刁节涛　田　曦　丁文霞

"九五"军队级重点教材

九五

数字电路与逻辑设计

SHUZI DIANLU YU LUOJI SHEJI

国防科技大学出版社

“九五”军队级重点教材

数字电路与逻辑设计

刁节涛　田　曦　丁文霞　编著

国防科技大学出版社
·长沙·

图书在版编目(CIP)数据

数字电路与逻辑设计/刁节涛,田曦,丁文霞编著．长沙:国防科技大学出版社,2000.7
ISBN 7-81024-638-0

Ⅰ.数…　Ⅱ.①刁…②田…③丁…　Ⅲ.数字电路-理论　Ⅳ.TN79

中国版本图书馆 CIP 数据核字(2000)第 25541 号

国防科技大学出版社出版发行
电话:(0731)4555681　邮政编码:410073
E-mail:gfkdcbs@public.cs.hn.cn
责任编辑:黄八一　责任校对:潘　生
新华书店总店北京发行所经销
国防科技大学印刷厂印装

*

787×1092　1/16　印张:22.25　字数:514 千
2000 年 6 月第 1 版 2006 年 1 月第 2 次印刷　印数:4001—7000 册

*

定价:32.00 元

前　言

数字电路课程是电子工程、自动化技术、计算机等电类专业和机电一体化等非电类专业的主要技术基础课程。教育部曾多次组织重点院校的专家教授们编写出版过多部统编教材,对该课程的发展起到了重要的推动作用。

随着电子科学技术的高速发展,近年来数字电路课程的教学内容有了很大的变化。由于大规模和超大规模集成电路(LSI/VLSI)的出现,使数字电路课程内容的讲授由单元电路,逻辑部件过渡到数字系统。对电路和系统的分析与设计方法也由传统的方法过渡到计算机辅助分析与设计。面对科学技术飞速发展的现实与现代化教学的需要,我们编写了"数字电路与逻辑设计"一书。我们希望该书能成为面向21世纪的换代教材,被广大读者所接受。

《数字电路与逻辑设计》一书阐述了数字逻辑电路的基本理论,包括逻辑代数、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、半导体存储器、D/A和A/D转换电路以及可编程逻辑器件的基本结构、开发应用及数字电路CAD技术的理论分析,实际应用和工程设计的基本方法。该书的选材符合当前数字电路的发展现状和工程应用的需求,范围较广,内容较全,深度适中。该书既保留了本课程的经典内容,又扩充了当前数字逻辑电路的新发展、新技术与新成果,内容取舍恰当,力图体现本教材的系统性、先进性以及理论结合实际的原则。全书吸收了国内外许多优秀教材、专著和学术论文的先进思想和科研成果,是一本较好地符合当前教学需要的教材。

该书既可用于高年级本科生、继续工程教育和研究生的教材,也可作为电子工程领域科技人员的参考书。

本书由刁节涛、田曦、丁文霞编写,陈善思教授主审。第一、二、三、四章由刁节涛编写,五、六、七章由丁文霞编写,八、九、十章由田曦编写,全书由刁节涛统稿、定稿。

本书在编写过程中得到了国防科技大学教保处,国防科技大学电子科学与工程学院、国防科技大学CAD实验中心的大力支持和帮助,在此一并表示衷心的感谢。

由于编者水平有限,书中难免会有一些错误及不足之处,恳请读者批评指正。

编　者

2000.4.

目　　录

第一章　逻辑代数基础

逻辑代数是分析和设计逻辑电路的数学工具。本章介绍数制、逻辑变量、逻辑函数、逻辑运算以及逻辑化简等。

第一节　数制与码制

§1.1.1　数制

用数字量表示物理量的大小时，仅用一位数码是不够的，因此往往用进位计数的方法组成数码使用，我们把多位数码中每一位的构成方法以及从低位到高位的进位规则称为数制。

在数字电路中经常使用的计数进制除了十进制以外，还经常使用二进制、八进制和十六进制。

一、十进制

十进制是人们熟悉而常用的数制，组成十进制的符号有 0、1、2、3、4、5、6、7、8、9，我们称这些符号为数码。

一种数制的基本特征是它的基数，十进制的计数规则是“逢十进一”，所以它的基数为10。

例如，十进制数 278.94 用位置计数法可以表示为：

$$N_{10} = 2 \times 10^2 + 7 \times 10^1 + 8 \times 10^0 + 9 \times 10^{-1} + 4 \times 10^{-2}$$

任意一个十进制数均可展开为：

$$N_{10} = \sum_{i=-m}^{n-1} a_i 10^i \qquad (1.1.1)$$

其中：设整数部分的位数是 n，小数部分位数是 m，它的简写形式为：

$$N_{10} = a_{n-1} a_{n-2} \cdots a_1 a_0 a_{-1} \cdots a_{-m}$$

二、二进制

目前在数字电路及数字系统中应用最广的是二进制，二进制有两个数码 0、1，基数是

2。二进制的计算规则是“逢二进一”。

根据式(1.1.1),任何一个二进制数均可展开为:

$$N_2 = \sum_{i=-m}^{n-1} b_i 2^i \tag{1.1.2}$$

其中:设整数部分的位数 n,小数部分的位数是 m,它的简写形式为:

$$N_2 = b_{n-1} b_{n-2} \cdots b_1 b_0 \cdot b_{-1} b_{-2} \cdots b_{-m}$$

如 11011.11,1010.101 等

例如 二进制 101.11 用位置计数法可表示为:

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

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 和 D 代替 2 和 10 这两个脚注。

三、八进制

八进制有 0~7,8 个数码,基数为 8,它的计算规则是“逢八进一”。

例如:八进制数 372.01 用位置计数法可表示为:

$$(372.01)_8 = 3 \times 8^2 + 7 \times 8^1 + 2 \times 8^0 + 0 \times 8^{-1} + 1 \times 8^{-2}$$

八进制数的一般表达式为:

$$N_8 = \sum_{i=-m}^{n-1} a_i 8^i \tag{1.1.3}$$

四、十六进制

组成十六进制数的符号有 0、1、2、3、4、5、6、7、8、9、$A(10)$、$B(11)$、$C(12)$、$D(13)$、$E(14)$、$F(15)$,其中字母 $A \sim F$ 表示 10 ~ 15,十六进制的计数规则是“逢十六进一”。

例如:十六进制数 $E5D7.A3$ 用位置计数法可表示为:

$$(E5D7.A3)_{16} = 14 \times 16^3 + 5 \times 16^2 + 13 \times 16^1 + 7 \times 16^0 + 10 \times 16^{-1} + 3 \times 16^{-2}$$

十六进制的一般表达式为:

$$N = \sum_{i=-m}^{n-1} a_i 16^i \tag{1.1.4}$$

§1.1.2 数制间的转换

一、各种进制转换成十进制

将二进制、八进制、十六进制转换为十进制的方法很简单,只要按照位置计数法表达式,求出系数与位权之积,然后把诸项乘积求和,即可得到转换结果。

例如 $(1010.011)_2 = 2^3 + 2^1 + 2^{-2} + 2^{-3} = (10.375)_{10}$

其它进制转换为十进制的方法是类似的,不再赘述。

二、十进制转换为其它进制

1.十进制数转换为二进制数

十进制数可分为整数和小数两部分。对整数和小数分别转换,再将结果排列在一起就得到完整的转换结果。

(1) 整数部分转换

先举一个例子,把十进制数 41 转换为二进制数,其除法算式如下:

$$
\begin{array}{lll}
2\,|\,\underline{41} & \cdots\cdots \text{余 } 1 \cdots \text{最低位 } b_0 \\
\quad 2\,|\,\underline{20} & \cdots\cdots \text{余 } 0 \cdots\cdots b_1 \\
\quad\quad 2\,|\,\underline{10} & \cdots\cdots \text{余 } 0 \cdots\cdots b_2 \\
\quad\quad\quad 2\,|\,\underline{5} & \cdots\cdots \text{余 } 1 \cdots\cdots b_3 \\
\quad\quad\quad\quad 2\,|\,\underline{2} & \cdots\cdots \text{余 } 0 \cdots\cdots b_4 \\
\quad\quad\quad\quad\quad 2\,|\,\underline{1} & \cdots\cdots \text{余 } 1 \cdots \text{最高位 } b_5 \\
\quad\quad\quad\quad\quad\quad\quad 0 &
\end{array}
$$

所以,$(41)_{10} = (101001)_2$

第一次除法的余数是二进制最低位的系数,最后一次相除的余数是二进制最高位的系数,将余数按顺序排列起来就是所得的二进制数。

(2) 小数部分转换

把十进制转换为二进制数采用基数乘法,现举例说明转换过程,将十进制小数$(0.39)_{10}$转换为二进制数:

$$
\begin{array}{ll}
0.39 \times 2 = 0.78 & b_{-1} = 0 \\
0.78 \times 2 = 1.56 & b_{-2} = 1 \\
0.56 \times 2 = 1.12 & b_{-3} = 1 \\
0.12 \times 2 = 0.24 & b_{-4} = 0 \\
0.24 \times 2 = 0.48 & b_{-5} = 0 \\
0.48 \times 2 = 0.96 & b_{-6} = 0 \\
0.96 \times 2 = 1.92 & b_{-7} = 1 \\
0.92 \times 2 = 1.84 & b_{-8} = 1 \\
0.84 \times 2 = 1.68 & b_{-9} = 1 \\
0.68 \times 2 = 1.36 & b_{-10} = 1
\end{array}
$$

换算到此为止,$(0.39)_{10} = (0.0110001111)_2 + e$,$e$ 为剩余误差,$e < 2^{-10}$。对十进制小数进行转换时,可以进行到所得乘积小数部分为 0 或者达到所需精度为止。

2.十进制数转换为任意进制数

把十进制数转换为 r 进制的方法类同于十进制数转换为二进制数,即整数部分采用

基数除法，小数部分采用基数乘法，不同点在于基数不是 2 而是 r。

例如：将$(153)_{10}$转换为八进制数，其除法算式如下：

```
8 | 153     ……余 1…a0 = 1
   8 | 19      ……余 3……a1 = 3
      8 | 2      ……余 2……a2 = 2
           0
```

所以 有$(153)_{10}=(231)_8$

将$(0.513)_{10}$转换为八进制数，其乘法算式如下：

$$
\begin{array}{ll}
0.513\times 8 = 4.104 & a_{-1}=4\\
0.104\times 8 = 0.832 & a_{-2}=0\\
0.832\times 8 = 6.656 & a_{-3}=6\\
0.656\times 8 = 5.248 & a_{-4}=5\\
0.248\times 8 = 1.984 & a_{-5}=1\\
0.984\times 8 = 7.872 & a_{-6}=7
\end{array}
$$

如转换到此已满足精度要求，将得到 $(0.513)_{10}=(0.406517)_8+e$，剩余误差 $e<8^{-6}$

如果被转换的十进制数既有整数部分又有小数部分，先将两部分分别转换，再把转换结果并列在一起。

例如：$(153.513)_{10}=(231.406517)_8+e$。

三、二进制与八进制之间的转换

八进制的基数 8 是 2 的幂$(8=2^3)$，一位八进制数对应三位二进制数，因此二进制和八进制的互换是非常容易的。

1.二进制转换成八进制

把二进制从小数点开始分别向右和向左划分成三位一组，每组便是一位八进制。

例如：$(\underline{110}\ \underline{011}\ \underline{100}.\underline{101})_2=(634.5)_8$

如果不能正常构成三位一组，则在二进制数整数部分高位添零或在小数部分低位添零来补足三位一组。

例如：$(10011101.01)_2=(\underline{010}\ \underline{011}\ \underline{101}.\underline{010})_2=(235.2)_8$

2.八进制转换成二进制

同样的道理，由八进制转换成二进制的方法与上述过程相反。

例如：$(753.4)_8=(\underline{111}\ \underline{101}\ \underline{011}.\underline{100})_2=(111101011.1)_2$

四、二进制与十六进制之间的转换

十六进制的基数 $16=2^4$，因此，一位十六进制对应四位二进制数，它们之间的互换也是非常容易的。

1.二进制转换成十六进制

把二进制从小数点开始分别向右或向左划分四位一组每组便是一位十六进制，如果不足四位时，在整数部分高位添零或在小数部分低位添零来补足四位。

例如：$(1011101000.011)_2 = (\underline{0010}\ \underline{1110}\ \underline{1000}.\underline{0110})_2 = (2E8.6)_{16}$

2.十六进制转换成二进制

十六进制转换成二进制的方法与上述过程正好相反

例如：$(3FD.B)_{16} = (0011\ 1111\ 1101.1011)_2 = (1111111101.1011)_2$

在计算机中八进制下标用英文字母"O"表示，十六进制下标用"H"表示。

第二节　逻辑代数基本概念和运算规则

逻辑代数是英国数学家乔治·布尔(Geroge Boole)于1847年提出的，所以又称布尔代数，它是分析和设计逻辑电路的数学工具。

§1.2.1　逻辑变量与逻辑函数

在逻辑代数中的变量称为逻辑变量，用字母A、B、C、表示。逻辑变量只能有两种可能的取值：真和假。常把真记作"1"，假记作"0"。这里"1"和"0"并不表示数量的大小，而是表示完全对立的两种状态。"1"表示条件具备或事情发生；"0"表示条件不具备或事情不发生。反之，亦然。

例如：在图1.2.1所示的电路中，指示灯是否点亮取决于开关A是否接通。如果我们定义：

$F = 1$表示灯亮　　　　$F = 0$表示灯灭；

$A = 1$表示开关接通　　　$A = 0$表示开关断开。

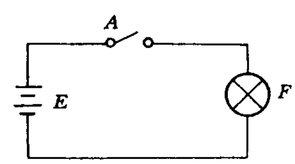

图1.2.1　指示灯开关电路

那么，F是A的函数，逻辑函数表达式为$F = f(A)$。F和A都称为逻辑变量，其中A称为输入逻辑变量(简称逻辑变量)，F称为输出逻辑变量(简称逻辑函数)。如果逻辑函数是多变量函数，即$F = f(A,B,C,\cdots)$，那么，逻辑函数的表达式就比较复杂，逻辑函数表达式由逻辑变量$A,B,C,\cdots$和算子"·"(与)、"+"(或)、"-"(非)及括号、等号等组成。

如：$F = A$；　$Z = A \cdot B$；　$G = \bar{A}$；　$H = A \cdot (B + \bar{C})$

在上述逻辑函数表达式中，A、B、C为逻辑变量，F、Z、G、H为逻辑函数，逻辑变量上加一横杠的为逻辑反变量，不加的为逻辑原变量。

§1.2.2　逻辑运算

图1.2.2中给出了三个指示灯的控制电路。在图(a)电路中，只有当两个开关同时闭

合时，指示灯才会亮；在图(b)电路中，只要有任何一个开关闭合，指示灯就亮；而在图(c)电路中开关断开时灯亮，开关闭合时灯反而不亮。

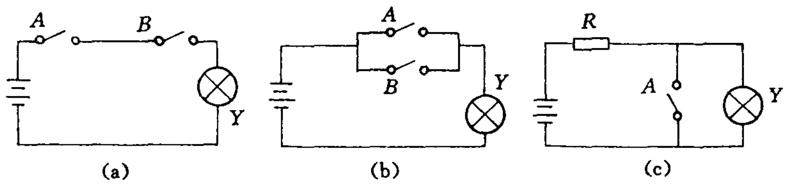

图 1.2.2　用于说明与、或、非定义的电路

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 1.2.2 中的三个电路代表了三种不同的因果关系：

图(a)的例子表明，只有决定事物结果全部条件同时具备时，结果才发生。这种因果关系叫做逻辑与或者叫做逻辑相乘。

图(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。这种因果关系叫做逻辑或，也叫做逻辑相加。

图(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。这种因果关系叫做逻辑非，也叫做逻辑求反。

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 表示灯不亮，则可以列出以 0、1 表示的与、或、非逻辑关系的图表，如表 1-2-1，1-2-2 和 1-2-3 所示。这种图表叫做逻辑真值表，或简称为真值表。

表 1-2-1　与逻辑运算的真值表

A　B	Y
0　0	0
0　1	0
1　0	0
1　1	1

表 1-2-2　或逻辑运算的真值表

A　B	Y
0　0	0
0　1	1
1　0	1
1　1	1

表 1-2-3　非逻辑运算的真值表

A	Y
0	1
1	0

在逻辑代数中，把与、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“·”表示与运算，以“+”表示或运算，以变量上边的“−”表示非运算。因此，A 和 B 进行与逻辑运算时可写成：

$$Y = A \cdot B$$

A 和 B 进行或逻辑运算时可以写成：

$$Y = A + B$$

对 A 进行非逻辑运算时可写成：

$$Y = \bar{A}$$

同时，把实现与逻辑运算的单元电路叫做与门，把实现或逻辑运算的单元电路叫做或

门，把实现非逻辑运算的单元电路叫做非门(也叫做反相器)。

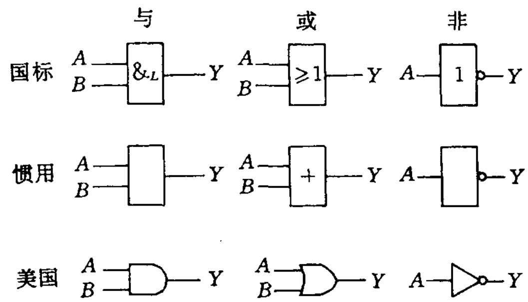

图 1.2.3　与、或、非的图形符号

与、或、非逻辑运算还可以用图 1.2.3 所示的图形符号来表示。

实际的逻辑问题往往比与、或、非复杂得多，不过它们都可以用与、或、非的组合来实现。最常见的复合逻辑运算有与非、与或、异或、同或等。表 1-2-4 ~ 1-2-8 给出了这些复合逻辑运算的真值表。图 1.2.4 是它们的图形逻辑符号和运算符号。

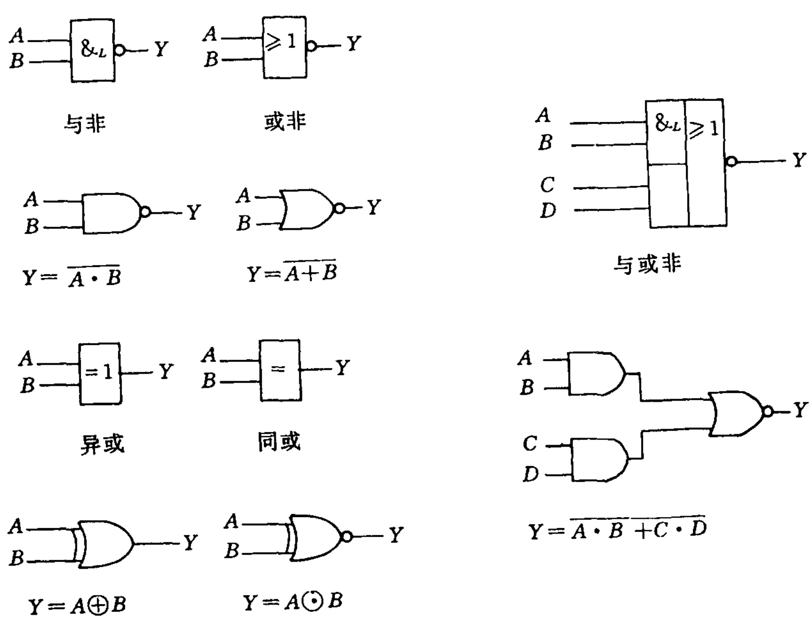

图 1.2.4　复合逻辑的图形符号和运算符号

表 1-2-4 与非逻辑的真值表

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

表 1-2-5 或非逻辑的真值表

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

表 1-2-6 与或非逻辑的真值表

A B	C D	Y
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

表 1-2-7 异或逻辑的真值表

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

表 1-2-8 同或逻辑的真值表

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

由表 1-2-4 可见,将 A、B 先进行与运算,然后将结果求反,最后得到的即 A、B 的与非运算结果。因此,可以把与非运算看作是与运算和非运算的组合。图 1.2.4 中图形符号上的小园圈表示与非运算。

在与或非逻辑中,A、B 之间以及 C、D 都是与的关系,只要 A、B 或 C、D 任何一组或两组同时为 1,输出 Y 就是 0。只有当每一组输入都不全是 1 时,输出 Y 才是 1。

异或是这样一种逻辑关系:当 A、B 不同时,输出 Y 为 1;而 A、B 相同时,输出 Y 为 0。异或也可以用与、或、非的组合表示。

$$A \oplus B = A \cdot \bar{B} + \bar{A} \cdot B \tag{1.2.4}$$

同或和异或相反,当 A、B 相同时,Y 等于 1,A、B 不同时,Y 等于 0。同或也可以写成与、或、非的组合形式。