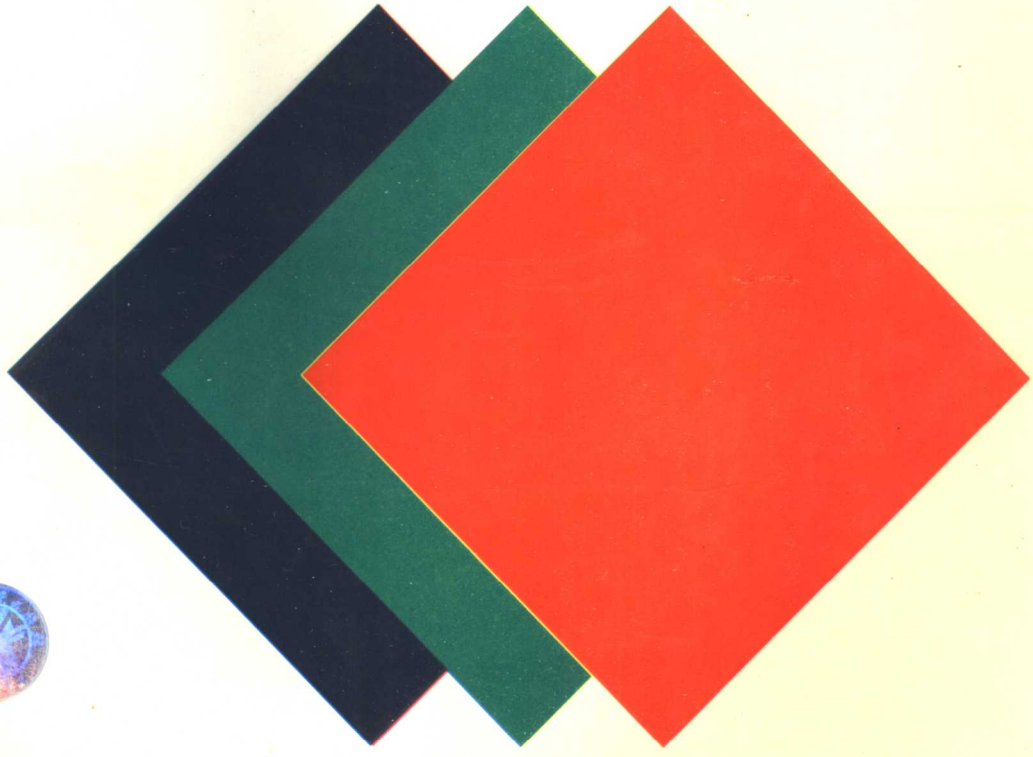
计算机系列教材

逻辑设计（第二版）

毛法尧　欧阳星明　任宏萍



华中理工大学出版社

图书在版编目（CIP）数据

逻辑设计/毛法尧等编．－2版．
武汉：华中理工大学出版社，1996年12月
ISBN 7-5609-1261-3
Ⅰ．逻…
Ⅱ．①毛…　②欧阳…　③任…
Ⅲ．逻辑设计
Ⅳ．TP302.2

内 容 简 介

本书是计算机系列教材之一。全书共分七章，并附有“学习指导与实验”，其主要内容包括数制与编码，逻辑代数基础，组合逻辑电路和时序逻辑电路的分析和设计，采用中、大规模集成电路的逻辑设计及数字系统的设计。

本书可供各类高等学校计算机学科及相关学科的有关专业作为教材，亦可供工程技术人员和其他读者学习、参考。

逻 辑 设 计

（第二版）

毛法尧　欧阳星明　任宏萍

责任编辑：黄以铭　吴凤萍

*

华中理工大学出版社出版发行

（武昌喻家山　邮编：430074）

新华书店湖北发行所经销

武汉市汉桥印刷厂印刷

*

开本：787×1092　1/16　印张：19　字数：468 000

1996年12月第2版　1998年6月第10次印刷

印数：49 001－55 000

ISBN 7-5609-1261-3/TP·175

定价：18.00元

前　言

本书是1990年出版的《逻辑设计》教材的第二版，随着计算机科学技术和教育的发展，教学内容和教学方法也不断更新，本书第二版作了较大幅度的修改。

逻辑设计是计算机科学和工程的重要技术基础。该课程主要介绍逻辑设计的理论基础和逻辑电路的分析和设计方法，重点是组合逻辑电路和同步时序逻辑电路的分析和设计，对于系统仅介绍数字系统的一般构成原则和设计过程。

本书在内容的选取、概念的引入、文字的叙述，以及例、习题的选择等方面，都力求遵循面向应用、重视实践、便于自学的原则。本书共分七章，并附有"学习指导与实验"。其中，第一章数制与编码，主要介绍数字系统中二进制数和各种编码的十进制数的表示及相互转换，带符号二进制数的表示与运算。第二章逻辑代数基础，介绍逻辑代数基本概念和运算规则，并着重介绍逻辑函数的表示和简化。第三章组合逻辑电路，着重讨论组合逻辑电路的分析和设计方法。第四章同步时序逻辑电路，主要讨论同步时序逻辑电路的分析和设计。第五章异步时序逻辑电路，仅对异步时序逻辑电路作一般介绍。第六章采用中、大规模集成电路的逻辑设计，详细说明某些典型中、大规模集成电路的逻辑功能，并介绍用中、大规模集成电路进行逻辑设计的方法。第七章数字系统设计，主要介绍寄存器传送方法，以及如何用这种方法构成数字系统的一般步骤。学习指导与实验方面的内容包括课程大纲、学习辅导、习题解答与若干具体实验指导和实践考核，编写这些内容的目的是为了便于读者自学和加强实践环节。

本书由毛法尧主编。其中第二、三章和各章习题及习题解答由欧阳星明编写；第一、四章和实验指导及实践考核由任宏萍编写；第五、六、七章和课程大纲及学习辅导由毛法尧编写。再版过程中，华中理工大学出版社给予了大力支持，在此，表示衷心感谢。

由于编者水平有限，书中不妥或错误之处在所难免，殷切希望广大读者批评指正。

编　者

1996年4月于华中理工大学（武汉）

《计算机系列教材》再版说明

我社出版的《计算机系列教材》已有12种相继问世。这12种教材分别是:《逻辑设计》、《PASCAL语言程序设计》、《IBM-PC宏汇编语言程序设计》、《计算机组成原理》、《数据结构》、《计算机操作系统》、《微型计算机及其应用》、《数据库系统概论》、《计算机及其使用基础》、《计算机的操作·应用·维护》、《计算方法》、《FOXBASE程序设计教程》等。这批教材自1990年12月先后出版以来,多数已重印了七八次,深受广大师生、计算机方面自学成才者的欢迎,并获有关专家的好评。其中,《IBM-PC宏汇编语言程序设计》荣获全国电子专业类优秀教材一等奖。

湖北省高等教育自学考试委员会已指定这批教材为湖北省高等教育自学考试计算机及应用专业的专业课教材,其中,《计算机及其使用基础》被指定为现行开考各专业的公共基础课教材。

这批教材都是约请具有丰富教学经验的各课程骨干教师编写的。每种教材都是按照计算机专业该课题教学大纲进行编撰,在内容的剪裁、文字的表述、例题和习题的选择上都力求遵循理论联系实际的原则,面向应用,重视实践,便于自学。每种教材都有"学习指导与实验"部分,其内容包括该课程大纲、学习辅导、习题解答和实验,目的是为了方便读者自学、复习和加强实践环节的指导。

为了适应计算机科学技术飞速发展的新形势,为了满足计算机专业教学的需要和培养新型计算机专业人才的需要,必须更新教材的内容,把已经陈旧、过时的内容删去,增添新的知识。为此,我社约请每种教材的作者们对原教材进行精心修订。这次修订,重点是内容增删,同时根据初版以来作者的教学实践和有关专家提出的改进意见,作出修改。

湖北省高等教育自学考试委员会办公室、华中理工大学成人教育学院有关领导对这批教材的初版和再版给予了大力支持,华中理工大学计算机科学工程系和计算机软件教研室、计算机及应用教研室有关领导和老师们对这批教材的初版和再版给予热情而具体的帮助,在此表示衷心的感谢。

相信这批再版计算机系列教材将会继续受普通高校、成人高等教育自学考试、函授大学、夜大学、广播电视大学、职工大学等计算机类有关专业师生的欢迎。这批系列教材也将继续是广大工程技术人员在职较系统地自学计算机知识颇好的读物。

华中理工大学出版社

1996年5月

目　录

第一章　数制与编码

在数字系统中存在着两种类型的运算，即逻辑运算和算术运算。这两种运算都与数有密切的关系，为此必须对数的表示形式和基本特征有所了解。电子数字计算机是数字系统中最常见的、最有代表性的一种设备，数在计算机系统中是如何表示的，其特征又有哪些呢？我们知道，人类表示数的方法通常是采用十进位计数制。例如，温度“零上 21.5℃，零下 3.8℃”可表示为

$$+21.5℃ \qquad -3.8℃$$

由此可见，要正确表示温度这个物理量数值的大小，必须具有以下 3 个条件。

(1) 选择一组数字符号和组合规则。如上例选用了十进制数的数字符号“1、2、3、5、8”等，并且按照“逢十进一”的规则进行组合。

(2) 确定小数点的符号和位置。

(3) 确定数的正、负符号的表示形式和位置。如上例在正数前面加上符号“+”，在负数前面加上符号“−”。

类似地，计算机表示数的方法，也需要选择一种进位计数制，以确定小数点在机器中的表示以及数的正、负符号的表示。因此，在本章，将以十进制数开始分析，进而讨论一般的进位计数表示、计数规则和各种数制的转换，重点讨论计算机采用的进位计数制——二进制的表示形式和特点。其次介绍正、负数的代码表示以及数的定点表示和浮点表示。最后介绍数的各种编码和编码方法。

1.1　进位计数制

用一组统一的符号和规则来表示数的方法，称为进位计数制，简称数制。日常生活中最常遇到的进位计数制是十进制，在这种数制中，采用了 10 个有序的数字符号 0、1、2、3、4、5、6、7、8、9 和一个小数点符号“.”，按照“逢十进一”的规则进行组合。

如果将数字符号中的数字并列的放在不同位置组成数串，就可以表示一个十进制数。

例如，十进制数 246.52，其中各位数字代表的意义为

2	4	6.	5	2
↑	↑	↑	↑	↑
代表 2×100	代表 4×10	代表 6×1	代表 $5\times\frac{1}{10}$	代表 $2\times\frac{1}{100}$

可以看出，处在不同位置的数字符号有着不同的意义（比如最左边的数字 2 与最右边的数字 2 所代表的意义不同），或者说各位的权（Weight）不同。十进制数 246.52 从左至右各位的权分别是 10^2、10^1、10^0、10^{-1}、10^{-2}。这样 246.52 按权展开的形式如下：

$$246.52=2\times10^2+4\times10^1+6\times10^0+5\times10^{-1}+2\times10^{-2}$$

由以上分析可知，一种进位计数制包含两个基本因素：基数和位权。

(1) 基数：一个数制中所允许使用的数字符号的个数称为该数制的基数。例如，十进制的基数为 10。

(2) 位权：在一个进位计数制表示的数中，不同数位上的固定常数称为位权值，简称位权。例如，十进制个位的位权为 10^0，十位的位权为 10^1，百位的位权为 10^2。

一般说来，对于任意一个十进制数 N，可以有两种表示形式：位置记数表示法和多项式表示法。

(1) 位置计数表示法，也称并列表示法。如十进制数可表示为

$$(N)_{10}=(a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m})_{10} \tag{1-1}$$

其中，n 为整数部分的位数；m 为小数部分的位数；a_i 为数字符号，取值范围为 $0\leqslant a_i\leqslant 9(-m\leqslant i\leqslant n-1)$；下标 10 为十进位计数制的基数。

(2) 多项式表示法，也称按权展开表示法。如十进制数可表示为

$$\begin{aligned}(N)_{10}=&a_{n-1}\times 10^{n-1}+a_{n-2}\times 10^{n-2}+\cdots+a_1\times 10^1+a_0\times 10^0\\&+a_{-1}\times 10^{-1}+a_{-2}\times 10^{-2}+\cdots+a_{-m}\times 10^{-m}\end{aligned} \tag{1-2}$$

或者写成和式

$$(N)_{10}=\left(\sum_{i=-m}^{n-1}a_i\times 10^i\right)_{10} \tag{1-3}$$

比较式(1-1)、(1-2)、(1-3)，可以看出，在十进制位置记数法中，位置为 i 的数字 a_i 具有的位权是 10^i。

在数字系统中使用的进位计数制并不限于十进制数，对于任意的 r 进制来说，数 N 的表示方法也有以上两种形式，即

位置记数表示法：

$$(N)_r=(a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m})_r$$

多项式表示法：

$$\begin{aligned}(N)_r=&a_{n-1}\times r^{n-1}+a_{n-2}\times r^{n-2}+\cdots+a_1\times r^1+a_0\times r^0+a_{-1}\times r^{-1}+a_{-2}\times r^{-2}\\&+\cdots+a_{-m}\times r^{-m}\\=&\left(\sum_{i=-m}^{n-1}a_i\times r^i\right)_r\end{aligned}$$

其中，n 为整数部分的位数；m 为小数部分的位数；a_i 为数字符号，$0\leqslant a_i\leqslant r-1$，$-m\leqslant i\leqslant n-1$；r 为进位计数制的基数。

r 进制的计数规则是“逢 r 进一”。

1.1.1 二进制数的表示

当进位基数 r=2 时，称为二进位计数制，简称二进制。它是数字系统中被广泛采用的一种数制。在二进制中只有 0、1 两个数字符号。例如，由 0、1 组合而成的数 $(1101)_2$ 就是一个二进制数。

二进制的计数规则是“逢二进一”。即每位计满 2 就向相邻高位进一。

对于任意一个二进制数 N，用位置计数表示法表示为

$$(N)_2=(a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m})_2$$

用多项式表示法表示为

$$(N)_2=a_{n-1}\times 2^{n-1}+a_{n-2}\times 2^{n-2}+\cdots+a_1\times 2^1+a_0\times 2^0$$

$$+a_{-1}\times 2^{-1}+a_{-2}\times 2^{-2}+\cdots+a_{-m}\times 2^{-m}$$
$$=\sum_{i=-m}^{n-1}a_i\times 2^i$$

其中，a_i 为数码 0 或 1；n 为整数部分的位数；m 为小数部分的位数。

通常，对二进制数的表示，可以在数字右下角标注 2 或 B。

1.1.2 八进制数的表示

当进位基数 r＝8 时，称为八进制。通常是从十进制数字符号中借用 r 个数字作为 r 进制的数字符号。例如，八进制采用的数字符号是 0，1，2，3，4，5，6，7。八进制的计数规则是“逢八进一”。数$(47.6)_8$就代表一个八进制数。任意一个八进制数 N 的多项式表示为

$$(N)_8=a_{n-1}\times 8^{n-1}+a_{n-2}\times 8^{n-2}+\cdots+a_1\times 8^1+a_0\times 8^0+a_{-1}\times 8^{-1}+\cdots+a_{-m}\times 8^{-m}$$
$$=\sum_{i=-m}^{n-1}a_i\times 8^i$$

其中，a_i 是 0～7 中的任何一个数字。

1.1.3 十六进制数的表示

十六进制的进位基数 r＝16。当基数 r＞10 时，除借用十进制的十个数字符号外，往往还借用英文字母作补充。例如，十六进制使用的十六个数字符号是 0，1，2，3，4，5，6，7，8，9，A，B，C，D，E，F。其中 A，B，C，D，E，F 分别代表十进制数字 10，11，12，13，14，15。十六进制的计数规则是“逢十六进一”。数$(54AF.2B)_{16}$就是一个十六进制数。

任意一个十六进制数 N 的多项式表示为

$$(N)_{16}=a_{n-1}\times 16^{n-1}+a_{n-2}\times 16^{n-2}+\cdots+a_1\times 16^1+a_0\times 16^0+a_{-1}\times 16^{-1}+\cdots+a_{-m}\times 16^{-m}$$
$$=\sum_{i=-m}^{n-1}a_i\times 16^i$$

其中，a_i 是 0～9，A～F 中的任何一个数字或字母。

表 1.1 中列出了 r 为 10，2，8，16 时各种数制的开头的 17 个自然数。

表 1.1 四种数制表示几个选定的数

十进制数	二进制数	八进制数	十六进制数	十进制数	二进制数	八进制数	十六进制数
0	0000	0	0	9	1001	11	9
1	0001	1	1	10	1010	12	A
2	0010	2	2	11	1011	13	B
3	0011	3	3	12	1100	14	C
4	0100	4	4	13	1101	15	D
5	0101	5	5	14	1110	16	E
6	0110	6	6	15	1111	17	F
7	0111	7	7	16	10000	20	10
8	1000	10	8				

1.1.4 二进制的特点

选择什么样的数制来表示数，这对数字系统的成本和性能均有很大的影响。在数字系统

中，常用二进制来表示数和进行运算，其原因是二进制具有如下特点：

（1）二进制只有 0 和 1 两个数字符号，容易物理实现。比如可用电压的高、低这两种不同的状态来表示二进制的数字符号 1、0；也可用脉冲的有、无来表示二进制的 1、0。如果数字系统采用十进制，则需 10 种不同的状态来表示十进制的 10 个数字符号，这显然是不容易实现的。

（2）二进制运算规则简单。二进制的运算规则是：

加法规则

$0+0=0$　　$0+1=1$

$1+0=1$　　$1+1=0$（同时向相邻高位进 1）

减法规则

$0-0=0$　　$0-1=1$（同时向相邻高位借 1）

$1-0=1$　　$1-1=0$

乘法规则

$0\times0=0$　　$0\times1=0$

$1\times0=0$　　$1\times1=1$

除法规则

$0\div1=0$　　$1\div1=1$

下面举几个二进制数运算的例子。

例 1.1　进行 1101+1011 运算。

解

$$\begin{array}{r} 1101 \\ +)\ 1011 \\ \hline 11000 \end{array}$$

两个二进制数的加法运算和十进制数的加法运算相似，但采用“逢二进一”的规则，每位数累计到 2 时，本位就记为 0，且向相邻高位进 1。

例 1.2　进行 11101−10011 运算。

解

$$\begin{array}{r} 11101 \\ -)\ 10011 \\ \hline 1010 \end{array}$$

二进制减法采用“借一当二”的法则，减法运算从低位起按位进行，在遇到 0 减 1 时，就要向相邻高位借 1，也就是从高位减去 1。

例 1.3　进行 1101×1011 运算。

解

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

二进制数的乘法运算和十进制数的乘法运算相似，所不同的是对部分积进行累加时要按“逢二进一”的原则。

例 1.4 进行 10010001÷1011 运算。

解

```
                1 1 0 1…商
1 0 1 1 ) 1 0 0 1 0 0 0 1
          1 0 1 1
          ---------
            1 1 1 0
            1 0 1 1
            -------
              1 1 0 1
              1 0 1 1
              -------
                1 0…余数
```

二进制的除法运算与十进制数的除法运算类似，但采用二进制数的运算规则。

(3) 二进制便于进行逻辑设计。二进制的数字符号 0、1，可与逻辑代数中逻辑变量的“假”、“真”对应起来。也就是说，可用一个逻辑变量来表示一个二进制数位。这样，就可以用逻辑代数的理论来设计二进制数字系统。(这一点将在后面几章中看到)

二进制的缺点是书写长，读或记很不方便。因此，人们也常采用八进制和十六进制作为辅助。因为八进制和十六进制数容易书写、阅读，便于记忆，且与二进制数的转换也十分容易。

下面将介绍这几种常用数制之间的转换。

1.2 数制转换

人们习惯使用十进制数，而计算机和其他数字系统内部大多采用二进制，因此，在信息处理过程中，计算机首先要把十进制数转换成二进制数，然后，进行加工处理，最后，还需将二进制数表示的结果信息转换成人们习惯的十进制数。这里就存在一个不同数制之间的转换问题。

所谓数制转换，是指一个数从一种数制的表示形式转换成等值的另一种数制的表示形式。数制转换的具体方法大致有两种，即多项式替代法和基数乘除法。前者用于将非十进制数转换为十进制数，后者则用于将十进制数转换为非十进制数。

1.2.1 多项式替代法

所谓多项式替代法，是将 r 进制数按多项式表示法展开，然后按十进制的运算规则求和，即可得到与 r 进制数等值的十进制数。

例 1.5 将二进制数 1 0 1 1 0 1.1 转换成十进制数。

解 首先，将二进制数的位置计数表示式展开成多项式表示式，则有

$$(101101.1)_2=(1\times2^5+0\times2^4+1\times2^3+1\times2^2+0\times2^1+1\times2^0+1\times2^{-1})_{10}$$

再将等式右边按十进制运算规则进行计算，得

$$(101101.1)_2=(32+8+4+1+0.5)_{10}=(45.5)_{10}$$

上述方法也适合于其他数制的数转换成十进制数。

例 1.6 $(147)_8=(?)_{10}$

解

$$(147)_8=(1\times8^2+4\times8^1+7\times8^0)_{10}$$
$$=(64+32+7)_{10}=(103)_{10}$$

例 1.7 $(1AF8)_{16}=(?)_{10}$

解

$$(1AF8)_{16}=(1\times16^2+A\times16^1+F\times16^0+8\times16^{-1})$$
$$=(1\times256+10\times16+15\times1+8\times16^{-1})_{10}$$

$$=(256+160+15+0.5)_{10}$$
$$=(431.5)_{10}$$

注意，十六进制数在转换成十进制数时，应将其中的字母 A～F 转换成相应的十进制数。

1.2.2 基数乘除法

基数乘除法用于将十进制数转换为非十进制数，其中基数乘法用于小数部分的转换，基数除法用于整数部分的转换。下面分别介绍这两种方法。

1. 基数除法

例 1.8 将十进制数 25 转换为二进制数。

解 将十进制整数逐次除以基数 2，直到商为 0 为止，然后将每次的余数按相反顺序排列起来，即为等值的二进制整数。其转换过程可简记为“除 2 倒取余”。即

	商	余数
2	25	…1…a_0 （最低位）
2	12	…0…a_1
2	6	…0…a_2
2	3	…1…a_3
2	1	…1…a_4 （最高位）
	0	

故有$(25)_{10}=(a_4a_3a_2a_1a_0)_2=(11001)_2$。

上述将十进制整数转换为二进制数的方法可推广用于将十进制整数转换到其他任意非十进制数。

例 1.9 将十进制数 850 转换为八进制数。

解

		余数	
8	850	… 2	（最低位）
8	106	… 2	
8	13	… 5	
8	1	… 1	（最高位）
	0		

故有$(850)_{10}=(1522)_8$。

例 1.10 将十进制数 698 转换为十六进制数。

解

		余数	
16	698	… 10	（最低位）
16	43	… 11	
16	2	… 2	（最高位）
	0		

由于$(10)_{10}=(A)_{16}$，$(11)_{10}=(B)_{16}$，$(2)_{10}=(2)_{16}$

所以$(698)_{10}=(2BA)_{16}$。

2. 基数乘法

例 1.11 将十进制数 0.625 转换为二进制小数。

解 将十进制小数部分逐次乘以基数 2，直到乘积的小数部分为 0（精确转换）；或者小数部分虽不为 0，但二进制小数的位数已达到精度要求（近似转换）为止。然后将每次乘积的整数部分顺序排列起来，即为等值的二进制小数。其转换过程可简记为“乘 2 顺取整”。即

$$
\begin{array}{rl}
0.625 & \\
\times) \quad 2 & \\
\hline
\boxed{1}.250 & \text{整数 } 1(a_{-1}) \text{ 最高小数位} \\
\times) \quad 2 & \\
\hline
\boxed{0}.500 & \text{整数 } 0(a_{-2}) \\
\times) \quad 2 & \\
\hline
\boxed{1}.000 & \text{整数 } 1(a_{-3}) \text{ 最低小数位}
\end{array}
$$

此时乘积的小数部分为 0，因此，不再继续乘以基数 2。将每次乘积的整数部分顺序排列，即可得到

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

必须指出：竖式中的整数不参加连乘。

在十进制的小数部分转换中，有时连续乘 2 不一定能使小数部分等于 0，这说明该十进制小数不能用有限位二进制小数表示。这时，只要取足够多的位数，使其误差达到所要求的精度就可以了。

例 1.12 将十进制数 0.18 转换成二进制数，精确到小数点后 4 位。

解

$$
\begin{array}{rl}
0.18 & \\
\times) \quad 2 & \\
\hline
\boxed{0}.36 & \text{整数 } 0(a_{-1}) \\
\times) \quad 2 & \\
\hline
\boxed{0}.72 & \text{整数 } 0(a_{-2}) \\
\times) \quad 2 & \\
\hline
\boxed{1}.44 & \text{整数 } 1(a_{-3}) \\
\times) \quad 2 & \\
\hline
\boxed{0}.88 & \text{整数 } 0(a_{-4}) \\
\times) \quad 2 & \\
\hline
\boxed{1}.76 & \text{整数 } 1(a_{-5})
\end{array}
$$

当$(0.18)_{10}$乘了四次基数 2，并去掉整数部分后，其乘积为$(0.88)_{10}$，小数部分仍不为 0。此题要求近似转换需精确到小数点后四位，因此，将$(0.88)_{10}$再乘一次 2，这样得到

$$(0.a_{-1}a_{-2}a_{-3}a_{-4}a_{-5})_2=(0.00101)_2$$

四舍五入后得

$$(0.18)_{10}\approx(0.0011)_2$$

用类似的方法可完成将十进制小数部分转换为非十进制小数。

例 1.13 将十进制数 0.15 转换为八进制数。

解 将十进制小数部分逐次乘以基数 8，即

$$\begin{array}{r} 0.25 \\ \times) \quad 8 \\ \hline \boxed{2}.00 \qquad \cdots 2 \end{array}$$

因为小数部分为 0，故得$(0.25)_{10}=(0.2)_8$。

如果一个十进制数既有整数部分又有小数部分，那么，转换时要分别按基数除法和基数乘法进行转换，然后再把转换的结果合并起来。

例 1.14 将$(25.625)_{10}$转换为二进制数。

解

$$\begin{aligned}(25.625)_{10}&=(25)_{10}+(0.625)_{10}\\&=(11001)_2+(0.101)_2\\&=(11001.101)_2\end{aligned}$$

即$(25.625)_{10}=(11001.101)_2$。

例 1.15 将$(46.4)_{10}$转换为八进制数。

解

整数部分

$$\begin{array}{r|l} 8 & 46 \quad \cdots \quad 6 \uparrow \\ 8 & 5 \quad \cdots \quad 5 \\ \hline & 0 \end{array}$$

小数部分

$$\begin{array}{r} 0.4 \\ \times) \quad 8 \\ \hline \boxed{3}.2 \quad \cdots \quad 3 \\ \times) \quad 8 \\ \hline \boxed{1}.6 \quad \cdots \quad 1 \\ \times) \quad 8 \\ \hline \boxed{4}.8 \quad \cdots \quad 4 \\ \vdots \end{array}$$

故有$(46.4)_{10}=(56.314\cdots)_8$。

例 1.16 将$(54.3)_{10}$转换为十六进制数。

解

整数部分

$$\begin{array}{r|l} 16 & 54 \quad \cdots \quad 6 \uparrow \\ 16 & 3 \quad \cdots \quad 3 \\ \hline & 0 \end{array}$$

小数部分

$$\begin{array}{r} 0.3 \\ \times) \quad 16 \\ \hline \boxed{4}.8 \quad \cdots \quad 4 \\ \times) \quad 16 \\ \hline \boxed{12}.8 \quad \cdots \quad 12 \\ \times) \quad 16 \\ \hline \boxed{12}.8 \quad \cdots \quad 12 \\ \vdots \end{array}$$

故有$(54.3)_{10}=(36.4CC\cdots)_{16}$。

1.2.3 2^n 进制之间数的转换

1. 二进制数与八进制数之间的转换

因为八进制的基数 r=8，而$8^1=2^3$（注意指数），由于这两种进位制的权之间有这种内在的

联系，因而它们之间的转换比较简单。$8^1=2^3$，也就是说每一位八进制数相当于三位二进制数，或反之。例如，一位八进制数$(5)_8$与三位二进制数$(101)_2$相互对应。

二进制数转换成八进制数的规则是：以小数点为界向左、右两边每三位分成一组，不满三位者加0。每组以其对应的八进制数符代替，并顺序排列，即为变换后的等值八进制数。

例1.17 $(11011110.1)_2=(?)_8$

解 以二进制数11011110.1的小数点为界，向左、向右每三位分一组，最左边和最右边的组如果不够三位，则需加0，即

二进制数：011 011 110 . 100

八进制数： 3 3 6 . 4

即$(11011110.1)_2=(336.4)_8$。

八进制数转换成二进制数的规则是，每位八进制数用相应的三位二进制数来表示即可。

例1.18 $(54.32)_8=(?)_2$

解 $(54.32)_8=101/100/.011/010$

$=(101100.011010)_2$

2. 二进制数与十六进制数之间的转换

因为十六进制的基数$r=16$，而$16^1=2^4$（注意指数），因此，一位十六进制数相当于四位二进制数，或者说，一位十六进制的数可用四位二进制数来表示，或反之。

例1.19 将$(101101.011)_2$转换成十六进制数。

解 以二进制数101101.011的小数点为界，向左、向右每四位分成一组，最左边和最右边的组如果不够四位，则需加0。

二进制数：0010 1101 . 0110

十六进制数：2 D 6

即$(101101.011)_2=(2D.6)_{16}$。

例1.20 将十六进制数$(E\ F.16)_{16}$转换成二进制数。

解 将十六进制数的各位用相应的四位二进制数代替。即

$$(E\ F.16)_{16}=1110/1111/.0001/0110$$

$$=(11101111.00010110)_2$$

在基数为2^n的数制中，其他数制之间数的转换可用二进制数作桥梁，使转换变得非常简单。

例如，将十六进制数$(E\ F.16)_{16}$转换成八进制数，可先将$(E\ F.16)_{16}$转换成相应的二进制数，然后再将该二进制数转换成八进制数。

由上例可知，$(E\ F.16)_{16}=(11101111.00010110)_2$，只需求$(11101111.00010110)_2=(?)_8$即可。

因为 二进制数：011 101 111 . 000 101 100

所以 八进制数： 3 5 7 . 0 5 4

故$(E\ F.16)_{16}=(357.054)_8$。