

微型计算机原理及应用

杨纪春 谢文佩 主编

机械工业出版社

TP36
YJC/1

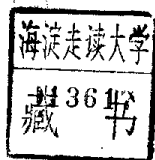
微型计算机原理及应用

主编 杨纪春 谢文佩

编者 (以姓氏笔划为序)

牛允鹏 王智相 纪志亭 沈长妹

杨纪春 李朝纯 黄秀权 谢文佩



机械工业出版社

内 容 简 介

本书是为高等院校机械、电气、仪表类等非计算机专业师生编写的微型计算机教材。本书以流行的8位机Z80为典型机种,介绍了微型计算机的结构特点,工作原理、程序设计的技巧以及接口技术。根据当前微机发展,还输入了单片机和16位微处理器的内容。

本书文笔流畅,由浅入深,通俗易懂。不但可作为高等院校,大专、职工大学教材,也可作为工程技术人员自学读物。

微型计算机原理及应用

主 编 杨纪春 谢文佩

责任编辑 李 欣

封面设计 刘 代

机械工业出版社(北京阜成门外百万庄南里一号)
(北京市书刊出版业营业许可证出字第1117号)

通县新建印刷厂印制

新华书店北京发行所发行,新华书店经售

开本 787×1092¹/₁₆·印张 17·字数 415 千字

1988年9月北京第一版,1988年9月北京第一次印刷

印数 00,001-11,050 定价:4.80元

ISBN 7-111-01229-1/TP·77

序

微型计算机的应用已经遍及各个领域,愈来愈多的大专院校学生和工程技术人员渴望学习和掌握微型计算机的基本知识和应用技术。

本教材是为尚未具有计算机专门知识的师生编写的。书中首先介绍了有关计算机的各种基本知识,包括计算机的组成、数制、和码制;然后以当前流行的8位微处理器Z80 CPU为例,详细地介绍了其结构和指令系统,以及汇编语言程序设计方法;接着介绍了组成微型计算机的各个部件,包括存储器、接口电路,进而介绍了一个典型的单板计算机的硬件电路及其监控程序。本书还介绍了各种输入输出方法, A/D、D/A 接口,使读者掌握微型计算机如何与外部电路的连接。为了适应微型计算机的迅速发展,书中还详细地介绍了新的16位微处理器和单片机。本书最后给出了几个应用实例。

本书内容广泛,叙述深入浅出。书中既包含了当前一般同类教材中有关8位机的基本内容,又有必要的计算机基础知识,还介绍了新的高档微处理器和单片机,是一本适宜于非计算机专业的院校学生使用的教材,也可作为广大工程技术人员的自学读物。

吴治衡

于吉林工业大学

1988.3.23

前 言

电子计算机的出现与发展,引起了人类社会一场新的技术革命。尤其是70年代在大规模集成电路基础上出现的微型计算机,由于它体积小、成本低,功耗小以及对环境要求低等一系列优点,使得它很快深入到了科学计算、生产过程控制、工商企业管理,甚至文化教育等领域。越来越多的非计算机专业工程技术人员开始使用微型计算机,许多高等学校的非计算机专业也相继开设了微型计算机课程。

为了适应这一形势,在原机械委教育局的关怀下,我们机械委所属部分院校的有关同志,在洛阳和合肥两次开会,商定编写一套适合理工科院校非计算机专业的计算机方面的教材,先后这本《微型计算机原理及应用》就是其中的一本。经过大家几个月的共同努力,本书终于同师生见面了。

本教材以理工科院校非计算机专业的师生为对象,同时兼顾到工程技术人员掌握微型计算机技术的自学需要。因此,本书在编写上考虑到以下几点:

1. 在开始安排了计算机基础知识一章,以减少非计算机专业人员阅读本书的困难。
2. 在基本概念及工作原理的叙述上,力求做到逻辑性强,通俗易懂。
3. 为了使同学们学完本课程后,在实际使用微机时还可以根据本书内容解决实际问题,故在内容上作了适当的扩充,并编入一些应用实例。
4. 本书以8位机Z80为典型机种。为适应当前微机的发展,也介绍了单片机和16位微处理器。

另外说明一点,目前流行的逻辑电路的画法,大家都比较习惯了,所以本书继续沿用。

本书由吉林工业大学杨纪春和合肥工业大学谢文佩主编。各章编者如下:第一章,第八章,武汉工学院李朝纯;第二章,哈尔滨科学技术大学纪志亭;第三章及第四章,谢文佩;第五章及第十一章,北京机械工业管理学院王智相;第六章,杨纪春;第七章,吉林工业大学沈长妹;第九章,安徽工学院牛允鹄;第十章,吉林工业大学黄秀权。

本书由吉林工业大学吴治衡教授主审,对我们的编写工作给予了许多的指导和帮助。在出版过程中,机械工业出版社及有关各院校教务处给予了大力支持,在此一并表示衷心的感谢!

由于编者水平有限,编写仓促,缺点和错误在所难免,恳请读者批评指正,以便在修订时更正修改。

编 者
1988年4月

目 录

序

前 言

第一章 微型计算机基础知识

§ 1-1 概述	1
§ 1-2 数制与编码	2
1.2.1 进位计数制及其特点	3
1.2.2 不同进位制之间的转换	5
1.2.3 二进制编码	7
1.2.4 数码、字母、符号的编码	8
1.2.5 计算机中带符号数的表示法	9
§ 1-3 微型计算机工作原理	12
1.3.1 电子计算机的基本结构及工作原理	12
1.3.2 微型计算机及微处理器的结构特点	14

第二章 Z80微处理器及其指令系统

§ 2-1 Z80 CPU 的内部结构	17
2.1.1 内部寄存器组	17
2.1.2 算术逻辑单元	19
2.1.3 指令寄存器IR	20
2.1.4 指令译码器	20
2.1.5 定时及控制电路	20
§ 2-2 Z80 CPU 的外形及引脚	20
2.2.1 地址总线	21
2.2.2 数据总线	21
2.2.3 控制总线	21
§ 2-3 Z80 CPU工作原理与时序	23
2.3.1 工作原理	23
2.3.2 Z80 CPU的时序	25
§ 2-4 Z80 CPU指令的格式	30
§ 2-5 Z80 CPU寻址方式	31
§ 2-6 Z80 指令系统	34
2.6.1 指令与指令系统	34
2.6.2 指令的分类	34
2.6.3 数据传送类指令	34
2.6.4 数据处理类指令	40
2.6.5 程序控制类指令	45
2.6.6 CPU控制类指令	48

第三章 汇编语言程序设计

§ 3-1 机器语言、汇编语言和高级语言	52
----------------------	----

3.1.1	机器语言	52
3.1.2	汇编语言	53
3.1.3	高级语言	53
3.1.4	三种语言的比较	54
§ 3-2	Z80 的汇编语言语法规则及伪指令	55
3.2.1	汇编语言语法规则	55
3.2.2	伪指令	56
3.2.3	包括伪指令的完整程序	59
§ 3-3	汇编语言程序设计	59
3.3.1	程序设计的步骤及流程图	60
3.3.2	直线程序	60
3.3.3	分支程序	61
3.3.4	循环程序	64
3.3.5	子程序及其调用	69
3.3.6	查表程序	75
第四章	半导体存储器	80
§ 4-1	概述	80
4.1.1	存储器的概念	80
4.1.2	同存储器有关的几个概念	80
4.1.3	存储器的分类	80
4.1.4	半导体存储器的分类	81
4.1.5	半导体存储器的组成	82
§ 4-2	读写存储器 RAM	83
4.2.1	静态存储器	83
4.2.2	动态存储器	86
§ 4-3	只读存储器 ROM	90
4.3.1	掩模式只读存储器 (ROM)	90
4.3.2	可编程只读存储器 (PROM)	91
4.3.3	可擦除的可编程序只读存储器 (EPROM)	92
§ 4-4	存储器的连接与扩充	94
4.4.1	存储器芯片的扩充	94
4.4.2	存储器同 CPU 的连接	96
4.4.3	存储器连接举例	97
第五章	输入输出与中断	102
§ 5-1	输入与输出	102
5.1.1	输入/输出端口的寻址方式	102
5.1.2	输入/输出的控制方式	103
5.1.3	Z80 CPU 的输入/输出时序	110
§ 5-2	中 断	111
5.2.1	中断概述	111
5.2.2	Z80 的中断方式	115
5.2.3	中断的优先权与中断嵌套	119
5.2.4	Z80 CPU 的中断响应时序	121

第六章 微型计算机接口电路	124
§6-1 概 述	124
§6-2 Intel 8212——8位通用输入输出接口电路	124
6.2.1 8212外形及引脚	125
6.2.2 8212内部结构和工作原理	125
6.2.3 8212应用举例	126
§6-3 可编程并行输入输出接口电路——Z80 PIO	127
6.3.1 PIO的功能	127
6.3.2 PIO的结构和外形	128
6.3.3 PIO的初始化编程	132
6.3.4 PIO同Z80 CPU的典型连接	136
6.3.5 PIO的工作时序	137
6.3.6 PIO应用举例	140
§6-4 可编程计数器/定时器电路——Z80 CTC	145
6.4.1 CTC的功能	145
6.4.2 CTC的结构和外形	145
6.4.3 CTC的初始化编程及命令字	149
6.4.4 CTC同Z80 CPU的典型连接	154
6.4.5 CTC的工作时序	154
6.4.6 CTC应用举例	155
第七章 A/D和D/A转换	160
§7-1 D/A 转换器与接口	160
7.1.1 并行D/A转换器	160
7.1.2 数字轴角转换器——步进电机	166
§7-2 A/D转换器与接口	169
7.2.1 A/D转换的过程与采样定理	169
7.2.2 A/D转换原理	171
7.2.3 A/D转换芯片及其同CPU的连接	176
第八章 典型微型计算机介绍	179
§8-1 TP801A 单板计算机的电路构成	179
§8-2 TPBUG-A 监控程序简介	185
8.2.1 TPBUG-A 监控程序的结构	185
8.2.2 TPBUG-A 的主要程序分析	186
第九章 16 位微处理器	192
§9-1 概 述	192
§9-2 Intel 8086结构	192
9.2.1 8086 概述	192
9.2.2 8086的结构特点	193
9.2.3 8086 CPU引脚	196
9.2.4 8086总线周期	196
9.2.5 8086寄存器结构	198
9.2.6 8086存储器结构	200
9.2.7 8086输入/输出结构	204

9.2.8 8086中断系统	204
9.2.9 8086的多处理器特征	207
§ 9-3 Motorola 68000	208
第十章 单片微型计算机	211
§ 10-1 概 述	211
10.1.1 单片机的组成	211
10.1.2 单片机的特点	212
10.1.3 MCS-51 系列单片机	213
§ 10-2 MCS-51系列单片机结构原理	214
10.2.1 MCS-51 系统结构	214
10.2.2 各功能部件的特点	216
10.2.3 MCS-51 引脚信号功能	219
10.2.4 CPU定时	220
§ 10-3 用8031 组成的系统举例	221
10.3.1 具有4k/2k EPROM, 2k RAM 的 8031 系统	221
10.3.2 具有2732编程器的8031系统	224
10.3.3 具有键盘、显示器的 8031 系统	224
第十一章 微机应用系统典型实例	225
§ 11-1 机床微机控制系统	225
11.1.1 车床微机控制系统的构成原理	225
11.1.2 步进电机的驱动	226
11.1.3 插补运算及其实现方法	229
§ 11-2 列车运行参数微机检测和控制系统	232
11.2.1 列车运行参数和控制要求	233
11.2.2 系统硬件设计	234
11.2.3 应用程序设计	237
附录1 Z80 指令系统表	240
附录2 ASCII 编码表(美国信息交换标准码)	264
参考文献	

第一章 微型计算机基础知识

§1-1 概 述

电子计算机是一门影响深远、发展迅速、应用广泛的多种学科的技术科学。自1946年发明直到现在的40多年间，它作为20世纪科学技术方面的卓越成就之一，有力地推动了各门科学技术的发展。它的应用深入到科学、文化、工农业生产、国防建设、财政金融及家庭事务。按照电子计算机所用的基本电子器件来划分，40多年来电子计算机大致经历了以下四个阶段：

第一代——电子管计算机（1946~1957年），其主要特征是采用电子管作为基本逻辑元件。1946年美国宾夕法尼亚大学制造的第一台计算机ENIAC，就是电子管计算机的代表。它共使用了18800个电子管，重达30 t，占房面积约170m²，耗电150 kw，但它的运算速度只有每秒5000次，价值却达40万美元。

这一代计算机，除了基本逻辑元件采用电子管以外，存储装置采用的是水银延迟电路、磁鼓。到这一代的后期，又出现了磁芯存储器。

这一代计算机的软件，还只有机器语言和汇编语言。

第二代——晶体管计算机（1958~1964年），它采用晶体管作为基本逻辑元件。这使得计算机的可靠性和速度均得到了提高。并且体积减小，重量减轻，能耗降低，成本下降。

这一代计算机所用的存储装置已基本上都是磁芯存储器。除了用磁芯作内存（又称主存储器）外，还开始使用磁鼓、磁带和磁盘作为外存储器（又称辅助存储器）。

这一代计算机所用的软件，已出现了高级语言，如FORTRAN、ALGOL 60等。还开始使用批量处理操作系统。

第三代——集成电路计算机（1965~1969年），它的基本逻辑元件是中、小规模集成电路。无论从运算速度、可靠性、体积、重量、能耗及成本等方面看，均比半导体计算机更为优越。

这一代计算机仍以磁芯存储器为内存的主流。外存储器虽然仍用磁鼓，但主要还是用磁带和磁盘。在软件方面，开始出现分时操作系统，高级语言中有了会话式语言，如BASIC语言。在程序设计方法上采用了结构化程序设计。

第四代——大规模集成电路计算机（1970年以后），它的基本逻辑元件已用大规模集成电路（每芯片1000个元件以上）取代了中、小规模集成电路。使得计算机体积更小、重量更轻、耗电更省，成本也大幅度下降。

这一代计算机在存储装置和软件方面都有较大的改进。首先用大规模集成电路的半导体存储器取代了磁芯存储器。软件上有了软件工程的概念，开始使用数据库。

微型计算机（Microcomputer）就属于第四代计算机。它的基本组成和工作原理同其它计算机没什么区别，但由于采用了大规模集成电路工艺，使它在结构上具有自己的特色。它把计算机的中央处理单元（CPU——Central Processing Unit，包括控制器、运算器及寄存器

组)集中制在一片或几片大规模集成电路芯片上,称为微处理器(Microprocessor)。

80年代初,美国和日本宣称开始了第五代计算机的研究。并宣布:第五代计算机将完全摆脱自计算机发明以来一直沿用的冯·诺依曼(John von Neumann)结构。采用新的并行结构、新的存储器结构、新的程序设计语言以及能处理符号而不只是处理数字的新的操作方式。还要求计算机能理解自然语言,具有一定智能。但是,这些研究工作还正在进行,目前还不能认为世界已进入第五代计算机时代。

微型计算机的出现,使计算机事业发生了深刻的变化。由于它在体积、重量、耗电及成本方面的巨大优势,已经使微型计算机进入办公室及家庭。在科学技术、工农业生产方面更得到了广泛的应用。从1971年第一个微处理器,Intel公司的Intel 4004诞生以来,微型计算机的发展以它的微处理器为标志也可分成四个阶段。

第一代——1971~1972年

美国Intel公司首先制成的Intel 4004四位微处理器,加上ROM、RAM及输入输出接口芯片,组成MCS-4微型计算机。它是这一代微型计算机的代表。属于这一代的还有低档的8位微处理器Intel 8008。

第二代——1973~1977年

这一代微处理器都是8位的。典型产品有Intel公司的 Intel 8080和 Motorola 公司的 M 6800,它们属于中档的8位机。

1976年至1977年,美国 Zilog 公司制成的 Z80和Intel公司制成的 Intel 8085,均属于高档的8位微处理器。

第三代——1978~1981年

采用超大规模集成电路工艺,是16位微处理器。典型产品有Intel公司于1978年制成的 Intel 8086, Zilog公司于1978年制成的 Z 8000, Motorola公司于1980年制成的 MC 68000。

第四代——1981年到现在

这一代微处理器的特点是32位,又称超级微处理器。

典型机种有Intel公司的 IAPX 432, 贝尔实验室的 MAC-32等。

目前,微型计算机正朝着高性能、低成本、多微处理器系统和网络化方向发展。

我国在微型计算机方面起步较晚。1974年开始研究,于1977年制出第一台微型计算机 DJS-050。1979年制出了 DJS-051、DJS-061,形成了 DJS-50和 DJS-60 两个系列,后来又出现了 DJS-40系列的微型计算机。目前,我国开发和成批生产了多种微型计算机系统,如与苹果机兼容的紫金 II 微型计算机,同IBM/PC机兼容的长城0520微型计算机等。各种适合我国国情的汉字化系统软件,也纷纷开发和推广,显示出广阔的发展前途。

§1-2 数制与编码

数是数值计算和数据处理的基础,数有大小、有正负、还有不同的计数制。在计算机中应采用什么计数制,如何表示数的大小、正负,这是每一个从事计算机工作的人首先需要了解的。

我们平时常用的计数制是十进制计数制,简称十进制,它有从0到9共十个数码。而电子计算机必须用二进制,它只有0和1两个数码、之所以这样,是因为电子计算机是由逻辑电路

组成的。电子电路通常只有两个稳定状态：饱和与截止，高电位与低电位等。这两种状态正好用来表示二进制的两个数码0与1。如果要表示十进制的十个数码，就十分困难，电路将十分复杂。因此，如果在计算机中采用二进制，将使计算机电路简单、成本低廉，而且运算迅速、可靠。另外，虽然计算机只处理二进制数，但在计算机的编程中，由于书写简短，而且同二进制数之间有十分简单的变换关系，还会用到八进制和十六进制数。在微型计算机中主要用十六进制数。

由于计算机只能通过电路的两个稳定状况来表示0和1，所以对于数的正负、大小及运算，就有它独特的用0和1组成的表示方法。另外，一些字母、符号等，计算机也要用0和1的组合才能表示。我们把这些表示方法统称为编码。

1.2.1 进位计数制及其特点

数制包括进位计数制和非进位计数制两种。非进位计数制的特点是，数码所代表的数值，同它在数中的位置无关。如罗马数字中的 I 总代表1，V 总代表5，X 总代表10，C 总代表100等。进位计数制的特点是，每个数码表示的数值，不仅取决于数码本身，而且还取决于该数码在数中的位置。如十进制的数5453.25中，左起第一个5和第三位的5，以及小数点后第二位的5，虽然都是用数码5来表示，但前者代表的值是5000 (5×10^3)，后者表示的则是50 (5×10^1)，而最低位的5所代表的值则为0.05 (5×10^{-2})。

非进位计数制由于表数不便，运算困难，已经不用。下面就电子计算机所常用的进位制分别加以讨论。

一、十进制 (Decimal notation)

十进制有以下特点：①有十个数码：0, 1, 2, 3, 4, 5, 6, 7, 8, 9；②逢十进一，借一当十。亦即相邻两位相同数码代表的值互为十倍关系。如5453.25这数，可以写成如下形式：

$$5453.25 = 5 \times 10^3 + 4 \times 10^2 + 5 \times 10^1 + 3 \times 10^0 + 2 \times 10^{-1} + 5 \times 10^{-2}$$

这称作十进制数5453.25的按权展开式。对于任何一个十进制数，可以用下列一般式来表示：

$$\begin{aligned} D &= D_{n-1} \cdot 10^{n-1} + D_{n-2} \cdot 10^{n-2} + \cdots + D_1 \cdot 10^1 + D_0 \cdot 10^0 + D_{-1} \cdot 10^{-1} + \cdots + D_{-m} \cdot 10^{-m} \\ &= \sum_{i=n-1}^{-m} D_i \cdot 10^i \end{aligned}$$

二、二进制 (Binary notation)

二进制的特点是：①有两个数码：0和1；②逢二进一，借一当二。如二进制数 111011.101 可以写成如下形式：

$$\begin{aligned} (111011.101)_2 &= 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} \\ &= 32 + 16 + 8 + 2 + 1 + 0.5 + 0.125 \\ &= (59.625)_{10} \end{aligned}$$

对于任何一个二进制数，可以写成下列一般表达式：

$$B = B_{n-1} \cdot 2^{n-1} + B_{n-2} \cdot 2^{n-2} + \cdots + B_1 \cdot 2^1 + B_0 \cdot 2^0 + B_{-1} \cdot 2^{-1} + \cdots + B_{-m} \cdot 2^{-m}$$

$$= \sum_{i=n-1}^{-m} B_i \cdot 2^i$$

三、八进制 (Octal notation)

特点: ①有八个数码: 0, 1, 2, 3, 4, 5, 6, 7; ②逢八进一, 借一当八。如:

$$(327)_8 = 3 \cdot 8^2 + 2 \cdot 8^1 + 7 \cdot 8^0 = 192 + 16 + 7 = (215)_{10}$$

一般表达式为:

$$\begin{aligned} Q &= Q_{n-1} \cdot 8^{n-1} + Q_{n-2} \cdot 8^{n-2} + \dots + Q_1 \cdot 8^1 + Q_0 \cdot 8^0 + Q_{-1} \cdot 8^{-1} + \dots + Q_{-m} \cdot 8^{-m} \\ &= \sum_{i=n-1}^{-m} Q_i \cdot 8^i \end{aligned}$$

四、十六进制 (Hexadecimal notation)

特点: ①有十六个数码: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F。其中A, B, C, D, E, F, 分别为十进制数的10, 11, 12, 13, 14, 15; ②逢十六进一, 借一当十六。如:

$$(327)_{16} = 3 \cdot 16^2 + 2 \cdot 16^1 + 7 \cdot 16^0 = 768 + 32 + 7 = (807)_{10}$$

一般式为:

$$\begin{aligned} H &= H_{n-1} \cdot 16^{n-1} + H_{n-2} \cdot 16^{n-2} + \dots + H_1 \cdot 16^1 + H_0 \cdot 16^0 + H_{-1} \cdot 16^{-1} + \dots \\ &\quad + H_{-m} \cdot 16^{-m} \\ &= \sum_{i=n-1}^{-m} H_i \cdot 16^i \end{aligned}$$

表 1-1 各种进制数的对应关系

十 进 制	二 进 制	八 进 制	十六进制
0	0 0 0 0	0	0
1	0 0 0 1	1	1
2	0 0 1 0	2	2
3	0 0 1 1	3	3
4	0 1 0 0	4	4
5	0 1 0 1	5	5
6	0 1 1 0	6	6
7	0 1 1 1	7	7
8	1 0 0 0	10	8
9	1 0 0 1	11	9
10	1 0 1 0	12	A
11	1 0 1 1	13	B
12	1 1 0 0	14	C
13	1 1 0 1	15	D
14	1 1 1 0	16	E
15	1 1 1 1	17	F

我们可以把各种进制制用统一的一般表达式来表示:

$$N = N_{n-1} \cdot P^{n-1} + N_{n-2} \cdot P^{n-2} + \dots + N_1 \cdot P^1 + N_0 \cdot P^0 + N_{-1} \cdot P^{-1} + \dots + N_{-m} \cdot P^{-m}$$

$$= \sum_{i=n-1}^{-m} N_i \cdot P^i$$

式中: N_i ——第 i 位的数码(系数), 进位制不同, 数码的个数不同;

P ——进位基数, 即数码的个数;

P^i ——权;

n ——整数部份位数, 为正整数;

m ——小数部份位数, 为正整数。

二进制数、八进制数、十六进制数同十进制数的对应关系见表1-1。

1.2.2 不同进位制之间的转换

一、二进制数与十进制数的转换

1. 二进制数转换成十进制数

二进制数转换成十进制数较方便, 只需按权展开, 然后相加即可。例:

$$\begin{aligned}(1101.01)_2 &= 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} \\ &= 8 + 4 + 0 + 1 + 0 + 0.25 \\ &= (13.25)_{10}\end{aligned}$$

2. 十进制数转换成二进制数

整数部分和小数部分要分别用不同的方法。

(1) 整数部分的转换——除2取余法

将十进制数反复除以2, 取其余数作为相应二进制数的最低位 K_0 , 其次为 K_1 , 直到最后一次相除商为0时得到最高位 K_{n-1} , 则 $K_{n-1}K_{n-2}K_{n-3} \cdots K_1K_0$ 即为转换所得的二进制数。

如: 将 $(205)_{10}$ 转换成二进制数

2 205	余1..... K_0
2 102	余0..... K_1
2 51	余1..... K_2
2 25	余1..... K_3
2 12	余0..... K_4
2 6	余0..... K_5
2 3	余1..... K_6
2 1	余1..... K_7

$$\text{即 } (205)_{10} = (K_7 K_6 K_5 K_4 K_3 K_2 K_1 K_0)_2 = (11001101)_2$$

(2) 小数部分的转换——乘2取整法

将十进制小数乘2, 取乘积的整数部分作为相应二进制数小数点后最高位 K_{-1} , 反复乘2, 逐次得到 $K_{-1}, K_{-2} \cdots K_{-m}$ 。直到积的小数部分为0或小数点后的位数达到精度要求为止。

如: 将 $(0.625)_{10}$ 转换为二进制数

$$\begin{array}{r}
 0.625 \\
 \times 2 \\
 \hline
 1.250 \quad \text{整数} 1 \cdots \cdots K_{-1} \\
 0.250 \\
 \times 2 \\
 \hline
 0.500 \quad \text{整数} 0 \cdots \cdots K_{-2} \\
 \times 2 \\
 \hline
 1.000 \quad \text{整数} 1 \cdots \cdots K_{-3}
 \end{array}$$

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

在实际转换中，不是任意十进制小数都能用有限位二进制数精确表示的，此时可按精度要求取足够的位数。

对于既有整数又有小数部分的十进制数，可以对它的整数部分和小数部分分别转换成二进制数，再把两者连接起来。如：

将 $(25.25)_{10}$ 转换成二进制数

$$\begin{array}{r}
 2 \overline{) 25} \quad 1 \\
 \underline{2 12} \quad 0 \\
 2 \overline{) 6} \quad 0 \\
 2 \overline{) 3} \quad 1 \\
 2 \overline{) 1} \quad 1 \\
 0
 \end{array}
 \quad
 \begin{array}{r}
 0.25 \\
 \times 2 \\
 \hline
 0.50 \quad 0 \\
 \times 2 \\
 \hline
 1.00 \quad 1
 \end{array}$$

$$\text{即 } (25.25)_{10} = (11001.01)_2$$

二、任意进位制数与十进制数的转换

上面的除2和乘2，实际上就是除进位基数P和乘进位基数P。因此可以仿照二进制，采用“按权展开相加”、“除P取余”、“乘P取整”的办法来实现任意进位制数同十进制数的转换。

关于以上方法的证明这里从略，读者若有兴趣，可以参阅有关的数字逻辑方面的教材。

三、二进制数与八进制数之间的转换

从表1-1可知，二进制数第四位的权是 $2^3=8$ ，而八进制数第二位的权是 $8^1=8$ ，因此，二进制的三位数对于八进制的一位数。当二进制数从第三位向第四位进位时，即 $0111 \rightarrow 1000$ ，八进制数相当于从最低位向第二位进位，即 $7 \rightarrow 10$ 。两种进位制数之间的转换是十分简便的。

1. 二进制数转换成八进制数

以小数点为基准，整数部分从右至左，每三位一组，最高有效位不足三位时，添0补足三位；小数部分从左至右，每三位一组，最低有效位不足三位时，添0补足三位。然后，将各组的三位二进制数按 2^2 ， 2^1 ， 2^0 权展开后相加，得到相应的一位八进制数。两者的对应关系见表1-1。

例 将 10111011.00110101 转换为八进制数

$$\begin{array}{ccccccc}
 \underline{010}, & \underline{111}, & \underline{011}, & \underline{001} & \underline{101} & \underline{010} \\
 2 & 7 & 3 & 1 & 5 & 2
 \end{array}$$

$$\therefore (10111011.00110101)_2 = (273.152)_8$$

2. 八进制数转换成二进制数

道理同上，把一位的八进制码写成对应的三位二进制码，并按权连接起来即可。如：

$$(27.461)_8 = (010 \ 111.10 \ 0110 \ 001)_2$$

四、 二进制数与十六进制数之间的转换

二进制数与十六进制数之间也存在着二进制同八进制之间相似的关系： $2^4=16^1$ 。从表1-1可知，二进制的四位数组对应于十六进制的一位数。

1. 二进制数转成十六进制数

以小数点为基准，整数部分从右往左，小数部分从左往右，每四位一组，最后不足四位时，添0补足。然后，把每组的四位二进制数码按权展开后相加，得到相应的一位十六进制数码，按权的顺序连接起来即得相应的十六进制数。

例，将110111110.100101111转换成十六进制数

0001 1011 1110. 1001 0111 1000

1 B E 9 7 8

$\therefore (110111110.100101111)_2 = (1BE.978)_{16}$

2. 十六进制数转换成二进制数

同八进制数的情况相似，但一位十六进制数码对应转换成四位二进制数码，然后按权连接起来。

例，(6AB.7A54)₁₆

0110 1010 1011.0111 1010 0101 0100

$\therefore (6AB.7A54)_{16} = (11010101011.01111010010100)_2$

从上面的讨论可以知道，八进制、十六进制数同二进制数之间有着十分简便的转换关系，而且八进制、尤其是十六进制的书式十分简短。因而在程序设计中，二进制的代码往往书写成八进制或十六进制形式。特别是十六进制，计算机的数据往往是8位或16位的二进制数，可以表示成两位或者四位的十六进制数码，因而得到广泛的应用。

在程序设计中，各种进制数的数还采用下面的表示法：

十进制数 在数字后面加字母D或不加字母，如325D或325；

二进制数 在数字后面加字母B，如1011B；

八进制数 在数字后面加字母Q，如427Q；

十六进制数 在数字后加字母H，如3ABH。

1-2-3 二进制编码

计算机只能处理由0和1组成的信息，但我们平时都习惯于处理十进制数，在送入计算机之前要转换成二进制数，这一过程如果由计算机自动完成，就方便了。为此，必须要使计算机能识别十进制数。针对这一目的，我们用0和1对十进制数进行编码，即用0和1的二进制形式来表示十进制数，称为十进制数的二进制编码。这里我们仅介绍一种编码方法——BCD码(Binary Coded Decimal)。

BCD码是用四位二进制编码来表示一位十进制数。它仅仅是形式上变成了由0和1组成的二进制形式，而实质上运算规则及数值都是十进制的。由于四位二进制码的权分别为8、4、2、1，所以又称它为8421码。它同十进制数的对应关系见表1-2。

从表1-2可看出，用四位二进制编码来表示十进制数，用BCD码是十分直观和方便的，它有以下特点：

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

(1) 编码有权, 权为8、4、2、1。

(2) 采用BCD码进行运算时要进行修正。

四位二进制编码可以有16种不同的组合, 分别表示十进制数0~15共16个数。而BCD码只要求表示十进制数0~9这十个数码, 只需要0000~1001这十组码就够了。1010~1111这六组码就是多余的, 对于BCD码来讲, 就是非法代码。在对BCD码进行十进制运算时, 由于计算机只有二进制的运算指令, 只能按二进制的规律进行运算。这样, 运算的结果势必出现非法代码。针对这一点, 必须采取相应的修正措施, 使运算符合十进制的规律。Z80的指令系统中就设置了一条十进制调整指令DAA, 以解决这个问题。关于DAA指令, 将在第二章中讲述。

1.2.4 数码、字母、符号的编码

计算机除了输入和处理十进制数外, 还会输入程序和各种命令, 里面包括许多数码、字母和符号。如果不用0和1进行编码, 计算机也是无法识别的。这些计算机通过编码能识别的数字、字母和符号, 我们通称为字符。

目前微型计算机普遍采用的字符编码方法是ASCII码(American Standard Code for Information Interchange)——美国信息交换标准码, 它用七位二进制码表示一个字符。由于 $2^7=128$, 所以, 共有128种不同的组合, 可以表示128个不同的字符, 包括: 数码0~9, 26个大写英文字母, 26个小写英文字母以及各种运算符号、标点符号及控制命令等。ASCII码的编码表见附录。

二进制的一位, 称为1 bit。而8个bit称为1个字节(Byte)。计算机能处理的最大位数称为计算机的字长(Word length)。显然, 字长越长, 精度越高, 但它受到计算机硬件的限制。数据一般以字节为单位处理, 因而ASCII码占 $D_7 \sim D_0$ 七位后, 空闲的 D_7 可用作奇偶校验位。

所谓奇偶校验, 是在代码传送过程中, 用以检验是否出现错误的方法。分奇校验和偶校验两种。奇校验, 规定正确的代码, 一个字节中1的个数必是奇数。若非奇数, 则用最高位 $B_7=1$ 来满足。在接收端若收到一个字节中1的个数不为奇数的代码, 就知道出错了。偶校验, 则规定正确的代码一个字节中1的个数必须是偶数, 其它方面同奇校验。