Standard Logic Gates

Six types of gates

- AND
- OR
- NOT
- XOR
- NAND
- NOR

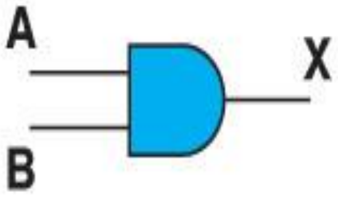
AND Gate

An AND gate accepts two input signals, If both are 1, the output is 1; otherwise, the output is 0.

Boolean Expression

$$X = A \cdot B$$

Logic Diagram Symbol



Truth Table

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

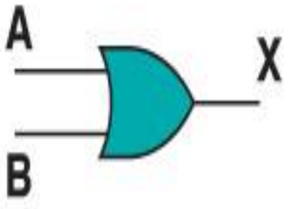
OR Gate

An OR gate accepts two input signals, If both are 0, the output is 0; otherwise, the output is 1.

Boolean Expression

$$X = A + B$$

Logic Diagram Symbol



Truth Table

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

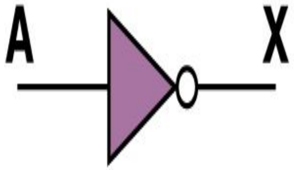
NOT Gate

A NOT gate accepts one input signal (0 or 1) and returns the opposite signal as output.

Boolean Expression

$$X = A'$$

Logic Diagram Symbol



Truth Table

A	X
0	1
1	0

NAND Gate

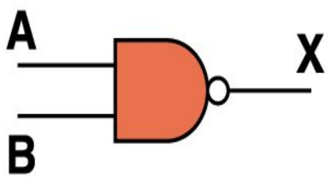
The NAND gate accepts two input signals

If both are 1, the output is 0; otherwise, the output is 1

Boolean Expression

$$X = (A \cdot B)'$$

Logic Diagram Symbol



Truth Table

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

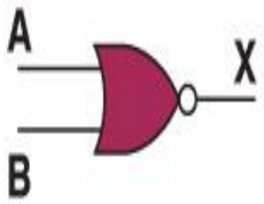
NOR Gate

The OR gate accepts two input signals, If both are 0, the output is 1; otherwise, the output is 0

Boolean Expression

$X = (A + B)'$

Logic Diagram Symbol



Truth Table

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

XOR Gate

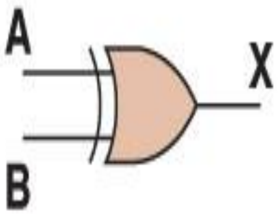
An XOR gate accepts two input signals, If both inputs are the same, the output is 0; otherwise, the output is 1.

X=A’B+AB’

Boolean Expression

$X = A \oplus B$

Logic Diagram Symbol

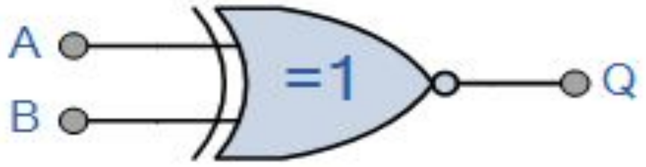


Truth Table

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

X-NOR

Symbol	Truth Table



2-input Ex-NOR Gate

B	A	Q
0	0	1
0	1	0
1	0	0
1	1	1

Boolean Expression $Q = \overline{A \oplus B}$

Read if A **AND** B the **SAME** gives Q

XOR Gate

Note the difference between the XOR gate and the OR gate; they differ only in one input situation

When both input signals are 1, the OR gate produces a 1 and the XOR produces a 0

XOR is called the *exclusive OR*

Review of Gate Processing

- An AND gate produces 1 if both input values are 1
- An OR gate produces 0 if both input values are 0
- A NOT gate inverts its single input.
- A NAND gate produces 0 if both inputs are 1.
- A NOR gate produces a 1 if both inputs are 0.
- An XOR gate produces 0 if input values are the same.
- An XNOR gate produces 0 if both input are differ/opposite .

Boolean Law’s

Identity Name	AND Form	OR Form
Identity Law	$1 \times X = X$	$0 + X = X$
Idempotent law	$X \times X = X$	$X + X = X$
Inverse Law	$X \times X' = 0$	$X + X' = 1$
Null Law	$0 \times X = 0$	$1 + X = 1$

Boolean Law’s Coti….

Identity Name	AND Form	OR Form
Commutative Law	$X \times Y = Y \times X$	$X + Y = Y + X$
Associative Law	$(X \times Y) \times Z = X \times (Y \times Z)$	$(X + Y) + Z = X + (Y + Z)$
Distributive Law	$X + (Y \times Z) = (X + Y)(X + Z)$	$X \times (Y + Z) = X \times Y + X \times Z$
Absorption Law	$X \times (X + Y) = X$	$X + X \times Y = X$
DeMorgan’s Law	$\overline{(X \times Y)} = X' + Y'$	$\overline{(X + Y)} = X' \times Y'$
Double Complement Law	$((X)')' = X$	

Universal Logic Gate

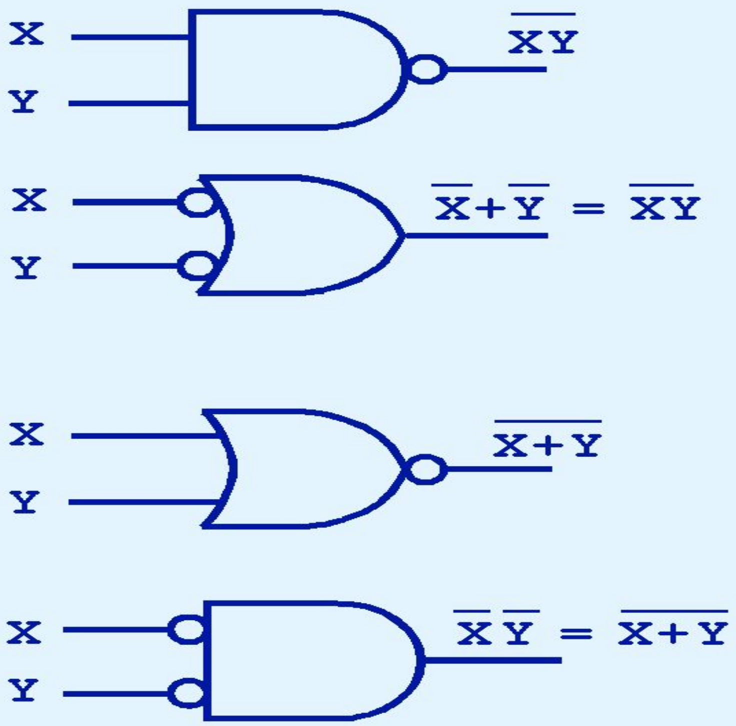
- A universal gate is a gate which can implement any Boolean function without need to use any other gate type.
- The NAND and NOR gates are universal gates.

X NAND Y

X	Y	X NAND Y
0	0	1
0	1	1
1	0	1
1	1	0

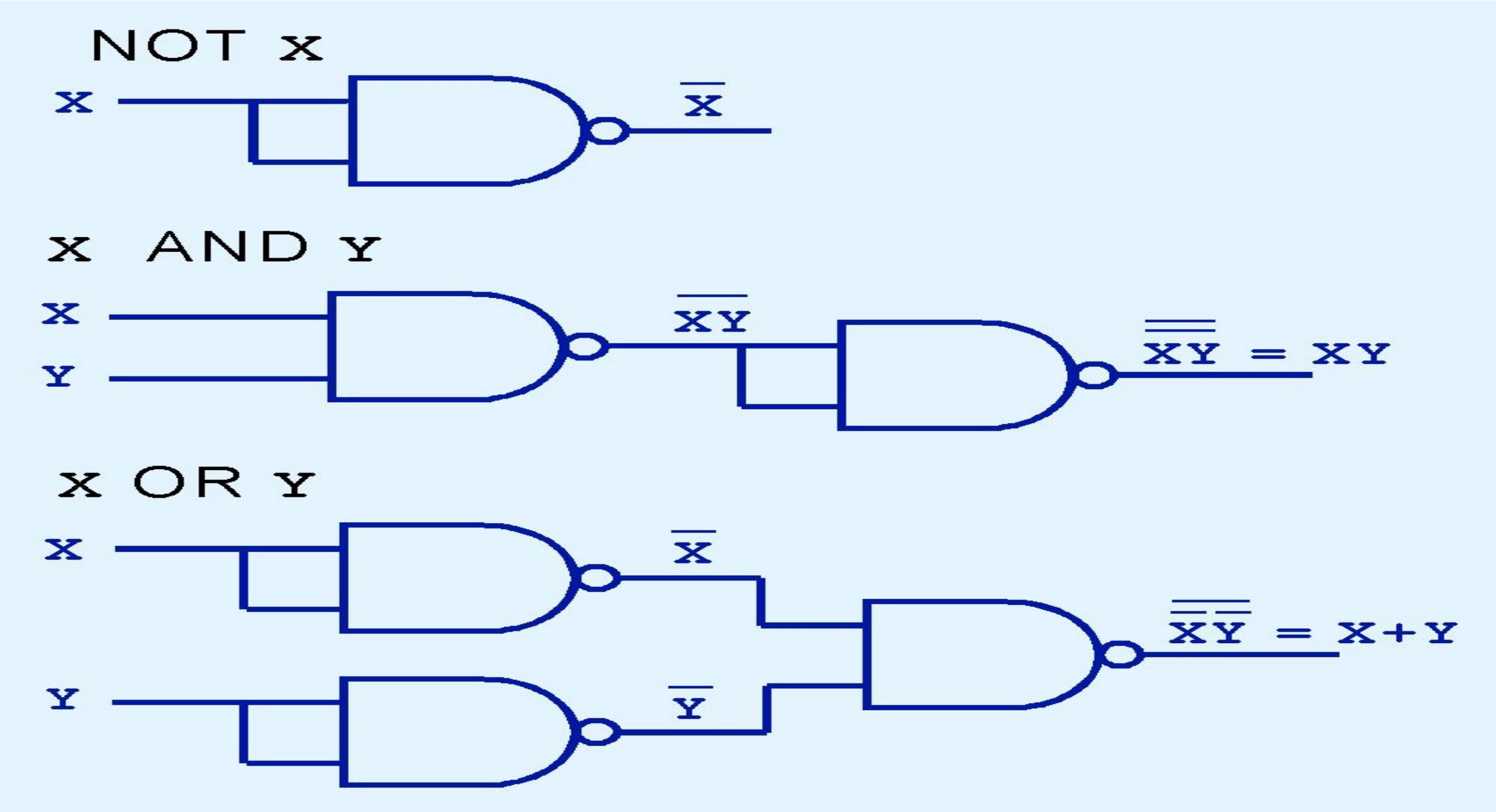
X NOR Y

X	Y	X NOR Y
0	0	1
0	1	0
1	0	0
1	1	0



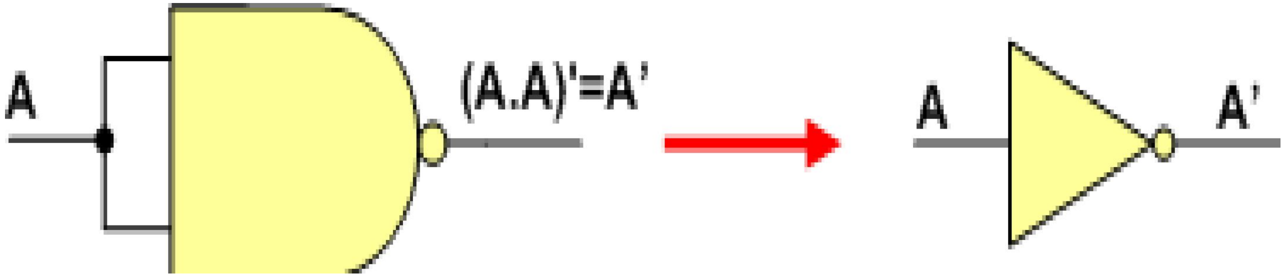
NAND Gate as universal Gate

- To prove that any Boolean function can be implemented using only NAND gates, we will show that the AND, OR, and NOT operations can be performed using only NAND gates.

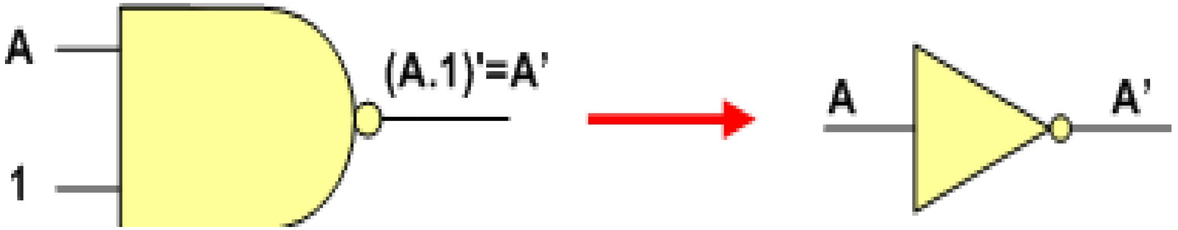


NAND Gate as universal Gate

- **Implementing an Inverter (NOT Gate) Using only NAND Gate**
- All NAND input pins connect to the input signal **A** gives an output **A**’.

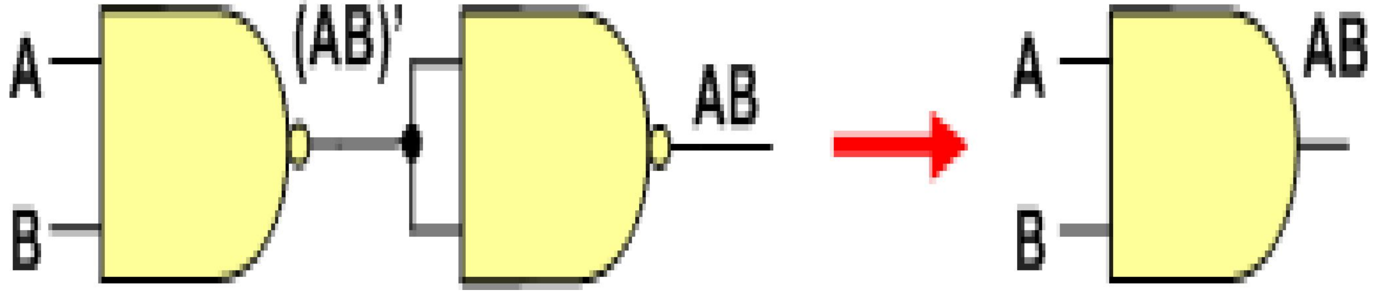


- One NAND input pin is connected to the input signal **A** while all other input pins are connected to logic **1**. The output will be **A’**.



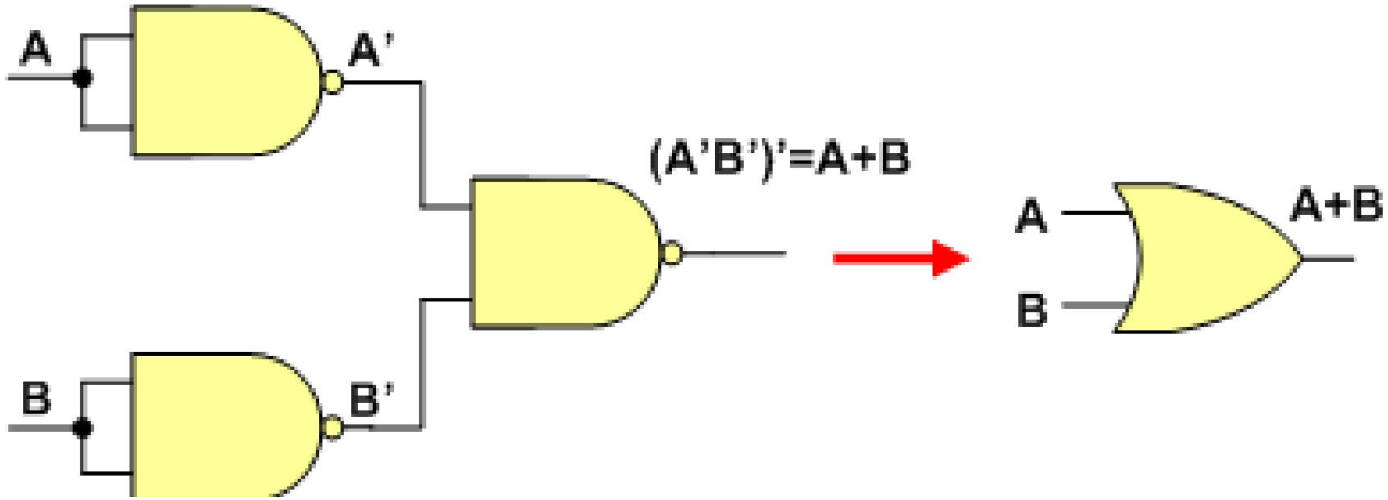
Implementing AND Using only NAND Gates

- **An AND gate** can be replaced by NAND gates as shown in the figure (The AND is replaced by a NAND gate with its output complemented by a NAND gate inverter).



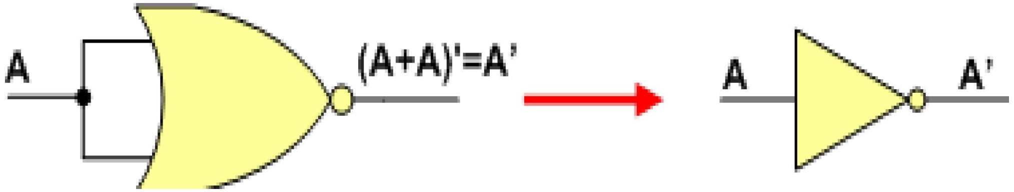
Implementing OR Using only NAND Gates

- **An OR gate** can be replaced by NAND gates as shown in the figure (The OR gate is replaced by a NAND gate with all its inputs complemented by NAND gate inverters).

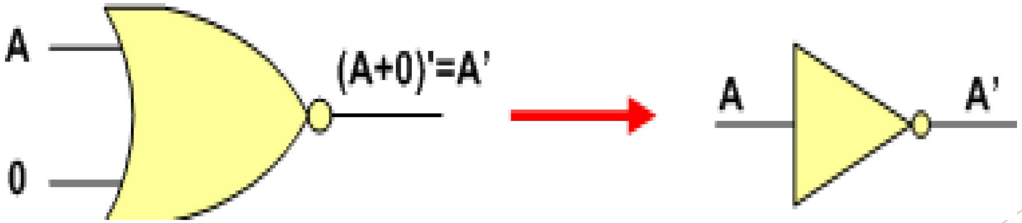


NOR Gate is as Universal Gate

- To prove that any Boolean function can be implemented using only NOR gates, we will show that the AND, OR, and NOT operations can be performed using only these gates.
- **Implementing an Inverter Using only NOR Gate**
- The figure shows two ways in which a NOR gate can be used as **an inverter (NOT gate)**.
- All NOR input pins connect to the input signal **A** gives an output **A'**.

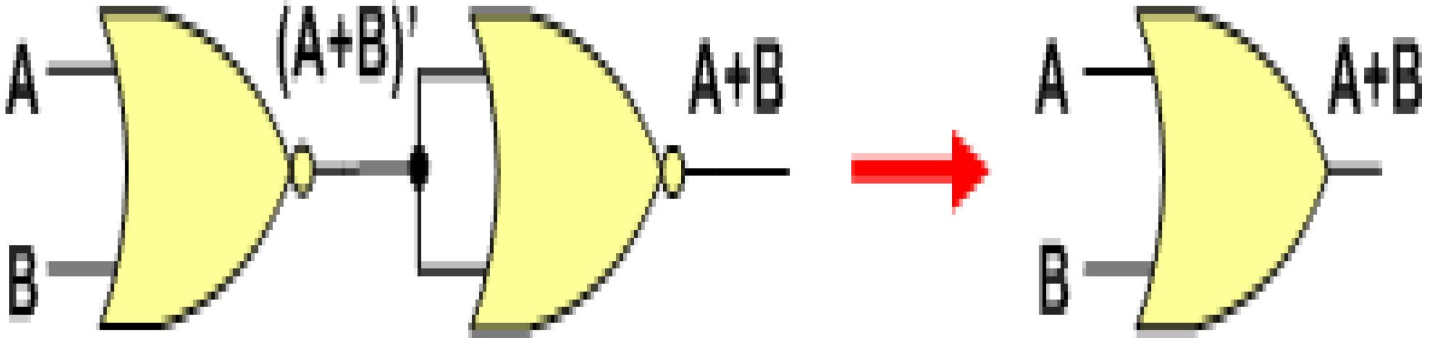


- One NOR input pin is connected to the input signal **A** while all other input pins are connected to logic **0**. The output will be **A'**.



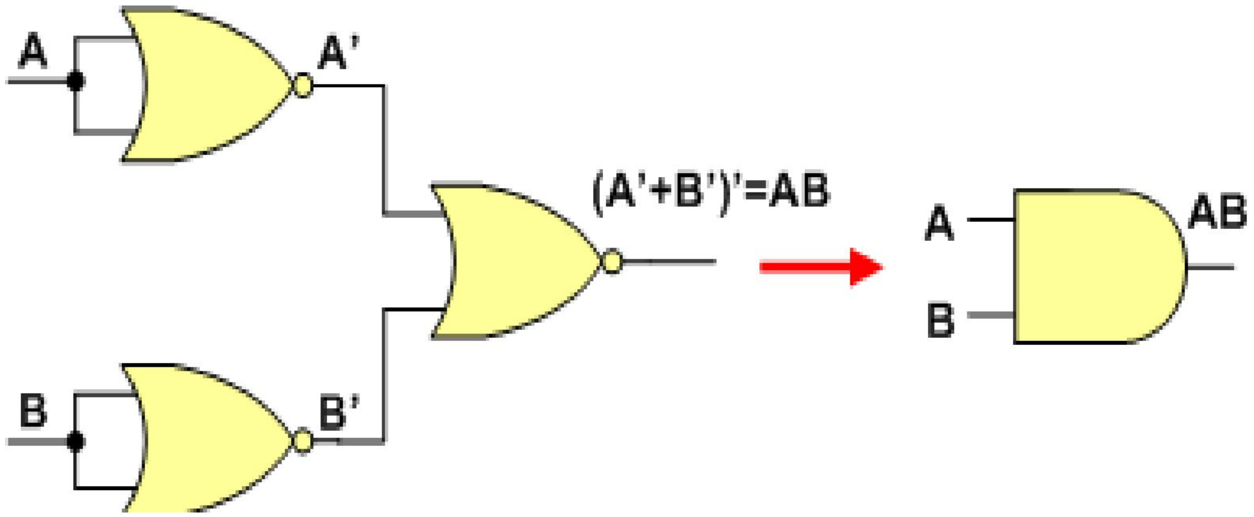
Implementing OR Using only NOR Gates

- **An OR gate** can be replaced by NOR gates as shown in the figure (The OR is replaced by a NOR gate with its output complemented by a NOR gate inverter)



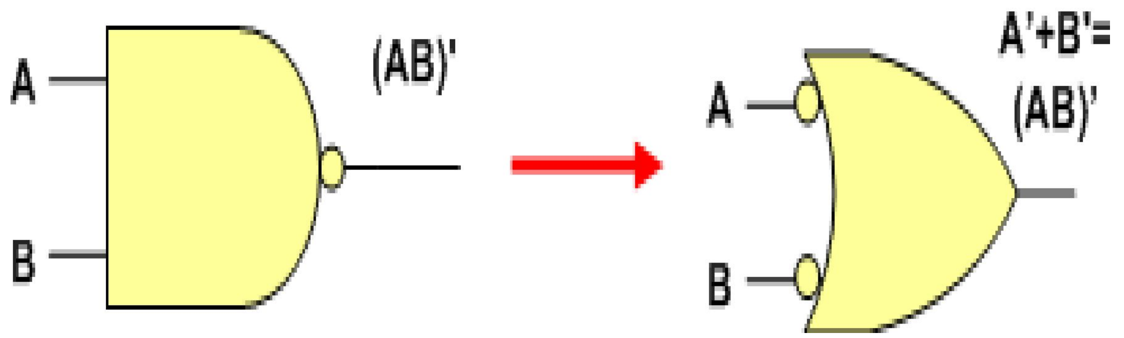
Implementing AND Using only NOR Gates

- **An AND gate** can be replaced by NOR gates as shown in the figure (The AND gate is replaced by a NOR gate with all its inputs complemented by NOR gate inverters)

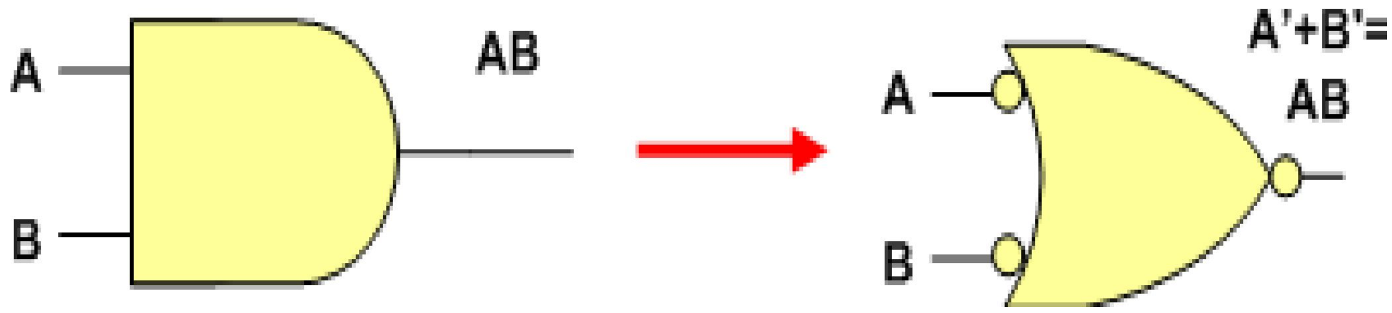


Equivalent Gates:

- A NAND gate is equivalent to an inverted-input OR gate.



- An AND gate is equivalent to an inverted-input NOR gate.



Equivalent Gates:

- A NOR gate is equivalent to an inverted-input AND gate.

- An OR gate is equivalent to an inverted-input NAND gate.