ASSIGNMENT - 2

1) Design a combinational circuit with four input lines that represent a decimal digit in BCD and four output lines that generate 9's complement of the input digit

In terms of decimal digits, the truth table will be as follows

Input	Output
0	9
1	8
2	7
3	6
4	5
5	4
6	3
7	2
8	1
9	0

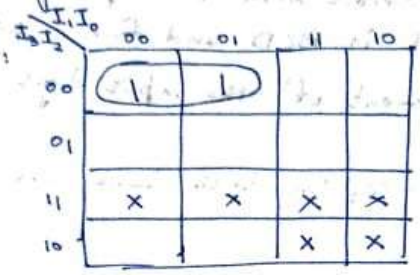
In terms of binary:

Input				Output			
I_3	I_2	I_1	I_0	O_3	O_2	O_1	O_0
0	0	0	0	1	0	0	1
0	0	0	1	1	0	0	0
0	0	1	0	0	1	1	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	0	0
0	1	1	0	0	0	1	1
0	1	1	1	0	0	1	0
1	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0

→ all other input treated as "don't care"

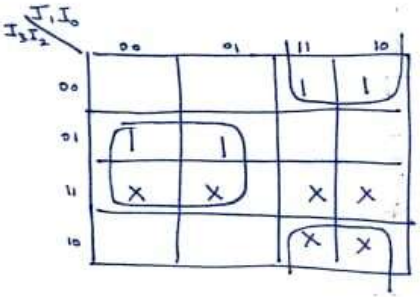
→ drawing K-Maps of the output

I) O_3 :



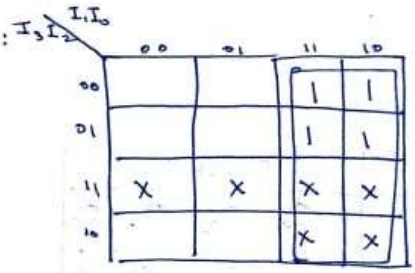
$O_3 = I_3' I_2' I_1'$

II) O_2 :



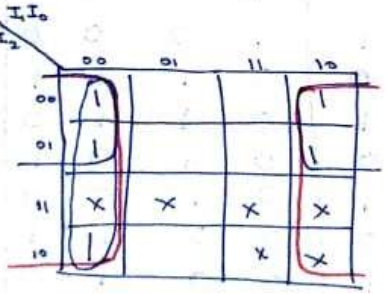
$O_2 = I_2 I_1' + I_2' I_1$
$= I_1 \text{ XOR } I_2$

III) O_1 :

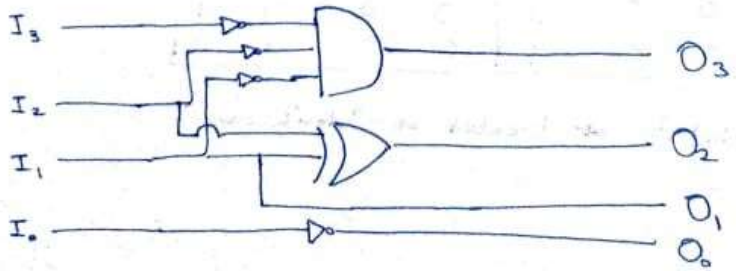


$O_1 = I_1$

IV) O_0 :

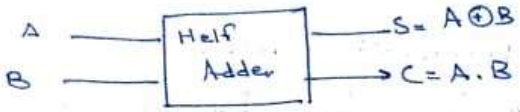


$O_0 = I_3' I_0' + I_1' I_0'$
$= I_0'$



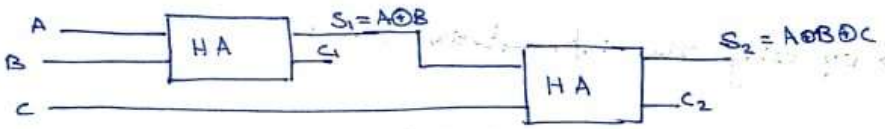
2) Implement the following boolean function using three half adder circuits.

~~i) D = A ⊕ B ⊕ C~~



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

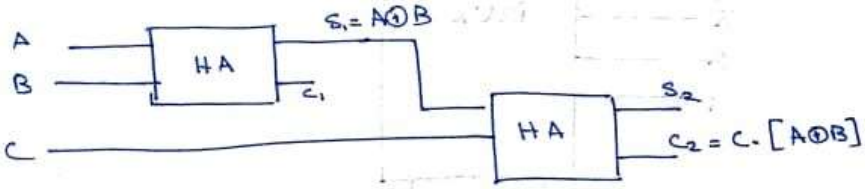
$= [A \oplus B] \oplus C$



ii) $E = \bar{A}BC + A\bar{B}C$

$= C.[\bar{A}B + A\bar{B}]$

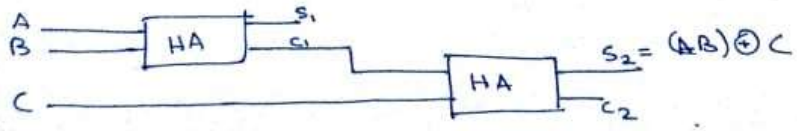
$= C.[A \oplus B]$



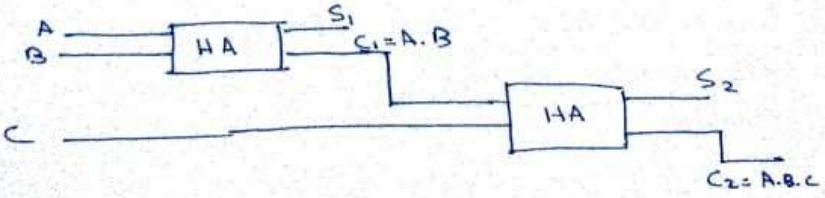
iii) $F = AB\bar{C} + C(\bar{A}+\bar{B})C$

$= AB\bar{C} + \overline{AB}\,C$

$= (AB) \oplus C$



iv) $G_1 = ABC = (AB).C$



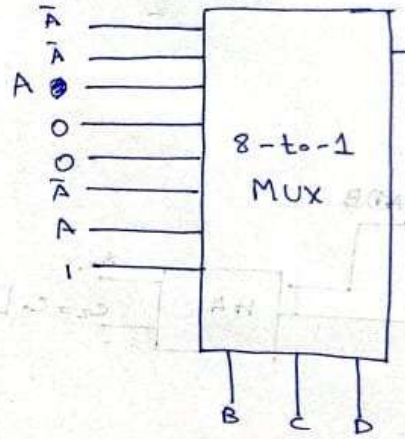
Q3) Implement $f(A,B,C,D) = \Sigma(0, 1, 5, 7, 10, 14, 15)$ using appropriate multiplexer.

→ Let's use 2^{4-1} - to - 1 Multiplexer

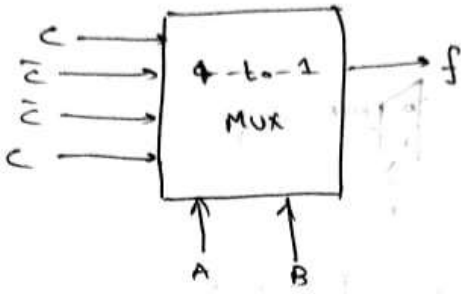
→ Drawing Implementation Table.

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{A}$	(0)	(1)	2	3	4	(5)	6	(7)
A	8	9	(10)	11	12	13	(14)	(5)
	$\bar{A}$	$\bar{A}$	$\emptyset A$	0	0	$\bar{A}$	A	1

→ by using the implementation table:



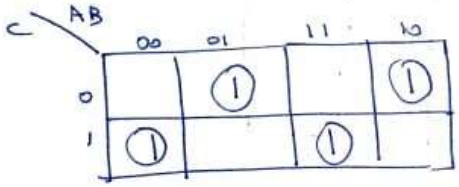
4) Which boolean Function is implemented by the following Multiplexer.



drawing reversed implementation table

	I_0	I_1	I_2	I_3
$\bar{C}$	0	(1)	(2)	3
C	(4)	5	6	(7)
	C	$\bar{C}$	$\bar{C}$	C

$\therefore f(C,A,B) = \Sigma(1,2,4,7)$

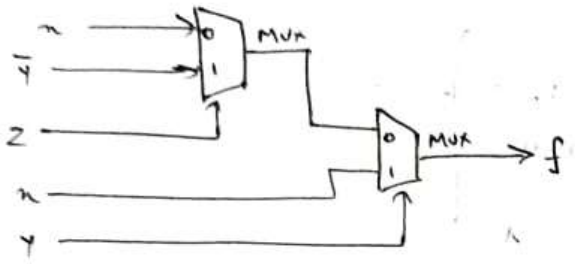


$\therefore f(C,A,B) = C'A'B + C'AB' + CA'B' + CAB$

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

$= C'(A \oplus B \text{ XOR } B) + C(A \text{ XNOR } B)$

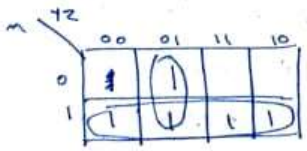
5) Which Function is realized by the circuit given below



using the circuit to make Truth Table

Input			output
x	y	z	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

* using K-Map



$f(x,y,z) = x + y'z$

Q6) A combinational circuit is defined by the following three functions:

$F1 = \bar{x}\bar{y} + xy\bar{z}$

$F2 = \bar{x} + y$

$F3 = xy + \bar{x}\bar{y}$

Design the circuit with decoder and external gates.

→ We know that for a 3-8 decoder with x, y, z as inputs.

$D_0 = x'y'z'$

$D_1 = x'y'z$

$D_2 = x'yz'$

$D_3 = x'yz$

$D_4 = xy'z'$

$D_5 = xy'z$

$D_6 = xyz'$

$D_7 = xyz$

solving F1:

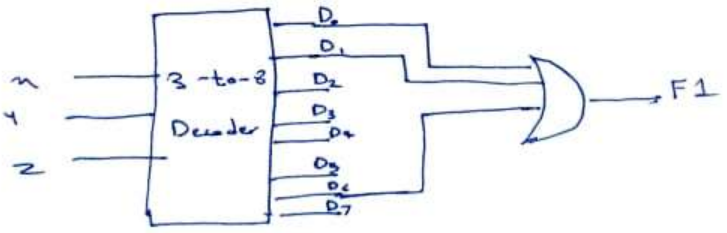
$F1 = \bar{x}\bar{y} + xy\bar{z}$

$= \bar{x}\bar{y}(z + \bar{z}) + xy\bar{z}$ $\quad [1 = a + \bar{a}]$

$= \bar{x}\bar{y}z + \bar{x}\bar{y}\bar{z} + xy\bar{z}$

$= \bar{x}\bar{y}\bar{z} + \bar{x}\bar{y}z + xy\bar{z}$

$= D_0 + D_1 + D_6$



→ we also know for a 2-to-4 decoder with input x and y, that

$D_0 = x'y'$

$D_1 = x'y$

$D_2 = xy'$

$D_3 = xy$

Solving F2:

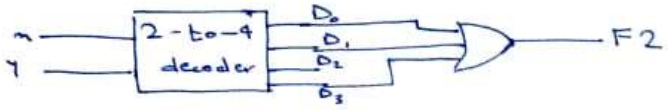
$F2 = \bar{x} + y$

$= \bar{x}(y + \bar{y}) + y(x + \bar{x})$ $\quad [a + \bar{a} = 1]$

$= \bar{x}y + \bar{x}\bar{y} + xy + \bar{x}y$

$= \bar{x}\bar{y} + \bar{x}y + xy$ $\quad [\alpha + \alpha = \alpha]$

$= D_0 + D_1 + D_3$

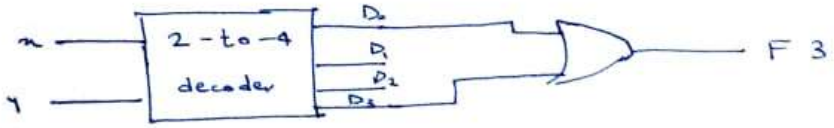


now solving F3:

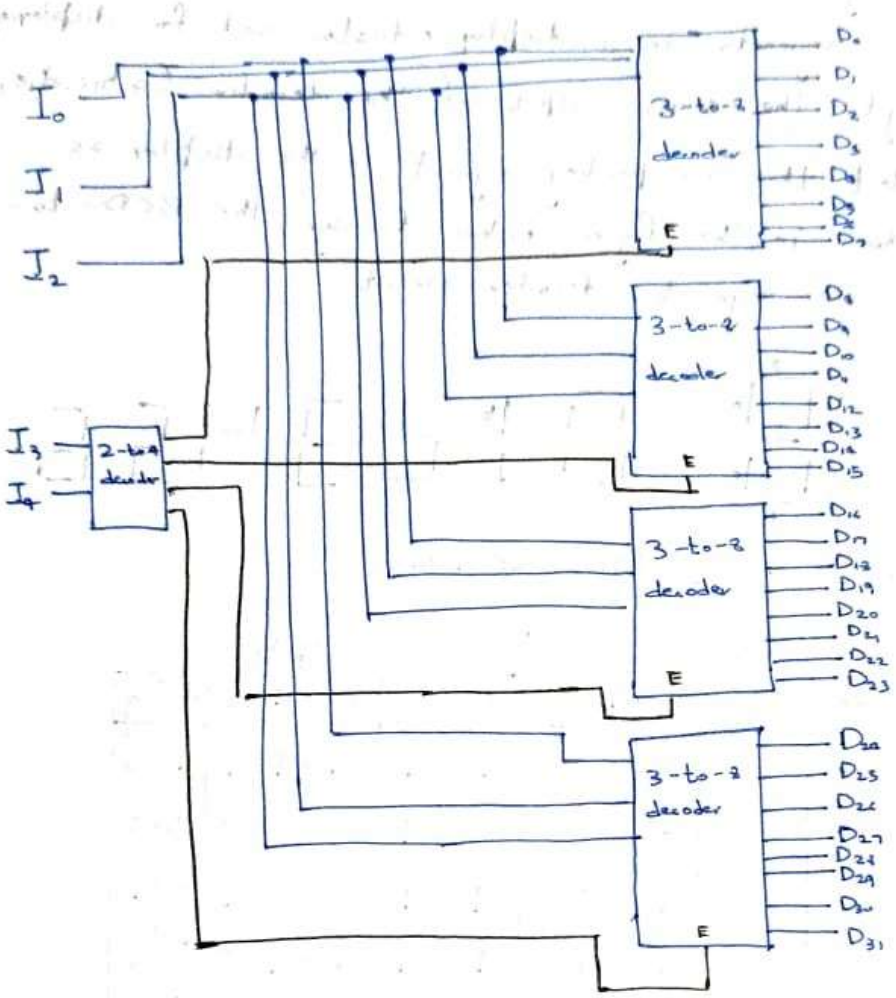
$F3 = xy + \bar{x}\bar{y}$

$= \bar{x}\bar{y} + xy$

$= D_0 + D_3$



7) Construct a 5-to-32 decoder with four 3-to-8 decoder or demultiplexers and a 2-to-4 decoder. Use a block diagram construction.



8) A BCD-to-seven-segment decoder is a combinational circuit that accepts a decimal digit in BCD and generates the appropriate outputs for the selection of segments in a display indicator used for displaying digit. The seven output of the decoder (a,b,c,d,e,f,g) select the corresponding segment in the display as shown in the figure below. Design the BCD-to-seven-segment decoder circuit

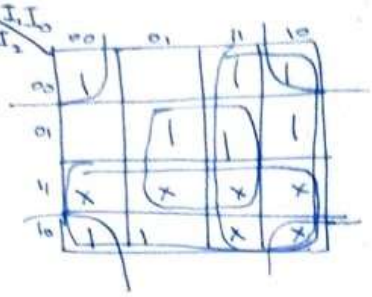
→ from the figure, making Truth Table.

No.	Input				Outputs						
	I_3	I_2	I_1	I_0	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

→ all other inputs are "don't care"

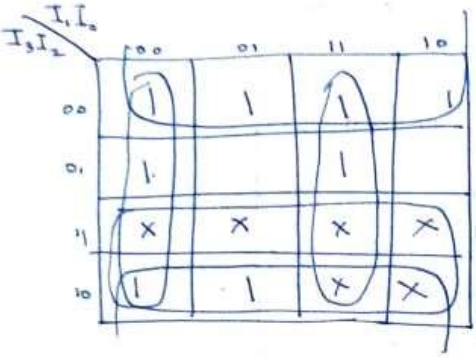
Simplifying using K-Map

I) a



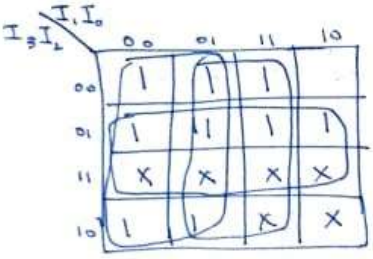
$a = I_3 + I_1 + I_2' I_0' + I_2 I_0$

II) b



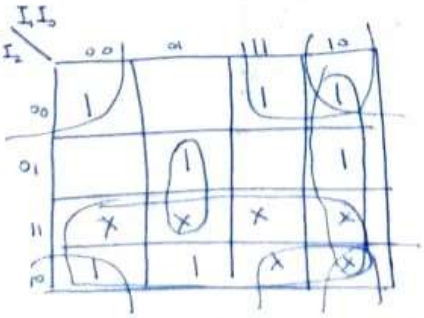
$b = I_3 + I_2' + I_1 I_0 + I_1' I_0'$

III) c



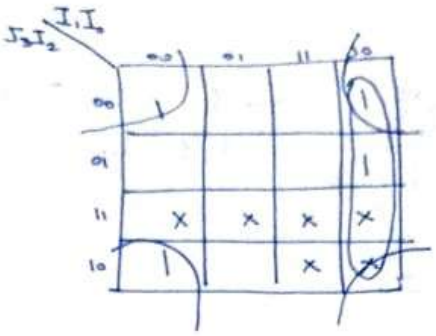
$c = I_1' + I_0 + I_2$

IV) d



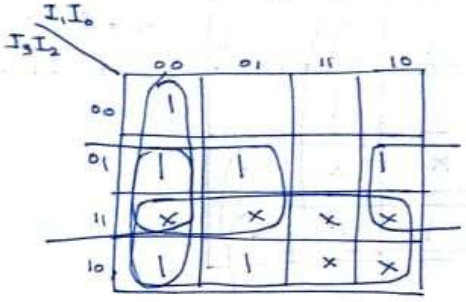
$d = I_3 + I_1 I_0' + I_2' I_1 + I_2' I_0' + I_2 I_1' I_0$

V) e



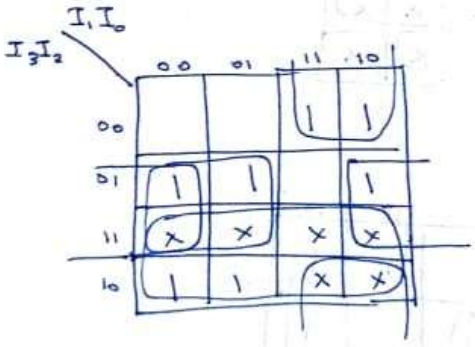
$e = I_2'I_0' + I_1I_0'$

VI) f



$f = I_3 + I_1'I_0' + I_2I_1' + I_2I_0'$

VII) g



$g = I_3 + I_2I_1' + I_2'I_1 + I_2I_0'$