CHAPTER ELEVEN

LOGIC GATES:

Logic Values:

-There are always two discrete states, and every logic value must assume one of them.

-These states are 1 and 0.

-There is no intermediate state which is neither 1 nor 0.

- Each logic input or output can exist in only one state at any one time.

- Each logic state has an inverse or a complement, which is the opposite of its current state.

- The complement of a true (one) state is a false (zero) state and vice versa.

-Logic value can be a constant or a variable.

- If it is a constant, then it remains in that state and if it is variable, then it may be switched between 0 and 1.

- A variable is often named by the action it causes to take place.

- The following logical variables are all self evident, i.e., start, stop, reset, count and add.

- The logical value which causes a variable to carry out the function suggested by its name is arbitrary.

- If a logical one causes the action, then this variable is called active high and if it is logic zero, then the variable is called active low.

- Thus if an active high variable is labeled start, a logical one (start or1) will initiate the action.

-If it is active low and it has been labeled start, then a logical zero will trigger the action.

Basic gates:

-The properties of the five basic gates from which all digital systems are constructed will be defined.

-The word gate conveys the idea of a two state device, which can be opened or closed.

- Also, a gate may also be thought of as a black box with one or more input terminals and an output terminal.

-The digital signal at the input terminal is processed by the gate, to produce a digital signal at its output terminal.

-The nature of the gate determines the actual processing which takes place.

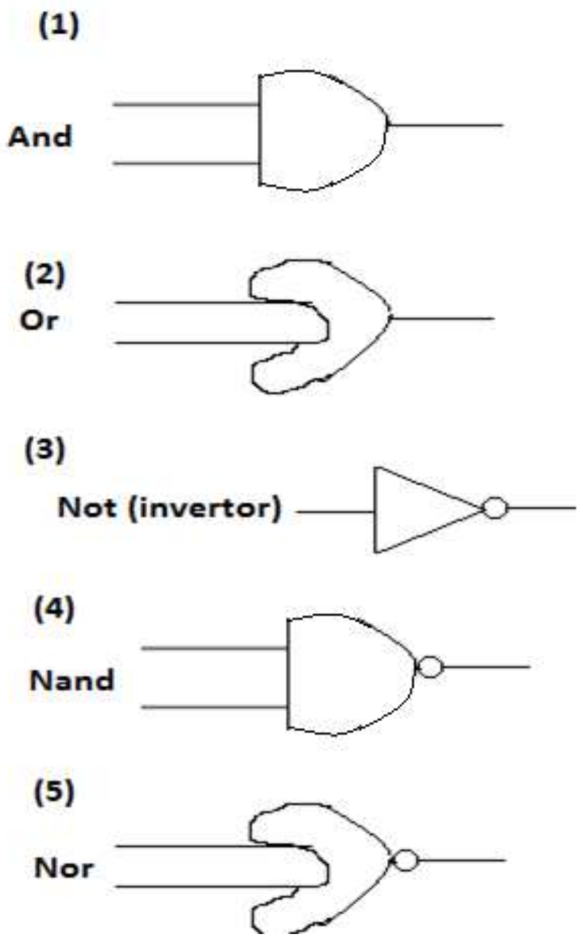
-If a gate has two input terminals A and B and an output terminal C, then the output may be written as $C = F(A,B)$.

- While F is the logical function, A and B are the values.

-Logic gates can be constructed using simple switches, relays, transistors, diodes or ICS.

-ICS are commonly used because of their low cost.

Gates and their symbols:



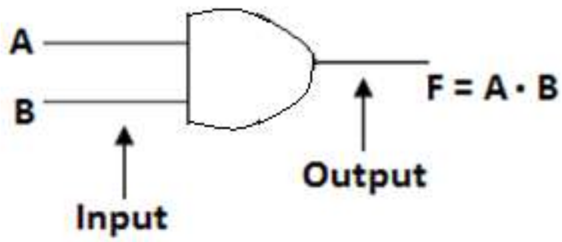
The nand and the nor gates are referred to as the universal gates.

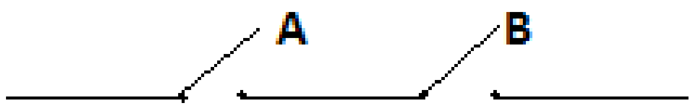
Truth table:

-This is a table of all the possible combinations of all the inputs to a gate, and the resulting outputs.

-Such a table enables us to check the operation of a logic gate system, and derive the Boolean equations.

The And gate:





-Above shows the switching circuit of the And gate.

-The output of the And gate is true if and only if each of its inputs is true.

-Using the electric circuit as illustration, the flow of current occurs if switches A and B are closed.

-The logic symbol for And is the dot and for this reason, A and B is written as A ·B or A ·B.

-A useful way of describing the relationship between the inputs of gates and their output, is by means of a truth table.

The truth table for the And gate (operation):

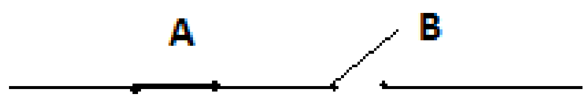
A	B	F = A ·B
0	0	0
0	1	0
1	0	0
1	1	1

NB: While zero represents false, one represents true.

- For example with reference to an electric circuit, an input of 0represents an open circuit while one represents a closed circuit.

Example (1). An input of A = 0 and B = 1 can be represented by

Example (2).An input of A = 1 and B = 0 can be represented by

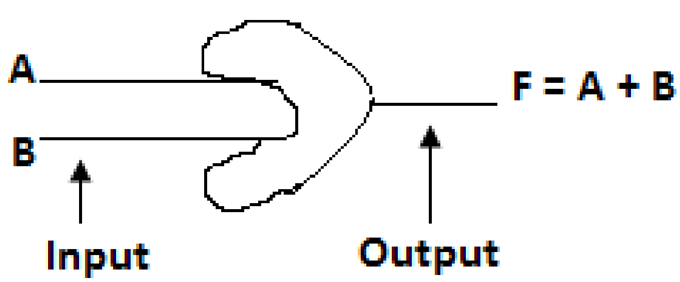


Example (3).An input of A = 1 and B = 1 can be represented by

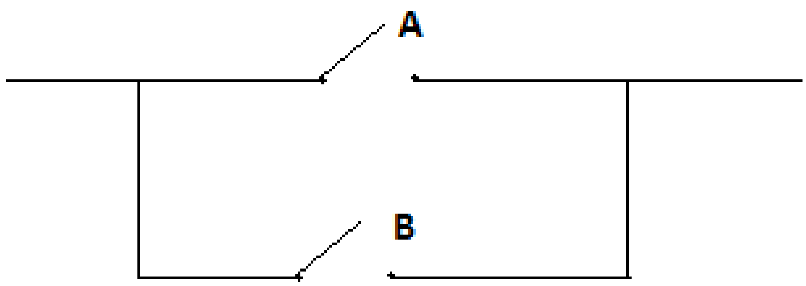
-With reference to the output, an output of 1 implies that current will flow or is flowing through the circuit.

- Lastly, an output of zero, is an indication that no current is flowing through the circuit.

<u>The or gate:</u>



-The Or gate can be represented by using the circuit diagram given next:



-From this circuit diagram, it can be seen that the output of an Or gate is true if one or both of the inputs is true, since the circuit is complete if either A or B is closed.

-The logical symbol for the Or operation is + so that A Or B = A + B.

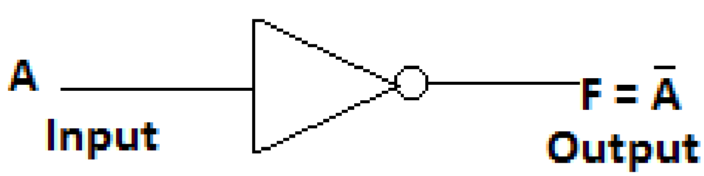
-The use of or here differs from its English meaning.

-The output is also true if both A and B are true.

The truth table for the Or gate:

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

The Not gate:

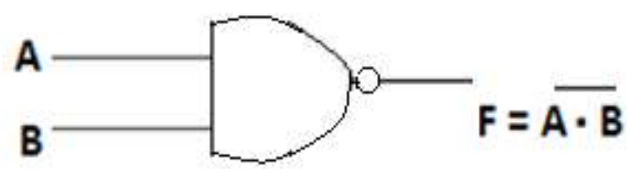


-The Not gate (inverter or complementor) inverts the state of the signal at its input.

Truth table for the Not gate or operation:

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

The Nand gate:



-The output had in the case of the Nand gate, is the inverse or opposite to that had with respect to the And gate, i.e. by assuming the gate to be a And gate.

-For example, to determine the output of a Nand gate, first determine what the output will be if the gate is an And gate.

-Finally, determine the inverse of this output.

-Assuming the output of the And gate is 1, then the output of the And gate will be 0.

Truth table for the Nand gate (operation):

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

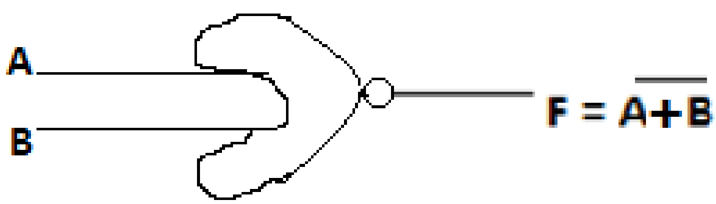
0	1	1
1	0	1
1	1	0

N/B: If A = 0 and B = 0, and assuming that the gate to be the And gate, then F = 0 and its inverse is F = $\overline{A.B}$ will be 1.

-Also, if the gate is the And gate and A = 1 while B = 1, then the output will be 1.

-The inverse of this output which is F = $\overline{A.B}$ will be 0.

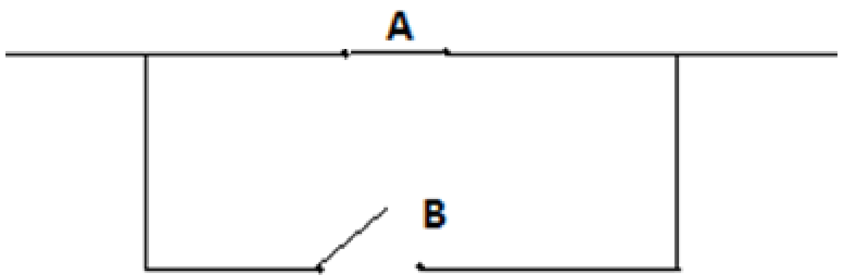
<u>The Nor gate:</u>



-The output of the Nor gate will be the inverse of that had, if the gate happens to be an Or gate.

Example (1):

If the gate under consideration is the Or gate, with A = 1 and B= 0, then the electric circuit diagram will be

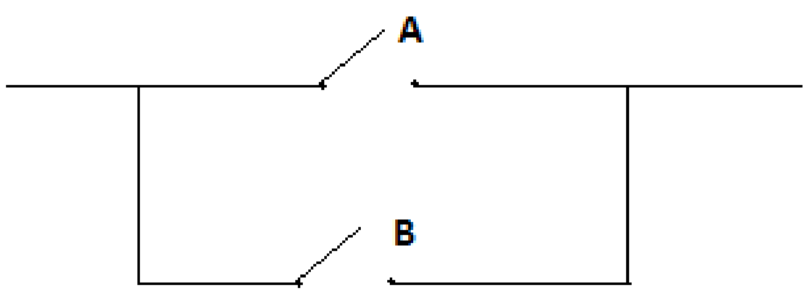


-The output will therefore be 1, and the output of the Nor gate associated with this case will therefore be 0.

-This implies that if the logic gate happens to be a Nor gate, then its output which is F = $\overline{A. B}$ will be 0.

Example2:

-If the gate is the Or gate with A = 0 and B = 0, then the electric circuit will be



-The output will be 0, and as such the output of the Nor gate associated with this case will be 1.

-This means that its output which is F = $\overline{A + B}$ will be 1, if the Or gate is replaced with a Nor gate.