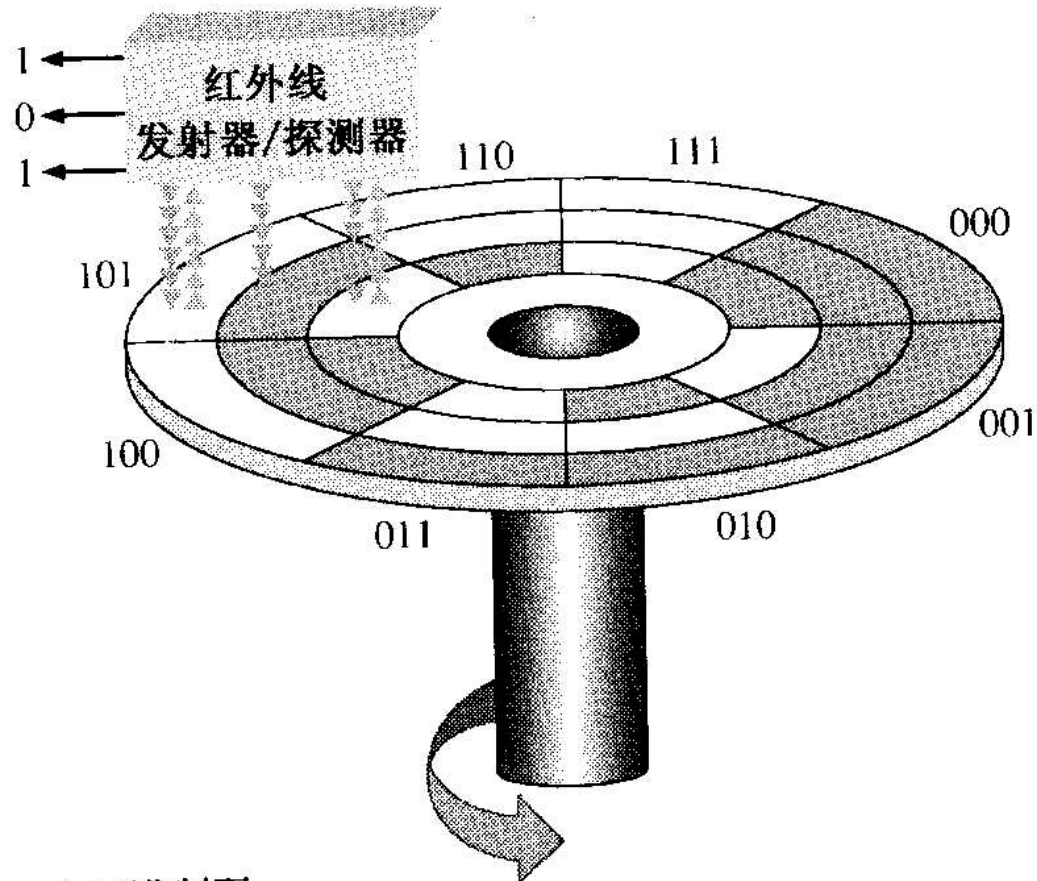


数字编码典型应用

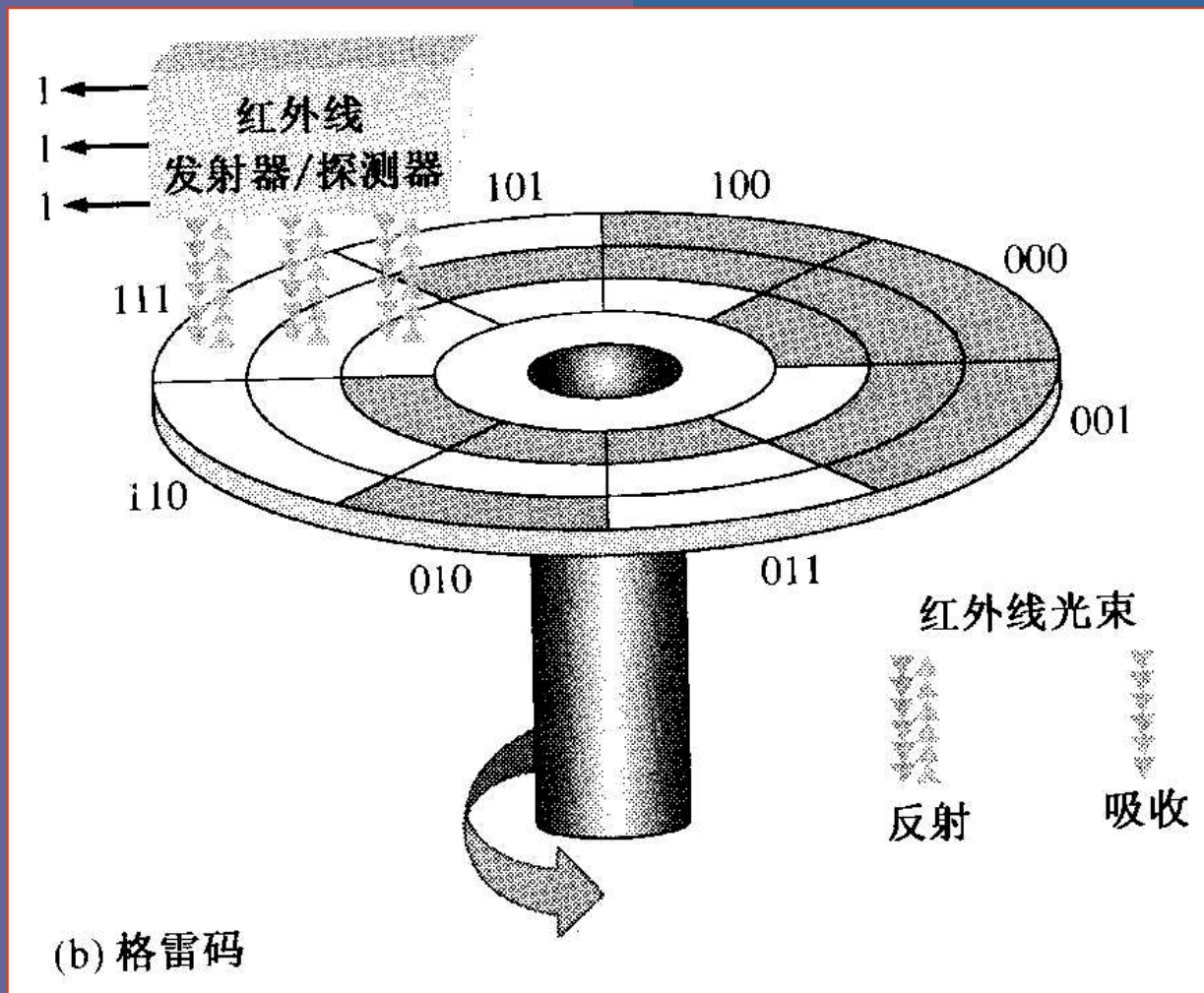
——三位轴位编码器

有三个同心圆环被分成8个扇区（扇区越多，位置就能表示得越准确）。每个圆环的每个扇区分为能够反射光束或不能反射光束两种，当圆环随着轴转动时，处于红外线(IR)发射器下部的扇区分别接收红外线发射器所产生的3条光束。当扇区反射光束时表示1，当扇区不反射光束时表示0。红外线探测器检测是否存在反射光束，然后产生相应的3位编码。红外线发射器 / 探测器处于固定位置，当轴逆时针360°旋转时，8个扇区在3条光束下移动。每一条光束被扇区的表面反射或吸收，产生表示轴位编码器位置的二进制数。



(a) 二进制码

在图(a)中，这些扇区直接以二进制格式排列，所以探测器输出从000到001，再到010，再到011，等等。当光束处于反射扇区上部时，输出为1；当光束处于非反射扇区上部时，输出为0。如果从一个扇区到另一个扇区的转换瞬间，一条光束稍微先于其他光束被反射或吸收，就会产生一个错误的输出



格雷码用来消除二进制所固有的这类出错问题。如图(b)所示，格雷码保证了在两个相邻的扇区中，只有一个位会发生改变。这就意味着即使光束没有精确对准，也绝对不会发生转移瞬间中的错误。例如，再一次考虑，当光束位于111扇区，准备进入下一个101扇区会发生什么情况。无论光束是否对准，在转移瞬间只有两个可能的输出111和101。在其他扇区的转移瞬间中，也会产生相同的结果

增量式编码器SCH50IF



二—十进制码 (Binary Coded Decimal,简称BCD)

- 用四位二进制代码对十进制数的各个数码进行编码。
- BCD码至少为四位。因为四位二进制码共有16个不同的代码，要从16个代码中选出10个表示十进制数符，则共有

$$A_{16}^{10} = \frac{16!}{(16-10)!} \approx 2.9 \times 10^{10}$$

余3码的优点：在用余3码作十进制加法运算的时候，若2数之和是 $(10)_{10}$ ，正好等于二进制数的 $16(10000)$ ，于是便从高位自动产生进位信号。

偏权码

有权码

常用BCD码编码表

无权码

十进制数	8421	2421	631-1	余3码	格雷码	5中取2码	左移码
0	0000	0000	0011	0011	0010	00011	00000
1	0001	0001	0010	0100	0110	00101	10000
2	0010	0010	0101	0101	0111	00110	11000
3	0011	0011	0111	0110	0101	01001	11100
4	0100	0100	0110	0111	0100	01010	11110
5	0101	1011	1001	1000	1100	01100	11111
6	0110	1100	1000	1001	1101	10001	01111
7	0111	1101	1010	1010	1111	10010	00111
8	1000	1110	1101	1011	1110	10100	00011
9	1001	1111	1100	1100	1010	11000	00001

BCD码加/减运算

由于BCD码与十进制数之间存在直接的对应关系，因此有时利用BCD码来完成十进制数的运算，不同BCD码的运算规则各不相同。

8421 BCD码的加/减运算

例如

十进制数相加: $6 + 5 = 11$

8421 BCD 码相加: $0110 + 0101 = 1011$

$9 + 9 = 18$

$1001 + 1001 = 10010$

X?

产生错误的原因是十进制数相加的进位原则是“逢十进一”，而上述BCD码相加采用了二进制数相加时的进位原则，四位二进制数是“逢十六进一”，两者相差6。因此，按二进制数运算规则得到的BCD码运算结果需要修正。

8421 BCD 码加法运算的修正方法是：

当和数大于 9 (1001) 或产生进位时，需对和数本位加 6 (0110) 修正；反之，当和数小于或等于 9 时，无需修正。

例 1 用 8421 BCD 码求 $6+5$ 和 $9+9$ 。

$$\begin{array}{r} 0110 \\ + 0101 \\ \hline 1011 \quad \text{大于 9} \\ + 0110 \quad \text{加 6 修正} \\ \hline 10001 \end{array}$$

$$\begin{array}{r} 1001 \\ + 1001 \\ \hline 10010 \quad \text{有进位} \\ + 0110 \quad \text{加 6 修正} \\ \hline 11000 \end{array}$$

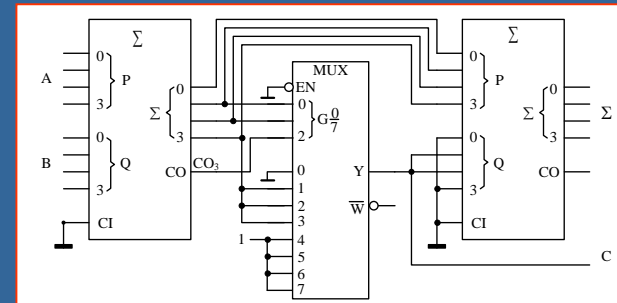
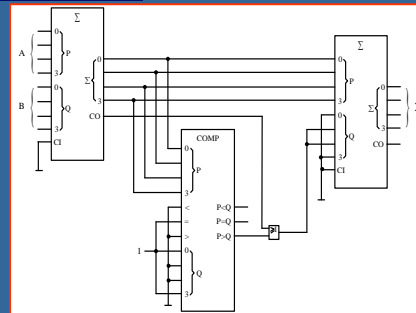
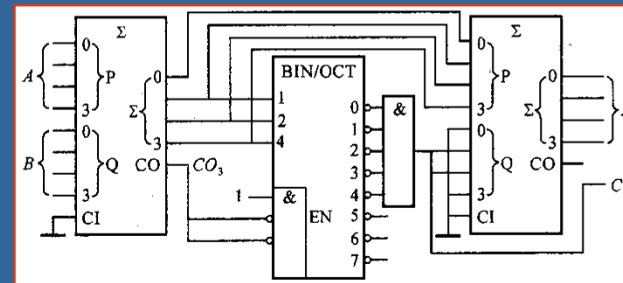
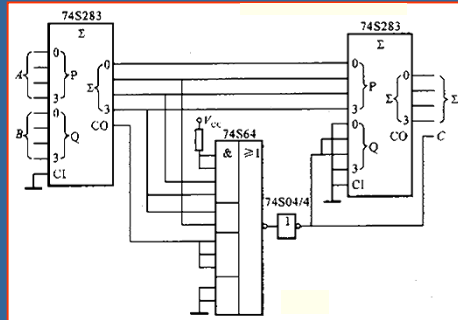
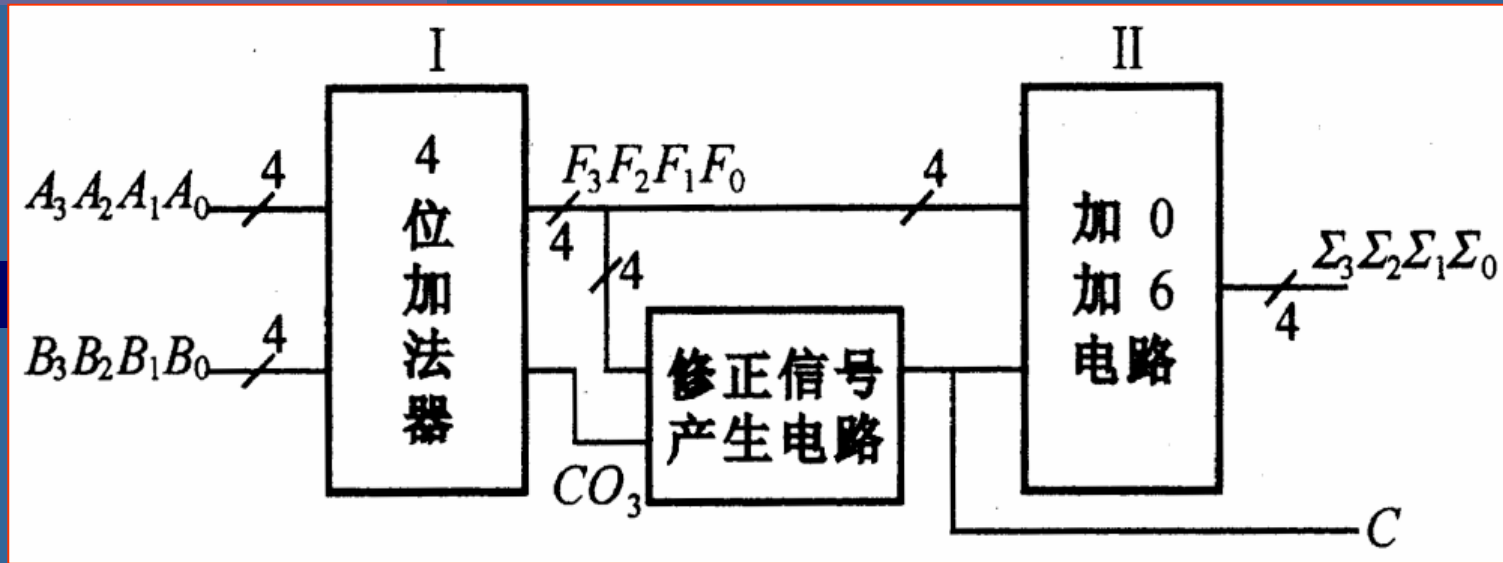
例 2 试用 8421 BCD 码求 $983+819$

$$\begin{array}{r} 1001\ 1000\ 0011 \\ + 1000\ 0001\ 1001 \\ \hline 10001\ 1001\ 1100 \\ + 0110\ 0000\ 0110 \quad \text{个位和百位需修正} \\ \hline 10111\ 1010\ 0010 \\ + \quad \quad 0110 \quad \text{十位需修正} \\ \hline 11000\ 0000\ 0010 \end{array}$$

注意：若由于修正而产生进位时，无需作第二次修正。即每一十进制位最多只能修正一次。

是否还要修正？

1位BCD码加法器方框图



8421BCD码进行减法运算

当 BCD 码向高位借位时，按二进制规则是“借一当十六”，而十进制数相减的借位原则是“借一当十”，所以发生借位时需要修正。修正方法是，当相减过程中出现向高位 BCD 码借位时，本位 BCD 码必需“减 6 修正”。

例 试用 8421 BCD 码完成 $255 - 98$

$$\begin{array}{r} 0010\ 0101\ 0101 \\ - 0000\ 1001\ 1000 \\ \hline 1\ 1011\ 1101 \\ - 0110\ 0110 \\ \hline 1\ 0101\ 0111 \end{array}$$

个位、十位均向高位借位
个位、十位减 6 修正

思考题：余3码的加法运算规则？

?

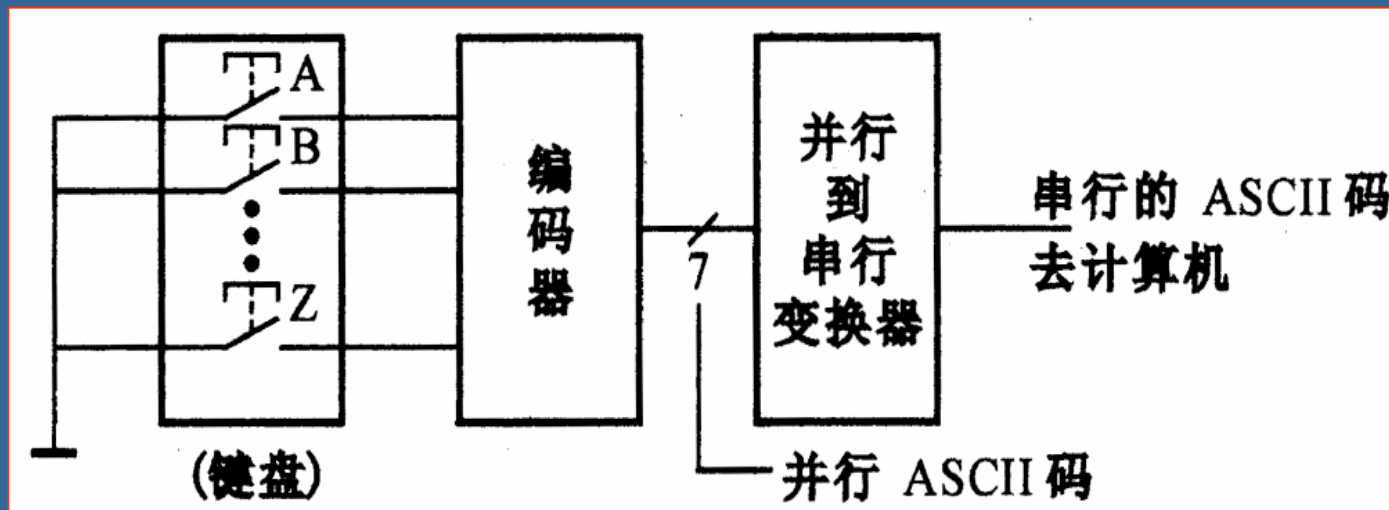
进行加法运算的规则是：
当两个余三码数相加不产生进位时，则应该从结果中减去0011；产生进位时则一方面将进位信号送给高位余三码，另一方面本位还要执行加上0011的修正操作。

字符编码 就是要显示某些文字所需要的编码

ASCII码: American Standard Information Interchange(美国信息交换标准码)

键盘及信号变换

读作
“aske
e”



计算机键盘具有一个专用微处理器，一直扫描键盘电路以检测何时按键被按下或者释放。由计算机软件产生的独特扫描编码表示该特殊的按键。该扫描编码随后被变换为字母数字编码(ASCH)，以备计算机使用。

部分字符的ASCII码

字符	ASCII 码	字符	ASCII 码	字符	ASCII 码
空	0100000	4	0110100	K	1001011
.	0101110	5	0110101	L	1001100
(0101000	6	0110110	M	1001101
+	0101011	7	0110111	N	1001110
\$	0100100	8	0111000	O	1001111
*	0101010	9	0111001	P	1010000
)	0101001	A	1000001	Q	1010001
-	0101101	B	1000010	R	1010010
/	0101111	C	1000011	S	1010011
,	0101100	D	1000100	T	1010100
'	0100111	E	1000101	U	1010101
=	0111101	F	1000110	V	1010110
0	0110000	G	1000111	W	1010111
1	0110001	H	1001000	X	1011000
2	0110010	I	1001001	Y	1011001
3	0110011	J	1001010	Z	1011010

思考题：

汉字是如何编码的？



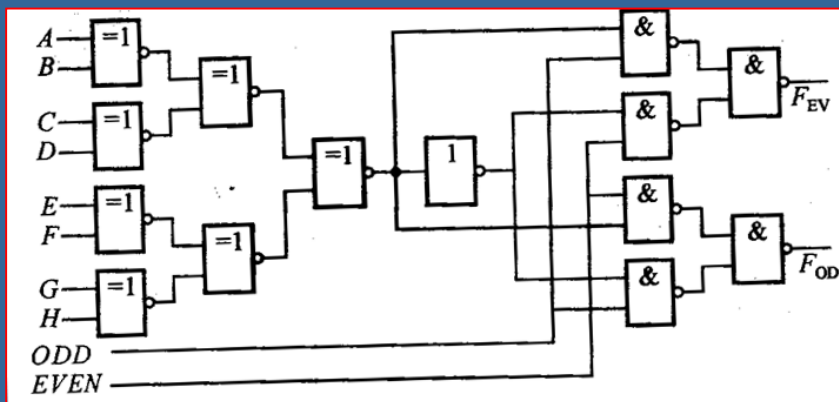
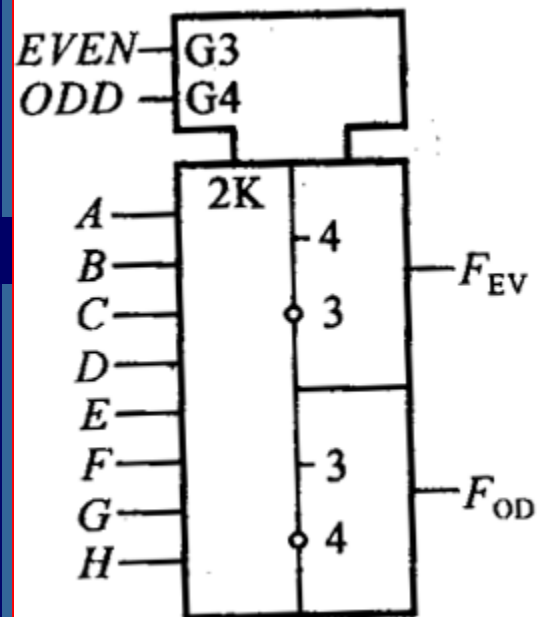
可靠性编码

产生误码的几种形式

正确代码		1001	说 明
误 码	a	0001	单向 1→0
	b	1101	单向 0→1
	c	0000	单向双错
	d	1111	单向双错
	e	0101	双向错误

奇偶校验码

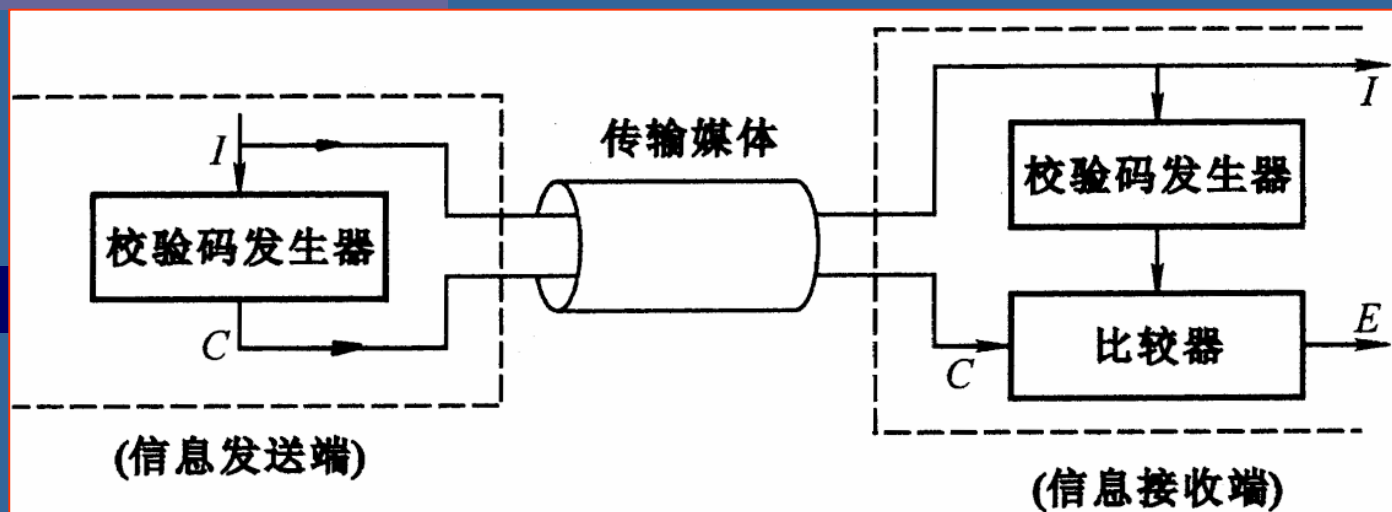
	信息码	奇校验码	偶校验码
0	0000	10000	00000
1	0001	00001	10001
2	0010	00010	10010
3	0011	10011	00011
4	0100	00100	10100
5	0101	10101	00101
6	0110	10110	00110
7	0111	00111	10111
8	1000	01000	11000
9	1001	11001	01001



74180
(奇偶校验器)

功能：用来校验某一组传输的数据有否错误的组合逻辑电路。

1 Berger码信息传递系统



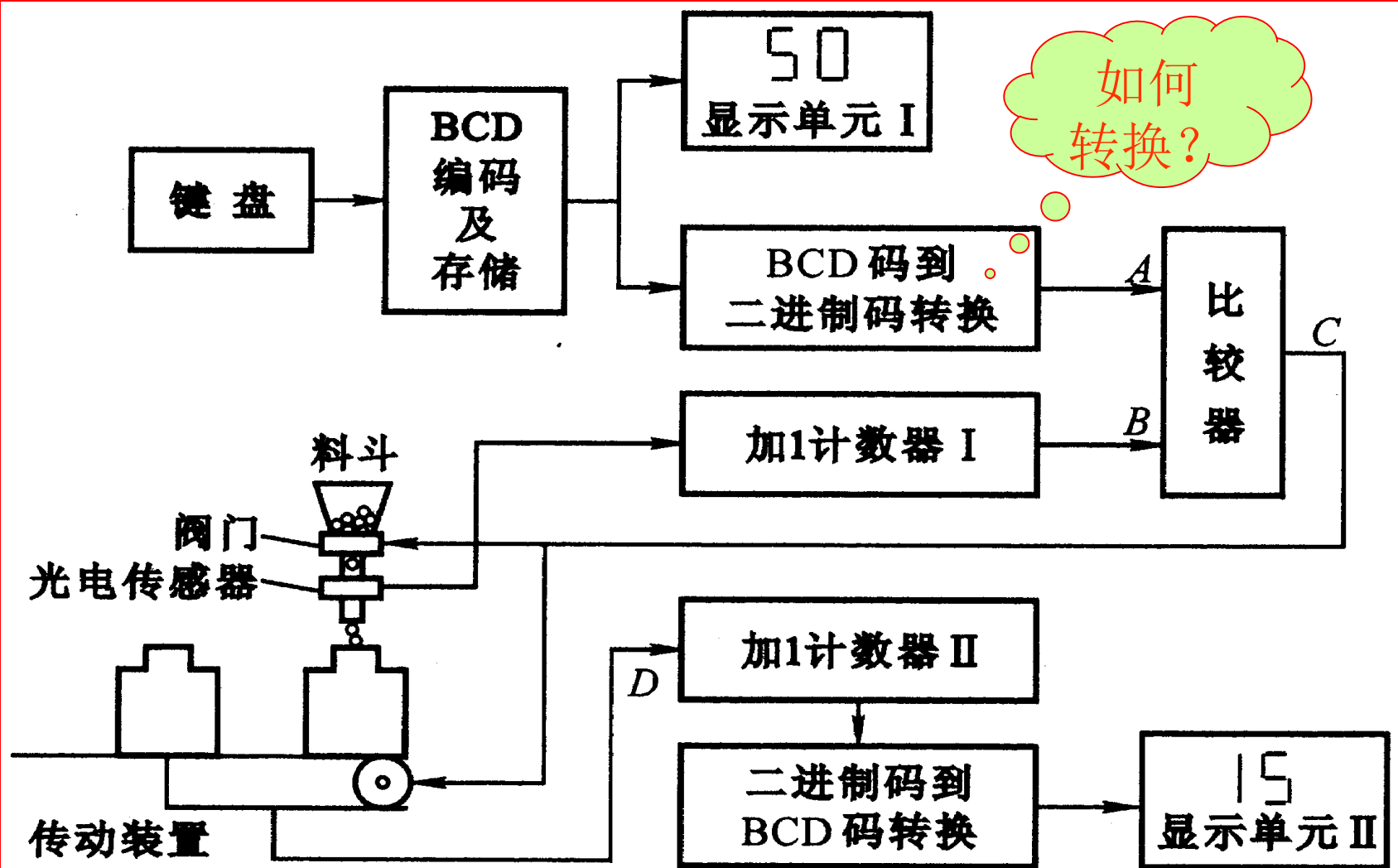
Berger码也是一种可分离码。设信息码 I 中含有的1的个数为是 k ，与 k 等值的二进制数的反码即为校验码。校验码的位数 c 由 $\text{Log}_2(i+1)$ 决定，其中 i 为信息码 I 的位数。

例：试构成0101000的Berger码。

由于 $I=0101000$ ， $i=7$ ，从而校验位的位数 $c=\text{Log}_2(7+1)=3$ ，且因 $k=2$ ，故校验码 $C=101$ ，由此得 I 的Berger码为0101000101。

Berger码可以检测各种单向多错，且是各种可分离码中需要校验位最少的一种编码方法。如图所示当传输的代码发生错误时，比较器的两个输入将不相同，从而给出错误指示 E 。在该系统中，仅增加了两个校验码发生器和一个比较器。

药片分装自动控制系统



如何
转换?

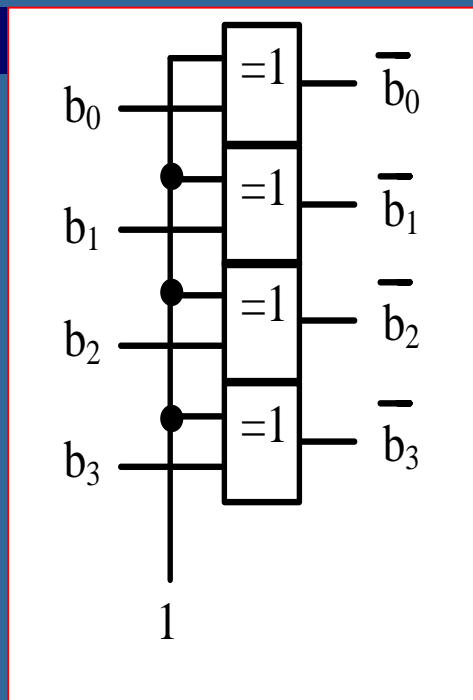
设显示单元最大显示数为50，且这里的BCD码为8421BCD码。则有真值表：

8421BCD 码								自然二进制码					
b_{80}	b_{40}	b_{20}	b_{10}	b_8	b_4	b_2	b_1	a_5	a_4	a_3	a_2	a_1	a_0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	0	1	0	0	0	0	0	1	0
.....												
0	0	0	0	1	0	0	1	0	0	1	0	0	1
0	0	0	1	0	0	0	0	0	0	1	0	1	0
.....												
0	1	0	1	0	0	0	0	1	1	0	0	1	0

习题课

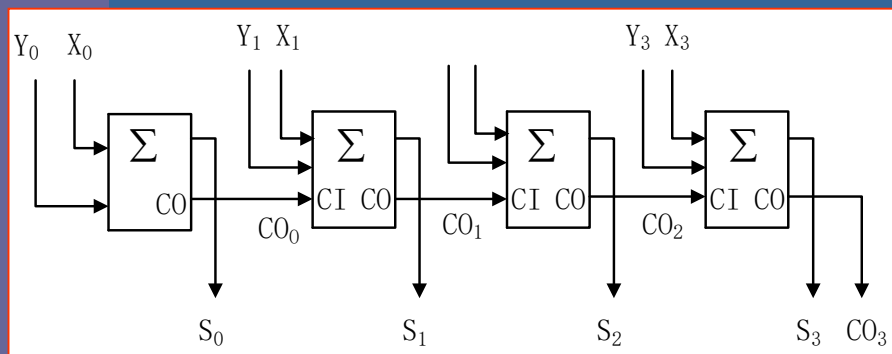
习题1

试用异或门作为一个模块实现一个4位二进制原码到反码的转换电路。



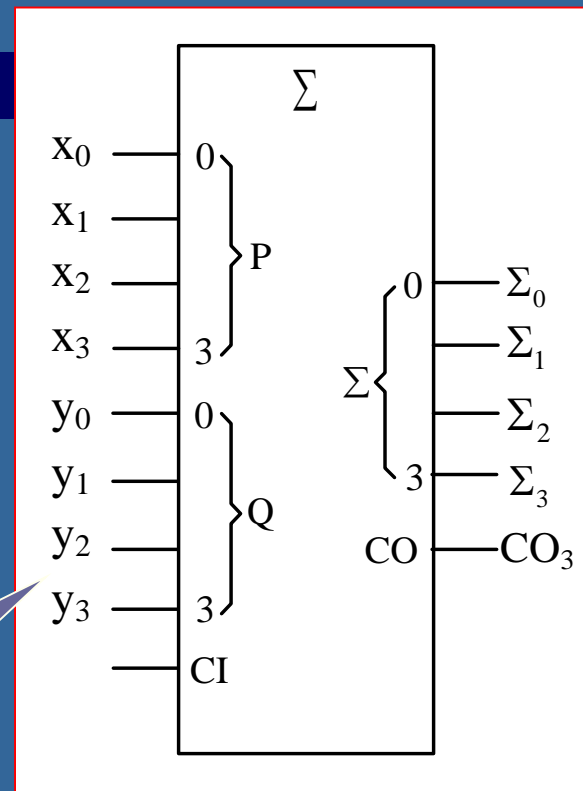
习题2

试用题1中的反码转换器及4位全加器作为模块构成4位二进制原码到补码的转换电路。

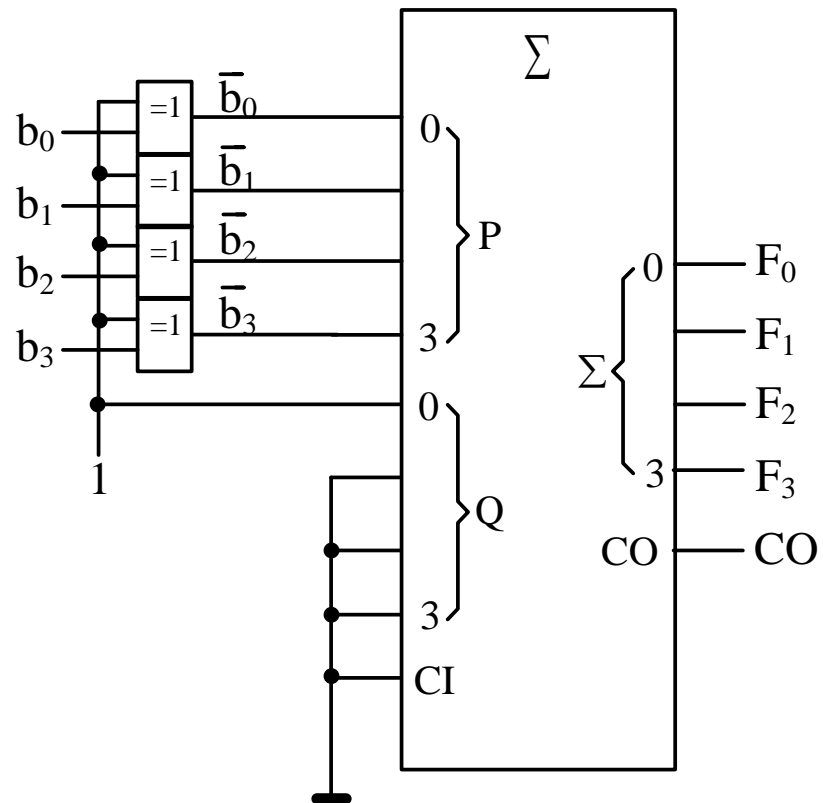


(P9 4位全加器)

4位全加
器逻辑
符号

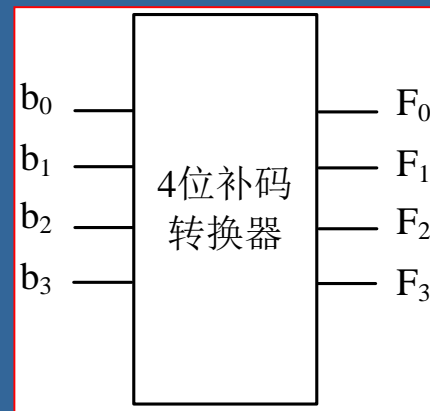
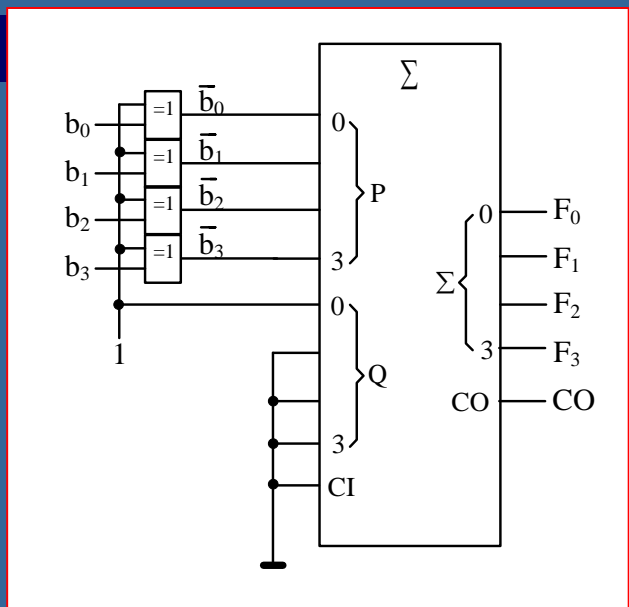


习题2 (续)

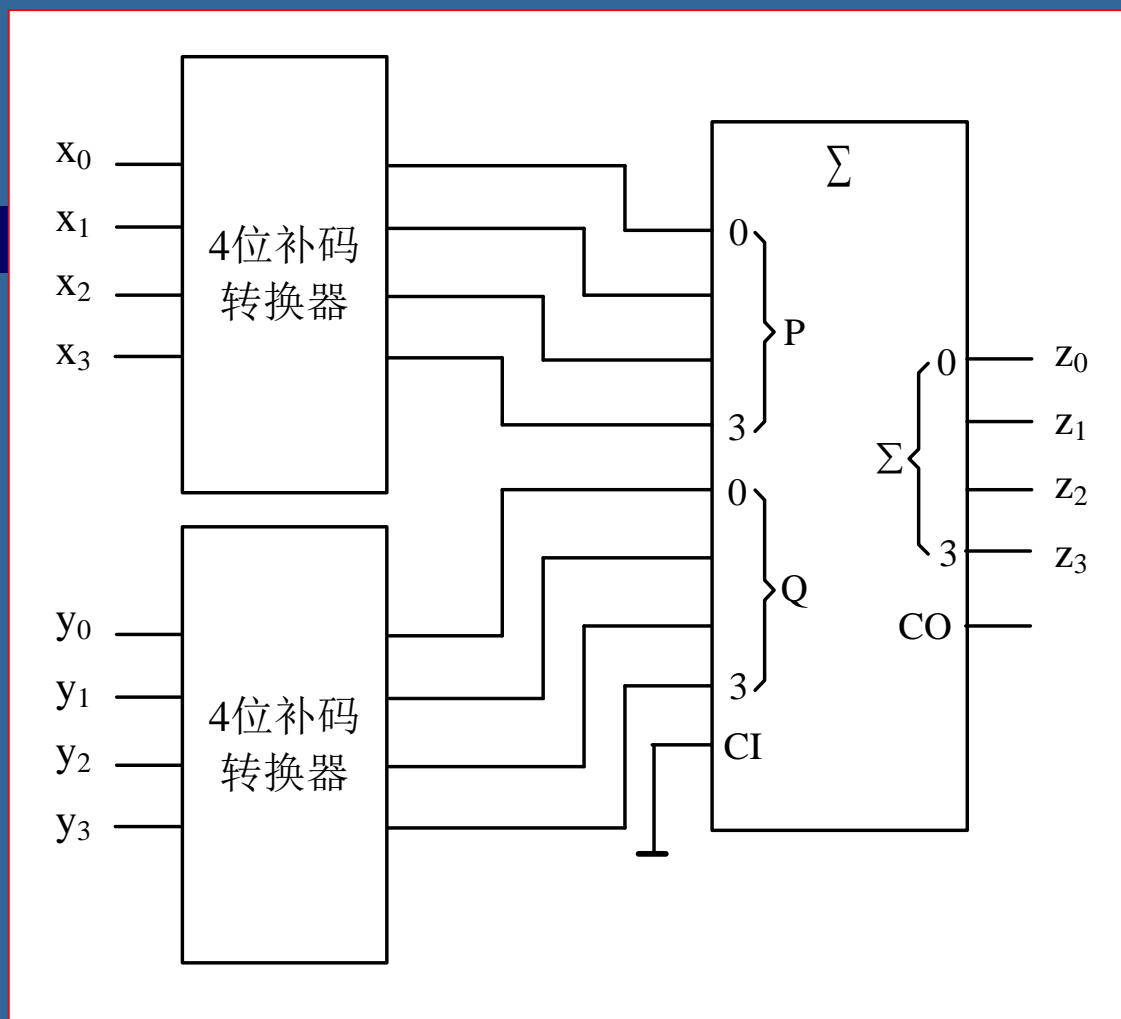


习题3

试用全加器及题2中的补码转换器作为模块构成4位补码运算电路。

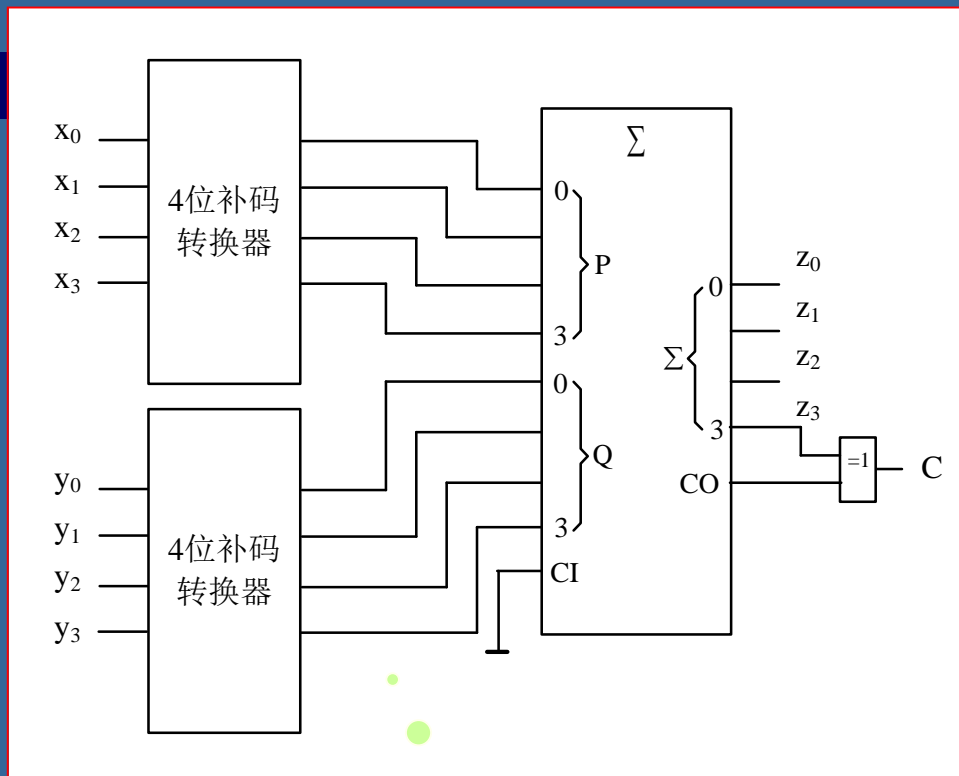


习题3 (续)



习题4

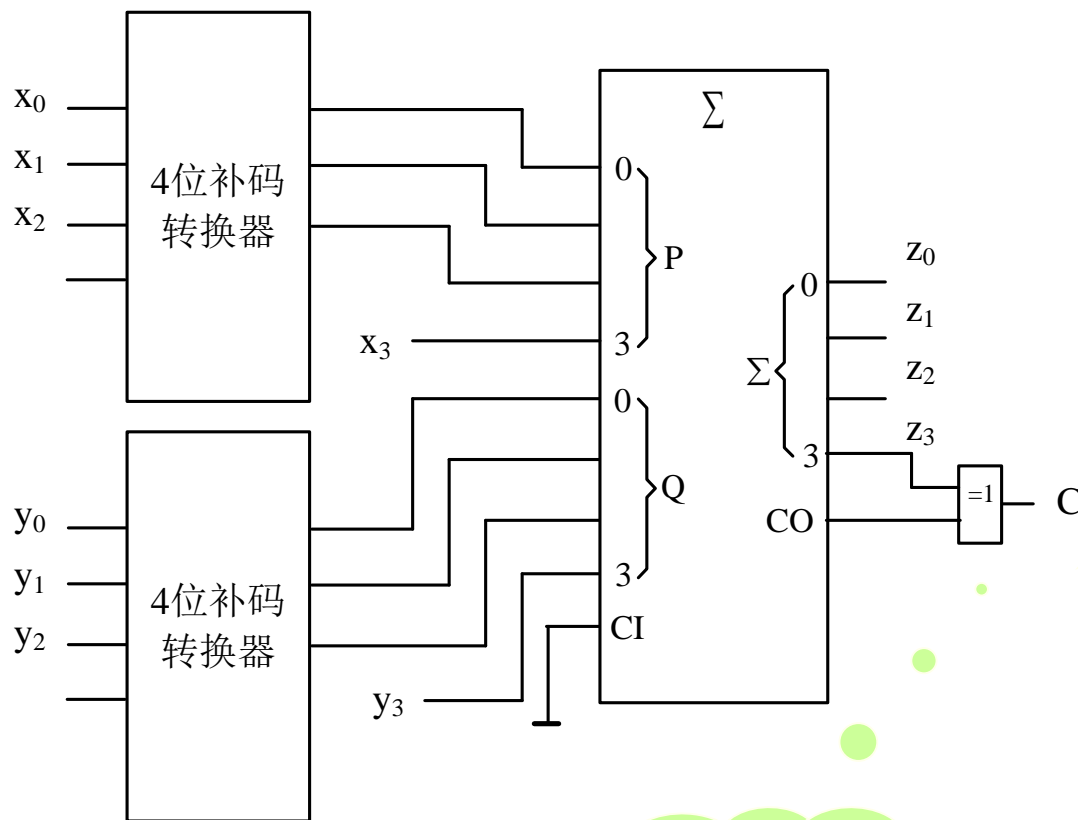
试在题3的基础上加入溢出判别电路构成更为完善的补码运算电路。



思考：本设计正确？

$x_3x_2x_1x_0$ 和 $y_3y_2y_1y_0$ 应是有符号数，在求其补码时，符号位是不变的；但在进行加法运算时需代入。

正数的数值位不变，负数的数值位求反加1



同符号补码相加才可能产生溢出

思考：
还有问题？

4位带符号的补码变换器

