21世纪高等学校规划教材

SHUZI DIANZI JISHU

数字电子技术

李积英　主编

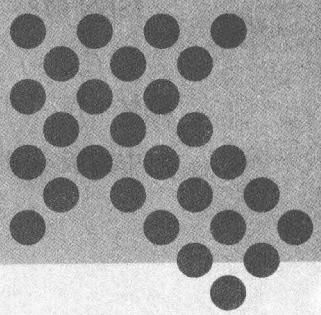
中国电力出版社
CHINA ELECTRIC POWER PRESS

21世纪高等学校规划教材

SHUZI DIANZI JISHU

数字电子技术

主编　李积英
编写　赵　贺　侯　越　曹　岩
主审　高普古



中国电力出版社
CHINA ELECTRIC POWER PRESS

内 容 提 要

本书为21世纪高等学校规划教材。

全书共分为9章，包括数字电路基础知识，集成逻辑门，组合逻辑电路，触发器，时序逻辑电路，脉冲波形的产生与变换，半导体存储器，可编程逻辑器件及数/模、模/数转换电路。本书在内容上强调基本概念和原理，重视电路的分析和设计方法，注意电路与工程应用相结合。

本书每章先综述所要介绍的内容，然后进行正文叙述，最后进行小结，每节后都有大量思考题与习题，以加深学生对所学知识的理解与运用。本书从课程教学目的出发，突出课程重点，突出基本分析和设计方法的介绍，注重方法的普遍性和实践性。

本书可作为高等学校电气类、电子信息类及其他相近专业教材，也可作为高职高专教材，同时还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/李积英主编. —北京：中国电力出版社，2011.11

21世纪高等学校规划教材

ISBN 978-7-5123-2299-8

Ⅰ.①数… Ⅱ.①李… 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（2011）第226806号

中国电力出版社出版、发行

（北京市东城区北京站西街19号 100005 http://www.cepp.sgcc.com.cn）

北京市同江印刷厂印刷

各地新华书店经售

*

2012年1月第一版 2012年1月北京第一次印刷

787毫米×1092毫米 16开本 17.75印张 433千字

定价 **31.00** 元

前　言

数字电子技术是一门重要的专业基础课程，其应用性很强，是电气信息类、电子信息类等专业的必修课。随着电子IC技术的不断发展和高等学校课程教学改革的不断深入，对数字电子技术课程的教学内容提出了更高的要求。为适应科学技术的发展以及社会对人才培养的要求，编者依据国家教委工科电工课程教学指导委员会审定通过的高等学校“电子线路”课程的基本要求，并结合多年的课堂教学实践和经验编写了本书。

本书在编写时注意了以下几点：①在内容的选取上，对原有教学内容进行了调整和充实，精简分立元件内容，增加集成电路内容，教学重点从逻辑电路分析转向面对问题的逻辑电路设计和集成电路芯片的应用；②在内容的次序安排上，从基本概念出发，引出基本电路、介绍分析方法和设计方法，最后通过具体实例加以总结和归纳，从而培养学生分析问题、解决问题的能力；③为了激发学生的学习兴趣，并与所学专业相结合，引入了工程应用方面的实例，如数字钟电路，555定时器的应用电路等。

本书共分为9章。第1章介绍了数字电子技术的基础知识，包括数制和码制、逻辑代数数学工具、逻辑函数的表示及其化简。第2章和第4章分别介绍了集成逻辑门电路和触发器，主要介绍基本数字器件集成逻辑门和触发器的外特性，对其内部结构和内部电路的分析计算作了许多精简。第3章和第5章分别介绍了组合逻辑电路和时序逻辑电路，这部分内容是数字电路逻辑设计的理论基础。除了介绍传统的分析方法和设计方法外，重点介绍了常见的各种中规模集成数字器件的基本功能和应用，以及以中规模器件为核心的组合逻辑电路和时序逻辑电路的分析和设计方法。第6章简要介绍了脉冲波形的产生和变换电路，重点介绍了555定时器及其应用电路，并增加了应用实例说明；删减了由分立元件构成的多谐振荡器、单稳态触发器和施密特触发器的内容。第7章和第8章介绍了半导体存储器和可编程逻辑器件；简要介绍了可编程逻辑器件的发展；分类及可编程逻辑器件的开发与测试技术。至于硬件描述语言等内容另有课程介绍，读者可参阅有关资料。第9章介绍了模/数和数/模转换电路。

本书由李积英主编，负责全书的组织、修改和定稿；曹岩编写第1、2章，李积英编写第3、4章，赵贺编写第5、6章，侯越编写第7、8、9章。

感谢清华大学高晋占教授百忙之中审阅了全书，并提出了许多宝贵的意见和建议。

感谢兰州交通大学电子与信息工程学院电子技术教研室各位老师的支持与帮助。

由于编者能力和水平有限，加之时间仓促，书中难免会有疏漏和不妥之处，恳请广大读者提出批评和改进意见。

编　者

2011年10月

目　录

第1章　数字电路基础知识

本章首先介绍数字信号、数字技术和数字系统等基本概念，然后介绍各种进制数的表示方法，最后介绍逻辑代数的基本概念、公式和定理，逻辑函数的代数化简法和卡诺图化简法。逻辑代数是分析及设计数字电路的基本工具，逻辑函数化简是数字电路分析及设计的基础。

1.1　概　　述

1.1.1　数字信号和模拟信号

工程上把电信号分为模拟信号和数字信号两大类。模拟信号是在时间和幅值上都连续变化的信号，例如温度、压力、磁场、电场等物理量通过传感器变成的电信号，模拟语音的音频信号和模拟图像的视频信号等，如图1.1.1（a）所示。对模拟信号进行传输、处理的电子线路称为模拟电路。数字信号是在时间和幅值上都离散的信号，通常是由逻辑0和1组成的信号，例如计算机中各部件之间传输的信息，如图1.1.1（b）所示。对数字信号进行传输、处理的电子线路称为数字电路，如数字电子钟电路、数字万用表电路等。

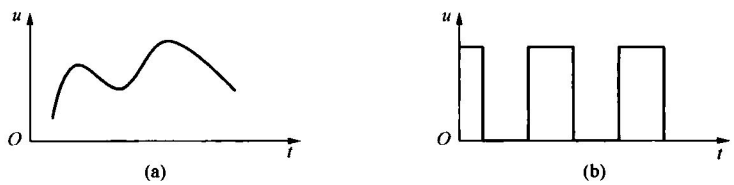


图1.1.1　模拟信号和数字信号的电压—时间波形
（a）模拟信号波形；（b）数字信号波形

1.1.2　数字技术和数字系统

数字信号和模拟信号之间可以相互转换。模拟信号经过取样、保持、量化和编码转换为数字信号的过程称为模数转换（A/D转换）。对模拟信号进行数字化处理时，只要取样率大于或等于信号最高频率的两倍，并有足够的二进制位数来表示每一个取样信号，就可以用序列的离散二进制数来表示模拟信号，并可以根据它的离散值恢复出原始的模拟信号。

数字技术就是为了适应和满足不同的应用需要，通过A/D转换电路将模拟信号转换成由0和1组成的数字信号的技术。所谓数字系统，是指输入和输出都是数字信号且具有存储、传输、处理信息能力的系统，微型计算机就是一个典型的数字系统。数字系统完成对数字信号的存储、运算、处理、变换和合成等。

随着数字技术的不断发展，用数字系统处理模拟信号将会越来越普遍，数字电路被广泛应用于数字电子计算机、数字通信系统、数字式仪表、数字控制装置及工业自动化系统等领域。数字系统具有如下几方面的优点。

（1）精度高。模拟系统的精度主要取决于电路中元件的精度，而元件的精度一般很难达到10^{-3}以上；数字系统的精度主要取决于表示信息的二进制的位数（即字长），数字系统17位字长就可达到10^{-5}的精度。

（2）可靠性强。数字系统只有两个电平信号“1”和“0”，受噪声和环境条件的影响小，不像模拟系统各参数易受温度、电磁感应、振动等环境条件的影响；另外，数字系统多采用大规模集成电路，其故障率远比采用众多分立元件构成的模拟系统低。

（3）应用范围广。数字系统不但适用于数值信息的处理，而且适用于非数值信息的处理；模拟系统只能处理数值信息。

（4）集成度高且成本低。由于数字电路主要工作在开关状态，对元件的参数要求不高，便于大规模集成和生产。随着微电子技术的发展，可以用更低的成本和更高的性能来开发较为复杂的数字系统。虽然当前模拟系统集成化的开发成本在不断的下降，性能也在不断增强，但由于数字器件的简单性，数字系统集成化的发展更为迅速。

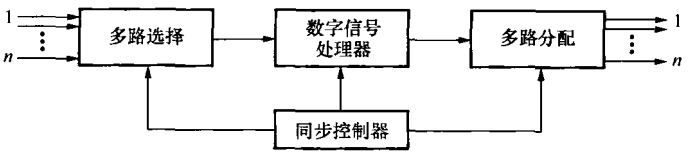


图 1.1.2 分时复用的数字系统

（5）使用效率高。数字系统的一个最大的优点是所谓的“分时复用”，即可利用同一数字信号处理器同时处理几个通道的信号。如图1.1.2所示，在“同步控制器”控制下，各路输入信号按先后顺序分别送入“数字信号处理器”进行处理，然后分别输出。处理器的运算速度越高，能同时处理的信道数也就越多。

1.1.3 数字逻辑和逻辑代数

数字电路与模拟电路之间，除了输入、输出处理的信号不同之外，还有一个主要区别就是输入和输出之间表达的关系不同。对模拟电路而言，输入和输出之间表达的是一种数值关系，而数字电路的输入和输出之间表达的是一种因果关系，即逻辑关系。因此，数字电路也称数字逻辑电路。

在数字电路中，输入和输出变量都是只有两种状态的逻辑变量。逻辑变量的两种状态分别是“真”和“假”，通常用数字1表示“真”，用数字0表示“假”。逻辑变量的取值只能在0和1中选择，而不能有第三种取值。数字电路中基本的逻辑运算有逻辑与、逻辑或和逻辑非，由这三种基本逻辑运算可以组成多种复合逻辑运算。

实现逻辑运算的电路，称为逻辑门。逻辑门是组成数字电路的最小单元。数字逻辑电路根据功能和结构特点不同，可分为组合逻辑电路和时序逻辑电路。组合逻辑电路完全是由逻辑门构成的，不包含存储器件；时序逻辑电路是包含存储器件的电路。数字逻辑电路的存储功能是由存储器件完成的，最基本的存储器件是触发器。在数字逻辑电路的实际应用中，通常既包括组合逻辑电路，也包括时序逻辑电路。

逻辑代数是分析和设计数字逻辑电路的数学工具，也称为布尔代数或开关代数。1938年，美国贝尔电话实验室的数学家、现代信息理论的创始人克劳德·山农提出了开关代数。他将近百年前英国数学家和逻辑学家乔治·布尔创立的布尔代数直接运用于开关电路的结果。尽管开关代数仅是布尔代数的一种特殊情况，即二值的布尔代数，但是大多数人还是习惯使用术语“布尔代数”。目前，一般情况下所提的布尔代数、逻辑代数都是开关代数，而

不是早期的布尔代数。

1.2 数制与码制

1.2.1 数制

一、各种计数体制及其表示方法

所谓数制就是计数的方法。在生产实践中，人们创造了许多计数方法，如七进制、十进制、十二进制等。而计数体制都采用位置计数法，即以特定的一些数字符号（也称数码）排列起来，每个符号处于不同位置作为各位的系数，每个位置都有一定的位权。其数值就是把各位的位权乘以该位的系数相加之和。

1. 十进制

十进制是以 10 为基数的计数体制。十进制数有 0、1、2、3、4、5、6、7、8、9 十个数码，其进位规则是逢十进一。各位的系数为其中之一，各位的位权是以 10 为底的整数幂。如 $431.25=4\times10^2+3\times10^1+1\times10^0+2\times10^{-1}+5\times10^{-2}$，式中，$10^2$、$10^1$、$10^0$、$10^{-1}$、$10^{-2}$ 是根据每一位数码所在的位置而定的，所以称之为位权。

任意一个十进制数 D 可展开为

$$\mathrm{D}=\sum_{i=-\infty}^{\infty}K_i10^i \tag{1.2.1}$$

式中：K_i 称为第 i 位的系数，它可以是 0～9 十个数码中的任何一个；10^i 称为第 i 位的权。

式（1.2.1）称为按权展开式。

通常，对十进制数的表示，可以在数字的右下角标注 10 或 D（D 代表 Decimal number），如 $(15)_{10}$和 $(15)_{\mathrm{D}}$。

若以 N 代替式（1.2.1）中的 10，则可得到任意进制数的按权展开式

$$\mathrm{D}=\sum_{i=-\infty}^{\infty}K_iN^i \tag{1.2.2}$$

式中：K_i 为第 i 位的系数；N 为计数基数；N^i 为第 i 位的权。

K_iN^i 为第 i 位的加权系数，故任意进制数的数值等于各加权系数之和。

2. 二进制

在数字电路中常采用的是二进制，因为二进制只有两个数码 0 和 1，可以与电路的两个状态（导通或截止）直接对应。二进制是以 2 为基数的计数体制，其进位规则是逢二进一。各位的系数为 0 或 1，各位的位权是以 2 为底的整数幂。为便于区分十进制数和二进制数，人们规定，凡注有下标 2 或 B（B 代表 Binary number）的数为二进制数。其按权展开式与十进制数相同，如 $(101.01)_2=1\times2^2+0\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}$。

一位二进制数也叫做一比特（bit），八位二进制数叫做一字节（Byte），十六位二进制数叫做一个字（Word）。二进制数的位数叫字长，例如一字节字长是八位，一字字长是十六位。

3. 八进制和十六进制

在数字系统中，二进制数位往往很长，读写不方便，一般采用八进制或十六进制对二进制数进行读和写。

八进制是以 8 为基数的计数体制，其进位规则是逢八进一。各位的系数为 0、1、2、3、4、5、6 和 7 八个数码，各位的位权是以 8 为底的整数幂。由于八进制的数码和十进制的前八个数码相同，但其位权基数不同，所以书写时要标注下标 8 或在括号外右下角标注英文字母 O（O 代表 Octal number）。其按权展开式与十进制数相同，如 $(354.2)_8=3\times8^2+5\times8^1+4\times8^0+2\times8^{-1}$。

十六进制是以 16 为基数的计数体制，其进位规律是逢十六进一。各位的系数为 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E 和 F 十六个数码，各位的位权是以 16 为底的整数幂。为便于区分十进制数和十六进制数，人们规定，凡注有下标 16 或 H（H 代表 Hexadecimal number）的数为十六进制数。其按权展开式与十进制数相同，如 $(3BD.2)_{16}=3\times16^2+11\times16^1+13\times16^0+2\times16^{-1}$。

二、数制之间的转换

1. 非十进制数转换为十进制数

将非十进制数转换为十进制数，通常采用“按权展开求累加和法”，即将非十进制数的按权展开式按照十进制的规律进行运算，就可以得到等值的十进制数。

【例 1.2.1】 将 $(101.11)_2$ 转换成十进制数。

解
$$\begin{aligned}(101.11)_2&=1\times2^2+0\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-2}\\&=4+0+1+0.5+0.25\\&=5.75\end{aligned}$$

2. 十进制数转换为非十进制数

十进制整数和小数转换成非十进制数的方法是不同的。整数部分可以采用连除法，即将原十进制数连续除以转换计数体制的基数，每次除完所得的余数就作为要转换的系数；先得到的余数为转换数的低位，后得到高位，直到除得的商为 0 为止。这种方法概括起来可说成“除基数、得余数、作系数、从低位、到高位”。十进制数小数部分转换成非十进制小数可采用连乘法，即将原十进制数纯小数乘以要转换的数制的基数，取其积的整数部分作系数，剩余的纯小数部分再乘基数，先得到的整数作新数的高位，后得到的作低位，直至其纯小数部分为 0 或到一定精度为止。这种方法概括起来可说成“乘基数、取整数、作系数、从高位、到低位”。

【例 1.2.2】 将 $(27.75)_{10}$ 转换成二进制数。

解 （1）将整数部分 27 转换成二进制数

2	27		余数
2	13	· · ·	1
2	6	· · ·	1
2	3	· · ·	0
2	1	· · ·	1
	0	· · ·	1

即 $(27)_{10}=(11\,011)_2$

（2）0.75 转换成二进制数

		整数
0.75×2=1.5	· · · · ·	1
0.5×2=1.0	· · · · ·	1

即 $(0.75)_{10} = (0.11)_2$

所以$(27.25)_{10} = (11\,011.11)_2$

3. 二进制数与八进制数、十六进制数间相互转换

由于 $2^3=8$，$2^4=16$，所以一位八进制数（或十六进制数）相当于 3（或 4）位二进制数，它们是完全对应的。因此二进制数转换成八进制数（或十六进制数）的规则如下：

从小数点算起，向左或向右每 3（或 4）位分成一组，最后不足 3（或 4）位者用 0 补齐，每组用 1 位等值的八进制数（或十六进制数）表示，即得到要转换的八进制数（或十六进制数）。

【例 1.2.3】 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 $(10\,111\,011.011\,1)_2 = (273.36)_8 = (BB.78)_{16}$

反之，八进制数（或十六进制数）转换成二进制数时，只要将每位八进制数（或十六进制数）分别写成相应的 3（或 4）位二进制数，按原来的顺序排列起来即可。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 1.2.4】 将 $(26.35)_8$ 转换成二进制数。

解

2		6	.	3		5	
0	1	0		1	1	0	
				0	1	1	1	0	1

即 $(26.35)_8 = (10\,110.011\,101)_2$

1.2.2 码制

码制是指用数字或字符进行的编码。常用的编码有多种，这里只介绍二—十进制编码。二—十进制码，也叫 BCD 码，是用二进制码表示的十进制数。由于十进制数有 0～9 共十个数码，所以用四位二进制码表示，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。四位二进制码共有十六种不同的组合（或叫状态），可以选取其中的任意十个组合代表 0～9 十个数字，这种表示方法称为编码。其方案有很多种，但常用的只有几种，见表 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。

（2）2421BCD 码。2421BCD 码是有固定权的码，各位的权值分别为 2、4、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。

（4）余 3 码。余 3 码是 8421BCD 码的每个码组加 0011 形成。余 3 码的各位无固定的权，称为无权码。余 3 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 3 码都互为自补代码。

（5）格雷码（Gray 码）。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，因而格雷码也叫余 3 循环码，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。

表 1.2.1 几种常用的BCD码

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

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。

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，若要确定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 $(95.7)_{10}=(10\ 010\ 101.011\ 1)_{8421BCD}=(11\ 001\ 000.101\ 0)_{余3}$。

1.3 逻辑代数

一个实际的数字系统，其电路是非常复杂的。在分析和设计数字电路时，常常借助于一种数学工具——逻辑代数。逻辑代数是按一定的逻辑规律进行运算的代数，和普通代数一样，也是用字母表示变量。但是，逻辑代数的变量（称为逻辑变量）的取值十分简单，不是0就是1，没有第三种可能。而且，这里的0和1不表示数量的大小，只表示两种不同的逻辑状态。例如，用1和0表示真和假、是和非、有和无、开和关等。所以逻辑代数要比普通代数简单得多。

1.3.1 逻辑变量与逻辑函数

研究事物原因（条件）和结果之间因果关系规律的命题称为逻辑命题。广义地讲，逻辑就是规律，一般人们称决定事物的因素（原因）为逻辑自变量，而称被决定的事物的结果为逻辑结果（或称逻辑因变量）。这就是所说的逻辑变量。

逻辑代数中的变量往往用字母A、B、C、…表示。每个变量只取“0”或“1”两种情况，即变量不是取“0”，就是取“1”，不可能有第三种情况。它相当于信号的有或无，电平的高或低，电路的导通或截止。这使逻辑代数可以直接用于双值系统逻辑电路的研究。

如果以逻辑变量作为输入，以运算结果作为输出，那么当输入变量的取值确定之后，输出的取值便随之而定。因此，输出逻辑变量与输入逻辑变量之间是一种函数关系，这种函数

关系称为逻辑函数。由于变量和函数的取值只有0和1两种状态，所以后面讨论的都是二值逻辑函数。

1.3.2　基本逻辑运算及其表示方法

在逻辑电路中，基本的逻辑关系只有逻辑与、逻辑或和逻辑非三种，把能够实现这三种逻辑运算的电路称为逻辑门电路，简称门电路，分别叫做与门、或门、非门。因此，在逻辑代数中，相应地也有三种基本逻辑运算，即与运算、或运算和非运算。

一、与运算和与门

1. 与运算

若决定某一事物结果的所有条件同时具备时，结果才会发生，这种因果关系叫做逻辑与，也叫逻辑乘、与逻辑。图1.3.1（a）所示的是两个串联的开关控制一个灯泡的电路，若把开关的闭合作为条件，灯泡亮作为结果，显然，只有开关A、B都闭合，灯泡才会亮。

2. 与门

在用电路实现逻辑运算时，用输入端的电压或电平表示自变量，用输出端的电压或电平表示因变量。能够实现与逻辑的基本单元电路叫做与门。图1.3.1（b）所示的是由二极管构成的与门电路。由该图可知，在输入端A、B中只要有一个信号为低电平0，则对应输入端相连的二极管获得正向电压而导通，在二极管的钳位作用下，使输出Y为低电平0；只有输入A、B同时为高电平1时，输出Y才为高电平1。可见，输出与输入之间存在与运算关系。与门逻辑图形符号如图1.3.1（c）所示。

显然对于输入变量的不同取值，输出变量均有与其对应的逻辑值。把输入、输出变量所有相互对应的逻辑值列在一个表格内，这种表格称为逻辑函数真值表，简称真值表，它能清楚地表示事物的因果关系。具有二输入变量的与逻辑函数真值表见表1.3.1。

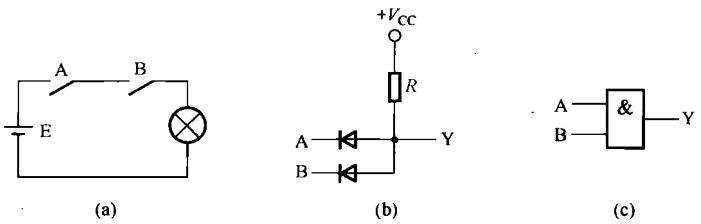


图1.3.1　与门电路和与门逻辑图形符号

（a）与运算关系；（b）二极管与门电路；（c）与门逻辑图形符号

表1.3.1　　与逻辑函数真值表

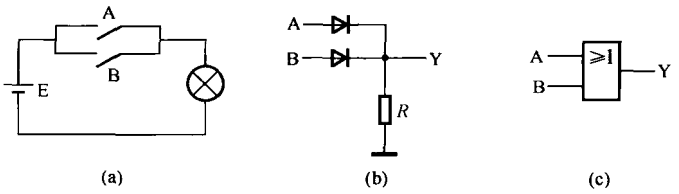
A	B	Y	A	B	Y
0	0	0	1	0	0
0	1	0	1	1	1

与逻辑关系用逻辑函数来表示，称为逻辑乘，即$Y=A\cdot B=AB$。

二、或运算和或门

1. 或运算

若决定某一事物结果的所有条件中只要有一个或一个以上条件具备时，结果就会发生，则这种因果关系叫做或逻辑，也称逻辑加。图1.3.2（a）所示的是两个并联的开关控制一个灯泡的电路，若把开关闭合作为条件，灯泡亮作为结果，显然，开关A、B只要有一个闭合，灯泡就会亮。



(a) (b) (c)

图 1.3.2 或门电路与或门逻辑图形符号

(a) 或运算关系；(b) 二极管或门电路；(c) 或门逻辑图形符号

2. 或门

能够实现或逻辑的基本单元电路叫做或门。如图 1.3.2（b）所示的是由二极管构成的或门电路。由该图可知，在输入端 A、B 中只要有一个信号为高电平 1，则对应输入端相连的二极管就会导通，输出 Y 即为高电平 1；只有输入 A、B 同时为低电平 0 时，输出 Y 才为低电平 0。或门逻辑图形符号如图 1.3.2（c）所示，或逻辑函数真值表见表 1.3.2。

表 1.3.2 或逻辑函数真值表

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

或逻辑关系用逻辑函数来表示，称为逻辑或，即 Y＝A＋B。

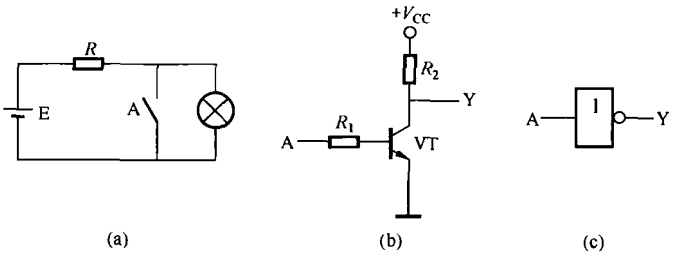
三、非运算和非门

1. 非运算

非逻辑是这样一种因果关系：某事情发生与否，仅取决于一个条件，而且是对该条件的否定，即条件具备时事情不发生，条件不具备时事情才发生，这样的因果关系叫做逻辑非，也称逻辑求反。图 1.3.3（a）所示是一个开关和一个灯泡并联的电路，若把开关闭合作为条件，灯泡亮作为结果，显然，开关 A 闭合，灯泡不亮；开关 A 打开，灯泡亮。

2. 非门

能够实现非逻辑的基本单元电路叫做非门（或称反相器）。图 1.3.3（b）所示的是晶体管非门电路。图中，当输入端 A 为高电平 1 时，晶体管处于饱和状态，输出 Y 为低电平 0；当输入端 A 为低电平 0 时，晶体管处于截止状态，输出 Y 为高电平 1。非门电路的逻辑图形符号如图 1.3.3（c）所示。非逻辑函数真值表见表 1.3.3。



(a) (b) (c)

图 1.3.3 非门电路和非门逻辑图形符号

(a) 非运算关系；(b) 三极管非门电路；(c) 非门逻辑图形符号

表 1.3.3 非逻辑函数真值表

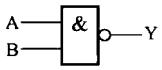
A	Y
0	1
1	0

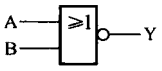
非逻辑关系用逻辑函数来表示，称为逻辑非，即 $Y=\overline{A}$。

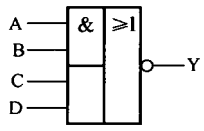
1.3.3 复合逻辑运算

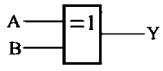
人们在研究实际问题时发现，事物的各个因素之间的逻辑关系往往要比单一的与、或、非复杂得多，不过它们都可以用与、或、非的组合来实现。含有两种或两种以上逻辑运算的逻辑函数称为复合逻辑函数。其逻辑表达式、真值表及逻辑图形符号见表 1.3.4。

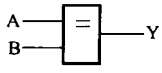
表 1.3.4 常用复合逻辑运算的逻辑表达式、真值表及逻辑图形符号

<table>
<tr><th>逻辑运算</th><th>逻辑表达式</th><th>真值表及逻辑图形符号</th></tr>
<tr><td>与非</td><td>$Y=\overline{AB}$</td><td>
<table>
<tr><th>A</th><th>B</th><th>Y</th></tr>
<tr><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
</table>


</td></tr>
<tr><td>或非</td><td>$Y=\overline{A+B}$</td><td>
<table>
<tr><th>A</th><th>B</th><th>Y</th></tr>
<tr><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
</table>


</td></tr>
<tr><td>与或非</td><td>$Y=\overline{AB+CD}$</td><td>
<table>
<tr><th>A</th><th>B</th><th>C</th><th>D</th><th>Y</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
</table>


</td></tr>
<tr><td>异或</td><td>$Y=\overline{A}B+A\overline{B}=A\oplus B$</td><td>
<table>
<tr><th>A</th><th>B</th><th>Y</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
</table>


</td></tr>
<tr><td>同或</td><td>$Y=\overline{A}\cdot\overline{B}+AB=A\odot B$</td><td>
<table>
<tr><th>A</th><th>B</th><th>Y</th></tr>
<tr><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table>


</td></tr>
</table>

需要说明的是在数字电路中，对逻辑变量的逻辑状态用不同的逻辑体制表示时，所得到的逻辑函数也就不同。

当逻辑电路中的高电平用逻辑 1 表示，低电平用逻辑 0 表示，这种逻辑赋值方法称之为正逻辑。反之，若高电平用逻辑 0 表示，低电平用逻辑 1 表示，这种逻辑赋值方法称之为负逻辑。对于同一电路，采用哪一种逻辑是人为进行的，并不牵涉电路本身结构的优劣，但是所采用的逻辑约定不同，电路所执行的逻辑操作可能完全不同。

正、负逻辑虽不相同，但它们之间存在着一定的关系。以上介绍各种门电路时采用的都是正逻辑。如果把表 1.3.1 的电平关系按负逻辑赋值，则所得到的真值表与表 1.3.2 相同，即图 1.3.1 按正逻辑是“与”门，按负逻辑是“或”门。还可以证明，在正逻辑和负逻辑中，“非”关系是相同的。

目前在逻辑电路中习惯采用正逻辑，今后如不加特殊说明，本书一律采用正逻辑。

1.3.4 逻辑代数的基本定律和规则

逻辑代数是研究逻辑电路的数学工具，它为分析和设计逻辑电路提供了理论基础。根据三种基本逻辑运算，可推导出一些基本公式和定律，形成一些运算规则。熟悉、掌握并且会运用这些规则，对于掌握数字电子技术十分重要。

一、基本定律和常用公式

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 1.3.5。其中有的定律与普通代数相似，有的定律与普通代数不同，使用时切勿混淆。

表 1.3.5　　逻辑代数基本定律

定律名称	定　律	
0－1 定律	$A\cdot 0=0$	$A+1=1$
自等定律	$A\cdot 1=A$	$A+0=A$
重叠定律	$A\cdot A=A$	$A+A=A$
互补定律	$A\cdot \overline{A}=0$	$A+\overline{A}=1$
交换定律	$A\cdot B=B\cdot A$	$A+B=B+A$
结合定律	$A\cdot (B\cdot C)=(A\cdot B)\cdot C$	$A+(B+C)=(A+B)+C$
分配定律	$A\cdot (B+C)=A\cdot B+A\cdot C$	$A+BC=(A+B)(A+C)$
吸收定律 1	$(A+B)\cdot (A+\overline{B})=A$	$A\cdot B+A\cdot \overline{B}=A$
吸收定律 2	$A\cdot (\overline{A}+B)=A\cdot B$	$A+\overline{A}\cdot B=A+B$
摩根定律	$\overline{A\cdot B}=\overline{A}+\overline{B}$	$\overline{A+B}=\overline{A}\cdot \overline{B}$
还原定律	$\overline{\overline{A}}=A$	

表 1.3.5 给出的这些定律的正确性可以用真值表的方法加以证明，若将变量的所有取值代入等式两边，两边的结果相等，则等式成立。

【例 1.3.1】　试用真值表验证$\overline{A+B}=\overline{A}\cdot \overline{B}$的正确性。

解　将变量 A 和 B 的所有取值代入等式两边，得出的真值表见表 1.3.6，结果表明$\overline{A+B}$与$\overline{A}\cdot \overline{B}$的结果相等，故等式成立。

表 1.3.6 $\overline{A+B}$与$\overline{A}\cdot\overline{B}$的真值表

A	B	$\overline{A}$	$\overline{B}$	$\overline{A+B}$	$\overline{A}\cdot\overline{B}$
0	0	1	1	1	1
0	1	1	0	0	0
1	0	0	1	0	0
1	1	0	0	0	0

二、三个重要规则

1. 代入规则

对于任何一个逻辑等式，将等式两边出现的同一变量都以某个逻辑变量或逻辑函数同时取代后，等式依然成立，这就是代入规则。

利用代入规则可以方便地扩展公式。例如，给定等式 $A+\overline{A}B=A+B$，若式中 A 都用 $A+C$ 来代替，则等式仍成立，即

$$(A+C)+\overline{A+C}\cdot B=(A+C)+B$$

代入规则对逻辑函数化简非常有用。

2. 对偶规则

对于任何一个逻辑式 Y，如果将其中的“·”换成“+”、“+”换成“·”，“0”换成“1”、“1”换成“0”，则得到一个新的函数式，这就是函数 Y 的对偶式，记作 Y'。

可以证明，若两个逻辑式相等，则它们的对偶式也相等，这就是对偶规则。

利用对偶规则可以帮助减少公式的记忆量。例如，表 1.3.5 中的基本定律的左边公式和右边公式就互为对偶，只需记住一边的公式即可。因为利用对偶规则，不难得出另一边的公式。

3. 反演规则

如果将逻辑函数表达式 Y 中所有的“·”换成“+”、“+”换成“·”，“0”换成“1”、“1”换成“0”，原变量换成反变量，反变量换成原变量，得到的逻辑函数式用 $\overline{Y}$ 表示，这一规则称为反演规则。其中函数式 $\overline{Y}$ 是原函数式 Y 的反函数。

例如，已知函数 $Y=AB+\overline{A}C$，则其反函数为 $\overline{Y}=(\overline{A}+\overline{B})(A+\overline{C})$。

显然，利用反演规则很容易求出一个函数的反函数。使用反演规则时，应注意正确使用括号来保持原来的运算顺序，否则就会发生错误；另外，不属于单个变量上的非号应保留不变，这点应特别注意。

【例 1.3.2】 若 $Y=(\overline{A}+BC)\overline{\overline{C}D}$，求 $\overline{Y}$ 和 Y'。

解 根据反演规则可知

$$\overline{Y}=A(\overline{B}+\overline{C})+\overline{C+\overline{D}}=A\overline{B}+A\overline{C}+\overline{C}D$$

根据对偶规则可知

$$Y'=\overline{A}(B+C)+\overline{\overline{C}+D}$$

1.4 逻辑函数及其表示方法

1.4.1 逻辑函数的表示方法

常用的逻辑函数的表示方法有逻辑真值表（简称真值表）、逻辑函数式（也称逻辑式或