电子技术类职业教育丛书

5

数字电路基础

王明臣 编

科学技术文献出版社

中国电子学会普及工作部
北京市职 业 教 育 研究会 组编

电子技术类职业教育丛书之五

数字电路基础

王明臣 编

科学技术文献出版社
1986

电子技术类职业教育丛书之五
数字电路基础
王明臣 编
科学技术文献出版社出版
一二〇一厂印刷
新华书店北京发行所发行　各地新华书店经售
*
787×1092毫米　16开本　印张：23.5　字数：598.4千字
1986年11月北京第一版第一次印刷
印数：1—9,000册
科技新书目：126—46
统一书号：15176·703　定价：4.85元

前　　言

为适应我国职业技术教育迅速发展的迫切需要，使教育更好地为四化建设服务，中国电子学会普及工作部和北京市职业教育研究会共同组织编写出版这套《电子技术类职业教育丛书》。

丛书包括：无线电数学、无线电电工基础、模拟低频电子电路、模拟高频电子电路、数字电路基础、盒式收录机原理与电路解说、黑白彩色电视机原理与电路解说（上、下册）、微型计算机原理和应用、家用录象机原理与电路解说，共十册。

这套丛书是参照电子技术类职业教育的教学计划和大纲编写的。它包括了电子技术专业的基础课、技术基础课和专业课，具有较强的系统性，每册内容又具有一定的独立性。丛书可作为职业教育参考教材，也可供具有中等文化程度的电子技术爱好者自学时选用。

在编写丛书过程中，编者注意到理论与实践密切结合，用具体应用实例来加深对理论概念的理解；以阐明分析问题的步骤和思路为线索突出物理概念，并有一定的理论分析以加深理解；在文字上力求深入浅出和通俗易懂。每章后面一般都有一定数量的习题，帮助读者巩固所学的内容。书后还附有习题解答或提示，以便于自我检查。

本套丛书部分内容曾作为中国电子学会举办的“全国电子技术自修班”教材使用过，充分听取了广大学员对本书的意见。对书中的遗误和不妥之处进行了必要的修改；对部分内容也作了适当的调整和增删。

中国电子学会普及工作部和北京市职业教育研究会的有关领导，对丛书的出版给予了大力支持，并直接组织指导了全套丛书的选题、编写、定稿和印刷出版等事宜；有关工作人员和编者们也为全套丛书尽早与读者见面做出了很大的努力。尽管如此，在较短的时间里，组织出版这样一套职业教育系列丛书，我们还是第一次尝试。书中的错误与不当之处在所难免，尤其是这套丛书是否能满足职工教育的要求，更有待于广大读者通过学习实践提出宝贵意见，以便于在此基础上编出更适合我国职业技术教育的丛书。

最后，我们还应向为这套丛书及时出版而付出辛勤劳动的出版、印刷等部门，以及所有参与此项工作的同志表示衷心的感谢。

丛书编辑委员会
一九八六年四月　于北京

编 者 的 话

本书是电子技术类职业教育丛书之五，主要讲述数字电路基础知识。全书包括：数制和码制、逻辑代数、基本逻辑电路、组合逻辑电路、时序逻辑电路、运算电路、存储器、模拟与数字转换电路等内容。本书既可作为电子技术类职业教育教材，也可作为具有中学以上文化程度并具有一定无线电技术基础知识的广大无线电爱好者参考。

此书曾作为中国电子学会普及工作部举办的"全国电子技术自修班"选用教材，原名为：《微型电子计算机数字电路基础》。这次作为职业教育丛书正式出版，在自修班两次试用的基础上，广泛听取了读者的意见，对全书的遗误之处作了修正，并考虑到数字电视、数字录像等新技术发展的需要，增编了模拟与数字转换电路一章，书名改为《数字电路基础》。

为了适应职业教育的特点，我们力求讲清基本物理概念，做到深入浅出和循序渐进，较系统地阐明问题的思考方法与分析步骤，以便于初学者掌握。各章后边附有习题，读者在学完每章的内容之后，应能独立完成所列的习题。对一些较难的习题，在书后附有答案或提示供读者参考。

在编写本书的过程中，参考了微型电子计算机和数字电路等有关方面的书籍和讲义。限于作者的水平，加之时间急迫，脱稿仓促，书中难免存在缺点和不当之处，诚恳地欢迎广大读者批评指正。

编 者

一九八六年四月 北京

目　　录

第一章　数制和码制……1
第一节　数制的概念……1
一　数的表示方法……1
二　各种进位制的相互转换关系……5
第二节　二进制数的正数、负数、整数及小数表示方法……11
一　正负符号的表示方法……11
二　小数的表示方法……11
第三节　二进制的四则运算……13
一　二进制的加法……14
二　二进制的减法……14
三　二进制的乘法……14
四　二进制的除法……15
第四节　原码、补码和反码……16
一　原　码……16
二　补　码……18
三　反　码……21
四　原码、补码与反码间的相互变换……23
五　补码加法的实现……24
第五节　二进制的编码……32
一　二进制编码的十进制数……32
二　字母和符号的编码……35
第二章　逻辑代数……39
第一节　逻辑代数的概念与表示形式……39
一　什么叫逻辑代数……39
二　逻辑代数的基本运算……40
三　逻辑函数及其表示方法……44
第二节　逻辑代数基本运算的定律、定理和重要规则……50
一　逻辑代数中的三大基本定律……50
二　逻辑代数基本定理……55
三　逻辑代数运算中的基本法则……62
第三节　逻辑代数的标准式及简化……65
一　逻辑代数的标准式……65
二　利用公式化简逻辑式……69
三　利用卡诺图化简逻辑表达式……71

第三章　基本逻辑电路……84
第一节　有触点开关逻辑电路……84
一　开关逻辑电路……84
二　继电器逻辑电路……86
三　有触点开关电路的分析与综合……92
第二节　逻辑门电路……94
一　基本逻辑门电路……95
二　复合逻辑门电路……99
三　其他逻辑门电路……104
第三节　逻辑门电路的分析与综合……119
一　逻辑门电路的分析……120
二　逻辑门电路的设计步骤与应用举例……122
第四节　正逻辑与负逻辑……126
一　逻辑的对偶性……126
二　莫根律与正、负逻辑的关系……128
三　反相符号……130
第四章　组合逻辑电路……135
第一节　编码器……135
一　二进制编码器……135
二　二-十进制编码器……137
三　代码转换电路……139
四　优先编码器……140
第二节　译码器……143
一　二进制译码器……143
二　二-十进制译码器及显示器……146
三　中规模集成二-十进制七段显示译码器……155
第三节　多路选择器……157
一　多路选择器工作原理……158
二　利用多路选择器实现数据传送……159
三　利用多路选择器实现组合逻辑……160
第四节　数码比较器与奇偶校验器……161
一　数码比较器……161
二　奇偶校验器……165
第五节　组合逻辑电路中的竞争与冒险……169
一　什么是竞争与冒险……169
二　冒险竞争的判断……171
三　冒险竞争的消除……173
第五章　时序逻辑电路……176
第一节　触发器……176

一　分立元件组成的触发器 ……176
二　集成触发器 ……181
三　各类触发器小结 ……194
第二节　寄存器 ……199
一　代码寄存器 ……199
二　移位寄存器 ……201
第三节　计数器 ……204
一　异步计数器 ……204
二　同步计数器 ……208
三　减法计数器与可逆计数器 ……215
第六章　运算电路 ……225
第一节　加法电路 ……225
一　二进制加法器 ……225
二　二-十进制加法器 ……232
第二节　减法电路 ……239
一　二进制减法器 ……239
二　二-十进制减法器 ……247
第三节　乘法电路 ……252
一　二进制乘法器 ……253
二　二-十进制乘法器 ……259
第四节　除法电路 ……263
一　二进制除法器 ……263
二　二-十进制除法器 ……268
第五节　运算电路的其他有关部件 ……271
一　累加器 ……271
二　实际加法器的组成 ……275
三　浮点运算电路 ……279
第七章　存储器 ……285
第一节　存储器的分类 ……285
一　按存取方式分类 ……285
二　按所用元件材料分类 ……286
第二节　磁芯存储器 ……286
一　磁芯存储器写入与读出工作原理 ……286
二　2D方式（字排列方式） ……287
三　3D方式（位排列方式） ……289
第三节　半导体存储器 ……289
一　MOS基本电路及其工作原理 ……290
二　随机存取存储器(RAM) ……295
三　只读存储器(ROM) ……306

第四节 电荷耦合器件(CCD)存储器 ……………………………………314
一、CCD的基本工作原理 ……………………………………314
二、数据电荷的“写入”和“读出” ……………………………………316
三、CCD存储器应用实例 ……………………………………317
第八章 模数转换与数模转换 ……………………………………319
第一节 模拟量和数字量 ……………………………………319
第二节 模数 变换的基本原理……………………………………320
一、取 样 ……………………………………320
二、量 化 ……………………………………320
三、编 码 ……………………………………321
第三节 模拟-数字转换器……………………………………321
一、串联型模数转换器 ……………………………………321
二、并联型模数转换器 ……………………………………322
三、串并型模数转换器 ……………………………………323
四、逐位逼近型模数转换器 ……………………………………325
第四节 数字-模拟转换器……………………………………326
一、数字-模拟转换基本原理……………………………………326
二、梯形权电阻数模转换器 ……………………………………327
三、R-2R梯形数模转换器……………………………………328
四、高速梯形数模转换器 ……………………………………328
习题解答 ……………………………………330

第一章　数制和码制

目　的　要　求

电子计算机是以二进制的数制进行计算的；此外，也用到八进制与十六进制的数制。对于初学者，首先应了解二进制的表示方法、特点以及它与十进制之间的关系和相互变换等。然后应进一步了解二进制的小数、正数和负数在电子计算机中的表示方法。通过本章的学习，要求读者能达到：

(1) 深入理解数制的概念及一般数的表示方法。

(2) 掌握二进制、十进制及其他进制数的表示方法，它们之间的相互关系与变换规则。

(3) 会对二进制数进行加、减、乘、除等四则运算。

(4) 掌握二进制中小数点的表示方法及正、负符号的表示方法。

(5) 深入理解原码、反码和补码的概念与它们之间的相互关系，学会利用补码把减法运算转换为加法运算。

(6) 能用不同形式的二进制编码来表示十进制的数、字母和各种符号。

第一节　数制的概念

数制是数的进位方式和制度。在不同的数制中，数的进位与计算方法各不一样。在日常生活中，我们用到各种进位制，如六十秒为一分钟，六十分钟为一小时，这就是六十进位制；旧秤的十六两为一斤，这就是十六进位制；二十四小时为一天，这又是二十四进位制……。不过，人们最常用的也最熟悉的还是十进位制，也叫十进制。

一、数的表示方法

如何表示一个数，在不同的数制中有不同的形式，但是它们又都遵循一个共同的规律，这就是都用位数和每位上的状态数(0，1，2，……)相结合来表示。

1．十进制

我们先来分析一下如何用十进制表示一个数。在十进制中，每一位最多有0～9十个数码(或状态)，因而状态数为十。任何一位上满十即向高位进一，所以任何一位上不可能出现十的数码。每相邻两位之间的关系为：向高位进位规则是“逢十进一”，给低位借位规则是“借一当十”。同一个数码，如果在数中所处的位置不同，它所代表的数值也不同。

例如：我们观察十进制中256.37这个数。数码2处在百位，它所代表的数值为二百；数码5在十位，它代表的数值为五十；数码6在个位，代表的数值为六；数码3在十分之一位，代表的数值为十分之三；数码7在百分之一位，代表的数值为百分之七。可见，数码0～9可以出现在不同的位置上，不同位置上的数码代表的数值就不同。为了确定某数位上的数码所代表的实际数值，必须乘上一个因子，例如：一百、十、一、十分之一或百分之一等，这个因子就称为权。或者说，权是表示某一数位上单位有效数字所代表的实际数值。

把上述十进制的数256.37写成按权的展开式，即为

$$256.37=2\times\underbrace{10^2+5\times 10^1+6\times 10^0+3\times 10^{-1}+7\times 10^{-2}}_{\text{十进制的权}}$$

式中，从左到右的10的幂是10^2、10^1、10^0、10^{-1}、10^{-2}，均为十进制的权。以十进制小数点为参考点，2、5、6数码分别为百位、十位和个位的状态数；3、7等数码又分别为十分之一位、百分之一位的状态数。其整数部分，离小数点越近，权越小；其小数部分则相反，离小数点越近，权越大。状态数改变位置，则它相应代表的数值也就改变了。

综上所述，对于任何一个十进制的数$(N)_{10}$，如果它有n位整数(小数点以前的位数)和m位小数(小数点以后的位数)，则可用下面的一般式来表示：

$$\begin{aligned}(N)_{10}&=A_{n-1}\cdot 10^{n-1}+A_{n-2}\cdot 10^{n-2}+\cdots\cdots+A_1\cdot 10^1+A_0\cdot 10^0+A_{-1}\cdot 10^{-1}+\cdots\cdots\\&\quad+A_{-m}\cdot 10^{-m}\\&=\sum_{i=n-1}^{-m}A_i\cdot 10^i\end{aligned}$$

上式中，N的下标10表示N为十进制的数，A表示数码0～9中的任何一个数，A_i表示第i位的数码，10^i表示十进制第i位的权，n表示整数的位数，为正整数，m表示小数的位数，为正整数。

2. 二进制

二进制的数码只有两个，0和1，即状态数为二。任何一位满二则向高位进1，故任何一位都不会出现二的数码。我们可把二进制的进位规则总结为“逢二进一”，借位规则为“借一当二”。

按照上述十进制数的一般表示法，把10改为2，就很容易写出二进制的一般表示式。例如，对于有n位整数和m位小数的二进制，可表示为：

$$\begin{aligned}(N)_2&=B_{n-1}\cdot 2^{n-1}+B_{n-2}\cdot 2^{n-2}+\cdots\cdots+B_1\cdot 2^1+B_0\cdot 2^0+B_{-1}\cdot 2^{-1}+B_{-2}\cdot 2^{-2}\\&\quad+\cdots\cdots+B_{-m}\cdot 2^{-m}\\&=\sum_{i=n-1}^{-m}B_i\cdot 2^i\end{aligned}$$

上式中，$(N)_2$表示二进制的数，B表示0与1两数码中的任何一个数，B_i表示第i位的数码，2^i表示二进制第i位的权，n表示整数的位数，为正整数，m表示小数的位数，为正整数。

例如：一个二进制的数$(1011.011)_2$可以表示为按权的展开式：

$$(1011.011)_2=1\times\underbrace{2^3+0\times 2^2+1\times 2^1+1\times 2^0+0\times 2^{-1}+1\times 2^{-2}+1\times 2^{-3}}_{\text{二进制的权}}$$

同十进制一样，二进制各位的权代表该位上单位数码所表示的实际数值。例如上面二进制数中，第一个(首位)数码1代表$1\times 2^3=8$，而最后一个(末位)数码1则代表$1\times 2^{-3}=\frac{1}{8}$。

从上述按权的展开式可以看出，二进制的权为2的幂，例如2^3、2^2、2^1、2^0、2^{-1}、2^{-2}……等。它们的排列规律与十进制相同，以小数点为参考点，其整数位离小数点越近权越小，其小数位离小数点越近则权越大。

在计算机中，二进制数所包含的位数通常称为比特(bit)，或者说一比特表示一位二进制的数。

十进制数与二进制数之间的对应关系，我们可以举如下的一些数来表示，只要我们按照二进制中“逢二进一”的进位规则，两者之间的关系是很容易理解的。

十进制	0	1	2	3	4	5	6	7	8	9	10	11
二进制	0	1	10	11	100	101	110	111	1000	1001	1010	1011

我们研究二进制，是由于在电子计算机和其他数字设备中广泛地采用二进制的形式，这又因为二进制有以下一些优点。

(1) 二进制数容易表示

我们知道，在电子计算机和数字设备中，任何数字都是由电路元件的状态来记忆和表示的。由于二进制每一位只有两个数码 1 和 0，所以凡是有两种不同工作状态的电子元件(双稳元件)，都可以用来表示一位二进制的数。例如，可以用一个指示灯来表示一位二进制数，灯亮表示“1”，灯灭表示“0”；一只开关也可以用来表示一位二进制数，开关接通表示“1”，断开表示“0”。实际上，有两个稳定状态的电子元件是很多的，例如继电器触点的闭合与断开，二极管、晶体管的导通与截止，电路中电压的高和低，脉冲的有和无等等。只要规定其中一种状态为“1”，另一种相反的状态为“0”，就可以用来表示一位二进制数。至于更大的数值，则可以用更多的位数来表示。

(2) 采用二进制可以节省电路元件，便于设计计算机和简化机器结构

任何一个数，是由每位的状态数及位数共同表示的。我们仅用十进制与二进制进行比较。

例如我们要表示 0 到15这个数值范围，若用十进制来表示，它需要二位十进制数，其个位可能出现 0 到 9 共10个状态，其十位可能出现 0 或 1 两个状态，共计12个状态。如果我们用二进制来表示 0 到15这个数值范围，共需要四位二进制数，或称 4 比特；每一位二进制数可能出现两个状态(0 或 1)，故四位共计八个状态。由于这些状态是由电子元件的工作状态来表示的，所以状态越少，相应的电路元件数目也越少，电路设计也越简单。

(3) 识别能力强，抗干扰性能好

由于电子计算机中的数由电子元件的工作状态来表示，所以这些电子元件的工作如果不稳定，必然会影响到机器的正常工作或造成动作错误。若我们选用二进制，每一位二进制相应的两个数码(0 或 1)就可由双稳元件的两个状态来表示。例如晶体管导通表示“1”，截止表示“0”，只要我们保证“1”状态时晶体管饱和导通，而“0”状态时晶体管充分截止，就可以防止由于电压波动或外来干扰而引起的动作错误，保证了机器工作的可靠性。

当然，二进制也有不足之处。既然在各种数制中任何一个数值都是由每位的状态数及位数共同表示的，所以每一位上的状态数(基数)越多，则一个数的位数就越少；反之，状态数越少，则位数必然越多。这也正是二进制的不便之处。二进制的主要缺点就是对同一个数用二进制表示时，其位数比用十进制表示时要多。例如65这个数，在十进制中只有两位就可以表示，但是在二进制中，它为1000001，需要七位来表示。数值越大，这种缺点越突出。所以，初学者往往很不习惯。

3. **其他进制**

在电子计算机中，为了避免二进制的位数太多的缺点，在某些场合还用到八进制与十六

进制。

（1） 八进制

八进制是基数为 8 的进位制，八进制的数码是0、1、2、3、4、5、6、7，任何一位达到 8 时则向高位进一，其进位规则是“逢八进一”，借位规则是“借一当八”。

如果一个八进制数其整数位数为n，而小数位数为m，仿照前述方法，可以写出八进制数按权的展开式如下：

$$
\begin{aligned}
(N)_8 &= C_{n-1}\cdot 8^{n-1}+C_{n-2}\cdot 8^{n-2}+\cdots\cdots+C_1\cdot 8^1+C_0\cdot 8^0+C_{-1}\cdot 8^{-1}+\cdots\cdots \\
&\quad +C_{-m}\cdot 8^{-m} \\
&= \sum_{i=n-1}^{-m} C_i\cdot 8^i
\end{aligned}
$$

式中，C为数码0～7中的任何一个数，C_i为第i位的数码，8^i为八进制第i位的权。

例如一个八进制数$(127.321)_8$，写成按权的展开式时为

$$(127.321)_8=1\times 8^2+2\times 8^1+7\times 8^0+3\times 8^{-1}+2\times 8^{-2}+1\times 8^{-3}$$

$$\underbrace{8^2\qquad 8^1\qquad 8^0\qquad 8^{-1}\qquad 8^{-2}\qquad 8^{-3}}_{\text{八进制的权}}$$

以十进制中0～11的数为例，它与二进制、八进制的对应关系如下表所示。

十进制	0	1	2	3	4	5	6	7	8	9	10	11
二进制	0	1	10	11	100	101	110	111	1000	1001	1010	1011
八进制	0	1	2	3	4	5	6	7	10	11	12	13

（2） 十六进制

十六进制是基数为16的进位制，任何一位上满16则向高位进一，所以任何一位可能出现的最大数码为15。为了表示方便，我们把每位上可能出现的10、11、12、13、14、15六个两位十进制的数分别用字母 A、B、C、D、E、F 来表示。这样，十六进制中的数码是0、1、2、3、4、5、7、8、9、A、B、C、D、E、F，共 16 个。十六进制的进位规则是“逢十六进一”，借位规则是“借一当十六”。

如果一个十六进制的数$(N)_{16}$，其整数位数为 n，小数位数为 m，则同样可以按权写成展开式如下：

$$
\begin{aligned}
(N)_{16} &= D_{n-1}\cdot 16^{n-1}+D_{n-2}\cdot 16^{n-2}+\cdots\cdots+D_1\cdot 16^1+D_0\cdot 16^0+D_{-1}\cdot 16^{-1}+\cdots\cdots \\
&\quad +D_{-m}\cdot 16^{-m} \\
&= \sum_{i=n-1}^{-m} D_i\cdot 16^i
\end{aligned}
$$

式中，D_i为第i位的数码，D为数码0～9和A～F中的任何一个，16^i为十六进制第i位的权。

例如一个十六进制的数$(8AD)_{16}$，可按权写成展开式如下：

$$(8AD)_{16}=8\times 16^2+A\times 16^1+D\times 16^0$$

$$=8\times16^2+10\times16^1+13\times16^0$$

$$\underbrace{16^2 \qquad 16^1 \qquad 16^0}_{\text{十六进制的权}}$$

以十进制中0～17的数为例，它与二进制、十六进制的对应关系如下表所示。

十进制	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
二进制	0	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110	1111	10000	10001
十六进制	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11

通过上面对十进制、二进制、八进制、十六进制等计数方法的分析，我们可以总结出如下两个结论，并可推广到任何进位制中数的一般表示公式。

（一）进位记数制的三要素

数码——组成一种进位记数制的基本成分，例如对于R进位制，数码为0～(R－1)，包括0在内共有R个数码。

基数——进位计数制中数码的个数（状态数），对于R进位制，基数为R。

进位和借位规则——对于任意的R进位制，进位和借位规则为“逢R进一，借一当R”。

（二）任何进位制数的按权展开式

对于任何进位制的数，如R进位制的数$(N)_R$，可以写成按权的一般展开式如下：

$$(N)_R=K_{n-1}\cdot R^{n-1}+K_{n-2}\cdot R^{n-2}+\cdots\cdots+K_1\cdot R^1+K_0\cdot R^0+K_{-1}\cdot R^{-1}+\cdots\cdots$$
$$+K_{-m}\cdot R^{-m}$$
$$=\sum_{i=n-1}^{-m}K_i\cdot R^i$$

上式中，K为数码0～(R－1)中的任一个数，K_i为第i位的数码，R为基数，R^i为R进制第i位的权。n为整数的位数，为正整数；m为小数的位数，也是正整数。

二、各种进位制的相互转换关系

上节我们分析了各种进位制中数的表示方法，对于同一个数可以用不同的进位制来表示。有时候，我们需要把一种进位制的数变换成另一种进位制的数。这一节主要介绍不同进位制数之间的转换规律。

1. 二进制与十进制数之间的转换

（一）二进制数转换成十进制数

二进制数转换成十进制数的方法可以是“按权相加”，即把给定的二进制数按权展开，然后相加，就可得到相应的十进制数。

例1：$(1010101)_2=(?)_{10}$

解：$(1010101)_2=1\times2^6+1\times2^4+1\times2^2+1\times2^0$

$=64+16+4+1$

$=(85)_{10}$

例2：$(0.1011)_2=(?)_{10}$

解：$(0.1011)_2=1\times2^{-1}+1\times2^{-3}+1\times2^{-4}$

$=0.5+0.125+0.0625$

$=(0.6875)_{10}$

若将上述两例合起来，则有

$$(1010101.1011)_2=(85.6875)_{10}$$

(二)十进制数转换成二进制数

十进制数转换成二进制数时，要将十进制数整数和小数分别转换，然后再合起来。

(1) 整数部分

十进制整数部分的转换可用“除 2 取余，逆序排列”法，即用十进制整数除以 2，得到一个商和余数，再将新的商除以 2……，一直除下去，直到商等于 0 时为止。然后将每次所得余数，按次序排列起来，先得到的余数为二进制整数的低位，后得到的余数为二进制整数的高位，这就是所谓“逆序排列”。

例 1：$(84)_{10}=(?\ )_2$

解：

$$\begin{array}{r|l}
2 & 84 \\
\hline
2 & 42 \cdots\cdots\text{余}0 \uparrow \\
\hline
2 & 21 \cdots\cdots\text{余}0 \\
\hline
2 & 10 \cdots\cdots\text{余}1 \\
\hline
2 & 5 \cdots\cdots\text{余}0 \\
\hline
2 & 2 \cdots\cdots\text{余}1 \\
\hline
2 & 1 \cdots\cdots\text{余}0 \\
\hline
 & 0 \cdots\cdots\text{余}1
\end{array}$$

$$\therefore (84)_{10}=(1010100)_2$$

(2) 小数部分

十进制小数部分的转换可用“乘 2 取整，顺序排列”法，即用十进制小数部分乘以 2，得到一个乘积。将乘积的整数部分取出，小数部分继续乘以 2，又得到一个新的乘积，再将新的乘积中的整数部分取出，小数部分继续乘以 2……一直乘下去，直到乘积的小数部分为 0，或者达到所要求的精度为止。然后将每次取出的整数按次序排列起来，先得到的整数为二进制小数的高位，后得到的整数为二进制小数的低位，这就是所谓“顺序排列”。

例 2：$(0.6875)_{10}=(?\ )_2$

解：

$$\begin{array}{lr}
 & 0.6875 \\
 & \times\quad 2 \\
\hline
\text{整 } 1 \cdots\cdots & 1.3750 \\
 & 0.3750 \\
 & \times\quad 2 \\
\hline
\text{整 } 0 \cdots\cdots & 0.7500 \\
 & 0.7500 \\
 & \times\quad 2 \\
\hline
\text{整 } 1 \cdots\cdots & 1.5000 \\
 & 0.5000 \\
 & \times\quad 2 \\
\hline
\downarrow \text{整 } 1 \cdots\cdots & 1.0000
\end{array}$$

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

若将上述两例合起来，则有

$$(84.6875)_{10}=(1010100.1011)_2$$

2. 任意进制数与十进制数之间的转换

任意进制数转换成十进制数的方法，类似于二进制数转换成十进制数的方法，即“按权相加”。

十进制数转换成任意进制数的方法，类似于十进制数转换成二进制数的方法，即用整数部分和小数部分分别转换。下面，举出几例予以说明。

例 1：$(125.54)_8=(?)_{10}$

解：$(125.54)_8=1\times8^2+2\times8^1+5\times8^0+5\times8^{-1}+4\times8^{-2}$

$$=64+16+5+0.625+0.0625$$

$$=(85.6875)_{10}$$

例 2：$(85.6875)_{10}=(?)_8$

解：整数部分

$$\begin{array}{r|l}
8 & 85 \\ \hline
8 & 10 \cdots\cdots\cdots\cdots 余5 \uparrow \\ \hline
8 & 1 \cdots\cdots\cdots\cdots 余2 \\ \hline
 & 0 \cdots\cdots\cdots\cdots 余1
\end{array}$$

$$\therefore (85)_{10}=(125)_8$$

小数部分

$$\begin{array}{rr}
 & 0.6875 \\
 & \times\quad 8 \\ \hline
\downarrow 整5 & \cdots\cdots 5.5000 \\
 & 0.5000 \\
 & \times\quad 8 \\ \hline
整4 & \cdots\cdots 4.0000
\end{array}$$

$$\therefore (0.6875)_{10}=(0.54)_8$$

把整数部分和小数部分合起来，则有

$$(85.6875)_{10}=(125.54)_8$$

例 3：$(55.B)_{16}=(?)_{10}$

解：$(55.B)_{16}=5\times16^1+5\times16^0+B\times16^{-1}$

$$=5\times16^1+5\times16^0+11\times16^{-1}$$

$$=80+5+0.6875$$

$$=(85.6875)_{10}$$

例 4：$(85.6875)_{10}=(?)_{16}$

解：整数部分

$$\begin{array}{r|l}
16 & 85 \\ \hline
16 & 5 \cdots\cdots\cdots\cdots 余5 \uparrow \\ \hline
 & 0 \cdots\cdots\cdots\cdots 余5
\end{array}$$

$$\therefore\quad (85)_{10}=(55)_{16}$$

小数部分

$$\begin{array}{rr}
 & 0.6875 \\
 & \times\quad 16 \\ \hline
 & 4\ 1250 \\
 & 6\ 875 \\ \hline
整11 & \cdots\cdots 11.0000
\end{array}$$

$$\because\quad (11)_{10}=(B)_{16}$$

$$\therefore\quad (0.6875)_{10}=(0.B)_{16}$$

把整数部分和小数部分合起来，则有

$$(85.6875)_{10}=(55.B)_{16}$$

3. 二进制数与八进制数之间的转换