Logic gates

Lecture 3

Prepared by : Eng.Amani Safwat

Logic Gates

Logic gates are electronic circuits used to perform logical operations on binary signals (0 and 1).

They are the basic building blocks of digital circuits such as computers, processors, and embedded systems.

Why We Use Logic Gates

We use logic gates to control the flow of signals within digital systems and to perform logical and arithmetic operations such as addition, comparison, and data storage.
They form the foundation of all digital systems

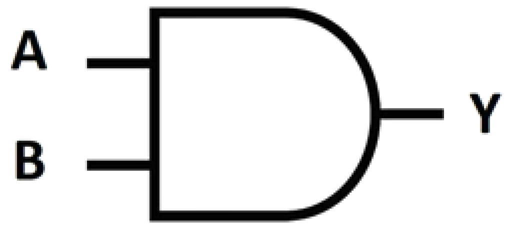
Types of Logic Gates

- **(Basic Gates):**
- AND Gate
- OR Gate
- NOT Gate
- **(Universal Gates):**
- NAND Gate
- NOR Gate
- **(Special Gates):**
- XOR Gate (Exclusive OR)
- XNOR Gate (Exclusive NOR)

AND Gate

Outputs 1 only if all inputs are 1
Used when all conditions must be true

Symbol



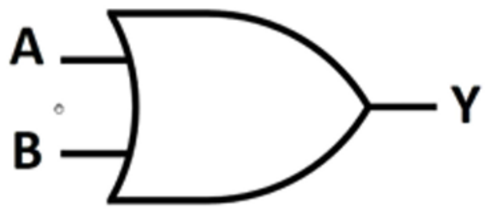
Truth Table		
A	**B**	**Y**
0	0	0
1	0	0
0	1	0
1	1	1

Equation :**Y= AB OR Y=A.B**

OR Gate

Outputs 1 if at least one input is 1
Used when only one condition needs to be true.

Symbol



Truth Table		
A	**B**	**Y**
0	0	0
1	0	1
0	1	1
1	1	1

Equation :**Y=A+B**

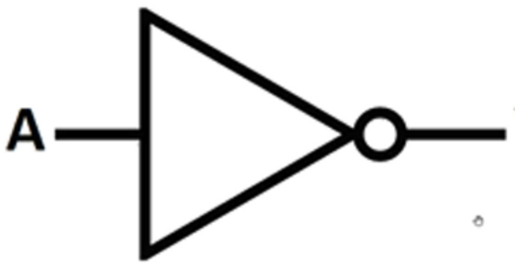
NOT Gate

Inverts the input; if input is 1, output is 0

Used to reverse logical states

Truth Table	
A	**Y**
0	1
1	0

Symbol



Equation **:Y=A**

NAND Gate

The output is 0 only if all inputs are 1 (inverse of AND).

Used as a universal gate to build any logic circuit.

Truth Table		
A	**B**	**Y**
0	0	1
1	0	1
0	1	1
1	1	0

Symbol

Equation :$Y=\overline{AB}$

NOR Gate

Outputs 1 only when all inputs are 0 (inverse of OR).

Used as a universal gate to build any circuit

Truth Table		
A	**B**	**Y**
0	0	1
1	0	0
0	1	0
1	1	0

Symbol

Equation :$Y=\overline{A+B}$

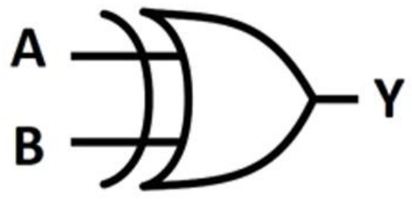
XOR Gate

Outputs 1 only when inputs are different

Used in adder and comparator circuits

Truth Table		
A	**B**	**Y**
0	0	0
1	0	1
0	1	1
1	1	0

Symbol



Equation :$Y = A \oplus B$

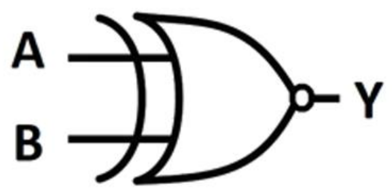
XNOR Gate

Outputs 1 only when inputs are the same

Used in equality comparator circuits

Truth Table		
A	**B**	**Y**
0	0	1
1	0	0
0	1	0
1	1	1

Symbol

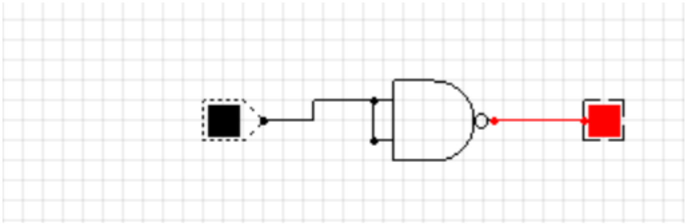


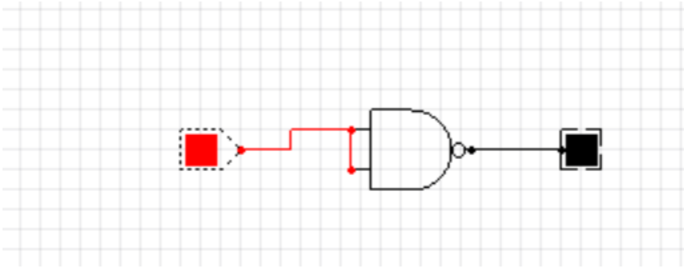
Equation :$Y=\overline{A \oplus B}$

Universal Gates

NAND Gate

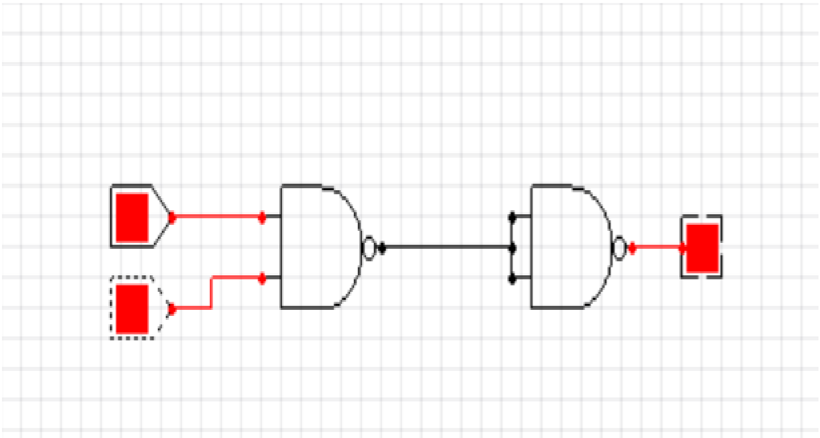
NOT using NAND:
Y = (A · A)′ → gives the same output as NOT.





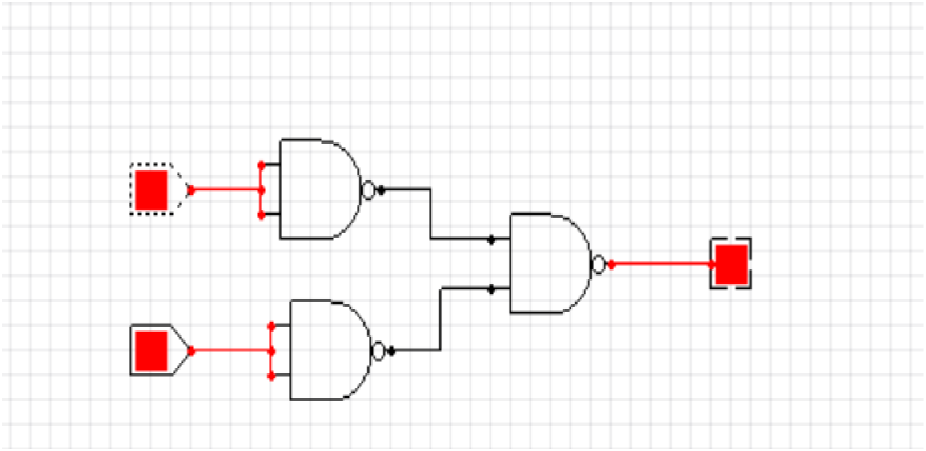
NAND Gate

AND using NAND:
Y = ((A · B)′)′ → We use two NAND gates.



NAND Gate

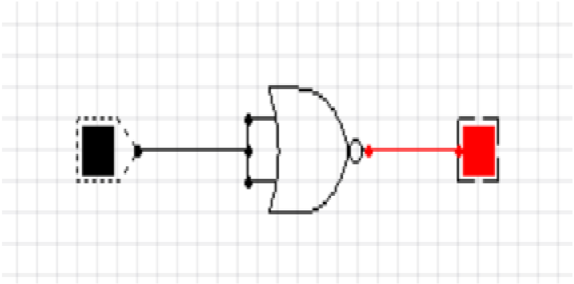
OR using NAND:
Y = ((A′) · (B′))′ → We use three NAND gates.

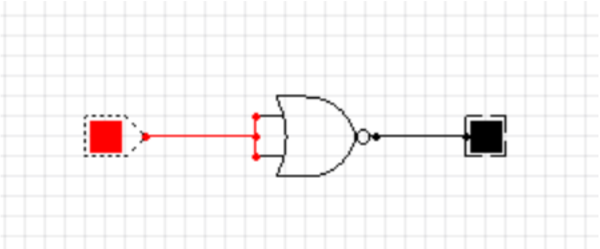


NOR Gate

NOT using NOR:

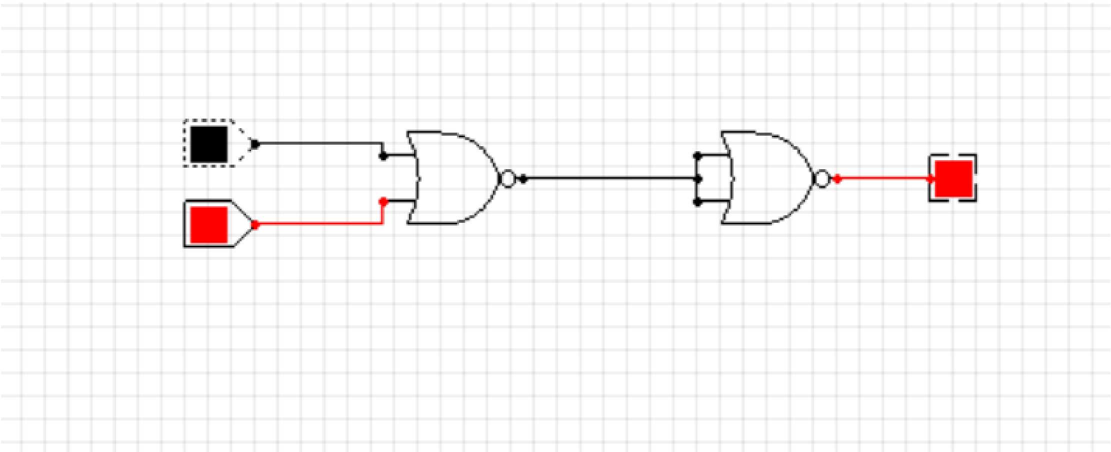
Y = (A + A)′ → Same result as NOT





NOR Gate

OR using NOR:
Y = ((A + B)′)′ → We use two NOR gates



NOR Gate

AND using NOR:
Y = ((A’) + (B’))’ → We use three NOR gates