Model Exam - II
2022
CS8351

A. Ashokmathra
312H2010H016
CSE - A
22/10/21
CS8351
Digital Principle Design

PART-B

11. a) $A \cdot B + A \cdot C \cdot D = (A+C)(A+D)(B+C)(B+D)$

11. b) $AC' + B'D + A'CD + ABCD$

$AC' \cdot 1 + B' \cdot D \cdot 1 + A'CD \cdot 1 + ABCD$

$AC' \cdot (B+B') + B' \cdot D \cdot (A+A') + A'CD \cdot (B+B') + ABCD$

$AC'B + AC'B' + AB'D + A'B'D + A'BCD + A'B'CD + ABCD$

$AC'B \cdot 1 + AC'B' \cdot 1 + AB'D \cdot 1 + A'B'D \cdot 1 + A'BCD + A'B'CD + ABCD$

$AC'B \cdot (D+D') + AC'B' \cdot (D+D') + AB'D \cdot (C+C') + A'B'D \cdot (C+C') + A'BCD + A'B'CD + ABCD$

$AB'C'D + ABC'D' + \underline{AB'C'D} + AB'C'D' + AB'CD + \underline{AB'C'D} + \underline{A'B'CD} + A'B'C'D + A'BCD + \underline{A'B'CD} + ABCD$

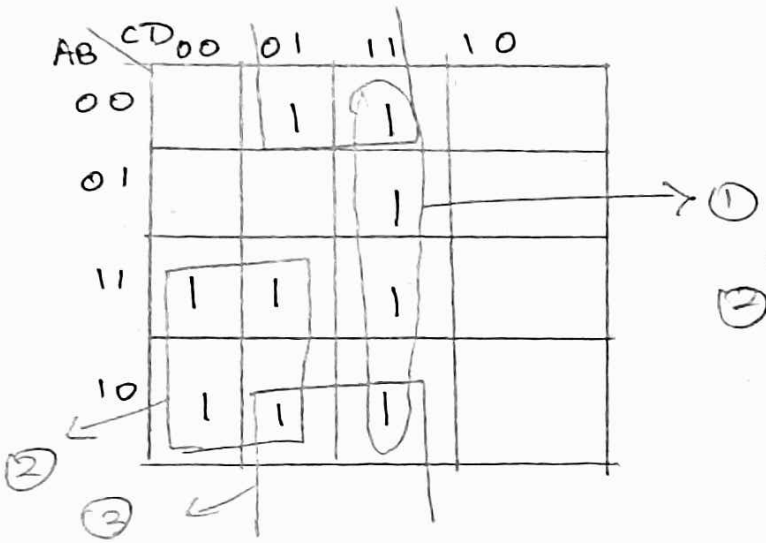
1101 1100 1001 1000 0111 0011 0001

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

0101 0111 1111

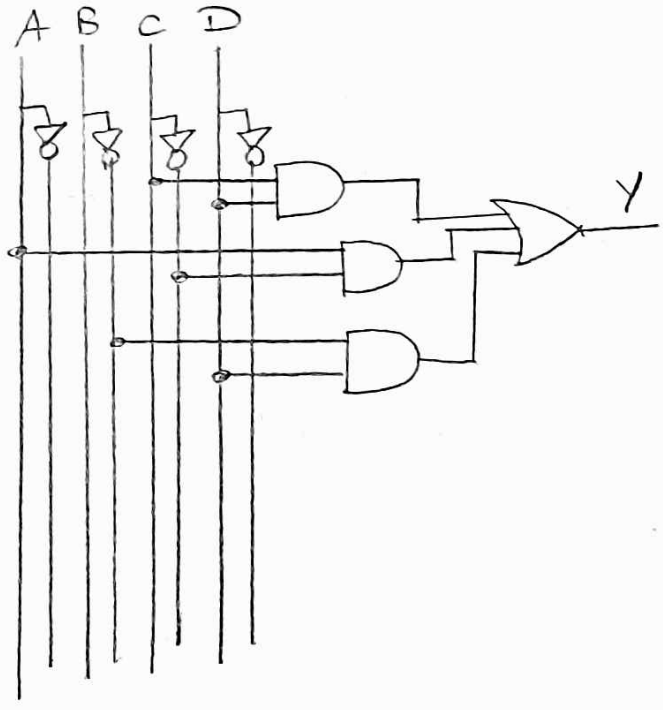
K-map (SOP)

$\Sigma m = (13, 12, 9, 8, 11, 3, 1, 7, 15)$

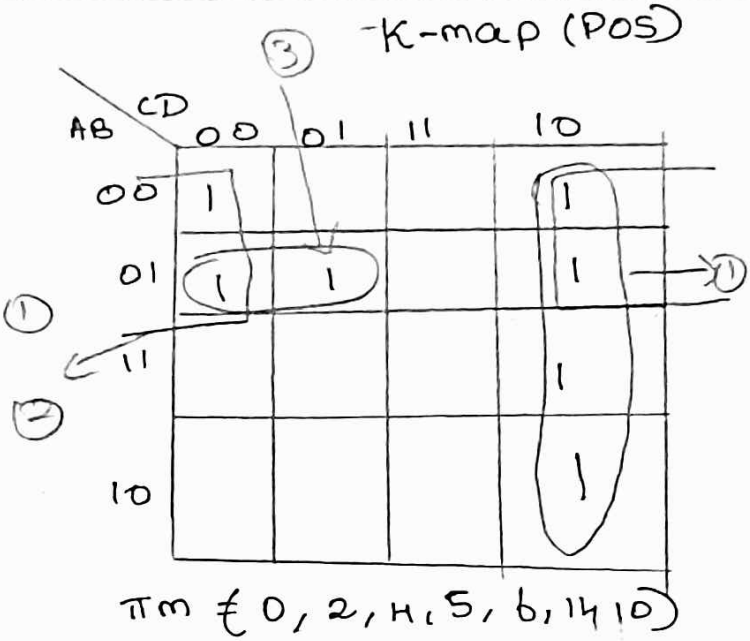


$Y = CD + A\bar{C} + \bar{B}D$

Logic Diagram



K-map (POS)



$\pi m \{0, 2, 4, 5, 6, 14, 10\}$

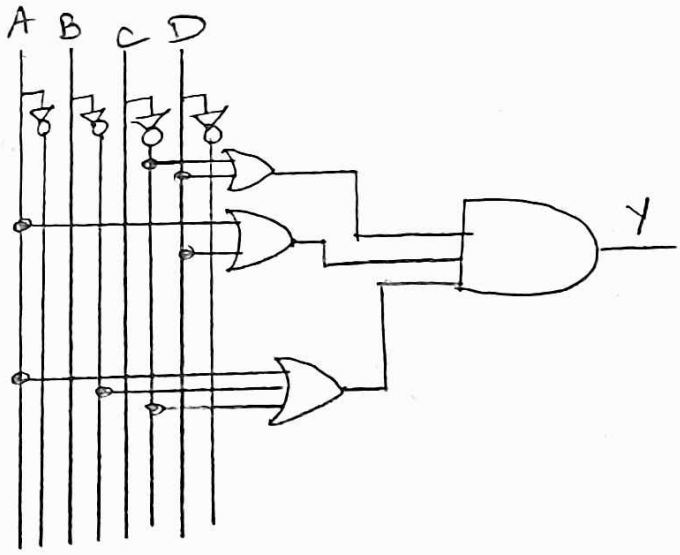
$\bar{Y} = C\bar{D} + \bar{A}\bar{D} + \bar{A}B\bar{C}$

Logic Diagram.

$Y = C\bar{D} + \bar{A}\bar{D} + \bar{A}B\bar{C}$

$Y = (\bar{C}+D)\cdot(A+\bar{D})(A+\bar{B}+C)$

Logic Diagram.



12. a Excess-3 to BCD code.

Excess-3 code is a self complementing code because the binary sum of a code and its 9's complement is equal to 9 complement can be generated by inverting each bit patter.

State Table.

Excess-3 code				BCD code			
E_3	E_2	E_1	E_0	B_3	B_2	B_1	B_0
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1

The above State table representing BCD equivalent excess-3 values. Since the BCD code has the range of (0-9) The table has given values with equivalent excess-3 code.

For B_3

E_3E_2 \ E_1E_0	00	01	11	10
00	0	x	0	x
01	0	0	0	0
11	1	x	x	x
10	0	0	1	0

For B_2

E_3E_2 \ E_1E_0	00	01	11	10
00	x	x		x
01			1	
11		x	x	x
10	1			1

For B_1

E_3E_2 \ E_1E_0	00	01	11	10
00	x	x		x
01		1		
11		x	x	x
10		1		1

$B_3 = E_3 E_2 + E_3 E_1 E_0$, $B_2 = \bar{E}_2 \bar{E}_1 + E_2 E_1 E_0 + E_3 E_1 \bar{E}_0$, $B_1 = \bar{E}_1 E_0 + E_1 \bar{E}_0$

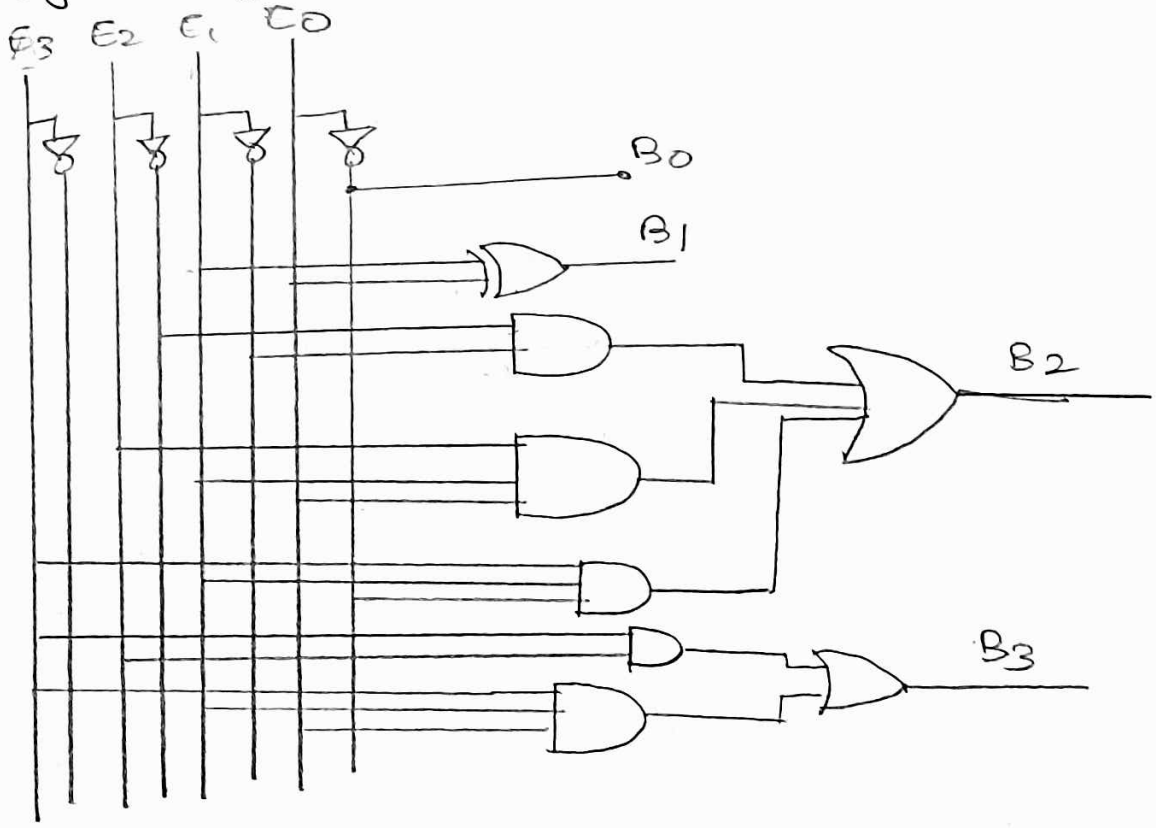
$B_1 = E_1 \oplus E_0$

For B_0

E_3E_2 \ E_1E_0	00	01	11	10
00	X	X		X
01	1			1
11	1	X	X	X
10	1			1

$B_0 = \bar{E}_0$

Logic Diagram.



15- a) 3 inputs, 4 product term, 2 output (PLA)

$F_1(A,B,C) = \Sigma m(3,5,6,7)$

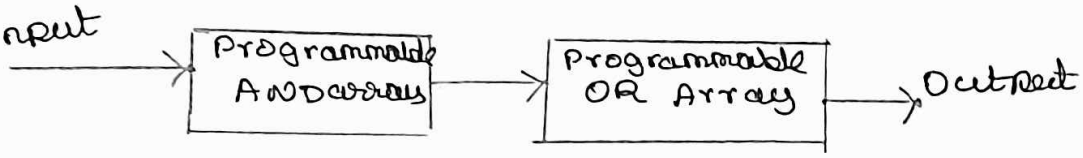
$F_2(A,B,C) = \Sigma m(0,2,4,7)$

Sol

Programmable Logic Array (PLA)

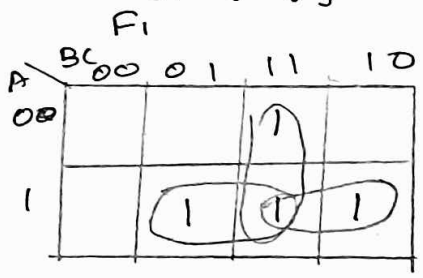
It is a type of fixed architecture logic device with programmable AND gate followed by programmable OR gate.

It is used to implement a complex combinational circuit.

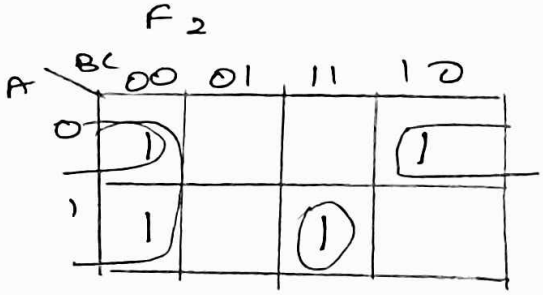


Step 1:

Simplify using K-map:



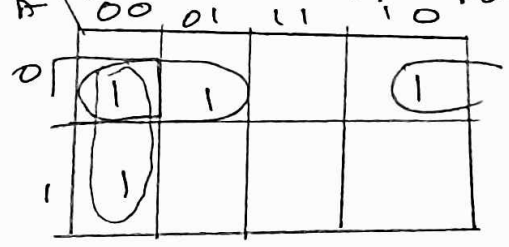
$F_1 = BC + AC + AB$



$F_2 = \bar{B}\bar{C} + \bar{A}\bar{C} + ABC$

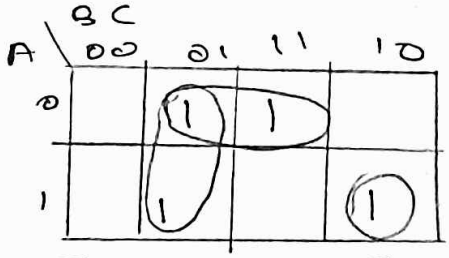
Since 6 distinct product terms are there we need only access of 4 product term.

For $\bar{F_1}$



$\bar{F_1} = \bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}\bar{C}$

For $\bar{F_2}$



$\bar{F_2} = AB\bar{C} + \bar{B}C + \bar{A}C$

Step 2 : Determine the number of product terms by comparing simplified outputs.

$F_1 F_2$ → 6 Product term

$F_1 \bar{F}_2$ → 6 Product term

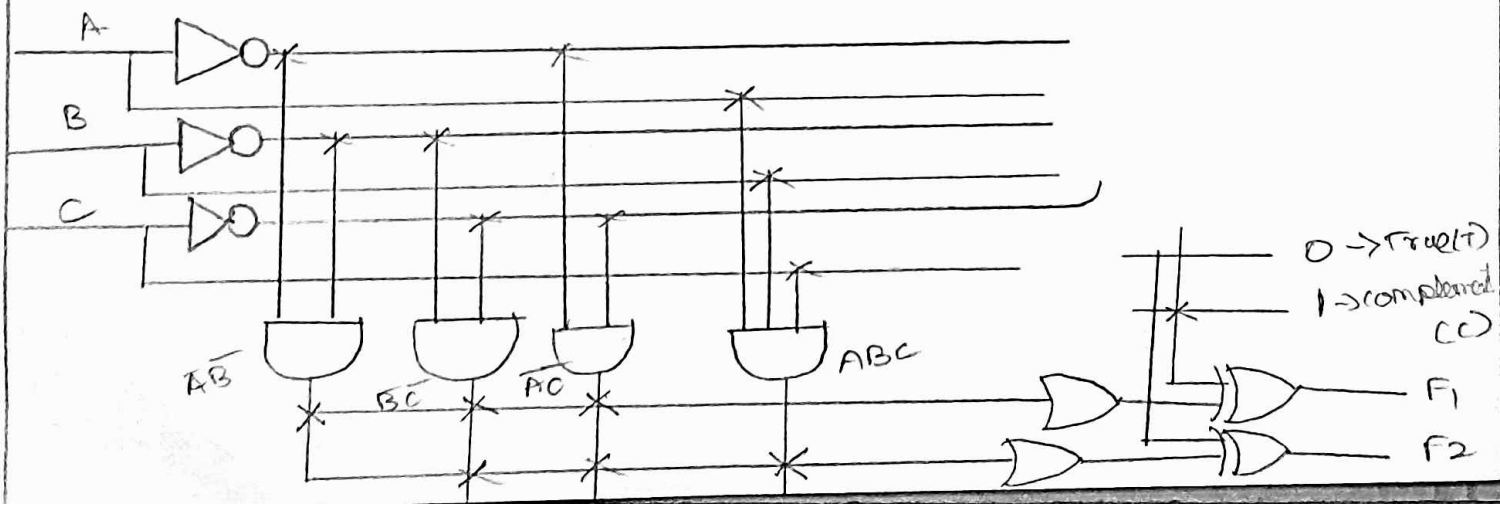
$\bar{F}_1 F_2$ → 4 Product term ✓

$\bar{F}_1 \bar{F}_2$ → 6 Product term

Step 3 : Take $\bar{F}_1 F_2$ to implement the function of PLA Programming table.

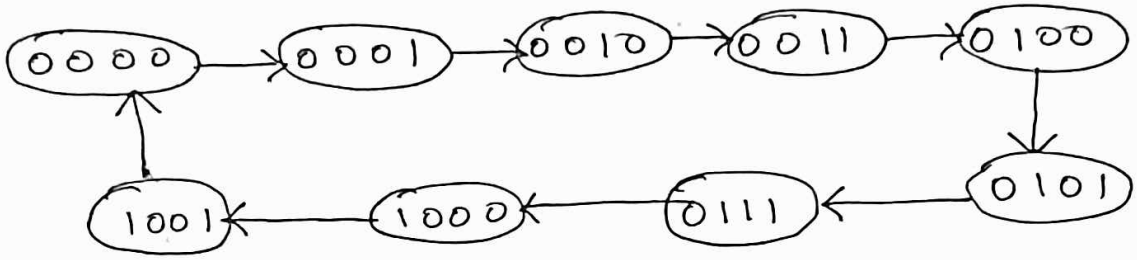
Product Term	Input			Output	
	A	B	C	F_1	F_2
$\bar{A}\bar{B}$	0	0	–	1	¢ –
$\bar{B}\bar{C}$	–	0	0	1	1
$\bar{A}\bar{C}$	0	–	0	1	1
ABC	1	1	1	–	1
				C	T

Step 4 : Logic Diagram.



13. a Design Synchronous Mod 10 Counter using JK Flip Flop.

State diagram



Excitation Table.

Q_n	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

State Table.

Present State				Next state				J				K			
Q_A	Q_B	Q_C	Q_D	Q_{A+1}	Q_{B+1}	Q_{C+1}	Q_{D+1}	J_A	J_B	J_C	J_D	K_A	K_B	K_C	K_D
0	0	0	0	0	0	0	1	0	0	0	1	x	x	x	x
0	0	0	1	0	0	1	0	0	0	1	x	x	x	x	1
0	0	1	0	0	0	1	1	0	0	x	1	x	x	0	x
0	0	1	1	0	1	0	0	0	1	x	x	x	x	1	1
0	1	0	0	0	1	0	1	0	x	0	1	x	0	x	x
0	1	0	1	0	1	1	0	0	x	1	x	x	0	x	1
0	1	1	0	0	1	1	1	0	x	x	1	x	0	0	x
0	1	1	1	1	0	0	0	1	x	x	x	x	1	1	1
1	0	0	0	1	0	0	1	x	0	0	1	0	x	x	x
1	0	0	1	0	0	0	1	x	0	0	x	1	x	x	0

K-map

JA

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00				
01			1	
11	X	X	X	X
10	X	X	X	X

$J_A = Q_B Q_C Q_D$

JB

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00			1	
01	X	X	X	X
11	X	X	X	X
10			X	X

$J_B = Q_C Q_D$

JC

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00		1	X	X
01		1	X	X
11	X	X	X	X
10			X	X

$J_C = \bar{Q}_A Q_D$

JD

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00	1	X	X	1
01	1	X	X	1
11	X	X	X	X
10	1	X	X	X

$J_D = 1$

KA

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	X	X	X	X
10		1	X	X

$K_A = Q_D$

KB

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00	X	X	X	X
01			1	
11	X	X	X	X
10	X		X	X

$K_B = Q_C Q_D$

K_C

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00	x	x	1	0
01	x	x	1	0
11	x	x	x	x
10	x	x	x	x

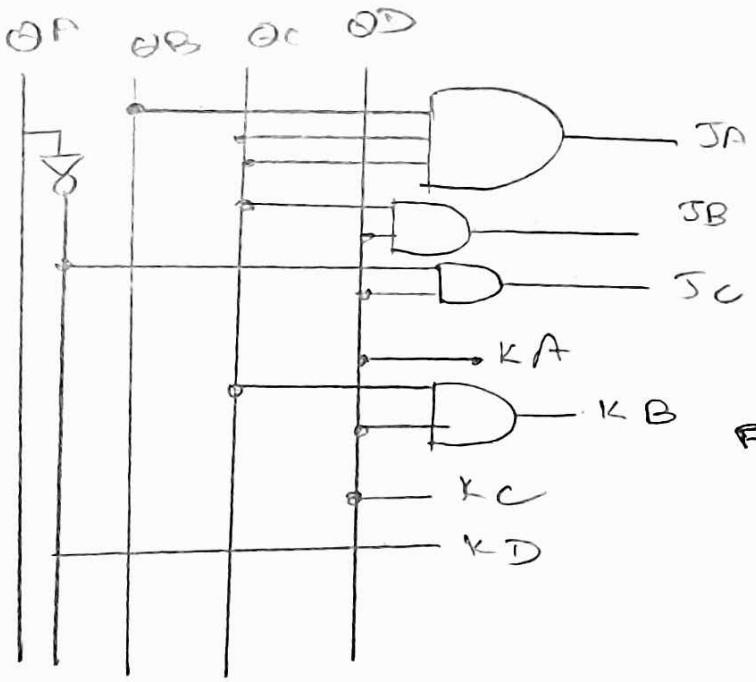
$K_C = Q_A Q_D$

K_D

$Q_A Q_B$ \ $Q_C Q_D$	00	01	11	10
00	x	1	1	x
01	x	1	1	x
11	x	x	x	x
10	x	0	x	x

$K_D = \overline{Q_A}$

Logic Diagram



For J

For K

PART-A

1. $X = 1010100$
   $Y = 1000011$

a) $X - Y$

$X = 1010100$

2's complement $Y = +0111101$

sum $= 10001000 1$

$X - Y = 00100001$

b)

$Y = 1000011$

$X = +0101100$

1101111

There is no end carry

$Y - X = (2\text{'s complement of } 110\,1111) = -0010001$

2. De Morgan Theorem

De-Morgan's Theorem 1: The complement of product of any number of variables is equivalent sum of individual complements.

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

A	B	AB	$(AB)'$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

A'	B'	$A'+B'$
1	1	1
1	0	1
0	1	1
0	0	0

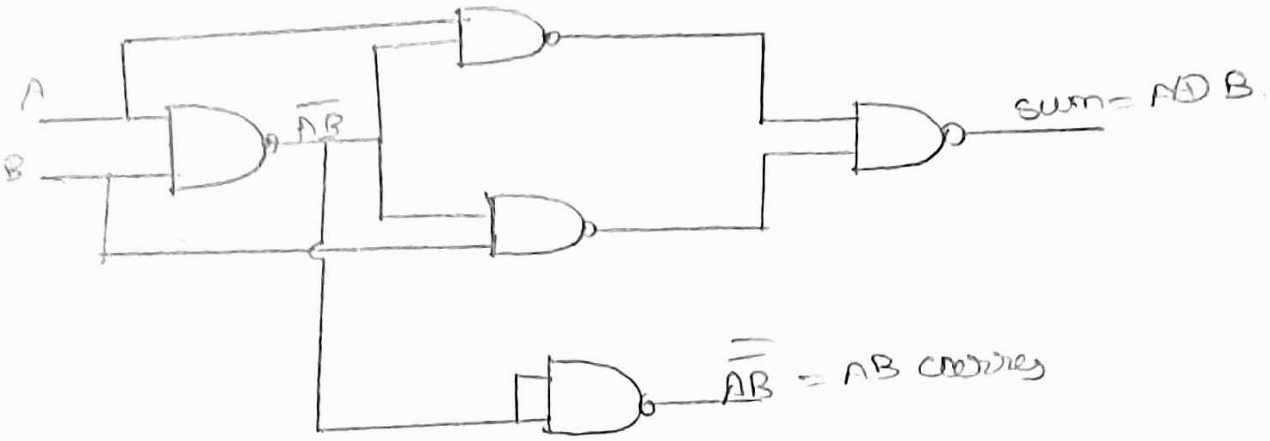
De Morgan Theorem 2: The complement of sum of any number of variables is equivalent to product of the individual component.

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

A	B	$A+B$	$(A+B)'$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

A'	B'	$A'B'$
1	1	1
1	0	0
0	1	0
0	0	0

3.



4. Priority Encoder.

A priority encoder is an encoder circuit that includes priority function. The operation of the priority encoder is such that if two or more inputs are activated at same time the output of the binary code will be generated to the highest number of input.

5. Use of Shift Register

→ Storage Device – The primary use of shift register is temporary storage.

→ Time delay generation

→ Serial to parallel converter (SIPO)

→ Parallel to Serial Converter (PISO)

→ Shift register counters.

6. Race around condition can be eliminated by

1. Using the edge triggered JK FF
2. Using the master slave J-K flipflop.

7. Shared Row Method

The method of making race free assignment by introducing extra rows in the flow table is sometimes referred shared row method.

8. Incorrect conclusion is the conclusion arrive in a counter when gets into a forbidden state and neither return normal counting sequence or to another acceptable state or initial state, the counter switches to another forbidden state and get stuck up in the cycle is forbidden codes.

9. a) Use of PLA

*. Combinational circuit can be implemented using PLA's

*. Sequential circuit can be be implemented using PLA's

*. Compact circuit can be built using PLA's which occupies less space.

10. Field Programmable Logic Array (FPGA)

It is an integrated circuit designed to be configured by a customer or a designer after manufacturing hence it is field programmable".

PART-C

16. b) Synchronous Counter by (0, 1, 3, 7, 6, 4)

	Present State			Next State			FF Input					
CLK	Q_C	Q_B	Q_A	Q_{C+1}	Q_{B+1}	Q_{A+1}	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	0	0	0	1	0	x	0	x	1	x
1	0	0	1	0	1	1	0	x	1	x	x	0
3	0	1	1	1	1	1	1	x	x	0	x	1
7	1	1	1	1	1	0	x	0	x	0	x	1
6	1	1	0	1	0	0	x	0	x	1	0	x
4	1	0	0	0	0	0	0	x	0	x	0	x
							x	1				