

微型计算机 原理及应用

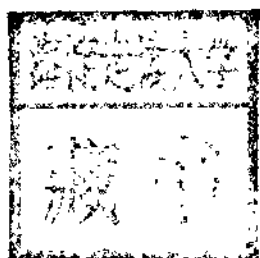
雷友琴 朱金钧 等编著

机械工业出版社

7-1260.1

微型计算机原理及应用

雷友琴 朱金钧 等编著



机械工业出版社

032561

本书以国内外广泛流行的 Intel-86 系列微处理器为背景,从微处理器的角度详细介绍了 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、中断系统及接口技术;从应用的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,可作为高等学校计算机及电类有关专业教材或技术人员培训教材,也适合于广大从事微型机科研、生产、教学和应用开发的科技人员自学或参考。

图书在版编目(CIP)数据

IS-07/23

微型计算机原理及应用/雷友琴 朱金钧等编著. —北京:机械工业出版社, 1996. 6

ISBN 7-111-05063-0

I. 微… II. 雷… III. 微型计算机-基本知识 IV. TP36

中国版本图书馆 CIP 数据核字(95)第 23910 号

出 版 人:马九荣(北京市百万庄南街 1 号 邮政编码 100037)

责任编辑:蒋克 王福俭 版式设计:张世琴

责任校对:张 仕 封面设计:姚 毅

责任印制:卢子祥

北京交通印务实业公司印刷 · 新华书店北京发行所发行

1996 年 6 月第 1 次印刷

787mm×1092mm¹/₁₆ · 20.25 印张 · 493 千字

0 001 3 500 册

定价:26.80 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

前 言

自 1981 年美国 IBM 公司推出第一台个人计算机 IBM PC 之后,PC 机系列及其兼容机得到迅速的发展。由于其软硬件资料公开、升级产品保持与早期产品高度兼容、硬件系统便于进一步开发与升级、软件系统资源十分丰富等一系列原因,PC 机系列成为微型机(下简称微机)市场的主流机种。近几年,人们又将 PC 机用于工业控制,开发出各种 STD 总线工业控制机和工业 PC(IPC)。在这种情况下,仅仅用高级语言编写一些科学计算及管理方面的软件已不能满足应用领域的需要。人们迫切地需要学习、熟悉 PC 机的系统结构及汇编语言编程技术,以便进一步开发各种 PC 机应用系统。基于这种需要,各高等院校也将以 8 位机为主流的微机原理课转向 16 位机。目前虽然存在许多以 PC 为背景的计算机原理和汇编语言编程方面的书,但这些书或者以汇编语言编程为主,或者侧重硬件系统描述,或者侧重于操作系统的应用,或者缺乏新内容的补充。总之,很难选到一本适合于大学本科电类专业微型机原理课教材的书。这就是我们编写《微型计算机原理及应用》一书的初衷。

根据多年微型机原理课的教学经验及科学研究,我们感觉电类专业微机系统原理及应用教材应具有以下特点:1)软硬件兼顾,既具有微机系统组成方面的内容,也具有汇编语言编程方面的内容。通过本书的学习使读者具有组织小型的微机硬件系统的能力和对现有 PC 系统进一步扩展开发的能力。2)教材应具有先进性。时至今日,仍然以讲解 8088 为 CPU 的 PC/XT 机(这种机型实际已很少应用)是远远不够的,必须包括 80386、80486 原理方面的内容。作为工科专业的教材,必须要包含新的工业 PC 机(IPC)方面的内容。3)作为对读者入门的启发与引导,必须有一些实际可引用的应用程序实例,并使读者可从中学习到较多的实用编程技术。4)通过本书的学习,应使读者对 PC-DOS 的全貌及文件系统、系统功能调用有全面的了解,而又不占过大的篇幅叙述 PC-DOS 操作系统使用方法。5)作为工科电类教材,应具有过程控制计算机系统方面的内容,即应具有 A/D、D/A 及典型芯片使用方面的知识。6)对初学者,上机及掌握调试程序的方法是比较困难的。教材应包括这方面的简洁明了的内容。7)应具有必备的附录材料,如指令系统表、常用系统功能调用表等,以方便读者使用。

《微型计算机原理及应用》一书就是按照上述设想编写的,既照顾到系统组成方面的硬件知识,又有较多的汇编语言编程方面的内容。在知识“新”这方面,介绍了高速缓存(Cache)技术;80386、80486 处理器工作原理,存储保护及扩充的指令功能,及一些典型外围芯片及接口技术;适当地介绍了 DOS 的层次结构和文件系统;在应用程序部分综合地介绍了系统调用方法、磁盘读写方法、高级语言与汇编语言接口技术、小汉字库的建立与应用等。并给出上机调试通过的实例程序。每章之后都附有适当的习题。

本书适用于高等学校工科电类及相关专业“微机原理及应用”课教材,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。本书力求做到系统、实用和先进性,在文字叙述上尽量做到通俗易懂、精炼。

本书共 8 章,第 2、3、4、6(部分)章由雷友琴编写,第 7 章由朱金钧编写,第 5 章及附录由李金忠编写,第 1、6(部分)章由周万珍编写。

IV

在本书的编写与出版过程中,得到高树森、麻新旗、钟连荣等同志的关心和支持,在此表示衷心感谢。

由于编者学识所限,虽经反复校审,也难免有错误和不当之处,敬请读者批评指正。

编著者
1995年9月

目 录

前言

绪论	1
----------	---

第 1 章 计算机基础

1 运算基础	4
1.1 计算机中的数制	4
1.2 计算机中的编码系统	6
1.3 带符号数的表示	7
1.4 数的定点和浮点表示	10
2 微型计算机的组成	11
2.1 常用的名词术语	11
2.2 微型计算机的硬件组成	12
2.3 微型计算机的软件系统	12
3 评估计算机的主要技术指标	13
3.1 字长	14
3.2 内存储器容量	14
3.3 指令执行时间	14
3.4 外部设备配备	14

习题与思考	14
-------------	----

第 2 章 80×86 微处理器

1 8086 微处理器(CPU)	16
1.1 8086 CPU 的结构	16
1.2 8086 微处理器的总线周期及引脚功能	20
2 8086 的存储器及 I/O 组织	22
2.1 8086 存储器的组织	22
2.2 8086 I/O 组织	24
3 8086 系统配置	25
3.1 最小模式和最大模式的概念	25
3.2 最小模式系统	25
3.3 最大模式系统	28
4 8086 的操作和时序	30
4.1 8086 系统的复位和起动作	31
4.2 最小模式下的总线读周期	32
4.3 最小模式下的总线写周期	33
4.4 最大模式下的总线读周期	34
4.5 最大模式下的总线写周期	35

4.6 最小模式下的总线保持(即总线请求响应)	36
-------------------------------	----

4.7 最大模式下的总线请求/允许	37
-------------------------	----

5 80386 微处理器

5.1 寄存器组	38
5.2 80386 的地址转换	41
5.3 80386 的存储管理与保护功能	42
5.4 80386 的流水线结构	44

6 80486 微处理器

6.1 80486 微处理器的特点	46
6.2 80486 微处理器的内部结构	46

习题与思考	50
-------------	----

第 3 章 指令系统

1 8086/8088 指令系统的寻址方式	52
1.1 立即寻址	53
1.2 寄存器寻址	53
1.3 直接寻址	53
1.4 寄存器间接寻址	53
1.5 变址寻址	54
1.6 基址加变址寻址	54
2 8086/8088 指令格式	56
3 8086/8088 指令系统	58
3.1 数据传送指令	58
3.2 算术运算类指令	61
3.3 逻辑运算指令	68
3.4 串操作指令	71
3.5 转移类指令	75
3.6 处理器控制指令	79
4 80×86 指令系统	80
4.1 80286 增强与增加的指令	81
4.2 80386/80486 增强与增加指令	82

习题与思考	84
-------------	----

第 4 章 汇编语言程序设计

1 汇编语言源程序的结构	86
2 汇编语言语句	87
2.1 语句格式	87

2.2 伪指令语句	89	4.2 PROM	132
3 宏指令语句	93	4.3 EPROM	132
3.1 宏定义及其调用	93	4.4 E ² PROM	134
3.2 带参数的宏	94	5 存储器芯片的选择	134
3.3 取消宏指令名的伪指令 PURGE	95	5.1 容量	134
3.4 重复伪指令 REPT	95	5.2 速度	134
4 系统功能调用	95	5.3 价格	134
4.1 键盘输入单字符——1号系统功能调用	96	6 存储器子系统与CPU主系统的连接	135
4.2 键盘输入字符串——0AH号系统功能调用	96	6.1 EPROM、RAM子系统与CPU主系统的连接	135
4.3 输出单字符——2号系统功能调用	97	6.2 8086CPU的最小模式与静态RAM的连接	135
4.4 输出字符串——9号系统功能调用	97	7 存储体系结构	137
4.5 返回操作系统——4CH号系统功能调用	97	7.1 存储器的多体结构	137
5 汇编语言程序设计方法	97	7.2 虚拟存储器	139
5.1 应用软件设计方法与程序的基本结构	97	7.3 高速缓冲存储器(Cache)	142
5.2 简单程序设计	99	习题与思考	144
5.3 分支程序设计	99	第6章 输入/输出和中断	145
5.4 循环程序设计	103	1 外设接口的一般结构	145
5.5 子程序设计	110	1.1 数据信息	145
6 汇编语言上机操作	117	1.2 状态信息	145
6.1 编辑源程序	117	1.3 控制信息	145
6.2 汇编源程序	118	2 CPU与外设交换数据的方式	146
6.3 连接程序	118	2.1 程序控制传递方式	146
6.4 调试程序	119	2.2 DMA(直接存储器存取)传递方式	149
6.5 运行程序	121	3 中断	150
习题与思考	121	3.1 概述	150
第5章 存储器	123	3.2 中断处理过程与中断管理	151
1 存储器概述	123	4 8086/8088的中断系统	154
2 随机存取存储器(RAM)	123	4.1 中断结构	154
2.1 静态RAM	123	4.2 内部中断——软中断	155
2.2 动态RAM	125	4.3 外部中断——硬中断	156
3 存储器的工作时序	129	4.4 各类中断的优先权及中断响应和处理流程	157
3.1 存储器的读周期	129	5 8259A可编程中断控制器	158
3.2 存储器的写周期	130	5.1 8259A的功能、结构及工作原理	158
3.3 8086CPU对存储器的读/写时序	130	5.2 8259A的编程	160
4 只读存储器(ROM)	131	5.3 8259A的工作方式	164
4.1 掩膜式ROM	131	5.4 由多片8259A组成的主从式中断系统	167

5.5 8259A 的编程实例	167	习题与思考	253
6 8237DMA 控制器	171	第 8 章 微型计算机系统	255
6.1 主要功能	171	1 微型计算机的硬件系统	255
6.2 8237 的结构和工作原理	172	1.1 IBM PC/XT 微机系统	256
6.3 8237 的编程和应用举例	180	1.2 IBM PC/AT 系统	258
习题与思考	182	1.3 工业控制 PC 机	261
第 7 章 接口技术	184	2 PC-DOS 操作系统	265
1 综述	184	2.1 PC-DOS 的层次结构	265
1.1 接口的功能	184	2.2 PC-DOS 的启动过程及内存 映象	266
1.2 接口与系统的连接	185	2.3 文件系统	266
2 并行通信和并行接口芯片	186	3 应用程序举例	269
2.1 并行通信	186	3.1 内存驻留及时钟显示程序	269
2.2 8255A 可编程并行接口芯片	187	3.2 C 语言与汇编语言混合编程	273
3 串行通信和串行接口芯片	205	3.3 文件读写、人机联系及屏幕图形显 示应用实例	278
3.1 串行通信的基本概念	206	附录	285
3.2 8251A 可编程串行通信接口	210	附录 A 8086/8088 指令系统	285
4 计数器/定时器接口电路	221	附录 B BIOS 功能调用	302
4.1 可编程计数器/定时器的工 作原理	222	附录 C MS-DOS(INT 21H)功能 调用	306
4.2 8253-PIT 可编程计数器/定 时器	223	附录 D ASCII 码编码表	312
5 模拟通道接口	233	附录 E PC/XT 及 PC/AT 总线	313
5.1 概述	233	参考文献	315
5.2 数/模(D/A)转换器	234		
5.3 模/数(A/D)转换器	242		

绪 论

电子数字计算机是本世纪重大的科技成就之一。自 1946 年第一台电子计算机问世以来,从器件发展的角度来看,计算机经历了电子管、晶体管、中小规模集成电路、超大规模集成电路四代,运算速度由每秒数千次,发展到每秒数亿次;功率消耗从数百千瓦下降到数瓦甚至更低;占地面积由数百平方米发展到台式机,甚至单片机,应用范围由单纯的数字计算发展到数据库管理、工业生产过程控制、智能化仪表及家用电器控制等。就计算机系统本身来看正向巨型机和微型机两个方向发展。尤其微型机(包括单片机)的应用已深入到日常工作和生活的各个领域。下面我们就微型机系统的发展、微型机系统的分类、组成和应用等问题,作概括性的介绍。

1 微型计算机系统的发展

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,目前仍在向超多功能、多媒体方向发展,势头方兴未艾。

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。集成度大约为 2000 等效晶体管/片,时钟频率为 1MHz,指令周期为 $20\mu\text{s}$ 。

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,指令周期约为 $2\mu\text{s}$ 。

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,时钟周期为 2.5~5MHz,指令周期为 $1\mu\text{s}$ 。

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,时钟周期为 5MHz,指令周期小于 $0.5\mu\text{s}$ 。

1980 年之后为第 5 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 $0.2\mu\text{s}$ 。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 16~20MHz,集成度高达 15~50 万管/片,指令周期为 $0.1\mu\text{s}$ 。其后又出现了 Intel80486、80586……,Motorola 68030、68040…集成度高达 100 万管/片,指令周期约为 $0.04\mu\text{s}$ 。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。

2 微型计算机系统的分类

从不同的角度可以将微型计算机进行不同的分类。根据微处理器的字长可将微型机分为:4 位机、8 位机、16 位机、32 位机。从微型机的结构出发,可将微型机分类为位片机(即只有一位微型机)、单片机、单板机、系统微型机。从微型机的用途出发,可将微型机分为通用微型机、工业控制微型机。从操作系统的角度可将微型机分为单用户微型机、多用户微型机等等。下面从微型机的结构出发,介绍各种微型机。

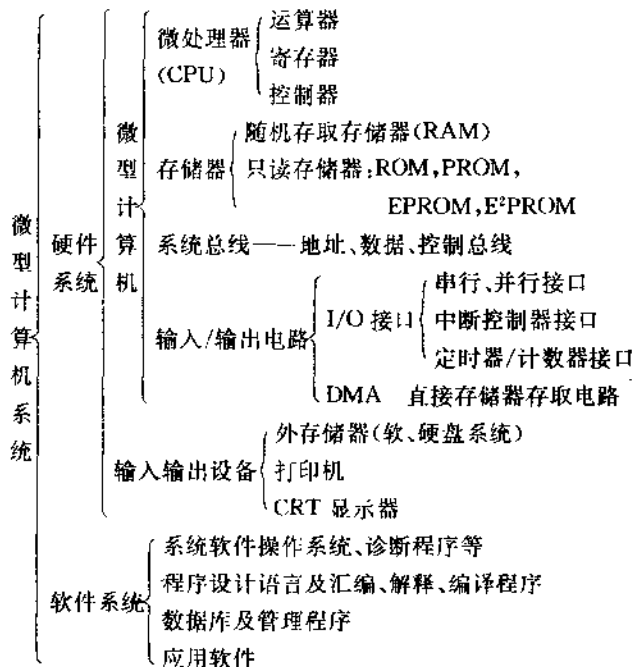
(1) 单片微型计算机(Single Chip Micro Computer) 它把微处理器(简称CPU)、存储器和I/O接口集成在一片超大规模集成电路芯片上,不加扩展或稍加扩展就能构成专用微型机系统,主要用于工业控制或智能仪表。

(2) 单板微型计算机(Single Board Micro Computer) 它把CPU、存储器、I/O接口、键盘及显示电路等组装在一块印制电路板上。由于单板结构,性能价格比较高,故主要用在工业控制及教学、实验系统。

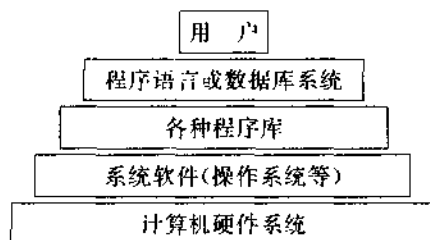
(3) 微型计算机系统 除主机板装有CPU和内存存储器外,大都配有标准键盘、EGA/VGA显示器及驱动器、软硬盘及其驱动接口、打印机接口、通用串并行接口等硬件系统,一般配有较完善的操作系统及功能软件。它可广泛应用于数据库管理系统,网络系统终端,工业控制系统的上位监控微机。

3 微型计算机系统的组成

现代微型机系统一般说来包括硬件系统和软件系统两部分。其组成情况如下表所示。



计算机系统硬件、软件及与用户之间的关系如下图所示。可以看出,可将计算机软件看作用户与计算机硬件系统的接口。软件之间又是逐层依赖的。



4 微型计算机系统的用途

微型计算机的应用几乎深入到生产与生活的各个领域。下面就主要方面简单介绍微型计算机的应用。

(1)数据库及管理系统 数据库及管理系统是微型计算机的主要应用方面,约占微型机总量的80%,小到一种单一的管理软件,如财务管理软件、人事管理软件;大到数据库管理系统,如经营决策系统、市场预测、飞机票预订系统,综合的信息管理系统(MIS)都属于微型机在管理方面的应用。

(2)科学计算 科学计算是微型计算机的基本应用之一。也是数字计算机的最早的应用,如工程设计、结构强度计算、天气预报等,都需要用到大量、高精度的计算。

(3)工业生产过程控制 各种生产过程都可采用计算机进行控制,如石油、化工、发电、电力调度、轧钢、冶炼、机床控制、交通管理等各个部门都可以采用微型机进行控制。可以采用由一台单片机、单板机、微型机构成的单机自动控制,也可采用由微型机构成的多级计算机系统,下位机负担直接数字控制,上位机进行监督控制并兼作管理。在过程控制系统中,还广泛采用集散控制系统,即一种又有集中又有分散的综合控制系统。

(4)通信技术 集电子技术、通信技术及计算机技术成就于一体的计算机网络在通信领域获得广泛的应用。随着多媒体技术及信息高速公路技术的发展,计算机通信将在未来的日子里获得越来越广泛的应用。

(5)智能仪表与家用电器 以单片机为核心构成的各种智能仪表,使仪表的体积、重量、精度得到质的飞跃,今后还会出现更多更新、功能更强的智能仪表。微型机在家用电器中也得到广泛应用。如智能化洗衣机、电冰箱、微波炉等已很普遍。估计微型机除在这些方面继续扩大与完善其应用外,在家庭服务方面也将得到广泛应用。

(6)计算机辅助设计 计算机仿真,计算机辅助设计、计算机辅助制造等是现代工程设计必不可少的设计阶段,它不仅可减少设计人员的劳动量,而且在设计阶段就可模拟、预估产品的结果,大大加快了工程系统的设计制造周期,节省了试制费用,对工程设计将产生越来越深远的影响。

上面只是列举了微型机应用的主要方面。现在可以说微型机已经深入到生产、生活的各个方面。有人把微型机的出现称之为第二次工业革命。目前已经出现了第五代计算机——人工神经网络计算机。随着计算机功能的完善,必将在计算机应用上发生更深远的变革。

为使我国早日赶上工业发达国家,在计算机知识学习、计算机系统的生产与应用方面投入更多的入力、物力,必将产生事半功倍的效果。

第1章 计算机基础

1 运算基础

1.1 计算机中的数制

计算机从诞生的那一天起,主要是作为一种计算工具出现的。所以它的最基本的功能是对数进行加工和处理。数在机器中是以器件的物理状态来表示的,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。因此,二进制数的表示是最简单且可靠的,另外,二进制数的运算规则也最简单。所以,目前计算机中的数全是用二进制数表示的。

1.1.1 二进制数

一个二进制数应具有以下两个基本特点:

1. 具有两个不同的数字符号,即0和1。
2. 逢二进一。

由于是逢二进一的,所以同一字符在不同的位置所表示的值是不同的。例如

1111.1101

小数点左边第一位的“1”代表的值是它本身;小数点左边第二位的“1”是由第一位进上来的,所以它的值为 1×2^1 ;同理,左边第三位的“1”表示的是 1×2^2 ;小数点右面第一位的“1”代表 1×2^{-1} ;右边第二位的“1”表示 1×2^{-2} ……。

一个二进制数的值可以用它的按权展开式来表示,即

$$(1111.1101)_2 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4} = (15.8125)_{10}$$

$$(101.01)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = (5.25)_{10}$$

于是,一个任意的二进制数A可以表示为

$$(A)_2 = a_{n-1} \times 2^{n-1} + a_{n-2} \times 2^{n-2} + \dots + a_1 \times 2^1 + a_0 \times 2^0 + a_{-1} \times 2^{-1} + a_{-2} \times 2^{-2} + \dots + a_{-m} \times$$

$$2^{-m} = \sum_{i=-m}^n a_i \times 2^i$$

其中n为整数部分位数,m为小数部分位数。

$$\text{例如: } 63 = 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

$$\text{则 } (63)_{10} = (111111)_2$$

1.1.2 十六进制数

目前大部分微型机的字长是4的倍数,所以广泛地采用十六进制数来表示。一个十六进制数的特点为

1. 具有十六个数字符号,采用0~9和A~F。这十六个数字符号与十进制数和二进制数之间的关系见表1-1。

表 1-1 二进制、十进制、十六进制数码对照表

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

2. 逢十六进一。

由于十六进制数是逢十六进一,所以同一个数字符号,在不同的数位所代表的值是不同的。小数点左边各位数的权是 16 的正次幂,小数点右边各位数的权是 16 的负次幂。一个十六进制数 B 可以按权展开为

$$(B)_{16} = B_{n-1} \times 16^{n-1} + B_{n-2} \times 16^{n-2} + \cdots + B_1 \times 16^1 + B_0 \times 16^0 + B_{-1} \times 16^{-1} + B_{-2} \times 16^{-2} + \cdots + B_{-m} \times 16^{-m} = \sum_{i=-m}^{n-1} B_i \times 16^i$$

其中, n 是整数部分的位数, m 是小数部分的位数, B_i 的取值范围在 0~9 和 A~F 中。

例如: $(10)_{16} = 1 \times 16^1 + 0 \times 16^0 = (16)_{10}$

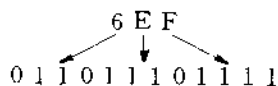
$(1AB)_{16} = 1 \times 16^2 + 10 \times 16^1 + 11 \times 16^0 = (427)_{10}$

但是在机器中数并不是用十六进制表示的,由于一开始提到的理由,在机器中数仍用二进制表示。由于二进制和十六进制之间存在一种特殊的关系,即 $2^4 = 16$,所以一位十六进制数可用四位二进制数表示,它们之间存在着直接且又唯一的对应关系,见表 1-1。

因此,二进制数和十六进制数之间的转换是十分简捷而又方便的。

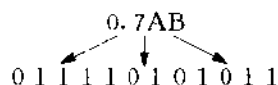
(1) 十六进制数转化为二进制数 不论十六进制的整数或小数,只要把每一位十六进制的数用相应的四位二进制数代替,就可以转换为二进制数。

例如: $(6EF)_{16}$ 可转化为



即 $(6EF)_{16} = (11011101111)_2$

$(0.7AB)_{16}$ 可转化为



即 $(0.7AB)_{16} = (0.011110101011)_2$

(2) 二进制数转换为十六进制数 二进制数的整数部分由小数点向左, 每四位一组, 最后不足四位的前面补 0; 小数部分由小数点向右, 每四位一组, 最后不足四位补 0, 然后把每四位二进制数用相应的十六进制代替, 即可转换为十六进制数。

例如: $(101011011.10111)_2$ 可转化为

0001	0101	1011	•	1011	1000
1	5	B		B	8

即 $(101011011.10111)_2 = (15B.B8)_{16}$

总之, 数在机器中是用二进制表示的。但是, 一个二进制数书写起来太长, 容易出错。而且当前大部分微型机的字长是 4 位、8 位、16 位或 32 位的, 都是四的整数倍, 故在书写时可用十六进制数表示, 1 个字节 (8 位) 可用 2 位十六进制数表示, 2 个字节 (1 个字) 可用 4 位十六进制数表示等, 手写方便且不容易出错。

1.2 计算机中的编码系统

如上节所述, 在计算机中, 数是用二进制表示的。而计算机又应能识别和处理各种字符, 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? 由于计算机中的基本物理器件是具有两个状态的器件, 所以各种字符也只能用若干位的二进制码组合表示。

1.2.1 二进制编码的十进制数

因二进制数实现容易、可靠, 且运算规律简单, 所以在计算机中用二进制数。但是, 二进制数不直观, 于是在计算机的输入和输出时, 通常还是采用十进制数, 不过, 这样的十进制数要用二进制编码表示。

1 位十进制数用 4 位二进制编码来表示, 表示的方法很多, 较常用的是 8421 BCD 码, 表 1-2 列出了部分 8421 BCD 码和十进制数的关系。

表 1-2 BCD 编码表

十进制数	8421 BCD 码	十进制数	8421 BCD 码
0	0000	8	1000
1	0001	9	1001
2	0010	10	0001 0000
3	0011	11	0001 0001
4	0100	12	0001 0010
5	0101	13	0001 0011
6	0110	14	0001 0100
7	0111	15	0001 0101

BCD 码有 10 个不同的字符,且它是逢十进一的,所以 BCD 码是十进制数;但它的每一位是用 4 位二进制编码来表示的,因此称它为十进制的二进制编码。BCD 码是比较直观的。例如:(01001001, 0111)BCD 十进制为 49.7。

所以,只要熟悉了 BCD 的 10 位编码,可很容易地实现十进制与 BCD 码之间的转换。

但是,BCD 码和二进制之间的转换是不直接的,要先经过十进制。即将 BCD 码先转换为十进制码,然后再转换为二进制;反之亦然。

1.2.2 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,目前微机中用的最普通的是采用 ASCII 码(American Standard Cord for Information Interchange,美国标准信息交换码),编码表见附录 D。

ASCII 码是用 7 位二进制编码,故可表示 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,必是相应用 0110000~0111001 来表示的。因微型机通常字长是 8 位,所以通常把 bit₇ 用做奇偶校验位,但在机器中表示时,常认为零,故用一个字节来表示一个 ASCII 码。于是 0~9 的 ASCII 码为 30H~39H;大写字母 A~Z 的 ASCII 码为 41H~5AH,小写字母 a~z 的 ASCII 码为 61H~7AH。

1.3 带符号数的表示

上面提到的二进制数没有提到符号问题,故是一种无符号数的表示。但是在计算机中,数当然会有正负,那么符号该怎样表示呢?通常一个数的最高位为符号位,即若是字长为 8 位,则 D₇ 位是符号位,D₆~D₀ 为数字位。符号位用 0 表示正,用 1 表示负。如

$$X = (00001100)_2 \text{ 其值为 } +12$$

$$\text{而 } Y = (10001100)_2 \text{ 其值为 } -12$$

为了运算方便(把减法变成加法),在机器中数有三种表示法——原码、反码和补码。

1.3.1 原码

按上所述,正数的符号位用 0 表示,负数的符号位用 1 表示。这种表示法称为原码。例如:

$$X = +124$$

$$\text{则 } [X]_{\text{原}} = 01111100$$

$$Y = -124$$

$$\text{则 } [Y]_{\text{原}} = 11111100$$

其中,最高位为符号位,后面 7 位是数值。如字长是 16 位,则 D₁₅ 是符号位,D₁₄~D₀ 是数值位。在用原码表示时,8 位二进制原码表示数的范围为 -127~+127;16 位二进制原码表示的范围为 -32767~+32767,原码表示简单易懂,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)就要做减法。为了把减法运算转换为加法运算就引入了反码和补码。

1.3.2 反码

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。如:

$$X = +4$$

$$\text{则 } [X]_{\text{反}} = [X]_{\text{原}} = 00000100$$

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。

$$\text{例如: } X = -4$$

则 $[X]_{\text{反}} = 11111011$

$X = -31$ 则 $[X]_{\text{反}} = 11100000$

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,这一点要十分注意。

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。

2. 8 位二进制反码表示的数值范围为 $+127 \sim -127$; 16 位反码表示的范围为 $+32767 \sim -32767$ 。

3. 当一个带符号数用反码表示时,最高位为符号位。当符号位为 0 时,即说明该数是正数,后面 7 位(或 15 位)是其数值;当符号位为 1 时,说明该数为负数,其值为后面 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 $Y = -1$

1.3.3 补码

1. 模的概念

在时钟上,正拨 12h,倒拨 12h,结果都不变,即在时钟上 $X+12=X-12(\text{mod } 12)$ 。

假定有 n 位计数器,其计数范围为 $0 \sim (2^n - 1)$,在该计数器上加 2^n 与减 2^n 结果不变,我们称 2^n 为 n 位计数系统的模。

总结上述可见,一个计数系统,某数加(减)其模,结果不变。

2. 补码的引入

在时钟上,如果现在是 6 点正,而时钟却指着 8 点正,快了 2h,校准的方法是正拨 10h,或倒拨 2h;结果都正确,即

$$8+10=6(\text{mod } 12) \text{ 顺拨}$$

$$8-2=6(\text{mod } 12) \text{ 倒拨}$$

即 $+10$ 和 (-2) 效果相同。可见在 -2 可用加 10 来做,结果一样。

在数学上,如 a 和 b 满足: $a \text{ MOD } M = (nM + b) \text{ MOD } M$ M 为模,则称 a 和 b 同余或称 a 和 b 互为补数。

在时钟上 $+10$ 与 (-2) , $+7$ 与 (-5) , $+6$ 与 (-6) 对 12 同余或称互为补数。引入补数的概念就可以将减法化为加法来做。

3. 补码的求法

对 n 位二进制数其模为 2^n ,则

$$[X]_{\text{补}} = (2^n + X) \text{ MOD } 2^n = X$$

如 $X > 0$ 则 $[X]_{\text{补}} = (X + 2^n) \text{ MOD } 2^n = X$,即正数的补码为原正数不变。如 $X < 0$ 则:

$$[X]_{\text{补}} = (2^n + X) \text{ MOD } 2^n = 2^n - 1 + X + 1 = \bar{X} + 1$$

即负数的补码等于负数的原码除符号位外求仅加 1。

$X = 0$ 则

$$[0]_{\text{补}} = 2^n + 0 = 0$$

即 0 的补码为 0,只有一种表示方法。

下面举例说明补码的求法与应用

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

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

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

8 位带符号数的补码表示也列在表 1-3 中。它有以下特点：

(1) $[+0]_{\text{补}} = [-0]_{\text{补}} = 00000000$

(2) 8 位二进制补码表示范围为 $+127 \sim -128$, 16 位二进制补码表示范围为 $+32767 \sim -32768$

(3) 一个用补码表示的二进制数, 最高位为符号位, 当符号位为“0”(即为正数)时, 其余几位即为此数的二进制值; 但当符号位为“1”即负数时, 其余几位不是此数的二进制值, 其值为后面各位按位取反, 在最低位加 1。例如:

$[X]_{\text{补}} = 00001111$, 则 $X = +15$

$[X]_{\text{补}} = 11111111$, 则 $X = -1$

当负数采用补码表示时, 就可以把减法运算转换为加法运算。例如

$$X = 64 - 9 = 64 + (-9)$$

$$[X]_{\text{补}} = [64]_{\text{补}} + [-9]_{\text{补}}$$

$$+64 = 01000000 \quad +9 = 00001001 \quad [-9]_{\text{补}} = 11110111$$

于是

$$\begin{array}{r} 0100\ 0000 \\ +1111\ 0111 \\ \hline 1\ 0011\ 0111 \end{array} \quad \begin{array}{r} 0100\ 0000 \\ -0000\ 1001 \\ \hline 0011\ 0111 \end{array}$$

自然丢失

加法和减法运算的结果完全相同。原因是当一个数超过 256 时, 便在机器上又体现从零开始, 即 8 位字长的最大数不能超过 256, 也就是模是 256。正是由于此, 减法运算才能转换为加法运算。

表 1-3 数的表示法

十进制数	二进制数	原 码	反 码	补 码
+0	+0000000	00000000	00000000	00000000
+1	+0000001	00000001	00000001	00000001
+2	+0000010	00000010	00000010	00000010
⋮	⋮	⋮	⋮	⋮
+126	+1111110	01111110	01111110	01111110
+127	+1111111	01111111	01111111	01111111
-0	-0000000	10000000	11111111	00000000
-1	-0000001	10000001	11111110	11111111
-2	-0000010	10000010	11111101	11111110
⋮	⋮	⋮	⋮	⋮
-126	-1111110	11111110	10000001	10000010
-127	-1111111	11111111	10000000	10000001
-128	-1000000	---	---	10000000