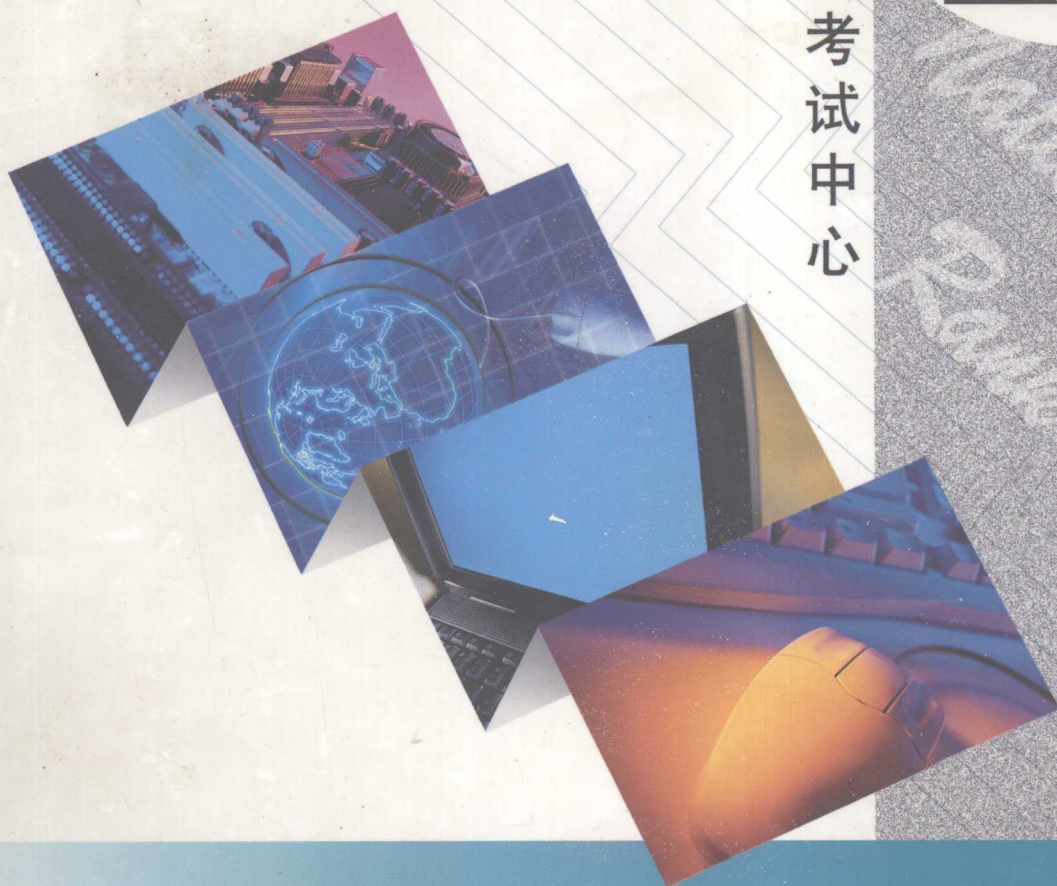


教育部
考试中心



全国计算机等级考试

四级考试指导 (2004 年版)



高等教育出版社

全国计算机等级考试

四级考试指导 (2004 年版)

教育部考试中心

主要内容

图书在版编目(CIP)数据

全国计算机等级考试四级考试指导(2004年版) / 教育部考试中心编. — 北京: 高等教育出版社, 2004.12
ISBN 7-04-013598-1

I. ①全... Ⅱ. 教... Ⅲ. 电子计算机—水平—考试—教材
IV. TP3

中国版本图书馆CIP数据核字(2004)第092022号

封面设计

责任编辑 李 颖
封面设计 李 颖
版式设计 李 颖
http://www.hep.com.cn

书号	定价	本	册	印	发
7-04-013598-1	13.00元	880×1230	1/16	北京中教馆印务有限公司	北京中教馆印务有限公司

2004年11月第1次印刷
2004年11月第1次印刷
5x 88.00元

高等教育出版社

地址: 北京市西城区德胜门内大街2号
电话: 010-62050668
邮编: 100088

内容提要

由教育部考试中心推出的全国计算机等级考试是一项客观、公正、科学的专门测试应试人员的计算机应用知识与技能的全国范围等级考试。它面向社会,服务于社会,为计算机信息技术的推广普及、为用人单位录用和考核工作人员发挥其应有的作用。

本书是按照教育部考试中心颁布的《全国计算机等级考试考试大纲(2004年版)》的有关要求,在1996年出版的《考试指导(四级)》基础上,由教育部考试中心组织、在全国计算机等级考试委员会指导下,邀请有关专家编写的,内容包括计算机系统组成及工作原理、数据结构与算法、离散数学、操作系统、软件工程、数据库、计算机体系结构、计算机网络与通讯,共8大部分内容。

图书在版编目(CIP)数据

全国计算机等级考试四级考试指导:2004年版/教育部考试中心. —北京:高等教育出版社,2004.8

ISBN 7-04-015268-1

I.全... II.教... III.电子计算机—水平考试—自学参考资料 IV.TP3

中国版本图书馆CIP数据核字(2004)第068055号

策划编辑 田晓兰 肖子东 责任编辑 肖子东 封面设计 于文燕 责任绘图 朱 静
版式设计 马静如 责任校对 康晓燕 责任印制 韩 刚

出版发行 高等教育出版社
社 址 北京市西城区德外大街4号
邮政编码 100011
总 机 010-58581000

经 销 新华书店北京发行所
印 刷 北京市鑫霸印务有限公司

开 本 880×1230 1/16
印 张 39.75
字 数 1 240 000

购书热线 010-64054588
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>

版 次 2004年9月第1版
印 次 2004年9月第1次印刷
定 价 69.00元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号:15268-00

大力推行全国计算机等级考试 为发展知识经济、信息产业和培养计算机 专门人才作出贡献

(代序)

中国科学院院士 北京大学信息与工程科学学部主任
全国计算机等级考试委员会主任委员 杨芙清

当今,人类正在步入一个以智力资源的占有和配置,知识生产、分配和使用为最重要因素的知识经济时代,也就是小平同志提出的“科学技术是第一生产力”的时代。科教是经济发展的基础,知识是人类创新的源泉。基础研究的科学发现、应用研究的原理探索和开发研究的技术发明,三者之间的联系愈来愈紧密,转换周期日趋缩短。世界各国的竞争已成为以经济为基础、以科技(特别是高科技)为先导的综合国力的竞争。

在高科技中,信息科学技术是知识高度密集、学科高度综合、具有科学与技术融合特征的学科。它直接渗透到经济、文化和社会的各个领域,迅速改变着人们的观念、生活和社会的结构,是当代发展知识经济的支柱之一。

在信息科学技术中,微电子是基础,计算机硬件及通信设施是载体,计算机软件是核心。软件是人类知识的固化,是知识经济的基本表征,软件已成为信息时代的新型“物理设施”。人类抽象的经验、知识正逐步由软件予以精确地体现。在信息时代,软件是信息化的核心,国民经济和国防建设、社会发展、人民生活都离不开软件,软件无处不在。软件产业是增长最快的朝阳产业,是具有高额附加值、高投入/高产出、无污染、低能耗的绿色产业。软件产业的发展将推动知识经济的进程,促进从注重量的增长向注重质的提高方向发展,是典型的知识型产业。软件产业是关系到国家经济安全和文化安全,体现国家综合实力,决定 21 世纪国际竞争地位的战略性的产业。

为了适应知识经济发展的需要,大力推动信息产业的发展,需要在全民中普及计算机的基本知识,广开渠道,培养和造就一批又一批能熟练运用计算机和软件技术的各行各业的专门人才。

1994 年,原国家教委(现教育部)推出了全国计算机等级考试,它是一种重视应试人员对计算机和软件的实际掌握能力的考试。它不限制报考人员的学历背景,任何年龄段的人员都可以报考。这就为培养各行各业计算机的应用人才,开辟了一条广阔的道路。

1994 年是推出计算机等级考试的第一年,当年参加考试的有 1 万余人;而 2003 年,报考人数已达 251 万余人。截止至 2003 年底,全国计算机等级考试共开考 18 次,考生人数累计超过 1 050 万人,其中有 350 多万考生获得了不同级别的证书。

事实说明,鼓励社会各阶层的人士通过各种途径掌握计算机应用技术,并运用等级考试对他们的才干予以认真的、有权威性的认证,是一种较好的人才培养的有效途径,是比较符合我国具体情况的。等级考试也为用人单位录用和考核人员提供了一种测评手段。从有关公司对等级考试所作的社会抽样调查结果看,不

论是管理人员还是应试人员,对该项考试的内容和形式都给予了充分肯定的评价。

计算机等级考试所取得的良好效果,也同全国各有关单位专家们在等级考试的大纲编写、试题设计、阅卷评分及效果分析等多项工作中所付出的大量心血和辛勤的劳动密切相关,他们为这项工作的顺利开展作出了重要的贡献。

计算机与软件技术是一项日新月异的高新技术。计算机等级考试的考试内容和考核形式也将跟随新技术的发展不断创新,需要及时推出新的考试科目,及时修订旧科目的考试大纲、教材,对考试命题以及上机考试系统进行改革和完善,从而使等级考试更能反映当前的应用实际,使培养计算机应用人才的基础工作更健康地向前发展。本书的修订,正是为了满足新时期新技术发展的需要,满足社会主义市场经济人才培养的需要。

从面临知识经济的机遇与挑战这样一个社会大环境的背景出发,考察全国计算机等级考试,就会看到,这一举措是符合知识经济和发展信息产业的方向的,是值得大力推行的。

我们相信,在21世纪知识经济和加快发展信息产业的形势下,在教育部考试中心的精心组织领导下,在全国各有关专家们的大力配合下,全国计算机等级考试一定会以更新的面貌出现,从而为我国培养计算机应用专门人才的宏大事业作出更多的贡献。

2003年12月

前 言

随着计算机技术的发展与普及,计算机已经成为各行各业最基本的工具之一,而且正迅速进入千家万户,有人还把它称为“第二文化”。因此,许多单位把具有一定计算机应用知识与能力作为录用、考核工作人员的重要条件。

为满足社会需要,于1994年教育部考试中心就推出了全国计算机等级考试,一方面向社会普及和推广计算机应用知识、技能,另一方面为用人单位录用和考核工作人员提供一个客观的依据。从1994年至今,全国计算机等级考试开考十年来,得到了社会各方面的大力支持,考试规模持续增长,赢得了良好的信誉,为社会主义市场经济建设发挥了重要作用。

该项考试面向社会,服务于社会,不以评价教学为目的,考核内容不是按照学校教学要求设定,而是根据使用计算机的不同应用层面要求,以应用能力为主,划分等级,分别考核,为人员择业、人才流动提供其计算机应用知识与能力水平之证明。

教育部考试中心在设计实施这项考试中,得到了全国著名计算机专家的热情支持与大力帮助。专家们对考试大纲、内容及考试的组织与实施都进行了认真的研究,精心的设计。为便于考生学习和考试,依照《考试大纲(2004年版)》的要求,教育部考试中心组织编写了四级考试指导(2004年版)。

参加本书编写的人员是:蒋本珊(第1章),唐发根(第2章),耿素云(第3章),孟祥武(第4章),金茂忠(第5章),周志逵(第6章),邝坚(第7章),严伟(第8章),由陈向群统稿。由于编写时间仓促,教材涉及面广,疏误之处必然存在,望读者提出宝贵意见,以便修订时改进。

编 者

2004年6月

目 录

第一章 计算机系统组成及工作原理	1	2.2.2 线性表的顺序存储结构	46
1.1 基本概念	1	2.2.3 线性表的链式存储结构	48
1.1.1 计算机系统的硬件组成	1	2.3 堆栈与队列	55
1.1.2 计算机系统的层次结构	3	2.3.1 堆栈的基本概念与操作	55
1.1.3 计算机的主要性能指标	3	2.3.2 堆栈的存储结构	55
1.2 运算方法基础与运算器	4	2.3.3 队列的基本概念与操作	56
1.2.1 数值数据在计算机中的表示	4	2.3.4 队列的存储结构	57
1.2.2 非数值数据在计算机中的表示	9	2.4 特殊矩阵的压缩存储	57
1.2.3 数据校验码	11	2.4.1 对称矩阵的压缩存储	58
1.2.4 基本的算术运算	13	2.4.2 对角矩阵的压缩存储	58
1.2.5 基本的逻辑运算	15	2.4.3 稀疏矩阵的三元组表存储	58
1.2.6 运算器的基本组成	16	2.5 树与二叉树	59
1.3 指令系统及控制器	18	2.5.1 树的基本概念	60
1.3.1 指令格式和指令的寻址方式	18	2.5.2 二叉树的基本概念	61
1.3.2 指令类型	21	2.5.3 二叉树的存储结构	62
1.3.3 控制器的组成	22	2.5.4 二叉树的遍历	64
1.3.4 CPU 的总体结构	23	2.5.5 二叉排序树	66
1.3.5 中断系统	24	2.6 图	68
1.4 存储系统	25	2.6.1 图的基本概念	68
1.4.1 存储系统原理	25	2.6.2 图的存储	69
1.4.2 半导体随机存储器和只读存储器	26	2.6.3 图的遍历	71
1.4.3 主存储器的组成与读写操作	28	2.6.4 最小生成树	72
1.4.4 外存储器的工作原理	31	2.6.5 最短路径	73
1.5 输入/输出设备与输入/输出系统	33	2.6.6 拓扑排序	74
1.5.1 常见输入/输出设备及 I/O 接口	33	2.6.7 关键路径	74
1.5.2 程序查询方式	35	2.7 文件及其查找	75
1.5.3 程序中断方式	35	2.7.1 文件的基本概念	75
1.5.4 DMA 方式	37	2.7.2 顺序文件及其查找方法	76
1.5.5 通道方式	39	2.7.3 索引文件及其查找方法	79
1.5.6 典型总线	41	2.7.4 B-树和B+树	80
练习题	42	2.7.5 散列表的查找	81
参考文献	42	2.8 内排序	84
第二章 数据结构与算法	43	2.8.1 排序的基本概念	84
2.1 基本概念	43	2.8.2 插入排序法	85
2.1.1 数据结构的基本概念	43	2.8.3 选择排序法	86
2.1.2 算法及算法分析	44	2.8.4 起泡排序法	87
2.2 线性表	45	2.8.5 Shell 排序法	88
2.2.1 线性表的定义及其基本操作	45	2.8.6 快速排序法	89
		2.8.7 堆排序法	91

2.8.8 二路归并排序法	92	4.1.2 操作系统的历史	173
练习题	94	4.1.3 操作系统的功能	173
第三章 离散数学	95	4.1.4 操作系统的类型	174
3.1 数理逻辑	95	4.1.5 研究操作系统的方法	176
3.1.1 命题逻辑基本概念	95	4.1.6 操作系统的硬件环境	177
3.1.2 命题公式及其分类	98	4.2 进程管理	180
3.1.3 命题逻辑等值演算	100	4.2.1 多道程序设计	180
3.1.4 主析取范式与主合取范式	102	4.2.2 进程	181
3.1.5 命题逻辑的推理理论	106	4.2.3 进程间的通信	184
3.1.6 谓词逻辑基本概念	110	4.2.4 进程控制	187
3.1.7 谓词公式的解释与分类	111	4.2.5 进程调度	187
3.1.8 谓词逻辑的等值演算与前束范式	113	4.2.6 死锁	189
3.1.9 谓词逻辑推理理论	115	4.3 作业管理	191
3.2 集合论	118	4.3.1 操作系统与用户的接口	191
3.2.1 集合的基本概念	118	4.3.2 作业管理概述	192
3.2.2 集合的基本运算	119	4.3.3 批处理方式下的作业管理	193
3.2.3 包含排斥原理	120	4.4 存储管理	196
3.2.4 有序对、卡氏积和二元关系	122	4.4.1 存储管理概述	197
3.2.5 关系的运算及性质	123	4.4.2 内存资源管理	198
3.2.6 复合关系	125	4.4.3 分区存储管理	199
3.2.7 关系的闭包	126	4.4.4 页式存储管理	200
3.2.8 等价关系与偏序关系	127	4.4.5 段式存储管理	202
3.2.9 函数	130	4.4.6 段页式存储管理	204
3.2.10 自然数和集合的基数	132	4.4.7 虚拟存储管理	206
3.3 代数结构	135	4.4.8 虚拟存储管理的性能问题	209
3.3.1 代数运算及其性质	135	4.4.9 交换技术	209
3.3.2 代数系统的同态与同构	137	4.5 文件管理	210
3.3.3 商代数与积代数	138	4.5.1 文件与文件系统	210
3.3.4 半群与群	139	4.5.2 文件结构和存取方式	211
3.3.5 循环群与置换群	141	4.5.3 文件目录	214
3.3.6 群的陪集分解、不变子群和商群	142	4.5.4 文件存储空间的管理	217
3.3.7 环和域	144	4.5.5 文件存取控制	218
3.3.8 格与布尔代数	145	4.5.6 文件的使用	220
3.4 图论	147	4.5.7 文件系统的安全	221
3.4.1 无向图与有向图	147	4.6 设备管理	221
3.4.2 通路、回路和图的连通性	152	4.6.1 设备管理概述	221
3.4.3 图的矩阵表示	154	4.6.2 通道技术	223
3.4.4 二部图	156	4.6.3 缓冲技术	224
3.4.5 欧拉图与哈密顿图	157	4.6.4 设备分配	225
3.4.6 平面图	158	4.6.5 设备处理	227
3.4.7 无向树及生成树	160	4.6.6 磁盘调度	227
3.4.8 根树及其应用	161	4.7 UNIX 操作系统的特点与使用	228
练习题	163	4.7.1 UNIX 操作系统概述	228
第四章 操作系统	172	4.7.2 UNIX 的结构	229
4.1 操作系统基本概念	172	4.7.3 UNIX 基本操作	229
4.1.1 操作系统概念	172	4.7.4 UNIX 文件系统	230

4.7.5	UNIX 的基本命令	230	5.7.2	软件开发工具	310
4.7.6	UNIX 系统 Shell	232	5.8	软件质量保证与软件质量度量	312
4.8	Linux 系统的特点及使用	235	5.8.1	软件质量概念	312
4.8.1	Linux 概述	235	5.8.2	软件质量保证(SQA)	314
4.8.2	Linux 结构	235	5.8.3	软件质量度量与评价	316
4.8.3	基本命令	236	5.8.4	软件技术评审	318
4.8.4	Shell	238	5.8.5	软件可靠性	320
4.9	Windows 操作系统的特点及使用	240	5.9	软件管理	321
4.9.1	Windows 概述	240	5.9.1	软件管理职能	321
4.9.2	Windows 操作系统进程管理	241	5.9.2	软件项目的组织与计划	322
4.9.3	Windows 操作系统存储管理	242	5.9.3	风险分析	324
4.9.4	Windows 操作系统文件系统	243	5.9.4	项目进度与跟踪	326
4.9.5	Windows 操作系统 I/O 设备管理	244	5.9.5	软件配置管理	327
练习题		245	5.9.6	软件过程成熟度模型(CMM)	328
第五章	软件工程	246	5.9.7	软件工程标准化与软件文档	329
5.1	软件工程的基本概念	246	5.9.8	软件产权保护	332
5.1.1	软件与软件危机	246	练习题		333
5.1.2	软件工程定义	246	参考文献		334
5.1.3	软件生存周期	247	第六章	数据库	335
5.1.4	软件过程模型	248	6.1	数据库的基本概念	335
5.2	结构化分析与设计	249	6.1.1	信息处理与数据库	335
5.2.1	问题定义、可行性研究和软件计划	249	6.1.2	数据模型	336
5.2.2	软件需求分析	250	6.1.3	数据库系统结构	341
5.2.3	软件体系结构设计	258	6.1.4	数据库系统组成	342
5.2.4	概要设计和详细设计	263	6.2	关系数据库	343
5.2.5	用户界面设计	276	6.2.1	关系模型的基本概念	343
5.3	原型化开发方法	281	6.2.2	关系代数	347
5.3.1	原型化开发方法的基本原理	281	6.2.3	元组关系演算	351
5.3.2	原型化开发模型和开发过程	282	6.2.4	域关系演算	353
5.4	面向对象分析与设计	284	6.2.5	查询优化	355
5.4.1	面向对象的基本概念	284	6.3	关系数据库标准语言 SQL	359
5.4.2	面向对象分析	285	6.3.1	SQL 语言的特点	359
5.4.3	统一建模语言(UML)	289	6.3.2	SQL 的系统结构	360
5.4.4	软件复用	291	6.3.3	SQL 的数据定义	361
5.5	软件测试	294	6.3.4	SQL 的数据操纵	363
5.5.1	软件测试的基本概念	294	6.3.5	视图	367
5.5.2	软件测试方法	296	6.3.6	数据控制	369
5.5.3	测试用例设计	297	6.3.7	嵌入式 SQL	370
5.5.4	软件测试过程	300	6.4	关系数据库设计理论	373
5.6	软件维护	304	6.4.1	关系模型的存储异常	373
5.6.1	软件维护的基本概念	304	6.4.2	函数依赖	374
5.6.2	软件维护活动和实施	305	6.4.3	函数依赖公理	376
5.6.3	软件可维护性	307	6.4.4	模式分解	377
5.6.4	软件维护的副作用	308	6.4.5	关系模式的规范化	380
5.7	软件工程环境与工具	309	6.5	数据库保护	384
5.7.1	软件工程环境	309	6.5.1	数据库恢复	384

6.5.2 并发控制	388	7.4.4 阵列处理机	486
6.5.3 完整性	393	7.4.5 多处理机	488
6.5.4 安全性	396	7.4.6 机群系统	491
6.6 数据库设计	399	7.5 系统性能评价	493
6.6.1 数据库设计概述	399	7.5.1 性能评价概念	493
6.6.2 需求分析	402	7.5.2 基准测试程序	494
6.6.3 概念设计	403	练习题	495
6.6.4 逻辑设计	407	参考文献	496
6.6.5 物理设计	409	第八章 计算机网络	497
6.6.6 数据库实施与维护	410	8.1 计算机网络与 Internet	497
6.7 数据库管理系统	410	8.1.1 网络发展	498
6.7.1 数据库管理系统的组成	410	8.1.2 网络协议	500
6.7.2 数据库系统的工作过程	411	8.1.3 网络软件	501
6.7.3 关系数据库管理系统	412	8.1.4 网络硬件	503
6.8 数据库新技术	418	8.1.5 协议层次与参考模型	509
6.8.1 数据库技术的发展	418	8.2 应用层	512
6.8.2 分布式数据库	419	8.2.1 应用层概述	512
6.8.3 并行数据库	422	8.2.2 万维网	514
6.8.4 多媒体数据库	423	8.2.3 文件传送	523
6.8.5 面向对象数据库和对象—关系数据 库	425	8.2.4 电子邮件	526
6.8.6 数据仓库	427	8.2.5 域名系统	530
6.8.7 数据挖掘	429	8.2.6 网络安全	534
6.8.8 Web 数据库	430	8.2.7 网络多媒体技术	536
练习题	431	8.3 传输层	539
第七章 计算机体系结构	433	8.3.1 传输层概述	539
7.1 体系结构的基本概念	433	8.3.2 无连接传输	543
7.1.1 计算机系统的层次结构	433	8.3.3 面向连接传输	544
7.1.2 计算机体系结构的定义	434	8.3.4 Socket 编程	554
7.1.3 计算机体系结构的分类	435	8.4 网络层与路由	564
7.1.4 体系结构发展的影响因素	437	8.4.1 网络层概述	565
7.1.5 体系结构的定量分析	439	8.4.2 路由原理	568
7.2 存储系统	442	8.4.3 Internet 协议	575
7.2.1 存储层次	442	8.4.4 Internet 路由	588
7.2.2 Cache 工作原理	443	8.4.5 服务质量	592
7.2.3 虚存工作原理	449	8.5 链路层与局域网	595
7.3 指令与时间并行性	454	8.5.1 数据链路层概述	595
7.3.1 指令优化策略	454	8.5.2 流量控制	597
7.3.2 流水线技术	456	8.5.3 差错控制	600
7.3.3 RISC	468	8.5.4 PPP: 点对点协议	603
7.4 并行处理技术	472	8.5.5 多路访问协议与 Ethernet	606
7.4.1 并行性概念	472	8.5.6 数据链路层交换	617
7.4.2 超标量与超流水线技术	474	练习题	623
7.4.3 向量处理机	480	参考文献	624

第一章 计算机系统组成及工作原理

1.1 基本概念

1.1.1 计算机系统的硬件组成

一、电子计算机的发展

电子计算机是一种不需要人工直接干预,能够自动、高速、准确地对各种信息进行处理和存储的电子设
备。一个完整的计算机系统包含硬件系统和软件系统两大部分,硬件通常是指一切看得见、摸得到的设备实
体。人们习惯把计算机的硬件发展历史分“代”,其实分代并没有统一的标准。若按计算机所采用的微电子
器件的发展,可以将计算机分成以下几代:

第一代(1946 年~1959 年):电子管计算机。

第二代(1959 年~1964 年):晶体管计算机。

第三代(1964 年~1975 年):小、中规模集成电路(SSI,MSI)计算机。

第四代(1975 年~1990 年):大、超大规模集成电路(LSI,VLSI)计算机。

第五代(1990 年~现在):甚大规模集成电路(ULSI)计算机。

现在世界已进入了计算机时代,计算机的发展趋势正向着“两极”分化。一极是微型机向更微型化、网
络化、高性能、多用途方向发展;另一极则是巨型机向更巨型化、超高速、并行处理、智能化方向发展。

二、计算机的基本组成

计算机的硬件系统由运算器、存储器、控制器、输入设备和输出设备五大基本部件组成。图 1.1 为计
算机最基本的组成框图。图中存储器由主存储器(内存储器)和辅助存储器(外存储器)组成。

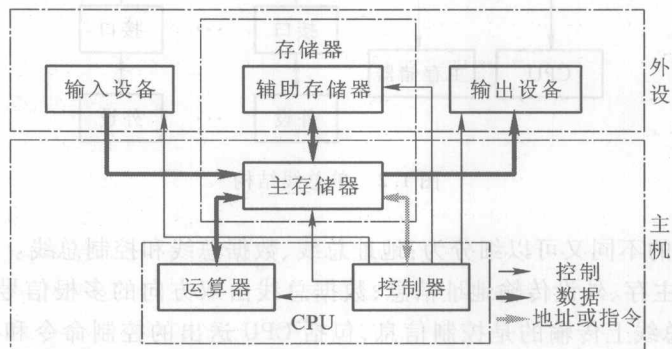


图 1.1 计算机的基本组成

通常将运算器和控制器合称为中央处理器(CPU)。中央处理器和主存储器一起组成主机部分。除去主
机以外的硬件装置(如输入设备、输出设备、辅助存储器等)称为外围设备或外部设备。

三、计算机主要部件的功能

1. 输入设备

输入设备的任务是把人们编好的程序和原始数据送到计算机中去,并且将它们转换成计算机所能识别和接受的信息方式。

2. 输出设备

输出设备的任务是将计算机的处理结果以人或其他设备所能接受的形式送出计算机。

3. 存储器

存储器是用来存放程序和数据的部件,规模较大的存储器往往分成若干级,称为存储系统。主存储器可由 CPU 直接访问,存取速度快但容量较小,一般用来存放当前正在执行的程序和数据。辅助存储器设置主机外部,存储容量大但存取速度较慢,一般用来存放暂时不参与运行的程序和数据,这些程序和数据在需要时可传送到主存。有些计算机在主存和 CPU 间又增设了一级 Cache(高速缓冲存储器)。Cache 的存取速度比主存更快,但容量更小,用来存放当前最急需处理的程序和数据,以便快速地向 CPU 提供指令和数据。

4. 运算器

运算器是对信息进行处理和运算的部件,通常的运算包括算术运算和逻辑运算。运算器的核心被称为算术逻辑运算部件(ALU),运算器中还有若干个通用寄存器或累加寄存器,用来暂存操作数和存放运算结果。

5. 控制器

控制器是整个计算机的指挥中心,它的主要功能是按照人们预先确定的操作步骤,控制整个计算机的各部件有条不紊地自动工作。控制器中包括一些专用的寄存器。

四、计算机的总线结构

将计算机的各大基本部件,按某种方式连接起来就构成了计算机的硬件系统。目前许多计算机(主要指小型、微型计算机)的各大基本部件之间是用总线(Bus)连接起来的。

所谓总线是一组能为多个部件服务的公共信息传送线路,它能分时地发送与接收各部件的信息。计算机中采用总线结构,既可以大大减少信息传送线的数目,又可以提高计算机扩充主存及外部设备的灵活性。

最简单的总线结构是单总线结构,如图 1.2 所示。各大部件都连接在单一的一组总线上,故将这个单总线称为系统总线。CPU 与主存、CPU 与外设之间可以直接进行信息交换,主存与外设、外设与外设之间也可以直接进行信息交换,而无须经过 CPU 的干预。

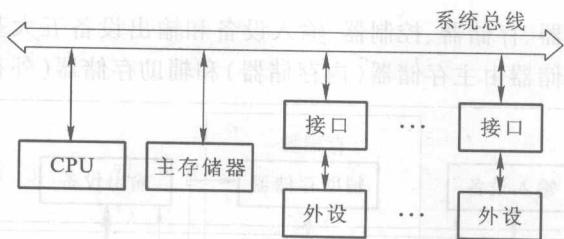


图 1.2 单总线结构

系统总线按传送信息的不同又可以细分为:地址总线、数据总线和控制总线。地址总线由单方向的多根信号线组成,用于 CPU 向主存、外设传输地址信息;数据总线由双方向的多根信号线组成,CPU 可以沿这些线读入或送出数据;控制总线上传输的是控制信息,包括 CPU 送出的控制命令和主存/外设反馈给 CPU 的状态信号。

1.1.2 计算机系统的层次结构

计算机系统是一个硬件与软件组成的综合体,我们可以把它看成是按功能划分的多级层次结构,如图1.3所示。

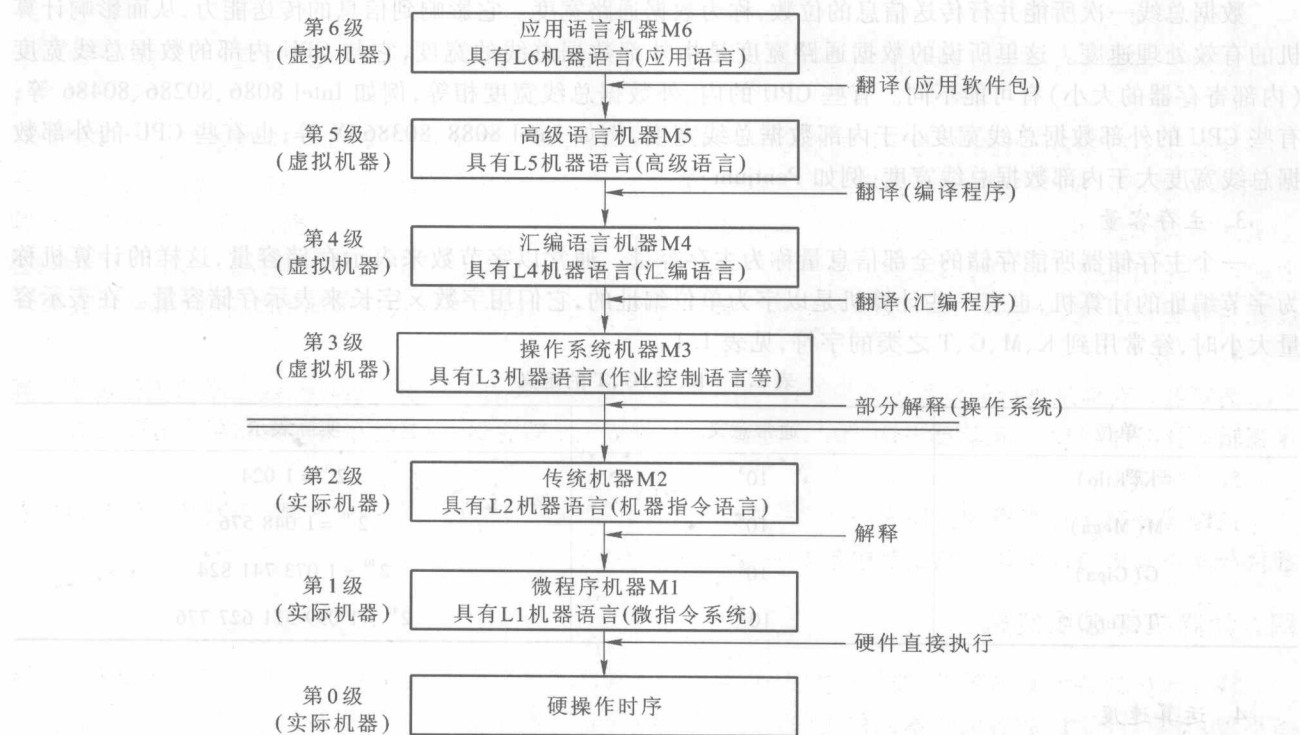


图 1.3 计算机系统的多层次结构

第0级为硬件组成的实体。

第1级是微程序级。这级的机器语言是微指令集,用微指令编写的微程序一般是直接由硬件执行的。

第2级是传统机器级。该级的机器语言是该机的指令集,用机器指令编写的程序可以由微程序进行解释。

第3级是操作系统级。从操作系统的基本功能来看,一方面它要直接管理传统机器中的软硬件资源,另一方面它又是传统机器的延伸。

第4级是汇编语言级。这级的机器语言是汇编语言,完成汇编语言翻译的程序叫做汇编程序。

第5级是高级语言级。这级的机器语言就是各种高级语言,通常用编译程序来完成高级语言的翻译工作。

第6级是应用语言级。这一级是为了使计算机满足某种用途而专门设计的,因此这一级语言就是各种面向问题的应用语言。

把计算机系统按功能划分成多级层次结构,有利于正确理解计算机系统的工作过程,明确软件、硬件在计算机系统中的地位 and 作用。

1.1.3 计算机的主要性能指标

1. 机器字长

机器字长是指参与运算的数的基本位数,它是由加法器、寄存器的位数决定的,所以机器字长一般等于内部寄存器的大小。字长标志着精度,字长越长,计算的精度就越高。

在计算机中为了更灵活地表达和处理信息,又以字节(Byte)为基本单位,用大写字母 B 表示。一个字节等于 8 位二进制位(bit)。

2. 数据通路宽度

数据总线一次所能并行传送信息的位数,称为数据通路宽度。它影响到信息的传送能力,从而影响计算机的有效处理速度。这里所说的数据通路宽度是指外部数据总线的宽度,它与 CPU 内部的数据总线宽度(内部寄存器的大小)有可能不同。有些 CPU 的内、外数据总线宽度相等,例如 Intel 8086、80286、80486 等;有些 CPU 的外部数据总线宽度小于内部数据总线宽度,例如 Intel 8088、80386SX 等;也有些 CPU 的外部数据总线宽度大于内部数据总线宽度,例如 Pentium 等。

3. 主存容量

一个主存储器所能存储的全部信息量称为主存容量。通常以字节数来表示存储容量,这样的计算机称为字节编址的计算机,也有一些计算机是以字为单位编址的,它们用字数 $\times$ 字长来表示存储容量。在表示容量大小时,经常用到 K、M、G、T 之类的字符,见表 1.1。

表 1.1 K、M、G、T 的定义

单位	通常意义	实际表示
K(Kilo)	10^3	$2^{10} = 1\ 024$
M(Mega)	10^6	$2^{20} = 1\ 048\ 576$
G(Giga)	10^9	$2^{30} = 1\ 073\ 741\ 824$
T(Tera)	10^{12}	$2^{40} = 1\ 099\ 511\ 627\ 776$

4. 运算速度

计算机的运算速度与许多因素有关,如机器的主频、所执行的操作、主存本身的速度等等。通常以 MIPS 和 MFLOPS 作为计量单位来衡量运算速度。

MIPS(Million Instruction Per Second)表示每秒执行多少百万条指令。对于一个给定的程序,MIPS 定义为

$$\text{MIPS} = \frac{\text{指令条数}}{\text{执行时间} \times 10^6}$$

MFLOPS(Million Floating-point Operation Per Second)表示每秒执行多少百万次浮点运算。对于一个给定的程序,MFLOPS 定义为

$$\text{MFLOPS} = \frac{\text{浮点操作次数}}{\text{执行时间} \times 10^6}$$

1.2 运算方法基础与运算器

1.2.1 数值数据在计算机中的表示

一、计算机中常用的进位计数制

1. 二进计数制

二进制是一种最简单的进位计数制,它只有两个不同的数码:“0”和“1”,逢二进一。

计算机采用二进制主要有以下几点原因:

- (1) 容易实现,可以用具有两种不同稳定状态的物理器件来表示“0”或“1”。
- (2) 运算规则简单,操作实现简便。
- (3) 数据的存储、传送和处理可靠。

(4) 能够方便地使用逻辑代数这一有力工具来分析、综合与设计计算机的逻辑线路。

但是,用二进制来表示一个数所使用的位数要比用十进制数表示时所使用的位数长得多,书写起来不方便。为此,在计算机中通常还使用八进制和十六进制。

2. 八进制计数制与十六进制计数制

八进制数的每个数位可取 8 个不同的数码(0~7)中的任意一个,基数 r 为 8,逢八进一。因为 $r=8=2^3$,所以只要把二进制中的 3 位数码编为一组就是一位八进制数码。

十六进制数的每个数位可取 16 个不同的数码和符号(0~9, A~F)中的任意一个,其中 A~F 表示十进制数值 10~15,基数 r 为 16,逢十六进一。因为 $r=16=2^4$,故 4 位二进制数码与 1 位十六进制数码相对应。

在计算机中常用后缀字母 B 表示二进制数, O 表示八进制数, H 表示十六进制数。十进制数的后缀是字母 D,在书写时可以省略。例如:有四个数分别为 375D、101B、76O、A17H,从后缀字母就可以知道它们分别是十进制数、二进制数、八进制数和十六进制数。

由二进制数转换成八进制或十六进制数是很容易的。对于一个二进制混合数(既包含小数又包含整数),在转换时应以小数点为界。其整数部分,从小数点开始往左数,将一串二进制数分为三位一组或四位一组,在数的最左边可根据需要加“0”;对于小数部分,从小数点开始往右数,也将一串二进制数分为三位一组或四位一组,在数的最右边也可根据需要加“0”。最终使总的位数成为三或四的倍数,然后分别用相应的八进制或十六进制数取代之。例如:

$$\begin{aligned} 11011.1010011011\text{B} &= \underline{011} \ \underline{011} \ . \ \underline{101} \ \underline{001} \ \underline{101} \ \underline{100}\text{B} \\ &= 33.5154\text{O} \ (\text{八进制}) \\ &= \underline{0001} \ \underline{1011} \ . \ \underline{1010} \ \underline{0110} \ \underline{1100}\text{B} \\ &= 1\text{B}.A6\text{CH} \ (\text{十六进制}) \end{aligned}$$

反之,由八进制或十六进制数转换成二进制数也是很容易的。

八进制和十六进制数之间的转换,可以通过二进制作作为中间桥梁。例如:

$$\begin{aligned} 3.145\text{O} &= \underline{011} \ . \ \underline{001} \ \underline{100} \ \underline{101}\text{B} \\ &= \underline{0011} \ . \ \underline{0011} \ \underline{0010} \ \underline{1000}\text{B} \\ &= 3.328\text{H} \end{aligned}$$

二、十进制与任意进制相互转换

1. 十进制整数转换为任意进制整数(除基取余法)

十进制整数转换为任意进制整数的规则是:除(以)基取余;先余为低,后余为高。

对于十进制整数转换为二进制整数来说,就是用 2 连续去除十进制整数,直至商等于 0 为止;然后逆序的排列每次的余数(先取得的余数为低位),便得到与该十进制数相对应的二进制数各位的数值。例如:

2	57	...	余数 1	(低位)
2	28	...	余数 0	
2	14	...	余数 0	
2	7	...	余数 1	
2	3	...	余数 1	
2	1	...	余数 1	(高位)
0				

所以, $57D = 111001B$

用与此类似的方法也可以完成十进制整数 $\rightarrow$ 八进制整数、十进制整数 $\rightarrow$ 十六进制整数的转换,不同的是此时应当分别用 8 或 16 连续去除十进制整数。

2. 十进制小数转换为任意进制小数(乘基取整法)

十进制小数转换为任意进制小数的规则是:乘(以)基取整;先整为高,后整为低。

对于十进制小数转换为二进制小数来说,就是连续用 2 去乘十进制小数,直至乘积的小数部分等于 0 或者达到精度要求为止;然后顺序的排列每次乘积的整数部分(先取得的整数为高位),便得到与该十进制数相对应的二进制数各位的数值。例如:

$$\begin{array}{rcl}
 0.6875 & & \\
 \times 2 & & \\
 \hline
 1.3750 \dots & \text{整数 1} & \text{(高位)} \\
 0.3750 & & \\
 \times 2 & & \\
 \hline
 0.7500 \dots & \text{整数 0} & \\
 0.7500 & & \\
 \times 2 & & \\
 \hline
 1.5000 \dots & \text{整数 1} & \\
 0.5000 & & \\
 \times 2 & & \\
 \hline
 1.0000 \dots & \text{整数 1} & \text{(低位)}
 \end{array}$$

所以, $0.6875D = 0.1011B$

用与此类似的方法也可以完成十进制小数 $\rightarrow$ 八进制小数、十进制小数 $\rightarrow$ 十六进制小数的转换,不同的是此时应当分别用 8 或 16 连续地去乘十进制小数。

对于十进制数到二进制数的转换,也可以不采用上述规则,只要基本上记住二进制数每一位上的权值,就很容易将这个二进制数写出来。例如:

$$\begin{aligned}
 327.6875D &= 256 + 64 + 4 + 2 + 1 + 0.5 + 0.125 + 0.0625 \\
 &= 2^8 + 2^6 + 2^2 + 2^1 + 2^0 + 2^{-1} + 2^{-3} + 2^{-4} \\
 &= 101000111.1011B
 \end{aligned}$$

三、数的机器码表示

1. 无符号数和带符号数

所谓无符号数,就是整个机器字长的全部二进制位均表示数值位(没有符号位)。若机器字长为 $n+1$ 位,无符号数的表示范围是 $0 \sim (2^{n+1} - 1)$;若字长为 8 位,则无符号数的表示范围为 $0 \sim 255$ 。

所谓带符号数,就是用“+”、“-”号加绝对值来表示数值的大小,这种形式表示的数在计算机技术中称为“真值”。“+”或“-”号在计算机中必须数码化,通常约定二进制数的最高位为符号位,“0”表示“+”号,“1”表示“-”号,以这种形式表示的数在计算机中称为机器数。常见的机器数有原码、反码、补码等三种不同的形式。(下面各例中假设字长为 8 位,1 位符号位,7 位数值位,若真值的位数不足,应补够相应的位数。)

2. 原码

原码是一种最简单的机器数表示法,其最高位为符号位,数值部分与真值相同。

例如, $X = +13 = +1101B$ $[X]_{\text{原}} = 00001101$

$X = -13 = -1101B$ $[X]_{\text{原}} = 10001101$

在原码表示中,真值 0 有两种不同的表示形式:

$[+0]_{\text{原}} = 00000000$

$[-0]_{\text{原}} = 10000000$

原码表示的优点是直观易懂,但实现加、减运算的规则较复杂。

3. 补码

补码的符号位表示与原码相同,其数值部分的表示与数的正负有关:对于正数,数值部分与真值形式相同;对于负数,将真值的数值部分按位取反,且在最低位上加 1。

例如: $X = +13 = +1101\text{B}$ $[X]_{\text{补}} = 00001101$

$X = -13 = -1101\text{B}$ $[X]_{\text{补}} = 11110011$

在补码表示中,真值 0 的表示形式是惟一的:

$[+0]_{\text{补}} = [-0]_{\text{补}} = 00000000$

补码表示可以简化加、减法运算的规则。

4. 反码

反码与补码有许多类似之处,对于正数,数值部分与真值形式相同;对于负数,将真值的数值部分按位取反。

例如: $X = +13 = +1101\text{B}$ $[X]_{\text{反}} = 00001101$

$X = -13 = -1101\text{B}$ $[X]_{\text{反}} = 11110010$

在反码表示中,真值 0 也有两种不同的表示形式:

$[+0]_{\text{反}} = 00000000$

$[-0]_{\text{反}} = 11111111$

5. 三种机器数的比较

三种机器数既有共同点,又有各自不同的性质,主要区别有以下几点:

(1) 对于正数它们都等于真值本身,而对于负数各有不同的表示。

(2) 最高位都表示符号位,补码和反码的符号位可作为数值位的一部分看待,和数值位一起参加运算;但原码的符号位不允许和数值位同等看待,必须分开进行处理。

(3) 对于真值 0,原码和反码各有两种不同的表示形式,而补码只有惟一的一种表示形式。

(4) 原码、反码表示的正、负数范围相对零来说是对称的;但补码负数表示范围较正数表示范围宽,能多表示一个最负的数(绝对值最大的负数),其值等于 -2^n (纯整数)或 -1 (纯小数)。

例如:表 1.2 列出 ± 0 、 ± 57 、 ± 127 和 -128 的 8 位二进制原码、反码和补码并将补码用十六进制的表示。

表 1.2 有关数据的原码、反码和补码

真值	原码(B)	反码(B)	补码(B)	补码(H)
+127	01111111	01111111	01111111	7F
+57	00111001	00111001	00111001	39
+0	00000000	00000000	00000000	00
-0	10000000	11111111	00000000	00
-57	10111001	11000110	11000111	C7
-127	11111111	10000000	10000001	81
-128	—	—	10000000	80

四、定点数和浮点数

1. 定点数

在定点表示法中,所有数据的小数点位置固定不变。定点数表示简单直观,但数的表示范围小,运算过程容易产生溢出。通常有两种定点数:定点小数和定点整数。

定点小数即纯小数,小数点的位置固定在最高有效数位之前、符号位之后,记做 $X_s.X_1X_2\cdots X_n$ 。