侯朝桢　郑　链　张文政　编

微机与单片机应用基础

WEI JI YU DAN PIAN JI YING YONG JI CHU

北京理工大学出版社

微机与单片机应用基础

侯朝桢　郑链　张文政　编

北京理工大学出版社

(京)新登字149号

内容简介

本书对微型计算机及单片机的基本原理、硬件、软件及其应用中的有关问题进行了系统、深入的分析和阐述。全书共九章：概述，Z80 CPU的指令系统与时序，Z80汇编语言程序设计，半导体存贮器，输入输出与中断，微机接口芯片与电路，微机系统，微机应用，单片机及其应用等。本书选材注意理论联系实际，选用了较多的实际系统与应用实例，以8位微机为重点，兼顾16位机及单片机，每章均附有思考题与习题，适于自学。通过本书的学习，使读者在微机与单片机应用方面打下较好的基础。

本书可作为大专院校电类非计算机专业微机课程的教材或参考书，也可供从事微机与单片机开发应用的工程技术人员阅读参考。

微机与单片机应用基础
侯朝桢 郑链 张文政 编
*
北京理工大学出版社出版发行
各地新华书店经售
国防工业出版社印刷厂印刷
*
787×1092毫米 16开本 25.5印张 597千字
1992年8月第一版 1992年8月第一次印刷
ISBN 7-81013-513-9/TP·47
印数：1—10000册 定价：6.95元

前 言

近十几年来，微型计算机得到了日益广泛的应用，在国防、工业、农业、商业、文教及日常生活等领域都获得了丰硕的成果，大大促进了我国社会主义现代化建设和国民经济的发展。特别是单片机由于其结构简单、价格低廉、使用方便，发展更为迅速。

目前，有关微型计算机与单片机原理方面的书籍已出版不少。各高校也都开设了“微型计算机原理”、“单片机及其应用”等课程。但将其基本原理与应用密切结合，比较适于自学的教材或参考书尚不多见。本书是在我们多年教学经验的基础上，从微型计算机与单片机的应用出发，较深入地阐述它们的基本原理、硬件与软件技术以及在实时控制、测试技术等方面的应用。本书选材力求理论联系实际，选用了较多的实际系统与应用实例，机型以目前应用广泛、适于教学的8位微处理器Z80为重点，兼顾16位机及单片机，每章均附有思考题与习题，适于自学。通过本书的学习，使读者在微型计算机与单片机应用方面打下较好的基础。书中应用实例中有关电路和程序都通过实验调试，其中一部分是我们在科研与教学过程中研制开发的。

全书共分九章。第一章概述，主要介绍微型计算机发展过程和特点，数制与码制，并着重介绍了Z80 CPU的结构。第二章Z80的指令系统与时序，主要讨论Z80的指令格式、寻址方式、指令系统和Z80 CPU的时序。第三章Z80汇编语言程序设计，是微机软件设计的中心，针对各类典型应用程序的设计，分别举例加以说明。第四章半导体存贮器，阐述了各类半导体存贮器的工作原理、电路、与CPU的连接等问题。第五章输入输出与中断，讨论了输入输出的数据传送方式、Z80中断方式和中断控制电路。第六章微型计算机的接口芯片和接口电路，主要分析了典型的计数器/定时器电路、并行接口电路、串行接口电路、直接存贮器存取电路、数/模与模/数转换电路等。第七章微型计算机系统，讨论了微机系统的组成、标准总线、TP801单板机、IBM PC机及32位微机等。第八章微型计算机应用，阐述了微机应用的设计思路，并以交通信号、步进电机、直流电机、水处理等控制为例，详细讨论了微机应用中的硬件设计与软件开发问题。最后，第九章单片机及其应用，以MCS-51单片机为重点，分析了单片机的硬件结构，程序编制，并举例说明单片机的应用问题。

本书由侯朝桢编写第一、四、七、八、九章，郑链编写第五、六章，张文政编写第二、三章，全书由侯朝桢统稿。北京理工大学江涛教授审阅全书并提出了许多宝贵意见，北京理工大学二系、六系、八系的有关老师对本书的出版给予了热情的支持和帮助，在此一并致谢。对被引用文献的作者表示感谢。

本书可作为大专院校电类非计算机专业的微机与单片机课程的教材或参考书，参考学时为54～99学时，也可供从事微机与单片机开发应用的工程技术人员自学参考。

由于作者的水平有限，经验不足，缺点和不足之处在所难免，恳请读者批评指正。

作者 1991.12

目　录

第一章 概 述

本章的主要内容是有关微型计算机的基本概念及其工作原理。扼要介绍和讨论了微型计算机中常用的数制和码制。对于本书重点分析的 Z80 微处理器，在本章中则较详细地讨论了它的硬件结构、引脚以及主要特点。

第一节 引 言

一、微型计算机的出现和发展

微型计算机出现于 1971 年，它是计算机发展过程中的必然产物。为了说明微型计算机的出现与发展，我们先回顾一下计算机的发展历史。

一般通称的计算机指的是数字电子计算机，它是用程序控制并以数字电子技术为基础的自动计算装置。世界上第一台计算机是美国宾夕法尼亚大学 1946 年研制的电子数字积分器ENIAC。它主要是为弹道设计服务的，共用 18800 只电子管，1500 个继电器，耗电 150kW，占地 $150m^2$，重量 30t，耗资百万美元，每秒只能完成 5000 次加法运算。与今天的微型计算机相比，真是又笨又贵，只能送进博物馆展览了，但 ENIAC 毕竟是计算机的先驱，计算机的新时代即由此开始。

第一代是电子管计算机（1946年至 1958年），其逻辑元件是电子管，内存贮器为磁鼓、磁芯，外存贮器采用磁带，软件以机器语言为主，后期才采用汇编语言，其应用主要是科学计算。

第二代是晶体管计算机（1958年至 1964 年），晶体管取代电子管作为计算机的逻辑元件，内存贮器仍为磁芯，外存贮器采用了磁盘。软件出现了高级语言及其编译程序。应用方面发展到各种事务的数据处理，开始用于工业控制。

第三代是集成电路计算机（1964 年至 1971 年），其逻辑元件为小规模和中规模集成电路，即 SSI 和 MSI。软件中已出现分时操作系统，会话式语言等。小型机发展迅速，多用于企事业管理和工业控制。

第四代是大规模集成电路计算机（1971 年以后），其逻辑元件与内存贮器皆采用大规模集成电路（LSI），即单个硅片上集成 1000～20000个管子的集成电路。计算机一方面向每秒运算一千万次以上的巨型机发展，现在每秒运算十亿次的计算机已经出现；另一方面是微型计算机迅速发展并获得广泛应用，如工业、农业、商业、办公自动化和家用电器控制等，都大量使用了微型计算机。

目前，世界上包括我国在内的许多国家正在研制第五代计算机，这是以人工智能和超大规模集成电路（VLSI）为标志的新一代计算机。

如上所述，微型计算机属于第四代计算机的产品。60年代末期，空间技术、军事与民用工业的发展都迫切要求研制体积小、成本低、可靠性高的新型计算机，而大规模集成电路技术与计算机技术的飞速发展，又在技术上与生产工艺上为微型计算机的产生奠

定了基础。

1969 年，美国一家只有 12 个人的电子器件公司接到日本订货，要求设计制造新型的台式计算器。接受这项任务的年轻工程师霍夫在电子技术与计算机技术方面都有比较丰富的经验，他大胆创新，将整个系统分成四个芯片：中央处理器、读写存贮器、只读存贮器和移位寄存器，仅用四个芯片就组装成了一台计算器。这就是今天微型计算机的雏形。单片的中央处理器叫做 CPU (Central Processing Unit)，后来就发展成为微处理器。随后，在此基础上，这家公司于 1971 年推出了世界上第一片 4 位微处理器4004，很快又推出了全世界第一片 8 位微处理器 8008。这家公司也发展成了全世界最大的微型计算机芯片公司，即美国 Intel 公司。

微型计算机的发展在短短的十多年间就经历了三代的演变，现在已进入第四代。第一代（1971 年～1973 年）是 4 位机与低挡 8 位机，即数据线或运算器的位数为 4 位或 8 位，其代表产品是 Intel 4004 和 8008，芯片集成度约 1200～2000管/片，运算速度较低。第二代（1973年～ 1978 年）是中、高挡 8 位机，典型产品有 Intel 8080/8085，MC6800，MC6809，Z80，R6500 等，集成度约 5000～9000 管/片，功能较强，在各个领域获得广泛应用，现在仍为微型计算机中的主流产品。第三代（1978 年～1981 年)是 16 位机，代表产品为 Intel 8086，Z8000，MC68000，单片集成度达 20000～68000 管/片，功能很强，已达到低挡小型机水平，应用更加广泛。第四代（1981 年以后）是 32 位机，代表产品是 iAPX432，MC32μP，Intel80386，MC68020 等，集成度约 10 万～45 万管/片，主要性能相当于高挡小型机，甚至接近中型机水平，其应用领域正向高层次扩展。

从微型计算机十多年的发展过程来看，每隔 2 ～ 3 年，集成度与速度就提高一倍，而成本降低一半，销售量则增长几倍，真可说是飞速发展，迅猛异常。微型计算机的突出贡献是把计算机从高级机房的禁锢下解放出来，使非计算机专业的人员也能得心应手地使用计算机。因此它在各行各业都获得了越来越广泛的应用，几乎是无处不在。特别是微型机与当代的新技术如信息技术、自动化技术、光电技术、新材料、新工艺等密切结合，更形成了一股汹涌澎湃的技术革命的洪流，这就是通常说的新技术革命或第三次浪潮。可以毫不夸张地说，微型计算机的发展是新技术革命的奠基石和催化剂。

二、微处理器与微型计算机系统

微型计算机的类型很多，分类方法也各不相同，例如可以分别按性能、价格、应用范围、生产工艺等来区分微型计算机。这里不做详细讨论，仅对各类微型计算机中常用的一些名词术语进行简单分析与介绍。

（一）微处理器 μP 或 MP(Microprocessor)

μP 是指由一片或几片大规模集成电路组成的具有运算器与控制器的中央处理部件 CPU，即微型化的 CPU，有时也称为微处理机，也可用 MPU 表示。微处理器并不是完整的计算机，它只是微型计算机的核心部分，单独使用微处理器并不能进行计算、控制等工作。如 Z80 CPU 就是一种常用的微处理器。

（二）微型计算机 μC 或 MC(Microcomputer)

μC 是以微处理器为核心，配上存贮器与输入/输出接口电路后组成的完整的计算

机。简称为微计算机或微型机或微机。如 TP801 即为一种较简单的微型机。

（三）微型计算机系统 μCS 或 MCS(Microcomputer System)

μCS 是以微型计算机为中心，配以相应的外围设备、辅助电路及高性能的系统软件所组成的系统。简称微计算机系统、微型机系统或微机系统。如 Apple Ⅱ 即为包含主机、磁盘驱动器与打印机的较简单的微机系统。

（四）单片机（Single Chip Computer）

单片机是将微计算机（包括CPU、存贮器及输入/输出接口电路）集成在一个芯片上，即单片微型计算机，简称单片计算机、单片微机或单片机。单片机虽然功能较差，但使用方便灵活，价格便宜，特别适合于智能化仪器仪表和家用电器等较简单的控制与应用，近年来发展很快。如 Intel 8051 即为常用的一种单片机。

（五）单板机（Single Board Computer）

单板机是将 CPU、存贮器、输入/输出接口电路等组装在一块印刷电路板上，即成为单板微型计算机、单板微机或简称单板机。单板机的性能一般比单片机高，而价格仍较便宜，故在教学和自动控制等领域使用非常广泛。TP801 就是一种广泛使用的单板机。

（六）位片式微机（Bit Slice Microcomputer）

位片式微机是指用多个芯片按位组合成的微计算机，其中每个芯片仅处理 1～4 位二进制数，称为位片式微型计算机或位片式微机。这种微机一般采用双极型半导体工艺，其速度比通常的微机快几倍到十几倍。但制造困难，集成度不高，故采用位片式结构，可用于高速控制和军用系统中。如 Intel 3000 系列的 2 位位片机，AMD 公司的Am 2900 系列为 4 位位片机。

应当指出，由于微机发展迅速，应用广泛，因而对上述术语的理解与说明并不完全统一，术语之间的界限也并不十分严格。例如单板机一般也具有盒式磁带机等简单的外部设备，完全可以看作小巧的微机系统，但由于其外设非常简单，通常把它归入微型机的范畴内。本书的上述说明是目前比较通用和一致的解释，并作为本书后续章节中统一使用的术语。

三、微型计算机的特点

综上所述，微型机是微型化的计算机，它由大规模集成电路所组成，并通过程序进行控制。与其他类型的计算机（如小型机、中型机、大型机、巨型机等）相比，具有以下特点。

（1）体积小，重量轻，功耗低。微型机采用大规模集成电路，CPU 集成于一片或几个芯片内，主机一般采用单片、单板或多板式。即使是微机系统也仅为台式配置，可全部放置在一个终端桌上，比起柜式或机架式配置要轻便灵活得多。单片 CPU 的功耗在几瓦以下，整个微机系统的功耗也在几百瓦以下，因而使用环境、电源、散热等问题都较易解决。

（2）价格便宜，批量大，通用性强。单片机价格为几十至几百元，单板机为几百元至几千元，微机系统为几千元至几万元。微机采用大批量生产的通用芯片组装而成，这就更有利于降低成本。因而使中小企事业单位及较简单的设备上都可以配置微机。

（3）使用方便灵活，对环境要求低，可靠性高。由于微机的芯片与部件已系列化、标准化，并且从单片、单板到微机系统实现了多层次、多规格与多品种的产品结构，因而适应性强，从航天技术到家用电器都获得了广泛应用。与数字逻辑电路相比，它仅需改变程序（软件）即可实现不同功能，而不必修改硬件，灵活性强。由于采用大规模集成电路，元件数量与接点都较少，微机本身也较简单，因此可靠性高，对环境要求低，使计算机从机房的禁锢下解放出来。一般微机的平均故障间隔时间 MTBF 可达上万小时。

（4）功能较低，字长较短。其数据线或运算器的位数一般为 4 位、8 位、16 位，最新的超级微型机为 32 位。微机的速度较低，运算速度一般为每秒几万至几十万次。它的存贮容量也较小，一般为几千字节(一字节为 8 位二进制数)至几万字节。外部设备也不够齐全，软件也不如小型机丰富成熟。

应该说明，上述特点并不是静止不变的，关于机型的划分也不是绝对的。微型机与小型机有时就难以区分，特别是小型机微型化与微型机性能提升后，两者的界限就更加模糊。以至商品介绍中常出现“微型机的价格、小型机的功能”这种不明确的说法。不过一般认为价格与体积仍是两者分界的重要标志。至于常用的计算器与微型机的区分主要是程序设计功能，但某些微机化的计算器（如 PC1500）也有很强的程序设计功能，这种计算器实际上应归入微机范畴之内。

第二节　计算机中的数制与码制

计算机最基本的功能是对数据进行运算和处理，数在计算机中则用电平高低等物理状态来表示。计算机中一般采用二进制数，原因是它只要求两个不同的物理状态。而为了简便和直观，在对微机中的数进行分析讨论时，则常用十六进制数。本节将对数制与码制问题进行简要分析。

一、进位计数制

按进位的方法进行计数称进位计数。常用的进位计数制有以下几种。

（1）十进制是人们最习惯的进位制。它只有 0，1，2，…，9 共 10 个不同的数字；采用“逢十进一”的进位法则，即相邻位的“权”相差十倍。例如

$$123.45 = 1\times10^2 + 2\times10^1 + 3\times10^0 + 4\times10^{-1} + 5\times10^{-2}$$

一般说来，任意一个十进制数 D 为

$$D = D_{n-1}\times10^{n-1} + D_{n-2}\times10^{n-2} + \cdots + D_1\times10^1 + D_0\times10^0 + D_{-1}\times10^{-1}$$

$$+ D_{-2}\times10^{-2} + \cdots + D_{-m}\times10^{-m} = \sum_{i=-m}^{n-1} D_i\times10^i$$

其中 D_i 表示第 i 位数码，可以是 0 ～ 9 中的任一数，其值取决于数值 D；m、n 为正整数，n 为小数点左边的位数，m 为小数点右边的位数。10 为十进制数的基数（或称底数），而 10^{n-1}、$\cdots 10^0$、$\cdots 10^{-m}$ 等则分别为该位的权。

（2）二进制数是计算机中最常用的，它仅有 0 和 1 两个数码，进位法则是“逢二

进一”。如

$$(1011.101)_2=1\times2^3+0\times2^2+1\times2^1+1\times2^0+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$$
$$=(11.625)_{10}$$

其中 $(\ )_2$ 与 $(\ )_{10}$ 分别代表二进制数与十进制数。

一般的二进制数 B 可表示为

$$B=B_{n-1}\times2^{n-1}+B_{n-2}\times2^{n-2}+\cdots+B_1\times2^1+B_0\times2^0+B_{-1}\times2^{-1}$$
$$+B_{-2}\times2^{-2}+\cdots+B_{-m}\times2^{-m}=\sum_{i=-m}^{n-1}B_i\times2^i$$

其中 B_i 只能取0或1，取决于数值 B，m、n 为正整数。二进制的基数为2。

（3）十六进制数在对微机的分析与研究中常用来表示其数据，它有16个不同的数码，即0～9、A、B、C、D、E、F，其中 A～F 分别等于十进制数10～15，如表1-1所示。

表1-1 二进制、十进制与十六进制数码对照表

十进制数	十六进制数	二进制数	十进制数	十六进制数	二进制数
0	0	0 0 0 0	9	9	1 0 0 1
1	1	0 0 0 1	10	A	1 0 1 0
2	2	0 0 1 0	11	B	1 0 1 1
3	3	0 0 1 1	12	C	1 1 0 0
4	4	0 1 0 0	13	D	1 1 0 1
5	5	0 1 0 1	14	E	1 1 1 0
6	6	0 1 1 0	15	F	1 1 1 1
7	7	0 1 1 1	16	10	1 0 0 0 0
8	8	1 0 0 0			

十六进制数的进位法则为“逢十六进一”。例如

$$(2A3F)_{16}=2\times16^3+10\times16^2+3\times16^1+15\times16^0=(10815)_{10}$$

一般的16进制数 H 可表示为

$$H=H_{n-1}\times16^{n-1}+H_{n-2}\times16^{n-2}+\cdots+H_1\times16^1+H_0\times16^0$$
$$+H_{-1}\times16^{-1}+H_{-2}\times16^{-2}+\cdots+H_{-m}\times16^{-m}=\sum_{i=-m}^{n-1}H_i\times16^i$$

其中 H_i 可取0～F 间的任一值，取决于数值 H；m、n 为正整数，基数为16，故称十六进制数。

（4）一般情况，数可写作

$$K=\sum_{i=-m}^{n-1}K_iJ^i$$

其中 J 为基数；K_i 为第 i 位数码，可取0～$(J-1)$ 中的任一值，取决于数 K；J^i 为 K_i 的“权”；m、n 为正整数，n 为小数点左边位数，m 为小数点右边位数。

通常用 $(K)_J$ 表示 J 进位制数。在微机中也常用某些字母来代表不同的进位制数。

例如十进制数的代号为D（Decimal）或省略；二进制数代号为B（Binery）或%；八进制数代号为Q或O（Octal），十六进制数代号为H(Hexadecimal）或＄。

二、各种进位制数间的转换

（一）二进制数转换为十进制数

根据其定义，可将二进制数按权展开后相加，如

$$(101.001)_2 = 1\times 2^2 + 1\times 2^0 + 1\times 2^{-3} = (5.125)_{10}$$

（二）十进制数转换为二进制数

我们先以 $(37)_{10}$ 为例，分析整数的变换，即

$$(37)_{10} = (K_{n-1}\cdots K_2K_1K_0)_2 = K_{n-1}\times 2^{n-1} + \cdots + K_1\times 2^1 + K_0\times 2^0$$
$$= 2\,(K_{n-1}\times 2^{n-2} + \cdots + K_1\times 2^0) + K_0$$

等式两端同除以2，得

$$(18) + \frac{1}{2} = (K_{n-1}\times 2^{n-2} + \cdots + K_1\times 2^0) + \frac{K_0}{2}$$

等式两端括弧内的整数部分与其余的小数部分应分别相等，故 $K_0 = 1$，即37除以2后的余数为 K_0，这样就分离出 K_0。再除以2，即可分离出 K_1，依此类推，可得其余的 $K_2\cdots K_{n-1}$。整个计算过程为

$$\begin{array}{r|l}
2 & \underline{37}\ \cdots\cdots \text{余 } 1 = K_0\ \text{（最低位）} \\
2 & \underline{18}\ \cdots\cdots \text{余 } 0 = K_1 \\
2 & \underline{9}\ \cdots\cdots \text{余 } 1 = K_2 \\
2 & \underline{4}\ \cdots\cdots \text{余 } 0 = K_3 \\
2 & \underline{2}\ \cdots\cdots \text{余 } 0 = K_4 \\
 & 1\ \cdots\cdots \text{余 } 1 = K_5\ \text{（最高位）}
\end{array}$$

则 $$(37)_{10} = (K_5K_4K_3K_2K_1K_0)_2 = (100101)_2$$

我们再以 $(0.625)_{10}$ 为例分析小数的变换，如

$$(0.625)_{10} = (0.K_{-1}K_{-2}\cdots K_{-m}) = K_{-1}\times 2^{-1} + K_{-2}\times 2^{-2} + \cdots + K_{-m}\times 2^{-m}$$

等式两端同乘以2，得

$$(1 + 0.25)_{10} = K_{-1} + (K_{-2}\times 2^{-1} + \cdots + K_{-m}\times 2^{-m+1})$$

由等式两端的整数部分相等，可分离出 $K_{-1} = 1$。再乘以2，可分离出 K_{-2}，依此类推，可得其余数码。整个转换过程为

$$\begin{array}{rl}
 & 0.625 \\
\times & \underline{\qquad 2} \\
 & 1.250\cdots\cdots \text{整数部分} = 1 = K_{-1}\ \text{（最高位）} \\
 & 0.250\ \text{（取小数部分）} \\
\times & \underline{\qquad 2} \\
 & 0.500\cdots\cdots \text{整数部分} = 0 = K_{-2} \\
\times & \underline{\qquad 2} \\
 & 1.000\cdots\cdots \text{整数部分} = 1 = K_{-3}\ \text{（最低位）}
\end{array}$$

则 $$(0.625)_{10} = (0.K_{-1}K_{-2}K_{-3}) = (0.101)_2$$

有的数的计算过程会无限地进行下去，这时可根据精度要求选取适当的位数。

归纳起来，将十进制数转换为二进制数（即十换二）时，其整数部分不断除以2，每次所得余数即为二进制数码，最后余数为最高有效位的数，称除2取余法；而小数部分则不断乘以2，每次乘积的整数部分依次排列成二进制数码，称乘2取整法。

（三）十六进制与十进制间的转换

1. 十六进制换十进制　将十六进制数按权展开，如

$$(F4)_{16}=F4\mathrm{H}=15\times16^{1}+4\times16^{0}=(244)_{10}=244\mathrm{D}=244$$

2. 十进制换十六进制　与十换二类似，整数部分用除16取余法，小数部分用乘16取整法。如 $(1200.28125)_{10}=(?)_{16}$ 换算如下：

$$
\begin{array}{r|l}
16 & 1200 \cdots\cdots 余\ 0=K_0 \\
\hline
16 & 75 \cdots\cdots 余\ 11=B\mathrm{H}=K_1 \\
\hline
 & \cdots\cdots 余\ 4=K_2
\end{array}
$$

而

$$
\begin{array}{rl}
0.28125 & \\
\times\quad 16 & \\
\hline
168750 & \\
+\quad 28125 & \\
\hline
4.50000 & \cdots\cdots 整数部分=4=K_{-1} \\
0.50000 & （取小数部分） \\
\times\quad 16 & \\
\hline
8.00000 & \cdots\cdots 整数部分=8=K_{-2}
\end{array}
$$

所以

$$(1200.28125)_{10}=(4B0.48)_{16}$$

（四）十六进制与二进制间的转换

在讨论微型机的数据时，经常使用十六进制数，由于 $2^4=16$，可用4位二进制数表示1位十六进制数（如表1-1所示），这两种数制间可以很方便地进行转换。对于十六换二，可将每位十六进制数用相应的4位二进制数代替，如

$$(04B2)_{16}=(0000\quad 0100\quad 1011\quad 0010)_2=(10010110010)_2$$

反过来，对二换十六则将二进制数以小数点为基准，分别向左和向右按4位一组转换为对应的十六进制数，如

$$(1110\quad 0011\cdot1010\quad 0111)_2=(E3\cdot A7)_{16}$$

三、二进制数的运算

二进制数只有0和1两个数码，它的加、减、乘、除等运算规则与十进制数相似，并且要简单得多。

（一）加法规则

（1）$0+0=0$

（2）$0+1=1+0=1$

（3）$1+1=10=0$ 及进位1

（4）$1+1+1=11=1$ 及进位1

例如，两个二进制数1011与1101相加，其过程为

$$
\begin{array}{rl}
1111 & \text{进位} \\
1011 & \text{被加数} \\
+\ 1101 & \text{加数} \\
\hline
11000 & \text{和}
\end{array}
$$

（二）减法规则

（1） 0 － 0 ＝ 0

（2） 1 － 0 ＝ 1

（3） 1 － 1 ＝ 0

（4） 0 － 1 ＝ 1 并向相邻高位借 1

例如，二进制数 1100 减去0011，其过程为

$$
\begin{array}{rl}
1100 & \text{被减数} \\
-0011 & \text{减数} \\
\hline
1001 & \text{差}
\end{array}
$$

（三）乘法规则

（1） 0 × 0 ＝ 0

（2） 0 × 1 ＝ 0

（3） 1 × 0 ＝ 0

（4） 1 × 1 ＝ 1

除了两个 1 相乘之积为 1 外，其他情况皆为 0 。

例如

$$
\begin{array}{rl}
1101 & \text{被乘数} \\
\times 1001 & \text{乘数} \\
\hline
\left.\begin{array}{r} 1101 \\ 0000 \\ 0000 \\ +1101 \end{array}\right\} & \text{部分积} \\
\hline
1110101 & \text{乘积}
\end{array}
$$

由上例可见，乘法是将被乘数与乘数的各位相乘，并依次移位后得到部分积，再将部分积相加得乘积。当用计算机实现时，则包含了移位和相加等操作。

（四）除法规则

除法是乘法的逆运算，与十进制除法相似。从最高位开始检查，逐次增加位数，直到该位之前的所有位的值等于或大于除数，即在该位商 1 ，将此部分被除数减除数，得到余数，然后将被除数的下一位移到余数上。若余数不够减（除数）则商 0 ，将被除数的再下一位移到余数上；若余数够减则商 1 ，余数减去除数。如此反复进行，直到被除数的所有位都下移完为止。

例如

$$
\begin{array}{lrl}
 & 1101 & \text{商} \\
\text{除数}\ 110 & \overline{)\ 1001110} & \text{被除数} \\
 & -\ 110 & \\
\cline{2-2}
 & 111 & \\
 & -110 & \\
\cline{2-2}
 & 110 & \\
 & -110 & \\
\cline{2-2}
 & 0 &
\end{array}
$$

总之，二进制数的加、减、乘、除运算在计算机里可以归结为加、减、移位等操作。为了简化设备，微机中一般只设置加法器，减法则按带符号数的加法处理。下面就讨论这个问题。

四、带符号数的表示法

（一）机器数与真值

在数学中，带符号数的正、负可分别用符号“+”、“-”来表示。而计算机里的电子器件一般只有“0”和“1”两种状态，因此数的符号也可用这两种数码来表示，在数的前面增设一个符号位。正数的符号位为“0”，负数的符号位为“1”。

例如，有两个数

$$X=+1010100\mathrm{B}$$

$$Y=-1010100\mathrm{B}$$

在计算机中可以表示为

$$X_1=01010100\mathrm{B}$$

$$Y_1=11010100\mathrm{B}$$

这种数码化了的带符号数就称为机器数，而它们所代表的原来的数则称为真值。机器数一般是计算机里存贮或进行运算的数码；真值则是数学上人们习惯使用的带符号数，即机器数所代表的真正数值。上例中的 X_1、Y_1 即为机函数，而 X、Y 则分别是它们的真值。机器数常有原码、反码和补码三种表示方法。

（二）原码

如上所述，正数的符号位用0表示，负数的符号位用1表示，其余各位为数值位，即为该数的绝对值。这种数的表示方法称为原码表示法。在前述例子中，X_1 与 Y_1 即为原码表示，可写作

$$X=+1010100\mathrm{B}=+84,\ [X]_{原}=X_1=01010100\mathrm{B}$$

$$Y=-1010100\mathrm{B}=-84,\ [Y]_{原}=Y_1=11010100\mathrm{B}$$

0的原码表示可以是（+0），也可认为是（-0），即0在原码中有两种表示法

$$[+0]_{原}=000\cdots0$$

$$[-0]_{原}=100\cdots0$$

在表1-2中列出了8位二进制数的各种表示法。最左边一栏为机器数，右边四栏为

表1-2　8位二进制数的表示法

二进制数码表示	无符号二进制数	原　码	反　码	补　码
0 0 0 0 0 0 0 0	0	+0	+0	+0
0 0 0 0 0 0 0 1	1	+1	+1	+1
0 0 0 0 0 0 1 0	2	+2	+2	+2
⋮	⋮	⋮	⋮	⋮
0 1 1 1 1 1 0 1	125	+125	+125	+125
0 1 1 1 1 1 1 0	126	+126	+126	+126
0 1 1 1 1 1 1 1	127	+127	+127	+127
1 0 0 0 0 0 0 0	128	-0	-127	-128
1 0 0 0 0 0 0 1	129	-1	-126	-127
1 0 0 0 0 0 1 0	130	-2	-125	-126
⋮	⋮	⋮	⋮	⋮
1 1 1 1 1 1 0 1	253	-125	-2	-3
1 1 1 1 1 1 1 0	254	-126	-1	-2
1 1 1 1 1 1 1 1	255	-127	-0	-1

各种表示方法所对应的真值。由表可见，8位二进制数的原码表示范围为－127～＋127。

原码表示简单易懂，与真值转换方便。但同号数相减或异号数相加时要进行减法运算。为了使加减法运算在计算机里能够进行简单的统一处理，就引出了数的反码和补码表示法。

（三）反码

正数的反码表示与原码相同，最高位为符号位，此位为0表示其为正数，其余位为数值位。如

$$X=+1010100\,\mathrm{B},\ [X]_{反}=[X]_{原}=01010100\,\mathrm{B}$$

负数的反码表示其符号位为1，数值位取反，即将正数数值位中的0变1，1变0。或者说将正数的各位全部取反（包括符号位）即为负数的反码表示。如

$$X=-1010100\,\mathrm{B};\ [X]_{反}=10101011\,\mathrm{B}$$

0的反码表示有两种：

$$[+0]_{反}=000\cdots0;\qquad [-0]_{反}=111\cdots1$$

由表1-2可见，8位二进制数的反码表示范围为－127～＋127。

（四）补码

正数的补码表示与原码和反码相同，而负数的补码表示为其反码的最低位加1。如

$$X=-1010100\,\mathrm{B}$$

$$[X]_{反}=10101011\,\mathrm{B}$$

$$[X]_{补}=10101100\,\mathrm{B}$$

0的补码表示只有一种，即

$$[+0]_{补}=00\cdots0\,\mathrm{B}$$

$$[-0]_{补}=[-0]_{反}+1=11\cdots1\,\mathrm{B}+1=00\cdots0\,\mathrm{B}$$

由表1-2可见，8位二进制数的补码表示范围为－128～＋127。由于补码表示中0只占一个机器数，故其范围比原码和反码扩大了（由－127扩大为－128）。

实际上，补码的概念可通过日常生活现象来理解。如果钟表由9点拨到5点，可以倒拨四格，即9－5＝4。但有的钟表不允许倒拨，就只有顺拨7格，也指向5点，当到达12点时，即由0重新开始，相当于自动丢失了12，即

$$9+7=12\ (\text{自动丢失})+4=4$$

这个自动丢失的数（12）称为模（modulo，简写为mod），是此系统的量程或系统内所能表示的最大数。对模12来说，可写作

$$9-5=9+7\qquad(\mathrm{mod}\ 12)$$

表明等号两边同除以模12，其余数相同，即（9－5）与（9＋7）对模12是同余的。也可说（－5）与（＋7）对模12互为补数。

一般说来，若数X、Y、K满足下式

$$Y=nK+X\qquad(n\text{为整数})$$

则称Y与X对模K同余，或互为补数，也可写作

$$Y\equiv X\qquad(\mathrm{mod}\ K)$$

这就说明，当某数减去小于模之另一数时（上例为9－5），可用加上此数的补数（上例为9＋7）来代替。而补数就是模与该数的负数之和。因此，利用补数的概念和补

码表示法就可以将减法运算转换为加法运算。其推导过程如下。

设二进制数共有 n 位，其模为 2^n，负数〔$-X$〕的反码表示为

$$[-X]_{反}=\underbrace{111\cdots1}_{n\text{位}}-X=(2^n-1)-X$$

而〔$-X$〕的补码表示则为

$$[-X]_{补}=[-X]_{反}+1=2^n-1-X+1=2^n-X$$

这就说明，以 2^n 为模，$[-X]_{补}$ 与 $-X$ 互为补数，减去 X 的运算可用加上 $[-X]_{补}$ 来实现。这种以 2^n 为模的补码也简称为以 2 为模的补码，而反码可看作以 (2^n-1) 为模的补码，简称对 1 的补码。而负数用补码表示时，除了变反加 1 外，也可用 (2^n-X) 来求取。例如

$$X=01010100\,\text{B}$$

$$[-X]_{补}=2^8-X=10101100\,\text{B}$$

由于补码的加减法运算十分方便，一般微机中的带符号数除特别指明外，都采用补码表示。

五、数的定点和浮点表示

（一）数的定点表示法

当数用定点表示时，小数点的位置是固定不变的，通常将小数点固定在数值部分的最高位之前或最低位之后。前者所表示之数为纯小数，其最大值必须小于 1；后者则表示整数，在微机中经常采用，本书中若不加特别说明，都采用这种定点整数的表示方法。例如，对 8 位二进制数来说，可分别表示为无符号数（定点正整数）或带符号数（用补码表示）两种情况。例如：11001111 可认为是无符号数或带符号数，其各位的含义分别如下所示

$$\text{带符号数：}\quad \overbrace{\underbrace{1}_{\text{数符}}\;\underbrace{1001111}_{\text{数值部分}}}^{\text{无符号数的数值部分}}\quad\text{（假想的小数点位置）}$$

而小数点在计算机里是表示不出来的，仅为其假想位置。上述两种情况，可分别写作

无符号数：11001111 B = 207 D

带符号数：$[11001111]_{补}=-[00110001\,\text{B}]=-49\,\text{D}$

设定点数总位数为 n，所表示的数为 N，则对无符号数来说，其表示范围为

$$0\leqslant N\leqslant 2^n-1\quad（当\ n=8\ 时，\ 0\leqslant N\leqslant 255）$$

而带符号数的表示范围为

$$-2^{n-1}\leqslant N\leqslant 2^{n-1}-1\quad（当\ n=8\ 时，\ -128\leqslant N\leqslant 127）$$

超出此范围则为溢出（超出上限为上溢出，超出下限为下溢出）。产生溢出时，所得结果必然出错。为了保证在计算过程中不发生溢出错误，在编制程序时要人为地设置合适的比例因子。

（二）数的浮点表示法

在浮点表示的数中，小数点的位置不是固定的，而是浮动的。浮点表示的二进制数 N 可写作