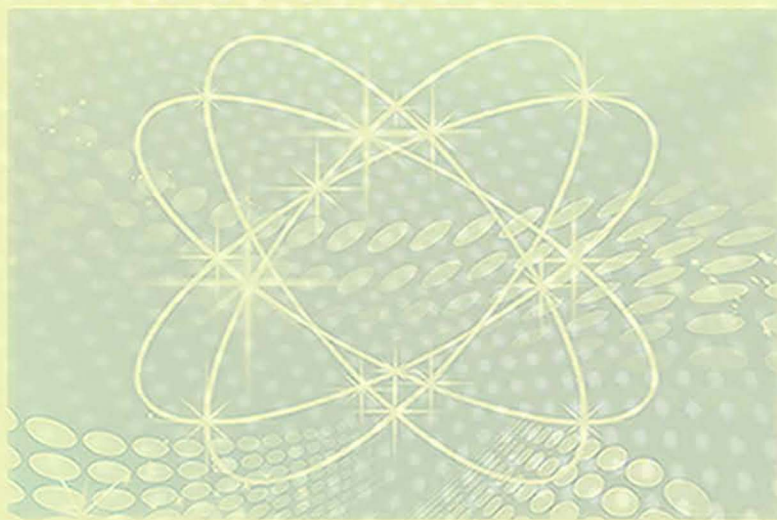


电子技术与实践

杨悦梅 主编



电子科技大学出版社

图书在版编目 (CIP) 数据

电子技术与实践 / 杨悦梅主编. —成都: 电子科技大学出版社, 2015. 6

ISBN 978-7-5647-3054-3

I. ①电… II. ①杨… III. ①电子技术—高等学校—教材 IV. ①TN

中国版本图书馆 CIP 数据核字 (2015) 第 131694 号

电子技术与实践

主 编 杨悦梅

出 版: 电子科技大学出版社 (成都市一环路东一段 159 号电子信息产业大厦 邮编: 610051)
策划编辑: 谢晓辉
责任编辑: 谢晓辉
主 页: www.uestcp.com.cn
电子邮箱: uestcp@uestcp.com.cn
发 行: 新华书店经销
印 刷: 成
成品尺寸: 185 mm × 260 mm 印张 12.25 字数 306 千字
版 次: 2015 年 6 月第一版
印 次: 2015 年 6 月第一次印刷
书 号: ISBN 978-7-5647-3054-3
定 价: 32.00 元

■ 版权所有 侵权必究 ■

◆ 本社发行部电话: 028-83202463; 本社邮购电话: 028-83201495。

◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

前 言

作为电类专业的一门重要的专业基础课，数字电子技术是每个电类专业的必修课程，同时这方面的书籍和教材也琳琅满目、层出不穷。广大从事高职教学的教师越来越感到现有的教学用书与高职学生的认知特点和培养目标不适应，总希望找到一本适合高职学生的得心应手的教材来施教，让学生看得懂、教师教得顺，这也是我们编写这本教材所希望达到的目的。

结合教育部《关于全面提高高等职业教育教学质量的若干意见》（教高[2006]16号文件）精神，本着“以服务为宗旨、以就业为导向、以能力为本位”的制导思想，我们明确本教材的编写以就业为导向，根据行业专家和企业技术人员对电类专业涵盖的岗位群进行工作任务和职业能力的分析，以电类专业共同具备的岗位职业能力为依据，遵循学生认知规律，紧密结合职业资格证书中对电子技能所作的要求，确定项目模块和课程内容。我们借鉴德国的职业教育理念，以项目和真实产品的工作过程作为联系课程、学校和行业企业的纽带。以项目为导向，以任务为驱动，以完成项目工作任务为主线，链接相应的理论知识和技能实训，融“教、学、做”为一体。

本教材采用项目制，主要内容包括五个模块项目，即信号灯控制电路的制作、优先选择器电路的设计与制作、四人数显抢答器的制作、篮球竞赛 30S 计时器电路的设计与制作、变音警笛电路的制作。为了满足很多学校课程综合实训的要求，提供了三个综合实训项目，即数字电子钟的设计与制作、改进型抢答器的设计与制作、多路彩灯控制器的设计与制作。

本教材由杭州科技职业技术学院杨悦梅老师通稿并编写了项目 1、项目 2，袁炼红老师编写了项目 3、项目 4，沈国泉老师编写了项目 5，安婷婷老师编写了综合实训项目。在编写过程中也得到了奚亢基、龚大墉老师的大力支持，在此一并致以谢意。

由于编者水平有限，书中难免存在不尽如人意的地方，也可能存在一些缺点和错误，敬请各位读者批评指正。

编 者

2016 年 1 月于杭州

目 录

项目 1 信号灯的逻辑控制	1	
学习目标	1	
工作任务	1	
项目技能实训 信号灯的逻辑控制	1	
一、实训目的	1	
二、实训设备和器件	1	
三、实训内容与步骤	1	
四、实训注意事项	2	
五、实训电路分析与实训报告	2	
六、小结	3	
知识链接一 数制与编码的基础知识	4	
一、数制	4	
二、编码	7	
知识链接二 逻辑代数的基础知识	7	
一、逻辑代数基本运算	7	
二、逻辑变量与逻辑函数	10	
三、逻辑运算	10	
四、逻辑函数的表示方法	15	
习题 1	23	
项目 2 优先报警器电路的设计与制作实训	26	
学习目标	26	
工作任务	26	
项目技能实训 优先选择器电路的设计与制作实训	26	
一、实训目的	26	
二、实训设备和器件	26	
三、实训内容与步骤	26	
四、实训注意事项	27	
五、实训电路分析与实训报告	27	
知识链接一 基本逻辑门电路	29	
一、二极管与门电路	29	
二、二极管或门电路	30	◆

三、非门电路——三极管反相器	32
知识链接二 数字集成芯片	33
一、TTL 集成门电路	33
二、CMOS 集成门电路	39
三、集成逻辑门电路常见封装及引脚排列	45
技能实训 常用集成门电路的逻辑功能测试	46
一、实验目的	46
二、实验设备与器件	46
三、实验内容与步骤	47
四、实训注意事项	48
五、完成电路的详细分析及编写项目实训报告	48
习题 2	49
项目 3 四人数显抢答器的制作与调试	52
学习目标	52
工作任务	52
项目技能实训 四人数显抢答器的制作与调试	52
一、实训目的	52
二、实训设备和器件	52
三、实训内容与步骤	52
四、实训注意事项	54
五、实训电路分析与实训报告	54
知识链接一 三人表决器的设计与调试	56
一、组合逻辑电路的分析	56
二、组合逻辑电路的设计	58
三、三人表决器的设计与结果演示	60
知识链接二 半加器、全加器的设计与性能测试	62
一、半加器的逻辑功能与电路设计	62
二、全加器的逻辑功能与电路设计	62
技能实训 一位十进制减法计数器的设计与调试	63
一、编码器	63
二、译码器	68
三、计数器电路设计及结果演示	74
习题 3	75
项目 4 篮球竞赛 30s 计时器设计电路	78
学习目标	78
工作任务	78

项目技能实训 篮球竞赛 30s 计时器电路的设计与制作实训	78	
一、实训目的	78	
二、实训设备和器件	78	
三、实训内容与步骤	78	
四、实训注意事项	80	
五、实训电路分析与实训报告	80	
知识链接一 触发器的基础知识	83	
一、基本 RS 触发器	83	
二、同步触发器	87	
三、主从触发器	90	
四、边沿触发器	94	
五、T 触发器和 T' 触发器	98	
六、触发器的相互转换	99	
知识链接二 常用集成触发器的产品介绍	100	
一、主从 RS 触发器 CT74LS71	100	
二、集成边沿 JK 触发器 CT74LS112	101	
三、集成维持阻塞 D 触发器 CT74LS74	102	
知识链接三 时序电路的基础知识	104	
一、时序逻辑电路的分析	104	
二、寄存器与移位寄存器	109	
三、计数器	114	
知识链接四 常用集成计数器产品的介绍	118	
一、集成计数器	118	
二、用中规模 N 进制集成计数器设计任意 (M) 进制计数器	123	
知识链接五 晶体振荡体的应用	131	
一、石英晶体振荡器的基本原理	131	
二、石英晶体振荡器的主要参数	132	
三、石英晶体振荡器的应用	132	
技能实训 集成计数器的应用	133	
一、实训目的	133	
二、实训设备和器件	133	
三、实训内容与步骤	133	
四、实验注意事项	134	
五、完成电路的详细分析及编写项目实训报告	134	
习题 4	135	目
项目 5 变音警笛电路的制作	141	录
学习目标	141	◆

工作任务	141
项目技能实训 变音警笛电路的制作	141
一、实训目的	141
二、实训设备和器件	141
三、实训电路与说明	141
四、实训电路的安装与调试	143
知识链接 555 集成定时器	143
一、电路组成	143
二、工作原理	144
三、555 定时器的应用举例	145
四、555 定时器专用软件介绍	150
技能实训 光电枪和靶的电路制作实训	153
一、实训目的	153
二、实训设别与器件	153
三、实训电路与说明	153
四、实训电路的安装与调试	154
习题 5	154
项目 6 综合实训	157
一、数字电子钟的制作	157
二、改进型抢答器的制作	166
三、多路彩灯控制器的设计与制作	171
附 录 常用数字集成电路引脚排列图	177

项目 1 信号灯的逻辑控制

学习目标

了解数字逻辑的概念，理解与、或、非三个基本逻辑关系；熟悉逻辑代数的基本定律、常用公式和逻辑运算法则；掌握逻辑函数的五种表示方法；掌握逻辑函数的公式化简和卡诺图化简方法；熟悉逻辑门电路的逻辑功能。

工作任务

1. 读懂信号灯逻辑控制电路的原理图，明确元器件连接和电路连线。
2. 画出布线图。
3. 完成电路所需元器件的选择与检测。
4. 制作电路。
5. 完成电路功能检测和故障排除。
6. 完成电路的详细分析及编写项目实训报告。

项目技能实训 信号灯的逻辑控制

一、实训目的

1. 了解逻辑控制的概念。
2. 掌握表示逻辑控制的基本方法。

二、实训设备和器件

发光二极管、限流电阻、继电器（两个）、直流电源、导线若干。

三、实训内容与步骤

图 1.1 为实训电路图。这是一个楼房照明灯的控制电路。设 A 、 B 分别代表上、下楼层的两个开关，发光二极管 F 代表照明灯。在楼上按下开关 A ，可以将照明灯打开，在楼下闭合开关 B ，又可以将灯关掉；反过来，也可以在楼下开灯，楼上关灯。

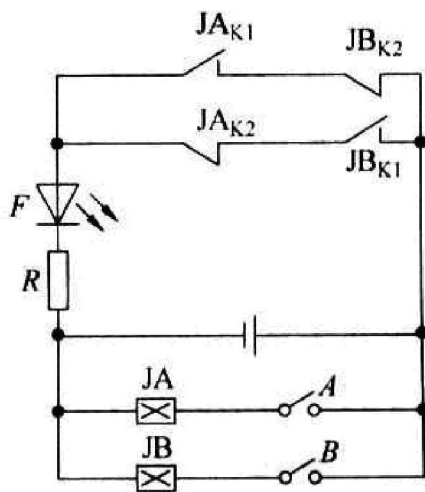


图 1.1 信号灯的逻辑控制电路

(1) 连接电路

按图 1.1 连接好电路，注意 JA、JB 两个继电器的开关不要接错。

(2) 实验开关和发光二极管的逻辑关系

接通电源，分别将开关 A、B 按表 1.1 的要求接通或者断开，观察发光二极管 F 的亮灭情况，并填入表 1.1 中。

表 1.1 电路功能测试表

开关 A	开关 B	发光二极管 F
断开	断开	
断开	闭合	
闭合	断开	
闭合	闭合	

四、实训注意事项

注意继电器引线不要接错，分清线圈端口和常开常闭端口，注意连线可靠。

五、实训电路分析与实训报告

根据上述电路，可做如下分析。

1. 图 1.1 中，JA 和 JB 分别代表继电器的两个线圈。JA_{K1}、JB_{K1} 代表继电器的常开触点，JA_{K2}、JB_{K2} 代表继电器的常闭触点。在项目图所示的状态下（开关 A、B 均断开），由于没有通路给发光二极管供电，所以发光二极管不亮；开关 A 闭合，继电器线圈 JA 通电，其常开触点 JA_{K1} 闭合，常闭触点 JA_{K2} 断开，JB_{K1}、JB_{K2} 则维持原来状态，此时图 1.1 最上面的一条电路连通，通过电源给发光二极管供电，发光二极管亮。同样道理，如果只闭合开关 B，也会给发光二极管构成通路使之点亮；当开关 A、B 均闭合时，由于没有通路，所以发光二极管不亮。

2. 发光二极管 F 的状态，我们称为输出，是由开关 A 、 B 来决定的，开关 A 、 B 称为输入。输出和输入是一种逻辑控制电路，输入量和输出量分别对应两种状态。

3. 从实验结果可以看出，当 A 、 B 同时闭合，或者同时断开，即处于相同状态时，二极管不亮；相反，当 A 、 B 处于不同状态时，发光二极管亮。如果定义开关闭合和灯亮为逻辑“1”，定义开关断开和发光二极管不亮为逻辑“0”，则 A 、 B 、 F 都可用两种逻辑状态“1”、“0”来描述，如表 1.2 所示。

表 1.2 逻辑功能表示

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

表 1.2 此时的“1”、“0”不代表任何数量的大小。表格的左边是两个输入状态的所有取值的组合，表格的右边是对应的输出状态。这样可以将表 1.1 重新用表 1.2 表示，这种表征逻辑事件输入和输出之间全部可能状态的表格称为逻辑事件的真值表。

4. 按电子实验室要求书写实训报告。

六、小结

1. 逻辑事件与逻辑控制

通过项目 1，我们初步认识了一个逻辑事件的控制电路。所谓逻辑，简单地说，就是表示事物的因果关系，即输入、输出之间变化的因果关系。而逻辑事件是这样的一类事物，它们具有如下共性：其存在或表现形式有且仅有两个相互对立的状态，而且它必定出现在这两个状态中的一个。例如：实训中的开关只有“闭合”和“断开”两种状态，而且开关的状态必为二者之一；发光二极管只有“亮”、“灭”两种对立状态。上述事件都是逻辑事件，又可以叫做逻辑量。

在现实生活中的一些实际关系，会使某些逻辑量的取值互相依赖，或互为因果。例如实训中开关的通断决定了发光二极管的亮、灭，反过来也可以从发光二极管的状态推出开关的相应状态，这样的关系称为逻辑控制。

在实际应用中，会遇到各种复杂的逻辑控制电路，但它们都是由基本的逻辑关系组成的。在数字电路中，有一些基本的逻辑控制电路，它们反映了这些基本的逻辑关系（又称逻辑运算）。这些基本的逻辑运算是构成各种复杂的逻辑电路的基础。

2. 表示逻辑控制的基本方法

不同的数码不仅可以表示数量的不同大小，而且还能用来表示不同的事物。在上述讨论的各类逻辑事件中，用 1 位二进制数码的 1 和 0 表示一个事物的两种不同逻辑状态。例如，可以用 1 和 0 分别表示一件事情的是和非、真和伪、有和无、好和坏，或者表示电

路的通和断、电灯的亮和暗、门的开和关等。

0 和 1 是二进制数的两个基本状态，它们是学习数字逻辑电路、计算机技术的基础。

知识链接一 数制与编码的基础知识

一、数制

数制是一种计数方法，它是计数进位制的简称。采用何种计数制方法应根据实际需要而定。日常生活中我们习惯用十进制，而在数字系统中进行数字的储存和处理采用的是二进制、八进制、十六进制。

我们先来看几个概念。

进位制：表示数时，仅用一位数码往往不够用，必须用进位计数的方法组成多位数码。多位数码每一位的构成以及从低位到高位进位的规则称为进位计数制，简称进位制。

基数：进位制的基数，就是在该进位制中可能用到的数码个数。

位权：在某一进位计数制的数中，每一位的大小都对应着该位上的数码乘上一个固定数，这个固定数就是这一位的权数。权数是一个幂。

1. 十进制 (Decimal)

日常生活中人们最习惯用的是十进制。十进制是以 10 为基数的计数制。在十进制中，有 0~9 共十个数码，它的进位规则是“逢十进一”。十进制的权展开式，如：

$$(12345)_{10}=1\times10^4+2\times10^3+3\times10^2+4\times10^1+5\times10^0$$

其中， 10^4 、 10^3 、 10^2 、 10^1 、 10^0 为万位、千位、百位、十位和个位的权，它们都是基数 10 的幂。同样的数码在不同的数位上代表的数值不同，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，称权展开式。又如：

$$(123.45)_{10}=1\times10^2+2\times10^1+3\times10^0+4\times10^{-1}+5\times10^{-2}$$

2. 二进制 (Binary)

数字电路中应用最广泛的是二进制。二进制是以 2 为基数的计数制。在二进制中，有 0、1 共两个数码，它的进位规则是“逢二进一”。二进制的权展开式，如：

$$(11011.01)_2=1\times2^4+1\times2^3+0\times2^2+1\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}$$

其中， 2^4 、 2^3 、 2^2 、 2^1 、 2^0 、 2^{-1} 、 2^{-2} 为各位的权，它们都是基数 2 的幂。

二进制只有 0 和 1 两个数码，它的每一位都可以用电子元件来实现，且运算规则简单，相应的运算电路也容易实现。

3. 八进制 (Octal)

八进制是以 8 为基数的计数制。在八进制中，有 0~7 共八个数码，它的进位规则是“逢八进一”。八进制的权展开式，如：

$$(1237.45)_8=1\times8^3+2\times8^2+3\times8^1+7\times8^0+4\times8^{-1}+5\times8^{-2}$$

4. 十六进制 (Hexadecimal)

十六进制是以 16 为基数的计数制。在十六进制中, 有 0~9、A~F 共 16 个数码, 它的进位规则是“逢十六进一”。十六进制的权展开式, 如:

$$(F81.A)_{16}=15 \times 16^2+8 \times 16^1+1 \times 16^0+10 \times 16^{-1}$$

5. 二进制、八进制、十六进制转换成十进制

分别写出二进制、八进制、十六进制数按位权展开式, 数码与位权的乘积称为加权系数。各位加权系数相加的结果便为对应的十进制数。如:

$$(11011.01)_2=1 \times 2^4+1 \times 2^3+0 \times 2^2+1 \times 2^1+1 \times 2^0+0 \times 2^{-1}+1 \times 2^{-2}=(27.25)_{10}$$

$$(1237.45)_8=1 \times 8^3+2 \times 8^2+3 \times 8^1+7 \times 8^0+4 \times 8^{-1}+5 \times 8^{-2}=(671.578125)_{10}$$

$$(F81.A)_{16}=15 \times 16^2+8 \times 16^1+1 \times 16^0+10 \times 16^{-1}=(3969.625)_{10}$$

6. 十进制转换成二进制

整数部分与小数部分转换方法不同, 必须分别进行转换, 然后再将两部分转换结果合并成完整的目标数制形式。

整数部分采用基数连除法, 先得到的余数为低位, 后得到的余数为高位。

小数部分采用基数连乘法, 先得到的整数为高位, 后得到的整数为低位。

如, 将十进制数 27.375 转换为二进制数, 可按以下方式来做:

整数部分: $(27)_{10}=(11011)_2$

2 27	余 数
2 13	1 $\uparrow$ b_0 低位
2 6	1 b_1
2 3	0 b_2
2 1	1 b_3
0	1 b_4 高位

小数部分: $(0.375)_{10}=(0.011)_2$

0.375 × 2 = 0.750	整数 0 $b_{-1}=0$ 高位
0.75 × 2 = 1.5	1 $b_{-2}=1$
0.5 × 2 = 1.0	1 $b_{-3}=1$ 低位

同理, 可采用同样的方法将十进制数转换成八进制、十六进制数, 但由于八进制和十六进制的基数较大, 做乘除法不是很方便, 因此需要将十进制转换成八进制、十六进制数时, 也可以将其先转成二进制, 然后再将二进制转成八进制、十六进制数。

7. 二进制数与八进制、十六进制数的转换

(1) 二进制数转换成八进制数

八进制数的基数 8 (2 的 3 次方), 故每位八进制数用三位二进制数构成。因此, 二进制数转换为八进制数的方法是: 整数部分从低位开始, 每三位二进制数为一组, 最后不足三位的, 则在高位加 0 补足三位为止; 小数点后的二进制数则从高位开始, 每三位

二进制数为一组，最后不足三位的，则在低位加 0 补足三位，然后用对应的八进制数来代替，再按顺序排列写出对应的八进制数。

如：将二进制数 $(10011.011101)_2$ 转换成八进制数

二进制数	<u>010</u>	<u>011</u>	<u>011</u>	<u>101</u>
	↓	↓	↓	↓
八进制数	2	3	3	5

因此， $(10011.011101)_2 = (23.35)_8$

(2) 八进制数转换成二进制数

将每位八进制数用三位二进制数来代替，再按原来的顺序排列起来，便得到了相应的二进制数。

如：将八进制数 $(345.726)_8$ 转换为二进制数

八进制数	<u>3</u>	<u>4</u>	<u>5</u>	<u>7</u>	<u>2</u>	<u>6</u>
	↓	↓	↓	↓	↓	↓
二进制数	011	100	101	111	010	110

因此， $(345.726)_8 = (11100101.11101011)_2$

(3) 二进制数转换成十六进制数

与上述相似，由于十六进制数的基数 16 (2 的 4 次方)，故每位十六进制数用四位二进制数构成。同样采用分组对应转换法，将二进制数转换成十六进制数，所不同的是此时每四位为一组，不足四位同样用“0”补足。

如：将二进制数 $(10011.011101)_2$ 转换成十六进制数

二进制数	<u>0001</u>	<u>0011</u>	<u>0111</u>	<u>0100</u>
	↓	↓	↓	↓
十六进制数	1	3	7	4

因此， $(10011.011101)_2 = (13.74)_{16}$

(4) 十六进制数转换成二进制数

将每位十六进制数用四位二进制数来代替，再按原来的顺序排列起来，便得到了相应的二进制数。

如：将十六进制数 $(3B.A78)_{16}$ 转换为二进制数

八进制数	<u>3</u>	<u>B</u>	<u>A</u>	<u>7</u>	<u>8</u>
	↓	↓	↓	↓	↓
二进制数	0011	1011	1010	0111	1000

因此, $(3B.A78)_{16} = (111011.101001111)_2$

由以上可见, 各种数制形式之间的转换方法中, 最基本的是十进制与二进制之间的转变, 八进制和十六进制可以借助二进制来实现相应的转换。

二、编码

数字系统只能识别 0 和 1, 怎样才能表示更多的数码、符号、字母呢? 用编码可以解决此问题。用一定位数的二进制数来表示十进制数码、字母、符号等信息称为编码。用以表示十进制数码、字母、符号等信息的一定位数的二进制数称为代码。

二-十进制代码: 用 4 位二进制数来表示十进制数中的 0~9 十个数码, 也称为 8421BCD 码, 简称 BCD 码。

8421BCD 码是一种应用十分广泛的代码。这种代码每位的权值是固定不变的, 为恒权码。它取了自然二进制数的前十种组合表示一位十进制数 0~9, 即 0000~1001, 从高位到低位的权值分别为 8, 4, 2, 1。去掉了自然二进制数的后六种组合 1010~1111, 8421BCD 码每组二进制代码各位加权系数的和便为它所代表的十进制数。如: 0101 按权展开式为: $0 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1 = 5$ 。表 1.3 所示是十进制与 8421BCD 码的对应关系。

如: $(19)_{10} = (00011001)_{8421BCD}$, $(47)_{10} = (01000111)_{8421BCD}$

表 1.3 十进制与 8421BCD 码的对应关系

十进制数	0	1	2	3	4	5	6	7	8	9
8421BCD 码	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

知识链接二 逻辑代数的基础知识

一、逻辑代数基本运算

1. 基本运算

有与 (AND)、或 (OR)、非 (NOT) 三种。为便于理解它们的含义, 先来看一个简单的例子, 如图 1.2 所示。

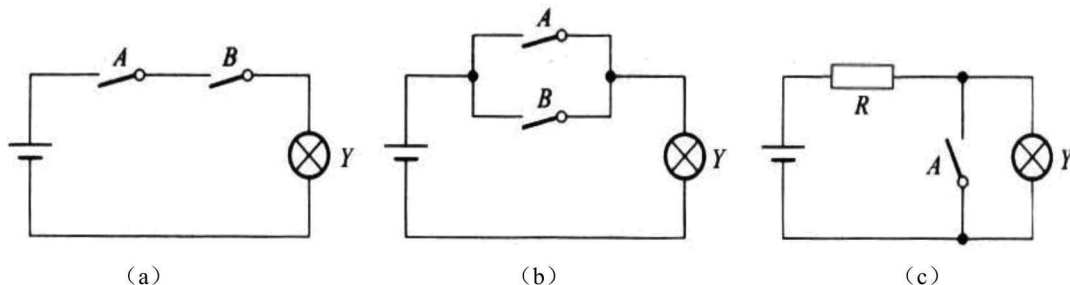


图 1.2 三个指示灯的控制电路

图 1.2 中给出了三个指示灯的控制电路。在图 1.2 (a) 电路中, 只有当两个开关同时闭合时, 指示灯才会亮; 在图 1.2 (b) 电路中, 只要有任何一个开关闭合, 指示灯就亮; 而在图 1.2 (c) 电路中, 开关断开时灯亮, 开关闭合时灯反而不亮。

如果把开关闭合作为条件 (或导致事物结果的原因), 把灯亮作为结果, 那么图 1.2 中的三个电路代表了三种不同的因果关系:

图 1.2 (a) 的例子表明, 只有决定事物结果的全部条件同时具备时, 结果才发生。这种因果关系称为逻辑与, 或称逻辑相乘。

图 1.2 (b) 的例子表明, 在决定事物结果的诸条件中只要有任何一个满足, 结果就会发生。这种因果关系称为逻辑或, 也称逻辑相加。

图 1.2 (c) 的例子表明, 只要条件具备了, 结果便不会发生; 而条件不具备时, 结果一定发生。这种因果关系称为逻辑非, 也称逻辑求反。

若以 A 、 B 表示开关的状态, 并以 “1” 表示开关闭合, 以 “0” 表示开关断开; 以 “ Y ” 表示指示灯的状态。并以 “1” 表示灯亮, 以 “0” 表示不亮, 则可以列出以 0、1 表示的与、或、非逻辑关系的图表。这种图表称为逻辑真值表 (Truth Table), 简称真值表, 如表 1.4~表 1.6 所示。

表 1.4 与逻辑真值表

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

表 1.5 或逻辑真值表

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

表 1.6 非逻辑真值表

A	Y
0	1
1	0

在逻辑代数中, 将与、或、非看作是逻辑变量 A 、 B 间的三种最基本的逻辑运算。并以 “ $\cdot$ ” 表示与运算, 以 “ $+$ ” 表示或运算, 以变量上的 “ $-$ ” 表示非运算。

因此, A 和 B 进行与逻辑运算时可写成

$$Y = A \cdot B \quad (1-1)$$

A 和 B 进行或逻辑运算时可写成

$$Y = A + B \quad (1-2)$$

对 A 进行非逻辑运算时可写成

$$Y = \bar{A} \quad (1-3)$$

同时, 将实现与逻辑运算的单元电路称为与门, 将实现或逻辑运算的单元电路称为或门, 将实现非逻辑运算的单元电路称为非门 (也称为反相器)。

与、或、非逻辑运算还可以用图形符号表示。图 1.3 中给出了被 IEEE（电气与电子工程师协会）和 IEC（国际电工协会）认定的两套与、或、非的图形符号，其中一套是目前在国外教材和 EDA 软件中普遍使用的特定外形符号，如图 1.3（a）所示；另一套是本书中采用矩形轮廓的符号，如图 1.3（b）所示。

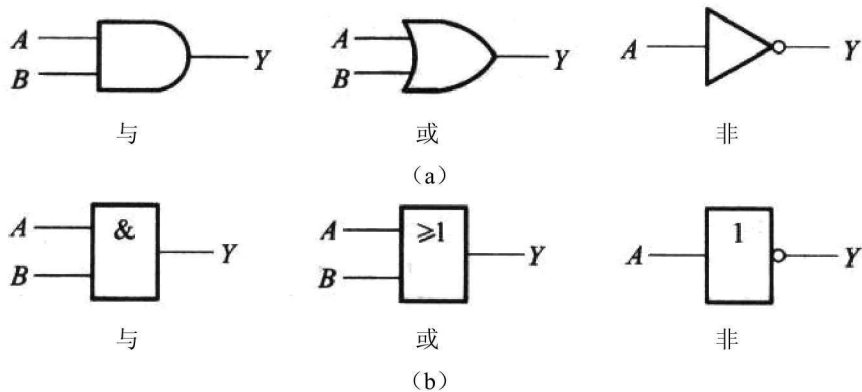


图 1.3 与、或、非的图形符号

2. 异或和同或

实训中所遇到的逻辑关系，称为异或。逻辑表达式 $Y = \overline{A}B + A\overline{B}$ 表示 A 和 B 的异或运算，其真值表如表 1.7 所示，逻辑符号如图 1.4 所示，真值表和实训中的表 1.2 是完全相同的。

表 1.7 异或真值表

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

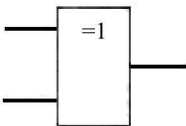


图 1.4 异或逻辑符号

从真值表可以看出，异或运算的含义是：当输入变量相同时，输出为“0”；当输入变量不同时，输出为“1”。 $Y = \overline{A}B + A\overline{B}$ 又可表示为 $Y = A \oplus B$ ，符号“ $\oplus$ ”读做异或。

逻辑表达式 $Y = \overline{\overline{A}B + A\overline{B}}$ 表示 A 和 B 的同或运算。其真值表如表 1.8 所示，逻辑符号如图 1.5 所示。

表 1.8 同或真值表

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

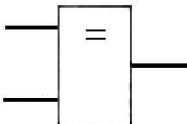


图 1.5 同或逻辑符号

从真值表可以看出，同或运算的含义是：当输入变量相同时，输出为“1”；当输入变量不同时，输出为“0”。 $Y = \overline{AB} + AB$ 又可表示为 $Y = A \odot B$ ，符号“ $\odot$ ”读做同或。

通过真值表可以看出，异或和同或互为非运算。 $A \odot B = \overline{A \oplus B}$

二、逻辑变量与逻辑函数

1849 年英国数学家乔治·布尔（George Boole）首先提出了进行逻辑运算的数学方法——布尔代数。后来，由于布尔代数被广泛应用于解决开关电路和数字逻辑电路的分析与设计中，所以也将布尔代数称为开关代数或逻辑代数。开关电路和数字逻辑电路的分析与设计中，虽然有些逻辑代数的运算公式在形式上和普通代数的运算公式雷同，但是两者所包含的物理意义有本质的不同。逻辑代数中也用字母表示变量。这种变量称为逻辑变量。逻辑运算表示的是逻辑变量以及常量之间逻辑状态的推理运算，而不是数量之间的运算。

当输入逻辑变量的取值确定之后，输出逻辑变量的取值也就被相应地确定了，输出逻辑变量与输入逻辑变量之间存在一定的对应关系，我们将这种对应关系称为逻辑函数。

三、逻辑运算

1. 基本定律

表 1.9 逻辑常量运算公式

与运算	或运算	非运算
$0 \cdot 0 = 0$	$0 + 0 = 0$	
$0 \cdot 1 = 0$	$0 + 1 = 1$	$\overline{1} = 0$
$1 \cdot 0 = 0$	$1 + 0 = 1$	$\overline{0} = 1$
$1 \cdot 1 = 1$	$1 + 1 = 1$	