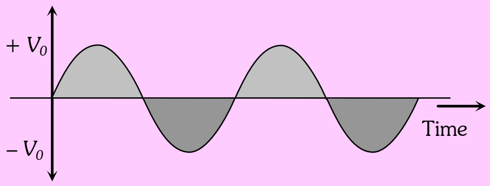
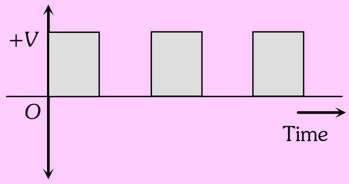
Digital Electronics

Voltage Signal and Binary System.

(1) Voltage signal

Analogue voltage signal	Digital voltage signal
The signal which represents the continuous variation of voltage with time is known as analogue voltage signal	The signal which has only two values. *i.e.* either a constant high value of voltage or zero value is called digital voltage signal





(2) Binary system

(i) A number system which has only two digits *i.e.* 0 (Low value) and 1 (High value) is known as binary system

(ii) The electrical circuit which operates only in these two state *i.e.* 1 (On or High) and 0 (*i.e.* Off or Low) are known as digital circuits.

(iii) Different names for the two states of digital signals :

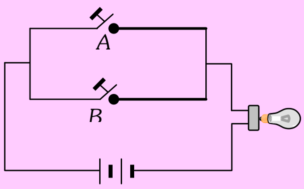
State Code	Name for the State							
1	On	Up	Closed	Excited	True	Pulse	High	Yes
0	Off	Down	Open	Unexcited	False	No pulse	Low	No

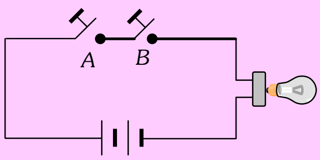
Boolean Algebra.

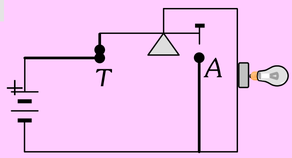
(1) In Boolean algebra only two states of variables (0 and 1) are allowed.

(2) The variables (A, B, C ….) of Boolean Algebra are subjected to three operations.

	OR Operation	AND Operation	NOT Operation
(i)	Represented by (+) sign	Represented by (·) sign	Represented by bar over the variables
(ii)	Boolean expression $Y = A + B$	Boolean expression $Y = A \cdot B$	Boolean expression $Y = \overline{A}$







A OFF $\rightarrow$ Lamp ON
A ON $\rightarrow$ Contact at T is

(3) Basic Boolean postulates and laws

(i) Boolean Postulates : $0 + A = A$, $1 \cdot A = A$, $1 + A = 1$, $0 \cdot A = 0$, $A + \overline{A} = 1$

(ii) Identity law : $A + A = A$, $A \cdot A = A$

(iii) Negation law : $\overline{\overline{A}} = A$

(iv) Commutative law : $A + B = B + A$, $A \cdot B = B \cdot A$

(v) Associative law : $(A+B) + C = A + (B+C)$, $(A \cdot B) \cdot C = A \cdot (B \cdot C)$

(vi) Distributive law : $A \cdot (B+C) = A \cdot B + A \cdot C$

(vii) De Morgan's laws : $\overline{A+B} = \overline{A} \cdot \overline{B}$ and $\overline{A \cdot B} = \overline{A} + \overline{B}$ also $A + \overline{A}B = A + B$ and $A(\overline{A} + B) = AB$

Logic Gates and Truth Table.

(1) **Logic gate :** The digital circuit that can be analysed with the help of Boolean algebra is called logic gate or logic circuit. A logic gate has two or more inputs but only one output.

There are primarily three logic gates namely the OR gate, the AND gate and the NOT gate.

(2) **Truth table :** The operation of a logic gate or circuit can be represented in a table which contains all possible inputs and their corresponding outputs is called the truth table. To write the truth table we use binary digits 1 and 0.

Different Logic Gates.

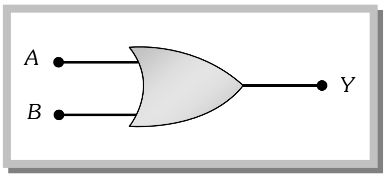
(1) The 'OR' gate

(i) It has two inputs (A and B) and only one output (Y)

(ii) Boolean expression is $Y = A + B$

(iii) Truth table and logic symbol

A	B	$Y = A + B$
0	0	0
0	1	1
1	0	1
1	1	1



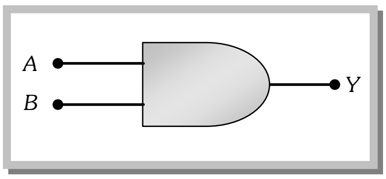
(2) The 'AND' gate

(i) It has two inputs and one output.

(ii) Boolean expression is $Y = A \cdot B$

(iii) Truth table and logic symbol :

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



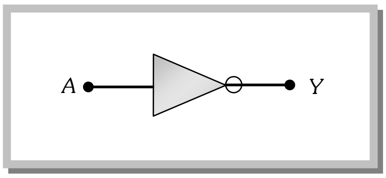
(3) **The 'NOT' gate**

(i) It has only one input and only one output

(ii) Boolean expression is $Y = \overline{A}$

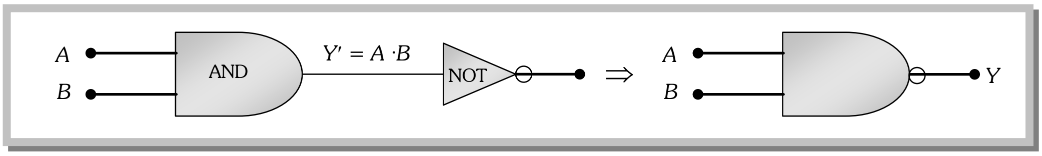
(iii) Truth table and logic symbol :

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



Combination of Logic Gates.

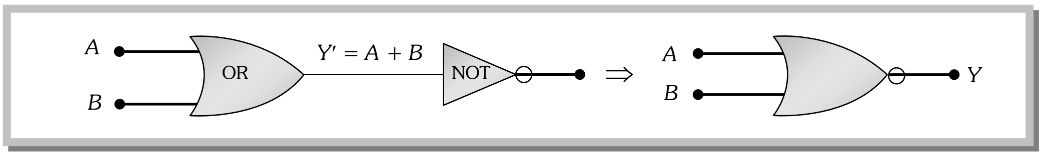
(1) **The 'NAND' gate :** From 'AND' and 'NOT' gate



Boolean expression and truth table : $Y = \overline{A \cdot B}$

A	B	$Y' = A \cdot B$	Y
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

(2) **The 'NOR' gate :** From 'OR' and 'NOT' gate



Boolean expression and truth table : $Y = \overline{A + B}$

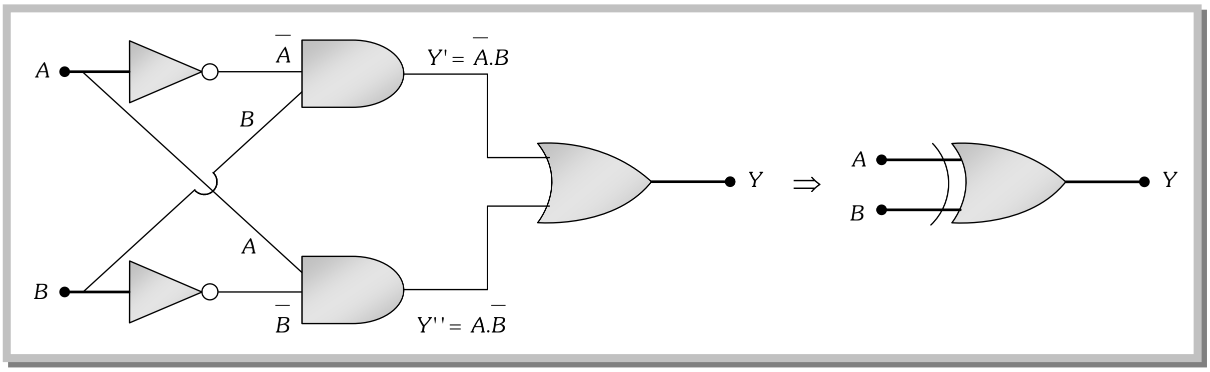
A	B	$Y' = A + B$	Y
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

(3) **The 'XOR' gate :** From 'NOT', 'AND' and 'OR' gate. Known as exclusive OR gate.

or

The logic gate which gives high output (*i.e.*, 1) if either input A or input B but not both are high (*i.e.* 1) is called exclusive OR gate or the XOR gate.

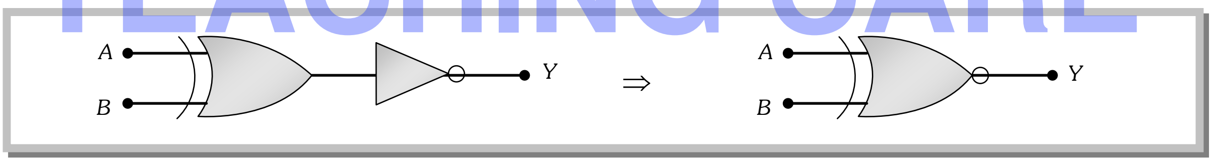
It may be noted that if both the inputs of the XOR gate are high, then the output is low (*i.e.*, 0).



Boolean expression and truth table : $Y = A \oplus B = \overline{A}B + \overline{AB}$

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

(4) **The exclusive nor (XNOR) gate :** XOR + NOT ⟶ XNOR



Boolean expression : $Y = A \odot B = \overline{A}\,\overline{B} + AB$

Logic Gates Using 'NAND' Gate.

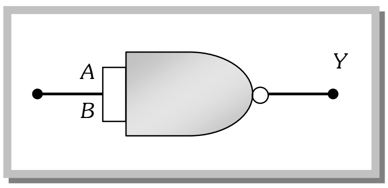
The NAND gate is the building block of the digital electronics. All the logic gates like the OR, the AND and the NOT can be constructed from the NAND gates.

(1) **Construction of the ' NOT' gate from the ' NAND' gate**

(i) When both the inputs (A and B) of the NAND gate are joined together then it works as the NOT gate.

(ii) Truth table and logic symbol

Input	Output
$A = B$	Y
0	1
1	0

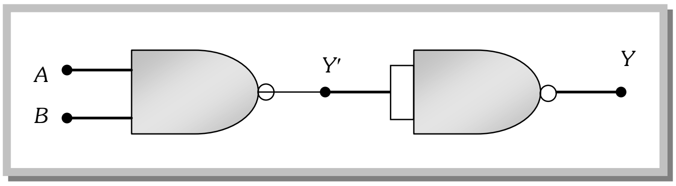


(2) **Construction of the 'AND' gate from the ' NAND' gate.**

(i) When the output of the NAND gate is given to the input of the NOT gate (made from the NAND gate), then the resultant logic gate works as the AND gate

(ii) Truth table and logic symbol

A	B	Y′	Y
0	0	1	0
0	1	1	0
1	0	1	0
1	1	0	1



(3) **Construction of the ‘ OR’ gate by the ‘NAND’ gate**

(i) When the outputs of two NOT gates (obtained from the NAND gate) is given to the inputs of the NAND gate, the resultant logic gate works as the OR gate

(ii) Truth table and logic symbol

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