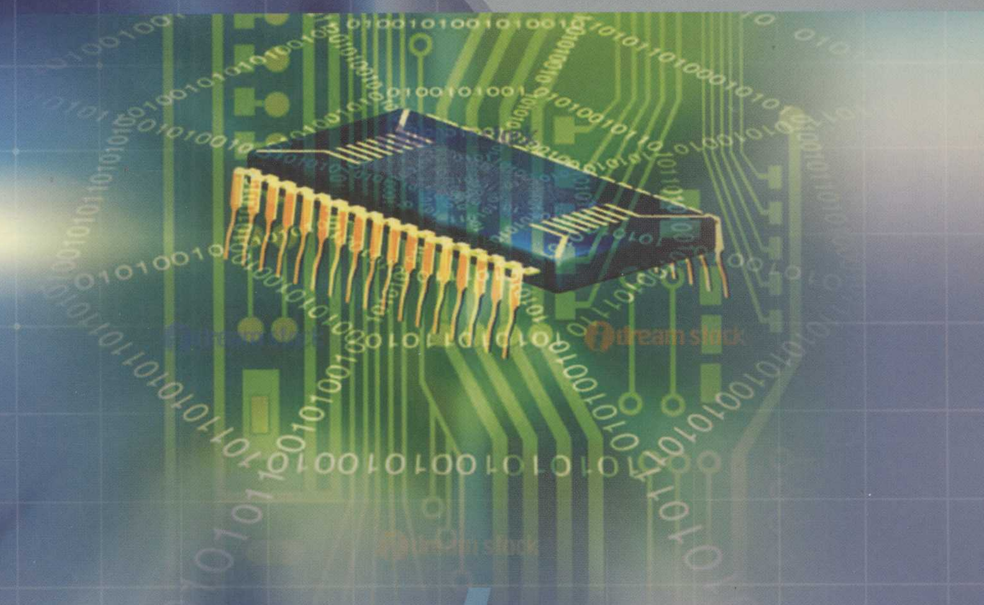
高等职业教育规划教材·电子信息类

数字电子技术

SHUZI

主　编　杨成利

副主编　董蕴华



黄河水利出版社

高等职业教育规划教材·电子信息类

数字电子技术

主　编　杨成利

副主编　董蕴华

黄河水利出版社

内 容 提 要

本书内容包括数制与代码、逻辑代数基础、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、脉冲波形的产生与整形、数／模和模／数转换器、半导体存储器与可编程逻辑器件，共九章，每章均有例题和习题。全书综合有关专业的大纲要求，尽力写出适应面较宽的教材。本教材适用于高等工科院校有关专业本科生、高职高专学生，也可供相关工程技术人员参考。

图书在版编目(CIP)数据

数字电子技术／杨成利主编. —郑州：黄河水利出版社，2005.6

ISBN 7－80621－721－5

Ⅰ.数…　Ⅱ.杨…　Ⅲ.数字电路－电子技术　Ⅳ.TN79

中国版本图书馆 CIP 数据核字(2005)第 052588 号

组稿编辑：郭安周　电话：0371－66024993　E-mail:guoanzhou@sohu.com

出 版 社：黄河水利出版社

地址：河南省郑州市金水路 11 号　　邮政编码：450003

发行单位：黄河水利出版社

发行部电话：0371－66026940　传真：0371－66022620

E-mail:yrcp@public.zz.ha.cn

承印单位：黄河水利委员会印刷厂

开本：787 mm×1 092 mm　1／16

印张：11

字数：250 千字　　印数：1—4 000

版次：2005 年 6 月第 1 版　　印次：2005 年 6 月第 1 次印刷

书号：ISBN 7－80621－721－5／TN·3　　定价：19.00 元

前　言

在当今中国的教育、科研、国防和经济领域里，电子技术的应用日益广泛，数字电路更是以系统集成化、设计自动化、用户专业化和测试智能化的趋势出现。数字电子技术作为电类专业的一门重要的技术基础课，它不仅有自身的理论体系，而且又有很强的实践性，为此，我们在编写过程中采取以下一些做法。

1．在讲授基础理论内容时，以“必须”和“够用”为尺度，删掉了与器件应用无直接联系的内容，同时又保持了课程体系的完整性。

2．在讲述集成电路时，重点介绍外部特性和正确的使用方法，对电路的内部结构仅作简单的定性介绍。

3．在处理传统内容和新技术的关系时，大幅度地削减了已经过时的内容，同时增加了新型器件及其应用的内容。

本书可作为高等工科院校本科、高职高专电类专业使用，也可供其他专业及相关工程技术人员参考。

本书由河南机电高等专科学校杨成利任主编、董蕴华任副主编。具体分工如下：第三、四、八章及附录三由杨成利编写；第二、五章及附录一由董蕴华编写；第一、七、九章及附录二由郑州大学牛虹编写；第六章由河南新乡市高级技工学校禹玺编写；附录四由河南机电高等专科学校杨其锋编写。全书由河南机电高等专科学校张亚华教授主审。

在本书的编写过程中，得到了河南机电高等专科学校于圣学教授和张新成、杜志勇、董作霖副教授的支持与帮助，在此表示感谢。

由于编者水平有限，书中难免出现不妥和错误之处，敬请读者批评指正。

编　者

2005 年 3 月

目 录

第 1 章　数制与代码

应知、应会要求

1. 理解“权”的概念，从而能写出任意进制数的按权展开式。
2. 掌握十进制、二进制、八进制、十六进制之间的转换方法。
3. 熟悉 8421BCD 码、2421BCD 码、5421BCD 码、余 3BCD 码和 Gray 码的编码方法以及它们之间的转换方法。

在日常生活中，我们习惯用十进制数，而在数字系统中进行数字的运算和处理，采用的是二进制数、八进制数、十六进制数。为了更好地实现人机对话，我们应当掌握各种数制、代码的特点及其相互之间的转换规律。本章讲述常用数制、常用代码及其相互之间的转换方法。

1.1　进位计数制及数制转换

1.1.1　数制

数制即指计数的方法，日常生活中经常使用的有十进制、二十四进制等。数字电路经常使用二进制、八进制及十六进制。

1.1.1.1　十进制

在日常生活中，人们通常采用十进制数来计算，每个数位规定使用的数码符号为 0、1、2、3、4、5、6、7、8、9，共十个数码，故计数的基数是 10。

十进制数的计算规律是由低位向高位进位，“逢十进一”，也就是说每位累计不能超过 9，计满 10 就应向高位进 1。

任意一个十进制数 D 可展开为

$$D=\sum_{i=-\infty}^{\infty}a_i 10^i$$

式中：a_i 是第 i 位的数码，它属于 0 ~ 9 十个数码中的任何一个；10^i 称为第 i 位的权。

例如：

$$632.45=6\times10^2+3\times10^1+2\times10^0+4\times10^{-1}+5\times10^{-2}$$

通常，对十进制数的表示，可以在数字的右下角标注 10 或 D。例如：$(25.7)_{10}$ 或 $(2.57)_D$。

1.1.1.2　二进制

在数字电路中广泛采用的是二进制数。在二进制中，每个数位规定使用的数码为 0、1，共两个数码，其进位基数为 2，计数规则是“逢二进一”，故称为二进制。

任意一个二进制数 D 可展开为

$$D=\sum_{i=-\infty}^{\infty}a_i 2^i$$

式中：a_i 是 i 位的数码，它属于 0、1 两个数码之一；2^i 称为第 i 位的权。

通常，对二进制数的表示，可以在数字右下角标注 2 或B。例如：

$$(1011.01)_2=1\times2^3+0\times2^2+1\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}$$

1.1.1.3 八进制

在八进制数中，每个数位规定使用的数码为 0、1、2、3、4、5、6、7，共 8 个数码，故其进位基数为 8，计数规则是“逢八进一”，故称为八进制。

任意一个八进制数 D 可展开为

$$D=\sum_{i=-\infty}^{\infty}a_i 8^i$$

式中：a_i 是第 i 位的数码，它属于 0 ~ 7 八个数码中的任何一个；8^i 称为第 i 位的权。

通常，对八进制数的表示，可以在数字右下角标注 8 或 O，例如：

$$(752.34)_8=7\times8^2+5\times8^1+2\times8^0+3\times8^{-1}+4\times8^{-2}$$

1.1.1.4 十六进制

在十六进制数中，每个数位规定使用的数码为 0、1、2、3、4、5、6、7、8、9、A(10)、B(11)、C(12)、D(13)、E(14)、F(15)，共十六个数码，故其进位基数为 16，计数规则是“逢十六进一”，故称为十六进制。

任意一个十六进制数 D 可展开为

$$D=\sum_{i=-\infty}^{\infty}a_i 16^i$$

式中：a_i 是第 i 位的数码，它属于 0 ~ F 共十六个数码中的任何一个；16^i 称为第 i 位的权。

通常，对十六进制数的表示，可以在数字右下角标注 16 或 H。例如：

$$(BD2.3C)_H=11\times16^2+13\times16^1+2\times16^0+3\times16^{-1}+12\times16^{-2}$$

1.1.2 数制转换

1.1.2.1 非十进制数转换成十进制数

把非十进制数转换成等值的十进制数，可采用按权展开，再把各项的数值相加的方法。

【例 1】 将 $(1101.11)_2$、$(165.2)_8$、$(2B.8)_{16}$ 分别转换成十进制数。

解：

$$\begin{aligned}(1101.11)_2&=1\times2^3+1\times2^2+0\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-2}\\&=8+4+0+1+0.5+0.25\\&=(13.75)_{10}\end{aligned}$$

$$\begin{aligned}(165.2)_8&=1\times8^2+6\times8^1+5\times8^0+2\times8^{-1}\\&=64+48+5+0.25=(117.25)_{10}\end{aligned}$$

$$(2B.8)_{16}=2\times16^1+11\times16^0+8\times16^{-1}$$

$=32+11+0.5=(43.5)_{10}$

1.1.2.2　十进制数转换成其他进制数

对于既有整数部分又有小数部分的十进制数转换为其他进制数，需先把整数部分和纯小数部分分别转换，再把两者的转换结果相加。

整数部分转换方法采用“除 N 取余”法，N 为进制基数。即先将十进制整数除以 N，取出余数（余数 $<N$）并记下，然后将所得商再除以 N，并再取余数，重复进行上述过程，直到商为 0 时为止，并按照和运算过程相反的顺序把各个余数排列起来，即为所求 N 进制数上述过程的整数部分

【例 2】　将 $(27)_{10}$ 转换成十六进制数。

解：

$$\begin{array}{rl} 16 \,|\, \underline{27} & \quad\text{余数} \\ 16 \,|\, \underline{1} & \cdots\cdots\ 11 \\ 0 & \cdots\cdots\ 1 \end{array}$$

即：$(27)_{10}=(1B)_{16}$

【例 3】　将 $(137)_{10}$ 转换成八进制数。

解：

$$\begin{array}{rl} 8 \,|\, \underline{137} & \quad\text{余数} \\ 8 \,|\, \underline{17} & \cdots\cdots\ 1 \\ 8 \,|\, \underline{2} & \cdots\cdots\ 1 \\ 0 & \cdots\cdots\ 2 \end{array}$$

即：$(137)_{10}=(211)_8$

【例 4】　将 $(28)_{10}$ 转换成二进制数。

解：

$$\begin{array}{rl} 2 \,|\, \underline{28} & \quad\text{余数} \\ 2 \,|\, \underline{14} & \cdots\cdots\ 0 \\ 2 \,|\, \underline{7} & \cdots\cdots\ 0 \\ 2 \,|\, \underline{3} & \cdots\cdots\ 1 \\ 2 \,|\, \underline{1} & \cdots\cdots\ 1 \\ 0 & \cdots\cdots\ 1 \end{array}$$

即：$(28)_{10}=(11100)_2$

纯小数部分转换方法采用“乘 N 取整”法，N 为进位基数。即先将十进制小数乘以 N，取其整数并记下，然后将积的小数部分再乘以 N，并再取整数，重复进行上述过程，直到小数部分为 0 或达到所要求的精度时为止，并按照和运算过程相同的顺序把各个整数排列起来，即为所求的 N 进制数的纯小数部分。

【例 5】　将 $(0.75)_{10}$ 转换成二进制数。

解：

$$\begin{array}{lr} & \text{整数} \\ 0.75\times 2=1.5 \cdots\cdots & 1 \\ 0.5\times 2=1 \cdots\cdots & 1 \end{array}$$

即：$(0.75)_{10}=(0.11)_2$

【例 6】　将$(0.85)_{10}$转换成八进制数，精确到小数点后 3 位。

解：

	整数
$0.85\times 8=6.8$ ……	6
$0.8\times 8=6.4$ ……	6
$0.4\times 8=3.2$ ……	3
$0.2\times 8=1.6$ ……	1

即：$(0.85)_{10}=(0.663)_8$

【例 7】　将$(28.75)_{10}$转换为二进制数。

解：由例 4 知$(28)_{10}=(11100)_2$，由例 5 知$(0.75)_{10}=(0.11)_2$，故$(28.75)_{10}=(11100.11)_2$。

1.1.2.3　二进制数转换成八进制数或十六进制数

八进制数的基数为 8$(8=2^3)$，十六进制数的基数为 16$(16=2^4)$。二进制数、八进制数和十六进制数之间具有 2 的整指数倍的关系，因而可直接进行转换。

将二进制数转换成八进制数(或十六进制数)时，其整数部分和小数部分可以同时进行转换，其方法为：从小数点开始，分别向左、向右，每 3 位(或 4 位)分一组。最后不满 3 位(或 4 位)的，则需加 0，使其足位。然后把每一组二进制数转换成八进制数(或十六进制数)，并保持原排序，即为等值的八进制数(或十六进制数)。

【例 8】　将$(10111011.01111)_2$转换成八进制数和十六进制数。

解：

八进制　　2　7　3　.　3　6

二进制　　0 1 0 1 1 1 0 1 1 . 0 1 1 1 1 0 0 0

十六进制　　B　B　.　7　8

即：$(10111011.01111)_2=(273.36)_8=(BB.78)_{16}$

1.1.2.4　八进制数或十六进制数转换成二进制数

八进制数(或十六进制数)转换成二进制数时，从小数点开始，分别向左、向右把八进制数(或十六进制数)的每一位数码分别转换成 3 位(或 4 位)的二进制数，并保持原排序。即为等值的二进制数。整数最高位一组左边的 0 及小数最低位一组右边的 0 可以省略。

【例 9】　将$(26.35)_8$转换成二进制数。

解：

2　6　·　3　5

↓　↓　　↓　↓

010　110　　011　101

即：$(26.35)_8=(10110.011101)_2$

1.2 数码的代码表示

1.2.1 十进制数的二进制编码

十进制数的二进制编码简称二—十进制码或BCD码(Binary Coded Decimal)。所谓BCD码是指用若干位二进制数来表示一位十进制数。这里的“二进制”并无“进位”的含义，只是强调采用的是二进制数的符号而已。

表示一位十进制数0~9十个数码，需用四位二进制数，而四位二进制数共16种组合，从中取出10种组合来表示“0~9”的编码方案约有2.9×10^{10}种，表1-1列举了目前常用的几种编码方案。

表1-1　几种常用的BCD码

十进制数	8421码	2421码	5421码	余3码	Gray码
0	0000	0000	0000	0011	0000
1	0001	0001	0001	0100	0001
2	0010	0010	0010	0101	0011
3	0011	0011	0011	0110	0010
4	0100	0100	0100	0111	0110
5	0101	1011	1000	1000	0111
6	0110	1100	1001	1001	0101
7	0111	1101	1010	1010	0100
8	1000	1110	1011	1011	1100
9	1001	1111	1100	1100	1000

1.2.1.1 8421BCD码

8421BCD码是有固定权的码，各位的权值分别为8、4、2、1。8421BCD码是最基本最简单的一种编码方案，应用十分广泛。

1.2.1.2 2421BCD码

2421BCD码是有固定权码，各位的权值分别为2、4、2、1。

1.2.1.3 5421BCD码

5421BCD码是有固定权的码，各位的权值分别为5、4、2、1。

1.2.1.4 余3BCD码

余3BCD码是8421BCD码的每个码组加0011形成,余3BCD码的各位无固定的权，称余3BCD码是无权码。余3BCD码也是一种对9的自补代码，其中0和9、1和8、2和7、3和6、4和5的余3BCD码互为自补代码。

用BCD码表示十进制数时,只要把十进制数的每一位数码分别用BCD码取代即可。反之，若要知道BCD码代表的十进制数，只要把BCD码以小数点为起点向左、向右每四位分一组，再写出每一组代码代表的十进制数，并保持原排序即可。例如：

$$(85.34)_{10}=(10000101.00110100)_{8421BCD}$$
$$=(10111000.01100111)_{余3BCD}$$

1.2.2 可靠性代码

代码在形成和传送过程中，都可能发生错误，为减少错误的发生，或者在发生错误时能迅速地发现或纠正，产生了几种可靠性代码。最常用的可靠性代码有格雷码和奇偶校验码。

1.2.2.1 格雷码(Gray 码)

格雷码有多种编码形式，但它们有一个共同的特点，就是任何相邻的两个代码之间(包括首、尾两个代码)只有一位不同，其余各位均相同。因而格雷码也叫循环码。格雷码属于无权码。

格雷码所具有的特点可以降低其产生错误的概率。例如按自然数规律计数，为完成十进制数 5 加 1 的运算，其相应的四位二进制代码从 0101 变为 0110，代码 0101 的最低两位都要改变。若两位的变化不是同时发生的(在实际电路中，没有绝对的同时改变)，那么，在计算过程中就可能短暂地出现其他代码(0111 或 0100)，尽管这种误码出现时间是短暂的，但有时却是不允许的，这会造成数字系统的逻辑错误，而采用格雷码，其相应的代码从 0111 变为 0101，只有一位发生变化，就不会出现这种错误。

1.2.2.2 奇偶校验码

奇偶校验码是一种可以检测奇数位错误的代码。它由信息位和校验位两部分组成。信息位可以是任何一种二进制代码，它代表要传送的信息本身；校验位仅有一位，它可以放在信息位的前面或后面。其编码方式有以下两种。

(1)使每一个码组中信息位和校验位的“1”的个数之和为奇数，称为奇校验。

(2)使每一个码组中信息位和校验位的“1”的个数之和为偶数，称为偶校验。

接收方对接收到的奇偶校验码进行检测，看每个码组中“1”的个数是否与发送方约定的相符，若不相符，则出现错码。

需要说明的是，奇偶校验码只能发现代码的一位(或奇数位)出错，而不能发现两位(或偶数位)出错。由于多位同时出错的概率要比一位出错的概率小得多，并且奇偶校验码容易实现，因而该码被广泛采用。

此外，在数字电路中，还有一些专门处理字母、标点符号、运算符号的二进制代码，如 ASCII 码、ISO 码等，读者可参阅有关书籍。

本章小结

数字系统中广泛使用二进制数，二进制数每位的权是基数 2 的整数幂。

通过对十进制数和二进制数的讨论，揭示了任意 N 进制数 D 按权展开式的一般表示式为

$$D=\sum_{i=-\infty}^{\infty}a_iN^i$$

式中：a_i是第 i 位的数码；N 为进位基数。

除十进制、二进制数以外，还讨论了八进制数和十六进制数，以及这几种数制之间相互转换的方法，特别是十进制数转换为二进制数的方法应牢固地掌握。

数字系统中最常用的编码有 8421BCD 码、2421BCD 码、5421BCD 码、余 3BCD 码等。8421BCD 码便于进行十进制数的运算，所以使用最为广泛。Gray 码是循环码，在计算机中使用较多。

奇偶校验码是一种简单的检错码，它只能检测出奇数位同时发生的错误，而不能纠错。

习　题

一、填空题

1. 数制是指_________的方法，数字电路中经常使用_________进制、_________进制及_________进制。其中，广泛采用的是_________进制。

2. 十进制数的二进制编码简称_________码。余 3BCD 码是 8421BCD 码的每个码组加_________形成。

3. 常用的两种可靠性代码为_________码和_________码。

4. $(1011001)_2=(\qquad)_8=(\qquad)_{16}=(\qquad)_{10}$

5. $(154.25)_{10}=(\qquad)_8=(\qquad)_{16}=(\qquad)_2$

6. $(154.23)_{10}=(\qquad)_{8421BCD}=(\qquad)_{余3BCD}$

二、选择题

1. 下列 BCD 码(　　)是无权码。

   A. 8421 码　　B. 余 3 码　　C. 2421 码　　D. 5421 码

2. $(A6C)_{16}$ 对应的二进制数为(　　)。

   A. $(A6C)_2$　　B. $(1D612)_2$　　C. $(101001101100)_2$　　D. $(100110110)_2$

3. $(15.5)_{10}$ 对应的 8421BCD 码为(　　)。

   A. 1101.101　　B. 10101.0101　　C. 1111.1　　D. 1111.101

4. $(15.5)_{10}$ 对应的二进制数为(　　)。

   A. 1101.101　　B. 10101.0101　　C. 1111.1　　D. 1111.101

三、判断题

1. 格雷码和余 3 码都是无权码。

2. $(1B.3)_{16}=(11011.011)_2$

3. $(100001110101.10010011)_{余3BCD}=(541.60)_{10}$

4. 二进制代码 1010101 的奇校验位为 0。

5. $(101.10001)_{8421BCD}=(5.11)_{10}$

四、计算题

1. 将下列几组数中的最大数和最小数用二进制数分别表示出来。

(1) $(39)_{10}$，$(101101)_2$，$(2E)_{16}$，$(48)_8$

(2) $(246)_{10}$，$(E9)_{16}$，$(401)_8$，$(100100100)_2$

(3) $(AF)_{16}$，$(101111000)_{8421BCD}$，$(254)_8$

2. 把下列各进制数按权展开。

①$(4517.239)_{10}$　②$(10110.0101)_2$　③$(105.744)_8$　④$(785.4AF)_{16}$

3. 将下列二进制数转换成十进制、八进制和十六进制数。

①101101　②10110.011

4. 求下列 BCD 码代表的十进制数。

①$(100010010110.00011001)_{8421BCD}$　②$(100010011010.01011001)_{余3BCD}$

5. 将下列 8421BCD 码转换成二进制数和十六进制数。

①00110111.01110101　②00111000

第 2 章　逻辑代数基础

应知、应会要求

1. 掌握逻辑代数的基本定律、规则。
2. 掌握逻辑函数的描述方法以及各方法之间的相互转换。
3. 掌握逻辑函数对偶式、反函数的求法以及利用代数法求函数最简与或式的方法。
4. 理解最小项、相邻项的概念，掌握四变量以内函数的卡诺图化简法。

1849 年，英国数学家乔治·布尔(G.Boole)提出了用数学分析方法表示命题陈述的逻辑关系，并将形式逻辑归结为一种代数演算，这种数学方法就是有名的“布尔代数”。因为布尔代数被广泛地应用于解决开关电路及数字电路的分析设计上，故又把布尔代数称为开关代数或逻辑代数。

目前，逻辑代数已成为研究数字系统逻辑设计的基础理论。无论何种形式的数字系统，都是由一些基本的逻辑电路所组成的。为了解决数字系统分析和设计中的各种具体问题，必须掌握逻辑代数这一重要数学工具。

本章从实用的角度介绍逻辑代数的基本运算、基本概念、基本定理和规则，并着重讨论逻辑函数的常用表示方法和逻辑函数的化简方法。

2.1　逻辑变量及基本运算

逻辑代数和普通代数一样，也用字母来表示变量，这种变量叫逻辑变量。所不同的是普通代数中的变量根据实际情况可以取多个值，而逻辑变量则不同，它的取值只有 0 或 1，而且这时 0 和 1 不再表示数量大小，只表示两种不同的逻辑状态。在数字系统中，开关的接通和断开、电压的高和低、晶体管的导通和截止等两种状态均可用 1 和 0 来表征。

在逻辑代数中，基本的逻辑运算有三种，即与运算、或运算、非运算。

2.1.1　逻辑代数的基本运算

下面举一例子对逻辑代数的基本运算进行说明。如图 2-1 所示为三种开关电路。

从图 2-1 中可以看出，若把开关的闭合作为条件，把灯泡的亮作为结果，那么三个电路图代表的逻辑关系如下：

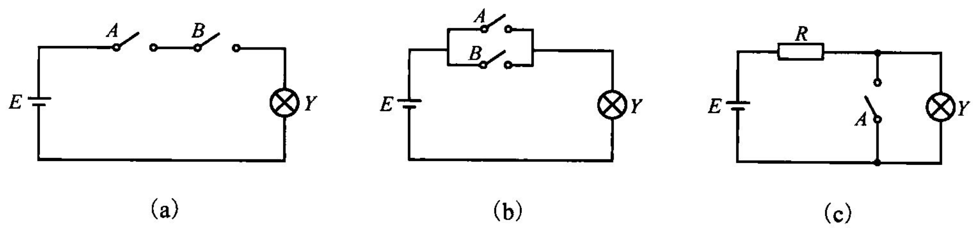


图 2-1　与、或、非三种逻辑开关电路

图 2-1(a)表示只有决定事件结果的全部条件均具备时结果才发生，这种逻辑关系叫与逻辑或逻辑乘。

图 2-1(b)表示决定事件结果的条件至少有一个满足，结果就能发生，这种逻辑关系叫或逻辑或逻辑加。

图 2-1(c)表示决定事件结果的条件不满足时，结果会发生；而当条件满足时，结果反而不会发生，这种逻辑关系叫非逻辑或逻辑反。

若以 A、B 来表示逻辑自变量，Y 表示逻辑因变量。开关断开状态用 0 表示，开关闭合状态用 1 表示；灯亮用 1 表示，灯灭用 0 表示，则可列出因变量和自变量间因果关系的表格，这种表格叫真值表。与逻辑、或逻辑、非逻辑的真值表见表 2-1、表 2-2、表 2-3。

表 2-1　与逻辑真值表

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

表 2-2　或逻辑真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

表 2-3　非逻辑真值表

A	Y
0	1
1	0

上述三种逻辑运算用式子表示出来叫逻辑表达式(也叫逻辑函数式)，即：

与逻辑表达式为

$$Y=A \cdot B$$

或逻辑表达式为

$$Y=A+B$$

非逻辑表达式为

$$Y=\overline{A}$$

上述三种逻辑运算用图形符号来表示叫逻辑符号，如图 2-2 所示。图 2-2(a)为与逻辑符号，图 2-2(b)为或逻辑符号，图 2-2(c)为非逻辑符号。

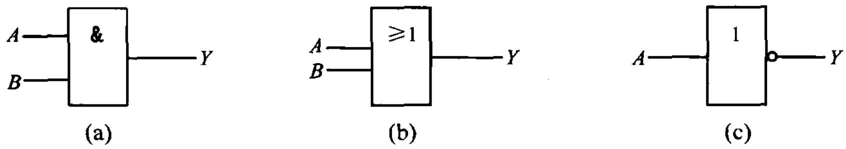
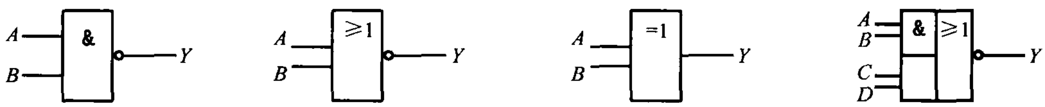


图 2-2　与逻辑、或逻辑、非逻辑的逻辑符号

把实现与逻辑、或逻辑、非逻辑的逻辑运算的电路分别叫做与门、或门、非门。

2.1.2　常用复合逻辑运算

由三种基本逻辑运算的简单组合叫复合逻辑运算。通常复合逻辑运算包含与非、或非、异或、与或非等，相应的逻辑函数式及逻辑符号如图 2-3 所示。



$Y=\overline{A\cdot B}$　(a)　$Y=\overline{A+B}$　(b)　$Y=A\oplus B$　(c)　$Y=\overline{A\cdot B+C\cdot D}$　(d)

图 2-3　复合逻辑函数的逻辑表达式和逻辑符号

图 2-3(a)为与非运算。它是与运算和非运算的组合，图形符号中小圈圈表示非运算。

图 2-3(b)为或非运算。它是或运算和非运算的组合。

图 2-3(c)为异或运算。它的运算规律是：当 A、B 取值不同时，输出 Y 为 1；当 A、B 取值相同时，输出 Y 为 0。异或运算可用与、或、非的组合表示，即

$$Y=A\oplus B=A\cdot\overline{B}+\overline{A}\cdot B$$

图 2-3(d)为与或非运算。它是与、或、非运算的组合。

以后，为了书写方便，常将与运算符号“·”省略，即将 $A\cdot B$ 简写成 AB。

2.2　逻辑代数的基本定律和规则

2.2.1　基本定律

基本定律反映了逻辑运算的一些基本规律，只有掌握了这些基本定律才能正确地分析和设计出逻辑电路。逻辑代数基本定律见表 2-4。

表 2-4 中每个定律(除否否律外)都是成对出现的，前 4 个定律和否否律可以直接代入“0”、“1”取值即可验证。5、6、7 和求反律 4 组定律可采用真值表加以验证。若将变量的所有取值代入等式两边，两边结果相等，则等式成立。

【例 1】　试用真值表验证 $\overline{A\cdot B}=\overline{A}+\overline{B}$。

解：将等式右边和左边看成两个逻辑式，将 A 和 B 所有取值代入，求出结果如表 2-5 所示。