Digital Electronics

Electronics: The study of motion of electron inside a semi-conductor is known as Electronics.

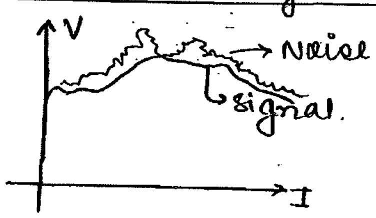
Note: there is a controlled conductivity in the case of semi-conductor

Gate: it is an electronic switching circuit made up of semiconductor switching devices.

Logic gates: Electronic switching ckt ie gate with logical ideas implementation by Mr. boolean are logical gates.

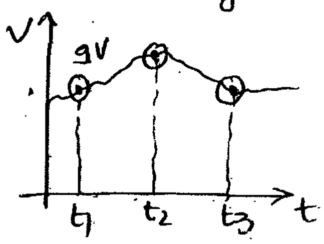
Need of digital signal:

- **Disadvantages of analog signal:**



(a) More bandwidth required.
(b) More power consumption.
(c) it is effected by noise more. (at recieving end if we amplify the signal, noise is also amplified)
(d) no chance for encryption.

- **advantages of digital signal:**



(a) Less bandwidth required.
(b) Less power consumption.
(c) it is effected less by noise.
(d) encryption is possible in digital signal.

Encryption is using digital data in the form of codes.

Block diagram:

Analog i/p → [ADC] → [digital system] → [DAC] → Analog O/P

Ex → μp, ALU

main systems: μp, ALU, computers, calculators ⇨ subsystems: Registers, counters, flip flops ⇨ Building blocks: logic gates.

⇩

contains diodes, BJT, mosfet

SYLLABUS

hapter 1: Basics:
- Boolean algebra.
- logic gates.
- K-maps.
- Number system.
- codes
- Data representation.

hapter 2: combinational circuits:
- Arthematic circuits
  - → half adder, half subtractor, full adder, full subtractor.
  - → parallel binary adder, look ahead carry adder
  - → BCD adder, 2's compliment adder subtractor.
- combinational circuits
  - → Decoder, multiplexer, demux, Encoder, decoder.
  - → comparator, code converter, parity generator, checker.

nit/chaper 3: sequential circuits:
- → Latches, flipflops, Registers, counters, state machines (mealy more, moore model).

hapter 4: logic families.
- → RTL, DTL, HTL, TTL, ECL
- → MOS logic: NMOS, PMOS, CMOS (mainly for Gate Exam)

hapter 5: ADC and DAC conversions.

Note: for 'n' switches we get 2^n possible states.

switch: → $s=0$ → o—o o—o
→ $s=1$ → o—o—o (or) o-o-o-o } one switch has two states

S_1	
0	OFF
1	ON

⇨ two switches have four states.

Boolean logical ideas:

These are categorized into three ways:

→ two function that produce constant '0' or '1'. (Null, identity)

→ four functions with unimary operations, complementary and transfer
↳ (NOT) ↳ Buffer.

→ ten functions with binary operations. (AND, OR, NAND, NOR, EX-OR, EX-NOR, Inhibition, Implication).

Note: for n input variables, we get (2^n) combinations and (2^{2^n}) [2^{2^n}] possible functions.

Ex = $n=2$ → $2^n \rightarrow 2^2 \rightarrow$ 4 combinations.
→ $2^{2^n} \rightarrow 2^{2^2} \rightarrow$ 16 functions/operations.

Truth tables:

		Null	AND	$x\cdot\bar{y}$ (inhibition)	x transfer Buffer	$\bar{x}\cdot y$ (inhibition)	transfer Buffer	Ex-OR $x\oplus y$	$x+y$	$\overline{x+y}$	coincidence $x\odot y$ (EXNOR)	NOT $\bar{y}$	$x+\bar{y}$ (implication)	NOT $\bar{x}$	$\bar{x}+y$ (implication)	NAND $(\overline{x\cdot y})$	Identity
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

f_0 = Null operation

$f_1 = x\cdot y = x \cap y$ (AND)

$f_2 = x\cdot\bar{y} = x/y$ = (x but not y) ⇒ Inhibition.

f_3 = x transfer ⇒ Buffer

$f_4 = \bar{x}\cdot y = y/x$ = (y but not x) ⇒ inhibition.

f_5 = y transfer ⇒ Buffer.

$f_6 = x \oplus y = \bar{x}\cdot y + x\bar{y}$ ⇒ Ex-OR (for diff i/p → o/p is high)

$f_7 = x+y = x \cup y$

$f_8 = \overline{x+y} = x \downarrow y$ ⇒ NOR

$f_9 = x \odot y = \bar{x}\bar{y} + xy$ → when i/p same → o/p is high
→ coraincidence logic gate (or) Equivalence logic gate.

$f_{10} = \bar{y}$ = complementary = NOT gate.

$f_{11} = x + \bar{y} = x \subset y$ = If y then x → implication

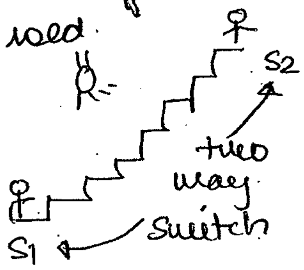
$f_{12} = \bar{x}$ = complementary = NOT gate.

$f_{13} = \bar{x} + y = x \supset y$ = If x then y = implication

$f_{14} = \overline{(x \cdot y)}$ = NAND = $x \uparrow y$

$f_{15} = 1$ = Identity.

Note: for staircase and acceletor operation EX-OR logic gate is used.



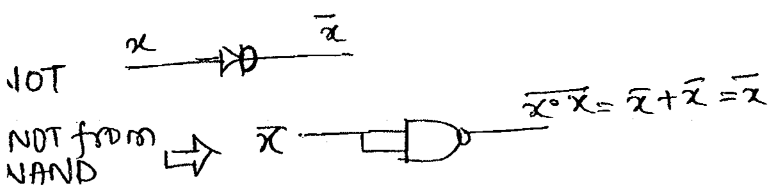
S_2	S_1	bulb $= S_2 \oplus S_1$
0	0	0
0	1	1
1	0	1
1	1	0

→ Basic logic gates: NOT, AND, OR

→ combinational logic gates: NAND (AND+NOT) & NOR (NOT+OR)

→ special purpose logic gates: EX-OR, EX-NOR (Arthematic operations).

∘ NAND & NOR are said to be universal logic gates.

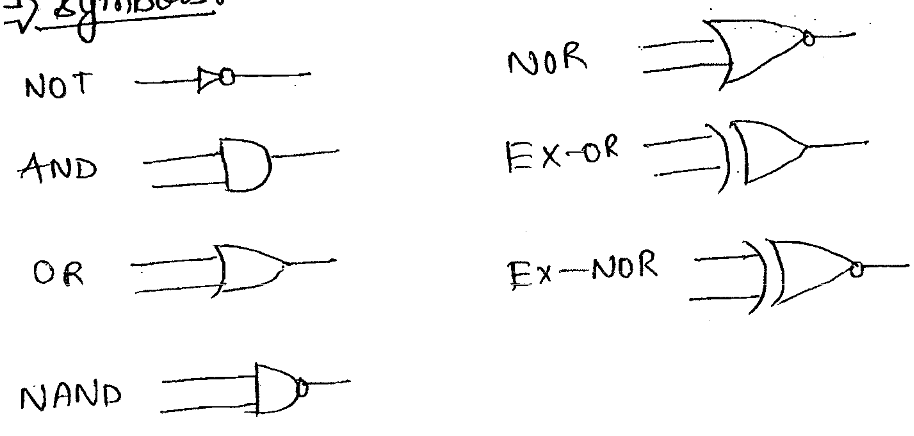


Demorgan's theorem.

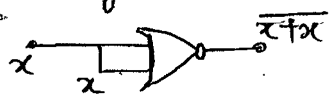
$\overline{x \cdot y} = (\bar{x} + \bar{y})$

$\overline{x + y} = (\bar{x} \cdot \bar{y})$

→ symbols:



NOT from NOR



$\Rightarrow \overline{x+x} = \bar{x}\cdot\bar{x} = \bar{x}$

⊛ Shortcut: No of Nand and NOR gate required:

Gate	No of Nand gates	No of NOR gates	NAND to NOR	(OR)	NOR to NAND
→ NOT	1	1	(4)		(4)
→ AND	2	3			
→ OR	3	2			
→ Ex-OR	4	5			
→ Ex-NOR	5	4			

⊛ Design of logic gates using universal logic gates:

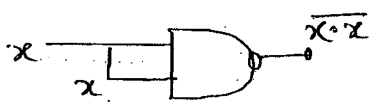
- universal logic gates are NAND and NOR.

(a) NOT from NAND:

→ output of NOT $\Rightarrow \bar{x}$

→ No of Nand gates req = 1

Circuit:



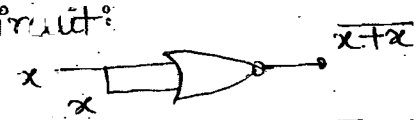
→ $\overline{x\cdot x} = \bar{x}+\bar{x} \Rightarrow \bar{x}$

(b) NOT from NOR:

→ output of NOT gate = $\bar{x}$

→ No of ~~NAND~~ NOR gates req = 1

circuit:



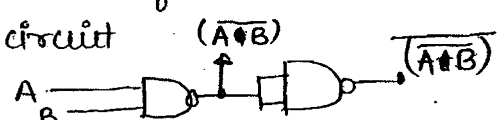
→ $\overline{x+x} = \bar{x}\cdot\bar{x} = \bar{x}$

(c) AND from NAND:

→ output of AND $\Rightarrow x\cdot y$

→ No of NAND req = 2

circuit

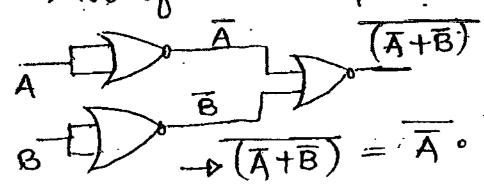


$\overline{(\overline{A\cdot B})} \Rightarrow A\cdot B$

(d) AND from NOR:

→ output of AND = $x\cdot y$

→ No of NOR req = 3

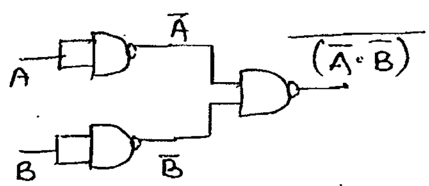


→ $\overline{(\bar{A}+\bar{B})} = \bar{\bar{A}}\cdot\bar{\bar{B}} = A\cdot B$

(e) OR from NAND:

→ output of OR $\Rightarrow x+y$

→ no of Nand $\Rightarrow$ 3

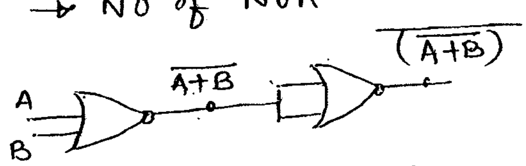


→ $\overline{(\bar{A}\cdot\bar{B})} \Rightarrow \bar{\bar{A}}+\bar{\bar{B}} = A+B$

(f) OR from NOR:

→ output of OR = $x+y$

→ No of NOR = 2

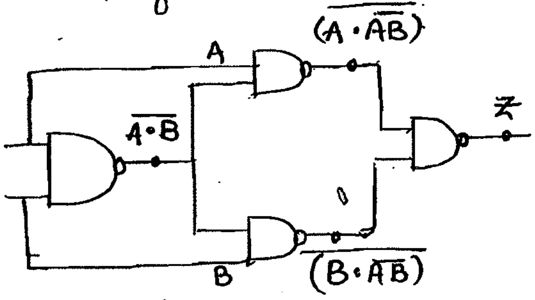


→ $\overline{(\overline{A+B})} = A+B$

g) EXOR from NAND: $(A \oplus B)$

→ output of EX-OR $= \bar{A}B + A\bar{B}$

→ no of NAND $= 4$



$z = \overline{\left[\overline{(A\cdot\overline{AB})}\cdot\overline{(B\cdot\overline{AB})}\right]}$

$z = \overline{\overline{(A\cdot\overline{AB})}} + \overline{\overline{(B\cdot\overline{AB})}}$

$z = A\cdot\overline{AB} + B\cdot\overline{AB}$

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

$z = \underbrace{A\cdot\bar{A}}_{0} + A\bar{B} + B\cdot\bar{A} + \underbrace{B\bar{B}}_{0}$

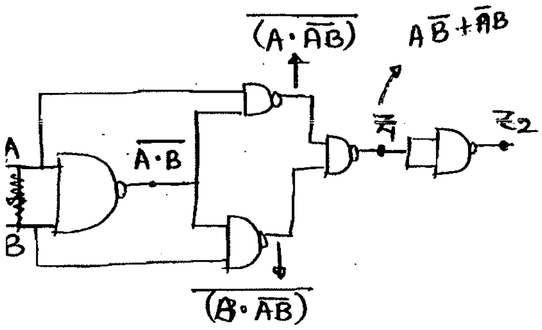
$= A\bar{B} + B\bar{A}$

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

) Ex-NOR from NAND: $(A \odot B)$

→ output of ExNOR $= \bar{A}\bar{B} + AB$

→ no of nand req $= 5$



$z_1 = A\bar{B} + \bar{A}B$

$z_2 = \overline{z_1}$

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

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

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

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

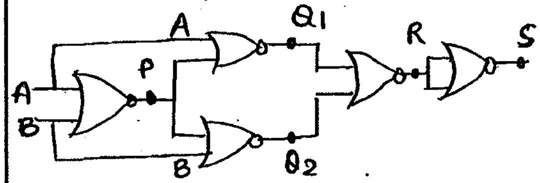
$= \underbrace{A\bar{A}}_{0} + \bar{A}\bar{B} + BA + \underbrace{B\bar{B}}_{0}$

$z_2 = \bar{A}\bar{B} + AB$

(h) Ex-OR from NOR:

→ output of Ex-OR $= \bar{A}B + A\bar{B}$

→ no of NOR $= 5$



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

$Q_1 = \overline{(A+P)} = \overline{(A+\overline{(A+B)})} = \bar{A}\cdot\overline{\overline{(A+B)}}$

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

$Q_2 = \overline{(B+P)} = \overline{(B+\overline{(A+B)})} = \bar{B}\cdot\overline{\overline{(A+B)}}$

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

$R = \overline{(Q_1+Q_2)} = \overline{[\bar{A}\cdot(A+B) + \bar{B}(A+B)]}$

$= \overline{[\underbrace{\bar{A}A}_{0} + \bar{A}B + \bar{B}A + \underbrace{\bar{B}B}_{0}]} = \overline{[\bar{A}B + \bar{B}A]}$

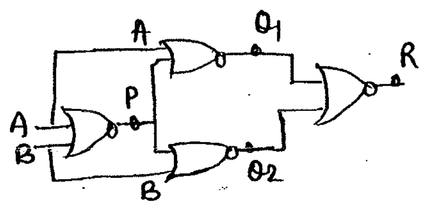
$R = \overline{(\bar{A}B + A\bar{B})}$

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

(J) Ex-NOR from NOR:

→ output of ExNOR $= \bar{A}\bar{B} + AB$

→ no of NOR $= 4$



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

$Q_1 = \overline{(A+P)} = \overline{A+\overline{(A+B)}} = \bar{A}\cdot\overline{\overline{(A+B)}}$

$= \bar{A}\cdot(A+B) = \underbrace{A\bar{A}}_{0} + \bar{A}B = \bar{A}B$

$Q_2 = \overline{(B+P)} = \overline{(B+\overline{A+B})} = \bar{B}\cdot\overline{\overline{(A+B)}}$

$= \bar{B}\cdot(A+B) = \bar{B}A + \underbrace{\bar{B}B}_{0} = A\bar{B}$

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

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

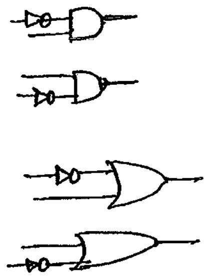
$= (\bar{\bar{A}}+\bar{B})\cdot(\bar{A}+\bar{\bar{B}})$

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

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

$= \underbrace{A\bar{A}}_{0} + AB + \bar{B}\bar{A} + \underbrace{\bar{B}B}_{0}$

Inhibition → $\bar{x}\cdot y$, $x\cdot\bar{y}$

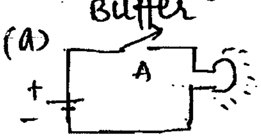


Implication → $\bar{x}+y$, $x+\bar{y}$

these are also universal logic gates.

⊛ switching circuits as logic gates:

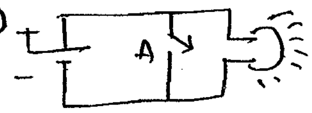
(a) Buffer



A	bulb
0	off(0)
1	ON(1)

$f = A$ (Buffer)

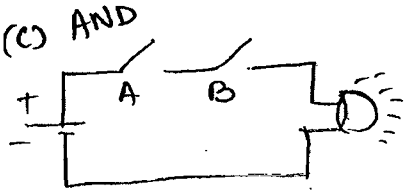
(b) NOT



A	bulb
0	ON(1)
1	off(0)

$f = \overline{A}$ (NOT)

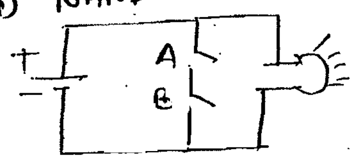
(c) AND



A	B	bulb
0	0	0
0	1	0
1	0	0
1	1	1

$f = A\cdot B$ (AND)

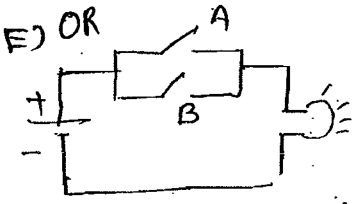
(d) NAND



A	B	bulb
0	0	1
0	1	1
1	0	1
1	1	0

$f = \overline{A\cdot B}$ = NAND.

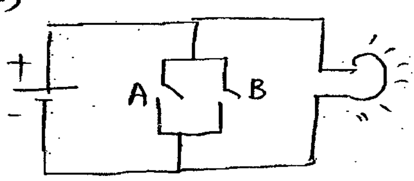
(E) OR



A	B	bulb
0	0	0
0	1	1
1	0	1
1	1	1

$f = A+B$ (OR)

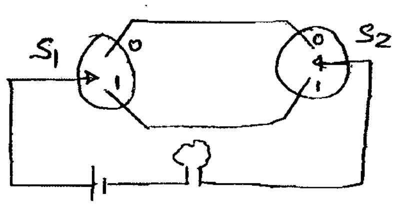
(F) NOR



A	B	bulb
0	0	1
0	1	0
1	0	0
1	1	0

$f = \overline{A+B}$ = NOR

(G)

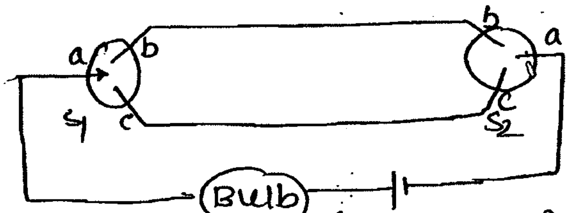


S_1	S_2	bulb
0	0	1
0	1	0
1	0	0
1	1	1

$f = S_1 \odot S_2$

⇒ workbook practice : CH2 : QT1, T2

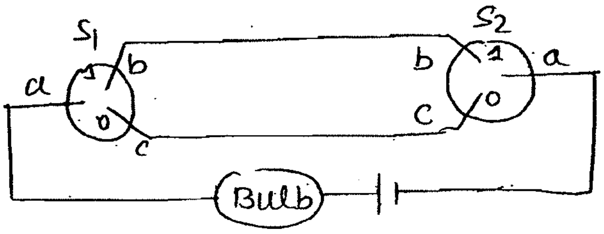
QT1 A two way switch has three terminals a, b and c. In ON position (logic value 1), a is connected to b, and in off position a is connected to c. Two of these switches are connected to bulb as shown in figure:



Which of the following expression, if true, will always result in lightning of bulb?

(a) $S_1 \cdot \overline{S_2}$ (b) $S_1 + S_2$ ~~(c)~~ $\overline{S_1 \oplus S_2}$ (d) $S_1 \oplus S_2$

Sol



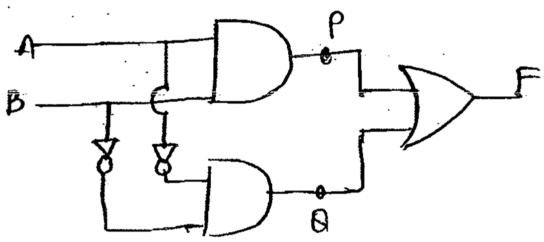
S_1	S_2	bulb
0	0	1
0	1	0
1	0	0
1	1	1

} $S_1 \odot S_2$

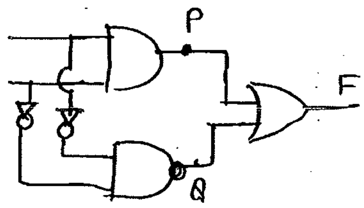
$S_1 \odot S_2 = \overline{S_1 \oplus S_2}$

T2 which one of the following represents coincidence logic gate?

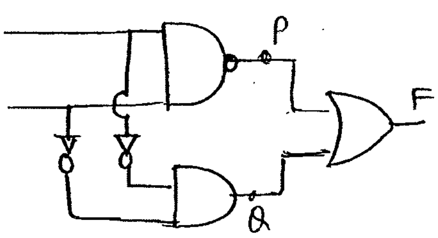
~~(a)~~



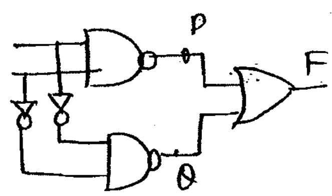
(c)



(b)



(d)

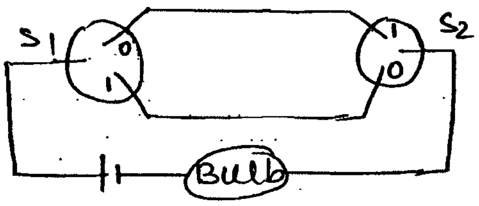


Solⁿ (a) truth table

A	B	P	Q	F
0	0	0	1	1
0	1	0	0	0
1	0	0	0	0
1	1	1	0	1

} $S_1 \odot S_2$ → coincidence logic gate

Q Ex:



S_1	S_2	bulb
0	0	0
0	1	1
1	0	1
1	1	0

$\} \; S_1 \oplus S_2$

(*) canonical forms:

	x	y	minterm → (product term)	f(ex)
m_0	0	0	→ $\bar{x}\cdot\bar{y}$	0
m_1	0	1	→ $\bar{x}\cdot y$	1 ✓
m_2	1	0	→ $x\cdot\bar{y}$	0
m_3	1	1	→ $x\cdot y$	1 ✓

$\boxed{f = \Sigma m(1,3)}$

$= f = m_1 + m_3 = \bar{x}\cdot y + x\cdot y$

SOP

$f = \Sigma m(1,3)$

	x	y	maxterm → (sum term)	f(ex)
M_0 →	0	0	→ $x+y$	0 ✓
M_1	0	1	→ $x+\bar{y}$	1
M_2	1	0	→ $\bar{x}+y$	0 ✓
M_3	1	1	→ $\bar{x}+\bar{y}$	1

$f = M_0 + M_2 = (x+y)\cdot(\bar{x}+y)$

⇒ POS

$\boxed{f = \pi M(0,2)}$

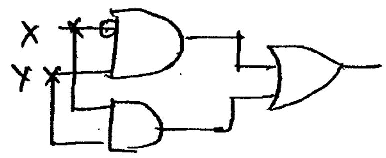
$\Rightarrow \overline{\bar{x}\cdot\bar{y}} = \bar{\bar{x}} + \bar{\bar{y}}$

$= x + y$

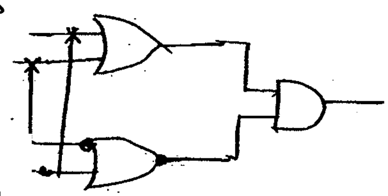
i.e. $\overline{(\text{minterm})} = (\text{maxterm})$

$\overline{m_0} = M_0$, $\overline{m_1} = M_1$, $\overline{m_2} = M_2$, $\overline{m_3} = M_3$ & $\overline{M_0} = m_0$, $\overline{M_1} = m_1$, $\overline{M_2} = m_2$, $\overline{M_3} = m_3$.

SOP → AND – OR



POS → OR – AND



$f = \Sigma m(1,3) = \pi M(0,2)$ ← missing no.

Q

x	y	f
0	0	0
0	1	1
1	0	1
1	1	0

SOP = ?
POS = ?

SOP(f) = $\bar{x}y + x\bar{y}$
POS(f) = $(\bar{x}+\bar{y})\cdot(x+y)$ } This is EX-OR gate

$= \bar{x}(x+y) + \bar{y}(x+y)$

$= x\bar{x} + \bar{x}y + \bar{y}x + \bar{y}y$ ($x\bar{x} \to 0$, $\bar{y}y \to 0$)

$= \bar{x}y + x\bar{y} = x \oplus y$

- In canonical sop form, each product term must contains all variable of given expression.

Note: In canonical sop, each product term is known as minterm.

- In canonical POS, each sum term is known as max term.

x	y	f
0	0	0
0	1	0
1	0	0
1	1	1

SOP = ?
POS = ?

$SOP(f) = x\cdot y$

$POS(f) = (x+y)(x+\bar{y})(\bar{x}+y)$

$= x+y\,[x(\bar{x}+y)+\bar{y}(\bar{x}+y)]$

$= (x+y)(\cancel{x\bar{x}}^{0} + xy + \bar{y}\bar{x} + \cancel{x\bar{y}}^{0})$

$= (x+y)(xy+\bar{y}\bar{x})$

$= x(xy+\bar{y}\bar{x}) + y(xy+\bar{y}\bar{x})$

$= xy + \cancel{\bar{y}\bar{x}x}^{0} + yx + \cancel{\bar{y}\bar{x}y}^{0}$

$= x\cdot y$

<u>Note</u>: In the min-term zero corresponding variable is complimented and one corresponding variable is taken without complimentation.

In the case of max term, 0 corresponding variable is taken without complimentation & 1 corresponding variable is taken with complimentation.

$f = \bar{x}\cdot y + x\cdot y \rightarrow$ canonical SOP

$\quad\; \Downarrow \qquad \Downarrow$

$\quad\; \frac{01}{1} \qquad \frac{11}{3}$

$f = \Sigma m(1,3) \rightarrow$ standard form

$f = \bar{x}\cdot y + x\cdot y$

$= y(\bar{x}+\bar{x})$

$= y(1)$

$= y \rightarrow$ minimal form

	x	y	f
0	0	0	1
1	0	1	0
2	1	0	1
3	1	1	0

find SOP & POS:
- → canonical form
- → standard form
- → minimal form

<u>SOP</u>: $f = \Sigma m(0,2) \rightarrow$ standard form

$f = m_0 + m_2$

$\;\; = 00 \cdot 10$

$f = \bar{x}\bar{y} + x\bar{y} \rightarrow$ canonical form

$= \bar{y}(\bar{x}+x)$

$= \bar{y}(1)$

$f = \bar{y} \rightarrow$ minimal form.

POS: $f = \pi M(1,3)$ → ~~canon~~ standard form.

$= M_1 + M_3$

$\Downarrow$ 01 $\quad \Downarrow$ 11

$f = x\bar{y} + \bar{x}\bar{y}$ → canonical form.

$= \bar{y}(x+\bar{x})$

$f = \bar{y}$ → minimal form.

Note: If more zero's is given in output function then we go for min term with (1)

If more 1's are given in output function then we go for max term with (0).

$101 \to n=3 \to 2^3 = 8$ combinations

$f = \Sigma m(1,5) = \pi M(0,2,3,4,6,7)$

$= \pi M(1,9) = \Sigma m(0,2,3,4,5,6,7,8,10,11,12,13,14,15)$

1001 ↓ $n=4 \to 2^4 = 16$ combinations.

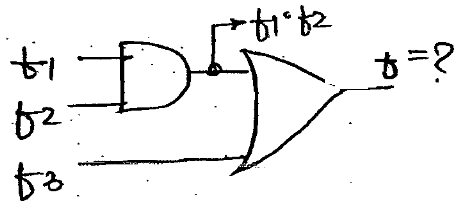
Ex: $f_1 = \Sigma m(0,1,2,4)$

$f_2 = \Sigma m(1,3,4,5)$

$f_3 = \Sigma m(2,3,4,5,6,7)$

AND = $f_1 \cdot f_2 = f_1 \cap f_2$

OR = $A + B = A \cup B$.



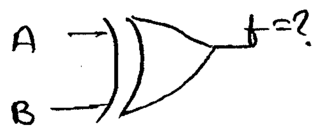
$f_1 \cdot f_2 = \Sigma m(1,4)$

$f_3 = \Sigma m(2,3,4,5,6,7)$

$f \Rightarrow f_1 \cdot f_2 + f_3 = \Sigma m(1,2,3,4,5,6,7)$.

Ex: $A = \Sigma m(0,1,4,5)$

$B = \Sigma m(2,3,\cancel{4},\cancel{5},6,7)$



$f = A \odot B$

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

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

$\bar{A}B$ → inhibition → B but not A.

$\bar{A}B = \Sigma m(2,3,6,7)$

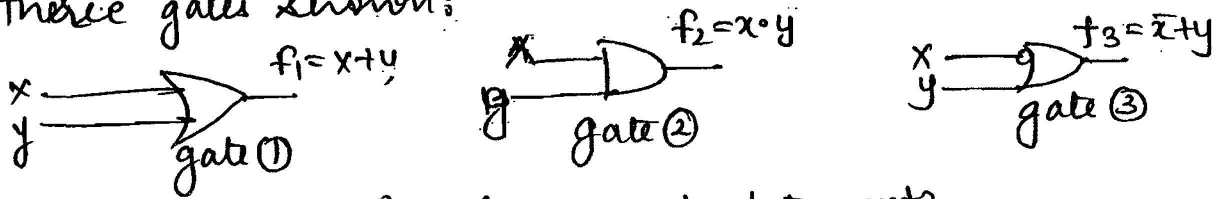
$A\bar{B}$ → inhibition → A but not B

$A\bar{B} = \Sigma m(0,1)$

$\bar{A}B + A\bar{B} = \Sigma m(0,1,2,3,6,7)$

▷ workbook practice: CH2 : Q 21

Q21 A universal logic gate can ~~be~~ implemented any boolean function by connecting sufficient no of them appropriately. Three gates shown:



which of the following is correct statement?

(a) Gate 1 is a universal gate.

(b) Gate 2 is a universal gate.

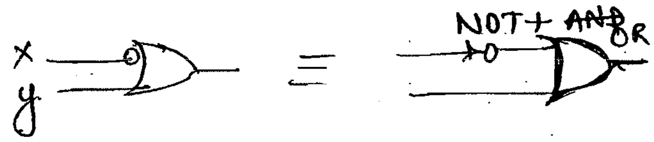
~~(c)~~ gate 3 is a universal gate.

(d) ~~gate~~ none of the gates are universal gate.

Ans: gate ①: $f_1 = x+y$ → It is OR gate
↳ not universal.

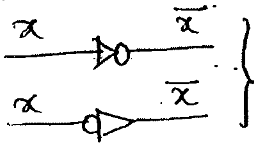
gate ②:- $f_2 = x \cdot y$ → it is AND gate
↳ not universal

gate ③:- $f_3 = \bar{x}+y$ → implication
↳ universal gate

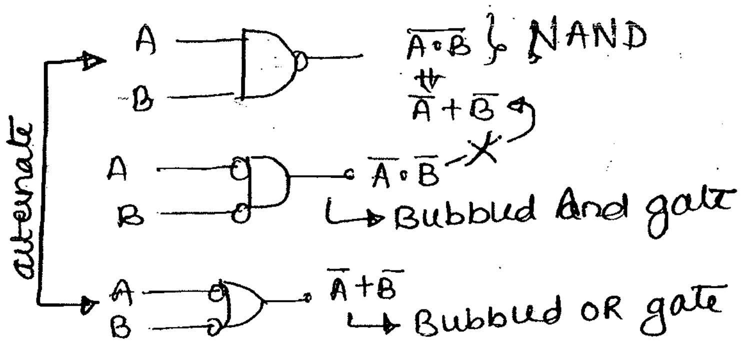


⇒ universal gate

Alternative logic gates:



} NOT



NAND $\xrightarrow{\text{alternate}}$ Bubbled OR gate

NOR $\xrightarrow{\text{alternate}}$ Bubbled And Gate

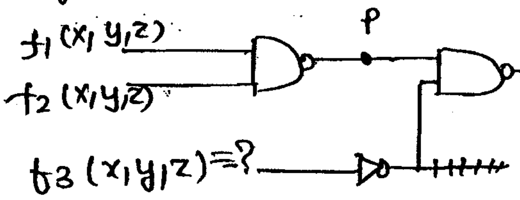
AND $\xrightarrow{\text{alternate}}$ Bubbled NOR gate

OR $\xrightarrow{\text{alternate}}$ Bubbled NAND gate

] shortcut.

⇨ workbook practice : CH2 : Q3, Q1, Q12

Q3 consider the ~~following~~ ~~logic~~ circuit whose inputs are functions f_1, f_2 and f_3 . and output is f.
given that:

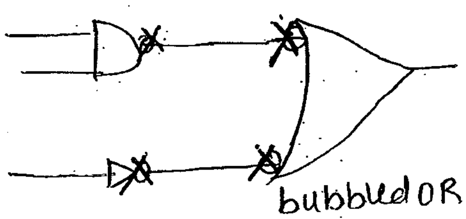


$f_1(x,y,z) = \Sigma(0,1,3,5)$
$f_2(x,y,z) = \Sigma(6,5)$
$f(x,y,z) = \Sigma(1,4,5)$.
f_3 is

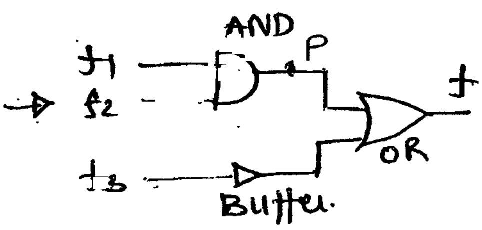
0, 1, 2, 3, 4, 5, 6, 7, 8

✓(a) $\Sigma(1,4,5)$ (b) $\Sigma(6,7)$
(c) $\Sigma(0,1,3,5)$ (d) None.

Sol^n $P = \overline{f_1 \cdot f_2} = \bar{f_1} + \bar{f_2}$
$= \Sigma(2,4,6,7,8) + \Sigma(1,2,3,4,7,8)$ X
$= \Sigma(1,2,3,4,6,7,8)$



→



$P = f_1 \cdot f_2 = f_1 \cap f_2$
$P = \Sigma m(5)$
$f_3 = \Sigma m(1,4,5)$
$f = P \cup f_3$
$\Sigma m(1,4,5) = \Sigma m(5) \cup \Sigma m(\text{——})$.
$\therefore f_3 = \Sigma m(1,4,5)$. Or $\Sigma m(1,4)$.

Q1 consider the following SOP expression F;
$F = ABC + \bar{A}\bar{B}C + A\bar{B}C + \bar{A}BC + \bar{A}\bar{B}\bar{C}$
the equivalent ~~sum of product~~ product of sum expression is:

✓(a) $F = (A+\bar{B}+C)(\bar{A}+B+C)(\bar{A}+\bar{B}+C)$
(b) $F = (A+\bar{B}+\bar{C})(A+B+C)(\bar{A}+\bar{B}+\bar{C})$
(c) $F = (\bar{A}+B+\bar{C})(A+\bar{B}+\bar{C})(A+B+C)$
(d) $F = (\bar{A}+\bar{B}+C)(A+B+\bar{C})(A+B+C)$

Sol^n: $F = ABC + \bar{A}\bar{B}C + A\bar{B}C + \bar{A}BC + \bar{A}\bar{B}\bar{C}$

minterms: $111 \to 7$, $001 \to 1$, $101 \to 5$, $011 \to 3$, $000 \to 0$

$f = \Sigma m(0,1,3,5,7)$

$f = \pi m(2,4,6)$

$2 \to 010 \to A+\bar{B}+C$
$4 \to 100 \to \bar{A}+B+C$
$6 \to 110 \to \bar{A}+\bar{B}+C$

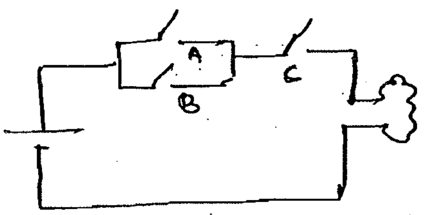
maxterms $f = (A+\bar{B}+C)(\bar{A}+B+C)(\bar{A}+\bar{B}+C)$

done without teacher logic good*

12 statement(I): A NAND gate represents universal logic gate.
statement (II): only two nand gates are sufficient to accomplish all basic gates.

Ans statement (I) → true
statement (II) → false } option (c) statement (I) is true but statement (II) is false.

9 NAND and NOR gates are universal gates because:
(a) they are available everywhere
(b) they are widely used in I.C packages.
Yes (c) they can be combined to produce AND, OR & NOT gate
(d) they can be manufactured easily.



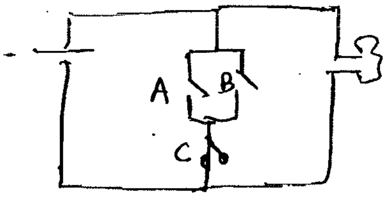
$f = ?$

bulb (1) ⇒ A=1, C=1 ; B don't care
⇒ B=1, C=1, A don't care

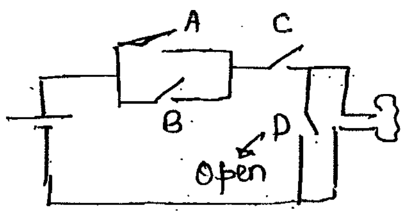
	A	B	C	f
(5)	1	0	1	1
(7)	1	1	1	1
(3)	0	1	1	1
(7)	1	1	1	1

$f = \Sigma m(3,5,7)$
$f = (A\cdot\bar{B}\cdot C) + (\bar{A}BC) + (ABC)$
$f = (A+B)\cdot C$

Q



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



$f = AC\bar{D} + BC\bar{D}$

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

Positive and negative logic

(+)ve logic — 0 → 0V, 1 → 5V

(−)ve logic — 0 → 5V, 1 → 1V

(+)ve logic — low → 0, high → 1

(−)ve logic — low → 1, high → 0.

High, low → (+)ve logic
−2V, −7V
1, 0

(+)ve And logic:

x	y	AND
0	0	0
0	1	0
1	0	0
1	1	1

(−)ve And logic

x	y	AND(−ve)
0	0	1
0	1	1
1	0	1
1	1	0

(+)ve OR logic.

(+)ve AND ≡ (−ve) OR & vice versa.

(+) ve NAND ≡ (−)ve NOR & vice versa.

Duality

$B = \{\cdot, +\}$

$B = \{0, 1\}$

Boolean laws are maintaining Duality.

step 1: Interchange $\{\cdot, +\}$ AND ↔ OR

step 2: Interchange 0, 1.

Ex

$x \cdot 0 = 0$ ✓

$x + 1 = 1$ ✓

Boolean law are valid in duality.

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

→ $f = A + B \cdot \bar{C}$

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

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

→ $(f^D)^D = f$ → it happens for every boolean expression.

if $f^D = f \rightarrow f$ is self dual

Ex = $f = AB + BC + CA$

$f^D = (A+B)\cdot(B+C)\cdot(C+A)$

$f^D = [AB + AC + BB + BC]\cdot(C+A)$

$= (AB + AC + BC + BB)(C+A)$

$f^D = BC + AB + AC$ (on solving) $= f$.

} f is self dual.

) checking of self Duality:

conditions: → it should be a netural function.
→ it should not contain mutually exclusive terms.

netural function:

x	y	z	f(given)	
0	0	0	0	M_0
0	0	1	0	M_1
0	1	0	1	m_2
0	1	1	0	M_3
1	0	0	1	m_4
1	0	1	1	m_5
1	1	0	0	M_6
1	1	1	1	m_7

minterms = m_2, m_4, m_5, m_7

maxterms = M_0, M_1, M_3, M_6

[netural function is no of minterms = no of maxterms]

mutually exclusive terms:

x	y	z
→ 0	0	0
→ 0	0	1
→ 0	1	0
→ 0	1	1
→ 1	0	0
→ 1	0	1
→ 1	1	0
→ 1	1	1

(0,7) (1,6) (2,5) (3,4)

$2 \times 2 \times 2 \times 2 = \underline{16}$

mutually exclusive terms:

$0 \rightarrow 000$, $7 \rightarrow 111$] (0,7) are mutually exclusive terms

$1 \rightarrow 001$, $6 \rightarrow 110$] (1,6)

$2 \rightarrow 010$, $5 \rightarrow 101$] (2,5)

$3 \rightarrow 011$, $4 \rightarrow 100$] (3,4)

: $f = \Sigma m(0,1,2,5)$

f is netural (min term = max term)

f is mutually exclusive (2,5) so it is not self dual.

ote: no of possible self dual forms $= 2^{2^{n-1}}$ (provided that it should be a netural function/form).

for mutually Exclusive terms:

if $n=2$ sum of terms $= 3$
if $n=3$ sum of terms $= 7$
if $n=4$ sum of terms $= 15$

} for Mutually exclusive terms, sum of terms $= (2^n - 1)$

Ex = $f_1 = \Sigma m(0,1,4)$ → not self dual.

$f_2 = \Sigma m(2,3,4,5)$ → not self dual.

$f_3 = \Sigma m(0,1,2,3,4)$ → self dual.

⇨ workbook practice: CH2 : Q22, 23

Q22 for the identity $AB + \bar{A}C + BC = AB + \bar{A}C$, the dual form is

(a) $(A+B)(\bar{A}+C)(B+C) = (A+B)(\bar{A}+C)$ ✓

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

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

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

$\{\cdot, +\}$ ← interchange → ~~$\{0, 1\}$~~

soln $AB + \bar{A}C + BC = AB + \bar{A}C$

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

Q23 The dual ~~form~~ of a boolean function $F(x_1, x_2 \dots x_n, +, \cdot, ^-)$, written as F^D, is the same expression as that of F with + and · swapped. F is said to be self dual if $F = F^D$. The no of self dual functions with n boolean variables is:

(a) 2^n

(b) 2^{n-1}

(c) 2^{2^n}

(d) $2^{2^{n-1}}$

⇨ If $n=1$

$f = A$, $f^D = A$ ①

$f = \bar{A}$, $f^D = \bar{A}$ ②

} no of possible self dual functions $= 2^{2^{n-1}} = 2^{2^{1-1}} = 2^{2^0} = 2$

⊛ complimenting a boolean Expression:

→ step①: Dual form

→ step②: compliment the individual variable.

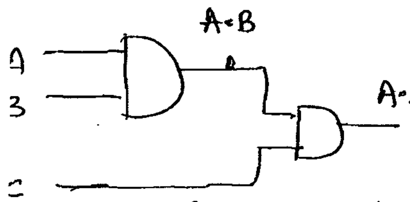
$f = A + B\cdot\bar{C}$

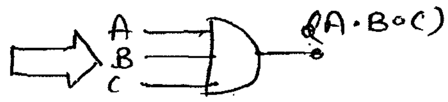
$\bar{f} = ?$

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

$\bar{f} = \bar{A}\cdot(\bar{B}+C)$ ✓

③ Degenerative forms:



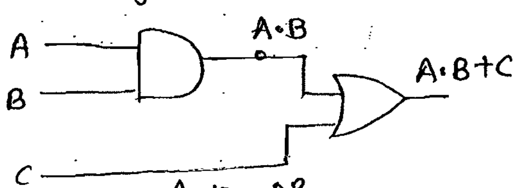


(AND–AND)

AND– AND is a Degenerated form for AND gate.

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 degenerated form for that single logic gate.

Non degenerative form:

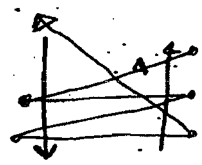


⇨ AND–OR is a non degenerative form

AND–OR.

Non degenerative form list: (shortcut):

AND – OR (8)	⟵ Dual form ⟶	OR – AND (1)
NAND – NAND (2)	⟵ Dual form ⟶	NOR – NOR (3)
NOR – OR (4)	⟵ Dual form ⟶	NAND – AND (5)
OR – ~~NOR~~NAND (6)	⟵ Dual form ⟶	AND – NOR (7)



In the above representation of Non degenerative form combinations present in the same row are dual forms

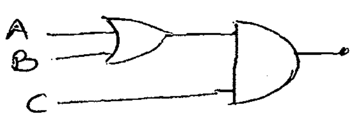
Ex $f = AB + C$

AND–OR

$f = AB + C$

$f^D = (A+B) \cdot C$

OR AND



AND–OR $\xleftrightarrow{\text{dual}}$ OR–AND

⇨ EX-OR: shortcut:

x	y	$f = x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

$$f = x \oplus y = \bar{x}y + x\bar{y}$$

shortcut

$$\left.\begin{array}{l} \overline{x \oplus y} \\ \bar{x} \oplus y \\ x \oplus \bar{y} \end{array}\right\} x \odot y$$

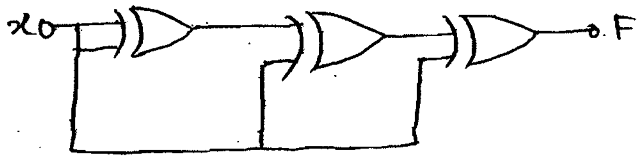
Short-cut (4)

$$\left\{\begin{array}{ll} x \oplus x & = 0 \\ x \oplus \bar{x} & = 1 \\ x \oplus 1 & = \bar{x} \\ x \oplus 0 & = x \end{array}\right.$$

$$\begin{aligned} x \oplus 1 &= \bar{x}\cdot 1 + x \cdot \bar{1} \\ &= \bar{x} + x \cdot 0 \\ &= \bar{x} + 0 \\ x \oplus 1 &= \bar{x} \end{aligned}$$

⇨ workbook chapter 2: Q2, Q4

Q2 for the circuit shown below the output F is given by:



(a) $F = 1$
(b) $F = 0$ ✓
(c) $F = X$
(d) $F = \overline{X}$

sol^n
$x \oplus x = 0$ (o/p of gate 1)
$0 \oplus x = x$ (o/p of gate 2)
$x \oplus x = 0$ (o/p of gate 3)
$\therefore F = 0$

Q4 consider the following boolean operator # with the following properties
$x \# 0 = x$, $x \# 1 = \bar{x}$, $x \# x = 0$ and $x \# \bar{x} = 1$. Then $x \# y$ is equivalent to:

(a) $x\bar{y} + \bar{x}y$ ✓ (b) $x\bar{y} + \bar{x}\bar{y}$ (c) $\bar{x}y + xy$ (d) $xy + \bar{x}\bar{y}$.

sol^n $x \# y = x \oplus y = x\bar{y} + \bar{x}y$

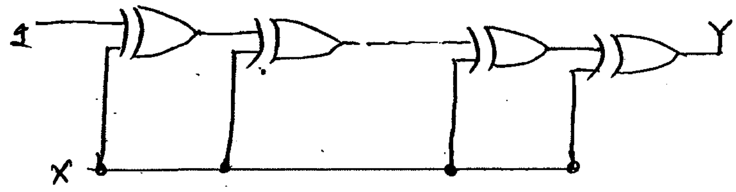
⇨ EX-NOR: shortcut:

shortcut (5)

$$\left\{\begin{array}{lll} x \odot x & = & \Rightarrow 1 \\ x \odot \bar{x} & = & \Rightarrow 0 \\ x \odot 1 & = & \Rightarrow x \\ x \odot 0 & = & \Rightarrow \bar{x} \end{array}\right.$$

⇨ workbook: CH2: Q17

Q17 If the input to the digital circuit of following circuit consisting of cascade of 20 XOR gates is x. Then what is the output of Y?



(a) 0
✓(b) 1
(c) x'
(d) x

Sol^n $1 \oplus x = \bar{x}$ (output of gate 1) → odd no gate
$x \oplus \bar{x} = 1$ (output of gate 2) → even no gate.

∴ o/p of gate 19 = $\bar{x}$
and o/p of gate 20 = 1
∴ $\underline{Y = 1}$.

→ for 2-variables:

EX–OR

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

$$\boxed{\begin{array}{l} \overline{x \oplus y} = x \odot y \\ \overline{\text{EX-OR}} = \text{EX-NOR} \end{array}}$$

EX–NOR

x	y	$x \odot y$
0	0	1
0	1	0
1	0	0
1	1	1

→ for 3-variables:

EX–OR → 0 → same i/p; 1 → diff i/p

x	y	z	$x \oplus y$	$x \oplus y \oplus z$
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	0
1	1	1	0	1

EX–NOR → 0 → diff i/p; 1 → same i/p

x	y	z	$x \odot y$	$x \odot y \odot z$
0	0	0	1	0
0	0	1	1	1
0	1	0	0	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	1	0
1	1	1	1	1

EXNOR = EX-OR

→ conclusion:

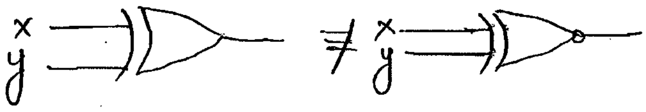
- for 2 variables,
  $\overline{x \oplus y} = x \odot y$ i.e. $\overline{\text{EX-OR}}$ = EX-NOR
- for 3 variables,
  $\overline{x \oplus y \oplus z} \neq \cancel{x \odot y \odot z}\ x \odot y \odot z$ but EX-OR = EXNOR

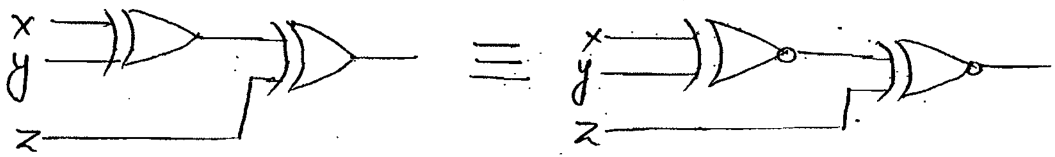
• for 4-variables,

$\overline{w\oplus x\oplus y\oplus z} = w\odot x\odot y\odot z$. ie $\overline{\text{EX-OR}}$ = EX-NOR.

Hence,

$$\boxed{\begin{array}{l}\overline{\text{EX-OR}} = \text{EX-NOR} \rightarrow \text{for even variables}\\ \text{EX-OR} = \text{EX-NOR} \rightarrow \text{for odd variables}\end{array}}$$





Note: two inputs EX-OR, EX-NOR gates are only ~~avialable~~ available.

⇨ **logic minimization techniques:**

→ venn diagrams.

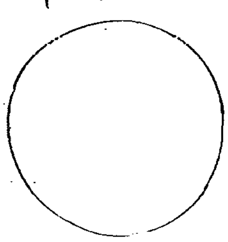
→ Boolean algebra.

→ K map.

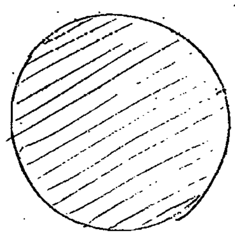
→ Tabulation method (Mc- clusky).

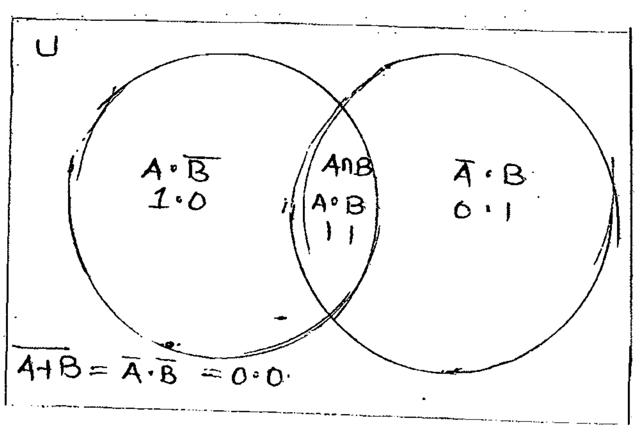
⇨ **venn diagrams:** (+ve logic)

A=0 → unshadded



A=1 → shaded.



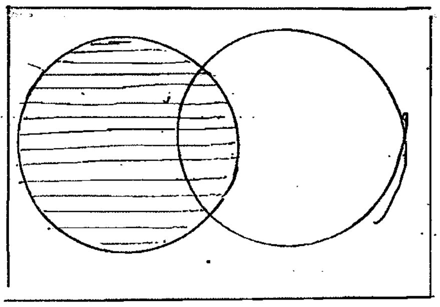


$A\cdot\bar{B}$ → inhibition → A but not B

$\bar{A}\cdot B$ → inhibition → B but not A.

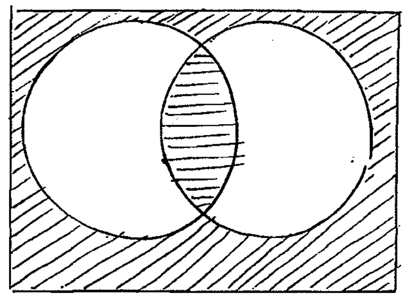
$A\cap B$ → A and B → $A\cdot B$
↳ (AND Gate)

1



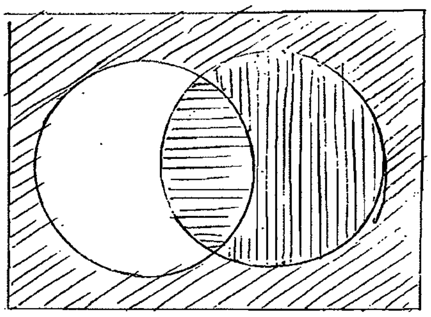
$f = A\cdot\bar{B} + A\cdot B$
$= A\cdot(\bar{B} + B)$
$= A\cdot(1)$
$= A$

2



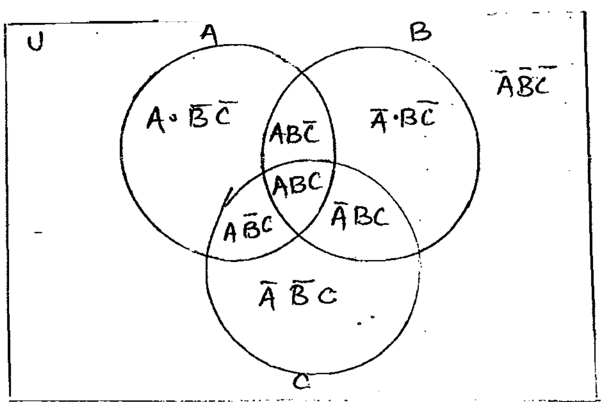
$f = A\cdot B + \bar{A}\cdot\bar{B}$
EX-NOR
unshaded region
$= f = \bar{A}B + A\bar{B}$
$= \overline{A \oplus B}$
$= A \odot B$

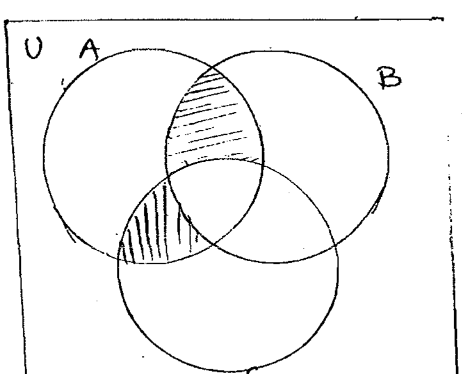
3



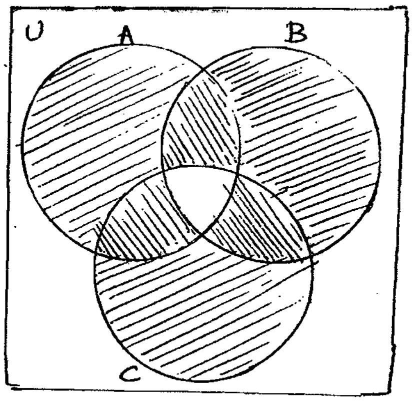
$f = A\cdot B + \bar{A}\cdot B + \bar{A}\cdot\bar{B}$
$= (A+\bar{A})\cdot B + \bar{A}\cdot\bar{B}$
$= B(1) + \bar{A}\cdot\bar{B}$
$= B + \bar{A}\cdot\bar{B}$
$= B + \bar{A} = \bar{A} + B$

unshaded region $= \overline{A\cdot\bar{B}}$
$= \bar{A} + \bar{\bar{B}}$
$= \bar{A} + B$ ~~un~~ shaded area





$f = A\bar{B}\bar{C} + A\bar{B}C$
$= A(\bar{B}\bar{C} + \bar{B}C)$
$= A(B \oplus C)$



shaded area $\Rightarrow f = A\bar{B}\bar{C} + AB\bar{C} + \bar{A}B\bar{C}$
$+ \bar{A}BC + \bar{A}\bar{B}C + A\bar{B}C$

unshaded area $\Rightarrow \overline{\bar{A}\bar{B}\bar{C}} + ABC$

⇨ functional completeness:

NAND } universal logic gates
NOR } ↳ (Any function completely expressed by these gates).

NAND → AND + NOT
NOR → OR + NOT.

{AND, NOT} → functionally completed
{OR, NOT} → functionally computed.

Eg: $f(A,B,C) = \bar{A} + B\bar{C}$ → checking of NOT operation

→ {·, NOT}
{+, NOT}

$f(A,A,A) = \bar{A} + A\cdot\bar{A}$
$= \bar{A} + 0$
$= \bar{A}$ → NOT ✓

similiarly, $f(B,B,B) = \bar{B}$ and $f(C,C,C) = \bar{C}$

$f(A,B,C) = \bar{A} + B\cdot\bar{C}$
$f(\bar{A},B,\bar{B}) = \bar{\bar{A}} + B\cdot\bar{\bar{B}}$
$= A + B\cdot B$
$= A + B$ → OR ✓

since we get {+, NOT}
∴ f is functionally completed.

Eg: $f(A,B) = \bar{A} + B$
→ $f(A,A) = \bar{A} + A$
$= 1$ NOT ✗

f is not functionally completed.
Hence, f is functionally-incomplete