29/08/23

Dr. Mohd Chand

Logic

Bank Loker System :

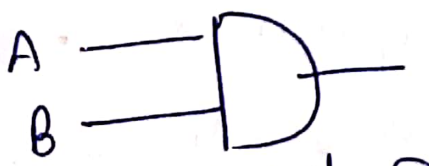
✓ Manager → A (Permanently).
Cashier 1 → B } koi Ek chalega
Cashier 2 → C

Truth table

A	B	C	O/P
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

open = 1
Lock = 0

Boolean Expression.



and gate

or gate

Number systems :-

~~1) Binary numbers~~ :

(1) Decimal number -

- Ten digits (0-9)
- Base or radix 10.
- Cofficecents are multiplied by power of 10.
- Positional weighted system.

$7392 \cdot 54 = 2 \times 10^0 + 9 \times 10^1 + 3 \times 10^2 + 7 \times 10^3.$

$5 \times 10^{-1} + 4 \times 10^{-2}$

$\longleftarrow$ Multiply with power 10

$\longrightarrow$ Multiply with (-) power of 10

(2) Binary number System

- Two digits (0, 1)
- Base or ~~r~~edix = 10
- cofficents are multiplied by Power of 2
- Positional weighted system

$1101.101 = 1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3.$
$1 \times 10^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$

(3) octal.

- 8 digits (0-7)
- Base or Redix = 8
- Cofficients are multiplied by power 8
- positional weighted System

$1101.101 = 1 \times 8^0 + 0 \times 8^1 + 1 \times 8^2 + 1 \times 8^3.$
$1 \times 8^{-1} + 0 \times 8^{-2} + 1 \times 8^{-3}$

(4) Hexadecimal (0-9) (A-F).

- 16 digits (0-9, A-F)
- Base or Redix = 16
- cofficients are multiplied by power 16.
- Positional weighted System.

Brtin

Binary addition

A	B	O/P
0	0	0
0	1	1
1	0	1
1	1	0 , carry 1

Binary subtraction

A	B	O/P
0	0	0
0	1	1 , borrow of 1
1	0	1
1	1	0

augend : 101101

adderd : 100111

sum 1010100

minuend : 101101

subtraherd : -100111

000110

Lab

IC → Integrated circuit

Anticlockwise

$\frac{1}{2}+\frac{1}{4}$

$\frac{3}{4}$.

4/09/23

Binary to decimal conversion

$(101.11)_2 = (\qquad)_{10}$

$\overleftarrow{\quad}$ $\overrightarrow{\quad}$

Redix/ Base

$101 = 1\times 2^0 + 0\times 2^1 + 1\times 2^2 = 1+0+4 = (5)_{10}$

$.11 = 1\times 2^{-1} + 1\times 2^{-2} = (0.75)_{10}$

$(101.11)_2 = (5.75)_{10}$

(b) $(11011.101)_2$

$11011 = 1\times 2^0 + 1\times 2^1 + 0\times 2^2 + 1\times 2^3 + 1\times 2^4$

$= 1+2+0+8+16 = 27$

$.101 = 1\times 2^{-1} + 0\times 2^{-2} + 1\times 2^{-3}$

$= \frac{1}{2} + 0 + \frac{1}{8} = 0.12 + 0.5 = 0.17$

Scanned with OKEN Scanner

$\frac{5}{8} = (0.62)_{10}$

$$8 \overline{)50}\ \ 0.6$$
$$48$$

$(11011.101)_2 = (27.625)_{10}$

<u>Decimal to Binary</u>

(a) $(52.75)_{10} = (?)$

Double - Dabble method.

$(52)_{10} = (110100)_2$

2	52	
2	26	0
2	13	0
2	6	1
2	3	0
2	1	1
	0	1

Downward to upward

$(0.75)_{10}$

upward to Downward

$0.75 \times 2 = 1.5$

$0.50 \times 2 = 1.00$

$(0.75)_{10} = (0.11)_2$

$(52.75)_{10} = (110100.11)_2$

(b) $(0.6875)_{10} = (?)$

$\frac{1}{8} + \frac{1}{2}$

$0.6875 \times 2 = 1.3750$

~~1.3750 × 2 = 1.7500~~

~~1.7500 × 2 = 5000~~

$.3750 \times 2 = 0.7500$

$.7500 \times 2 = 1.5000$

$.5000 \times 2 = 1.0000$

$(0.6875)_{10} = (.1011)_2$

0.12
8)10
8
20
1.75
×2
3.50
3750
×2
7500
×2
15000
×2

Homework

$(0.63)_{10}$ 4/5 tk Karke stop Kardenge

$0.63 \times 2 = 1.26$

$0.26 \times 2 = 0.52$

$0.52 \times 2 = 1.04$

$0.04 \times 2 = 0.08$

$0.08 \times 2 = 0.16$

Down to up

$(0.63)_{10} = (.00106)_2$

③ octal to decimal

(a) $(258.21)_8 = 8\times8^0 + 5\times8^1 + 2\times8^2 .$
$2\times8^{-1} + 1\times8^{-2}$

$\frac{2}{8} + \frac{1}{64}$ $= (8+40+128).(0.25+0.15625)$

$= \cancel{124}\ 176.40625$

octal to Binary

Decimal no.	Binary Equivalent 4	2	1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Scanned with OKEN Scanner

4	2	0
2^2	2^1	2^0

$1 \quad 0 \quad 1 = (5)_8$

$(258.21)_8 = X$

$(275.21)_8$

$= (010111101.010001)_2$

$(257.21)_8$

$= (010101111.010001)_2$

$(1\underbrace{011}\underbrace{101}.\underbrace{001}10)_2$

$= (\underbrace{001}\underbrace{011}\underbrace{101}.\underbrace{001}\underbrace{100})_2$

$= (135.14)_8$

	Hexadecimal number	Binary Equivalent			
		8	4	2	1
	0	0	0	0	0
	1.	0	0	0	1
	2	0	0	1	0
	3	0	0	1	1
	4	0	1	0	0
	5	0	1	0	1
	6.	0	1	1	0
	7	0	1	1	1
	8	1	0	0	0
	9	1	0	0	1
10	A	1	0	1	0
11	B	1	0	1	1
12	C	1	1	0	0
13	D	1	1	0	1
14	E	1	1	1	0
15	F	1	1	1	1

[illegible]

8 4 2 1

(A) $(A42D.1E)_{16}$

$(1010010000101101.00011110)_2$

(b) $(\underbrace{10}\underbrace{1101}\underbrace{1011})_2$

MSB LSB

$(2DB)_{16}$

<u>complements</u> [(r-1)'s complement]

- Dimnished radix complement
- Radix complement ((r) complement).

1) 9's complement.

r = 10

25678

$$\begin{array}{r} 99999 \\ 25678 \\ \hline 74321 \\ \hline \end{array}$$

9's complement = 74321

10's complement = 74321 + 1
= 74322.

9's & 10's are valid decimal number (jinka Base 10 hoga)

1's complement & 2's complement valid only on Binary.

0	1	1	carry
1	0	1	
0	0	0	
1	1	0	

1's complement

1011101

```
 1111111
 1011101
 -------
 0100010
```

2's comlement = 0100010 + 1
= 0100011

Subtraction with complement -

1) using 10's complement, subtract.

72532 - 3250

```
minuend, M     =  72532
subtrahend N   =  03250
                  -----
                  69282
                  -----
```

```
9's complement =  99999
                  69282
                  -----
                  30717
```

10's Subtrahend Ka nikalta hai

9's complement of N = 99999 − 03250 = 96749

10's complement of N = 96750
Minuend M = + 72532
169282

if M > N, carry is discard.
∴ sum = 69282

3250 − 72532

M minuent = 3250
N Subtrahend = 72532

9's complement = 99999 − 72532 = 27467

10's comlement N = 27468
+ 3250
= 30718

Sum

if M < N, take 10's complement sum and place a negetive sign infront

So 9's complement =
$$\begin{array}{r} 99999 \\ -\ 30718 \\ \hline 69281 \end{array}$$

10's complement = 69282

Answer = −69282

Binary subtraction −

x = 1010100

y = 1000011

Perform x − y using 2's complement −

2's complement of y

$\boxed{\text{1's com}}$

$$\begin{array}{r} 1111111 \\ -\ 1000011 \\ \hline 0111100 \end{array}$$

0111100

1's comp + 1

$\boxed{\text{2's complement}}$ = 0111101

$$\begin{array}{r} 1010100 \\ +\ 0111101 \\ \hline \end{array}$$

Sum = $\boxed{1}$0010001

└→ discard

$\boxed{\begin{array}{l} \text{Signed Binary numbers} \\ 0 \rightarrow +\text{ve} \\ 1 \rightarrow -\text{ve} \end{array}}$

result = 0010001

0 1 1
1 0 1
1 0 0 0

2) Perform Y - X using 2's complement

X = 1010100

Y = 1000011

~~0101011~~
~~+ 1~~
~~0101100~~

2's complement of X = 0101011 + 1
= 0101100

  1000011
+ 0101100

  1101111

If MSB = 1, result is negetive and 2's complement of sum

Result = 0010001

0101011
0000001

0101100

Signed Binary Numbers -

we assume $0 \rightarrow +ve$, $1 \rightarrow -ve$ } Left most Bit

Q1
$$\begin{array}{l} 8421 \\ \hline 1011 \end{array} \text{ (unsigned)} = (11)_{10}$$

1011 (signed) $= (-3)_{10}$

$\rightarrow 1\times8 + 0\times4 + 2\times1 + 1\times1 = 11$

~~8x10+0x4+0x2x1~~

$\boxed{1}011$

$\rightarrow 1\times1$
$\rightarrow 1\times2$ $+ = (-3)_{10}$

yeh sirf sign btaega (Baaki Koi matlab nhi hai isse)

16+8+1

11001 (unsigned) $= (25)_{10}$

$\boxed{1}1001$ (signed $= (-9)_{10}$

$\rightarrow 1\times1$
$\rightarrow 1\times8$ $+ = 9$

$\rightarrow -ve$

Homework

Subtraction using 1's complement

* Practically we implement 2's complement in hardware desigen

Ques1 $\underbrace{(110101)_2}_{x} - \underbrace{(100101)_2}_{y}$

y's 1 complement = 011010

Add x & 1's complement of y

$$\begin{array}{r} 110101 \\ +011010 \\ \hline \boxed{1}001111 \end{array}$$

carry will be add in LSB (Last significant bit)

$$\begin{array}{r} 001111 \\ +1 \\ \hline 010000 \end{array}$$

Ans"

$\boxed{y - x}$

x's 1 complement = 001010.

Add y & x 1's complement.

$$\begin{array}{r} 100101 \\ 001010 \\ \hline \boxed{1}01111 \end{array}$$

only for 1

If no carry over then take 1 complement of sum and -ve sign.

1's complement of sum = 010000

12/09/23

$$\begin{array}{r} +6 \\ +13 \\ \hline +19 \end{array}$$

(ussaly 8 bits 'm kaam hota hai)

	128	64	32	16	8	4	2	1
6	0	0	0	0	0	1	1	0
13	0	0	0	0	1	1	0	1

$$\begin{array}{r} 0\,0\,0\,0\,0\,1\,1\,0 \\ +\ 0\,0\,0\,0\,1\,1\,0\,1 \\ \hline 0\,0\,0\,1\,0\,0\,1\,1 \end{array}$$

+6
-13

6 = 0000 0110
13 = 0000 1101

1's complement of 13 = 1111 0010

2's complement of = 1111 0011

Now
sum of 6 & 13 's complement is

$$\begin{array}{r} 0000\ 0110 \\ +\ 1111\ 0011 \\ \hline 1111\ 1001 \\ \hline \end{array}$$

→ 2 complement (-ve)

= 0000 0110 + 1

= 0000 0111 Result //

-6   6 = 0000 0110
13   13 = 0000 1101

1's complement of 6 = 1111 1001
2's complement of 6 = 1111 1010

Now sum of 2's complement of 6 and 13.

$$\begin{array}{r} 0000\ 1101 \\ +\ 1111\ 1010 \\ \hline 1\,0000\ 0111 \\ \hline \end{array} = (7)_{10}$$

discard

Result = 00000111 Ans//

Ques -6 + (-13)

6	0 0 0 0 0 1 1 0
13	0 0 0 0 1 1 0 1

1's complement of 6 = 1111 1001

2's complement of 6 = 1111 1010

1's complement of 13 = 1111 0010

2's complement of 13 = 1111 0011

sum =

```
   1 1 1 1       1
     1 1 1 1   1 0 1 0
 +   1 1 1 1   0 0 1 1
 ---------------------
   [1] 1 1 1 0   1 1 0 1
```

Carry is discarded

1110 1101

Now Result is -ve & 2's complement

$\boxed{-6+(-13)}$

$= 0001\ 0011$ Ans // /Result

Binary codes

* Maximum Information with limited no. of Bits.

Koi word baar Repeat ho rha hai to uske lie Ek particular code Braya jata hai

For Ex: ADGIFM = 001.

compressed meaning = Bits Kam hai.

Home-assignment

Perform $(-6)-(-13)$ with Signed Binary numbers using 2's complement

$$
\begin{array}{r}
\scriptstyle 1\,1\,1\,1\,1\,1 \\
1111\ 1010 \\
+\ 0000\ 1101 \\
\hline
\boxed{1}0000\ 0111
\end{array}
$$

Discard

0000 0111 Answer//.

13/09/23

Binary codes -

- It is a pattern of 0's & 1's
- circuit elements are being manipulated Economically for use in computers.
- Minimum number of bits required to code 2^n distinct quantities. = n
- Maximum number of bith required to code 2^n distinct quantities = NO Maximum limit.

- 2-bit code for $2^2 (= 4)$ quantities / Elements.
- 3 bit code for $2^3 (= 8)$ quantities / Elements
- 4 bit code for $2^4 (= 16)$ quantities / Elements.

Encoding (Bhejna) msge

decoding

colour coding

4 bit code

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

10-bit code

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 0 0 0 1 0
0 0 0 0 0 0 0 1 0 0
0 0 0 0 0 0 ɸ 0 0 0

{Apna new create kia hai}

Binary coded Decimal (BCD)

Decimal symbol	BCD digit
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1

$(9)_{10} = (1001)_2$

$(9)_{BCD} = (1001)_2$

$(12)_{10} = (1100)_2$

$(12)_{BCD} = (0001\ 0010)_2$

If a number has K elements then length of that BCD is 4K.

Maximum Information can be transferred.

$(256)_{BCD} = 4 \times k = 4 \times 3 = 12$

$\downarrow$

$(0010\ 0101\ 0110)_2$

BCD Addition -

Procedure

1) when the Binary sum ≤ 1001 (without a carry), the corresponding BCD digit is correct.

2) when the Binary sum is greater than ($\geq$) 1010, the Result is invalid. The addition of $(0110)_2$ $[6]_{10}$ to the Binary sum converts it

to the correct BCD digit and also produces a carry as required

$$\begin{array}{r} 4 \\ +5 \\ \hline 9 \\ \hline \end{array} \qquad \begin{array}{r} 0100 \\ 0101 \\ \hline 1001 \\ \hline \end{array} \longrightarrow \text{Rule } ①$$

$$\begin{array}{r} 4 \\ +8 \\ \hline 12 \end{array} \qquad \begin{array}{r} 0100 \\ 1000 \\ \hline 1100 \\ +0110 \\ \hline 10010 \\ \hline \end{array} \longrightarrow \text{Rule } ②$$

BCD digit code 0001 0010

BCD addition

$$\begin{array}{rcr} 184 & = & 0\,0\,0\,1\,1\,0\,0\,0\,0\,1\,0\,0 \\ 576 & = & 0\,1\,0\,1\,0\,1\,1\,1\,0\,1\,1\,0 \\ \hline & & 0\,1\,1\,0\,1\,1\,1\,1\,1\,0\,1\,0 \\ \hline \end{array}$$

wrong way.

184	0001	1000	0100
576	0101	0111	0110
	0110	1111	1010
			+0110
			10000

Wrong ×

Right way

184	0001	1000	0100
+576	0101	0111	0110
760	0111	10000	1010
		0110	0110
	0111	10110	0000
	7	6	0

Excess - 3 code

↳ BCD code + $\underbrace{0011}_{(3)_{10}}$

digit	BCD ~~digit~~ code	Excess - 3 code BCD + 0011
0	0 0 0 0	0011
1	0 0 0 1	0100
2	0 0 1 0	0101
3	0 0 1 1	0110
4	0 1 0 0	0111
5	0 1 0 1	1000
6	0 1 1 0	1001
7	0 1 1 1	1010
8	1 0 0 0	1011
9	1 0 0 1	1100

weighted code

$$\begin{array}{r} 0011 \\ 0011 \\ \hline 110 \end{array} \qquad \begin{array}{r} 0101 \\ 0100 \\ \hline 1010 \end{array} \qquad \begin{array}{r} 0110 \\ +0011 \\ \hline 1001 \end{array}$$

self complementary

weighted code ×

Grey codes –

$n = 1$ (2^1 code)	$n = 2$ (2^2 code)	$n = 3$ (2^3 code)
0	0 0	0 0 0
1	0 1	0 0 1
	1 1	0 1 1
	1 0	0 1 0
		1 1 0
		1 1 1
		1 0 1
		1 0 0

Binary to Grey

$(9)_{10} = (1001)_2$

MSB

Grey = (1101)

carry generate hua to discard kardo

Grey to Binary

Grey = (1011)

MSB

1101

Boolean Algebra and Logic gates -

- Mathematical methods that simplyfy circuits.
- To reduce overall cost of the desigen.
- Boolean algebra will help in optimizing circuit.

Table - postulates and theorems -

	(a)	(b)
Postulate 2	$x+0=x$	$x\cdot 1=x$
Postulate 5	$x+x'=1$	$x\cdot x'=0$
Theorem 1	$x+x=x$	$x\cdot x=x$
Theorem 2	$x+1=1$	$x\cdot 0=0$
Theorem 3, involution	$(x')'=x$	
Postulate 3, commutative	$x+y=y+x$	$x\cdot y=y\cdot x$
Theorem 4, associative	$x+(y+z)=(x+y)+z$	$x\cdot(y\cdot z)=(x\cdot y)\cdot z$
Postulate 4, distributive	$x(y+z)=xy+xz.$	$x+(y\cdot z)=(x+y)\cdot(x+z)$

Imp

- Theorem 5, De Morgan: $(x+y)' = x'y'$ | (b) $(x \cdot y)' = x' + y'$

Theorem 6, absorption: $x + xy = x$ | (b) $x \cdot (x+y) = x$

---

optimization → best decide karna.

$\oplus$ modulo-2-addition.

$+$ Binary addition.

$+ \xrightarrow{\text{Replace}} \cdot$ | $0 \longrightarrow 1$

$\cdot \longrightarrow +$ | $1 \longrightarrow 0$

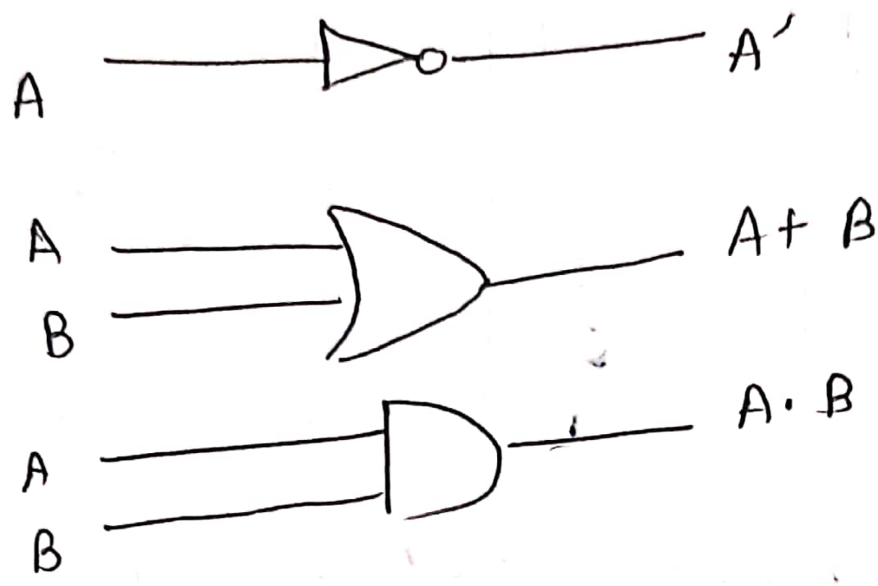
Boolean functions -

$F_1 = x + y'z$

$F_2 = x'y'z + x'yz + xy'$

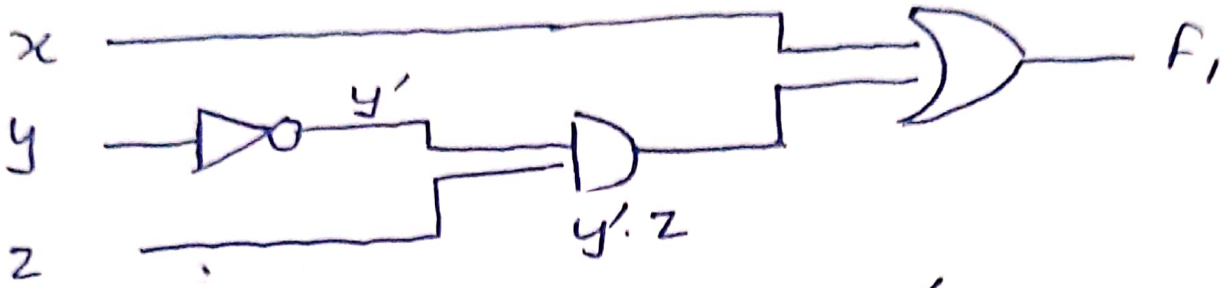
x	y	z	F_1	F_2
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	0
1	1	1	1	0

Logic symbols

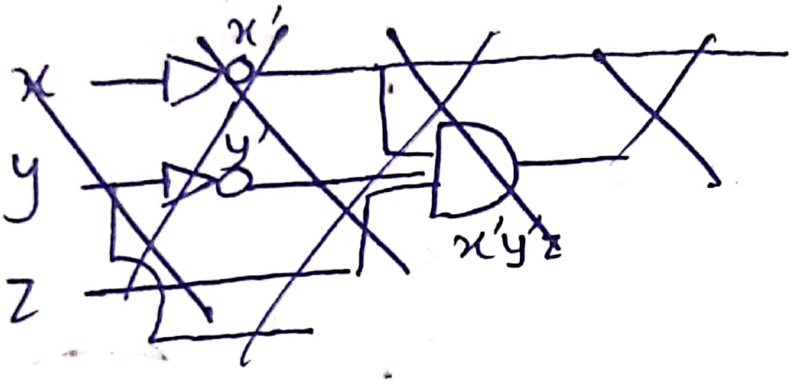


Circuit desigen

$F_1 = x + y'z$



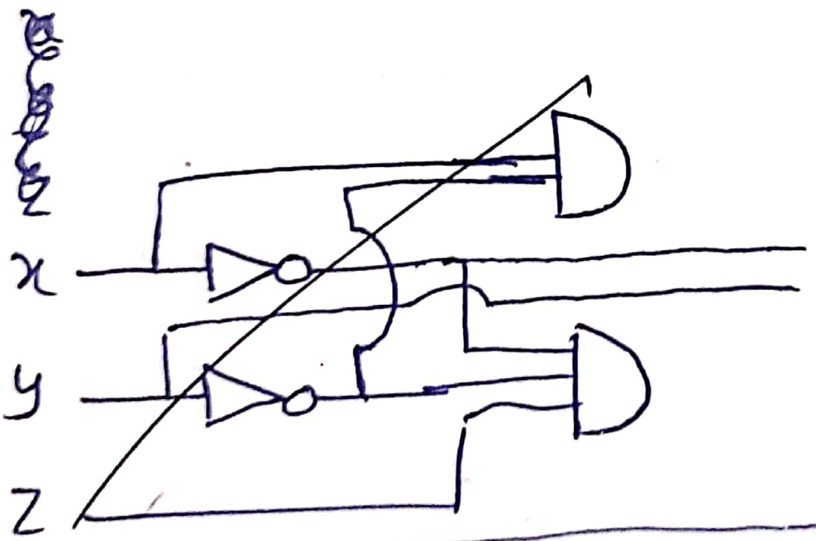
$F_2 = x'y'z + x'yz + xy'$



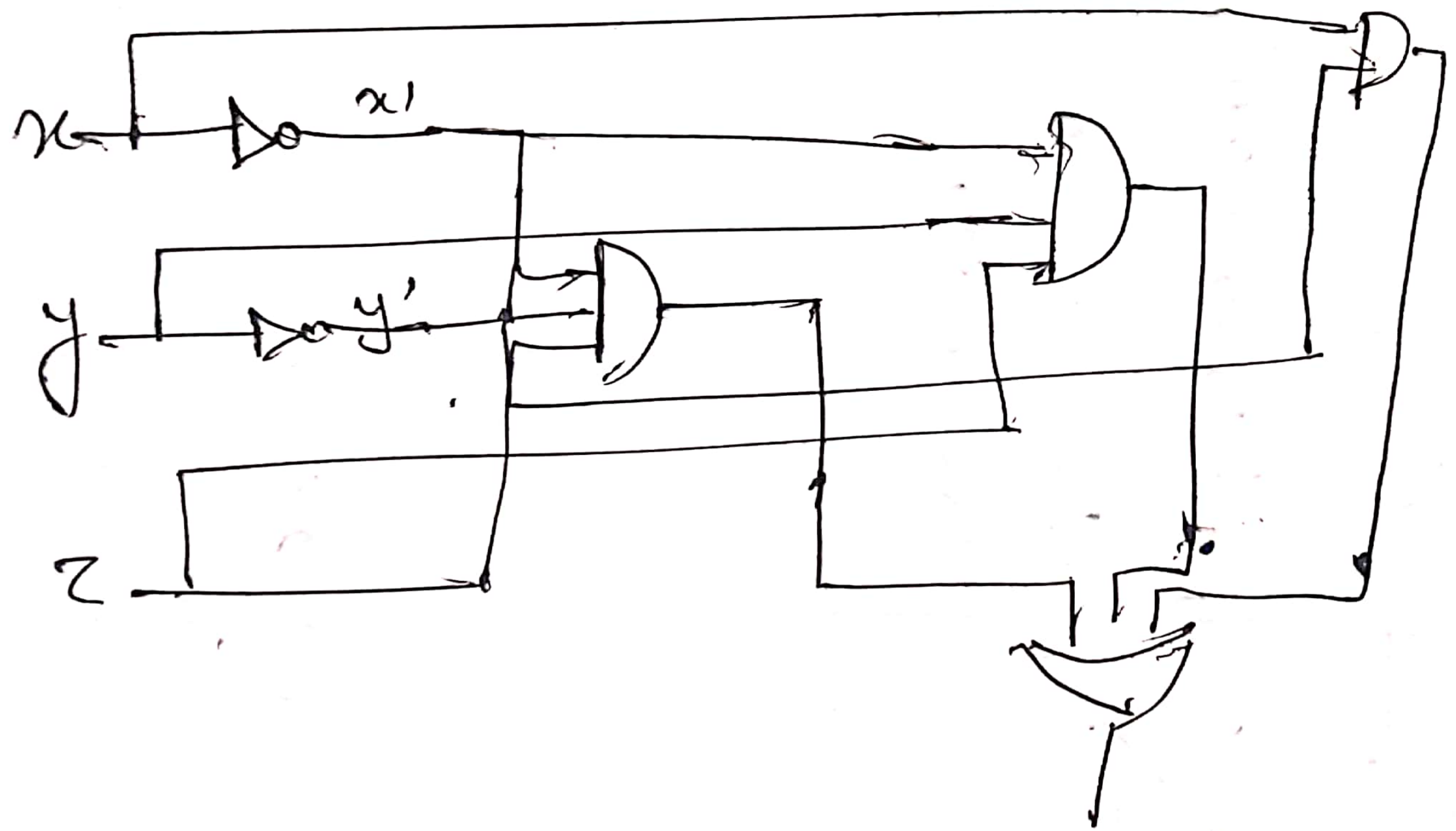
$F_2 = x'z(y'+y) + xy'$

$F_2^m = x'z + xy'$

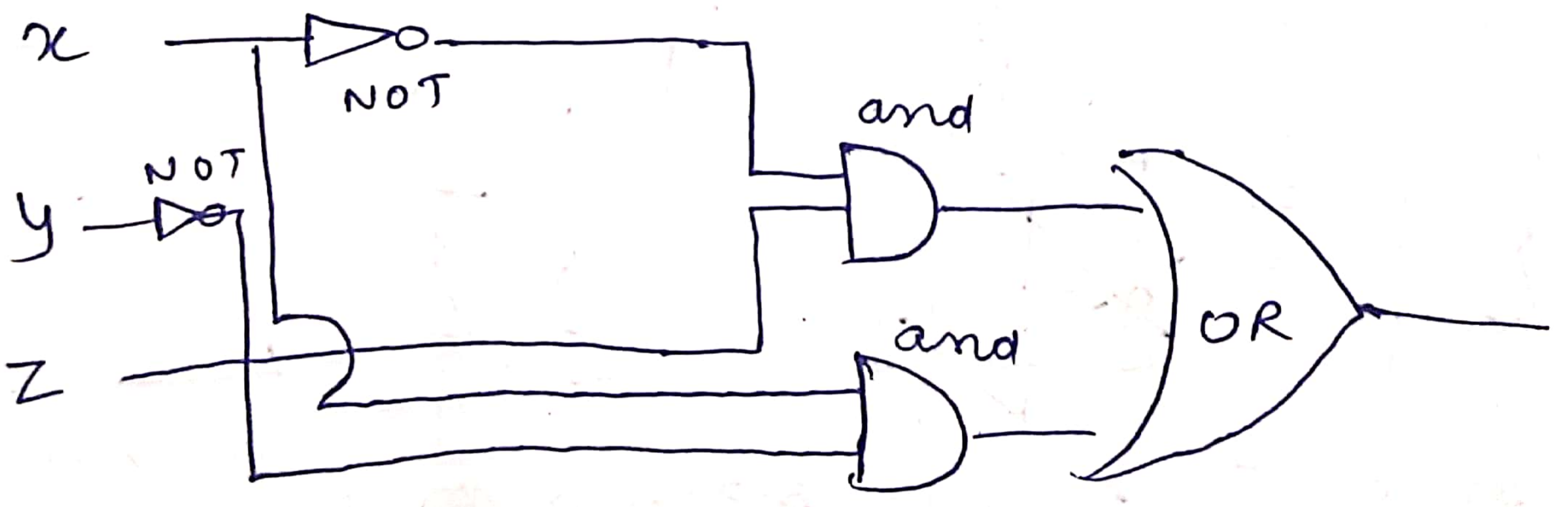
↳ modified



		terms	literal
x			
y	F_2	3	8
z	F_2^m	2	4



K-map



Ques Simplify the Boolean functions to a minimum no. of literal ?

(a) $x(x'+y) = xx' + xy$

$= 0 + xy$

$= xy$

Modified
term = 1
Literal = 2

(b) $x + x'y =$

$(x+x') + (x+y)$

$= 1 + (x+y)$

term = 2
Literal = 2

(c) $(x+y)(x+y') = x.x + x.y' + y.x + y.y'$

$= x + xy' + y.x + 0$

$= x + xy' + yx$

Modified	term	literal
	3	5

$= x + yy'$

$= x + 0$

Modified	term	literal
	1	1

25/09/23.

Canonical or standard forms-

A	B	C	Min terms (m)	Max terms (M)	
0	0	0	$A'B'C'$ (m_0)	$A+B+C$	M_0
0	0	1	$A'B'C$ (m_1)	$A+B+C'$	M_1
0	1	0	$A'BC'$ (m_2)	$A+B'+C$	M_2
0	1	1	$A'BC$ (m_3)	$A+B'+C'$	M_3
1	0	0	$AB'C'$ (m_4)	$A'+B+C$	M_4
1	0	1	$AB'C$ (m_5)	$A'+B+C'$	M_5
1	1	0	ABC' (m_6)	$A'+B'+C$	M_6
1	1	1	ABC (m_7)	$A'+B'+C'$	M_7

Product = 1 (minterm) $\rightarrow$ AND

Sum = 0 (Max term) $\rightarrow$ OR

f $\rightarrow$ Min terms

F $\rightarrow$ Max terms

x	y	z	f_1	F_2
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$f_1 = 1$

$F_2 = 0$

$\cancel{f_1 = x'y'z + xy'z' + xyz}$

$\cancel{= m_1 + m_4 + m_7 = \Sigma(m_1, m_4, m_7)}$

$\cancel{f_2 = (x+y+z)\cdot(x+y+z')\cdot(x+y'+z)\cdot(x'+y+z)}$

$f_1 = x'y'z + xy'z + xyz$

$\downarrow$

SOP's

$= m_1 + m_4 + m_7$

$= \Sigma(m_1, m_4, m_7)$

$= \Sigma(1, 4, 7)$

$$F_2 = (x+y+z)\cdot(x+y+z')(x+y'+z)\cdot(x'+y+z)$$

$$= M_0 \cdot M_1 \cdot M_2 \cdot M_4$$

$$= \pi(M_0, M_1, M_2, M_4)$$

$$= \pi(0,1,2,4)$$

$\rightarrow$ POS's

SOP $\rightarrow$ Sum of product $\longrightarrow f_1$

POS $\rightarrow$ Product of sum. $\longrightarrow F_2$

Q Express the Boolean function $f = A + B'C$ as a sum of minterms or SOP's

$$f = A + B'C$$

$$f = A\cdot(B+B')(C+C') + BC'(A+A')$$

$$f = (AB + AB')(C+C') + BC'A + A'BC'$$

$$= ABC + \underline{ABC'} + AB'C + \underline{ABC'} + BC'A + A'BC'$$

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

$= m_7 + m_6 + m_5 + m_4 + m_1$

$= \Sigma(1, 4, 5, 6, 7)$

$= \Sigma(m_1, m_4, m_5, m_6, m_7)$

<u>Ques</u> Express the Boolean function

$F = AB + A'C$

as a POS's on Product of Max terms.

using distributive law,

$x + yz = (x+y)(x+z)$

$F = \underbrace{AB}_{x} + \underbrace{A'C}_{yz}$

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

$= (\underbrace{A'}_{x} + \underbrace{A}_{y}\underbrace{B}_{z})(\underbrace{C}_{x} + \underbrace{A}_{y}\underbrace{B}_{z})$

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

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

$$(\underbrace{A'+B}_{x}+\underbrace{CC'}_{yz})(\underbrace{A+C}_{x}+\underbrace{BB'}_{yz})(\underbrace{B+C}_{x}+\underbrace{AA'}_{yz})$$

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

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

$$= M_4, M_5, M_0, M_2$$

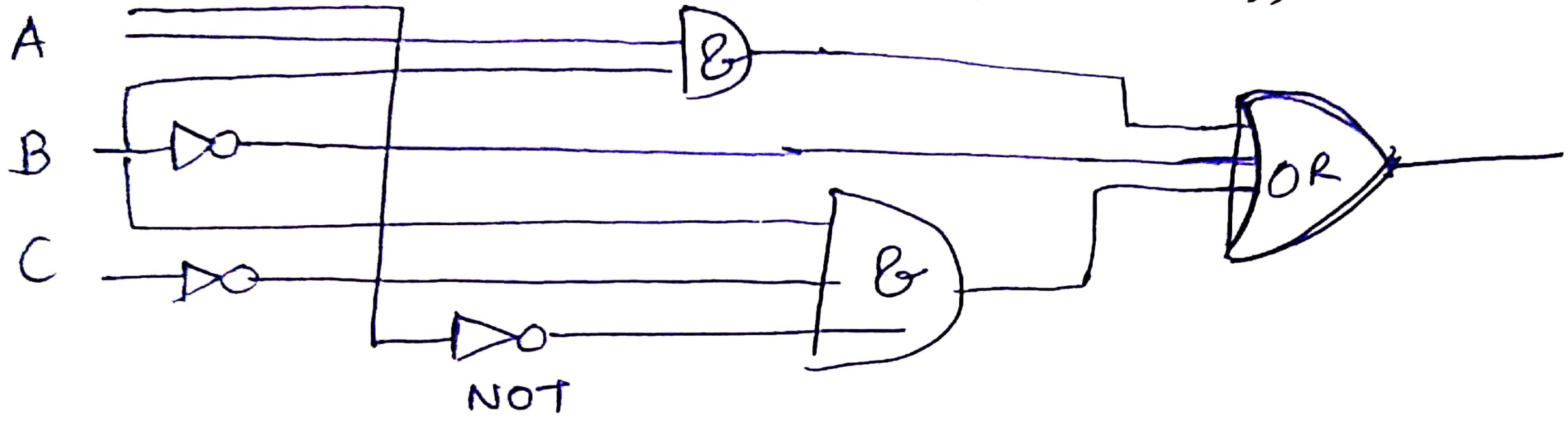
$$= \pi(M_0, M_2, M_4, M_5)$$

$$= \pi(0,2,4,5)$$

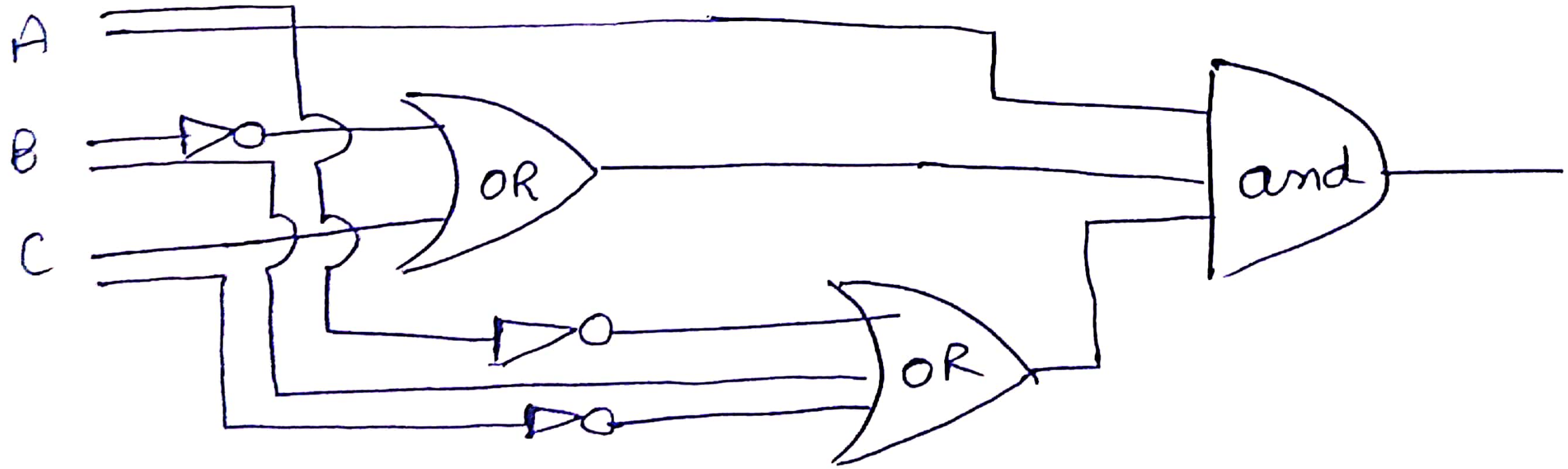
1) SOP's (ORing all minterms)

$$f = B' + AB + A'BC'$$

$f = B' + AB + A'BC'$ (Minterms)



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



Other Logic functions

no. of variables $= n$

No. of Boolean function $= 2^{2n}$

Let $n = 2$

Total functions $= 2^{2n} = 2^{2\times 2} = 16$

A	B	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

A	B	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

$F_1 = 0$

$F_2 = A \cdot B$

$F_3 = A \cdot B'$

$F_4 = A$

$F_5 = A'B$

$F_6 = B.$

$F_7 =$

$F_8 =$

$F_9 =$

$F_{10} =$

$F_{11} = B'$

$F_{12} = 1$

$F_{13} = A'$

$F_{14} =$

$F_{15} = (AB)'$

$F_{16} = 1$

$(A'+B')'$

$A'+B$

$A' \quad B$

$1 + 0 = 0$

$1 \quad 1 = 1$

$0 \quad 0 = 0$

$0 \quad 1 = 0$

$A \quad B'$

$A' \quad B'$

$0 \quad 0 = 0 \quad 1$

$0 \quad 1 = 1 \quad 0$

$1 \quad 0 = 1 \quad 0$

$1 \quad 1 = 0 \quad 1$

UNIT - 2

Gate level minimization ~

- To design a logic circuit.
- To optimize the designing of logic circuit.
- To Reduce the Boolean expression.

→ The map method

① Karnaugh map method (K-map)
② QM method.

Karnaugh map method {K-map}

2 variable K-map method

$\square$ single cell

No. of variables = 2
No. of cells = $2^2 = 4$ cells.

A \ B		B' 0	B 1
A'	0	0	1
A	1	2	3

A	B	
0	0	m_0
0	1	m_1
1	0	m_2
1	1	m_3

$m_0 = A'B'$ $m_2 = A \cdot B'$

$m_1 = A'B$ $m_3 = AB$

3- variable K-map method

No. of variables = 3

No. of cells = $2^3 = 8$

A \ BC	$B'C'$ 00	$B'C$ 01	BC 11	BC' 10
A' 0	0	1	3	2
A 1	4	5	7	6

$m_0 = A'B'C'$

A	B	C	
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3

4-variable K-map

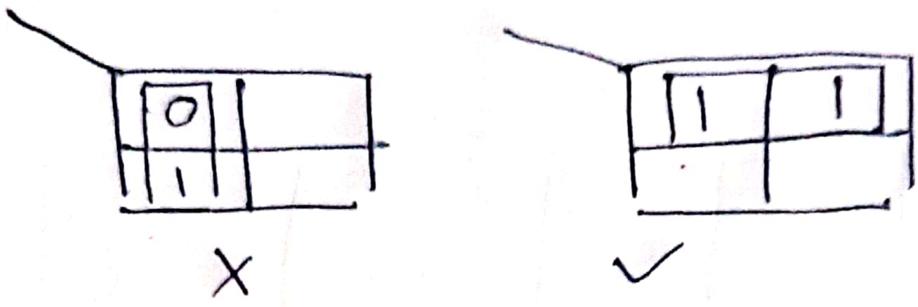
No. of variables = 4

No. of cells = $2^4 = 16$

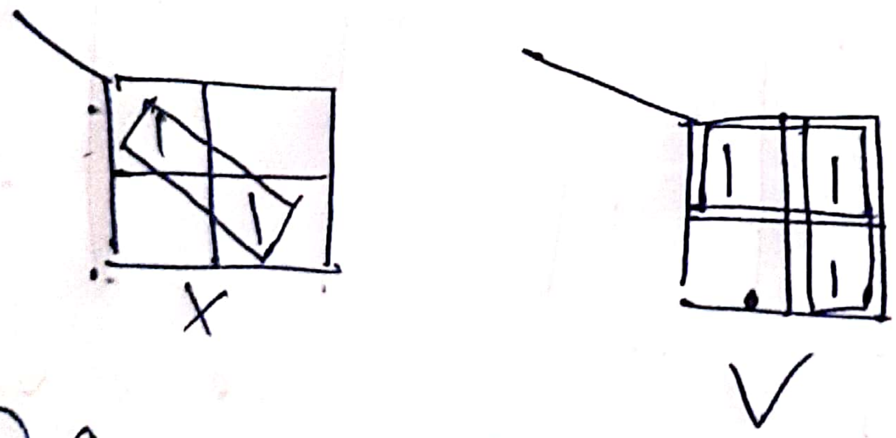
AB \ CD	$C'D'$ 00	$C'D$ 01	CD 11	CD' 10
$A'B'$ 00	0 m_0	1 m_1	3 m_3	2 m_2
$A'B$ 01	4 m_4	5 m_5	7 m_7	6 m_6
AB 11	12 m_{12}	13 m_{13}	15 m_{15}	14 m_{14}
AB' 10	8 m_8	9 m_9	11 m_{11}	10 m_{10}

Rules for minimization of SOP's

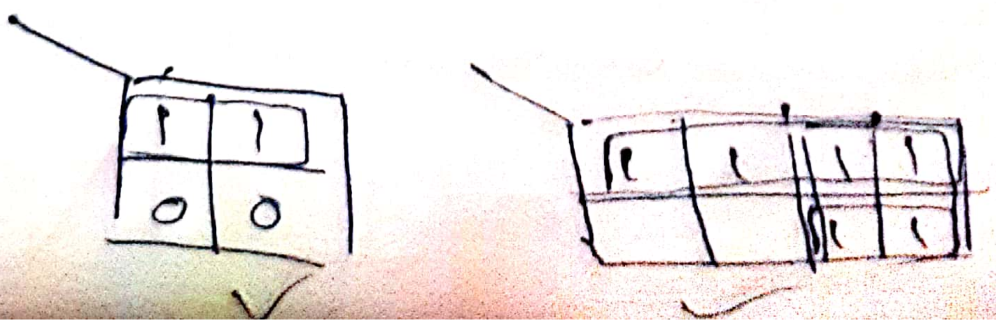
① Groups may not include any cell containing a "0"

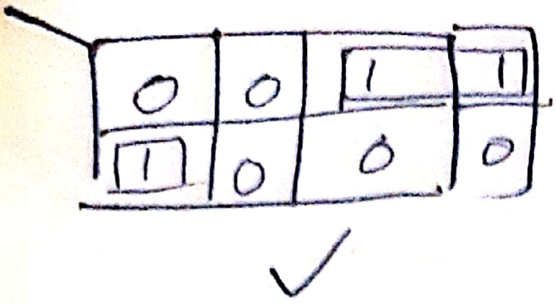


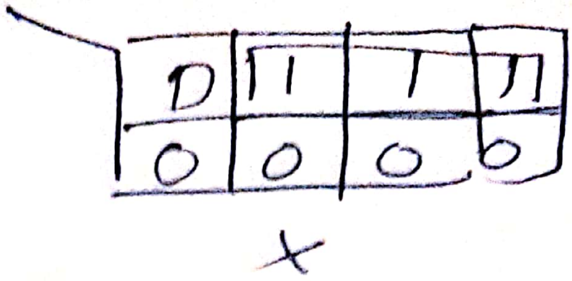
② Groups may be horizontal or vertical but not diagnol.



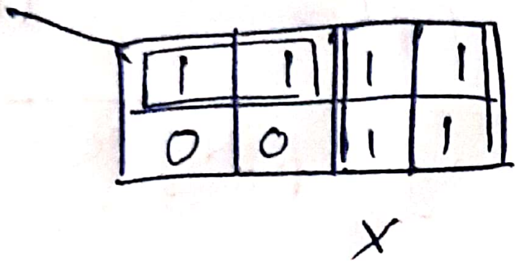
③ Groups must contain 1, 2, 4, 8 or in general 2^n cells.

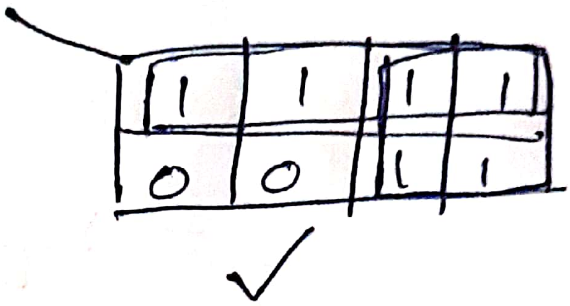




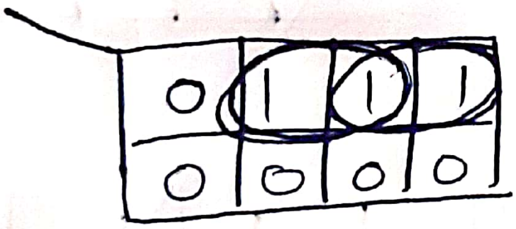


(4) Each group should be as large as possible

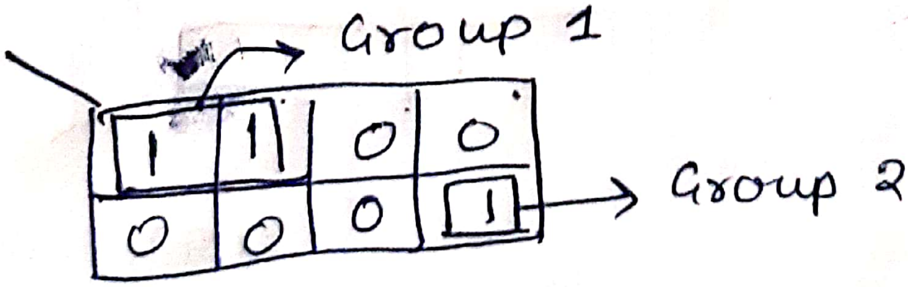




(5) Groups may overlap.



(6) Each cell containing a ~~one~~ '1' must be in atleast one group.



7. Groups may wrap around the corners the table

1	1	1	1
1	1		

1	1	0	1
1	0	0	1

Solved Examples -

Q1 Simplify the Boolean function -

$F(A, B, C) = \Sigma(2, 3, 4, 5)$

Range $= 0$ to $(2^n - 1)$

$n \rightarrow$ no. of variables

A \ BC	B'C' 00	B'C 01	BC 11	BC' 10
A' 0	[0]	[1]	1 [3]	1 [2]
A 1	1 [4]	1 [5]	[7]	[6]

f_1

L to R $\rightarrow$ adj. cells

f_2

<u>SoP</u>

$f_1 = A'BC + A'BC'$

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

$= A'B$

$f_2 = AB'C' + AB'C$

$f_2 = AB'(C' + C)$

$f_2 = AB'$

(jo change ho gya discard kro)

$F = f_1 + f_2$

$= A'B + AB'$

Ques $F(x, y, z) = \Sigma(3, 4, 6, 7)$

x \ yz	y'z' 00	y'z 01	yz 11	yz' 10
x'	(0)	(1)	1 (3)	(2)
x	1 (4)	(5)	1 (7)	1 (6)

$\longrightarrow f_2$

f_1

$f_1 = yz \qquad f_2 = xz'$

$f = f_1 + f_2 = yz + xz'$

4-variable

Ques Simplify the function

$F(w, x, y, z) = \Sigma(0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$

wx \ yz	y'z' 00	y'z 01	yz 11	yz' 10
w'x' 00	1 (0)	1 (1)	0 (3)	1 (2)
w'x 01	1 (4)	1 (5)	0 (7)	1 (6)
wx 11	1 (12)	1 (13)	0 (15)	1 (14)
wx' 10	1 (8)	1 (9)	0 (11)	0 (10)

f_1 f_2 f_3

$f_1 = y'$

$f_2 = z w'$

$f_3 = z' x$

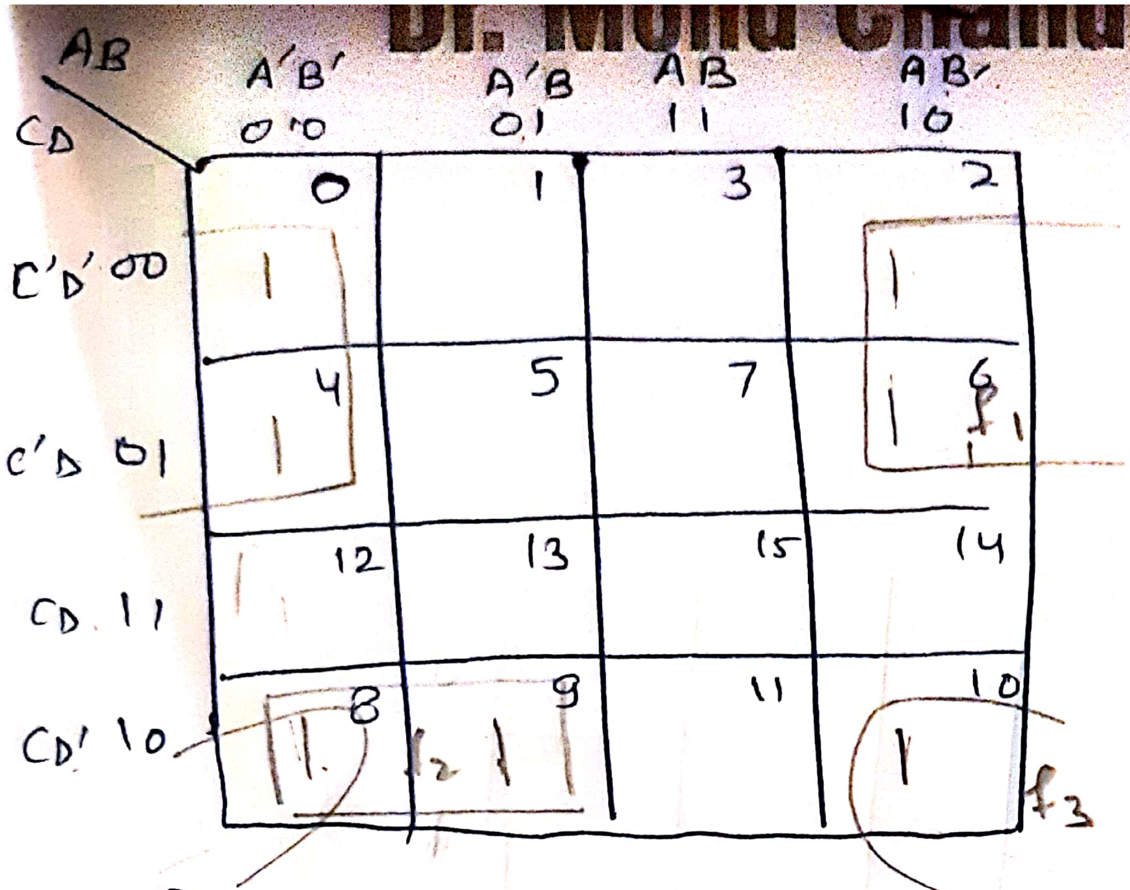
$$f = f_1 + f_2 + f_3$$
$$= y' + z' w' + z' x$$
$$= y' + z' (w' + x)$$

Q2 Simplify the Boolean function -

$F = A'B'C' + B'CD' + A'BCD' + AB'C'$

using K-map?

Scanned with OKEN Scanner



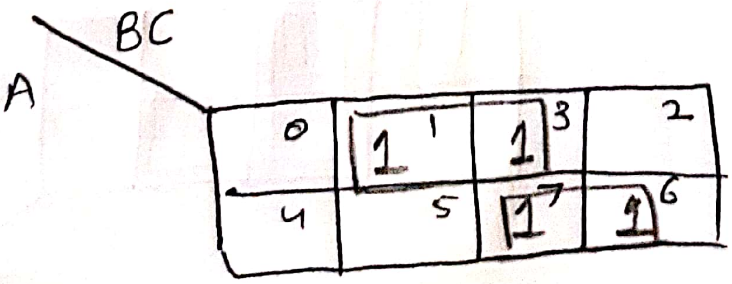
$f_1 = B'C'$ $\quad$ $f_2 = CD'A'$

$f_3 = B'CD'$

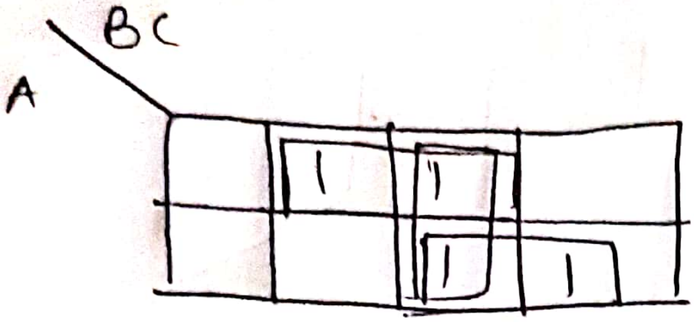
$f = B'C' + B'CD' + CD'A'$

Essential prime ~~Implications~~ Implicants - (non-overlapping groups)

→ Prime Implicants - (overlapped groups)



Essential Prime Implicants



Prime Implicants

5-variable K-map

A = 0

DE \ BC	B'E' 00	B'C 01	B'C 11	BC' 10
DE 00	0	1	3	2
D'E 01	4	5	7	6
DE 11	12	13	15	14
DE' 10	8	9	11	10

A = 1

DE \ BC	B'C' 00	B'C 01	BC 11	BC' 10
D'E' 00	16	17	19	18
D'E 01	20	21	23	22
DE 11	28	29	31	30
DE' 10	24	25	27	26

17/10/23

Q1 Simplify the Boolean function

$F(A, B, C, D, E) = \Sigma m(0, 5, 6, 8, 9, 10, 11, 16, 20, 24, 25, 26, 27)$.

A = 0

DE \ BC	B'C' 00	B'C 01	BC 11	BC' 10
D'E' 00	0: 1 F_2	1	3	2
D'E 01	4	5: 1 f_3	7	6: 1 f_4
DE 11	12	13	15	14
DE' 10	8: 1	9: 1	11: 1	10: 1

F_1

A = 1

DE \ BC	B'C' 00	B'C 01	BC 11	BC' 10
D'E' 00	16: 1 F_5	17	19	18
D'E 01	20: 1	21	23	22
DE 11	28	29	31	30
DE' 10	24: 1	25: 1	27: 1	26: 1

$F_1 = A'DE'$

$F_2 = B'C'D'E'$

$F_3 = A'B'CD'E$

$F_4 = A'DE'B'C$

$F_5 = AB'D'E'$

$F = F_1 + F_2 + F_3 + F_4 + F_5$

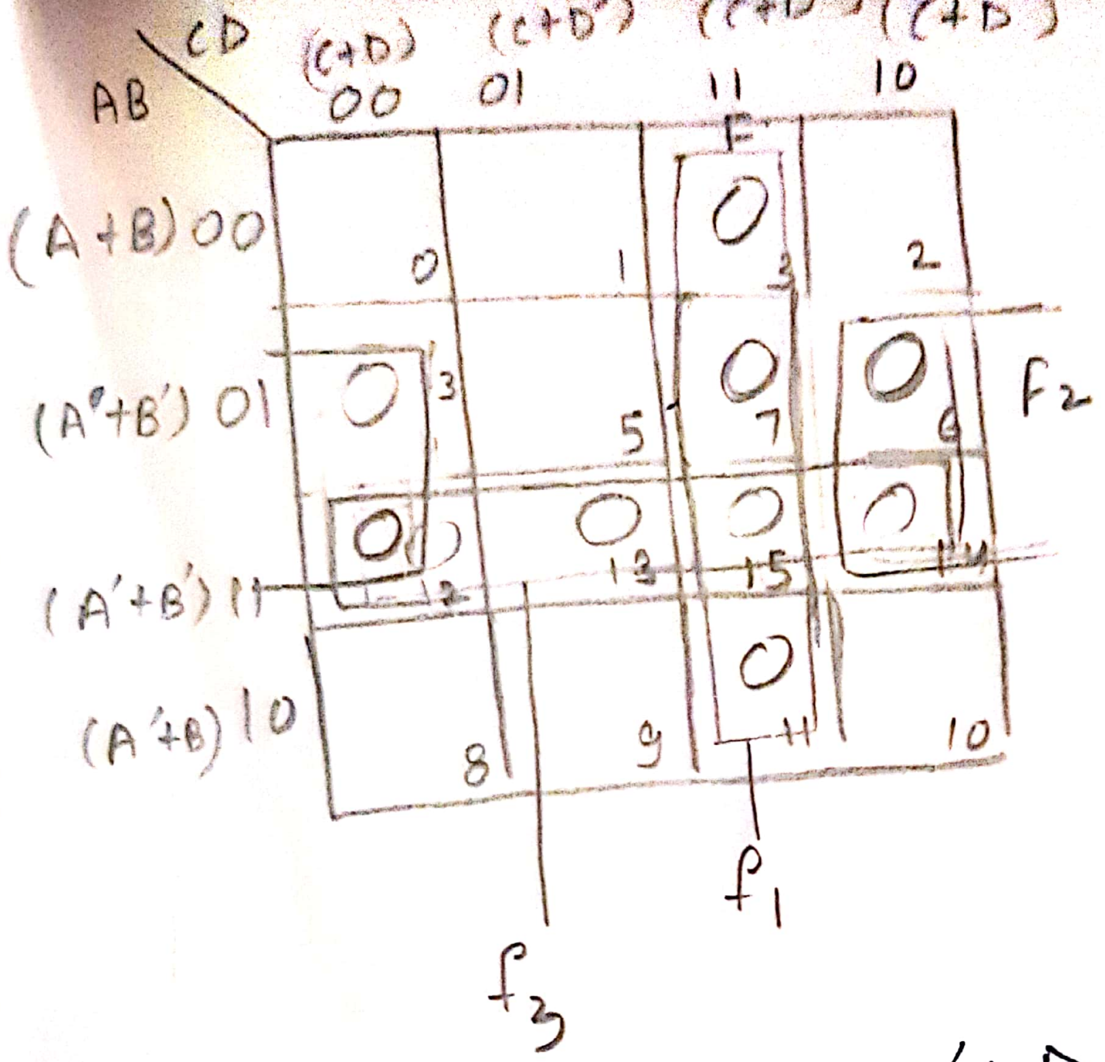
$= DE' + B'C'D'E' + A'B'CD'E + A'DE'B'C$
$+ AB'D'E'$

$\boxed{\begin{array}{l} \Sigma - SOP \\ \pi - POS \end{array}}$

POS Simplification

Q1 $F(A,B,C,D) = \Sigma(0,1,\cancel{8}2,5,8,9,10)$

$F(A,B,C,D) = \pi(3,4,6,7,11,12,13,14,15)$



$$f_2 = B' + D$$

$$f_1 = C' + D'$$

$$f_3 = A' + B'$$

$$F = (f_1)(f_2)(f_3)$$

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