
2.8.4	进程的退出与消亡	(85)
2.8.5	相关的系统调用	(86)
2.8.6	信号	(87)
2.8.7	信号量与 P/V 操作	(89)
2.8.8	等待队列	(90)
2.8.9	管道	(90)
2.8.10	Linux 内核体系结构	(92)
2.8.11	并发程序设计实例	(92)
2.9	小结	(94)
	习题 2	(95)

第 3 章 存储管理	(97)
3.1 概述	(98)
3.1.1 存储管理的功能	(98)
3.1.2 内存的分配与回收	(98)
3.1.3 地址重定位	(98)
3.1.4 存储保护	(100)
3.1.5 虚拟存储器	(101)
3.2 连续存储管理	(102)
3.2.1 固定分区存储管理	(102)
3.2.2 可变式分区存储管理	(103)
3.3 分页式存储管理	(108)
3.3.1 分页式存储管理中存储块的分配与回收	(109)
3.3.2 分页式存储管理的地址重定位	(110)
3.3.3 联想存储器	(111)
3.3.4 存储保护	(112)
3.4 分段式存储管理	(112)
3.4.1 分段式存储管理的基本思想	(112)
3.4.2 分段式存储管理的地址重定位	(113)
3.4.3 分段与分页的区别	(114)
3.5 段页式存储管理	(114)
3.6 虚拟存储器管理	(116)
3.6.1 虚拟存储器的概念	(116)
3.6.2 请求页式存储管理	(116)
3.6.3 页面置换算法	(118)
3.6.4 请求段式存储管理	(121)
3.7 各种存储分配策略的比较	(123)
3.8 Linux 存储管理	(125)
3.8.1 Linux 存储管理概述	(125)
3.8.2 内存分配	(126)
3.8.3 请求调页	(128)
3.8.4 页面交换	(129)
3.9 小结	(130)
习题 3	(131)
第 4 章 文件管理	(132)
4.1 文件系统概述	(132)
4.1.1 文件及文件系统	(132)
4.1.2 文件类型	(134)
4.1.3 文件的操作	(135)
4.2 文件的组织结构及存取方法	(136)
4.2.1 文件的逻辑结构	(136)

4.2.2	文件的物理结构	(136)
4.2.3	文件的存取方法	(138)
4.3	文件存储空间管理	(139)
4.3.1	磁盘空闲空间分配策略	(140)
4.3.2	空闲空间管理	(142)
4.4	文件目录管理	(145)
4.4.1	文件目录概念	(145)
4.4.2	文件目录结构	(146)
4.4.3	文件目录操作	(149)
4.5	文件的安全与保护	(150)
4.5.1	用户认证	(151)
4.5.2	保护域	(152)
4.5.3	存取控制	(153)
4.5.4	存取权限	(154)
4.5.5	保护模型	(154)
4.5.6	隐藏通道	(156)
4.6	文件系统性能改善	(157)
4.7	Linux 中的文件系统	(159)
4.7.1	虚拟文件系统 VFS	(159)
4.7.2	Linux 虚拟文件系统的数据结构	(161)
4.7.3	对虚拟文件系统的管理	(162)
4.7.4	EXT2 文件系统	(163)
4.7.5	EXT2 磁盘重要数据结构	(165)
4.7.6	EXT3 文件系统	(173)
4.8	小结	(175)
	习题 4	(176)
第 5 章	设备管理	(177)
5.1	设备管理概述	(177)
5.2	输入/输出硬件组织	(179)
5.2.1	输入/输出设备的分类	(179)
5.2.2	设备控制器	(180)
5.2.3	通道	(182)
5.3	输入/输出软件组织	(183)
5.3.1	输入/输出软件的层次结构	(184)
5.3.2	中断处理程序	(184)
5.3.3	设备驱动程序	(186)
5.3.4	与设备无关的 I/O 软件	(187)
5.3.5	用户层的 I/O 软件	(189)
5.4	缓冲技术	(190)
5.4.1	缓冲技术的引入	(190)

5.4.2	单缓冲	(190)
5.4.3	双缓冲	(191)
5.4.4	缓冲池	(192)
5.5	设备的分配与回收	(193)
5.5.1	设备分配所用的数据结构	(193)
5.5.2	设备分配与回收	(197)
5.6	Linux 设备管理	(199)
5.6.1	Linux 设备管理基础	(200)
5.6.2	Linux 设备驱动程序	(203)
5.6.3	Linux 的中断处理	(205)
5.7	设备驱动程序的设计	(206)
5.7.1	设备驱动程序的统一模型与其性能的一般指南	(206)
5.7.2	Linux 下 PCI 设备驱动程序的设计	(208)
5.8	小结	(215)
习题 5		(216)
第 6 章	网络操作系统	(217)
6.1	网络操作系统概述	(217)
6.1.1	网络拓扑结构	(217)
6.1.2	通信与协议	(221)
6.1.3	网络操作系统的功能和构成	(226)
6.1.4	Linux 网络操作系统的实现	(228)
6.2	网络通信技术	(230)
6.2.1	原语通信	(230)
6.2.2	远程过程调用	(231)
6.3	网络资源共享技术	(233)
6.3.1	网络资源共享的概念	(233)
6.3.2	硬盘共享	(233)
6.3.3	打印机共享	(235)
6.3.4	数据和文件资源的共享	(236)
6.3.5	在混合网络中访问网络资源	(237)
6.4	网络文件系统	(238)
6.4.1	网络文件系统的概述	(238)
6.4.2	网络文件系统的实现	(239)
6.4.3	Linux 网络文件系统的体系结构	(240)
6.4.4	Linux 网络文件系统的安装	(242)
6.5	小结	(243)
习题 6		(244)
附录 A	缩略词	(245)
附录 B	Linux 系统常用系统调用	(250)
参考文献		(257)

第 1 章 操作系统概述



教学要点

本章主要介绍操作系统的基本概念、发展历史、功能和主要特征，以及常用的操作系统。应重点掌握操作系统的基本概念和功能，并对操作系统的发展历史有一定的了解，进而对目前流行的操作系统及其发展方向能有深刻的认识。

在现代计算机系统中，如果不安装操作系统，很难想象还会有谁使用它。操作系统是计算机系统中重要的系统软件，是整个计算机系统的控制中心。操作系统不仅将裸机改造成为功能强、服务质量高、使用方便灵活、运行安全可靠的虚拟机来为用户提供良好的使用环境，而且采用合理有效的方法来组织多个用户共享计算机系统中的各种资源，最大限度地提高系统资源的利用率。

1.1 操作系统的概念

操作系统是配置在计算机硬件平台上的第一层软件，是一组系统软件。一个新的操作系统往往融合了计算机发展中的一些传统的技术和新的研究成果。在计算机系统中，处理器、内存、磁盘、终端、网卡等硬件资源通过主板连接构成了看得见摸得着的计算机硬件系统。为了使这些硬件资源高效地、尽可能并行地供用户程序使用，为了给用户提供使用这些硬件的通用方法，必须为计算机配备操作系统。操作系统的工作就是管理计算机的硬件资源和软件资源，并组织用户尽可能方便地使用这些资源。操作系统是软硬件资源的控制中心，它以尽量合理有效的方法组织用户共享计算机的各种资源。为了使读者对操作系统有更清楚的了解，下面从系统软件、资源管理器和接口三个方面介绍操作系统。

1.1.1 操作系统的地位

计算机系统可以看作由硬件和软件按层次结构组成的系统，如图 1.1 所示。硬件系统是指构成计算机系统所必需的硬件设备。现代计算机系统一般都包含处理器、内存、磁盘驱动器、光盘驱动器、打印机、时钟、鼠标、键盘、显示器、网络接口等硬件。计算机硬件系统构成了计算机本身和用户作业的物质基础。

只有硬件系统而没有软件系统的计算机称为裸机。用户直接使用裸机不仅不方便，而且系统效率将严重降低。软件系统是为计算机系统配置的程序和数据的集合。软件系统又有应

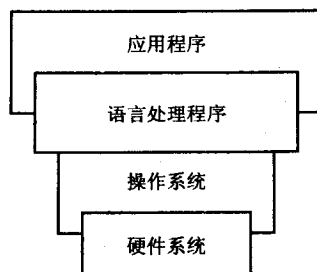


图 1.1 计算机系统组成

用软件和系统软件之分。应用软件是为解决某一具体应用问题而开发的软件，如财务软件、字处理软件等；系统软件是专门为计算机系统所配置的，如操作系统、各种语言处理程序等。操作系统是以硬件为基础的系统软件，是硬件层的第一次扩充，在这一层上实现了操作系统的全部功能，并提供了相应的接口。其他各软件层都是在操作系统的基础上开发出来的。语言处理程序层包括各种程序设计语言的编译程序及动态调试程序等实用性程序。语言处理程序层是操作系统层的扩充，而应用程序层是语言处理程序层的进一步扩充。在应用程序层，用户可以使用各种程序设计语言，在操作系统的支持下，编写并运行满足用户需要的各种应用程序。

由此可见，操作系统是计算机系统中最重要系统软件。

1.1.2 资源管理功能

从资源管理的角度，可以把计算机系统资源分为四大类：处理器、存储器、输入/输出设备和信息。前三类为硬件资源，信息为软件资源。操作系统的任务就是使整个计算机系统的资源得到充分有效地利用，并在相互竞争的程序之间合理有序地控制系统资源的分配，从而实现了对计算机系统工作流程的控制。

作为资源管理器，操作系统要完成以下工作。

(1) 跟踪资源状态

时刻维护系统资源的全局信息，掌握系统资源的种类和数量，以及已分配和未分配的情况。

(2) 分配资源

处理对资源的使用请求，协调请求中的冲突，确定资源分配算法。当有多个用户争用某个资源时，进行裁决。同时，根据资源分配的条件、原则和环境决定是立即分配还是暂缓分配。

(3) 回收资源

用户程序在资源使用完毕之后要释放资源。此时，资源管理器应及时回收资源，以便下次重新分配。

(4) 保护资源

资源管理器负责对资源进行保护，防止资源被有意或无意地破坏。

系统资源的使用方法和策略决定了操作系统的规模、类型、功能与实现方法。基于这一点，可以把操作系统看成是由一组资源管理器（即资源管理程序）组成的。根据资源的分类情况，可以为操作系统建立相应的四类管理器：处理器管理、存储器管理、输入/输出设备管理和信息管理（通常指文件系统）。

1.1.3 接口功能

在计算机系统的四个层次中，硬件是最低层。操作系统处于用户与计算机系统硬件之间，用户通过操作系统来使用计算机。对多数计算机而言，在机器语言级的体系结构（包括指令系统、存储组织、I/O和总线结构）上编程是相当困难的，尤其是输入/输出操作。为了让用户和程序员在使用计算机时不涉及硬件细节，使程序员与硬件细节独立开来，需要建立一种高度抽象。这种抽象就是为用户提供一台等价的扩展计算机，这样的计算机称为虚拟计算机，简称虚拟机。操作系统作为虚拟机，为用户使用计算机提供了方便，用户可不必了解计算机硬件的工

作细节，只需要通过操作系统来使用计算机，操作系统就成了用户与计算机之间的接口。

用户可通过三种方式使用计算机。

(1) 命令方式

用户可通过键盘输入由操作系统提供的一组命令，来直接操纵计算机系统。

(2) 系统调用方式

用户可在自己的应用程序中，通过调用操作系统提供的一组系统调用，来操纵计算机系统。

(3) 图形、窗口方式

用户通过屏幕上的窗口和图标，来操纵计算机系统和运行自己的程序。

有了这几种方式，用户就可以不涉及硬件的实现细节，方便而有效地取得操作系统为用户所提供的各种服务，合理地组织计算机工作流程。所以说，操作系统是用户与计算机硬件系统之间的接口。

综上所述，操作系统是一组程序 and 数据的集合，它能够控制和管理计算机系统的硬件和软件资源，合理地组织计算机工作流程，并为用户使用计算机提供方便。

设置操作系统的目的在于提高计算机系统的效率，增强系统的处理能力，提高系统资源的利用率，方便用户使用计算机。

1.2 操作系统的发展历史

1.2.1 手工操作阶段

在计算机刚刚出现时，由于存储容量小，运算速度慢，输入/输出设备只有纸带输入机、卡片阅读机、打印机和控制台。人们使用这样的计算机只能采用人工操作方式，根本没有操作系统。在人工操作方式下，用户一个挨一个地轮流使用计算机。每个用户的使用过程大致如下：先把手工编写的程序（机器语言程序）穿成纸带（或卡片），装上输入机，然后经人工操作把程序和数据输入计算机，接着通过控制台开关启动程序。待计算完毕，用户拿走打印结果，并卸下纸带（或卡片）。在这个过程中，需要人工装纸带、人工控制程序运行、人工卸纸带，进行一系列的“人工干预”。这种由一道程序独占机器的情况，在计算机运算速度较慢的时候是可以容忍的，因为此时计算所需的时间相对较长，人工操作时间所占比例还不算很大。

随着计算机技术的发展，计算机在速度、容量、外设的功能和种类等方面都有了很大的发展。比如，计算机的速度有了几十倍、上百倍的提高，手工操作的慢速度和计算机运算的高速度之间形成了一对矛盾，即人机矛盾。

随着计算机速度的持续提高，人机矛盾已到了不可容忍的地步。为了解决这一矛盾，只有设法去掉人工干预，实现作业的自动过渡，这样就出现了批处理技术。

由于当时软件发展处于初级阶段，用于管理的软件还没有产生，因此所有的运行管理和具体操作都由用户自己承担。为了实现作业建立和作业过渡的自动化，引入了批量监督程序（常驻内存的核心代码），每一种语言翻译程序（汇编语言或高级语言的编译程序）或实用程序（如链接程序）都作为监督程序的子例程。子例程相当于子程序，例程就是完成一定功能的特定程序。

1.2.2 联机批处理系统

监督程序的工作对象是以作业流形式提供的。每个用户需要计算机解决的计算工作均组

织成一个作业。每个作业有一个与程序分开的说明文件，即作业说明书，它提供了用户标识、用户要使用的编译程序，以及所需要的系统资源等基本信息。每个作业还包含一个程序和—些原始数据，最后是该作业的终止信息。终止信息给监督程序一个信号，表示此作业已经结束，应为下一个用户作业做好服务准备。

各用户把自己的作业交给机房，由操作员把一批作业装到输入设备上（如果输入设备是纸带输入机，则这一批作业在一盘纸带上；如果输入设备是读卡机，则作业在一叠卡片上），然后在监督程序控制下送到外部存储器，如磁带、磁鼓或磁盘。

为了执行一个作业，批处理监督程序将解释这个作业的说明书，若系统资源能满足其要求，则将该作业调入内存，并从外部存储器上输入所需要的编译程序。编译程序将用户源程序翻译成目标代码，然后由链接装配程序把编译后的目标代码及其所需的子程序装配成一个可执行的程序，接着开始执行。计算完成后输出该作业的计算结果。只有一个作业处理完毕后，监督程序才可以自动地调度下一个作业进行处理，依次重复上述过程，直到该批作业全部处理完毕。在这种批处理系统中，作业的输入/输出是联机的，也就是说作业从输入机到磁带，由磁带调入内存，以及结果的输出打印都是由 CPU 直接控制的。

随着 CPU 速度的不断提高，CPU 和输入/输出设备之间的速度差距就形成了一对矛盾。因为在进行输入/输出时，CPU 是空闲的，高速的 CPU 要等待慢速的输入/输出设备的工作，不能发挥 CPU 应有的效率。

1.2.3 脱机批处理系统

为了克服联机批处理的缺点，在批处理系统中引入了脱机输入/输出技术，从而形成了脱机批处理系统。脱机批处理系统由主机和卫星机组成，卫星机又称外围计算机，它不与主机直接连接，只与外部设备打交道。作业通过卫星机输入到磁带上，当主机需要输入作业时，就把输入带同主机连上。主机从输入带上把作业调入内存，并予以执行。作业完成后，主机把结果记录到输出带上，再由卫星机把输出带上的信息打印输出。这样，主机摆脱了慢速的输入/输出工作，可以较充分地发挥它的高速计算能力。同时，由于主机和卫星机可以并行操作，因此脱机批处理系统与早期的联机批处理系统相比大大提高了系统的处理能力。

批处理系统是在解决人机矛盾及高速度的 CPU 和低速度的 I/O 设备间矛盾的过程中发展起来的。它的出现改善了 CPU 和外设的使用情况，实现了作业的自动定序、自动过渡，从而使整个计算机系统的处理能力得以提高。但批处理系统仍存在许多缺陷，比如卫星机与主机之间的磁带装卸仍需人工完成，操作员需要监督机器的状态等。如果一个程序进入死循环，系统就会踏步不前，只有当操作员提出请求，要求终止该作业，删除它并重新启动，系统才能恢复正常运行。当目标程序执行一条引起停机的非法指令时，机器就会错误地停止运行。此时，只有操作员进行干预，即在控制台上按启动按钮后，程序才会重新启动运行。并且，由于系统没有任何保护自己的措施，无法防止用户程序破坏监督程序和系统程序。系统保护的问题亟待解决。

1.2.4 执行系统

20 世纪 60 年代初期，计算机硬件获得了两个方面的迅速发展，一是通道的引入，二是中断技术的出现，这两项重大成果使操作系统进入了执行系统阶段。

通道是一种输入/输出专用处理器，它能控制一台或多台外部设备工作，负责外部设备与

内存之间的数据传输。它一旦被启动，就能独立于 CPU 运行，这样就可使 CPU 和通道并行操作，而且 CPU 和各种外部设备也能并行操作。

中断是指当 CPU 接到外部硬件（如 I/O 设备）发来的信号时，马上停止原来的工作，转去处理这一事件，在处理完了以后，CPU 又回到原来的工作点继续工作。

借助于通道、中断技术，输入/输出工作可以在 CPU 控制之下完成。这时，原有的监督程序不仅要负责调度作业自动地运行，而且还要提供输入/输出控制功能（即用户不能直接使用启动外设的指令，它的输入/输出请求必须通过系统去执行），它比原有的功能增强了。这个扩展后的监督程序常驻内存，称为执行系统。执行系统比脱机处理前进了一步，它节省了卫星机，降低了成本，而且同样能支持主机和通道、主机和外部设备的并行操作。在执行系统中，用户程序的输入/输出工作是委托给执行系统实现的，由执行系统检查其命令的合法性，提高了系统的安全性，可以避免不合法的输入/输出命令对系统的威胁。

批处理系统和执行系统的普及，发展了标准文件管理系统和外部设备的自动调节控制功能。许多成功的批处理操作系统在 20 世纪 50 年代末到 60 年代初期开发成功，比较著名的有 FMS（FORTRAN Monitor System）和 IBSYS（IBM 为 7094 机配备的操作系统）。

1.2.5 多道批处理系统

中断和通道技术出现以后，输入/输出设备和 CPU 可以并行操作，初步解决了高速 CPU 和低速外部设备之间的矛盾，提高了计算机的工作效率。但人们很快就发现，这种并行是有限度的，并不能完全消除 CPU 对外部传输的等待。比如，一个作业在运行过程中请求输入一批数据，当纸带输入机花 1000ms 输入 1000 个字符后，CPU 只花 300ms 就处理完了，而这时，第二批输入数据还需 700ms 才能输入完毕。因此，尽管 CPU 具有和外部设备并行工作的能力，但是在这种情况下无法让它多做工作，如图 1.2 所示。

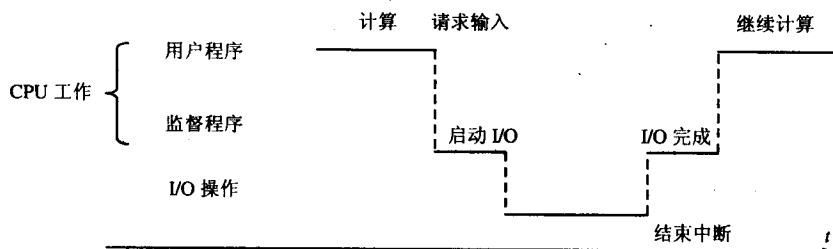


图 1.2 单道程序工作示例

在输入操作结束之前，CPU 处于空闲状态，其原因是输入/输出操作与本道作业相关。商业数据处理、文献情报检索等任务涉及的计算量比较少，而输入/输出量比较大，所以需要较多地调用外部设备。当由慢速的机械传动读卡机、纸带输入机或从磁带、磁盘等设备输入数据到存储器时，CPU 不得不等待。在处理结束后，又有很多时间被耗费在处理器等待通道将结果送到磁带、磁盘或用机械打印机打印在纸上。而科学和工程计算任务，主要涉及的是计算量大而使用外部设备较少的作业，因而当 CPU 运算时，外部设备经常处于空闲状态。此外，计算机在处理一些小题目时，存储器空间也未能得到充分利用。以上种种情况说明了单道程序工作时，计算机系统的各部件的效能没有得到充分发挥。那么，为了提高设备的利用率，能否在系统内同时存放几道程序呢？于是引入了多道程序的概念。

多道程序设计技术是在计算机内存中同时存放几道相互独立的程序，使它们在管理程序