

面向21世纪高等院校计算机教材系列



微型计算机原理 及应用技术

● 朱金钧 麻新旗 等编著

 机械工业出版社
China Machine Press



面向 21 世纪高等院校计算机教材系列

微型计算机原理 及应用技术

朱金钧 麻新旗 等编著



机械工业出版社

本书以 Intel-86 系列微处理器为背景,从微处理器的角度介绍了 Intel-86 系列微处理器的结构、工作原理、指令系统及汇编语言、程序设计等内容;从微型机系统组成的角度介绍了存储器结构、中断系统和接口技术;从应用角度介绍了典型的微型机系统及工业 PC 机系统,并引入适量的可直接引用的编程实例。采用软硬件结合的方法,全面介绍了微型计算机系统的组成原理及应用。

本书内容比较丰富,注重系统性、先进性和实用性的结合,可作为高等学校计算机及电类相关专业课程教材或技术人员培训教材,也适合广大从事微型机科研、生产、教学和应用开发的科技人员用于自学或参考。

图书在版编目 (CIP) 数据

微型计算机原理及应用技术/朱金钧等编著. —北京:机械工业出版社, 2002. 8

面向 21 世纪高等院校计算机教材系列

ISBN 7-111-10801-9

I. 微... II. 朱... III. 微型计算机—高等学校—教材 IV. TP36

中国版本图书馆 CIP 数据核字 (2002) 第 062860 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 时 静

责任印制: 闫 焱

北京京丰印刷厂印刷·新华书店北京发行所发行

2002 年 9 月第 1 版·第 1 次印刷

787mm×1092mm 1/16·19.5 印张·479 千字

0 001—8 000 册

定价: 27.00 元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话 (010) 68993821、68326677-2527

封面无防伪标均为盗版

出版说明

随着计算机技术的飞速发展,计算机在经济与社会发展中的地位日益重要。在高等院校的培养目标中,都将计算机知识与应用能力作为其重要的组成部分。为此,国家教育部根据高等院校非计算机专业的计算机培养目标,提出了“计算机文化基础”、“计算机技术基础”和“计算机应用基础”三个层次教育的课程体系。根据计算机科学发展迅速的学科特点,计算机教育应面向社会,面向潮流,与社会接轨,与时代同行。随着计算机软硬件的不断更新换代,计算机教学内容也必须随之不断更新。

为满足高等院校计算机教材的需求,机械工业出版社聘请了清华大学、北京交通大学、北京邮电大学等院校的老师,经过反复研讨,结合当前计算机发展需要和编者长期从事计算机教学的经验精心编写出“面向 21 世纪高等院校计算机教材”。

本套教材理论教学和实践教学相结合,图文并茂、内容实用、内容层次分明、讲解清晰、系统全面,其中溶入了教师大量的教学经验,是各类高等院校、高等职业学校及相关院校的最佳教材,也可作为培训班和自学使用。

前 言

由于微型计算机具有体积小、重量轻、耗电少、可靠性高、结构灵活和价格低廉等优点,从 20 世纪 70 年代至今其得到了飞速发展。字长从 4 位、8 位、16 位、32 位到 64 位,工作主频从 1MHz 开始提高到现在已经上市的 2.4GHz。当前,一台 64 位微型计算机的功能已经赶上或超过了 20 世纪 70 年代中型机的功能。1998 年底 IBM 公司公布了其研究成果“太平洋蓝”计算机,其运算速度达到每秒 3.9 万亿次,而今 Intel 公司已把 2.4GHz 主频的 Pentium4 投放到商品市场上出售,再加上多处理机系统计算机技术越来越完善,微型计算机以其无与伦比的优点在计算机领域中占据的份额越来越大。目前,它已深入到工业、农业、国防等国民经济的各个领域,同时也广泛进入机关、学校和我们的家庭。

根据多年的微机原理课的教学实践及科学研究,我们认为微机原理及应用课程教材应具有以下特点:(1)重点突出、层次清楚、由浅入深、循序渐进。例如在第 1 章中特意加进了计算机组织与结构的核心内容,为后续章节的学习奠定了基础,有利于读者跨进微型计算机世界的大门。全书共分为 7 章,每一章均可作为一个相对独立的层次,从 CPU 到指令系统,从指令系统到程序设计,从接口到计算机的应用,引导读者由浅入深的进行学习。(2)软硬件结合。计算机科学本身就是一个软硬件结合的科学,微机原理作为一门计算机专业基础课更是如此。书中第 3、4 章为汇编语言程序设计,就是为了利用各种各样的接口通过编程达到应用的目的。(3)应具有先进性。时至今日,仅仅讲述以 Intel8086 为处理器的微型计算机是远远不够的,书中介绍了 80386、80486 乃至 Pentium 原理方面的内容,还加进了当前微型计算机的高速缓存等内容。(4)同时将书中不能详述的内容收入附录,有利于读者查询,灵活应用所学知识。

本书适用于高等院校计算机及相关专业“微机原理及应用”课程教材,也可作为广大从事微机开发及应用的工程技术人员的参考书。本书在力求达到上述特点外,在文字叙述上尽量做到通俗易懂,语言精炼。

全书由朱金钧教授和麻新旗副教授统稿。第 1、2 章由麻新旗执笔,第 3、4 章由张会莉执笔,第 5 章和附录由周万珍执笔,第 6 章由朱金钧执笔,第 7 章由薛增涛执笔。

本书在编写出版过程中得到王俊社、刘希岭、侯洁、贾克、汤景军、张智慧、宋东宇等同志的大力支持,在此深表感谢。

由于编者水平有限,书中难免有错误和不妥之处,敬请读者提出宝贵意见。

编者

目 录

出版说明

前言

第 1 章 计算机基础	1
1.1 引言	1
1.1.1 计算机发展概况	1
1.1.2 计算机的主要特点	2
1.1.3 计算机的分类和应用	3
1.2 计算机中数的表示方法	3
1.2.1 进位计数制	3
1.2.2 计算机中的编码系统	6
1.2.3 带符号数的表示	7
1.2.4 数的定点和浮点表示	9
1.3 计算机系统的组成及其工作原理	12
1.3.1 计算机的硬件系统	12
1.3.2 计算机的软件系统	16
1.3.3 计算机的主要技术指标	18
1.4 习题与思考	19
第 2 章 80X86 微处理器	21
2.1 8086 微处理器(CPU)	21
2.1.1 8086CPU 的结构	21
2.1.2 8086 CPU 引脚的功能	25
2.2 8086 系统的存储器及 I/O 组织	28
2.2.1 8086 系统的存储器组织	28
2.2.2 8086 系统的 I/O 组织	30
2.3 8086 系统配置	31
2.3.1 最小模式和最大模式的概念	31
2.3.2 最小模式系统	31
2.3.3 最大模式系统	34
2.4 8086 CPU 的操作时序	37
2.4.1 8086 CPU 的复位操作时序	38
2.4.2 最小模式下的总线读周期	38
2.4.3 最小模式下的总线写周期	40
2.4.4 最大模式下的总线读周期	41
2.4.5 最大模式下的总线写周期	42

2.4.6	最小模式下的总线保持(即总线请求/响应)周期	43
2.4.7	最大模式下的总线请求/允许周期	44
2.5	80386 微处理器	45
2.5.1	寄存器组	45
2.5.2	80386 的地址转换	49
2.5.3	80386 的存储管理与保护功能	49
2.5.4	80386 的流水线结构	51
2.6	80486 微处理器	53
2.6.1	80486 微处理器的特点	53
2.6.2	80486 微处理器的内部结构	54
2.7	Pentium 微处理器	58
2.7.1	Pentium 体系结构的特点	58
2.7.2	相对 486 体系结构的增强点	61
2.8	习题与思考	61
第 3 章	指令系统	63
3.1	寻址方式	63
3.1.1	操作数类型	63
3.1.2	寻址方式	64
3.2	8086 指令系统	69
3.2.1	数据传送指令	69
3.2.2	算术运算指令	74
3.2.3	逻辑运算指令	82
3.2.4	移位指令	83
3.2.5	字符串操作指令	86
3.2.6	转移指令	90
3.2.7	处理器控制指令	95
3.2.8	输入输出指令	96
3.2.9	中断指令	97
3.3	80X86 的增强型指令	97
3.3.1	80286 增强和新增指令	97
3.3.2	80386 增强和新增指令	99
3.3.3	80486 新增指令	101
3.4	习题与思考	101
第 4 章	汇编语言程序设计	104
4.1	宏汇编语言的基本语法	104
4.1.1	常数、变量和标号	104
4.1.2	表达式与运算符	105
4.2	伪指令	109
4.2.1	伪指令语句的格式	109

4.2.2 常用伪指令	110
4.3 宏指令	118
4.4 汇编语言程序的结构	120
4.4.1 汇编语言源程序的结构	120
4.4.2 汇编语言和 DOS 操作系统之间的接口	121
4.5 DOS 系统功能调用	122
4.6 汇编语言程序设计	125
4.6.1 程序设计步骤	125
4.6.2 汇编语言程序设计方法	126
4.6.3 程序设计举例	140
4.7 习题与思考	151
第 5 章 存储器	155
5.1 概述	155
5.1.1 存储系统的层次结构	155
5.1.2 存储器的分类	157
5.1.3 存储器的基本组成:	158
5.1.4 存储器的技术指标	159
5.2 半导体读写存储器	160
5.2.1 静态 RAM	160
5.2.2 动态 RAM	163
5.2.3 存储器的工作时序	166
5.3 半导体只读存储器	168
5.3.1 掩膜式只读存储器 ROM	168
5.3.2 可编程的只读存储器 PROM	169
5.3.3 可编程、可擦除的只读存储器——EPROM	169
5.4 存储器与 CPU 的连接	170
5.4.1 存储器与 CPU 连接时考虑的问题	171
5.4.2 常用的译码电路	171
5.4.3 存储器连接举例	172
5.5 习题与思考	177
第 6 章 输入/输出和中断	178
6.1 外设接口的一般结构	178
6.1.1 数据信息	178
6.1.2 状态信息	179
6.1.3 控制信息	179
6.2 CPU 与外设交换数据的方式	179
6.2.1 程序控制传递方式	179
6.2.2 DMA(直接存储器存取)传递方式	182
6.3 中断	183

6.3.1	概述	183
6.3.2	中断过程与中断管理	184
6.4	8086/8088 的中断系统	187
6.4.1	中断结构	187
6.4.2	内部中断——软中断	188
6.4.3	外部中断——硬中断	189
6.4.4	各类中断的优先权及中断响应	190
6.5	8259A 可编程中断控制器	190
6.5.1	8259A 的功能、结构及工作原理	190
6.5.2	8259A 的编程	193
6.5.3	8259A 的工作方式	197
6.5.4	由多片 8259A 组成的主从式中断系统	200
6.5.5	8259A 的编程实例	200
6.6	8237DMA 控制器	205
6.6.1	主要功能	205
6.6.2	8237A 的结构和工作原理	205
6.6.3	8237A 的编程和应用举例	213
6.7	习题与思考	216
第 7 章	接口技术	218
7.1	概述	218
7.1.1	接口的功能	218
7.1.2	接口与系统的连接	219
7.2	并行通信和并行接口芯片	220
7.2.1	并行通信	220
7.2.2	8255A 可编程并行接口芯片	221
7.3	串行通信和串行接口芯片	242
7.3.1	串行通信的基本概念	243
7.3.2	8251A 可编程串行通信接口	247
7.4	计数器/定时器接口电路	260
7.4.1	可编程计数器/定时器的工作原理	260
7.4.2	8253 可编程计数器/定时器	262
7.5	模拟通道接口	273
7.5.1	概述	273
7.5.2	数/模(D/A)转换器	274
7.5.3	模/数(A/D)转换器	278
7.6	习题与思考	289
附录 A	BIOS 功能调用	292
附录 B	MC-DOS(INT 21H)功能调用	296
附录 C	ASCII 码编码表	301

第 1 章 计算机基础

电子数字计算机是 20 世纪最重大的科技成就之一。自 1946 年第一台电子计算机问世以来,计算机技术得到迅速发展,并已广泛应用于工农业生产、科学研究、国防及人们日常工作和生活的各个领域。伴随人类进入 21 世纪,以高科技为支撑的信息化社会已经到来,以“信息”为主导的新兴产业正在全球经济领域掀起一场空前的革命,“知识”是这场革命的直接推动力,而计算机及其应用技术则是知识经济的基础,掌握计算机知识和应用能力的多少已成为衡量现代人文化水平的一个重要标志。随着信息化时代的到来,计算机技术的进一步发展和应用必将对社会发展和人类文明产生更大的促进作用,对社会政治、经济、文化和人类生活的各个方面也将产生更深远的影响。

1.1 引言

1.1.1 计算机发展概况

“计算”是人类生活中的一项重要活动。人类祖先在史前时期就知道用石块和贝壳计数。随着历史的发展,人类创造了简单的计算工具,如我国在唐宋时期就开始使用算盘。经过长期艰苦的努力和探索,科学家们发明了机械式计算器、继电器式计算机,在 1946 年终于研制成功第一台电子数字计算机 ENIAC。

在推动计算机发展的诸多因素中,电子器件的发展是最为活跃的。因此,人们常常把计算机的发展以电子器件为标志划分为:

第一代(1945~1958年):电子管计算机。采用水银延迟线作为内存,磁鼓作为外存。体积大、耗电多、运算速度慢。最初只能使用二进制表示的机器语言,到 50 年代中期才出现汇编语言。这个时期,计算机主要用于科学计算和军事方面,应用很不普遍。

第二代(1958~1964年):晶体管计算机。内存主要采用磁芯,外存大量采用磁盘,输入输出设备有了较大改进。体积显著减小、可靠性提高、运算速度可达每秒百万次。软件方面出现了高级程序设计语言和编译系统。计算机开始广泛应用于以管理为目的的信息处理。

第三代(1964~1971年):集成电路计算机。主要采用中、小规模集成电路,运算速度达每秒千万次,可靠性大大提高,体积进一步缩小,价格大大降低。软件方面进步很大,有了操作系统,开展了计算机语言的标准化工作并提出了结构化程序设计方法,出现了计算机网络。计算机应用开始向社会化发展,其应用领域和普及程度迅速扩大。

第四代(1971年至今):大规模集成电路计算机。大规模集成电路的出现使计算机发生了巨大的变化,特别是出现了微处理器,从而推出了微型计算机。微型计算机的出现和发展是计算机发展史上的重大事件,使得计算机在存储容量、运算速度、可靠性和性能价格比等方面都比上一代计算机有了较大突破。各种系统软件、应用软件大量推出,功能配置空前完善,充分发挥了计算机的功能,把计算机的发展和应用带入了一个全新时代。计算机已经应用到几乎

所有的领域,成为人类社会活动中不可缺少的工具。

微型计算机的发展过程,也就是微处理器的发展过程,自1971年第一个微处理器出现以来,微处理器的发展已经经历了4代,目前正处在第5代微处理器发展阶段,每一代的性能都提高了近一个数量级。几乎每两年就有一个质的变化,目前仍在向多功能、多媒体方向发展。

1971~1973年为第一代微处理器,代表产品为Intel4004、8008。前者为4位机,后者为8位机。集成度约为2000等效晶体管/片,时钟频率为1MHz,指令周期为20 μ s。

1973~1975年为第二代微处理器,代表产品有Intel8080、M6800,字长为8位,集成度为5000管/片,时钟频率为2MHz,指令周期为2 μ s。

1975~1977年为第三代微处理器,代表产品有Intel8085、Z80、M6802等,字长为8位,集成度为1万管/片,时钟频率为2.5~5MHz,指令周期为1 μ s。

1978~1980年为第四代微处理器,代表产品有Intel8086、M6809、Z8000等,字长为16位,集成度约为3万管/片,时钟频率为5MHz,指令周期小于0.5 μ s。

1980年之后为第五代微处理器,代表产品有80286、Motorola68010等,字长为16位,集成度达10万管/片,时钟频率为10MHz,指令周期约为0.2 μ s。1983年之后又出现了Intel80386、Motorola68020等微处理器,字长为32位,时钟频率为16MHz以上,集成度高达15~50万管/片,指令周期为0.1 μ s。其后又出现了Intel80486、Pentium系列,集成度达数百万管/片以上,时钟频率高达上千兆赫兹,指令周期只有几个~几十个毫微秒。在这些处理器的芯片上已经包含大容量的高速缓冲存储器(CACHE),原来属于大型机的存储管理技术已经移植到芯片上。

从计算机发展来看,目前的趋势是向两极化(应用于国防尖端等大型领域的巨型机和应用用于其他更广泛领域的微型机)、网络化、多媒体、智能化的方向发展,其他新型计算机如神经网络计算机、生物计算机、光子计算机等非冯·诺依曼体系结构的计算机则是现代计算机技术研究的另一个焦点。

1.1.2 计算机的主要特点

电子计算机是一种不需人的直接干预就能够高速、自动地进行算术和逻辑运算的电子装置。存储程序与程序控制是计算机的重要工作原则,是它能够高速自动运算的基础。

所谓存储程序,是把计算过程表示为由许多条命令(指令)组成的命令序列(程序),与数据一起预先存入计算机的存储器内。只要发出运行命令,计算机就会按照规定的顺序一条一条地取出指令和执行指令,自动地完成预定的信息处理任务。由于在程序执行过程中不需要人的干预,因此,计算机能自动高速的执行程序。人们可以把任何信息处理任务分解成一系列基本的算术和逻辑操作,并按照执行的先后顺序把它们组成程序,存入计算机并使之执行,因此计算机可以完成信息处理任务,具有很强的通用性。

计算机处理的各种符号,包括数值、文字、符号、图像、声音等,都通过数字化编码技术,用数字量表示,所以计算机运算非常准确。同时,计算机可以对这些数字量进行各种大小关系的比较判断,并根据比较结果决定下一步的处理,这就是“条件转移”概念,即计算机的逻辑性。

综上所述,计算机具有五大特点:自动性、高速性、准确性、逻辑性和通用性。

1.1.3 计算机的分类和应用

电子计算机的分类方法多种多样。从原理上,可分为处理连续变化信号的模拟计算机和处理离散信号的数字计算机;从用途上,可分为专用计算机和通用计算机。现在常说的“计算机”实际上是指通用电子数字计算机,仅从它来看,又可分成巨型机、大型机、小型机和微型机等。微型机按字长又可分为4位、8位、16位、32位和64位机,按结构可分为位片机、单片机、单板机和微机系统等。

计算机的应用形式和应用领域千变万化、日新月异,已深入到人类社会生活的各个领域,大到国防技术、航空航天技术、核能技术,中到管理信息系统、科学研究、工业设计和仿真、生产过程控制、多媒体与信息高速公路技术、文化教育、医疗等,小到智能仪表、家用电器等,无一不是计算机信息处理与控制的应用领域。按照计算机应用的性质和形式,可分为数值计算、数据处理(包括办公自动化、数据库应用系统)、生产自动化(包括过程控制、计算机辅助系统如计算机辅助设计CAD、辅助制造CAM、辅助测试CAT、辅助工程CAE等)、计算机模拟、人工智能、计算机网络应用、远程教育等。

1.2 计算机中数的表示方法

计算机的基本功能是对数的运算和处理。在计算机中,通过数字化编码技术,对所表示的数值、文字、符号及控制信息等进行数字编码,这种数字化表示方法不仅要适合于人的自然习惯,同时要满足机器中所用器件、线路的工作状态以及数据可靠传输与易于校验纠错等方面的要求。一个具有两种不同的稳定状态且能相互转换的器件,就可以用来表示一位二进制数,由于表示二进制的器件易于制造且工作可靠,并且二进制数的运算规则也最简单,因此目前计算机中均采用二进制数来表示各种信息及进行信息处理。

1.2.1 进位计数制

按进位的方法进行计数称为进位计数制。数是客观事物的量在人大脑中的反映,可用不同的数制来量度。同一个量用不同的数制量度的结果不同。在日常生活中,我们最熟悉、最常用的是十进制、七进制(星期)、十二进制和六十进制(时间)等。在计算机中,常采用二进制和十六进制。

一个R进制数具有以下主要特点:

- (1) 具有R个不同的数字符号:0、1、2、……、R-1。
- (2) 逢R进一。

任一R进制数S可用其若干个数字符号的组合来表示,如 $a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m}$, (这种书写方法称为位置表示法),其中n为小数点前的位数,m为小数点后的位数, a_i 是R进制的一个数字符号,R称为基数。上述R进制数S可用多项式(称为按权展开式)表示为:

$$S = a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m} \\ = a_{n-1} \times R^{n-1} + a_{n-2} \times R^{n-2} + \cdots + a_1 \times R^1 + a_0 \times R^0 + a_{-1} \times R^{-1} + \cdots + a_{-m} \times R^{-m}$$

显然,位置表示法中,每个数码 a_i 所代表的数值等于该数码乘以一个与所在数位有关的常数 R^i ,如 R^1 、 R^0 、 R^{-1} 、 R^{-2} 等,这些常数称为位权,简称“权”。显然,同一个数码所处位置不

同,其权也不同,代表的数值大小也不同,这正是位置表示法的含义。

1. 十进制数

特点:(1) 具有十个不同的数字符号,即 0~9。

(2) 逢十进一。

一个十进制数可以用它的按权展开式表示。例如:

$$(758.75)_{10} = 7 \times 10^2 + 5 \times 10^1 + 8 \times 10^0 + 7 \times 10^{-1} + 5 \times 10^{-2}$$

其中用 $()_{10}$ 形式表示括号中的数是十进制数,其他进制数也用类似的形式表达。

2. 二进制数

特点:(1) 具有两个不同的数字符号,即 0 和 1。

(2) 逢二进一。

一个二进制数可以用它的按权展开式表示。例如:

$$\begin{aligned} (10110.101)_2 &= 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= (22.625)_{10} \end{aligned}$$

3. 十六进制数

特点:(1) 具有十六个不同的数字符号,即 0~9 和 A~F。

(2) 逢十六进一。

一个十六进制数可以用它的按权展开式表示。例如:

$$(1AF.4)_{16} = 1 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 + 4 \times 16^{-1} = (430.25)_{10}$$

表 1-1 给出了上述三种进位计数制的对应关系。

表 1-1 三种数制对照表

十进制	二进制	十六进制	十进制	二进制	十六进制
0	0000	0	9	1001	9
1	0001	1	10	1010	A
2	0010	2	11	1011	B
3	0011	3	12	1100	C
4	0100	4	13	1101	D
5	0101	5	14	1110	E
6	0110	6	15	1111	F
7	0111	7	16	10000	10
8	1000	8	17	10001	11

除了用括号加下标的形式表示不同进制的数据外,程序中通常采用加不同的后缀字母来表示不同进制的数据,规定:二进制数加后缀字母 B,如 100B;十进制数加后缀字母 D 或省略后缀,约定无后缀的数为十进制数,如 100D 或 100;十六进制数加后缀字母 H,如 100H。

4. 各种数制之间的转换

(1) 二进制、十六进制转换成十进制。采用按权展开式计算求和的方法,如上例。

(2) 十进制转换成二进制、十六进制。整数部分采用除基取余法,小数部分采用乘基取整

法。

【例 1-1】十进制数 22.625 转换为二进制数。

整数部分: $2 \overline{)22}$ $2 \overline{)11} \cdots \text{余 } 0 \text{ (低位)}$ $2 \overline{)5} \cdots \text{余 } 1$ $2 \overline{)2} \cdots \text{余 } 1$ $2 \overline{)1} \cdots \text{余 } 0$ $0 \cdots \text{余 } 1 \text{ (高位)}$	小数部分: 0.625 $\times 2$ $\hline \boxed{1}.25 \cdots \text{取整数 } 1 \text{ (高位)}$ $\times 2$ $\hline \boxed{0}.5 \cdots \text{取整数 } 0$ $\times 2$ $\hline \boxed{1}.0 \cdots \text{取整数 } 0 \text{ (低位)}$
---	---

所以: $(22)_{10} = (10110)_2$
 结果: $(22.625)_{10} = (10110.101)_2$

【例 1-2】十进制数 430.25 转换为十六进制数。

整数部分: $16 \overline{)430}$ $16 \overline{)26} \cdots \text{余 } 14 \rightarrow \text{E (低位)}$ $16 \overline{)1} \cdots \text{余 } 10 \rightarrow \text{A}$ $0 \cdots \text{余 } 1 \text{ (高位)}$	小数部分: 0.25 $\times 16$ $\hline \boxed{4}.0 \cdots \text{取整数 } 4$
---	--

结果: $(430.25)_{10} = (1AE.4)_{16}$

注意: ① 整数部分转换, 每次只求整数商, 将余数作为转换结果的一位, 重复对整数商除基数, 一直除到商为 0 为止。② 小数部分转换, 每次把乘积的整数取走作为转换结果的一位, 对剩下的小数继续进行乘法运算。对某些数可以乘到积的小数为 0 (如上述两例), 这种转换结果是精确的; 对某些数 (如 0.3) 永远不能乘到积的小数为 0, 这时要根据精度要求, 取适当的结果位数即可, 这种转换结果是不精确的。

(3) 二进制与十六进制间的相互转换。由表 1-1 可以看出, 每一位十六进制数对应四位二进制数, 因此, 只要用四位二进制数代替对应的一位十六进制即可完成十六进制到二进制的转换。

例如: 十六进制数 $1 \quad A \quad E \quad . \quad 4$
 $\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 $0001 \quad 1010 \quad 1110 \quad . \quad 0100$

即 $(1AE.4)_{16} = (110101110.01)_2$

若要将二进制数转换为十六进制数, 只要以小数点为分界, 分别向左和向右每四位二进制位分为一组 (若最高位或最低位不够四位则补 0), 对应转换为十六进制数即可。

例如: 二进制数 110101110.01
 \downarrow
 $\underline{0001} \quad \underline{1010} \quad \underline{1110} \quad . \quad \underline{0100}$
 $\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 十六进制数 $1 \quad A \quad E \quad . \quad 4$

即 $(110101110.01)_2 = (1AE.4)_{16}$

5. 二进制数的运算

二进制与十进制的运算方法基本相同,但在二进制运算时,逢二进一,借一当二。

【例 1-3】 $10100 + 1101 = 100001$

$$\begin{array}{r} 10100 \\ + 1101 \\ \hline 100001 \end{array}$$

【例 1-5】 $1101 \times 1011 = 10001111$

$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline 1101 \\ 1101 \\ 0000 \\ + 1101 \\ \hline 10001111 \end{array}$$

【例 1-4】 $100001 - 10100 = 1101$

$$\begin{array}{r} 100001 \\ - 10100 \\ \hline 1101 \end{array}$$

【例 1-6】 $11100 \div 101 = 101 \cdots 11$

$$\begin{array}{r} 101 \cdots \text{商} \\ \overline{)11100} \\ \underline{101} \\ 1000 \\ \underline{101} \\ 11 \cdots \text{余数} \end{array}$$

1.2.2 计算机中的编码系统

在计算机中,数是用二进制表示的。而计算机还应该能够识别和处理各种文字信息,如大小写英文字母、标点符号、运算符号等,这些符号又怎样表示呢? 由于计算机的基本物理器件是具有两个状态的器件,所以各种文字信息也只能用若干位的二进制编码组合来表示。

1. 二进制编码的十进制数(BCD 码——Binary Coded Decimal)

因为二进制数实现容易、可靠,且运算规律简单,所以在计算机内部采用二进制数。但是,二进制数不直观,在计算机进行输入和输出与人交换信息时,通常还是采用十进制数。为了转换直观方便,要将十进制数用二进制编码来表示,这就是 BCD 码。

1 位十进制数需要用 4 位二进制编码,编码的表示方法很多,较常用的是 8421BCD 码,表 1-2 列出了 8421BCD 码和十进制数的对应关系。

BCD 码是十进制数,有 10 个不同的数字符号,且是逢十进位的;但它的每一位是用 4 位二进制编码来表示的,因此称为二进制编码的十进制数。BCD 码比较直观,例如十进制数 65 用 BCD 码书写为 01100101,BCD 码 01001001.0111 表示的十进制数为 49.7。

所以,只要熟悉了 BCD 码的 10 个编码,就可以很容易地实现十进制与 BCD 码之间的转换。

表 1-2 BCD 编码表

十 进 制	8421BCD 码	十 进 制	8421BCD 码
0	0000	6	0110
1	0001	7	0111
2	0010	8	1000
3	0011	9	1001
4	0100	10	0001 0000
5	0101	11	0001 0001

虽然 BCD 码是用二进制编码方式表示的,但它与二进制之间不能直接转换,要用十进制作为中间桥梁,即先将 BCD 码转换为十进制数,然后再转换为二进制数;反之亦然。

2. 字母与字符的编码

如上所述,字母和字符也必须按照特定的规则,用二进制编码才能在机器中表示。编码可以有各种方式,目前微机中最普遍采用的是 ASCII 码(American Standard Code for Information Interchange, 美国标准信息交换码),编码表见附录 C。

ASCII 码采用 7 位二进制编码,故可表示 $2^7 = 128$ 个字符,其中包括数码(0~9),以及英文字母等可打印的字符。从表中可以看到,数码 0~9 相应的用 $(0110000)_2 \sim (0111001)_2$ 来表示。因微型机字长或内存单元通常是 8 位,所以通常把最高位用作奇偶校验位,但在机器中表示时,一般认为是 0,故用一个字节(8 位二进制数)来表示一个字符的 ASCII 码值。于是 0~9 的 ASCII 码为 $(00110000)_2 \sim (00111001)_2$ 即 $(30)_{16} \sim (39)_{16}$, 大写字母 A~Z 的 ASCII 码为 $(41)_{16} \sim (5A)_{16}$, 小写字母 a~z 的 ASCII 码为 $(61)_{16} \sim (7A)_{16}$ 。

另外,在计算机中,汉字编码采用国标码(GB18030-2000),它采用单、双、四字节混合编码,每个字节的最高位为 1,并以此来区分汉字和 ASCII 码。

1.2.3 带符号数的表示

上面提到的二进制数没有涉及符号问题,故是一种无符号数的表示。但是在实际应用中,数会有正负,那么计算机中如何表示数的符号呢?通常规定数据的最高位为符号位,即若是字长为 8 位,则 D_7 位是符号位, $D_6 \sim D_0$ 为数字位。并规定符号位用 0 表示正,用 1 表示负。

如: $X = (00001100)_2$ 则 X 的值为 +12; 而: $Y = (10001100)_2$ 则 Y 的值为 -12。

在计算机中,带符号数有三种表示法——原码、反码和补码。用原码、反码、补码表示的数称为机器数,机器数对应的数学值称为真值。

1. 原码

正数的符号位用 0 表示,负数的符号位用 1 表示,数值位保持不变。这种表示法称为原码。原码的定义为:

若 $X \geq +0$ 则 $[X]_{原} = X$

若 $X \leq -0$ 则 $[X]_{原} = 2^{n-1} - X$ 其中 n 为原码的位数。

例如,设用 8 位表示原码,则 $[+124]_{原} = 01111100$, $[-124]_{原} = 11111100$

其中,最高位 D_7 为符号位,后 7 位 $D_6 \sim D_0$ 是数值位。如果字长是 16 位,则 D_{15} 是符号位, $D_{14} \sim D_0$ 是数值位。在用原码表示时,8 位二进制原码真值范围为 $-127 \sim +127$; 16 位二进制原码真值范围为 $-32767 \sim +32767$ 。8 位二进制原码的表示见表 1-3。原码表示法简单易懂,而且与真值转换方便。但若是两个异号数相加(或两个同号数相减)就需要做减法。为了简化运算器设计(把减法运算转换为加法运算),引入了反码和补码。

2. 反码

反码的定义为:若 $X \geq +0$ 则 $[X]_{反} = X$

若 $X \leq -0$ 则 $[X]_{反} = 2^n + X - 1$ 其中 n 为反码的位数。

显然,正数的反码表示与原码相同,最高位为符号位,其余位为数值位。

例如: $X = +4$, 则 $[X]_{\text{反}} = [X]_{\text{原}} = 00000100$

而负数的反码应当表示为该数的原码除符号位外按位取反。

例如: $X = -4$, 则 $[X]_{\text{原}} = 10000100\text{B}$, $[X]_{\text{反}} = 11111011$

$X = -31$, 则 $[X]_{\text{原}} = 10011111\text{B}$, $[X]_{\text{反}} = 11100000$

负数的反码表示与原码有很大的区别:最高位相同,仍是“1”,但数据位的值完全相反,这一点要特别注意。8位二进制反码的表示见表 1-3,它有以下特点:

(1) “0”有两种表示方法: $[+0]_{\text{反}} = 00000000$, $[-0]_{\text{反}} = 11111111$

(2) 8位二进制反码真值范围为 $-127 \sim +127$; 16位反码真值范围为 $-32767 \sim +32767$ 。

(3) 当一个带符号数用反码表示时,最高位为符号位。若符号位为 0,说明该数是正数,后面 7 位(或 15 位)是其真值;若符号位为 1,说明该数为负数,其真值为后面 7 位(或 15 位)按位取反。例如:已知 $[X]_{\text{反}} = 00000101$, 则 $X = +5$; $[Y]_{\text{反}} = 11111110$, 则 $Y = -1$ 。

反码表示对计算机的结构有特殊要求,现在很少采用。

3. 补码

(1) 模的概念:在钟表上,指针正拨 12 小时或倒拨 12 小时,其时间值是相等的,即在钟表上 $X + 12 = X - 12 \pmod{12}$ 。

对于 n 位二进制计数器,其计数范围为 $0 \sim (2^n - 1)$,在该计数器上加 2^n 或减 2^n 结果是不变的,我们称 2^n 为 n 位计数系统的模。对钟表来说,它的模为 12。

由上述可见,一个计数系统,某数加(或减)其模,结果不变。

(2) 补码的引入:在钟表上,如果现在时间是 6 点整,而钟表却指着 8 点整,快了 2 小时,校准的方法是正拨 10 小时或倒拨 2 小时,结果都正确,即:

$$8 + 10 = 6 \pmod{12} \text{ 顺拨} \qquad 8 - 2 = 6 \pmod{12} \text{ 倒拨}$$

即加 10 和减 2 效果相同,可见减 2 可用加 10 来计算。

在数学上,如 a 和 b 满足: $a \text{ MOD } M = (nM + b) \text{ MOD } M$ (其中 n 为正整数, M 为模),则称 a 和 b 互为补数(以 M 为模)。

在时钟上 $+10$ 与 (-2) , $+7$ 与 (-5) , $+6$ 与 (-6) 对 12 同余或称互为补数。

引入补数的概念后就可以将减法转化为加法来计算。

(3) 补码的求法:对 n 位二进制数,模为 2^n ,则 $[X]_{\text{补}} = (2^n + X) \text{ MOD } 2^n$, i 为正整数。

补码的定义为:若 $X \geq +0$ 则 $[X]_{\text{补}} = X$

若 $X \leq -0$ 则 $[X]_{\text{补}} = 2^n + X$

其中 n 为补码的位数。如果 $X \geq 0$,则 $[X]_{\text{补}} = (2^n + X) \text{ MOD } 2^n = X$,即正数的补码为原正数不变。如果 $X \leq 0$,则 $[X]_{\text{补}} = (2^n + X) \text{ MOD } 2^n = 2^n - 1 + X + 1 = [X]_{\text{反}} + 1$,即负数的补码等于负数的反码加 1,也就是等于负数原码除符号位外求反加 1。下面举例说明补码的求法与应用。

$$[+3]_{\text{补}} = [+3]_{\text{原}} = [+3]_{\text{反}} = 00000011$$

$$[-3]_{\text{补}} = [-3]_{\text{反}} + 1 = 11111100 + 1 = 11111101$$

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

$$[-0]_{\text{补}} = [-0]_{\text{反}} + 1 = 11111111 + 1 = 00000000$$