

21世纪高等学校规划教材 | 计算机科学与技术

# 操作系统原理

韩其睿 主编



清华大学出版社

21世纪高等学校规划教材 | 计

# 操作系统原理

韩其睿 主编

清华大学出版社  
北京

## 内 容 简 介

本书是一本为高等学校师生编写的教材,讲述了当代操作系统的基本原理,全书由7章组成,详细介绍了进程的概念、进程间通信、线程、信号量、消息传递、处理机调度、存储管理、输入/输出设备管理、文件系统等。考虑到实验教学的要求,本书安排了Linux操作系统的一些实例,还配有丰富的习题以及习题答案。

本书可作为高等学校计算机技术、软件工程、网络工程专业学生的教材,也可供相关技术人员参考。

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

操作系统原理/韩其睿主编. —北京:清华大学出版社,2013

21世纪高等学校规划教材·计算机科学与技术

ISBN 978-7-302-32725-7

I. ①操… II. ①韩… III. ①操作系统—高等学校—教材 IV. ①TP316

中国版本图书馆CIP数据核字(2013)第130858号

责任编辑:高买花 王冰飞

封面设计:傅瑞学

责任校对:白蕾

责任印制:何芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

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)

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:13.75

字 数:333千字

版 次:2013年8月第1版

印 次:2013年8月第1次印刷

印 数:1~2000

定 价:25.00元

# 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。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作,提高教学质量的若干意见》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上。精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展,顺应并符合21世纪教学发展的规律,代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版

社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。推出的特色精品教材包括:

(1) 21 世纪高等学校规划教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。

(2) 21 世纪高等学校规划教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 21 世纪高等学校规划教材·电子信息——高等学校电子信息相关专业的教材。

(4) 21 世纪高等学校规划教材·软件工程——高等学校软件工程相关专业的教材。

(5) 21 世纪高等学校规划教材·信息管理与信息系统。

(6) 21 世纪高等学校规划教材·财经管理与应用。

(7) 21 世纪高等学校规划教材·电子商务。

(8) 21 世纪高等学校规划教材·物联网。

清华大学出版社经过三十多年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

联系人:魏江江

E-mail: weijj@tup.tsinghua.edu.cn

# 前言

本书是一本为高等学校师生编写的教材,旨在介绍操作系统的概念、结构和原理,目标是向读者展现操作系统的本质特点。

操作系统作为计算机系统软件的核心,无论理论上还是实践上都有着丰富的内容。计算机系统和用途多种多样,包括面向单用户的个人计算机、中等规模的共享系统、大型计算机和超级计算机以及诸如实时系统的专门系统,适应它们的操作系统也在不断地发展。这些操作系统在结构上、原理上和技术上各有不同的特点。

虽然操作系统多种多样,但是一些基本概念和原理被广泛使用,构成操作系统的理论基础。本书比较详细地讨论了操作系统原理,包括进程的概念、进程间通信、线程、信号量、消息传递、调度算法、存储管理、输入/输出设备管理、文件系统等。

全书共分7章:第1章简要介绍系统硬件结构,操作系统的概念、功能、特征、结构、发展历史、类型等。第2章主要讨论进程和线程的概念,包括进程状态、控制、组织和进程通信。第3章主要讨论处理机管理、调度策略和算法。第4章详细介绍进程的竞争与死锁的处理方法。第5章介绍存储管理的概念和方法。第6章对文件系统做详细介绍,包括磁盘的组织、文件的目录结构、文件保护和文件共享等内容。第7章讨论输入/输出管理,包括I/O控制方式、缓冲技术等。考虑到实验教学的要求,每一章最后都根据本章介绍的原理给出了Linux系统实例。

Linux操作系统作为目前一种主流操作系统,具有一系列的优点。Linux系统的一个重要特点是其源代码的开放性,这为操作系统的学习和实践提供了方便。

本书第1章由韩其睿编写;第2章第1节至第3节由任淑霞编写,第4节至第7节由姚清爽编写;第3章由王佳欣编写;第4章第1节至第4节、第7节由马洁编写,第5节、第6节由冯堃编写;第5章第1节、第2节由孙学梅编写,第3节至第5节由赵茜编写;第6章由李昕编写;第7章由陈香凝编写。全书由韩其睿负责审阅定稿。

编者

2013年6月

# 目 录

第 1 章 计算机系统概述 .....	1
1.1 计算机硬件结构 .....	1
1.1.1 基本构成 .....	1
1.1.2 处理器 .....	2
1.1.3 存储设备 .....	4
1.1.4 I/O 结构 .....	5
1.2 操作系统的概念 .....	7
1.2.1 操作系统的定义与地位 .....	7
1.2.2 如何理解操作系统 .....	8
1.2.3 操作系统的特征 .....	9
1.2.4 操作系统的功能 .....	10
1.3 操作系统的发展和分类 .....	11
1.3.1 操作系统的发展 .....	11
1.3.2 操作系统的分类 .....	15
1.4 操作系统的结构 .....	16
1.4.1 简单模块组合结构 .....	17
1.4.2 层次结构 .....	17
1.4.3 微内核结构 .....	18
1.5 UNIX 操作系统 .....	18
1.5.1 UNIX 操作系统的历史 .....	18
1.5.2 UNIX 操作系统的结构 .....	19
1.6 Linux 操作系统 .....	20
1.6.1 Linux 操作系统的历史 .....	20
1.6.2 Linux 操作系统的特点 .....	21
1.6.3 Linux 操作系统的结构 .....	22
1.7 Windows 操作系统 .....	23
1.7.1 Windows 操作系统的历史 .....	23
1.7.2 Windows 操作系统的结构 .....	24
1.7.3 Windows 2000/XP 的特点 .....	26
本章小结 .....	26
习题 1 .....	27

<b>第 2 章 进程与线程</b> .....	30
2.1 进程的概念 .....	30
2.1.1 多道程序设计 .....	30
2.1.2 进程的概念 .....	31
2.2 进程的状态与转换 .....	32
2.2.1 进程的状态 .....	32
2.2.2 进程状态的转换 .....	32
2.3 进程的描述与控制 .....	34
2.3.1 进程的描述 .....	34
2.3.2 进程的控制 .....	35
2.4 进程的组织 .....	39
2.4.1 进程的组成 .....	39
2.4.2 PCB 的组织方式 .....	39
2.5 进程的通信 .....	40
2.5.1 共享存储器系统 .....	41
2.5.2 消息传递系统 .....	41
2.5.3 管道通信 .....	42
2.6 线程 .....	42
2.6.1 线程的概念 .....	42
2.6.2 线程与进程的比较 .....	44
2.6.3 线程的实现与模型 .....	45
2.7 Linux 系统中的进程 .....	48
2.7.1 Linux 中进程的概念 .....	48
2.7.2 Linux 的进程控制 .....	49
2.7.3 Linux 中进程的通信 .....	50
本章小结 .....	51
习题 2 .....	52
<b>第 3 章 处理机调度</b> .....	55
3.1 调度类型 .....	55
3.2 进程调度 .....	57
3.2.1 引起进程调度的原因 .....	58
3.2.2 进程调度的方式 .....	58
3.3 调度准则 .....	58
3.4 调度算法 .....	60
3.4.1 先来先服务法 .....	60
3.4.2 短作业优先法 .....	61
3.4.3 最短剩余时间优先法 .....	62



3.4.4	高响应比优先法	62
3.4.5	优先级调度	63
3.4.6	时间片轮转法	64
3.4.7	多级队列法	66
3.4.8	多级反馈队列法	66
3.5	线程调度	67
3.6	多处理器调度	68
3.7	实时调度	69
3.7.1	实时任务类型	69
3.7.2	实时调度算法	70
3.8	Linux 系统进程调度	70
3.8.1	Linux 的进程调度	70
3.8.2	Linux 进程调度时机	70
3.8.3	Linux 进程调度策略	71
	本章小结	72
	习题 3	72
<b>第 4 章</b>	<b>进程同步与死锁</b>	<b>78</b>
4.1	进程同步的基本概念	78
4.1.1	并发性	78
4.1.2	与时间有关的错误	79
4.1.3	进程的同步与互斥	80
4.1.4	临界资源和临界区	80
4.2	互斥实现方法	82
4.2.1	硬件方法	82
4.2.2	软件方法	84
4.3	信号量	85
4.3.1	整型信号量机制	86
4.3.2	记录型信号量机制	86
4.3.3	AND 型信号量机制	87
4.4	经典的进程同步问题	88
4.4.1	生产者-消费者问题	88
4.4.2	读者-写者问题	89
4.4.3	哲学家进餐问题	91
4.4.4	打瞌睡的理发师问题	92
4.5	管程	93
4.5.1	使用信号的管程	94
4.5.2	使用通知和广播的管程	97
4.6	死锁	97

4.6.1	死锁的概念 .....	98
4.6.2	死锁的处理策略 .....	99
4.6.3	死锁的预防与避免 .....	99
4.6.4	死锁的检测与恢复 .....	103
4.6.5	处理死锁的综合方式 .....	105
4.7	Linux 系统的进程同步和死锁 .....	106
本章小结	.....	107
习题 4	.....	108
<b>第 5 章</b>	<b>存储管理</b> .....	<b>119</b>
5.1	存储管理的概念 .....	119
5.1.1	地址空间 .....	119
5.1.2	程序装入与链接 .....	120
5.1.3	重定位 .....	121
5.2	内存管理 .....	123
5.2.1	固定分区 .....	123
5.2.2	动态分区 .....	124
5.2.3	覆盖和交换技术 .....	125
5.2.4	分页存储管理 .....	128
5.2.5	分段存储管理 .....	132
5.2.6	段页式存储管理 .....	135
5.3	虚拟存储器管理 .....	137
5.3.1	虚拟存储的概念 .....	138
5.3.2	虚拟页式存储管理 .....	138
5.3.3	虚拟段式存储管理 .....	139
5.3.4	虚拟段页式存储管理 .....	140
5.4	页面置换算法 .....	141
5.4.1	最优页面置换算法 .....	142
5.4.2	最近最少使用页面置换算法 .....	142
5.4.3	先进先出页面置换算法 .....	142
5.4.4	时钟页面置换算法 .....	143
5.4.5	抖动和工作集 .....	144
5.4.6	局部分配策略和全局分配策略 .....	145
5.4.7	页面置换算法小结 .....	146
5.5	Linux 内存管理 .....	146
5.5.1	基本概念和特点 .....	146
5.5.2	Linux 内存管理基本思想和实现 .....	147
5.5.3	Linux 中的分页管理机制 .....	148
5.5.4	Linux 中的虚存段式管理机制 .....	148

本章小结 .....	149
习题 5 .....	149
<b>第 6 章 文件管理 .....</b>	<b>158</b>
6.1 磁盘组织与管理 .....	158
6.1.1 磁盘结构 .....	158
6.1.2 磁盘调度算法 .....	159
6.1.3 常见的磁盘管理任务 .....	160
6.2 文件和文件系统 .....	160
6.2.1 文件和文件系统的概念 .....	160
6.2.2 文件的分类 .....	161
6.3 文件的结构和组织 .....	161
6.3.1 文件的逻辑结构 .....	161
6.3.2 文件的物理结构 .....	161
6.4 文件的目录 .....	162
6.4.1 一级目录 .....	163
6.4.2 二级目录 .....	163
6.4.3 多级目录 .....	163
6.4.4 图形目录 .....	164
6.5 文件的共享 .....	164
6.5.1 共享动机 .....	164
6.5.2 共享方式 .....	164
6.6 文件保护 .....	165
6.6.1 访问类型 .....	165
6.6.2 访问控制 .....	165
6.7 存取方式和存储空间的管理 .....	165
6.7.1 空闲区表 .....	166
6.7.2 位示图 .....	166
6.7.3 空闲块链 .....	166
6.7.4 成组链接 .....	167
6.8 文件系统实现 .....	168
6.8.1 文件系统层次结构 .....	168
6.8.2 目录实现 .....	168
6.8.3 文件实现 .....	169
6.9 文件系统的安全与可靠性 .....	170
6.9.1 文件系统的安全 .....	170
6.9.2 文件系统的可靠性 .....	170
6.10 Linux 系统的文件管理 .....	171
本章小结 .....	175

习题 6 .....	175
<b>第 7 章 输入/输出管理</b> .....	<b>179</b>
7.1 I/O 管理概述 .....	179
7.1.1 I/O 设备分类 .....	179
7.1.2 I/O 设备管理功能 .....	180
7.1.3 设备控制器与 I/O 通道 .....	180
7.2 I/O 软件层次 .....	183
7.2.1 中断处理程序 .....	183
7.2.2 设备驱动程序 .....	183
7.2.3 与设备无关的 I/O 软件 .....	184
7.2.4 用户空间的 I/O 软件 .....	186
7.3 输入/输出控制方式 .....	186
7.3.1 程序直接控制方式 .....	186
7.3.2 中断控制方式 .....	187
7.3.3 直接内存存取控制方式 .....	188
7.3.4 通道控制方式 .....	189
7.4 缓冲技术 .....	189
7.4.1 缓冲技术的引入 .....	189
7.4.2 缓冲的分类 .....	190
7.5 设备分配 .....	193
7.5.1 设备分配中的数据结构 .....	193
7.5.2 设备分配策略 .....	194
7.5.3 设备分配程序 .....	196
7.5.4 Spooling 系统 .....	197
7.6 Linux 的 I/O 管理 .....	198
7.6.1 Linux 的 I/O 管理概述 .....	199
7.6.2 Linux 的 I/O 控制 .....	199
7.6.3 字符设备与块设备管理 .....	201
本章小结 .....	202
习题 7 .....	203

# 第 1 章

## 计算机系统概述

本章简要介绍计算机系统、计算机硬件结构及其相应的技术；讨论操作系统的定义、地位、特征、功能等；介绍操作系统的发展历史和分类；并对现代操作系统的结构进行论述；最后分别介绍 UNIX 系统、Linux 系统和 Windows 系统的发展、结构和特点。

当今计算机系统由硬件系统和软件系统两大部分组成。硬件系统是指计算机的物理设备,包括 CPU、存储器、输入/输出控制器和总线等。软件系统是各种程序和数据的集合。硬件系统是软件系统运行的基础,软件系统对硬件系统的功能进行扩充和完善。硬件系统和软件系统的有机结合构成了功能强大的计算机系统。计算机系统所有的硬件和软件,统称为计算机的资源。人们总是希望能够充分、有效、方便地利用计算机的所有资源,操作系统(Operating System, OS)正是因此而发展起来的。

### 1.1 计算机硬件结构

硬件是计算机系统的核心,单纯的硬件系统通常称为裸机。硬件系统是计算机系统快速可靠工作的基础,操作系统在硬件基础上工作,并对硬件资源进行控制和管理。随着电子技术的发展,计算机硬件技术不断地改进和创新,同时影响着操作系统的发展。因此掌握一些硬件知识对理解操作系统的原理和技术是必要的。

#### 1.1.1 基本构成

计算机硬件系统有 4 个主要的结构化部件,它们是:处理器(控制器和运算器)、主存储器、输入/输出模块(I/O 控制器、外部设备)和系统总线。这些部件以某种方式连接,协同工作,以实现计算机的执行程序这一主要功能。硬件系统连接结构如图 1-1 所示。

##### 1. 处理器(Processor)

处理器用于控制计算机的操作,执行数据处理功能,是计算机硬件的核心,又称 CPU (Central Processing Unit),处理器由控制器、

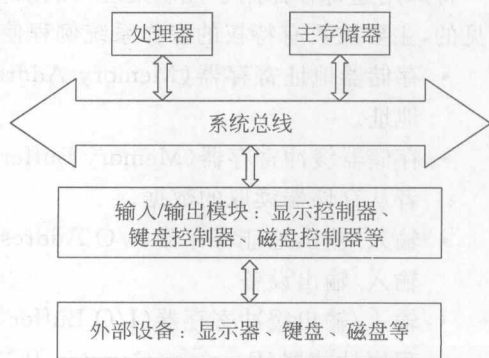


图 1-1 计算机硬件系统连接结构

运算器和一组寄存器组成。

## 2. 主存储器(Main Memory)

主存储器又称内存,用于存储数据和程序。处理器需要的程序和数据都在主存储器中存放,处理器与主存储器直接交换信息。其他部件与处理器的信息交换要通过主存储器。

## 3. 输入/输出模块(I/O Modules)

输入/输出模块用于完成用户信息(程序和数据)和计算机能处理的二进制信息之间的变换和移动。输入/输出模块包括一些通信设备、终端、辅助存储器和 I/O 控制器等。

## 4. 系统总线(System Bus)

系统总线是为处理器、主存储器和输入/输出模块之间提供信息通信的设施。

按照冯·诺依曼型计算机工作原理,用户使用输入设备将程序和数据通过 I/O 模块送入到主存储器中,处理器则从主存储器中逐条取出程序指令,按指令要求完成读取数据、处理数据、写入数据等操作,并将最终处理结果通过 I/O 模块交由输出设备输出。整个运行过程都是在处理器的控制下完成的,指令和数据都通过系统总线来传送。

### 1.1.2 处理器

处理器由控制器和运算器组成。控制器负责根据读入的指令决定计算机的操作,如:是否访问 I/O 设备,是否访问内存或者是否进行数据运算。运算器负责对数据进行加减乘除等算术运算和比较两个数大小等逻辑运算。计算机的全部工作都是在处理器的控制下进行的。如何有效地使用处理器一直是计算机技术研究的重要内容。

#### 1. 寄存器

处理器中包括一组寄存器(Register),有的处理器有上百个寄存器。它们提供一定的存储能力。寄存器的访问速度快但价格昂贵。处理器将指令和数据先加载到寄存器中,然后再进行处理,这样可以提升处理器的处理速度。寄存器可分为特殊用途寄存器和一般用途寄存器。

特殊用途寄存器用于控制处理器的操作,在大多数计算机中,此类寄存器对于用户是不可见的,主要被具有特权的操作系统例程使用。这类寄存器包括:

- 存储器地址寄存器(Memory Address Register, MAR),存放下一次读写的存储器地址。
- 存储器缓冲寄存器(Memory Buffer Register, MBR),存放将要写入存储器的数据或者从存储器读取的数据。
- 输入/输出地址寄存器(I/O Address Register, I/O AR),存放下一次读写的特定的输入/输出设备。
- 输入/输出缓冲寄存器(I/O Buffer Register, I/O BR),存放将要输入/输出的数据。
- 程序计数器(Program Counter, PC),存放下一次读取指令的存储器地址。
- 指令寄存器(Instruction Register, IR),存放从主存储器中取出的指令。

一般用途寄存器用于存取数据和内存地址,如数据寄存器(Data Register)和地址寄存器(Address Register)等。此类寄存器对于用户是可见的。由于寄存器的存取速度比主存储器的存取速度快,用户在编程时可以优先使用这些寄存器,减少处理器对主存储器的访问次数,提高处理器的使用效率。

## 2. 指令的执行

计算机按照程序进行工作。程序是计算机指令的集合。程序存放在内存中。计算机执行程序时,每次在内存中取出一条指令,执行该指令;然后再取下一条指令并执行。程序执行是由不断重复的取指令和执行指令过程组成的。一个单一的指令需要的处理称为一个指令周期,其过程如图 1-2 所示。



图 1-2 指令周期

在每个指令周期的开始,处理器从存储器中取一条指令,指令的地址靠程序计数器保存。一般情况下,每执行完一条指令后,程序计数器总是按顺序指向下一条指令。取出的指令存入处理器的一个寄存器中,该寄存器称为指令寄存器,处理器根据指令的操作内容执行操作。这些操作动作一般包括如下几种。

- 数据传输:将数据在内存和寄存器间进行传递。
- 输入/输出:通过 I/O 模块将数据传送到输出设备或从输入设备输入数据。
- 数据处理:对数据执行算术运算或逻辑运算。
- 程序控制:某些指令可以改变程序的执行顺序。处理器可从该类指令中取出下一条指令在内存中的位置,并将其保存在程序计数器中。

人们利用计算机的指令,可以编写出功能强大的各种复杂程序,计算机通过处理器按程序不断地取出指令,执行指令,如此反复,完成各项任务。

## 3. 数据交换

计算机的主要功能是处理数据。计算机处理器一般只与内存交换数据。要处理的数据先装入内存,处理器根据指令从内存中取出数据,进行处理,处理后的数据也是先送到内存,然后再送到其他设备(如输出设备或磁盘)。内存与外部设备的数据交换是在处理器的控制下完成的。这是计算机数据交换的主要方式。

为了减轻处理器的压力,在某些情况下,允许 I/O 模块直接与内存交换数据,现在大多数控制器都提供了这种功能,称为直接内存存取(Direct Memory Access, DMA)。此时处理器允许 I/O 模块具有对存储器读写数据的特权,这样 I/O 设备与存储器之间的数据传递就无须通过处理器来完成。

另外为了加快处理数据的速度,有些 I/O 模块可以与处理器直接交换数据,正如处理器可以通过指定存储单元的地址启动对存储器的读写一样,处理器可以不经内存,直接从 I/O 模块中读数据,送入到处理器的寄存器;或将处理器的寄存器中的数据直接写向 I/O 模块。

### 1.1.3 存储设备

计算机处理的信息都在存储设备上存储。计算机存储器的设计目标归纳为3个问题：容量、速度和价格。从计算机的性能方面来看，存储器的存储容量越大越好，存储器的存储容量大，才能将大型软件一次存入并运行；存储器的速度越快越好，为了达到最佳性能，存储器的存取速度最好能跟得上处理器的速度，这样计算机系统的整体速度才能提高。但是，大的容量和快的速度是建立在高的价格上的。在实际的计算机系统中，存储器的设计往往是综合考虑容量、速度和价格因素给出的折中方案。

#### 1. 存储器的层次结构

基于对存储器的容量、速度和价格的综合考虑，在计算机系统中使用多种不同的存储器，它们一般包括：寄存器、高速缓冲存储器、主存储器（内存）、辅助存储器（外存）。这些存储器大都采用一种层次结构，如图1-3所示。

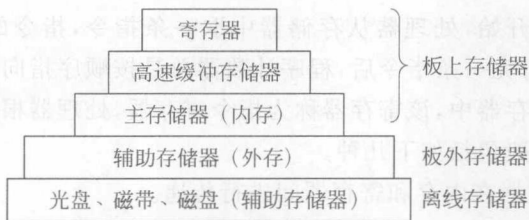


图 1-3 存储器的层次结构

速度最快的寄存器是交换数据最频繁的器件，它和控制器、计算器一起放在处理器的芯片上，以充分提高系统效率。寄存器价格较贵，因此容量较小。

与处理器直接打交道的内存采用了速度较快的半导体存储器，计算机工作时，处理机频繁地与内存交换信息，因此，内存的速度是至关重要的，它直接影响计算机系统的速度。由于半导体存储器价格贵，因此其容量往往做得不是很大。

处理器的速度远远高于主存储器的存取速度，因此使用高速缓存作为主存储器和处理器的寄存器之间的数据移动的缓冲，可以提升系统的性能。

存储容量较大的辅助存储器采用较为便宜的磁盘存储器，但其存取速度较慢。它作为内存存储器的后备设备存在，与处理器交换信息的频率大大降低，其存取速度对计算机系统的性能影响相对较小。由于其价格便宜，可以做成大容量存储设备。

这种层次结构使得各种存储器针对各自的要求，发挥各自的优势，从而达到容量、速度和价格综合指标的最优化。

#### 2. 寄存器

寄存器集成在处理器芯片上，详细参见1.1.2处理器部分。

#### 3. 主存储器（内存）

主存储器又称内存，是计算机内部的主要存储器，内存可分为只读存储器（Read Only



Memory, ROM)和随机存取存储器(Random Access Memory, RAM)。ROM 中的数据是一次性写入的,之后便不能修改,常用于存放一些基本的程序和数据,如启动程序、初始化数据等。ROM 中的数据可永久保存,不会因为电源关闭而丢失。RAM 的内容则可以快速地进行读写,用于操作系统、用户程序以及数据的存放。目前使用的 RAM 通常是易失性的,即计算机关机时, RAM 中存储的内容会丢失。

#### 4. 高速缓冲存储器

高速缓冲存储器简称高速缓存(Caching)。在计算机运行过程中,处理器需要从主存储器中取出指令和数据,并将处理的结果写入主存储器。长期以来,处理器的速度远远高于主存储器的存取速度,使得处理器经常要停下来等待从主存储器中读取数据,这就影响了计算机系统的性能。如果采用和处理器中寄存器同样的技术构造主存储器,可以大大提高主存储器的存取速度,使其和处理器的速度匹配,但是这样做成本太高。高速缓存是既可提升系统性能又不增加许多成本的重要机制。

高速缓存用于主存储器和处理器中寄存器之间的数据移动,如图 1-4 所示。由主存储器到高速缓存的数据移动为块移动,由高速缓存到处理器寄存器的数据移动为字移动。

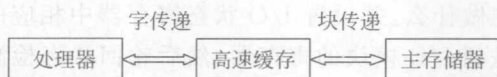


图 1-4 高速缓存技术原理

高速缓存采用存取速度接近于处理器速度的存储器,但其容量较主存储器要小很多。高速缓存器包含主存储器中部分数据的副本。当处理器要读取数据时,首先检查这个数据是否在高速缓存器中。如果在,则从高速缓存器中读取该数据;如果不在,则按照一定数量大小将该数据连同其附近的数据由主存储器送入高速缓存器,处理器再从高速缓存器中读取该数据。按照局部性原理,设计适当的高速缓存器的大小及良好的管理策略,可使得高速缓存在提升系统性能上达到良好效果。但是,如果处理不当,高速缓存非但不能提升系统性能,甚至还会使系统性能下降。

#### 5. 磁盘

寄存器、高速缓存、主存储器都具有一类特点:速度较快、容量较小、价格较贵和数据易失性等。作为计算机系统中常见的辅助存储器的磁盘,则具有速度较慢、容量较大、价格较便宜和数据非易失性等特点。磁盘采用磁介质存储数据,在电源关闭后,磁盘中的数据仍然保存。因此,磁盘常用于存储大量的程序和数据,作为主存储器的后备,如在内存扩充技术中,将磁盘作为虚拟内存使用。除磁盘外,光盘、磁带作为辅助存储器也在广泛使用,它们都具有容量较大、价格较便宜和数据非易失性等特点。

### 1.1.4 I/O 结构

处理器通过总线和 I/O 模块与外部设备打交道。处理器将信息通过总线传递到 I/O 模块, I/O 模块再传递到对应的外部设备。