高等学校教材

数字逻辑

（第二版）

鲍家元　毛文林

高等教育出版社

高等学校教材

数 字 逻 辑

（第二版）

鲍家元　毛文林

高等教育出版社

内容提要

全书共分八章。第一章、第二章作为数字逻辑的理论基础，讨论了数制、编码和逻辑代数基础。第三章至第六章在小规模集成电路分析和设计的基础上，讨论组合逻辑和时序逻辑技术中的基本概念、基本方法以及工程实践中的文档和工程设计的问题，并以较大篇幅介绍了一些常用的、具有代表性的 MSI 器件原理、设计和应用。第七章讨论了可编程逻辑器件 PLD，以可编程阵列逻辑 PAL 为重点讨论其逻辑结构和应用特点。第八章对系统级逻辑设计的方法及描述工具 ASM 和 MDS 做了介绍，并以数字系统控制器设计为重点，讨论了系统设计思想及描述工具的应用。

本书可作为计算机、电子工程及应用数字技术的各类专业本科生的教材。

图书在版编目(CIP)数据

数字逻辑/鲍家元，毛文林.—2 版.—北京：高等教育出版社，2002.9(2006.重印)

本科计算机专业教材

ISBN 7-04-010595-0

Ⅰ.数... Ⅱ.①鲍...②毛... Ⅲ.数字逻辑-高等学校-教材 Ⅳ.TP302.2

中国版本图书馆 CIP 数据核字(2002)第 021843 号

责任编辑 康兆华　**封面设计** 王凌波　**责任绘图** 朱　静　**版式设计** 胡志萍　王　莹
责任校对 俞声佳　**责任印制** 毛斯璐

出版发行	高等教育出版社	**购书热线**	010-58581118
社　　址	北京市西城区德外大街 4 号	**免费咨询**	800-810-0598
邮政编码	100011	**网　　址**	http://www.hep.edu.cn
总　　机	010-58581000		http://www.hep.com.cn
		网上订购	http://www.landraco.com
经　　销	蓝色畅想图书发行有限公司		http://www.landraco.com.cn
印　　刷	唐山市润丰印务有限公司	**畅想教育**	http://www.widedu.com
		版　　次	1997 年 7 月第 1 版
开　　本	787×1092　1/16		2002 年 9 月第 2 版
印　　张	24	**印　　次**	2006 年 12 月第 6 次印刷
字　　数	580 000	**定　　价**	29.90 元

物料号　10595-00

第二版前言

本教材初版于1997年7月由高等教育出版社出版,因其突出的特点而受到高校师生普遍欢迎。通过四年来的试用,编者广泛听取意见,获得不少宝贵的修改建议;随着高等教育改革的深入开展,课程的设置和内容也发生了一些调整和变化;由于计算机相关课程教育扩大到高等院校的各专业,教材的适用范围也在发生变化。这些都要求对教材进行认真的修改,以适应新世纪对高等教育重点教材的要求。

这次修编工作有以下几点特色:

1. 精炼内容,突出教材特点。

高等教育改革发展迅猛,尤其对于计算机专业,相关的课程设置和课程内容在不断地调整。例如,许多院校开设了"大规模集成电路设计"、"数字系统设计"、"硬件描述语言"等选修课和"数字逻辑课程设计"实验课,在其教材和实验指导书中对课程内容都有详细的讲解。在本书中,则对相关内容进行了删节和浓缩,如初版第七章中有关ABEL语言的论述和第九章的"数字系统中CAD技术及其他设计技术"等。这就使本书的篇幅得到大幅度压缩。

2. 注重学生能力的培养。

在第二版中进一步突出了对学生能力的培养,不仅提出了问题及解答范例,而且分析其产生的规律和解决方法。例如,时序电路设计中的"电路挂起故障排除"是设计者无法回避的一个重要问题,也是学生学习的难点。本次增加了这一部分内容,希望学生在深入解剖典型电路的基础上掌握其设计规律,并鼓励学生寻找新的解决方法。书中内容不必全部在课堂上讲授,大量的应用例子和参考内容可由学生自己阅读,以提高学生的自学能力。

3. 扩大教材适用范围。

考虑到其他专业学生学习本课程的要求,对初版书中部分内容进行了补充和改写,以避免学生因某些前期知识的缺乏而产生学习困难。如增加了一些与数字电路相关的内容。

4. 对初版书中各类错误做了全面修订。

本书授课时数为64学时,配套实验或实验课另行安排。

本书第二版由毛文林执笔,鲍家元教授对本书修订提出了指导意见。限于修编者的水平,书中疏漏和不足之处在所难免,恳请读者批评指正。

毛文林

2001年12月

于西安交通大学

初版前言

本教材是由国家教育委员会高等学校理科计算机科学教学指导委员会计算机及应用教材建设组评选推荐出版。

本教材是按照1992年召开的全国数字逻辑教材详细大纲审定会制定的《数字逻辑教材详细大纲》,并参考美国《ACM/IEEE-CS 91教程》中"数字逻辑"及"数字系统"知识单元所提出的专题及主要概念编写而成。

数字逻辑课程是计算机科学专业、电子工程类专业及主要应用数字技术的各类专业的技术基础课程。它的先修课程是"电子技术基础"及"程序设计语言",后续课程有"计算机组成原理"、"微机原理与接口技术"等。

数字逻辑是数字技术实践的基础。它涉及数字技术中的基本原理、基本分析与设计方法,具有很强的工程实践性。因此,本教材是按照数字逻辑设计原理及实践的指导思想进行编写的。本教材与国内已出版的同类教材在编写上有以下三点差异:其一,在重点讨论数字逻辑的基本原理、基本分析和设计方法的基础上,辅以大量的工程实践举例(特别是中、大规模集成电路设计及应用举例),使读者更好地理解和掌握基本分析和设计方法并灵活应用,更好地培养工程实践能力;其二,强调硬件逻辑设计及软件逻辑设计的结合,这是近十年来数字逻辑设计的发展趋势;其三,在讨论基本原理之后,给出一些推演性问题及应用举例,而且教材中有意不做详细讨论,留给读者去思考、去完善,以培养学生独立分析问题和解决问题的能力。在编写本教材时,力求反映当代已在工程实践中应用的数字逻辑新技术。

全书共分九章。第一、二章作为数字逻辑的理论基础,讨论了数制、编码和逻辑代数基础。第三、四、五、六章在小规模集成电路分析和设计的基础上,讨论组合逻辑和时序逻辑技术中的基本概念、基本方法以及工程实践中文档和设计中的问题,并以较大篇幅介绍了一些常用的、具有代表性的MSI器件原理、设计和应用。第七章讨论了可编程逻辑器件PLD,以可编程阵列逻辑PAL为重点讨论其逻辑结构,并较详细介绍了一种广为应用的编程语言ABEL及其编程应用。第八章对系统级逻辑设计的方法及描述工具ASM和MDS做了介绍,并以数字系统控制器设计为重点,讨论了系统设计思想及描述工具的应用。第九章是数字系统逻辑设计更高一级设计方法和专用集成电路ASIC的概述。在本书中,除突出讨论了数字逻辑中的一系列概念、原理和方法外,还在全书中反复强调了标准化,信号按时间排序,抽象模型,系统的模块化,大系统的复杂性、可靠性,次佳设计和折衷等概念。这些都是实际工程设计中必须建立的重要思想。

本书中逻辑符号采用国标(GB4728.12-85)及国外常用逻辑符号(MIL-STD-806标准)两种表示方法,目的是使读者熟悉两种符号,便于查阅国内外技术文献。

本书是作者根据长期从事教学及工程实践的体会编写而成。本书力求保持数字逻辑在内容上的完整性、先进性及工程实践性。本书中至少有三分之一的内容(如大量的应用举例)不必在课堂上讲授而让学生自学,以培养学生的自学能力及独立钻研能力。全书的授课时数约为70学

时，对计算机软件专业，可将目录中打"＊"号的章、节删去，这样授课时间大约需要45学时。

本书第一章由毛文林提出初稿，第二至九章由李跃提出初稿，鲍家元对第二至六章和第八章的初稿进行了大幅度的修改、补充并统编全书。本书在编写指导思想和编写大纲上得到清华大学王尔乾教授的许多帮助；本书稿承北京理工大学刘明业教授审阅，刘教授以其深厚的理论造诣和丰富的实践经验对书稿提出了许多宝贵修改意见；在此一并表示衷心的感谢。

由于未能收集到更多的国内外最新教材和参考文献，加之限于作者水平，本书中错误及不妥之处难以避免，敬请读者和专家指正。

鲍家元
1996年5月
于西安交通大学

目　录

第一章 数制和编码

1.1 进位计数制

在各种进位计数制中，十进计数制（十进制）是人们最熟悉的。分析十进计数制，从中可以总结出以下特点：

(1) 必须具有 10 个有序的数字符号：0、1、2、3、4、5、6、7、8、9 和小数点符号“.”。

(2) 遵循“逢十进一”的计数规则。

十就是这种计数制的进位基数，或称基数。这样的若干符号并列在一起就可以表示一个十进制数。例如，378.65 共有 5 位数字，最左面为百位（3 代表 300），其后第二位为十位（7 代表 70），第三位为个位（8 在这里就代表 8），小数点右边的第一位为十分位（6 代表$\frac{6}{10}$），小数点右边第二位为百分位（5 代表$\frac{5}{100}$）。可见，数字符号在不同的位置代表着不同的数值，即有着不同的“权”(weight)。378.65 实质上可以表示成下列形式：

$$378.65 = 3\times10^2 + 7\times10^1 + 8\times10^0 + 6\times10^{-1} + 5\times10^{-2} \tag{1.1}$$

上式左边的形式，通常称之为十进制数的位置记数法，也叫并列表示法。上式右边的形式，通常称之为十进制数的多项式表示法，也叫按权展开式。

一般来说，任何一个十进制数 N，可以表示成：

$$(N)_{10} = (K_{n-1}K_{n-2}\cdots K_1K_0.K_{-1}K_{-2}\cdots K_{-m})_{10} \tag{1.2}$$

其中，n 表示整数位数，m 表示小数位数，K_i 是十进制中十个数字中的任何一个，即

$$0\leqslant K_i\leqslant 9$$

括号外的下标为进位制的基数，本例中“10”代表十进制。

同样，数 N 也可用多项式表示法写为：

$$\begin{aligned}(N)_{10} &= K_{n-1}(10)^{n-1} + K_{n-2}(10)^{n-2} + \cdots + K_1(10)^1 + K_0(10)^0 \\ &\quad + K_{-1}(10)^{-1} + K_{-2}(10)^{-2} + \cdots + K_{-m}(10)^{-m} \\ &= \sum_{i=-m}^{n-1} K_i\cdot 10^i \end{aligned} \tag{1.3}$$

对其他进位计数制来说，也具有上述特点。如基数为 R 的进位计数制，必须满足：

(1) 具有 R 个有序的数字符号：$0,1,2,\cdots,R-1$ 及小数点符号“.”。

(2) 遵循“逢 R 进一”的计数规则。

对 R 进制中的数 N，可用位置计数法表示为：

$$(N)_R = (A_{n-1}A_{n-2}\cdots A_1A_0.A_{-1}A_{-2}\cdots A_{-m})_R \tag{1.4}$$

该数 N 也可用多项式表示法写为

$$\begin{aligned}(N)_R &= (A_{n-1}R^{n-1} + A_{n-2}R^{n-2} + \cdots + A_1R^1 + A_0R^0 \\ &\quad + A_{-1}R^{-1} + A_{-2}R^{-2} + \cdots + A_{-m}R^{-m})_R \\ &= (\sum_{i=-m}^{n-1} A_i \cdot R^i)_R\end{aligned} \tag{1.5}$$

其中，n 表示整数位数，m 表示小数位数，A_i 是 R 进制中的数字符号之一，即

$$0 \leqslant A_i \leqslant R-1$$

因而，式(1.5)中括号里的 R，在写成 R 进制的数字符号时，已不能仅仅用一位数字符号来表示，而应记做“10”。括号外的 R，是计数制的基数标志，用十进制数字表示。如 $R=10$，则括号和记数标志均可省去。

表 1.1 列出几种进位制整数数列前面的一部分。从这个表上可以比较出这几种进位制数值的对应关系。例如：

$$(14)_{10} = (1110)_2 = (112)_3 = (32)_4 = (16)_8 = (E)_{16}$$

表 1.1　数制表(整数)

$R=10$	$R=2$	$R=3$	$R=4$	$R=8$	$R=16$
0	0	0	0	0	0
1	1	1	1	1	1
2	10	2	2	2	2
3	11	10	3	3	3
4	100	11	10	4	4
5	101	12	11	5	5
6	110	20	12	6	6
7	111	21	13	7	7
8	1 000	22	20	10	8
9	1 001	100	21	11	9
10	1 010	101	22	12	A
11	1 011	102	23	13	B
12	1 100	110	30	14	C
13	1 101	111	31	15	D
14	1 110	112	32	16	E
15	1 111	120	33	17	F
16	10 000	121	100	20	10
17	10 001	122	101	21	11
⋮	⋮	⋮	⋮	⋮	⋮

基数 $R=2$，是二进位计数制(二进制)。由于二进制中仅有两个数字符号 0 和 1，所以很容易用电子元件特性来表示，因而它是计算机运算的基础。在二进制中，相应的运算规则有：

加法	乘法
$0+0=0$	$0\times0=0$
$0+1=1+0=1$	$0\times1=1\times0=0$
$1+1=10$	$1\times1=1$

这种运算的简便性导致了完成运算的电路的简化和控制的简化。

掌握二进制数的一个基本要求，是熟悉二进制的权(2 的幂)，必须像熟悉 10 的幂那样熟悉 2 的幂。表 1.2 列出了部分常用的二进制的权。

表 1.2 二进制($R=2$)各位的权(R^i)

i	R^i	i	R^i	i	R^i
−7	0.007 812 5	0	1	7	128
−6	0.015 625	1	2	8	256
−5	0.031 25	2	4	9	512
−4	0.062 5	3	8	10	1 024
−3	0.125	4	16	11	2 048
−2	0.25	5	32	12	4 096
−1	0.5	6	64	13	8 192

1.2 进位计数制的相互转换

将一个数从一种进位计数制表示法转换成另外一种进位计数制表示法，称为数制转换。除了基数为 2^K 的进位制之间转换可用直接转换法外，其他进位制之间均需采用数学计算的方法来转换。通常用于数制转换的两种方法是多项式替代法和基数乘除法，这两种方法有不同的适用范围。

1.2.1 多项式替代法

若将 α 进制(基数 $R=\alpha$)的数$(N)_\alpha$ 转换成β 进制(基数 $R=\beta$)的数$(N)_\beta$，即从$(N)_\alpha$ 求得$(N)_\beta$，可以先用公式(1.5)将$(N)_\alpha$ 写成多项式：

$$(N)_\alpha=(A_{n-1}10^{n-1}+A_{n-2}10^{n-2}+\cdots+A_1 10^1+A_0 10^0+A_{-1}10^{-1}+\cdots+A_{-m}10^{-m})_\alpha \tag{1.6}$$

式中 10 即 α 进制中的基数α。将上式右边的所有数值符号 $A_i(0\leqslant A_i\leqslant\alpha-1)$和 10 替换成 β 进制中的相应值，并按 β 进制的计算规则计算，所得结果就是$(N)_\beta$，即：

$$(N)_\beta=(A_{n-1}\alpha^{n-1}+A_{n-2}\alpha^{n-2}+\cdots+A_1\alpha^1+A_0\alpha^0+A_{-1}\alpha^{-1}+\cdots+A_{-m}\alpha^{-m})_\beta \tag{1.7}$$

例 1 将$(1CE8)_{16}$转换为十进制。

$N=(1\times10^3+C\times10^2+E\times10^1+8\times10^0)_{16}$	括号中 10 为 16 进制基数
$=(1\times16^3+12\times16^2+14\times16^1+8\times16^0)_{10}$	转换成十进制符号
$=(7\,400)_{10}$	十进制计算结果
$=7\,400$	十进制基数符号可省

例 2 将$(121.2)_3$转换为二进制。

$$N=(1\times10^2+2\times10^1+1\times10^0+2\times10^{-1})_3$$
$$=(1\times11^2+10\times11^1+1\times11^0+10\times11^{-1})_2$$
$$=(1\,001+110+1+0.101010\cdots)_2$$
$$=(10000.101010\cdots)_2$$

其中的计算是按二进制计算规则进行的，如：

$$10\times11^{-1}=10\div11=0.\dot{1}0101\dot{0}\cdots$$

是循环小数，故$(121.2)_3=(10000.101010\cdots)_2$

从上面两例可以看出，用多项式替代法实现$(N)_\alpha$转换为$(N)_\beta$，计算是在β进制中进行的，因此读者必须熟悉β进制的运算。换句话说，把其他进制的数转换为十进制数时，采用多项式替代法就很方便。如要把十进制数转换为其他进制数时，可以采用基数乘除法。

1.2.2 基数乘除法

与多项式替代法相反，用基数乘除法将$(N)_\alpha$转换成$(N)_\beta$时，计算是在α进制中进行的。整数的转换和小数的转换方法不同，整数转换要用基数除法，而小数的转换要用基数乘法。若一个数包括整数和小数两部分，可以将它们分别转换，然后合并起来。

1. 整数转换(基数除法)

假设α进制表示的整数$(N)_\alpha$转换为β进制的整数$(N)_\beta$的并列记数法表示为：

$$(N)_\beta=(b_{n-1}\cdots b_1b_0)_\beta$$

若用多项式表示法表示，应为：

$$(N)_\beta=(b_{n-1}\cdot10^{n-1}+\cdots+b_1\cdot10^1+b_0\cdot10^0)_\beta$$

如果把其中的$b_i(0\leqslant b_i\leqslant\beta-1)$和 10 都换成$\alpha$进制的相应值，则这个数在$\alpha$进制中可以表示为：

$$(N)_\alpha=(b_{n-1}\cdot\beta^{n-1}+\cdots+b_1\cdot\beta^1+b_0\cdot\beta^0)_\alpha \tag{1.8}$$

上式左、右两边都表示是在α进制中。如通过α进制的计算，确定每个b_i，再分别转换为β进制的相应值，就求出了在β进制中的$(N)_\beta$。

为了方便起见，这里把式(1.8)表示为：

$$N=((\cdots((b_{n-1})\cdot\beta+b_{n-2})\cdot\beta+\cdots)\cdot\beta+b_1)\cdot\beta+b_0 \tag{1.9}$$

将上式除以β，可得$(\cdots((b_{n-1})\cdot\beta+b_{n-2})\cdot\beta+\cdots)\cdot\beta+b_1$和余数$b_0$；若将所得商再除以$\beta$，又可得到余数$b_1$；重复以上过程，直至求得最后一个余数$b_{n-1}$。

例 1 将十进制的 179 转换成二进制数。

$179\div2=89$ 余 $1\to b_0$

$89 \div 2 = 44$ 余 $1 \rightarrow b_1$

$44 \div 2 = 22$ 余 $0 \rightarrow b_2$

$22 \div 2 = 11$ 余 $0 \rightarrow b_3$

$11 \div 2 = 5$ 余 $1 \rightarrow b_4$

$5 \div 2 = 2$ 余 $1 \rightarrow b_5$

$2 \div 2 = 1$ 余 $0 \rightarrow b_6$

$1 \div 2 = 0$ 余 $1 \rightarrow b_7$

即 $(179)_{10} = (10110011)_2$

例 2 将$(3\ 417)_{10}$转换成十六进制数。

这里可将上例的除法表示简化为：

$16 \underline{|3\ 417}$ 余 $9 \rightarrow b_0$

$16 \underline{|213}$ 余 $5 \rightarrow b_1$

$16 \underline{|13}$ 余 13(即 D)$\rightarrow b_2$

0

即 $(3\ 417)_{10} = (\mathrm{D}59)_{16}$

2. 小数转换(基数乘法)

从一种进制转化到另一种进制时，小数的位数变化是很大的，有时甚至不可能用有限位数准确地实现转换。如$\frac{1}{3}$在三进制中仅一位小数，而在十进制中则是无限循环小数，即

$$(0.1)_3 = (0.3333\cdots)_{10}$$

当小数$(N)_\alpha$采用基数乘法转换为$(N)_\beta$时，$(N)_\beta$的小数位数到底应取多少呢？这里有两种情况：

一是β进制的小数位数已确定，如人为规定或设备限定，则应根据要求转换。

二是要使$(N)_\beta$的精度不低于$(N)_\alpha$的精度，即要根据$(N)_\alpha$的位数来确定$(N)_\beta$的位数(见本节 1.2.5)。

现设小数$(N)_\alpha$转换为β进制后记做$(N)_\beta$，并取 m 位，用位置记数法和多项式表示法记为：

$$(N)_\beta = (0.C_{-1}C_{-2}\cdots C_{-m})_\beta$$

和

$$(N)_\beta = (C_{-1}\cdot 10^{-1} + C_{-2}\cdot 10^{-2} + \cdots + C_{-m}\cdot 10^{-m})_\beta \tag{1.10}$$

如将式(1.10)中的 C_{-i}($1 \leqslant i \leqslant m, 0 \leqslant C_{-i} \leqslant \beta - 1$)和 10 转换成 α 进制中的相应值 C_{-i} 和 β，则$(N)_\beta$在α进制中可表示为：

$$(N)_\alpha = (C_{-1}\cdot \beta^{-1} + C_{-2}\cdot \beta^{-2} + \cdots + C_{-m}\cdot \beta^{-m})_\alpha \tag{1.11}$$

由于式(1.11)左、右两边均以 α 进制表示，所以可通过 α 进制的计算，求出每个 C_{-i}，再把它们分别转换成 β 进制的相应值，也就求出了$(N)_\beta$。下面在确定 C_{-i}的过程中，为了方便起见，省略进制的下标，并把原数称为 N_0。

用 β 乘式(1.11)两边，得

$$\beta \cdot N_0 = C_{-1} + C_{-2} \cdot \beta^{-1} + \cdots + C_{-m} \cdot \beta^{-m+1}$$

可知上式的整数部分即是 C_{-1}，再令小数部分为 N_1：

$$N_1 = C_{-2} \cdot \beta^{-1} + C_{-3} \cdot \beta^{-2} + \cdots + C_{-m} \cdot \beta^{-m+1} \tag{1.12}$$

再用 β 乘式(1.12)两边，得

$$\beta \cdot N_1 = C_{-2} + C_{-3} \cdot \beta^{-1} + \cdots + C_{-m} \cdot \beta^{-m+2}$$

上式右边的整数部分即是 C_{-2}。

重复上述乘法运算，即可依次求得 C_{-i}，直至 C_{-m}，再将所有的 C_{-i} 转换成 β 进制的相应值，并用位置记数法列出，即是所求的 $(N)_\beta$。

上述计算过程中，如还未求到 C_{-m}，小数部分已为零。表示此数已精确转换，余下位数均为零；如求到 C_{-m} 时，小数部分仍不为零，这表示 $(N)_\beta$ 为 $(N)_\alpha$ 的近似值，即转换有误差。

例 3 将 $(0.432\,1)_{10}$ 转换为十六进制(取小数四位)，即

$N_0 = 0.432\,1$ $\qquad \beta = 16$

$\beta \cdot N_0 = 16 \times 0.432\,1 = 6.913\,6$； $\qquad N_1 = 0.913\,6$， $C_{-1} = 6$

$\beta \cdot N_1 = 16 \times 0.913\,6 = 14.617\,6$； $\qquad N_2 = 0.617\,6$， $C_{-2} = 14$(即 E)

$\beta \cdot N_2 = 16 \times 0.617\,6 = 9.881\,6$； $\qquad N_3 = 0.881\,6$， $C_{-3} = 9$

$\beta \cdot N_3 = 16 \times 0.881\,6 = 14.105\,6$； $\qquad N_4 = 0.105\,6$， $C_{-4} = 14$(即 E)

即 $\qquad (0.432\,1)_{10} \approx (0.6E9E)_{16}$

例 4 将 $(0.375)_{10}$ 转换成二进制数。

计算过程可用下列算式表示：

$$\begin{array}{r l}
0\ .375 & \\
\times\quad 2 & \\
\hline
[0].750 & \longrightarrow C_{-1} = 0 \\
\times\quad 2 & \\
\hline
[1].500 & \longrightarrow C_{-2} = 1 \\
\times\quad 2 & \\
\hline
[1].000 & \longrightarrow C_{-3} = 1
\end{array}$$

即 $\qquad (0.375)_{10} = (0.011)_2$

1.2.3 任意两种进制之间的转换

通过上面两节的介绍可以知道：将 $(N)_\alpha$ 转换成 $(N)_\beta$ 时，如果熟悉 α 进制的运算规则就采用基数乘除法；如果熟悉 β 进制的运算规则就采用多项式替代法。

如果对 α 进制和 β 进制的运算规则均不熟悉，则可利用十进制作为桥梁。

一般说来，若需将 α 进制的数转换成 β 进制的数，可先用多项式替代法将 $(N)_\alpha$ 转换成 $(N)_{10}$，再用基数乘除法将 $(N)_{10}$ 转换成 $(N)_\beta$。这样，所有的计算均在十进制中进行。

例 把 $(1023.231)_4$ 转换成五进制数。

第一步：采用多项式替代法将该数转换成十进制数。

$$\begin{aligned}N &= 1\times4^3+0\times4^2+2\times4^1+3\times4^0+2\times4^{-1}+3\times4^{-2}+1\times4^{-3}\\ &= 64+0+8+3+0.5+0.187\,5+0.015\,625\\ &= 75.703\,125\end{aligned}$$

即 $(1023.231)_4=(75.703\,125)_{10}$

第二步:采用基数乘除法将该数从十进制转换成五进制数。

整数部分

$$\begin{array}{l}5\,\underline{|\,75}\cdots\cdots0\\ 5\,\underline{|\,15}\cdots\cdots0\\ \;5\,\underline{|\,3}\cdots\cdots3\\ \quad\;0\end{array}$$

小数部分

$$\begin{array}{r}0.703\,125\\ \underline{\times\qquad\quad 5}\\ [3].515\,625\\ \underline{\times\qquad\quad 5}\\ [2].578\,125\\ \underline{\times\qquad\quad 5}\\ [2].890\,625\\ \underline{\times\qquad\quad 5}\\ [4].453\,125\\ \vdots\quad\end{array}$$

即 $(75.703\,125)_{10}\approx(300.3224)_5$ (取小数四位)

$(1023.231)_4\approx(300.3224)_5$ (取小数四位)

1.2.4 直接转换法

要将 α 进制的数转换成 β 进制的数时,如果基数 α、β 都是 2^k(k 为正整数)时,可以进行直接转换。

二进制在电子计算机和数字逻辑系统中获得广泛应用,但它写起来很长,不易读不易记。基数为 2^k 的进位制实质上是将若干位二进制的字符串用一个数字字符来表示,这样读、写、记录都很方便。如八进制的一个数字字符可以表示三位二进制字符串,而十六进制的一个数字字符可以表示四位二进制字符串。因而,基数为 2^k 的进位制与二进制之间的数制转换,可以用划分相应字符串的方法直接转换。

如八进制数转换为二进制数时,将每位八进制数字符号展开为相应的三位二进制字符串,舍去多余的 0(整数部分最高位的 0 和小数部分最低位的 0)即可。

二进制数转换为八进制数时,整数部分应从小数点开始向左数,每三位对应八进制一位,最高有效位不足三位时,可在高位加 0 补足三位。小数部分则应从小数点向右数,每三位对应八进制一位,最低有效位不足三位时,应在低位加 0 补足三位。

例 1 将 $(372.34)_8$ 转换成二进制数。

$$\begin{array}{lccccccc}\text{八进制数:} & \underline{\;3\;} & \underline{\;7\;} & \underline{\;2\;} & . & \underline{\;3\;} & \underline{\;4\;}\\ & 011 & 111 & 010 & . & 011 & 100\end{array}$$

二进制数:11111010.0111

即 $(372.34)_8=(11111010.0111)_2$ 或写成 $(374.34)_O=(11111010.0111)_B$

式中下标 O 表示 Octal——八进制，下标 B 表示 Binary——二进制。

例 2 将$(10101100.11)_2$转换成八进制数。

二进制数：$\underline{0\,1\,0}\ \underline{1\,0\,1}\ \underline{1\,0\,0}\ .\ \underline{1\,1\,0}$

八进制数：　2　5　4　.　6

即　$(10101100.11)_B=(254.6)_O$

十六进制与二进制之间的转换方法相同，只是每一位十六进制数对应四位二进制数而已。十六进制与八进制之间的转换，可以通过二进制作为过渡，重新分组后直接进行。

例 3 将$(AF.16C)_{16}$转换为八进制。

十六进制：$\underline{\ A\ }\ \ \underline{\ F\ }\ .\ \underline{\ 1\ }\ \ \underline{\ 6\ }\ \ \underline{\ C\ }$

二进制：$\underline{0\,1\,0}\ \underline{1\,0\,1}\ \underline{1\,1\,1}\ .\ \underline{0\,0\,0}\ \underline{1\,0\,1}\ \underline{1\,0\,1}\ \underline{1\,0\,0}$

八进制：　2　5　7　.　0　5　5　4

即　$(AF.16C)_{16}=(257.0554)_8$　或写成　$(AF.16C)_H=(257.0554)_O$

式中下标 H 表示 Hexadecimal——十六进制。

1.2.5 数制转换时小数位数的确定

设 α 进制小数有 k 位，转换成 β 进制后维持相同的精度需要 j 位，这时应有：

$$(0.1)_\alpha^k=(0.1)_\beta^j$$

在十进制中可表示为：

$$\left(\frac{1}{\alpha}\right)^k=\left(\frac{1}{\beta}\right)^j$$

即　$\alpha^k=\beta^j$

两边取对数

$$\lg\alpha^k=\lg\beta^j$$

即

$$k\cdot\lg\alpha=j\cdot\lg\beta$$

所以

$$j=k\cdot\frac{\lg\alpha}{\lg\beta}$$

由于要求转换后的精度不低于原精度，j 应取符合下列不等式的整数：

$$k\cdot\frac{\lg\alpha}{\lg\beta}\leqslant j<k\cdot\frac{\lg\alpha}{\lg\beta}+1 \tag{1.13}$$

例如，将$(0.432\,1)_{10}$转换为十六进制时，小数位数应取多少？

由于原数精度为$(0.1)_{10}^4$，j 应满足下列不等式：

$$4\cdot\frac{\lg 10}{\lg 16}\leqslant j<4\cdot\frac{\lg 10}{\lg 16}+1$$

即　$3.320\leqslant j<4.320$

所以，将$(0.432\,1)_{10}$转换为十六进制时，小数位数应取 4 位。