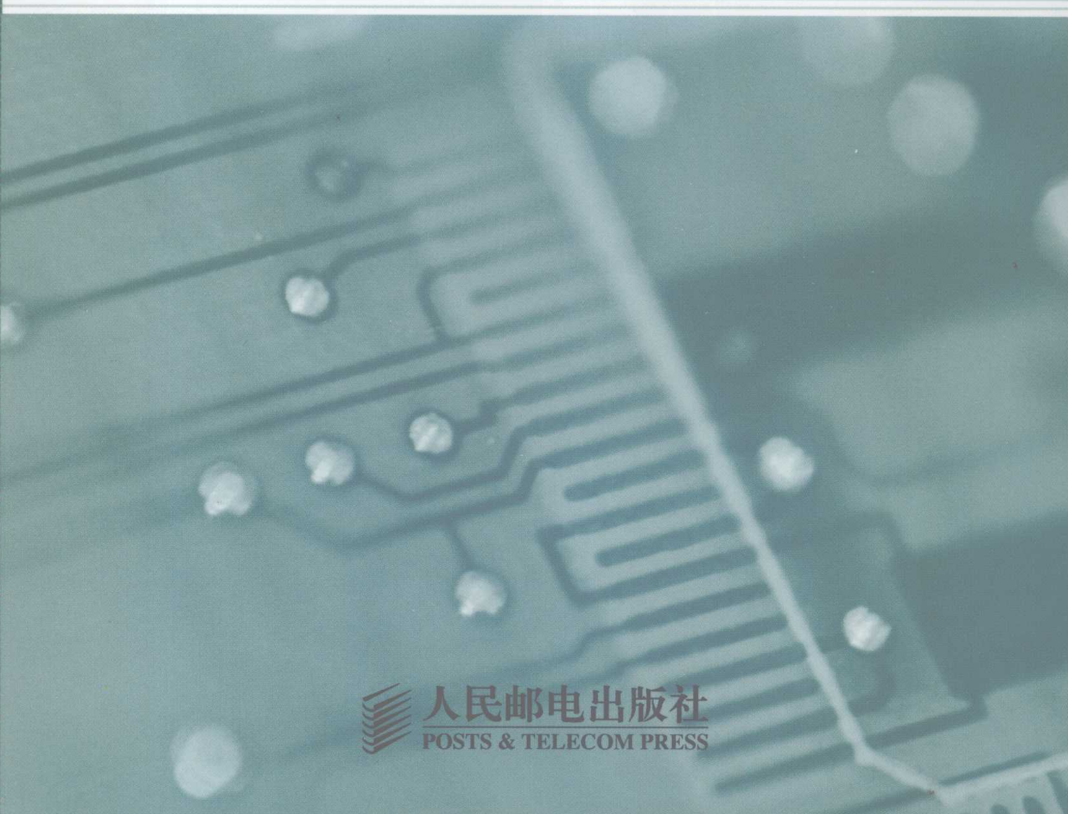
高职高专21世纪规划教材

GAOZHI GAOZHUAN 21 SHIJI GUIHUA JIAOCAI

数字电子技术基础

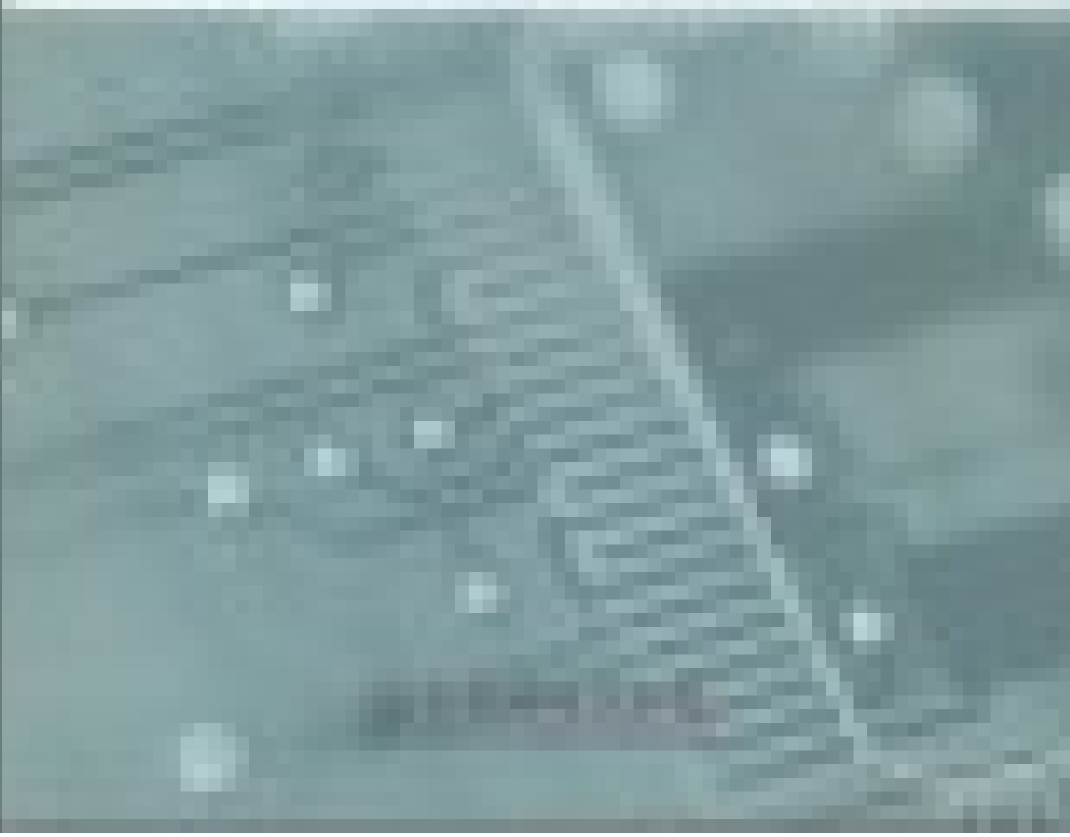
■ 杨碧石 主编　陈兵飞 束慧 副主编 ■



人民邮电出版社

POSTS & TELECOM PRESS

数字电子技术基础



高职高专21世纪规划教材

数字电子技术基础

杨碧石　主　编
陈兵飞　束　慧　副主编

人民邮电出版社
北京

图书在版编目（CIP）数据

数字电子技术基础 / 杨碧石主编；陈兵飞，束慧副主编．—北京：人民邮电出版社，2007.10

（高职高专 21 世纪规划教材）

ISBN 978-7-115-16649-4

Ⅰ．数…　Ⅱ．①杨…②陈…③束…　Ⅲ．数字电路—电子技术—高等学校：技术学校—教材　Ⅳ．TN79

中国版本图书馆 CIP 数据核字（2007）第 123744 号

内 容 提 要

本书介绍逻辑代数的基本知识及其数字逻辑电路的基本分析和设计方法。全书共分8章。主要内容包括逻辑代数的基本知识，组合逻辑电路的分析与设计，时序逻辑电路的分析与设计，脉冲波形产生电路，数模和模数转换电路，半导体存储器和可编程逻辑器件等。本书每章后面都配有实验与实训及习题，便于读者巩固所学理论知识，提高分析问题和解决问题的能力。

本书可作为高职高专院校电子、电气、自动化、计算机等有关专业的教材，也可作为自学者及科技人员参考用书。

高职高专 21 世纪规划教材

数字电子技术基础

◆ 主　　编　杨碧石
副 主 编　陈兵飞　束　慧
责任编辑　王　平

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号
邮编　100061　　电子函件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
河北省邮电印刷厂印刷
新华书店总店北京发行所经销

◆ 开本：787×1092　1/16
印张：12.75
字数：301 千字　　　2007 年 10 月第 1 版
印数：1 – 3 000 册　　　2007 年 10 月河北第 1 次印刷

ISBN 978-7-115-16649-4/TN

定价：19.00 元

读者服务热线：(010)67170985　印装质量热线：(010)67129223

编者的话

电子技术是目前发展最快的学科之一。随着科学技术的发展，许多专业相继开设了电子技术课程。电子技术分为“模拟电子技术”和“数字电子技术”，它们均是学习其他有关课程的基础。本书只讨论“数字电子技术”。

编写本书时，我们注重精选教材内容，教材教学内容较好地体现了基础性、先进性和前沿性，突出重点，加强基本概念、基本原理、单元电路、集成电路器件的分析，循序渐进地培养和提高逻辑电路的设计能力，为社会输送具有独立分析问题和解决问题的应用型人才。根据高职教育的要求，本课程的工程性较强，我们着重介绍数字逻辑电路的分析及集成电路的实际应用。

本教材是根据《高职高专教育专业人才培养目标及规格》要求，结合作者多年的教学改革和实践经验，以培养高素质、应用型、具备综合工作能力的人才为出发点编写而成的。

全书共分 8 章，第 1 章主要讲述逻辑代数的基本知识，为学习以后各章创造必要的条件；第 2 章主要讨论门电路的组成原理、逻辑状态的分析，这是本课程的重要基础；第 3 章主要讨论组合逻辑电路的分析和设计；第 4 章主要讨论触发器电路的组成原理、逻辑状态的分析，这是时序逻辑电路的基础；第 5 章主要讨论时序逻辑电路的分析和设计；第 6 章主要介绍脉冲波形产生电路；第 7 章主要讨论数模和模数转换电路；第 8 章主要介绍半导体存储器和可编程逻辑器件。

本教材由杨碧石主编。其中第 1、3、4、5 章由杨碧石编写，第 2、8 章由陈兵飞编写，第 6、7 章由束慧编写，全书由杨碧石统稿。在本书编写与整理过程中，许多专家及同行给予了大力支持和帮助，并提出了一些宝贵意见，在此，向他们表示衷心的感谢。

由于笔者水平所限，书中难免会有错误和不妥之处，敬请广大读者批评指正。可通过 E-mail 发至 ntybs@126.com 和 ntybs@mail.ntvc.edu.cn 与我们联系。

编者

2007 年 8 月

目　录

第 1 章

逻辑代数基础

本章主要介绍数制与码制、逻辑代数的基本定律、逻辑函数的表示方法和逻辑函数的化简方法等。

数制是多位数码中每一位的构成方法以及从低位到高位的进位规则，其中包括十进制、二进制、八进制和十六进制等，应熟练掌握数制间的相互转换；码制是为了便于记忆和处理，在编制代码时要遵循的规则，应掌握常用的 BCD 码。

与、或、非既是 3 种基本逻辑关系，也是 3 种基本逻辑运算，与非、或非、与或非、异或则是由 3 种基本逻辑运算复合而成的 4 种常用逻辑运算。书中还给出了表示这些运算的逻辑符号，要注意理解和记忆。

逻辑代数的基本定律与常用公式是推演、变换和化简逻辑函数的依据，有些与普通代数相同，有些则完全不一样，例如摩根定理、重叠律、非非律等，要特别注意记住这些特殊的公式。

逻辑函数常用的表示方法有 5 种：真值表、卡诺图、函数式、逻辑图和波形图。它们各有其特点，但本质相通，可以互相转换。尤其是由真值表到逻辑图和由逻辑图到真值表的转换，直接涉及数字电路的分析设计与综合问题，更加重要，一定要掌握。

逻辑函数公式化简法和卡诺图化简法，是应该熟练掌握的内容。公式化简法没有什么局限性，但也无一定步骤可以遵循，要想迅速得到函数的最简与或表达式，不仅和对公式、定律的熟悉程度有关，而且还和运算技巧有联系。卡诺图化简法则不同，它简单、直观，有可以遵循的明确步骤，不易出错，初学者也易于掌握。但是，当函数变量多于 5 个时，卡诺图化简法就失去优势，没有实用价值了。

1.1 概　　述

在自然界中有各种物理量，尽管它们的性质各异，但就其变化规律的特点而言，不外乎有两大类。其中一类物理量的变化在时间上和数值上都是连续的，这一类物理量称为模拟量，这种模拟量的信号叫模拟信号，如电视的图像和伴音信号、生产过程中由传感器检测的由某种物理量（如温度、压力）转化成的电信号等，传输、处理模拟信号的电路称为模拟电路。另一类物理量的变化在时间和数值上都是离散的，这一类物理量称为数字量，这种数字量的信号叫数字信号，如电子表的秒信号、生产中自动记录零件个数的计数信号、由计算机键盘输入计算机的信号等，它们的变化发生在一系列离散的瞬间，数值大小的增减总是最小数量

单位的整数倍，传输、处理数字信号的电路称为数字电路。

1.1.1 数字电路的特点和分类

数字电路的工作信号一般都是数字信号。在电路中，它往往表现为突变的电压或电流，并且只有两个可能的状态。所以，数字电路中的半导体器件应工作在开关状态。利用器件导通和截止两种不同的工作状态，代表不同的数字信息，完成信号的传递和处理任务。

通常用 0 和 1 组成的二值量表示数字信号最为简单，故常用的数字信号是用电压的高、低，脉冲的有、无，分别代表两个离散数值 1 和 0。数字电路在结构、工作状态、研究内容和分析方法等方面都具有自己的特点：数字电路中半导体器件工作在开关状态，这和二值量或二进制信号的要求是相对应的，分别用 1 和 0 两个数码来表示；数字电路的基本单元电路比较简单，对元件的精度要求不高，便于电路集成化、系列化生产，并具有使用方便、可靠性高、价格低廉等优点；数字电路能够对数字信号进行各种逻辑运算和算术运算，所以在各种数控装置、智能仪表以及计算机等领域得到广泛应用。

数字电路按组成的结构可分为分立元件电路和集成电路两大类。其中集成电路按集成度分为小规模（SSI 集成度为 1～10 门/片）、中规模（MSI 集成度为 10～100 门/片）、大规模（LSI 集成度为 100～1000 门/片）和超大规模（VLSI 集成度大于 1000 门/片）集成电路；按电路所用器件的不同，可分为双极型和单极型电路。其中双极型电路有 DTL、TTL、ECL、IIL、HTL 等多种，单极型电路有 JFET、NMOS、PMOS、CMOS 等 4 种；按电路的逻辑功能的不同特点，又可分为组合逻辑电路和时序逻辑电路两大类。

1.1.2 数制和码制

1. 数制

用数字量表示物理量的大小时，仅用 1 位数码往往不够，因此经常需要用进位计数的方法组成多位数码使用。我们把多位数码中每一位的构成方法以及从低位到高位的进位规则称为数制。

在数字电路中经常使用的计数进制除了十进制以外，还有二进制、八进制和十六进制。

（1）十进制

十数制是我们最为熟悉的进位计数制。它将 0、1、2、3、4、5、6、7、8、9 十个数字符号，按照一定的规律排列起来，表示数值的大小。例如：

$$1886 = 1\times10^3+8\times10^2+8\times10^1+6\times10^0$$

从这个 4 位十进制数，不难发现十进制数的特点如下。

① 每一位数必然是 10 个数字符号中的一个。所以它计数的基数为 10。

② 同一个数字符号在不同的数位代表的数值不同，这个 4 位数的位值依次分别为 1000、100、10、1；位值又称为权值或位权，它是 10 的幂。

③ 低位数和相邻的高位数之间的进位关系是“逢十进一”。

有了基数和位权的概念，对于任一个十进制数 N 按其位权值展开均可表示为

$$(N)_{10} = a_{n-1}\times10^{n-1}+a_{n-2}\times10^{n-2}+\cdots+a_1\times10^1+a_0\times10^0+a_{-1}\times10^{-1}+\cdots+a_{-m}\times10^{-m}= \sum_{i=n-1}^{-m} a_i\times10^i \qquad (1.1)$$

式（1.1）中 a_i 为 0～9 中任一数码，n 和 m 为正整数，n 为整数部分的位数，m 为小数部分的位数。那么，对于任意进制数，我们可以写成下式：

$$(N)_R = \sum_{i=n-1}^{-m} r_i \times R^i \tag{1.2}$$

式（1.2）中 r_i 为任意进制中第 i 位的数码，数码可以是 0～R−1 中任一个，n 和 m 为正整数，n 为整数部分的位数，m 为小数部分的位数，R 为进位基数，R^i 为第 i 位的权值。

本书中常用的进位计数制是十进制（Decimal）、二进制（Binary）、八进制（Octadic）和十六进制（Hexadecimal）。因此，当基数 R 为 10 时，表示十进制数可用$(N)_{10}$表示。同样二进制数、八进制数、十六进制数可分别用$(N)_2$、$(N)_8$、$(N)_{16}$表示。

（2）二进制

二进制数是在数字电路中应用最广的计数体制。它只有 0 和 1 两个数字符号，所以计数的基数为 2。各位数的权值是 2 的幂，低位和相邻高位之间的进位关系是“逢二进一”。因此任意一个二进制数$(N)_2$可以表示为

$$(N)_2 = b_{n-1}\times 2^{n-1}+b_{n-2}\times 2^{n-2}+\cdots+b_1\times 2^1+b_0\times 2^0+b_{-1}\times 2^{-1}+\cdots+b_{-m}\times 2^{-m}= \sum_{i=n-1}^{-m} b_i\times 2^i \tag{1.3}$$

式（1.3）中 b_i 只能取 0 或者 1 两个数码，2^i 为第 i 位的权值。例如：

$(1101.101)_2 = 1\times 2^3+1\times 2^2+0\times 2^1+1\times 2^0+1\times 2^{-1}+0\times 2^{-2}+1\times 2^{-3}$

二进制数的运算规则为

加法：0+0＝0、0+1=1+0＝1、1+1=10；乘法：0×0＝0、0×1=1×0＝0、1×1=1。

【例 1.1】 一个八位二进制整数为$(N)_2=(10011110)_2$，求其对应十进制的数值。

解：将二进制数按权展开，求各位数值之和，可得

$$\begin{aligned}(N)_2 = (10011110)_2 &= (1\times 2^7+0\times 2^6+0\times 2^5+1\times 2^4+1\times 2^3+1\times 2^2+1\times 2^1+0\times 2^0)_{10}\\ &= (128+16+8+4+2)_{10} = (158)_{10}\end{aligned}$$

（3）八进制

二进制数虽在计算机中易于实现，然而它最大的缺点是不便读写，与十进制数相比，表示同一个数时二进制用的位数较多。为此，在数字系统中，又常使用八进制数和十六进制数。

计数基数 R=8 时，称为八进制。它有 0～7 八个数字符号，各位数的权值是 8 的幂，低位和相邻高位之间的关系是“逢八进一”。因此任意一个八进制数$(N)_8$可以表示为

$$(N)_8 = q_{n-1}\times 8^{n-1}\times q_{n-2}\times 8^{n-2}+\cdots+q_1\times 8^1+q_0\times 8^0+q_{-1}\times 8^{-1}+\cdots+q_{-m}\times 8^{-m} = \sum_{i=n-1}^{-m} q_i\times 8^i \tag{1.4}$$

式中 q_i 只能取 0～7 中的某一数码。例如：$(325.7)_8=3\times 8^2+2\times 8^1+5\times 8^0+7\times 8^{-1}$

【例 1.2】 求 3 位八进制$(N)_8=(236)_8$所对应的十进制数的值。

解：按权展开，求各位数值之和，可得

$$(236)_8 = (2\times 8^2+3\times 8^1+6\times 8^0)_{10} = (128+24+6)_{10} = (158)_{10}$$

（4）十六进制

在十六进制数中，计数基数为 16，有 16 个不同的数字符号：0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。这里十进制数的 10～15 分别用 A～F 6 个英文字母表示。低位和相邻高位间的关系是“逢十六进一”。因此，任意一个十六进制数$(N)_{16}$可表示为

$$(N)_{16} = h_{n-1}\times 16^{n-1}+h_{n-2}\times 16^{n-2}+\cdots+h_1\times 16^1+h_0\times 16^0+h_{-1}\times 16^{-1}+\cdots+h_{-m}\times 16^{-m} = \sum_{i=n-1}^{-m} h_i\times 16^i \tag{1.5}$$

式中 h_i 只能取 0～F 中的某一个数码。例如：$(3A.9E)_{16} = 3\times 16^1+A\times 16^0+9\times 16^{-1}+E\times 16^{-2}$

【例 1.3】 求二位十六进制数$(N)_{16}=(9E)_{16}$所对应的十进制数的值。

解：按权展开，求各位数值之和，可得 $(9E)_{16}=(9\times16+14\times16^0)_{10}=(158)_{10}$

同一个十进制数，当分别由二进制、八进制、十六进制来表示时，八进制、十六进制要比二进制简单得多，而二进制转换成八进制和十六进制十分方便，因此，书写计算机程序时，广泛使用八进制和十六进制。

表 1.1 列出了几种常用计数进制对照表。

表 1.1 中几种数制各有其优缺点，应用场合也不相同。十进制数虽然是生活中最常用、最习惯的一种进位制数，但其 10 个数码在数字电路中难于找到 10 个状态与之对应。而二进制数只有 0、1 两个数码，可用来表示电路的两种工作状态。所以在数字电路中采用二进制。当二进制中其数位较多而不易读写时，常采用八进制和十六进制。

2. 数制转换

数制之间的转换可归为两类：十进制数和非十进制之间的转换；2^n进制数之间的转换。

（1）非十进制数转换成十进制数

由二进制、八进制、十六进制数的一般表达式可知，只要将它们按权展开，求各位数值之和，即可得到对应的十进制数。

表 1.1　　几种常用计数进制对照表

十 进 制	二 进 制	八 进 制	十 六 进 制
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

【例 1.4】 将非十进制数$(1011.011)_2$、$(27.46)_8$、$(C2)_{16}$转换成十进制数。

解：按权展开，求各位数值之和。

$$(1011.011)_2=(1\times2^3+1\times2^1+1\times2^0+1\times2^{-2}+1\times2^{-3})_{10}=(8+2+1+0.25+0.125)_{10}=(11.375)_{10}$$

$$(27.46)_8=(2\times8^1+7\times8^0+4\times8^{-1}+6\times8^{-2})_{10}=(16+7+0.5+0.09375)_{10}=(23.59375)_{10}$$

$$(C2)_{16}=(12\times16^1+2\times16^0)_{10}=(194)_{10}$$

（2）十进制数转换成非十进制数

十进制数转换成非十进制数时，要将其整数部分和小数部分分别转换，结果合并为目的数制形式。

① 整数部分的转换

整数部分的转换采用基数除法。所谓基数除法即用目的数制的基数去除十进制整数，第一次除所得的余数为目的数的最低位，把得到的商再除以该基数，所得余数为目的数的次低位，依次类推，直至商为 0 时，所得余数为目的数的最高位。此法也叫除基取余法。

② 小数部分的转换

小数部分的转换是采用基数乘法进行的。所谓基数乘法即用该小数乘目的数制的基数，第一次乘得结果的整数部分为目的数的最高位（当然是小数部分的最高位），其小数部分再乘基数，所得结果的整数部分作为目的数的第二位，依次类推，直至小数部分为 0 或达到要求精度为止。此法也叫乘基取整法。

【例 1.5】 把$(26)_{10}$分别转换成二进制数、八进制数、十六进制数。

2 | 26	0	低位	8 | 26	2		16 | 26	10
2 | 13	1		8 | 3	3		16 | 1	1
2 | 6	0		0			0	
2 | 3	1	↑					
2 | 1	1						
0		高位					

$(26)_{10}=(11010)_2$　　　$(26)_{10}=(32)_8$　　　$(26)_{10}=(1A)_{16}$

【例 1.6】 把$(0.875)_{10}$转换成二进制数。

$0.875\times2=1.750$　　1

$0.75\times2=1.500$　　1

$0.500\times2=1.000$　　1

所以，$(0.875)_{10}=(0.111)_2$

【例 1.7】 把$(0.423)_{10}$换成二进制数（保留 4 位小数）。

$0.423\times2=0.846$　　0

$0.846\times2=1.692$　　1

$0.692\times2=1.384$　　1

$0.384\times2=0.768$　　0

$0.768\times2=1.536$　　1

一般保留 4 位小数，则第 5 位小数采取“零舍一入”的原则，所以$(0.432)_{10}=(0.0111)_2$

从本例可知，十进制小数有时不能用二进制小数精确地表示出来，这时只能根据精度要求，求到一定的位数，近似地表示。

（3）2^n 进位制数之间的转换

① 二进制数与八进制数之间的转换

八进制的基数 $8=2^3$，所以 3 位二进制数构成 1 位八进制数。若要将二进制数转换成八进制数时，只要将二进制整数部分自右往左每 3 位分一组，最后不足 3 位时左边用 0 补足；小数部分则自左往右每 3 位分为一组，最后不足 3 位时在右面用 0 补足。再把每 3 位二进制数对应的八进制数码写出即可。

【例 1.8】 试将二进制数$(1010011100.101110111)_2$转换成八进制数。

001 010 011 100 . 101 110 111

$$\begin{array}{ccccccc} 1 & 2 & 3 & 4. & 5 & 6 & 7 \end{array}$$

$$(1010011100.101110111)_2 = (1234.567)_8$$

如果将八进制数转换成二进制数时，只要写出每位数码所对应的二进制数，依次排好即可。

【例 1.9】 试将八进制数$(463.57)_8$转换成二进制数。

$$\begin{array}{cccccc} 4 & 6 & 3 & . & 5 & 7 \\ 100 & 110 & 011 & . & 101 & 111 \end{array}$$

$$(463.57)_8 = (100110011.101111)_2$$

②二进制数与十六进制数之间的转换

由于十六进制的基数 $16=2^4$，所以 4 位二进制数对应 1 位十六进制数。按照上述转换步骤，只要将二进制数按 4 位分组，即可实现它们之间的转换。

【例 1.10】 试将二进制数$(10110100111100.100101111)_2$转换成十六进制数。

$$\begin{array}{cccccccc} 0010 & 1101 & 0011 & 1100 & . & 1001 & 0111 & 1000 \\ 2 & D & 3 & C & . & 9 & 7 & 8 \end{array}$$

$$(10110100111100.1001011110)_2 = (2D3C.978)_{16}$$

【例 1.11】 试将十六进制数$(3AF6.5B)_{16}$转换成二进制数。

$$\begin{array}{ccccccc} 3 & A & F & 6 & . & 5 & B \\ 0011 & 1010 & 1111 & 0110 & . & 0101 & 1011 \end{array}$$

$$(3AF6.5B)_{16} = (0011101011110110.01011011)_2$$

如果要进行八进制和十六进制数之间的转换，均可通过二进制作为转换媒介。

3. 码制

在数字电路系统中，由 0 和 1 组成的二进制数码不仅可以表示数值的大小，而且还可以表示特定的信息。这种具有特定含义的数码称之为二进制代码。常见的代码有二-十进制码和格雷码。

（1）二-十进制代码（BCD 码）

用 4 位二进制数组成一组代码来表示 0～9 十个数字，这种代码称为二-十进制代码（Binary Coded Decimal）简称 BCD 码。常见的 BCD 码有 3 种。

① 8421 码

BCD 码可以分为有权码和无权码。所谓有权码即每一位都有固定数值的码。有权码中用得最多的是 8421BCD 码，该码共有 4 位，其位权值自高位至低位分别为 8、4、2、1，故称 8421 码，它属于恒权码。每个代码的各位数值之和就是它表示的十进制数。8421 码与十进制数之间的关系是 4 位二进制代码表示 1 位十进制数。例如：$(69)_{10} = (01101001)_{8421}$

② 2421 码

2421 码也是一种有权码，也属于恒权码。该码从高位到低位的权分别是 2、4、2、1，也是 4 位二进制代码表示 1 位十进制数。该码中，0 和 9、1 和 8、2 和 7、3 和 6、4 和 5 互为反码，即两码对应位的取值相反。这种码不具备单值性，易产生伪码。

③ 余 3 码

余 3 码所组成的 4 位二进制数，正好比它代表的十进制数多 3，故称余 3 码，两个余 3 码相加时，其和要比对应表示的十进制数之和多 6。在余 3 码中，0 和 9，1 和 8，2 和 7，3 和 6，4 和 5 也互为反码。余 3 码不能由各位二进制数的权来决定某代表的十进制数，故属于无权码。

3 种常见的 BCD 码表示法如表 1.2 所示。

表 1.2　　常见 BCD 码

十进制整数	8421 码	2421 码	余 3 码
0	0000	0000	0011
1	0001	0001	0100
2	0010	0010	0101
3	0011	0011	0110
4	0100	0100	0111
5	0101	1011	1000
6	0110	1100	1001
7	0111	1101	1010
8	1000	1110	1011
9	1001	1111	1100

（2）格雷码

格雷码（Gray Code）的特点是：相邻两个代码之间仅有 1 位不同，其余各位均相同。计数电路按格雷码计数时，每次状态更新仅有 1 位代码变化。减少了出错的可能性。格雷码属于无权码。它有多种代码形式，其中最常用的一种是循环码。表 1.3 给出了 4 位循环码的编码表。

循环码中，不仅相邻两个代码只有 1 位不同，而且首尾（0 和 15）两个代码也仅有 1 位不同，构成一个“循环”，故称为循环码。此外这种代码还具有“反射性”。即以中间为对称的两个代码（如 0 和 15；1 和 14、……、7 和 8）也只有 1 位不同，所以又把它称为反射码。

表 1.3　　4 位循环码编码表

十进制数	循环码	十进制数	循环码
0	0000	8	1100
1	0001	9	1101
2	0011	10	1111
3	0010	11	1110
4	0110	12	1010
5	0111	13	1011
6	0101	14	1001
7	0100	15	1000

1.2　逻辑代数的基本运算

在客观世界中，事物的发展变化通常都是有一定因果关系的。例如，电灯的亮、灭决定于电源是否接通，如果接通了，电灯就会亮，否则就灭。电源接通与否是因，电灯亮不亮是果。这种因果关系，一般称为逻辑关系，反映和处理这种关系的数学工具，就是逻辑代数。

逻辑代数，是英国数学家 George Boole 在 19 世纪中叶创立的，所以也叫布尔代数。直到 20 世纪 30 年代，美国人 Claude E.Shannon 在开关电路中才找到了它的用途，并且很快就成为分析设计和综合开关电路的重要数学工具，因此又常称之为开关代数。

和普通代数比较起来，在逻辑代数中，虽然也用英文字母表示变量，但情况要简单得多。在二值逻辑中，变量取值不是 1 就是 0，没有第三种可能。而且这里的 0 和 1 并不是表示数值的大小，它们所代表的是两种不同的逻辑状态。例如，用 1 和 0 分别表示一件事的是与非、真与假，电压的高与低，电流的有与无，一个开关的开通与关断，一盏电灯的亮与灭等。在逻辑代数中，有些公式和定理与普通代数并无区别，有些则完全不同。

逻辑代数基本运算有与、或、非 3 种。下面结合指示灯控制电路的实例分别进行讨论。