SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

19ECB202 – LINEAR AND DIGITAL CIRCUITS

II YEAR/ III SEMESTER

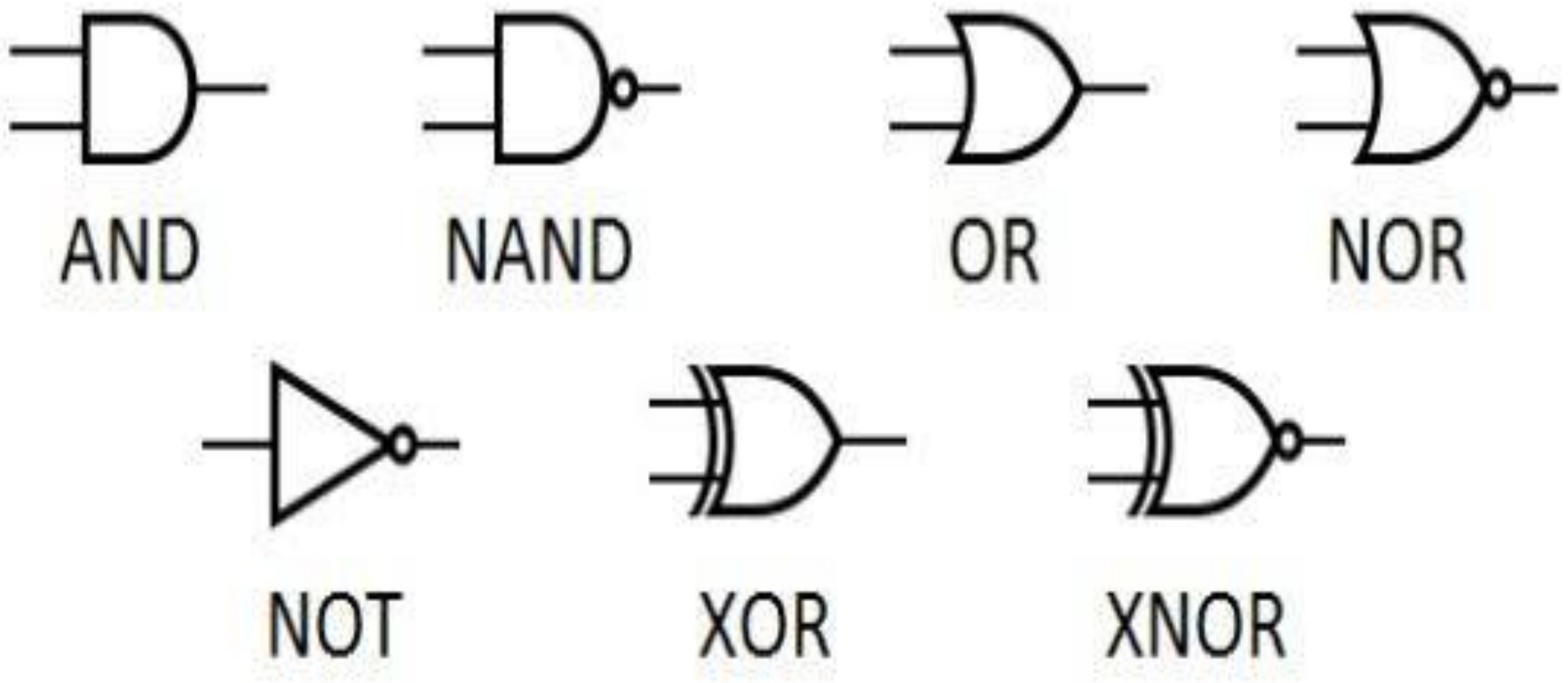
UNIT 3 – GATES AND MINIMIZATION TECHNIQUES

TOPIC 1 - Logic Gates, NAND and NOR Implementation

What is a Logic Gate?

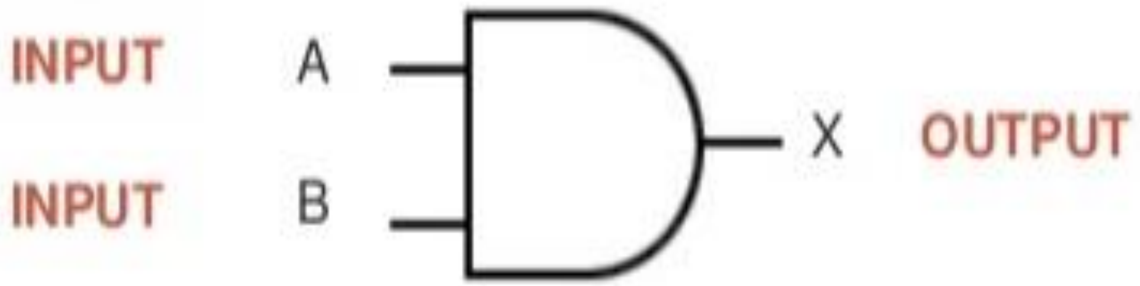
- A logic gate is an electronic circuit which makes logical decisions based on the combination of digital signals present on its inputs.

- The most common basic logic gates are



AND Gate

- The AND gate is an electronic circuit that gives a **high** output (1) only if **all** its inputs are high. A dot (.) is used to show the AND operation i.e. A.B



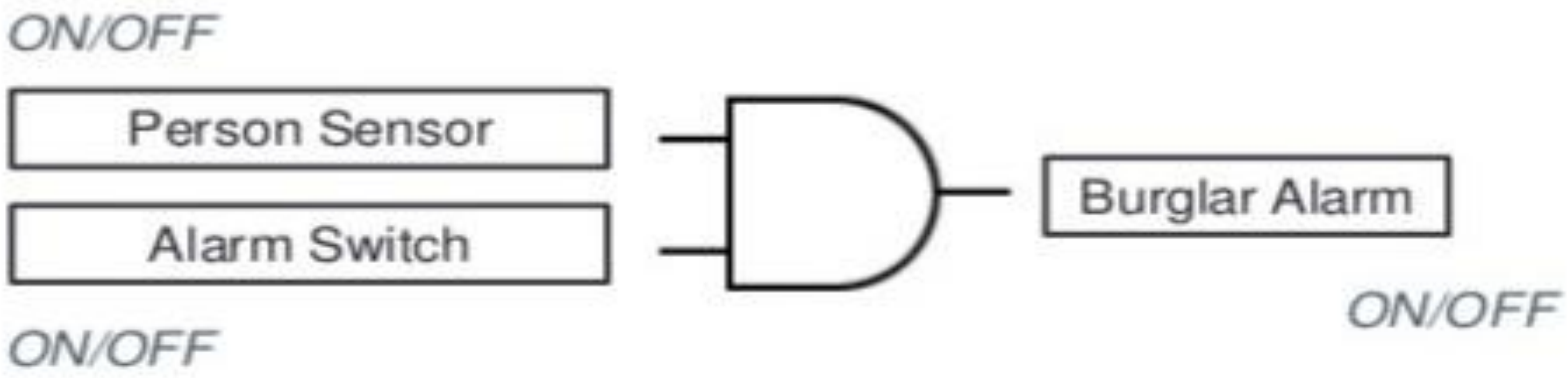
The output will be positive (true) when both inputs (the input one AND the input two) are positive (true).

X = A AND B

2 Input AND gate

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

AND Gate

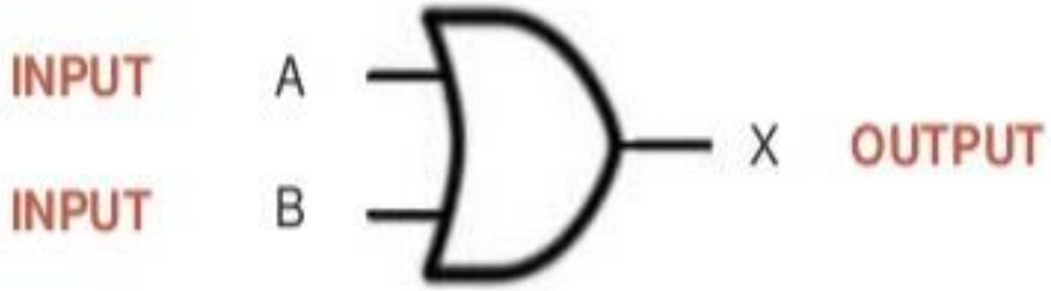


If both the Person Sensor AND the Alarm Switch are on then the Burglar Alarm is activated.

OR Gate

- The OR gate is an electronic circuit that gives a high output (1) if **one or more** of its inputs are high. A plus (+) is used to show the OR operation.

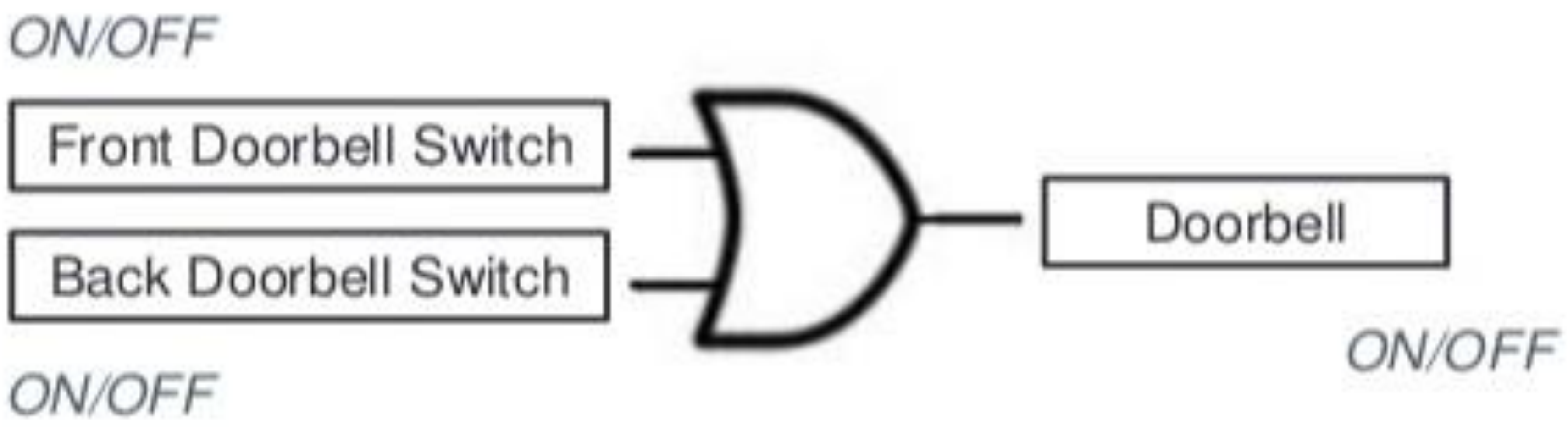
2 Input OR gate		
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1



The output will be positive (True) if at least one input is true.

X = A OR B

OR Gate

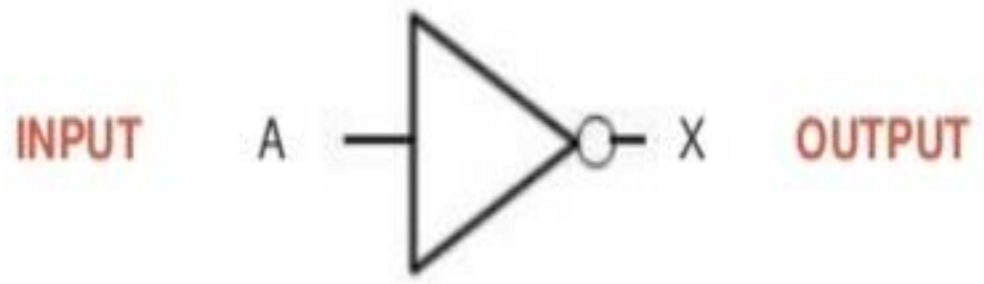


If either the Front Doorbell Switch OR the Back Doorbell Switch is pressed then the Doorbell rings.

NOT Gate

- ➢The NOT gate is an electronic circuit that produces an inverted version of the input at its output. It is also known as an *inverter*.
- ➢ If the input variable is A, the inverted output is known as NOT A.
- ➢ This is also shown as A', or A with a bar over the top, as shown at the outputs.

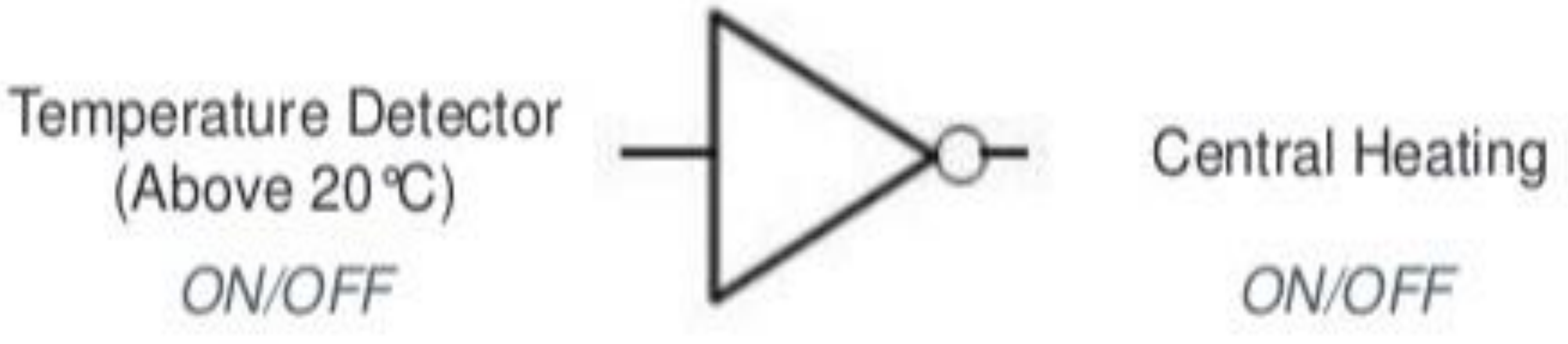
NOT gate	
A	$\bar{A}$
0	1
1	0



A NOT gate (inverter) has only one input. It reverses the logic state.

Y = NOT A

NOT Gate



If the temperature is above 20°C then the Central Heating is switched off.

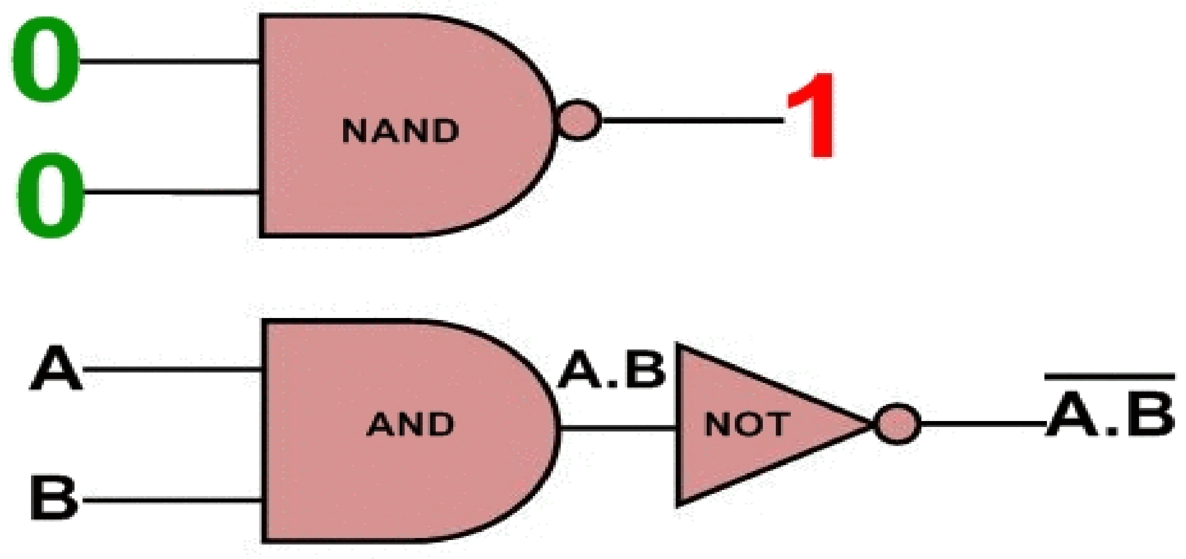
If the temperature is below 20°C then the Central Heating is switched on.

NAND Gate

- This is a NOT-AND gate which is equal to an AND gate followed by a NOT gate.
- The outputs of all NAND gates are high if any of the inputs are low.
- The symbol is an AND gate with a small circle on the output.
- The small circle represents inversion

2 Input NAND gate

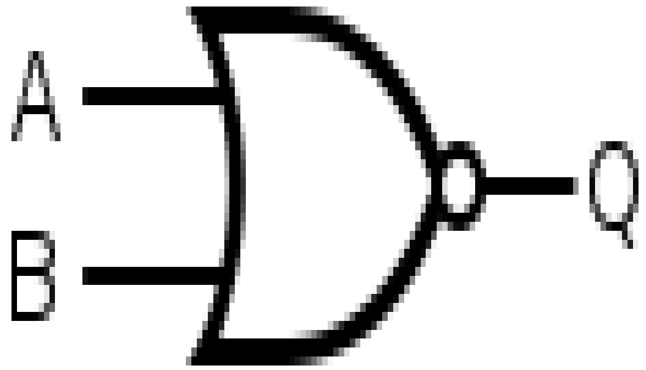
A	B	$\overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0



NAND gate

NOR Gate

- This is a NOT-OR gate which is equal to an OR gate followed by a NOT gate.
- The outputs of all NOR gates are low if **any** of the inputs are high.
- The symbol is an OR gate with a small circle on the output.
- The small circle represents inversion.



2 Input NOR gate		
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

EXOR Gate

- The '**Exclusive-OR**' gate is a circuit which will give a high output if **either, but not both**, of its two inputs are high.
- An encircled plus sign ( ) is used to show the EXOR operation.

(a) Circuit symbol

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

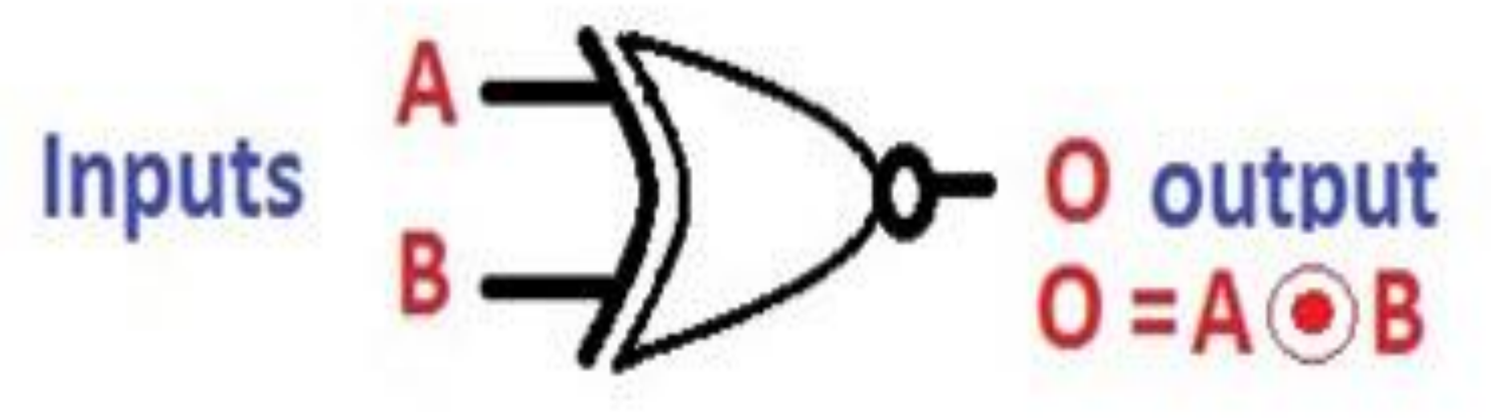
(b) Truth table

$$C = A \oplus B$$

(c) Boolean expression

EXNOR Gate

- The '**Exclusive-NOR'** gate circuit does the opposite to the EOR gate.
- It will give a low output if **either, but not both**, of its two inputs are high.
- The symbol is an EXOR gate with a small circle on the output.
- The small circle represents inversion



Symbol

Inputs		Output
A	B	O
0	0	1
0	1	0
1	0	0
1	1	1

Truth table

Universal Gates

- A gate that can be used to create any logic gate is called universal gate.

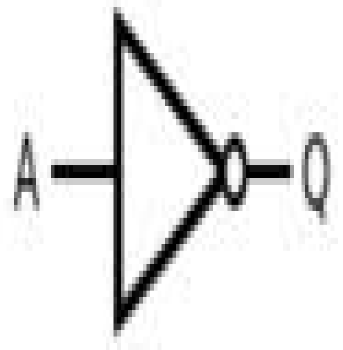
- NAND and NOR gates are called **universal gates** because they can perform all the three basic logic functions OR, AND and NOT.

NAND and NOR implementation

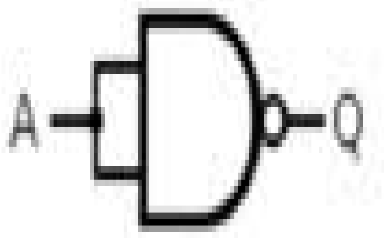
- Any Boolean function can be created using AND OR and NOT gates.
- AND, OR and NOT gates can be implemented using NAND and NOR gates.

NAND implementation - Implementation of NOT and AND using NAND gate

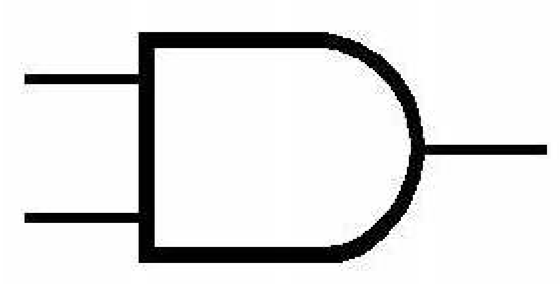
- A NAND gate with single input acts like a NOT gate.
- As a NAND gate is the invert of AND so by putting an inverter on the output of NAND we can have AND gate.



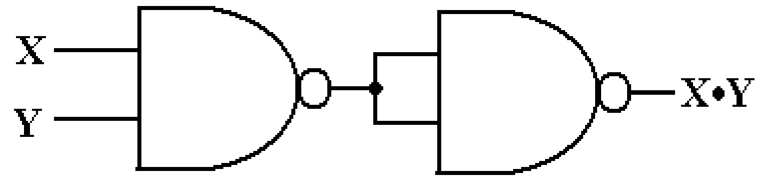
NOT gate



NAND construction of NOT gate



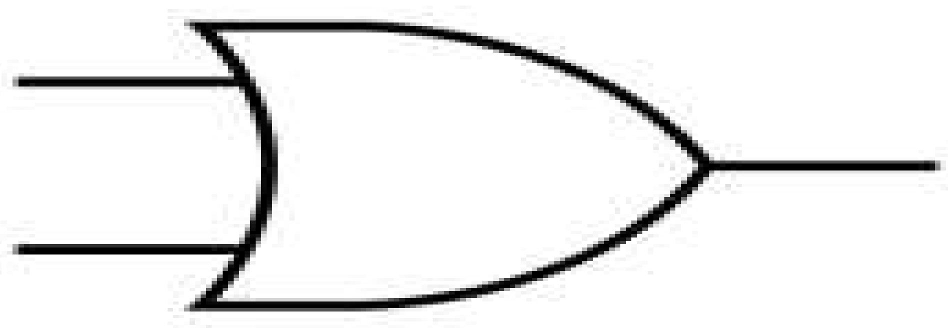
AND Gate



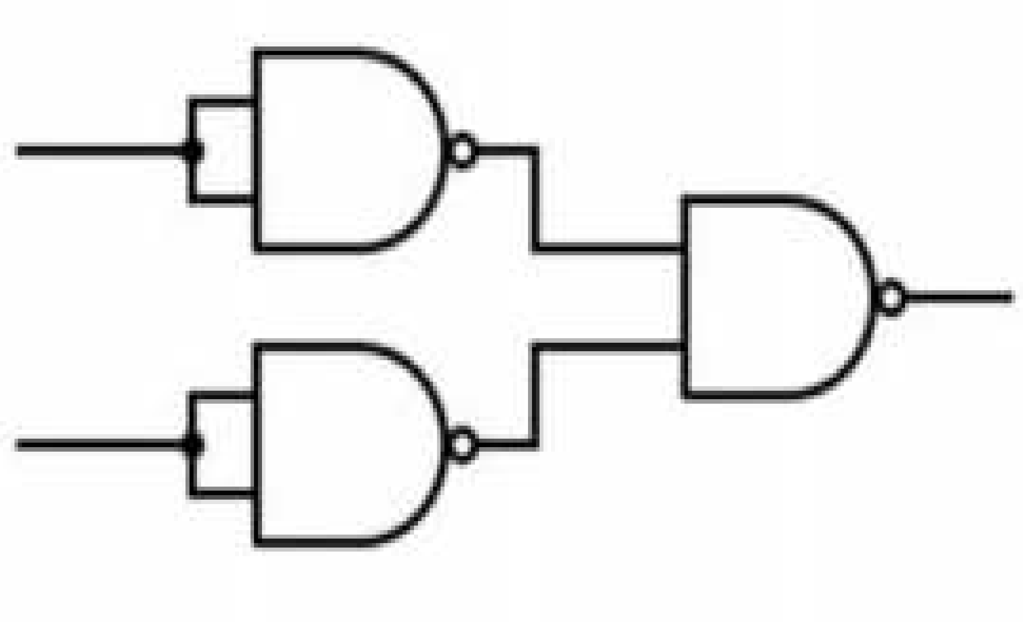
NAND Construction of AND Gate

NAND implementation - Implementation of N OR using NAND gate

- By putting additional inverters in the input we can achieve an OR gate by a NAND gate De Morgan's Law is the base of it.

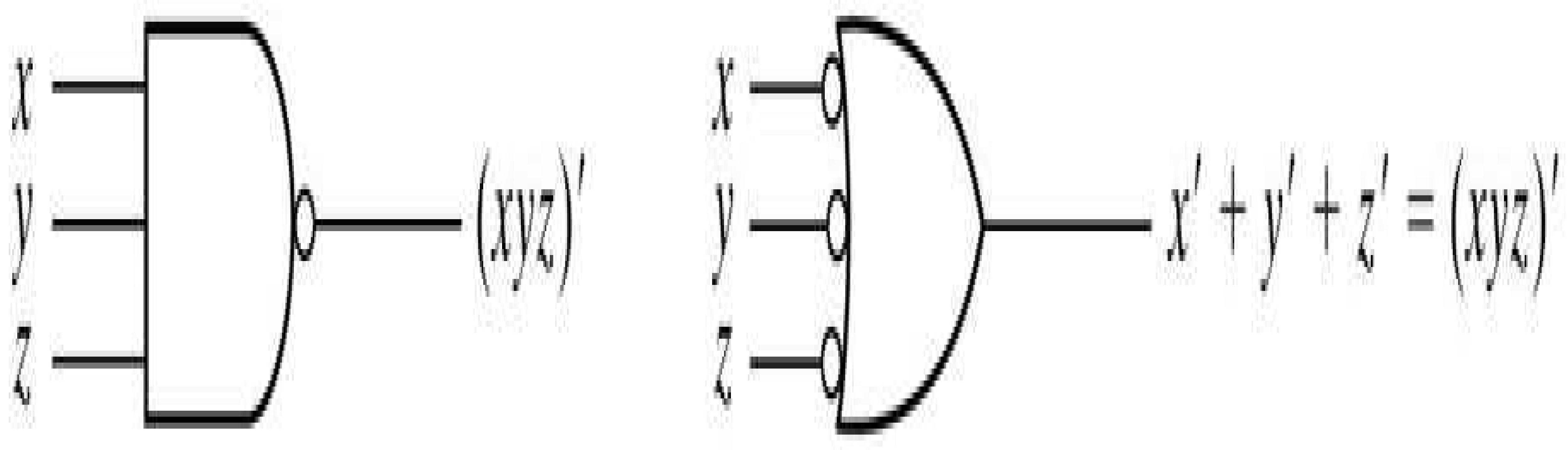


OR Gate



NAND Implementation of OR Gate

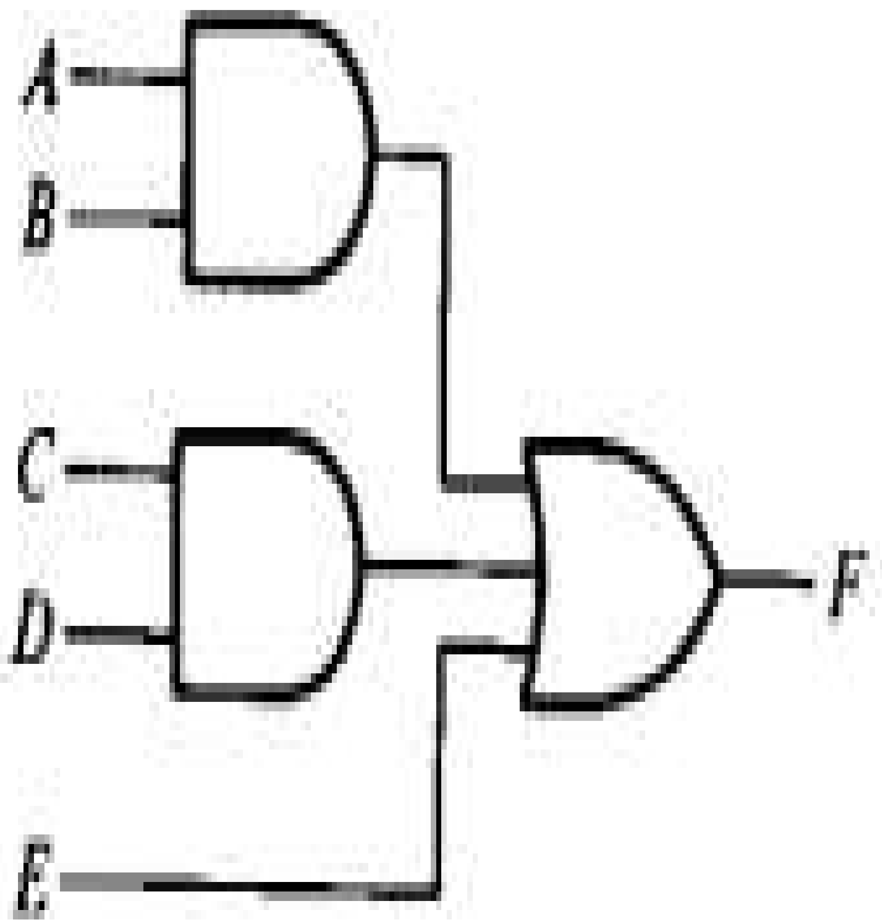
Symbolic Equivalence of NAND Gate

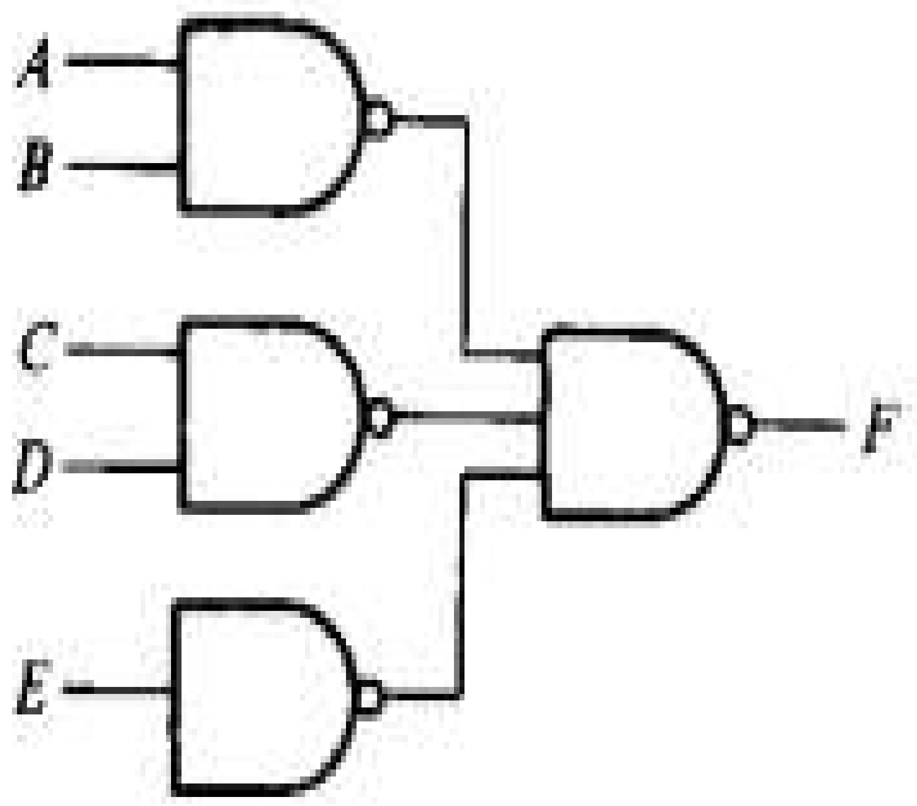


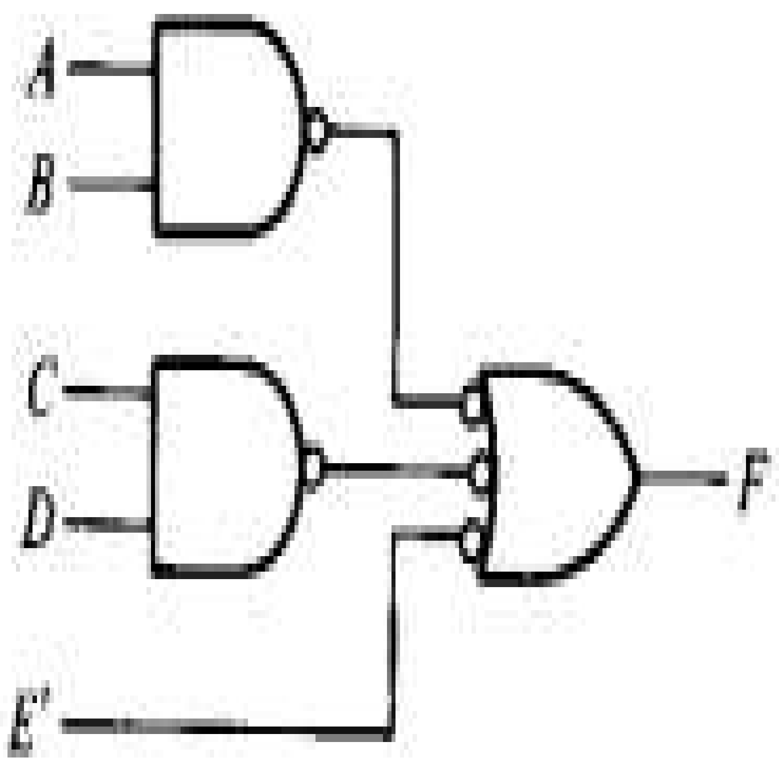
Two level implementation of NAND gate

➢The implementation of Boolean function with NAND gate requires that the function be simplified into sum of products form.

Eg. F = A.B + C.D + E

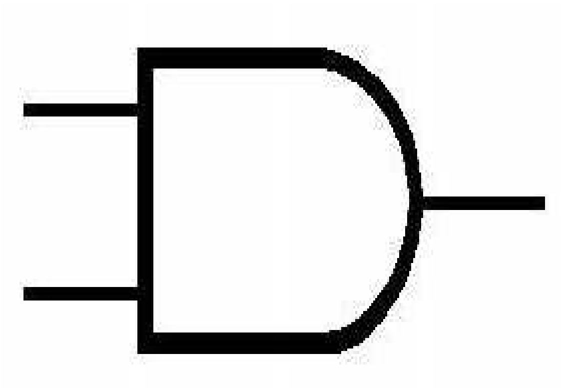




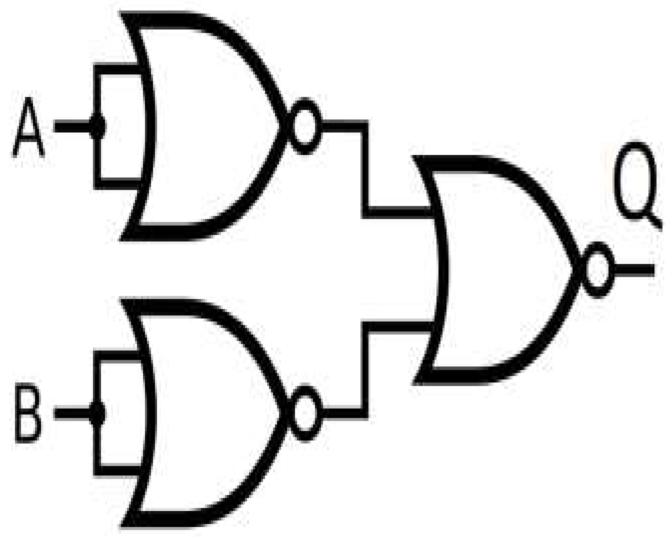


NOR implementation - Implementation of NOT and AND using NOR gate

➢By De Morgan's theorem putting two extra inverters in input we achieve AND gate by NOR gate



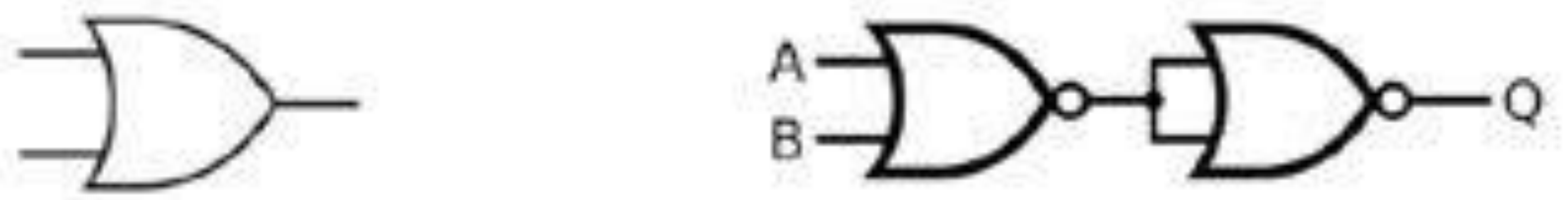
AND gate



NOR implementation of AND gate

NOR implementation - Implementation of OR gate using NOR gate

➢As NOR is the invert of OR gate so by putting an inverter in the output of NOR we get OR gate

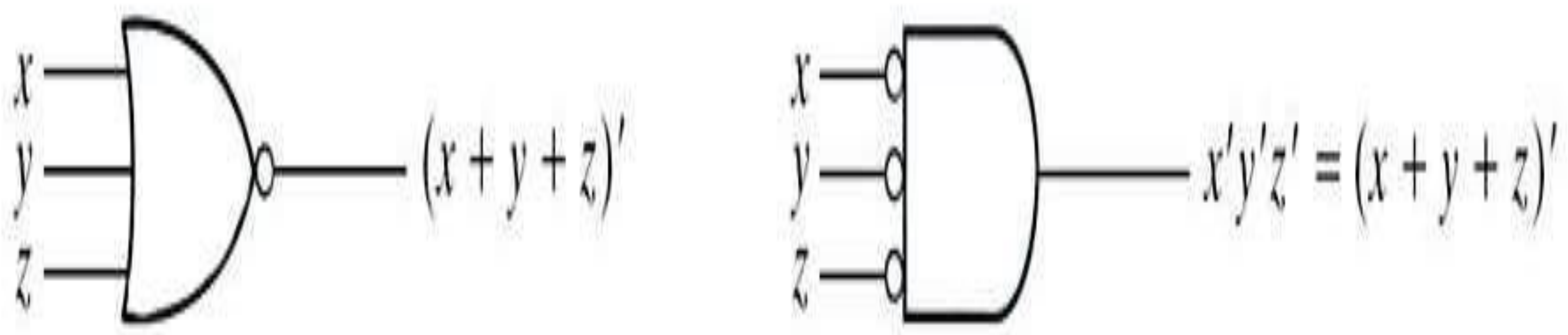


OR gate

NOR implementation of OR gate

Graphical equivalence of NOR gate

➢By De Morgan's Law we can describe NOR gate graphically by the following symbols

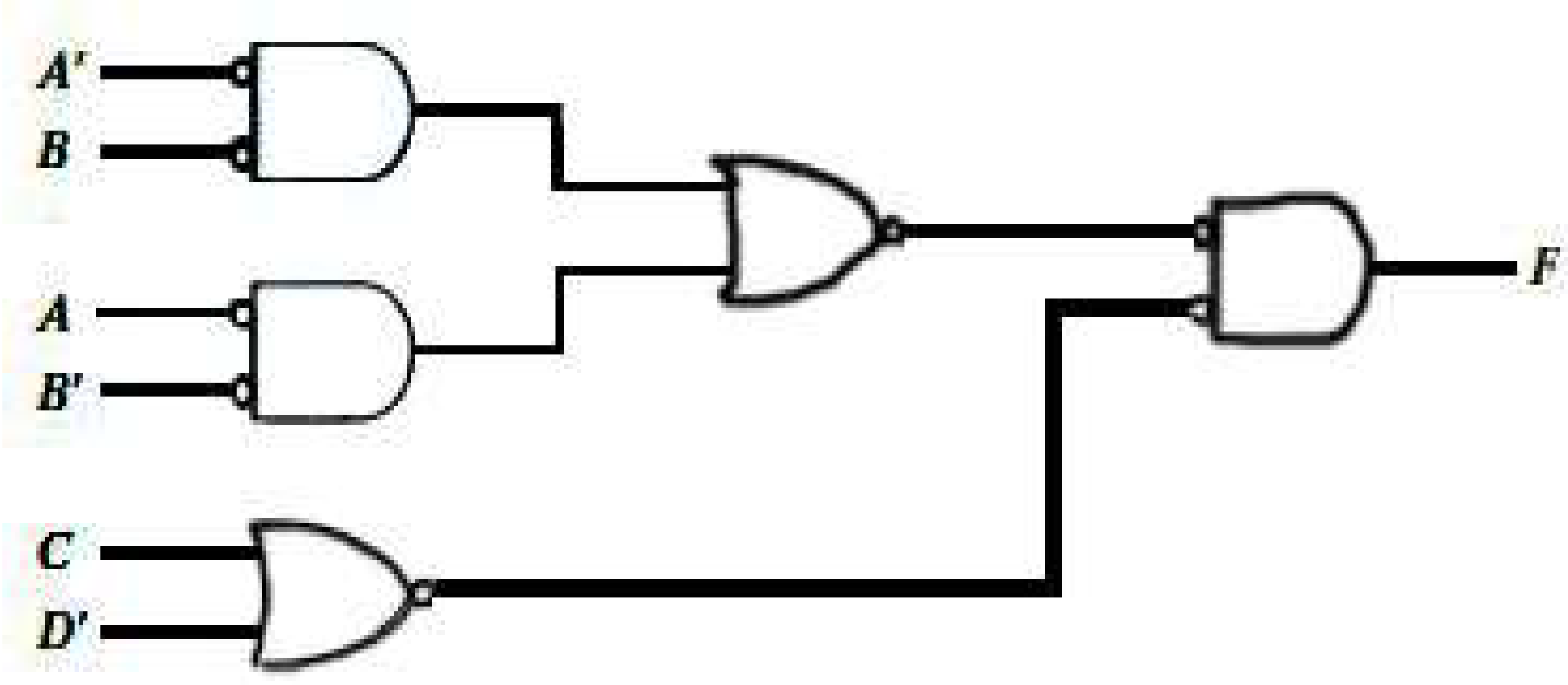


Two level implementation of NOR gate

➢A two-level implementation with NOR gates requires that the function be simplified into product of sums form.

Eg.

F = (AB' + A'B)(C + D')



Implementation using NOR gate

THANK YOU