Unit 2

Digital Signals:

Digital signal is a signal that is being used to represent data as a sequence of discrete values. In most digital circuits, the signal can have two possible valid values; this is called a **binary signal** or **logic signal**. They are represented by two voltage bands: one near a reference value (typically termed as *ground* or zero volts), and the other a value near the supply voltage. These correspond to the two values "zero" and "one" (or "false" and "true").

Example of digital wave form: 0101100100

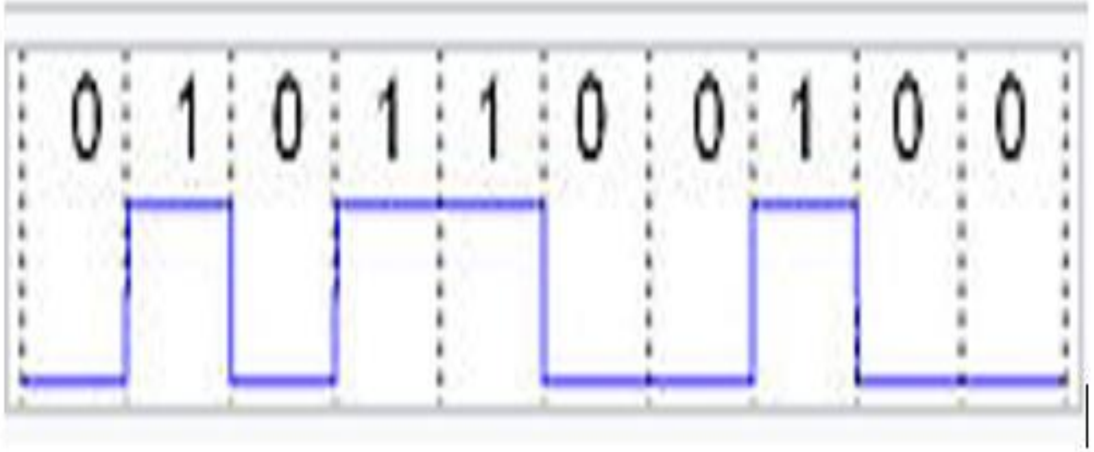


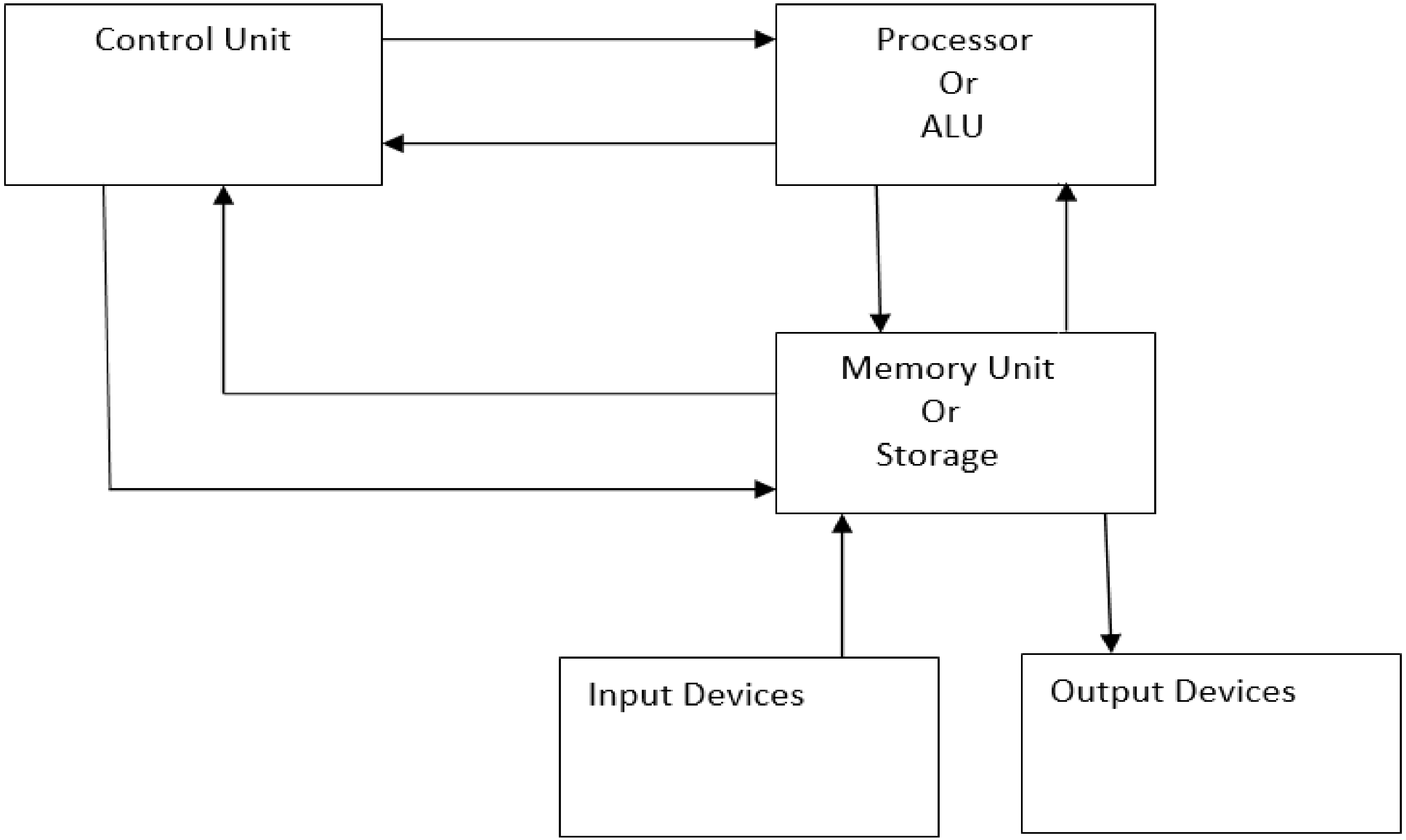
Fig: Digital Wave Form

Analog signal: Analog signal represents continuous values; at any given time it represents a real number within a continuous range of values.

Digital System: Digital system is data technology that uses discrete (discontinuous) values. This system operates on binary digits (signals) 0 and 1. Working of digital computers, communication systems, calculators etc. are based on digital techniques and these systems are known as digital system.

Analog System: An analog system is a system that use analog i.e. continuous range of values to present information. Like voltmeter, automobile speedometer are the example of analog system.

Block diagram of digital computer



Functions or working principle of digital computer:

1. Memory unit stores programs as well as input and output.
2. The control unit supervises the flow of information between various units and retrieves the instructions stored in memory unit.
3. After getting control signal from control unit, memory unit sends the data to the processor.
4. For each instruction, control unit informs the processor to execute the operation according to the instructions.
5. After getting control signal processor sends the processed information to memory unit sends that information to the output unit.

Advantages of digital system:

1. In case of digital system large numbers of ICs are available for performing various operations, hence digital systems are highly reliable, accurate, small in size and speed of operation is very high.
2. Computer controlled digital systems can be controlled by software that allows new functions to be added without changing hardware.

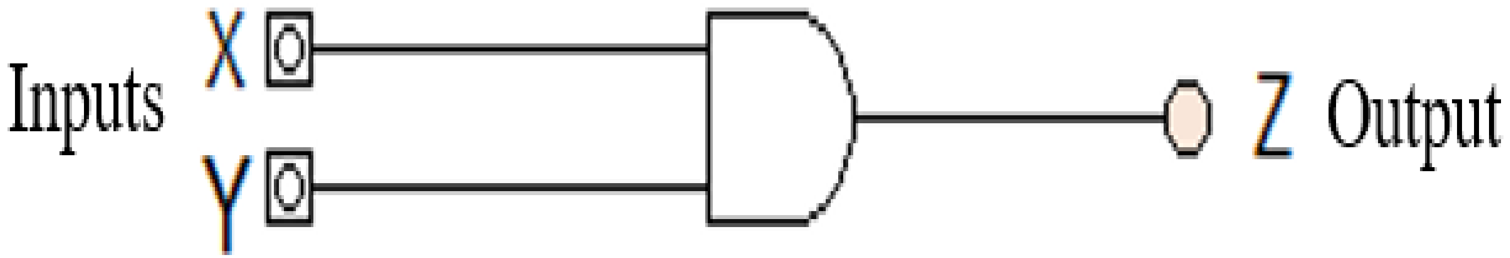
Disadvantages of digital system:

1. It is difficult to install digital system, because it requires many more complex electronic circuits and ICs.
2. In digital systems, if a single piece of digital data lost, large blocks of related data can completely change.

Logic gates: Logic gates are the basic building blocks of digital computer. Logic gates have one or more than one input and only one output. Input and output values are the logical values true (1) and false (0). Logic gates are also called combinational logic circuits. Basic logic gates are AND, OR and NOT.

1. **AND gate:** The output of AND gate is 1 if and only if all of the inputs are 1, otherwise the output value is 0.

Logic diagram of AND gate:



Algebraic Function:

Z = X AND Y

= X.Y

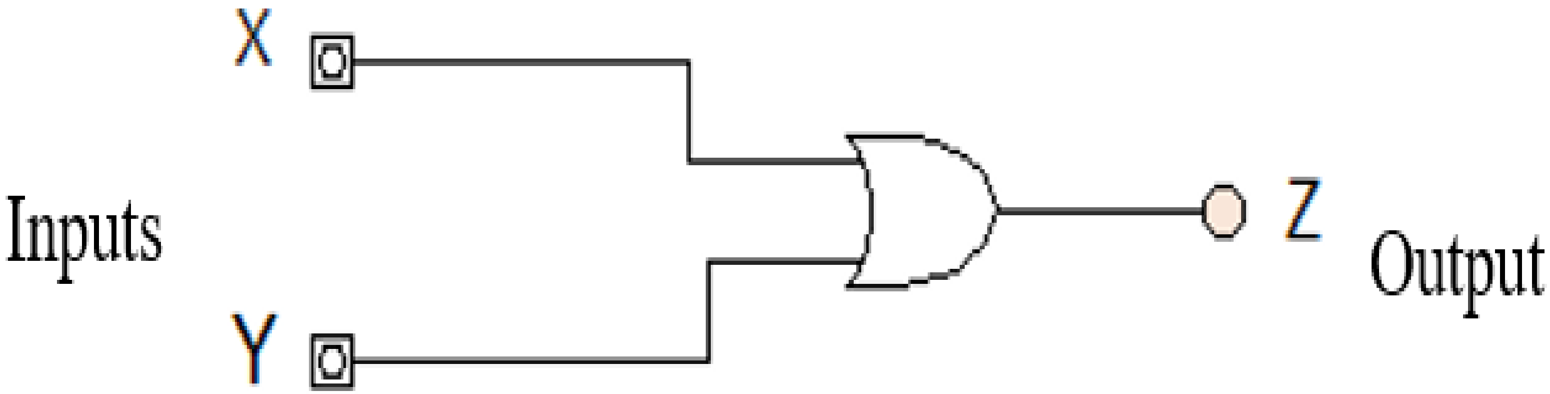
Truth Table:

2^n combination of variables. i.e. $2^2 = 4$ binary Combinations from 0 to 3.

Inputs		Outputs
X	**Y**	**Z**
0	0	0
0	1	0
1	0	0
1	1	1

2. **OR gate:** The output of OR gate is 1 if any one of the input value is 1 and output is 0 if all the inputs are 0.

Logic diagram of OR gate:



Algebraic Function:

$Z = X \text{ OR } Y$ or $X + Y$

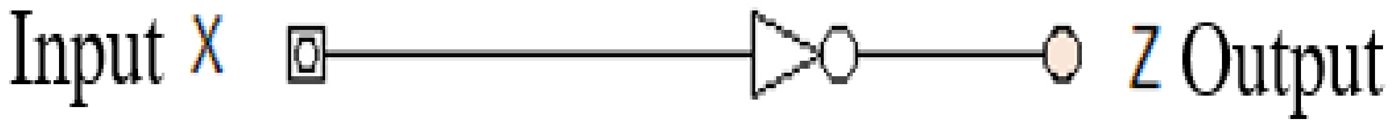
Truth Table:

2^n combination of variables. i.e. $2^2 = 4$ binary Combinations from 0 to 3.

Inputs		Outputs
X	**Y**	**Z**
0	0	0
0	1	1
1	0	1
1	1	1

3. NOT gate (inverter): A NOT has only one input and one output the output of NOT gate is the inversion of the input i.e. complement of the input.

Logic diagram of NOT gate:



Algebraic Function:

$$Z = \text{NOT } X$$
$$= \bar{X}$$

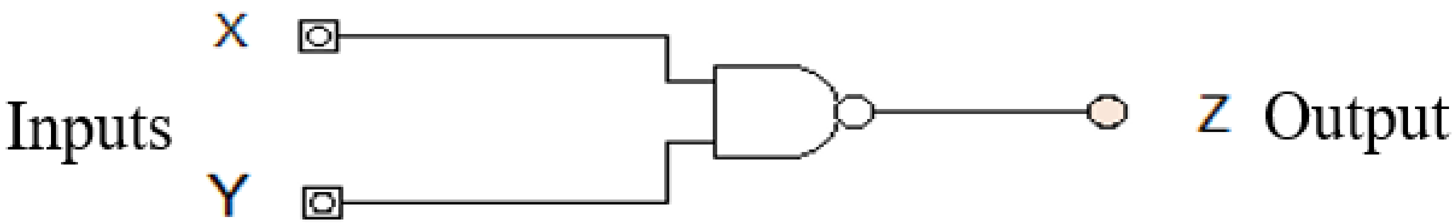
Truth Table:

Input	Output
X	Z
0	1
1	0

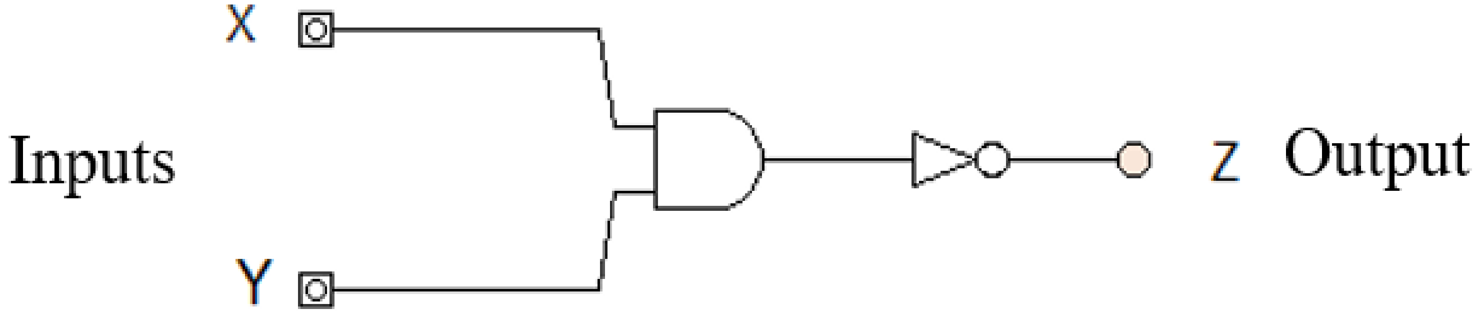
Combined Gates:

NAND gate: NAND gate is the combination of AND and NOT gates. An AND gate with inverter at the output. The output of NAND gate is the complement of AND gate.

Logic Diagram of NAND gate:



Or



Algebraic Function:

$$Z = \overline{X \text{ AND } Y}$$
$$= (X.Y)'$$

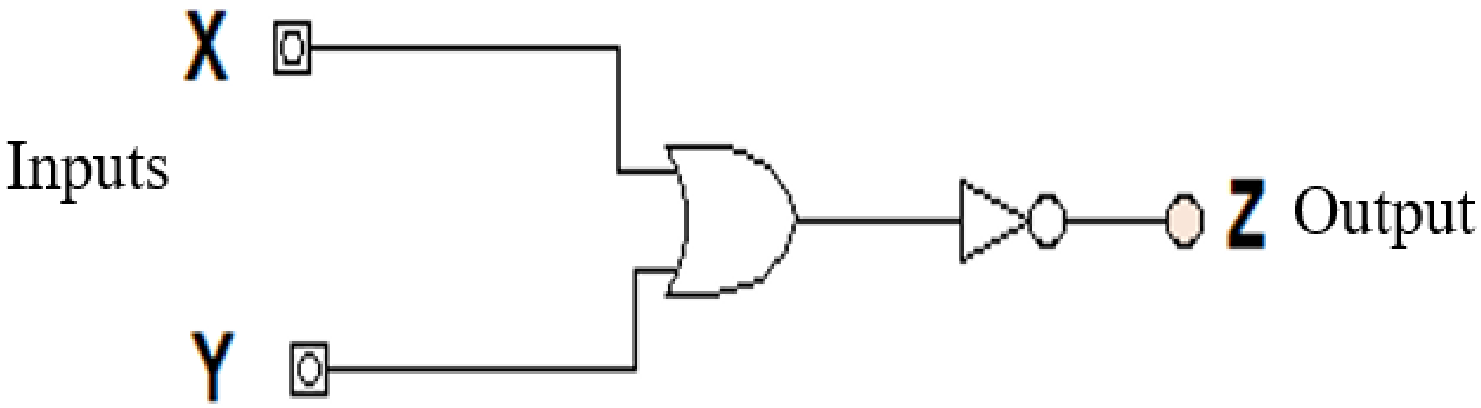
Truth Table:

2^n combination of variables. i.e. $2^2 = 4$ binary Combinations from 0 to 3.

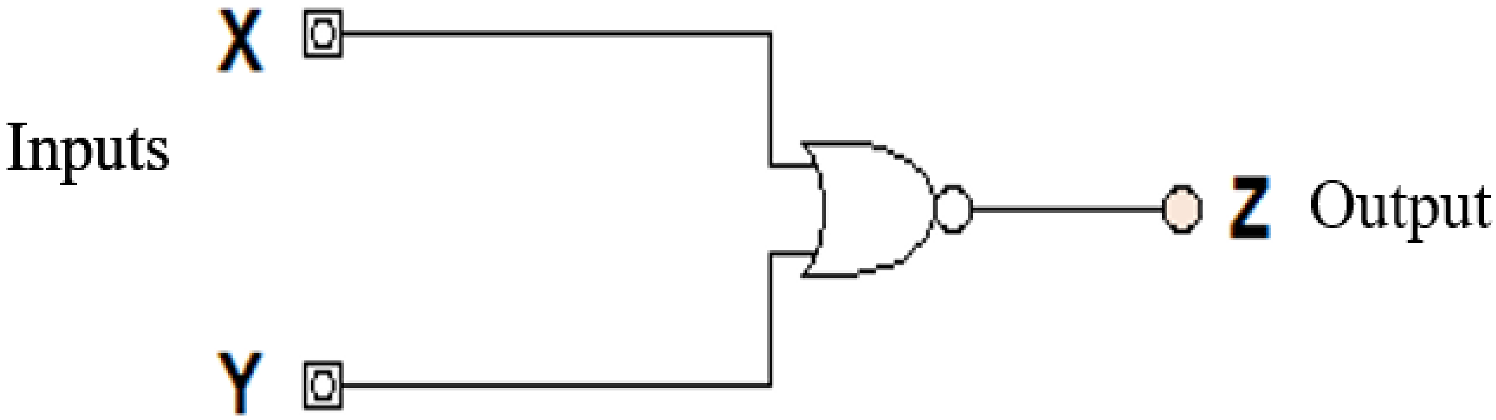
Inputs		Outputs
X	**Y**	**Z**
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gate: NOR gate is the combination of OR and NOT gate. The output of NOR gate is the complement of OR gate.

Logic Diagram of NAND gate:



Or



Algebraic Function:

$$Z = \overline{X \text{ OR } Y}$$
$$= (X+Y)'$$

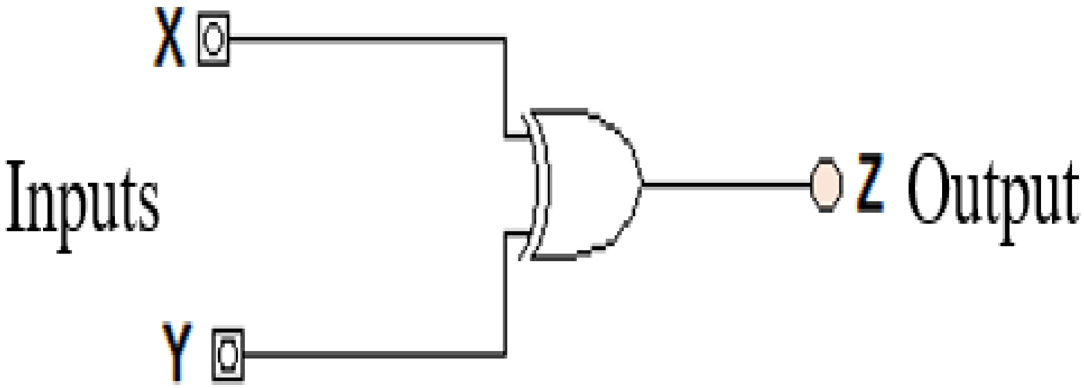
Truth Table:

2^n combination of variables. i.e. $2^2 = 4$ binary Combinations from 0 to 3.

Inputs		**Outputs**
X	**Y**	**Z**
0	0	1
0	1	0
1	0	0
1	1	0

Exclusive – OR (XOR) Gate: An XOR gate gives an output value 1 when there are different input values and the output value is low when there are same input values.

Logic Diagram of XOR gate:



Algebraic Function:

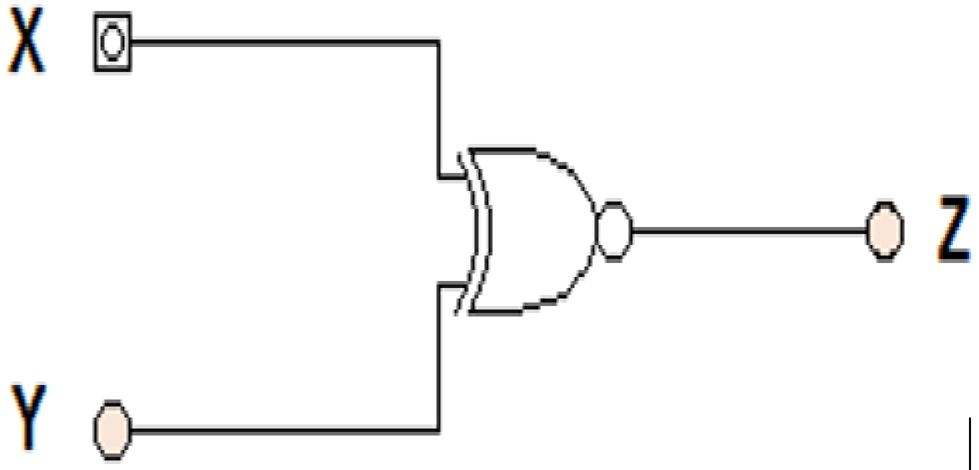
$$Z = X\overline{Y} + \overline{X}Y$$

Truth Table:

Inputs		Outputs
X	**Y**	**Z**
0	0	0
0	1	1
1	0	1
1	1	0

Exclusive – NOR (X-NOR) Gate: An X-NOR gate gives an output value 1 when there are same input values and the output value is low when there are different input values.

Logic Diagram of XNOR gate:



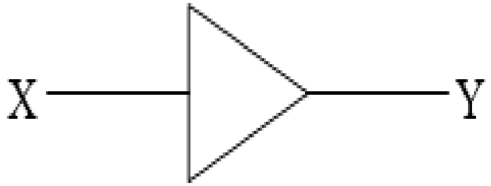
Algebraic Function:

$Z = XY + \bar{X}\bar{Y}$

Truth Table:

Inputs		Outputs
X	**Y**	**Z**
0	0	1
0	1	0
1	0	0
1	1	1

Buffer: Buffer produces the transfer function but does not produce any particular logic operation, since the binary value of the output is equal to the binary value of the input. Buffer will delay the time between input and output. It can also amplify the signal if the current is too weak.



Logic function: Y = X

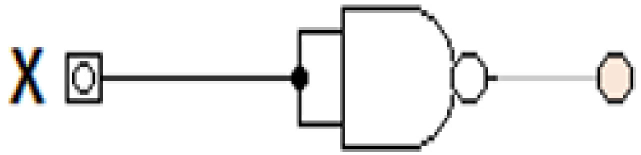
Truth Table:

Input	Output
X	Y
0	0
1	1

Universal Gates: NAND and NOR gates are called universal gates because, we can build any gate using NAND or NOR gates. Basic logic gates AND, OR and NOT can be realized by using only NAND or NOR gates.

NAND as a universal gate:

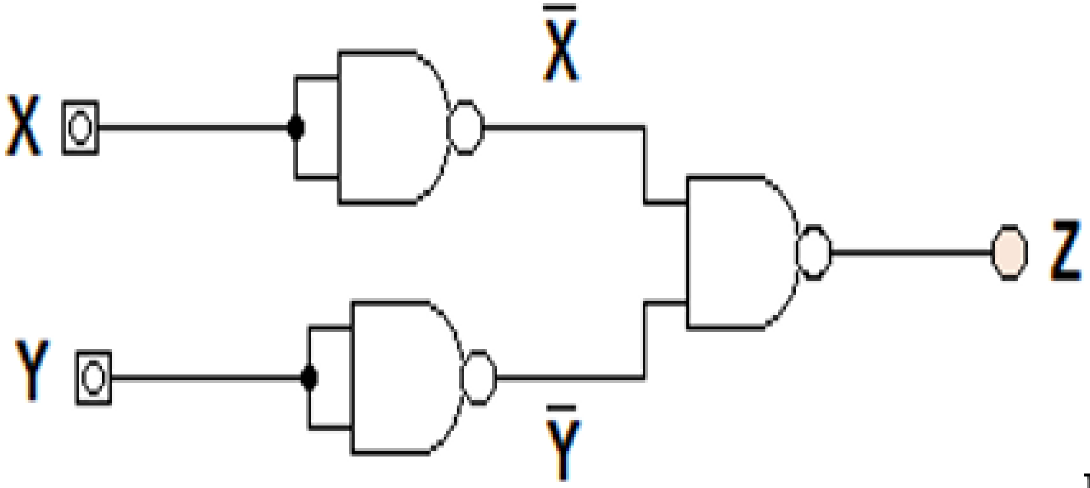
i) **NOT gate:** NOT gate can be realized using single input NAND gate.



$Y = \overline{X}$

Properties of NOT gate.

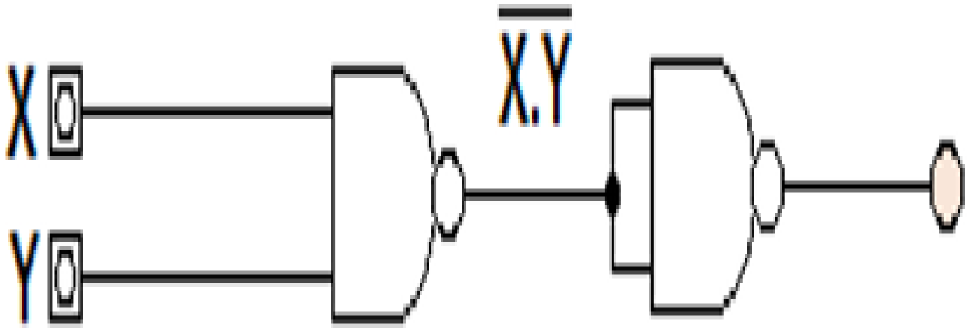
ii) **OR gate:** OR gate can be realized using three NAND gates.



$Z = \overline{\overline{X}.\overline{Y}} = \overline{\overline{X}} + \overline{\overline{Y}} = X + Y$

Properties of OR gate.

iii) **AND gate:** AND gate can be realized using two NAND gates.

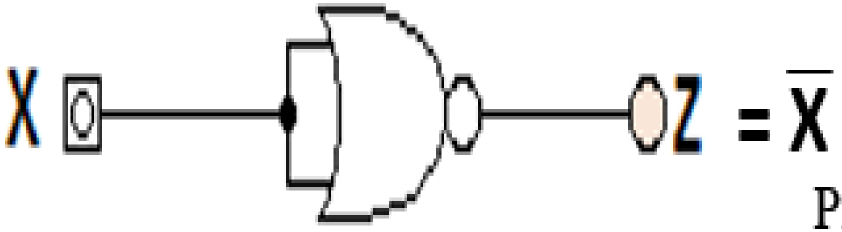


$$Z = \overline{\overline{X.Y}} = \overline{\overline{X}+\overline{Y}} = \overline{\overline{X}}.\overline{\overline{Y}} = X.Y$$

Properties of AND gate.

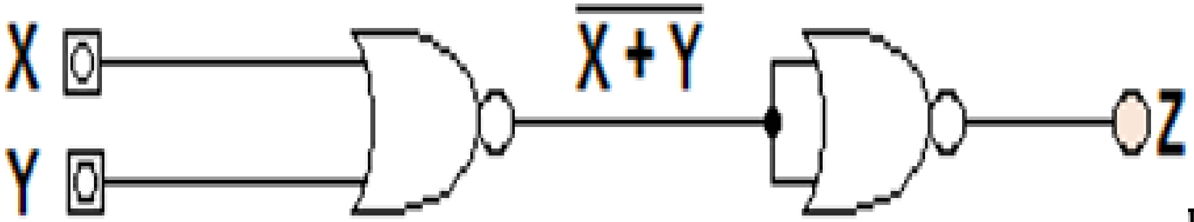
NOR as a universal gate:

i) **NOT gate:** NOT gate can be realized using a single input NOR gate.



Properties of NOT gate.

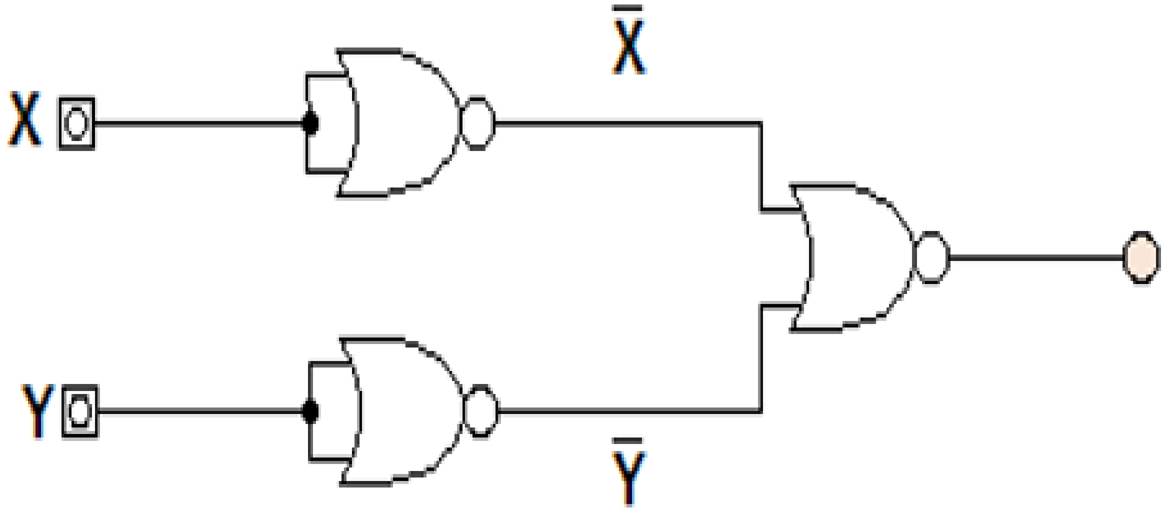
ii) **OR gate:** OR gate can be realized using two NOR gates.



$$= \overline{\overline{X+Y}} = \overline{\bar{X}\,.\,\bar{Y}} = \bar{\bar{X}} + \bar{\bar{Y}} = X + Y$$

Properties of OR gate.

iii) AND gate: AND gate can be realized using three NOR gates.

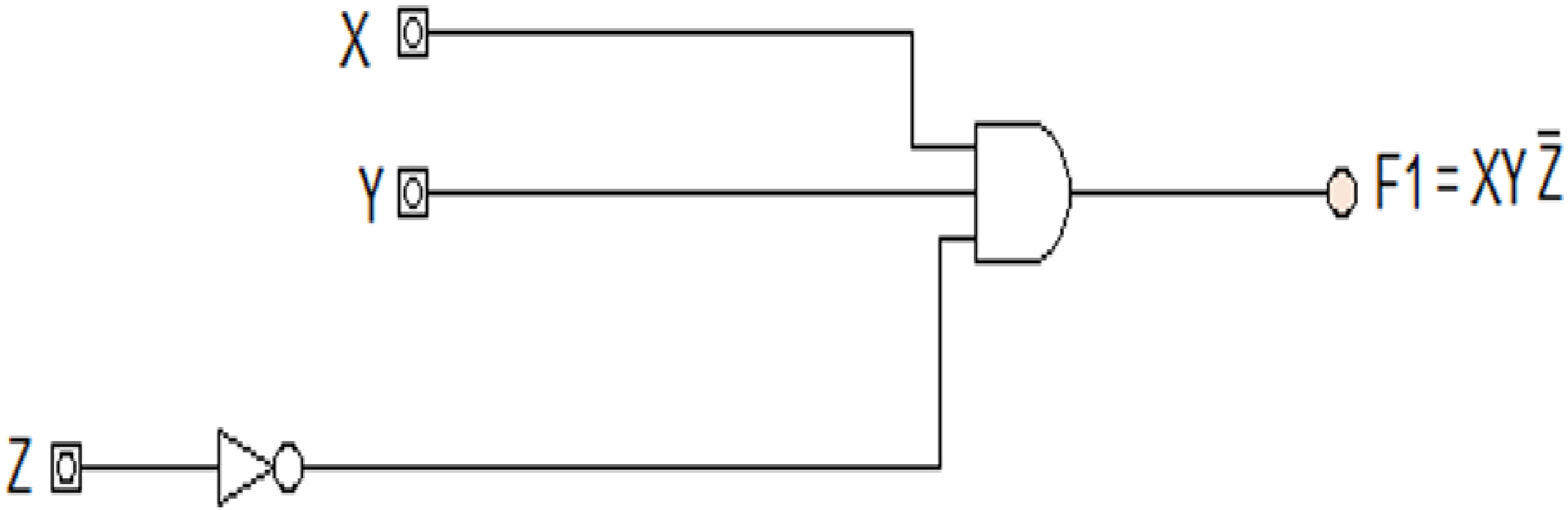


$$Z = \overline{\overline{X} + \overline{Y}} = \overline{\overline{X}} \, . \, \overline{\overline{Y}} = X \, . \, Y$$

Properties of AND gate.

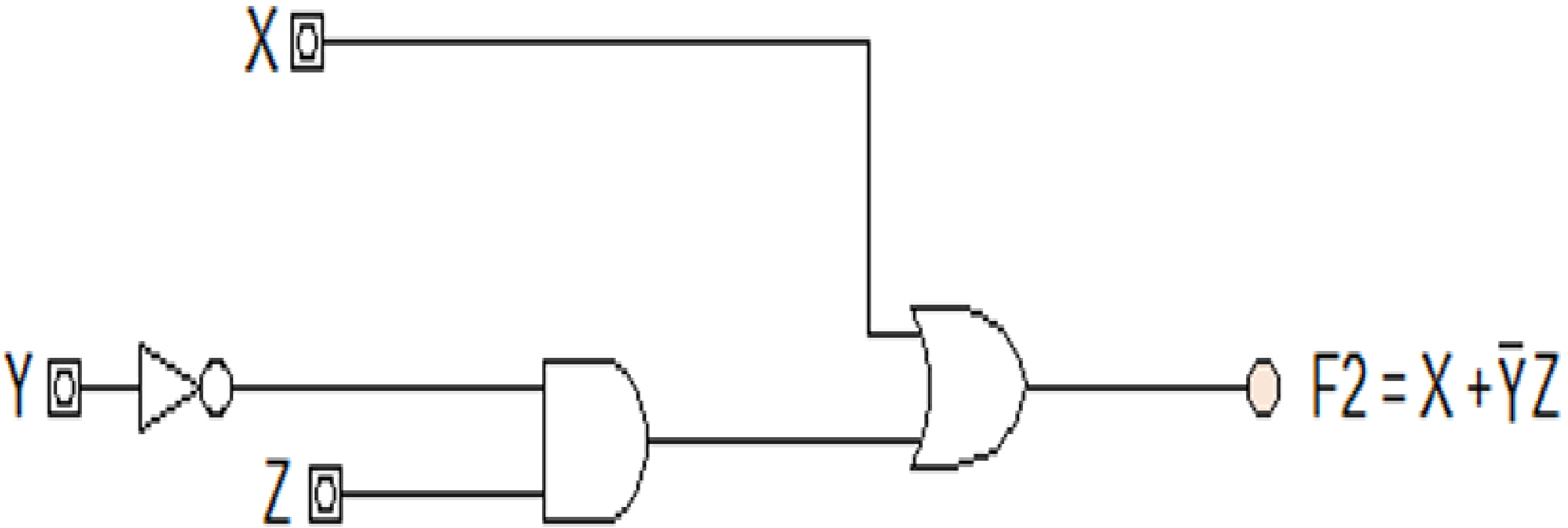
Implementation/realization of Boolean function with logic gates and truth table:

i) $F1 = XY\bar{Z}$



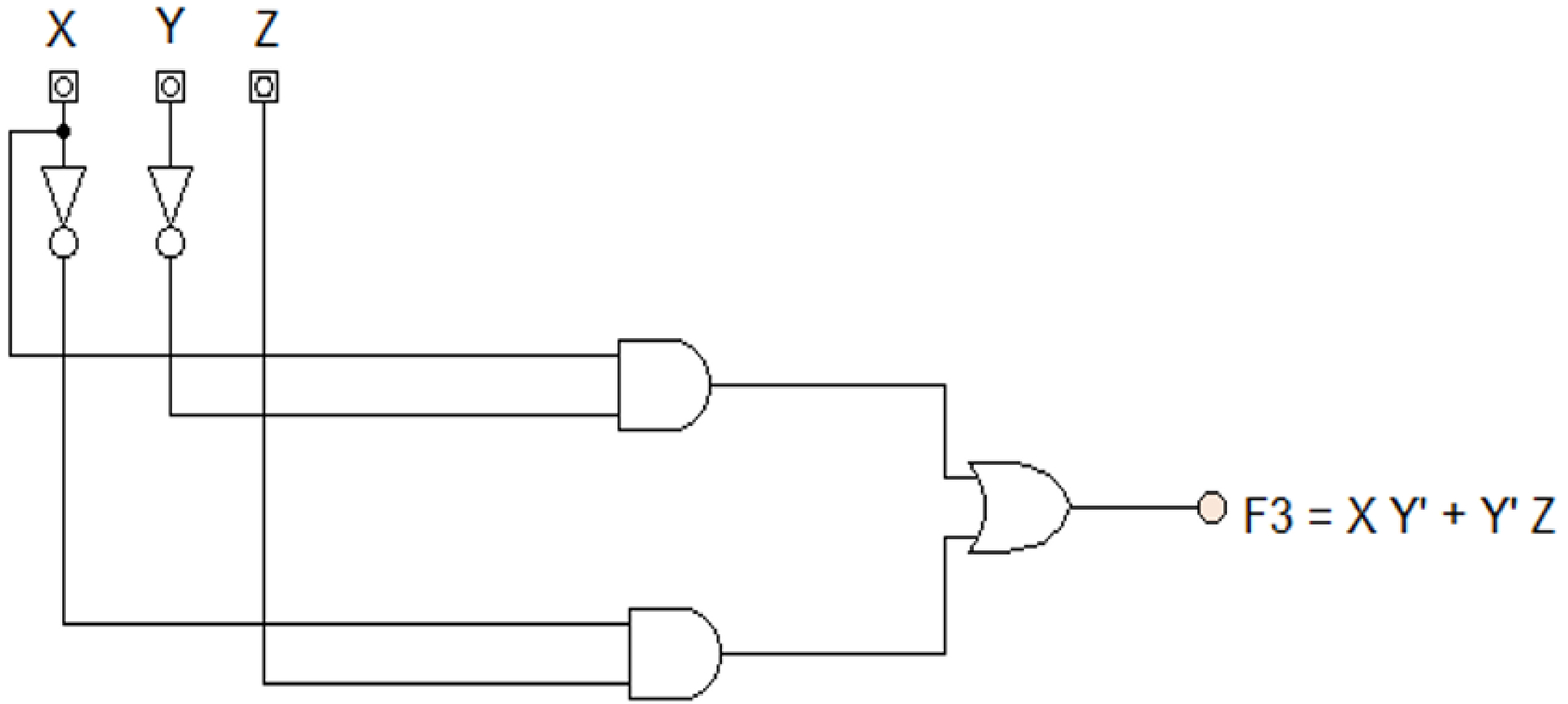
Implementation/realization of Boolean function with logic gates and truth table:

ii) $F2 = X + \bar{Y}Z$



Implementation/realization of Boolean function with logic gates and truth table:

iii) $F3 = X\bar{Y} + \bar{X}Z$



Implementation/realization of Boolean function with logic gates and truth table:

Truth table for F1, F2 and F3:

X	Y	Z	$F1 = XY\bar{Z}$	$F2 = X + \bar{Y}Z$	$F3 = X\bar{Y} + \bar{X}Z$
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	1	0
1	1	1	0	1	0