* CODE CONVERSION

* Sometimes it's necessary to use the O/P of one system as the I/P to another.

* A conversion circuit must be inserted between the two system if each uses different codes for the same information.

* So code converter circuit makes two systems compatible even though each uses a different binary code.

Ex-1

Binery to grey code Converter

Binary Code				Grey code			
A	B	C	D	w	x	y	z
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

$w = \Sigma(8, 9, 10, 11, 12, 13, 14, 15)$

$x = \Sigma(4, 5, 6, 7, 8, 9, 10, 11)$

$y = \Sigma(2, 3, 4, 5, 10, 11, 12, 13)$

$z = \Sigma(1, 2, 5, 6, 9, 10, 13, 14)$

W

AB \ CD	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$\boxed{W = A}$

X

AB \ CD	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$X = AB' + A'B$

$\boxed{X = A \oplus B}$

y

AB \ CD	00	01	11	10
00			1	1
01	1	1		
11	1	1		
10			1	1

$y = B'C' + B'C$

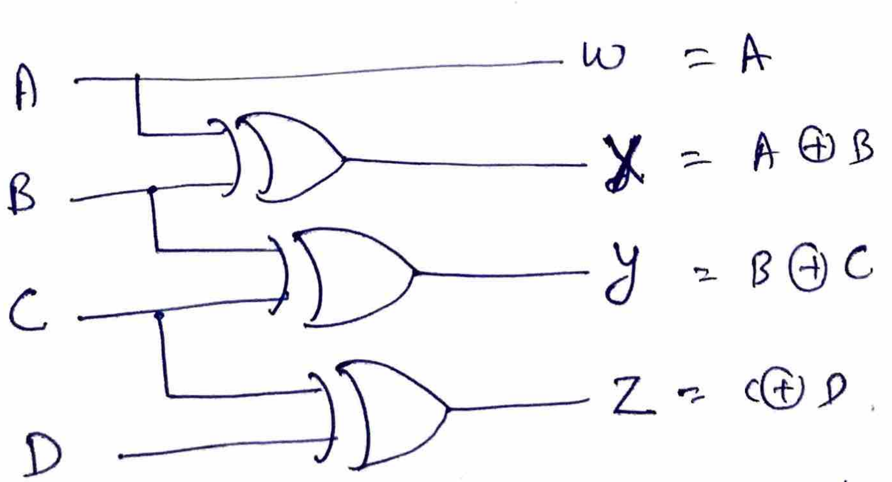
$\boxed{y = B \oplus C}$

Z

AB \ CD	00	01	11	10
00		1		1
01		1		1
11		1		1
10		1		1

$Z = C'D + CD'$

$\boxed{Z = C \oplus D}$



grey code

Binary

Ex-2

grey code to Binary Conversion

grey code				(in Binary)	Binary code			
A	B	C	D		W	X	Y	Z
0	0	0	0	– 0	0	0	0	0
0	0	0	1	– 1	0	0	0	1
0	0	1	1	– 3	0	0	1	0
0	0	1	0	– 2	0	0	1	1
0	1	1	0	– 6	0	1	0	0
0	1	1	1	– 7	0	1	0	1
0	1	0	1	– 5	0	1	1	0
0	1	0	0	– 4	0	1	1	1
1	1	0	0	– 12	1	0	0	0
1	1	0	1	– 13	1	0	0	1
1	1	1	1	– 15	1	0	1	0
1	1	1	0	– 14	1	0	1	1
1	0	1	0	– 10	1	1	0	0
1	0	1	1	– 11	1	1	0	1
1	0	0	1	– 9	1	1	1	0
1	0	0	0	– 8	1	1	1	1

$$W = \Sigma(8, 9, 10, 11, 12, 13, 14, 15)$$

$$X = \Sigma(4, 5, 6, 7, 8, 9, 10, 11)$$

$$Y = \Sigma(2, 3, 4, 5, 8, 9, 14, 15)$$

$$Z = \Sigma(1, 2, 4, 7, 13, 14, 8, 11)$$

w

AB \ CD	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$w = A$

X

AB \ CD	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$X = A'B + AB' = A \oplus B$

y

AB \ CD	00	01	11	10
00			1	1
01	1	1		
11			1	1
10	1	1		

$y = A'B'C + A'BC' + ABC + AB'C'$

$= A'(B'C + BC') + A(BC + B'C')$

$= A'(B \oplus C) + A(B \oplus C)'$

$= A \oplus B \oplus C$

z

AB \ CD	00	01	11	10
00		1		1
01	1		1	
11		1		1
10	1		1	

$z = A'B'C'D + A'B'CD' + A'BC'D' + A'BCD + ABC'D + ABCD' + AB'C'D' + AB'CD$

$= A'B'(C'D + CD') + A'B(C'D' + CD) + AB(C'D + CD') + AB'(C'D' + CD)$

$= A'B'(C \oplus D) + AB(C \oplus D) + A'B(C \oplus D)' + A'B(C \oplus D)'$

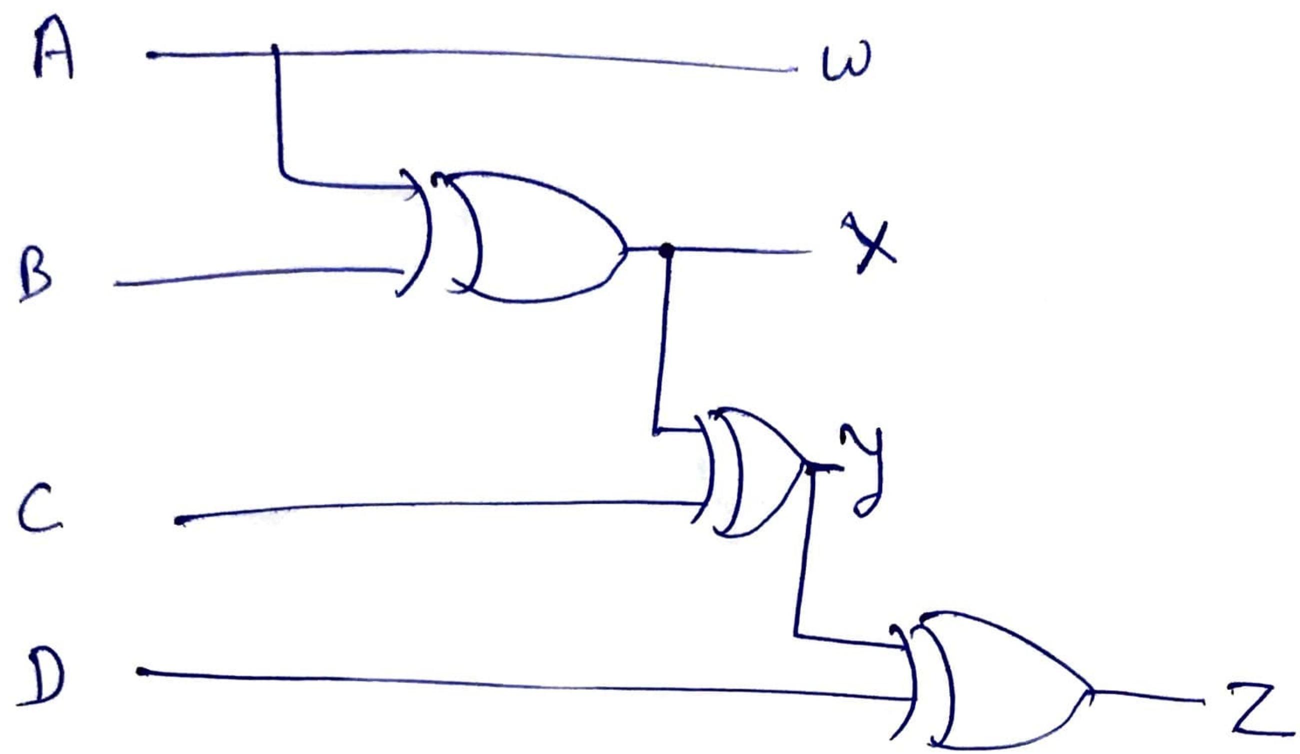
$= A \oplus B \oplus C \oplus D$

$\boxed{w = A}$

$\boxed{X = A \oplus B}$

$\boxed{y = X \oplus C}$

$\boxed{Z = y \oplus D}$



Grey to Binary code converter

Ex-3

84-2-1 to BCD Code converter

	8	4	-2	-1	BCD			
	A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	0	0	0
7	0	1	1	1	0	0	0	1
6	0	1	1	0	0	0	1	0
5	0	1	0	1	0	0	1	1
4	0	1	0	0	0	1	0	0
11	1	0	1	1	0	1	0	1
10	1	0	1	0	0	1	1	0
9	1	0	0	1	0	1	1	1
8	1	0	0	0	1	0	0	0
15	1	1	1	1	1	0	0	1

find don't cares in input

$d = \Sigma(1, 2, 3, 12, 13, 14)$

$W = \Sigma(8, 15)$

$X = \Sigma(4, 11, 10, 9)$

$Y = \Sigma(5, 6, 9, 10)$

$Z = D$

W

AB \ CD	00	01	11	10
00		X	X	X
01				
11	X	X	1	X
10	1			

$\boxed{W = AC'D' + AB}$

AB \ CD	00	01	11	10
00		X	X	X
01	1			
11	X	X		X
10		1	1	1

$\boxed{X = BC'D' + B'D + B'C}$

y

AB \ CD	00	01	11	10
00		X	X	X
01		1		1
11	X	X		X
10		1		1

$y = C'D + CD'$

$\boxed{y = C \oplus D}$

$\boxed{Z = D}$

Ex

BCD to Excess-3 Code Converter

Input BCD A	B	C	D	O/P Excess-3 w	x	y	z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

AB \ CD	00	01	11	10
00				
01		1	1	1
11	X	X	X	X
10	1	1	X	X

$$\boxed{W = A + BC + BD}$$

AB \ CD	00	01	11	10
00		1	1	1
01	1			
11	X	X	X	X
10		1	X	X

$$\boxed{X = B'D + B'C + BC'D'}$$

AB \ CD	00	01	11	10
00	1		1	
01	1		1	
11	X	X	X	X
10	1		X	X

$$\boxed{Y = C'D' + CD}$$

$$\boxed{Z = D'}$$

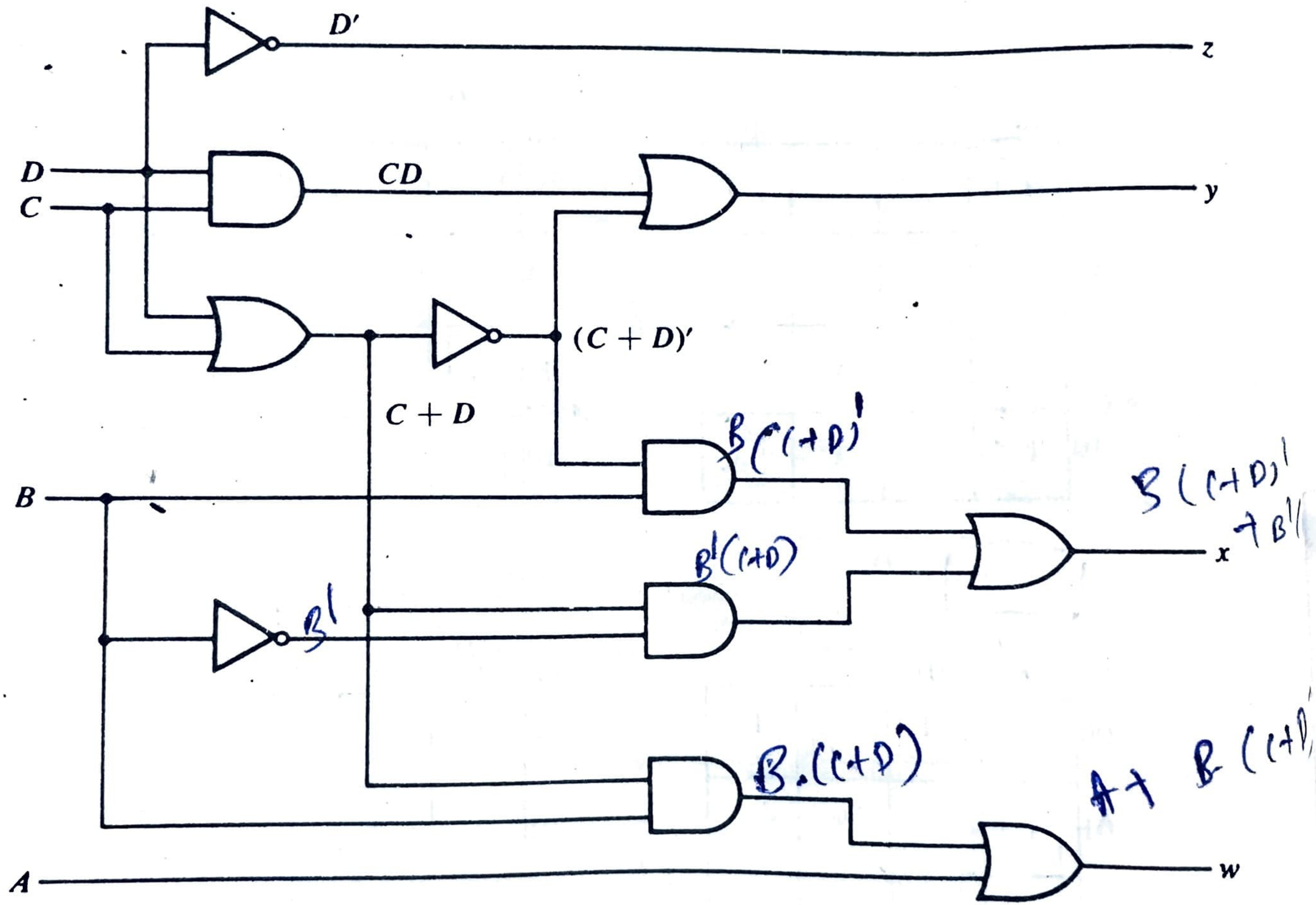


Figure 4-8 Logic diagram for BCD-to-excess-3 code converter

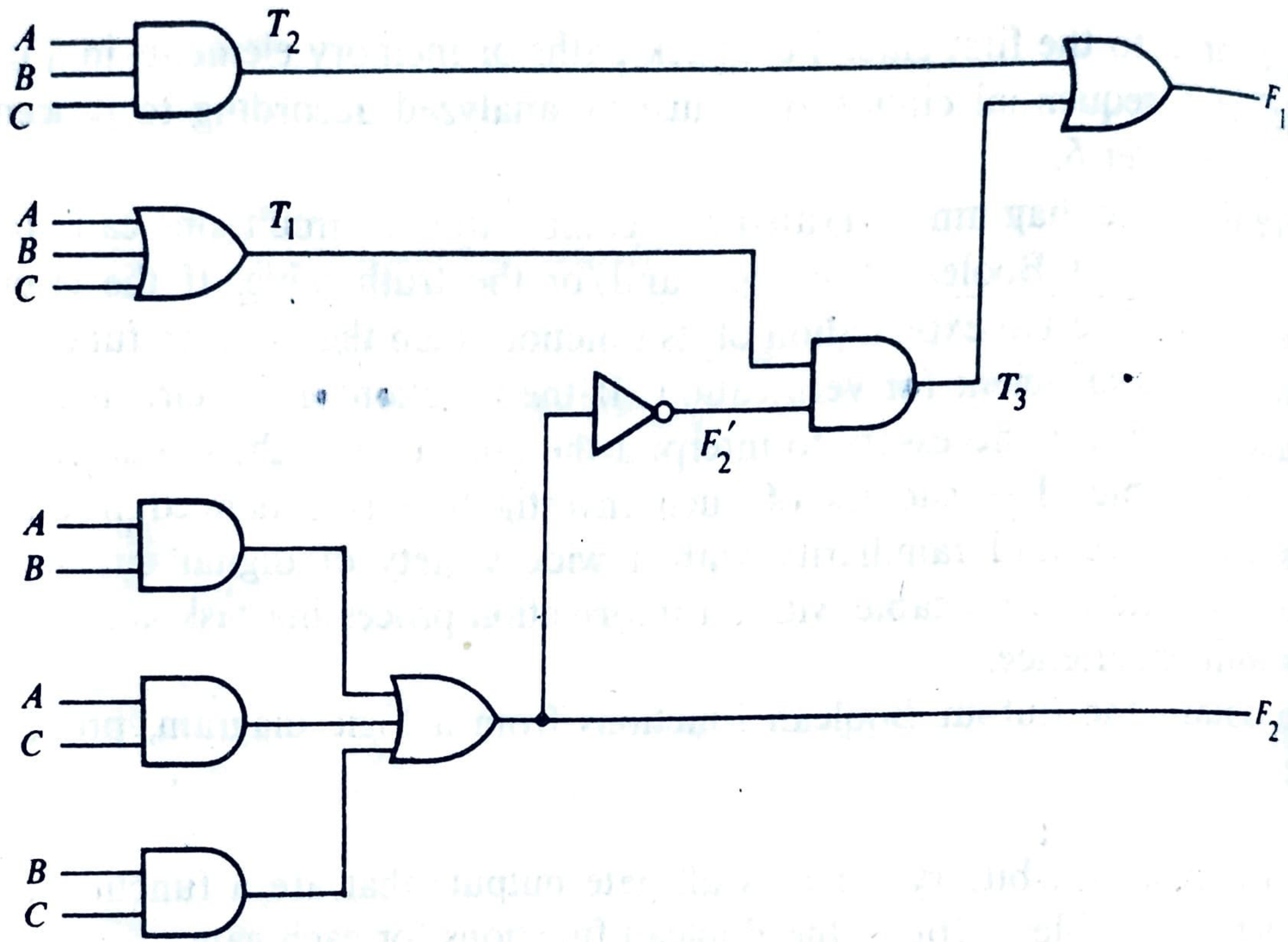


Figure 4-9 Logic diagram for analysis example

substitutions as follows:

$$F_1 = T_3 + T_2 = F_2'T_1 + ABC = (AB + AC + BC)'(A + B + C) + ABC$$
$$= (A' + B')(A' + C')(B' + C')(A + B + C) + ABC$$
$$= (A' + B'C')(AB' + AC' + BC' + B'C) + ABC$$
$$= A'BC' + A'B'C + AB'C' + ABC$$

TABLE 4-2 Truth table for logic diagram of Fig. 4-9

A	B	C	F_2	F_2'	T_1	T_2	T_3	F_1
0	0	0	0	1	0	0	0	0
0	0	1	0	1	1	0	1	1
0	1	0	0	1	1	0	1	1
0	1	1	1	0	1	0	0	0
1	0	0	0	1	1	0	1	1
1	0	1	1	0	1	0	0	0
1	1	0	1	0	1	0	0	0
1	1	1	1	0	1	1	0	1

Ex Design a combinational circuit with four i/p lines that represent a decimal digit in BCD & four o/p lines that generates the 9's complement of the i/p digits.

Input ABCD	output wxyz
0000	1001
0001	1000
0010	0111
0011	0110
0100	0101
0101	0100
0110	0011
0111	0010
1000	0001
1001	0000

dont cares = $\Sigma(10, 11, 12, 13, 14, 15)$

AB \ CD	00	01	11	10
00	1	1		
01				
11	X	X	X	X
10			X	X

$\boxed{w = A'B'C'}$

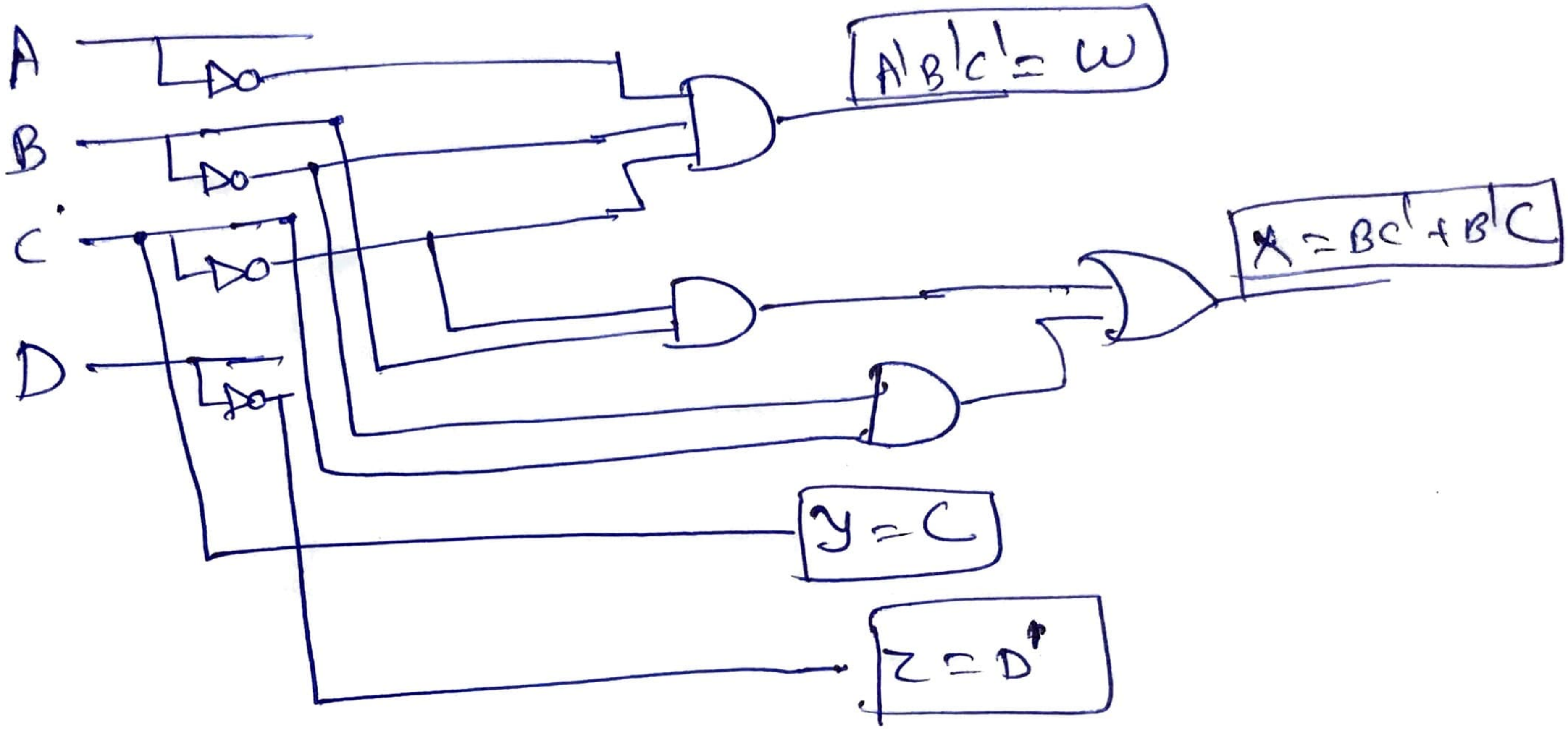
AB \ CD	00	01	11	10
00			1	1
01	1	1		
11	X	X	X	X
10			X	X

$\boxed{x = B'C + BC'}$

AB \ CD	00	01	11	10
00			1	1
01			1	1
11	X	X	X	X
10			X	X

$\boxed{y = C}$

$\boxed{Z = D'}$



logic diagram.

Ex Design a combinational ckt that accepts a three-bit number & generates an o/p binary number equal to the square of the i/p number.

Input A B C	Output X_6 X_5 X_4 X_3 X_2 X_1	
0 0 0	0 0 0 0 0 0	(0)
0 0 1	0 0 0 0 0 1	(1)
0 1 0	0 0 0 1 0 0	(4)
0 1 1	0 0 1 0 0 1	(9)
1 0 0	0 1 0 0 0 0	(16)
1 0 1	0 1 1 0 0 1	(25)
1 1 0	1 0 0 1 0 0	(36)
1 1 1	1 1 0 0 0 1	(49)

$\boxed{X_1 = C}$

$\boxed{X_2 = 0}$

$\boxed{X_3 = BC'}$

$\boxed{X_4 = C.(A \oplus B)}$

$\boxed{X_5 = A(B' + C)}$

$\boxed{X_6 = AB}$

A \ BC	00	01	11	10
0				1
1				1

$\boxed{X_3 = BC'}$

A \ BC	00	01	11	10
0			1	
1		1		

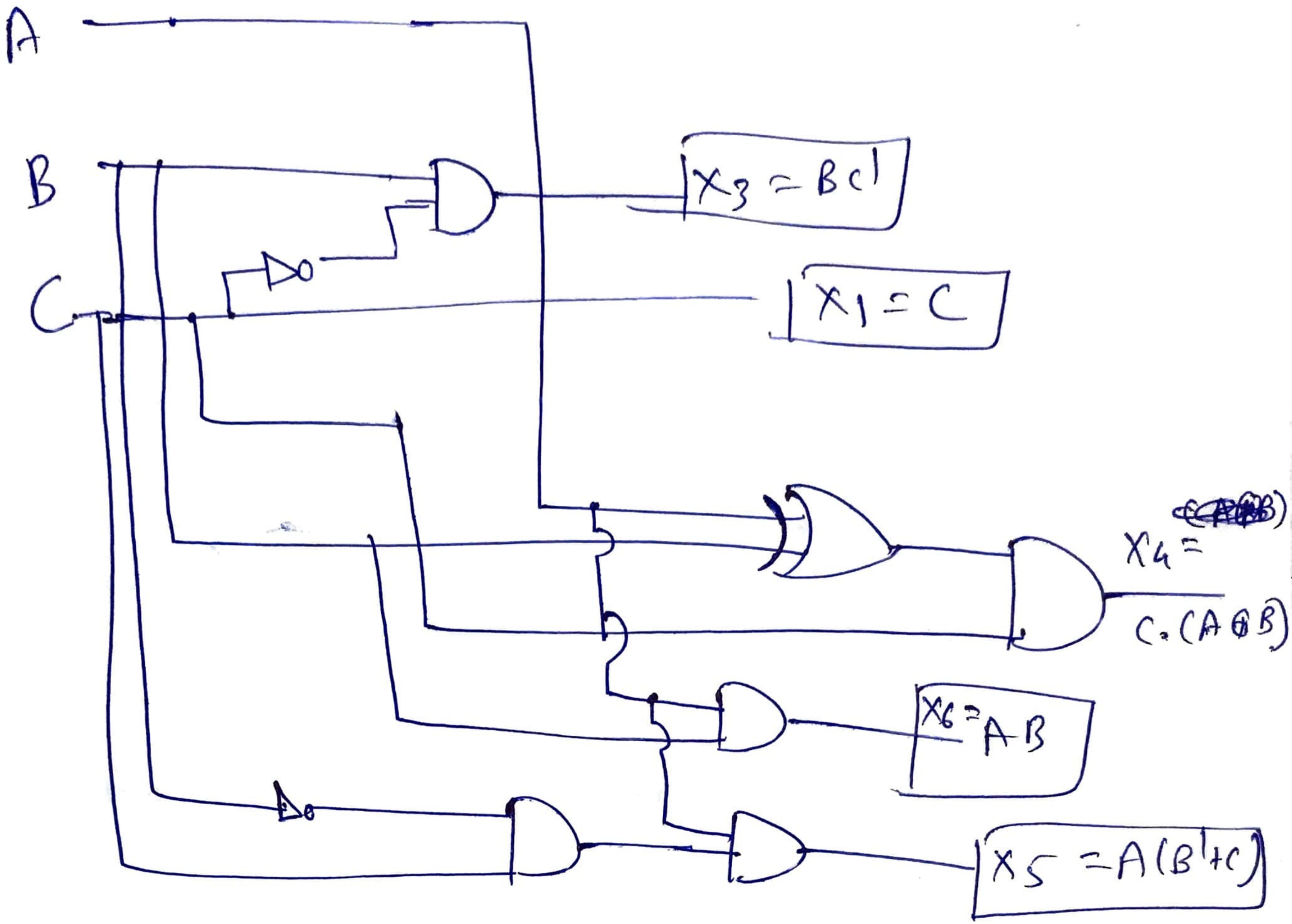
$$\boxed{X_4 = A'BC + AB'C \\ C(A \oplus B)}$$

A \ BC	00	01	11	10
0				
1			1	1

$\boxed{X_6 = AB}$

A \ BC	00	01	11	10
0				
1	1	1	1	

$\boxed{X_5 = A \cdot B' + AC}$


A
B
C
X3 = BC'
X1 = C
X4 =
C.(A⊕B)
X6 = A.B
X5 = A(B'+C)

Ex A Comb^n ckt has 4 i/p & 1 o/p.

o/p is equal to 1 when

(a) all i/p are 1

(b) none i/p are 1

(3) an odd no. of i/p are 1

- Design ckt

A B C D	X
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	0
0 1 1 0	0
0 1 1 1	1
1 0 0 0	1
1 0 0 1	0
1 0 1 0	0
1 0 1 1	1
1 1 0 0	0
1 1 0 1	1
1 1 1 0	1
1 1 1 1	1

AB \ CD	00	01	11	10
00	1	1		1
01	1		1	
11		1	1	1
10	1		1	

$$X = A'B'C' + A'B'D' + A'C'D' + B'C'D' + ABD + ABC + BCD + ACD$$

Ex: multiply two bit binary no & O/P is also binary.

two no.s are $a_1 a_0$ & $b_1 b_0$

— Design a ckt

$a_1a_0 \backslash b_1b_0$	00	01	11	10
00	0000	0000	0000	0000
01	0000	0001	0011	0010
11	0000	0011	1001	0110
10	0000	0010	0110	0100

four O/Ps $w\ x\ y\ z$

$$w = \Sigma(15) = a_1 a_0 b_1 b_0$$

$$x = \Sigma(10, 11, 14) = a_1 a_0' b_1 + a_1 b_1 b_0'$$

$$y = \Sigma(6, 7, 9, 11, 13, 14) = a_1 b_0 b_1' + a_0 a_1' b_1 + a_0 b_0' b_1 + a_0' a_1 b_0$$

$$z = \Sigma(5, 7, 13, 15) = a_0 b_0$$

Ex

Design a Comb^n ckt whose I/P is a four-bit no & whose O/P is 2's ~~comM~~ Complement of the i/p no.

~~A B C D~~

P Q R S	A B C D
0 0 0 0	0 0 0 0
0 0 0 1	1 1 1 1
0 0 1 0	1 1 1 0
0 0 1 1	1 1 0 1
0 1 0 0	1 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 0 1 0
0 1 1 1	1 0 0 1
1 0 0 0	1 0 0 0
1 0 0 1	0 1 1 1
1 0 1 0	0 1 1 0
1 0 1 1	0 1 0 1
1 1 0 0	0 1 0 0
1 1 0 1	0 0 1 1
1 1 1 0	0 0 1 0
1 1 1 1	0 0 0 1

[illegible]

$A = \Sigma(1,2,3,4,5,6,7,8)$

$B = \Sigma(1,2,3,4,9,10,11,12)$

$C = \Sigma(1,2,5,6,9,10,13,14)$

$\boxed{D = S}$

$A = A'(B+C+D) + AB'C'D'$

$B = (C+D)B' + BC'D'$

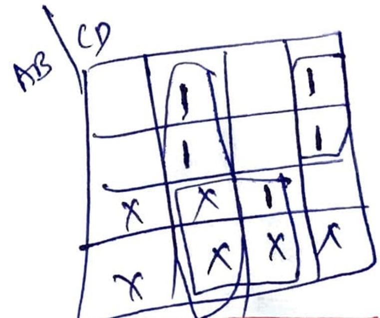
$C = CD' + C'D$

Ex Design a comb^n ckt that multiply by 5 an i/p decimal digit represented in BCD. O/P is Also in BCD.

A	B	C	D	y_4	y_3	y_2	y_1	x_4	x_3	x_2	x_1	
0	0	0	0	0	0	0	0	0	0	0	0	(0)
0	0	0	1	0	0	0	0	0	1	0	1	(5)
0	0	1	0									
0	0	1	1									
0	1	0	0									
0	1	0	1									
0	1	1	0									
0	1	1	1	0	1	0	0	0	0	0	0	(40)
1	0	0	0	0	1	0	0	0	1	0	1	(45)
1	0	0	1									

Ex Design ckt that converts from 2,4,2,1 code to 8,4,−2,−1 code

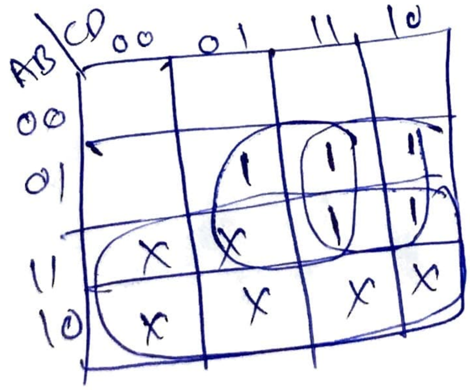
2	4	2	1	8	4	−2	−1
A	B	C	D	w	x	y	z
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	1
0	0	1	0	0	1	1	0
0	0	1	1	0	1	0	1
0	1	0	0	0	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	1	1	0	1	0	0	0
1	1	1	1	1	1	1	1



$\boxed{y = C'D + AD + A'CD'}$

don't care = $\Sigma(8, 9, 10, 11, 12, 13)$

w



$\boxed{w = A + BC + BD}$

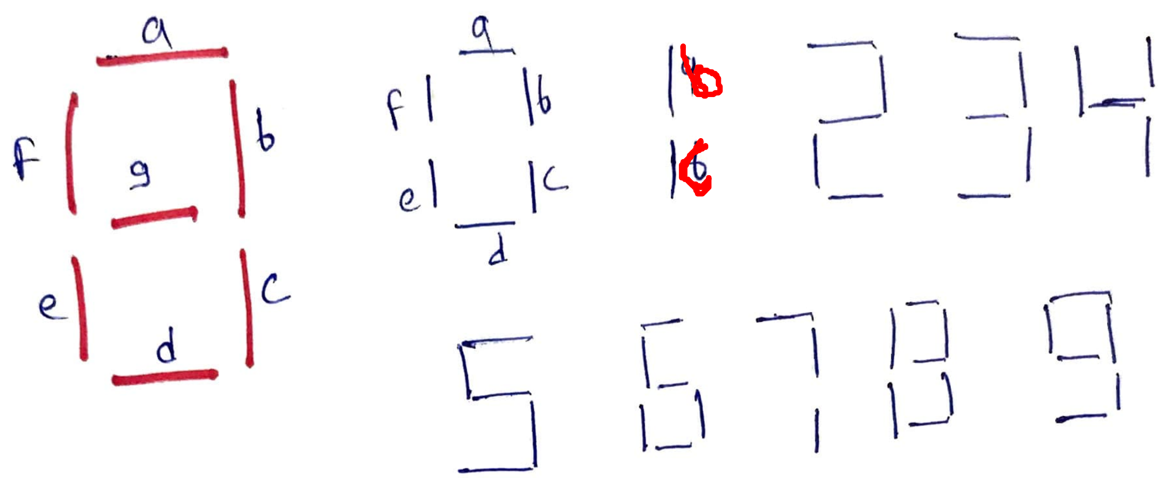
x

$\boxed{x = B'C + B'D + BC'D' + AD}$

$\boxed{z = D}$

Ex Convert (4-digit) binary to BCD

Ex



Design BCD to seven segmet decoder

A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1