

普通高等教育“十一五”规划教材

微型计算机原理与应用

第2版

主 编 杨有君 史志才
副主编 张志弘 韩彦铎 陈 波



机械工业出版社
CHINA MACHINE PRESS



普通高等教育“十一五”规划教材

微型计算机原理与应用

第2版

主 编 杨有君 史志才
副主编 张志弘 韩彦铎 陈 波
参 编 王 红 王德安
主 审 季振洲 赵长德



机械工业出版社

本书主要介绍以 8086/8088CPU 为核心的 16 位微型计算机系统及其接口技术。虽然目前高档微机系统已很普及,但 8086CPU 作为主流微型计算机的基础,能够系统、全面地反映微型计算机系统的工作原理,采用 8086/8088CPU 介绍微型计算机技术仍具有典型性。

本书在原有教材的基础上进行了修订,并结合计算机技术的最新发展适当地增加了一些新知识和新技术;力求做到深入浅出,语言精炼,内容完整和系统性强。

在编写本书的过程中,注重理论分析与实际技术相结合,既有原理描述,又有实际应用分析;全书结构组织合理,内容衔接自然,文字通俗流畅,易于理解和学习。例如,本书第二章第一节以计算机信息流的思想讲述微型计算机系统工作的全过程,使学生非常直观、系统地了解计算机工作的基本原理。

图书在版编目(CIP)数据

微型计算机原理与应用/杨有君,史志才主编. —2 版. 北京:机械工业出版社, 2007. 8

普通高等教育“十一五”规划教材
ISBN 978-7-111-05359-0

I. 微… II. ①杨…②史… 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(2007)第 102894 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:邓海平 版式设计:霍永明 责任校对:李婷

封面设计:张 静 责任印制:洪汉军

北京瑞德印刷有限公司印刷(三河市明辉装订厂装订)

2008 年 1 月第 2 版第 1 次印刷

184mm×260mm·24 印张·591 千字

标准书号:ISBN 978-7-111-05359-0

定价:36.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换
销售服务热线电话:(010) 68326294

购书热线电话:(010) 88379639 88379641 88379643

编辑热线电话:(010) 88379715

封面无防伪标均为盗版

前 言

本书第1版是根据1993年9月国家教委工科计算机基础课程教学指导委员会确定的高等学校工科本科非计算机专业“微机原理与应用”课程教学基本要求而组织编写的,1997年5月出版发行,本次修订为第2版。此次修订的目的是为了跟踪微型计算机技术的最新发展,同时对书中的内容进行了调整和增补,扩大了本书的适应面,使此书不但适用于机械电子工程等非计算机专业的本科教学,同时也可以作为计算机科学与技术、自动化、电子、通信等本科专业的教材或教学参考书。

本书主要介绍以8086/8088CPU为核心的16位微型计算机系统及其接口技术。虽然目前高档微机系统已很普及,但8086CPU作为主流微型计算机的基础,能够系统、全面地反映微型计算机系统的工作原理,采用8086/8088CPU介绍微型计算机技术仍具有典型性。

本书编写的主要目的是使学生从理论和实践上掌握微型计算机的基本组成、工作原理、接口技术以及汇编语言程序设计方法,建立微型计算机系统的整体概念,使学生具有微机系统软硬件开发的初步能力。本书注重理论分析与实际技术相结合,既有原理描述,又有实际应用分析;全书结构组织合理,内容衔接自然,文字通俗流畅,易于理解和学习。例如,本书第二章第一节以计算机信息流的思想讲述微型计算机系统工作的全过程,使学生非常直观、系统地了解计算机工作的基本原理。

本书各章的组织安排为:第一章介绍微型计算机系统的基础知识,第二章介绍微型计算机的工作原理和8086微处理器,第三章介绍8086的寻址方式和指令系统,第四章为8086汇编语言程序设计,第五章介绍存储器系统,第六章为输入/输出及中断技术,第七章介绍用于扩充微机系统功能的可编程接口芯片,第八章为数/模转换和模/数转换技术,第九章介绍微型计算机的发展与应用。

本书由杨有君、史志才主编,张志弘、韩彦铎、陈波任副主编,第一章由王红、张志弘、韩彦铎负责修订和编写,第二、三、四章、附录由陈波负责修订和编写,第五、九章由王红、张志弘负责修订和编写,第六、七、八章由史志才、韩彦铎负责修订和编写;杨有君、史志才负责全书的组织和统稿;王德安参与了本书的修订和源程序的调试工作。

本书在原有教材的基础上进行了修订,并结合计算机技术的最新发展适当增加了一些新知识和新技术;在此对原书的几位作者和主审——清华大学的赵长德教授表示衷心的感谢,感谢他们为本书的修订和出版奠定了基础。此次修订得到了哈尔滨工业大学计算机学院博士生导师季振洲教授的审阅和指导,在此表示衷心的感谢。

本书力求做到深入浅出,文辞精炼,内容完整和系统性强。由于编者水平有限,书中定有不足之处,敬请读者提出宝贵意见。作者的E-mail为szc1964@163.com。

编者

于大连市

目 录

前言

第一章 微型计算机系统基础知识 1

- 第一节 引言 1
- 第二节 微型计算机的应用领域 5
- 第三节 计算机中的数和编码 6
- 第四节 微型计算机硬件基础知识 18
- 思考题与习题 27

第二章 微型计算机的工作原理和

8086 微处理器 29

- 第一节 微型计算机的工作原理 29
- 第二节 8086CPU 的结构特点 39
- 第三节 8086/8088 的引脚功能 44
- 第四节 时钟发生器和总线管理芯片 47
- 第五节 8086/8088 最小模式系统 51
- 第六节 8086/8088 最大模式系统 55
- 第七节 总线占用与释放 61
- 思考题与习题 63

第三章 8086 的寻址方式和指令

系统 65

- 第一节 8086 寻址方式 66
- 第二节 8086 指令系统概况 70
- 第三节 8086 指令系统 73
- 第四节 80X86 指令系统介绍 110
- 思考题与习题 112

第四章 8086 汇编语言程序设计 116

- 第一节 程序设计语言概论 116
- 第二节 汇编语言和汇编程序 118
- 第三节 汇编语言数据 119
- 第四节 伪指令 123
- 第五节 8086 汇编语言程序设计基础 130
- 第六节 汇编语言程序的上机过程和 DOS 功能调用 132
- 第七节 8086 汇编语言程序设计 137
- 第八节 汇编语言高级应用技术 150
- 思考题与习题 157

第五章 存储器 161

第一节 存储器简介 161

- 第二节 半导体存储器 165
- 第三节 存储器的组成 178
- 第四节 存储器与 CPU 的连接 182
- 思考题与习题 191

第六章 输入/输出系统及中断技术 192

- 第一节 输入/输出系统概述 192
- 第二节 输入/输出端口的编址与寻址 195
- 第三节 CPU 与外设间数据的传输方式 196
- 第四节 8086 CPU 的中断方式 203
- 第五节 可编程中断控制器 8259A 及其应用 210
- 思考题与习题 228

第七章 可编程接口芯片 229

- 第一节 并行通信与可编程接口芯片 8255A 229
- 第二节 可编程计数器/定时器 8253 239
- 第三节 串行通信与可编程接口芯片 8251A 246
- 第四节 DMA 控制器 8237A 261
- 思考题与习题 274

第八章 数/模转换和模/数转换 276

- 第一节 概述 276
- 第二节 数/模 (D/A) 转换 277
- 第三节 模/数 (A/D) 转换 284
- 第四节 部分模/数和数/模转换芯片简介 293
- 思考题与习题 297

第九章 微型计算机的发展与应用 299

- 第一节 微型计算机的发展 299
- 第二节 X86 系列机性能的发展 300
- 第三节 微型计算机系统构成 314
- 第四节 总线 326
- 第五节 应用举例 331
- 思考题与习题 339

附录 340

- 附录 A 8086/8088 指令编码格式 340

附录 B 8086/8088 指令编码一览表	351	附录 D MS-DOS 的功能调用表	371
附录 C 常用指令对标志寄存器标志位的 影响表	369	参考文献	376



第一章

微型计算机系统基础知识

第一节 引言

计算机是人类历史上最伟大的发明创造之一，它的出现虽然仅仅具有几十年的历史，但对人类社会的进步与发展却起到了巨大的推动作用。世界上第一台由程序控制的计算机诞生于电子管时代，这就是1946年由美国宾夕法尼亚大学研制出来的电子数字计算机（Electronic Numerical Integrator And Computer），简称ENIAC电子计算机。这台计算机共用18 800个电子管，1 500个继电器，占地 170m^2 ，重达30t，功耗150kW，运算速度为每秒5 000次加法运算。这个庞然大物对现今种类繁多、性能优异、体积小巧的计算机来说，已是相形见绌了。但却是它奠定了电子计算机的技术基础，可以称得上是当今计算机的始祖。

与ENIAC计算机研制的同时，冯·诺依曼同莫尔小组合作研制出了EDVAC计算机，在这台计算机中确立了计算机的5个基本部件：输入部分、输出部分、运算器、存储器和控制器。这样的组成以及采用的二进制原则等理论延用至今，所以现代的一般计算机被称作冯·诺依曼结构计算机。

半个世纪以来，计算机的发展日新月异，令人瞩目。而带动计算机高速发展的一个极为重要的因素是计算机所采用的物理器件在不断进步，先后经历了电子管、晶体管、集成电路和大规模与超大规模集成电路4个时代。所以一般依此把电子计算机的发展也归结为4个时代，即电子管时代（约为1946~1958年）、晶体管时代（约为1958~1964年）、集成电路时代（约为1964~1971年）和大规模与超大规模集成电路时代（自1971年至今）。

计算机通常按性能、价格和体积分为巨型机、大型机、中型机、小型机和微型机5种类型。微型机或微机是微型计算机的简称，是第四代计算机产品，具有体积小、重量小、价格低、可靠性高、软件丰富、使用方便等优点。这些特点正顺应了科学技术发展的新需求，所以微型机一经问世，便备受青睐，以迅猛的势头、勃勃的生机不断改型换代，飞跃发展。如

今微型机已无孔不入地渗透到社会的各个领域和角落，已成为人们工作、生活中不可缺少的一部分。要想走在科技发展的前沿，当务之急是熟练掌握和应用微型机技术。

一、微处理器

微处理器 (Microprocessor) 是微型机的核心，它是将微型机的运算器和控制器集成起来的芯片，通常也称它为中央处理单元 CPU (Central Processing Unit)。CPU 有单片的，也有性能较高的多片型 CPU。

第一个微处理器是在 1971 年由美国 Intel 公司生产的 4 位芯片 4004，由于在市场上受到了意外的欢迎，公司很快在此基础上推出了 4 位的 4040 和 8 位的 8080 微处理器。这 3 种处理器即为第一代微处理器，它们的集成度约为 2 000 个晶体管/片，时钟频率为 1 MHz，平均指令执行时间约为 $2\mu\text{s}$ 。

1974 ~ 1978 年，8 位微处理器的性能得到完善。这期间，最具实力的微处理器生产厂家要属 Intel 公司、Motorola 公司和 Zilog 公司。典型产品如 Intel 8080/8085、MC6800、Z80 等，集成度可达 9 000 个晶体管/片，时钟频率为 2 ~ 4MHz，平均指令执行时间为 1 ~ $2\mu\text{s}$ ，各项性能指标都比第一代微处理器有较大改进，被称为第二代微处理器。

20 世纪 70 年代末到 80 年代初，以超大规模集成电路为基础的第三代微处理器产品更为引人注目，16 位微处理器得到推广和应用，性能进一步提高。Intel 8086 是最早开发成功的 16 位微处理器，集成度为 29 000 个晶体管/片。为方便原来的 8 位机用户，Intel 公司又推出了准 16 位微处理器 8088。8088 的内部数据总线为 16 位，而外部与存储器及输入/输出接口相连的数据线则为 8 位。这是一种巧妙的过渡性安排。以 8088 为 CPU 的 IBM-PC、PC/XT 系列微机一度成为世界上微机的主流机型，主导市场多年。紧接着，80286 微处理器的面市，使 16 位微处理器有了更优越的性能。以 80286 为核心的 IBM-PC / AT 成为当时的高档 16 位微机。

微处理器的前景看好，吸引着众多商家为之角逐。在激烈的市场竞争中，Intel 公司逐渐站稳脚跟，成为生产微处理器的最主要厂商。1985 ~ 1990 年，Intel 公司推出了第四代 32 位微处理器系列产品 80386、80486，处理器也由单处理器向多处理器的方向发展。1993 年 3 月，Intel 公司的微处理器 Pentium 问世，拉开了第五代 64 位微处理器大发展的序幕。而后，芯片市场开始打破 Intel 公司一家独霸的格局，世界各大芯片制造厂商接二连三地推出新产品。1995 年底，Intel 推出的 Pentium Pro 有 4 个定点处理器和两个浮点处理器，主频速度 133MHz。而 IBM、Apple 和 Motorola 合作生产的 PowerPC 604 及 Cyrix 公司、AMD 公司的 586 芯片也堪与之匹敌。微处理器在技术发展和市场需求的刺激下可谓群星荟萃，色彩纷呈，产品性能不断提高，价格不断下降，将微处理器的发展推向了前所未有的高峰，由此带动了微型计算机技术的飞速发展。

表 1-1 以 Intel 公司的产品为例，来说明微处理器在性能上的提高。

表 1-1 Intel 系列微处理器的发展

生产年代	型 号	位 数	集成度 (晶体管/片)
1971	4040	4	2 000
1974	8080	8	5 000
1975	8085	8	9 000
1978	8086	16	29 000

(续)

生产年代	型 号	位 数	集成度 (晶体管/片)
1982	80286	16	100 000
1985	80386	32	275 000
1990	80486	32	1 000 000
1993	Pentium	64	3 100 000
1995	Pentium Pro	64	5 500 000

二、微型计算机

微型计算机 (Microcomputer) 是 CPU 通过总线与存储器和输入/输出接口电路相连构成的整体。有时也称微型计算机为主机，其基本结构如图 1-1 所示。

除了核心部件 CPU 外，微型计算机还必须有存储器和输入/输出接口 (简称 I/O 接口) 两大重要组成部分。这些组成部分位于微机系统的主板上，并通过总线连接起来。

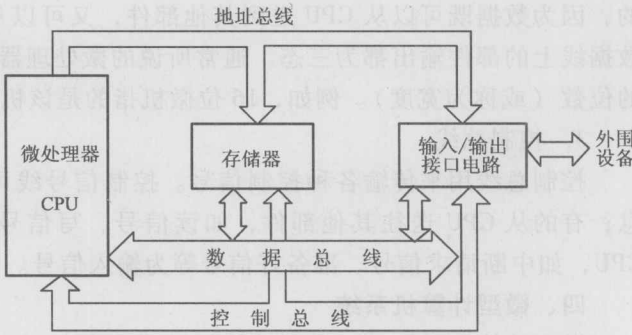


图 1-1 微型计算机的基本结构

存储器好比一个大仓库，是微机用于存放程序和数据等信息的记忆部件，能够按指定位置取出 (读出) 或者存入 (写入) 信息。存储器有内存 (主存) 和外存 (辅存) 之分。内存包括只读存储器 (ROM) 和读写存储器 (RAM)，工作速度快，存储容量较小，价格较高；内存插在微机主板的插槽上，通过总线和 CPU 直接相连，并与 CPU 交换信息。外存是内存的延伸，速度较慢，但容量大，价格低，掉电信息不丢失，可以存放经常变动或暂时不用的信息，不能直接与 CPU 打交道。要运行的程序必须首先从外存装入内存，然后才能运行。存储器容量的大小、速度的快慢是刻画微机性能的重要指标。

输入/输出接口电路是主机与外部设备相连的关键环节。外部设备 (简称外设，如磁盘、光驱、键盘、显示器、打印机等) 种类繁多，与 CPU 交换信息时往往出现速度不匹配、时序不匹配、信息格式不匹配、信息类型与电平不匹配等问题，这就通常要由 I/O 接口作为桥梁来协调解决。微机系统中除有标准的显示和键盘接口外，还提供若干个 USB 接口、RS232 串行口和并行口以连接其他外设 (如 U 盘、打印机等)。

有关存储器和 I/O 接口的详细内容将在后续章节加以介绍。

三、总线

总线 (BUS) 是微机用来传递信息的通信线，是计算机内部传递信息的主要通道。计算机内的主要部件，如 CPU、存储器和 I/O 接口等都是通过总线相互连接和交换信息。为方便功能扩展，人们在微机内定义了多种总线标准，如 ISA 总线、VESA 总线和 PCI 总线等。这些总线标准的确定，为微机系统功能扩展提供了可靠保证。一个部件只要符合某种总线标准，就可以连接到采用该总线标准的微机系统中。

各种总线标准尽管有所差异,但通常都包含3组功能不同的总线:地址总线 AB (Address Bus)、数据总线 DB (Data Bus) 和控制总线 CB (Control Bus)。

1. 地址总线

地址总线是用来传送地址信息的总线。只能从 CPU 单方向输出地址线,以寻找内存单元或 I/O 接口的地址。地址总线的位数决定了 CPU 可以直接寻址的内存空间大小。例如,16 位微机的地址总线一般为 20 位,所以最大内存容量为 2^{20} 即 1M 字节。

2. 数据总线

数据总线用来传输数据。所谓数据,在微机中是个广义概念,不同场合含义不同。它可能是真正要处理的数据,也可能是指令代码、状态变量或控制变量等信息。数据总线是双向的,因为数据既可以从 CPU 送到其他部件,又可以从其他部件送入 CPU,这就要求挂接在数据线上的部件输出都为三态。通常所说的微处理器或微型计算机的位数,就是指数据总线的位数(或称为宽度)。例如,16 位微机指的是该机的数据总线宽度为 16 位。

3. 控制总线

控制总线用来传输各种控制信号。控制信号线可能表示控制信息,也可能表示状态信息;有的从 CPU 送往其他部件,如读信号、写信号等为输出信号;有的从其他部件引入 CPU,如中断请求信号、准备好信号等为输入信号。

四、微型计算机系统

微型计算机系统是以微型计算机(即主机)为中心,配上外部设备、电源和系统软件等,能够独立工作的完整计算机。微型计算机系统的硬件主要由主机和外设两部分组成。

外部设备(简称外设)用于使计算机实现各种信息的输入和输出。外设的种类很多。输入设备是将数据、程序等信息转换成计算机能接受的信息形式,输入到计算机内部,如键盘、鼠标等。输出设备是将计算机内部的二进制信息,转换成人或设备可以接受的信息形式输出,如显示器、打印机等。还有的外设既可以作为输入设备,也可以作为输出设备来使用,如硬盘、软盘等。

系统软件是用于使用和管理计算机各个组成部分的一系列软件,它包括操作系统和各种实用程序,并为用户使用微型计算机系统提供手段。

操作系统是系统软件的核心,它负责管理计算机的所有资源,协调计算机执行的各种操作。随着微机的不断发展,微机环境下的操作系统也经历了由简单到复杂的发展过程。例如,从 8 位操作系统发展到 32 位操作系统,从字符界面到图形界面,从单任务到多任务,从单用户到多用户,这都表明操作系统功能的不断完善和增强。目前可供用户选择的操作系统很多,如各种版本的 DOS、Windows、UNIX、Linux、OS/2 及 Macintosh 等。

实用程序包括各种语言的汇编或解释、编译程序以及在此基础上编写的应用程序。应用程序包括编辑和排版软件、程序调试软件、网络及管理软件、计算机辅助设计/辅助制造/辅助教学和工具软件等。当今计算机软件极其丰富,拓宽了计算机的应用领域,计算机网络的普及和多媒体技术的发展,使计算机从商用机发展为家用电脑。

可见,微机的硬件建立了微机应用的物质基础,而计算机软件则最有效地发挥了微机的功能,为用户使用微机提供了方便、快捷和可靠的手段,硬件与软件的结合才能构成完整的微型计算机系统。

五、微机中的信息单位

微机内的各种信息均以二进制代码表示。常用的信息单位有位、字节和字等。

(1) 位 位 (bit) 是计算机所能表示的最基本的数据单元。在计算机中, 位就是一个二进制数位, 每一位只能为“0”或“1”两种状态。由若干个二进制位按一定的规则, 可以表示出各种数据、字符、指令等信息。

(2) 字节 相邻的 8 位二进制数位称为一个字节 (Byte)。

所有的程序和数据都是在计算机的硬件设备上存储和运行的。而计算机并不是随意地存储各种长度的二进制数。在以 8086 为 CPU 的微机系统中, 硬件的存储单元是一个字节。

(3) 字和字长 字 (Word) 是计算机内部进行数据处理的基本单位。计算机的每一个字所包含的二进制位数称为字长。计算机的字长一般指数据总线的宽度, 通常所说的微机是 4 位、8 位、16 位、32 位或 64 位等, 就是根据微机的字长来区分的。

在汇编语言表示中, 常把两个字节, 即 16 位二进制数规定为一个字; 把 32 位二进制数, 即两个字规定为一个双字 (Double Word)。

(4) 准 16 位机或准 32 位机 准 16 位 CPU 是指芯片内部数据总线宽度是 16 位, 而片外数据总线宽度则为 8 位的 CPU。例如, Intel 8088 就是典型的准 16 位 CPU 芯片。

同样, 芯片内部数据总线为 32 位, 片外则为 16 位的 CPU 称为准 32 位 CPU。例如 386/SX 等都为这种结构的准 32 位 CPU。

由准 16 位 CPU 或准 32 位 CPU 所构成的微型计算机称作准 16 位机或准 32 位机。

第二节 微型计算机的应用领域

随着微处理器的飞速发展, 微型计算机技术已经成为当今科学领域发展最快、应用范围最广的前沿学科。世界上计算机生产行业竞争越来越激烈, 如著名的微型机生产厂商 HP 公司、IBM 公司、APPLE 公司、DELL 公司、DEC 公司、联想电脑等, 它们在争夺市场上可谓群雄逐鹿, 各显神通, 促使微型机在性能成倍提高的同时, 价格大幅下降。微型机的品种也愈加多样化, 采用开放的硬件结构使配置更加方便、灵活; 操作系统及各类应用软件越来越丰富, 尤其是近年来计算机网络的不断应用与普及, 多媒体技术的广为流传, 使得微型机的应用达到前所未有的高潮。概括起来, 目前主要有以下几个方面的应用:

(1) 科学计算 科学计算自始至终是计算机的重要应用领域。当今的计算机具有更快和更强的数据运算能力, 在航天、军工、气象等领域得到广泛的应用。而如今许多微型机系统具有较强的运算处理能力, 由多个微处理器模块或微型计算机系统构成功能强大的系统, 其性能可与大型机相匹敌, 而成本却大大低于大型机, 大有取代大型机, 甚至巨型机的趋势。

(2) 信息处理及计算机网络 微型机配上数据库管理软件后, 可以灵活地对各种信息按要求分类、检索、存储和打印等, 尤其是运用于网络系统。计算机网络近几年发展迅速, 除本单位、本部门的局域网络外, 覆盖全国乃至世界的广域网络也相继建立, 因特网也已普及, 网络应用范围也在不断扩大。国内“三金”工程的建立更加加速了社会信息化的进程。

(3) 生产过程自动化 过程控制是计算机应用较早的领域之一。计算机在工业领域的应用改变了传统的制造方式。近几年, 计算机辅助设计 (CAD)、计算机辅助制造 (CAM) 得到长足发展, 生产过程自动化不但是生产装置的自动化, 而且发展成为计算机集成制造系统 (CIMS), 大大地提高了生产效率。

(4) 智能化仪器及装置 微型机体积小、价格低、耗电少,这使得它在仪器、仪表中得到广泛应用。现在的仪器、仪表,大多采用微处理器来代替传统的机械部件或分立的电子部件,使产品在测量精度和性能上有很大提高。在人们日常使用的各种家用电器中,现在也更多地采用了微处理器(甚至是单片微型计算机或嵌入式系统)进行控制,如手机、全自动洗衣机、电视机、电冰箱、空调等。

(5) 家用电脑 计算机多媒体技术的应用和计算机通信功能的完善,计算机性能的提高和价格的降低,使计算机正快速地走向家庭。家用电脑的普及正在替代传统的家用电器,改变着人们的生活方式。如家庭影院、家庭教育、家庭办公等正在成为现实。

第三节 计算机中的数和编码

一、数制

数制是利用符号来计数的科学方法。人类最初使用的数制是十进制,而且至今人们仍然最习惯采用十进制来计数。但十进制不是惟一的数制,在不同的场合,根据不同的需要,可以灵活地选用不同的数制。例如,二进制、十二进制、十六进制等计数方法,就是人们在长期的实践中探索出来的。

计算机是电子设备,它的电子器件只能识别电压,特别是大部分微处理器的操作都取决于电压电平的高和低两种可靠状态,这比较符合二进制的特点。因此,目前计算机中的所有信息都以二进制的代码形式来表示。同时在计算机的设计和使用,也常涉及十进制、八进制、十六进制等。

(一) 常用的数制简介

表示数制时,常用到基和权两个概念。基是数制所使用的数码个数,权表示数制每一位所具有的值。

1. 十进制 (Decimal System)

一个任意的十进制数 D 可以表示为

$$D = D_{n-1}10^{n-1} + D_{n-2}10^{n-2} + \cdots + D_110^1 + D_010^0 + D_{-1}10^{-1} + D_{-2}10^{-2} + \cdots + D_{-m}10^{-m} \\ = \sum D_i 10^i \quad (n \text{ 位整数}, m \text{ 位小数})$$

其中, D_i 为 $0 \sim 9$ 的 10 个数码中的一个。

例如,十进制数 65 498.13 可以展开为

$$6 \times 10^4 + 5 \times 10^3 + 4 \times 10^2 + 9 \times 10^1 + 8 \times 10^0 + 1 \times 10^{-1} + 3 \times 10^{-2}$$

由此看出,十进制的特点是:

- 1) 基为 10,即所用的数码为 $0 \sim 9$,共 10 个。
- 2) 各位的权是以 10 为底的幂次。
- 3) 逢 10 进位。

其他数制也具有类似的特点,只是基和权不同而已。

2. 二进制 (Binary System)

二进制的特点是:

- 1) 基为 2,即可用的数码为 0 和 1 两个。
- 2) 各位的权是以 2 为底的幂次。

3) 逢 2 进位。

一个任意的二进制数 B 可以表示为

$$B = B_{n-1}2^{n-1} + B_{n-2}2^{n-2} + \cdots + B_12^1 + B_02^0 + B_{-1}2^{-1} + B_{-2}2^{-2} + \cdots + B_{-m}2^{-m} = \sum B_i 2^i \quad (n \text{ 位整数, } m \text{ 位小数})$$

其中, n 为整数部分的位数, m 为小数部分的位数; B_i 为 0 或 1。

例如, 二进制数 10101.01 可以展开为

$$1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = 21.25$$

3. 八进制 (Octave System)

八进制的特点是:

- 1) 基为 8, 即可用的数码为 0~7, 共 8 个。
- 2) 各位的权是以 8 为底的幂次。
- 3) 逢 8 进位。

4. 十六进制 (Hexadecimal System)

十六进制的特点是:

- 1) 基为 16, 即可用 16 个数码, 包括 0~9 的 10 个数字, 另外用 A~F (或 a~f, 大、小写英文字母无区别) 表示 10~15。
- 2) 各位的权是以 16 为底的幂次。
- 3) 逢 16 进位。

在汇编语言程序设计中或平时书写时, 可以将一个常数用任意的数制来表示, 只是需要在该数后面紧跟一个英文字母, 通常为所用进制的英文单词的第一个字母, 来表明采用何种数制。二进制数后跟 B; 八进制数后跟 O 或 Q; 十进制数后跟 D, 数字后不跟任何字母也表示十进制数; 十六进制数后跟 H。

例如, 23H 表示 23 是十六进制数。

(二) 数制之间的转换

各种数制之间都可以相互转换。

1. 二进制与十进制之间的转换

(1) 二进制转换为十进制 将二进制数各位的数码乘以对应位的权, 累加之和即是等值的十进制数。

【例 1-1】 求 10001101B 的十进制数。

其办法是

$$10001101B = 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 141$$

因此, 10001101B 的十进制数是 141。

二进制小数转换为十进制小数时, 与上述方法类似, 只不过二进制小数各位的权是 2^{-1} 、 2^{-2} 、 2^{-3} 、...

以上把二进制数转换为十进制数的方法, 实际上适合于把任意进制数转换成十进制数。即把任意进制数各位的权与该位上的数码相乘, 然后求这些多项式之和就得到相应的十进制数。

【例 1-2】 求十六进制数 3AFBH 转换成十进制数。

$$3AFBH = 3 \times 16^3 + 10 \times 16^2 + 15 \times 16^1 + 11 \times 16^0 = 15\,099$$

(2) 十进制转换为二进制 十进制中整数部分可以采用“除基取余法”：

用 2 除该十进制数得到商和余数，该余数为二进制代码的最低有效位 (LSB)。再用 2 除该商，又得到新的商和余数，该余数为二进制代码的次低有效位。依此类推，直到商为 0，此时余数为二进制代码的最高有效位 (MSB)。将这些余数由高到低排列起来，就得到了该十进制数的二进制代码。

【例 1-3】 求 20 的二进制代码。

过程是：

2	20	余数
2	10	0 (LSB)
2	5	0
2	2	1
2	1	0
	0	1 (MSB)

由此得到 20 的二进制代码是 10100B。

十进制中小数部分要转换为二进制数可以采用“乘基取整法”：

用 2 乘该十进制数得到整数和小数，该整数为小数点后的第一位。再用 2 乘该小数，又得到新的整数和小数，该整数为小数点后的第二位。依此类推，直到小数为 0，此时整数为小数点后的最后一位。

【例 1-4】 求 0.125 的二进制数。

过程是：

	0.125	整数部分
×	2	
	0.250	0 (小数点后的第 1 位)
	0.25	
×	2	
	0.50	0 (小数点后的第 2 位)
	0.5	
×	2	
	1.0	1 (小数点后的第 3 位)

小数点后第一位是 0 就不用继续乘了，这时得到十进制数 0.125 的二进制数为 0.001B。

一个十进制小数转换为二进制小数，可能因永远得不到 0.0 的乘积而无法得到精确的转换值，这时可以用适当位数的近似值来表示。

一个既有整数又有小数部分的十进制数，可分别按“除基取余法”和“乘基取整法”对整数部分和小数部分加以转换，然后将转换后的二进制数合在一起，即为完整的二进制数。

【例 1-5】 求 20.125 的二进制代码。

由前面两例中分别对整数 20 和小数 0.125 进行二进制代码的转换，得到 10100B 和 0.001B，相加后得到 20.125 的二进制代码为 10100.001B。

对于十进制数与其他进制的转换，如八进制、十六进制等的转换，也可按上述方法进行，不同之处只是所用的“基”值要相应改变而已。

2. 二进制与八进制、十六进制的转换

二进制与八进制、十六进制数之间的转换十分方便,因为每3个二进制位正好对应一个八进制位;每4个二进制位对应一个十六进制位。

表1-2给出了一组十进制与二进制、八进制、十六进制的对应关系。由表可以看出,采用十六进制或八进制可以简化二进制代码的书写,易读且便于记忆。

表 1-2 几种进制数的对应关系

十进制数	二进制数	八进制数	十六进制数
0	0000	00	0 = 0 + 0
1	0001	01	1 = 1 + 0
2	0010	02	2 = 0 + 1
3	0011	03	3 = 1 + 1
4	0100	04	4 = 1 + 1
5	0101	05	5 = 1 + 1
6	0110	06	6 = 1 + 1
7	0111	07	7 = 1 + 1
8	1000	10	8 = 1 + 1
9	1001	11	9 = 1 + 1
10	1010	12	A = 1 + 1
11	1011	13	B = 1 + 1
12	1100	14	C = 1 + 1
13	1101	15	D = 1 + 1
14	1110	16	E = 1 + 1
15	1111	17	F = 1 + 1

(1) 二进制转换为八进制或十六进制 将二进制数转换为对应的八进制或十六进制数时,二进制数的整数部分由小数点向左,每3位或4位一分,最前面不足3位或4位的在前面补0,小数部分由小数点向右每3位或4位一分,最后不足3位或4位的在后面补0,然后把分好的每3位或4位二进制数用对应的八进制数或十六进制数代替即可。

【例 1-6】 将 11011010011.101100101B 转换为十六进制数。

转换过程为

0110, 1101, 0011.1011, 0010, 1000

6 D 3 . B 2 8

因此,上述二进制数对应的十六进制数是 6D3.B28H。

(2) 八进制或十六进制转换为二进制 把八进制或十六进制数的每一位用对应的3位或4位二进制数代替,即可转换为对应的二进制数。

【例 1-7】 将 9CE.1D5H 转换为二进制数。

9 C E . 1 D 5
1001 1100 1110.0001 1101 0101

所以 9CE.1D5H = 100111001110.000111010101B。

二、二进制的运算

在计算机内，一般都采用二进制来实现各种算术运算及逻辑运算。原因是二进制数的运算规则比较简单，“1”和“0”正好对应逻辑电路的高低电平，容易用电路实现。

(一) 算术运算

在 8086 指令系统中，有支持加、减、乘、除运算的算术运算指令。下面介绍一下它们的运算规则。

1. 加法

两个二进制数相加时，对应位相加。规则是

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 0 \quad \text{向高位产生进位}$$

【例 1-8】 求 $1011\text{B} + 1001\text{B} = ?$

$$\begin{array}{r} 1011 \\ +) 1001 \\ \hline 10100 \end{array}$$

求得 $1011\text{B} + 1001\text{B} = 10100\text{B}$ 。

2. 减法

两个二进制数相减时，对应位相减。规则是

$$0 - 0 = 0$$

$$0 - 1 = 1 \quad \text{向高位产生借位}$$

$$1 - 0 = 1$$

$$1 - 1 = 0$$

【例 1-9】 求 $1010\text{B} - 101\text{B} = ?$

$$\begin{array}{r} 1010 \\ -) 101 \\ \hline 0101 \end{array}$$

求得 $1010\text{B} - 101\text{B} = 101\text{B}$ 。

3. 乘法

二进制乘法与十进制乘法类似，用部分积右移、相加等操作来完成。规则是

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

【例 1-10】 求 $1011\text{B} \times 1101\text{B} = ?$

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

求得 $1011\text{B} \times 1101\text{B} = 10001111\text{B}$ 。

4. 除法

二进制除法与十进制除法类似，用上商、相减等操作来完成。

【例 1-11】 求 $11100110\text{B} \div 1010\text{B} = ?$

$$\begin{array}{r}
 10111 \\
 1010 \overline{) 11100110} \\
 \underline{1010} \\
 10001 \\
 \underline{1010} \\
 1111 \\
 \underline{1010} \\
 1010 \\
 \underline{1010} \\
 0
 \end{array}$$

求得 $11100110\text{B} \div 1010\text{B} = 10111\text{B}$ 。

(二) 逻辑运算

二进制数同样适于逻辑运算。二进制数中的 1 和 0 两个数码，可代表逻辑代数中的真和伪。两个二进制数进行逻辑运算时，对应的二进制数位分别独立运算，而与相邻位的值无关，并且运算结果只能是 0 或 1 两种数值。

1. 或运算

“或”运算的运算符为“ $\vee$ ”。规则是

$$0 \vee 0 = 0$$

$$0 \vee 1 = 1$$

$$1 \vee 0 = 1$$

$$1 \vee 1 = 1$$

例如：

$$\begin{array}{r}
 1001 \\
 \vee) 1100 \\
 \hline
 1101
 \end{array}$$

可见，进行或运算时，两者有一个为真（“1”），其结果必为真；两者都为伪（“0”），结果才为伪。此结论可推广至多变量的或运算；即各变量有一个为真，其结果必为真；都为伪时结果才为伪。

2. 与运算

“与”运算的运算符为“ $\wedge$ ”。规则是

$$0 \wedge 0 = 0$$

$$0 \wedge 1 = 0$$

$$1 \wedge 0 = 0$$

$$1 \wedge 1 = 1$$

例如：