Lecture Three / Logic Gats

1. Set of Gats **AND, OR, NOT, XOR, NOR, NAND.**
2. The Half-Adder Circuit.
3. The Full-Adder Circuit.
4. The Half-Subtractor Circuit.
5. The Full-Subtractor Circuit.

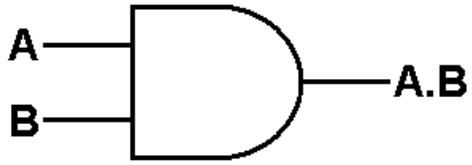
1. Basic Revision of Logic Gates

In the following chapter, we assume a two input logic gate with inputs **A** and **B**.

1.1 Simple Gates

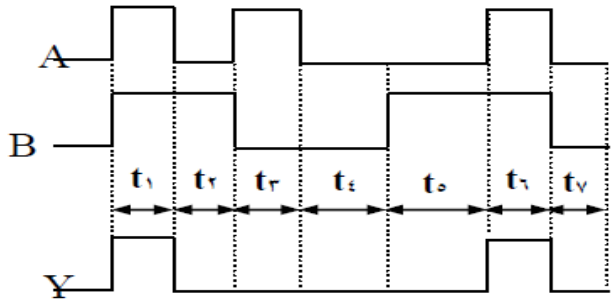
i. AND Gate

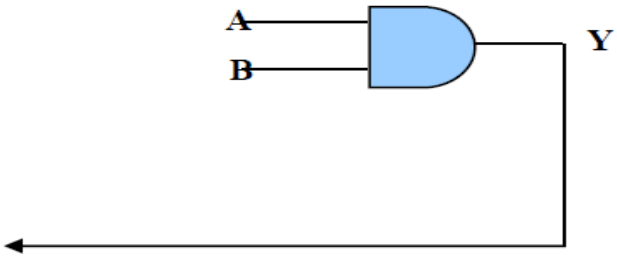
The symbol and truth table for an **AND** gate is shown below:



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

The Timing Diagram for an **AND** gate is shown below:



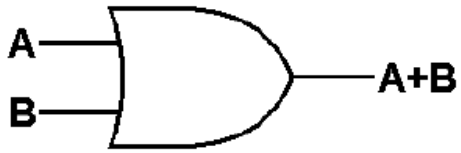


Summary: If inputs **A AND B** are high then the output is high

Abbreviations: often a '**.**' (dot) is used to signify the **AND** operator when writing Boolean equations.

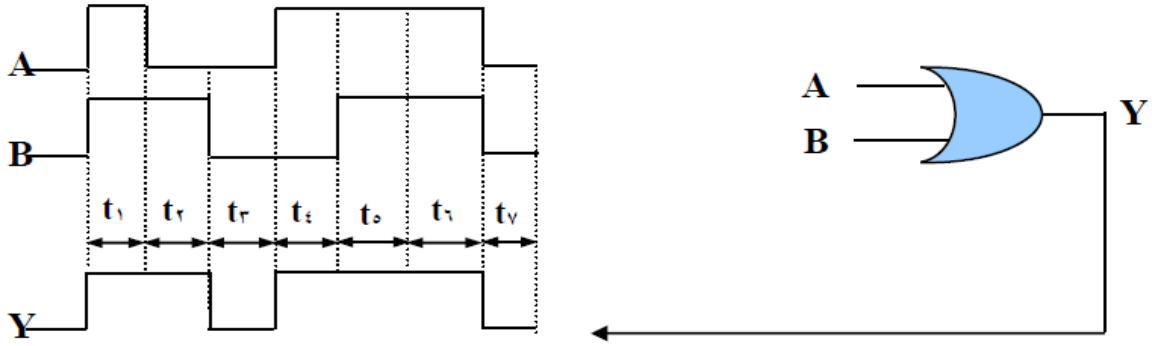
ii. OR Gate (Sometimes referred to as an Inclusive OR)

The symbol and truth table for an **OR** gate is shown below:



A	*B*	*A+B*
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**

The Timing Diagram for an **AND** gate is shown below:

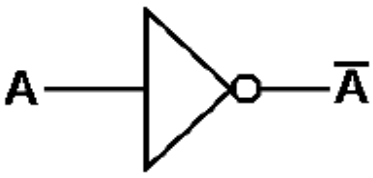


Summary: If inputs **A OR B** are high (or both) then the output is high.

Abbreviations: often a '+' is used to signify the **OR** operator when writing Boolean equations.

iii. NOT Gate

The symbol and truth table for an **NOT** gate is shown below:



A	$\overline{A}$
0	**0**
0	**1**

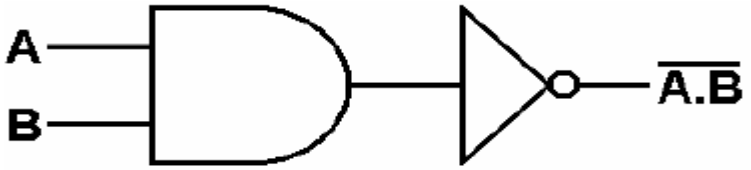
Summary: Output is **NOT** what the input is. They are also called '**inverters**'.

Abbreviations: Many sometimes a line above the thing that is being notted, sometimes a dash '**'** and sometimes an '**!**' exclamation mark.

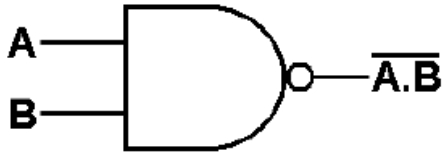
1.2 Complex Gates

i. NAND Gate

Made by combining an **AND** gate and a **NOT** gate. The equivalent circuit is:

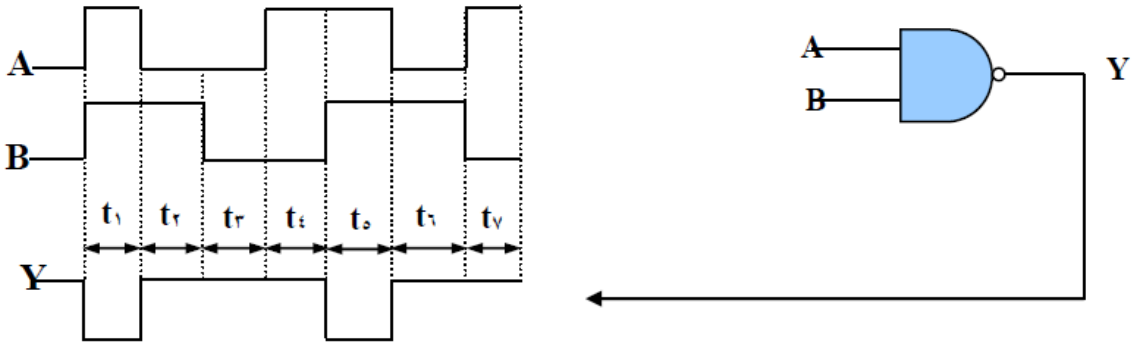


The symbol and truth table for an **NAND** gate is shown below:



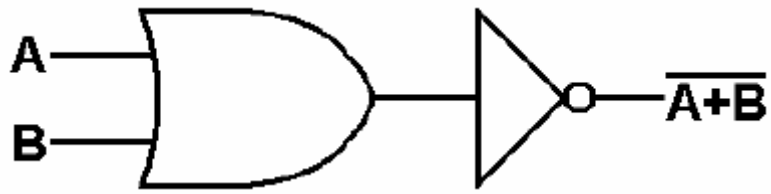
A	B	$\overline{A.B}$
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

The Timing Diagram for an **NAND** gate is shown below:

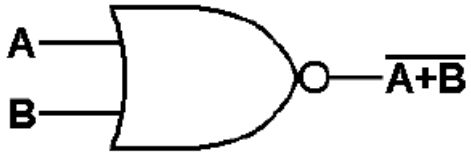


ii. NOR Gate

Made by combining an **OR** gate and a **NOT** gate. The equivalent circuit is:

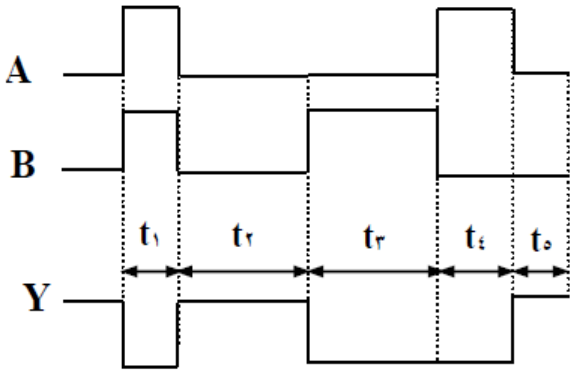


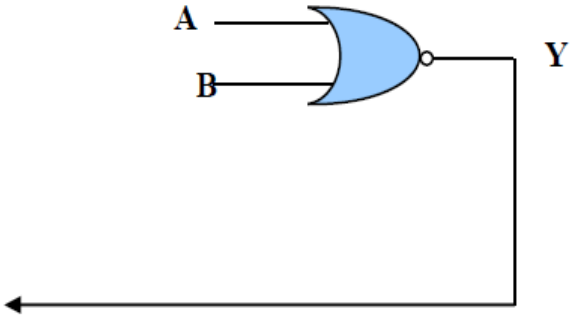
The symbol and truth table for an **NOR** gate is shown below:



A	B	$\overline{A+B}$
0	0	**1**
0	1	**0**
1	0	**0**
1	1	**0**

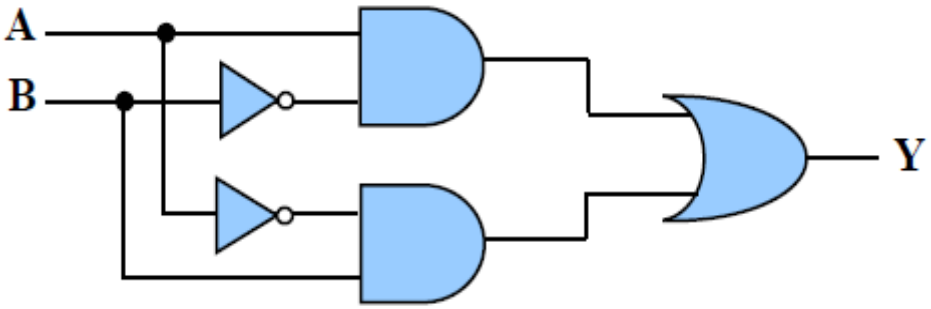
The Timing Diagram for an **NOR** gate is shown below:



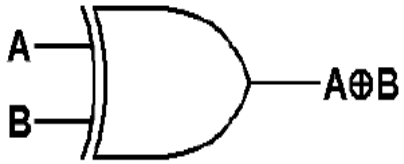


iii. XOR Gate (Exclusive OR)

This gate gives a high output when A OR B are high, but not both. The equivalent circuit is:



The symbol for a **XOR** gate is:

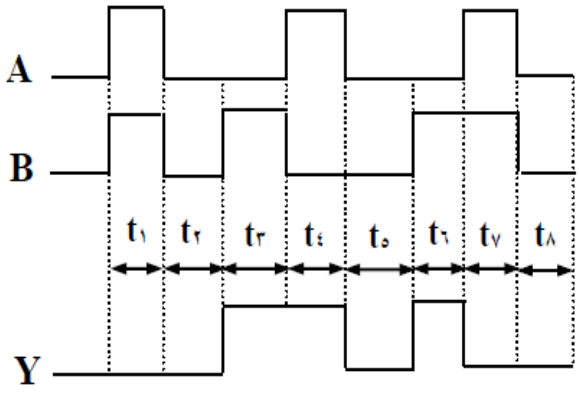


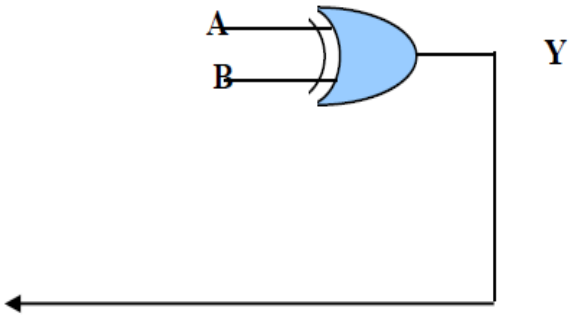
A	B	$A \oplus B$
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

Summary: $A \oplus B = A.\overline{B} + \overline{A}.B$

Abbreviations: **XOR** gates are represented with a $\oplus$ symbol.

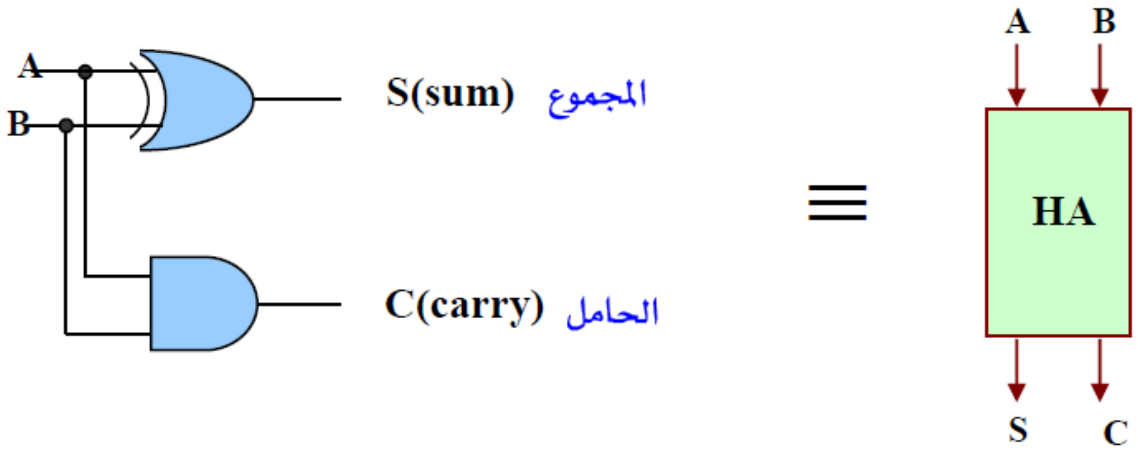
The Timing Diagram for an **XOR** gate is shown below:





2. <u>The Half-Adder Circuit:</u> The basic digital arithmetic circuit is the addition of two binary digits. Input variables of a Half-Adder call augends & addend bits. The output variables the sum & carry. The symbol and truth table for an Half-Adder gate is shown below:

❖ ***<u>Logic diagram for half adder:</u>***



❖ ***<u>Truth table for half adder:</u>***

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

❖***<u>Half- Adder questions:</u>*** The Boolean functions for the two outputs can be obtained directly from the truth table as:

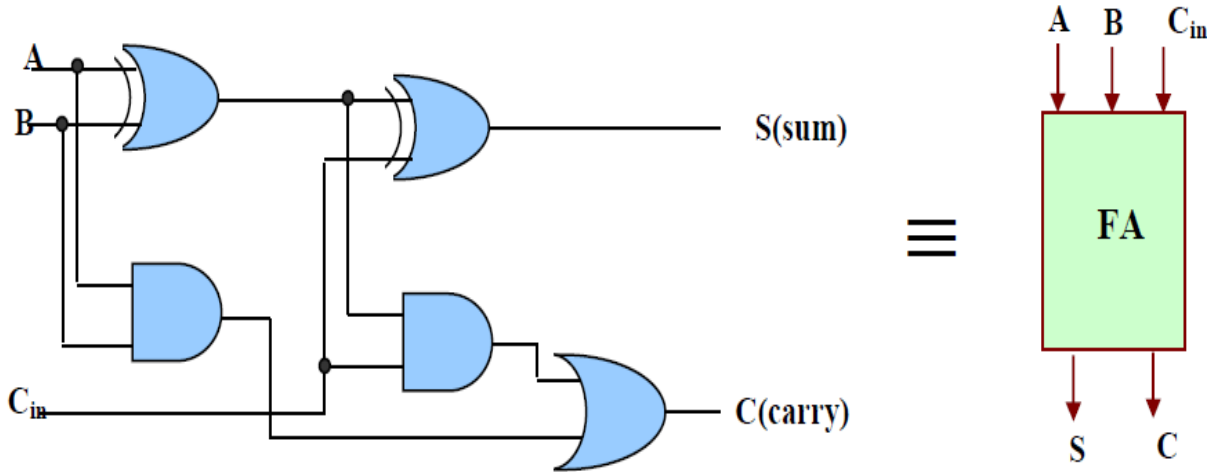
$$S = \bar{A}B + A\bar{B}$$

$$S = A \oplus B$$

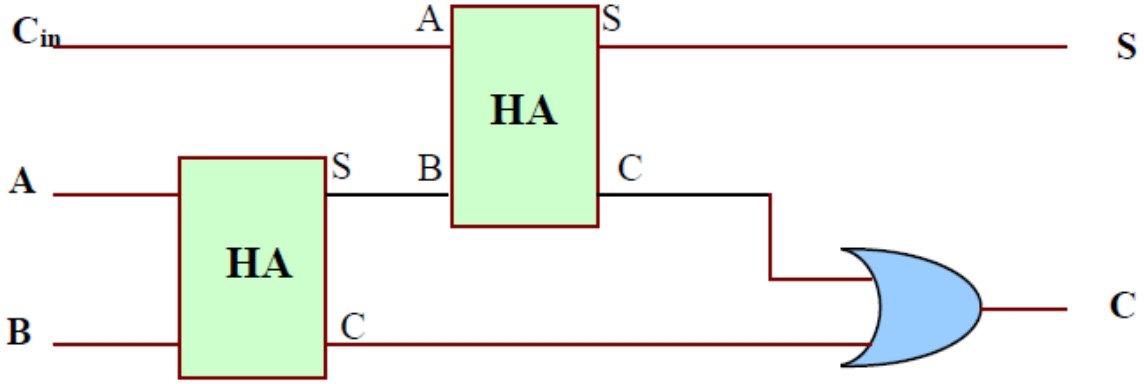
$$C = A * B$$

3. <u>The Full-Adder Circuit:</u> A full-adder is a combinational circuit that forms the arithmetic sum of three input bits. It consists of three inputs & two outputs. The symbol and truth table for an Full-Adder gate is shown below:

❖ ***<u>Logic circuit for full adder (Logic Diagram):</u>***



❖ ***<u>The block diagram for full-adder:</u>***



❖ ***Truth table for full adder:***

Inputs			Out puts	
A	B	C_{in}	C	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	0
1	1	1	1	1

❖ ***Full-Adder questions:*** The Boolean functions for the two outputs can be obtained directly from the truth table as:

$$S = \overline{A}\overline{B}C_{in} + \overline{A}B\overline{C_{in}} + A\overline{B}\overline{C_{in}} + ABC_{in}$$

$$S = A \oplus B \oplus C_{in}$$

$$S = (A \oplus B) \oplus C_{in}$$

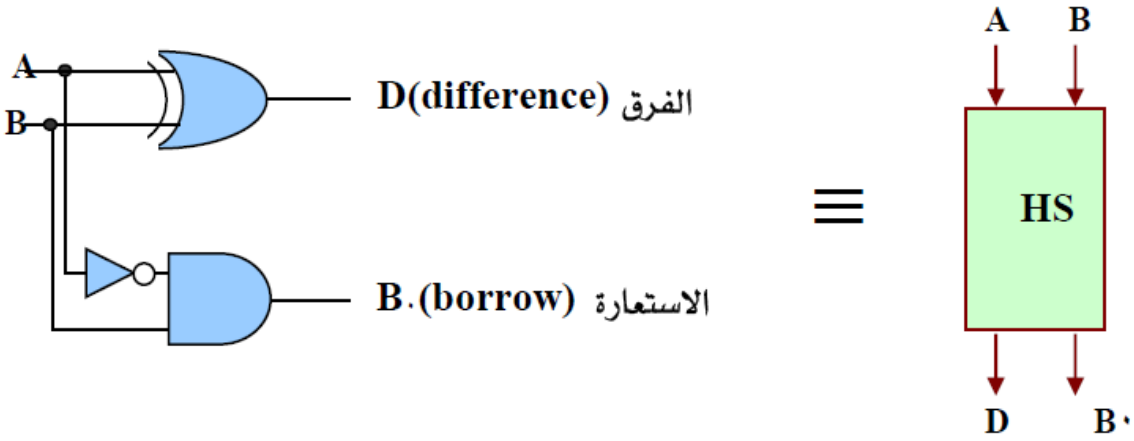
$$C = AB + A\overline{B}C_{in} + \overline{A}BC_{in}$$

$$C = AB + C_{in}(A\overline{B} + \overline{A}B)$$

$$C = AB + C_{in}(A \oplus B)$$

4. <u>The Half-Subtractor Circuit:</u> A half subtractor is an arithmetic circuit that subtracts two bits and produces their difference. The block diagram of half subtractor is shown. The circuit has two inputs minuend (A) and subtrahend (B) and two output bits, one is the difference bit (D) and the other is the borrow bit (B).

❖ ***<u>Logic circuit for Half-Subtractor:</u>***



❖ ***<u>Truth table for Half-Subtractor:</u>***

INPUTS			OUTPUTS	
A	B		B	D
0	0		0	0
0	1		1	1
1	0		0	1
1	1		0	0

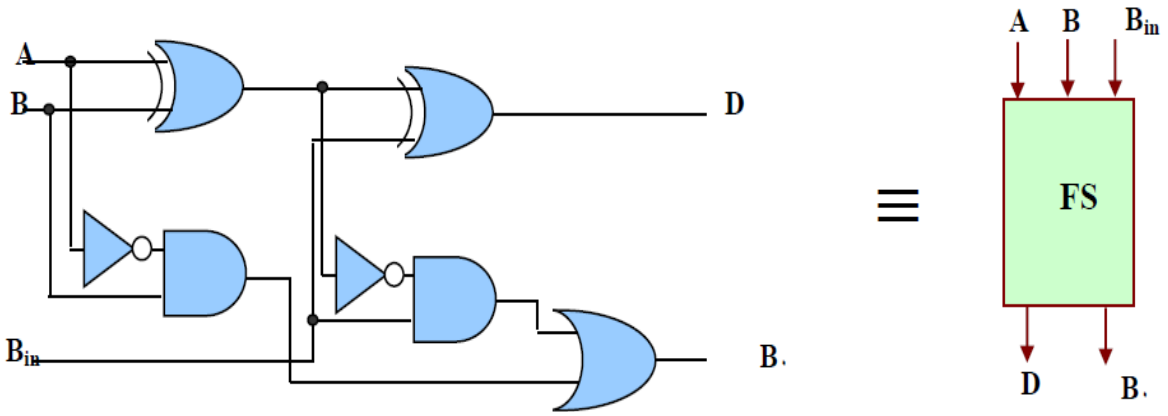
❖ ***<u>Half-Subtractor questions:</u>***

$$D = (\bar{A}B + \bar{B}A) = A \oplus B$$

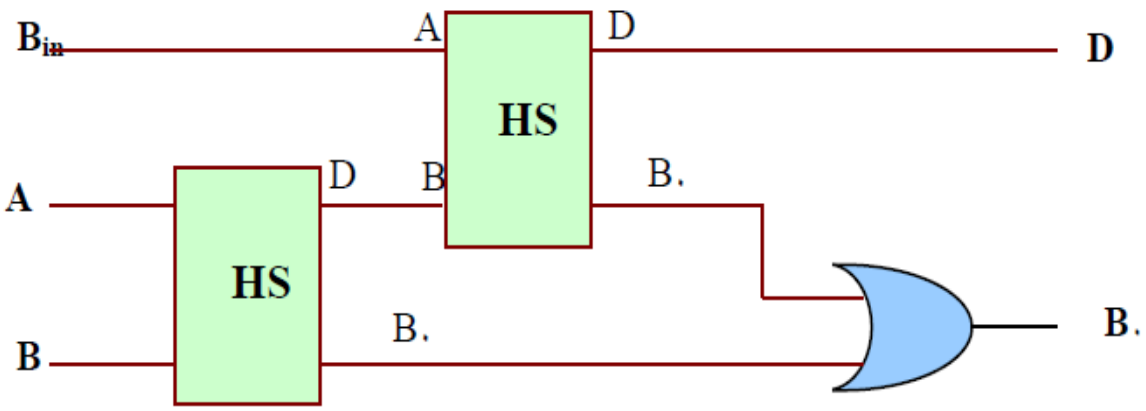
$$B = \bar{A}B$$

5.<u>The Full-Subtractor Circuit:</u> A full subtractor is a combinational circuit that performs a subtraction between two bits, taking into account that a 1 may have been borrowed by a lower significant bit. The block diagram of full subtractor is shown. The circuit has three inputs and two outputs.

❖ ***<u>Logic circuit for Full-Subtractor:</u>***



❖ ***<u>The block diagram for Full-Subtractor:</u>***



❖ *Truth table for Full-Subtractor:*

INPUTS			OUTPUTS	
A	B	B_{in}	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

❖ *Full-Subtractor questions:*

$$D = \overline{A}\overline{B}B_{in} + \overline{A}B\overline{B_{in}} + A\overline{B}\overline{B_{in}} + ABB_{in}$$

$$D = A \oplus B \oplus B_{in}$$

$$B = \overline{A}B + \overline{A}B_{in} + BB_{in}$$