



数字电路与逻辑设计

宗 汝

西安电子科技大学电子工程学院

Email: zongru@xidian.edu.cn

目 录

- 一、数制与编码
- 二、逻辑代数基础
- 三、组合逻辑电路
- 四、触发器
- 五、时序电路的分析与设计

第一章 数值与编码

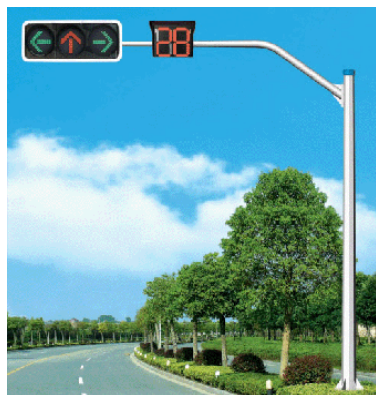
1.1 数字电路概述

1.2 数制

1.3 编码

1.1 数字电路概述

一、我们身边的数字电路应用

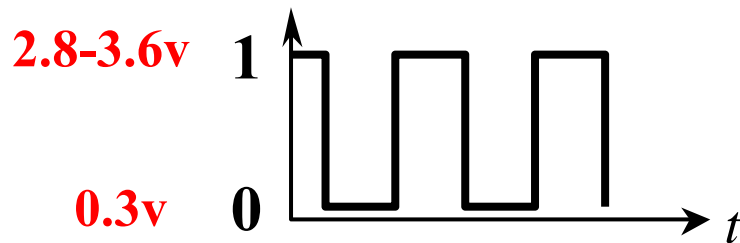


二、什么是数字电路

数字电路：用数字信号进行算术运算和逻辑运算的电路，称为数字电路或数字系统。

数字电路具有逻辑运算和逻辑处理功能，又称数字逻辑电路。

数字信号：时间和幅值的变化是离散的信号。即时间上离散，幅值上整数化（低电平表示逻辑**0**，高电平表示逻辑**1**）。

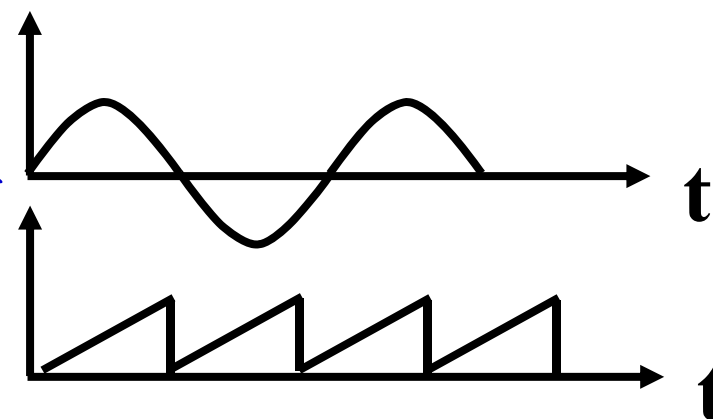


1.1 模拟信号与数字信号

模拟信号:

在时间和数值上连续变化的信号

时间上连续, 幅值上连续

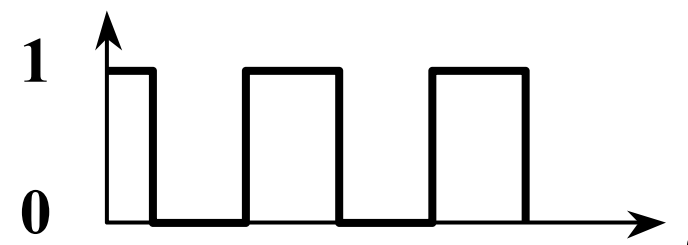


对模拟信号进行传输、处理的电子线路称为**模拟电路**。

数字信号:

在时间和数值上变化是离散的信号

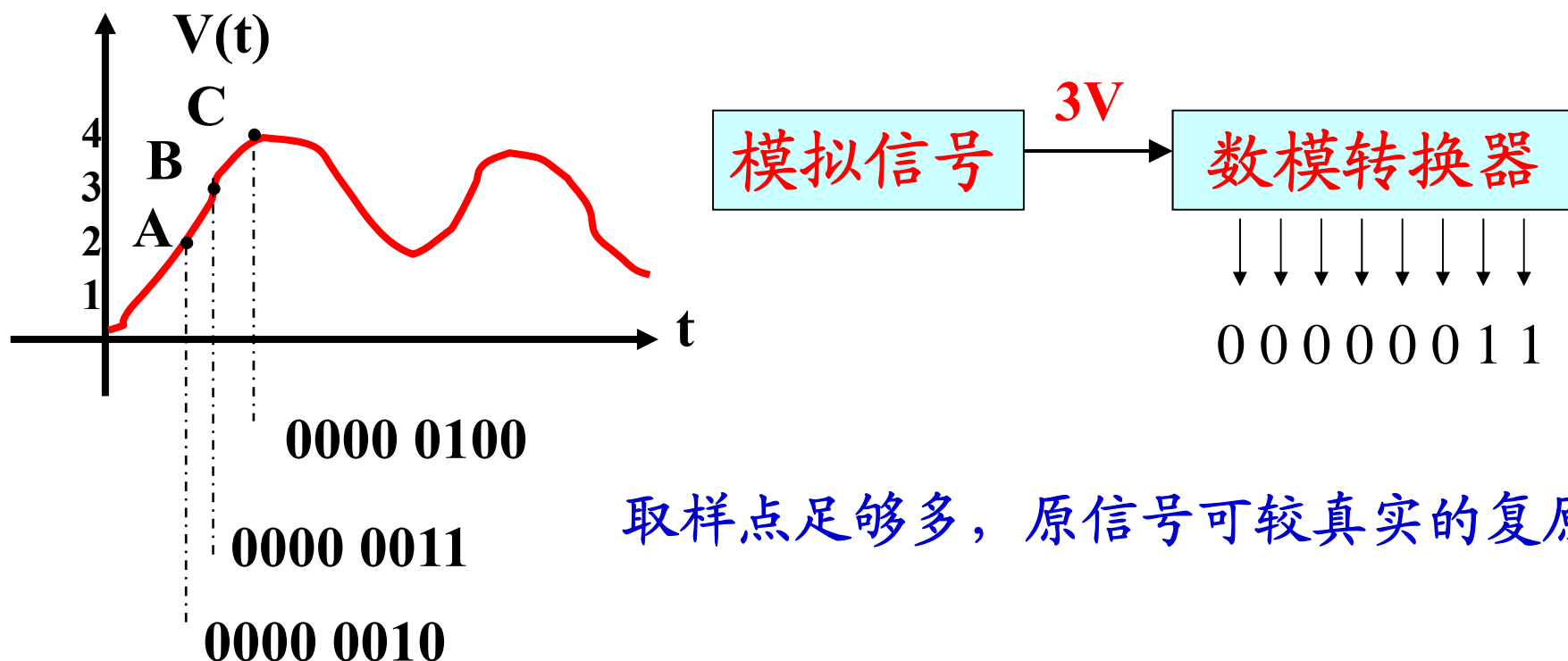
时间上离散, 幅值上整数化



对数字信号进行传输、处理的电子线路称为**数字电路**。

1.1 模拟信号与数字信号

模拟量用数字0、1的编码表示



取样点足够多，原信号可较真实的复原

1.2 电子技术概述

十九世纪末、二十世纪初开始发展的新兴技术。

电子技术由模拟电子技术和数字电子技术两部分构成。

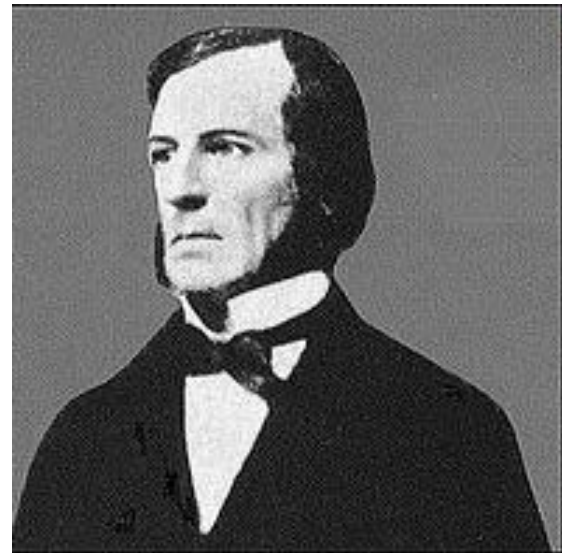
电子技术的发展与电子器件的发展紧密相关。

1.3 数字电路

同期理论的发展

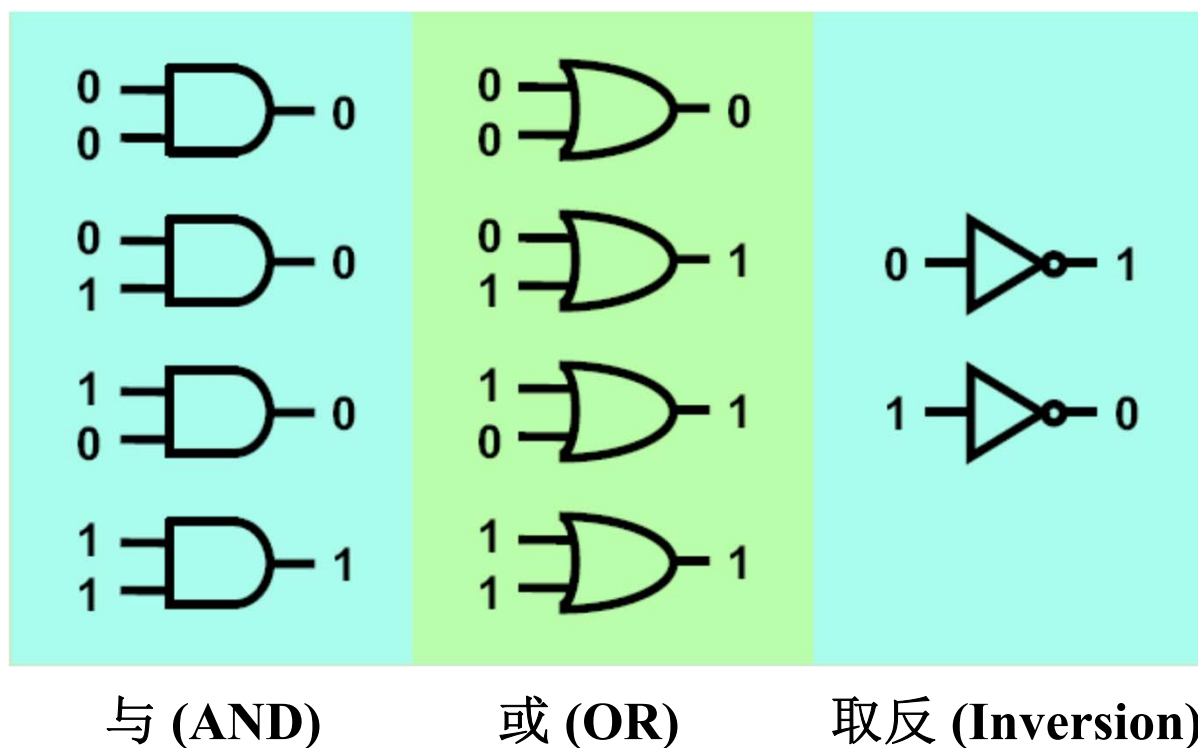
布尔代数 (Boolean Logic) ——现代数字计算机逻辑的基础

1854年, George Boole指出:
逻辑不仅仅是哲学, 也是数学



George Boole
(2 Nov. 1815 – 8 Dec. 1864)

1.3 数字电路



1.3 数字电路

数字电路的优点

- (1) 便于高度集成化。
- (2) 工作可靠性高、抗干扰能力强。
- (3) 数字信息便于长期保存。
- (4) 数字集成电路产品系列多、通用性强、成本低。
- (5) 保密性好。

1.4 数制

1.4.1 进位计数制

1.4.2 二进制数

1.4.3 八进制数和十六进制数

1.4.4 数制转换

进位计数制

N进制：以N为基数的记数体制

基 数
(Base)

1. 有N个数码(Digit): $0 \sim (N-1)$

2. 逢N进1

第i位的权(The ith power of N)

3.
$$(D)_N = \sum_{i=-\infty}^{\infty} K_i \times N^i$$

第i位的系数

对于一个基数为 $R(R \geq 2)$ 的 R 进制计数制，数 N 可以写为

$$\begin{aligned}(N)_R &= a_{n-1}a_{n-2} \cdots a_1a_0 \cdot a_{-1}a_{-2} \cdots a_{-m} & 0 \leq a_i \leq R-1 \\&= a_{n-1} \times R^{n-1} + a_{n-2} \times R^{n-2} + \cdots + a_1 \times R^1 + a_0 \times R^0 + a_{-1} \times R^{-1} \\&\quad + a_{-2} \times R^{-2} + \cdots + a_{-m} \times R^{-m} \\&= \sum_{i=-m}^{n-1} a_i R^i\end{aligned}$$

式中， n 代表整数位数， m 代表小数位数， a_i 为第 i 位数码，它可以是 $0、1、\dots、(R-1)$ 个不同数码中的任何一个， R^i 为第 i 位数码的权值。

十进制 (Decimal number)

基数

$N = 10$

系数

$K_i : 0 \sim 9$

第*i*位的权

$N^i : 10^i$

$$7642_{10} = 7 \times 10^3 + 6 \times 10^2 + 4 \times 10^1 + 2 \times 10^0$$

位置记数法或
并列表示法

多项式表示法或
按权展开法

二进制数(Binary number)

基数

$$N = 2$$

系数

$$K_i: 0, 1$$

第i位的权

$$N^i: 2^i$$

$$101111_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 47_{10}$$

八进制数 (Octal number)

基数

$$N = 8$$

系数

$$K_i: 0 \sim 7$$

第*i*位的权

$$N^i: 8^i$$

$$1352_8 = 1 \times 8^3 + 3 \times 8^2 + 5 \times 8^1 + 2 \times 8^0 = 746_{10}$$

十六进制 (Hexadecimal number)

$$N = 16$$

$$K_i : 0 \sim 9 \text{ A,B,C,D,E,F}$$

$$N^i : 16^i$$

$$2EA_{16} = 2 \times 16^2 + 14 \times 16^1 + 10 \times 16^0 = 746_{10}$$

数制转换

1. 二进制到十进制——按权展开法

$$101111_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 47_{10}$$

同理，若将任意进制数转换为十进制数，只需将数 $(N)_R$ 写成按权展开的多项式表示式，并按十进制数规则进行运算，便可求得相应的十进制数 $(M)_{10}$ 。

数制转换

2.十进制数转换成二进制数

(1) 整数转换——除2取余法

例：把53.375转换为二进制数

整数部分：

2		<u>53</u>	余数=1=b ₀
2		<u>26</u>	余数=0=b ₁
2		<u>13</u>	余数=1=b ₂
2		<u>6</u>	余数=0=b ₃
2		<u>3</u>	余数=1=b ₄
2		<u>1</u>	余数=1=b ₅
		0		

数制转换原理

2.十进制数转换成二进制数

(1) 整数转换——除2取余法

如将十进制整数 $(N)_{10}$ 转换为二进制整数 $(M)_2$ 。

$$\begin{aligned}(N)_{10} &= (a_{n-1}a_{n-2} \cdots a_1a_0)_2 & a_i &= 0, \text{ or } 1 \\ &= a_{n-1} \times 2^{n-1} + a_{n-2} \times 2^{n-2} + \cdots + a_1 \times 2^1 + a_0 \times 2^0 \\ &= 2(a_{n-1} \times 2^{n-2} + a_{n-2} \times 2^{n-3} + \cdots + a_2 \times 2^1 + a_1) + a_0 \\ &= 2Q_1 + a_0\end{aligned}$$

将上式两边同除以2，所得的商为：

$$Q_1 = (a_{n-1} \times 2^{n-2} + a_{n-2} \times 2^{n-3} + \cdots + a_2 \times 2^1 + a_1)$$

余数就是 a_0 。

数制转换原理

前面得到的商: $Q_1 = (a_{n-1} \times 2^{n-2} + a_{n-2} \times 2^{n-3} + \cdots + a_2 \times 2^1 + a_1)$

同理, 这个商 Q_1 又可以写成:

$$Q_1 = 2(a_{n-1} \times 2^{n-3} + a_{n-2} \times 2^{n-4} + \cdots + a_2) + a_1$$

显然, 若将上式两边再同时除以2, 则所得余数是 a_1 。重复上述过程, 直到商为0, 就可得二进制数的数码 a_0 、 a_1 、 \cdots 、 a_{n-1} 。

即:

$$(N)_{10} = (a_{n-1}a_{n-2} \cdots a_1a_0)_2$$

可见，十进制整数转换为二进制整数，采用**连除2取余法**(先余的是高位，后余的是低位)：

- (1) 将N除以2，记下所得的商和余数；
- (2) 将上一步所得的商再除以2，记下所得的商和余数；
- (3) 重复第(2)步，直到商为0为止；
- (4) 按照**和运算过程相反的顺序**把各个余数排列起来，即为所求的2进制数。

2.十进制数转换成二进制数

(2) 小数转换——乘2取整法

小数部分: 0.375

$$\begin{array}{r} \times \quad 2 \\ \hline 0.750 \end{array} \dots\dots \text{整数部分} = 0 = b_{-1}$$

$$\begin{array}{r} \times \quad 2 \\ \hline 1.500 \end{array} \dots\dots \text{整数部分} = 1 = b_{-2}$$

$$\begin{array}{r} \times \quad 2 \\ \hline 1.000 \end{array} \dots\dots \text{整数部分} = 1 = b_{-3}$$

② 小数转换——乘2取整法。

如将十进制小数 $(N)_{10}$ 转换为二进制小数 $(M)_2$ 。

$$\begin{aligned}(N)_{10} &= (a_{-1}a_{-2}\cdots a_{-m})_2 \\ &= a_{-1} \times 2^{-1} + a_{-2} \times 2^{-2} + \cdots + a_{-m} \times 2^{-m} \\ &= 2^{-1} \times (a_{-1} + a_{-2} \times 2^{-1} + \cdots + a_{-m} \times 2^{-m+1})\end{aligned}$$

上式两边同时乘以2，便得到

$$2(N)_{10} = a_{-1} + \underline{(a_{-2} \times 2^{-1} + \cdots + a_{-m} \times 2^{-m+1})}$$

$$2(N)_{10} = a_{-1} + \underline{F_1}$$

因此， $2(N)_{10}$ 乘积的整数部分就是 a_{-1} 。若将 $2(N)_{10}$ 乘积的小数部分 F_1 再乘以2，则可以依次得到 a_{-2}, \dots, a_{-m} 。

可见，十进制小数转换为二进制小数，采用**连乘2取整法**（先取的是高位，后去的是低位）：

- （1）将N乘以2，记下整数部分；
- （2）将上一步乘积中的小数部分再乘以2，记下整数部分；
- （3）重复第（2）步，直到小数部分为0或满足精度要求为止；
- （4）按照**和运算过程相同的顺序**把各个整数部分排列起来，即为所求的2进制小数。

整数部分: $53D = 110101B$
小数部分: $0.375D = 0.011B$
所以 $53.375D = 110101.011B$

练习:

$173.8125D = ?$

$173.8125D = 10101101.1101B$

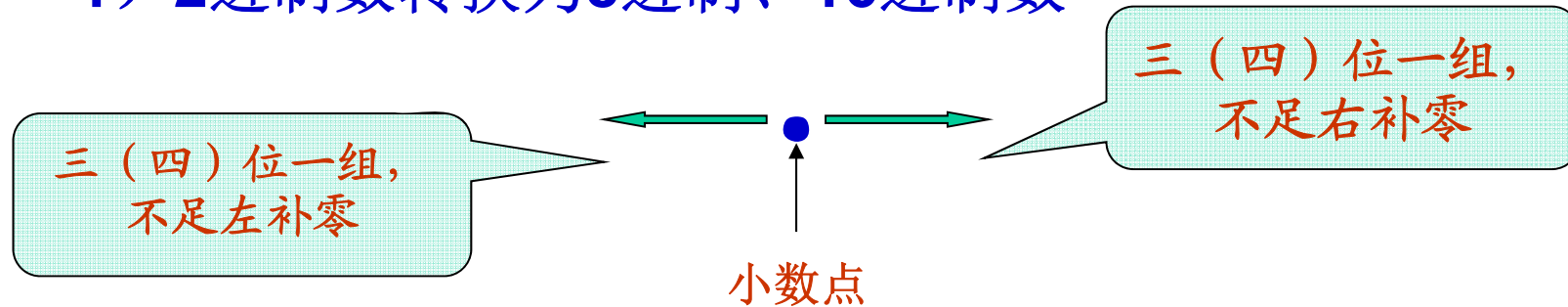
常用2的幂级数

n	2ⁿ	n	2ⁿ
1	2	9	512
2	4	10	1024
3	8	11	2048
4	16	12	4096
5	32	13	8192
6	64	14	16384
7	128	15	32768
8	256	16	65536

3.二进制数和八进制数(十六)间的转换

八进制数和十六进制数的基数分别为 $8=2^3$, $16=2^4$,

1) 2进制数转换为8进制、16进制数



2) 8进制、16进制数转换为2进制数

8进制数 \longrightarrow 2进制数: 1位变3位

16进制数 \longrightarrow 2进制数: 1位变4位

4. 八进制 (十六) 与十进制之间的转换

Example 1:

$$(110101.011000111)_2 = (\quad ? \quad)_8 = (\quad ? \quad)_{16}$$

110 101 . 011 000 111

6 5 . 3 0 7

110101.011000111

0011 0101 . 0110 0011 1000

3 5 . 6 3 8

$$(110101.011000111)_2 = (65.307)_8 = (35.638)_{16}$$

4. 八进制 (十六) 与十进制之间的转换

Example 2:

$$(2EA)_{16} = (\quad ? \quad)_{10}$$

2 E A

0010 1110 1010

$$(001011101010)_2 = 512 + 128 + 64 + 32 + 10 = 746_{10}$$

$$(2EA)_{16} = (746)_{10}$$

1.5 二进制编码 (BCD码)

对于若干个不同的数据或信息，按照一定的规律分别给其指定一个代表符号的过程叫**编码**。

用一定位数的二进制数来表示十进制数码、字母、符号等信息称为**二进制编码**。

这一定位数的二进制数就称为**代码**。

对于**N**个信息，要用几位二进制数才能满足编码呢？

$$2^n \geq N$$

一、二—十进制编码--BCD码

用 4 位二进制数码表示一位十进制数的 0 ~ 9 十个状态，称这些代码为二 - 十进制代码，即 **BCD (Binary Coded Decimal)** 代码。

几种常见的BCD码

十进制数 \ 编码种类	8421码	余3码	2421码	5421码	余3循环码
0	0000	0011	0000	0000	0010
1	0001	0100	0001	0001	0110
2	0010	0101	0010	0010	0111
3	0011	0110	0011	0011	0101
4	0100	0111	0100	0100	0100
5	0101	1000	1011	1000	1100
6	0110	1001	1100	1001	1101
7	0111	1010	1101	1010	1111
8	1000	1011	1110	1011	1110
9	1001	1100	1111	1100	1010
权	8421		2421	5421	

8421BCD码和十进制间的转换

直接按位（按组）转换。

如： $(3.6)_{10} = (0011.0110)_{8421BCD} = (11.0110)_{8421BCD}$

$(101\ 0001\ 0111\ 1001)_{8421BCD} = (5179)_{10}$



补0

Why BCD?

- 十进制——人类认知形式
- 二进制——机器处理形制

二、可靠性编码

1.格雷码（**Gray**码） 格雷码是一种典型的循环码。

循环码特点：

- ①**相邻性**：任意两个相邻码组间仅有一位的状态不同。
- ②**循环性**：首尾两个码组也具有相邻性。

十进制数	格雷码	十进制数	格雷码
0	0000	8	1100
1	0001	9	1101
2	0011	10	1111
3	0010	11	1110
4	0110	12	1010
5	0111	13	1011
6	0101	14	1001
7	0100	15	1000

一种典型的格雷码

两位格雷码

0	0
0	1

1	1
1	0

三位格雷码

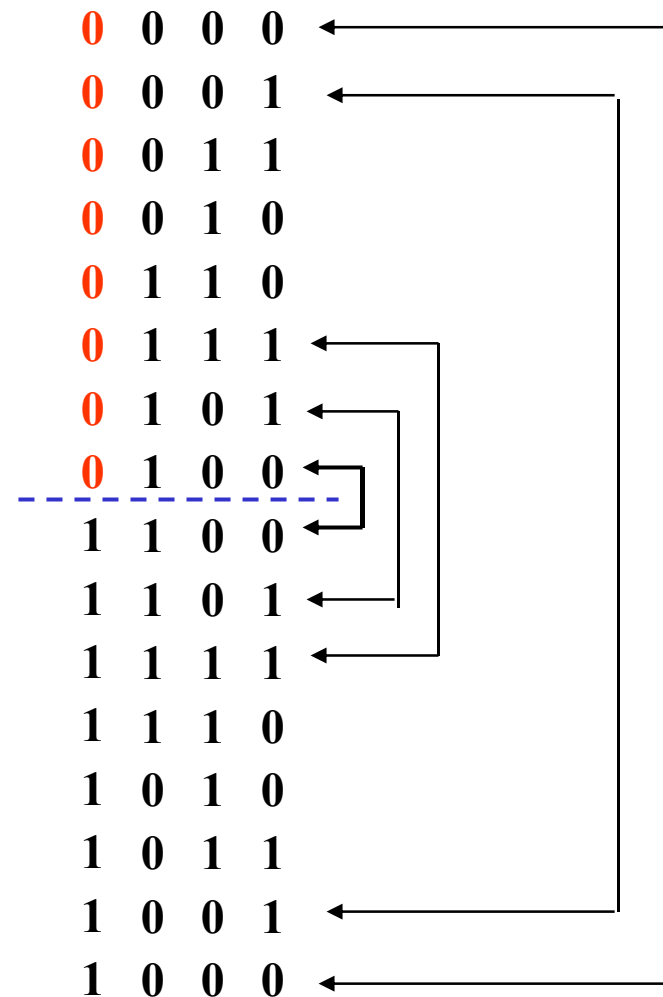
0	0	0
0	0	1
0	1	1
0	1	0

1	1	0
1	1	1
1	0	1
1	0	0

四位格雷码

0	0	0	0
0	0	0	1
0	0	1	1
0	0	1	0
0	1	1	0
0	1	1	1
0	1	0	1
0	1	0	0

1	1	0	0
1	1	0	1
1	1	1	1
1	1	1	0
1	0	1	0
1	0	1	1
1	0	0	1
1	0	0	0



格雷码具有反射特性。反射特性是指以编码最高位0和1的交界处为对称轴，处于对称位置的各代码除最高位不同外，其余各位均相同。低位镜像，高位取反

Why Gray?

- 编码的相邻特性

2. 奇偶校验码

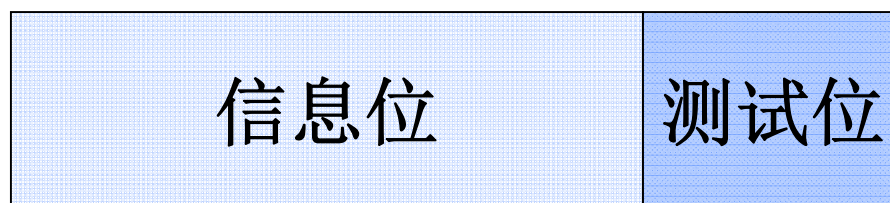
代码(或数据)在传输和处理过程中,有时会出现代码中的某一位由 **0** 错变成 **1**, 或 **1** 变成 **0**。奇偶校验码由信息位和一位奇偶检验位两部分组成。

信息位: 是位数不限的任一种二进制代码。

检验位: 仅有一位, 它可以放在信息位的前面, 也可以放在信息位的后面。

奇校验码 (odd codes)

信息位与测试位1的个数之和为奇数



偶校验码 (Even codes)

信息位与测试位1的个数之和为偶数 .

8421BCD奇偶校验码

十进制数	8421BCD奇校验码	8421BCD偶校验码
	信息位 校验位	信息位 校验位
0	0 0 0 0 1	0 0 0 0 0
1	0 0 0 1 0	0 0 0 1 1
2	0 0 1 0 0	0 0 1 0 1
3	0 0 1 1 1	0 0 1 1 0
4	0 1 0 0 0	0 1 0 0 1
5	0 1 0 1 1	0 1 0 1 0
6	0 1 1 0 1	0 1 1 0 0
7	0 1 1 1 0	0 1 1 1 1
8	1 0 0 0 0	1 0 0 0 1
9	1 0 0 1 1	1 0 0 1 0

Why Odd or Even?

- 检测传输或处理错误

3. ASCII码

ASCII码即 American Standard Code for Information Interchange (美国国家标准信息交换码) 的英文缩写。

常用的有两种：

ASCII-7编码用7 位二进制编码表示一个字符，共可表示 128 个不同的字符。通常使用时在最高位添 0 凑成 8 位二进制编码，或根据实际情况将最高位用做校验位。

ASCII-8编码用 8 位二进制编码表示一个字符，共可表示 256 个不同的字符。

		0	1	2	3	4	5	6	7
		B ₇ B ₆ B ₅							
		B ₄ B ₃ B ₂ B ₁							
0	0 0 0 0	NUL	DLE	Sp	0	@	P	'	p
1	0 0 0 1	SOH	DC1	!	1	A	Q	a	q
2	0 0 1 0	STX	DC2	"	2	B	R	b	r
3	0 0 1 1	ETX	DC3	#	3	C	S	c	s
4	0 1 0 0	EOT	DC4	\$	4	D	T	d	t
5	0 1 0 1	ENQ	NAK	%	5	E	U	e	u
6	0 1 1 0	ACK	SYN	&	6	F	V	f	v
7	0 1 1 1	BEL	ETB	'	7	G	W	g	w
8	1 0 0 0	BS	CAN	(8	H	X	h	x
9	1 0 0 1	HT	EM)	9	I	Y	i	y
A	1 0 1 0	LF	SUB	*	:	J	Z	j	z
B	1 0 1 1	VT	ESC	+	;	K	[k	{
C	1 1 0 0	FF	FS	,	<	L	\	l	
D	1 1 0 1	CR	GS	-	=	M]	m	}
E	1 1 1 0	SO	RS	•	>	N	^	n	~
F	1 1 1 1	SI	US	/	?	O	-	o	DEL

小结

进位计数制

二进制数

八进制数和十六进制数

数制转换

BCD码

格雷码

奇偶校验码

练习：数制转换

1. 把下列二进制数转换为十进制数

a. 110100101

b. 00010111

$$\begin{aligned} \text{a. } 110100101 &= 1 + 4 + 32 + 128 + 256 \\ &= 421_{10} \end{aligned}$$

$$\text{b. } 00010111 = 1 + 2 + 4 + 16 = 23_{10}$$

练习：数制转换

2.把下列十进制数转换为二进制数，假设下列数是无符号数（正数），用12位表示。

a. 47 b. 98 c. 5000

a. 47

$$47 < 64$$

没有 2^6 位或者更高位

$$47 - 32 = 15$$

得到一个 2^5 位

$$15 < 16$$

没有 2^4 位

$$15 - 8 = 7$$

2^3 位

$$7 = 111$$

$$47_{10} = 000000101111_2$$

练习：数制转换

b. 98

$$98_{10} = 000001100010_2$$

C. 5000

不能用 12 bit表示, 因为 $5000 > 2^{12}$

练习：数制转换

3. 把下列数转换为

- i. 八进制数
- ii. 十六进制数

a. 1101011011_2

b. 611_{10}

a. i $011\ 010\ 110\ 111 = 3267_8$

ii $0110\ 1011\ 0111 = 6B7_{16}$

b. i . $611_{10} = 512 + 64 + 32 + 2 + 1$
 $= 2^9 + 2^6 + 2^5 + 2^1 + 2^0$
 $= 1001100011_2 = 001\ 001\ 100\ 011_2 = 1143_8$

ii . $611_{10} = 1001100011_2 = 0010\ 0110\ 0011_2 = 263_{16}$

练习：数制转换

4. 把下列数转换为十进制数

a. 2170_8

b. $1C3_{16}$

$$\begin{aligned} \text{a. } 2170_8 &= 10 \ 001 \ 111 \ 000_2 = 2^{10} + 2^6 + 2^5 + 2^4 + 2^3 \\ &= 1144_{10} \end{aligned}$$

$$\begin{aligned} \text{b. } 1C3_{16} &= 1 \ 1100 \ 0011_2 = 2^8 + 2^7 + 2^6 + 2^1 + 2^0 \\ &= 256 + 128 + 64 + 2 + 1 = 451_{10} \end{aligned}$$

练习：数制转换

5. 用三位4种BCD码来表示以下两个数

a. 491 b. 27

i BCD 8421

iii BCD 2421

ii BCD 5421

iv BCD excess 3

a. 491

b. 27

8421 0100 1001 0001

0000 0010 0111

5421 0100 1100 0001

0000 0010 1010

2421 0100 1111 0001

0000 1111 0001

exs3 0111 1100 0100

0011 0101 1010

练习：数制转换

6. 当以下数为BCD码或者是无符号二进制数时，十进制数为多少？

- i BCD 8421
- ii BCD 5421
- iii BCD 2421
- iv BCD excess 3
- v Binary unsigned

a. 1000 0111

b. 1100 1001

十进制数	8421 BCD	5421 BCD	2421 BCD	余3码 (Es3)
0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 0 0 1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 0 1 0	0 0 1 0	0 1 0 1
3	0 0 1 1	0 0 1 1	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 0 0	0 1 0 0	0 1 1 1
5	0 1 0 1	1 0 0 0	1 0 1 1	1 0 0 0
6	0 1 1 0	1 0 0 1	1 1 0 0	1 0 0 1
7	0 1 1 1	1 0 1 0	1 1 0 1	1 0 1 0
8	1 0 0 0	1 0 1 1	1 1 1 0	1 0 1 1
9	1 0 0 1	1 1 0 0	1 1 1 1	1 1 0 0
不用 unused	1 0 1 0	0 1 0 1	0 1 0 1	0 0 0 0
	1 0 1 1	0 1 1 0	0 1 1 0	0 0 0 1
	1 1 0 0	0 1 1 1	0 1 1 1	0 0 1 0
	1 1 0 1	1 1 0 1	1 0 0 0	1 1 0 1
	1 1 1 0	1 1 1 0	1 0 0 1	1 1 1 0
	1 1 1 1	1 1 1 1	1 0 1 0	1 1 1 1

a. $(1000\ 0111)_{8421\text{BCD}} = 87_{10}$

$(1000\ 0111)_{5421\text{BCD}} / \quad \because 0111 \quad \text{unused}$

$(1000\ 0111)_{2421\text{BCD}} / \quad \because 1000\ 0111 \text{ unused}$

$(1000\ 0111)_{\text{BCDexs3}} = 54_{10}$

$(1000\ 0111)_2 = 135_{10}$

练习：数制转换

b. 1100 1001

$(1100\ 1001)_{8421\text{BCD}} / \quad \because 1100 \text{ useless}$

$(1100\ 1001)_{5421\text{BCD}} = 96_{10}$

$(1100\ 1001)_{2421\text{BCD}} / \quad \because 1001 \text{ useless}$

$(1100\ 1001)_{\text{BCDexs3}} = 96_{10}$

$(1100\ 1001)_2 = 201_{10}$

第1章作业（P14）：

1-1、（2）；（3）

1-2、

1-3、

1-8、①③④