



工业和信息化部高职高专“十二五”规划教材立项项目

高等职业教育电子技术技能培养规划教材

Gaodeng Zhiye Jiaoyu Dianzi Jishu Jishu Peiyang Guihua Jiaocai

数字电子技术 基础 (第2版)

焦素敏 主编 范艳峰 刘林芝 副主编



Fundamentals of Digital Electronic Technology (2nd Edition)

内容由浅入深，知识衔接紧凑
理论简明扼要，突出能力培养
实用资料丰富，案例典型真实



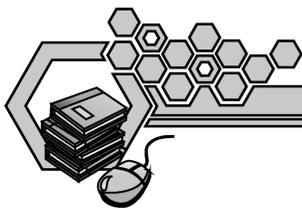
人民邮电出版社
POSTS & TELECOM PRESS

工业和信息化高职高专“十二五”规划教材立项项目
高等职业教育电子技术技能培养规划教材

数字电子技术基础 (第2版)

焦素敏 主编
范艳峰 刘林芝 副主编

人民邮电出版社
北京



第 2 版前言

本书是根据国家教育部最新制定的高职高专教育数字逻辑电路课程教学的基本要求和高职高专人才培养的规格和特点,并结合现代数字电子技术的发展趋势而编写的。

本书的主要内容有数字电子技术理论基础、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、脉冲波形的产生与变换、数模和模数转换器、存储器及可编程逻辑器件和数字电路 EDA 简介等。本书各章还配有本章小结、自我检测题与参考答案、思考题与习题、技能训练、综合训练、实用资料速查、读图练习等内容。

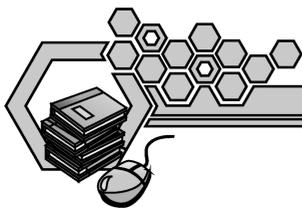
本书在内容及章节编排上,充分考虑高职高专教育电子、通信、电气及计算机各专业的需要,以够用和实用为教学改革方向,删去了繁琐的理论推导过程,侧重基本分析方法、设计方法和集成电路芯片的应用。在注重基本概念和基础理论的同时,更强调应用能力的培养,将案例教学融入本书的编写中,每章增加了大量的自我检测题及参考答案、技能训练和实用资料速查、读图练习和综合训练等内容,使读者能够很快地把理论与实际应用紧密结合起来,既能帮助提高读者的理解能力,又能培养读者的学习兴趣。此外,PLD 及数字电路 EDA 的简介,使读者在传统数字电路的基础上,对现代电子技术的发展方向——EDA 技术有一个简单的了解并能够快速入门。全书知识衔接紧凑,叙述通俗易懂,对数字电子技术的相关专业术语给出了英文表示,适合作为高职高专教育电子、通信、电气及计算机等专业的教材,也适于成人自学和职业技术培训使用。

本书由河南工业大学焦素敏编写第 1 章、第 4 章、第 5 章、第 8 章和第 9 章,刘林芝编写第 2 章、第 3 章,范艳峰编写第 6 章、第 7 章。全书由焦素敏组织、统稿和定稿,并担任主编。

由于时间仓促加之编者水平有限,书中不妥和错误之处在所难免,恳请读者批评指正。

编 者

2012 年 1 月

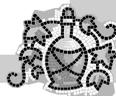


目 录

第 1 章 数字电子技术理论基础	1
1.1 数字电路概述	1
1.1.1 数字信号与数字电路	1
1.1.2 数字电路的特点	1
1.2 数制和码制	2
1.2.1 数制	2
1.2.2 数制转换	3
1.2.3 码制	4
1.3 逻辑函数及其表示方法	6
1.3.1 逻辑代数	6
1.3.2 3 种基本逻辑运算	6
1.3.3 常用的复合逻辑运算	8
1.3.4 逻辑函数的表示方法及相互转换	10
1.4 逻辑代数的基本定律和规则	12
1.4.1 逻辑代数的基本定律	12
1.4.2 逻辑代数的基本规则	13
1.5 逻辑函数的公式化简法	14
1.5.1 逻辑函数的不同表达方式	14
1.5.2 逻辑函数的公式化简法	14
1.6 逻辑函数的卡诺图化简法	15
1.6.1 逻辑函数的最小项及其表达式	15
1.6.2 逻辑函数的卡诺图表示法	17
1.6.3 用卡诺图化简逻辑函数	18
1.7 具有无关项的逻辑函数及其化简	20
1.7.1 逻辑函数中的约束项	20
1.7.2 利用无关项化简逻辑函数	21
本章小结	22
自我检测题	22
习题	24
第 2 章 逻辑门电路	27
2.1 二极管和三极管的开关特性	27
2.1.1 二极管的开关特性	27



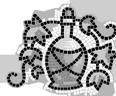
2.1.2 三极管的开关特性	28
2.2 基本逻辑门电路	30
2.2.1 3种基本门电路	30
2.2.2 DTL与非门	31
2.3 TTL逻辑门电路	32
2.3.1 TTL与非门的工作原理	32
2.3.2 TTL与非门的外特性及有关参数	33
2.4 其他类型的TTL门电路	37
2.4.1 集电极开路与非门(OC门)	38
2.4.2 三态门(TS门)	40
2.4.3 TTL与或非门和异或门	41
2.5 CMOS反相器门电路	42
2.5.1 MOS管的开关特性	42
2.5.2 CMOS反相器	43
2.6 其他CMOS门电路	44
2.6.1 CMOS与非门	44
2.6.2 CMOS或非门	44
2.6.3 CMOS传输门(TG门)	44
2.7 正负逻辑问题	45
2.8 门电路在实际应用中应注意的问题	46
2.8.1 多余输入端的处理	46
2.8.2 TTL和CMOS电路外接负载问题	47
2.8.3 TTL与CMOS电路的接口技术	47
技能训练 集成门电路逻辑功能的测试	48
实用资料速查:集成门电路相关资料	49
本章小结	52
自我检测题	52
习题	53
第3章 组合逻辑电路	55
3.1 组合逻辑电路的分析方法和设计方法	55
3.1.1 组合逻辑电路的基本概念	55
3.1.2 组合逻辑电路的分析方法	56
3.1.3 组合逻辑电路的设计方法	57
3.2 编码器	59
3.2.1 编码器的原理和分类	59
3.2.2 集成编码器	62
3.3 译码器和数据分配器	64
3.3.1 译码器的原理及分类	64
3.3.2 集成译码器	67



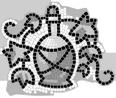
3.3.3 数据分配器	71
3.4 数据选择器	72
3.4.1 数据选择器的原理	72
3.4.2 集成数据选择器	74
3.5 数值比较器	75
3.5.1 数值比较器的原理	75
3.5.2 集成数值比较器	77
3.6 算术运算电路	78
3.6.1 半加器和全加器	78
3.6.2 集成算术运算电路	79
3.7 组合逻辑电路中的竞争与冒险	79
3.7.1 产生竞争冒险的原因	80
3.7.2 冒险的消除方法	81
技能训练 1 组合逻辑电路的设计与测试	82
技能训练 2 译码器的使用	83
技能训练 3 编码器、显示译码器及数字显示电路	84
技能训练 4 数据选择器	86
实用资料速查:常用组合逻辑电路功能部件相关资料	87
本章小结	90
自我检测题	91
习题	92
第 4 章 触发器	94
4.1 触发器的电路结构及工作原理	94
4.1.1 基本 RS 触发器	94
4.1.2 同步 RS 触发器	96
4.1.3 主从触发器和边沿触发器	100
4.2 触发器的功能分类及相互转换	104
4.2.1 触发器的功能分类	104
4.2.2 不同类型时钟触发器的相互转换	106
4.2.3 集成触发器及主要参数	108
技能训练 触发器	111
本章小结	112
自我检测题	113
习题	114
第 5 章 时序逻辑电路	118
5.1 时序逻辑电路的基本概念	118
5.2 时序逻辑电路的分析方法和设计方法	119
5.2.1 同步时序逻辑电路的分析	120



5.2.2	异步时序逻辑电路的分析	122
5.2.3	同步时序逻辑电路的设计方法	123
5.3	寄存器和锁存器	125
5.3.1	数码寄存器	125
5.3.2	移位寄存器	126
5.3.3	锁存器	128
5.3.4	寄存器集成电路介绍	129
5.4	计数器	131
5.4.1	二进制计数器	131
5.4.2	十进制计数器	134
5.4.3	集成计数器介绍	138
5.5	节拍脉冲发生器	143
	技能训练 集成计数器及应用	144
	实用资料速查:常用时序逻辑电路功能部件相关资料	145
	本章小结	146
	自我检测题	147
	习题	149
第6章	脉冲波形的产生与变换	151
6.1	555 定时器	151
6.2	多谐振荡器	153
6.2.1	由 555 定时器组成的多谐振荡器	153
6.2.2	石英晶体多谐振荡器	154
6.2.3	多谐振荡器的应用	156
6.3	单稳态触发器	156
6.3.1	由 555 定时器组成的单稳态触发器	156
6.3.2	集成单稳态触发器	157
6.3.3	单稳态触发器的应用	159
6.4	施密特触发器	161
6.4.1	由门电路组成的施密特触发器	161
6.4.2	由 555 定时器构成的施密特触发器	162
6.4.3	集成施密特触发器	163
6.4.4	施密特触发器的应用	164
	技能训练 555 时基电路	165
	读图练习 ASCII 键盘编码电路	166
	综合训练 数字钟的设计与实现	170
	本章小结	172
	自我检测题	173
	习题	174



第 7 章 数模和模数转换器	175
7.1 D/A 转换器	175
7.1.1 二进制权电阻网络 D/A 转换器	175
7.1.2 $R-2RT$ 型网络 D/A 转换器	177
7.1.3 D/A 转换器的主要技术参数	179
7.1.4 集成 D/A 转换器	179
7.2 A/D 转换器	182
7.2.1 概述	182
7.2.2 常用的 A/D 转换器类型	183
7.2.3 集成 A/D 转换器及其应用	187
技能训练 1 模数转换器 ADC0809	190
技能训练 2 数模转换器 DAC0832	191
读图练习 3 位半数字电压表	193
本章小结	198
自我检测题	198
习题	199
第 8 章 半导体存储器及可编程逻辑器件	201
8.1 随机存取存储器(RAM)	201
8.1.1 RAM 的结构和工作原理	201
8.1.2 RAM 的存储元	203
8.1.3 RAM 的扩展	204
8.2 只读存储器(ROM)	205
8.2.1 ROM 的结构和工作原理	205
8.2.2 ROM 的扩展	209
8.3 可编程逻辑器件(PLD)	210
8.3.1 概述	210
8.3.2 PAL 和 GAL	211
* 8.3.3 CPLD/FPGA 简介	216
本章小结	223
自我检测题	224
习题	225
* 第 9 章 数字电路 EDA 简介	226
9.1 VHDL 入门	226
9.1.1 组合逻辑电路设计举例	227
9.1.2 时序逻辑电路设计举例	233
9.2 EDA 工具软件 MAX+plus II 使用入门	236
9.2.1 原理图输入设计方法	236



9.2.2 文本编辑——VHDL 设计	244
本章小结	252
习题	253
参考文献	254

第1章

数字电子技术理论基础

数字电路是以数字量为研究对象的电子电路。本章主要讨论数字电子技术的基础理论知识,包括数制和码制,逻辑代数及其化简。同时,还给出了逻辑函数的概念、表示方法及相互转换。

1.1 数字电路概述

1.1.1 数字信号与数字电路

电子电路中的信号分为两类。一类在时间和幅度上都是连续的,称为模拟信号,如图 1.1 所示,如电压、电流、温度、声音等信号。传送和处理模拟信号的电路称为模拟电路(Analog Circuit)。另一类在时间和幅度上都是离散的,称为数字信号,如图 1.2 所示,如计时装置的时基信号、灯光闪烁等信号都属于数字信号。传送和处理数字信号的电路称为数字电路(Digital Circuit)。

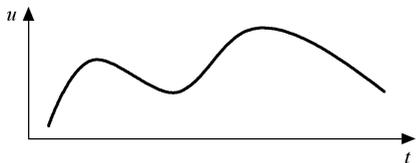


图 1.1 模拟信号

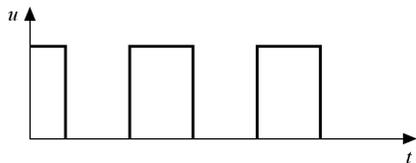
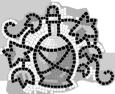


图 1.2 数字信号

1.1.2 数字电路的特点

数字电路与模拟电路相比具有以下特点。

- ① 数字电路的工作信号是离散的数字信号。数字信号常用 0、1 二元数值表示。



- ② 数字电路中,半导体器件均工作在开关状态,即工作在截止区和饱和区。
- ③ 数字电路研究的主要问题是输入、输出之间的逻辑关系。
- ④ 数字电路的主要分析工具是逻辑代数。

1.2 数制和码制

1.2.1 数制

数制即指计数的方法,日常生活中最常用的是十进制计数,而在数字电路和计算机中最常用的是二进制、八进制和十六进制。

1. 十进制数

在十进制数中,每一位都采用0~9共10个数码中的任何一个来表示,所以十进制的计数基数是10,超过9就必须用多位数来表示。其相邻的低位和高位间的运算关系是“逢十进一”,即

$$9 + 1 = 10$$

数码处在不同位置时,所代表的数值是不同的。例如:

$$5555 = 5 \times 10^3 + 5 \times 10^2 + 5 \times 10^1 + 5 \times 10^0$$

式中, 10^3 、 10^2 、 10^1 、 10^0 称为十进制数各数位的权或位权,都是10的幂。因此,任意一个十进制数都可以表示为各个数位上的数码与其对应的权的乘积之和,称为权展开式,用通式可表示为

$$(N)_{10} = a_{n-1} \times 10^{n-1} + a_{n-2} \times 10^{n-2} + \dots + a_1 \times 10^1 + a_0 \times 10^0 + a_{-1} \times 10^{-1} + a_{-2} \times 10^{-2} + \dots + a_{-m} \times 10^{-m} = \sum_{i=-m}^{n-1} a_i \times 10^i$$

式中, a_i 为0~9中的任一数码;10为进制的基数;10的*i*次幂为第*i*位的权; m 、 n 为正整数, n 为整数部分的位数, m 为小数部分的位数。

2. 二进制数

二进制计数体制中只有0和1两个数码,其基数是2,运算规律是“逢二进一”,即

$$1 + 1 = 10$$

二进制数同样也可按权展开,用通式可表示为

$$(N)_2 = \sum_{i=-m}^{n-1} b_i \times 2^i$$

例如:

$$(101.01)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = (5.25)_{10}$$

上式中用下标2和10分别表示括号里的数是二进制数和十进制数。

3. 八进制数

八进制数有0~7共8个数码,计数基数是8,运算规律是“逢八进一”,即

$$7 + 1 = 10$$

八进制数中每个数位的权都是8的幂。例如:

$$(207.04)_8 = 2 \times 8^2 + 0 \times 8^1 + 7 \times 8^0 + 0 \times 8^{-1} + 4 \times 8^{-2} = (135.0625)_{10}$$



4. 十六进制数

二进制数在计算机系统中处理很方便,但当位数较多时,书写及记忆都比较难,为了减少位数,通常将二进制数用十六进制来表示,它是计算机系统中除二进制数之外使用较多的进制。十六进制中有0~9,A(10),B(11),C(12),D(13),E(14),F(15)共16个不同的数码,计数基数是16,运算规律是“逢十六进一”,即

$$F + 1 = 10$$

十六进制数中每个数位的权都是16的幂。例如:

$$(D8.A)_{16} = 13 \times 16^1 + 8 \times 16^0 + 10 \times 16^{-1} = (216.625)_{10}$$

1.2.2 数制转换

1. 十进制数与二进制数的相互转换

(1) 二进制数转换成十进制数

二进制数转换成十进制数的方法是按权展开,再求加权系数之和。

【例 1.1】 将二进制数 $(1101010)_2$ 转换成十进制数。

$$\begin{aligned} \text{解: } (1101010)_2 &= 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ &= 2^6 + 2^5 + 2^3 + 2^1 \\ &= 64 + 32 + 8 + 2 \\ &= (106)_{10} \end{aligned}$$

(2) 十进制数转换为二进制数

十进制数转换为二进制数时,对整数部分可采用“除2取余、逆序排列”法,对小数部分可采用“乘2取整、顺序排列”法。

【例 1.2】 将十进制数 $(44.375)_{10}$ 转换成二进制数。

解:可将 $(44.375)_{10}$ 的整数部分和小数部分分别进行转换,步骤如下:

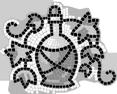
整数部分			小数部分				
2	44	余数	低位	0.375		整数	高位
2	22 0	↑	× 2			
2	11 0		0.750	0	
2	5 1		0.750			
2	2 1		× 2		1	
2	1 0		1.500		
2	0 1		0.500			
			× 2		1		↓ 低位
			1.000			

故 $(44.375)_{10} = (101100.011)_2$

2. 十进制数与其他进制数的相互转换

十进制数和其他进制数的相互转换与十进制数和二进制数的相互转换方法完全类似。

把十进制数转换为其他进制数时,可将十进制数分为整数和小数两部分进行。整数部分的转



换采用“除基取余,逆序排列”法。小数部分的转换采用“乘基取整,顺序排列”法。

把其他进制数转换为十进制数时,可将其他进制数按加权系数展开式展开,求得和即为相应的十进制数。

3. 二进制数与八进制数的相互转换

(1) 二进制数转换为八进制数

二进制数转换为八进制数时,可将二进制数由小数点开始,整数部分向左,小数部分向右,每3位分成一组,不够3位补零,则每组二进制数便是一位八进制数。

【例 1.3】 将二进制数 $(1101010.1101)_2$ 转换为八进制数。

解: $(1101010.1101)_2 = (001,101,010.110,100)_2 = (152.64)_8$

(2) 八进制数转换为二进制数

八进制数转换为二进制数时,只要将每位八进制数用3位二进制数表示即可。

【例 1.4】 将八进制数 $(207.04)_8$ 转换为二进制数。

解: $(207.04)_8 = (010,000,111.000,100)_2$

4. 二进制数与十六进制数的相互转换

(1) 二进制数转换为十六进制数

二进制数转换为十六进制数时,只要将二进制数的整数部分自右向左每4位一组,不足4位时在左边补零;小数部分则自左向右每4位一组,最后不足4位时在右边补零。再把每4位二进制数对应的十六进制数写出来即可。

【例 1.5】 将二进制数 $(1101010.1101)_2$ 转换为十六进制数。

解: $(1101010.1101)_2 = (0110,1010.1101)_2 = (6A.D)_{16}$

(2) 十六进制数转换为二进制数

十六进制数转换为二进制数时正好与(1)所述相反,只要将每位的十六进制数对应的4位二进制写出来就行了。

在数制使用时,常将各种数制用简码来表示:如十进制(Decimal)数用D表示或省略;二进制(Binary)用B来表示;八进制(Octal)用O来表示;十六进制(Hexadecimal)数用H来表示。如:十进制数123表示为123D或者123;二进制数1011表示为1011B;八进制数173表示为173O;十六进制数3A4表示为3A4H。

1.2.3 码制

数码不但可以用来表示数量的大小,还可以用来表示不同的事物。当用数码作为代号表示不同的事物时,称其为代码(Code)。一定的代码有一定的规则,这些规则称为码制。给不同事物赋予一定代码的过程称为编码。

日常生活中,人们习惯于十进制数码,而数字系统只能对二进制代码进行处理,这就需要用4位二进制数来表示一位十进制数,这种用来表示十进制数的4位二进制代码称为二-十进制代码(Binary Coded Decimal),简称BCD码。由于4位二进制数有 $2^4 = 16$ 种组合方式,可任选其中10种来表示0~9这10个数码,因此编码方案很多。常见的BCD码有以下几种。



1. 8421 码

8421 码是 BCD 码中使用最多的一种有权码(每位均有固定权值),其权值由高到低依次为 $8(2^3)$ 、 $4(2^2)$ 、 $2(2^1)$ 、 $1(2^0)$,故称 8421BCD 码。8421BCD 码的特点是,如果将代码看成是一个 4 位二进制数,则它的数值正好等于它所代表的十进制数的大小。即假设 8421 码为 $a_3a_2a_1a_0$,则其表示的十进制数为

$$8a_3 + 4a_2 + 2a_1 + 1a_0$$

【例 1.6】 将 $(35)_{10}$ 和 $(79.4)_{10}$ 分别用 8421 码表示。

解: $(35)_{10} = (0011\ 0101)_{8421}$

$(79.4)_{10} = (0111\ 1001.0100)_{8421}$

2. 2421 码

2421 码也是一种有权码,其权值由高到低依次为 2、4、2、1,假设 2421 码为 $a_3a_2a_1a_0$,则其表示的十进制数为

$$2a_3 + 4a_2 + 2a_1 + 1a_0$$

3. 5421 码

5421 码也是一种有权码,其权值由高到低依次为 5、4、2、1,假设 5421 码为 $a_3a_2a_1a_0$,则其表示的十进制数为

$$5a_3 + 4a_2 + 2a_1 + 1a_0$$

4. 余 3 码

余 3 码各位没有固定的权值,是一种无权代码。它是对相应的 8421 码加 0011 得到的,因此叫做余 3 码。

5. 格雷(Gray)码

格雷码也叫循环码,它也是一种无权码。格雷码的特点是,任何两个相邻的代码只有一位不同,其他位都相同。这样,在代码转换过程中就不会产生过渡“噪声”。

上述几种常用的二-十进制编码如表 1.1 所示。

表 1.1 几种常用的二-十进制编码

十进制数	8421 码	2421 码	5421 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



续表

十进制数	8421 码	2421 码	5421 码	余 3 码	格 雷 码
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,国际上还有一些专门处理字母、数字字符的二进制代码,如 ASCII 等。读者可参阅有关书籍。

1.3 逻辑函数及其表示方法

1.3.1 逻辑代数

逻辑代数(Logic Algebra)又叫布尔代数或开关代数,是由英国数学家乔治·布尔于 1847 年创立的。逻辑代数与普通代数都由字母来代替变量,但逻辑代数与普通代数的概念不同,它不表示数量大小之间的关系,而是描述客观事物之间逻辑关系的一种数学方法。逻辑变量的取值只有两种,即逻辑 0 和逻辑 1,它们并不表示数量的大小,而是表示两种对立的逻辑状态,如开关的通与断、电位的高与低、灯的亮与灭等。0 和 1 称为逻辑常量。

例如,在图 1.3 所示的指示灯控制电路中,我们用字母 Y 表示指示灯,用 A、B 表示两个开关。指示灯 Y 的亮与灭两种状态取决于开关 A、B 的通断状态。我们将 A、B 称为输入逻辑变量,将 Y 称为输出逻辑变量。

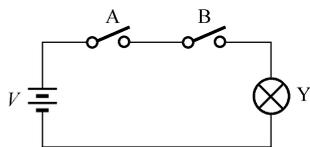


图 1.3 指示灯控制电路

逻辑代数有两种逻辑体制,其中,正逻辑体制规定,高电平为逻辑 1,低电平为逻辑 0;负逻辑体制规定,低电平为逻辑 1,高电平为逻辑 0。

1.3.2 3 种基本逻辑运算

在逻辑代数中有 3 种基本的逻辑运算:与运算、或运算、非运算。

1. 与运算

只有当决定一件事情的所有条件都具备时,这件事情才会发生,这种因果关系称为“与”(and)逻辑运算。例如,在图 1.3 所示电路中,两个开关串联控制一个指示灯。显然,只有当两个开关都接通时,灯才能亮,否则,灯灭。该电路的与逻辑关系如表 1.2 所示。

如果用 1 表示开关闭合和灯亮,用 0 表示开关断开和灯灭;则电路中指示灯 Y 和开关 A、B 之间的关系如表 1.3 所示,这种反映逻辑关系的表格称为逻辑真值表。

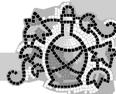


表 1.2 与逻辑关系表

开关 A	开关 B	灯 Y
断	断	灭
断	通	灭
通	断	灭
通	通	亮

表 1.3 与逻辑真值表

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

在逻辑代数中,与逻辑运算又叫逻辑乘,两变量的与运算可用逻辑表达式表示为

$$Y = A \cdot B$$

读做“Y 等于 A 与 B”。意思是:若 A、B 均为 1,则 Y 为 1;否则 Y 为 0。与运算规则可以归纳为“有 0 出 0,全 1 为 1”。

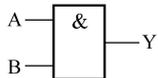
在数字电路中,实现与逻辑关系的逻辑电路称为与门,其逻辑电路符号如图 1.4 所示,其中(a)图是国际惯用符号、(b)图是国标符号。本书以后采用国际惯用符号。

2. 或运算

当决定事件发生的条件具备一个或一个以上时,事件就发生;只有当所有条件均不具备时,事件才不会发生。这种因果之间的关系就是“或”(or)逻辑的运算关系。例如,在图 1.5 所示的电路中,只要开关 A、B 中任意一个接通或者两个都接通,灯就亮;只有当开关 A、B 均断开时,灯才不亮。由此可得或逻辑关系表如表 1.4 所示,或逻辑真值表如表 1.5 所示。



(a) 特定外形符号



(b) 矩形轮廓符号

图 1.4 与逻辑电路符号

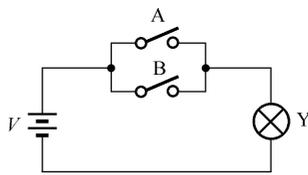


图 1.5 或逻辑关系电路

表 1.4 或逻辑关系表

开关 A	开关 B	灯 Y
断	断	灭
断	通	亮
通	断	亮
通	通	亮

表 1.5 或逻辑真值表

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

在逻辑代数中,或逻辑运算又叫逻辑加,两变量的或运算可用逻辑表达式表示为

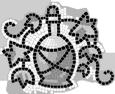
$$Y = A + B$$

读做“Y 等于 A 或 B”,意思是:若 A、B 均为 0,则 Y 为 0;否则 Y 为 1。或运算规则可以归纳为“全 0 出 0,有 1 为 1”。

在数字电路中,实现或逻辑关系的逻辑电路称为或门,其逻辑电路符号如图 1.6 所示。

3. 非运算

非(not)运算关系是,当条件具备时,事件不发生;当条件不具备时,事件能发生。即某事件发



生与否,仅取决于一个条件,而且是对该条件的否定。

例如,在图 1.7 所示电路中,当开关 A 接通时,灯 Y 不亮;而当开关 A 断开时,灯 Y 亮。由此可得非逻辑关系表和非逻辑真值表,如表 1.6 和表 1.7 所示。

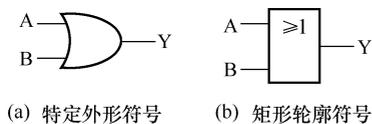


图 1.6 或逻辑电路符号

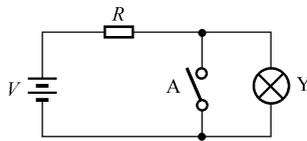


图 1.7 非逻辑关系电路

表 1.6 非逻辑关系表

开关 A	灯 Y
断	亮
通	灭

表 1.7 非逻辑真值表

A	Y
0	1
1	0

在逻辑代数中,非逻辑运算又称逻辑反。非逻辑关系的表达式为

$$Y = \bar{A}$$

读做“Y 等于 A 非”,意思是:若 A 为 0,则 Y 为 1;若 A 为 1,则 Y 为 0。非逻辑运算规则可以归纳为“有 0 出 1,是 1 为 0”。非逻辑电路符号如图 1.8 所示。

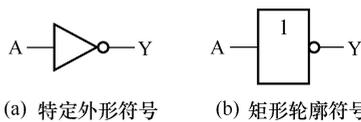


图 1.8 非逻辑电路符号

1.3.3 常用的复合逻辑运算

复合逻辑是指由与、或、非 3 种基本逻辑关系组合而成的逻辑关系。常用的复合逻辑运算主要包括与非、或非、与或非、异或、同或等。

1. 与非

与非逻辑运算是由与、非两种基本运算按照“先与后非”的顺序复合而成的。两变量与非逻辑的逻辑表达式为

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

两变量与非逻辑真值表如表 1.8 所示,与非逻辑符号如图 1.9 所示。对于与非逻辑,只有当其全部输入为 1 时,输出才为 0。

表 1.8 与非逻辑真值表

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

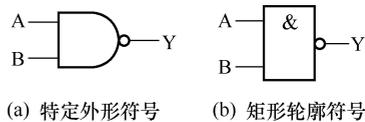


图 1.9 与非逻辑符号