脉冲与数字电路

王馥熏
骆克敏 编

武汉测绘科技大学出版社

前　言

本教材主要根据计算机专业“脉冲与数字电路”课程教材大纲，并结合近几年计算机学科发展而编写的。

全书十一章分为五大部分。第一部分（第一章概论，第八章脉冲波形的产生与整形，第十章传输线）介绍各种数制和相应的代码，以及代码信号（数字信号）的产生、传输。第二部分（第二章晶体二极管及三极管的有关特性，第三章逻辑门电路，第四章逻辑代数基本原理，第六章触发器）介绍逻辑运算的数学描述方法——逻辑代数（布尔代数）和实现逻辑运算所用的基本门电路和触发器。第三部分（第五章组合逻辑电路，第七章时序逻辑电路）介绍用基本门电路和触发器构成的各种逻辑电路的分析（和基本设计）方法。第四部分（第九章大规模集成电路基础）介绍大规模集成电路的基本器件和电路形式，并引入微处理器芯片。第五部分（第十一章 A/D——D/A 转换）介绍数字信号和模拟信号的互相转换。

本书是计算机专业“脉冲与数字电路”课程的教材，也可作为其它专业相应课程的有关教学参考书或教材，因此教材编写时按模块结构组织各章内容。各专业都可按其需要取用相应章节内容作为参考或教学。各章都有相应的例题和习题。

教材以中小规模集成电路为主，由浅入深，讲清基本概念，介绍各类电路的使用特性，分析具体的实用电路，并引导学生掌握基本的分析和设计方法。

教材的第一、二、三、六、八、十一等章由骆克敏同志编写，第四、五、七、九、十章由王馥熏同志编写，最后王馥熏同志执笔统一全稿。

本教材承蒙华中理工大学毛法尧教授和武汉测绘科技大学钟钦华教授审阅并提出了许多宝贵的意见。在此致以衷心的感谢。

由于时间仓促和水平有限，本教材会有不妥以至错误之处，殷切期望有关专家和读者提出批评和建议，为提高该教材的质量而努力。

编　者

1994 年 2 月

目　录

第一章　概　　论

1—1　脉冲电路与数字逻辑电路

脉冲与数字电路课程包含了两部分内容：脉冲电路和数字逻辑电路。

一、脉冲电路

脉冲电路主要讲述脉冲讯号的产生、变换、整形、传输和应用。产生脉冲讯电路所用的器件最早是用电子管，后来用晶体管，现在又采用了集成电路器件。

所谓脉冲信号就是"在短暂时间里发生跃变的电压或电流波形"。至于"短暂时间"可以和电路的过渡过程时间相比拟，其数量级在微秒或毫微秒。(即 10^{-6}秒或 10^{-9}秒)。一般认为，除了正弦波以外的一切波形都可以称为脉冲波形，常见的有以下几种：(图 1—1)。

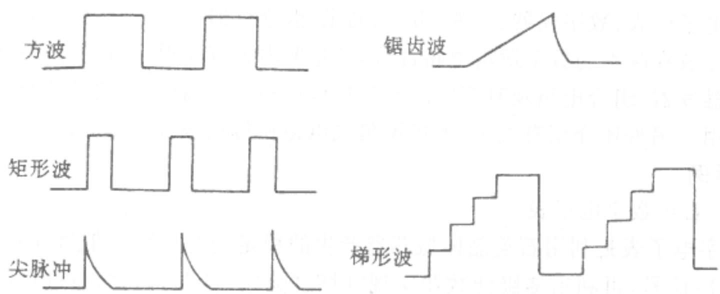


图 1—1　各种脉冲波形

上面绘出的都是理想的波形。实际波形(例如方波)如图 1—2 所示，其形状可用一些参数来表示：

T：脉冲周期。脉冲频率为 $f=\frac{1}{T}$

V_m：脉冲最大幅度

t_k：脉冲宽度，在 $0.5V_m$ 处测定

t_r：脉冲上升时间，(又称前沿)，从 $0.1V_m$ 到 $0.9V_m$ 之间测定

t_f：脉冲下降时间，(又称后沿)，从 $0.9V_m$ 到 $0.1V_m$ 之间测定

k：空度系数，用 $k=\frac{T}{t_k}$来表示。

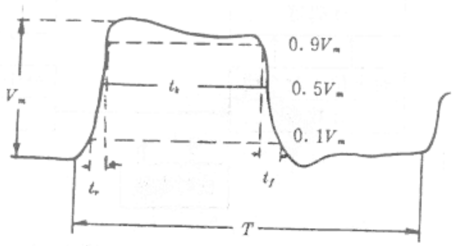


图 1—2　脉冲参数

上述参数中 T、t_k、t_r、t_f 的计量单位用秒、毫秒($1ms=10^{-3}s$)、微秒($1\mu s=10^{-6}s$)、毫微秒($1ns=10^{-9}s$)、V_m 用伏。

此为试读，需要完整PDF请访问：www.ertongbook.com

二、数字逻辑电路

数字逻辑电路是产生数字信号以及执行数字和逻辑运算的电子电路。现在都采用集成电路组成，故称"数字集成逻辑电路"，数字讯号是一种在时间上和数值上均不连续的，离散的讯号。例如电报讯号、雷达讯号等，它与大家所熟悉的模拟信号不同，模拟信号在幅度和时间上都是连续的，如正弦波等。因此，我们可以把脉冲信号也看作数字信号的一种。例如矩形波在数字电路中就得到大量的应用。

数字是人们非常熟悉的，常用的符号是 0，1，2，3，4，5，6，7，8，9 十种，运算时"逢十进一"，故称十进制计数。钟表的分、秒是逢六十进位，称六十进制，其它还有八进制，十六进制等等。但是由于在电路中人们无法用十种状态来进行数的表示，所以逻辑代数中采用了二进制的方法。二进制符号只有两个。即：0 和 1，运算时"逢二进一"，这样两个符号就可以用电讯号来实现。例如有电流表示"1"，无电流表示"0"；有电压表示"1"，无电压表示"0"。

这种二值信号还能描述客观事物中大量的相互对立的事件。例如事情的真与假、大与小、好与坏……。计算机对这种二值信号既可以按其数值大小去处理(进行数学上的计算)，也可以对相互对立的事件进行逻辑思维的推理(进行逻辑运算)。

随着科学技术的发展，数字逻辑电路已由小规模集成电路进入了中、大规模集成电路，目前又制造了超大规模集成电路，为数字逻辑电路的应用和发展开辟了广阔的前景，应用范围越来越广泛。如电子仪表、数字仪器、电视、雷达、通讯、医疗仪器、电子计算机等等。

本课程重点放在基本的数字逻辑电路部分，其主要内容有逻辑门电路，逻辑代数(又称"布尔代数")各种触发器、组合电路及时序电路的分析设计和应用。脉冲电路部分介绍脉冲波形的产生、变换、应用。另外还介绍有关中、大规模集成电路和传输线等工作原理。

三、应用举例

【例 1】 石英数字电子表

石英式数字电子表是利用石英晶体振荡器产生的稳定的时标脉冲信号，经过分频器得到每秒 1Hz 的脉冲信号，再利用多级计数器实现"时"、"分"、"秒"的计数，通过译码电路接到显示屏加以显示。"时""分""秒"的计数分别用 24 进制计数器和 60 进制计数器，显示屏可以用数码管或液晶体显示，具体框图如图 1—3 所示

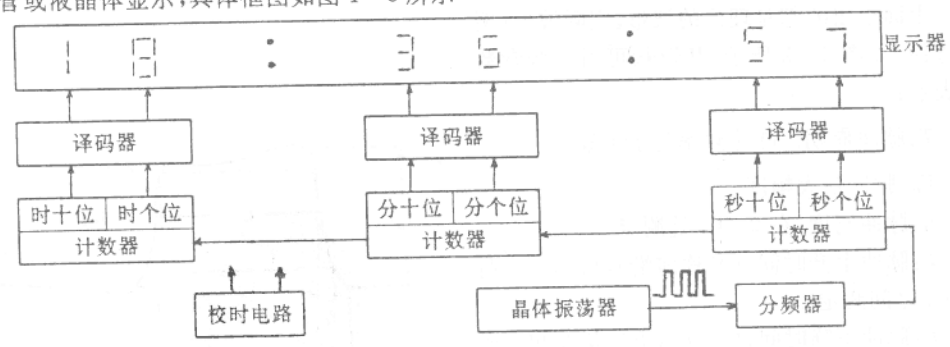


图 1—3 石英数字式电子表逻辑图

【例 2】 数字式电机测速装置

图 1—4 是一种测定电机转速的数字装置框图。在电机的马达主轴上外接一个带小孔的圆盘，光源通过小孔投射到光电信号转换器上，电机马达转动一周，光电信号转换器接收到一次光照，并产生一个电信号，经放大、整形后得到一个电脉冲信号，接到框图中的门电路输入端。

门电路的另一端接入由脉冲发生器发来的脉中宽度为1秒的门控信号。因此，门电路被定时的打开1秒钟，“放行”由马达转速产生的电脉冲，并进入计数和显示装置。显示屏显示的数值即为电机马达的每秒钟转速。

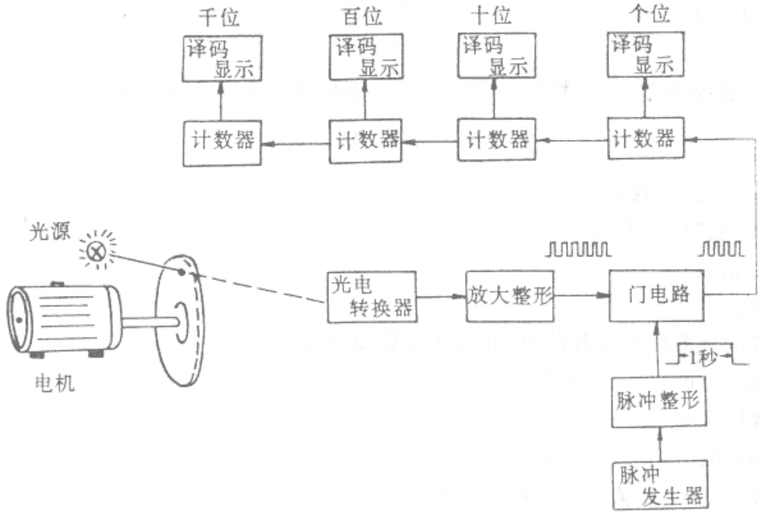


图1—4　数字式电机测速逻辑图

1—2　数　　制

十进制数制是我们日常使用最多、最习惯的一种，而在数字逻辑电路中却以二进制为基础进行处理，这样可以更好地利用电讯号来表达。下面讨论这两种数制互相转换的方法，此外，还介绍八进制、十六进制以及二～十进制(BCD)的数码表示法。

一、十进制数

十进制数由0,1,2,3,4,5,6,7,8,9十个数字符号来表示，把不同的数符组合起来构成一个数时，数符所处的位置不同代表的意义也不同，例如一个十进制数1987.12，小数点左边第一位为个位，第二位为十位，第三位为百位，第四位为千位，而小数点右边第一位为$\frac{1}{10}$位，第二位为$\frac{1}{100}$位，以此类推。写成一个完整的表达式是：$1987.12=1\times1000+9\times100+8\times10+7\times1+1\times\frac{1}{10}+2\times\frac{1}{100}=1\times10^3+9\times10^2+8\times10^1+7\times10^0+1\times10^{-1}+2\times10^{-2}$，我们把$10^{-2}$、$10^{-1}$、$10^0$、$10^1$、$10^2$、$10^3$等称为不同数位的“权”。

推而广之，一个任意的十进制数$(N)_{10}$可以写成一个多项式：

$$
\begin{aligned}
(N)_{10} &= a_{n-1}10^{n-1} + a_{n-2}\cdot10^{n-2} + \cdots + a_2\cdot10^2 + a_1\cdot10^1 \\
&\quad + a_0\cdot10^0 + a_{-1}\cdot10^{-1} + a_{-2}\cdot10^{-2} + \cdots + a_{-m}\cdot10^{-m} \\
&= \sum_{j=-m}^{n-1} a_j\cdot10^j
\end{aligned}
$$

式中数N括号外的右下标为数制，10即为十进制，2为二进制，其余类推。

a_i 可取 0～9 的任意数符，m、n 为正整数（n 为整数位，m 为小数位）：在进位计数制中，独立的数符个数称为该数制的基数。因此，10 为十进制的基数。

二、二进制数

二进制只有两个数符，即 0，1。以 2 为基数，"逢二进一"。二进制的表达式如下：

$$11011.101 = 1\times 2^4 + 1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 1\times 2^0 + 1\times 2^{-1} + 0\times 2^{-2} + 1\times 2^{-3}$$

此处 2^n 作为二进制各数位的"权"。同理，一个二进制多项式也可以写成：

$$(N)_2 = \sum_{i=-m}^{n-1} b_i \cdot 2^i$$ 式中：

$(N)_2$：表示二进制数；

b_i：只取"0"或"1"二个数符；

m、n：取正整数；

2：为基数。

二进制数的运算规则比较简单，其加法运算规则是：

0＋0＝0；0＋1＝1；1＋0＝1；1＋1＝10

乘法运算规则是：

0×0＝0；0×1＝1；1×0＝1；1×1＝1

十进制数的 0～9 转换成二进制数的表达方式如下：

$(0)_{10}=(0)_2$　　$(1)_{10}=(1)_2$

$(2)_{10}=(10)_2$　　$(3)_{10}=(11)_2$

$(4)_{10}=(100)_2$　　$(5)_{10}=(101)_2$

$(6)_{10}=(110)_2$　　$(7)_{10}=(111)_2$

$(8)_{10}=(1000)_2$　　$(9)_{10}=(1001)_2$

八进制数和十六进制数的表示方法见表 1—1 所示：

表 1—1　几种数制中数的表示法

十进制	二进制	八进制	十六进制	十进制	二进制	八进制	十六进制
0	0	0	0	9	1001	11	9
1	1	1	1	10	1010	12	A
2	10	2	2	11	1011	13	B
3	11	3	3	12	1100	14	C
4	100	4	4	13	1101	15	D
5	101	5	5	14	1110	16	E
6	110	6	6	15	1111	17	F
7	111	7	7	16	10000	20	10
8	1000	10	8	17	10001	21	11

三、数制的相互转换

1. 二进制数转换成十进制数

一个二进制数转换成十进制数时，只要将二进制数的每一位的数符乘以该位的权，然后按十进制计数规则进行算术运算，其结果就是相应的十进制数。为了计算方便，下面给出了二进制的权 2^n 和 2^{-n} 值。

表 1—2 二进制的权 2^n 和 2^{-n}

n	2^n	2^{-n}
0	1	1.0
1	2	0.5
2	4	0.25
3	8	0.125
4	16	0.0625
5	32	0.03125
6	64	0.015625
7	128	0.0078125
8	256	0.00390625
⋮	⋮	⋮

【例 1】 将$(11111000011.001)_2$化为十进制数

解：
$$\begin{aligned}(11111000011.001)_2 &= 1\times2^{10}+1\times2^9+1\times2^8+1\times2^7+1\times2^6+0\times2^5+0\times2^4+0\times2^3\\ &\quad+0\times2^2+1\times2^1+1\times2^0+0\times2^{-1}+0\times2^{-2}+1\times2^{-3}\\ &=1024+512+256+128+64+2+1+0.125=(1987.125)_{10}\end{aligned}$$

从式中可以看到，同样一个数用二进制表示比用十进制表示的位数要多。

2. 将十进制数转换成二进制数（基数乘、除法）

十进制整数转换成二进制整数要用逐项除 2 的方法，其步骤如下：

1)将十进制数除以 2，记下余数 0 或 1 作为转换后二进制数的最低位的值；

2)把上面所得的商数再除以 2，余数 0 或 1 作为次低位；

3)按上办法继续将商除 2，记下余数，直到最后商数为 0 止，此时的余数为转换后二进制数的最高位；

4)将余数从高位到低位写成二进制数表达式。

【例 2】将$(87)_{10}$换成二进制数。

			余数	
2	87			
2	43	……	1	（最低位）
2	21	……	1	
2	10	……	1	
2	5	……	0	
2	2	……	1	
2	1	……	0	
	0	……	1	（最高位）

所以$(87)_{10}=(1010111)_2$

十进制数转换成二进制数时也可以用查表(1—2)方法来进行。

【例 3】$(19)_{10}$换成二进制数

①查表中不超过 19 的 2^n 最大值是 $2^4=16=10000$

②$19-16=3$

③查表中不超过 3 的 2^n 的最大值是 $2^1=2=00010$

④$3-2=1$

⑤查表：　　　　　　　　　　　　　　$1=00001$

⑥总和得结果：　$(19)_{10}$　　　　　　$=(10011)_2$

对于十进制的小数转换为二进制小数的方法通常用乘2的方法，步骤如下：

1)将纯小数乘以2，其乘积≥1时，记录1；乘积<1时，记录为0，此即二进制小数的最高位数；

2)将上一步中所得的数丢掉整数部分，又将其小数部分乘2，同上一样，乘积≥1时记1；乘积<1时记0；

3)不断计算下去，直至乘2的结果到0为止，或已达到所求数的精度，运算结束，同时得到小数的最低位。

【例4】将$(0.6875)_{10}$转换成二进制小数。

解：

	乘积	二进制数
0.6875 × 2 = 1.3750	>1	1(高位)
0.3750 × 2 = 0.7500	<1	0
0.7500 × 2 = 1.5000	>1	1
0.5000 × 2 = 1.0000	=1	1(低位)

结果：$(0.6875)_{10}=(0.1011)_2$

3. 八进制和二进制转换(直接法)

八进制数码为0,1,2,3,4,5,6,7,“逢八进一”其基数为8，即$2^3=8$，而二进制的基数为2，即$2^1=2$，二者间的幂次比值为3∶1，所以一位八进制数相当于三位二进制的数。

【例5】$(123.45)_8$转换成二进制数

解：

$$\begin{array}{cccccc} 1 & 2 & 3 & . & 4 & 5 \\ \downarrow & \downarrow & \downarrow & & \downarrow & \downarrow \\ 001 & 010 & 011 & . & 100 & 101 \end{array}$$

所以　$(123.45)_8=(001010011.100101)_2$

同样，可以将二进制数分成三位一组写成八进制数。八进制比二进制位数缩短了三分之二，而且更接近于人们所习惯的读数方法。

4. 十六进制和二进制转换(直接法)

十六进制的符号是0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F。“逢十六进一”，即计数从0到F时，若再加1则记作10(参见表1—1)，其基数为16。

同理，因为$2^4=16$，所以若从二进制转换成十六进制时，只要从低位向高位每四位一组，最高位不足四位可在位首添加0，每组写成一个十六进制的数码。

【例6】将$(001100110000111)_2$转换成十六进制

解：分组写法如下：

$$\begin{array}{cccc} 0001 & 1001 & 1000 & 0111 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 9 & 8 & 7 \end{array}$$

结果是：$(0011001100000111)_2=(1987)_{16}$

【例 7】 将$(ADF)_{16}$转换成二进制

解：由表(1—1)查得：

$$\begin{array}{ccc} A & D & F \\ \downarrow & \downarrow & \downarrow \\ 1010 & 1101 & 1111 \end{array}$$

所以 $(ADF)_{16}=(101011011111)_2$

至于八进制、十六进制与十进制之间的转换，读者可参考其他书籍，此处不再叙述。

总之，在数制转换过程中，若 α 代表转换前的数制，β 代表转换后的数制，则在 $\alpha\to\beta$ 时，可作如下的判断和处理：

1) β 为十进数制：则 α 数制中的数按位乘以各数位的"权"值展开，并在十进制数中求其值的大小；

2) α 为十进制数：则将 α 数制中的整数部分用 β 数制的基数相除，其小数部分用 β 数制的基数相乘，(基数乘、除法)

3) α 和 β 数制的基数存在幂次比时，采用直接转换方法；

4) α 和 β 数制都为任意进制时，采用过渡方法，即用十进制数作为过渡数制。

1—3 数的编码

十进制数普及于人类的日常生活之中，而在数字系统中一般以二进制数为基础进行识别、记忆和运算，在具体的数字逻辑电路中二、十进制数都有其相应的代码形式。

表 1—3 数和数码的表示形式

十进制数	十进制数的代码	二进制数	二进制数码
0	0000	0	00000
1	0001	1	00001
2	0010	10	00010
3	0011	11	00011
4	0100	100	00100
5	0101	101	00101
6	0110	110	00110
7	0111	111	00111
8	1000	1000	01000
9	1001	1001	01001
10	0001 0000	1010	01010
11	0001 0001	1011	01011
12	0001 0010	1100	01100
13	0001 0011	1101	01101
14	0001 0100	1110	01110
15	0001 0101	1111	01111
16	0001 0110	10000	10000

此为试读，需要完整PDF请访问：www.ertongbook.com

一、十进制数和 BCD 码

用二进制数表示十进制数时，数值越大，数位越多，见表 1－3。用二进制代码表示十进制数时则有所不同，要包含十个不等数值或十个不同状态的可能性，至少需要四位二进制代码。于是$(1)_{10}$的数字代码为 0001，数值$(9)_{10}$的代码形式为 1001。代码具有组成特性，这种组成特性表示代码所采用的位数必须代表和考虑整个数值范围。因此数值$(0)_{10}$也必须用 0000 四位代码表示。数值$(87)_{10}$的表示方法为 1000，0111，前四位 1000 对应十位数 8，后四位 0111 对应个位数 7。

四位二进制代码共有十六个数值或十六种组合状态，用它表示不等的十个数值，就有六个多余状态称冗余状态。广义地说可以用十六种组合状态的任一种组合代表十进制数的任意一个数值，也可以取任意十种组合状态表示十个不同的数值，这种方法称为二—十进制编码即 BCD 码，表 1—4 中列有多种常用的 BCD 码，包括有权 BCD 码和无权 BCD 码。

表 1—4 中 8421 码、2421 码、5121 码称为有权 BCD 码，每位二进制数符乘以相应位的"权"，相加后即得到相应的十进制数。

如$(1101)_{2421}=1\times2+1\times4+0\times2+1\times1=(7)_{10}$

$(1000)_{8421}=1\times2^3+0\times2^2+0\times2^1+0\times2^0=(8)_{10}$

8421 码是最常见的 BCD 码，其特点是用四位二进制代码构成 16 种组合状态时，按其编码值的大小顺序从 0000 取用至 1001 以代表十进制制中的 0－9，其余 6 种组合状态为冗余状态。

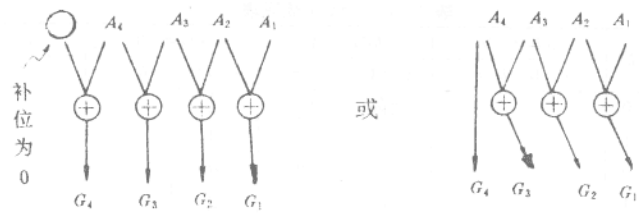
表 1—4 常见的 BCD 码

十进制 \ BCD 码	8421 码	2421 码	5121 码	余 3 码	余 3 循环码
0	0000	0000	0000	0011	0010
1	0001	0001	0001	0100	0110
2	0010	0010 (1000)	0010	0101	0111
3	0011	0011 (1001)	0011	0110	0101
4	0100	0100 (1010)	0111	0111	0100
5	0101	0101 (1011)	1000	1000	1100
6	0110	0110 (1100)	1001	1001	1101
7	0111	0111 (1101)	1010	1010	1111
8	1000	1110	1011	1011	1110
9	1001	1111	1111	1100	1010

余 3 码和余 3 循环码属无权 BCD 码，它们不能用上述方法求得各位代码所对应的十进制数，余 3 码是在 8421 码中加上数值 3，即$(0000)_{8421}=(0011)_{余3}$。

循环码的特点是上下二数码之间只有一位代码值不相同，余 3 循环码可以由余 3 码演变而来。

设余 3 码的四位代码符号为 $A_4A_3A_2A_1$，余 3 循环码为 $G_4G_3G_2G_1$，则二者间存在如下的运算关系



符号⊕为"异或"逻辑运算符号，如果 A_4 和 A_3 代码相异，G_3 必为 1，否则 G_3 必为 0。"异或"逻辑将在第三章中介绍。

运用上述演变规则也可以从 8421 码求得对应的 8421 循环码。循环码又叫格雷码(Gray 码)。

表 1—5　我国通用七单位代码表

				b_7	0	0	0	0	1	1	1	1
				b_6	0	0	1	1	0	0	1	1
				b_5	0	1	0	1	0	1	0	1
b_4	b_3	b_2	b_1	列 / 行	0	1	2	3	4	5	6	7
0	0	0	0	0			SP	0	@	P	、	p
0	0	0	1	1			!	1	A	Q	a	q
0	0	1	0	2			"	2	B	R	b	r
0	0	1	1	3	控		#	3	C	S	c	s
0	1	0	0	4			¥	4	D	T	d	t
0	1	0	1	5	制		%	5	E	U	e	u
0	1	1	0	6			&	6	F	V	f	v
0	1	1	1	7	字		'	7	G	W	g	w
1	0	0	0	8			(	8	H	X	h	x
1	0	0	1	9	符		)	9	I	Y	i	y
1	0	1	0	10			☆	:	J	Z	j	z
1	0	1	1	11			+	;	K	[	k	{
1	1	0	0	12			,	<	L	\	l	|
1	1	0	1	13			—	=	M	]	m	}
1	1	1	0	14			.	>	N	∧	n	—
1	1	1	1	15			/	?	O	—	o	DEL

二、二进制数和二进制代码

用二进制代码表示二进制数也必须遵循代码的组成特性，如果要表示$(0)_{10}$至$(16)_{10}$数值范围内的各种数值，相应的二进制数所对应的二进制代码见表 1—3，此时所用二进制代码必须为五位，从 00000——10000，见表 1—3 第 4 栏。

二进制代码除了限于上述数值编码外，在计算机的控制和操作使用中，还用于对字母符号，图形符号，标点符号，机器操作符号的编码，类似的这类编码有早期的五位博多码，近期的我国通用七单位码(表 1—5)及美国标准信息交换码 ASCII 码(见表 1—6)。ASCII 码共有七位

有效代码，能对 $2^7=128$ 个控制符，字母和数字符号进行编码。例如：字母 A 的七位二进制码 $b_7b_6b_5b_4b_3b_2b_1=1000001$。

表 1—6 ASCII 编码表

b_4	b_3	b_2	b_1	$b_7b_6=00$		$b_7b_6=01$		$b_7b_6=10$		$b_7b_6=11$	
				$b_5=0$	$b_5=1$	$b_5=0$	$b_5=1$	$b_5=0$	$b_1=$	$b_5=0$	$b_5=1$
0	0	0	0			间隔	0	@	P		p
0	0	0	1			!	1	A	Q	a	q
0	0	1	0			//	2	B	R	b	r
0	0	1	1			#	3	C	S	c	s
0	1	0	0			S	4	D	T	d	t
0	1	0	1			%	5	E	U	e	u
0	1	1	0			&	6	F	V	f	v
0	1	1	1			、	7	G	W	g	w
1	0	0	0	控 制 符		(	8	H	X	h	x
1	0	0	1			)	9	I	Y	i	y
1	0	1	0			*	:	J	Z	j	z
1	0	1	1			+	;	K	〔	k	{
1	1	0	0			,	<	L	\	l	|
1	1	0	1			—	=	M	〕	m	}
1	1	1	0			.	>	N	^	n	~
1	1	1	1			/	?	O	—	o	抹掉

在计算机中，一般用一个字节（8 位二进制代码）来表示一个字符，其中低 7 位为 ASCII 码，最高位设置一校验位——奇（偶）校验位。所谓奇校验位是指："在校验位上所设置的代码值必须保证该字节 8 位代码中"1"的个数为奇数"。例如上述字符 A 的 ASCII 码为 1000001，则在最高位应设置代码"1"，表示成：11000001；所谓偶校验位的含义与上述相反，最高校验位的代码设置应保证该字节中的"1"的个数为偶数，如有偶校验位的字符 A 的 8 位代码是 01000001。

字符信号在加工处理和传送过程中，如果有某位代码的"1"、"0"错翻，则就破坏了奇、偶的设置规则，就能及时发现错误。

三、带符号的二进制数的代码表示法

二进制数带上符号后有正数和负数之分，例 +0.1101、−1001。

负数的二进制代码有三种表示方法：

1）原码表示法，直接用符号位和数值代码来表示。

2）反码表示法，用符号位和数值代码的反码来表示。所谓"反码"即将原码数值代码中的"0"写成"1"，"1"写成"0"，逐位取反。

3）补码表示法，用符号位和数值代码的补码来表示。所谓"补码"，即在反码的末位加 1。

探讨负数的多种表示法是为了简化数值间的减法运算。事实已经证明负数的补码表示法更有利于减法运算，即可以用同一加法电路进行数值的加、减运算。下面用表 1—7 说明负数的三种代码表示法，对于正数，它们的二进制原码、反码、补码相同。从表中可以看出，正、负数的代码表示法都引进了符号位，正数为 0，负数为 1，即用数码 0 替代"＋"号，数码"1"替代"－"号，数值位和符号位间用逗号"，"加以划分。

表 1—7　数的原码、反码和补码形式

数值范围　$\|(0)_{10}\sim(63)_{10}\|$				
正　　数	负　　数			
例 $+(9)_{10}=+(001001)$	例 $-(9)_{10}=-(001001)_2$			
二进制的{原码 反码 补码}相同	原　码	反　码	补　码	补码变形码
数值范围六位 0　001001 ↓　↓ 正数符号位　数值代码	数值范围六位 1　001001 ↓　↓ 负数符号位　数值代号	1　110110 ↓　↓ 负数符号位　原码逐位取反	1　110111 ↓　↓ 负数符号位　反码末位加 1	11　110111 ↓　↓ 负数二位符号位　反码末位加 1

在作补码的加、减运算时，还需考虑数值位的溢出问题。

【例 1】　$A=+100011, B=+110010$，求 $A+B=?$

解：　$A_{补}=0,100011, B_{补}=0,110010$

则　$A_{补}+B_{补}=[A+B]_{补}$

$$\begin{array}{lr} & A_{补} \quad 0,100011 \\ +) & B_{补} \quad 0,110010 \\ \hline & 1,010101 \end{array}$$

正数相加得负数，其结果显然是错误的，其实质是二数相加后其数值大小超过六位代码所能表示的数值范围，所以在计算机中称为“溢出”。

【例 2】　$A=-100011 \quad B=-110010$　求 $A+B=?$

解：　$A_{补}=1,011101 \quad B_{补}=1,001110$

$A_{补}+B_{补}=[A+B]_{补}$

$$\begin{array}{lr} & A_{补} \quad 1,011101 \\ +) & B_{补} \quad 1,001110 \\ \hline & 10,101011 \end{array}$$

二负数相加得到正数？这种错误原因同上，也称为数位的溢出。

为判断补码加、减法运算的溢出问题，采用二位符号位的补码变形码形式，见表 1—7。

用变形补码进行数值加减运算时，若出现运算结果中的二位符号位不相同，则最高符号位代替数值的正、负符号，另一符号位作为数值处理，若运算结果中的二位符号位相同，则 00 代表正值，11 代表负值。

上述例 1、例 2 的变形补码运算如下：

$A_{补}+B_{补}=[A+B]_{补}$

$$\begin{array}{lr} & A_{补} \quad 00,100011 \\ +) & B_{补} \quad 00,110010 \\ \hline & 01,010101 \end{array}$$

运算结果为 $[A+B]_{补}=01,010101$

相应的 $A+B=+1010101$

$A_{补}+B_{补}=[A+B]_{补}$

$$\begin{array}{lr} & A_{补} \quad 11,011101 \\ +) & B_{补} \quad 11,001110 \\ \hline & 1\,10,101011 \end{array}$$

丢去　符号位　作数位处理

由于运算结果为负数的补码形式，则相应的 $A+B=-1010101$。

对于补码、反码的加、减运算法则这里不作详细讨论。

【本章小结】

1. 脉冲波形的主要参数有：脉冲周期 T；脉冲宽度 T_w；脉冲波形的前沿时间 t_r；脉冲波形的后沿时间 t_f 以及脉冲波形的最大幅度 V_m。上述的时间参数的计量单位可以为秒(S)、毫秒(ms)、微秒(μs)以及毫微秒(ns)。

2. 在数字电路和计算机中，最常用的数制为二进制、十进制和十六进制数。二进制（或任意进制数）转换为十进制数时，各数位的数符乘以各数位的不同“权”值，并在十进制计算规则下求其总和，即可得到转换后的十进制数。

3. 十进制数转换为二进制（或任意进制）数时，通常采用基数乘（小数部分的转换）、除法（整数部分的转换）。二进制和十六进制数的相互转换，可采用直接转换法。任意数制间的转换用过渡法。

4. 在计算机中，十进制数以二—十进制编码(BCD)码形式出现。BCD 码可以分为有权 BCD 码和无权 BCD 码。在无权 BCD 码中以格雷码较为常见，它和有权 BCD 码的转换关系符合下列公式：

$$G_i = B_i \oplus B_{i+1}$$

【思考题和习题】

1. 脉冲波形的基本参数有哪几个？

2. 数字信号和模拟信号的区别何在？

3. 何谓数制中的数符、基数和数位的“权”？

4. 将下列各数按基数展开成该数的多项式表示形式。

   $(10101)_2$　　$(3472)_{10}$　　$(138)_8$　　$(429)_{16}$

5. 将下列十进制数转换为二进制数。

   6　　13　　28　　250　　4096

6. 将下列二进制数转换为十进制数。

   101　　1101　　11010　　110101　　0.01001　　1010101.001

7. 将下列二进制数转换为十六进制数。

   10101101　　100101011　　10110001010

8. 将下列十六进制数转换成二进制数。

   4A　　B5　　CE　　AD7　　9BC8

9. 将下列十进制数写成 8421BCD 码。

   957　　3421　　860　　2607　　1988

10. 将下列 8421 码改写成十进制数。

   0111101　　100001011000　　11101110101

11. 将下列十进制小数转换成二进制数。

   0.18　　0.765　　0.625　　0.904

12. 将下列十进制数表示为 2421 码。

   145　　726　　6725

13. 将下列 2421 码表示为十进制数。

   1111　　1011　　0100　　1110　　0001

14. 何谓 BCD 码、有权 BCD 码，无权 BCD 码？

15. 带符号数的数码表示方式有几种？相互间有何直接或间接的转换关系？

第二章　晶体二极管和三极管的开关特性

晶体二极管和三极管是组成脉冲数字电路的基本器件，它们在电路中当作开关来使用，晶体管导通时就象开关接通，而截止时犹如开关断开。因此，晶体管一直处于开关工作的状态之中。本章主要分析在开关状态下，晶体二极管和三极管的特性及其简单的应用。

2—1　半导体二极管

一、PN 结

众所周知，晶体管器件都是由半导体材料制成，半导体材料主要使用硅和锗两种，其导电性能介于导体和绝缘体之间。

单晶状态下的半导体材料的性能是稳定的，原子之间排列整齐。硅(S_i)和锗(G_e)的原子结构外层电子都是四个，故称为四价元素。如图 2－1 所示。单晶状态下的排列如图 2－2(a)所示，原子间距离相等。(距离约 2.35×10^{-4}微米)，互相以共价键结构进行组合。

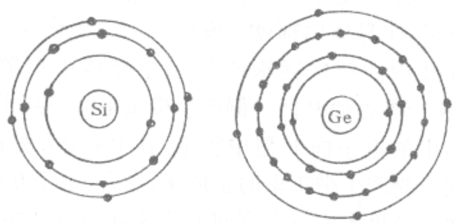
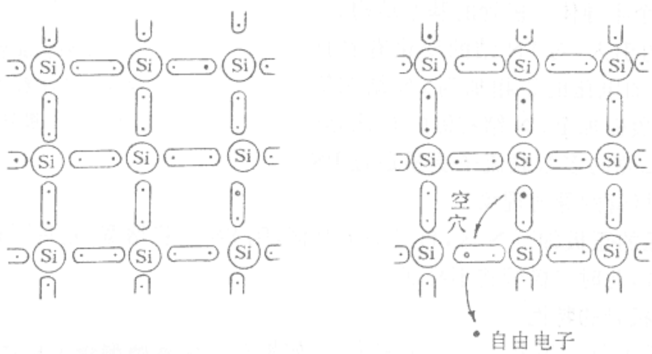


图 2－1　硅和锗原子结构图

如果在外界条件发生变化时，(如温度变化)，少数电子挣脱出来成为自由电子，这是一个带负电荷的电子载流子，而电子“跑”出来后留下了一个空穴，如果这个空穴又被另一个电子来填补，而



(a)　　(b)

图 2－2　硅晶体共价键结构与电子空穴对的产生

另一个电子原来的位置上又出现一个新的空穴，这种空穴的移动形成了带正电荷的空穴载流子。见图 2－2(b)所示。

在单晶半导体材料中人为地掺入不同的元素(称杂质)，则半导体性能将会发生很大的变

此为试读，需要完整PDF请访问：www.ertongbook.com