UNIT V: Logic Gates

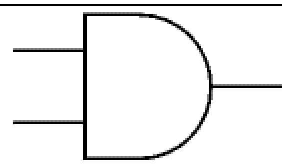
A logic gate is a building block of a digital circuit. Most logic gates have two inputs and one output and are based on Boolean algebra. At any given moment, every terminal is in one of the two binary conditions *false* (high) or *true* (low). False represents 0, and true represents 1. Depending on the type of logic gate being used and the combination of inputs, the binary output will differ. A logic gate can be thought of like a light switch, wherein one position the output is off—0, and in another, it is on—1. Logic gates are commonly used in integrated circuits (IC).

❖ Basic Logic gates

➢ AND gate

An AND gate can have two or more inputs, its output is true if all inputs are true. The output Q is true if input A AND input B are both true: **Q = A . B**

Input A	Input B	Output Q
0	0	0
0	1	0
1	0	0
1	1	1

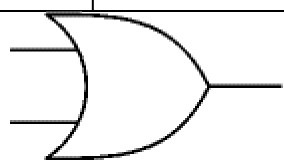


Logical symbol

- **OR gate**

An OR gate can have two or more inputs, its output is true if at least one input is true.
The output Q is true if input A OR input B is true (or both of them are true): **Q = A + B**

Input A	Input B	Output Q
0	0	0
0	1	1
1	0	1
1	1	1

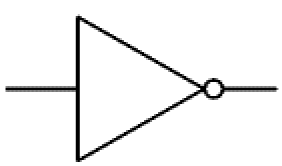


Logical symbol

- **NOT gate (inverter)**

A NOT gate can only have one input and the output is the inverse of the input. A NOT gate is also called an inverter.
The output Q is true when the input A is NOT true: **Q =** $\overline{A}$

Input A	Output Q
0	1
1	0



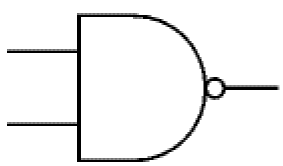
Logical symbol

❖ Universal gates

➢ NAND gate

This is an AND gate with the output inverted, as shown by the 'o' on the symbol output. A NAND gate can have two or more inputs, its output is true if NOT all inputs are true. The output Q is true if input A AND input B are NOT both true: **Q =** $\overline{A \cdot B}$

Input A	Input B	Output Q
0	0	1
0	1	1
1	0	1
1	1	0

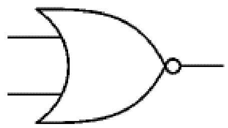


Logical symbol

➢ NOR gate

This is an OR gate with the output inverted, as shown by the 'o' on the symbol output. A NOR gate can have two or more inputs, its output is true if no inputs are true. The output Q is true if NOT inputs A OR B are true: **Q =** $\overline{A + B}$

Input A	Input B	Output Q
0	0	1
0	1	0
1	0	0
1	1	0



Logical symbol

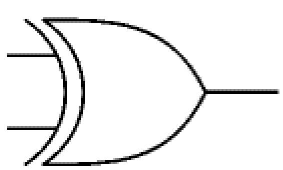
❖ <u>Exclusive gates</u>

➢ EX-OR gate

EXclusive-OR. This is like an OR gate but excluding both inputs being true. The output is true if inputs A and B are DIFFERENT. EX-OR gates can only have 2 inputs.

The output Q is true if either input A is true OR input B is true, *but not when both of them are true*: **Q = A** $\oplus$ **B**

Input A	Input B	Output Q
0	0	0
0	1	1
1	0	1
1	1	0

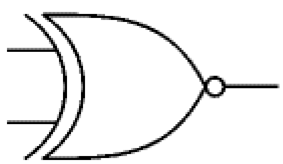


Logical symbol

➢ EX-NOR gate

EXclusive-NOR. This is an EX-OR gate with the output inverted, as shown by the 'o' on the symbol output. EX-NOR gates can only have 2 inputs. The output Q is true if inputs A and B are the *SAME* (both true or both false): **Q =** $\overline{A \oplus B}$

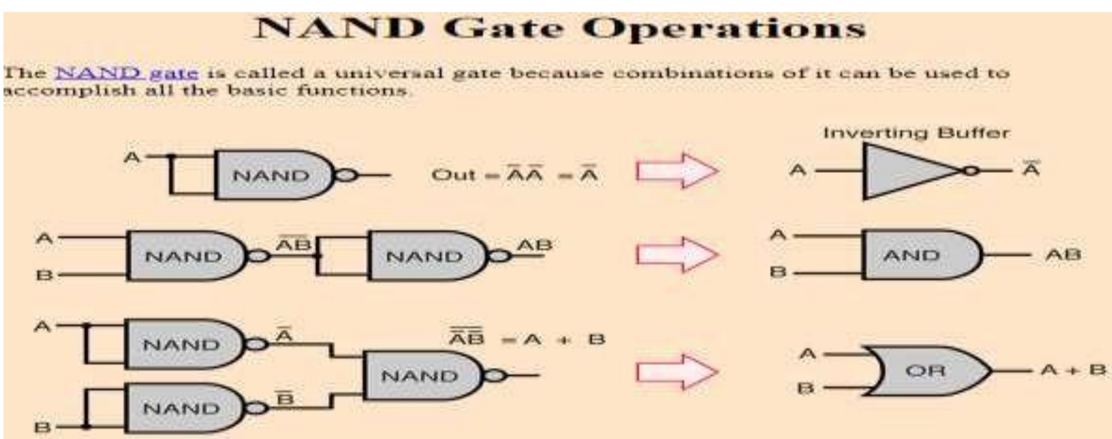
Input A	Input B	Output Q
0	0	1
0	1	0
1	0	0
1	1	1



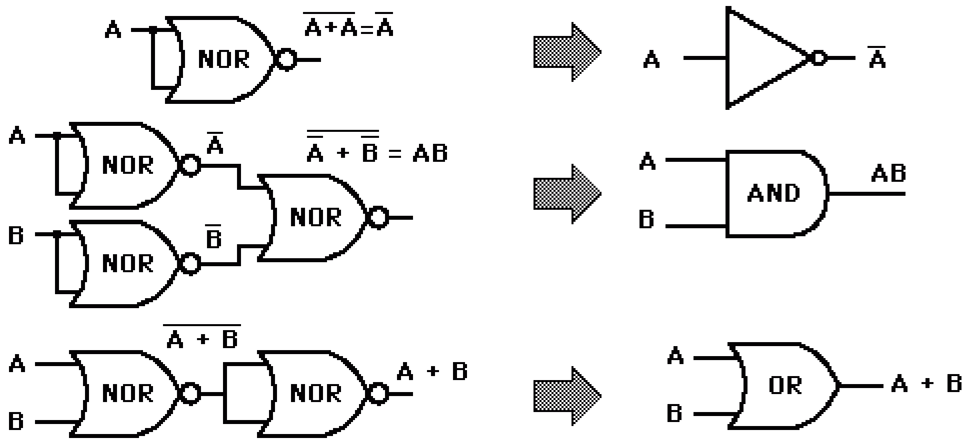
Logical symbol

✓ Why NAND and NOR gates are called Universal gates?

NOR gates and NAND gates have the particular property that any **one of them** can create any logical Boolean expression if appropriately designed. Meaning that you can create any logical Boolean expression using ONLY NOR gates or ONLY NAND gates. Other logical gates do not have this property.



NOR Gate Operations



Combinational Circuits

Combinational Circuits (CC) are circuits made up of different types of logic gates. A **logic gate** is a basic building block of any electronic circuit. The output of the combinational circuit depends on the values at the input at any given time. The circuits do not make use of any memory or storage device.

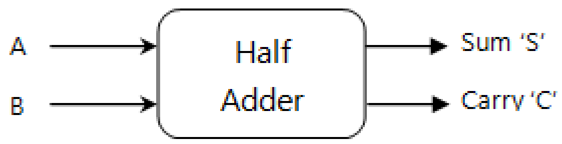
❖ <u>The Adder</u>

An adder is a digital circuit that is used to perform the addition of numeric values. It is one of the most basic circuits and is found in arithmetic logic units of computing devices. There are two types of adders. **Half adders** compute single digit numbers, while **full adders** compute larger numbers.

➢ Half Adder

The half adder adds two single digit binary numbers and forms the foundation for all addition operations in computing. If we have two single binary digits, A and B, then the half adder adds them with the circuit carrying two outputs, the sum and the carry. The carry represents any overflow from the addition of the two numbers. This is represented in the following block diagram figure:

Figure 1: Half Adder



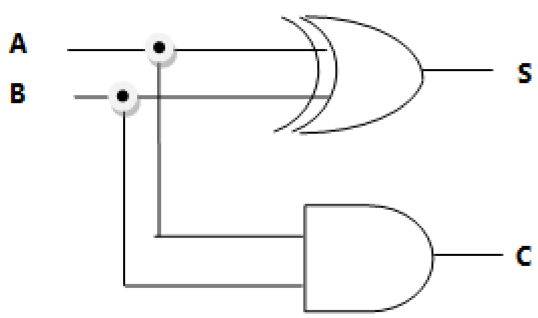
In addition, the following truth table demonstrates all the possible outputs for various input combinations of the half adder.

Table 1: Truth Table - Half Adder

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

The next figure represents the logic circuit of the half adder:

Figure 2: Half Adder- Logic Circuit

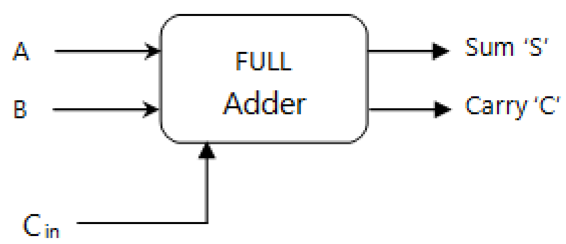


The sum S is represented by the Boolean Expression S = A'B + AB' = $\mathbf{A} \oplus \mathbf{B}$ and C = AB

➢ Full Adder

The full adder overcomes the disadvantages of the half adder in that it can add two single bit numbers in addition to the carry digit at its input as seen in this figure:

Figure : Full Adder



The next truth table shown here demonstrates all the possible outputs for various input combinations with the carry input digit:

Table 2: Truth Table - Full Adder

A	B	Cin	Co	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	1
1	1	1	1	1

Boolean expression for the full adder is S = A'B'Cin + A'BCin' + AB'Cin' + ABCin and C = A'BCin + AB'Cin + ABCin' + ABCin. This is where A and B are all the possible binary inputs and C is the carry in.