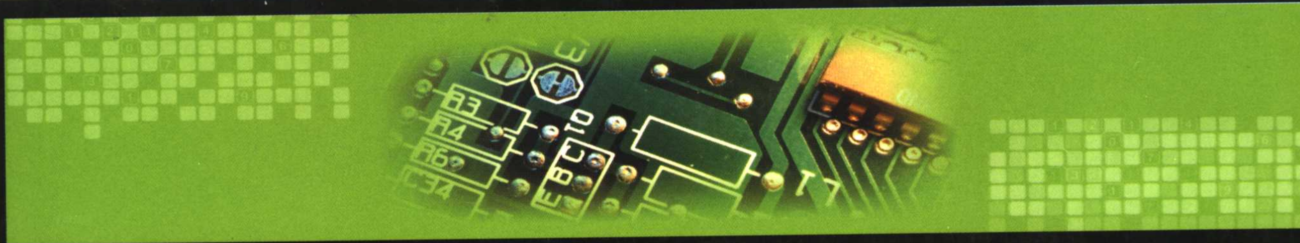
数字电路与逻辑设计实用教程



韩桂英 主编
逄凌滨 于为民 副主编

国防工業出版社
National Defense Industry Press

数字电路与逻辑设计
实用教程

韩桂英 主编
逢凌滨 于为民 副主编

国防工业出版社
·北京·

内 容 简 介

本书包括大学本科教学大纲要求的全部内容和部分扩展内容。要求学生掌握现代数字系统分析与设计的基本原理和基本方法。教学重点是布尔代数基本定律、组合逻辑电路和时序逻辑电路分析与设计。由于可编程器件的使用越来越广泛，其编程技术也是学习的重要内容。

内容包括：数制与编码、逻辑代数、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、脉冲信号的产生与整形、A/D 和 D/A 转换器、可编程逻辑器件和 VHDL。

各章都附有习题。

本书可作为高等学校计算机科学与技术、电子科学与技术、通信工程与电子信息工程、自动化等专业的教材，也可供相关专业科技人员参考。

图书在版编目（CIP）数据

数字电路与逻辑设计实用教程/韩桂英主编. —北京：国防工业出版社，2005.8

ISBN 7-118-04115-7

Ⅰ.数… Ⅱ.韩… Ⅲ.数字电路－逻辑设计－教材 Ⅳ.TN79

中国版本图书馆 CIP 数据核字（2005）第 094619 号

国防工业出版社出版发行

（北京市海淀区紫竹院南路 23 号）

（邮政编码 100044）

国防工业出版社印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 16¼ 373 千字

2005 年 8 月第 1 版 2005 年 8 月北京第 1 次印刷

印数：1—4000 册 定价：26.00 元

国防书店：（010）68428422 发行邮购：（010）68414474

发行传真：（010）68411535 发行业务：（010）68472764

前　言

当今数字技术已成为新技术发展的一个标志，数字技术的普及、微型计算机的迅速发展和应用使数字技术成为计算机应用和自动控制领域中不可缺少的专业基础知识和技术。

为了培养21世纪电气类人才，重基础、宽口径，深化培养复合型、应用型人才的办学理念，结合多年来的教学实践特编写本教材。

在编写过程中习题做到文字叙述通俗易懂、逻辑性强，内容安排由浅入深、循序渐进、理论联系实际，习题与课程内容相配合，做到便于自学。

本书共分10章，重点介绍中、大规模集成电路的应用。

第1章和第2章介绍了数字电路的基础知识，内容包括数字与编码、逻辑代数和逻辑门电路。这些是研究数字逻辑电路的组成和应用所必备的。

第3章是组合逻辑电路。主要介绍了常用中、小规模集成组合电路及其应用。

第4章～第6章为时序逻辑电路。主要介绍了常用中、小规模集成时序逻辑电路及其应用。

第7章为脉冲产生与整形电路。

第8章和第9章为可编程逻辑器件和半导体存储器。

第10章为数模与模数转换器。

全书采用国家标准的图形符号。

本书第1章～第3章由于为民编写，第4章和第5章由李锡祚编写，第6章由韩桂英编写，第7章～第10章由逄凌滨和常治学编写，课后习题由刘忠富、薛原、石立新编写。全书由韩桂英负责统编定稿。

目　录

第1章　逻辑代数基础 …… 1

1.1　概述 …… 1
一、数字信号与数字电路 …… 1
二、数制和代码 …… 2
三、算术运算和逻辑运算 …… 5
1.2　逻辑代数中的三种基本运算 …… 7
一、逻辑变量与逻辑函数 …… 7
二、基本逻辑运算和复合逻辑运算 …… 7
1.3　逻辑代数的基本公式和常用公式 …… 10
一、基本公式 …… 10
二、常用公式 …… 11
1.4　逻辑代数的基本运算规则 …… 12
一、代入规则 …… 12
二、反演规则 …… 12
三、对偶规则 …… 13
1.5　逻辑函数及其表示方法 …… 13
一、逻辑函数 …… 13
二、逻辑函数的表示方法 …… 14
三、逻辑函数的两种标准形式 …… 15
1.6　逻辑函数的公式化简法 …… 18
一、逻辑函数的最简形式 …… 18
二、常用的公式化简方法 …… 19
1.7　逻辑函数的卡诺图化简法 …… 20
一、逻辑函数的卡诺图表示法 …… 20
二、用卡诺图化简逻辑函数 …… 22
1.8　具有无关项的逻辑函数的化简 …… 24
一、逻辑函数式中的无关项 …… 24
二、化简具有约束项的逻辑函数 …… 24
习题 …… 26

第2章　逻辑门电路 …… 29

2.1　分立元件门电路 …… 29

一、半导体二极管、三极管的开关特性 …… 29
二、分立元件门电路 …… 30
2.2　TTL 集成门电路 …… 33
一、概述 …… 33
二、TTL 与非门 …… 34
三、其他逻辑功能的 TTL 门电路 …… 39
2.3　MOS 门电路 …… 44
一、CMOS 反相器 …… 44
二、CMOS 与非门和或非门 …… 46
三、漏极开路的 CMOS 门和 CMOS 三态门 …… 47
四、CMOS 传输门和模拟开关 …… 48
2.4　TTL 与 CMOS 电路的连接 …… 48
一、两类集成门电路相互连接的条件 …… 48
二、接口电路 …… 49
习题 …… 49

第 3 章　组合逻辑电路 …… 56

3.1　组合逻辑电路的分析和设计 …… 56
一、组合电路逻辑功能的特点 …… 56
二、组合逻辑电路的分析 …… 56
三、组合逻辑电路的设计 …… 57
3.2　常用组合电路 …… 58
一、编码器和译码器 …… 58
二、数据选择器 …… 69
三、数值比较器 …… 72
四、算术运算电路 …… 74
3.3　组合逻辑电路中的竞争和冒险 …… 80
一、竞争与冒险 …… 80
二、冒险现象的判别 …… 82
三、消除冒险现象的方法 …… 83
习题 …… 83

第 4 章　触发器 …… 87

4.1　基本 RS 触发器 …… 87
一、电路结构与工作原理 …… 87
二、逻辑功能描述 …… 88
三、动作特点 …… 90
四、应用举例 …… 90
4.2　时钟触发器 …… 91

一、电平触发的时钟触发器 …… 91
二、主从触发的时钟触发器 …… 93
三、边沿触发的时钟触发器 …… 98
4.3　各种逻辑功能触发器描述及相互转换 …… 101
一、各种逻辑功能触发器描述 …… 101
二、各种触发器之间的转换 …… 102
习题 …… 106

第5章　时序逻辑电路 …… 112

5.1　时序逻辑电路综述 …… 112
一、时序逻辑电路的基本概念 …… 112
二、时序逻辑电路的描述 …… 113
5.2　时序逻辑电路的分析 …… 114
5.3　时序逻辑电路设计 …… 117
习题 …… 123

第6章　常用时序逻辑功能器件及其应用 …… 126

6.1　计数器 …… 126
一、计数器的分类 …… 126
二、二进制计数器 …… 127
三、十进制计数器 …… 131
四、集成计数器 …… 132
6.2　寄存器 …… 136
一、锁存器 …… 137
二、数码寄存器 …… 138
三、移位寄存器 …… 138
6.3　顺序脉冲发生器 …… 143
6.4　MSI 时序电路的设计 …… 144
一、任意进制计数器 …… 144
二、其他时序电路设计 …… 150
习题 …… 153

第7章　脉冲信号的产生与整形 …… 157

7.1　555 定时器及应用 …… 157
一、555 定时器的电路结构与工作原理 …… 157
二、用 555 定时器组成单稳态触发器电路 …… 159
三、用 555 定时器组成多谐振荡器 …… 161
四、用 555 定时器组成的施密特触发器 …… 162
7.2　单稳态触发器 …… 165

一、由门电路组成的微分型单稳态触发器 …… 165
二、集成单稳态触发器 …… 167
7.3　多谐振荡器 …… 171
一、门电路组成的多谐振荡器 …… 171
二、石英晶体多谐振荡器 …… 174
7.4　施密特触发器 …… 175
一、门电路组成的施密特触发器 …… 175
二、集成的施密特触发器 …… 177
习题 …… 178

第8章　可编程逻辑器件 …… 182

8.1　概述 …… 182
8.2　可编程阵列逻辑 …… 183
8.3　通用阵列逻辑 …… 185
一、GAL的电路结构 …… 186
二、输出逻辑宏单元 …… 187
8.4　现场可编程门阵列 …… 190
一、FPGA的基本结构 …… 190
二、FPGA的IOB和CLB …… 192
三、FPGA的可编程连线资源 …… 193
四、编程数据的装载 …… 195
8.5　在系统可编程逻辑器件 …… 197
一、低密度ISP—PLD …… 198
二、高密度ISP—PLD …… 199
8.6　PLD的编程 …… 203
习题 …… 204

第9章　半导体存储器 …… 207

9.1　只读存储器 …… 207
一、掩模只读存储器 …… 207
二、可编程只读存储器 …… 209
三、可擦除的可编程只读存储器 …… 210
四、ROM的应用举例 …… 215
9.2　随机存储器 …… 215
一、静态RAM(SRAM)存储单元 …… 215
二、SRAM的结构和工作原理 …… 216
9.3　存储器容量的扩展 …… 218
一、字长(位数)的扩展 …… 218
二、字数的扩展 …… 218

习题 …… 219

第 10 章　数模与模数转换器 …… 222

10.1　D/A 转换器 …… 222

一、权电阻网络 D/A 转换器 …… 222

二、倒 T 型电阻网络 D/A 转换器 …… 224

三、权电流型 D/A 转换器 …… 226

10.2　D/A 转换器的转换精度和转换速度 …… 228

一、D/A 转换器的转换精度 …… 228

二、D/A 转换器的转换速度 …… 231

10.3　集成 D/A 转换器应用举例 …… 232

一、D/A 转换器的输出方式 …… 232

二、D/A 转换器的应用举例 …… 234

10.4　A/D 转换器 …… 236

一、A/D 转换器的基本工作原理 …… 236

二、取样－保持电路 …… 238

三、直接 A/D 转换器 …… 239

四、逐次逼近型 A/D 转换器 …… 241

五、间接 A/D 转换器 …… 243

六、A/D 转换器的转换精度与转换速度 …… 247

习题 …… 248

参考文献 …… 252

第 1 章　逻辑代数基础

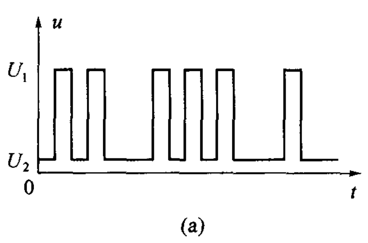
在电子技术领域，为便于存储、分析和传输，往往将模拟信号进行编码，转换为数字信号。本章介绍数的进制、代码和逻辑代数的基本知识。首先简要讲述数字电路中常见的数制和代码，然后介绍分析数字电路的基本数学工具——逻辑代数。在讲解逻辑代数基本概念、基本公式、常用公式和重要定理的基础上，着重阐述逻辑函数的表示方法，以及应用公式和卡诺图化简逻辑函数的方法。

1.1　概　述

一、数字信号与数字电路

1. 数字信号与模拟信号

自然界的各种物理量，按其变化规律可以分为两大类。一种变化在时间和数量上是不连续的，如信号灯闪亮的次数、电子表的秒指示等，它们的变化发生在离散的瞬间，其数值的大小都是某个最小数量单位的整数倍。这一类的物理量叫做数字量，表示数字量的信号叫做数字信号，如图 1.1.1(a)所示。工作在数字信号下的电路叫做数字电路。另一类物理量在时间、数量上是连续的，例如从热电偶得到的电压信号、自然界温度的变化等，这一类的物理量叫做模拟量，表示模拟量的信号叫做模拟信号，如图 1.1.1(b)所示。工作在模拟信号下的电路叫做模拟电路。



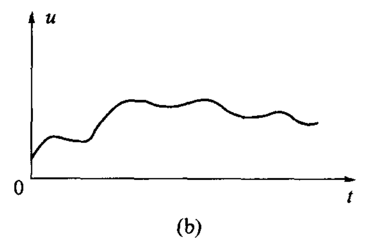


图 1.1.1　数字信号和模拟信号

(a) 数字信号；(b) 模拟信号。

2. 数字电路的特点

数字电路的工作信号一般都是数字信号。在电路中，它往往表现为突变的电压或电流，并且只有两个可能的状态。所以，数字电路中的二极管和三极管多数工作在开关状态。利用导通和截止两种不同的工作状态，代表不同的数字信息，完成信号的传递和处理任务。

在数字电路中，重点研究的问题是输入信号和输出信号之间的逻辑关系。为了分析

这些逻辑关系，需要使用一套新的数学工具，即逻辑代数。表示电路功能的方法也往往是用真值表、逻辑函数式、卡诺图、特性方程以及状态转换图等。

二、数制和代码

1. 数的几种常用进制

1）十进制

十进制是我们所熟悉的计数体制，它用 0 ~ 9 十个数字符号，按照一定的规律排列起来，表示数值的大小。例如：

$$123.45 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2}$$

从这个 5 位十进制数，不难发现十进制数的特点：它的基数是 **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。任意一个十进制数 D 均可展开为

$$D = \sum k_i \times 10^i \tag{1.1.1}$$

式中 k_i 是第 i 位的系数，它可以是 0 ~ 9 这十个数码中的任何一个。若整数部分的位数是 n，小数部分的位数是 m，则 i 包含从 $n-1$ 到 0 的所有正整数和从 -1 到 $-m$ 的所有负整数。

若以 N 取代式(1.1.1)中的 10，即可得到任意进制（N 进制）数展开式的普遍形式

$$D = \sum k_i N^i \tag{1.1.2}$$

式中 i 的取值与式(1.1.1)的规定相同，N 称为计数的基数，k_i 为第 i 位的系数，N^i 称为第 i 位的权。

2）二进制

目前在数字电路中应用最广的是二进制。在二进制数中，每一位仅有 0 和 1 两个可能的数字符号，所以计数的基数为 **2**。低位和相邻高位间的进位关系是“逢二进一”，故称为二进制。

根据式(1.1.2)，任何一个二进制数均可展开为

$$D = \sum k_i 2^i \tag{1.1.3}$$

并计算出它所表示的十进制数的大小。例如：

$$(101.11)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = (5.75)_{10}$$

上式中分别使用下脚注的 2 和 10 表示括号里的数是二进制和十进制数。有时也用 B (Binary) 和 D(Decimal) 代替 2 和 10 这两个脚注。

3）十六进制

十六进制数用 0 ~ 9、A、B、C、D、E、F 等 16 个符号表示。任意一个十六进制数均可表示为

$$D = \sum k_i 16^i \tag{1.1.4}$$

例如：

$$(2B.6F)_{16} = 2 \times 16^1 + 11 \times 16^0 + 6 \times 16^{-1} + 15 \times 16^{-2} = (43.43359)_{10}$$

式中的下脚注 16 表示括号里的数是十六进制，有时也用 H(Hexadecimal) 标注。

由于目前在微型计算机中普遍采用 8 位、16 位和 32 位二进制并行运算，而 8 位、16

位和32位的二进制数可以用2位、4位和8位的十六进制数表示，因而用十六进制符号书写程序十分简便。

2. 数制转换

1）二—十转换

把二进制数转换为等值的十进制数，称为二—十转换。转换时只要将二进制数按式(1.1.3)展开，然后把所有各项的数值按十进制数相加，就可以得到等值的十进制数了。例如：

$$(1101.01)_2 = 1\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 + 0\times2^{-1} + 1\times2^{-2} = (13.75)_{10}$$

2）十—二转换

把十进制数转换为二进制数，整数部分用“除二取余法”，小数部分用“乘2取整法”，具体操作举例如下。

例 1.1.1　将$(25)_{10}$化为二进制数

解：

$$\begin{array}{r|l l l}
2 & 25 & \cdots\cdots 余 1 & \cdots\cdots K_0 \\ \hline
2 & 12 & \cdots\cdots 余 0 & \cdots\cdots K_1 \\ \hline
2 & 6 & \cdots\cdots 余 0 & \cdots\cdots K_2 \\ \hline
2 & 3 & \cdots\cdots 余 1 & \cdots\cdots K_3 \\ \hline
2 & 1 & \cdots\cdots 余 1 & \cdots\cdots K_4 \\ \hline
 & 0 & &
\end{array}$$

故$(25)_{10} = (11001)_2$

例 1.1.2　把$(0.8125)_{10}$转换成二进制数

解：

$$\begin{array}{r}
0.8125 \\
\times\quad\quad 2 \\ \hline
1.6250
\end{array}\cdots\cdots\cdots\cdots 整数部分 = 1 = k_{-1}$$

$$\begin{array}{r}
0.6250 \\
\times\quad\quad 2 \\ \hline
1.2500
\end{array}\cdots\cdots\cdots\cdots 整数部分 = 1 = k_{-2}$$

$$\begin{array}{r}
0.2500 \\
\times\quad\quad 2 \\ \hline
0.5000
\end{array}\cdots\cdots\cdots\cdots 整数部分 = 0 = k_{-3}$$

$$\begin{array}{r}
0.5000 \\
\times\quad\quad 2 \\ \hline
1.0000
\end{array}\cdots\cdots\cdots\cdots 整数部分 = 1 = k_{-4}$$

故$(0.8125)_{10} = (0.1101)_2$

3）二—十六转换

把二进制数转换成等值的十六进制数，称为二—十六转换。

由于4位二进制数恰好有16个状态，而把这4位二进制数看作一个整体时，它的进位输出又正好是逢十六进一，所以只要从低位到高位将每4位二进制数分为一组，并代之以等值的十六进制数，即可得到对应的十六进制数。

例如，将$(01101010.11010010)_2$化为十六进制数时可得

$$
\begin{array}{cccccc}
(0110, & 1010. & 1101, & 0010)_2 \\
\downarrow & \downarrow & \downarrow & \downarrow \\
=(6 & A\ . & D & 2\ )_{16}
\end{array}
$$

4）十六—二转换

十六—二转换是指把十六进制数转换成等值的二进制数。转换时只需将十六进制数的每一位用等值的4位二进制数代替就行了。

例如，将$(8FB.C5)_{16}$化为二进制数时得到

$$
\begin{array}{ccccc}
(8 & F & B. & C & 5)_{16} \\
\downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
=(1000 & 1111 & 1011. & 1100 & 0101)_2
\end{array}
$$

5）十六进制数与十进制数的转换

在将十六进制数转换为十进制数时，可根据式(1.1.4)将各位数按权展开后相加求得。在将十进制数转换为十六进制数时，可以先转换成二进制数，然后再将得到的二进制数转换为等值的十六进制数。

3. 代码

不同的数码不仅可以表示不同的数量大小，而且还能用来表示不同的事物。在后一种情况下，这些数码已没有表示数量大小的含义，只是表示不同事物的代号而已。这些数码称为代码。

为了便于记忆和处理，在编制代码时总要遵循一定的规则，这些规则就叫做码制。

例如在用4位二进制数码表示1位十进制数的0～9这十个状态时，就有多种不同的码制。通常将这些代码称为二—十进制代码，简称**BCD**(Binary Coded Decimal)代码。

表1.1.1列出了几种常见的BCD代码，它们的码制规则各不相同。

表1.1.1　几种常见的BCD代码

编码种类 / 十进制数	8421码	余3码	2421码	5211码	余3循环码
0	**0000**	**0011**	**0000**	**0000**	**0010**
1	**0001**	**0100**	**0001**	**0001**	**0110**
2	**0010**	**0101**	**0010**	**0100**	**0111**
3	**0011**	**0110**	**0011**	**0101**	**0101**
4	**0100**	**0111**	**0100**	**0111**	**0100**
5	**0101**	**1000**	**1011**	**1000**	**1100**

（续）

编码种类 十进制数	8421 码	余 3 码	2421 码	5211 码	余 3 循环码
6	0110	1001	1100	1001	1101
7	0111	1010	1101	1100	1111
8	1000	1011	1110	1101	1110
9	1001	1100	1111	1111	1010
权	8421		2421	5211	

8421 码是 BCD 代码中最常用的一种。在这种编码方式中每一位二值代码的 1 都代表一个固定的数值，把每一位的 1 代表的十进制数加起来，得到的结果就是它所代表的十进制数码。由于代码中从左到右每一位的 1 分别表示 8、4、2、1，所以把这种代码叫做 8421 码。每一位的 1 代表的十进制数称为这一位的权。8421 码中每一位的权是固定不变的，它属于恒权代码。

余 3 码的编码规则与 8421 码不同，如果把每一个余 3 码看作 4 位二进制数，则它的数值要比它所表示的十进制数码多 3，故而将这种代码叫做余 3 码。

如果将两个余 3 码相加，所得的和将比十进制数和所对应的二进制数多 6。因此，在余 3 码作十进制加法运算时，若两数之和为 10，正好等于二进制数的 16，于是便从高位自动产生进位信号。

此外，从表 1.1.1 中还可以看出，0 和 9、1 和 8、2 和 7、3 和 6、4 和 5 的余 3 码互为反码，这对于求取对 10 的补码是很方便的。

余 3 码不是恒权代码。如果试图把每个代码视为二进制数，并使它所等效的十进制数与所表示的代码相等，那么代码中每一位的 1 所代表的十进制数在各个代码中不是固定的。

2421 码是一种恒权代码，它的 0 和 9、1 和 8、2 和 7、3 和 6、4 和 5 也互为反码，这个特点和余 3 码相仿。

5211 码是另一种恒权代码。学了计数器的分频作用后可以发现，如果按 8421 码接成十进制计数器，则连续输入计数脉冲的 4 个触发器输出脉冲对于计数脉冲的分频比从低位到高位依次为 5∶2∶1∶1。可见，5211 码每一位的权正好与 8421 码十进制计数器 4 个触发器输出脉冲的分频比相对应。这种对应关系在构成某些数字系统时很有用。

余 3 循环码是一种变权码，每一位的 1 在不同代码中并不代表固定的数值。它的主要特点是相邻的两个代码之间仅有一位的状态不同。因此，按余 3 循环码接成计数器时，每次状态转换过程中只有一个触发器翻转，译码时不会发生竞争冒险现象（详见第 3 章）。

三、算术运算和逻辑运算

在数字电路中，1 位二进制数码的 0 和 1 不仅可以表示数量的大小，而且可以表示两种不同的逻辑状态。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、电灯的亮和暗等等。这种只有两种对立逻辑状态的逻辑关

系称为二值逻辑。

当两个二进制数码表示两个数量大小时，它们之间可以进行数值运算，这种运算称为算术运算。二进制算术运算和十进制算术运算的规则基本相同，惟一区别在于二进制数是逢二进一，而不是十进制数的逢十进一。

例如，两个二进制数 1001 和 0101 的算术运算有

加法运算

$$\begin{array}{r} 1001 \\ +\ 0101 \\ \hline 1110 \end{array}$$

减法运算

$$\begin{array}{r} 1001 \\ -\ 0101 \\ \hline 0100 \end{array}$$

乘法运算

$$\begin{array}{r} 1001 \\ \times\ 0101 \\ \hline 1001 \\ 0000 \\ 1001 \\ 0000 \\ \hline 0101101 \end{array}$$

除法运算

$$\begin{array}{r} 1.11\cdots \\ 0101\overline{)1001} \\ 0101 \\ \hline 1000 \\ 0101 \\ \hline 0110 \\ 0101 \\ \hline 0010 \end{array}$$

在数字电路和数字电子计算机中，二进制数的正、负号也是用 0 和 1 表示。在定点运算的情况下，以最高位作为符号位，正数为 0，负数为 1，以下各位用 0 和 1 表示数值。用这种方式表示的数码称为原码。例如：

为了简化运算电路，在数字电路中两数相减的运算是用它们的补码相加来完成的。二进制数的补码是这样定义的：

最高位为符号位，正数为 0，负数为 1；

正数的补码和它的原码相同；

负数的补码可以通过将原码的数值位逐位求反，然后在最低位上加 1 得到。

例 1.1.3 计算 $(+1001)_2-(0101)_2$

解：根据二进制数的运算规则可知

$$\begin{array}{r} \mathbf{1001} \\ -\ \mathbf{0101} \\ \hline \mathbf{0100} \end{array}$$

在采用补码运算时，首先求出 $(+1001)_2$ 和 $(-0101)_2$ 的补码，它们是

$$[+\mathbf{1001}]_{补}=\underbrace{\boxed{\mathbf{0}}}_{符号位}\ \mathbf{1001}$$

$$[-\mathbf{0101}]_{补}=\underbrace{\boxed{\mathbf{1}}}_{符号位}\ \mathbf{1011}$$

然后将两个补码相加并舍去进位

$$
\begin{array}{r}
01001 \\
+\ 11011 \\
\hline
\text{舍去} \leftarrow 1\ \boxed{0}0100
\end{array}
$$

则得到与前面一样的结果。这样就把减法运算转化成了加法运算。

此外,也不难发现,乘法运算可以用加法和移位两种操作实现,而除法运算可以用减法和移位操作实现。因此,二进制数的加、减、乘、除运算都可以用加法运算电路完成,这就大大简化了运算电路的结构。

当两个二进制数码表示不同的逻辑状态时,它们之间可以按照指定的某种因果关系进行所谓逻辑运算。这种逻辑运算和算术运算有着本质的不同。下面将重点介绍逻辑运算的各种规律。

1.2　逻辑代数中的三种基本运算

逻辑代数是一种描述客观事物逻辑关系的数学方法。它是英国数学家乔治·布尔(Georeg Bool)在 1847 年首先提出来的,所以又称布尔代数。

一、逻辑变量与逻辑函数

事物的发展和变化通常是按照一定的因果关系进行的。例如,照明电路中电灯是否能亮取决于电源是否接通和灯泡的好坏。后两者是因,前者是果。这种因果关系一般称为逻辑关系。逻辑代数正是反映这种逻辑关系的数学工具。

由于事物包含相互对立而又相互联系的两个方面,如上例所说亮与暗、通与断、好与坏等,所以在逻辑代数中,为了描述事物两种对立的逻辑状态,采用的是仅有两个取值的变量。这种变量称为逻辑变量。

逻辑变量和普通代数变量一样,都是用字母表示。但是,它又和普通代数变量有着本质区别:所研究的逻辑变量的取值只有 0 和 1 两种可能,而且这里的 0 和 1 不是表示数值大小,而是代表逻辑变量的两种对立状态。

在数字电路中的二进制数码,有时可做二进制数表示数值的大小,此时它们之间可以进行算术运算;有时还可以作为逻辑变量表示不同的逻辑状态,此时它们之间只能按照某种逻辑关系进行逻辑运算。

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,输出与输入之间乃是一种函数关系,这种函数关系称为逻辑函数,写作

$$Y = F(A,B,C,\cdots)$$

由于变量和输出(函数)的取值只有 0 和 1 两种状态,所以讨论的都是二值逻辑函数。

二、基本逻辑运算和复合逻辑运算

逻辑代数的基本运算有与、或、非 3 种。下面结合指示灯控制电路的实例分别讨论。

1. 基本逻辑运算

1）与运算（逻辑与）

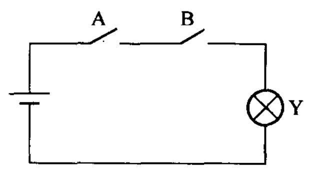


图 1.2.1　串联开关电路

图 1.2.1 给出了指示灯的两开关串联控制电路。由图可知，只有 A 和 B 两个开关全部闭合时，指示灯 Y 才会亮；如果有一个开关不闭合，或两个开关均不闭合，则指示灯不亮。

由此得到这样的逻辑关系：只有决定事物结果（灯亮）的若干条件（开关 A 和 B 闭合）全部满足时，结果才会发生。这种条件和结果的关系称为逻辑与，也叫与逻辑关系。

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 表示灯的状态，灯亮用 1 表示，灯不亮用 0 表示，则可得到表 1.2.1。这种用逻辑变量的真正取值反映逻辑关系的表格称为逻辑真值表，简称真值表。

在逻辑代数中，把逻辑变量之间逻辑与关系称作与运算，也叫逻辑乘，并用符号“·”表示与。因此，A、B 和 Y 的与逻辑关系可写成

$$Y = A \cdot B \tag{1.2.1}$$

此式称为与逻辑表达式。

与逻辑关系还可以用逻辑符号表示，如图 1.2.2 所示。

2）或运算（逻辑或）

图 1.2.3 给出了指示灯的两个开关并联控制电路。显而易见，只要任何一个开关（A 或 B）闭合或两个都闭合，指示灯 Y 都会亮；如果两个开关均不闭合，则灯不亮。由此可得到另一种逻辑关系：在决定事物结果的若干个条件中，只要满足一个或一个以上条件时，结果就会发生；否则，结果就不会发生。这种因果关系称为逻辑或，也叫或逻辑关系。

表 1.2.1　与逻辑真值表

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

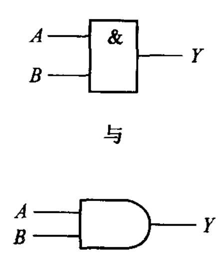


图 1.2.2　与逻辑符号

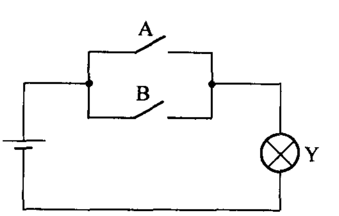


图 1.2.3　并联开关

按照前述假设，用二值逻辑变量可以列出或逻辑关系的真值表，如表 1.2.2 所示。

逻辑变量之间逻辑或关系，也称为或运算，也叫逻辑加，并用符号“+”表示。因此，A、B 和 Y 的逻辑关系表达式为

$$Y = A + B \tag{1.2.2}$$

或逻辑关系也可以用逻辑符号表示，图 1.2.4 为或逻辑符号。