CMR INSTITUTE OF TECHNOLOGY	USN										

Internal Assesment Test - 1

Sub:	Digital System Design using Verilog							Code:	BEC302
Date:	06/11/2024	Duration:	90 mins	Max Marks:	50	Sem:	III	Branch:	ECE

Answer any Five Questions

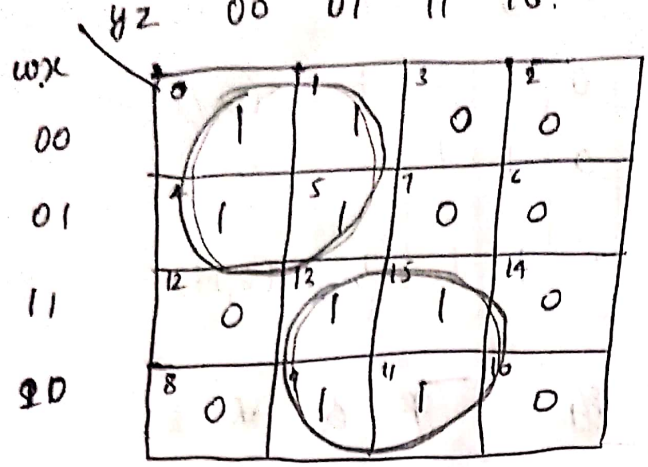
		Marks	CO	RBT
1.	a) Solve the function K=f(w,x,y,z) = ∑(0,1,4,5,9,11,13,15) using K-map b) Simplify the function F= f (P, Q, R, S) =∑ (0, 3, 5, 6, 7, 11,14) using QM method.	[10]	CO1	L2
2.	a) Define encoder. Write the truth table, equations and circuit diagram of 8- to-3-line priority encoder b) Implement the function f(w,x,y,z) = ∑(0, 1, 5, 6, 7, 9, 12, 15) using 8-to-1 line multiplexer	[10]	CO2	L3
3.	a) Define decoder. Write the truth table, equations and circuit diagram of 3- to-8-line decoder b) Explain the operation of 8-to-1 line multiplexer with block diagram truth table, and equations	[10]	CO2	L2
4.	a) Construct parallel binary adder/ subtractor using full adder block and EX-OR gates. Also explain the operation of it. b) Design two- bit comparator using cascade connection of one bit comparators and explain its operation.	[10]	CO2	L2
5.	Explain different types of PLD's with example	[10]	CO2	L2
6.	With neat diagram and truth table, explain the operation of S R & JK flip-flop	[10]	CO3	L2
7	With diagram, wave forms & truth table explain the working of MS-JK flip-flop	[10]	CO3	L2
8	Convert following Boolean function into: i) f(abc)= (a' + b) (b +c') Min term canonical form, ii) f(xyz)= x + x'z'(y + z') Max term canonical form	[10]	CO1	L3

CI CCI HOD

CMR

IAT - 1

1) a) $K = f(w,x,y,z) = \Sigma(0,1,4,5,9,11,13,15)$.



The equation after solving the k-map.

$$y = \bar{w}\bar{y} + wz.$$

$\therefore y = \bar{w}\bar{y} + wz$

b) $F = f(P,Q,R,S) = \Sigma(0,3,5,6,7,11,14)$ by QM method.

Given that minterms function

$F = f(P,Q,R,S) = \Sigma(0,3,5,6,7,11,14)$.

8 4 2 1

(i) Stage 1

P	Q	R	S	No. of 1's
0	0	0	0	0
0	0	1	1	3
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	1	1	11
1	1	1	0	14

(ii) Stage 2

P	Q	R	S	I/P
0	-	1	1	(3,7)
-	0	1	1	(3,11)
0	1	-	1	(5,7)
0	1	1	-	(6,7)
-	1	1	0	(6,14)

(iii) Stage 3 ~~I/P P Q R S~~

~~(3,11,6,14)~~

	0	3	5	6	7	11	14
0 (p'q'r's')	⊗						
(3,7) p'r's		x			x		
(3,11) q'r's		x				⊗	
(5,7) p'qs			⊗		x		
(6,7) p'qr				x	x		
(6,14) qrs'				x			⊗

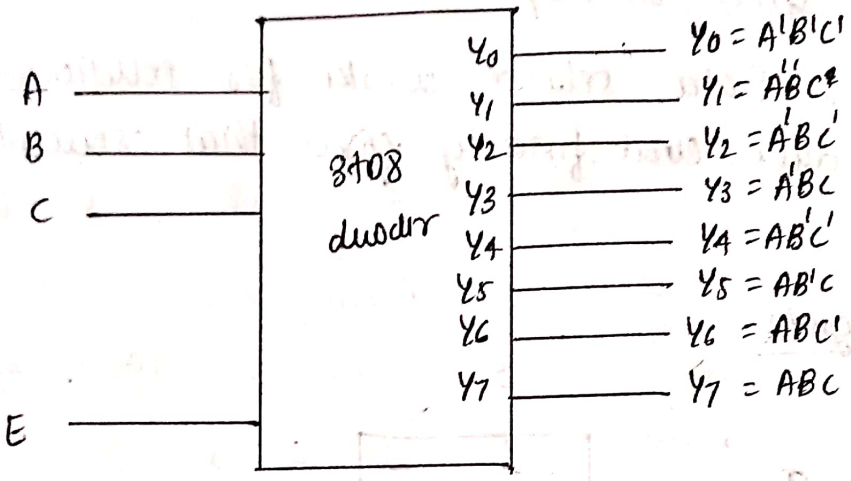
∴ essential prime implicants : 0, 5, 11, 14

The equation : $y = p'q'r's' + q'rs + p'qs + qrs'$

3) (a) decoder :

It is a combinational logic circuit where the decoder takes n inputs and gives 2^n outputs and there input is selected a selector ; the operation takes place when this selector is high.

Truth table : 3-to-8 decoder



Diagram

E	A	B	C	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	X	X	X	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	1	0	0	0	0	0
1	0	1	1	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	0	0	1	0	0
1	1	1	0	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	0	1

~~$Y = A'B'C' + A'B'C + A'BC' + A'BC + AB'C' + AB'C + ABC' + ABC$~~

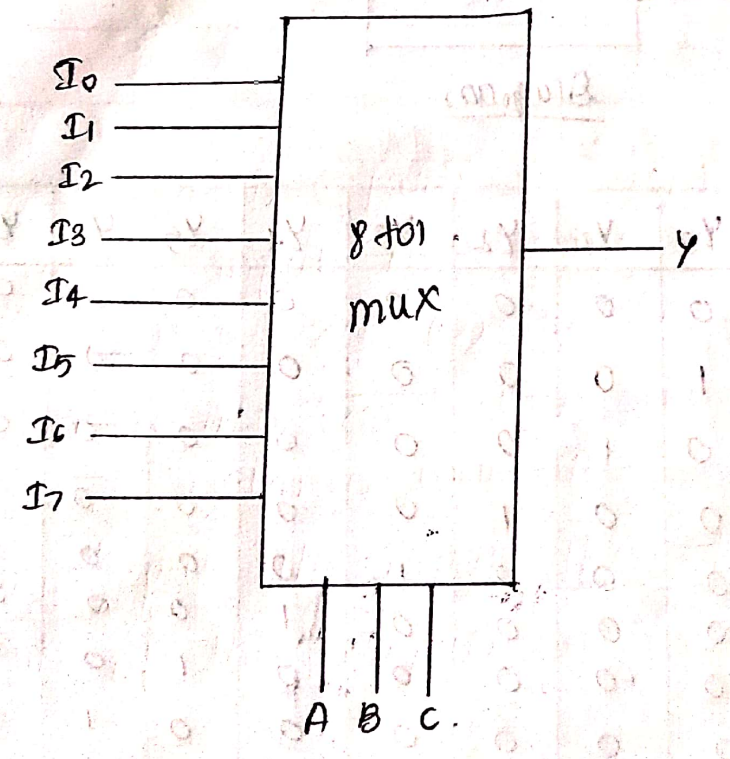
b) 8 to 1 mux.

multiplexer : It is combinational circuit which has many inputs and 1 output. It is also called as data selector and operates as data selector.

It is having 2^n selector lines which divides the input and gives as output.

Application : Data selector works for selection of 1 data in many data and finally gives that selected data as output.

Block diagram



Truth table

A	B	C	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

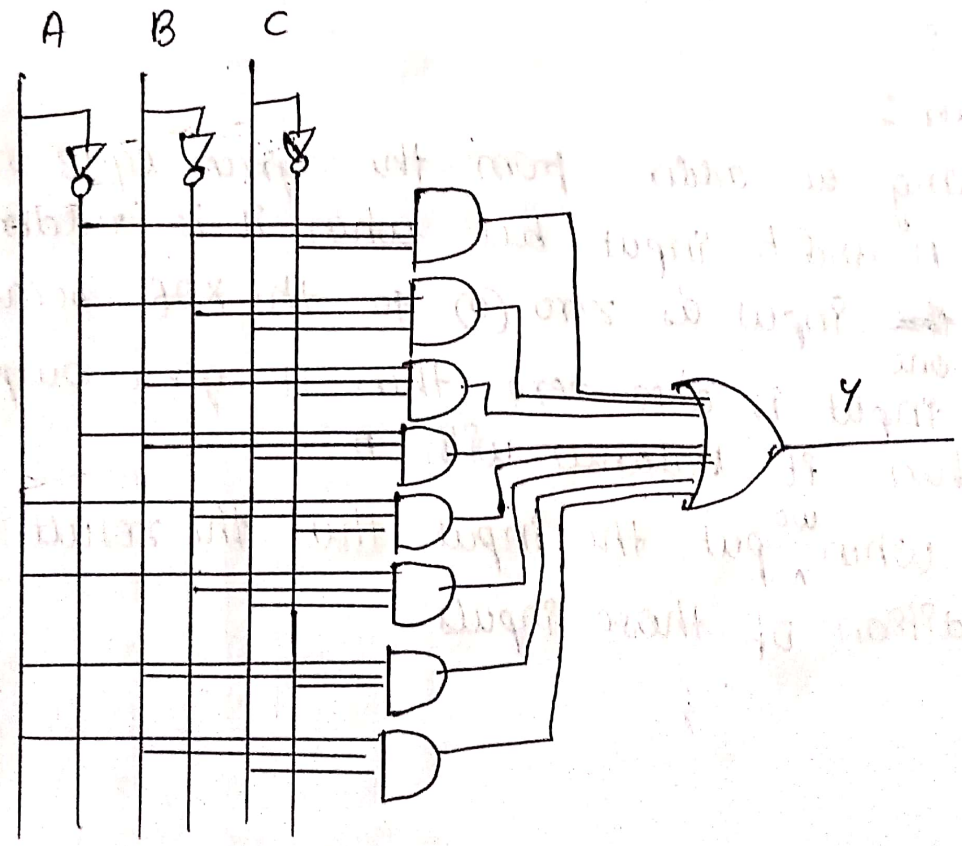
$I_0 = A'B'C'$ $I_4 = AB'C'$

$I_1 = A'B'C$ $I_5 = AB'C$

$I_2 = A'BC'$ $I_6 = ABC'$

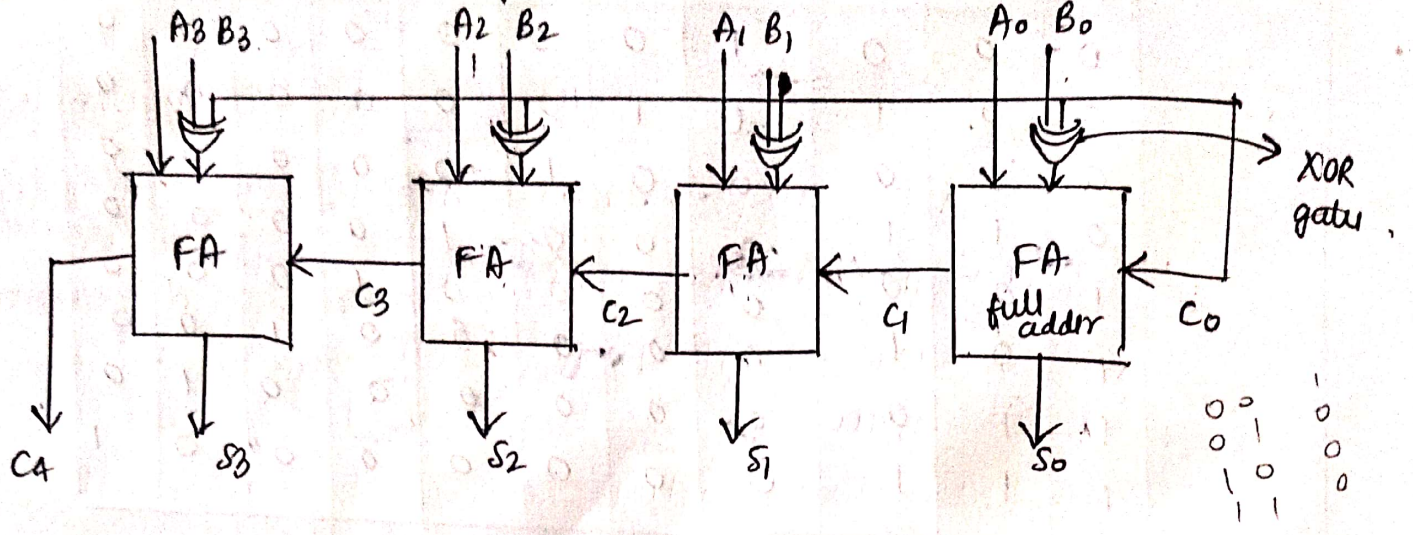
$I_3 = A'BC$ $I_7 = ABC$

logic diagram



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

4) a) Parallel binary adder / subtractor.



Construction:

Take four full adder and connect them and add the XOR gate, put one I/P from C_0 and another from the given problem. to full adder. Then one more input A to the full adder. Connect the sum and carry lines as shown.

Operation:

(i) Adder:

Working as adder from the given equn or problem provide A and B input but when it is adder then provide one input as zero (0) to the XOR because if another input is also zero then it gives output of XOR as 1 then it balances with A.

When we put the input then the result will be the addition of those inputs.

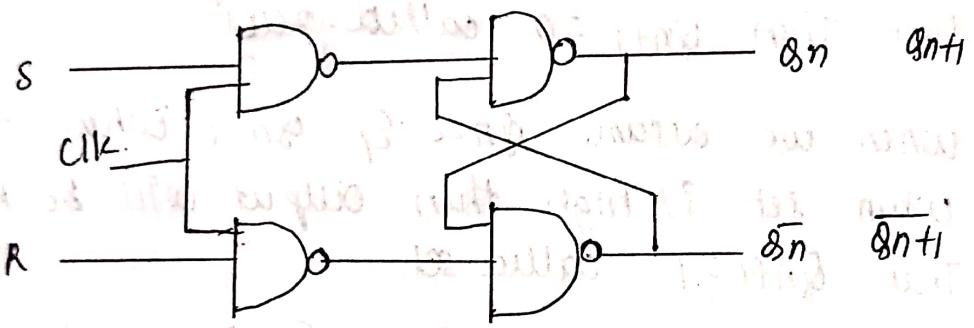
(iii) Subtractor :

When parallel binary adder work as the subtractor then we have to give one input as 1 to the XOR gate. Then the parallel binary adder works as subtractor to the given equation or problem.

The result given by this will be subtraction of the both inputs.

6) SR Flip flop with NAND Gate

Diagram



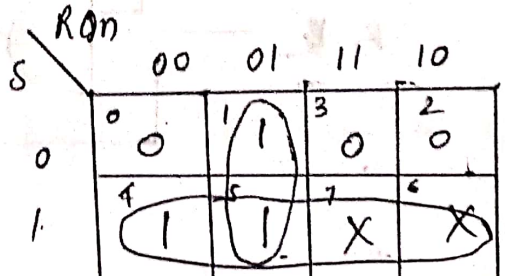
Truth table

S	R	Qn	Qn+1	
0	0	0	0	No change.
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	set.
1	0	1	1	
1	1	0	ND	not defined
1	1	1	ND.	

Excitation table

Qn	Qn+1	S	R
0	0	0	X
0	1	0	1
1	0	1	0
1	1	X	0

K map



$$Y = S + \bar{R}Q_n$$

Operation : The SR flip flop is made up of SR latch and along with two NAND gate.

The SR flip flop has two inputs S and R.

(i) When we assume that the (present output) $Q_n = 0$ when $S = R = 0$ and $Q_n = 1$ when $S = R = 0$. In this time the next output does not change. $Q_{n+1} = 0, = 1$

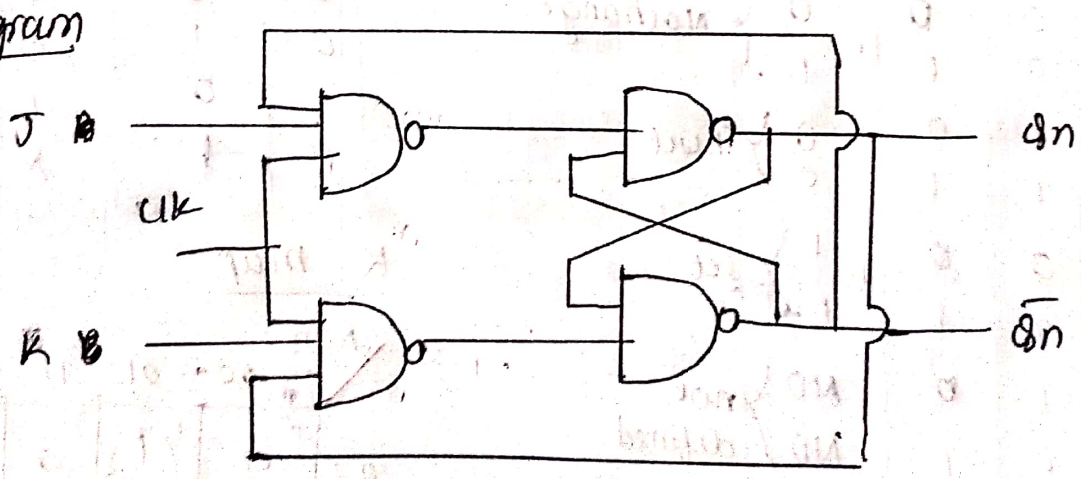
(ii) When we assume $Q_n = 0$ ~~when~~ and $Q_n = 1$ when $S = R = (0,1)$. When reset is high then output will be low. Then $Q_{n+1} = 0$. called reset.

(iii) When we assume $Q_n = 0$ & $Q_n = 1$ when $S = R = (1,0)$ when set is high then output will be high. Then $Q_{n+1} = 1$ called set.

(iv) When we assume $Q_n = 0$ & $Q_n = 1$ when $S = R = (1,1)$ then which is not possible.

JK flip flop

Diagram



CMR

Truth table

J	K	Qn	Qn+1	
0	0	0	0	No change
0	0	1	1	
0	1	0	0	reset
0	1	1	0	
1	0	0	1	set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

Excitation table

Qn	Qn+1	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

K-map

S \ RQn	00	01	11	10
0	0	1	0	0
1	1	1	0	1

$Y = \bar{R}Q_n + S\bar{R}\bar{Q_n}$

operation :

(i) when we assume that $Q_n = 0$ and $Q_n = 1$ when $J = K = 0$ then the next state output $Q_{n+1} = 0$ no change in the output

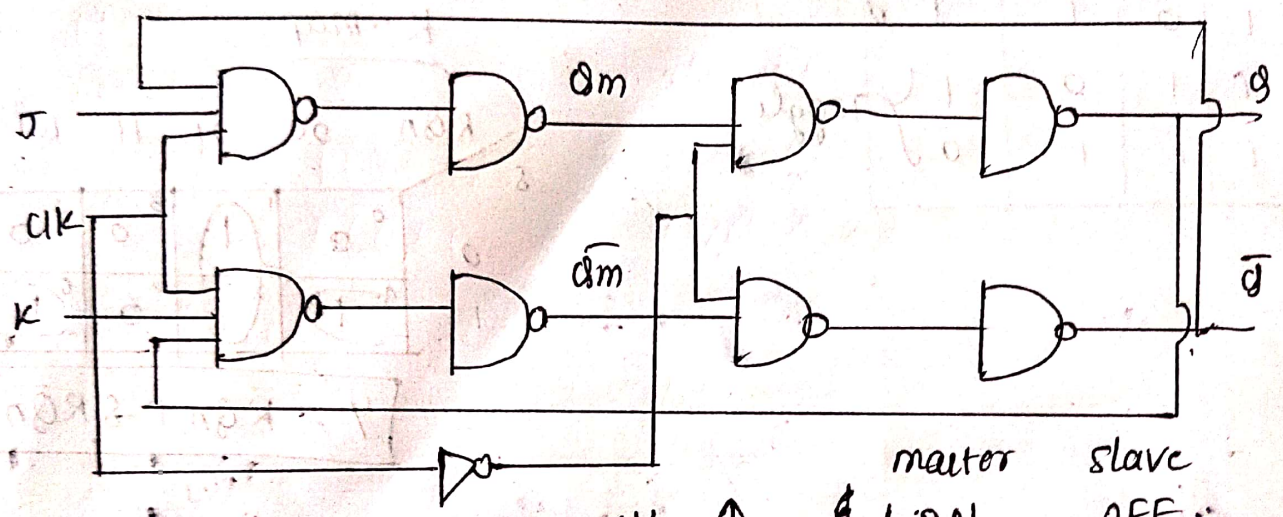
(ii) when we assume $Q_n = 0$ and $Q_n = 1$ when $J = K = (0,1)$ then the next state output $Q_{n+1} = 0$ because K = reset means zero.

(iii) When we assume $Q_n = 0$ & $Q_n = 1$ when $J = K = (1,0)$ then the $Q_{n+1} = 1$ because J = set means one

(iv) Here the JK has used toggling to get output for when $J=K=1$ and assume $Q_n=0$ and 1 then this toggle and gives output as $Q_{n+1}=1$ and $Q_{n+}=0$ when $Q_n=0$ and $Q_n=1$.

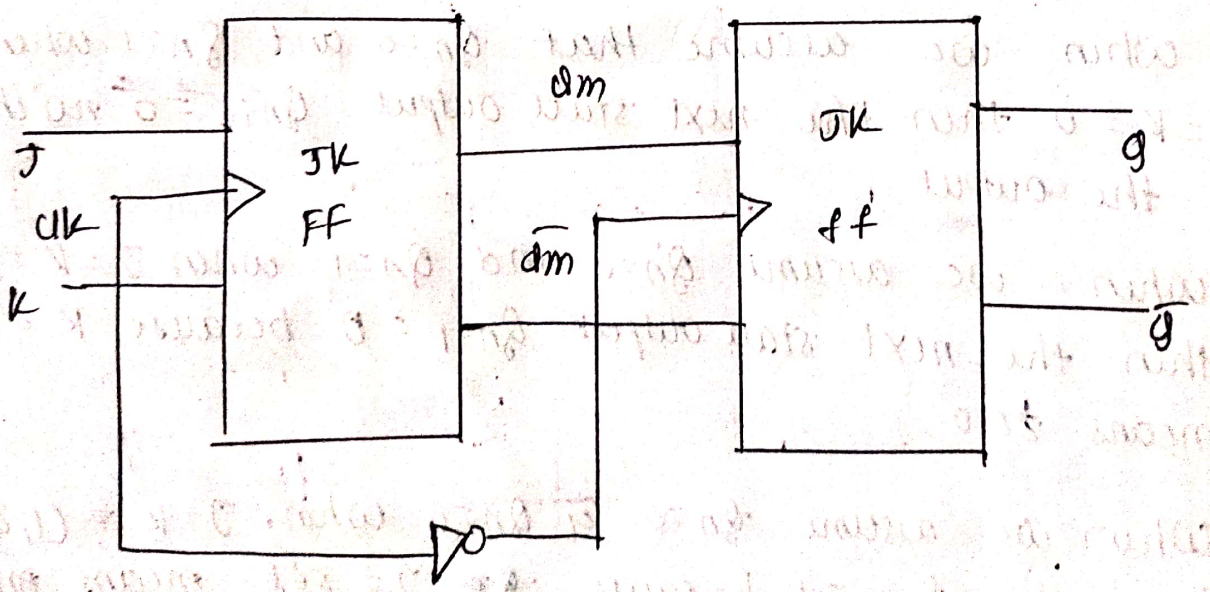
7) Master-slave JK flip flop.

Diagram



	master	slave
clk ↑	ON	OFF
clk ↓	OFF	ON

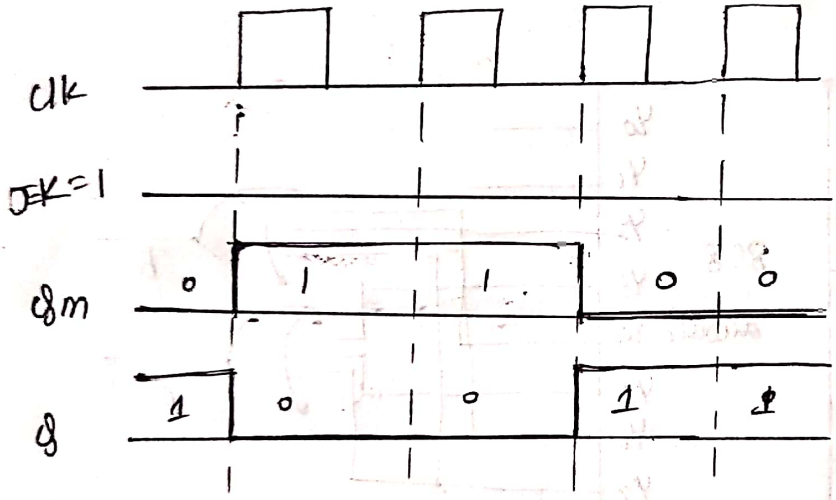
Block diagram



Truth table

clk	J	K	Q_m	$\bar{Q}$	
1	0	0	0	0	} No change.
1	0	0	1	1	
1	0	1	0	0	} Reset
1	0	1	1	0	
1	1	0	0	1	} set
1	1	0	1	1	
1	1	1	0	1	} Toggle.
1	1	1	1	0	

Waveform



Working : In the normal JK flip flop is used to remove the not defined for last term and we use toggle but we use MS JKFF because in JKFF we have high racing in it which can't be controlled so, we use this to remove racing from it. In waveform when the clock is high. then Q_m it takes two terminal & two terminal as low.

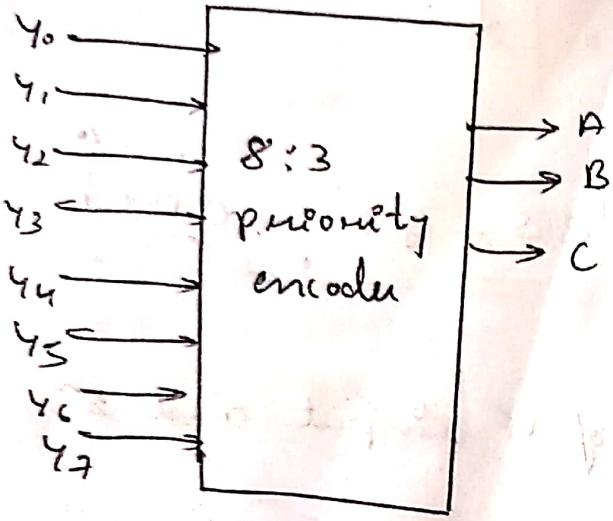
Encoder continuity:

(2) Reducible Form to Decimal Form.

(12)

The output for priority encoder depends upon the highest priority in input side: If the encoder consists of 2 or more highest priority then it choose which as more significant higher.

Logical diagrams



-10-

Truth Table

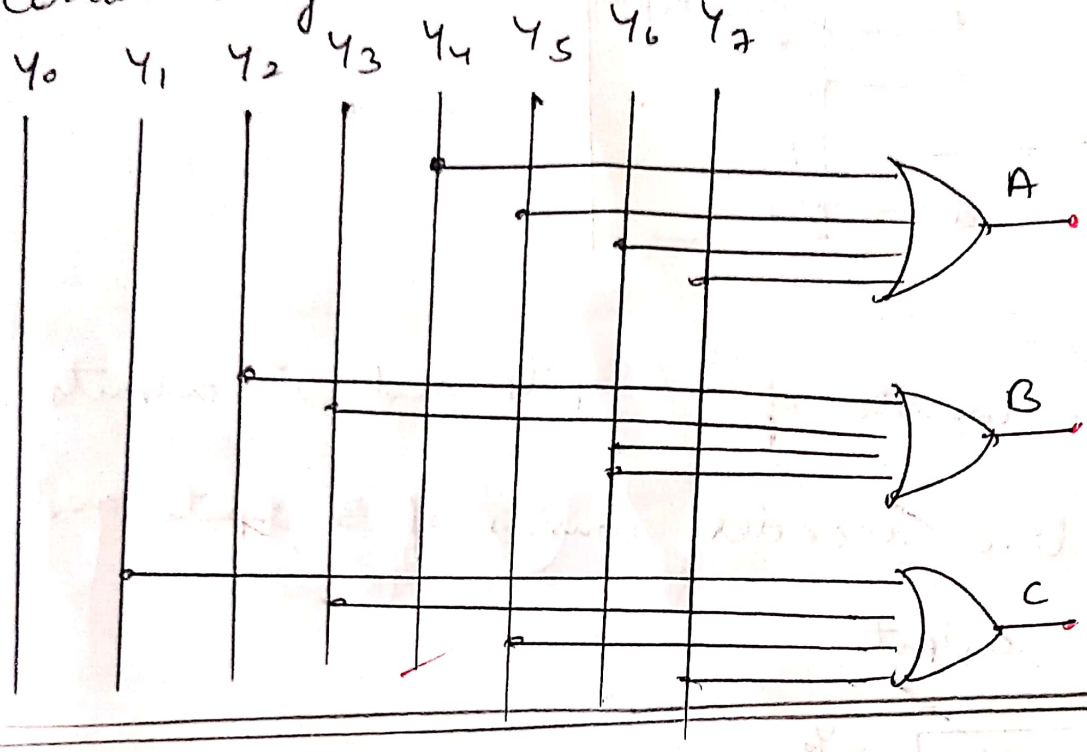
Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	A	B	C
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	X	X	0	0	1
0	0	1	X	X	X	X	X	0	1	0
0	0	0	1	X	X	X	X	0	1	1
0	0	0	0	1	X	X	X	1	0	0
0	0	0	0	0	1	X	X	1	0	1
0	0	0	0	0	0	1	X	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$A = Y_4 + Y_5 + Y_6 + Y_7$

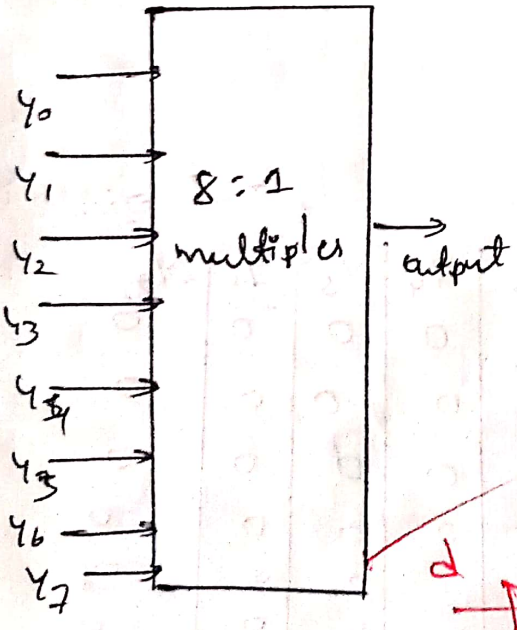
$B = Y_2 + Y_3 + Y_6 + Y_7$

$C = Y_1 + Y_3 + Y_5 + Y_7$

Circuit diagram

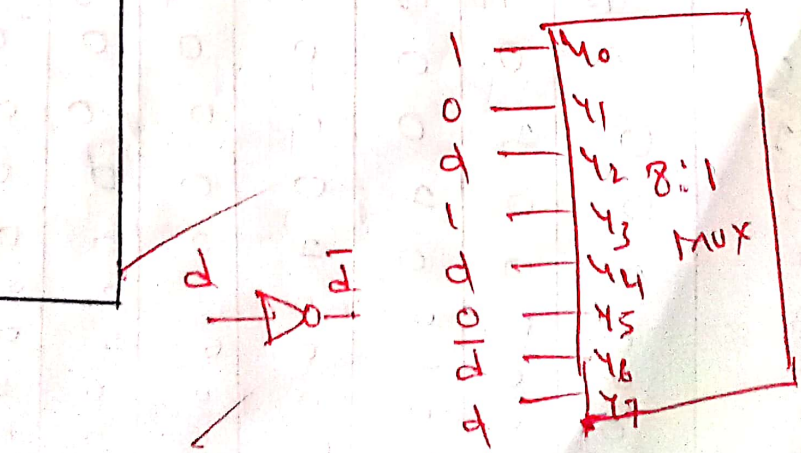


b) $f(w, x, y, z) = \Sigma(0, 1, 5, 6, 7, 9, 12, 15)$



	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
$\bar{d}$	1 0	0 2	0 4	1 6	0 8	0 10	1 12	0 14
d	1 1	0 3	1 5	1 7	1 9	0 11	0 13	1 15

1 0 d 1 d 0 $\bar{d}$ d..



(5) The different types of PLD's are

PROM = Programmable read only memory

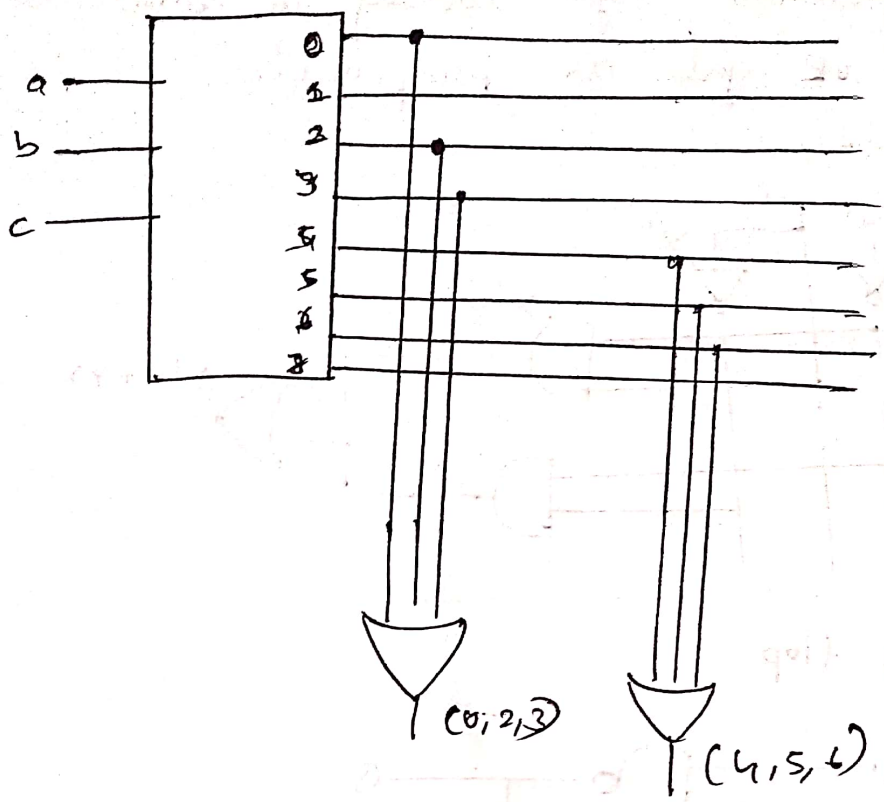
PLA = Programmable Logical array

PAL = Programmable array logic

① Programmable read only memory (PROM) in which programmable OR gate and fixed AND.

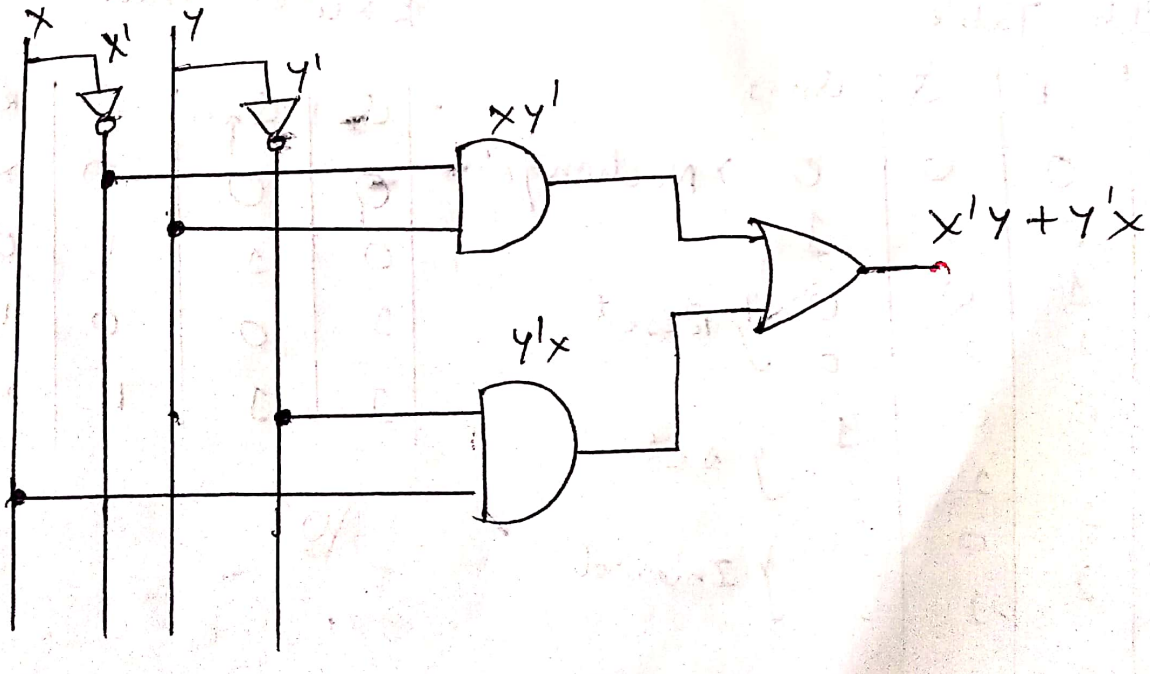
Example : $f(x) = \Sigma(0, 2, 3)$ and $f(x) = \Sigma(4, 5, 6)$

$f(a, b, c)$ $f(a, b, c)$



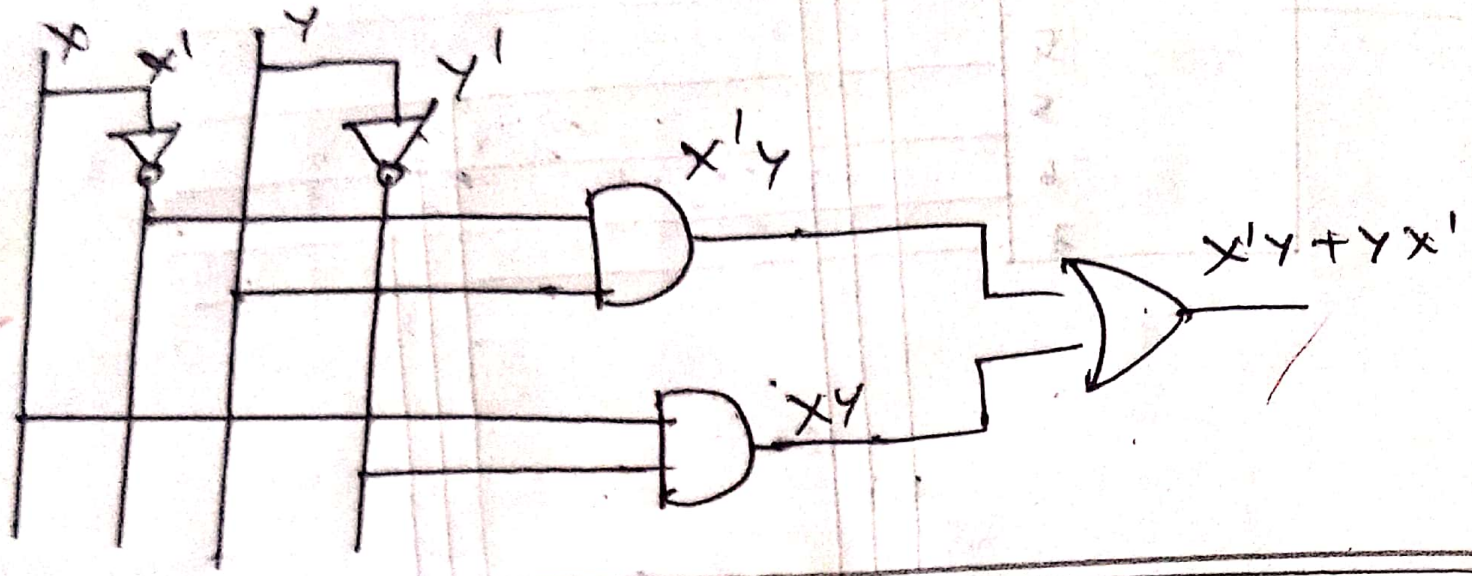
② PAL :- Programmable array logic in which programmable AND gate and fixed OR gate

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



(3) PLA :- Programmable logic array in which both AND and OR gate are programble.

$f(x) = x'y + yx'$



10

nt No.

Page No.:

Q.No: (8). Convert following Boolean function into

i. $f(abc) = (\bar{a}+b)(b+\bar{c})$ - min term

ii $= (\bar{a}+b+c\bar{c})(b+\bar{c}+a\bar{a})$

$(\bar{a}+b+c)(\bar{a}+b+\bar{c})(a+b+\bar{c})(\bar{a}+b+\bar{c})$

= POS Truth table

D	a	b	c	y
0	1	1	1	→ 0
1	1	1	0	→ 1 → 1
2	1	0	1	→ 0
3	1	0	0	→ 0
4	0	1	1	→ 4 → 1
5	0	1	0	→ 5 → 1
6	0	0	1	→ 0
7	0	0	0	→ 0

$\therefore f(abc) = \Pi(1,4,5)$

$\therefore f(abc) = \Sigma(0,2,3,6,7)$

min term.

(2.) $f(x,y,z) = x + \bar{x}\bar{z}(y+\bar{z})$

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

$x(y+\bar{y})(z+\bar{z}) + \bar{x}y\bar{z} + \bar{x}\bar{z}(y+\bar{y})$

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

Repeated so

$\Sigma(0,2,4,5,6,7) = \Pi(1,3)$

max term

	x	y	z	f
0	0	0	0	1
1	0	0	1	0 → 0
2	0	1	0	1
3	0	1	1	0 → 3
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1