



卓越工程师培养计划

▪ EDA ▪

<http://www.phei.com.cn>

杜树春 编著



基于 Proteus

的数字集成电路 快速上手



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

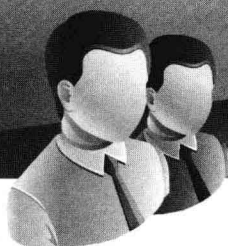


卓越工程师培养计划

▪ EDA ▪

<http://www.phei.com.cn>

杜树春 编著



基于Proteus 的数字集成电路 快速上手

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书主要介绍用 Proteus 软件来测试数字集成电路的方法。这种测试方法方便、迅速、麻烦少、成本低，可以说是一条让人快速掌握数字集成电路用法的捷径。

本书共分 8 章，第 1 章是集成电路基础知识介绍，第 2 章介绍用 Proteus 软件来测试数字集成电路的方法，第 3~8 章讲解用 Proteus 软件来测试数字集成电路的实例。

本书通俗易懂，特别适合从事电子电路设计的初学者自学使用，也可作为高等学校相关专业的教学用书。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目（CIP）数据

基于 Proteus 的数字集成电路快速上手 / 杜树春编著. —北京：电子工业出版社，2012.10

（卓越工程师培养计划）

ISBN 978-7-121-18601-1

I. ①基… II. ①杜… III. ①数字集成电路—电路设计—计算机辅助设计 IV. ①TN431.202

中国版本图书馆 CIP 数据核字（2012）第 226973 号

策划编辑：张 剑（zhang@phei.com.cn）

责任编辑：刘真平

印 刷：三河市鑫金马印装有限公司

装 订：三河市鑫金马印装有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1 092 1/16 印张：12 字数：307.2 千字

印 次：2012 年 10 月第 1 次印刷

印 数：4000 册 定价：29.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前 言

随着科技的发展,数字集成电路芯片获得越来越广泛的应用,尤其是在微处理器、单片机及嵌入式系统出现以后,数字集成电路芯片应用势头不减。因为数字集成电路芯片除了单独使用外,大量的与微处理器、单片机及嵌入式系统配合使用。发展到现在,数字集成电路功能各异的芯片成千上万,而且不断地在增加,没有一个人能掌握所有的集成电路的特性和用法。多数人都是用一个学一个,在不断应用中增加自己的集成电路知识。

假设我们现在要使用一个陌生的数字集成电路芯片,在采用传统的测试数字集成电路功能的方法时要分以下步骤:① 先了解该数字集成电路的特性,包括芯片引脚排列、功能表或真值表等(必要时要从网上下载详细介绍该芯片性能的“PDF”文件);② 购买该数字集成电路芯片,画出测试该数字集成电路基本特性的电路原理图,在面包板上或印制电路板上照图搭出实际电路,加上直流电源,还需要万用表、示波器和信号发生器等测试仪器的配合才能完成对该数字集成电路芯片的性能测试。有没有一种快速测试数字集成电路芯片的办法呢?有。办法就是用 Proteus 软件来测试数字集成电路。

若采用 Proteus 软件来测试数字集成电路,上述第②步就可以简化。具体如下:无须一开始就购买数字集成电路芯片,只需在计算机上的 Proteus 软件环境下,画出测试该数字集成电路基本特性的电路原理图,再通过单击鼠标来设定数字集成电路高低电平的输入,立即就可显示该数字集成电路应有的输出。采用这种方法调试还有一大优点,无须实际的万用表、示波器和信号发生器等测试仪器的配合,用同样在计算机上的虚拟示波器、虚拟信号发生器和虚拟电流电压表就可以了。这种测试方法方便、迅速、麻烦少、成本低,不失为一条快速掌握数字集成电路用法的捷径。本书主要介绍这种用 Proteus 软件来测试数字集成电路的方法。

本书共分 8 章,第 1 章是集成电路基础知识介绍,第 2 章介绍用 Proteus 软件来测试数字集成电路的方法,第 3~8 章讲解用 Proteus 软件来测试数字集成电路的实例。这几章在章节的顺序安排上和普通“数字电子技术”或“数字电子技术基础”教科书一致。第 3 章是门电路,第 4 章是组合逻辑电路,第 5 章是触发器,第 6 章是时序逻辑电路,第 7 章是脉冲波形的产生与变换,第 8 章是 D/A 和 A/D 转换器。

目前,一般的工科院校电子、计算机、通信、机电等专业都开设有模拟电子技术和数字电子技术课,本书可作为学生学习数字电子技术这门课的辅助教材。本书适合以下人员阅读或参考:一是学习数字电子技术的大、中专及高等职业学校、中等职业学校的在校学生;二是和电子专业有关的广大工程技术人员;三是广大电子科技爱好者。

数字电子技术是一门实践性很强的课程,本书偏重于实践,书中所有实例均可重复,且易于重复,使读者在极短的时间内便可掌握一个集成电路芯片的主要功能。本书通俗易懂,特别适合初学者或入门者自学。

在编写过程中,得到了杜菲、姜鸿斌等同志的协助,在此表示衷心感谢。

由于编著者水平有限且时间仓促,书中难免存在缺点和错误,恳请读者批评指正。我的电子邮箱为 E-mail: dushuchun@263.net。

编 著 者

2012 年 3 月 14 日

目 录

第 1 章 集成电路的基础知识	1
1.1 数制和码制	1
1.2 3 种基本门电路	4
1.3 常用的复合门电路	7
1.4 基本逻辑门电路	10
1.5 三态输出门、集电极开路门与漏极开路门	13
1.6 数字集成电路的主要种类和型号	15
1.7 数字集成电路的参数	17
1.8 习题	19
第 2 章 Proteus 软件用法	20
2.1 进入 Proteus ISIS	20
2.2 Proteus ISIS 工作界面	21
2.3 Proteus ISIS 电路原理图设计	25
2.4 Proteus ISIS 原理图设计中若干注意事项	32
2.5 Proteus VSM 仿真工具简介	34
2.6 用 Proteus 软件测试数字集成电路的方法	37
2.6.1 8 输入与非门 CD4068 功能测试	37
2.6.2 多路开关 CD4066 功能测试	40
2.6.3 十进制同步可逆计数器 74LS190 功能测试	41
第 3 章 门电路	45
3.1 基本门电路	45
3.1.1 与门	45
3.1.2 或门	47
3.1.3 非门	48
3.2 复合门电路	49
3.2.1 与非门	50
3.2.2 或非门	51
3.2.3 异或门	52
3.2.4 同或门	54
3.2.5 与或非门	55
3.3 特殊门电路	57
3.3.1 集电极开路门 (OC 门)	57

3.3.2	漏极开路门 (OD 门)	59
3.3.3	三态输出门 (TS 门)	61
3.4	习题	62
第 4 章	组合逻辑电路	63
4.1	译码器	63
4.1.1	二进制译码器	63
4.1.2	二-十进制译码器	65
4.1.3	显示译码器	67
4.2	优先编码器	70
4.3	数据选择器	73
4.4	数值比较器	75
4.5	加法器	78
4.6	奇偶产生器/校验器	79
4.6.1	9 位奇偶产生器/校验器	80
4.6.2	8 位奇偶产生器/校验器 74LS180	82
4.7	习题	85
第 5 章	触发器	86
5.1	触发器介绍	86
5.2	常见触发器	88
5.3	基本 RS 触发器	90
5.4	同步 RS 触发器	91
5.5	集成电路基本 RS 触发器 74LS279	93
5.6	集成电路 JK 触发器 74LS73	95
5.7	集成电路双 D 触发器 74LS74	97
5.8	集成电路八 D 触发器 74LS273	100
5.9	集成电路八 D 锁存器 74LS373	101
5.10	习题	104
第 6 章	时序逻辑电路	105
6.1	计数器	105
6.1.1	4 位二进制同步加法计数器 74LS161	105
6.1.2	4 位二进制同步可逆计数器 74LS191	108
6.1.3	4 位十进制异步减法计数器 CD4522	111
6.1.4	4 位十进制同步加法计数器 74LS160	114
6.1.5	双 4 位十进制加法计数器 74LS390	116
6.2	寄存器	119
6.2.1	4 位数码寄存器 74LS175	120

6.2.2	4 位双向移位寄存器 74LS194	122
6.2.3	串入并出移位寄存器 74LS164	125
6.2.4	并入串出移位寄存器 74LS165	127
6.2.5	并入并出 8 位通用移位存储寄存器 74LS299	130
6.3	习题	135
第 7 章	脉冲波形的产生与变换	136
7.1	多谐振荡器	136
7.1.1	对称式多谐振荡器	136
7.1.2	非对称式多谐振荡器	137
7.1.3	环形振荡器	139
7.1.4	石英晶体振荡器	140
7.2	整形电路	141
7.2.1	用门电路组成的施密特触发器	143
7.2.2	集成电路施密特触发器 74LS14	144
7.2.3	集成电路施密特触发器 CD40106	145
7.2.4	集成单稳态触发器 74LS121	146
7.2.5	可重复触发单稳态触发器 74LS123	149
7.2.6	非重复触发单稳态触发器 74LS221	151
7.3	555 定时器	153
7.3.1	用 NE555 构成多谐振荡器	154
7.3.2	用 NE555 构成施密特触发器	155
7.3.3	用 NE555 构成单稳态触发器	156
7.4	习题	158
第 8 章	D/A 和 A/D 转换器	159
8.1	D/A 转换器	159
8.2	A/D 转换器	167
8.3	A/D 转换器和 D/A 转换器使用说明	173
附录 A	美国标准信息交换码 (ASCII 码) 表	174
附录 B	Proteus 提供的所有元件分类和子类列表	175
附录 C	数字电路基本逻辑单元图形符号对照表	180
参考文献		182

第1章 集成电路的基础知识

数字集成电路发明于 1961 年, 50 多年来, 数字集成电路技术取得了飞速的发展, 其集成度不断提高, 功能日趋完善。数字集成电路已广泛应用于人类生活的各个领域。本章将分几个小节介绍数字集成电路基础知识。这些小节分别是数制和码制, 三种基本门电路, 常用的复合门电路, 基本逻辑门电路, 三态输出门、集电极开路门与漏极开路门电路, 数字集成电路的主要种类和型号及数字集成电路的四电压参数。

1.1 数制和码制

1. 数制 (Number Systems)

数码: 由数字符号构成且表示物理量大小的数字和数字组合。**计数制 (简称数制):** 多位数码中每一位的构成方法, 以及从低位到高位进制规则。常用的数制有十进制、二进制和十六进制。

1) 十进制 十进制是人们最熟悉的计数制。在十进制数中, 每一位有 0~9 十个数码, 所以计数的基数是 10。超过 9 的数必须用多位数表示, 其中低位和相邻高位之间的关系是“逢十进一”, 故称为十进制。其特点为:

- 数字符号 (系数): 0、1、2、3、4、5、6、7、8、9
- 计数规则: 逢十进一
- 基数: 10
- 权: 10 的幂

例: $(1999)_{10} = (1 \times 10^3 + 9 \times 10^2 + 9 \times 10^1 + 9 \times 10^0)_{10}$

2) 二进制 二进制是数字电路和计算机中使用的计数制。在二进制数中, 每一位有 0 和 1 两个数码, 所以计数的基数是 2。低位和相邻高位之间的关系是“逢二进一”, 故称为二进制。其特点为:

- 数字符号: 0、1
- 计数规则: 逢二进一
- 基数: 2
- 权: 2 的幂

例: $(1011101)_2 = (1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0)_{10}$
 $= (64 + 0 + 16 + 8 + 4 + 0 + 1)_{10}$
 $= (93)_{10}$

3) 十六进制 十六进制是一种便于二进制数和十进制数相互转换的数制, 十六进制数的每一位有 16 个不同的数码, 分别用 0~9、A (10)、B (11)、C (12)、D (13)、E (14)、F (15) 表示, 低位和相邻高位之间的关系是“逢十六进一”。其特点为:

- 数字符号: 0~9、A、B、C、D、E、F
- 计数规则: 逢十六进一

➤ 基数: 16

➤ 权: 16 的幂

$$\begin{aligned}\text{例: } (5D)_{16} &= (5 \times 16^1 + 13 \times 16^0)_{10} \\ &= (80 + 13)_{10} \\ &= (93)_{10}\end{aligned}$$

2. 数制之间的转换 (Conversion)

我们所熟悉的数制是十进制, 数字电路和计算机使用的数制是二进制, 十六进制则是两者相互转换的桥梁。

1) 二进制与十六进制之间的转换

(1) 二进制→十六进制: 用“四位分组法”。因为 4 位二进制数对应 1 位十六进制数, 将待转换的二进制数从后往前, 每四位分成一组 (如不够四位在前面补 0), 再把每组二进制数写成对应的十六进制数。十进制数、二进制数和十六进制数对照表如表 1-1 所示。例如:

表 1-1 十进制数、二进制数和十六进制数对照表

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	0	0	9	1001	9
1	1	1	10	1010	A
2	10	2	11	1011	B
3	11	3	12	1100	C
4	100	4	13	1101	D
5	101	5	14	1110	E
6	110	6	15	1111	F
7	111	7	16	10000	10
8	1000	8			

$$\begin{aligned}(10111010110)_2 &= (0101\ 1101\ 0110)_2 \\ &= (5D6)_{16}\end{aligned}$$

(2) 十六进制→二进制: 只需将每个十六进制数直接写成二进制数即可。例如:

$$\begin{aligned}(9A7E)_{16} &= (1001\ 1010\ 0111\ 1110)_2 \\ &= (1001101001111110)_2\end{aligned}$$

2) 十进制与十六进制之间的转换

(1) 十六进制→十进制: 用“按权相加法”, 例如:

$$\begin{aligned}(5D)_{16} &= (5 \times 16^1 + 13 \times 16^0)_{10} \\ &= (80 + 13)_{10} \\ &= (93)_{10}\end{aligned}$$

(2) 十进制→十六进制: 根据十进制数的大小 (它在 $0 \sim 16 \sim 16^2 \sim 16^3 \dots$ 中的位置) 确定除数 16^N , $N=1, 2, 3, \dots$ 。例如: 把十进制数 810 转化成十六进制数。

解: 因为 810 这个数在 $16^2 \sim 16^3$ 之间, 故先用 16^2 作为除数。

$$810/16^2=810/256=3 \text{ 余 } 42$$

$$42/16=2 \text{ 余 } 10$$

所以 $810=32AH$

3) 十进制与二进制之间的转换

(1) 二进制 $\rightarrow$ 十进制: 用“按权相加法”, 例如:

$$\begin{aligned}(1011101)_2 &= (1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0)_{10} \\ &= (64 + 0 + 16 + 8 + 4 + 0 + 1)_{10} \\ &= (93)_{10}\end{aligned}$$

(2) 十进制 $\rightarrow$ 二进制: 整数部分的转换用“除 2 取余法”。

例: 求 $(217)_{10} = (\quad)_2$

解: $\because$

2 217		余 1	b0
2 108		余 0	b1
2 54		余 0	b2
2 27		余 1	b3
2 13		余 1	b4
2 6		余 0	b5
2 3		余 1	b6
2 1		余 1	b7
0			

$$\therefore (217)_{10} = (11011001)_2$$

3. 编码 (Encode)

由于数字电路及计算机只能处理二进制数和二进制编码, 因此, 任何进入计算机的信息必须转化为二进制数或二进制编码。几种常用的编码是: 二进制码、二-十进制码 (BCD 码) 和美国标准信息交换码 (ASCII 码)。

1) 二进制码 (Binary-Coded) 用 0、1 组成的二进制数码不仅可以表示数值的大小, 而且可以用来表示特定的信息。这种具有特定含义的二进制数码称为二进制代码。建立这种代码与它表示的对象 (如十进制数、字母、特定符号、逻辑值等) 的一一对应关系的过程称为编码; 将代码所表示的特定信息翻译出来称为译码, 分别由编码器和译码器来实现。

2) 二-十进制编码 (Binary-Coded Decimal) 二-十进制编码就是用 4 位二进制数来表示 0~9 这 10 个十进制数符号, 简称 BCD 码。由于 4 位二进制数从 0000~1111 共有 16 种组合, 而十进制数只有 10 个数码符号, 因此有多种 BCD 码, 如 8421 码、2421 码等, 常用的是 8421-BCD 码。

8421-BCD 码用 4 位二进制数的前 10 种组合来表示 0~9 这 10 个十进制数, 它和十六进制数的前 10 个数是一样的。表 1-2 所示为十进制数与 BCD 码对照表。

表 1-2 十进制数与 BCD 码对照表

十进制数	BCD 码	十进制数	BCD 码
0	0000	2	0010
1	0001	3	0011

续表

十进制数	BCD 码	十进制数	BCD 码
4	0100	7	0111
5	0101	8	1000
6	0110	9	1001

从表中可见, 8421-BCD 码有如下特点:

- 选取 0000~1001 表示十进制数 0~9。
- 按自然顺序的二进制数表示所对应的十进制数字。
- 是有权码, 从高位到低位的权依次为 8、4、2、1, 故称为 8421 码。
- 1010~1111 等六种状态是不用的, 称为禁用码。

例: $(1985)_{10} = (0001\ 1001\ 1000\ 0101)_{8421BCD}$

3) 美国标准信息交换码 (ASCII 码) 美国标准信息交换码 (American Standard Code for Information Interchange), 简称 ASCII 码。用一个 7 位二进制数来表示一个特定的字符, 可表示 $2^7=128$ 个符号。这 128 个符号共分为两类: 一类是图形字符, 共 96 个; 一类是控制字符, 共 32 个。96 个图形字符包括十进制数码符号 10 个、大小写英文字符 52 个和其他字符 34 个。32 个控制字符包括回车符、换行符等。0~9 十个数字对应的 ASCII 码为 30H~39H; 大写英文字母 A~F 对应的 ASCII 码为 41H~46H; 小写英文字母 a~f 对应的 ASCII 码为 61H~66H。美国标准信息交换码 (ASCII 码) 表见附录 A。

1.2 3 种基本门电路

逻辑代数的 3 种基本运算分别是“与”逻辑运算、“或”逻辑运算和“非”逻辑运算。实现这三种基本运算的电路称为门电路, 分别叫与门 (AND Gate)、或门 (OR Gate) 和非门 (NOT Gate)。

1. 与逻辑运算 (AND)

如图 1-1 所示, 一根水管上串接 A 和 B 两个阀门, 只有 A 阀门和 B 阀门都打开时, 水流才能通过。其他情形, 一个也不开, 只打开一个, 水流都不能通过。如果阀门打开用“1”表示, 关闭用“0”表示, 则 A 阀门与 B 阀门都为“1”时才有水流出, 这就是“与”的概念。

再如图 1-2 所示, 一闭合电路中串联两个开关 A 和 B, 只有开关 A 和 B 都合上时, 电路中才有电流流过, 灯才能点亮。开关 A 和 B 状态及灯 L 状态之间有如表 1-3 所示的关系。



图 1-1 两只阀门相“与”

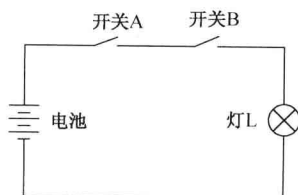


图 1-2 两只开关相“与”

表 1-3 开关 A 和 B 状态及灯 L 状态之间的关系

开关 A	开关 B	灯 L
断开	断开	不亮
断开	闭合	不亮
闭合	断开	不亮
闭合	闭合	亮

如果规定开关接通为“1”状态，开关断开为“0”状态，灯亮为“1”状态，灯灭为“0”状态，则把表 1-3 中的“断开”和“不亮”换为“0”，“闭合”和“亮”换为“1”，就是下面的与逻辑真值表。关于“与”逻辑，我们可以归纳出以下几点。

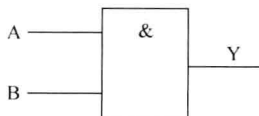
1) 与逻辑定义 当决定某一事件的所有条件都具备时，事件才能发生。这种决定事件的因果关系称为“与逻辑关系”。

2) 与逻辑真值表

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

3) 与逻辑函数式 $Y = A \cdot B$ 。

4) 与逻辑符号



5) 与逻辑运算 $0 \cdot 0 = 0$; $0 \cdot 1 = 0$; $1 \cdot 0 = 0$; $1 \cdot 1 = 1$ 。

2. 或运算 (OR)

如图 1-3 所示，一根水管上并接 A 和 B 两个阀门，只要 A 阀门和 B 阀门中有一个打开，水流就能通过。只有 A 阀门和 B 阀门一个也不开，水流才不能通过。如果阀门打开用“1”表示，关闭用“0”表示，则 A 阀门和 B 阀门中有一个为“1”时就有水流出，这就是“或”的概念。

再如图 1-4 所示，一闭合电路中并联两个开关 A 和 B，只要开关 A 和 B 中有一个合上，电路中就有电流流过，灯就能点亮。开关 A 和 B 状态及灯 L 状态之间有如表 1-4 所示的关系。

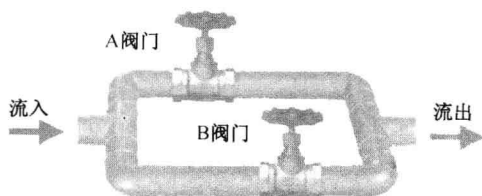


图 1-3 两只阀门相“或”

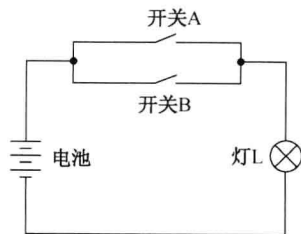


图 1-4 两只开关相“或”

表 1-4 开关 A 和 B 状态及灯 L 状态之间的关系

开关 A	开关 B	灯 L
断开	断开	不亮
断开	闭合	亮
闭合	断开	亮
闭合	闭合	亮

如果规定开关接通为“1”状态，开关断开为“0”状态，灯亮为“1”状态，灯灭为“0”状态，则把表 1-4 中的“断开”和“不亮”换为“0”，“闭合”和“亮”换为“1”，就是下面的或逻辑真值表。关于“或”逻辑，我们可以归纳出以下几点。

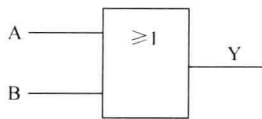
1) **或逻辑定义** 当决定某一事件的一个或多个条件满足时，事件便能发生。这种决定事件的因果关系称为“或逻辑关系”。

2) 或逻辑真值表

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

3) 或逻辑函数式 $Y=A+B$ 。

4) 或逻辑符号



5) 或逻辑运算 $0+0=0$; $0+1=1$; $1+0=1$; $1+1=1$ 。

3. 非运算 (NOT)

如图 1-5 所示，在一具有电源（电池）的闭合电路中，R 是限流电阻，Y 是灯，A 是开关。当开关 A 闭合时，灯不亮（因为此时电流经开关流过返回电池了）；当开关 A 打开时，灯亮。这种逻辑关系称为非逻辑关系。此时开关 A 状态及灯 Y 状态之间有如表 1-5 所示的关系。

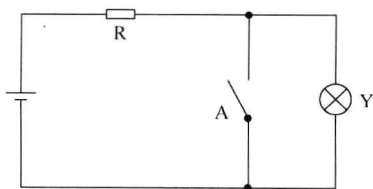


图 1-5 实现非逻辑的电路图

表 1-5 开关 A 状态及灯 Y 状态之间的关系

开关 A	灯 Y
断开	亮
闭合	不亮

如果规定开关闭合为“1”状态，开关断开为“0”状态，灯亮为“1”状态，灯灭为“0”状态，则把表 1-5 中的“断开”和“不亮”换为“0”，“闭合”和“亮”换为“1”，就是下面的非逻辑真值表。关于“非”逻辑，我们也可以归纳出以下几点。

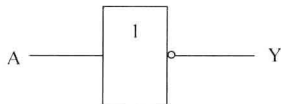
1) 非逻辑定义 条件具备时，事件不能发生；条件不具备时事件一定发生。这种决定事件的因果关系称为“非逻辑关系”。

2) 非逻辑真值表

A	Y
0	1
1	0

3) 非逻辑函数式 $Y = A'$ (或 $Y = \overline{A}$)。用上画线“—”和“'”均可表示非逻辑。本书大多采用“'”。

4) 非逻辑符号



5) 非逻辑运算 $\overline{0} = 1$; $\overline{1} = 0$ 。

1.3 常用的复合门电路

任何复杂的逻辑运算都可以由以上 3 种基本运算组合而成。常用的几种逻辑运算分别是与非逻辑运算、或非逻辑运算、异或逻辑运算、同或逻辑运算和与或非逻辑运算。实现这 5 种运算的电路称为与非门 (NAND Gate)、或非门 (NOR Gate)、异或门 (Exclusive-OR Gate)、同或门 (Exclusive-NOR Gate) 和与或非门 (AND-NOR Gate)。其实，与非门由与门和非门组合而成；或非门由或门和非门组合而成；与或非门由与门、或门和非门组合而成；异或门是或门的变形，同或门 (又叫异或非门) 是异或门和非门的结合。

1. 与非逻辑运算 (NAND)

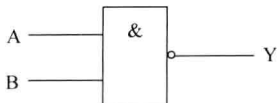
1) 与非逻辑定义 当决定某一事件的所有条件都具备时，事件不发生。这种决定事件的因果关系称为“与非逻辑关系”。

2) 与非逻辑真值表

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

3) 与非逻辑函数式 $Y = (A \cdot B)'$ 。

4) 与非逻辑符号



5) 与非逻辑运算 $(0 \cdot 0)' = 1$; $(0 \cdot 1)' = 1$; $(1 \cdot 0)' = 1$; $(1 \cdot 1)' = 0$ 。

2. 或非运算 (NOR)

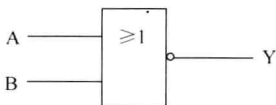
1) 或非逻辑定义 当决定某一事件的一个或多个条件满足时，事件便不发生。这种决定事件的因果关系称为“或非逻辑关系”。

2) 或非逻辑真值表

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

3) 或非逻辑函数式 $Y = (A + B)'$ 。

4) 或非逻辑符号



5) 或非逻辑运算 $(0 + 0)' = 1$; $(0 + 1)' = 0$; $(1 + 0)' = 0$; $(1 + 1)' = 0$ 。

3. 异或运算 (Exclusive-OR)

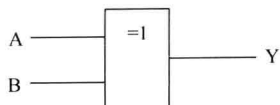
1) 异或逻辑定义 当决定某一事件的所有条件都不相同时，事件才能发生。这种决定事件的因果关系称为“异或逻辑关系”。

2) 异或逻辑真值表

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

3) 异或逻辑函数式 $Y = A \oplus B$ 。

4) 异或逻辑符号



5) 异或逻辑运算 $0 \oplus 0=0$; $0 \oplus 1=1$; $1 \oplus 0=1$; $1 \oplus 1=0$ 。

4. 同或运算 (Exclusive-NOR)

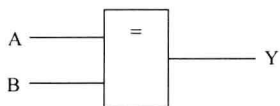
1) 同或逻辑定义 当决定某一事件的所有条件都相同时, 事件才能发生。这种决定事件的因果关系称为“同或逻辑关系”。

2) 同或逻辑真值表

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

3) 同或逻辑函数式 $Y=A \odot B$ 。

4) 同或逻辑符号



5) 同或逻辑运算 $0 \odot 0=1$; $0 \odot 1=0$; $1 \odot 0=0$; $1 \odot 1=1$ 。

5. 与或非运算 (AND-OR-NOT)

1) 与或非逻辑定义 “与”、“或”和“非”的复合运算称为与或非运算。

2) 与或非逻辑真值表

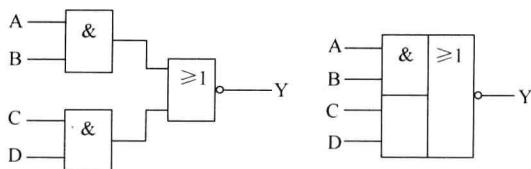
A	B	C	D	Y
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1

续表

A	B	C	D	Y
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

3) 与或非逻辑函数式 $Y = (A \cdot B + C \cdot D)'$ 。

4) 与或非逻辑符号



1.4 基本逻辑门电路

能够实现逻辑运算的电路称为逻辑门电路。逻辑门电路可以用分立元件电路实现，也可以用 TTL 集成电路实现，还可以用 CMOS 集成电路实现。逻辑门电路用分立元件电路实现时，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。基本逻辑门电路包括与门、或门、非门、与非门、或非门、异或门、同或门和与或非门等门电路。下面我们介绍与门、或门、非门、与非门、TTL 与非门五种门电路。

1. 二极管与门电路

二极管与门电路如图 1-6 所示，其工作情况如下。

- $V_A = V_B = 0V$ 时， VD_1 和 VD_2 都导通，则 $V_L \approx 0V$ 。
- $V_A = 0V$ ， $V_B = 5V$ 时， VD_1 导通，则 $V_L \approx 0V$ ， VD_2 受反向电压而截止。
- $V_A = 5V$ ， $V_B = 0V$ 时， VD_2 导通，则 $V_L \approx 0V$ ， VD_1 受反向电压而截止。
- $V_A = V_B = 5V$ 时， VD_1 和 VD_2 都截止， $V_L = V_{CC} = 5V$ 。

将上述结果进行归纳，按正逻辑体制（高电平为逻辑 1，低电平为逻辑 0 的体制），很容易看出该电路实现的逻辑运算为 $L = A \cdot B$ 。增加一个输入端和一个二极管，就可变成三输入端与门。以此类推，可构成更多输入端的与门。

2. 二极管或门电路

二极管或门电路如图 1-7 所示。同理可分析出，该电路实现的逻辑运算为 $L = A + B$ 。同样，可用增加输入端和二极管的方法，构成更多输入端的或门。