21世纪高校规划教材

数字电子技术

主编　王连英　章小印

江西高校出版社

21 世纪高校规划教材

数字电子技术

主　编　王连英　章小印
副主编　江路明　兰巧云　樊辉娜

江 西 高 校 出 版 社

图书在版编目(CIP)数据

数字电子技术/王连英,章小印主编 .—南昌:江西高校出版社,2007.8

ISBN 978-7-81075-968-7

Ⅰ.数… Ⅱ.①王… ②章… Ⅲ.数字电路-电子技术 Ⅳ.TN79

中国版本图书馆 CIP 数据核字(2007)第 126041 号

出版发行	江西高校出版社
社　　址	江西省南昌市洪都北大道 96 号
邮政编码	330046
电　　话	(0791)8529392,8504319
网　　址	www.juacp.com
印　　刷	江西教育印刷厂
照　　排	江西太元科技有限公司照排部
经　　销	各地新华书店
开　　本	787mm×1092mm　1/16
印　　张	19.5
字　　数	474 千字
版　　次	2007 年 8 月第 1 版第 1 次印刷
印　　数	1～2000 册
书　　号	ISBN 978-7-81075-968-7
定　　价	28.00 元

版权所有　侵权必究

前 言

本书是为高职高专电类、机电类专业编写的教学用书，可作为《数字电子技术》和《电子技术基础》数字部分课程的教材，也可供从事电子技术工作的工程技术人员参考。

《数字电子技术》是一门发展迅速、实践性和应用性很强的电子技术基础课程。为提高学生的实际动手能力，从培养生产一线的高级技术人才的目的出发，以培养学生的综合工作能力为线索，结合我们多年从事数字电子技术课程教学的改革和实践编写了本教材。编写本教材的指导思想是：注重基本理论、基本分析方法的介绍和应用，理论联系实际，以设计制作为课题，以能力培养为目的。

本书主要内容有逻辑代数基础、基本门电路、组合逻辑电路、时序逻辑电路、脉冲波形的产生与变换、半导体存储器与可编程器件、模数与数模转换器、课程设计与制作等。为配合教学每章都有小结、习题及思考题、自我测试、实验与技能操作训练。

为了能够适应新技术发展的需要，本书压缩、精简了传统分立元件和集成电路内部电路的内容，以较多的篇幅介绍了集成电路新元件、新技术的设计和应用，根据教学的实际需要可进行适当的选择。最后在附录A还简要介绍了电子电路仿真(EWB)软件的使用方法，理论上使用EWB可以实现本教材的全部实验，视需要可选择其中的部分内容安排学生进行仿真技术训练。

本书由王连英、章小印任主编，江路明、兰巧云、樊辉娜任副主编，朱彩莲、冯小玲参编。其中，江西应用技术职业学院的江路明编写了第6章、第8章；江西机电职业技术学院的樊辉娜编写了第4章；江西工业工程职业技术学院的朱彩莲编写了第3章，章小印编写了第5章；江西信息应用职业技术学院的兰巧云编写了第7章；江西现代职业技术学院的冯小玲编写了第2章，王连英编写了第1章和附录。全书由江西现代职业技术学院王连英和江西工业工程职业技术学院章小印统稿。

本书在编写过程中参阅了国内部分知名专家、学者编写的数字电子技术课程的教材，甚至直接引用了其中一些经典论述，在此表示诚挚的感谢。

由于编者水平有限，时间匆忙，书中难免有疏漏和错误之处，恳请读者批评指正。

编 者

2007年5月

目　录

此为试读，需要完整PDF请访问：www.ertongbook.com

第1章　逻辑代数基础

电子系统中一般含有模拟和数字两种构件。

模拟电路处理的信号是模拟信号，模拟信号在时间和幅值上都是连续变化的。

数字电路处理的信号是数字信号，数字信号在时间和幅值上都是离散变化的。

从本章开始，我们将着重介绍有关数字电路的基础理论及其实际应用技术。

1.1　数制与编码

1.1.1　数制及其转换

1. 数制

数制即计数进位制的简称。日常生活中，人们最熟悉的是十进制，但是十进制并非唯一的计数方法，如一年等于十二个月，一分钟等于六十秒等。

数字电路中广泛采用的是二进制。在二进制数中，每一位仅有 **0** 和 **1** 两个可能的数码，计数基数为 2。但如果数值很大，用二进制表示的位数会很多，既难记忆又不便于读写，故又常采用八进制和十六进制。

(1)十进制

① 十进制数采用 0、1、2、3、4、5、6、7、8、9 这十个不同数码来表示。通常把数制中所有的数码个数称为基数。十进制的基数为 10。

② 十进制的计数规律是"逢十进一"。

③ 任意一个十进制数可展开为

$$\begin{aligned}(N)_{10} &= K_{n-1}\times 10^{n-1} + K_{n-2}\times 10^{n-2} + \cdots + K_1\times 10^1 + K_0\times 10^0 + K_{-1}\times 10^{-1} \\ &\quad + \cdots + K_{-m}\times 10^{-m} \\ &= \sum_{i=-m}^{n-1} K_i \times 10^i \qquad (1.1.1)\end{aligned}$$

其中 i 表示位数，m 为小数位数，n 为整数位数，m、n 为正整数；K_i 表示第 i 位的数码，可以是 0 ～ 9 十个数码中的任何一个；10^i 表示以基数 10 为底的 i 次幂，我们称为第 i 位的权。式(1.1.1)称为十进制数的位权展开式。

例如，十进制数 858.38 的展开式为

$$(858.38)_{10} = 8\times 10^2 + 5\times 10^1 + 8\times 10^0 + 3\times 10^{-1} + 8\times 10^{-2}$$

虽然上数中有三个数码都是"8"，但最左的一位是百位数，表示 800，即 8×10^2，它的权值为 10^2；中间的一位是个位数，表示 8，即 8×10^0，它的权值为 10^0；最右边的一位是小数点后两位数，表示 0.08，即 8×10^{-2}，它的权值为 10^{-2}。数码与权的乘积称为加权系数，如上述的 8×10^2、8×10^0、8×10^{-2}。所以同一数码所处位置不同，代表的数值大小也不同。

④ 任何一个十进制数，例如 858.38，可以书写成 858.38，或 $(858.38)_{10}$，或 858.38_D，或

858.38D 等形式。

(2) 二进制

① 二进制数采用**0**、**1** 这两个数码来表示。二进制的基数为 2。

② 二进制的计数规律是“逢二进一”,例如 $\mathbf{1+1=10}$。

③ 任意一个二进制数可展开为

$$
\begin{aligned}
(N)_2 &= K_{n-1}\times 2^{n-1}+K_{n-2}\times 2^{n-2}+\cdots+K_1\times 2^1+K_0\times 2^0+K_{-1}\times 2^{-1}\\
&\quad+\cdots+K_{-m}\times 2^{-m}\\
&=\sum_{i=-m}^{n-1}K_i\times 2^i \qquad (1.1.2)
\end{aligned}
$$

i 表示位数,m 为小数位数,n 为整数位数,m、n 为正整数;K_i 表示第 i 位的数码,可以是 **0**、**1** 两个数码中的任何一个;2^i 表示以基数 2 为底的 i 次幂,我们称为第 i 位的权。式(1.1.2) 称为二进制数的位权展开式。

例如,二进制数 **101.01** 的展开式为

$$
\begin{aligned}
(\mathbf{101.01})_2 &= \mathbf{1}\times 2^2+\mathbf{0}\times 2^1+\mathbf{1}\times 2^0+\mathbf{0}\times 2^{-1}+\mathbf{1}\times 2^{-2}\\
&=(4+0+1+0+0.25)_{10}\\
&=(5.25)_{10}
\end{aligned}
$$

二进制数$(\mathbf{101.01})_2$ 对应的十进制数为 5.25。

④ 任何一个二进制数例如 101.01,可以书写成$(\mathbf{101.01})_2$,或 $\mathbf{101.01}_B$,或 **101.01**B 等形式。

(3) 八进制

① 八进制数采用 0、1、2、3、4、5、6、7 这八个不同数码来表示。八进制的基数为 8。

② 八进制的计数规律是“逢八进一”,例如 $7+1=10$。

③ 任意一个八进制数可展开为

$$
\begin{aligned}
(N)_8 &= K_{n-1}\times 8^{n-1}+K_{n-2}\times 8^{n-2}+\cdots+K_1\times 8^1+K_0\times 8^0+K_{-1}\times 8^{-1}\\
&\quad+\cdots+K_{-m}\times 8^{-m}\\
&=\sum_{i=-m}^{n-1}K_i\times 8^i \qquad (1.1.3)
\end{aligned}
$$

i 表示位数,m 为小数位数,n 为整数位数,m、n 为正整数;K_i 表示第 i 位的数码,可以是 0 ~ 7 八个数码中的任何一个;8^i 表示以基数 8 为底的 i 次幂,我们称为第 i 位的权。式(1.1.3) 称为八进制数的位权展开式。

例如,八进制数 57.2 的展开式为

$$
\begin{aligned}
(57.2)_8 &= 5\times 8^1+7\times 8^0+2\times 8^{-1}\\
&=(40+7+0.25)_{10}\\
&=(47.25)_{10}
\end{aligned}
$$

八进制数$(57.2)_8$ 对应的十进制数为 47.25。

④ 任何一个八进制数例如 57.2,可以书写成$(57.2)_8$,或 57.2_O,或 57.2O 等形式。

(4) 十六进制

① 十六进制数采用 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F 这十六个不同数码来表示。十六进制的基数为 16。

此为试读,需要完整PDF请访问: www.ertongbook.com

② 十六进制的计数规律是"逢十六进一",例如 $F+1=10$。

③ 任意一个十六进制数可展开为

$$(N)_{16} = K_{n-1}\times 16^{n-1} + K_{n-2}\times 16^{n-2} + \cdots + K_1\times 16^1 + K_0\times 16^0 + K_{-1}\times 16^{-1} + \cdots + K_{-m}\times 16^{-m} = \sum_{i=-m}^{n-1} K_i \times 16^i \tag{1.1.4}$$

i 表示位数,m 为小数位数,n 为整数位数,m、n 为正整数;K_i 表示第 i 位的数码,可以是 0～F 十个数码中的任何一个;16^i 表示以基数 16 为底的 i 次幂,我们称为第 i 位的权。式(1.1.4)称为十六进制数的位权展开式。

例如,十六进制数 2A.8 的展开式为

$$\begin{aligned}(2A.8)_{16} &= 2\times 16^1 + A\times 16^0 + 8\times 16^{-1}\\ &= (32+10+0.5)_{10}\\ &= (42.5)_{10}\end{aligned}$$

十六进制数$(2A.8)_{16}$对应的十进制数为 42.5。

④ 任何一个十六进制数例如 2A.8,可以书写成 $(2A.8)_{16}$,或 $2A.8_H$,或 2A.8H 等形式。

表 1.1.1 给出了一组各种进位制数间的对应关系。

表 1.1.1　几种进位制的对应关系表

十进制	二进制	八进制	十六进制
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

2. 数制间的相互转换

(1) 二制数和其他进制数转换成十进制数

根据式(1.1.2)把二进制数按权位展开,然后把各项数值按十进制数相加,即可得到对

应的十进制数。

[例 1.1.1]　将二进制数 **1011.101** 转换为十进制数。

[解]　$(\mathbf{1011.101})_B = \mathbf{1}\times 2^3+\mathbf{0}\times 2^2+\mathbf{1}\times 2^1+\mathbf{1}\times 2^0+\mathbf{1}\times 2^{-1}+\mathbf{0}\times 2^{-2}+\mathbf{1}\times 2^{-3}$

$= 8+0+2+1+0.5+0+0.125$

$= (11.625)_D$

其他进制数同样按位权展开相加，即可得到对应的十进制数。

[例 1.1.2]　将八进制数 256 转换为十进制数。

[解]　$(256)_O = 2\times 8^1+5\times 8^0+6\times 8^{-1}$

$= 16+5+0.75$

$= (21.75)_D$

(2)十进制数转换成二进制数

十进制数转换为等值的二进制数，整数部分和小数部分转换方法不同，需分开转换。

① 整数部分采用“除 2 取余法”，即把整数部分逐次用 2 去除，并依次记下余数，一直除到商为 0 为止，然后把全部余数从后向前排列，即为转换后的二进制数。

[例 1.1.3]　把十进制数 59 转换为二进制数。

[解]　过程如下：

2	59		
2	29	………	余数 $=1=k_0$
2	14	………	余数 $=1=k_1$
2	7	………	余数 $=1=k_2$
2	3	………	余数 $=1=k_3$
2	1	………	余数 $=1=k_4$
	0	………	余数 $=1=k_5$

所以$(59)_D=(\mathbf{111011})_B$

② 小数部分采用“乘 2 取整法”，即把小数部分不断地用 2 去乘，所得的整数从小数点开始自左往右排列，直到一定精度为止。

[例 1.1.4]　把十进制数 0.56 转换为二进制数。

[解]　过程如下：

$0.56\times 2=1.12$	取整 $=1$，	$k_{-1}=\mathbf{1}$
$0.12\times 2=0.24$	取整 $=0$，	$k_{-2}=\mathbf{0}$
$0.24\times 2=0.48$	取整 $=0$，	$k_{-3}=\mathbf{0}$
$0.48\times 2=0.96$	取整 $=0$，	$k_{-4}=\mathbf{0}$

如果需要还可以继续下去。若只保留三位小数，对应的乘积纯小数为 0.48，因为 $0.48<0.5$，根据四舍五入原则，$k_{-3}=0$；所以，$(0.56)_D=(\mathbf{0.100})_B$

若保留四位小数，对应的乘积纯小数为 0.96，因为 $0.96>0.5$，根据四舍五入原则

$k_{-4}=\mathbf{0}+\mathbf{1}=\mathbf{1}$，所以$(0.56)_D=(\mathbf{0.1001})_B$

采用类似的方法，很容易把十进制数转换为八进制数、十六进制数，即整数部分采用“除基数(8 或 16)取余法”，小数部分采用“乘基数(8 或 16)取整法”。

(3)二进制数、八进制数、十六进制数之间的互换

因为 8 和 16 都是 2 的整数幂，所以二进制数与八进制数和十六进制数之间的互换是比较容易的。

① 二进制数、八进制数之间的互换

三位二进制数正好表示 0～7 八个数字，因此一个二进制数转换为八进制数时，整数部分从最低位开始，每三位分成一组，每一组对应一位八进制数，若最后不足三位时，应在前面加**0**，补足三位再转换；小数部分从最高位开始，每三位分成一组，每一组对应一位八进制数，若最后不足三位时，应在后面加**0**，补足三位再转换。反之，一个八进制数转换为二进制数时，每一位八进制数对应三位二进制数。

[例 1.1.5] 将二进制数**1011001.1011**转换成八进制数。

[解]

二进制数	**001**	**011**	**001.**	**101**	**100**
	↓	↓	↓	↓	↓
八进制数	1	3	1.	5	4

所以$(\mathbf{1011001.1011})_B=(131.54)_O$

[例 1.1.6] 将八进制数 37.6 转换成二进制数。

[解]

八进制数	3	7.	6
	↓	↓	↓
二进制数	**011**	**111.**	**110**

所以 $(37.6)_O=(\mathbf{11111.11})_B$

② 二进制数、十六进制数之间的互换

四位二进制数正好表示 0～F 十六个数字。因此一个二进制数转换为十六进制数时，整数部分从最低位开始，每四位分成一组，每一组对应一位十六进制数，若最后不足四位时，应在前面加 **0**，补足四位再转换；小数部分从最高位开始，每四位分成一组，每一组对应一位十六进制数，若最后不足四位时，应在后面加 **0**，补足四位再转换。反之，一个十六进制数转换为二进制数时，每一位十六进制数对应四位二进制数。

[例 1.1.7] 将二进制数 **1011001.1011** 转换成十六进制数。

[解]

二进制数	**0101**	**1001.**	**1011**
	↓	↓	↓
十六进制数	5	9.	B

所以 $(\mathbf{1011001.1011})_B=(59.B)_H$

[例 1.1.8] 将十六进制数 C7.3 转换成二进制数。

[解]

十六进制数	C	7.	3
	↓	↓	↓
二进制数	**1100**	**0111.**	**0011**

所以 $(C7.3)_H=(\mathbf{11000111.0011})_B$

1.1.2 编码

在日常生活中，不同的数码组合不仅表示数量的大小，还能表示不同的事物。例如，我们的身份证号码，这些号码并不具备数量大小的含义，仅仅代表不同的人，我们称这类号码

组合称为代码。形成代码的规则称码制，用数字或某种文字和符号来表示某一对象或信号的过程，称为编码。

数字电路中，常用四位二进制数码来表示一位十进制数，这种方法称为二进制编码的十进制数，简称二—十进制编码，或 BCD 码。四位二进制码有 16 种组合，而每位十进制数只需用 10 种组合，因此用四位二进制数码来表示一位十进制数时，有许多的选择方法。表 1.1.2 列出了几种常用的 BCD 码。

表 1.1.2　几种常用的 BCD 码

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

1. 有权 BCD 码

有权 BCD 码的四位二进制代码中每一位数码都有确定的位权值，将四位二进制代码按相应的位权展开，就可得到该代码所对应的十进制数。8421 码、2421 码、5421 码都属于有权码。其中 8421BCD 码是一种最简单、最常用有权码。

［例 1.1.9］　写出$(457.39)_{10}$对应的 8421BCD 码。

［解］　$(457.39)_{10} = (\mathbf{0100\ 0101\ 0111\ .\ 0011\ 1001})_{8421BCD}$

2. 无权 BCD 码

无权 BCD 码就是没有确定的位权值。例如余 3 码是由 8421 码加 3(**0011**)得来的。另一种常用的无权码叫格雷码，格雷码的特点是任意两个相邻的码组之间只有一位数不同。

1.2　逻辑代数

数字电路的结构是以二值数字逻辑为基础的，其中的工作信号是离散的数字信号 **0** 和 **1**。数字电路的工作过程就是在一定的工作条件下体现一种因果判断关系，即输出与输入之间的逻辑关系。研究数字电路逻辑关系的数学工具就是逻辑代数。逻辑代数中的变量叫做逻辑变量，逻辑变量常用字母 $A \sim Z$ 表示。逻辑变量的取值只有两种，即 **1** 或 **0**。**1** 或 **0** 不表示数值，只表示两个对立的逻辑状态。

1.2.1 三种基本的逻辑关系和运算

基本的逻辑关系只有三种,即与逻辑、或逻辑和非逻辑。与之对应,在逻辑代数中最基本的逻辑运算也有三种,与运算、或运算和非运算。下面将通过一些简单的实例来说明这三种逻辑关系和运算。

1. 与逻辑关系和运算

只有决定事物结果的全部条件同时具备时,结果才发生,这种因果关系叫做与逻辑,实现这一逻辑关系的运算就是与逻辑运算。

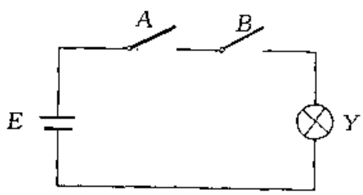

图 1.2.1

如图 1.2.1 所示电路中,只有当电路中的两个开关都合上,灯泡才会亮。此时灯泡的状态(结果)和开关的状态(条件)之间就是与逻辑关系。

用逻辑变量 A、B 代表两个开关的状态,两个变量只有两种取值,假设 **1** 代表接通,**0** 代表断开;用 Y 代表灯泡的状态,假设 **1** 代表亮,**0** 代表不亮。

在逻辑代数中,与逻辑运算可写成表达式

$$Y = A \cdot B \quad 或 \quad Y = AB$$

把 A 和 B 两个逻辑变量的全部可能取值及运算后的结果列成表,这种表格称为真值表。与运算真值表如表 1.2.1 所示。真值表是表示逻辑关系的重要方法之一。

与逻辑关系还可以用逻辑符号表示,如图 1.2.2 所示。在数字电路中,实现与逻辑功能的电路称为与门。

表 1.2.1 与运算真值表

A	B	$Y=AB$
0	**0**	**0**
0	**1**	**0**
1	**0**	**0**
1	**1**	**1**

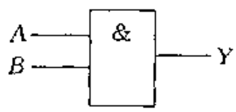

图 1.2.2

2. 或逻辑关系和运算

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,这种因果关系叫做或逻辑,实现这一逻辑关系的运算就是或逻辑运算。

如图 1.2.3 所示,只要电路中的任何一开关合上,灯泡就会亮。此时灯泡的状态(结果)和开关的状态(条件)之间就是或逻辑关系。

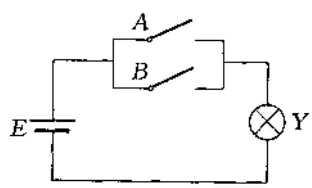

图 1.2.3

在逻辑代数中,或逻辑运算可写成表达式

$$Y = A + B$$

或运算真值表如表 1.2.2 所示。

或逻辑关系还可以用逻辑符号表示,如图 1.2.4 所示。在数字电路中,实现或逻辑功能的电路称为或门。

表 1.2.2　或运算真值表

A	B	$Y=A+B$
0	0	0
0	1	1
1	0	1
1	1	1

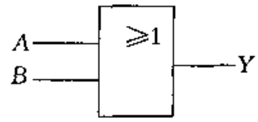

图 1.2.4

3. 非逻辑关系和运算

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，这种因果关系叫做非逻辑，实现这一逻辑关系的运算就是非逻辑运算。

如图 1.2.5 所示，只要电路中的开关断开，灯泡就会亮，开关闭合，灯泡反而不亮。此时灯泡的状态(结果)和开关的状态(条件)之间就是非逻辑关系。

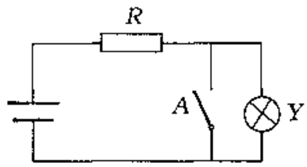

图 1.2.5

在逻辑代数中，非逻辑运算可写成表达式：$Y=\overline{A}$

非运算真值表如表 1.2.3 所示非逻辑关系还可以用逻辑符号表示，如图 1.2.6 所示。在数字电路中，实现非逻辑功能的电路称为非门。

表 1.2.3　非运算真值表

A	$Y=\overline{A}$
0	1
1	0

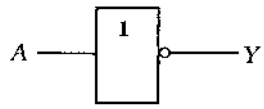

图 1.2.6

1.2.2　常用的逻辑关系和运算

实际应用中我们会碰到比与、或、非更复杂的逻辑问题，但它们都可以用与、或、非三种基本逻辑运算作特定组合来实现。常用的复合逻辑关系和运算有与非、或非、与或非、异或、同或等。下面分别说明。

1. 与非逻辑

与非逻辑是用与及非逻辑组合成的逻辑函数，其真值表和逻辑符号如表 1.2.4 和图 1.2.7 所示，表达式为

$$Y=\overline{AB}$$

表 1.2.4　与非运算真值表

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

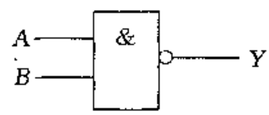

图 1.2.7

由真值表可知，只要输入变量有一个为 0，结果就为 1；两输入变量都为 1 时，结果才为 0。

此为试读，需要完整PDF请访问：www.ertongbook.com