10/09/2020
Thursday

Module-2

Boolen Algebra and Logic Gates :

- * Closure Law
- * Commutative Law
- * Associative Law
- * Distributive Law
- * Inverse Law
- * Identity Law

} Basic Rules of Boolean Algebra.

x	y	$x \cdot y$	$x+y$	x'
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

} Proof for postulates 1, 2, 3, 5, 6.

* $x \cdot (y+z) = x \cdot y + x \cdot z$ [Distributive Law]

Prove by drawing truth table with all eight combinations of x, y and z.

↓

Proof for postulate 4. (Refer pdf)

~~1) Theorem~~ 1 (a) : ~~$x+x = x$~~

~~$x+x = (x+x) \cdot 1$~~

~~$= (x+x)(x+x')$~~

~~$= x + x \cdot x'$~~

~~$=$~~

Theorems and Postulates :

(Huntington)

Postulates :

1. a) Closure w.r.t the operator $+$.
   b) Closure w.r.t the operator $\cdot$.
2. a) An identity element w.r.t $+$, designated by 0:
   $x+0 = 0+x = x$.
   b) An identity element w.r.t $\cdot$, designated by 1:
   $x \cdot 1 = 1 \cdot x = x$.
3. a) Commutative w.r.t $+$: $x+y = y+x$.
   b) Commutative w.r.t $\cdot$: $x \cdot y = y \cdot x$.
4. a) $\cdot$ is distributed over $+$: $x \cdot (y+z) = (x \cdot y) + (x \cdot z)$.
   b) $+$ is distributed over $\cdot$: $x + (y \cdot z) = (x+y) \cdot (x+z)$.
   [Only for Boolean algebra]

~~Proofs~~

5. For every element $x \in B$, there exists an element $x' \in B$ (called the complement of x) such that
(a) $x + x' = 1$ and (b) $x \cdot x' = 0$.

6. There exists at least two elements $x, y \in B$ such that $x \neq y$.

* Boolen has no additive & multiplicative inverse. Therefore, no subtraction or division operators.

* Huntington postulates do not have associative law but it is valid ~~for~~ Boolean algebra.

* Postulate 5 is not available in ordinary algebra.

Basic Theorems :

1. Theorem 1(a) : $x + x = x$.

Refer Duality Principle!

$$x + x = (x + x) \cdot 1$$
$$= (x+x)(x+x')$$
$$= x + xx'$$
$$= x + 0$$
$$= \underline{x}$$

Theorem 1(b) : $x \cdot x = x$.

$$x \cdot x = xx + 0$$
$$= xx + xx'$$
$$= x(x + x')$$
$$= x \cdot 1$$
$$= \underline{x}$$

~~(Dual)~~ ~~(1a is the~~
(1b is the dual of 1a)

2. Theorem 2(a) : $x + 1 = 1$.

$$x + 1 = 1 \cdot (x+1)$$
$$= (x+x')(x+1)$$
$$= x + x' \cdot 1$$
$$= x + x'$$
$$= \underline{1}$$

Theorem 2(b) : $x \cdot 0 = 0$ by duality.

3. Theorem 3 : $(x')' = x$

$$x + x' = 1 \quad , \quad x \cdot x' = 0 .$$

Complement of x' is x and is also $(x')'$.
Therefore, since the complement is unique, $(x')' = x$.

6. Theorem 6(a) : $x + xy = x$

$$x+xy = x\cdot 1 + xy$$
$$= x(1+y)$$
$$= x(y+1)$$
$$= x\cdot 1$$
$$= x$$

Theorem 6(b) : $x(x+y) = x$ by duality.

~~Proof~~

4. Theorem 4(a) : $x+(y+z) = (x+y)+z$.

Theorem 4(b) : $x(yz) = (xy)z$.

5. Theorem 5(a) : $(x+y)' = x'y'$.

Theorem 5(b) : $(xy)' = x'+y'$

* Theorems can be proven using truth tables of all possible combinations.

* Algebraic proof of theorems 4 and 5 are long. But, can be proved using truth tables. (Refer pdf)

Postulates and theorems of Boolean Algebra

- Postulate 2 a) $x+0 = x$ b) $x\cdot 1 = x$
- Postulate 5 a) $x+x' = 1$ b) $x\cdot x' = 0$
- Theorem 1 a) $x+x = x$ b) $x\cdot x = x$
- Theorem 2 a) $x+1 = 1$ b) $x\cdot 0 = 0$
- Theorem 3, involution $(x')' = x$
- Postulate 3, commutative a) $x+y = y+x$ b) $xy = yx$
- Theorem 4, associative a) $x+(y+z) = (x+y)+z$ b) $x(yz) = (xy)z$
- Postulate 4, distributive a) $x(y+z) = xy+xz$ b) $x+yz = (x+y)(x+z)$
- Theorem 5, DeMorgan a) $(x+y)' = x'y'$ b) $(xy)' = x'+y'$
- Theorem 6, absorption a) $x+xy = x$ b) $x(x+y) = x$

Operator Precedence

1. Parantheses – ( )
2. NOT – '
3. AND – ·
4. OR – +

(Refer pdf for Venn Diagram, Boolean Functions and Algebraic Manipulation)

Problems

1. Simplify the following Boolean functions to a minimum no of literals.

i) $x+x'y$

Ans. $(x+x')(x+y) = 1.(x+y) = \underline{x+y}$

ii) $x(x'+y)$

Ans. $xx'+xy = 0+xy = \underline{xy}$

iii) $x'y'z+x'yz+xy'$

Ans. $x'z(y'+y)+xy' = \underline{x'z+xy'}$

iv) $xy+x'z+yz$

Ans. $xy+x'z+yz(x+x') = xy+x'z+xyz+x'yz$

$= xy(1+z) + \cancel{x'z(y+y)}\ x'z(1+y)$

$= \underline{xy+x'z}$

v) $(x+y)(x'+z)(y+z)$

Ans. $\underline{(x+y)(x'+z)}$ by duality from function (iv).

(Refer pdf for Complement of a function.)

2. Find the complement of the functions $F_1 = x'yz'+x'y'z$ and $F_2 = x(y'z'+yz)$.

Ans. $F_1' = (x'yz'+x'y'z)' = (x'yz')'.(x'y'z)'$

$= \underline{(x+y'+z)(x+y+z')}$

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

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

$= \underline{x'+(y+z)(y'+z')}$

* Dual of the function and complement each literal.
[AND ↔ OR , 1 ↔ 0]

3. Find the complement of the above functions F_1 and F_2 by taking their duals and complementing each literal.

Ans. Dual of $F_1 = (x'+y+z')(x'+y'+z)$

Complement each literal $\Rightarrow (x+y'+z)(x+y+z') = F_1'$.

Dual of $F_2 = x + (y'+z')(y+z)$

Complement each literal $\Rightarrow x' + (y+z)(y'+z') = F_2'$.

Minterms and Maxterms

x	y	z	Term	Designation	Term	Designation
0	0	0	$x'y'z'$	m_0	$x+y+z$	M_0
0	0	1	$x'y'z$	m_4	$x+y+z'$	M_1
0	1	0	$x'yz'$	m_2	$x+y'+z$	M_2
0	1	1	$x'yz$	m_3	$x+y'+z'$	M_3
1	0	0	$xy'z'$	m_4	$x'+y+z$	M_4
1	0	1	$xy'z$	m_5	$x'+y+z'$	M_5
1	1	0	xyz'	m_6	$x'+y'+z$	M_6
1	1	1	xyz	m_7	$x'+y'+z'$	M_7

Minterm - Standard Product

Maxterm - Standard Sum

Functions of Three Variables

x	y	z	Function F_1	Function 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

$\therefore f_1 = x'y'z + xy'z' + xyz = m_1 + m_4 + m_7$

$\therefore f_2 = x'yz + xy'z + xyz' + xyz$

$= m_3 + m_5 + m_6 + m_7$

* Any ~~Boolen~~ Boolean function can be expressed as a sum of minterms.

To find complement :-

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

If we take the complement of f_1', we obtain f_1:

$f_1 = (x+y+z)(x+y'+z)(x+y'+z')(x'+y+z')(x'+y'+z)$

$= M_0 \cdot M_2 \cdot M_3 \cdot M_5 \cdot M_6$

Similarly,

$f_2 = (x+y+z)(x+y+z')(x+y'+z)\overset{x'+y+z}{(\quad)}$

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

$= M_0 M_1 M_2 M_4$

* Any Boolean function can be expressed as a product of maxterms.

* Canonical form : A ~~product~~ Boolean function that can be expressed as a product of maxterms and$^{\text{or}}$ a sum of minterms is called Canonical form,

1. Express the Boolean function $F = A + B'C$ in a sum of minterms.

Ans. F has 3 variables A, B and C.

A is missing 2 variables.

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

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

$= ABC + ABC' + AB'C + AB'C'$

$B'C$ is missing 1 variable.

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

$\therefore F = A + B'C$

$= ABC + ABC' + AB'C + AB'C' +$

$AB'C + A'B'C$

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

Canonical form

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

$F(A,B,C) = \sum(1,4,5,6,7)$

* To express as product of maxterms, use distributive law - $x + yz = (x+y)(x+z)$.

2. Express the Boolean function $F = xy + x'z$ in a product of maxterm form.

Ans. $F = xy + x'z = (xy + x')(xy + z)$

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

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

Each OR term is missing one variable.

$x' + y = x' + y + zz' = (x' + y + z)(x' + y + z')$

$x + z = x + z + yy' = (x + z + y)(x + z + y')$

$y + z = y + z + xx' = (y + z + x)(y + z + x')$

$\therefore F = (x + y + z)(x + y' + z)(x' + y + z)(x' + y + z')$

$= M_0 M_2 M_4 M_5$

$F(x,y,z) = \Pi(0,2,4,5)$

Canonical form.

Conversion of minterms to maxterms and vice versa.

(OR)

Conversion between Canonical Forms.

* A	B	C	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

$F(A,B,C) = \sum(1,4,5,6,7)$

$F'(A,B,C) = \sum(0,2,3) = m_0 + m_2 + m_3$

$\therefore F = (m_0+m_2+m_3)' = m_0' \cdot m_2' \cdot m_3'$

$= M_0 M_2 M_3 = \pi(0,2,3)$

ie, $\underline{\sum(1,4,5,6,7)} = \overline{\pi(0,2,3)}$

Standard Forms

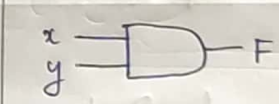
1. Sum of products :-

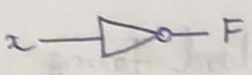
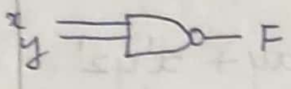
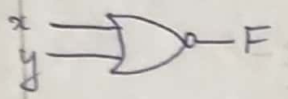
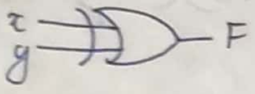
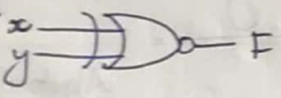
$F_1 = y' + xy + x'yz'$

2. Product of sums :-

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

Digital Logic Gates

Name	Graphic symbol	Algebraic function	Truth table
AND		$F = xy$	
OR		$F = x+y$	

Inverter (NOT)		$F = x'$	:
Buffer		$F = x$	:
NAND		$F = (xy)'$	:
NOR		$F = (x+y)'$	:
Exclusive-OR (XOR)		$F = xy' + x'y$ $= x \oplus y$	x y | F 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 0
Exclusive-NOR or equivalence		$F = xy + x'y'$ $= x \odot y$	x y | F 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 1

Simplification of Boolean Expression

1. **Karnaugh Map or K MAP :-**

(K-Map)

*

m_0	m_1
m_2	m_3

Two-variable map

x \ y	0	1 (y)
0	$x'y'$	$x'y$
x 1	xy'	xy

*

m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6

Three-variable map

x \ yz	00	01	11 (y)	10 (y)
0	$x'y'z'$	$x'y'z$	$x'yz$	$x'yz'$
x 1	$xy'z'$	$xy'z$	xyz	xyz'

(Similar to Gray code) ① (Only 1 bit has to be changed adjacently.)

$$m_5 + m_7 = xy'z + xyz$$
$$= xz(y+y')$$
$$= \underline{xz}$$

$m_1 + m_3 = x'z$ $m_1 + m_5 = y'z$

$m_0 + m_1 = x'y'$ $m_2 + m_6 = yz'$

$m_0 + m_2 = x'z'$

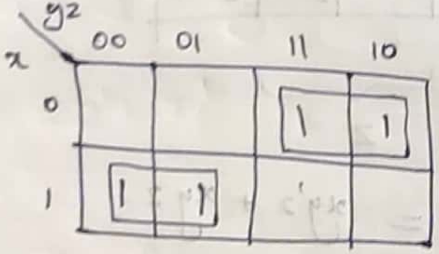
$m_4 + m_6 = xz'$

* When adjacent minterms are added (terms at the end too), ~~two variabl~~ a variable will be ~~elem~~ eliminated. [Those with change of one bit]

① Simplify the Boolean Function.

$F(x,y,z) = \Sigma(2,3,4,5)$

Ans.



x \ yz	00	01	11	10
0			1	1
1	1	1		

$x'yz + x'yz' = x'y$

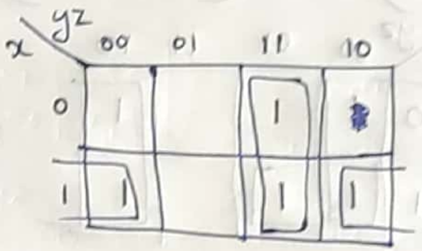
$m_4 + m_5 = xy'$ (~~Sine~~ since z-bit is changed)

$\therefore F = \underline{x'y + xy'}$

② Simplify the Boolean function.

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

Ans.



x \ yz	00	01	11	10
0			1	
1	1		1	1

~~$F = x'yz + xy'z' +$~~

$m_3 + m_7 = yz$

$m_4 + m_6 = xz'$

$\therefore F = \underline{xz' + yz}$

* When four adjacent minterms are combined, two variables will be eliminated.

$m_0 + m_2 + m_4 + m_6 = z'$

③ Simplify the Boolean function.

$F(x,y,z) = \Sigma(0,2,4,5,6)$

Ans.

x \ yz	00	01	11	10
0	1			1
1	1	1		1

$m_0 + m_2 + m_4 + m_6 = z'$

$m_4 + m_5 = xy'$

$\therefore F = z' + xy'$

④ Given the following Boolean function :

$F = A'C + A'B + AB'C + BC$.

a.) Express it in sum of minterms.

b.) Find the minimal sum of products expression.

Ans. a.) $F = A'BC + A'B'C + A'BC +$
$A'BC' + AB'C + ABC + A'BC$

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

$= m_3 + m_1 + m_2 + m_5 + m_7$

$= \Sigma(1,2,3,5,7)$

b.)

x \ yz	00	01	11	10
0		1	1	1
1		1	1	

$m_1 + m_3 + m_5 + m_7 = z$

$m_2 + m_3 = x'y$

$\therefore F = z + x'y$, which is the required expression.

Four Variable K-Map

~~m_0~~	~~m_1~~	~~m_3~~	~~m_2~~
~~m_4~~	~~m_5~~	~~m_7~~	~~m_6~~
~~m_{12}~~	~~m_{13}~~	~~m_{15}~~	~~m_{10}~~
~~m_{12}~~	~~m_{13}~~		

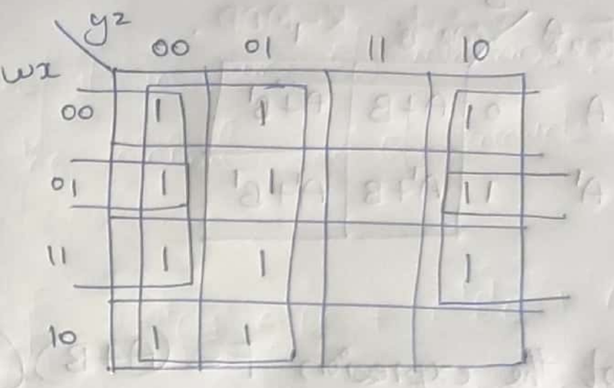
m_0	m_1	m_3	m_2
m_4	m_5	m_7	m_6
m_{12}	m_{13}	m_{15}	m_{14}
m_8	m_9	m_{11}	m_{10}

wx \ yz	00	01	11	10
00	$w'x'y'z'$	$w'x'y'z$	$w'x'yz$	$w'x'yz'$
01	$w'xy'z'$	$w'xy'z$	$w'xyz$	$w'xyz'$
11	$wxy'z'$	$wxy'z$	$wxyz$	$xwyz'$
10	$wx'y'z'$	$wx'y'z$	$wx'yz$	$wx'yz'$

Simplify the Boolean function.

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

Ans. On next page.



wx \ yz	00	01	11	10
00	1	1		1
01	1	1		1
11	1	1		1
10	1	1		

8, 4, 2

Check in this order!

$m_0 + m_1 + m_4 + m_5 + m_{12} + m_{13} + m_8 + m_9$

$= y'$

$m_2 + m_6 + m_0 + m_4 = w'z'$

$m_6 + m_{14} + m_4 + m_{12} = xz'$

$\therefore F = y' + w'z' + xz'$

Product of sums simplification. (POS)

A function in two variable (A, B) has four possible maxterm, $(A+B)$, $(A+\bar{B})$, $(\bar{A}+B)$, $(\bar{A}+\bar{B})$.

	A \ B	B 0	B' 1
A	0	A+B	A+B'
A'	1	A'+B	A'+B'

1. Plot the expression F = (A+B)(A'+B)(A'+B') on the K-MAP.

Ans.

A \ B	0	1
0	0	1
1	0	0

2. Reduce the expression F = (A+B)(A+B')(A'+B')

Ans.

A \ B	0	1
0	0	0
1		0

= AB'

~~(or)~~

M₀M₁ = A

(or)

M₁M₃ = B'

∴ F = AB'

~~3. Reduce the expression F = (A+B)~~

3. Simplify the following Boolean function in

a) SOP b) POS

F(A,B,C) = Σ(1,2,4,6,7)

Ans. a)

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

∴ F = AC' + AB + BC' + A'B'C

b) In Product of maxterms it is represented as π(0,3,5)

3-Variable K-MAP

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

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

* Four Variable K-MAP.

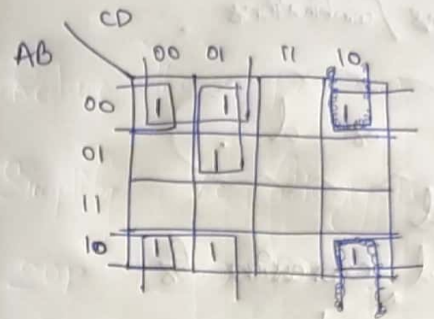
M_0	M_1	M_3	M_2
M_4	M_5	M_7	M_6
M_{12}	M_{13}	M_{15}	M_{14}
M_8	M_9	M_{11}	M_{10}

AB \ CD	00	01	11	10
00	$A+B+C+D$	$A+B+C+D'$	$A+B+C'+D'$	$A+B+C'+D$
01	$A+B'+C+D$	$A+B'+C+D'$	$A+B'+C'+D'$	$A+B'+C'+D$
11	$A'+B'+C+D$	$A'+B'+C+D'$	$A'+B'+C'+D'$	$A'+B'+C'+D$
10	$A'+B+C+D$	$A'+B+C+D'$	$A'+B+C'+D'$	$A'+B+C'+D$

1. Simplify the following Boolean function in
a.) SOP and b.) POS

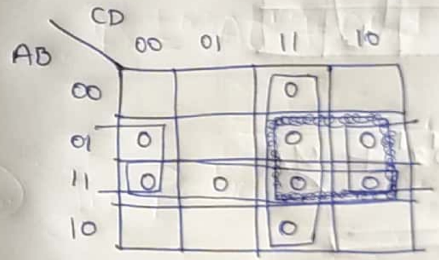
$F(A,B,C,D) = \Sigma(0,1,2,5,8,9,10)$.

Ans. a.) on next page



$\therefore F(A,B,C,D) = B'C' + A'C'D + B'D'$

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



$\therefore F(A,B,C,D) = \cancel{(B'+C')}(C'+D')(A'+B')(B'+D)$

* Don't Care Conditions.

The combination for which the values of the expression are not specified are called ~~Donts~~

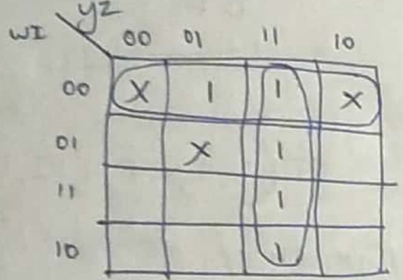
don't care conditions / combinations.

Symbol - X or d.

1) Find the simplified (minimised) expression for

a) $F(w,x,y,z) = \Sigma(1, 3, 7, 11, 15)$ and the don't care conditions, $d(w,x,y,z) = \Sigma(0, 2, 5)$.

Ans.

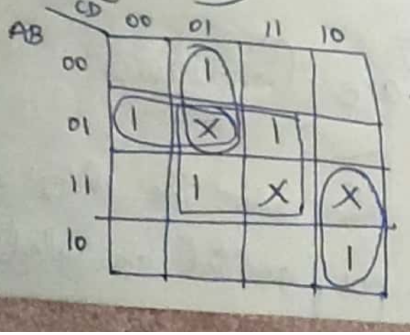


wx \ yz	00	01	11	10
00	X	1	1	X
01		X	1	
11			1	
10			1	

$\therefore F = w'x' + yz$

~~2~~ b) $F(A,B,C,D) = \Sigma_m(1, 4, 7, 10, 13) + d(5, 14, 15)$.

Ans.

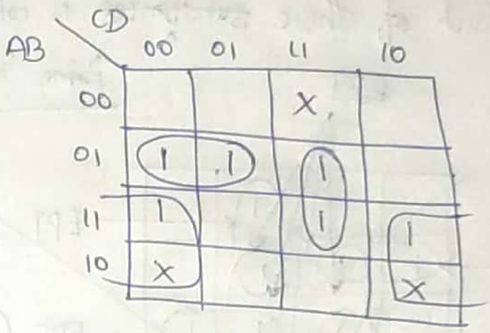


AB \ CD	00	01	11	10
00		1		
01	1	X	1	
11		1	X	X
10				1

$\therefore F = BD + ACD' + A'BC' + A'C'D$

c) ~~F(w,x~~ $F(A,B,C,D) = \Sigma_m(4, 5, 7, 12, 14, 15) + d(3, 8, 10)$

Ans.

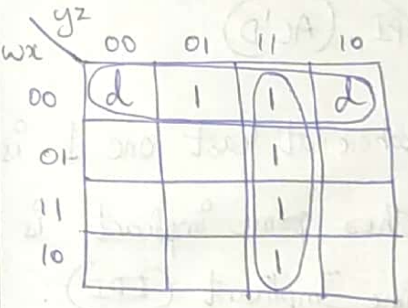


AB \ CD	00	01	11	10
00			X	
01	1	1	1	
11	1		1	1
10	X			X

$\therefore F = AD' + BCD + A'BC'$

d) $F(w,x,y,z) = \Sigma_m(1, \underset{2}{\blacksquare}3, 7, 11, 15) + d(0, \blacksquare)$.

Ans.



wx \ yz	00	01	11	10
00	d	1	1	d
01			1	
11			1	
10			1	

$\therefore F = w'x' + yz$

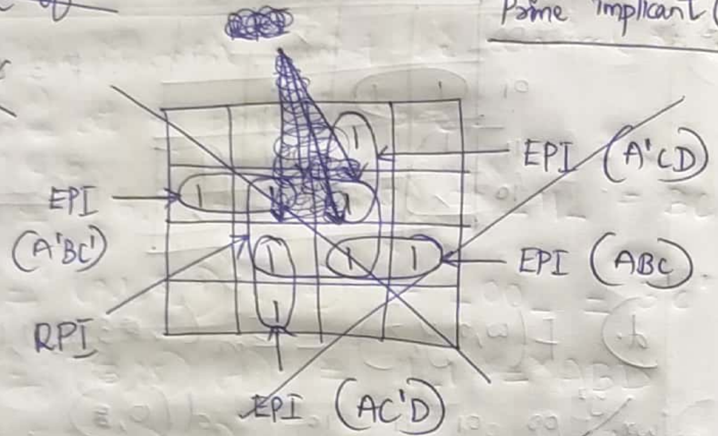
(Tabular Method)

2. Quine - McClusky Method :-

Prime implicants, Essential prime implicants, Redundant prime implicants and ~~Selective~~ Selective prime implicants

Each square or rectangle made up of the group of adjacent minterm is called a subcube. ~~Each of~~ Each of these ~~substances~~ subcubes is called **Prime Implicant (PI).**

~~Eg:~~



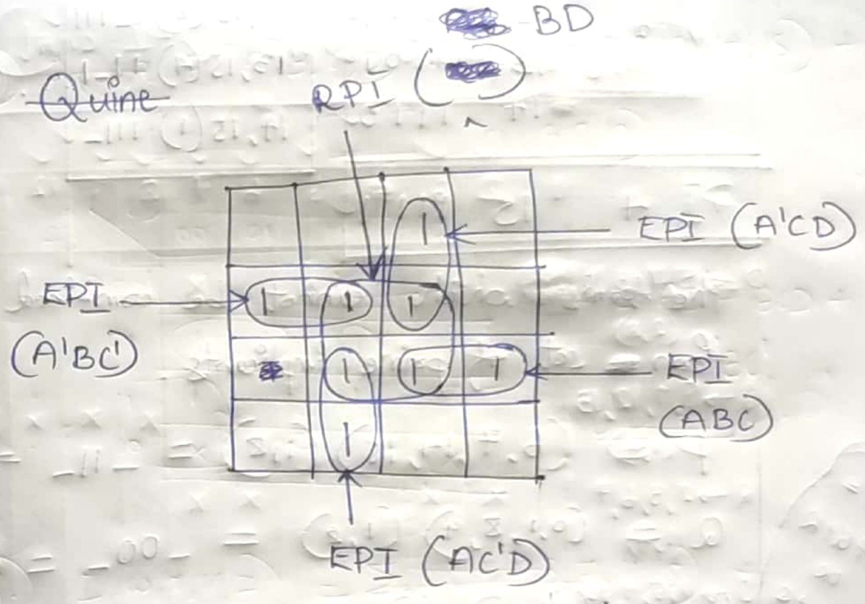
The prime implicant whose at least one 1 is already covered by other prime implicant is called **Essential Prime Implicant (EPI).**

The prime implicant whose each 1 is covered by at least one EPI is called **Redundant Prime Implicant (RPI).**

Neither essential nor redundant are called **Selective Prime Implicants (SPI).**

Eg:-

~~Quine~~



Quine-McClusky Method

1. Obtain the set of PIs for the boolean expression $F = \sum_m (0,1,6,7,8,9,13,14,15)$ using Tabular method.

Ans. ~~Group the minterms in~~

Column 1

Index	Minterm	Binary
Index 0	0	0000✓
,, 1	1	0001✓
	8	1000✓
,, 2	6	0110✓
	9	1001✓

Column 2

Pairs	ABCD
0,1 (1)	000_ ✓
0,8 (8)	_000 ✓
1,9 (8)	_001 ✓
8,9 (1)	100_ ✓
6,7 (1)	011_ ✓
6,14 (8)	_110 ✓
9,13 (4)	1_01 .. Ⓢ

Column 3

Quad	ABCD
0,1,8,9 (1,8)	_00_ ... Ⓠ
6,7,14,15 (1,8)	_11_ ... Ⓟ

" 3	7	0111✓	7,15 (8) _111✓
	13	1101✓	13,15 (4) 11_1 .. Ⓡ
	14	1110✓	14,15 (1) 111_✓
" 4	15	1111✓	

The terms which cannot be combined further form set of prime implicants.

f = P+Q+R or P+Q+S

$P = (6,7,14,15)\ (1,8) = \_11\_ = BC$

$Q = (0,1,8,9)\ (1,8) = \_00\_ = B'C'$

$R = (13,15)\ (4) = 11\_1 = ABD$

$S = (9,13)\ (4) = 1\_01 = AC'D$

2. Obtain the minimal expression for

$f = \sum_m (1,2,3,5,6,7,8,9,12,13,15)$

using the tabular method.

Ans.

Column 1

Index	Minterm	Binary
1	1	0001✓
	2	~~1000~~ 0010✓
	8	1000✓
2	3	0011✓
	5	0101✓
	6	0110✓
	9	1001✓
	12	1100✓
3	7	0111✓
	13	1101✓
4	15	1111

Column 2

Pair	ABCD
(1,3) (2)	00_1✓
(1,5) (4)	0_01✓
(1,9) (8)	_001✓
(2,3) (1)	001_✓
(2,6) (4)	0_10✓
(8,9) (1)	100_✓
(8,12) (4)	1_00✓
(3,7) (4)	0_11✓
(5,7) (2)	01_1✓
(5,13) (8)	_101✓
(6,7) (1)	011_✓
(9,13) (4)	1_01✓
(12,13) (1)	110_✓
(7,15) (8)	_111✓
(13,15) (2)	11_1✓

Column 3

Quad	ABCD	
(1,3,5,7) (2,4)	0__1	Ⓣ
(1,5,9,13) (4,8)	__01	Ⓢ
(2,3,6,7) (1,4)	0_1_	Ⓡ
(8,9,12,13) (1,4)	1_0_	Ⓠ
(5,7,13,15) (2,8)	_1_1	Ⓟ

~~P = x1x1 = BD~~

~~Q = 1x0x =~~

P = (5,7,13,15) (2,8) = x1x1 = BD

Q = (8,9,12,13) (1,4) = 1x0x = AC'

R = (2,3,6,7) (1,4) = 0x1x = A'C

S = (1,5,9,13) (4,8) = xx01 = C'D

T = (1,3,5,7) (2,4) = 0xx1 = A'D

Prime Implicant Chart (PI chart)

PI	1	✓ 2	✓ 3		✓ 5	✓ 6	✓ 7	✓ 8	✓ 9	✓ 12	✓ 13		✓ 15
P → 5,7,13,15 (2,8)					×		×				×		×
Q → 8,9,12,13 (1,4)								×	×	×	×		
R → 2,3,6,7 (1,4)		×	×			×	×						
S → 1,5,9,13 (4,8)	×				×				×		×		
T → 1,3,5,7 (2,4)	×		×		×		×						

2, 6 → Covered only by R. So, R is needed.

15 → P is needed.

8, 12 → Q is needed.

∴ P, Q, R → EPIs. (So, tick for 5,7,13,15, 8,9,12,13, 2,3,6,7)

Check if all minterms are included!

Two minimal expressions are possible ⇒

$P+Q+R+S = BD + AC' + A'C + C'D$

or

$P+Q+R+T = BD + AC' + A'C + A'D$

In 1),
P → EPI
Q → EPI
R → RPI
S → RPI

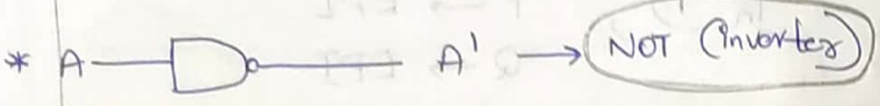
In 2),
P → EPI
Q → EPI
R → EPI
S → RPI
T → RPI

We have to add all EPIs, all SPIs, and some RPIs (so that all minterms are included). If we do not select an RPI, make sure its minterms are present in the remaining PIs.

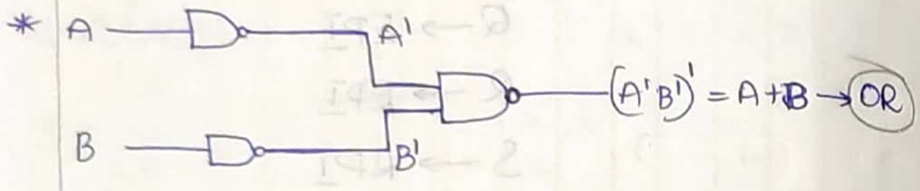
Multi level NAND Circuits

NAND & NOR – Universal gates.

NAND can give outputs of AND, OR and NOT.

* 

* 

* 

NAND and NOR can construct any digital circuit.

1. F = A(B+CD) + BC'

Ans. Steps.

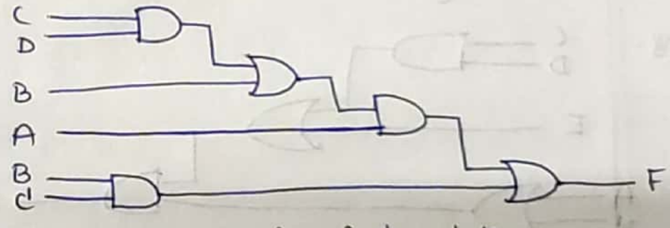
1. From the given algebraic expression, draw the logic diagram with AND, OR and NOT gates. Assume that both the normal and complement inputs are available.

2. Draw a second logic diagram with the equivalent NAND logic gates.

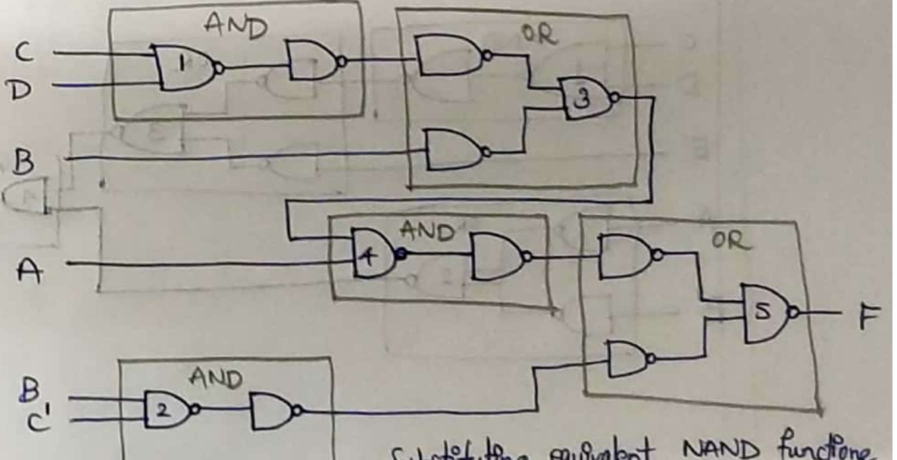
3. Remove any two cascaded inverters from the diagram, since double inversion does not perform a logic function. Remove inverters connected to single external inputs and complement the corresponding input variable. The new logic diagram obtained is the required NAND gate implementation.

[Cascading - Double NAND]

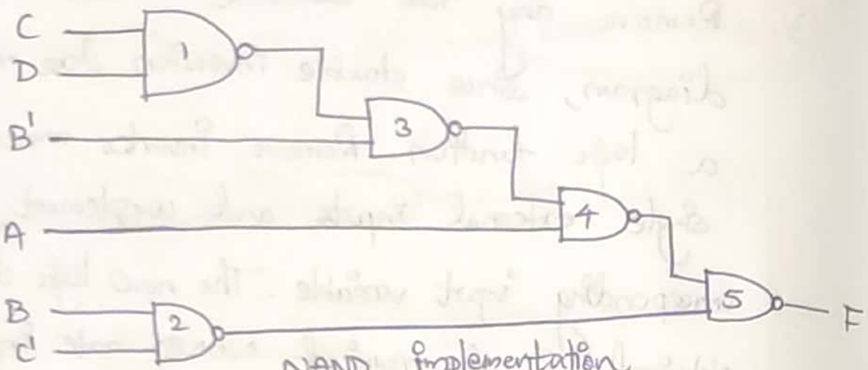
Answer.



AND/OR implementation



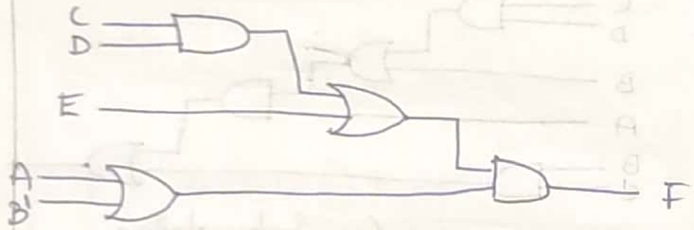
Substituting equivalent NAND functions

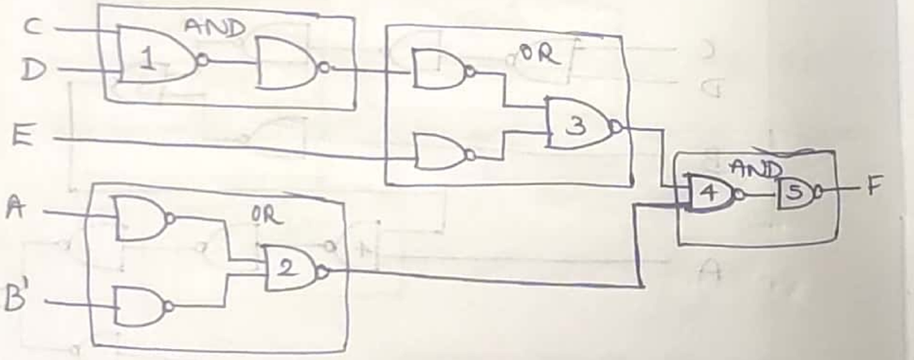


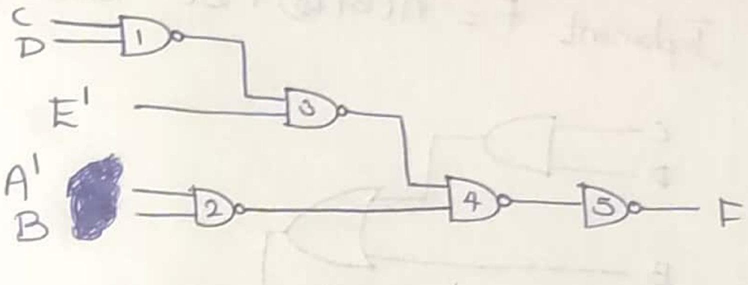
NAND implementation

2. Implementation of (A+B')(CD+E) with NAND gate.

Ans.





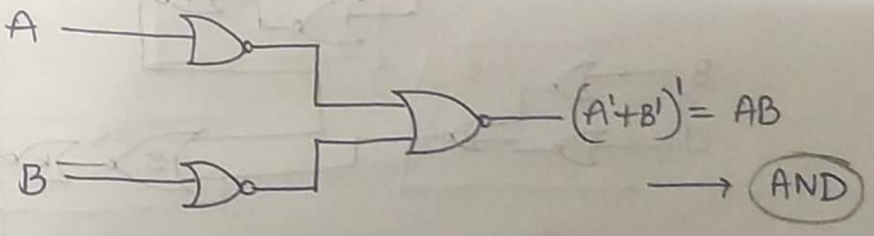


Multi Level NOR Circuits

Implementation of NOT, OR, and AND by NOR gates

* 

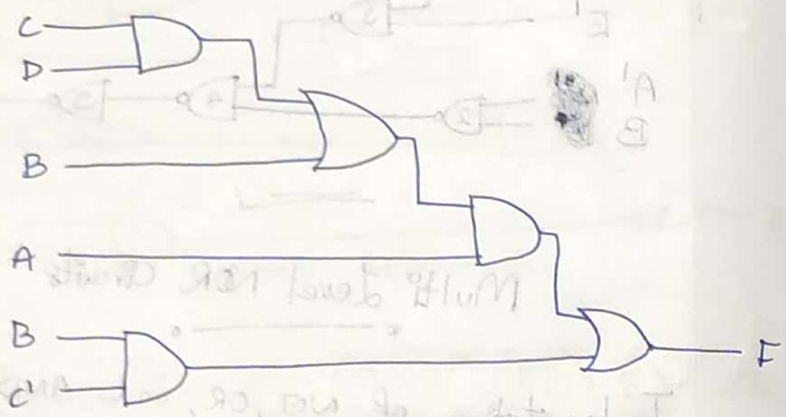
* 

* 

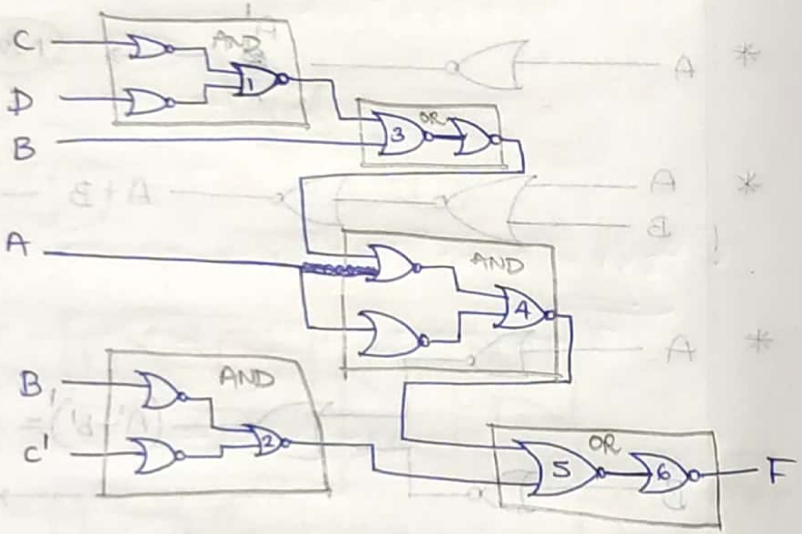
* Steps for NOR implementation is the same as those for NAND implementation. (NOR gates instead of NAND gates).

1. Implement $F = A(B+CD) + BC'$ with NOR gates.

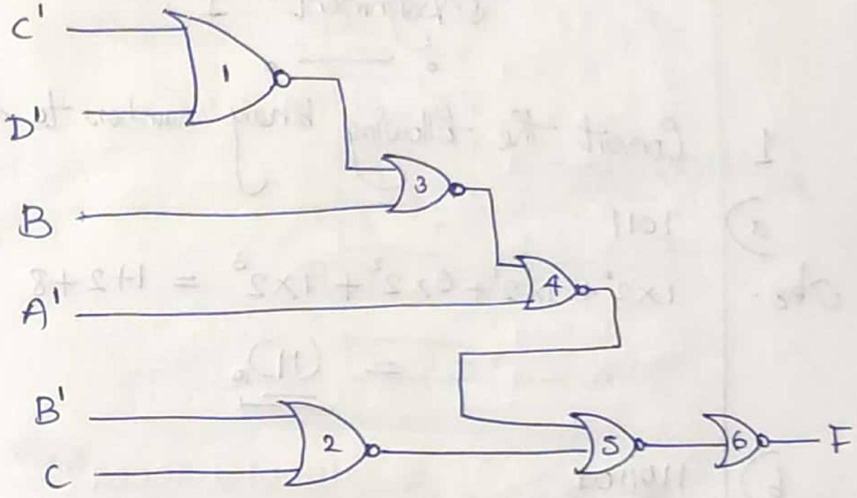
Ans.



AND/OR implementation



Substituting equivalent NOR functions



NOR implementation