

普通高等学校“十五”重点规划教材

21 世纪课程教材

Intel 8086/8088系列 微型计算机 原理及接口技术

杨志坚 主编

赵建国 吕英俊 副主编



中国电力出版社

www.infopower.com.cn

Intel 8086/8088系列 微型计算机 原理及接口技术

杨志坚 主编
赵建国 吕英俊 副主编

中国电力出版社

内容提要

本书以 Intel 8086/8088 机型为背景,系统地阐述了微处理器的内部结构、工作原理和微机应用系统的设计方法。同时,还对微型计算机技术的新发展做了较大篇幅的介绍。全书共 12 章,内容包括:计算机基础知识、8086/8088 微处理器、8086 的寻址方式及指令系统、8086 汇编语言基础、汇编语言源程序的汇编及上机过程、8086 汇编语言程序设计方法、存储器、输入/输出及 DMA 控制器、中断、串行和并行输入/输出、计数器/定时器 8253、从 8086 到 80X86 等。

本书内容丰富、系统性强、深入浅出、通俗易懂。既有基本理论的阐述,又有相应的硬件连接和软件设计的典型实例。其独特的编写/编排方法有利于学生的学习和教师的授课。

本书可用作高等院校非计算机专业本/专科学生《微型计算机原理及应用》课程的教材,也可供从事微型计算机应用开发的工程技术人员参考。由于该书具有便于自学的特点,因此,它也适于函授或自考生作为学习用书。

图书在版编目(CIP)数据

Intel 8086/8088 系列微型计算机原理及接口技术 / 杨志坚主编. —北京:中国电力出版社, 2003

ISBN 7-5083-1558-8

I. I… II. 杨… III. ①微型计算机, Intel 8086/8088 系列—理论 ②微型计算机, Intel 8086/8088 系列—接口设备 IV. TP36

中国版本图书馆 CIP 数据核字 (2003) 第 071932 号

—

书 名: Intel 8086/8088 系列微型计算机原理及接口技术

出版发行: 中国电力出版社

地址: 北京市三里河路6号 邮政编码: 100044

电话: (010) 88515918 传真: (010) 88518169

本书如有印装质量问题, 我社负责退换

印 刷: 汇鑫印务有限公司

开 本: 787×1092 1/16 印 张: 19 字 数: 421千字

书 号: ISBN 7-5083-1558-8

版 次: 2003年9月北京第一版

印 次: 2003年9月第一次印刷

定 价: 25.00 元

版权所有, 翻印必究

前 言

近年来,微型计算机因具有功能强大、小巧玲珑、可靠性高和价格低廉等特点而得到了飞速发展,它的应用已经渗透到工业、农业、国防、文教、卫生、航空、航天、科学研究和人们日常生活等各个角落。微型计算机技术有力地推动了各行业的技术改造和产品的更新换代,加快了人类发展的进程。

我们所说的“微型计算机的应用”,不仅仅是指把微型计算机当作我们日常工作、学习和生活中的一种工具(即熟悉微型计算机的基本操作及使用方法),更重要的是指以下两个方面:一是能利用程序设计语言开发一些必要的应用程序;二是要具备一定的微型计算机应用系统的开发能力。前者,只涉及软件方面的内容;而后者将要涉及软件和硬件两方面的知识。要进行微型计算机应用系统的开发不仅要懂得微处理器及其接口的基本工作原理和应用技术,而且还要掌握汇编语言(这是一种能直接控制硬件的惟一语言)程序的设计方法。这也正是《微型计算机原理及应用》这门课程所要介绍的主要内容。

在科学技术飞速发展的今天,具备微型计算机应用系统的开发能力是所有科技人员和高等院校的大学生必不可少的技能。

多年来,《微型计算机原理及应用》这门课程一直存在着“学生难学,教师难教”的现象。为实现我国二十一世纪新型人才的培养目标——注重基础、注重能力,同时,也是为了适应国家教育部大量缩减课程学时的要求,我们编写了这本教材。本书的编写主要突出了以下几个特点:

- (1)强调基本概念、基本内容、基本方法。
- (2)系统性强、概念明确、条理清晰。
- (3)语言通俗易懂、深入浅出。
- (4)举例及习题典型。
- (5)各章均附有对本章内容进行归纳的“小结”。
- (6)独特的编写/编排方法便于教师的“教”与学生的“学”。
- (7)学习难度降低,适于学生自学。
- (8)内容注重“应用”。

本书以 8086/8088 机型为背景,系统地阐述了微处理器的内部结构、工作原理和微机应用系统的设计方法。同时还对微型计算机技术的新发展做了较大篇幅的介绍。

全书共 12 章:第 1 章主要介绍了微型计算机的发展简史、微机/微机系统的基本组成、计算机中的数及其运算和字符的编码;第 2 章讲述了 8086/8088 微处理器的内部结构、外部引脚及其工作原理。同时,还介绍了 8086/8088 CPU 在最小/最大模式下的典型配置和总线时序;第 3 章详细介绍了 8086/8088 的寻址方式及指令系统;第 4 章讲解是 8086/8088 汇编语言的基础知识;第 5 章主要介绍 ASM-86 汇编语言源程序的汇编和上机步骤及各步骤的作用;第 6 章则通过大量示例介绍了 8086/8088 汇编语言程序的设计方法;第 7 章主要涉及半导体存储器(ROM 和 RAM)的内部结构、外部引脚及使用方法等方面的内容;第 8 章重点介绍了计算机

系统中的输入/输出操作及 DMA 控制器 8237A 的工作等原理及使用方法;第 9 章详细讲解了 8086/8088 的中断系统及中断管理器 8259A 的内部结构、外部引脚、工作原理和使用方法;第 10 章对串行输入/输出接口芯片 8251A 和并行输入/输出接口芯片 8255A 的内部结构、外部引脚和使用方法做了全面的介绍;第 11 章重点讲解了计数器/定时器 8253 的编程结构、工作原理及具体的使用方法;最后一章通过与 8086 比较的方式,介绍了 80X86 新的结构/功能及先进的设计思想。

本教材的第 3 至第 6 章和第 8 至第 11 章由杨志坚编写,第 12 章由赵建国编写;第 1、2、7 章由吕英俊(山东科技大学)编写。徐建政、刘志珍、秦安文和孙怡等也参加了部分内容的编写。全书由杨志坚负责统编。本书在编写过程中参阅了国内现有的一些《微型计算机原理及应用》课程的教材及有关资料,得到了不少启发并吸取了许多先进的经验。

虽然,这是一本为高等院校非计算机专业本/专科学生学习《微型计算机原理及应用》课程而编写的教材,但由于它具有便于自学的特点,因此,本书也非常适于用作函授生或自考生的学习用书。

山东大学计算机科学与技术学院的贾智平教授和电气工程学院的孟昭敦教授对本书的编写提出了不少宝贵意见;我校教材管理中心的王保祥主任对本书的出版给予了大力支持;付连强、李义、冷雪飞、张婷婷等同学对书中的程序进行了调试,在此表示衷心的感谢。

由于编者水平有限,书中错误和不妥之处在所难免,恳请各位读者批评指正。

编 者

目 录

前 言

第 1 章 计算机基础知识	1
1.1 微型计算机的发展简史	1
1.2 微型计算机和微型计算机系统	2
1.3 进位计数制与不同进制数间的转换	2
1.4 计算机中数的表示及运算	5
1.5 字符编码	7
1.6 逻辑运算	8
1.7 小结	9
1.8 习题	9
第 2 章 8086/8088 微处理器	10
2.1 8086/8088 CPU 概述	10
2.2 8086 CPU 的内部结构	10
2.3 8086 存储器组织	12
2.4 8086 输入/输出组织	14
2.5 8086 堆栈	15
2.6 8086 CPU 引脚及功能	15
2.7 8086 在最小模式和最大模式下的典型配置	19
2.8 8086 的操作及总线时序	22
2.9 小结	32
2.10 习题	33
第 3 章 8086 的寻址方式及指令系统	34
3.1 概述	34
3.2 8086 的寻址方式	34
3.3 8086 指令格式	39
3.4 8086 指令系统	42
3.5 小结	70
3.6 习题	70
第 4 章 8086 汇编语言基础	73
4.1 汇编语言简介	73
4.2 ASM - 86 的标记	74
4.3 表达式	75
4.4 语句	79
4.5 伪指令	79
4.6 小结	86

4.7 习题	87
第 5 章 汇编语言源程序的汇编及上机过程	88
5.1 ASM - 86 汇编语言源程序的结构	88
5.2 宏指令	89
5.3 主要的宏操作伪指令	91
5.4 ASM - 86 汇编语言源程序的汇编与连接	95
5.5 汇编语言程序的上机过程	96
5.6 汇编语言和操作系统 PC DOS 的接口	100
5.7 小结	102
5.8 习题	102
第 6 章 8086 汇编语言程序设计方法	103
6.1 汇编语言程序设计步骤	103
6.2 顺序结构程序设计	103
6.3 分支结构程序设计	105
6.4 循环结构程序设计	108
6.5 过程及过程的设计	111
6.6 程序设计举例	119
6.7 小结	126
6.8 习题	127
第 7 章 存储器	128
7.1 概述	128
7.2 静态 RAM 存储器	129
7.3 动态 RAM 存储器	133
7.4 只读存储器	140
7.5 存储器与 CPU 连接时应考虑的问题	142
7.6 小结	144
7.7 习题	145
第 8 章 输入/输出及 DMA 控制器	146
8.1 概述	146
8.2 CPU 与外设间信息传送的控制方式	147
8.3 DMA 控制器 8237A	149
8.4 小结	158
8.5 习题	159
第 9 章 中断	160
9.1 中断的基本概念	160
9.2 8086 的中断系统	160
9.3 中断控制器 8259A	167
9.4 小结	179
9.5 习题	180

第 10 章 串行和并行输入/输出	181
10.1 并行输入/输出接口芯片 8255A	181
10.2 串行输入/输出接口芯片 8251A	195
10.3 小结	210
10.4 习题	211
第 11 章 计数器/定时器 8253	213
11.1 概述	213
11.2 可编程计数器/定时器 8253	213
11.3 小结	223
11.4 习题	223
第 12 章 从 8086 到 80X86	225
12.1 Intel 80X86 微处理器系列	225
12.2 16 位微处理器的发展	225
12.3 80386 微处理器	232
12.4 80486 微处理器	237
12.5 80X86 的指令系统	252
12.6 习题	256
附录	257
附录一 ASCII 码表	257
附录二 8086 指令系统表	258
附录三 8086/8088 指令编码一览表	263
附录四 ROM BIOS 中断调用	272
附录五 DOS 系统功能调用一览表	278
附录六 IBM PC 机中断向量的布局	283
附录七 MASM5.0 宏汇编程序出错信息	283
附录八 80486 指令系统表	287
参考文献	291

第1章 计算机基础知识

1.1 微型计算机的发展简史

世界上第一台电子计算机于1946年在美国宾夕法尼亚大学诞生,在迄今五十多年的时间里,计算机及应用技术得到了突飞猛进的发展。

通常,电子计算机按其体积、性能和价格被分为巨型机、大型机、中型机、小型机和微型机。就系统结构和基本工作原理而言,微型机跟其他几类计算机没有本质上的区别,所不同的是微型机采用了集成度很高的器件和部件。

由于微处理器是微型计算机的核心部件,它在很大程度上决定了微型计算机及其系统的主要性能指标,因此,微型计算机的发展过程就是微处理器的发展过程。

世界上第一个微处理器(微型机的核心部件)是美国 Intel 公司 1971 年生产的 Intel 4004,从那时到现在,微型计算机已经历了四个发展时代,并且,又出现了第五代。第一代(1971 ~ 1973)是 4 位和低档 8 位微处理器时代,以 Intel 4004 和 Intel 8008 为代表。属于微型计算机发展的初级阶段,这些微处理器的集成度大约为 2000 管/片,时钟频率为 1MHz,指令的平均执行时间为 $10 \sim 20\mu\text{s}$ 。其应用主要局限于消费领域。如家用电器、计算器等。第二代(1974 ~ 1977)是 8 位微处理器时代,其代表产品有 Intel 8080/8085、MC6800、Z80 和 MC6809,这些微处理器的时钟频率为 $2 \sim 4\text{MHz}$,平均指令执行时间为 $1 \sim 2\mu\text{s}$,集成度为 5000 ~ 10000 管/片。这个时期,微处理器的设计和生产已相当成熟,并且,配套的各类器件也很齐全,因此,8 位微处理器一度成为应用的主流,并被广泛应用于教学、实验、工业控制和智能仪器等领域。第三代(1978 ~ 1984)是 16 位微处理器时代,代表产品有 Intel 8086/8088、Z8000、MC68000 和 Intel 80286 等,这些微处理器的时钟频率为 $4 \sim 10\text{MHz}$,集成度为 20000 ~ 100000 管/片,指令的平均执行时间约为 $0.5\mu\text{s}$ 。其性能已可与过去的中档小型计算机相比。它们的应用范围已涉及实时控制、数据管理和组联局域网等方面。第四代是(1985 ~ 1999)32 位微处理器时代。代表产品是 Intel 80386/ 80486、MC68020/MC68040 和 Pentium(奔腾)、Pentium Pro、Alpha21164、AMD - K6、Pentium II 和 Pentium III 等。这些微处理器的时钟频率为 $20 \sim 500\text{MHz}$,甚至更高,集成度为 270000 ~ 8200000 管/片,指令的平均执行时间约为 $1 \sim 100\text{nS}$ (Pentium III 初期产品的时钟频率已达 1GHz)。第五代(2000 年至今):2000 年 8 月,Intel 向世界展示了 Itanium(安腾) CPU,新一代字长 64 位的微处理器已经诞生。在新世纪到来之时,微处理器也迎来了一个新的发展时代。

微处理器技术的发展过程是微处理器在其结构体系上不断改进、优化的过程,是集成度、功能和速度不断提高的过程,也是性价比不断增长的过程。然而,这一切都离不开集成电路技术的有力支持。

微处理器技术迅速发展的动力主要来源于微型计算机在世界范围内的广泛应用和普及。我们相信,随着信息高速公路、网络技术及多媒体技术的建立和发展,随着特大规模集成电路和巨大规模集成电路技术的不断提高,微处理器技术及微型计算机的应用必定还会向纵深发展,其前景更加宽广、美好。

1.2 微型计算机和微型计算机系统

由图 1-1 可以看出,微型计算机是由微处理器、存储器、输入/输出接口电路、系统总线和总线控制逻辑构成的。

微处理器(亦称 MPU 或 CPU)是微型计算机的核心部分(人们常把它比作人的大脑)。它是微型计算机的运算和控制中心,主要由运算器、控制器、寄存器和内部总线组成。它的性能将在很大程度上决定着整个微型计算机的主要性能指标。

存储器包括只读存储器(ROM)和随机存储器(RAM),分别用于存放程序和数据。

I/O 接口电路处于微处理器与外部设备之间,用于协调二者的信息交换。由于微处理器与外部设备之间的信息传输总是通过 I/O 接口进行的(见图 1-1),所以,人们有时会把外部设备跟 I/O 接口混为一谈。

系统总线包括地址总线、数据总线和控制总线。微处理器通过地址总线(属单向信号线,只能输出)输出地址信息,用于选中它要寻址的存储器或 I/O 接口;数据总线(属双向信号线,可以输出或输入)用于传输数据信息,它是微处理器与存储器和外部设备之间进行数据传输的必经之路。控制总线则用于传送控制信息。该总线上的信息可能是由微处理器本身发出的,也可能是由外部发给微处理器的。

总线控制逻辑是微处理器与系统总线之间的接口。

微型计算机系统是由微型计算机、软件系统和外部设备组成的。其中,软件系统包括系统软件(如操作系统、语言处理程序、数据库管理系统和网络系统等)和应用软件(如字处理软件、电子表格软件、绘图软件、网络通信软件以及用户为满足自己的需要所开发的程序等)。

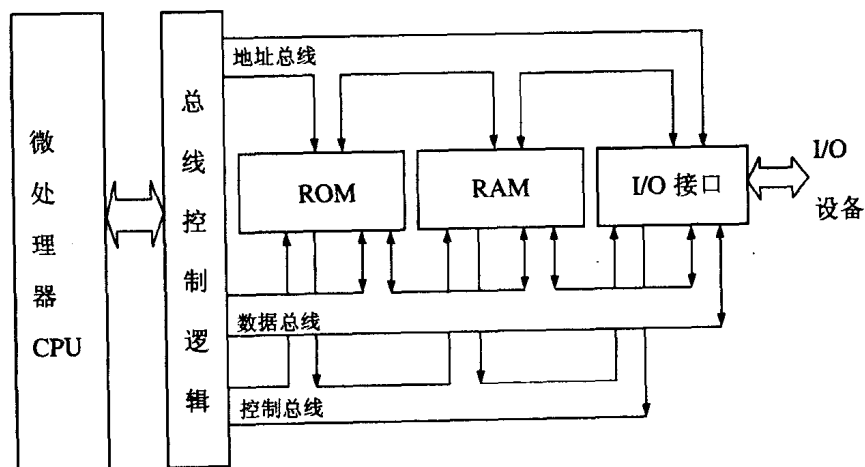


图 1-1 微型计算机的结构

1.3 进位计数制与不同进制数间的转换

1.3.1 进位计数制

按进位的方法进行计数,称进位计数制。当今,计数制有十进制、二进制、十六进制和八进

制等。其中,十进制是最常见的,它有 0、1、2、…、9 十个数符,计算原则是“逢十进一,借一当十”。

在计算机系统中,广泛使用的是二进制,它有 0、1 两个数符,计算原则是“逢二进一,借一当二”。

一个 R 进制的数 N_R 可表示为

$$N_R = \sum_{i=-m}^{n-1} a_i R^i$$

式中, n 和 m 分别为 N_R 小数点左边和右边的位数。 a_i 和 R^i 分别为 N_R 第 i 位的数符和第 i 位的权, R 称进位制的基数(即该进制数中允许使用的数符个数)。在 R 进制中,允许使用的数符有 R 个,计算原则是“逢 R 进一,借一为 R ”。

虽然,在计算机中采用二进制进行各种运算和操作具有许多优点,但书写冗长,读起来麻烦,且不便记忆。因此,人们在程序设计和计算中常采用十六进制或八进制。至于十六进制,它有 0、1、2、…、9、A、B、C、D、E、F 共十六个数符,计算原则是“逢十六进一,借一当十六”;八进制具有 0、1、2、…、7 共八个数符,其计算原则是“逢八进一,借一当八”。十进制数(0~15)的二进制、八进制和十六进制数表示见表 1-1。

表 1-1 十进制数(0~15)的二进制、八进制和十六进制数表示

十进制	二进制	八进制	十六进制	十进制	二进制	八进制	十六进制
0	0	0	0	8	1000	10	8
1	1	1	1	9	1001	11	9
2	10	2	2	10	1010	12	A
3	11	3	3	11	1011	13	B
4	100	4	4	12	1100	14	C
5	101	5	5	13	1101	15	D
6	110	6	6	14	1110	16	E
7	111	7	7	15	1111	17	F

注 1. 常采用以下两种方法来区别不同进制的数:①加下标:如 $(15)_{16}$ 、 $(10)_8$ 、 $(110)_2$ 、 $(34)_{10}$ 。②加后缀:如 15H、10Q、110B、34D(其中,H 表示十六进制,Q 表示八进制,B 表示二进制,D 表示十进制)。

2. 究竟采用哪一种进制完全取决于人们的需要和方便。

3. 当采用不同进制来描述同一数量的物质时,将有不同的表示形式。

1.3.2 不同进制数之间的转换

一、二进制数转换为十进制数

各位二进制数符与其对应位的权之乘积的和即为该二进制数所对应的十进制数。例如

$$\begin{aligned} 1101.101B &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 8 + 4 + 0 + 1 + 0.5 + 0 + 0.125 = 13.625D \end{aligned}$$

二、十进制数转换为二进制数

当把十进制数转换为二进制数时应将该十进制数的整数部分和小数部分的转换分别进行。具体地讲就是:

(1)把要转换的十进制整数部分不断除以 2,并记下余数,直到商为 0 为止(即除 2 取余法)。

(2)把要转换的十进制小数部分不断乘以 2,并记下整数部分,直到结果的小数部分为 0 为止(即乘 2 取整法)。

【例 1-1】 求 117.8125D 所对应的二进制数。

解:		商数	余数	对应的 二进制位
整数部分的转换	117/2	58	1	$a_0 = 1$
	58/2	29	0	$a_1 = 0$
	29/2	14	1	$a_2 = 1$
	14/2	7	0	$a_3 = 0$
	7/2	3	1	$a_4 = 1$
	3/2	1	1	$a_5 = 1$
	1/2	0	1	$a_6 = 1$
		结果的 小数部分	结果的 整数部分	对应的 二进制位
小数部分的转换	0.8125×2	0.625	1	$a_{-1} = 1$
	0.625×2	0.25	1	$a_{-2} = 1$
	0.25×2	0.5	0	$a_{-3} = 0$
	0.5×2	0.0	1	$a_{-4} = 1$

因此 $117.8125D = 1110101.1101B$

不难看出,无论是整数部分的转换还是小数部分的转换,首先得到的一位将是其相应二进制数最靠近小数点的整数位和小数位(即 a_0 和 a_{-1})。

三、十六进制数和二进制数之间的转换

十六进制数转变为二进制数的方法是按表 1-2 中的对应关系把每位十六进制数写成四位二进制数。

【例 1-2】 把 38.A6H 转换为二进制数。

解:因为 $38H = 00111000$, $A6H = 10100110$

所以 $38.A6H = 00111000.10100110 = 111000.1010011$

把二进制数转换为十六进制数的方法是从二进制数的小数点开始分别向左、向右每四位划一段(不足四位时,用 0 补足),然后,把每段按表 1-2 的对位关系用一位十六进制数代之即可。

【例 1-3】 把 110010.111B 转换为十六进制数。

解:将二进制数 110010.111 每四位划一段,变成‘00110010.1110’,经查表得

$110010.111B = 32.EH$

表 1-2 四位二进制数和一位十六进制数之间的对应关系

四位 二进制	一位 十六进制	四位 二进制	一位 十六进制	四位 二进制	一位 十六进制	四位 二进制	一位 十六进制
0000	0	0100	4	1000	8	1100	C
0001	1	0101	5	1001	9	1101	D
0010	2	0110	6	1010	A	1110	E
0011	3	0111	7	1011	B	1111	F

四、十六进制数和十进制数之间的转换

各位十六进制数与其对应位的权之乘积的和即该十六进制数所对应的十进制数。

【例 1-4】

解:把 4F6.AC 转换为十进制数

$$\begin{aligned} N &= 4F6.AC \\ &= 4 \times 16^2 + 15 \times 16^1 + 6 \times 16^0 + 10 \times 16^{-1} + 12 \times 16^{-2} \\ &= 4 \times 256 + 15 \times 16 + 6 \times 1 + 10 \times 0.0625 + 12 \times 0.00390625 \\ &= 1270.671875 \end{aligned}$$

十进制数转换为十六进制数的方法类似于十进制数到二进制数的转换方法,也是把整数部分和小数部分的转换分别进行,即:

(1)把待转换的十进制整数部分不断乘以 16,并记下余数,直到商为 0 为止(除 16 取余法);

(2)把待转换的小数部分不断乘以 16,记下整数部分,直到结果的小数部分为 0 为止(乘 16 取整法)。

【例 1-5】 求 1270.671875D 所对应的十六进制数。

解:		商 数	余数	对应的十六进制位
整数部分的转换	1270/16	79	6	$a_0 = 6$
	79/16	4	15	$a_1 = F$
	4/16	0	4	$a_2 = 4$
		结果的 小数部分	结果的 整数部分	对应的二 十进制位
小数部分的转换	0.671875×16	0.75	10	$a_{-1} = A$
	0.75×16	0.00	12	$a_{-2} = C$

因此 $1270.671875D = 4F6.ACH$

1.4 计算机中数的表示及运算

在计算机中既可以使用无符号数也可使用带符号数,由于计算机仅“认得”二进制,故数的符号也必须二进制化。通常,人们规定用最高位来表示数的符号:‘0’表示正数,‘1’表示负数。一个数在机器(计算机)中的表示形式称该数的机器数。反之,一个机器数所表示的那个数的本身称该机器数的真值。

1.4.1 无符号数的表示及运算

无符号数和它的机器数在表示形式上是完全一致的,其特点是每一个二进制位均表示数值,且计算规则为:1+0=1,0+1=1,0+0=0,1+1=10。

【例 1-6】 计算 157+163。

解:由于 157=10011101,163=10100011

因此 $157 + 163 = 10011101 + 10100011 = 101000000 = 320$

1.4.2 带符号数的表示及运算

由于‘+’和‘-’符号的0、1化,所以,带符号数和它的机器数在表示形式上是不同的。在计算机中可以用不同的码制来表示带符号数,常见的有原码、反码及补码表示法,现介绍如下(设一带符号数由机器中的8个二进制位表示)。

一、原码表示法

最高位表示符号(“0”表示正数,“1”表示负数),其余位表示数值。如

$$\begin{array}{ccc} [+8]_{\text{原}} = 0 \underbrace{0001000}_{\text{数值位}} & & [-8]_{\text{原}} = 1 \underbrace{0001000}_{\text{数值位}} \\ \uparrow & & \uparrow \\ \text{符号} & & \text{符号} \end{array}$$

二、反码表示法

正数的反码与其原码相同,如 $[+8]_{\text{反}} = [+8]_{\text{原}} = 00001000$

负数的反码是将其原码的数值位逐位取反,符号位不变。如 $[-8]_{\text{反}} = 11110111$

三、补码表示法

正数的补码与其原码相同,如 $[+8]_{\text{补}} = [+8]_{\text{原}} = 00001000$

负数的补码是将其反码在末位上加1,即得所求。如 $[-8]_{\text{补}} = 11111000$

四、补码运算的规则

补码运算时,符号位应跟其他位一样参加运算,并且,

$$(1) [X]_{\text{补}} + [Y]_{\text{补}} = [X + Y]_{\text{补}};$$

$$(2) [X]_{\text{补}} + [-Y]_{\text{补}} = [X - Y]_{\text{补}}。$$

五、说明

(1)引入补码的原因是为了用加法代替减法。通常,在计算机内部用补码来表示带符号数。

(2)8位二进制补码所能表示的带符号数的范围是 $-128 \sim +127$,当运算结果超出了这一范围时,将会出错。这在计算机术语中称作“溢出”,为防止由于“溢出”而产生运算结果的错误,可采用增加二进制位数的方法加以解决。如要求计算机计算 $(-98)_{10} + (-32)_{10}$,若采用8位二进制补码,其运算结果必定产生溢出,而改用16位二进制补码后,则“溢出”问题就不会出现了。

(3)补码的符号扩展问题。所谓符号扩展问题是指一个带符号数由较少二进制位扩展到位数较多时应注意的问题。对于用补码表示的数而言,正数的符号扩展应该在前面补0,而负数的符号扩展则应该在前面补1,如8位二进制补码表示中, $[+46]_{\text{补}} = 00101110$, $[-46]_{\text{补}} = 11010010$,若改用16位二进制补码表示,则

$$[+46]_{\text{补}} = 0000000000101110, \quad [-46]_{\text{补}} = 1111111111010010$$

(4)带符号数X的补码定义式

$$\text{整数补码 } [X]_{\text{补}} = \begin{cases} X & +2^{n-1} > X \geq 0 \\ 2^n + X & 0 > X \geq -2^{n-1} \end{cases} \quad \text{其中, } n \text{ 为二进制补码的位数。}$$

$$\text{小数补码 } [X]_{\text{补}} = \begin{cases} X & +1 > X \geq 0 \\ 2 + X & 0 > X \geq -1 \end{cases}$$

(5)由补码求真值的方法

先看 $[X]_{\text{补}}$ 的最高位,若为‘0’,说明X为正数,且其 $[X]_{\text{补}}$ 的其余位即X的数值;若为‘1’,

说明 X 为负数,其数值应通过将 $[X]_{\text{补}}$ 的其余位逐位取反,然后末位加 1 的方法得到。

【例 1-7】 已知 $[X]_{\text{补}} = 01111000$,求 X 的真值。

解:由于最高位为 0,故 X 为正,数值为 1111000,所以, $X = +120$ 。

【例 1-8】 已知 $[X]_{\text{补}} = 10110010$,求 X 的真值。

解:由于最高位为 1,故 X 为负,将 $[X]_{\text{补}}$ 的其余位 0110010 逐位取反且末位加 1,便得 X 的数值 1001110,所以 $X = -78$ 。

(6)“溢出”的判断方法

在计算机中,“溢出”的判断是由下面的逻辑表达式来实现的(设采用 n 位二进制补码进行运算):

$$OF = CY_n \oplus CY_{n-1}$$

其中,OF 是“溢出”标志位; CY_n 是最高位的进位(第 n 位向第 n+1 位的进位); CY_{n-1} 是次高位的进位(第 n-1 位向第 n 位的进位)。

(7) $[[X]_{\text{补}}]_{\text{求补}} = [-X]_{\text{补}}$

所谓“求补”就是将二进制补码的每一位(包括符号位)逐位取反,末尾加 1。

1.4.3 二进制编码的十进制数

计算机只“认得”二进制数,而人们习惯使用十进制数,为便于“人一机对话”,在兼顾了人与计算机各自特点的基础上,人们创造了一类二进制编码的十进制数,称作 BCD 码(即 Binary Coded Decimal),计算机中采用的 BCD 码多数是 8421BCD 码。

8421BCD 码具有十个不同的数字符号,且是逢十进位的,所以它是十进制数,但由于它是用四位二进制对一位十进制数符(0、1、2、…、9)进行编码的,故称二进制编码的十进制数。8421BCD 码跟十进制数的对应关系见表 1-3。有了这个表,8421BCD 码与十进制数的转换就变得非常容易。

例如 $(0001\ 0101\ 0110)_{\text{BCD}} = (156)$

又如 $(29)_{10} = (0010\ 1001)_{\text{BCD}}$

但是,8421BCD 码与二进制数间的转换是不直接的,须用十进制数作为过渡,即先把 BCD 码转换为十进制数,然后再转换为二进制数,反之亦然。

如 $(0001\ 0000\ 0000)_{\text{BCD}} = (100)_{10} = (1100100)_2$

又如 $(11111110)_2 = (254)_{10} = (0010\ 0101\ 0100)_{\text{BCD}}$

表 1-3 十进制数与 8421BCD 码的对应关系

十进制数	0	1	2	3	4	5	6	7	8	9
BCD 码	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

1.5 字符编码

计算机除了处理数字之外,还需处理字符(包括字母、数字、标点及各种算符等),这些字符在机器中也必须按特定的规则,用二进制进行编码,称字符编码。字符编码主用于计算机与 I/O 设备的信息交换之中。美国信息交换标准代码 ASCII 码(American Standard Code for Informa-

tion Interchange)是目前计算机中使用最为广泛的字符编码,ASCII 码的编码见附录一。

ASCII 码采用七位二进制编码,共有 128 个字符,其中,33 个为不可打印和显示的控制字符,95 个是可打印字符。在微机中,一个 ASCII 字符占用一个字节(8 位二进制),该字节的最高位在进行信息交换时常用作奇偶校验位,但读取字符时,规定把它当作‘0’。从附录一可以看出,数字 0~9 所对应的 ASCII 码为 30H~39H;大写字母 A~Z 所对应的 ASCII 码为 41H~5AH;小写字母 a~z 所对应的 ASCII 码为 61H~7AH(比大写字母的 ASCII 码大 20H);空格(SPACE)为 20H;回车(CR)为 0DH;……。

1.6 逻辑运算

逻辑数据是需要计算机处理的数值型数据中的另一类数据,这种数据只有两个取值(0 和 1),可以用来表示逻辑状态,进行逻辑判断及运算。计算机能进行的基本逻辑运算有“与”、“或”、“非”及“异或”等运算。

1.6.1 “与”运算(AND)

“与”运算可用符号‘·’或者‘ $\wedge$ ’来表示,其运算规则如表 1-4(a)所示。即只有当 A、B 两个量的取值均为 1 时, $A \wedge B$ 的运算结果才为 1,否则, $A \wedge B = 0$ 。

表 1-4(a) 基本逻辑运算的真值表(1)

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

1.6.2 “或”运算(OR)

“或”运算可用符号 + 或者 $\vee$ 来表示。或运算规则见表 1-4(b)。即 A、B 两变量中只要有一个变量取值为 1,则 $A \vee B$ 的运算结果就为 1,否则 $A \vee B = 0$ 。

表 1-4(b) 基本逻辑运算的真值表(2)

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

表 1-4(c) 基本逻辑运算的真值表(3)

A	$\bar{A}$
0	1
1	0

表 1-4(d) 基本逻辑运算的真值表(4)

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	0

1.6.3 “非”运算(NOT)

如变量为 A,则其“非”运算的结果用 $\bar{A}$ 来表示。“非”运算规则如表 1-4(c)所示。

1.6.4 “异或”运算(XOR 即 EXCLUSIVE-OR)

“异或”可用符号 $\oplus$ 或者 $\vee$ 来表示,其运算规则见表 1-4(d)。即当 A、B 两个变量取值相异时, $A \vee B$ 的运算结果为 1,取值相同时 $A \vee B = 0$ 。

1.7 小 结

本章主要内容小结和基本要求:

- (1)微型计算机是由微处理器、存储器(ROM 和 RAM)以及 I/O 接口组成的。
- (2)应掌握的不同进制有:十进制、二进制、八进制和十六进制。
- (3)计算机中的数分为无符号数和有符号数。其中,有符号数通常是用补码来表示的。
- (4)计算机中引入补码的原因是用加法来代替减法。
- (5)8421BCD 码是一种二进制编码的十进制数。
- (6)ASCII 码是字符的二进制编码。
- (7)本章内容是学好《微型计算机原理及应用》课程的前提。

1.8 习 题

1. 将下列十进制数转换成二进制数和十六进制数:

- (1)120 (2)254 (3)1000 (4)4096

2. 将下列二进制数转换成十六进制数和十进制数:

- (1)1110000 (2)10010011 (3)1101101011 (4)1100001010000

3. 写出下列十进制数的 8421BCD 码:

- (1)34 (2)129 (3)2048 (4)5760

4. 完成下列十六进制数的运算:

- (1)8A + 39 (2)1234 + A6 (3)ABCD - E5 (4)7AB - FO

5. 下列各数均为十六进制表示的 8 位二进制数,请说明它们分别被看作是无符号数或用补码表示的带符号数时,它们所表示的十进制数是什么?

- (1)FF (2)6A (3)80 (4)74 (5)32

6. 请写出下列字符串的 ASCII 码值:

I am a student.

This is a number 3579.

7. 写出 $(+120)_{10}$ 和 $(-120)_{10}$ 的原码、反码和补码。

8. 已知 $[X]_{\text{补}} = 01001000$ 、 $[Y]_{\text{补}} = 11001000$, 分别求其真值。

9. 下列各数均为十进制数,请用 8 位二进制补码计算下列各题,并分别说明其运算结果是否发生溢出。

- (1)85 + (-76) (2)85 - 76 (3)85 - (-76)
(4)(-85) - 76 (5)(-85) - (-76) (6)(-85) + 76