BOOLEAN LOGIC

10/10/11

⇒ BOOLEAN LOGIC -

x	y	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Boolean logical ideas are characterised into three ways :

(1) Two fun^n can produce constant 0, 1

(2) four fun^n with unary operation (complement tranfer.

(3) Ten fun^n with binary operation (AND, OR, NAND, NOR, EXOR, EX-NOR, INHIBITION, IMPLICATION)

$f_0 = 0$ NULL OPERATION
$f_1 = x.y$ AND OPER.
$f_2 = x.y'$ INHIBITION (x but not y) (x/y)
$f_3 = x$ TRANSFER OPERATION
$f_4 = x'y$ INHIBITION (y but not x) (y/x)
$f_5 = y$ TRANSFER OPER.

$f_6 = xy' + x'y = x \oplus y$ EX-OR (x or y, BUT NOT BOTH)

$f_7 = x + y =$ OR OPERATION

$F_8 = (x+y)' = x \downarrow y$ NOR OPERATION

$f_9 = xy + x'y' = x \odot y$ EX-NOR (x EQUAL y) (COINCIDENCE EQUIVALENCE)

$F_{10} = y'$ COMPLEMENTARY

$f_{11} = x + y' = x \subset y$ IMPLICATION (IF y THEN x)

$f_{12} = x'$ COMPLEMENTARY

$f_{13} = x' + y = x \supset y$ IMPLICATION (IF x THEN y)

$F_{14} = (xy)' = x \uparrow y$ NAND OPERATION

$f_{15} = 1$ IDENTITY OPER.

NOTE -

- To get a logic gate from a logic operation it must be either commutative or associative.

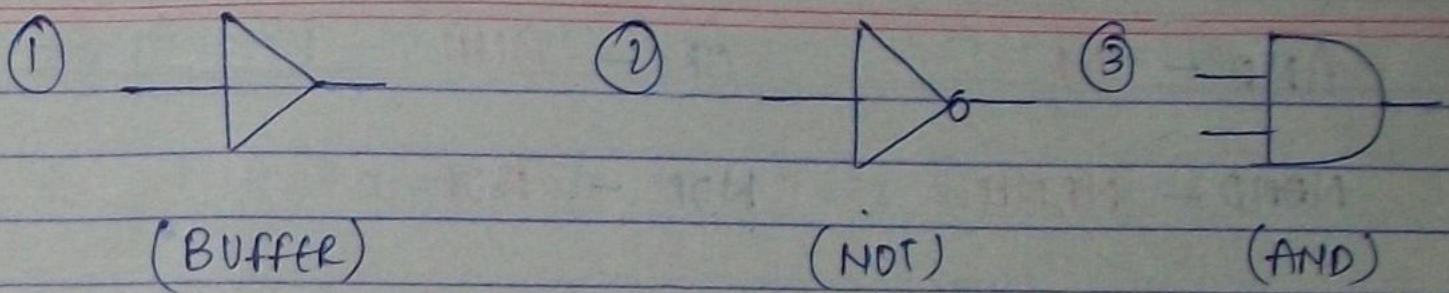
- Inhibition / Implication neither commutative nor associative.

- NAND / NOR commutative but not associative

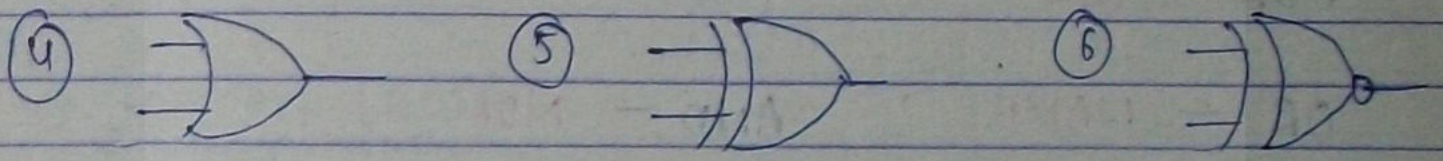
- NAND / NOR universal gates

NAND ⇄ NOR (4 / 4) (IMPLEMENT)

NOT	1	1
AND	2	3
OR	3	2
EX-OR	4	5
EX-NOR	5	4



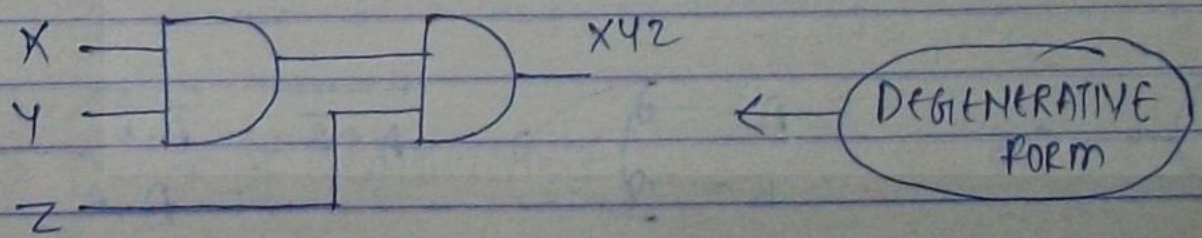
(BUFFER) (NOT) (AND)

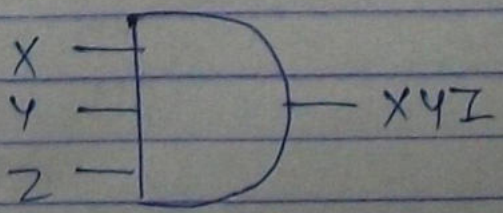


(OR) (EX-OR) (EX-NOR)

⇒ DEGENERATIVE FORMS -

If a two level logic gate system O/P is expressed with a single logic gate then that two level logic gate system is known as degenerative form for the singe logic gate.





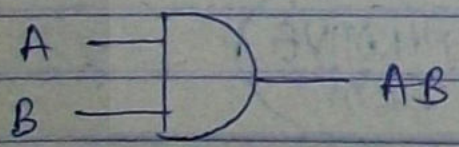
NON-DEGENERATIVE FORM –

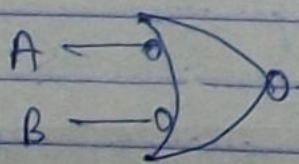
AND – OR	OR – AND
NAND – NAND	NOR – NOR
NOR – OR	NAND – AND
OR – NAND	AND – NOR

⇒ ALTERNATIVE LOGIC GATES –

NAND	BUBBLED →	OR
NOR	BUBBLED →	AND
AND	BUBBLED →	NOR
OR	BUBBLED →	NAND

Ex – AND





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

$= A \cdot B$

⇒ DUALITY –

1. Interchange $(\cdot \rightarrow +)$ or $(+ \rightarrow \cdot)$
2. Interchange $(0 \rightarrow 1)$ or $(1 \rightarrow 0)$

⇒ BOOLEAN LAW –

AND	OR
$x \cdot x = x$	$x + x = x$
$x \cdot 0 = 0$	$x + 0 = x$
$x \cdot 1 = x$	$x + \bar{x} = 1$
$x \cdot \bar{x} = 0$	$x + 1 = 1$

Ex – $f = \bar{x} \cdot y + z(x + y)$, $f^D = ?$

$f^D = \bar{x} + y \cdot z + (x \cdot y)$

⇒ ORDER OF IMPORTANCE –

1. [ ]
2. NOT
3. AND
4. OR

⇒ POSITIVE - NEGATIVE LOGIC –

#	(+ve) LOGIC	(-ve) LOGIC
	HIGH → 1	HIGH → 0
	LOW → 0	LOW → 1

Ex = −7V → 1 . = −ve logic
−2V → 0

A	B	f = AB
0	0	0
0	1	0
1	0	0
1	1	1

A	B	f = A+B
1	1	1
1	0	1
0	1	1
0	0	0

(*) Negative AND is positive OR and vice versa

(*) Negative NAND is positive NOR and vice versa

⇒ SPECIALITY OF EX-OR GATES –

$X \oplus Y = X'Y + XY'$

$X \oplus X = 0$

$X \oplus X' = 1$

$X \oplus 1 = X'$

$X \oplus 0 = X$

$\overline{X \oplus Y}$
$X' \oplus Y$
$X \oplus Y'$
$= X \odot Y$

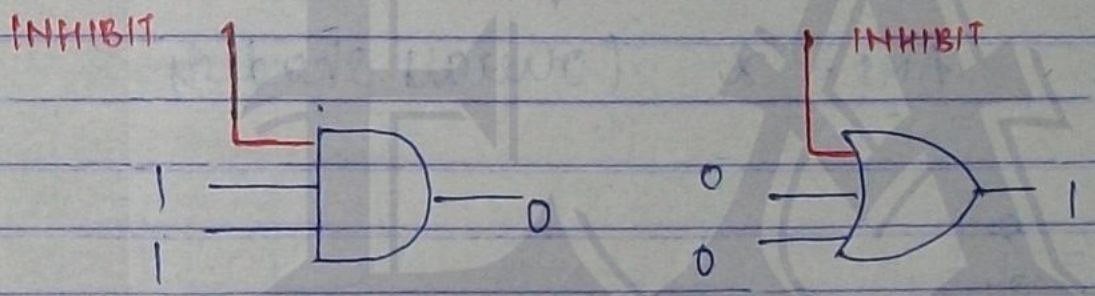
NOTE-

- In the case of symmetrical series of circuit the O/P cat:

O/P of 1st gate = O/P of every odd no. gate.

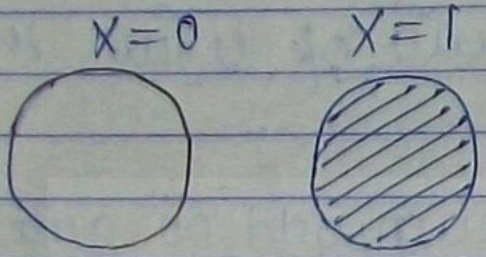
O/P of 2nd gate = O/P of every even no. gate.

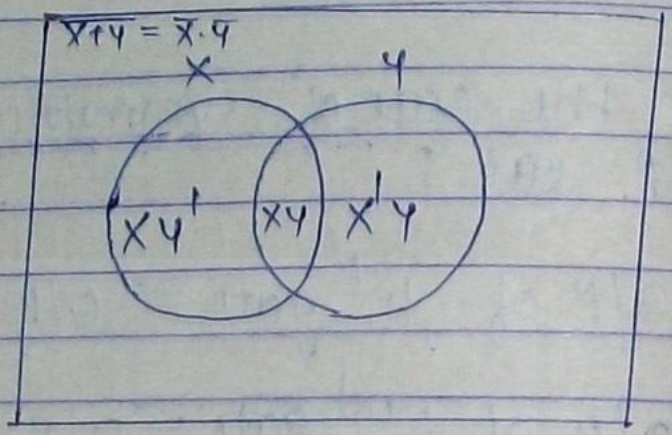
⇒ INHIBITING THE LOGIC GATE-



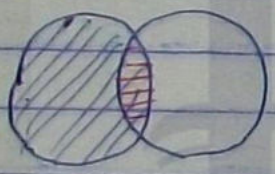
	DISABLE	ENABLE
NAND	L	H
NOR	H	L
AND	L	H
OR	H	L

⇒ VENN DIAGRAM –



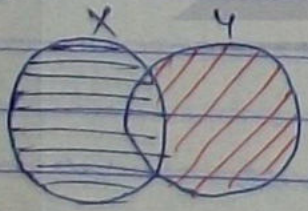


Ex – ① $X + XY = ?$

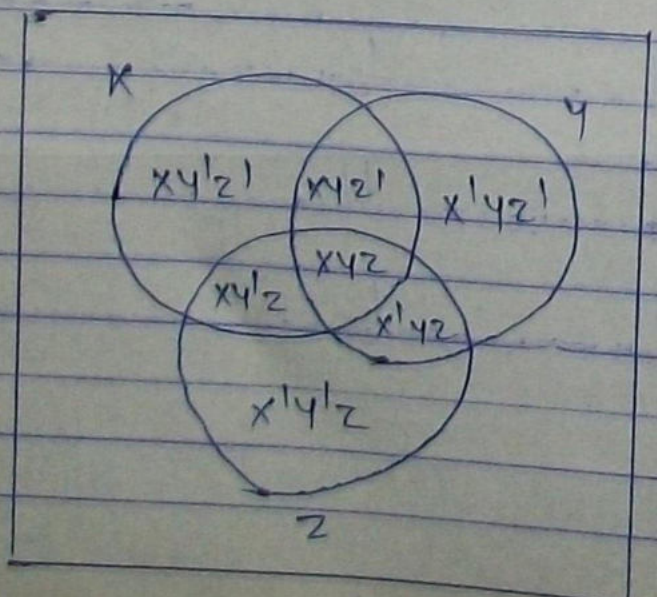


ANS = X (overall shaded)

② $X + X'Y = ?$



ANS = X + Y (overall shaded)



FOR THREE VARIABLE