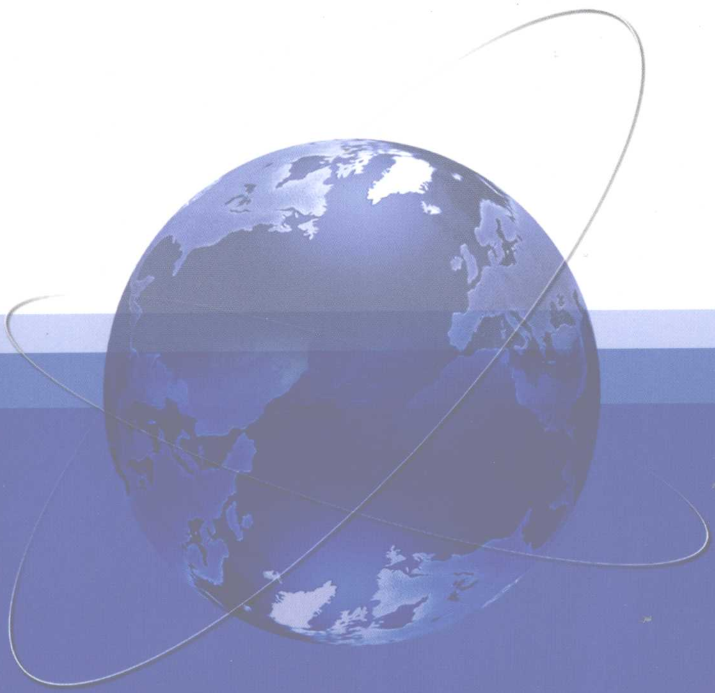
普通高等教育“十一五”国家级规划教材

21世纪高职高专规划教材（电工电子类）

数字电子技术



张明莉　王斌　主编

配多媒体课件

机械工业出版社
CHINA MACHINE PRESS

普通高等教育“十一五”国家级规划教材
21世纪高职高专规划教材（电工电子类）

数字电子技术

主　编　张明莉　王　斌
副主编　李宗宝　丛　振
参　编　艾　兰　曾献芳　倪　琳
　　　　刘阿玲　许　娅　王松林

机械工业出版社

作为“普通高等教育‘十一五’国家级规划教材”，本书以工程技术应用为出发点，由浅入深地介绍了逻辑代数基础，集成逻辑门电路，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电路，数/模、模/数转换器，半导体存储器和可编程逻辑器件，数字电路读图练习，仿真软件 Multisim7 的简介及应用等内容。本书力求全面地介绍数字电子技术的知识，并将理论与仿真实验、实训相结合，以达到举一反三、融会贯通的目的。本书简单地介绍了 Multisim7 软件，并在许多相应的章节配有相关技术应用的内容，便于读者全面掌握数字电子技术的应用。

本书可作为电气、电子信息、计算机以及部分非电类专科和高职学生的入门教材，也可作为相关专业的本科学生及工程技术人员的参考用书。

本书配有电子教案，凡一次性购书 30 本以上者免费赠送一份电子教案。请与本书责任编辑余茂祚联系（联系电话：010-88379759，电子邮箱：yumaozuo@163.com）。

图书在版编目（CIP）数据

数字电子技术/张明莉，王斌主编. —北京：机械工业出版社，2008.9

普通高等教育“十一五”国家级规划教材. 21世纪高职高专规划教材. 电工电子类

ISBN 978-7-111-25067-8

Ⅰ. 数… Ⅱ. ①张…②王… 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（2008）第 136890 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）
策划编辑：余茂祚 责任编辑：余茂祚 王 琪
版式设计：霍永明 责任校对：陈延翔
封面设计：饶 薇 责任印制：李 妍
北京地质印刷厂印刷

2009 年 1 月第 1 版第 1 次印刷
184mm×260mm · 13.75 印张 · 337 千字
0001—4000 册
标准书号：ISBN 978-7-111-25067-8
定价：23.00 元

21 世纪高职高专规划教材
编委会名单

前 言

本书是根据教育部有关文件精神，由中国机械工业教育协会和机械工业出版社组织全国80多所院校编写的高职高专规划教材之一。

本书是由多年从事电子技术基础课程教学和电子技术应用工程项目开发的教师编写的。本书将基础理论与应用技术紧密结合，内容由浅入深，有利于培养学生的自学、应变和创新能力。本书的主要内容包括逻辑代数基础，集成逻辑门电路，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电路，数/模、模/数转换器，半导体存储器和可编程逻辑器件，数字电路读图练习，仿真软件 Multisim7 的简介及应用，大部分章节后还有相应知识的技术应用实例。本书的参考学时为：理论教学学时 60，仿真实验 14。读者可以根据课时的要求自行取舍学习内容。

本书由北京联合大学、扬州工业职业技术学院、大连职业技术学院、安徽水利电力职业技术学院、无锡交通高等技术学校、安徽铜陵学院、辽宁辽东学院、安徽商贸职业技术学院部分教师共同编写。第 1 章、第 10 章由张明莉编写；第 2 章由艾兰编写；第 3 章由李宗宝、王松林编写；第 4 章由曾献芳编写；第 5 章由倪琳编写；第 6 章由丛振编写；第 7 章由刘阿玲编写；第 8 章由许娅编写；第 9 章由李宗宝编写；多媒体教学光盘由王斌制作。张明莉、王斌主编对全书进行了统稿修订；李宗宝、丛振副主编审阅了书稿。

本书在编写过程中得到了刘继承、王传新、李淑芬、王珏、梁爱琴、贺玲芳、耿钰、张世德、田文杰、吉素霞、宋玉秋、张兆莉、孙雪、胡立栓等老师的帮助，在此一并表示感谢。

由于编者水平有限，书中错误和不妥之处难免，希望广大读者批评指正。

编 者

目录

第 1 章　逻辑代数基础

逻辑代数是分析和设计数字电路的基本数学工具。本章介绍了数字逻辑的基本知识，首先介绍了数制、码制以及各种数制之间的转换关系，然后介绍了逻辑函数的表示及化简方法。

1.1　概述

1.1.1　数字信号和模拟信号

在电子技术应用中，电信号按其变化规律可以分为两大类：模拟信号和数字信号。模拟信号的变化在时间和数值上都是连续的。例如，电话线中的语音信号就是随时间作连续变化的模拟信号，它的电压信号在正常情况下是连续变化的，不会出现跳变。传输、处理模拟信号的电路称为模拟电路，而传输、处理数字信号的电路称为数字电路。数字信号在时间和数值上是断续变化的离散信号，这类信号在两种稳定状态之间作阶跃变化，这两种状态常用 0 和 1 来表示，因而称之为二值信息。模拟电压信号和数字电压信号如图 1-1 所示。

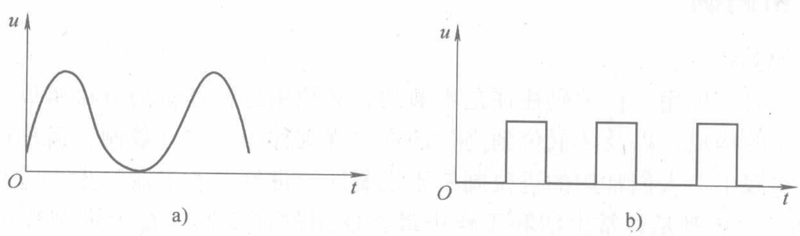


图 1-1　模拟电压信号和数字电压信号

a）模拟电压信号　b）数字电压信号

1.1.2　数字电路的特点

由于数字信号采用二值信息来表示脉冲的有、无或电平的高、低，所以数字电路在结构和工作状态、研究内容和分析方法上都与模拟电路不同。与模拟电路相比，数字电路具有如下特点：

1）由于数字信号是二值信息，所以在电路中工作的半导体管通常处在开关状态，即工作在饱和区和截止区，放大区只是过渡状态。

2）数字电路主要的研究对象是电路输入和输出间的逻辑关系。输入信号的状态（0 或 1）和输出信号的状态（0 或 1）之间的逻辑关系，反映了电路的逻辑功能。分析数字电路的方法主要有逻辑代数和卡诺图法等。

3）数字信息便于长期保存。

4）组成数字电路的单元结构较简单，便于集成化和系列化生产，工作可靠、精度高。数字集成电路产品系列多、通用性强、成本低。

1.1.3　数字电路的分类和应用

1）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（在一块硅片上包含组件数量的多少）可分为小规模集成（SSI）、中规模集成（MSI）、大规模集成（LSI）和超大规模集成（VLSI）电路。其中 SSI 电路主要是一些逻辑单元电路，如逻辑门电路、集成触发器，它的集成度为 1 ~ 10 门/片或 10 ~ 100 元器件/片；MSI 电路主要是一些逻辑功能部件，包括译码器、编码器、选择器、算术运算器、计数器、寄存器、比较器、转换电路等，它的集成度为 10 ~ 100 门/片或 100 ~ 1000 元器件/片；LSI 电路主要是由一些数字逻辑系统，如中央控制器、存储器、串并行接口电路等构成，它的集成度大于 100 门/片或大于 1000 元器件/片；VLSI 电路是高集成度的数字逻辑系统，如在一个硅片上集成一个完整的微型计算机，它的集成度大于 1000 门/片或大于 10 万元器件/片。

2）数字电路按电路所使用的器件的不同可分为双极型电路（如 DTL、TTL、ECL、IIL、HTL 电路等）和单极型电路（如 NMOS、PMOS、CMOS、HCMOS 电路等）。

3）数字电路按电路的逻辑功能的不同可分为组合逻辑电路和时序逻辑电路。组合逻辑电路没有记忆功能，其输出信号只与当时的输入信号有关，而与电路以前的状态无关；时序逻辑电路具有记忆功能，其输出信号不仅和当时的输入信号有关，而且与电路以前的状态有关。

本书以 SSI 电路为基础，以 MSI 电路为主，着重介绍各种逻辑单元电路和逻辑部件的工作原理、分析设计方法和应用举例。

1.2 数制和码制

1.2.1 进位计数制

在表示数字时，仅用一位数码往往是不够的，必须用进位计数的方法组成多位数码。多位数码中每一位的构成，以及从低位到高位的进位规则称为进位计数制，简称进位制。在日常生活及生产实践中，人们常用的进位制有十进制、二进制、八进制、十六进制等。

1. 十进制　十进制是日常生活和工作中最常使用的计数制。在十进制数中，每一位有 0 ~ 9 共 10 个数字符号，所以计数制的基数是 10，且低位和相邻高位的进位关系是“逢十进一”。任意一个十进制数可以表示为

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

式中　10——计数制的基数；

K_i——第 i 位的系数，可以是 0 ~ 9 共 10 个数字符号中的任意一个；

10^i——第 i 位的权；

n、m——正整数，n 表示整数部分位数，m 表示小数部分位数。

例如，十进制数 2008.5 可表示为

$$
\begin{aligned}
(2008.5)_{10} &= 2 \times 10^3 + 0 \times 10^2 + 0 \times 10^1 + 8 \times 10^0 + 5 \times 10^{-1} \\
&= 2000 + 0 + 0 + 8 + 0.5
\end{aligned}
$$

式中，下脚标 10 表示括号里的数是十进制数。

若以 R 取代式（1-1）中的 10，即可得到 R 进制数的普遍形式

$$
(D)_R = \sum_{i=-m}^{n-1} K_i R^i \tag{1-2}
$$

式中　R——基数，表示有 R 个不同的数字符号且逢 R 进位。

例如，八进制数 127.5 可表示为

$$(127.5)_8 = 1\times8^2 + 2\times8^1 + 7\times8^0 + 5\times8^{-1} = 64 + 16 + 7 + 0.625 = (87.625)_{10}$$

式中，下脚标 8 表示括号里的数是八进制数。

2. 二进制　目前在数字系统中广泛采用二进制。这是因为二进制数的每一位只取 0 或 1 两个数字符号，可以用某些电子元器件的两个不同稳定状态（如晶体管的饱和与截止）来表示。任意二进制数可以表示为

$$(D)_2 = \sum_{i=-m}^{n-1} K_i 2^i \tag{1-3}$$

式中　2——计数制的基数；

2^i——第 i 位的权，低位和相邻高位的进位关系是“逢二进一”。

例如，二进制数 1011.1011 可表示为

$$\begin{aligned}(1011.1011)_2 &= 1\times2^3 + 0\times2^2 + 1\times2^1 + 1\times2^0 + 1\times2^{-1} + 0\times2^{-2} + 1\times2^{-3} + 1\times2^{-4}\\ &= 8 + 0 + 2 + 1 + 0.5 + 0.25 + 0.125 + 0.0625\\ &= (11.6875)_{10}\end{aligned}$$

式中，下脚标 2 表示括号里的数是二进制数。

3. 十六进制　采用二进制计数，对数字系统来说，处理、存储、传输极为方便；然而若需表示一个较大的数，则位数较多，读出和书写都不方便，因此常采用十六进制数进行书写或打印。十六进制数有 16 个数字符号且“逢十六进一”，见表 1-1。

表 1-1　十进制数、二进制数和十六进制数的对应

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

任意一个十六进制数可以表示为

$$(D)_{16} = \sum_{i=-m}^{n-1} K_i 16^i \tag{1-4}$$

例如，十六进制数 5BC.6 可表示为

$$\begin{aligned}(5BC.6)_{16} &= 5\times16^2 + 11\times16^1 + 12\times16^0 + 6\times16^{-1}\\ &= 1280 + 176 + 12 + 0.375\\ &= (1468.375)_{10}\end{aligned}$$

式中，下脚标 16 表示括号里的数是十六进制数。

八进制和十六进制可以根据需要，借助于它们与计算规则简单的二进制关系，对十进制

进行不同的编码或转换。

1.2.2 几种数制之间的转换

数字系统采用的是二进制数，书写时采用十六进制数，因此必然产生各种进位计数制的相互转换。

1. 二进制数与十六进制数的转换 把二进制数转换为等值的十六进制数时，由于4位二进制数恰好有16个状态，所以每1位十六进制数正好对应4位二进制数，见表1-1。

二进制数转换成十六进制数时，以小数点为界，整数部分自右向左，从低位向高位每4位二进制数分成一组，最后一组不足4位时，左边高位用0补足；小数部分则自左向右，从高位向低位每4位一组，最后一组不足4位时，右边低位用0补足，然后用1位对应的十六进制数表示。例如

$$
\begin{aligned}
(1011101.1010011)_2 &= (01011101.10100110)_2 \\
&= (5D.A6)_{16}
\end{aligned}
$$

十六进制数转换为二进制数时，将1位十六进制数用对应的4位二进制数表示，依次排列，然后去掉整数部分最高位的0和小数部分最低位的0，例如

$$
\begin{aligned}
(7BC.4)_{16} &= (011110111100.0100)_2 \\
&= (11110111100.01)_2
\end{aligned}
$$

2. 二进制数与十进制数间的转换

(1) 二进制数转换成十进制数。将二进制数按二进制的权展开，然后把所有各项的数值按十进制数相加，就可以得到相应的十进制数。例如

$$
\begin{aligned}
(11101.0101)_2 &= 2^4 + 2^3 + + 2^2 + 2^0 + 2^{-2} + 2^{-4} \\
&= 16 + 8 + 4 + 1 + \frac{1}{4} + \frac{1}{16} \\
&= (29.3125)_{10} \\
&\approx (29.31)_{10}
\end{aligned}
$$

各位二进制数的权见表1-2。

表1-2 各位二进制数的权

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2^n	1	2	4	8	16	32	64	128	256	512	1024	2048	4096	8192	16384	32768

小数部分转换时应注意精度。上例最低位为1/16，转换为十进制数后只需保留到百分位。当m较大时，按权相加进行转换很繁琐，可以采用将小数部分$x \times 2^m$变为整数进行转换后再除以2^m。如上例小数部分$(0.0101)_2 \times 2^4 \div 2^4 = (101)_2 \div 2^4 = 5 \div 16 \approx 0.31$。

(2) 十进制数转换为二进制数。需将十进制数的整数部分和小数部分分别进行转换。

1) 整数部分。除2取余，第一次除以2所得的余数是转换后所得的二进制数的最低位，第二次除以2所得的余数是转换后所得的二进制数的倒数第二位，……，以此类推，最后一次除以2且商为零时所得的余数是二进制的最高位。由最高位向最低位排列所得的余数即为应得的二进制数。

例如，将$(35)_{10}$转换为二进制数，即由

2 |35 ……余1……最低位 b_0

2 |17 ……余1…… b_1

2 |8 ……余0…… b_2

2 |4 ……余0…… b_3

2 |2 ……余0…… b_4

2 |1 ……余1……最高位 b_5

0

可得转换结果为 $(35)_{10} = (100011)_2$。

在十进制数转换为二进制数的过程中，也可以采用十六进数作为中间过渡。

例如，将 $(725)_{10}$转换为二进制数，即由

16 |725 ……余5 ……最低位 b_0

16 |45 ……余13…… b_1

16 |2 ……余2 ……最高位 b_2

0

可得转换结果为 $(725)_{10} = (2D5)_{16} = (1011010101)_2$

2）小数部分。先乘 2 取整，即把要转换的十进制小数乘以二进制的基数 2，积的整数部分取出作为小数点后二进制的最高位。然后继续将取整后积的小数部分乘以 2，再取积的整数部分作为二进制小数的次高位，……，这样依次相乘，直至积的小数部分为 0 或达到所需精度为止。最后，所得整数部分按先高位后低位顺序排列，即为二进制小数部分的结果。

例如，将 $(0.8125)_{10}$转换为二进制数，即由

0.8125 整数部分

× 2

1.6250 ……1…… 最高位 b_{-1}

0.6250

× 2

1.2500 ……1…… b_{-2}

0.2500

× 2

0.5000 ……0…… b_{-3}

0.5000

× 2

1.0000 ……1…… 最低位 b_{-4}

可得转换结果为 $(0.8125)_{10}=(0.1101)_2$

1.2.3 码制

用二进制表示文字、符号等信息的过程叫做编码。编码之后的二进制数称为二进制代码。代码的内涵因被编码对象的不同而有所差别。生活中的身份证号、邮政编码、火车车次、飞机航班、学生学号等都是编码的应用。不同的是，在实际生活中，人们常采用十进制进行代码；而在数字系统中，则采用二进制进行编码。

二进制编码十进制（Binary Coded Decimal，BCD）是用二进制代码的形式表示十进制数。十进制数有 10 个数字符号，需用 4 位二进制码表示。4 位二进制码有 16 种组合，而表示十进制数只需要 10 种组合，因此用 4 位二进制码表示十进制数有多种选取方式。表 1-3 列出了常用的 BCD 码。

表 1-3 常用的 BCD 码

十进制数	8421BCD 码	余 3 码	格雷码
0	0000	0011	0000
1	0001	0100	0001
2	0010	0101	0011
3	0011	0110	0010
4	0100	0111	0110
5	0101	1000	1110
6	0110	1001	1010
7	0111	1010	1000
8	1000	1011	1100
9	1001	1100	0100

例如，将 $(725)_{10}$ 表示成 8421BCD 码，即为 $(011100100101)_{8421BCD}$。

8421BCD 码用 4 位二进制码表示 1 位十进制数，每位二进制码都有固定的位权，所以这种代码称为有权码，8421BCD 码也由此得名。显然，在 8421BCD 码中 1010～1111 这 6 个代码不存在，被称为“伪码”。

余 3 码的编码规律是：余 3 码为 8421BCD 码加上二进制的 011（二进制的 011 对应于十进制的 3）对应的十进制数多 3。例如，$(0100)_{8421BCD}=(0111)_{余3}$。

从表 1-3 中还可以看出，格雷码在任意两组相邻代码之间只有一位不同，而且整个 4 位二进制代码的首、尾之间也只相差 1 位二进制码，所以格雷码又称循环码。格雷码在译码时不会发生第 3 章中所述的竞争冒险现象，因而常用于模拟量与数字量的转换。当模拟量发生微小变化而可能引起数字量发生变化时，格雷码仅改变一位，与其他代码同时改变两位或多位的情况相比更为可靠，即可减小出错的可能性。

1.3 逻辑函数的基本概念、公式和定理

1.3.1 3 种最基本的逻辑运算

与、或、非是 3 种最基本的逻辑运算。图 1-2 给出了 3 种指示灯控制电路。如果把开关是否闭合作为条件（逻辑变量），把灯的亮灭作为结果（逻辑函数），那么这 3 个电路表示

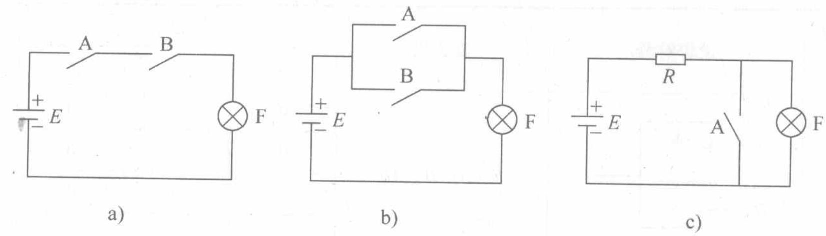


图 1-2　3 种指示灯控制电路

a）与电路　b）或电路　c）非电路

了与、或、非 3 种不同的因果逻辑关系。

1. 与运算　与运算的逻辑关系是：只有决定事物结果的全部条件同时具备时，结果才会发生。如果以 1 表示图 1-2 中开关闭合或灯亮，以 0 表示开关断开或灯灭，那么图 1-2a 表明，只有逻辑变量 A 和 B 同时为 1 时，逻辑函数的输出 F 才为 1。与的运算符是“·”，书写时可省略。实现逻辑与的电路称为与门，其逻辑表达式为 $F=AB$，与门可以有多个输入变量，如 $F=ABC\cdots$。逻辑符号见表 1-4。

2. 或运算　或运算的逻辑关系是：在决定事物结果的诸多条件中只要有一个满足，结果就会发生。如图 1-2b 中逻辑变量 A 或 B 任一为 1 时，逻辑函数的输出 F 即为 1。或的运算符是“+”。实现逻辑或的电路称为或门，其逻辑表达式为 $F=A+B$，或门也可以有多个输入变量，如 $F=A+B+C+D+\cdots$。逻辑符号见表 1-4。

3. 非运算　非运算的逻辑关系是：条件具备时，结果不发生；条件不具备时，则结果一定发生。如图 1-2c 中逻辑变量 A 为 1 时，逻辑函数的输出 F 为 0；逻辑变量 A 为 0 时，逻辑函数的输出 F 则为 1。非运算的运算符是逻辑变量上方有符号“ - ”，如 $\overline{A}$，称为“A 非”。实现逻辑非的电路称为非门，其逻辑表达式为 $F=\overline{A}$。逻辑符号见表 1-4。

1.3.2　复合逻辑运算

实际的逻辑问题往往比较复杂，不过它们都可以在与、或、非 3 种基本逻辑运算的基础上加以组合来实现。最常见的复合逻辑运算有与非、或非、与或非、异或、同或等。表 1-4 列出了几种基本的逻辑运算，其中后 5 种是与、或、非运算的组合形式。

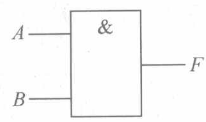
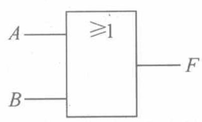
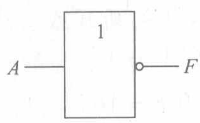
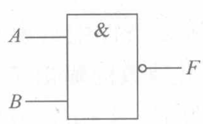
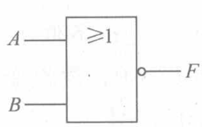
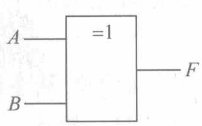
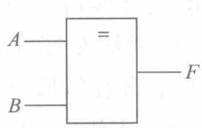
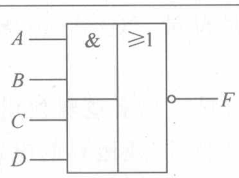
1. 与非、或非运算　由表 1-4 的相应真值表看出：与非运算是先与后非；或非运算是先或后非。实现与非、或非运算的电路为与非门、或非门，其逻辑符号见表 1-4。

2. 异或、同或运算　异或运算的逻辑关系为 $F=A\oplus B=\overline{A}B+A\overline{B}$。可见，只有逻辑变量 A 和 B 的取值不同时，逻辑函数的输出才为 1。异或的运算符是“$\oplus$”。实现异或运算的电路称为异或门，其逻辑符号见表 1-4。

同或运算的逻辑关系为 $F=A\odot B=\overline{A}\,\overline{B}+AB=\overline{A\oplus B}$。可见，只有逻辑变量 A 和 B 的取值相同时，逻辑函数的输出才为 1。同或的运算符是“$\odot$”。实现同或运算的电路称为同或门，其逻辑符号见表 1-4。对于两个输入变量的同或门和异或门，由真值表可以推出同或等于异或的非。

3. 与或非运算　与或非运算的顺序为先与再或最后取非。实现与或非运算的电路称为与或非门，其逻辑符号见表 1-4。

表 1-4 几种基本的逻辑运算

表示方法 / 逻辑运算	逻辑符号	表达式	真值表 A	真值表 B	真值表 F	基本运算规则
与		$F=A\cdot B=AB$	0	0	0	$A\cdot A=A$
			0	1	0	$A\cdot\overline{A}=0$
			1	0	0	$A\cdot 1=A$
			1	1	1	$A\cdot 0=0$
或		$F=A+B$	0	0	0	$A+A=A$
			0	1	1	$A+\overline{A}=1$
			1	0	1	$A+1=1$
			1	1	1	$A+0=A$
非		$F=\overline{A}$	0	—	1	$\overline{\overline{A}}=A$
			1	—	0	
与非		$F=\overline{AB}$	0	0	1	$\overline{AB}=\overline{A}+\overline{B}$
			0	1	1	
			1	0	1	
			1	1	0	
或非		$F=\overline{A+B}$	0	0	1	$\overline{A+B}=\overline{A}\,\overline{B}$
			0	1	0	
			1	0	0	
			1	1	0	
异或		$F=A\oplus B$	0	0	0	$A\oplus 0=A$
			0	1	1	$A\oplus 1=\overline{A}$
			1	0	1	$A\oplus A=0$
			1	1	0	$A\oplus\overline{A}=1$
同或		$F=A\odot B$	0	0	1	$A\odot 0=\overline{A}$
			0	1	0	$A\odot 1=A$
			1	0	0	$A\odot A=1$
			1	1	1	$A\odot\overline{A}=0$
与或非		$F=\overline{AB+CD}$	略			

1.3.3 基本公式、定理和常用规则

1. 逻辑代数基本公式　逻辑代数中的基本公式有以下 9 个：

1）与运算：$A\cdot 0=0$；$A\cdot 1=A$；$A\cdot A=A$；$A\cdot\overline{A}=0$

2）或运算：$A+0=A$；$A+1=1$；$A+A=A$；$A+\overline{A}=1$

非运算：$\overline{\overline{A}}=A$

2. 逻辑代数常用公式　逻辑代数常用公式较多，这里仅列出以下4个：

1）$A+AB=A$

2）$A+\overline{A}B=A+B$

3）$AB+\overline{A}B=B$

4）$AB+\overline{A}C+BC=AB+\overline{A}C$

3. 逻辑代数定律

1）交换律：$A\cdot B=B\cdot A$；$A+B=B+A$

2）结合律：$A\cdot(B\cdot C)=(A\cdot B)\cdot C$；$A+(B+C)=(A+B)+C$

3）分配律：$A\cdot(B+C)=AB+AC$；$A+BC=(A+B)(A+C)$

4）反演律（摩根公式）：$\overline{A+B}=\overline{A}\cdot\overline{B}$；$\overline{A\cdot B}=\overline{A}+\overline{B}$

4. 代入定理　在任何一个包含变量A的逻辑等式中，若以另一个逻辑式代换原式中所有A，所得等式仍然成立，反之亦然。

以上公式和定律都可以得到证明，有一种最直接的证明方法就是将变量的可能取值组合一一带入验证。而一旦证明了某几个公式，就可以依据这几个已经证明了的公式推导出其他全部公式。

上述公式和定律在数字电路分析中主要用于逻辑运算及逻辑函数式的变换和化简。

例1　已知$Y=\overline{A}B+A\overline{B}$，求$\overline{Y}$。

解　根据代入定理，可将$\overline{A}B$、$A\overline{B}$分别代换反演律中的两个变量，再利用反演律及其他有关公式，逐步演算、变换：

$$\begin{aligned}\overline{Y}&=\overline{\overline{A}B+A\overline{B}}\\&=\overline{\overline{A}B}\cdot\overline{A\overline{B}}\\&=(\overline{\overline{A}}+\overline{B})\cdot(\overline{A}+\overline{\overline{B}})\\&=(A+\overline{B})\cdot(\overline{A}+B)\\&=\overline{A}A+\overline{A}\,\overline{B}+AB+B\overline{B}\\&=0+\overline{A}\,\overline{B}+AB+0\\&=\overline{A}\,\overline{B}+AB\end{aligned}$$

由此可见，偶数个变量的异或运算的非便是同或运算，异或与同或互补。

用Multisim7仿真验证：此题的仿真电路及仿真结果如图1-3所示。

例2　利用公式化简下列逻辑式：1）$Y_1=A\overline{B}+A\overline{C}+BC$；2）$Y_2=A\overline{B}+ACD+\overline{A}\,\overline{B}+\overline{A}CD$。

解　1）

$$\begin{aligned}Y_1&=A\overline{B}+A\overline{C}+BC\\&=A(\overline{B}+\overline{C})+BC\\&=A\overline{BC}+BC\\&=A+BC\end{aligned}$$

2）

$$\begin{aligned}Y_2&=A\overline{B}+ACD+\overline{A}\,\overline{B}+\overline{A}CD\\&=A(\overline{B}+CD)+\overline{A}(\overline{B}+CD)\end{aligned}$$

$$= (\overline{B} + CD)(A + \overline{A})$$
$$= \overline{B} + CD$$

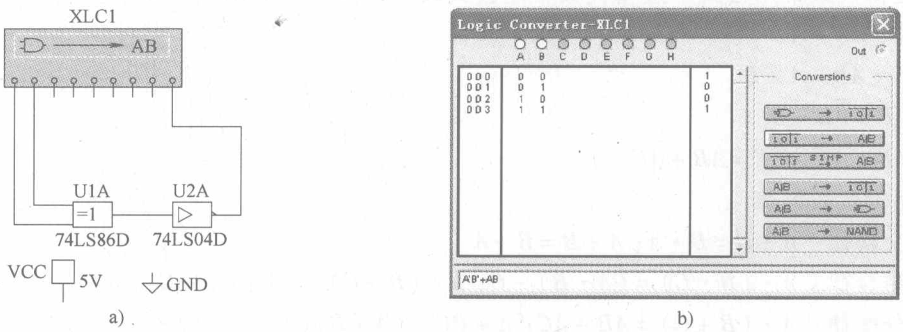


图 1-3　例 1 的仿真电路及仿真结果

a）仿真电路　b）仿真结果

用 Multisim7 仿真验证：

首先调出逻辑转换仪，双击打开转换窗口，并在其中输入 $A\overline{B}+ACD+\overline{A}\,\overline{B}+\overline{A}CD$（在 Multisim7 中，某逻辑自变量的非用自变量右上角加“′”来表示。如 $\overline{B}$ 用 B'表示），单击逻辑式转换成真值表功能的按钮，将逻辑式转换成真值表。然后再单击真值表转换成化简后的逻辑式的按钮，得到了最简逻辑式，如图 1-4 所示。

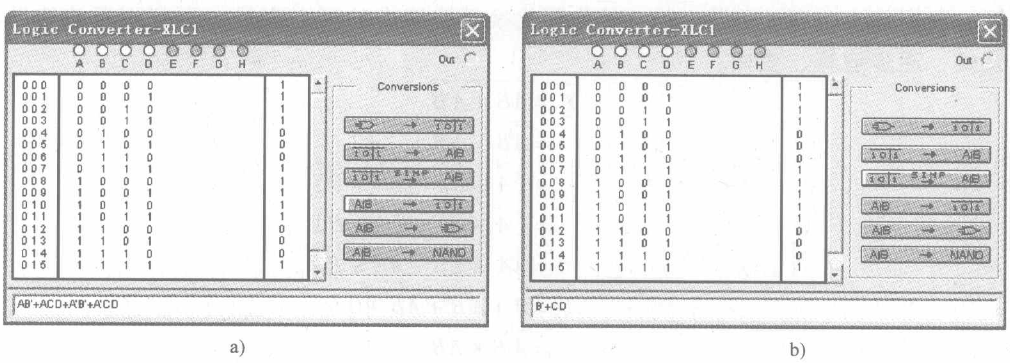


图 1-4　例 2 的仿真结果

a）由逻辑式转换成真值表　b）由真值表转换成最简逻辑式

1.4　逻辑函数的几种表示方法及相互转换

常用逻辑函数的表示方法有真值表、逻辑函数表达式、卡诺图和逻辑图、波形图。逻辑函数的几种表示方法可以相互转换。

1.4.1　由真值表求逻辑函数表达式和逻辑图

1. 真值表　真值表是由逻辑变量所有可能取值的组合及其对应函数值所构成的表格。

例 3　某产品有 3 项指标，当两项及以上指标达到标准时，则此产品为合格产品，其他情况均为不合格。画出筛选此产品为合格产品的逻辑电路。

解　设此产品的 3 项指标为 A、B、C，达标为 1，不达标为 0。设 F 为筛选结果，$F=1$