“十三五”普通高等教育规划教材

SHUZI DIANZI JISHU JICHU

数字电子技术基础

张志恒　主编

中国电力出版社
CHINA ELECTRIC POWER PRESS

"十三五"普通高等教育规划教材

数字电子技术基础

主　编　张志恒
副主编　赵世平
编　写　向远红
主　审　夏路易

中国电力出版社
CHINA ELECTRIC POWER PRESS

内 容 提 要

本书为“十三五”普通高等教育规划教材。

全书共分九章，主要内容包括数字逻辑基础、逻辑门电路、组合逻辑电路、集成触发器、时序逻辑电路、脉冲波形的产生与整形、半导体存储器、可编程逻辑器件、数/模和模/数转换器。通过本书的学习，学生能在规定的学时内掌握具有实用价值的数字电子技术的基本内容。

本书主要作为普通高等院校电气信息类、自动化类等相关专业的专业教材，也可作为高职高专教材，同时还可供从事电子技术工作的工程技术人员参考。

图书在版编目（CIP）数据

数字电子技术基础/张志恒主编．—北京：中国电力出版社，2017.4
“十三五”普通高等教育规划教材
ISBN 978-7-5198-0387-2

Ⅰ.①数…　Ⅱ.①张…　Ⅲ.①数字电路—电子技术—高等学校—教材　Ⅳ.①TN79

中国版本图书馆 CIP 数据核字（2017）第 028678 号

出版发行：中国电力出版社
地　　址：北京市东城区北京站西街 19 号（邮政编码 100005）
网　　址：http://www.cepp.sgcc.com.cn
责任编辑：乔　莉　牛梦洁　（010－63412535）
责任校对：常燕昆
装帧设计：郝晓燕　赵姗杉
责任印制：吴　迪

印　　刷：北京雁林吉兆印刷有限公司
版　　次：2017 年 4 月第一版
印　　次：2017 年 4 月北京第一次印刷
开　　本：787 毫米×1092 毫米　16 开本
印　　张：18.25
字　　数：440 千字
定　　价：42.00 元

前　言

数字电子技术基础课程是电子信息类、电气信息类、自动化类等相关专业的主要专业基础课程。国家教委曾多次组织重点院校的专家教授们编写出版过多本统编教材，对该课程的发展起到了重要的推动作用。

随着电子科学技术的高速发展，近年来数字电子技术基础课程的教学内容有了较大变化，其中基于 EDA 技术和可编程逻辑器件的现代数字系统设计得到了广泛应用。但由于可编程逻辑器件等新型器件仍属于半导体器件，所以过去讲授的半导体器件工作原理的理论基础对这些新型器件仍然适用。同时，传统教材中的逻辑代数、逻辑门、触发器、组合电路、时序电路等基本概念、分析方法、设计方法也是使用新型器件时必备的基础理论。因此，本书的对应章节一方面延续和保持了数字电路基础内容的完整性和理论的系统性，另一方面增加了数字电路基本内容的 VHDL 语言描述，使读者能够在学习数字逻辑单元电路时逐步掌握现代数字系统设计的基础知识。此外，本书在可编程逻辑器件一章，重点介绍了 FPLA、PAL 器件及其应用，GAL、CPLD 和 FPGA 的电路结构、工作原理和器件技术特性，并详细介绍了可编程逻辑器件的配置和基于 EDA 工具的现代数字系统设计流程。

本书由山西大学张志恒主编，太原理工大学夏路易教授主审。本书分为九章，第一、二、九章由山西大学张志恒编写，第三、六、七章由山西大学向远红编写，第四、五、八章及全书的 VHDL 内容由山西大学赵世平编写。全书由张志恒统稿。在本书的编写过程中参考了 Altera 公司的相关技术资料，所列举的程序代码均经过编译仿真检验。本书在编写过程中参考了一些已经出版的经典教材和文献，在此谨向作者们表示衷心的感谢。借此机会也向所有关心、支持和帮助过本书编写、修改、出版、发行工作的同志致以诚挚的谢意。

由于作者水平有限，书中一定还有许多不完善之处，殷切期望读者给予批评指正。

编　者

2016 年 12 月

目 录

第一章 数字逻辑基础

内容提要

随着时代的发展，信息时代已经真正到来，“数字”这两个字在我们的信息交流中越来越重要，数字通信、数字电视、数字控制等数字化技术已经成为当今电子技术的发展潮流。数字电路是数字电子技术的核心，是计算机及数字通信技术的核心基础。本章主要介绍描述数字电路逻辑功能的数学方法。首先介绍数字电路中常用的计数体制和几种常用的编码，然后介绍逻辑代数的基本运算规则和逻辑函数的表示方法，最后着重讲述了逻辑函数的代数化简法和卡诺图化简法。

第一节 概 述

一、信号的分类

电子电路中的信号可分为两类：一类是在时间和数值上连续变化的信号，如表示温度、湿度、压力、速度等物理量的信号，这一类信号称为模拟信号。用以产生、传递和处理模拟信号的电路称为模拟电路。另一类信号是在时间和大小上都是离散的信号，其大小变化是某个最小量的整数倍，如自动生产线上输出的零件数目等，这一类物理量称为数字量，把表示数字量的信号称为数字信号。用以产生、传递和处理数字信号的电路称为数字电路。

数字电路又称为逻辑电路，研究信号输入、输出之间逻辑关系。与模拟电路相比较，它具有以下特点：

(1) 模拟电路注重研究信号的放大、相位关系，波形失真情况等，采用的晶体管一般工作在放大状态。数字电路注重研究信号输入、输出之间的逻辑关系，采用的晶体管一般工作在截止或饱和状态（开关状态）。

(2) 数字电路分析和设计的数学工具是逻辑代数。逻辑代数以二进制为基础，其变量称为逻辑变量，用英文字母 A、B、C 等表示。逻辑变量取值（逻辑值）只有 1 和 0 两种，这里的 1 和 0 并不表示具体数值大小，而是表示逻辑变量的两种相反的状态，如是和否、真和假。

二、计数体制

表示“数”的计数体制很多，最常用的是进位计数制，进位计数制只用几个“数码”就能将任意大小的数表示出来。在数字系统中最常用的进位计数制是十进制、二进制、八进制和十六进制。

1. 十进制

十进制是一种进位计数制，它只用 10 个数码 0、1、2、3、4、5、6、7、8、9 就能将任意大小的数表示出来，计数规律是以“10”为基数，“逢十进一”，超过 9 时要用多位数表示，其中低位数和相邻高位数之间的进位关系是“逢十进一，借一做十”。例如 2638.63，它可以表示为

$$2638.63 = 2\times10^3 + 6\times10^2 + 3\times10^1 + 8\times10^0 + 6\times10^{-1} + 3\times10^{-2}$$

等号左边的形式称为十进制数的标位计数法，或称为并列表示法。2、6、3、8、6、3是等号右边形式中千位、百位、十位、个位、十分位、百分位的系数。由此可见处于不同位置上的数字符号具有不同的意义，或者说有着不同的权，即乘数 10^3、10^2、10^1、10^0、10^{-1}、10^{-2} 是十进制数2638.63各位的“权”。乘上权的系数称为加权系数，十进制数的数值就是各加权系数之和。因此，称上式等号右边形式为按权展开式。十进制数中有一个基本特征数“10”，它表征了该进位计数制所具有的数码个数及进位规则，称为十进制的“基数”。

基数和权是各种进位制的两个要素，正确理解其含意，便可掌握进位计数制的全部内容。对于一个任意大小的十进制数，可以表示为

$$(D)_{10} = \sum_{i=-m}^{n-1} a_i \times 10^i \tag{1-1}$$

式中：a_i 为第 i 位的系数；10^i 为第 i 位的权；m、n 为正整数，m 表示小数部分的位数，n 表示整数部分的位数；下角标10表示括号里的数是十进制数。

式（1-1）称为任意十进制数的按权展开式。

若以 N 代替式（1-1）中的10，就可得到任意进制（N 进制）数的按权展开式为

$$(D)_N = \sum_{i=-m}^{n-1} a_i \times N^i \tag{1-2}$$

式中：N 为计数制的基数；a_i 为第 i 位的系数；N^i 为第 i 位的权。

2. 二进制

二进制是以“2”为基数，“逢二进一”的进位计数制。二进制数中每一位仅有0或1两个可能的数码。任何一个二进制数均可展开为

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

式中：a_i 为基数“2”的第 i 次幂的系数。

例如

$$(1011.101)_2 = 1\times2^3 + 0\times2^2 + 1\times2^1 + 1\times2^0 + 1\times2^{-1} + 0\times2^{-2} + 1\times2^{-3}$$

3. 八进制

八进制是以“8”为基数，“逢八进一”的进位计数制。在八进制数中，每一位可以用0～7这8个数码中的任何一个表示。任何一个八进制数均可展开为

$$(D)_8 = \sum_{i=-m}^{n-1} a_i \times 8^i \tag{1-4}$$

式中：a_i 为基数“8”的第 i 次幂的系数，它可以是0～7这8个数码中的任何一个。例如

$$(2647.356)_8 = 2\times8^3 + 6\times8^2 + 4\times8^1 + 7\times8^0 + 3\times8^{-1} + 5\times8^{-2} + 6\times8^{-3}$$

4. 十六进制

十六进制是以“16”为基数，“逢十六进一”的进位计数制。在十六进制数中，每一位可以用0～9、A（10）、B（11）、C（12）、D（13）、E（14）、F（15）这16个数码中的任何一个表示。任何一个十六进制数均可展开为

$$(D)_{16} = \sum_{i=-m}^{n-1} a_i \times 16^i \tag{1-5}$$

式中：a_i 为基数“16”的第 i 次幂的系数。

例如

$$(3E5D.2C8)_{16} = 3\times16^{3}+E\times16^{2}+5\times16^{1}+D\times16^{0}+2\times16^{-1}+C\times16^{-2}+8\times16^{-3}$$

三、各种进制数之间的互相转换

1. 二、八、十六进制数转换为十进制数

不同进制只是描述数值的方式不同，它们是可以互相转换的，转换的前提是保证转换前后数值大小相等。

要将一个 N 进制数转换为一个十进制数，只要将该 N 进制数按式（1 - 2）展开，然后把各项数值按十进制数的计算规律相加，就可以得到等值的十进制数了。例如

$$\begin{aligned}(1011.11)_{2} &= 1\times2^{3}+0\times2^{2}+1\times2^{1}+1\times2^{0}+1\times2^{-1}+1\times2^{-2}\\ &= 8+2+1+0.5+0.25=(11.75)_{10}\\ (365.17)_{8} &= 3\times8^{2}+6\times8^{1}+5\times8^{0}+1\times8^{-1}+7\times8^{-2}\\ &= 192+48+5+0.125+0.109375=(245.234375)_{10}\\ (5D.41)_{16} &= 5\times16^{1}+D\times16^{0}+4\times16^{-1}+1\times16^{-2}\\ &= 80+13+0.25+0.00390625=(93.25390625)_{10}\end{aligned}$$

2. 十进制数转换为二、八、十六进制数

十进制数转换为 N 进制数，需对整数和小数部分分别进行转换。

（1）整数部分的转换。假设十进制整数为 $(D)_{10}$，它所对应的任意 N 进制数为 $(a_{n}a_{n-1}\cdots a_{1}a_{0})_{N}$，则有

$$(D)_{10}=a_{n}N^{n}+a_{n-1}N^{n-1}+\cdots+a_{1}N^{1}+a_{0}=N(a_{n}N^{n-1}+a_{n-1}N^{n-2}+\cdots+a_{1})+a_{0} \tag{1-6}$$

将式（1 - 6）两边同除以 N，那么两边的商和余数必然对应相等，所得的商为 $(a_{n}N^{n-1}+a_{n-1}N^{n-2}+\cdots+a_{1})$，所得的余数就是 a_0。

同理，这个商又可以写为

$$\frac{(D)_{10}-a_{0}}{N}=N(a_{n}N^{n-2}+a_{n-1}N^{n-3}+\cdots+a_{2})+a_{1} \tag{1-7}$$

将式（1 - 7）两边再除以 N，则所得之余数即为 a_1。

依此类推，反复将每次得到的商再除以 N，直至最后商为 0，便可以求出对应于任意 N 进制数的每一位系数。

［例 1 - 1］ 将 $(41)_{10}$ 转换为二进制数和八进制数。

解 1）

$$\begin{array}{r|l} 2 & 41 \quad \cdots\cdots 1 \quad a_0 \quad \text{低位}\\ \hline 2 & 20 \quad \cdots\cdots 0 \quad a_1\\ \hline 2 & 10 \quad \cdots\cdots 0 \quad a_2\\ \hline 2 & 5 \quad \cdots\cdots 1 \quad a_3\\ \hline 2 & 2 \quad \cdots\cdots 0 \quad a_4\\ \hline & 1 \quad \cdots\cdots 1 \quad a_5 \quad \text{高位}\end{array}$$

于是得到 $(41)_{10}=(101001)_{2}$

2）

$$\begin{array}{r|l} 8 & 41 \quad \cdots\cdots 1 \quad a_0 \quad \text{低位}\\ \hline 8 & 5 \quad \cdots\cdots 5 \quad a_1 \quad \text{高位}\\ \hline \end{array}$$

于是得到 $(41)_{10}=(51)_8$

[例 1-2] 将 $(2803)_{10}$转换为十六进制数。

解

$$\begin{array}{r|l}
16 & 2803 \quad \cdots\cdots \text{余数}\ (3)_{10}=(3)_{16} \quad a_0 \quad \text{低位} \\
16 & 175 \quad \cdots\cdots \text{余数}\ (15)_{10}=(\mathrm{F})_{16} \quad a_1 \\
16 & 10 \quad \cdots\cdots \text{余数}\ (10)_{10}=(\mathrm{A})_{16} \quad a_2 \quad \text{高位} \\
 & 0
\end{array}$$

于是得到 $(2803)_{10}=(\mathrm{AF3})_{16}$

(2) 小数部分的转换。假设十进制小数为 $(D)_{10}$，对应的任意 N 进制数为 $(0.a_{-1}a_{-2}\cdots a_{-m+1}a_{-m})_N$，则有

$$(D)_{10}=a_{-1}N^{-1}+a_{-2}N^{-2}+a_{-3}N^{-3}+\cdots+a_{-m}N^{-m} \tag{1-8}$$

将式 (1-8) 两边同乘以 N，得

$$N(D)_{10}=a_{-1}+(a_{-2}N^{-1}+a_{-3}N^{-2}+\cdots+a_{-m}N^{-m+1}) \tag{1-9}$$

可以看出用 N 乘以 $(D)_{10}$所得乘积的整数部分就是 a_{-1}。乘积的小数部分又可写为

$$N(D)_{10}-a_{-1}=a_{-2}N^{-1}+a_{-3}N^{-2}+\cdots+a_{-m}N^{-m+1} \tag{1-10}$$

将式 (1-10) 两边再乘以 N，得

$$N[N(D)_{10}-a_{-1}]=a_{-2}+a_{-3}N^{-1}+a_{-4}N^{-2}+\cdots+a_{-m}N^{-m+2} \tag{1-11}$$

所得的乘积的整数部分就是 a_{-2}。

依此类推，将每次乘以 N 后所得乘积的小数部分再乘以 N，直至最后乘积的小数部分为 0 或达到一定的精确度为止，便可求得任意 N 进制小数的每一位系数。

[例 1-3] 将 $(0.285)_{10}$转换为二进制数、八进制数和十六进制数，要求精确到小数点后 3 位。

解 1)

$$\begin{array}{r l}
 & 0.285 \\
\times & 2 \\ \hline
 & 0.570 \quad \cdots\cdots \quad \text{整数为 0，} a_{-1}=0 \\
\times & 2 \\ \hline
 & 1.140 \quad \cdots\cdots \quad \text{整数为 1，} a_{-2}=1 \\
- & 1 \\ \hline
 & 0.140 \\
\times & 2 \\ \hline
 & 0.280 \quad \cdots\cdots \quad \text{整数为 0，} a_{-3}=0
\end{array}$$

于是得到 $(0.285)_{10}\approx(0.010)_2$

2)

$$\begin{array}{r l}
 & 0.285 \\
\times & 8 \\ \hline
 & 2.280 \quad \cdots\cdots \quad \text{整数为 2，} a_{-1}=2 \\
- & 2 \\ \hline
 & 0.280 \\
\times & 8 \\ \hline
 & 2.240
\end{array}$$

$$\begin{array}{lr} & 2.240 \\ - & 2 \\ \hline & 0.240 \\ \times & 8 \\ \hline & 1.920 \end{array}$$

…… 整数为 2，$a_{-2}=2$

…… 整数为 1，$a_{-3}=1$

于是得到 $(0.285)_{10}\approx(0.221)_8$

3)

$$\begin{array}{lr} & 0.285 \\ \times & 16 \\ \hline & 4.560 \\ - & 4 \\ \hline & 0.560 \\ \times & 16 \\ \hline & 8.960 \\ - & 8 \\ \hline & 0.960 \\ \times & 16 \\ \hline & 15.360 \end{array}$$

…… 整数为 4，$a_{-1}=4$

…… 整数为 8，$a_{-2}=8$

…… 整数为 15，$a_{-3}=\mathrm{F}$

于是得到 $(0.285)_{10}\approx(0.48\mathrm{F})_{16}$

由［例 1-3］可以发现，在实现小数转换时，不一定能正好转换为有限位小数，因而必须考虑转换精确度问题，即根据需要来确定转换位数。

3. 二进制数与八进制数的相互转换

因为 3 位二进制数从（000）～（111）共有 8 个不同的状态，如果 111 再加 1 就成为 4 位二进制数 1000，且前 3 位回到 000，满足逢八进一的计数规律，所以 3 位二进制数恰好相当于 1 位八进制数。这样将二进制数转换为八进制数时，对整数部分，只需从最低位到高位每 3 位二进制数分为一组，高位不足 3 位时补 0，然后将每一组二进制数用一个等值的八进制数代替即可。小数部分从小数点后一位开始向后每 3 位二进制数分为一组，低位不足 3 位时补 0，然后将每一组二进制数用一个等值的八进制数代替即可。同理，八进制转换为二进制时，每一位八进制数只需用等值的 3 位二进制数代替即可。

［例 1-4］ 将 $(1100101.10101)_2$ 转换为八进制数，$(627.25)_8$ 转换为二进制数。

解 (1)

$$\begin{array}{cccccc} (001, & 100, & 101. & 101, & 010)_2 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ =(1 & 4 & 5. & 5 & 2)_8 \end{array}$$

(2)

$$\begin{array}{cccccc} (6 & 2 & 7. & 2 & 5)_8 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ =(110 & 010 & 111. & 010 & 101)_2 \end{array}$$

4. 二进制数与十六进制数之间的转换

因为 4 位二进制数从（0000）～（1111）共有 16 个不同的状态，如果 1111 再加 1 就成为 5 位二进制数 10000，且前 4 位回到 0000，满足逢十六进一的计数规律，所以 4 位二进制数恰好相当于 1 位十六进制数。这样将二进制数转换为十六进制数时，对整数部分，只需从最低位到高位每 4 位二进制数分为一组，高位不足 4 位时补 0 凑足 4 位，然后将每一组二进

制数用一个等值的十六进制数代替即可。小数部分从小数点后一位开始向后每4位二进制数分为一组，低位不足4位时补0凑足4位，然后将每一组二进制数用一个等值的十六进制数代替即可。同理，十六进制数变为二进制数时，每一位十六进制数只需用等值的4位二进制数代替即可

［例1-5］ 将$(1100011.10101)_2$转换为十六进制数，$(3D9.5C)_{16}$转换为二进制数。

解 (1) $(0110,\ 0011.1010,\ 1000)_2$

$$= (6\quad 3.\quad A\quad 8)_{16}$$

(2) $(3\quad D\quad 9.\ 5\quad C)_{16}$

$$= (0011\quad 1101\quad 1001.0101\quad 1100)_2$$

四、二进制码

数字系统中的信息包括数字、文字、符号等，它们都可以用多位二进制数码来表示。用二进制数码表示数字、文字、符号等信息的过程叫编码，用来进行编码的二进制数码称为二进制代码。如果需要编码的信息量为N，则用以编码的一组二进制代码所需位数n应满足$2^n \geqslant N$。例如：若信息量$N=45$，则编码所需的二进制代码位数$n=6$。

1. 二—十进制编码

凡是利用若干位二进制数码来表示1位十进制数码的方法称为二—十进制编码，简称为二—十进制码，即BCD（Binary-Coded Decimal）码。

1位十进制数有0～9这10个不同数码，需要用4位二进制数才能表示。4位二进制数码有$2^4=16$种不同的组合，因此，从16种组合状态中选用其中10种组合状态来表示1位十进制数0～9的编码方法很多，常用的二—十进制码有以下几种：

(1) 8421BCD码（简称8421码）。这种编码的4位二进制数码从高位至低位每位的权分别为8、4、2、1，故称为8421码，如表1-1所示，它是一种有权码。就0000～1001这10个二进制数而言，8421码和通常的4位二进制数没有区别，但需注意8421码中没有1010～1111这几个组合，这和通常的4位二进制数不同。8421码容易识别，转换也很方便，是广泛应用的一种编码。

对于有权码，应满足关系式

$$(D)_{10} = \sum_{i=0}^{3} a_i \times W_i \tag{1-12}$$

式中：a_i为i位的二进制数码（0或1）；W_i为i位的权。

例如8421码中的0110所代表的十进制数为

$$(0110)_{8421} = 0\times8+1\times4+1\times2+0\times1 = (6)_{10}$$

十进制数和8421码之间可直接按位转换，例如

$$(94.12)_{10} = (10010100.00010010)_{8421}$$

(2) 余3码。每个1位十进制数码用余3码表示时，比8421码多3（即多0011），故称为余3码。例如1位十进制数码$(5)_{10}$，用余3码表示为1000，而用8421码表示为0101，1000－0101＝0011。余3码是一种无权码。

常用的几种二—十进制编码如表1-1所示。

表 1-1　常用的几种二—十进制编码

十进制数 \ 编码种类	8421 码	2421A 码	2421B 码	5421 码	余 3 码
0	0000	0000	0000	0000	0011
1	0001	0001	0001	0001	0100
2	0010	0010	0010	0010	0101
3	0011	0011	0011	0011	0110
4	0100	0100	0100	0100	0111
5	0101	0101	1011	1000	1000
6	0110	0110	1100	1001	1001
7	0111	0111	1101	1010	1010
8	1000	1110	1110	1011	1011
9	1001	1111	1111	1100	1100
权	8421	2421	2421	5421	无

表 1-1 中还列出了 2421 码和 5421 码，它们都是有权码。2421 码和 5421 码的编码方案不是唯一的，例如 5421 码的 $(7)_{10}$，既可以用 1010 表示，也可以用 0111 表示；2421 码的 $(5)_{10}$，既可以用 0101 表示，也可以用 1011 表示。

从表 1-1 中可以看出，同一代码用不同的编码方法时，表示的意义不同。如表中 0100 代码，在 8421 码、2421A 码、2421B 码和 5421 码中代表 $(4)_{10}$，而在余 3 码中代表 $(1)_{10}$。

2. 格雷码（Gray 码）

格雷码的基本特点是任何两个代码之间仅有一位不同，因而又称为单位距离码。

格雷码属于无权码，它有多种编码形式，其中最常用的一种是循环码，表 1-2 所示为 4 位代码的循环码编码表。

表 1-2　4 位代码的循环码编码表

十进制数	G_3	G_2	G_1	G_0	十进制数	G_3	G_2	G_1	G_0
0	0	0	0	0	8	1	1	0	0
1	0	0	0	1	9	1	1	0	1
2	0	0	1	1	10	1	1	1	1
3	0	0	1	0	11	1	1	1	0
4	0	1	1	0	12	1	0	1	0
5	0	1	1	1	13	1	0	1	1
6	0	1	0	1	14	1	0	0	1
7	0	1	0	0	15	1	0	0	0

如表 1-2 所示，格雷码从一个代码变为相邻的另一个代码时，其中只有 1 位二进制数码变化，例如从 7 变到 8，即由 0100 变到 1100，只有最左 1 位发生变化，而 8421 码则由 0111 变为 1000，4 位都发生了变化。显然，采用格雷码可减少代码在进行变化时产生错误

的概率。

格雷码广泛用于输入、输出设备和模拟—数字转换器等。

第二节 逻 辑 代 数

在客观世界中，许多事物之间的关系具有因果性。例如，照明线路中开关与灯的关系，灯亮与灯灭取决于开关的闭合与断开。开关闭合与否是因，灯亮不亮是果，这种因果关系称为逻辑关系。逻辑，就是指“条件”与“结果”之间的因果关系。用以分析研究这种逻辑关系的数学工具就是逻辑代数，也称为布尔代数。

在日常生活中，许多事物都只有相互对立的两种不同状态，如开关的闭合与断开、灯的亮与灭、一件事情的真与假、电压的高与低、电流的有与无等。如果用 0 来代表其中的一种状态，用 1 来代表对立的另一种状态，这样就使在逻辑代数中，不论是代表“因”的自变量，还是代表“果”的因变量，都只有两种不同的取值可能，其值不是 1，就是 0，不可能有第三种取值。注意，这里 0 和 1 并不表示数值的大小，而是表示事物的两种对立状态，称为逻辑 0 状态和逻辑 1 状态。

一、逻辑代数中的三种基本运算

在逻辑代数中，最基本的逻辑运算有与、或、非三种运算。

1. 与运算

图 1 - 1 所示是一个简单电路，开关是否闭合与灯是否点亮之间的关系，就是一种“与”逻辑关系。图中用逻辑变量 A 和 B 分别表示两个开关，并用 1 和 0 分别表示开关处于“闭合”和“断开”状态。用逻辑变量 Y 表示灯，并用 1 和 0 分别表示灯“亮”和“灭”。如果将 A、B 逻辑变量的所有取值和与其一一对应的逻辑值 Y 之间的关系以表格的形式表示出来，如表 1 - 3 所示，则称为逻辑真值表，或简称为真值表。由表 1 - 3 不难看出，要想使灯“亮”这个结果发生，必须使它的两个条件“A”和“B”开关都闭合，或者说只有变量 A 和 B 都是 1 时，输出 Y 才为 1。因此，从这个电路可总结出这样的逻辑关系：当决定一件事情发生（灯亮）的各个条件（开关“A”、“B”闭合）全部具备时，这件事情才会发生。这种逻辑关系称为与逻辑。表示与逻辑的逻辑表达式为

$$Y = A \cdot B \tag{1-13}$$

式中：“·”为与运算符号，也表示逻辑“乘”，可省略不写。

式（1 - 13）读作 Y 等于 A 与 B。实现与运算的逻辑电路称为与门，逻辑符号如图 1 - 2 所示。

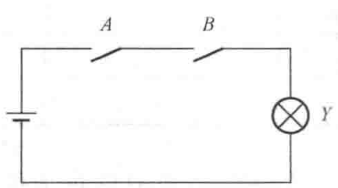

图 1 - 1 与逻辑电路图

表 1 - 3 与逻辑真值表

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

与运算可以推广到多个逻辑变量，即

$$Y = A \cdot B \cdot C \cdot \cdots \tag{1-14}$$

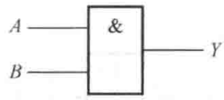

图 1-2　与运算的逻辑符号

2. 或运算

图 1-3 所示简单电路，开关是否闭合与灯是否点亮之间的关系，就是一种“或”逻辑关系。或逻辑真值表如表 1-4 所示。由表 1-4 不难看出，要想使灯“亮”这个结果发生，只要它的两个条件“A”和“B”中的一个开关闭合就可以，或者说只要变量 A 和 B 有一个是 1，输出 Y 就为 1。因此，从这个电路可总结出这样的逻辑关系：当决定一件事情发生（灯亮）的各个条件（开关“A”、“B”闭合）中只要有一个条件具备，这件事情就会发生。这种逻辑关系称为或逻辑。表示或逻辑的逻辑表达式为

$$Y = A + B \tag{1-15}$$

式中：“+”为或运算符号，也表示逻辑“加”。

式 (1-15) 读作 Y 等于 A 或 B。实现或运算的逻辑电路称为或门，逻辑符号如图 1-4 所示。

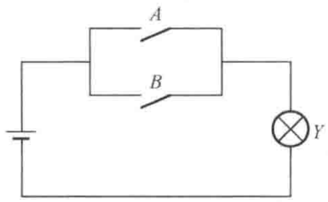

图 1-3　或逻辑电路图

表 1-4　　或逻辑真值表

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

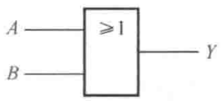

图 1-4　或运算的逻辑符号

或运算也可推广到多个逻辑变量，即

$$Y = A + B + C + \cdots \tag{1-16}$$

3. 非运算

图 1-5 所示简单电路，开关是否闭合与灯是否点亮之间的关系，就是一种“非”逻辑关系。非逻辑真值表如表 1-5 所示。由表 1-5 可以看出，要想使灯“亮”这个结果发生，必须使它的条件“A”开关不闭合，或者说只要变量 A 是 0，输出 Y 就为 1，若变量 A 是 1，则输出 Y 为 0。因此，从这个电路可总结出这样的逻辑关系：当决定一件事情发生（灯亮）的条件（开关“A”闭合）具备时，这件事情不会发生；而条件不具备时，这件事情就会发生。这种逻辑关系称为非逻辑。表示非逻辑的逻辑表达式为

$$Y = \overline{A} \tag{1-17}$$

式中：A 上的“—”为非运算符号。

式 (1-17) 读作 Y 等于 A 非。一般将 A 叫作原变量，$\overline{A}$ 叫作反变量。实现非运算的逻辑电路称为非门，非运算的逻辑符号如图 1-6 所示。

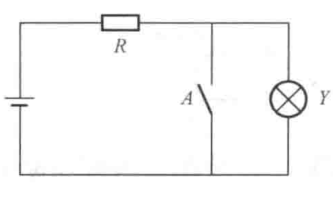

图 1-5　非逻辑电路图

表 1-5　　非逻辑真值表

A	Y
0	1
1	0

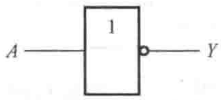

图 1-6 非运算的逻辑符号

二、逻辑代数中几种常用的复合运算

由与、或、非三种基本逻辑运算可以组合成若干常用的复合逻辑运算。

1. 与非运算

与非逻辑表达式为

$$Y=\overline{A\cdot B} \tag{1-18}$$

与非真值表如表 1-6 所示，实现该逻辑功能的门电路逻辑符号如图 1-7 所示。

表 1-6 与非真值表

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

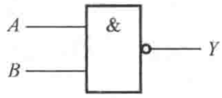

图 1-7 与非运算的逻辑符号

2. 或非运算

或非逻辑表达式为

$$Y=\overline{A+B} \tag{1-19}$$

或非真值表如表 1-7 所示，实现该逻辑功能的门电路逻辑符号如图 1-8 所示。

表 1-7 或非真值表

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

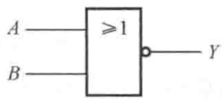

图 1-8 或非运算的逻辑符号

3. 异或运算

异或逻辑表达式为

$$Y=\overline{A}B+A\overline{B}=A\oplus B \tag{1-20}$$

式中：$\oplus$为异或运算符号。

式（1-20）读作 Y 等于 A 异或 B。当 A、B 逻辑状态相同时 $Y=0$；当 A、B 的状态不同时 $Y=1$。异或真值表如表 1-8 所示，实现该逻辑功能的门电路逻辑符号如图 1-9 所示。

表 1-8 异或真值表

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

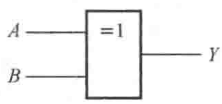

图 1-9 异或运算的逻辑符号