

可下载教学资料

<http://www.tup.tsinghua.edu.cn>

21世纪普通高校计算机公共课程规划教材

# 计算机操作系统教程

马海波 王德广 编著

清华大学出版社

21世纪普通高校计算机公共课程规划教材

# 计算机操作系统教程

马海波 王德广 编著

清华大学出版社  
北京

## 内 容 简 介

本书采用理论与实际相结合的方式,以传统操作系统知识为主,参照本科教学大纲和考研统考大纲,既讲述传统和现代理论,又介绍最新的开发和应用技术。全书共分5章,内容包括操作系统概述、进程管理、内存管理、设备管理和文件管理。

操作系统是计算机科学与技术专业、网络工程专业、软件工程专业的基础课程,本书面向高校相关专业师生,同时也可作为计算机研究与开发人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

## 图书在版编目(CIP)数据

计算机操作系统教程/马海波,王德广编著. —北京:清华大学出版社,2009.9  
(21世纪普通高校计算机公共课程规划教材)

ISBN 978-7-302-20462-6

I. 计… II. ①马… ②王… 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(2009)第151088号

责任编辑:梁颖 林都嘉

责任校对:梁毅

责任印制:王秀菊

出版发行:清华大学出版社

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

社总机:010-62770175

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

地址:北京清华大学学研大厦A座

邮编:100084

邮购:010-62786544

印刷者:北京密云胶印厂

装订者:三河市兴旺装订有限公司

经销:全国新华书店

开本:185×260 印张:18.5 字数:462千字

版次:2009年9月第1版 印次:2009年9月第1次印刷

印数:1~4000

定价:28.00元

---

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系  
调换。联系电话:(010)62770177 转 3103 产品编号:030175-01

# 出版说明

---

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和教学方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程(简称‘质量工程’)”,通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机公共课程领域,以公共基础课为主、专业基础课为辅,横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

(1) 面向多层次、多学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映各层次对基本理论和原理的需求,同时加强实践和应用环节。

(2) 反映教学需要,促进教学发展。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现教学质量和教学改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材配套,同一门课程有针对不同层次、面向不同专业的多本具有各自内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制定教材规划时要依靠各课程专家在调查研究本课程教

材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主题。书稿完成后要真实实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪普通高校计算机公共课程规划教材编委会

联系人: 梁颖 liangying@tup.tsinghua.edu.cn

# 前 言

---

操作系统是计算机科学与技术专业、网络工程专业、软件工程专业必修的一门重要的专业基础课程,同时也是从事计算机研究与应用开发人员的必修课程。

从 2009 年开始,计算机科学与技术学科硕士研究生入学考试的专业综合考试由国家统一命题,考试内容包括 4 门课程,即数据结构、计算机组成原理、操作系统和计算机网络。由此可见,操作系统在计算机学科中的重要地位。作者根据多年的教学和科研的经验与体会,并汲取国内外操作系统方面优秀教材的精华,结合我国的国情,本着提高学生素质、培养创新意识的精神,遵循本科教学大纲的要求,兼顾考研的需要,力求编写一本在国内外有较大影响的教材。

以往学生在学习操作系统的课程中,感觉非常的吃力,往往像学习文科专业一样死记硬背,难以理解其中的原理,所以许多知识只知其然,不知其所以然。本教材力图从计算机和管理者相结合的角度,试图让学生从一个团队管理者的角度去发现问题,提出问题,然后启发学生思考问题,并找出解决问题的方法,尽可能使学生学习操作系统的课程变得轻松、有兴趣,激发学生主动学习的热情,提高学习的效率和效果。

本教材采用理论与实际相结合的方式,以传统操作系统的知识为主,编写过程中参照本科教学大纲和考研统考大纲;既讲述传统和现代理论,又介绍最新开发和应用技术,同时还强调学生的动手实践能力,学以致用;编写力图简单、通俗,尽量涵盖所有的知识点,力争让学生轻松、快乐地掌握操作系统的精髓。

本书第 1、2、3 章由王德广编写,第 4、5 章由马海波编写,编写过程中主要参考的资料见本书最后的参考文献,也包括 Internet 上提供的一些技术资料,在此一并表示感谢。感谢申广忠、张雪对本书的校对,并提出了宝贵的意见,同时制作了课件和课后习题。

由于时间仓促,作者水平有限,书中难免有纰漏,欢迎广大读者多提宝贵意见。

编 者

2009 年 5 月 8 日

于大连交通大学

# 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 什么是操作系统	1
1.1.2 计算机系统概述	3
1.1.3 操作系统主要功能	13
1.1.4 操作系统的地位	16
1.1.5 操作系统的主要特性	18
1.2 操作系统的发展历史	20
1.2.1 手工操作阶段	20
1.2.2 单道批处理系统	22
1.2.3 多道批处理系统	23
1.2.4 分时系统	27
1.2.5 实时系统	29
1.3 操作系统的进一步发展	30
1.3.1 个人 PC 操作系统	30
1.3.2 网络操作系统	31
1.3.3 分布式操作系统	31
1.3.4 嵌入式操作系统	32
1.3.5 大型机操作系统	33
1.3.6 服务器操作系统	33
1.3.7 多处理器操作系统	33
1.3.8 智能卡操作系统	34
1.4 操作系统大观	34
1.4.1 Windows 操作系统	34
1.4.2 UNIX 操作系统	35
1.4.3 Linux 操作系统	36
1.4.4 FreeBSD 操作系统	38
1.4.5 Netware 操作系统	40
1.4.6 AIX 操作系统	40
1.4.7 Solaris 操作系统	41

1.4.8	Mac OS 操作系统 .....	41
1.4.9	其他操作系统 .....	42
1.5	操作系统的结构 .....	45
1.5.1	操作系统的构件 .....	45
1.5.2	整体结构 .....	47
1.5.3	层次结构 .....	47
1.5.4	虚拟机 .....	48
1.5.5	客户机/服务器(C/S)系统 .....	48
1.6	系统调用 .....	48
1.6.1	什么是系统调用 .....	48
1.6.2	系统调用的实现 .....	50
1.6.3	Trap 与 Interrupt .....	51
1.6.4	系统调用与过程(函数)的区别 .....	52
	小结 .....	52
	课后习题 .....	53
	讨论题 .....	55
<b>第 2 章</b>	<b>进程管理</b> .....	<b>56</b>
2.1	进程 .....	56
2.1.1	进程的引入 .....	56
2.1.2	进程的状态 .....	61
2.1.3	进程控制块 .....	64
2.1.4	进程的映像 .....	68
2.1.5	OS 的控制结构 .....	70
2.1.6	进程的控制 .....	72
2.2	线程 .....	76
2.2.1	线程的引入 .....	76
2.2.2	线程控制块 .....	80
2.2.3	线程的状态 .....	81
2.2.4	线程的级别 .....	81
2.3	进程间通信 .....	85
2.3.1	竞争与协作 .....	85
2.3.2	临界区 .....	86
2.3.3	忙等待的互斥 .....	88
2.3.4	睡眠与唤醒 .....	91
2.3.5	信号量及 P、V 原语 .....	93
2.3.6	经典 IPC 问题 .....	100
2.3.7	管程(monitor) .....	105
2.3.8	消息(message) .....	107



2.4	进程调度 .....	109
2.4.1	调度的概念 .....	109
2.4.2	批处理系统的调度 .....	115
2.4.3	交互式系统的调度 .....	118
2.4.4	实时系统的调度 .....	124
2.4.5	策略与机制 .....	125
2.4.6	线程调度 .....	125
2.5	死锁 .....	126
2.5.1	死锁产生的原因 .....	127
2.5.2	死锁产生的条件 .....	128
2.5.3	解决死锁的方法 .....	128
	小结 .....	138
	课后习题 .....	140
	思考题 .....	142
<b>第3章</b>	<b>内存管理 .....</b>	<b>143</b>
3.1	内存介绍 .....	143
3.1.1	内存 .....	143
3.1.2	相关概念 .....	147
3.2	单道系统的内存管理 .....	151
3.3	多道系统的分区管理 .....	153
3.3.1	固定分区管理 .....	153
3.3.2	动态分区管理 .....	154
3.3.3	查找算法 .....	158
3.3.4	可重定位分区分配 .....	159
3.3.5	分区管理的优缺点 .....	161
3.4	页式管理 .....	162
3.4.1	页式管理的实现原理 .....	162
3.4.2	页式地址变换 .....	164
3.4.3	相联存储器和块表 .....	165
3.4.4	页的共享与保护 .....	167
3.5	段式管理 .....	168
3.5.1	段式管理的引入 .....	168
3.5.2	段式管理的实现原理 .....	168
3.5.3	段的共享与保护 .....	170
3.5.4	分段和分页的比较 .....	170
3.6	内存扩充 .....	171
3.6.1	覆盖技术 .....	171
3.6.2	交换技术 .....	172

3.6.3	虚拟存储技术	172
3.7	请求页式管理	175
3.7.1	请求页式管理的思想及实现	175
3.7.2	多级页表	181
3.7.3	反置页表	183
3.7.4	请求页式管理的页面置换算法	184
3.7.5	请求页式管理设计时应注意的几个问题	193
3.7.6	请求页式管理性能	195
3.8	请求段式管理	196
3.8.1	请求段式管理的思想及实现	196
3.8.2	段的动态链接	198
3.8.3	段的共享	198
3.9	请求段页式管理	199
3.9.1	请求段页式管理的基本思想	199
3.9.2	请求段页式管理的实现原理	200
小结		202
课后习题		204
思考题		206
<b>第4章</b>	<b>设备管理</b>	<b>208</b>
4.1	概述	208
4.1.1	设备管理的目的	208
4.1.2	设备管理的任务	208
4.1.3	设备管理的功能	209
4.2	I/O 系统	209
4.2.1	I/O 设备分类	209
4.2.2	设备控制器	211
4.2.3	I/O 通道	212
4.2.4	总线系统	213
4.3	I/O 控制方式	214
4.3.1	程序直接控制方式	214
4.3.2	中断驱动 I/O 方式	215
4.3.3	DMA 控制方式	218
4.3.4	I/O 通道控制方式	219
4.4	缓冲技术	220
4.4.1	缓冲的引入	220
4.4.2	单缓冲	221
4.4.3	双缓冲	222
4.4.4	循环缓冲	222

4.4.5	缓冲池	223
4.5	设备分配	225
4.5.1	设备控制数据结构	225
4.5.2	设备分配原则	226
4.5.3	设备独立性	228
4.5.4	设备分配和回收算法	228
4.5.5	虚拟设备	229
4.6	设备处理	232
4.6.1	设备处理程序	232
4.6.2	设备驱动程序处理过程	232
4.6.3	中断处理程序	233
4.7	磁盘组织与管理	235
4.7.1	磁盘性能简述	235
4.7.2	磁盘调度	237
4.7.3	磁盘高速缓存	241
4.7.4	磁盘容错技术	242
	小结	246
	课后习题	247
	思考题	249
<b>第5章</b>	<b>文件管理</b>	<b>250</b>
5.1	文件	250
5.1.1	文件定义	250
5.1.2	文件分类	250
5.1.3	文件操作	252
5.2	文件系统	252
5.2.1	文件系统的功能	253
5.2.2	文件系统要解决的问题	253
5.2.3	理想文件系统的特征	253
5.2.4	文件系统的结构模型	254
5.3	文件逻辑结构	255
5.3.1	逻辑结构类型	255
5.3.2	顺序文件	257
5.3.3	链接文件	257
5.3.4	索引文件	258
5.3.5	索引顺序文件	259
5.4	文件分配	260
5.4.1	连续分配	261
5.4.2	链接分配	262

5.4.3	索引链接分配·····	263
5.5	目录管理·····	265
5.5.1	目录管理的要求·····	265
5.5.2	文件控制块 FCB 和索引结点·····	265
5.5.3	索引结点的管理·····	266
5.5.4	目录结构·····	267
5.5.5	目录查询技术·····	270
5.6	文件存储空间管理·····	272
5.6.1	空闲表法·····	272
5.6.2	位示图法·····	273
5.6.3	空闲链表法·····	274
5.6.4	空闲块成组链接法·····	274
5.7	文件的共享与保护·····	276
5.7.1	基于索引结点的共享方式·····	276
5.7.2	符号链接法文件共享·····	277
5.7.3	文件的安全保护·····	278
	小结·····	280
	课后习题·····	281
	思考题·····	282
	参考文献·····	283

完整的计算机系统是由硬件和软件两大部分组成。操作系统(Operating System, OS)是计算机系统所有软件中最基础、最核心、最底层的软件,它介于计算机硬件和终端用户之间,为终端用户操作计算机硬件提供环境,管理和控制计算机的硬件和软件资源,扩展计算机的功能。

## 1.1 操作系统的概念

### 1.1.1 什么是操作系统

多数计算机用户都有一些使用操作系统的体验,也知道一些操作系统的名称,如 Windows XP、Windows 2000、Linux 等。但什么是操作系统呢?业界至今没有形成一个统一的标准化的定义。之所以会出现这种情况,一方面是因为操作系统执行两个相对独立的任务,即计算机功能及性能的扩展和计算机资源的管理;另外一方面的原因是要取决于从什么角度看待操作系统,即是用户的观点还是系统的观点?

#### 1. 用户观点

作为一个普通的计算机用户,你希望操作系统有什么样的功能呢?

假设刚从电子城配了一台个人计算机(Personal Computer, PC),可惜当时由于时间紧,没安装 Windows XP,当把计算机在家里连接好以后并打开,没有出现熟悉的 Windows XP 的画面,这时你的感觉如何?

鼠标不能用,没有熟悉的桌面,不知道文件在哪里,没有播放 MP3 的播放器,没有播放视频的暴风影音,没有熟悉的 IE 浏览器或 Firefox 浏览器,没有迅雷或 BT 软件,下载不了想要的文件,其实也根本找不到所需要的文件资源,因为你根本不知道怎样操作才能上网……这一切如何面对?

怎么才能控制和使用这台计算机?应该有一个软件提供给我们,让我们能够方便地操控面前的这台机器。

**第一个结论:从用户的观点看,操作系统应该提供一个人与计算机打交道的接口,也就是人机接口。用户可以通过操作系统提供的这个接口使用和控制计算机,从而达到让计算机为用户服务的目的。用汽车做一个不完全一致的类比,汽车的设计者必须提供给用户方向盘和变速箱(不论是自动挡的还是手动挡的),如果没有方向盘和变速箱,让驾驶员怎么开这台车呢?**

那么接着往下想,这种接口是什么样子的呢?是一个接口,还是几种不同的接口?

第一,对于那些非计算机专业的人士(如会计人员、管理人员、销售员等)来说,他们具备的计算机知识有限,他们使用计算机主要是把计算机当成他们自己专业的辅助工具,为了使他们自己的业务完成得更加轻松、准确。他们并不会花大量的时间研究计算机的技术,所以应该给他们提供一种**图形界面的接口**(Graphical User Interface, GUI)。使用这种接口,用户可以通过鼠标操作计算机,让工作变得更轻松。用户不必知道什么命令和原理,但要求操作系统的编程人员必须把图形界面设计得非常人性化,界面非常美观、赏心悦目,让使用者在使用计算机的过程中,感觉是一种享受。但必须清楚,这种赏心悦目的界面背后必然耗费计算机的大量资源,好在这种用户对自己的资源要求不是很高。

第二,对于那些用计算机完成科学计算的用户来说,他们不追求界面的美观,他们也稍微懂得一些计算机的基本操作知识,他们的需求是让计算机尽最大可能地发挥它的效率,以便更快地完成他们的计算任务。对于这种用户,GUI 图形界面接口对他们来说,就不是最好的选择,因为他们希望计算机的所有资源尽可能地用于科学计算。那么就应该提供一种**命令行**(Command Line, CL)**接口**,这种接口不使用图形界面,那么图形界面所耗费的大量的计算机资源(主要是 CPU 和内存的资源)也就不存在了,系统资源会尽可能地用于用户的计算需求。

第三,对于那些计算机的专业人士呢,操作系统应该提供什么样的接口呢?回想一下,大家都使用过打印机,当在 Word 中输入一行文字以后,直接单击“打印”按钮输出,打印机会将一张完整的 A4 纸走出,但这张 A4 纸上只有一行文字,这是最常见的现象。但是如果去超市购物,在收银台的微型打印机里走出的纸,也是一张完整的吗?肯定不是的,超市微打的走纸长度是和购物的多少有关系,购买商品的种类越多,走纸越多。那么为什么在 Word 中完成不了这个功能呢?这就需要计算机的软件编程人员对微型打印机进行编程控制,也就是说,操作系统需要给计算机专业人员提供一个**软件编程的接口**。计算机的专业人员通过操作系统提供的这个编程接口,可以控制计算机的其他外设,修改他们的工作方式。

另一方面,裸机(仅有硬件的计算机)提供的机器语言难记、难用又难懂,一般的程序员和普通用户不愿意、也没有能力在这种环境下编程或使用计算机。在裸机之上安装操作系统之后,就把硬件的细节与程序员及用户隔离开。用户可以使用系统提供的各种命令,直接打开文件、读写文件、更改目录、将文件复制出来,等等。在做这些事情的时候,用户只关心自己要实现的目标,并未考虑到硬件如何动作,从而隐藏了底层硬件的特性。经过操作系统的加工,呈现在用户面前的机器是功能更强、使用更方便的机器。

对于用户来说,不同的用户,对计算机的使用所追求的目标是不同的。有的用户追求计算机的数据处理能力、有的用户追求计算机的实时处理能力、有的用户追求对多用户、多任务的并发能力。所以哪怕是同一台计算机,由于装上了不同的操作系统,所表现出来的能力也是不同的。也就是说,操作系统能够虚拟出一台功能比以前更强的计算机,扩展了现有计算机的能力,根据要求的不同,计算机的表现可以不同。

从上述角度看,操作系统的作用是为用户提供一台等价的扩展机器(Extended Machine),或者称虚拟机(Virtual Machine)。

## 2. 系统观点

系统观点是从系统内部实现的角度来看待操作系统的作用。操作系统是硬件之上的第一层软件,它要管理计算机系统中各种硬件资源和软件资源的分配问题,如 CPU 时间、内存空间、文件存储空间、I/O 设备等,要解决大量对资源请求的冲突问题,决定把资源分配给

谁、何时分配、分配多少等,使得资源的利用高效且公平。现代计算机由 CPU、内存、各种 I/O 设备、总线组成,操作系统的功能就是管理这些硬件资源和数据、程序等软件资源,控制和协调各种程序对这些资源的使用,尽可能最大化地发挥各种资源的作用。这样,操作系统就是资源的分配者。

再者,操作系统要对 I/O 设备和用户程序加以控制,保证设备的正常运行,防止非法操作,及时诊断设备的故障等。从这个意义上讲,操作系统又是工作流程的调度者。

从以上的论述中可以看到,操作系统的概念如图 1-1 所示。

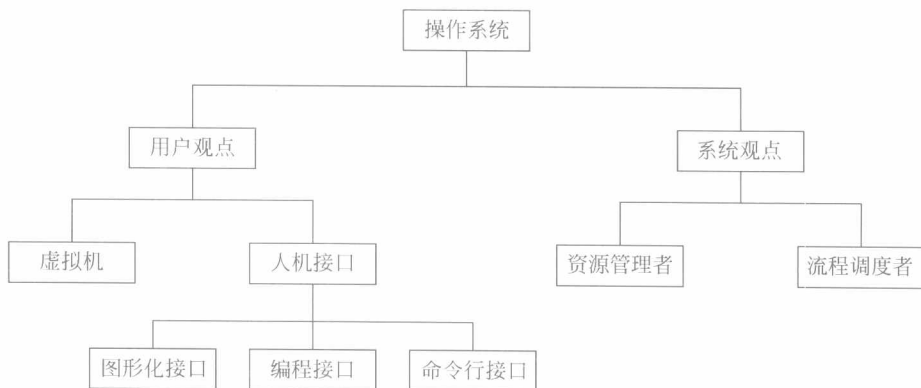


图 1-1 操作系统

### 1.1.2 计算机系统概述

计算机系统是由硬件和软件组成。硬件是软件建立与活动的基础,而软件又是对硬件功能的扩充。没有硬件,就失去了计算机系统的物理基础,软件也就无法存在了。反过来,如果只有硬件而没有软件,就会像最初的计算机那样,很难使用,没有活力,应用价值有限。硬件与软件有机地结合在一起,相辅相成,才使计算机技术飞速发展,且在当今信息时代占据举足轻重的地位。

操作系统与运行操作系统的计算机硬件联系密切。操作系统扩展了计算机的指令集,并管理计算机的资源,为了能够更好地工作,操作系统必须了解和掌握大量的硬件,需要了解硬件如何面对程序员。下面简述一下计算机的硬件结构,这些正是操作系统所管理的对象。

从概念上讲,一台简单的个人计算机可以抽象为如图 1-2 所示的模型。CPU、内存以及 I/O 设备都由一条系统总线连接起来,并通过总线与其他的设备进行通信。

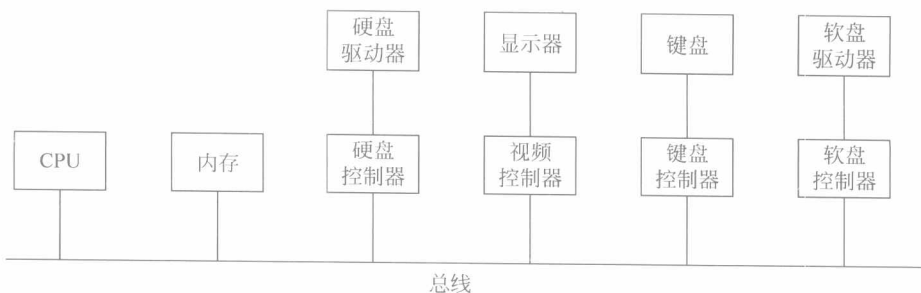


图 1-2 微型计算机结构组成

## 1. CPU(处理器)

计算机的“大脑”是 CPU,它从内存中取出指令并执行。在每个 CPU 的基本周期中,首先从内存中取出指令,解码以确定操作类型和操作数,接着执行之,然后取指、解码并执行下一条指令。按照这一方式,程序被执行完成。

每个 CPU 都有一套可执行的专门指令集,所以,Pentium 不能执行 SPARC 程序,而 SPARC 也不能执行 Pentium 程序。由于通过访问内存得到的指令或数据的时间要比执行指令花费的时间长得多,因此所有的 CPU 都有一些用来保存关键变量和临时结果的寄存器。这样,通常在指令集中提供一些指令,用以将一个字从内存调入寄存器,以及将一个字从寄存器写回内存。其他指令可以把来自寄存器和内存的操作数组合,或者用二者产生一个结果,例如,将两个字相加并把结果存在寄存器或内存中。

除了用来保存变量和临时结果的通用寄存器之外,多数计算机还有一些对程序员可见的专用寄存器。其中之一就是程序计数器,它保存了将要取出的下一条指令的内存地址,在指令取出之后,程序计数器就被更新以便指向后继的指令。

另外一个寄存器是堆栈指针寄存器,它指向内存中当前栈的顶端。该栈含有已经进入但是还没有退出的每一个过程的中间数据,在堆栈中保存了有关的输入参数、局部变量以及那些没有保存在寄存器中的临时变量。

第三个寄存器是程序状态字(Program Status Word,PSW)寄存器。这个寄存器包含了条件码位(它由比较指令设置)、CPU 优先级、模式(用户态或核心态)以及各种其他的控制位。用户程序通常读入整个 PSW,但是,只对其中的少量字段进行写操作。在系统调用和 I/O 操作中,PSW 的作用非常重要。

在多道程序设计中,操作系统经常会中止正在运行的某个程序并启动另外一个程序。每次它中止一个正在运行的程序时,操作系统必须保存所有寄存器的内容,这样在稍后该程序被再次运行时,可以把这些寄存器的内容重新装入。

为了改善性能,CPU 设计师早就放弃了一次取指、译码和执行一条指令的简单模型,现在许多 CPU 具有同时取出多条指令的机制。例如,一个 CPU 可以有分开的取指单元、译码单元和执行单元,于是当它执行指令  $n$  时,它还可以对指令  $n+1$  译码,并且读取指令  $n+2$ ,这样的一种机制称为流水线(pipeline),在图 1-3(a)中是一个有着三个阶段的流水线示意图。更长的流水线也是常见的。在多数的流水线设计中,一旦一条指令被取进流水线中,它就必须被执行完毕,即便前一条取出的指令是条件转移,它也必须被执行完毕。流水线使得编译器和操作系统的编写者很头痛,因为它造成了在机器中实现这些软件的复杂性问题。

比流水线更先进的设计是一种超标量 CPU,如图 1-3(b)所示。在这种设计中,有多个执行单元,例如,一个 CPU 用于整数算术运算,一个 CPU 用于浮点算术运算,而再一个用于布尔运算。两个或更多的指令被同时取指、译码并转储到一个保持缓冲区中,直至它们执行完毕。只要有一个执行单元空闲,就检查保持缓冲区中是否还有可处理的指令,如果有,就把指令从缓冲区中移出并执行之。这种设计存在一种隐含的作用:程序的指令经常不按顺序执行。在多数情况下,硬件负责保证这种运算的结果与顺序执行指令时的结果相同。但是,仍然有部分令人烦恼的复杂情形被强加给操作系统处理,在后面同步信号量的内容中会讨论这种情况。



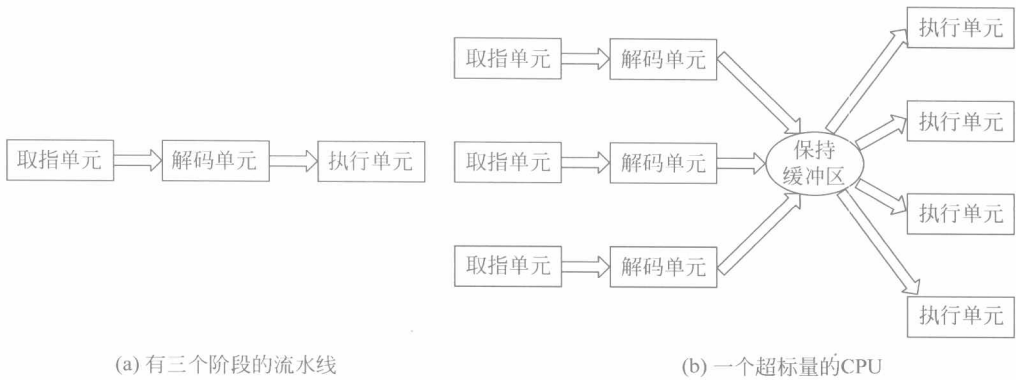


图 1-3

除了用在嵌入式系统中的非常简单的 CPU 之外,多数 CPU 都有两种模式,即核心态(管态)和用户态(目态),在 PSW 寄存器中有一个模式位控制这两种模式。当 CPU 执行操作系统的代码时,CPU 处于核心态,CPU 可以执行指令集中的每一条指令,并且使用硬件的每一种功能,访问整个硬件。相反,当 CPU 执行用户程序的代码时,CPU 处于用户态,仅允许执行整个指令集的一个子集和访问所有功能的一个子集。一般而言,在用户态中有关 I/O 和内存保护的所有指令是禁止的。当然,将 PSW 寄存器中的模式位设置成核心态也是禁止的。

为了从操作系统中获得服务,用户程序必须使用系统调用(System Call),系统调用陷入内核并调用操作系统。TRAP 指令从用户态切换成核心态,并启用操作系统。当有关工作完成之后,在系统调用后面的指令处把控制权返回给用户程序。在 1.6 节将具体解释系统调用的过程。

## 2. 存储器

在任何一种计算机中的第二种主要的部件就是存储器。在理想的情形下,存储器应该极为快速(快于执行一个指令,这样 CPU 不会受到存储器的限制)、充分大,并且非常便宜。但是目前的技术无法同时满足所有这三个目标,于是采用了不同的处理方式。

### 1) 存储器系统结构

存储器系统以一种分层次的结构构造,如图 1-4 所示。

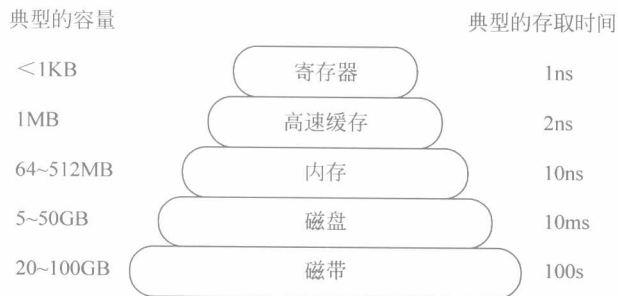


图 1-4 典型的存储层次结构(数字是粗略的估计)