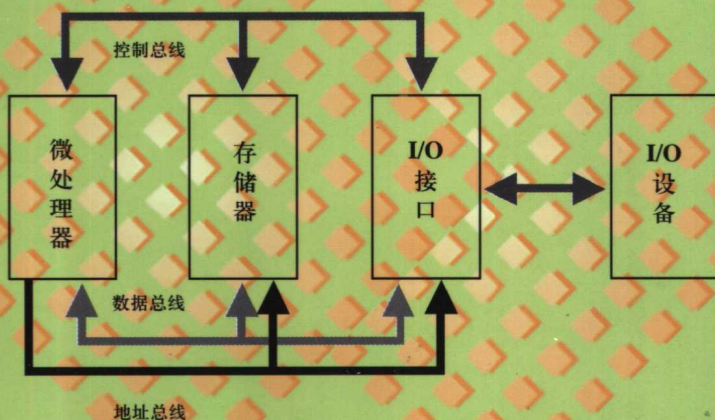


21世纪大学课程辅导丛书

微机原理与接口技术

重点难点及典型题精解

马瑞芳 王会燃



西安交通大学出版社

21 世纪大学课程辅导丛书

微机原理与接口技术

重点难点及典型题精解

马瑞芳 王会燃

西安交通大学出版社

内 容 提 要

本书是大专院校学生学习“微机原理与接口技术”课程的辅导教材,其中包括计算机基础知识、微处理器结构、存储器、指令系统、汇编语言程序设计、输入/输出与中断系统、I/O 接口电路、80x86 微处理器的结构及其应用等方面的内容。本书突出一个“练”字,通过大量习题和测试题的练习,促进读者对基本知识的理解和掌握。每章在提出基本要求、简要回顾基本知识点后,给出了典型例题和详解,然后提供了大量的习题,再给出一套自我测试题。在全书的最后,提供了几套课程综合测试题。

书中编制例题、习题及测试题共有 350 多道,每道题都附有参考答案。本书可作为大学本科生、自学考试生、大专生的学习辅导教材,也可作为研究生入学考试的参考资料。

图书在版编目(CIP)数据

微机原理与接口技术重点难点及典型题精解/马瑞芳,
王会燃编. —西安:西安交通大学出版社,2002.8

(21 世纪大学课程辅导丛书)

ISBN 7-5605-1566-5

I. 微… II. ①马…②王… III. ① 微型计算机—
基础理论—高等学校—教学参考资料②微型计算机—接
口—高等学校—教学参考资料 IV. TP36

中国版本图书馆 CIP 数据核字(2002)第 060248 号

*

西安交通大学出版社出版发行
(西安市兴庆南路 25 号 邮政编码:710049 电话:(029)2668315)
西安建筑科技大学印刷厂印装
各地新华书店经销

*

开本:787mm×1092mm 1/16 印张:16.5 字数:398 千字
2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷
印数:0 001~4 000 定价:19.50 元

发行科电话:(029)2668357,2667874



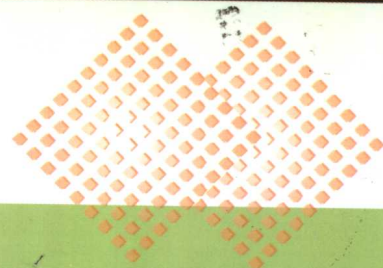
马瑞芳 西安交通大学电信学院计算机系副教授，主要从事计算机网络通信方面的研究工作，主讲过“微机原理及接口技术”、“汇编语言程序设计”、“单片机原理及其应用”等课程，主持和参加科研项目7项，编著教材4部，发表论文16篇，获省级、校级奖励7项。



王会燃 生于1959年，1982年在西安交通大学取得学士学位，1988年在西安交通大学取得博士学位，1987~1988年在德国柏林工业大学访问研究一年，1999年获教授职称。现工作于西安工程科技学院，任环境工程系系主任。主要从事CAD/CAM和计算机控制方面的教学和研究工作，主持参加省部级和其他科研项目10余项，编著教材3部，发表论文30余篇。

微机原理与接口技术

重点难点及典型题精解



目 录

前言

第 1 章 计算机基础知识

1.1 基本知识点	1
1.1.1 基本概念	1
1.1.2 数据表示与编码	1
1.1.3 CPU 执行指令的过程	3
1.2 重点与难点	4
1.3 典型题精解	5
1.4 习题	13
1.5 自我测试题	15

第 2 章 微处理器结构

2.1 基本知识点	16
2.1.1 8086/8088 概述	16
2.1.2 8086/8088 的主要区别	16
2.1.3 8086CPU 的功能结构	16
2.1.4 8086/8088CPU 的工作模式和引脚信号	19
2.1.5 8086/8088 的主要操作	23
2.1.6 最小工作模式	24
2.1.7 最大工作模式	27
2.2 重点与难点	28
2.3 典型题精解	29
2.4 习题	33
2.5 自我测试题	35

第 3 章 存储器

3.1 基本知识点	36
3.1.1 存储器的分类及特点	36
3.1.2 读写存储器 RAM	39
3.1.3 只读存储器 ROM	39
3.1.4 IBM PC/XT 的存储器	41
3.2 重点与难点	42

3.3 典型题精解·····	42
3.4 习题·····	44
3.5 自我测试题·····	45
 第4章 指令系统	
4.1 基本知识点·····	46
4.1.1 计算机的指令格式·····	46
4.1.2 8086/8088 CPU 与数据有关的基本寻址方式 ·····	46
4.1.3 与转移指令有关的寻址方式·····	48
4.1.4 8086/8088 指令系统分类·····	50
4.1.5 学习本章需注意的问题·····	52
4.2 重点与难点·····	52
4.3 典型题精解·····	53
4.4 习题·····	76
4.5 自我测试题·····	79
 第5章 汇编语言程序设计	
5.1 基本知识点·····	80
5.1.1 汇编语言语句的种类和格式·····	80
5.1.2 程序设计的基本步骤·····	81
5.1.3 程序设计的基本方法·····	82
5.1.4 常用的 DOS 中断和功能调用 ·····	83
5.1.5 汇编语言程序常用的几种退出方法·····	85
5.2 重点与难点·····	87
5.3 典型题精解·····	88
5.4 习题 ·····	116
5.5 自我测试题 ·····	121
 第6章 输入/输出与中断系统	
6.1 基本知识点 ·····	123
6.1.1 输入/输出·····	123
6.1.2 中断 ·····	125
6.1.3 中断控制器 8259A ·····	126
6.1.4 DMA 及 DMA 控制器 8237 ·····	132
6.2 重点与难点 ·····	138
6.3 典型题精解 ·····	139
6.4 习题 ·····	143
6.5 自我测试题 ·····	144

第7章 I/O 接口电路

7.1 基本知识	145
7.1.1 接口电路	145
7.1.2 并行接口芯片 8255A	145
7.1.3 串行通信接口芯片 8250	149
7.1.4 计数/定时器 8253	157
7.1.5 系统总线	159
7.2 重点与难点	160
7.3 典型题精解	161
7.4 习题	174
7.5 自我测试题	174

第8章 80x86 微处理器的结构及其应用

8.1 基本知识	175
8.1.1 80386 微处理器	175
8.1.2 80386CPU 的寻址方式及其指令系统	177
8.1.3 存储器管理	178
8.2 重点与难点	180
8.3 典型题精解	181
8.4 习题	185
8.5 自我测试题	185

附录1 综合测试题

综合测试题(一)	186
综合测试题(二)	188
综合测试题(三)	190

附录2 习题/自我测试题参考答案

第1章 习题	193
第1章 自我测试题	196
第2章 习题	198
第2章 自我测试题	204
第3章 习题	204
第3章 自我测试题	209
第4章 习题	211
第4章 自我测试题	215
第5章 习题	216
第5章 自我测试题	219
第6章 习题	222

第 6 章	自我测试题	232
第 7 章	习题	232
第 7 章	自我测试题	241
第 8 章	习题	243
第 8 章	自我测试题	244

附录 3 综合测试题参考答案

综合测试题(一)	245
综合测试题(二)	248
综合测试题(三)	250

参考文献

第 1 章 计算机基础知识

本章介绍计算机基础知识及其计算机中数据的表示与编码。

1.1 基本知识点

1.1.1 基本概念

1. CPU

中央处理单元(central processing unit)也称微处理器,由控制器和运算器组成。运算器执行所有的算术和逻辑运算指令;控制器通过从存储器中逐条取出指令,经译码分析后向计算机的各个部件发出取数、执行、存数等控制命令,协调计算机各部件之间的工作。微处理器芯片集成了计算机的控制逻辑和运算单元。

2. 存储器

存储器是独立于 CPU 之外的芯片或芯片组,是用来存放数据和程序的装置。

3. 输入/输出接口电路

输入/输出接口电路常常独立地做在一个芯片上。它是介于主机和外设之间的一种起缓冲、转换、匹配作用的电路,是外设与主机进行数据交换的桥梁。

4. 微型计算机

由微处理器、主存储器和输入输出接口电路组成。

5. 微型计算机(microcomputer)的特点

与大、中、小型计算机的区别就在于微型计算机的中央处理器(CPU)是集中在一块芯片上。为了区别于大、中、小型计算机的 CPU,称微型计算机的 CPU 芯片为微处理器(micro-processing unit 或 micro-processor)。

6. 微型计算机系统

由微型计算机配以相应的外围设备如显示器(CRT)、键盘及打印机等组成。

1.1.2 数据表示与编码

1. 微处理器常用的数制

微处理器常用的数是二进制数、十进制数和十六进制数。

二进制数是以 2 为基的表示法,数由两个数字构成(0 和 1),一个二进制数的后缀为 B,如 10110111B。

十进制数是以 10 为基的表示法,数由 10 个数字构成(0~9),一个十进制数的后缀为 D,

如 2001D。在大多数情况下,十进制数的后缀可省略。

十六进制数是以 16 为基的表示法,数由 16 个数字构成(0~9,A(10),B(11),C(12),D(13),E(14),F(15)),一个十六进制数的后缀为 H。

2. 不同进位计数制之间的数据转换

要把一种进制数转换为另一种进制的数,其实质是进行基数的转换。

基数转换是依据两个有理数相等时,其整数部分与小数部分分别相等的原则。因此在转换时,其整数部分与小数部分应分别进行转换,将转换后的结果合并,整数部分与小数部分之间用小数点隔开,就得到相应的转换结果。

3. 二进制数转换为十进制数

转换规则是“按权值相加”。也就是说,只要把二进制数中数位是“1”的那些位的权值相加,其和就是等效的十进制数(参见例 1.4)。

4. 十进制数转换为二进制数

十进制数转换为二进制数,它的整数和小数部分要分别进行转换。转换结束后将整数转换结果写在左边,小数转换结果写在右边,中间点上小数点。整数部分转换规则:将十进制整数用基数 2 连续去除,直到商为 0 为止,将每次除得的余数反向排列,就可得到十进制数整数部分的转换结果。反向排列是指最后得到的余数排在前边,作为结果的最高位,最先得到的余数排在后边,作为结果的最低位(参见例 1.5)。

小数部分的转换规则:将十进制数的小数部分用基数 2 连续去乘,直到小数部分为 0 或达到精度为止,将每次所得的乘积的整数部分正向排列,就可得到十进制小数的转换结果。正向排列是指最先得到的整数为结果的最高位,最后得到的整数为结果的最低位(参见例 1.5)。

5. 二进制数转换为(八进制)十六进制数

将二进制数以小数点为界,向左、向右分别按(3 位)4 位一组划分,不足(3 位)4 位的部分用“0”补足,将每一组数写成对应的(八进制)十六进制数,就可得到转换结果。

6. 原码表示法

将数的真值形式中的正(负)号用代码 0(1)来表示,数值部分用二进制来表示。

原码的特点如下:

(1) “0”的原码有两种表示法,它们是

$$[+0]_{\text{原}} = 00000000\text{B}$$

$$[-0]_{\text{原}} = 10000000\text{B}$$

(2) n 位二进制原码所能表示的数值范围为:

$$-(2^{n-1}-1) \sim (2^{n-1}-1)$$

(3) 原码表示一个数时,最高位为符号位。

符号位为 0 时,其后面的 $n-1$ 位为数值部分,这个数为正数。符号位为 1 时,其后面的 $n-1$ 位为数值部分,这个数为负数。

7. 补码表示法

正数的补码表示与正数的原码相同,负数的补码表示为它的原码表示除符号位外其余位按位取反且在最低位加 1 形成。

补码的特点如下:

(1) “0”的补码表示是唯一的。

$$[+0]_{\text{补}} = [-0]_{\text{补}} = 00000000\text{B}$$

(2) 补码运算时符号位无需单独处理。

(3) 采用补码运算时,减法可用加法来实现。

8. 补码加法和减法的规则

补码的加法和减法规则是:

$$[x + y]_{\text{补}} = [x]_{\text{补}} + [y]_{\text{补}}$$

$$[x - y]_{\text{补}} = [x]_{\text{补}} + [-y]_{\text{补}}$$

9. ASCII 码

ASCII(american standard code for information interchange, 美国信息交换标准代码)码是字符数据的通用编码,采用七位二进制编码,可表示 128 个字符,其中包括数字(0~9)、字母(大、小写)、通用控制字符和专用字符。

10. BCD 码

BCD (binary code decimal)码是一种二进制编码的十进制数。常用的 BCD 码是 8421 BCD 码。它的每一位十进制数都分别用 4 位二进制数来表示。BCD 码有两种格式:

(1) 压缩格式的 BCD 码 在一个字节内能表示 2 位十进制数,如

$$82\text{D} = 1000\ 0010\text{BCD} \quad (\text{其中 D 表示十进制数,BCD 表示 BCD 码})$$

(2) 非压缩格式的 BCD 码 在一个字节内只表示 1 位十进制数的 BCD 码,这个字节的低 4 位表示 8421 BCD 码,而高 4 位无意义,如

$$82\text{D} = \times \times \times \times 1000 \quad \times \times \times \times 0010\text{BCD}$$

1.1.3 CPU 执行指令的过程

1. CPU 执行指令时通常涉及到的硬件寄存器

PC——程序计数器。它存放待执行指令的地址,也称为指令地址计数器。在指令执行前给 PC 赋予程序的首条指令的地址,方式有:手工和系统自动装入程序入口地址。

IR——指令寄存器。用于存放从存储器中取出的当前指令。

PSW——程序状态字。用于存放指令的执行结果和控制执行特定功能的标志。如指令的执行结果有无溢出、有无进位等,可作为分支转移与否的依据。

2. CPU 执行一条指令的过程

概括地说,CPU 执行一条指令的过程分为取指令、分析指令和执行指令这三个阶段(如图 1.1 所示)。

(1) 取指令阶段 CPU 根据 PC 中的指令地址取出存放在存储器中的指令,存入指令寄存器 IR 中。PC 自动修改,指向下一条待取出的指令。

(2) 分析指令阶段 对 IR 中的指令进行译码分析。

(3) 执行指令阶段 根据指令分析的结果,按照下列四种情况之一执行:

① 若为暂停指令,则暂停指令的执行。

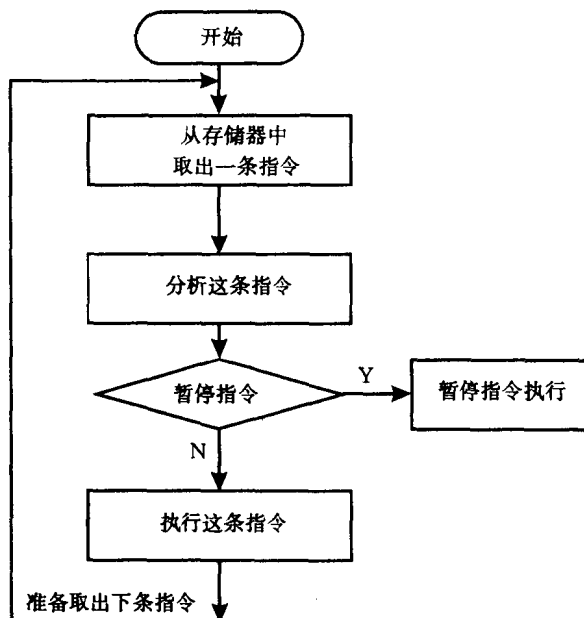


图 1.1 微处理器基本工作流程图

② 若为无条件转移指令,则转移地址置入 PC 中,为取出下一条指令做好准备,转(1)。

③ 若为条件转移指令,则检查判断 PSW 中的条件是否满足。若条件满足,将转移地址置入 PC 中,转(1);若条件不满足,直接转(1)。

④ 若为其他指令,则执行该指令的功能,转(1)。

1.2 重点与难点

本章主要掌握:

1. 计算机的基本概念

- (1) CPU
- (2) 存储器
- (3) 输入输出接口电路
- (4) 微型计算机
- (5) 微型计算机系统

2. 计算机中常用的数制

- (1) 二进制数表示法
- (2) 十进制数表示法
- (3) 十六进制数表示法
- (4) 各种进制数之间的转换

3. 机器数的编码表示及其运算

- (1) 原码表示法
- (2) 补码表示法
- (3) 补码加法和减法规则

4. CPU 执行指令的过程

- (1) 取指令
- (2) 分析指令
- (3) 执行指令

5. 微型计算机的基本组成

- (1) 运算器
- (2) 控制器
- (3) 存储器
- (4) 输入设备
- (5) 输出设备

本章的重点是掌握计算机中常用数据的表示法及其各种数制之间的转换,机器数的编码表示及其运算,理解 CPU 执行指令的过程,结合程序的执行搞清楚微型计算机的组成及其各模块的功能。

本章的难点是理解微型计算机的硬件和软件是如何协调工作的及其 CPU 执行指令的过程。

1.3 典型题精解

例 1.1 以十进制数 999.99 为例,说明十进制数的特点。

答 十进制数是人们日常生活中最常用的一种进位计数制,也是我们最熟悉的一种数制。十进制数有两个主要特点:

- (1) 十进制数的基是 10。
- 十进制数有 10 个不同的数字符号,即 0,1,2,3,4,5,6,7,8,9。
- (2) 十进制数逢 10 进位。

由于十进制数是逢 10 进位的,因此同一个数字符号在不同的位置(数的排列先后)代表的数值不同。例如 999.99 这个数,小数点左边第 1 个数“9”位于个位,它的值就是 9 本身;小数点左边第 2 个数“9”位于十位,它的值就是 $9 \times 10 = 90$;小数点左边第 3 个数“9”位于百位,它的值就是 $9 \times 100 = 900$;而小数点右面第 1 个数“9”位于十分位,它的值是 $9 \times 10^{-1} = 0.9$;小数点右面第 2 个数“9”位于百分位,它的值是 $9 \times 10^{-2} = 0.09$ 。999.99 这个数可以写成

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

一般来说,任意一个十进制数 D (decimal number)都可以表示为:

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

例 1.2 以二进制数 10110111.110 为例,说明二进制数的特点。

答 二进制数各位上的数码只有 0 和 1 两种取值,用计算机内部的电路实现时,比十进制数要方便得多。实际上,在计算机电子线路中用 +5 V 表示 1,用 0 V 表示 0,若干位排列起来就可表示一个二进制数。若把 0 V~5 V 分成 10 段来表示十进制数,那么可靠性随电压波动就会急剧下降。即使电压稳定,线路干扰也会降低可靠性。所以,数据在计算机内部存储、运算及传送均采用二进制数。当把计算机中存放的数据及其运算结果呈现给用户时,系统的翻译转换程序将数据自动转换成各种用户容易理解的数据(如十进制数值),以及图像、图表曲线等。

归结起来,二进制数具有如下特点:

(1) 具有两个不同的基本符号 0, 1;

(2) 逢 2 进 1,借 1 当 2;

(3) 一般表达式 $B = \sum_{i=-m}^{n-1} B_i 2^i$ 。

以二进制数 10110111.110 为例说明上述三个特点。这个二进制数中只有“0”和“1”两种符号,若给这个二进制数加“1”,算式如下:

$$\begin{array}{r} 10110111.110 \\ +00000001.000 \\ \hline 10111000.110 \end{array}$$

给这个二进制数加“1”,相当于给这个数加上“00000001.000”,它是包括 8 位整数和 3 位小数的二进制数。从上述运算中可以看出运用的二进制加法规则如下:

$$0+0=0$$

$$1+0=1$$

$$0+1=1$$

$$1+1=0 \text{——向高位进位为 1(逢 2 进 1)}$$

对于任意两个二进制数相减,算式如下:

$$\begin{array}{r} 10110111.110 \\ -00001010.010 \\ \hline 10101101.100 \end{array}$$

从这个运算中可以看出运用的二进制减法规则如下:

$$0-0=0$$

$$1-0=1$$

$$0-1=1 \text{——向高位借 1,借 1 当 2}$$

$$1-1=0$$

二进制数 10110111.110 可用如下方法来表示:

$$\begin{aligned} (10110111.110)_2 &= 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 \\ &\quad + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} \end{aligned}$$

计算这个表达式的和值,就是相应十进制数的结果,即 183.75。也就是说,二进制数 10110111.110 与十进制数 183.75 是等值的。

例 1.3 简述十六进制数的特点及其与二进制数/十进制数之间的关系。

答 十进制数在日常生活中常用,二进制数适用于计算机内部表示。由于用二进制表示

一个数时数位段多,给人们的记忆、书写都带来一些不方便,而用十六进制数表示一个数时,恰恰克服了二进制数表示的这些缺点。十六进制数的1位可以表示二进制数的4位,为此,人们在书写计算机程序时常常采用十六进制数。

十六进制数的特点:

(1) 具有16个不同的基本符号0~9,A~F;

(2) 逢16进1,借1当16;

(3) 一般表达式是 $H = \sum_{i=-m}^{n-1} H_i 16^i$ 。

表1.1列出了十进制数、二进制数和十六进制数的对应关系。

表 1.1 十进制数、二进制数和十六进制数的对应关系

十进制数	二进制数	十六进制数
0	00000000	00
1	00000001	01
2	00000010	02
3	00000011	03
4	00000100	04
5	00000101	05
6	00000110	06
7	00000111	07
8	00001000	08
9	00001001	09
10	00001010	0A
11	00001011	0B
12	00001100	0C
13	00001101	0D
14	00001110	0E
15	00001111	0F
16	00010000	10

到此为止,我们已经学习了数常用的四种进制的表示方法。书写规则如下:

- 二进制数后缀为字母 B(binary digit);
- 十进制数后缀为字母 D (decimal digit),十进制也可省略后缀 D;
- 十六进制数后缀为字母 H(hexadecimal digit);
- 八进制数后缀为字母 Q(octal digit)。这是由于字母“O”容易与数字“0”相混淆,所以八进制数后缀采用字母 Q。

例 1.4 将下列二进制数转换为十进制数。

(1) 101101.10B

(2) 111.111B

解 把二进制数转换为十进制数的方法是将二进制数位中的数值为1的那些位的权值相加,其和就是等效的十进制数。

$$\begin{aligned}
 (1) \quad 101101.10B &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} \\
 &= 45.5D
 \end{aligned}$$

$$(2) 111.111\text{B} = 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\ = 7.875\text{D}$$

二进制数转换为十进制数时,要求读者熟记位-权表(表 1.2)。

例 1.5 将下列十进制数转换为二进制数。

(1) 23.75 (2) 7.875

解 根据十进制数转换为二进制数的转换规则,整数部分和小数部分分别进行转换。

(1) 23.75

• 整数部分转换:将十进制整数用基数 2 连续去除,直到商为 0 为止,将每次所得的余数反向排列。

$$\begin{array}{rcl} \text{首先, 23 除以 2} & \longrightarrow & 2 \quad \overline{) 23} \\ \text{接着, 11 除以 2} & \longrightarrow & 2 \quad \overline{) 11} \quad \longleftarrow \text{第 1 次得商 11, 余数为 1} \\ \text{5 除以} & \longrightarrow & 2 \quad \overline{) 5} \quad \longleftarrow \text{第 2 次得商 5, 余数为 1} \\ & & 2 \quad \overline{) 2} \quad \longleftarrow \text{第 3 次得商 2, 余数为 1} \\ & & 2 \quad \overline{) 1} \quad \longleftarrow \text{第 4 次得商 1, 余数为 0} \\ & & 0 \quad \longleftarrow \text{第 5 次得商 0, 余数为 1} \end{array}$$

将所得余数反向排列,为整数转换的结果,即 $23\text{D} = 10111\text{B}$ 。

• 小数部分转换:将十进制数的小数部分用基数 2 连续去乘,直到小数部分为 0 或达到精度为止,将每次所得乘积的整数部分正向排列。

$$\begin{array}{rcl} & 0.75 & \\ \text{第 1 次得整数“1”} & \begin{array}{r} \times \quad 2 \\ \hline 1.50 \end{array} & \longleftarrow \text{首先, 0.75 乘以 2} \\ & & \\ \text{第 2 次得整数“1”} & \begin{array}{r} \times \quad 2 \\ \hline 1.00 \end{array} & \begin{array}{l} \longleftarrow 0.50 \text{ 乘以 2} \\ \longleftarrow \text{最后, 余数为 0} \end{array} \end{array}$$

将所得整数顺向排列,为小数部分转换结果,即 $0.75\text{D} = 0.11\text{B}$ 。

最后将整数部分和小数部分的转换结果合并起来,中间点上小数点,就得到 23.75 的转换结果为: $23.75\text{D} = 10111.11\text{B}$ 。

(2) 7.875

• 整数部分

$$\begin{array}{rcl} 2 \quad \overline{) 7} & \text{余数} & \\ 2 \quad \overline{) 3} & 1 & \\ 2 \quad \overline{) 1} & 1 & \uparrow 111 \\ 0 & 1 & \end{array}$$

• 小数部分

$$\begin{array}{rcl} & 0.875 & \\ \text{第 1 次得整数“1”} & \begin{array}{r} \times \quad 2 \\ \hline 1.750 \end{array} & \text{余数} \\ & & 1 \\ \text{第 2 次得整数“1”} & \begin{array}{r} \times \quad 2 \\ \hline 1.500 \end{array} & 1 \\ & & \downarrow 111 \\ \text{第 3 次得整数“1”} & \begin{array}{r} \times \quad 2 \\ \hline 1.000 \end{array} & 1 \end{array}$$

转换结果为: $7.875\text{D} = 111.111\text{B}$ 。

表 1.2 二进制数的位-权表

I	2^I
-3	0.125
-2	0.25
-1	0.5
0	0
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1 024
11	2 048
12	4 096
13	8 192
14	16 384
15	32 768