



教育部职业教育与成人教育司推荐教材
职业教育电力技术类专业教学用书

数字电子 技术基础

张 钢 方舒燕 主编



中国电力出版社

<http://jc.cepp.com.cn>



教育部职业教育与成人教育司推荐教材
职业教育电力技术类专业教学用书

数字电子 技术基础

主 编 张 钢 方舒燕
编 写 赵 峥
主 审 郝 波 尹常永



中国电力出版社

<http://jc.cepp.com.cn>

内 容 提 要

本教材充分考虑了职业技术教育《数字电子技术》课程教学的要求,在内容安排上既涵盖了课程必需的基本内容,又增加了许多能反映电子技术最新成就的相关内容。为此,对课程的内容进行了适当的调整,精简压缩了分立元件,优化了小规模集成电路与脉冲数字电路的内容,突出讲述了中规模集成电路及其应用实例,引入可编程逻辑器件与电子电路仿真软件。

本书遵循教学规律,力求由浅入深,由易到难,由简到繁,循序渐进,便于自学。突出数学电路学习的特点和重点。

本教材适用于各类职业院校多种专业《数字电子技术》课程的教学。如电力类、电气类、机电类、计算机类和电子类专业等。各专业可根据需要及教学时数情况对内容酌情取舍。

图书在版编目(CIP)数据

数字电子技术基础/张钢,方舒燕主编. —北京:中国电力出版社,2005.8

教育部职业教育与成人教育司推荐教材

ISBN 7-5083-3459-0

I. 数… II. ①张…②方… III. 数字电路—电子技术—高等学校:技术学校—教材 IV. TN79

中国版本图书馆 CIP 数据核字(2005)第 076725 号

中国电力出版社出版、发行

(北京三里河路 6 号 100044 <http://jc.cepp.com.cn>)

北京丰源印刷厂印刷

各地新华书店经售

*

2005 年 8 月第一版 2006 年 8 月北京第二次印刷

787 毫米×1092 毫米 16 开本 17.5 印张 376 千字

印数 4001—8000 册 定价 22.50 元

版 权 专 有 翻 印 必 究

(本书如有印装质量问题,我社发行部负责退换)

前言

本书为教育部职业教育与成人教育司推荐教材,是根据教育部审定的电力技术类专业主干课程的教学大纲编写而成的,并列入教育部《2004~2007年职业教育教材开发编写计划》。本书经中国电力教育协会和中国电力出版社组织专家评审,又列为全国电力职业教育规划教材,作为职业教育电力技术类专业教学用书。

本书体现了职业教育的性质、任务和培养目标;符合职业教育的课程教学基本要求和有关岗位资格和技术等级要求;具有思想性、科学性、适合国情的先进性和教学适应性;符合职业教育的特点和规律,具有明显的职业教育特色;符合国家有关部门颁发的技术质量标准。本书既可以作为学历教育教学用书,也可作为职业资格和岗位技能培训教材。

电子技术的发展日新月异,而数字电子技术的知识更新速度更是高于整个科技领域发展的平均速度。通信、计算机、广播、电视、数码产品直到家用电器,几乎各个领域都在向数字化发展,这就对以数字系统应用为主要教学内容的《数字电子技术》课程提出了更高的要求。

根据职业技术教育以培养工程应用型技术人才为主的培养目标,本书力求在保证必要的基本理论、基本知识、基本分析方法和技能的基础上,贯彻理论与工程实践相结合,以应用为目的,以必需、够用为度和掌握概念、强化应用等原则,力求突出职业教学的特色。

本教材充分考虑了职业技术教育《数字电子技术》课程教学的要求,在内容安排上既涵盖了课程必需的基本内容,又增加了许多能反映电子技术最新成就的相关内容。为此,对课程的内容进行了适当的调整,精简压缩了分立元件、优化了小规模集成电路与脉冲数字电路的内容,突出讲述了中规模集成电路及其应用实例,引入可编程逻辑器件与电子电路仿真软件。

本书遵循教学规律,力求由浅入深,由易到难,由简到繁,循序渐进,便于自学。突出数学电路学习的特点和重点。

在内容上,掌握数制、码制、逻辑代数、卡诺图、真值表、逻辑符号和逻辑函数式表示方法和分析方法的基础上,以集成电路为主线,介绍各种常用的组合和时序逻辑电路的逻辑功能、使用方法,并适当介绍一些常用 DAC、ADC 和可编程逻辑器件的基本知识。在介绍集成数字电路的各章中,列举了各种典型的应用电路,使理论与实践应用相结合,拓宽读者的应用思路,了解应用规律和方法。力图做到取材精练,组织合理,深入浅出,富有新意。考虑到计算机的日益普及,本书最后一章介绍了电子电路仿真软件,以期使读者了解数字电路的仿真方法,有条件的读者可在计算机上完成数字电路的虚拟实验,相信这些会十分有助于理论课的深化学习。

本教材适用于各类职业院校多种专业《数字电子技术》课程的教学。如电力类、电气类、机电类、计算机类和电子类专业等。各专业可根据需要及教学时数情况对内容酌情取舍。

目前国内教材上电子元器件的图形符号和字母代号难以统一。考虑到读者使用方便，本书采用了国标图形符号和字母代号。

全书共分九章。张钢编写了第五、六、七、八章，第一章至第四章由方舒燕编写，赵峥编写第九章及附录。全书由张钢、方舒燕任主编，负责全书的组织、修改和定稿工作。

本教材由郝波副教授和尹常永副教授主审，他们对书稿进行了认真详尽的审阅，提出了许多很有价值的宝贵意见，在此表示深切的谢意。

本教材的编写过程中得到了各级领导和各方面的大力支持，在此向他们表示衷心的感谢。另外，教材的编写参考了一些相关著作与资料。谨向书籍和文章的作者表示衷心的感谢。

由于编者水平有限，时间紧促，书中难免存在错漏和不妥之处。敬请广大读者提出批评和指正。

编 者

2005年5月

目 录

前言

第一章 数字逻辑基础	1
第一节 数制和代码	2
第二节 逻辑代数基础	7
第三节 逻辑函数的化简	13
习题	19
第二章 集成逻辑门电路	23
第一节 晶体二极管和三极管的开关特性	23
第二节 基本逻辑门电路	25
第三节 TTL集成逻辑门电路	27
第四节 CMOS集成逻辑门电路	34
第五节 各种类型门电路接口技术	38
习题	42
第三章 组合逻辑电路	45
第一节 组合逻辑电路的分析	45
第二节 组合逻辑电路的设计	48
第三节 编码器	50
第四节 译码器	53
第五节 加法器	61
第六节 数值比较器和数据选择器	63
第七节 中规模组合集成电路综合应用	70
第八节 组合逻辑电路中的竞争冒险	71
习题	74
第四章 触发器	78
第一节 基本 RS 触发器	78
第二节 同步触发器	82
第三节 常用钟控触发器	86
习题	95
第五章 时序逻辑电路	98
第一节 概述	98
第二节 时序逻辑电路的分析方法	100
第三节 异步计数器	103
第四节 同步计数器	110
第五节 任意进制计数器	116
第六节 寄存器	121
第七节 时序逻辑电路综合应用	128

习题	132
第六章 脉冲波形的产生与变换	136
第一节 集成 555 定时器	136
第二节 单稳态触发器	138
第三节 施密特触发器	145
第四节 多谐振荡器	149
习题	155
第七章 数模与模数转换器	158
第一节 概述	158
第二节 D/A 转换器	159
第三节 A/D 转换器	166
习题	177
第八章 半导体存储器与可编程逻辑器件	179
第一节 概述	179
第二节 只读存储器 ROM	180
第三节 随机存储器 RAM	188
第四节 可编程逻辑器件概述	192
第五节 低密度可编程逻辑器件	195
第六节 高密度可编程逻辑器件	202
习题	206
第九章 数字电路的计算机仿真	208
第一节 电子工作室简介	208
第二节 菜单栏与元器件库	209
第三节 EWB 的基本操作	214
第四节 电路分析方法	222
第五节 数字电路的 EWB 仿真举例	223
习题	227
附录一 国产半导体集成电路型号命名	228
附录二 常用逻辑符号对照表	230
附录三 54/74 常用数字集成电路一览表	232
附录四 数的原码、反码和补码	236
主要参考文献	239

数 字 逻 辑 基 础

引言：数字电路是一门研究数字信号的编码、运算、记忆、计数、存储、分配、测量与传输的科学技术。简单地说是用数字信号去实现运算、控制、测量的一门学科。数字化已成为当今现代电子技术的发展潮流。本章是基础知识，介绍的主要内容有：常用的数制和码制；与、或、非三种基本逻辑关系及与非、或非、与或非、异或和同或几种复合逻辑关系以及实现这些逻辑关系的对应的门电路；逻辑函数的几种表示方法——真值表、逻辑表达式、逻辑电路图、卡诺图，以及它们的特点和相互转换的方法；逻辑函数的化简方法——公式法和卡诺图法。

电子电路所处理的电信号可以分为**模拟信号**和**数字信号**两大类。**模拟信号**是指在时间和数值上都连续变化的信号，如温度、压力、正弦波电压和电流以及广播电视系统中传送的各种语言信号和图像信号等，模拟信号可以用计量仪器测量出某个时刻模拟量的瞬时值，或某一段时间之内的平均值，或有效值。传送和处理模拟信号的电路称为**模拟电路**（Analog Circuit）；**数字信号**是在时间和数值上都是断续变化的离散信号，数字信号是从脉冲演变而来的，一般是在两个稳定状态之间阶跃式变化的信号，如记录个数的记数信号、灯光闪烁等，传送和处理数字信号的电路称为**数字电路**（Digital Circuit）。

【例 1.0.1】 判断下列仪表或设备反映的物理量哪些是模拟量？哪些是数字量？

(1) 水银温度计；(2) 产品数量统计仪表；(3) 电阻箱；(4) 收音机音量旋钮控制。

解 (1) 为模拟量；

(2) 为数字量；

(3) 为数字量；

(4) 为模拟量。

数字电路广泛应用在通信系统、测量仪表、控制装置和电子计算机等领域。数字手表、数字电视、数字通信、数字控制……数字化已成为当今现代电子技术的发展潮流。

与模拟电路相比，数字电路有很多优点：

(1) 数字电路易于设计，使用方便；因为数字电路采用开关电路，它不要求物理量的精确数值，只要求物理量的范围。

(2) 数字电路便于信息的储存和传输；数字电路的信息储存是由特定的开关电路完成的。根据需要，开关电路就能将信息锁存并保持下来，数字信号可以无限地长期存储。

(3) 数字电路的准确度和精确度高；数字电路可以控制精确数字，只需按要求增减开关电路就可以；

(4) 数字电路工作可靠性高，抗干扰能力强。在数字电路中，因为不要求物理量的准确值。所以只要干扰不影响对高、低电平的分分就可以。

(5) 数字电路便于实现程控；便于采用数字计算机和微处理器来处理信息和参与控制。

(6) 数字电路便于集成化、系列化生产，成本低。

数字电路的特点：工作信号是离散的数字信号；在稳定状态时，电子器件(如二极管、三

极管)均工作在开关状态,即工作在饱和区和截止区;数字电路研究的主要问题是输入和输出之间的**逻辑关系**;主要分析工具是逻辑代数。

数字电路的分类:根据电路结构不同,数字电路可分为分立组件电路和集成电路两种;**集成度**是指每一芯片所包含的晶体管的个数,根据集成度不同,数字电路可分为:

- (1) 小规模集成电路 (SSI, 每片数十个器件);
- (2) 中规模集成电路 (MSI, 每片数百个器件);
- (3) 大规模集成电路 (LSI, 每片数千个器件);
- (4) 超大规模集成电路 (VLSI, 每片器件数目大于 1 万)。

根据电路制造工艺不同,数字电路主要分为**双极型**(TTL 电路)和**单极型**(CMOS 电路)两种;近年来,可编程逻辑器件(PLD)特别是现场可编程门阵列(FPGA)的飞速进步,使数字电子技术开创了新局面,不仅规模大,而且将硬件与软件相结合,使器件的功能更加完善,使用也更加灵活。

根据电路的工作原理不同,数字电路可分为**组合逻辑电路**和**时序逻辑电路**两种。

随着数字电子技术的迅速发展,尤其是计算机的普遍应用,用数字电路和计算机来处理模拟信号的系统越来越多。**数字系统的构成**如图 1.0.1 所示。

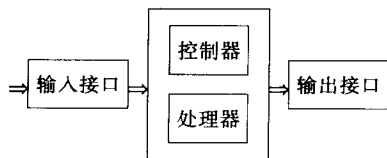


图 1.0.1 数字系统框图

数字系统通常由输入接口、输出接口、数据处理和控制器构成。输入接口和输出接口的主要任务是将模拟量转换为数字量,或将数字量转换为模拟量,处理器的主要作用是控制系统内部各部件的工作,使它们按照一定的程序操作。通常以是否有控制器作为区分功能部件和数字系统的标志,凡是包含控制器且能够按顺序进行

操作的系统,无论规模大小,一律称为**数字系统**。

第一节 数制和代码

一、数制

数制(Digital Number Systems)即计数体制,也就是计数方法。日常生活中最常用的是十进制数,而数字系统和计算机中主要采用的是二进制数。另外还有十六进制数和八进制数。

1. 十进制

十进制(Decimal)数采用 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 十个不同的数码来表示任何一个数。进位规律是“逢十进一,借一当十”,其基数是 10。各数码在不同数制时,所代表的数值是不同的。

位权:在一个进位计数制表示的数中,处于不同数位的数码,代表不同的数值,某一个数位的数值是由这一位数码的值乘上处于这位的一个固定常数,不同数位上的固定常数称为位权值,简称位权,或权。不同数位有不同的位权值,例如: $286.32_{(10)} = 2 \times 10^2 + 8 \times 10^1 + 6 \times 10^0 + 3 \times 10^{-1} + 2 \times 10^{-2}$ 。其中, 10^2 、 10^1 、 10^0 、 10^{-1} 、 10^{-2} 等分别称为十进制数各数位的权,都是 10 的幂。任何一个十进制数都可以写成以 10 为底的幂之和的形式,即:

$$(N)_{10} = K_{n-1} \times 10^{n-1} + K_{n-2} \times 10^{n-2} + \cdots + K_0 \times 10^0 + K_{-1} \times 10^{-1}$$

$$+ K_{-2} \times 10^{-2} + \cdots + K_{-m} \times 10^{-m} = \sum_{i=-m}^{n-1} K_i \times 10^i$$

式中, i 为数字中各数码 K 的位置号, 式中 K_i 为基数“10”的第 i 次幂的系数。小数点前的第一位 $i=0$, 第二位 $i=1$ 。依次类推, 小数点后第一位 $i=-1$, 第二位 $i=-2$, $\cdots$ 。 10^i 为第 i 位的权。

从数字电路的角度来看, 采用十进制是不方便的, 因为构成数字电路的基本思路是把电路的状态与数码对应起来, 而十进制的十个数码, 必须有十个不同的而且能严格区分的电路状态与之对应起来, 这样将在技术上带来许多困难且不经济, 因而在数字电路中一般不直接采用十进制, 而采用二进制。

2. 二进制

二进制 (Binary) 数采用 0、1 两个数码来表示任何一个数。进位规律是“逢二进一, 借一当二”, 其基数是 2。其中, 2^2 、 2^1 、 2^0 、 2^{-1} 、 2^{-2} 等分别称为二进制数各数字的权, 都是 2 的幂。二进制数也可以按权展开, 即:

$$(N)_2 = K_{n-1} \times 2^{n-1} + K_{n-2} \times 2^{n-2} + \cdots + K_0 \times 2^0 + K_{-1} \times 2^{-1} + K_{-2} \times 2^{-2} + \cdots + K_{-m} \times 2^{-m} = \sum_{i=-m}^{n-1} K_i \times 2^i$$

例如: $(1101.01)_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$

二进制的优点:

(1) 二进制的数字装置简单可靠, 应用元件少; 二进制只有两个数码 0 和 1, 因此, 它的每一位都可以用任何具有两个不同稳定状态的元件来表示, 如三极管的饱和和截止, 继电器结点的闭合和断开, 灯泡的亮和不亮等。只要规定一种状态表示 1, 另一种状态表示 0, 就可以表示二进制数。这样, 数码的存储、分析和传输, 就可以用最简单而可靠的方式进行。

(2) 二进制的基本运算规则简单, 与十进制数的运算相似, 运算操作方便。

二进制的缺点: 用二进制表示一个数时, 位数多, 使用起来不方便也不习惯, 因此在运算时, 原始数据多用人们习惯的十进制, 在送入计算机时, 就必须将十进制数据转换成数字系统能接受的二进制数, 而运算结束后再将二进制数转换为十进制数, 表示最终结果。

3. 十六进制

十六进制 (Hexadecimal) 采用 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F 十六个不同的数码来表示任何一个数, 符号 A~F 分别代表十进制的 10~15。进位规律是“逢十六进一, 借一当十六”, 其基数是 16。每个数字的权是 16 的幂。十六进制按权展开为:

$$(N)_{16} = K_{n-1} \times 16^{n-1} + K_{n-2} \times 16^{n-2} + \cdots + K_0 \times 16^0 + K_{-1} \times 16^{-1} + K_{-2} \times 16^{-2} + \cdots + K_{-m} \times 16^{-m} = \sum_{i=-m}^{n-1} K_i \times 16^i$$

例如: $(3DA)_{16} = 3 \times 16^2 + D \times 16^1 + A \times 16^0 = 3 \times 16^2 + 13 \times 16^1 + 10 \times 16^0 = (986)_{10}$

此外还有八进制 (**Octal**), 其采用 0, 1, 2, 3, 4, 5, 6, 7 八个不同的数码来表示任何一个数。进位规律是“逢八进一, 借一当八”, 其基数是 8。每个数字的权是 8 的幂。八进制按权展开为:

$$(N)_8 = \sum_{i=-m}^{n-1} K_i 8^i$$

二、数制的转换

各种数制之间可以互相转换，以方便设计和使用。

1. 任意进制转换成十进制

任意进制转换成十进制的方法是按权展开，求和即可。

例如： $(11010)_2 = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 26$

$(11010.11)_2 = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = (26.75)_{10}$

$(1AB)_{16} = 1 \times 16^2 + A \times 16^1 + B \times 16^0 = 1 \times 16^2 + 10 \times 16^1 + 11 \times 16^0 = (427)_{10}$

$(245)_8 = 2 \times 8^2 + 4 \times 8^1 + 5 \times 8^0 = (165)_{10}$

2. 十进制转换成任意进制

十进制数整数转换为任意进制数都可以采用“除基取余法”，即“除基数，得余数，从低位到高位排列”。其步骤为：

- (1) 将给定的十进制数除以基数，余数就是欲转换进制数的最低位；
- (2) 将上一步得到的商继续除基数，余数即是次低位；
- (3) 重复用得到的商除基数，直至商为0，此时的余数为最高位。

【例 1.1.1】将十进制数 $(13)_{10}$ 转换为二进制数。

解 这里基数是2。

13	6	3	1	0	商
÷ 2	1	0	1	1	余数

故 $(13)_{10} = (1101)_2$

【例 1.1.2】将十进制数 $(32)_{10}$ 转换为十六进制数。

解 这里基数是16。

32	2	0	商
÷ 16	0	2	余数

故 $(32)_{10} = (20)_{16}$

十进制数小数转换为任意进制小数可以采用“乘基取整法”。即“乘基数，取整数，从高位到低位排列”。步骤为：

- (1) 将给定的十进制小数乘以要转换的数制的基数，其乘积的整数就是欲转换进制数的最高位；
- (2) 将上一步得到的乘积的小数部分继续乘以基数，乘积的整数部分即是次高位；
- (3) 重复用得到的积乘基数，直到其纯小数部分为0或者满足一定误差要求为止。

【例 1.1.3】将十进制数 $(0.135)_{10}$ 转换为二进制数（精确到第五位）。

解 $0.135 \times 2 = 0.270 \cdots 0$ 最高位

$0.270 \times 2 = 0.540 \cdots 0$

$$0.540 \times 2 = 1.080 \cdots 1$$

$$0.080 \times 2 = 0.160 \cdots 0$$

$$0.160 \times 2 = 0.320 \cdots 0 \quad \text{最低位}$$

$$\text{则 } (0.135)_{10} = (0.00100)_2。$$

3. 二进制与十六进制、八进制之间的转换

因为每一个十六进制数码都可以用 4 位二进制数来表示, 所以可以将二进制数每 4 位一组, 写出各组的数值。如果是整数, 从右至左划分, 从左至右读写, 就是十六进制数。注意整数按 4 位一组划分时, 最高位一组不够 4 位时用 0 补齐。如果是小数, 从小数点后第一位从左至右划分, 从左至右读写, 最低位一组不够 4 位时用 0 补齐。

同理, 十六进制转换为二进制时, 可将十六进制数的每一位写成 4 位二进制数, 不改变顺序, 即可将十六进制转换为二进制。

【例 1.1.4】 将二进制数 $(1101010)_2$ 转换为十六进制数。

$$\text{解 } (1101010)_2 = (0110, 1010)_2 = (6A)_{16}$$

【例 1.1.5】 将二进制数 $(0.101011)_2$ 转换为十六进制数。

$$\text{解 } (0.101011)_2 = (0.1010, 1100)_2 = (0.AC)_{16}$$

【例 1.1.6】 将十六进制数 $(A.3B)_{16}$ 转换为二进制数。

$$\text{解 } (A.3B)_{16} = (1010.0011, 1011)_2$$

十进制转换为十六进制也可以先转换为二进制, 由二进制转换为十六进制。

因为每一个八进制数码都可以用 3 位二进制数来表示, 所以可以将二进制数每 3 位一组, 写出各组的数值。如果是整数, 从右至左划分, 从左至右读写, 就是八进制数。注意整数按 3 位一组划分时, 最高位一组不够 3 位时用 0 补齐。如果是小数, 从小数点后第一位从左至右划分, 从左至右读写, 最低位一组不够 3 位时用 0 补齐。

同理, 八进制转换为二进制时, 可将八进制数的每一位写成 3 位二进制数, 不改变顺序, 即可将八进制转换为二进制。

【例 1.1.7】 将二进制数 $(10110.1011)_2$ 转换为八进制数。

$$\text{解 } (10110.1011)_2 = (010, 110.101, 100)_2 = (26.54)_8$$

【例 1.1.8】 将八进制数 $(3.7)_8$ 转换为二进制数。

$$\text{解 } (3.7)_8 = (011.111)_2$$

另外十进制与十六进制、八进制之间的转换, 也可以利用二进制为中介。先把十进制数转换为二进制, 再利用二进制与十六进制、八进制的转换关系得到。

三、码制

数字系统中的信息可分为两类, 一类是数值, 另一类是文字符号 (包括控制符)。为了表示文字符号信息, 往往也采用一定位数的二进制码表示, 这个特定的二进制码称为**代码**。建立代码与十进制数、字母、符号的一一对应关系的方法称为**编码**。不同的编码方式称为**码制**。常用的编码有二-十进制码 (BCD 码)、格雷码、余 3 码及字符代码等。

1. 二-十进制码

用二进制代码表示一个给定的十进制数 0~9, 称为二-十进制编码, 简称**BCD 码** (Binary Coded Decimal)。表 1.1.1 给出了几种常用的 BCD 码。

表 1.1.1

常用的 BCD 码

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

因为 4 位二进制代码共有 16 个不同的组合，用它对 0~9 十个十进制数编码总有 6 个不用的状态，叫它无关状态，或称为伪码。例如 8421 码中的 1010~1111 为 6 个伪码。

BCD 码分为有权码和无权码。

有权码都是将自然 4 位二进制数的 16 个组合去掉 6 个而得到的，只不过舍去的组合不同，被保留的十个组合中的每一位都是有位权的，如表 1.1.1 中的 8421 码 $b_3b_2b_1b_0$ ，每位都有相应的位权值，如 b_0 的位权为 $2^0 = 1$ ， b_1 的位权为 $2^1 = 2$ ， b_2 的位权为 $2^2 = 4$ ， b_3 的位权为 $2^3 = 8$ ，由于每位的位权值分别为 8，4，2，1，所以这种代码称为 8421BCD 码。它们的权展开式的计算结果分别对应十个阿拉伯数字，因而也叫二-十进制码。

表 1.1.1 中，余 3 码和格雷码为无权码。

余 3 码是由 8421 码加 3 (0011) 得到的，不能用权展开式来表示其转换关系。

格雷码的特点是按照“相邻性”编码的，即相邻两码之间只有一位数字不同。一般可在下面情况下使用：如果用其他代码转换时，若代码的变化位数多于一位时可能产生错误或模糊的结果。例如当二进制代码从 0111 转换成 1000 时，需要所有的位数都变化，不同位数的过渡时间可能有较大区别，它取决于构成这些位的器件或电路。因此从 0111 到 1000 可能出现一个或几个中间状态。如果最高位变化的快，将会出现如下过渡状态：

0111	十进制数 7
1111	错误码
1000	十进制数 8

虽然 1111 状态的出现只是暂时的，但由这些位所控制的器件就可能出现误操作。使用格雷码是因为每次变换只有一位发生变化，各位之间不会出现竞争，可以避免这种错误。格雷码还常用于模拟量与数字量的转换。

还有其他编码方法，如奇偶校验码、汉明码等等。

2. 字符编码

常用的字母和字符编码有 ASCII 码和 ISO 码。ASCII 码是美国标准信息交换码的简称，其编码如表 1.1.2 所示。这是一组 8 位二进制代码，用 $b_0 \sim b_6$ 7 位表示 $2^7 = 128$ 个不同的字符信息，比如键盘上的每个英文字符都可以用特定的 ASCII 码表示。第 8 位为奇偶校验位。

表 1.1.2

ASCII 编码

字符 $b_3 b_2 b_1 b_0$	$b_6 b_5 b_4$	000	001	010	011	100	101	110	111
0000		NUL	DLE	SP	0	@	P	\	p
0001		SOH	DC1	!	1	A	Q	a	q
0010		STX	DC2	"	2	B	R	b	r
0011		ETX	DC3	#	3	C	S	c	s
0100		EOT	DC4	\$	4	D	T	d	t
0101		ENQ	NAK	%	5	E	U	e	u
0110		ACK	SYN	&	6	F	V	f	v
0111		BEL	ETB	,	7	G	W	g	w
1000		BS	CAN	(	8	H	X	h	x
1001		HT	EM	)	9	I	Y	i	y
1010		LF	SUB	*	:	J	Z	j	z
1011		VT	ESC	+	;	K	[	k	{
1100		FF	FS	,	>	L	\	l	
1101		CR	GS	-	=	M	]	m	~
1110		SO	RS	。	<	N	^	n	~
1111		SI	US	/	?	O	-	o	DEL

【例 1.1.9】下面是按 ASCII 码所编的信息表示什么含义？

1001000 1000101 1001100 1010000

解 从表 1.1.2 中可查出每个七位代码所代表的字母，结果是 HELP。

ASCII 码常用在计算机和输入/输出装置（如视频端或打印机）之间字母和数字的转换。

第二节 逻辑代数基础

一、逻辑变量与逻辑函数

广义的讲，逻辑就是规律。逻辑代数 (Logic Algebra) 也称为布尔代数，它是一种描述事物逻辑关系的数学方法，是研究逻辑电路的数学工具。逻辑代数中的变量和普通代数中的变量一样，也由字母表示。在对实际问题进行逻辑抽象时，我们一般称决定事物的原因为逻辑自变量，而称被决定事物的结果为逻辑因变量。

以某种形式表达的逻辑自变量和逻辑因变量的函数关系称为逻辑函数 (Logic Function)。它是由逻辑变量、常量通过运算符连接起来的代数式。一般写作

$$L = F(A, B, C, D, \dots) \quad (1.2.1)$$

与普通代数不同的是，逻辑代数的变量只有 0 和 1 两个取值。而且这里的“0”和“1”不表示数值的大小，只表示两种相互对立的逻辑状态。如：用“1”和“0”表示灯的亮和灭、门的开与关、电平的高与低等等。因此，常把“1”状态称为逻辑 1，“0”状态称为逻辑 0。逻辑代数有一系列的定律和规则，用它们对逻辑表达式进行处理，可以完成电路的化简、变换、分析和设计。

二值数字逻辑的产生，是基于客观世界的许多事物可以用彼此相关又互相对立的两种状态来描述，例如，是与非、真与假、开与关、低与高，如此等等。而且在电路上，可以用电子器件的开关特性来实现，由此形成离散信号电压或数字电压。这些数字电压通常用逻辑电平来表示，如高电平、低电平。应当注意，逻辑电平不是物理量，而是物理量的相对表示。

由于逻辑代数可以使用二值函数进行逻辑运算,一些用语言描述显得十分复杂的逻辑命题,使用数学语言后,就变成了简单的代数式。逻辑电路中的一个命题,不仅包含“肯定”和“否定”两重含义,而且包含条件与结果的多种组合。

在数字电路中,有两种逻辑体制,即**正逻辑体制**和**负逻辑体制**。若用逻辑“1”表示电路中的高电平,用逻辑“0”表示电路中的低电平,用H对应二进制的“1”,用L对应二进制的“0”,称为正逻辑体制;反之,称为负逻辑体制。对逻辑变量的逻辑状态采用不同的逻辑体制,所得到的逻辑函数也就不同。在一个数字电路中一般只能使用一种逻辑体制,混用时必须有严格的分界面,一般情况下采用**正逻辑体制**。

二、逻辑关系和运算

基本逻辑关系有**与逻辑**、**或逻辑**和**非逻辑**三种。相应的逻辑运算有**与运算**、**或运算**和**非运算**。

1. 与逻辑

与逻辑即**逻辑乘**。在图1.2.1(a)所示指示灯控制电路中,开关A、B如果有一个断开或者两个都断开,指示灯不亮;只有当两个开关都闭合时,指示灯才亮。指示灯的亮灭与开关的通断存在的这种逻辑关系,即只有决定事物结果(灯亮)的几个条件全都具备时,这种结果才会发生。这种逻辑关系称为“与”逻辑。

在数字电路中,研究的主要对象是输入变量和输出变量之间的逻辑关系,把输入变量可能的取值组合状态及其对应的输出状态列成表格,经逻辑赋值称为**真值表**。用真值表可直观地表示电路的输出与输入之间的逻辑关系。若用逻辑“1”表示开关闭合、指示灯亮,用逻辑“0”表示开关断开、指示灯灭,则与逻辑的真值表如表1.2.1所示。

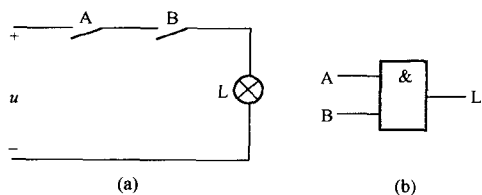


图 1.2.1 与逻辑电路及与门的逻辑符号

(a) 与逻辑电路; (b) 与门的逻辑符号

表 1.2.1 与逻辑真值表和逻辑规律

A	B	L	逻辑规律
0	0	0	有0为0 全1为1
0	1	0	
1	0	0	
1	1	1	

为便于分析和运算,通常用代数式表示逻辑关系,称为**逻辑表达式**。

与逻辑的逻辑表达式为

$$L = A \cdot B \quad \text{或} \quad L = AB \quad (1.2.2)$$

式中的“ $\cdot$ ”读作“与”。由真值表可知

$$0 \cdot 0 = 0 \quad 0 \cdot 1 = 0 \quad 1 \cdot 0 = 0 \quad 1 \cdot 1 = 1$$

与逻辑允许有两个或两个以上的输入变量,实现与逻辑运算的电路称为**与门**。与门的逻辑符号如图1.2.1(b)所示。

2. 或逻辑

或逻辑即**逻辑加**。在图1.2.2(a)所示指示灯控制电路中,开关A、B如果有一个闭合或者两个都闭合,指示灯亮;只有当两个开关都断开时,指示灯才不亮。指示灯的亮灭与开关的通断存在的这种逻辑关系,即在决定事物结果(灯亮)的几个条件中,只要有一个或一个以上条件满足时,结果就会发生。这种逻辑关系称为“或”逻辑。其真值表如表1.2.2表示。

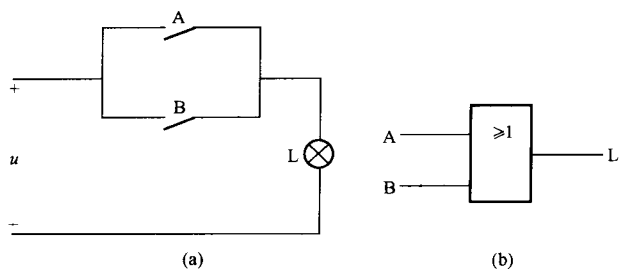


图 1.2.2 或逻辑电路及或门的逻辑符号

(a) 或逻辑电路；(b) 或门的逻辑符号

表 1.2.2 或逻辑真值表和逻辑规律

A	B	L	逻辑规律
0	0	0	有 1 为 1 全 0 为 0
0	1	1	
1	0	1	
1	1	1	

或逻辑的逻辑表达式为

$$L = A + B \quad (1.2.3)$$

式中的“+”读作“或”。从真值表可知

$$0 + 0 = 0 \quad 0 + 1 = 1 \quad 1 + 0 = 1 \quad 1 + 1 = 1$$

或逻辑允许有两个或两个以上的输入变量，实现或逻辑运算的电路称为或门。或门的逻辑符号如图 1.2.2 (b) 所示。

3. 非逻辑

非逻辑即逻辑非。在图 1.2.3 (a) 所示指示灯控制电路中，开关 A 闭合时，指示灯不亮；当开关 A 断开时，指示灯才亮。指示灯的亮灭与开关的通断存在的这种逻辑关系，即当决定事物结果（灯亮）的条件具备时，结果不发生；而当条件不具备时，结果才会发生。这种逻辑关系称为“非”逻辑，其真值表见表 1.2.3 所示。

非逻辑的逻辑表达式为

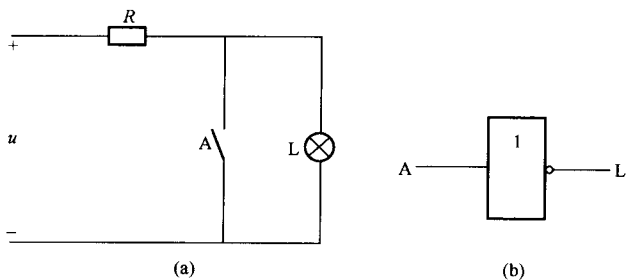


图 1.2.3 非逻辑电路及非门的逻辑符号

(a) 非逻辑电路；(b) 非门的逻辑符号

表 1.2.3 非逻辑真值表和逻辑规律

A	L	逻辑规律
0	1	有 0 为 1
1	0	有 1 为 0

$$L = \bar{A} \quad (1.2.4)$$

式中的“ $\bar{A}$ ”读作“A 非”或“A 反”。由真值表可知

$$\bar{0} = 1 \quad \bar{1} = 0$$

非逻辑只允许有一个逻辑变量。实现非逻辑运算的电路称为非门，也叫反相器。非门的逻辑符号如图 1.2.3 (b) 所示。

4. 复合逻辑

复合逻辑由基本逻辑组合而成。常见的复合逻辑有：与非、或非、与或非、异或和同或这五种。

对应的运算电路称为与非门、或非门、与或非门、异或门和同或门。其真值表、逻辑表达式和逻辑规律如表 1.2.4~表 1.2.8 所示。其逻辑符号如图 1.2.4~图 1.2.8 所示。

从表 1.2.6 和 1.2.7 可看出，异或逻辑和同或逻辑在逻辑上互为反函数，即

$$A \oplus B = \overline{A \odot B} \quad (1.2.5)$$

$$A \odot B = \overline{A \oplus B} \quad (1.2.6)$$

表 1.2.4 与非逻辑真值表、逻辑表达式和逻辑规律

逻辑表达式	真值表			逻辑规律
$L = \overline{AB}$	A	B	L	有 0 为 1 全 1 为 0
	0	0	1	
	0	1	1	
	1	0	1	
	1	1	0	

表 1.2.5 或非逻辑真值表、逻辑表达式和逻辑规律

逻辑表达式	真值表			逻辑规律
$L = \overline{A + B}$	A	B	L	有 1 为 0 全 0 为 1
	0	0	1	
	0	1	0	
	1	0	0	
	1	1	0	

表 1.2.6 异或逻辑真值表、逻辑表达式和逻辑规律

逻辑表达式	真值表			逻辑规律
$L = A\overline{B} + \overline{A}B$ $= A \oplus B$	A	B	L	不同为 1 相同为 0
	0	0	0	
	0	1	1	
	1	0	1	
	1	1	0	

表 1.2.7 同或逻辑真值表、逻辑表达式和逻辑规律

逻辑表达式	真值表			逻辑规律
$L = AB + \overline{A}\overline{B}$ $= A \odot B$	A	B	L	不同为 0 相同为 1
	0	0	1	
	0	1	0	
	1	0	0	
	1	1	1	

表 1.2.8 与或非逻辑真值表、逻辑表达式和逻辑规律

逻辑表达式	真值表				逻辑规律	
$L = \overline{AB + CD}$	A	B	C	D	L	两组输入均有 0 为 1 一组输入全 1 时为 0
	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	
	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	

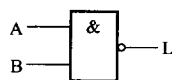


图 1.2.4 与非门逻辑符号

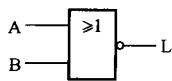


图 1.2.5 或非门逻辑符号

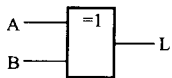


图 1.2.6 异或门逻辑符号

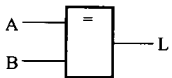


图 1.2.7 同或门逻辑符号

每个异或和同或逻辑门只允许有两个输入变量。例如：若要实现 $A \oplus B \oplus C$ 逻辑函数，必须用两个异或门，如图 1.2.9 所示。

三、逻辑函数的表示方法及其相互转换

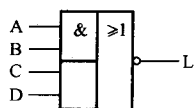


图 1.2.8 与或非门逻辑符号

在逻辑函数中，各逻辑变量之间的逻辑关系可以用逻辑表达式来表示。逻辑表达式右边的字母 A、B、C、D…称为输入逻辑变量，左边的字母 L 称为输出逻辑变量。字母上面没有非运算符的称为原变量，有非运算符的称为反变量。即为

$$L = F(A, B, C, D, \dots) \quad (1.2.7)$$