

微型计算机 硬件软件及其应用

周明德 编著

清华大学出版社

微 型 计 算 机

硬 件 软 件 及 其 应 用

周 明 德 编 著

清 华 大 学 出 版 社

1982年

内 容 简 介

本书共分十一章，先由一个简单的模型机入手，由浅入深地阐明了计算机的概念和原理；然后详细地介绍和分析了 Z80 的指令系统，通过大量的例子着重讲解了汇编语言；分析了 Z80 的时序；分别介绍了各种类型的半导体存储器工作原理、以及与 CPU 的接口及有关的问题、Z80 的输入输出指令、CPU 与外设交换信息的几种方法、着重介绍了中断的概念和 Z80 的中断方式及特点；也介绍了 8255A、Z80-CTC、Z80-PIO 等并行接口片子及它们的应用；还介绍了串行通信技术和 8251、5501 等串行接口电路及它们的应用；以 CROMEMCO 系统和 Z80 STARTER KIT 单板机为例分析和介绍了微型机的监控和调试程序；最后以 CROMEMCO 系统为例，阐述了系统的组成和连接（S-100 总线）、几块主要电路板的原理、以及以 CDOS 为例分析了磁盘操作系统及其对系统的管理。

本书是清华大学自动化系“计算机原理”课和“微型计算机”课的教材。本书还集有对大量国外的资料、手册和近几年出版的微型机的书籍进行提炼、概括、归纳和整理的资料，对微型计算机的原理、硬件、软件和应用作了深入和全面的分析，介绍了大量经过检验的应用程序，使本书在选材上具有先进性、系统性和实用性。所以本书可作为具有数字电路基础的科技人员学习和应用微型计算机的自学教材，以及作为为科技人员举办的微型机训练班的教材。

微 型 计 算 机

硬件、软件及其应用

周明德 编著

☆

清华大学出版社出版

北京 海淀 清华园

北京景山学校印刷厂印装

新华书店北京发行所发行·各地新华书店经售

开本：787×1092 1/16 印张：31 字数：774（千字）

1982年10月第一版

1982年11月第一次印刷

印数：1~70,000

统一书号：15235·44

定价：3.20 元

前 言

近年来, 计算机技术已取得惊人的迅猛发展, 使得计算机渗透到国防尖端、工业、农业、企业管理、日常生活的各个领域, 其作用和成就正日益卓著, 成了工业发展水平的标志之一, 是发展新技术、改造老技术的强有力武器。计算机的生产、推广和应用已成为我国四个现代化的战略产业。

自1946年第一台电子计算机设计和运行以后, 计算机主要是朝着大型和快速的方向发展。但是, 70年代以来, 由于大规模集成电路技术的发展, 微型计算机异军突起, 发展更为迅速。其CPU集成度几乎是每两年翻一翻, 且性能增长一个数量级。例如: 1971—1972出现的Intel4004和4040 (4位), 其集成度为2000晶体管/片; 在1976年生产的8085 (8位), 集成度为9000晶体管/片; 而1980年生产的*iAPX43201* (32位), 集成度则为100000晶体管/片。各种微型机的年产量达到了几十万台。

如果说大型计算机在国防尖端和科学研究中起过巨大的作用的话, 那么微型计算机 (简称微型机) 由于它成本低、体积小才真正使计算机能渗透和占领各个技术领域, 为计算机应用的普及和推广开创了现实的可能性。

例如, 在工业控制中, 60年代发展起来的各种类型的数控机床, 数控部分的成本往往要比机床的成本高一个数量级, 而且体积庞大。因而只有在某些不用数控机床无法加工的情况下才得到了应用和发展。而现在就可采用微型机——通常可用一个单板微型计算机 (简称单板机) 来实现控制, 成本只要几千元, 因而大大扩大了其应用范围, 不光在国防工业、重工业、在许多轻纺工业中, 例如织袜机, 织帽机, 编织毛衣机等控制中已得到了应用。而且出现微型机后, 就可以利用单板机来实现单机控制; 用功能强一点的微型机做上一级的控制; 管理和控制各个单板机以及它们之间的通讯。这样的分级分布控制系统就性能好、成本低, 为生产线以及整个车间的自动化提供了较现实的可能性。

把微型计算机与测量仪器、仪表联接起来, 可以存贮测量的信息, 可以对测量结果进行分析、综合和作各种处理, 大大提高测量效率, 实现测量的自动化。

利用微型机实现仓库管理、旅馆管理、飞机订票等信息处理和管理工作, 为计算机在管理中的应用开创了非常广阔的前景……。

微型机本身仍在迅速发展, 日新月异。现在一台微型机的功能, 不仅超过了50年代初期占地上百平方米, 功耗上百千瓦的庞大的电子管计算机, 而且也赶上和超过了60年代的小型机的功能。国外还在大力研究微型机网络, 企望在功能上能与大型机和巨型机相匹敌, 而成本却要便宜得多。所以, 微型计算机是计算机发展的一个重要的方面军。

引进、推广和应用微型计算机, 开展微型计算机的科研和生产工作, 是我国技术改造的一个经济而有效的途径。近年来, 我国微型机的科研、生产和应用也有了迅速的发展。因此, 在各个技术领域的科技人员中普及和提高微型计算机的知识, 使微型机在各个行业中得到广泛应用和充分发挥它的效用就成了当务之急。

本书是作者在清华大学自动化系讲授“计算机原理”课和“微型计算机”课的教材, 今也是为顾及具有数字电路基础的工程科技人员学习和应用微型计算机而编写的。在编写时我

们立足于适应一般科技人员，即是从具有一定的数字电路的基础知识，但还不了解计算机而需要学习和掌握计算机，或要把计算机引入、应用到各个工程科技领域的科技人员的角度出发，通过微型计算机介绍计算机的原理、结构和特点，着重对微型计算机的原理、硬件和软件作了全面和系统的分析。

本书在编写时是着重从应用的角度来分析和解决问题的。在硬件部分，本书不是注重于对计算机内部电路的分析和设计，而着重于了解和掌握这些电路的功能，着重于它们的应用。着重介绍利用微型机解决各种工程技术问题所需掌握的知识，所需解决的接口技术和接口电路。在软件部分，本书着重介绍了掌握和应用微型机的基本工具——汇编语言，掌握和应用微型机所需要的系统软件，特别是介绍了大量的经过检验的应用程序。

本书是一本教材，所以对大量的国外资料、手册和近几年出版的微型机的书籍进行了提炼、概括、归纳和整理。注意了选材上的先进性、系统性和完整性。同时从认识规律出发，尽可能做到深入浅出，循序渐进。所以，本书也可作为具有数字电路基础的工程科技人员学习和应用微型机的自学教材，也可作为各种为工程科技人员举办的微型机训练班的教材。

本书承蒙清华大学计算机系朱家维同志进行了全面而详细的审核与校对，提出了大量极其宝贵的意见；计算机学院的林定基同志也对本书提出了许多宝贵的意见。在此，表示衷心的感谢。

由于编者水平有限，时间又很仓促，书中一定存在不少缺点和错误，敬请读者批评指正。

周明德写于清华大学

1981年9月

目 录

前言	(1)
第一章 概述	(1)
第一节 引言	(1)
第二节 计算机中的数和编码系统	(1)
一、进位计数制	(2)
二、进位制数之间的转换	(4)
三、二进制编码	(7)
四、二进制数的运算	(8)
五、带符号数的表示法	(12)
第三节 计算机基础	(16)
一、计算机的基本结构	(16)
二、指令程序和指令系统	(17)
三、初级计算机	(19)
四、简单程序举例	(21)
五、寻址方式	(25)
六、分支	(30)
七、程序举例	(34)
第四节 计算机的硬件和软件	(38)
一、系统软件	(38)
二、应用软件	(39)
三、数据库 (Data Base) 及数据库管理系统	(39)
第五节 微型计算机的结构特点	(40)
一、微型机的外部结构特点	(40)
二、微型机的内部结构特点	(41)
第六节 Z80 的 CPU 结构	(42)
一、Z80 的内部结构	(43)
二、Z80 引脚及其功能	(44)
第二章 Z80 的指令系统和汇编语言程序设计	(47)
第一节 Z80 的寻址方式	(47)
一、立即寻址	(47)
二、立即扩展寻址	(47)
三、寄存器寻址	(48)
四、扩展寻址	(48)
五、寄存器间接寻址	(48)
六、变址寻址	(49)
七、零页寻址	(49)
八、相对寻址	(50)
九、位寻址	(50)
十、隐含寻址	(51)
第二节 Z80 的指令系统	(51)
一、数的传送和互换	(51)
二、数据块传送和搜索指令	(83)
三、算术和逻辑指令	(66)

四、循环和移位指令	(75)
五、位操作指令	(80)
六、转移指令	(81)
七、子程序调用和返回指令	(83)
第三节 汇编语言程序设计和实例	(86)
一、机器语言、汇编语言和高级语言	(86)
二、汇编语言源程序的格式	(88)
三、伪指令	(89)
四、程序举例	(91)
第四节 宏指令和条件汇编	(104)
一、宏定义和宏调用	(104)
二、条件汇编	(111)
第五节 汇编程序	(116)
一、概述	(116)
二、两次扫描的汇编程序	(119)
第三章 Z80-CPU 的时序	(126)
第一节 概述	(126)
一、指令周期、机器周期和 T 周期	(126)
二、CPU 的时序和存储器以及外设的时序	(127)
三、学习 CPU 时序的目的	(128)
第二节 Z80 的典型时序分析	(129)
一、取指令码 (M_1 周期)	(129)
二、存储器读或写周期	(130)
三、输入或输出周期	(132)
四、总线请求和响应周期	(133)
五、中断请求和响应周期	(134)
六、非屏蔽中断响应	(136)
七、暂停状态的脱离	(136)
第四章 半导体存储器	(138)
第一节 半导体存储器的分类	(138)
一、RAM 的种类	(139)
二、ROM 的种类	(139)
第二节 读写存储器 RAM	(140)
一、基本存储电路	(140)
二、RAM 的结构	(143)
三、RAM 与 CPU 的连接	(146)
第三节 只读存储器	(168)
一、掩模只读存储器	(168)
二、可编程的只读存储器	(170)
三、可擦去的 PROM-EPROM	(171)
第五章 输入和输出	(176)
第一节 输入输出的寻址方式	(176)
第二节 Z80 的输入输出指令和时序	(177)
一、直接寻址的 I/O 指令	(177)
二、用寄存器 C 间址的 I/O 指令	(177)
三、数据块输入或输出指令	(178)

四、Z80 CPU I/O 时序.....	(179)
第三节 CPU 与外设数据传送的方式.....	(179)
一、CPU 与 I/O 之间的接口信号	(179)
二、无条件传送方式	(180)
三、查询传送方式	(182)
四、中断传送方式	(186)
五、直接数据通道传送——DMA	(186)
第四节 用 8212 作为一个输入输出接口	(189)
一、8212 介绍	(189)
二、用 8212 作为 CPU 与纸带读入机的接口.....	(190)
第六章 中断.....	(193)
第一节 引言	(193)
一、为什么要用中断	(193)
二、中断源	(193)
三、中断系统的功能	(194)
第二节 最简单的中断情况	(194)
一、CPU 响应中断的条件.....	(194)
二、CPU 对中断的响应.....	(196)
第三节 矢量中断.....	(197)
一、RST 指令.....	(197)
二、RST 指令的时序.....	(197)
三、RST 指令的形成.....	(198)
第四节 中断优先权	(199)
一、用软件确定中断优先权	(199)
二、硬件优先权排队电路	(200)
三、一个例子	(202)
第五节 Z80 的中断方式	(208)
一、非屏蔽中断和屏蔽中断	(208)
二、屏蔽中断模式 0	(210)
三、屏蔽中断模式 1	(210)
四、屏蔽中断模式 2	(211)
第六节 Z80 中的优先权排队电路.....	(213)
一、链形优先权结构	(213)
二、屏蔽中断时序	(214)
三、中断嵌套	(217)
四、Z80 中断控制逻辑	(220)
第七章 并行接口片子.....	(222)
第一节 可编程的输入输出接口 8255A	(222)
一、8255A 的结构	(222)
二、方式选择	(224)
三、方式 0 的功能和应用举例	(228)
四、方式 1 的功能和应用举例	(234)
五、方式 2 的功能和应用举例	(244)
第二节 Z80—CTC	(254)
一、概述	(254)
二、CTC 工作方式和编程.....	(257)
三、CTC 时序.....	(259)

四、CTC 中断	(261)
五、CTC 应用举例	(263)
第三节 Z80—PIO	(265)
一、概述	(265)
二、PIO 编程	(268)
三、PIO 时序	(271)
四、应用举例	(274)
第八章 数/模和模/数转换	(278)
第一节 D/A 转换器	(278)
一、CPU 与 D/A 的接口	(278)
二、CPU 与 10 位 D/A 转换器的接口	(286)
第二节 A/D 转换	(288)
一、概述	(288)
二、用软件实现 A/D 转换	(291)
三、A/D 转换片子与 CPU 的接口	(294)
第三节 用 A/D 转换构成的数据采集系统	(296)
一、数据的采集	(296)
二、定时	(297)
三、数据的输出	(301)
第九章 串行通讯及接口电路	(308)
第一节 串行通讯	(308)
一、概述	(308)
二、串行传送中的几个问题	(310)
三、串行 I/O 的实现	(315)
第二节 串行接口电路	(319)
一、概述	(319)
二、Intel 8251A 可编程通讯接口	(320)
三、TMS 5501 多功能输入输出控制器	(328)
第三节 串行通讯应用举例	(336)
一、磁带记录的标准	(337)
二、接口电路	(338)
三、信息由 CPU 写入磁带的软件	(339)
四、从磁带读入数据的软件	(344)
第十章 监控和调试程序	(349)
第一节 监控和调试程序简介	(349)
一、RDOS 的功能简介	(349)
二、Z80 STARTER KIT 监控调试程序简介	(352)
第二节 ZBUG 的几个主要程序分析	(355)
一、键盘输入程序	(355)
二、显示程序	(362)
三、初始引导程序	(364)
四、检查和修改存储器内容	(365)
五、显示和修改寄存器内容	(368)
六、设置断点	(372)
七、单步程序	(373)
八、非屏蔽中断服务程序	(374)
九、执行键处理	(377)

第十一章 微型计算机系统介绍	(381)
第一节 CROMEMCO 系统Ⅲ的组成	(381)
第二节 系统的连接——S-100 总线	(382)
第三节 几块板的功能介绍	(385)
一、ZPU 板	(385)
二、64KZ 板	(387)
三、PRI 板	(393)
四、4FDC 板	(396)
第四节 CROMEMCO 系统的管理	(421)
一、CDOS 的功能	(421)
二、CDOS 介绍	(423)
附录	(433)
附录 1	(433)
附录 2	(435)
附录 3	(445)
附录 4	(465)

第一章 概 述

第一节 引 言

随着计算机应用的推广和普及,随着大规模集成电路技术的飞速发展,70年代初诞生了一代新型的电子计算机——微型计算机(Microcomputer)。它利用大规模集成电路技术把计算机的中央处理单元(CPU——Central Processing Unit)——即计算机的运算器和控制器集成在一个芯片上称为微处理器(Microprocessor),典型的型号有8080,6800,Z80等。同样利用大规模集成电路技术做成了容量相当大的存贮器(Memory)芯片如 1024×1 , 4096×1 , $16K \times 1$ 直至 $64K \times 1$ 的读写存贮器(或称为随机存取存贮器)RAM(Random Access Memory)和只读存贮器ROM(Read Only Memory);同时又把各种通用的或专用的,可编程的接口电路(与外部设备接口的)集成在一个片子上。这样,把CPU(例如8080,6800或Z80)配上一定容量的RAM,ROM以及接口电路(并行接口电路PIO,串行接口电路SIO)和必要的外设,例如CRT(带键盘),打印机等等就形成了一个微型计算机(或称为微型计算机系统)。

在有些专用的场合,还把CPU,一定容量的RAM和ROM,以及输入输出接口电路,集成在一个芯片上,就成了单片计算机(Single Chip Computer)。或把CPU, RAM和ROM,输入输出接口装在一块印刷电路板上,就成为单板计算机(Single Board Computer)。

总之,微型计算机以利用大规模和超大规模集成电路技术为特征,大大缩小了计算机的体积,同时也大大降低了成本(在国外,一个典型的微型计算机系统,包括:64K内存,两个8"软盘,CRT,行打印机等约为1~2万美元;而一个单板计算机,它通常具有1~2K ROM管理程序,1~2K RAM,以及几个输入输出接口,约为几百美元)。但是功能却赶上甚至超过了60年代的小型计算机。因而发展极为迅速,应用极为广泛,已经深入到工农业生产、国防、文教、科研,以及日常生活等各个领域。

在我国微型计算机的科研和生产工作已经逐步发展,应用也已开始深入到各个领域。

本书是从应用的角度,较全面系统地分析和论述了微型计算机系统的硬件和软件。硬件方面着重于分析CPU的结构和时序;分析半导体存贮器的选用及与CPU的接口;各种外设与CPU的接口技术以及接口片子;中断功能和D/A、A/D转换片子及与CPU的接口等,在软件方面,重点介绍了汇编语言,介绍了如何用汇编语言编程序,同时介绍了汇编程序,监控与调试程序。在书中介绍了大量的各种应用程序的例子。所有这些都是从工程技术人员如何利用微型计算机来解决各种工程问题,如何编应用程序的角度来分析、论述问题的。

第二节 计算机中的数和编码系统

计算机的最基本功能是进行数的计算和处理加工。数在机中是以器件的物理状态来表示的,为了使表示更为方便和可靠,在计算机中主要采用了二进制数字系统。或者说,计算机只认得二进制数,也即要机器处理的所有的数,都要用二进制数字系统来表示;所有的字

母、符号亦都要用二进制编码来表示。所以，我们的分析从二进制数字系统着手。

一、进位计数制

在日常生活中，可迁到各种进位计数制，如：10进制、12进制、16进制、60进制等。当然，最常用的是十进制。

(一) 十进制数

一个十进制数有两个主要特点：

1. 它有十个不同的数字符号，即：0、1、2、…8、9。

2. 它是逢“十”进位的。因此，同一个数字符号在不同的位置（或数位）代表的数值是不同的。例如，在999.99这个数中，小数点左面第一位的9代表个位，就是它本身的数值9；小数点左面第二位的9就代表十位，它的值为 9×10^1 ；左面第三位的9就代表百位，它的值为 9×10^2 ；而小数点右面第一位它的值就为 9×10^{-1} ；右面第二位它的值就为 9×10^{-2} ……。所以，这个数可以写成

$$999.99 = 9 \times 10^2 + 9 \times 10^1 + 9 \times 10^0 + 9 \times 10^{-1} + 9 \times 10^{-2}$$

一般地说，任意一个十进制数A，都可以表示为：

$$\begin{aligned} A &= A_{n-1} \cdot 10^{n-1} + A_{n-2} \cdot 10^{n-2} + \dots + A_1 \cdot 10^1 + A_0 \cdot 10^0 \\ &\quad + A_{-1} \cdot 10^{-1} + A_{-2} \cdot 10^{-2} + \dots + A_{-m} \cdot 10^{-m} \\ &= \sum_{i=n-1}^{-m} A_i \cdot 10^i \end{aligned}$$

其中i表示数的某一位； A_i 表示第i位的数码，它可以是0—9中的任一个，由具体的数A确定；m和n为正整数，n为小数点左面的位数，m为小数点右边的位数。式中10就称为计数制的底数（或称为基数），所以，这是十进制数。

(二) 二进制数的表示

与十进制数类似，它也有两个主要特点：

1. 它的数值部分，只需用两个符号0和1来表示。

2. 它是逢“二”进位的。因此，不同的数码在不同的数位所代表的值也是不同的。例

$$(1001)_2 = 1 \times 2^3 + 1 \times 2^0 = (8+1)_{10} = (9)_{10}$$

$$(11011.101)_2 = 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-3} = (27.625)_{10}$$

一般地说，任意一个二进制数B，都可以表示为：

$$\begin{aligned} B &= B_{n-1} \cdot 2^{n-1} + B_{n-2} \cdot 2^{n-2} + \dots + B_1 \cdot 2^1 + B_0 \cdot 2^0 \\ &\quad + B_{-1} \cdot 2^{-1} + B_{-2} \cdot 2^{-2} + \dots + B_{-m} \cdot 2^{-m} \\ &= \sum_{i=n-1}^{-m} B_i \cdot 2^i \end{aligned}$$

其中 B_i 只能取1或0，由具体的数B确定。n、m为正整数，n为小数点左面的位数，m为小数点右面的位数。二是进位制的基数，故称为二进制。

(三) 八进制数的表示

类似地它也有两个主要特点：

1. 用八个不同的数码符号0—7来表示数值。

2. 逢“八”进位。因此，在不同数位，数码所表示的值是不同的。例：

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

$$(372.135)_8 = 3 \times 8^2 + 7 \times 8^1 + 2 \times 8^0 + 1 \times 8^{-1} + 3 \times 8^{-2} + 5 \times 8^{-3} \\ = (250.18164)_{10}$$

所以，任意一个 8 进制数 C 可表示为：

$$C = C_{n-1} \cdot 8^{n-1} + C_{n-2} \cdot 8^{n-2} + \dots + C_1 \cdot 8^1 + C_0 \cdot 8^0 \\ + C_{-1} \cdot 8^{-1} + \dots + C_{-m} \cdot 8^{-m} \\ = \sum_{i=n-1}^{-m} C_i \cdot 8^i$$

其中， C_i 可取 0—7 之间的值，取决于数值 C； n 、 m 为正整数， n 为小数点左边的位数， m 为小数点右边的位数；8 为基数，故称为 8 进制。

(四) 16 进制数的表示

它也有两个主要特点：

1. 用 16 个不同的数码符号 0—9 以及 A、B、C、D、E、F 来表示数值，它与十进制和二进制数之间的关系如表 1-1 所示。

2. 逢“16”进位。因此，在不同数位，数码所表示的值是不同的。例：

$$(327)_{16} = 3 \times 16^2 + 2 \times 16^1 + 7 \times 16^0 = (807)_{10} \\ (3AB.11)_{16} = 3 \times 16^2 + A \times 16^1 + B \times 16^0 + 1 \times 16^{-1} + 1 \times 16^{-2} \\ = (939.0664)_{10}$$

所以，一个任意的 16 进制数 D，可以表示为：

$$D = D_{n-1} \cdot 16^{n-1} + D_{n-2} \cdot 16^{n-2} + \dots + D_1 \cdot 16^1 + D_0 \cdot 16^0 + D_{-1} \cdot 16^{-1} + \dots + D_{-m} \cdot 16^{-m} \\ = \sum_{i=n-1}^{-m} D_i \cdot 16^i$$

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

其中， D_i 可取 0—F 之间的值，取决于数值 D； n 、 m 为正整数， n 为小数点左边的位数， m 为小数点右边的位数；16 为基数，故称为 16 进制。

综上所述，可以把它们的特点概括为：

1. 每一种计数制都有一个固定的基数 J，它的每一位可能取 J 个不同的数值。

2. 它是逢“J”进位的。因此，它的每一个数位 i ，对应一个固定的值 J^i ， J^i 就称为该位的“权”，小数点左面各位的权依次是基数 J 的正次幂，而小数点右面各位的权依次是基数 J 的负次幂。与此相关，若小数点向左移一位（或数向右移一位），则等于减小了 J 倍；若小数点向右移一位，则等于增加了 J 倍。

二、进位制数之间的转换

(一) 二进制数转换成十进制数

这比较方便，根据二进制数的定义，只要将它按权展开相加就行。

例：

$$(111.101)_2 = 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-3} \\ = (7.625)_{10}$$

(二) 十进制整数转换成二进制整数

若要把十进制整数215转换成二进制数。即要变成如下的形式

$$(215)_{10} = (K_{n-1}K_{n-2}\dots\dots K_1K_0)_2$$

问题就是要找到 $K_{n-1}-K_0$ 的值，而这些值不是1就是0，取决于要转换的十进制数(例中即为215)。根据二进制的定义

$$(K_{n-1}K_{n-2}\dots\dots K_1K_0)_2 = K_{n-1} \cdot 2^{n-1} + K_{n-2} \cdot 2^{n-2} + \dots + K_1 \cdot 2^1 + K_0 \cdot 2^0$$

$$\therefore (215)_{10} = K_{n-1} \cdot 2^{n-1} + K_{n-2} \cdot 2^{n-2} + \dots + K_1 \cdot 2^1 + K_0 \cdot 2^0$$

显然，等式右边，除了最后一项 K_0 以外，都包含有2的因子，它们都能被2除尽。故，用2去除十进制数 $(215)_{10}$ ，则它的余数即为 K_0 。

$$\begin{array}{r} 2 \overline{) 215} \\ 107 \quad \dots\dots \text{余} 1 = K_0 \end{array}$$

$(215)_{10}$ 被2除且去掉余数后为

$$(107)_{10} = K_{n-1} \cdot 2^{n-2} + K_{n-2} \cdot 2^{n-3} + \dots + K_2 \cdot 2^1 + K_1$$

显然，上面等式的右边除了最后一项 K_1 以外，都包含有2的因子，都能被2除尽，故若用2去除107，所得的余数必为 K_1 。

$$\begin{array}{r} 2 \overline{) 107} \\ 53 \quad \dots\dots \text{余} 1 = K_1 \end{array}$$

用这样的方法一直继续下去，直到商为0，就可以得到 $K_{n-1}, K_{n-2}, \dots\dots K_1, K_0$ 。

$$\begin{array}{r} 2 \overline{) 215} \\ 2 \overline{) 107} \dots\dots \text{余} 1 = K_0 \\ 2 \overline{) 53} \dots\dots \text{余} 1 = K_1 \\ 2 \overline{) 26} \dots\dots \text{余} 1 = K_2 \\ 2 \overline{) 13} \dots\dots \text{余} 0 = K_3 \\ 2 \overline{) 6} \dots\dots \text{余} 1 = K_4 \\ 2 \overline{) 3} \dots\dots \text{余} 0 = K_5 \\ 2 \overline{) 1} \dots\dots \text{余} 1 = K_6 \\ 0 \dots\dots \text{余} 1 = K_7 \end{array}$$

$$\therefore (215)_{10} = K_7K_6K_5K_4K_3K_2K_1K_0 = (11010111)_2$$

由此可概括出把十进制整数转换为二进制整数的方法是：用2不断地去除要转换的十进制数，直至商为0。每次的余数即为二进制数码，最初得到的为整数的最低有效数(LSD—Least Significant Digit) K_0 ，最后得到的为最高有效数(MSD—Most Significant Digit) K_{n-1} 。

(三) 十进制小数转换为二进制小数

若要把十进制小数0.6875转换为二进制小数形式。即：

$$(0.6875)_{10} = (0.K_{-1}K_{-2}\dots K_{-n})_2$$

就是要由 $(0.6875)_{10}$ 来确定 $K_{-1}-K_{-n}$ 的值。按照二进制小数的定义，可把上式写成

$$(0.6875)_{10} = K_{-1} \cdot 2^{-1} + K_{-2} \cdot 2^{-2} + \dots + K_{-n} \cdot 2^{-n}$$

若两边都乘 2，则得

$$(1.375)_{10} = K_{-1} + (K_{-2} \cdot 2^{-1} + K_{-3} \cdot 2^{-2} + \dots + K_{-n} \cdot 2^{-n+1})$$

显然等式右边括号内的数是 <1 的（因乘 2 以前是 <0.5 的）。而两个小数若相等，必然是整数部分和小数部分分别相等。故 $K_{-1} = 1$ 。剩下的

$$(0.375)_{10} = K_{-2} \cdot 2^{-1} + K_{-3} \cdot 2^{-2} + \dots + K_{-n} \cdot 2^{-n+1}$$

若两边再乘以 2，则得

$$(0.75)_{10} = K_{-2} + K_{-3} \cdot 2^{-1} + K_{-4} \cdot 2^{-2} + \dots + K_{-n} \cdot 2^{-n+2}$$

于是 $K_{-2} = 0$ 。如此继续下去，可逐个得到 K_{-1} 、 K_{-2} 、 $\dots$ 、 K_{-n} 的值。

$$\begin{array}{r} 0.6875 \\ \times \quad 2 \\ \hline 1.3750 \quad \dots\dots \text{整数部分} = 1 = K_{-1} \\ 0.375 \\ \times \quad 2 \\ \hline 0.750 \quad \dots\dots \text{整数部分} = 0 = K_{-2} \\ \times \quad 2 \\ \hline 1.500 \quad \dots\dots \text{整数部分} = 1 = K_{-3} \\ 0.5 \\ \times \quad 2 \\ \hline 1.0 \quad \dots\dots \text{整数部分} = 1 = K_{-4} \end{array}$$

$$\therefore (0.6875)_{10} = (0.1011)_2$$

要注意的是，在十进制小数转换成二进制小数时，不断用 2 去乘不一定都能使尾数部分等于 0，过程可能会无限制的进行下去。这时只要根据精度要求，二进制小数取足够的位数就行。

由此可概括出十进制小数转换为二进制小数的办法是：不断用 2 去乘要转换的十进制小数，将每次所得的整数（0 或 1），依次记为 K_{-1} 、 K_{-2} 、 $\dots$ 。若乘积的小数部分最后能为 0，那么最后一次乘积的整数部分记为 K_{-n} 。则

$$0.K_{-1}K_{-2}\dots K_{-n}$$

即为十进制小数的二进制表达式。

但十进制小数，并不都是能用有限位的二进制小数精确表示的。则可根据精度要求取 m 位，得到十进制小数的二进制的近似表达式。

十进制整数和小数都能分别转换为二进制的整数和小数，所以一个具有整数和小数部分的十进制数，在转换为二进制数时，只要把它分为整数和小数两部分，然后再把它们分别转换为二进制表达式，最后用小数点把这两部分连起来就可以了。

（四）任意进位制数与十进制数之间的转换

一般地说，任意进位制数与十进制数之间的转换的原理和方法，跟二进制与十进制之间的转换的原理和方法类似。比如，八进制与十进制之间，八进制数转换为十进制数，只要将它按权展开，然后相加就行。十进制转换为八进制，就要把整数和小数分别进行转换。十进制整数转换为八进制整数，只要不断用 8 去除，每次所得的余数就为八进制的系数，最先得到的是八进制整数的最低有效数，最后得到的为最高有效数；而十进制小数在转换时只要不断用 8 去乘，每次所得的整数部分，即为八进制小数的系数。最先得到的为八进制小数的最高有效数，最后得到的为最低有效数。

例：将十进制数 835.6875 转换为八进制数。

首先把整数部分和小数部分分开，分别把它们转换为八进制：

$$\begin{array}{r} 8 \ 8 \ 8 \ 5 \\ 8 \ 1 \ 0 \ 4 \dots\dots \text{余 } 3 = K_0 \\ 8 \ 1 \ 3 \dots\dots \text{余 } 0 = K_1 \\ 8 \ 1 \dots\dots \text{余 } 5 = K_2 \\ 0 \dots\dots \text{余 } 1 = K_3 \end{array}$$

所以

$$(835)_{10} = (1503)_8$$

$$\begin{array}{r} 0.6 \ 8 \ 7 \ 5 \\ \times \quad \quad \quad 8 \\ \hline 5.5 \ 0 \ 0 \ 0 \dots\dots \text{整数部分} = 5 = K_{-1} \end{array}$$

$$\begin{array}{r} 0.5 \\ \times \quad \quad \quad 8 \\ \hline 4.0 \dots\dots \text{整数部分} = 4 = K_{-2} \end{array}$$

$$\therefore (835.6875)_{10} = (1503.54)_8$$

(五) 八进制与二进制之间的转换

因为 $2^3 = 8$ ，所以一位八进制数相当于三位二进制数，它们是完全对应的。因此，八进制与二进制之间的转换十分方便。

1. 八进制转换为二进制

每位八进制数，可以用三位二进制来表示。

例：(563)₈，分别把 5、6、3 用三位二进制数表示如下：

$$\begin{array}{c} 563 \\ \swarrow \quad | \quad \searrow \\ 101 \ 110 \ 011 \end{array}$$

即

$$(563)_8 = (101 \ 110 \ 011)_2$$

例：(0.764)₈，每一位可分别用三位二进制数表示如下：

$$\begin{array}{c} 0.764 \\ \swarrow \quad | \quad \searrow \\ 111 \ 110 \ 100 \end{array}$$

$$\therefore (0.764)_8 = (0.111 \ 110 \ 100)_2 = (0.111 \ 1101)_2$$

2. 二进制转换为八进制

二进制整数转换为八进制时，可以从最低位（小数点之左第一位）开始，每三位分为一组，不够三位的补足三位，然后把每三位二进制数用相应的八进制表示就行了。

例：(1110 111 0011)₂可转换如下：

$$\begin{array}{cccc} 011, & 101, & 110, & 011 \\ | & | & | & | \\ 3 & 5 & 6 & 3 \end{array}$$

$$\therefore (1110 \ 111 \ 0011)_2 = (3563)_8$$

二进制小数转换为八进制时，从小数点右面第一位开始，每三位分为一组，最后不足三位的补足三位，然后把每一组二进制数用相应的八进制表示即可。

例: $(0.101\ 100\ 0111)_2$ 可转换如下:

$$\begin{array}{cccc} 0.101, & 100, & 011, & 100 \\ | & | & | & | \\ 5 & 4 & 3 & 4 \end{array}$$

$$\therefore (0.101\ 100\ 0111)_2 = (0.5434)_8$$

可见, 八进制与二进制之间的转换是十分方便的。在计算机中, 数是以二进制表示的, 但二进制书写起来太长, 易错, 通常用八进制 (或16进制) 来书写。另外, 从十进制转换为八进制计算过程短、方便。所以, 在从十进制转换为二进制时, 常是先转换为八进制, 然后再转换为二进制。

(六) 十六进制与二进制之间的转换

在微型计算机中, 目前通用的字长为 8 位, 则可用两位16进制数表示, 故16进制在微型机中应用十分普遍。

而 $16 = 2^4$, 所以, 在16进制与二进制之间也存在着类似于 8 进制与二进制之间的简单而又直接的联系: 用四位二进制数表示一位16进制数, 只要熟悉这个联系, 16进制与二进制之间的转换也是十分方便的。

1. 16进制转换为二进制

不论是16进制的整数或小数, 只要把每一位16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: $(3AB)_{16}$ 可转换为

$$\begin{array}{ccc} & 3 & A & B \\ & / & | & \backslash \\ 0011 & 1010 & 1011 \end{array}$$

$$\therefore (3AB)_{16} = (0011\ 1010\ 1011)_2 = (11\ 1010\ 1011)_2$$

$(0.7A53)_{16}$ 可转换为

$$\begin{array}{cccc} & 0.7 & A & 5 & 3 \\ & / & | & | & \backslash \\ 0111 & 1010 & 0101 & 0011 \end{array}$$

$$\therefore (0.7A53)_{16} = (0.0111\ 1010\ 0101\ 0011)_2$$

2. 二进制转换为16进制

二进制的整数部分由小数点向左, 每四位一分, 最后不足四位的前面补 0; 小数部分由小数点向右, 每四位一分, 最后不足四位的后面补 0。然后把每四位二进制数用相应的16进制数代替, 即可转换为16进制数

例: $(11011\ 1110\ 0011.1001\ 01111)_2$ 可转换为

$$\begin{array}{ccccccc} 0001, & 1011, & 1110, & 0011, & 1001, & 0111, & 1000 \\ | & | & | & | & | & | & | \\ 1 & B & E & 3 & 9 & 7 & 8 \end{array}$$

$$\therefore (11011\ 1110\ 0011.1001\ 01111)_2 = (1\ BE\ 3.9\ 7\ 8)_{16} \text{①}$$

三、二进制编码

如前所述, 在计算机中, 是采用二进制数, 因而, 要在计算机中表示的数、字母、符号

①以后我们用数字后面加字母B (Binary) 表示为二进制数; 以字母Q (Octal) 表示为8进制数; 以字母D (Decimal) 或不加字母表示为十进制数; 用字母H (Hexadecimal) 表示为16进制数。