18-02-20

MODULE-3

DIGITAL ELECTRONICS - BOOLEAN ALGEBRA AND LOGIC GATES.

Boolean Algebra is somehow similar to normal / ordinary algebra which can take any value for different variables.

Scientists George Boole, invented an algebra to solve logical problems where only 2 valued logic is involved. i.e 0 & 1. Therefore this algebra is known as Boolean Algebra.

Some Basic Laws of Boolean Algebra:

① OR Operation Rules:

$A + 0 = A$

$A + 1 = \cancel{A}1$

$A + A = A$

$A + \bar{A} = 1$

② AND operation Rules:

$A \cdot 0 = 0$

$A \cdot 1 = A$

$A \cdot A = A$

$A \cdot \bar{A} = 0$

③ NOT operation Rules:

$\bar{\bar{A}} = A$ (complementation law).

④ Commutative Law:

① $A \cdot B = B \cdot A$ ② $A + B = B + A$.

⑤ Associative Law:

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

(2) $A \cdot (B C) = (A \cdot B) \cdot C$

⑥ Distributive Law:

(1) $A \cdot (B+C) = A \cdot B + A \cdot C$

⑦ Absorption Laws:

i) $A \cdot (\bar{A}+B) = AB$

ii) $A + \bar{A}B = A+B$.

iii) $A + BC = (A+B)(A+C)$.

$A \cdot (\bar{A}+B)$

A	B	$\bar{A}$	$\bar{A}+B$	$A \cdot (\bar{A}+B)$
0	0	1	1	1
0	1	1	1	1
1	0	0	0	0
1	1	0	1	0

$$A \cdot (\bar{A}+B) = A \cdot \bar{A} + AB \quad (\because A \cdot \bar{A} = 0)$$
$$= 0 + AB \quad (\text{OR operation})$$
$$= AB$$

$$A + \bar{A}B = A \cdot A + A \cdot \bar{A} B$$
$$= \underline{A + B}.$$

$$A + \bar{A}B = A \cdot 1 + \bar{A} B$$
$$= A(1$$

v) $A + \bar{A}B = A \cdot (B + \bar{B}) = AB$

A	B	$\bar{A}$	$\bar{A}B$	$A+B$	$A+\bar{A}B$
0	0	1	0	0	0
0	1	1	1	1	1
1	0	0	0	1	1
1	1	0	0	1	1

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

$$A(1+B) + \bar{A}B$$

$$A + AB + \bar{A}B$$

$$A + B(A+\bar{A})$$

$$= A+B$$

iii) $A + BC$

A	B	C	BC	$A+BC$	$A+B$	$A+C$	$(A+B)(A+C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

Demorgan's law

i) $\overline{A+B} = \bar{A} \cdot \bar{B}$

A	B	$A+B$	$\overline{A+B}$	$\bar{A}$	$\bar{B}$	$\bar{A}\cdot\bar{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

ii) $\overline{A \cdot B} = \bar{A} + \bar{B}$

A	B	$A \cdot B$	$\overline{A \cdot B}$	$\bar{A}$	$\bar{B}$	$\bar{A}+\bar{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Simplify the Boolean equ.

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

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

$\cancel{=} \cancel{A} \cdot B \cancel{+} \cancel{\bar{A}}$

$Y = A \cdot B + \bar{A} C + ABC + \bar{A} B C$

$= AB(1+C) + \bar{A} C (1+B)$

$= AB + \bar{A} C$

Standard form of Boolean Equation:-
All the boolean expressions can be converted into either 2 standard forms.

i) SOP (Sum of Product form).
ii) POS (Product of Sum form)

SOP - The logical sum of 2 or more product terms is called Sum of Product.

Ex - $Y = AB + BC + AC$
↓
$Y(A, B, C)$ function of 3 variables.

Minterms :

In product term term/minterm are in complemented or uncomplemented form in multiplication format.

A function having 2 variables has different possible combination of minterms as

A	B	minterm	minterm designation (m)
0	0	$\bar{A}\cdot\bar{B}$	m_0
0	1	$\bar{A}\cdot B$	m_1
1	0	$A\cdot\bar{B}$	m_2
1	1	$A\cdot B$	m_3

0 → complement form.

1 → uncomplement form.

~~Y = A(A~~

Q. $Y(A,B) = A+B$. Find the Canonical SOP expression.

$$Y(A,B) = A\cdot 1 + B\cdot 1$$

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

$$= AB + A\bar{B} + BA + B\bar{A}$$

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

$$= m_3 + m_2 + m_1$$

$$\boxed{m_3+m_2+m_1 = \sum_m (m_3, m_2, m_1)}$$

→ Canonical expression.

POS :- Product of Sum form :

Ex - $Y = (A+B).(A+C).(B+C)$.

sum term/ maxterms.

1 - ~~un~~complemented form.

0 - uncomplemented form

A	B	Maxterm	Maxterm design.
0	0	$A+B$	M_0
0	1	$A+\bar{B}$	M_1
1	0	$\bar{A}+B$	M_2
1	1	$\bar{A}+\bar{B}$	M_3

$$Y = (A+B) \cancel{A} .$$
$$= (A+0)+(B+0)$$
$$= (A+B.\bar{B})+(B+A.\bar{A})$$
$$= B(\bar{B}+1)+A(A+\bar{A})$$
$$= B$$

Dt 19-02-20.

$Y(A,B) = (A+B)$ Find the canonical POS expression.

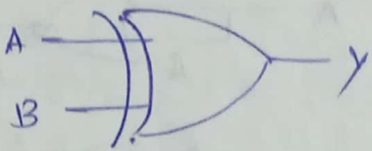
$Y(A,B) = A+B$

$=$

LOGIC GATES:

EX-OR

A	B	$Y = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0



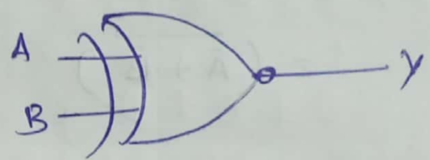
$A \oplus B = \bar{A}B + A\bar{B}$

It is also called as Inequality detector.

X-NOR: XOR + NOT.

A	B	$Y = \overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

$\overline{A \oplus B} = \bar{A}\bar{B} + AB$.



This is called as equality detector.

OR, AND, NOT - Basic Gates

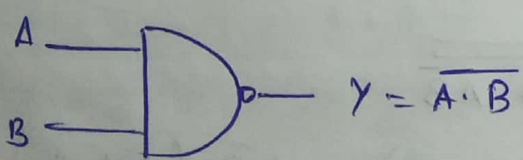
NOR, NAND, XOR, X-NOR - Derived Gates.

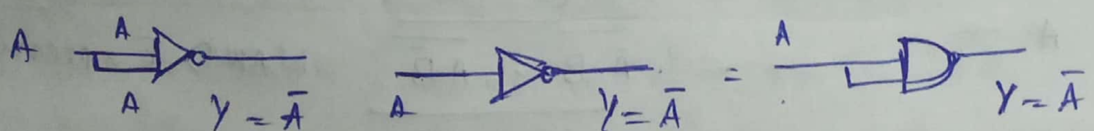
NOR & NAND - Universal gates.

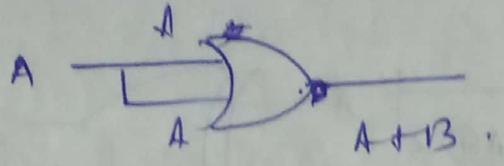
By using universal gates other gates can be derived.

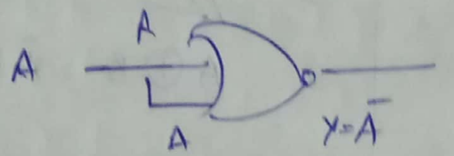
Complementation rule is used.

1. NOT gate using NAND & NOR gate:

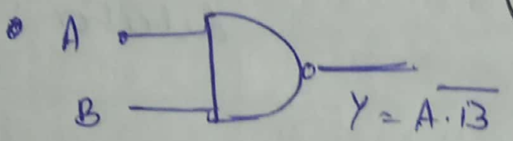






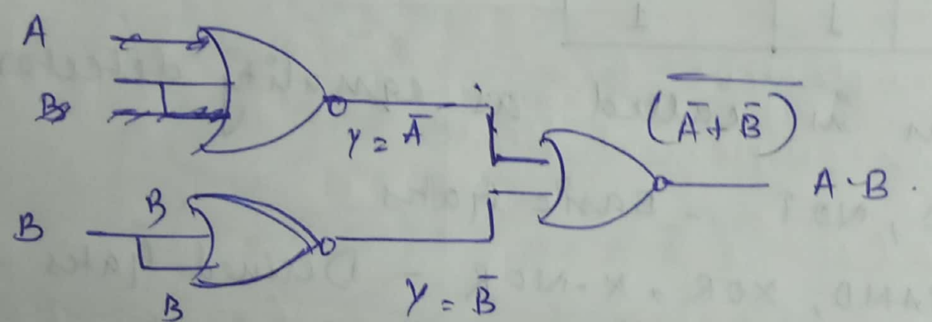


2. Find AND gate using NAND & NOR.



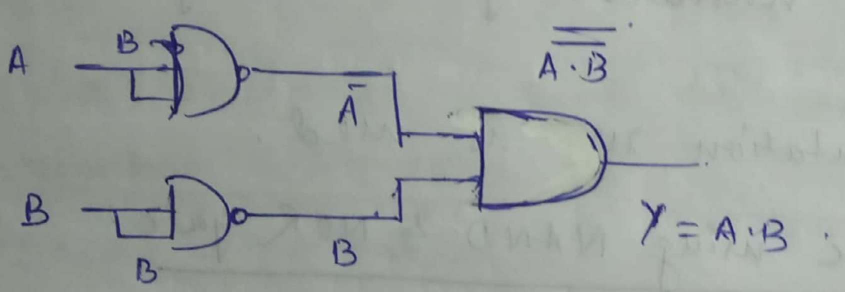
A $Y = \overline{\overline{A \cdot B}}$

$= (\overline{\bar{A} + \bar{B}})$



$Y = \overline{\overline{A \cdot B}}$

$\overline{\bar{A} + \bar{B}}$



3. EX-OR.

$Y = \overline{A \oplus B} = \cdot \overline{\overline{\bar{A} B + A \bar{B}}}$

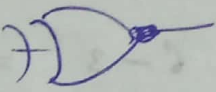
$\neq A +$ $= \bar{A} B (1 + B) + A \bar{B} (1 + A)$

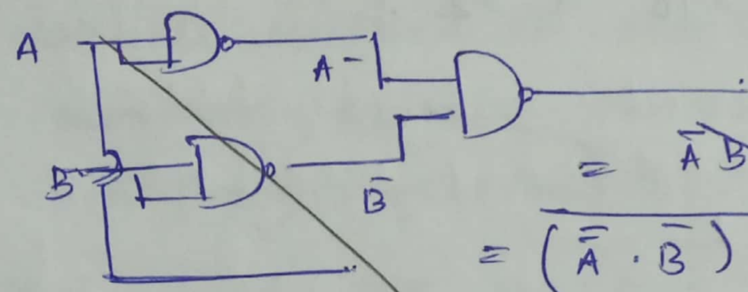
A $A \neq$ $= \overline{\overline{\bar{A} B} \cdot \overline{A \bar{B}}}$ NAND Expression.

$= \overline{A \bar{B} + \bar{A} B}$

B

Scanned by CamScanner

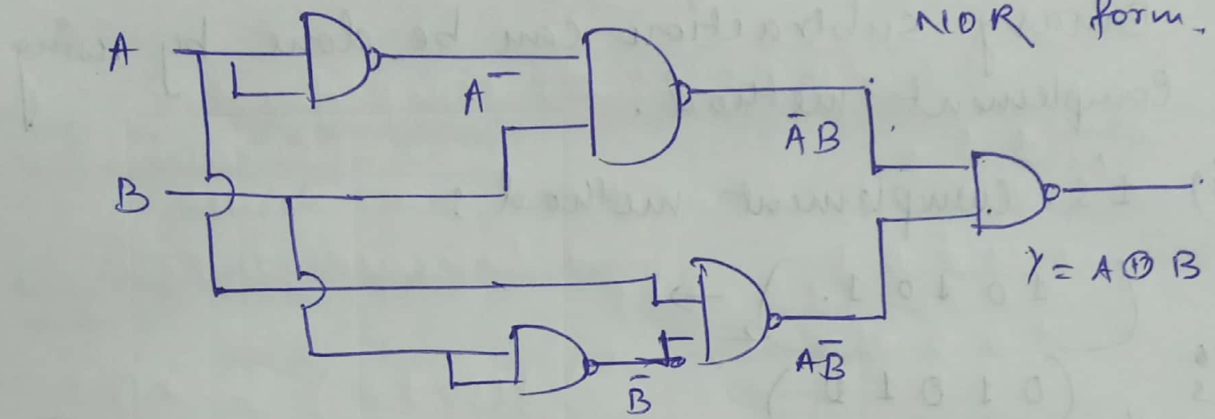


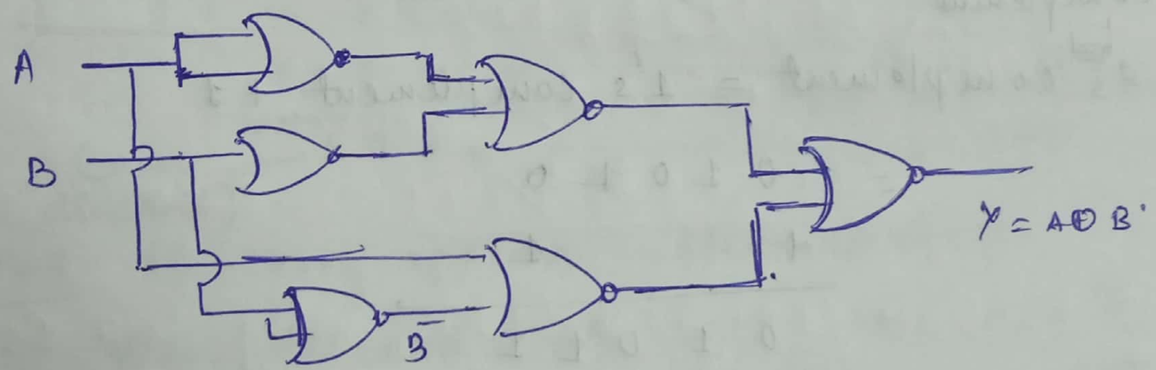


$= \overline{\bar{A}B}\cdot\overline{A\bar{B}}$

$= \overline{(\bar{\bar{A}}\cdot\bar{B}) + (\bar{A}+\bar{B})}$.

$= \underbrace{\overline{(A+\bar{B})} + \overline{(\bar{A}+B)}}_{\text{NOR form}}$





Arithmetic Operations used in Binary System :

1. Addition :

$1 + 1 = 10$ (carry)

$1 + 0 = 1$

$0 + 1 = 1$

$0 + 0 = 0$.

$$\begin{array}{r} 10 \\ +11 \\ \hline 101 \end{array}$$

2 + 3 = binary.

$(10)_2 + (11)_2 = (101)_2 = (5)_{10}$

2. Subtraction :

$0 - 0 = 0$ $\quad 0 - 1 = 10$.

$1 - 0 = 1$

$1 - 1 = 0$

$6-3 = (3)_{10} = (11)_2$.

$(110)_2 - (11)_2$

```
 1 1 0
   1 1
------
 0 1 1 .
```

Complement Method of Subtraction:

Binary subtraction can be done by using complement method.

i) 1's complement method:

$(10101.)_2 \rightarrow$

1's complement (01010)

2's complement = 1's complement + 1

```
=  0 1 0 1 0
 +         1
 ------------
   0 1 0 1 1
```

~~Positive~~ Complement only exist for (−ve) number, positive numbers have no complement number.

Ex: Subtraction using 1's complement method.

Subtract 11 from 13

$13 + (-11) = +2$.

11 - 1011 — 0100

13 - 1101 — 0010

−11 = 0100 — 1's complement of $(1011)_2$.

Complement method of subtraction is an alternate method of addition of different numbers, either (positive or -ve)
(-ve, -ve) (-ve +ve)

Step-1 Find the expression for addition of numbers.

13 + (-11).,

Step-2 Find complement of the -ve number which we want to subtract.

Step-3 Then add both 1's complement result and the given number.

$$\begin{array}{rr} 13 - & 1101 \\ + & 0100 \\ \hline & 10001 \end{array}$$

→ Intermediat result

Carry ← (discard)

→ $(17)_{10}$.

Step-4 If carry appears after addition then discard the carry and add it to the intermediate result ~~and add~~ to get the final result.

$$\begin{array}{rr} & 0001 \\ + & 1 \\ \hline (+) & 0010 \end{array}$$

→ $+(2)_2$.

Step-5 If carry appears final answer is a +ve number. So put +ve sign.

Step-6 If carry appears after addition in 2's complement method then discard the carry & the answer is a +ve number.

Subtract 11 from 13 using 2's complement

$13 + (-11)$

$\Rightarrow$ 11 - 1011.

13 - 1101

11 - 0100 — 1's complement.

+ 1

0101 — 2's complement.

$\Rightarrow$
```
  1 1 1
  0 1 0 1
+ 1 1 0 1
---------
1 0 0 0 1 0
```
~~10~~. → Intermediate Result.

carry (discard)

$\Rightarrow +(0010) = +(2)_2$

$\Rightarrow$
```
  0 0 0 0
+       1
---------
(+1). 0 0 0 1
```

Q. Subtract $(13)_{10}$ from $(11)_{10}$ using both 1's & 2's complement.

1. $11 + (-13)$

13 - 1101

11 - 01011

$\Rightarrow$ 13 - 0010 — 1's complement.

$\Rightarrow$
```
  1 0 1 1.
+ 0 0 1 0
---------
  1 1 0 1
```
no carry appears ← 1101 - Intermediate Result.

Step-7 $\Rightarrow$ ~~1101~~ If no carry appears, the final answer is -ve. Find the 1's complement of the result & put -ve sign

$\Rightarrow -(0010)_2 = -2$

c) 11 + (-13)

0 0 1 0 — 1's complement.
      1
0 0 1 1 — 2's complement

Final answer is a 1's complement of the intermediate result.

⇒
  0 0 1 1
  1 0 1 1
  1 1 1 0 → intermediate result.

Step - 8 Get the 2's complement of the intermediate result.

  0 0 0 1
+   1
$-(0010)_2 \Rightarrow -(2)_{10}$.

Signed Binary Numbers :

Using 1's & 2's complement method :

for +ve → 0 is assigned.
-ve → 1 is assigned.

		1's complement	2's complement
+9	0, 0001 001	same number.	
-9	1, 0001 001	1, 1110 110	.
+15	0, 000 1111.	same number	.
-15	~~10~~, 000 1111	1, 1110000	.

Add −9 with −6. $-9 + (-6)$

-9 1, 0 0 0 1 0 0 1 — 1, 1 1 1 0 1 1 0 — 1's complement

-6 1, 0 0 0 0 0 1 1 0 — 1, 1 1 1 1 1 0 0 1 — 1's complement

+

1, 1 1 1 0 1 1 0

⇒ 1, 1 1 1 1 0 0 1

1 1, 1 1 0 1 1 1 1. → intermediate

carry discard it and add it to the result

⇒ 1, 1 1 0 1 1 1 1

\+ 1

1, 1 1 1 0 0 0 0

↓ 1's complement

⇒

⇒ 0, 0 0 0 1 1 1 1. $= (-15)_{10}$

Subtract −6 from −9.

$-~~6~~ - (-~~9~~) = -~~6~~ + ~~9~~$.

⇒ $-9 - (-6) = -9 + 6$