SCHOOL OF ELECTRICAL ENGINEERING
FACULTY OF ENGINEERING

SKEE 1223 DIGITAL ELECTRONICS

CHAPTER 3: GATES AND BOOLEAN ALGEBRA

DR. MOHD SAIFUL AZIMI BIN MAHMUD

P19a-04-03-30

School of Electrical Engineering

Faculty of Engineering

Universiti Teknologi Malaysia

019-7112948

azimi@utm.my

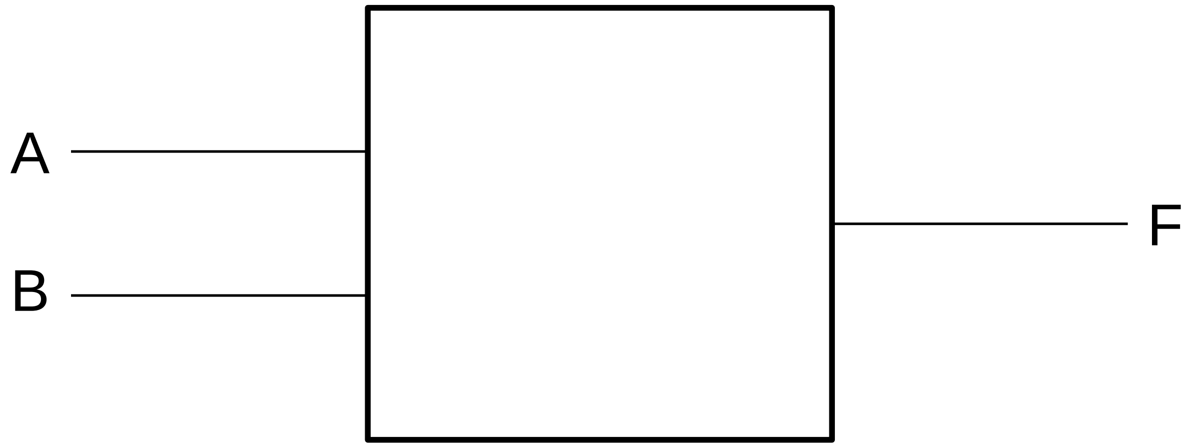
innovative • entrepreneurial • global

LOGIC GATES

LOGIC GATES

INTRODUCTION

- **Logic gate** is fundamental building blocks for all digital circuits.
- Each gate **at least have one input** and **only one output**.
- The output depends on the function of the gate and combination of all its inputs.

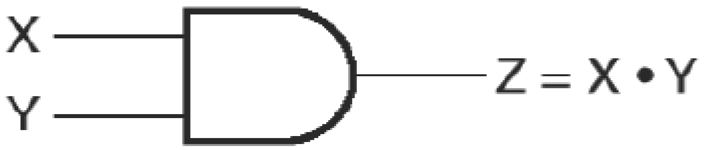


A generic two-input logic gate

LOGIC GATES

AND GATE: INTRODUCTION

- It has two or more inputs and one output.
- Performs as **logic multiplication**.



AND symbol

Logic expression

$Z = XY$ or $Z = X \cdot Y$

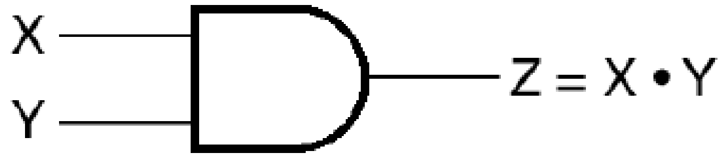
AND Truth Table

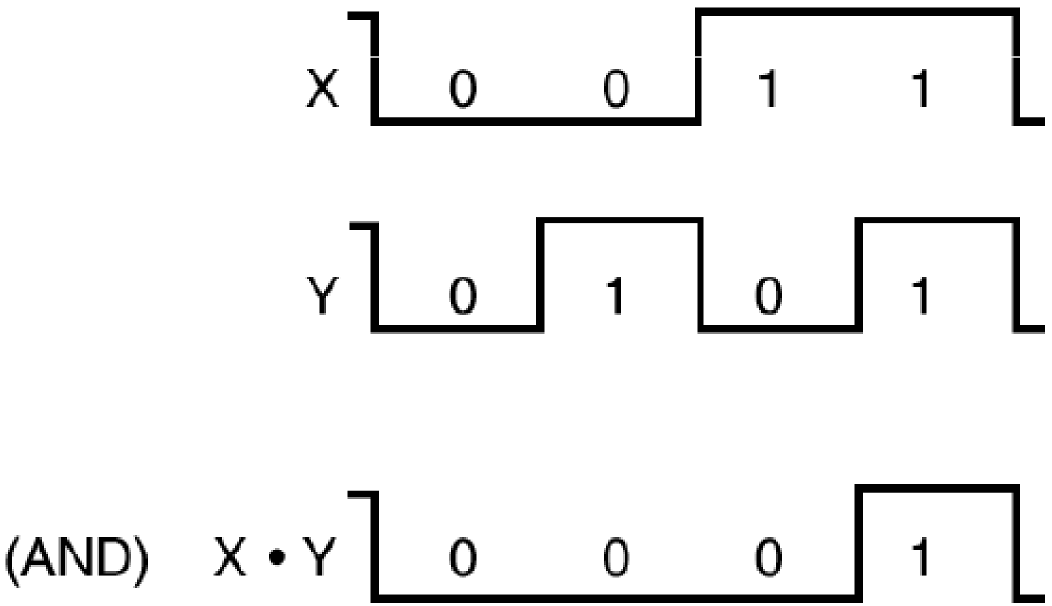
Input		Output
X	Y	**Z**
0	0	**0**
0	1	**0**
1	0	**0**
1	1	**1**

Logic operation

- Output Z is **1** when **all input X and Y** are **1**.
- Output Z is **0** when **at least one of X and Y** is **0**.

LOGIC GATES
AND GATE: TIMING DIAGRAM

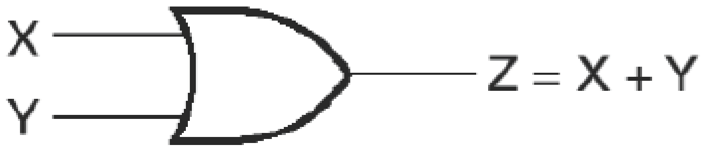




LOGIC GATES

OR GATE: INTRODUCTION

- It has two or more inputs and one output.
- Performs as **logic addition**.



OR symbol

Logic expression

$$Z = X + Y$$

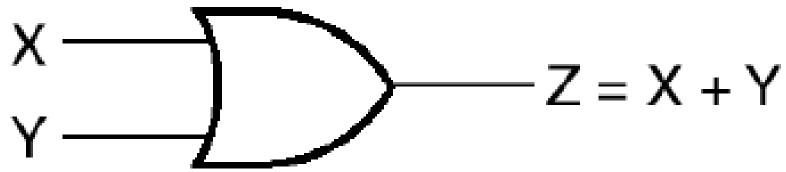
OR Truth Table

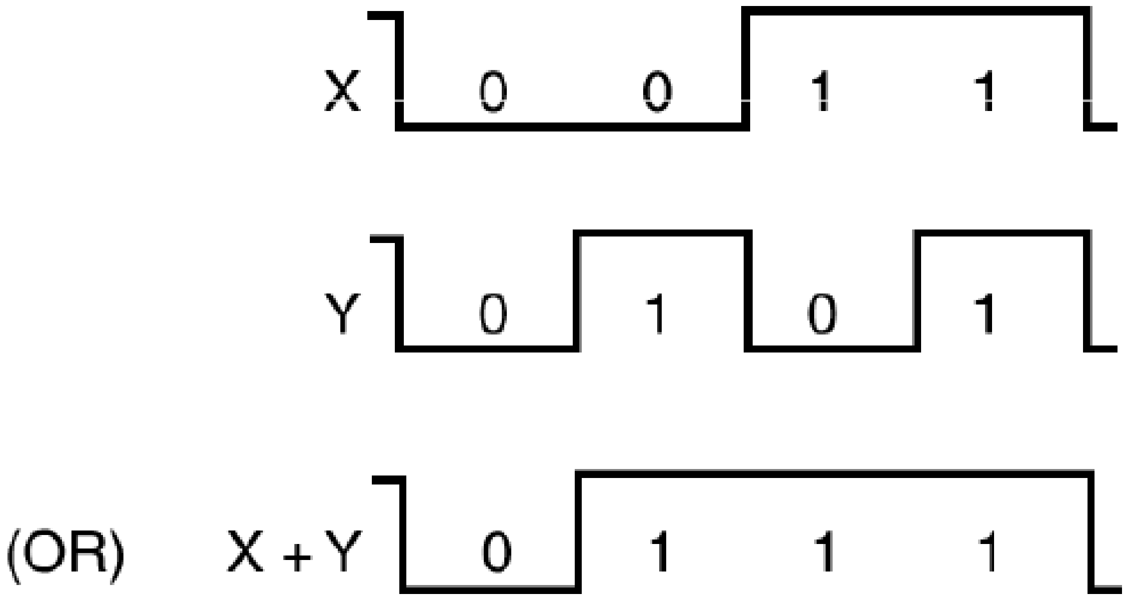
Input		Output
X	Y	**Z**
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**

Logic operation

- Output Z is **1** when **at least one of X and Y** is **1**.
- Output Z is **0** when **all input X and Y** are **0**.

LOGIC GATES
OR GATE: TIMING DIAGRAM

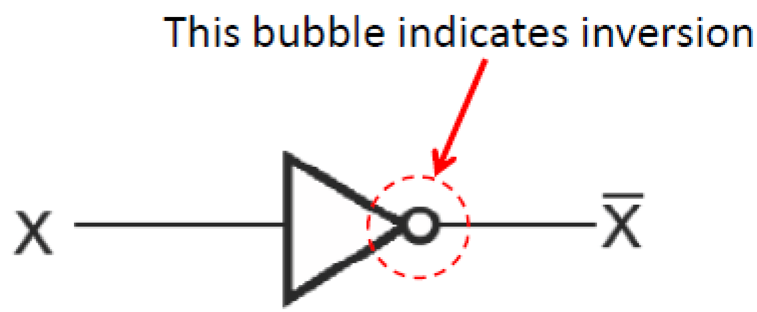




LOGIC GATES

NOT GATE: INTRODUCTION

- Also known as **inverter gate**.
- It only has one input and one output.
- Performs as **logic inversion**.



NOT symbol

Logic expression

$Z = \bar{X}$ or $Z = X'$

Logic operation

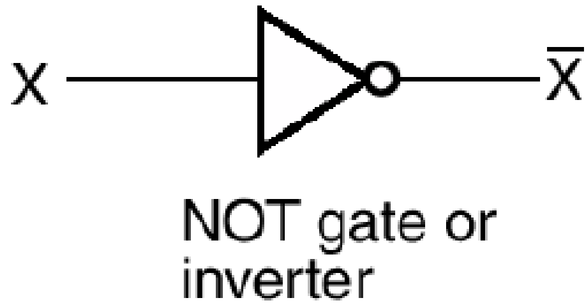
- Output Z is opposite of input X.

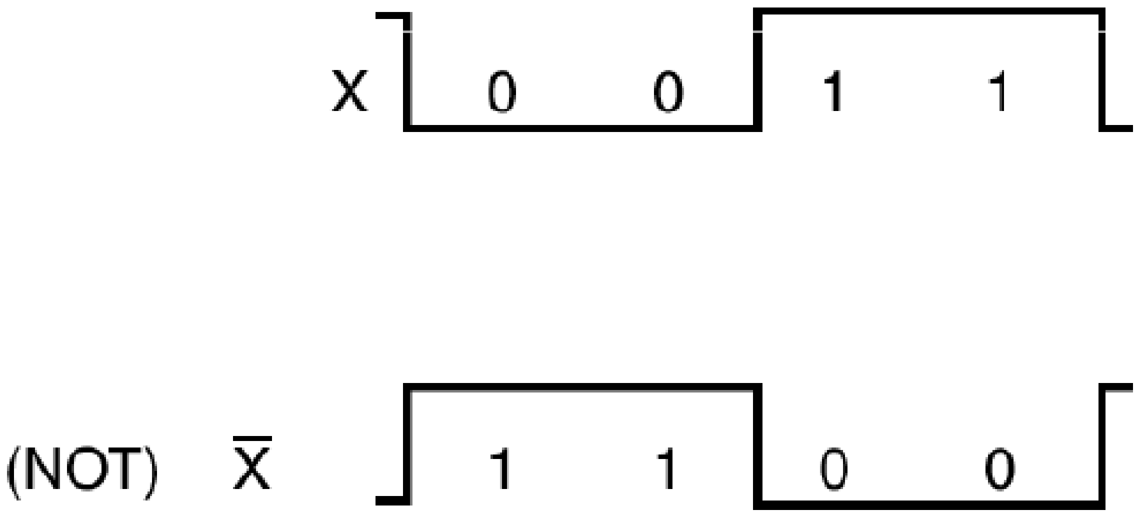
NOT Truth Table

Input	**Output**
X	**Z**
0	**1**
1	**0**

LOGIC GATES

NOT GATE: TIMING DIAGRAM





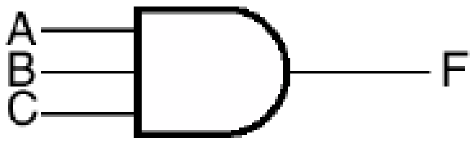
LOGIC GATES

MORE INPUTS GATE?

- Works the same way.

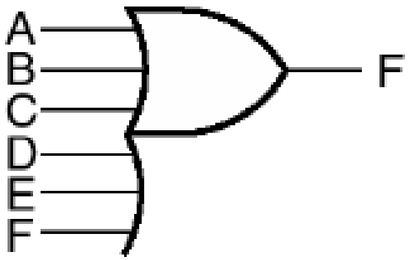
Example

Three inputs AND gate



- Logic expression: $F = ABC$
- Output: F is 1 when all inputs are 1.

Six inputs OR gate

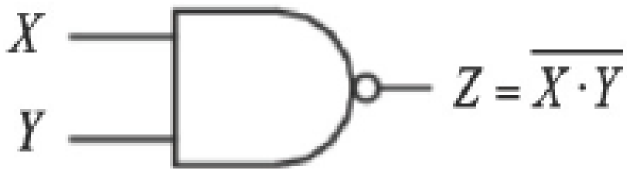


- Logic expression: $F = A + B + C + D + E + F$
- Output: F is 1 when at least one input is 1.

LOGIC GATES

NAND GATE: INTRODUCTION

- A **Universal gate**: used in combinations to perform AND, OR and NOT operations.
- NAND is a contraction of NOT-AND (implies AND function with an inverted output).



NAND symbol

Logic expression

$$Z = \overline{XY}$$

Logic operation

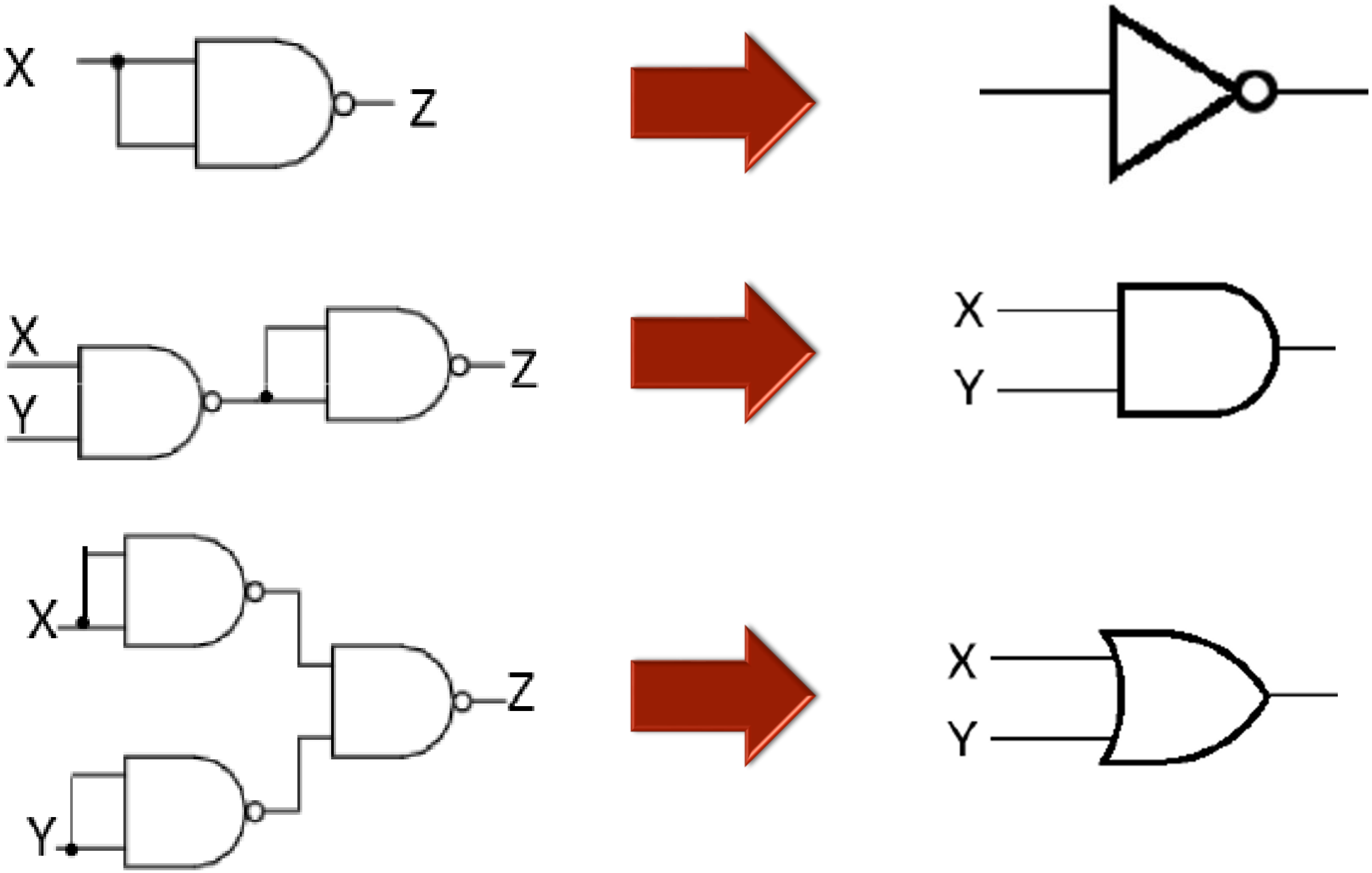
- Output Z is **1** when **at least one of X or Y** is **0**.
- Output Z is **0** when **all input X and Y** are **1**.

NAND Truth Table

Input		**Output**
X	Y	**Z**
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

LOGIC GATES

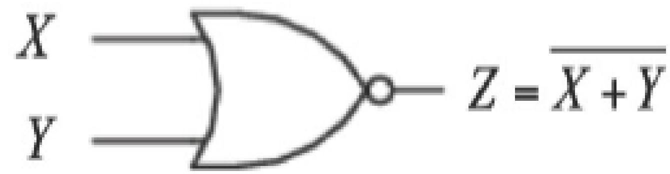
NAND GATE IS UNIVERSAL GATE



LOGIC GATES

NOR GATE: INTRODUCTION

- A **Universal gate**: used in combinations to perform AND, OR and NOT operations.
- NOR is a contraction of NOT-OR (implies OR function with an inverted output).



NOR symbol

Logic expression

$$Z = \overline{X + Y}$$

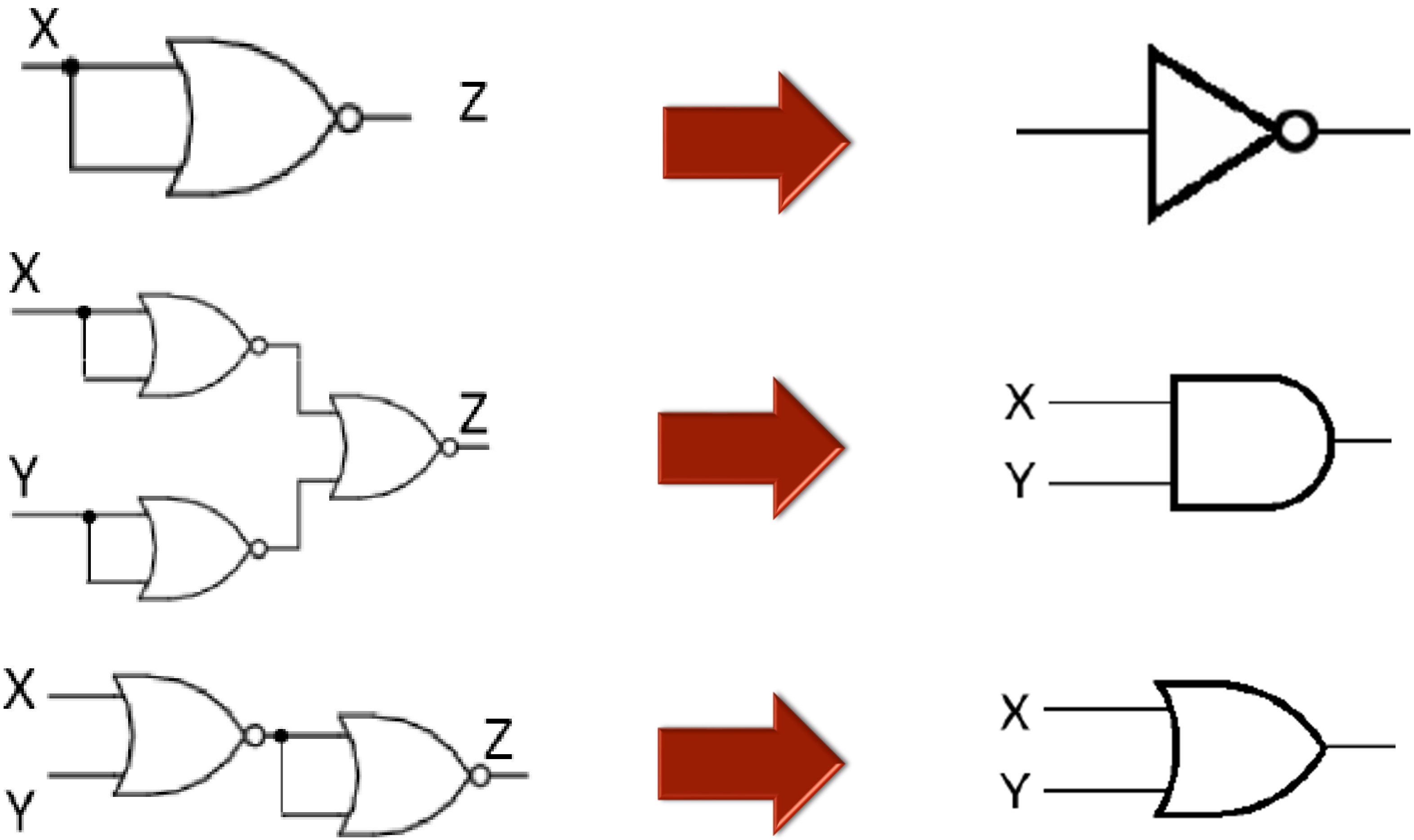
Logic operation

- Output Z is **1** when **all input X and Y** are **0**.
- Output Z is **0** when **at least one of X or Y** is **1**.

NOR Truth Table

Input		Output
X	Y	**Z**
0	0	**1**
0	1	**0**
1	0	**0**
1	1	**0**

LOGIC GATES
NOR GATE IS UNIVERSAL GATE



LOGIC GATES

EXCLUSIVE-OR GATE (XOR GATE): INTRODUCTION

- XOR gate only has two inputs.

XOR symbol

Logic expression

$Z = \bar{X}Y + X\bar{Y}$ or $Z = X \oplus Y$

Logic operation

- Output Z is **1** when **input *X* and *Y*** are **different**.
- Output Z is **0** when **input *X* and *Y*** are **same**.

XOR Truth Table

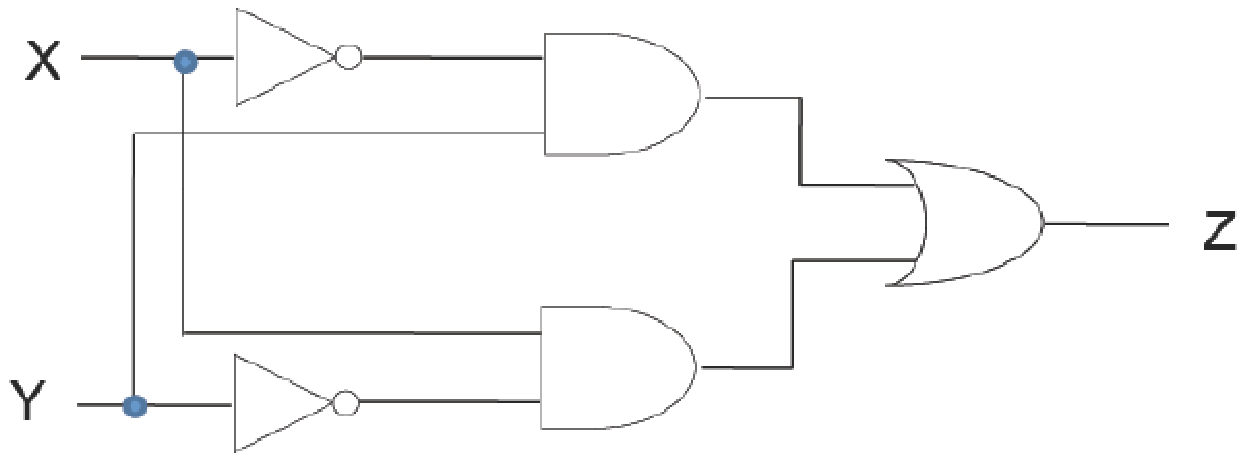
Input		**Output**
X	Y	**Z**
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

LOGIC GATES

EXCLUSIVE-OR GATE (XOR GATE): INTRODUCTION

- How to make XOR gate using basic gates (AND, OR and NOT)?

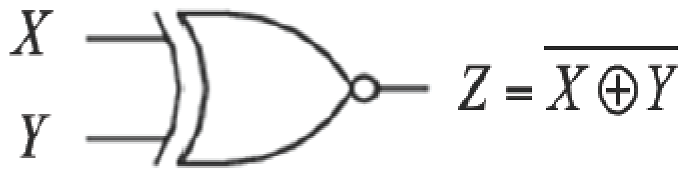
$$Z = X \oplus Y$$
$$Z = \bar{X} \cdot Y + X \cdot \bar{Y}$$



LOGIC GATES

EXCLUSIVE-NOR GATE (XNOR GATE): INTRODUCTION

- XNOR gate only has two inputs.
- The bubble on the output of XNOR symbol indicate that its outputs opposite that of XOR gate.



XNOR symbol

Logic expression

$Z = \overline{\bar{X}Y + X\bar{Y}}$ or $Z = \overline{X \oplus Y}$

Logic operation

- Output Z is **1** when **input X and Y** are **same**.
- Output Z is **0** when **input X and Y** are **different**.

XNOR Truth Table

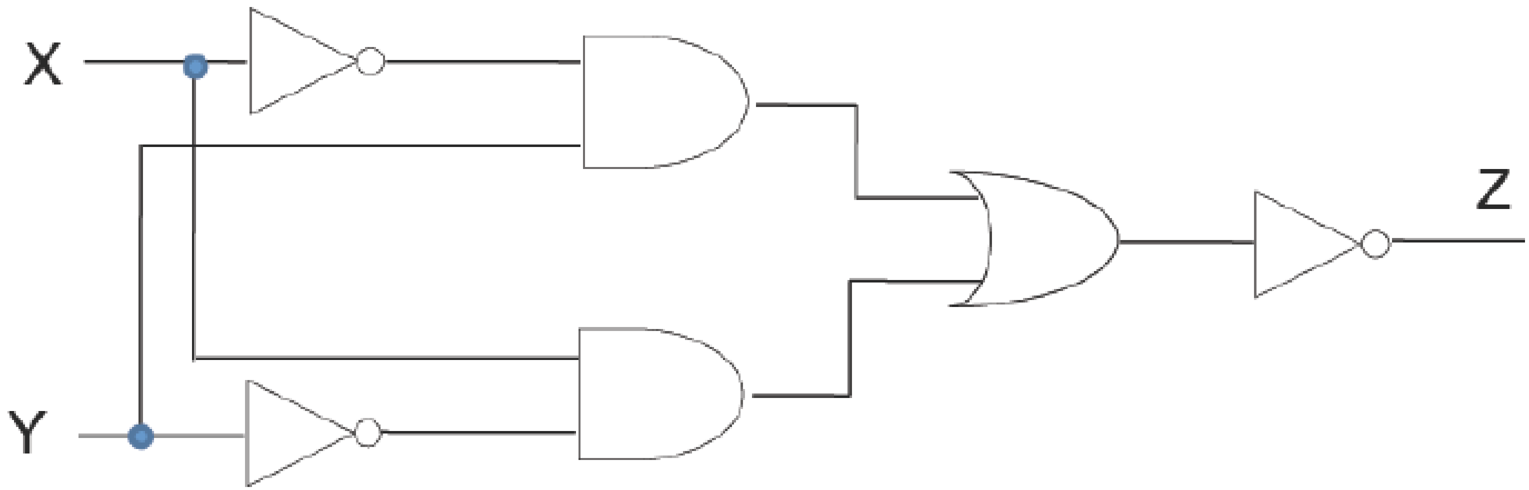
Input		**Output**
X	Y	**Z**
0	0	**1**
0	1	**0**
1	0	**0**
1	1	**1**

LOGIC GATES

EXCLUSIVE-NOR GATE (XNOR GATE): INTRODUCTION

- How to make XNOR gate using basic gates (AND, OR and NOT)?

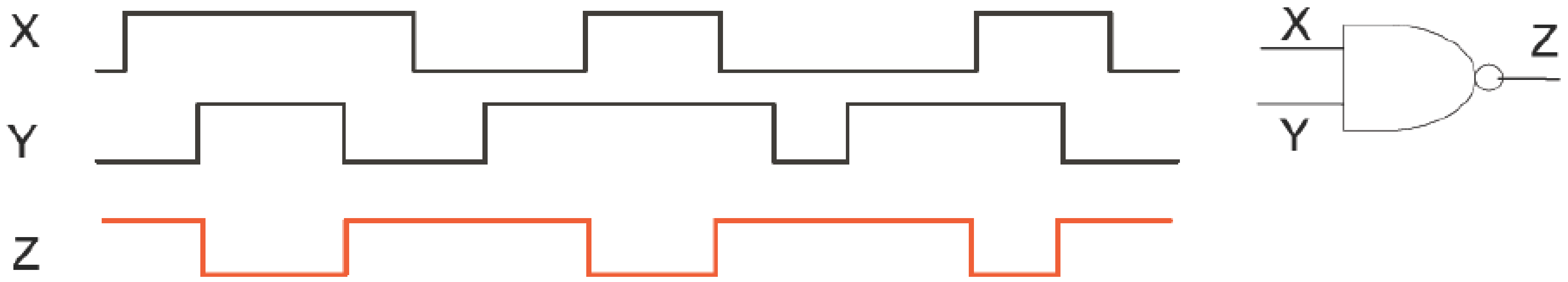
$$Z = \overline{X \oplus Y}$$
$$Z = \overline{\bar{X} \cdot Y + X \cdot \bar{Y}}$$

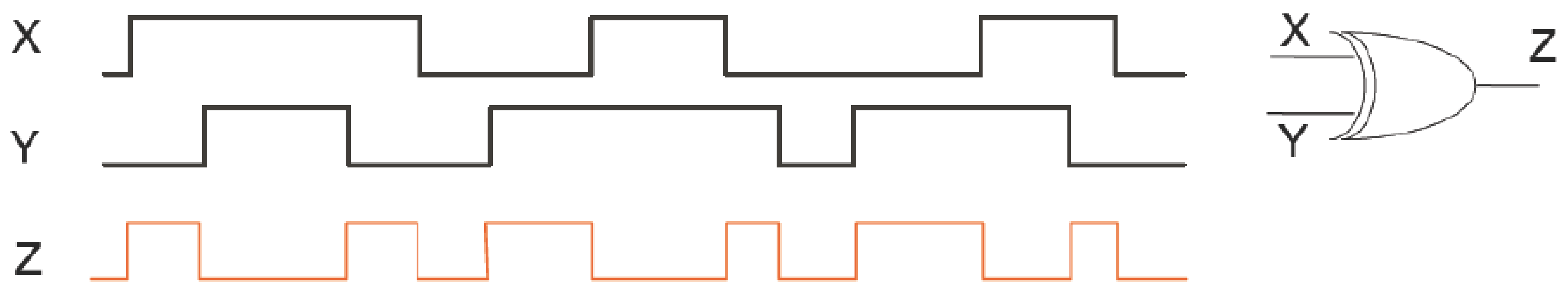


LOGIC GATES

ASSESSMENT 1

Draw the timing diagram for the following:

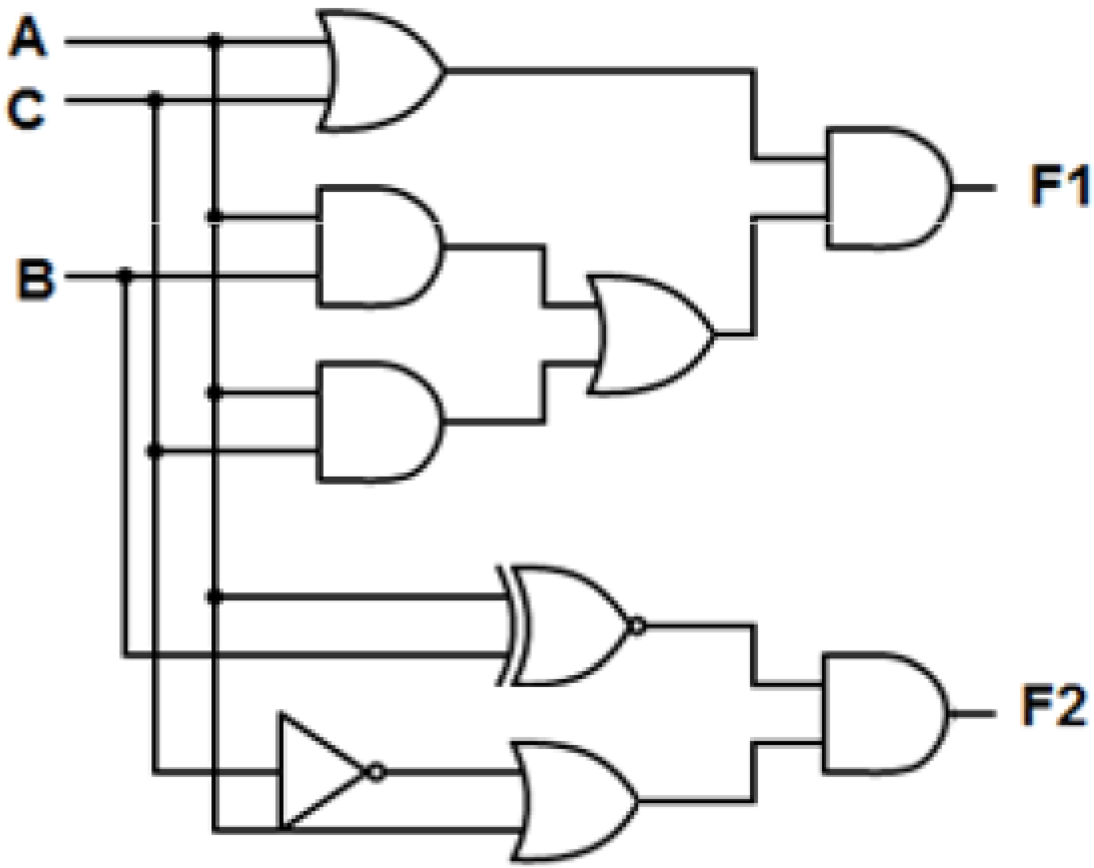




LOGIC GATES

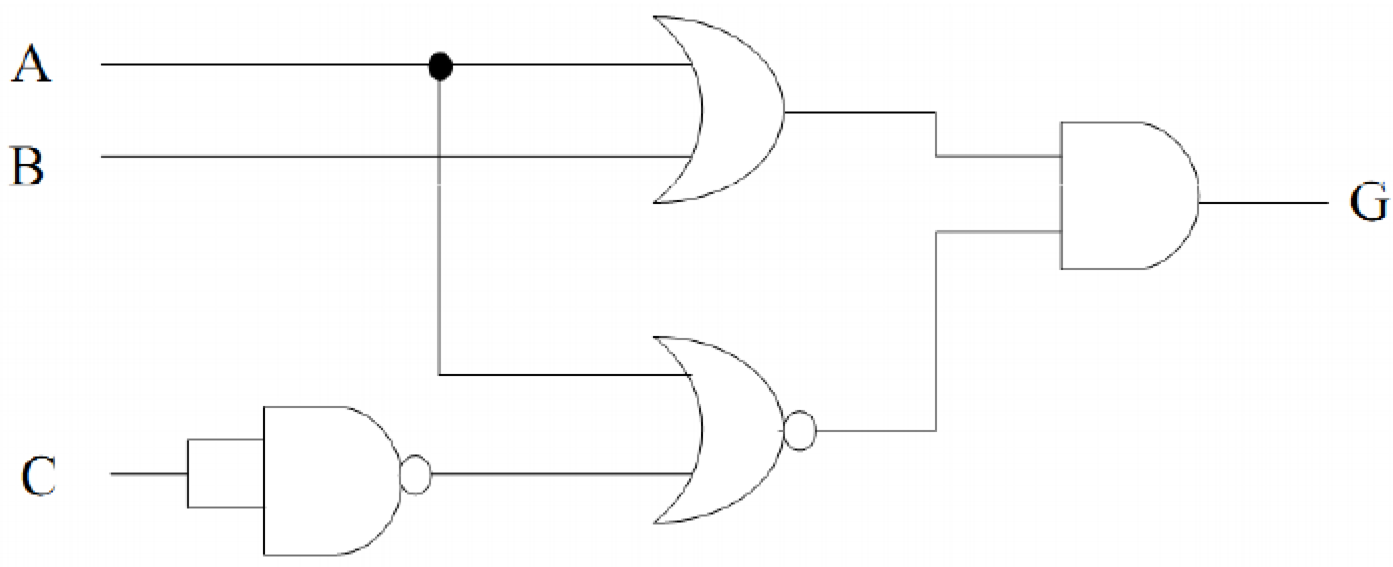
ASSESSMENT 2

For the given circuit as shown below, obtain Boolean expression of F1 and F2?



LOGIC GATES
HOMEWORK

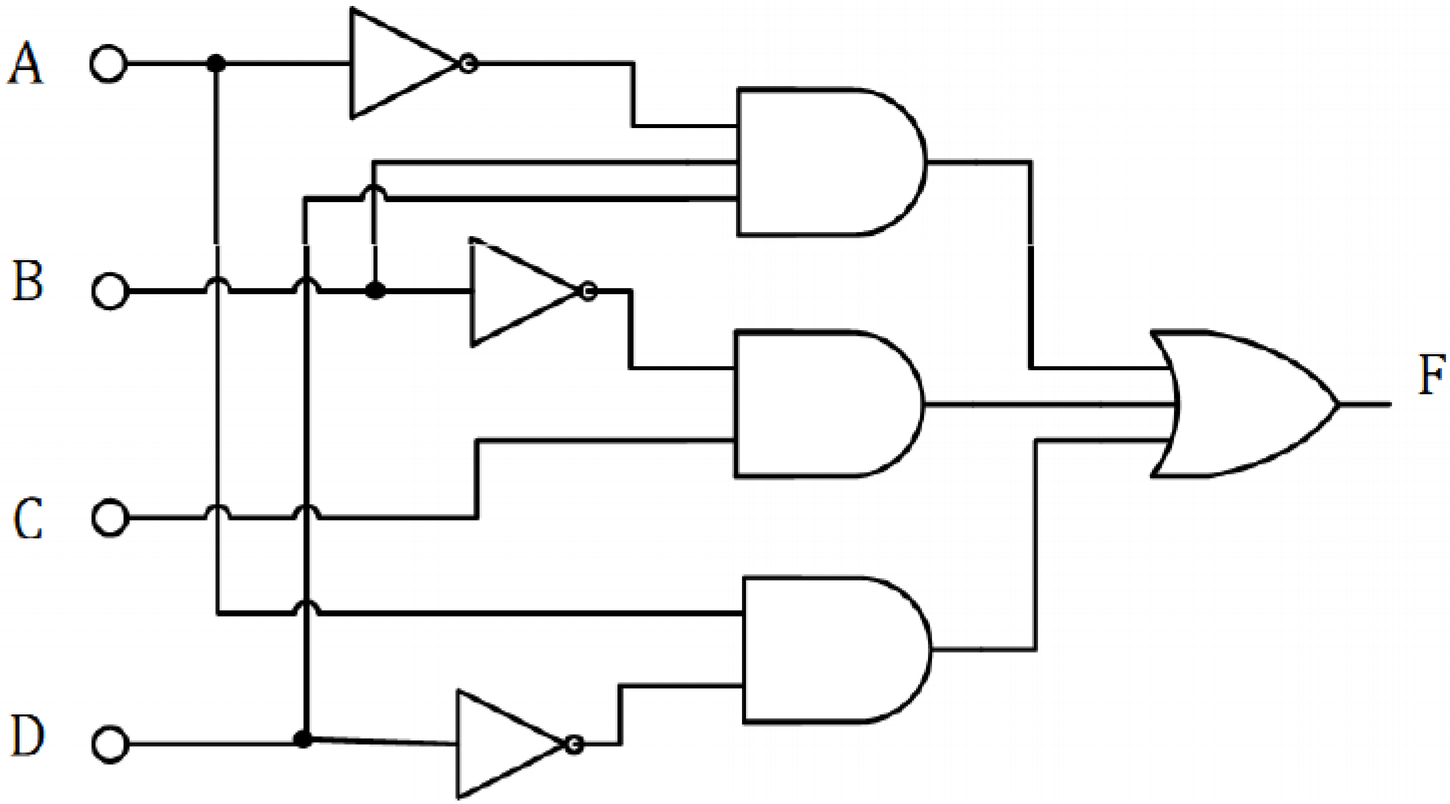
Write the algebraic expression for the following circuit.



LOGIC GATES

HOMEWORK

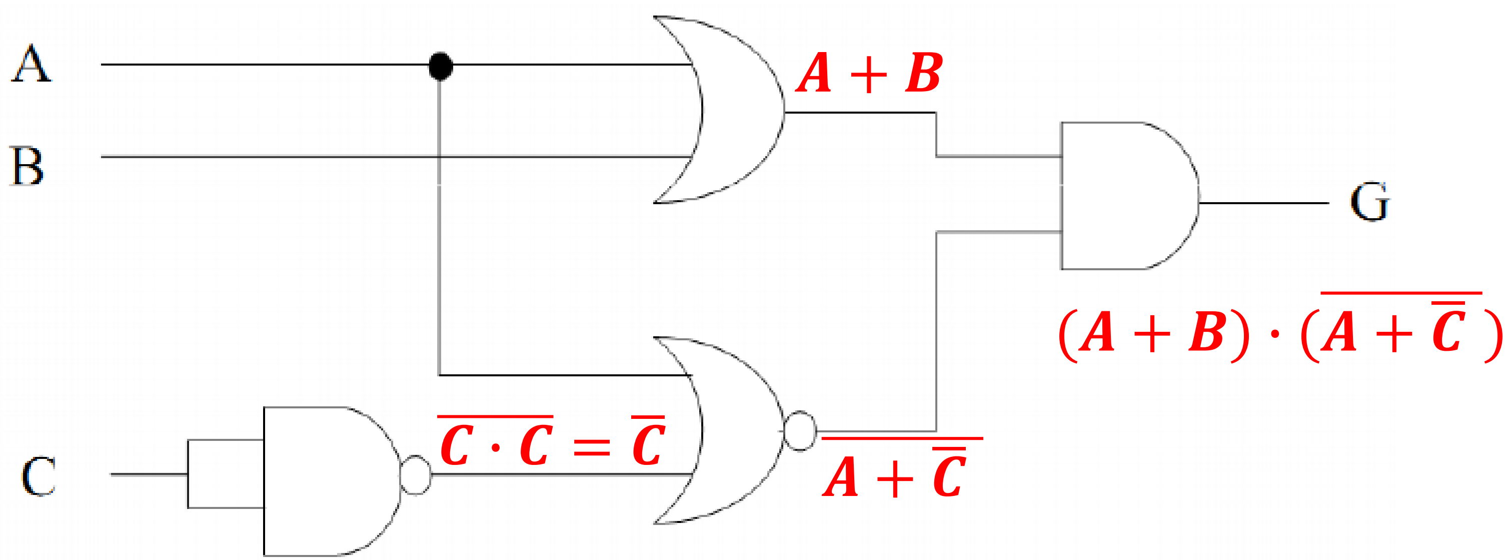
Write the algebraic expression for the following circuit.



LOGIC GATES

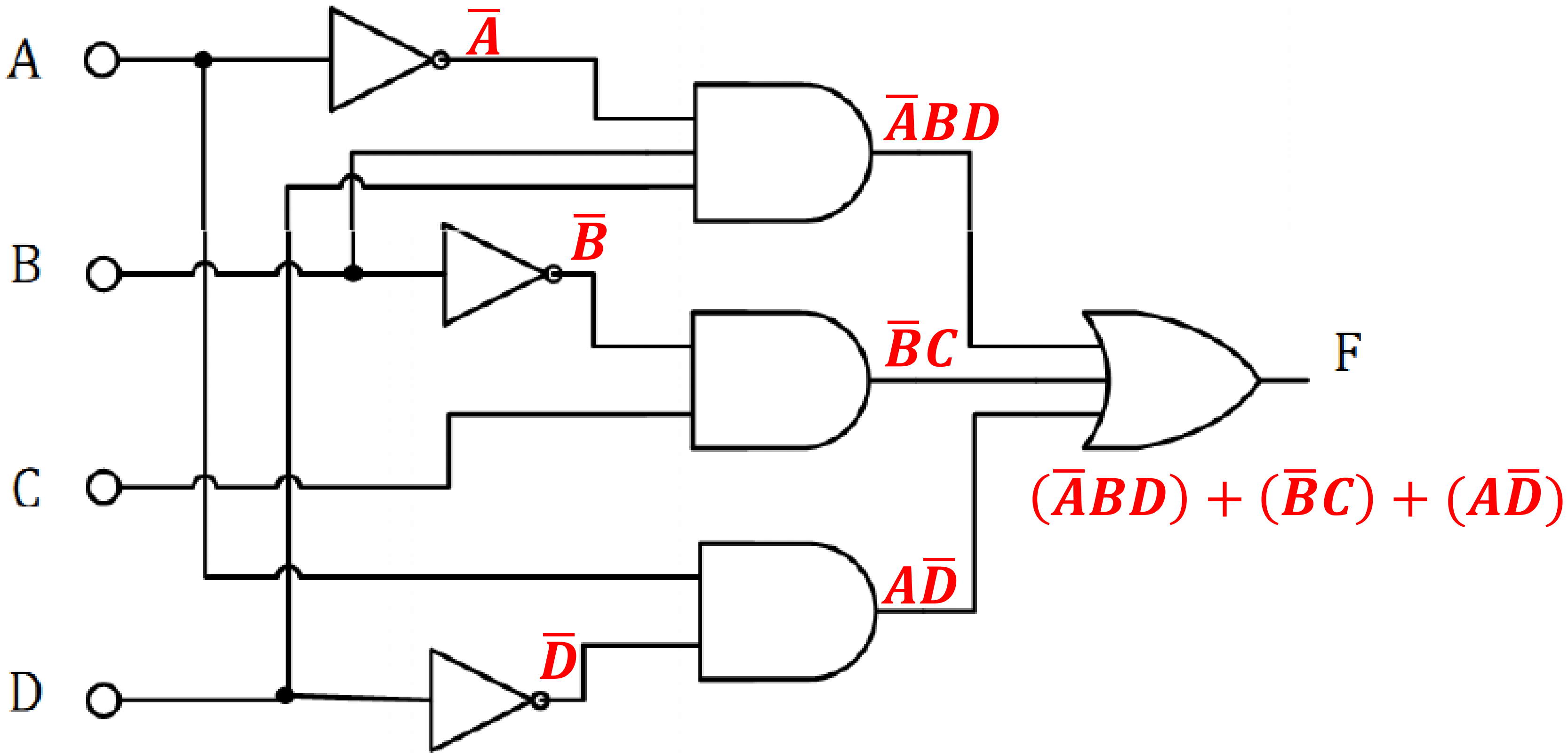
HOMEWORK

Write the algebraic expression for the following circuit.



LOGIC GATES
HOMEWORK

Write the algebraic expression for the following circuit.



BOOLEAN THEOREM

BOOLEAN THEOREM

BOOLEAN ALGEBRA

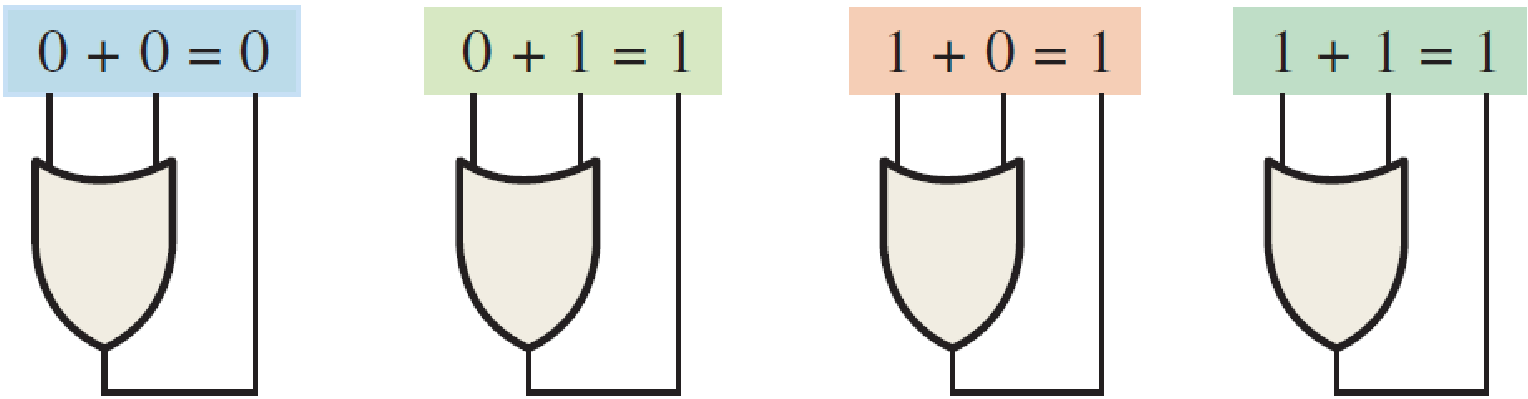
- Boolean algebra is the mathematics of digital systems.
- It is important in digital circuit analysis.
- Three terms that are used in Boolean algebra:

Variable	Complement	Literal
A symbol (letter) used to represent logical quantity. Example: F, X, Y, Z and etc. Any variable that can have **0** and **1** value.	The **inverse** of variable. Example: Complement of $A = \bar{A} = A'$	Variable or complement of variable.

BOOLEAN THEOREM

BOOLEAN ALGEBRA (ADDITION)

- In Boolean algebra, a **sum term** is a sum of literals.
- In logic circuits, a **sum term** is produced by an **OR operation** with no AND operation involved.
- Example: $A + B$, $A + \bar{B}$, $A + B + \bar{C}$

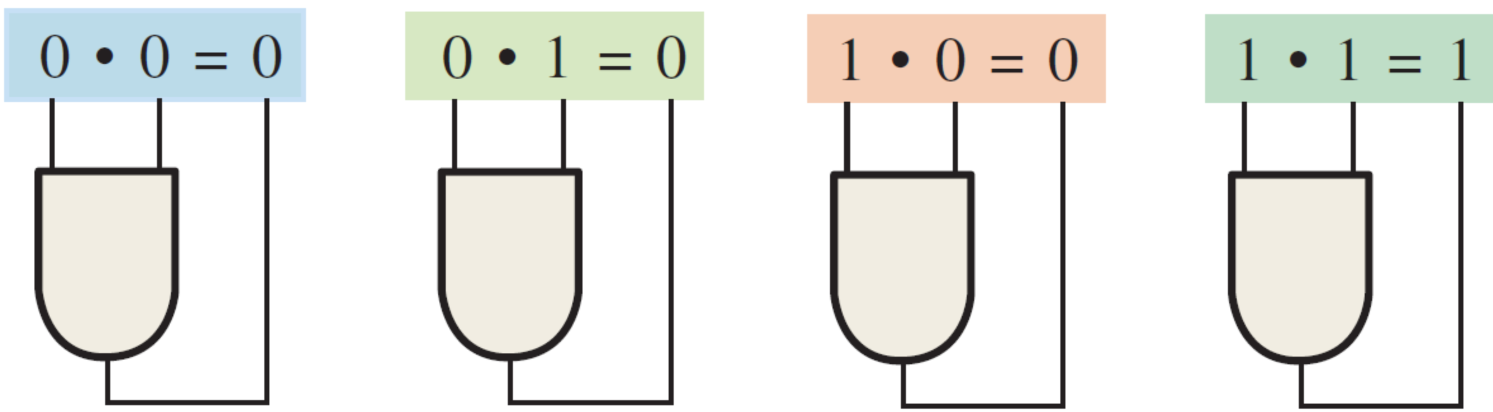


- A sum term equal to **1** when **one or more** of the literals are 1.
- A sum term equal to **0** only if **each** of the literals is 0.

BOOLEAN THEOREM

BOOLEAN ALGEBRA (MULTIPLICATION)

- In Boolean algebra, a **product term** is a product of literals.
- In logic circuits, a **product term** is produced by an **AND operation** with no OR operation involved.
- Example: $AB,\ A\bar{B},\ AB + \bar{C}$



- A product term equal to **1** only if **each** of the literals is 1.
- A product term equal to **0** when **one or more** of the literals are 0.

LAW AND RULES OF BOOLEAN ALGEBRA

LAW AND RULES OF BOOLEAN ALGEBRA

LAWS OF BOOLEAN ALGEBRA

There are **three** categories basic laws of Boolean algebra:

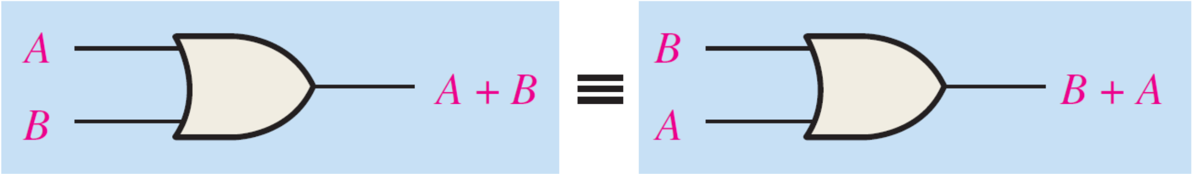
Commutative Laws	Associative Laws	Distributive Laws
For addition and multiplication	For addition and multiplication	Same as in ordinary algebra

LAW AND RULES OF BOOLEAN ALGEBRA

COMMUTATIVE LAWS

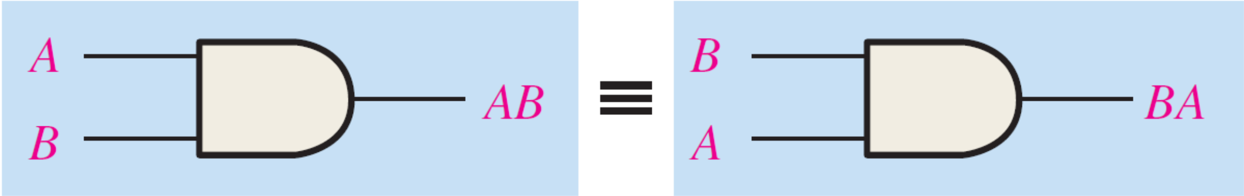
The commutative law for addition: The order variable are ORed make no different.

$$A + B = B + A$$



The commutative law for multiplication: The order variable are ANDed make no different.

$$A \cdot B = B \cdot A$$

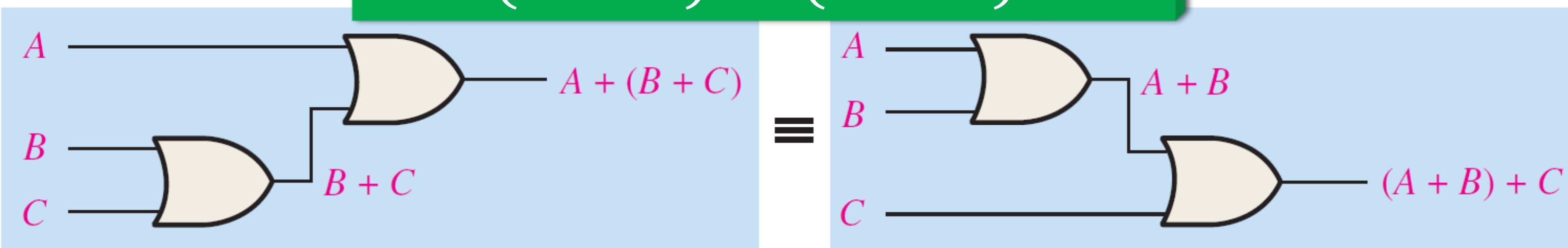


LAW AND RULES OF BOOLEAN ALGEBRA

ASSOCIATIVE LAWS

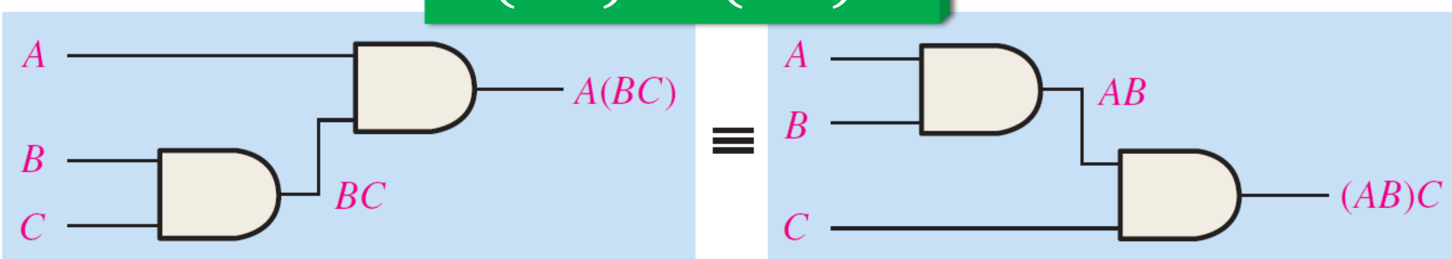
The associative law for addition: When ORing more than two variables, result are same regardless the grouping of variable.

$$A + (B + C) = (A + B) + C$$



The commutative law for multiplication: When ANDing more than two variables, result are same regardless the grouping of variable.

$$A(BC) = (AB)C$$

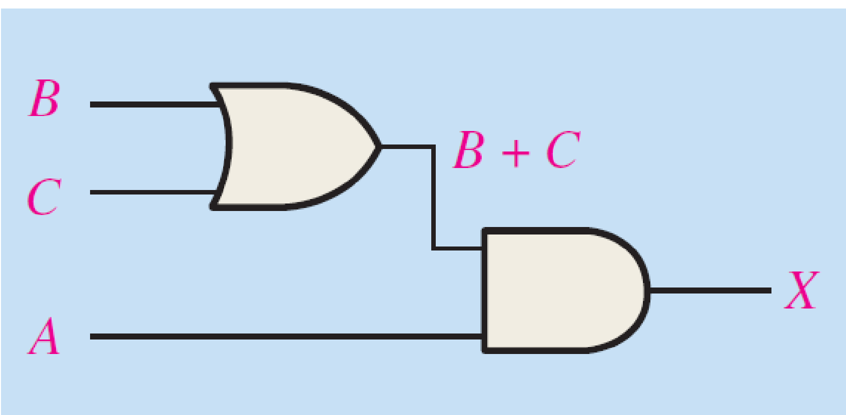


LAW AND RULES OF BOOLEAN ALGEBRA

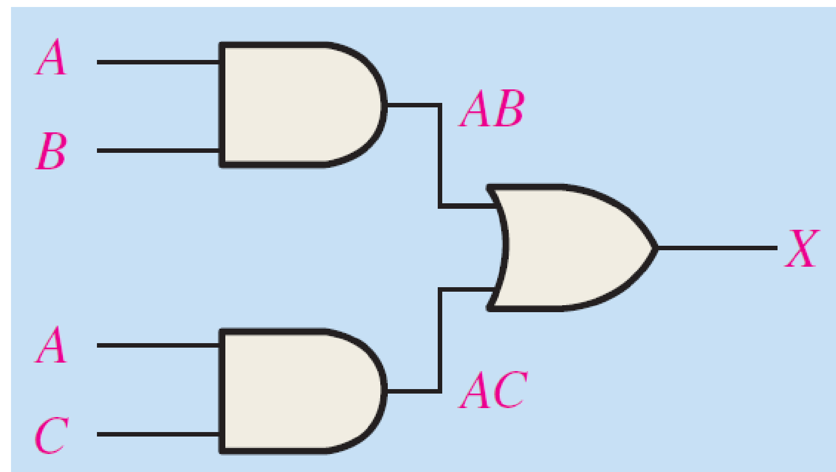
DISTRIBUTIVE LAWS

Expanding an expression by multiplying term by term

$$A\,(B + C) = AB + AC$$



≡



$X = A(B + C)$

$X = AB + AC$

LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

- List of Basic 12 rules that are useful in **manipulating and simplifying** Boolean expression.

1. $A + 0 = A$	7. $A.A = A$
2. $A + 1 = 1$	8. $A \cdot \bar{A} = 0$
3. $A \cdot 0 = 0$	9. $\bar{\bar{A}} = A$
4. $A \cdot 1 = 1$	10. $A + AB = A$
5. $A + A = A$	11. $A + \bar{A}B = A + B$
6. $A + \bar{A} = 1$	12. $(A + B)(A + C) = A + BC$

- Rule 1 to 9 can be viewed in terms of their application to logic gate.
- Rule 10 to 12 is derived in terms of simpler rules and laws previously discussed.

LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

Rule 1: $A + 0 = A$

$$X = A + 0 = A$$

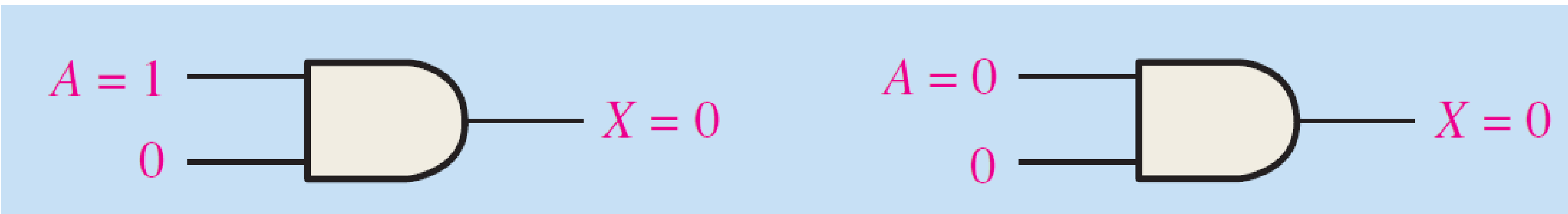
Rule 2: $A + 1 = 1$

$$X = A + 1 = 1$$

LAW AND RULES OF BOOLEAN ALGEBRA

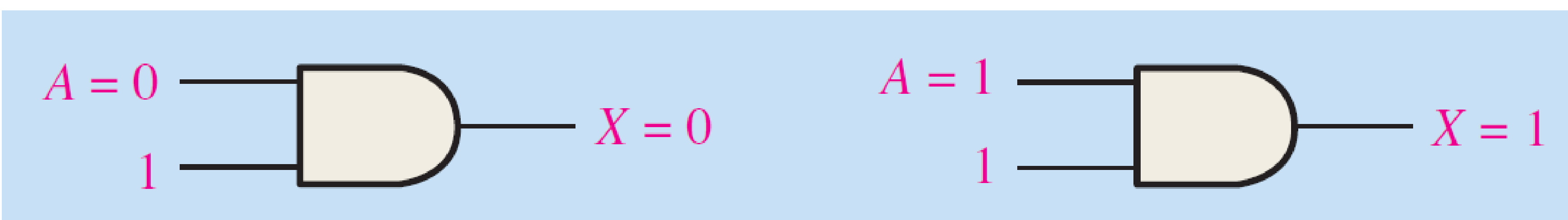
BASIC 12 RULES

Rule 3: $A \cdot 0 = 0$



$$X = A \cdot 0 = 0$$

Rule 4: $A \cdot 1 = A$

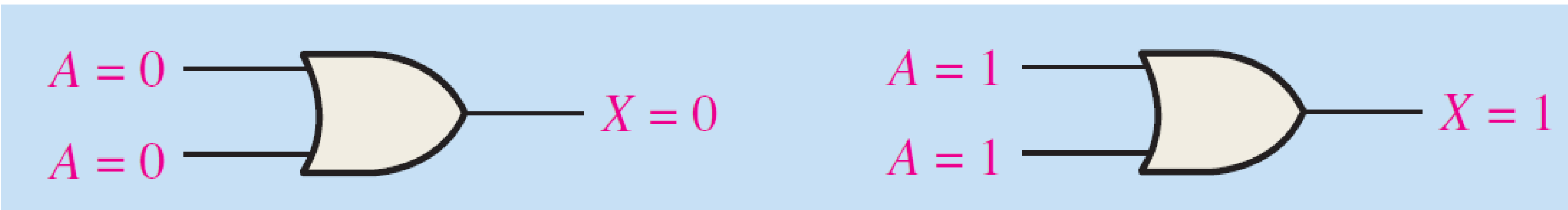


$$X = A \cdot 1 = A$$

LAW AND RULES OF BOOLEAN ALGEBRA

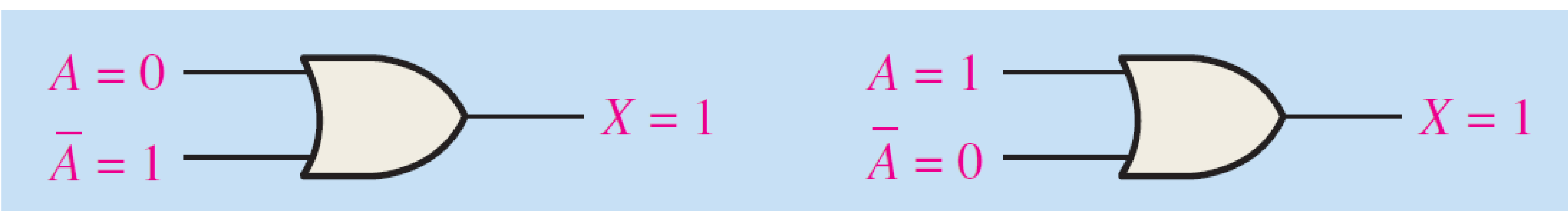
BASIC 12 RULES

Rule 5: $A + A = A$



$$X = A + A = A$$

Rule 6: $A + \overline{A} = 1$

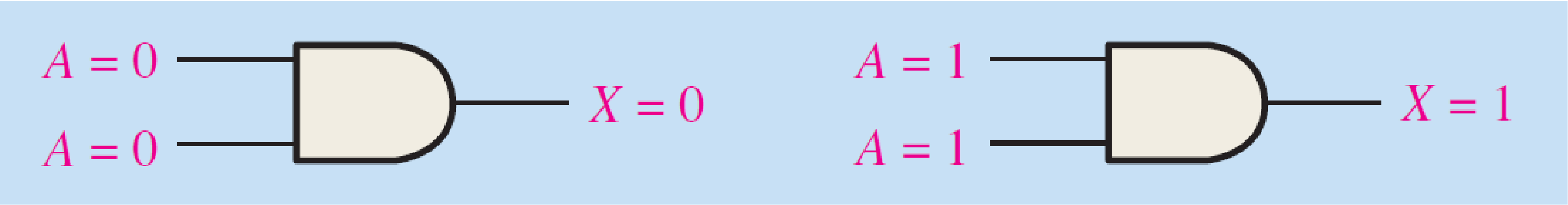


$$X = A + \overline{A} = 1$$

LAW AND RULES OF BOOLEAN ALGEBRA

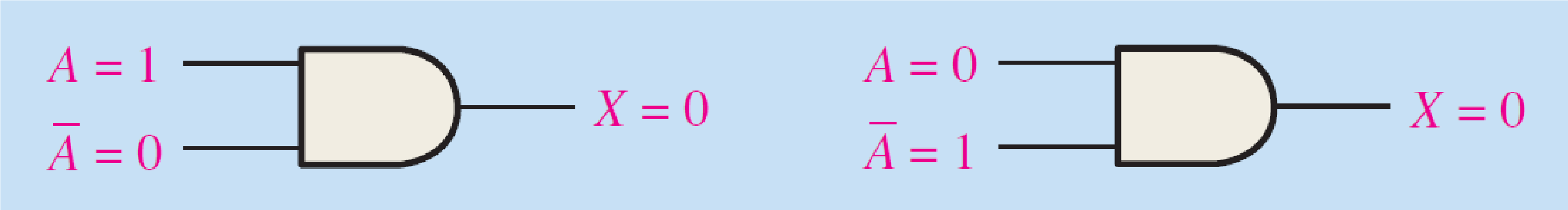
BASIC 12 RULES

Rule 7: $A \cdot A = A$



$$X = A \bullet A = A$$

Rule 8: $A \cdot \overline{A} = 0$



$$X = A \bullet \overline{A} = 0$$

LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

Rule 9: $\overline{\overline{A}} = A$

$$\overline{\overline{A}} = A$$

LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

Rule 10: $\boldsymbol{A + AB = A}$

$$A + AB = A(1 + B)$$
$$= A(1)$$
$$= \boldsymbol{A}$$

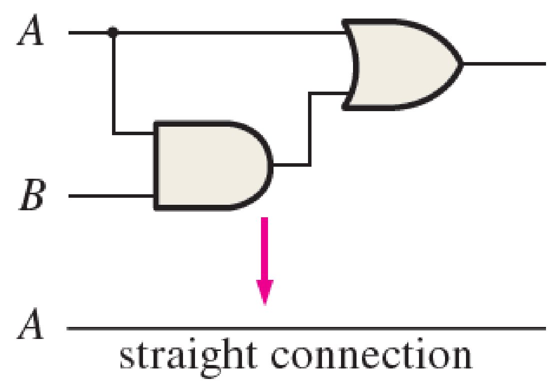
Factoring (Distributive law)

Rule 2: $1 + B = 1$

Rule 4: $A \cdot 1 = A$

A	B	AB	$A + AB$
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

equal



LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

Rule 11: $\boldsymbol{A + \bar{A}B = A + B}$

$$
\begin{aligned}
A + \bar{A}B &= A + AB + \bar{A}B \\
&= A + B(A + \bar{A}) \\
&= A + B(1) \\
&= \boldsymbol{A + B}
\end{aligned}
$$

Rule 10: $A + AB$

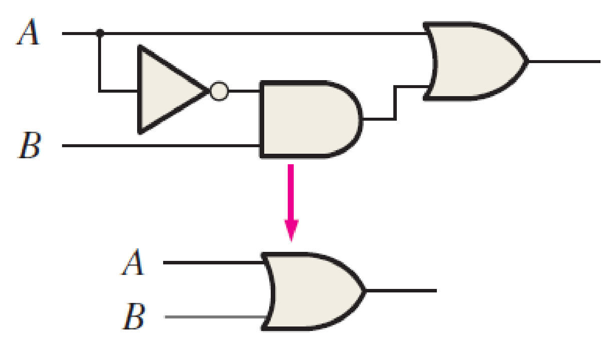
Factoring (Distributive law)

Rule 6: $A + \bar{A} = 1$

Rule 4: $\mathrm{B} \cdot 1 = B$

A	B	$\bar{A}B$	$A + \bar{A}B$	$A + B$
0	0	0	0	0
0	1	1	1	1
1	0	0	1	1
1	1	0	1	1

equal ($A + \bar{A}B$ and $A + B$ columns)



LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

Rule 12: $(\boldsymbol{A}+\boldsymbol{B})(\boldsymbol{A}+\boldsymbol{C})=\boldsymbol{A}+\boldsymbol{BC}$

$(A+B)(A+C)$	$=AA+AC+AB+BC$	Distributive law
	$=A+AC+AB+BC$	Rule 7: $A\cdot A=A$
	$=A(1+C)+AB+BC$	Factoring (Distributive law)
	$=A(1)+AB+BC$	Rule 2: $C+1=1$
	$=A+AB+BC$	Rule 4: $A\cdot 1=A$
	$=A(1+B)+BC$	Factoring (Distributive law)
	$=A(1)+BC$	Rule 2: $\mathrm{B}+1=1$
	$=\boldsymbol{A}+\boldsymbol{BC}$	Rule 4: $A\cdot 1=A$

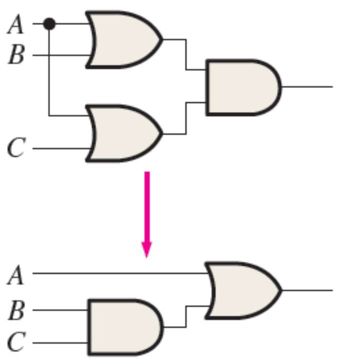
LAW AND RULES OF BOOLEAN ALGEBRA

BASIC 12 RULES

Rule 12: $(A + B)(A + C) = A + BC$

A	B	C	$A+B$	$A+C$	$(A+B)(A+C)$	BC	$A+BC$
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	1	1	1	1
1	0	0	1	1	1	0	1
1	0	1	1	1	1	0	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

equal ($(A+B)(A+C)$ and $A+BC$ columns)



LAW AND RULES OF BOOLEAN ALGEBRA

DEMORGAN'S THEOREM

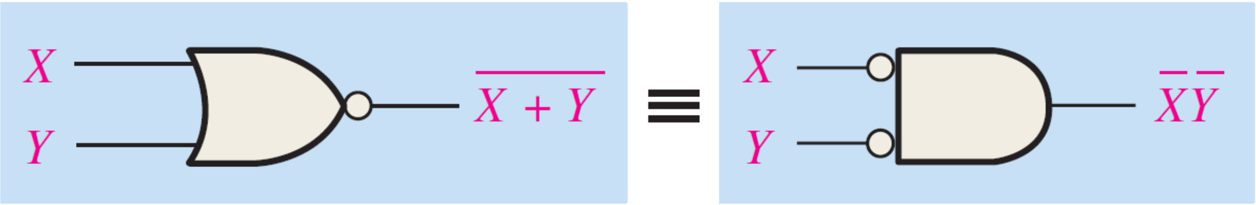
DeMorgan proposed two theorems that are important part of Boolean algebra.

1st Theorem

$$\overline{X+Y} = \bar{X} \cdot \bar{Y}$$

Truth Table

Input		Output	
X	Y	**X + Y**	**X . Y**
0	0	**1**	**1**
0	1	**0**	**0**
1	0	**0**	**0**
1	1	**0**	**0**



NOR Negative-AND

LAW AND RULES OF BOOLEAN ALGEBRA

DEMORGAN'S THEOREM

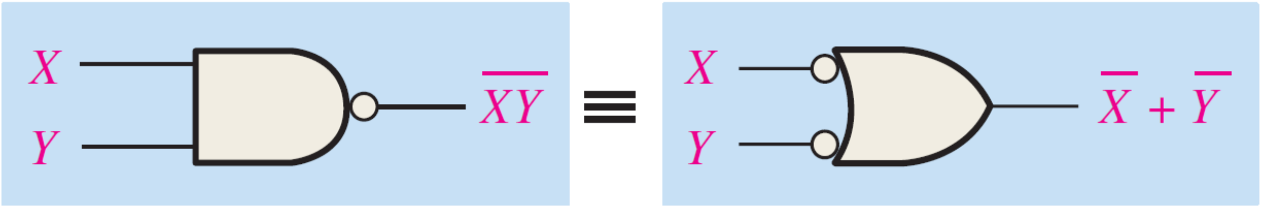
DeMorgan proposed two theorems that are important part of Boolean algebra.

2nd Theorem

$$\overline{X.Y} = \bar{X} + \bar{Y}$$

Truth Table

Input		Output	
X	Y	**X . Y**	**X + Y**
0	0	**1**	**1**
0	1	**1**	**1**
1	0	**1**	**1**
1	1	**0**	**0**



NAND Negative-OR

LAW AND RULES OF BOOLEAN ALGEBRA

HOMEWORK

Apply DeMorgan's Theorem to each of following expressions:

1. $F = \overline{(A + B + C)D}$
2. $F = \overline{ABC + DEF}$
3. $F = \overline{A\bar{B} + \bar{C}D + EF}$

BOOLEAN EXPRESSION SIMPLIFICATION

BOOLEAN EXPRESSION SIMPLIFICATION

BOOLEAN SIMPLIFICATION

Simplify Boolean expression below:

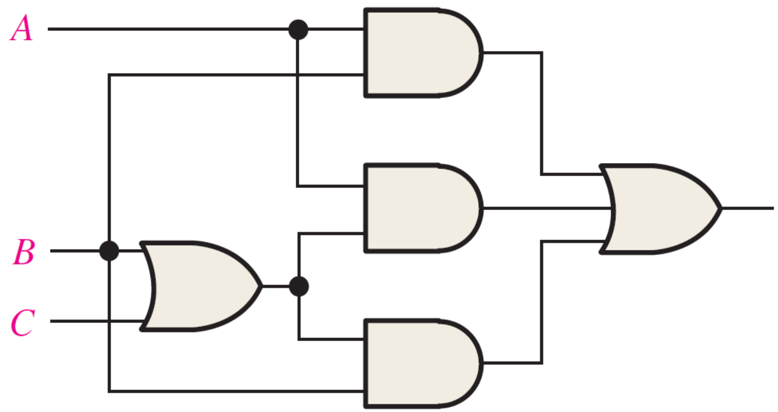
$\boldsymbol{AB + A(B + C) + B(B + C)}$

$= AB + AB + AC + BB + BC$	$Distributive\ Law$
$= AB + AC + B + BC$	$Rule\ 5: AB + AB = AB,$ $Rule\ 7: BB = B$
$= AB + AC + B(1 + C)$	$Distributive\ Law$
	$Rule\ 2: C + 1 = 1$
$= AB + AC + B$	$Distributive\ Law$
$= B(A + 1) + AC$	$Rule\ 2: A + 1 = 1$
$= B + AC$	

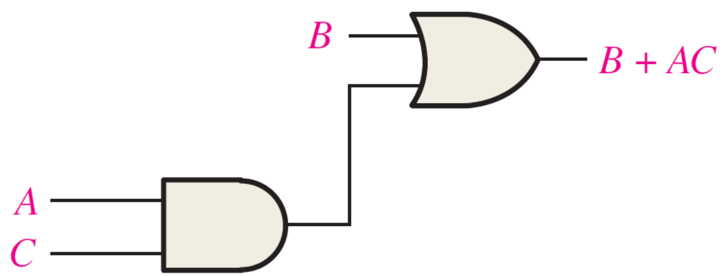
BOOLEAN EXPRESSION SIMPLIFICATION

BOOLEAN SIMPLIFICATION

Original Expression: $\boldsymbol{AB + A(B + C) + B(B + C)}$



Simplified Expression: $\boldsymbol{B + AC}$

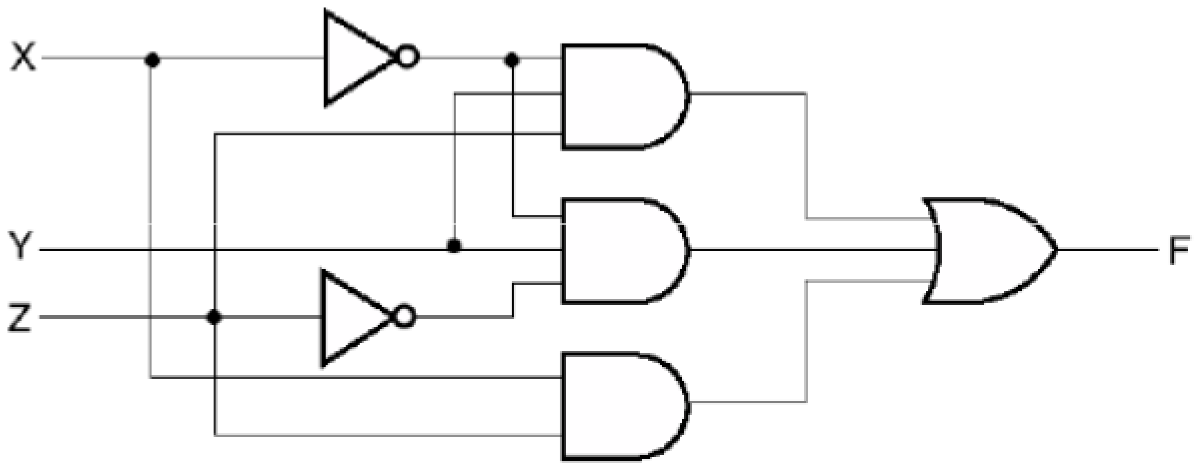


Faster, compact design and lower cost

BOOLEAN EXPRESSION SIMPLIFICATION

ASSESEMENT 3

Find the Boolean expression for given logic circuit. Then simplify the Boolean expression.



$$F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$$

BOOLEAN EXPRESSION SIMPLIFICATION

ASSESSMENT 3 (SOLUTION)

Do the simplification of Boolean expression as follows:

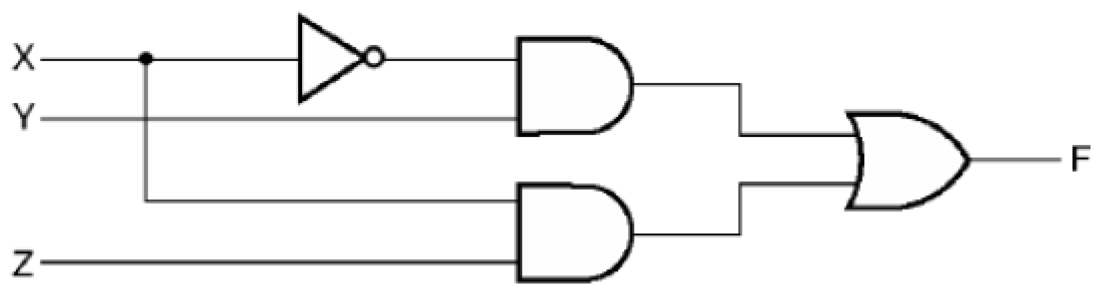
$$F = \bar{X}YZ + \bar{X}Y\bar{Z} + XZ$$

$$F = \bar{X}Y(Z + \bar{Z}) + XZ \qquad \textit{Distributive Law}$$

$$F = \bar{X}Y(1) + XZ \qquad \textit{Rule } 6: Z + \bar{Z} = 1$$

$$F = \bar{X}Y + XZ \qquad \textit{Rule } 4: \bar{X}Y \cdot 1 = \bar{X}Y$$

Simplified Boolean expression, $\boldsymbol{F = \bar{X}Y + XZ}$



BOOLEAN EXPRESSION SIMPLIFICATION

HOMEWORK

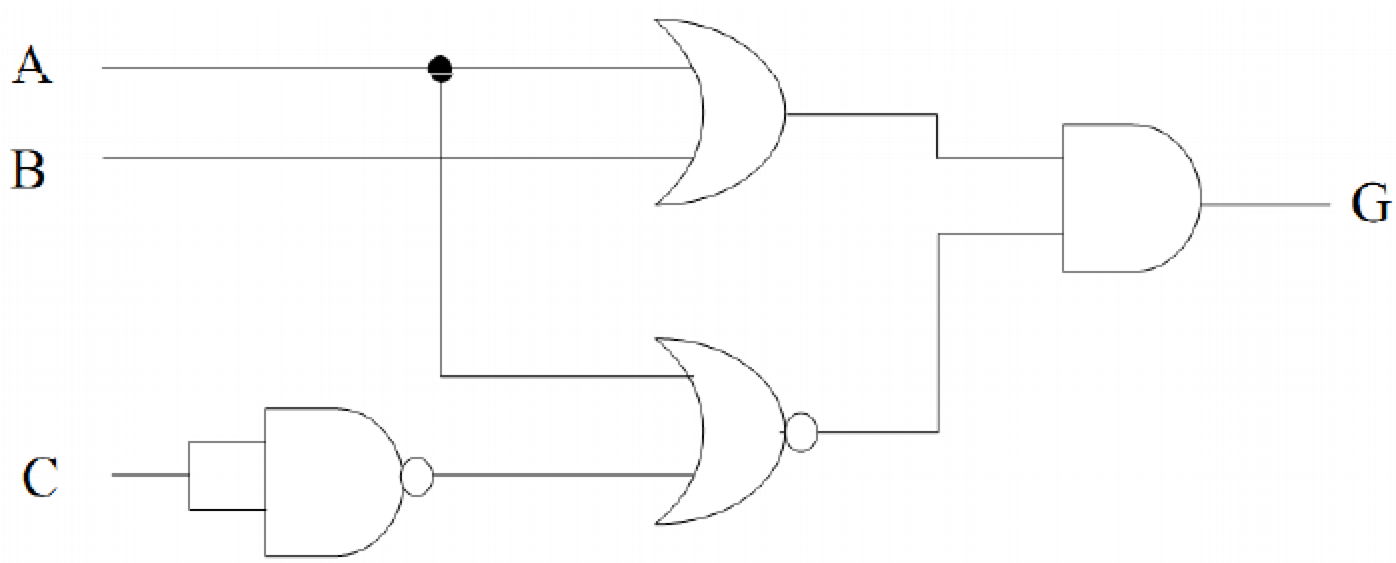
Simplify the following using Boolean algebra techniques and draw the simplified logic circuit:

1. $A\bar{B} + AB$
2. $AB + A(B + C) + B(B + C)$
3. $[A\bar{B}(C + BD) + \bar{A}\bar{B}]C$
4. $\overline{AB + AC} + \bar{A}\bar{B}C$

BOOLEAN EXPRESSION SIMPLIFICATION

HOMEWORK

1. Write the algebraic expression for the following circuit.
2. Produce a truth table for the circuit.
3. Design a simpler circuit having the same output.



BOOLEAN EXPRESSION SIMPLIFICATION

HOMEWORK

1. Derive a Boolean expression from the truth table.
2. Simplify the Boolean expression using Boolean algebra.
3. Draw a logic circuit for the simplified Boolean expression using only OR and AND gates.
4. Draw the logic circuit using only 2-input NAND gates.

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