Lecture -1
Logic Gates & Boolean Algebra

Prepared By: Asif Mahfuz

Logic Gates

- Logic Gates are the basic building blocks of any digital system.
- A logic gate can have one or more than one input but only one output.
- The relationship between the input/s and the output is based on a certain logic.
- The gates are named based on the logic.

Basic Logic Gates

- NOT gate
- AND gate
- OR gate

Universal Logic Gates

- NAND gate
- NOR gate

Exclusive Logic Gates

- XOR gate
- XNOR gate

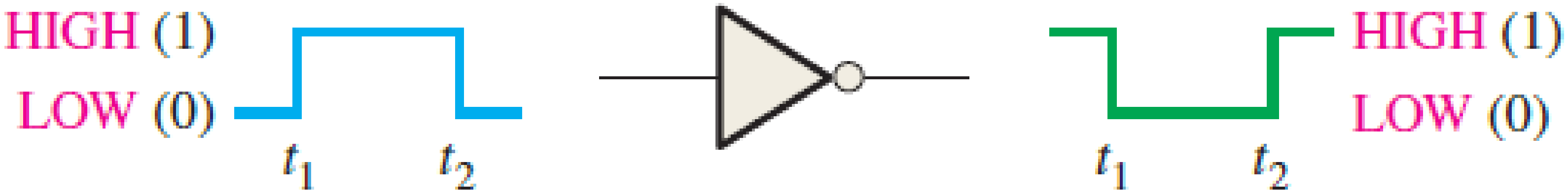
NOT Gate (Inverter)

- The output an inverter (NOT gate) is the opposite of its input.

Inverter truth table.

Input	Output
LOW (0)	HIGH (1)
HIGH (1)	LOW (0)

$X = \overline{X}$

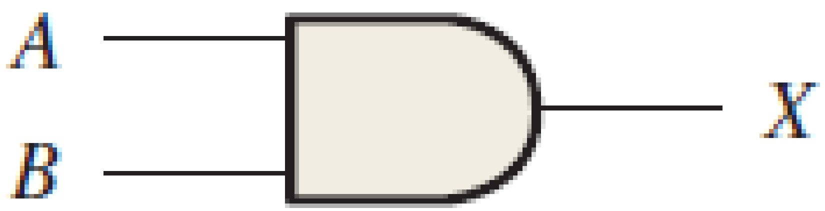


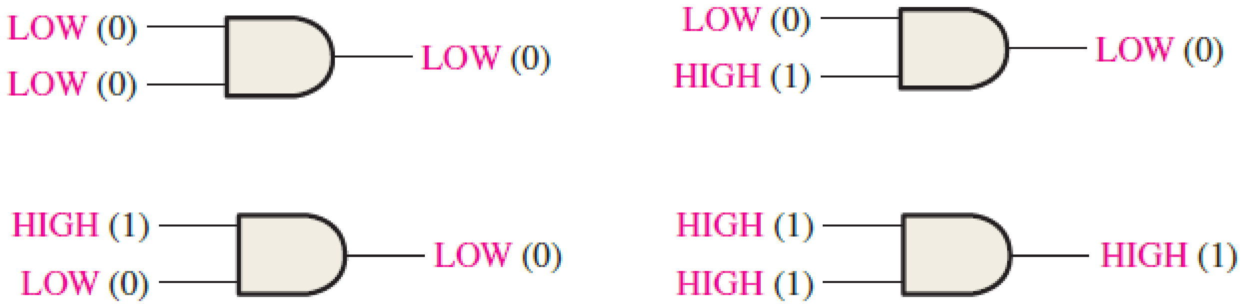
Input pulse

Output pulse

AND Gate

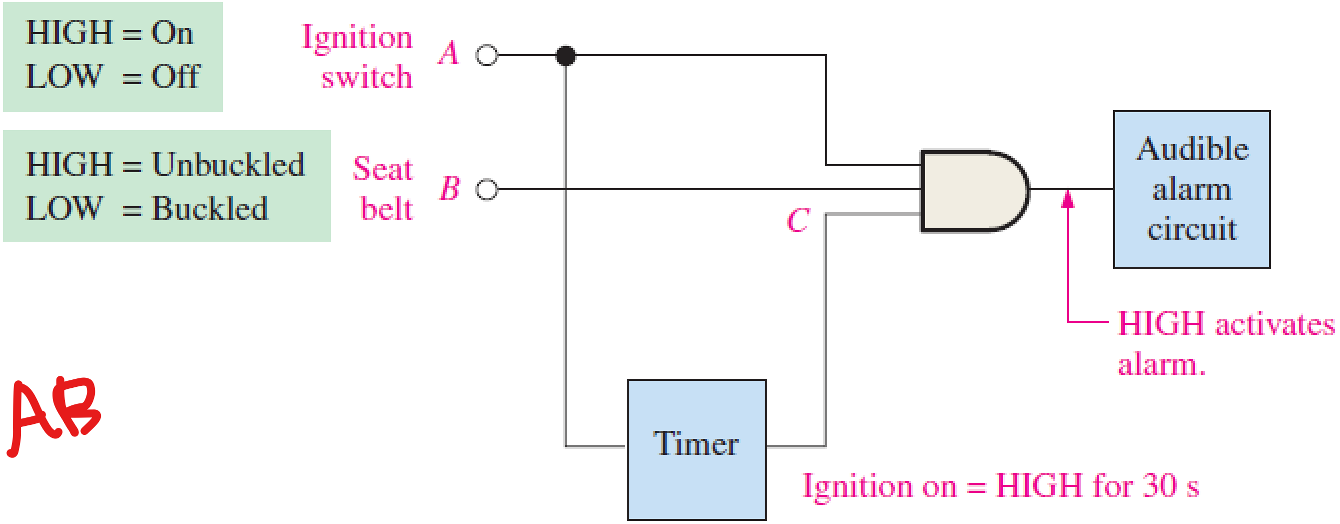
- The output an AND gate is HIGH only when both the inputs are HIGH.





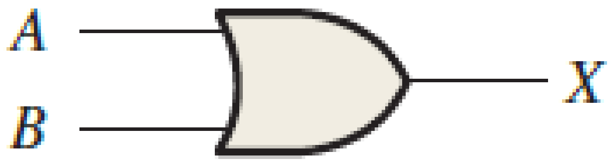
Inputs		Output
A	*B*	*X*
0	0	0
0	1	0
1	0	0
1	1	1

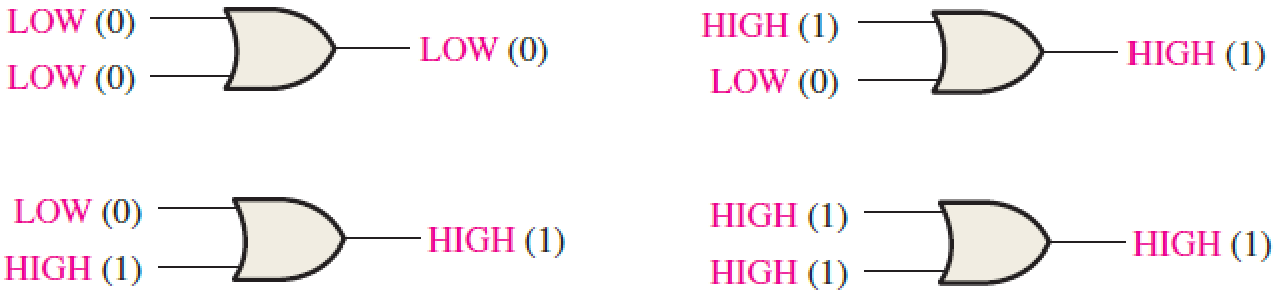
$x = AB$



OR Gate

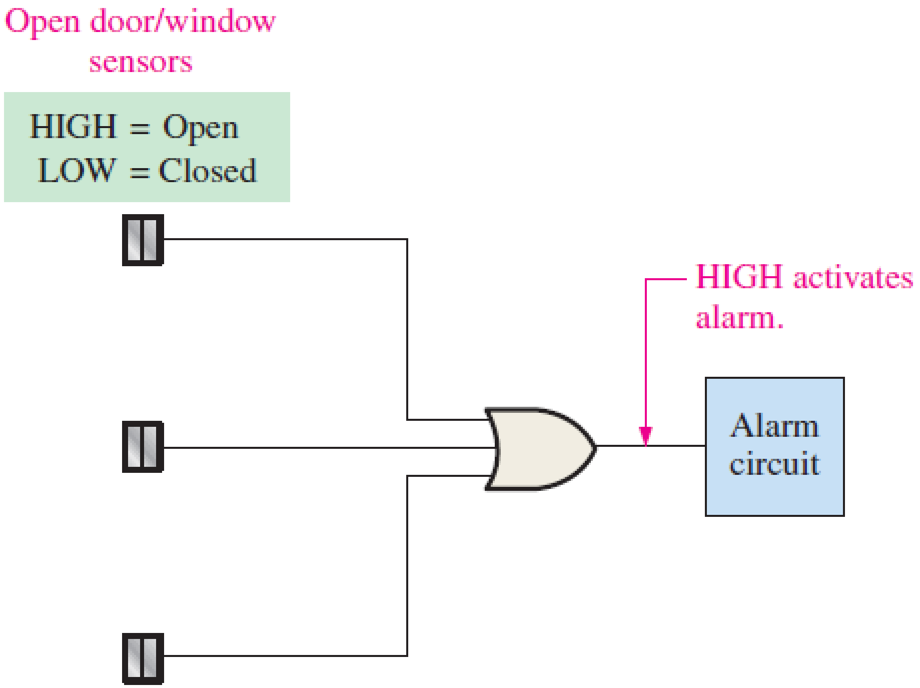
- The output an OR gate is HIGH when anyone or both the inputs are HIGH.





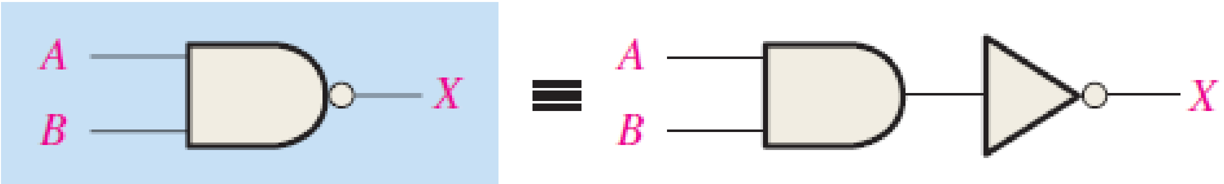
Inputs		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

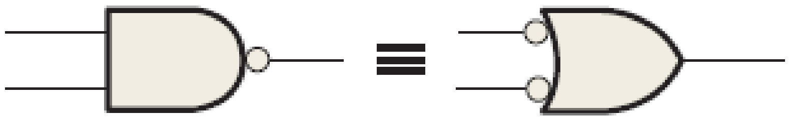
$X = A + B$



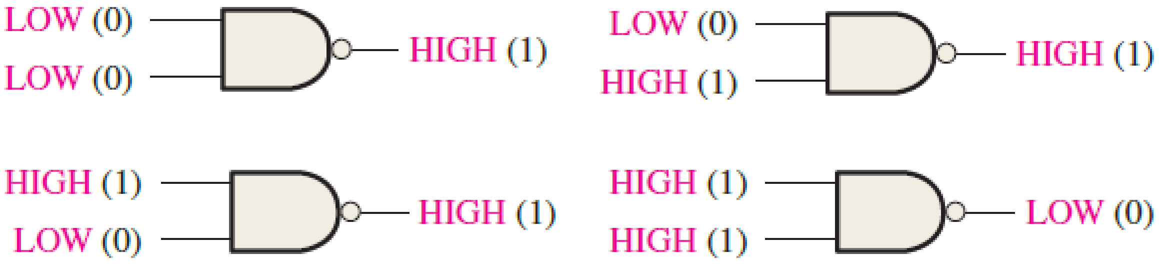
NAND Gate

- The output of a NAND gate is HIGH whenever one or more inputs are LOW.



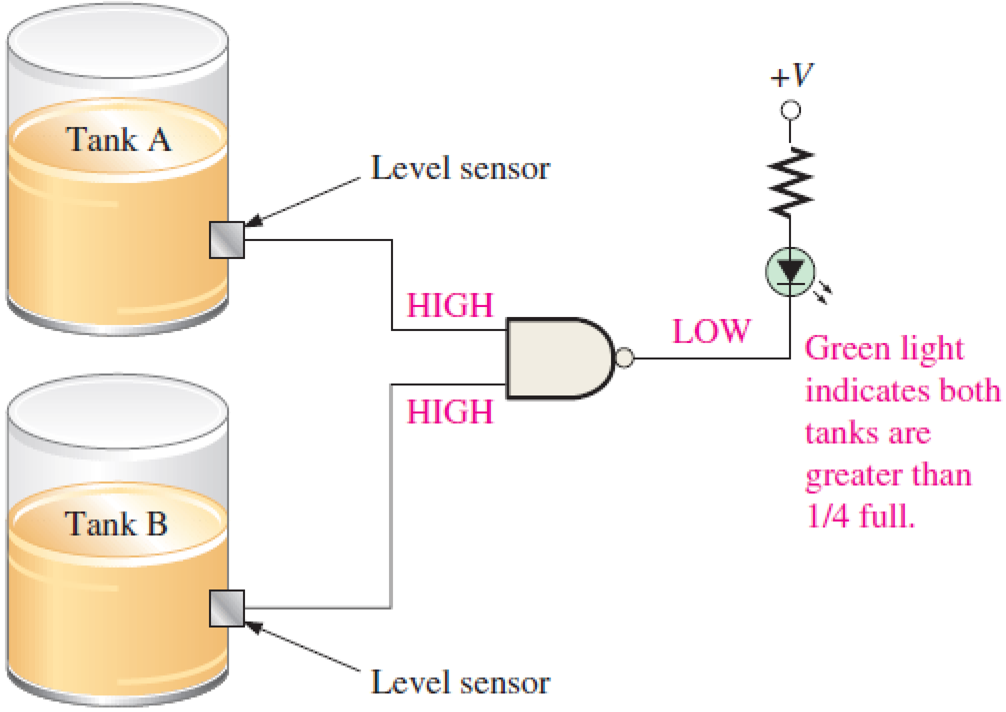


NAND Negative-OR



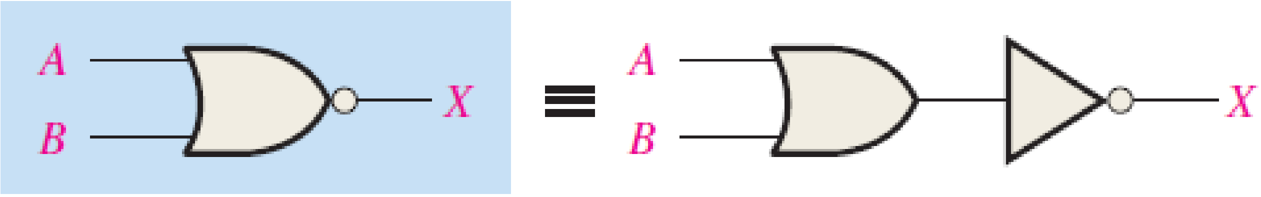
Inputs		Output
A	*B*	*X*
0	0	1
0	1	1
1	0	1
1	1	0

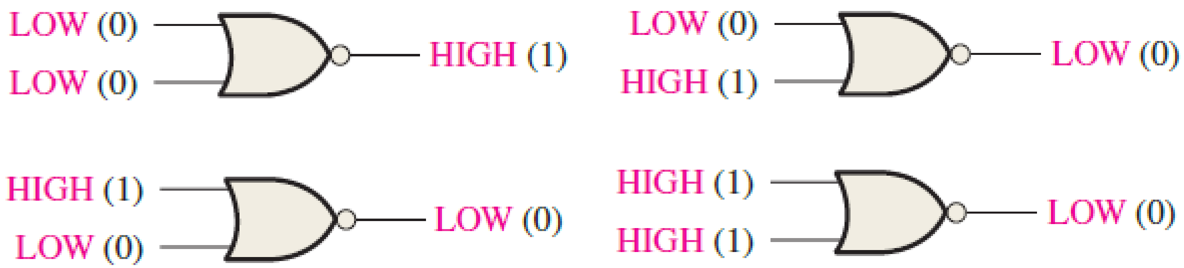
$x = \overline{AB}$

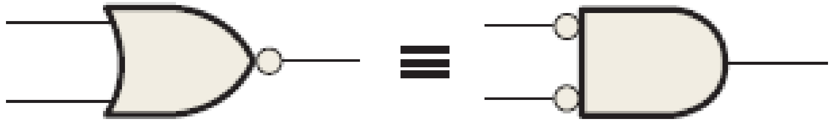


NOR Gate

- The output of a NOR gate is LOW whenever one or more inputs are HIGH.



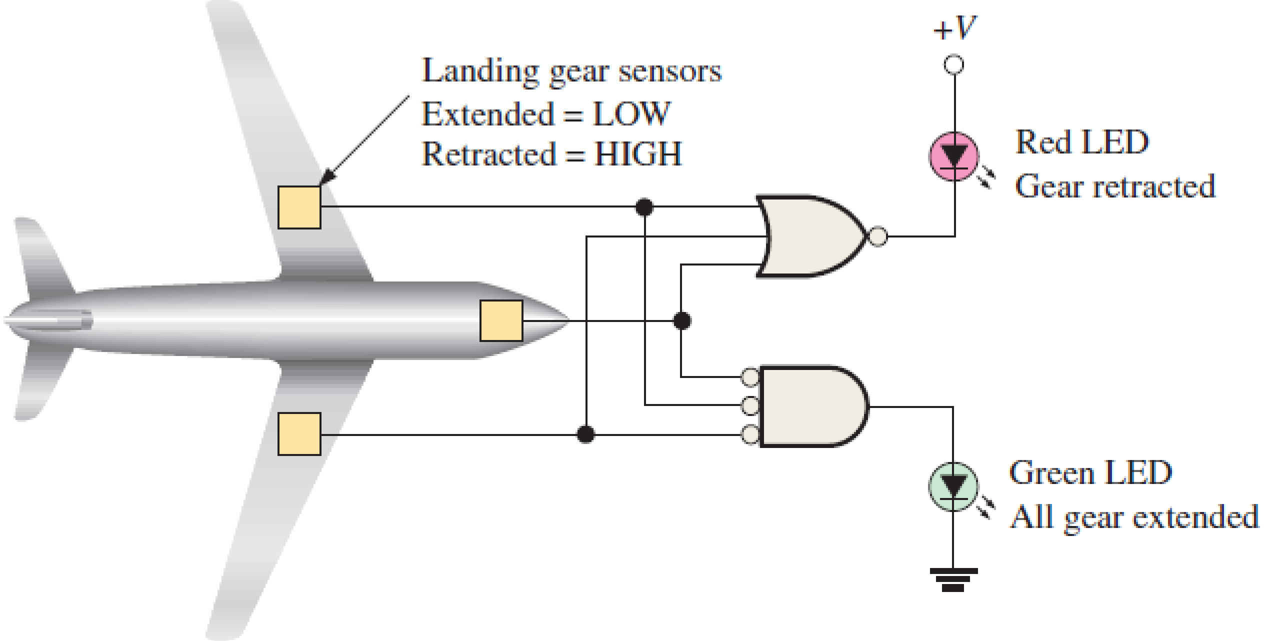




NOR Negative-AND

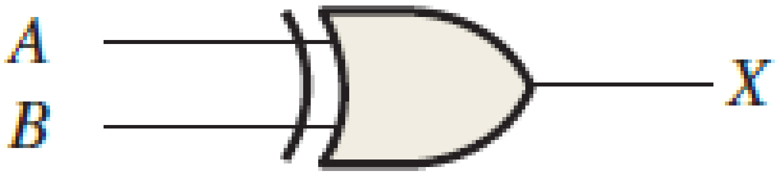
Inputs		Output
A	*B*	*X*
0	0	1
0	1	0
1	0	0
1	1	0

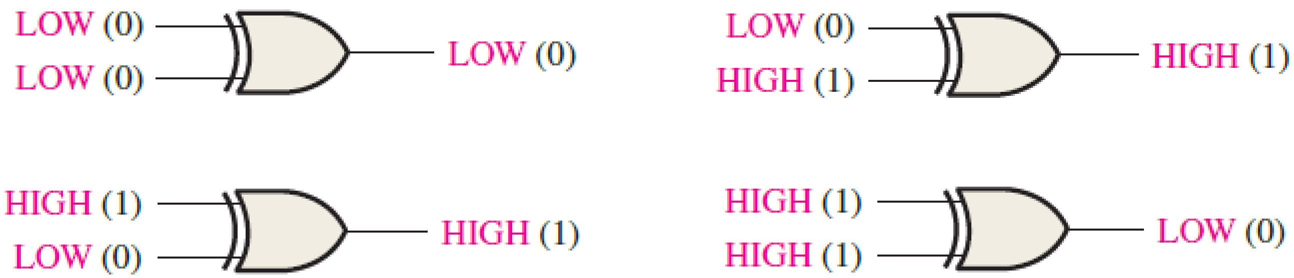
$X = \overline{A+B}$



XOR Gate

- The output of a XOR gate is HIGH whenever the two inputs are different.



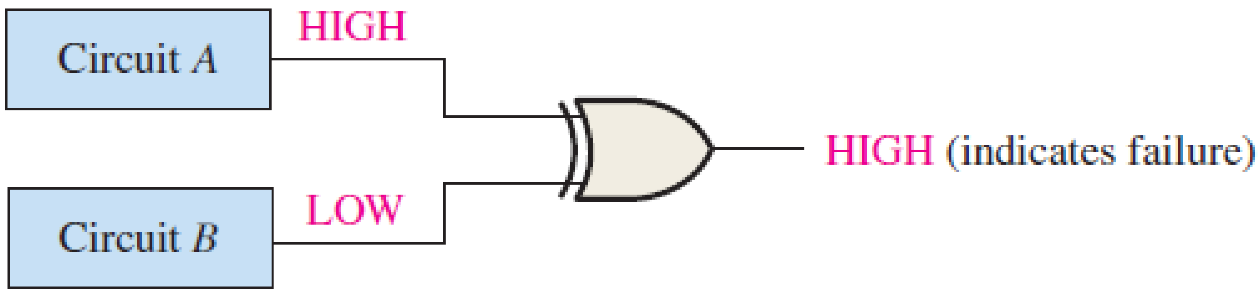


Truth table for an exclusive-OR gate.

Inputs		Output
A	*B*	*X*
0	0	0
0	1	1
1	0	1
1	1	0

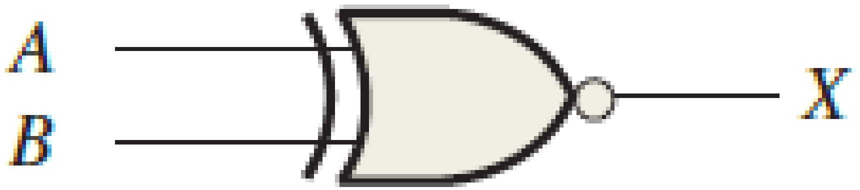
$X = A \oplus B$

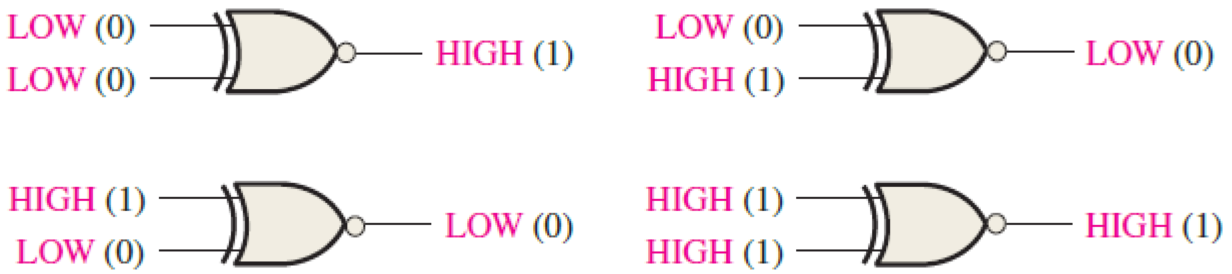
Two circuits are supposed to work parallelly in a process. If any of the circuit fails an indicator is activated.



XNOR Gate

- The output of a XNOR gate is gate is HIGH whenever the two inputs are same.



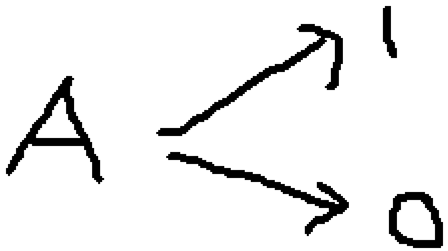


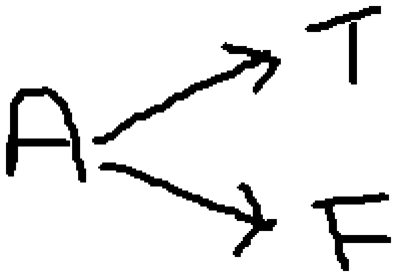
Inputs		Output
A	*B*	*X*
0	0	1
0	1	0
1	0	0
1	1	1

$x = \overline{A \oplus B}$

Boolean Algebra

- Boolean Variable: These are variables which can either take the value 1 or 0.





- Boolean Logic Expression: A Boolean logic expression is an expression constituted of only Boolean variables. The output of a Boolean logic expression is a Boolean value i.e. either True/False.

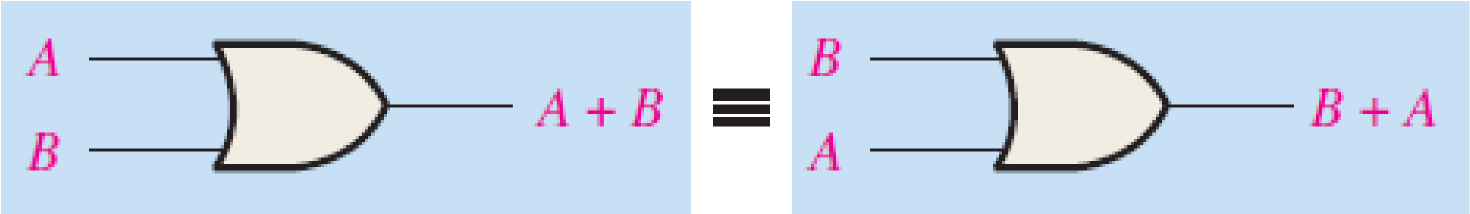
$$AB + A(B + C) + B(B + C)$$

- Boolean Algebra: It is the mathematics of digital logic. Usually Boolean algebra is used to simplify Boolean expressions or Boolean Function.

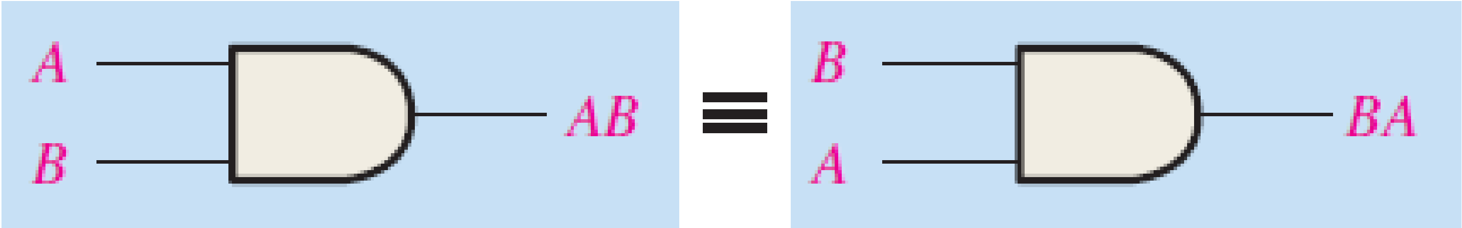
Laws of Boolean Algebra

Commutative Law:

- The commutative law for addition can be written as **A+B=B+A**



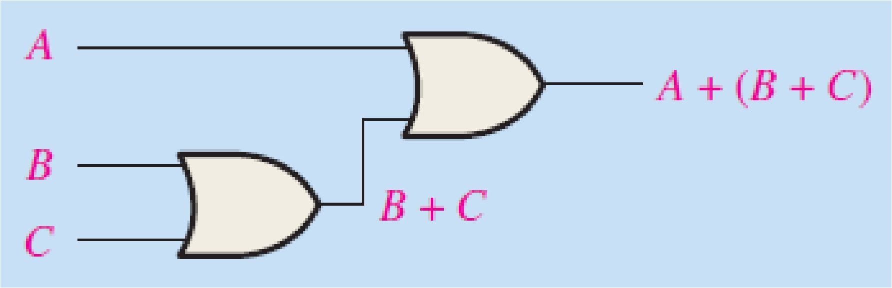
- The commutative law for multiplication can be written as **AB=BA**



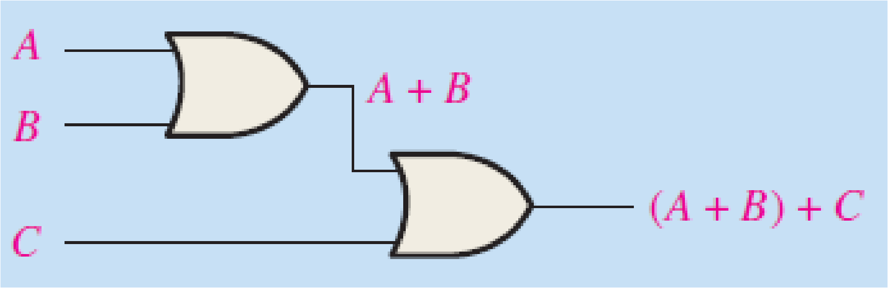
Laws of Boolean Algebra

Associative Law:

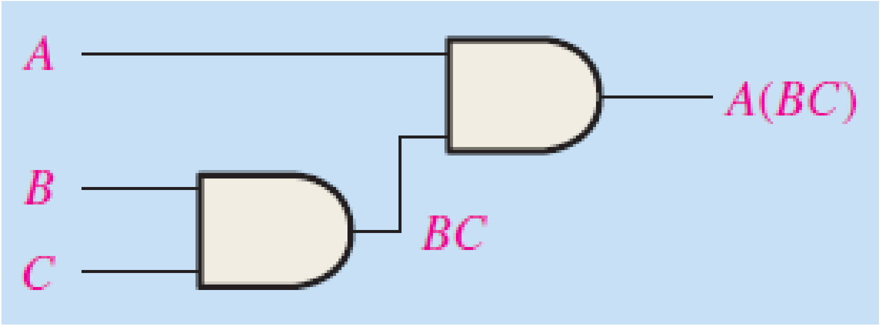
- The associative law of addition for three variables is written as **A+(B+C)=(A+B)+C**



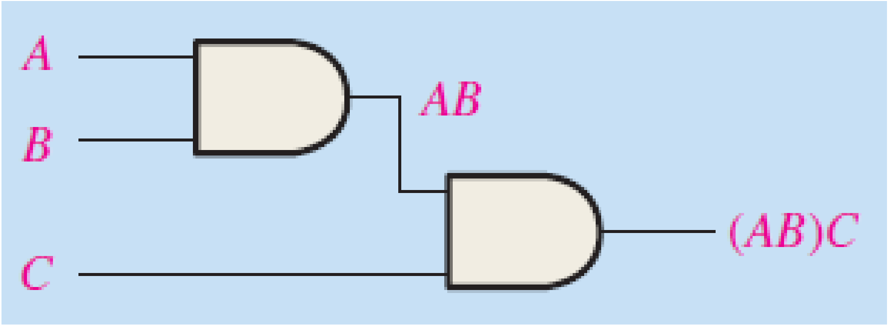
≡



- The associative law of multiplication for three variables is written as **A(BC)=(AB)C**



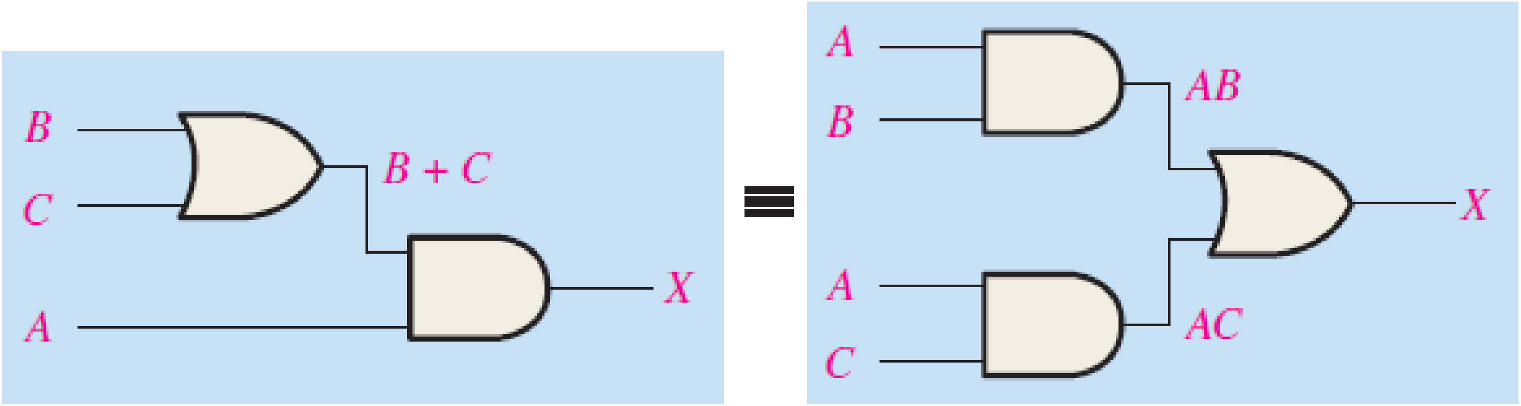
≡



Laws of Boolean Algebra

Distributive Law:

- The distributive law for three variables is written as **A(B+C)=AB+AC**



$X = A(B + C)$ $\quad$ $X = AB + AC$

Rules of Boolean Algebra

Basic rules of Boolean algebra.

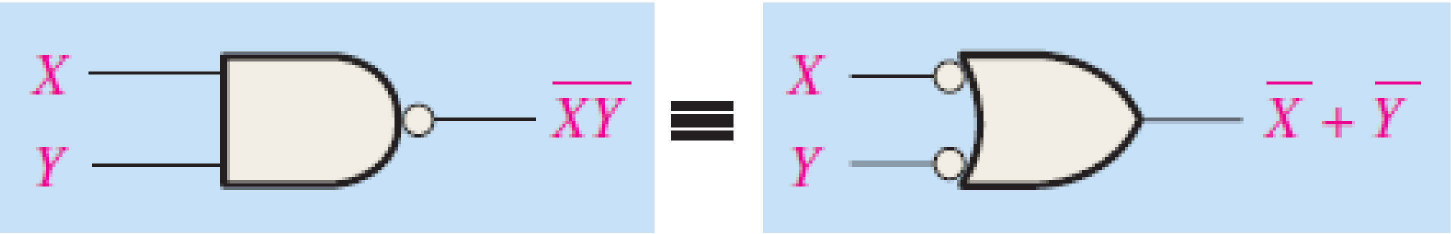
1. $A + 0 = A$	7. $A \cdot A = A$
2. $A + 1 = 1$	8. $A \cdot \overline{A} = 0$
3. $A \cdot 0 = 0$	9. $\overline{\overline{A}} = A$
4. $A \cdot 1 = A$	10. $A + AB = A$
5. $A + A = A$	11. $A + \overline{A}B = A + B$
6. $A + \overline{A} = 1$	12. $(A + B)(A + C) = A + BC$

De Morgan's Theorem

The first theorem is stated as follows:

- The complement of a product of variables is equal to the sum of the complements of complements of the variable.
- The formula of this theorem for two variables is written as:

$$\overline{\mathbf{XY}} = \overline{\mathbf{X}} + \overline{\mathbf{Y}}$$



NAND   Negative-OR

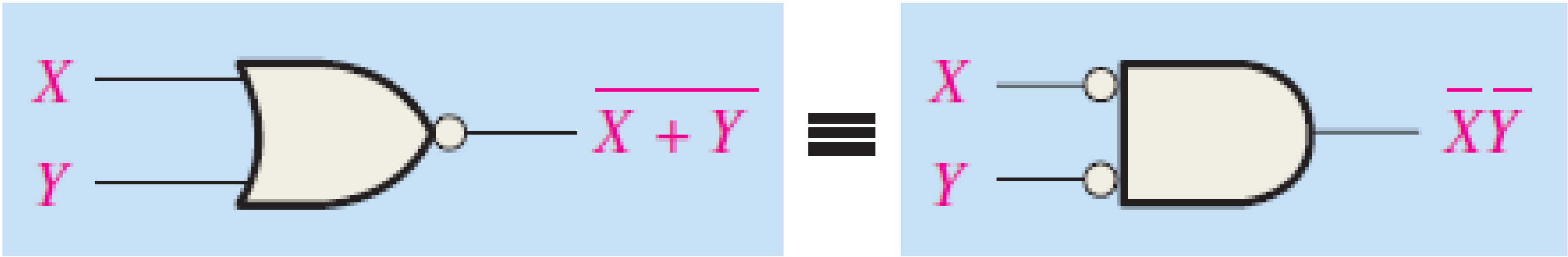
Inputs		Output	
X	Y	$\overline{XY}$	$\overline{X} + \overline{Y}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

De Morgan’s Theorem

The second theorem is stated as follows:

- The complement of a sum of variables is equal to the product of the complements of the variables.
- The formula of this theorem for two variables is written as:

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



NOR   Negative-AND

Inputs		Output	
X	Y	$\overline{X+Y}$	$\overline{X}\,\overline{Y}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

Application of De Morgan's Theorem

Apply DeMorgan's theorems to the expressions $\overline{XYZ}$ and $\overline{X + Y + Z}$.

Solution

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

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

Apply DeMorgan's theorems to the expressions $\overline{WXYZ}$ and $\overline{W + X + Y + Z}$.

Solution

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

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

Apply DeMorgan's theorems to each expression:

(a) $\overline{(\overline{A + B}) + \overline{C}}$

(b) $\overline{(\overline{A} + B) + CD}$

(c) $\overline{(A + B)\overline{C}\,\overline{D} + E + \overline{F}}$

Solution

(a) $\overline{(\overline{A + B}) + \overline{C}} = (\overline{\overline{A + B}})\overline{\overline{C}} = (A + B)C$

(b) $\overline{(\overline{A} + B) + CD} = (\overline{\overline{A} + B})\overline{CD} = (\overline{\overline{A}}\,\overline{B})(\overline{C} + \overline{D}) = A\overline{B}(\overline{C} + \overline{D})$

(c) $\overline{(A + B)\overline{C}\,\overline{D} + E + \overline{F}} = ((\overline{(A + B)\overline{C}\,\overline{D}})(\overline{E + \overline{F}}) = (\overline{A}\,\overline{B} + C + D)\overline{E}F$

Application of De Morgan's Theorem

Apply DeMorgan's theorem to the expression $\overline{\overline{X} + \overline{Y} + \overline{Z}}$.

Apply DeMorgan's theorem to the expression $\overline{\overline{W}\,\overline{X}\,\overline{Y}\,\overline{Z}}$.

Apply DeMorgan's theorems to each of the following expressions:

(a) $\overline{(A + B + C)D}$

(b) $\overline{ABC + DEF}$

(c) $\overline{A\overline{B} + \overline{C}D + EF}$

The Boolean expression for an exclusive-OR gate is $A\overline{B} + \overline{A}B$. With this as a starting point, use DeMorgan's theorems and any other rules or laws that are applicable to develop an expression for the exclusive-NOR gate.

Starting with the expression for a 4-input NAND gate, use DeMorgan's theorems to develop an expression for a 4-input negative-OR gate.

Apply DeMorgan's theorems to the following expressions:

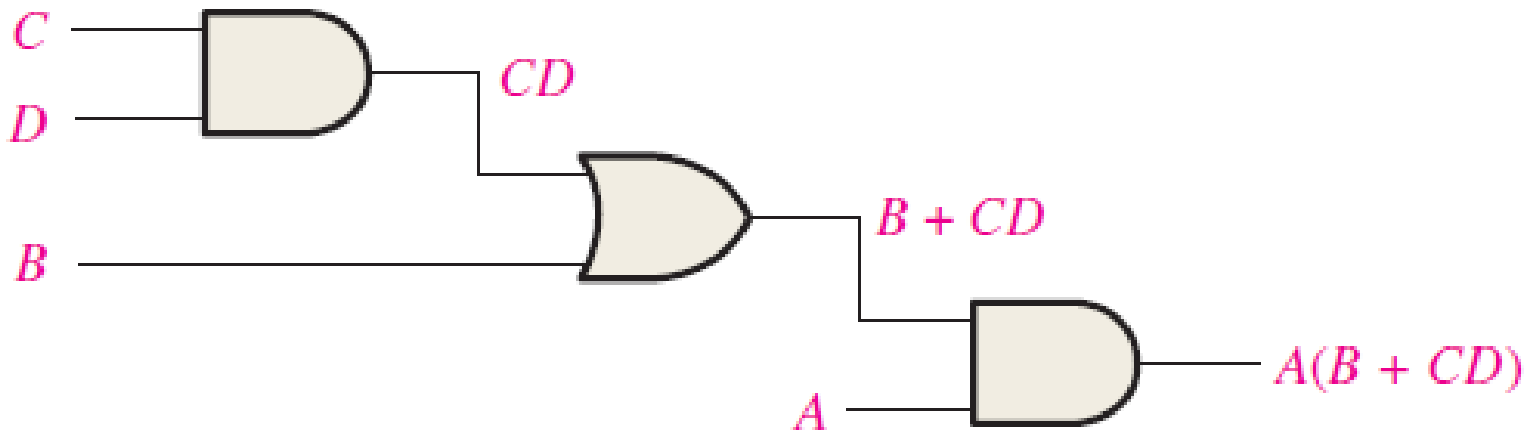
(a) $\overline{ABC} + (\overline{\overline{D} + E})$ (b) $\overline{(A + B)C}$ (c) $\overline{A + B + C} + \overline{\overline{D}E}$

Boolean Analysis of Logic Circuit

Boolean algebra provides a concise way to express the operation of a logic circuit formed by a combination of logic gates so that the output can be determined for various combinations of input values.

- A logic circuit can be expressed by Boolean expression and Boolean expression can be implemented by a logic circuit.
- The following Boolean expression can be implemented by the logic circuit below:

$$\mathbf{A(B + CD)}$$



Constructing a Truth-table from a Boolean Expression

- Once we have the Boolean expression describing a process or a logical circuit, a truth-table to show the operation for all possible combination can be constructed.
- First, we need to determine the number of inputs in the expression.
- Then, we need to note down all possible combination of the inputs.
- Lastly, we will evaluation the expression for all possible combination.

$$\mathbf{A(B + CD)}$$

Inputs				Output
A	B	C	D	$A(B + CD)$
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

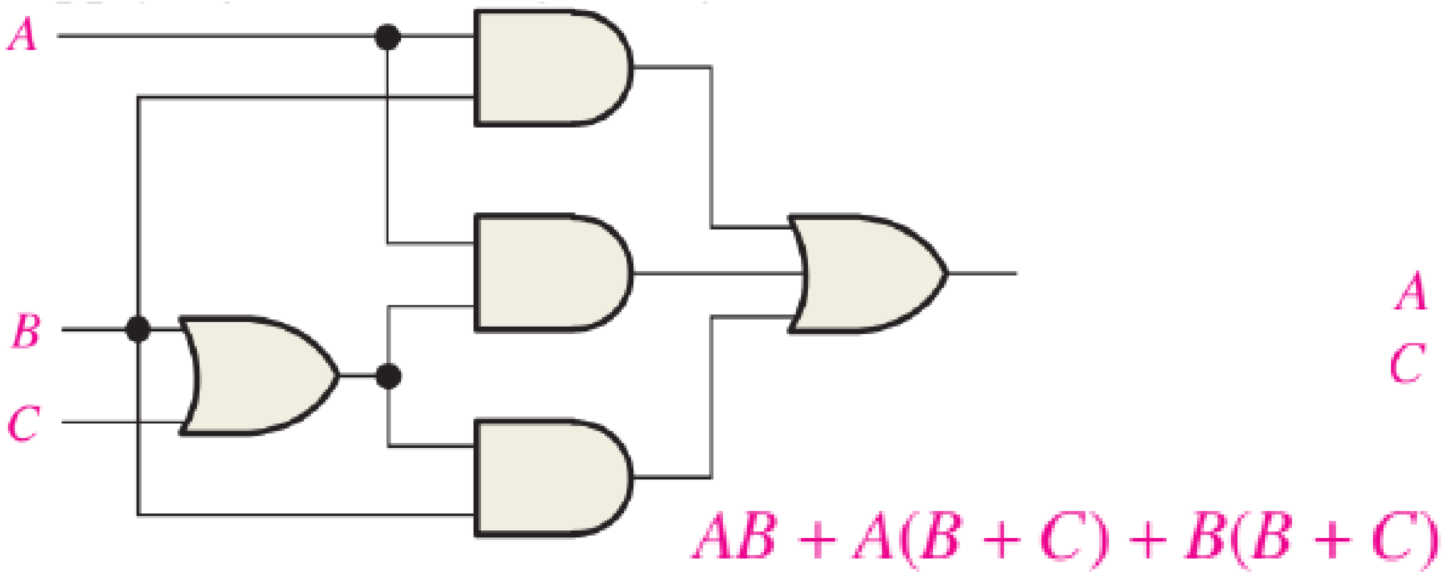
Logic Simplification

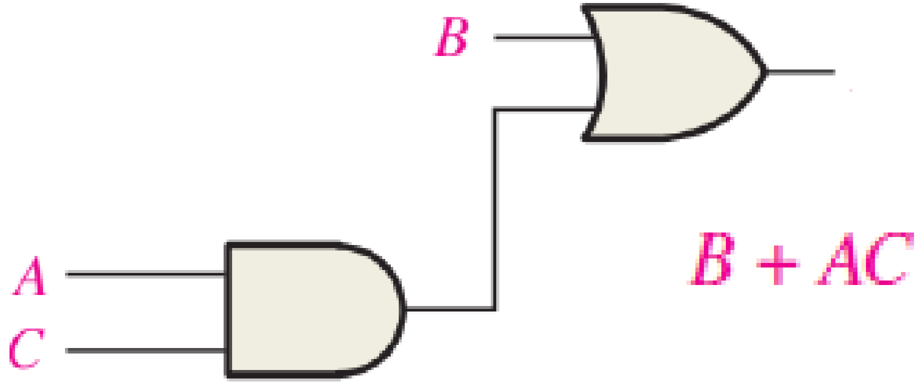
- We know that a Boolean expression can be implemented by a logical circuit.
- A large Boolean expression can often be simplified to a simpler and shorter expression.
- This is done by applying the laws and rules of Boolean algebra.
- Simplifying makes implementation simpler and thus requires lesser number of gates.
- Boolean algebra can be used to simplify the following expression:

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

- The simplified expression is:

$$\mathbf{B + AC}$$





Logic Simplification

Simplify the Boolean expression $A\overline{B} + A(\overline{B + C}) + B(\overline{B + C})$.

Simplify the following Boolean expression:

$$[A\overline{B}(C + BD) + \overline{A}\,\overline{B}]C$$

Simplify the following Boolean expression:

$$\overline{A}BC + A\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,\overline{C} + A\overline{B}C + ABC$$

Simplify the following Boolean expression:

$$\overline{AB + AC} + \overline{A}\,\overline{B}C$$

Simplify the Boolean expression $\overline{AB} + \overline{AC} + \overline{A}\,\overline{B}\,\overline{C}$.

***Applying rules of Boolean algebra and DeMorgan's Theorem show that:

i) $\mathbf{MN + \overline{MO} + M\overline{N}O(MN + O) = 1}$

ii) $\mathbf{\overline{A}B\overline{C} + \overline{A}BC + AB\overline{C} + ABC = B}$

1. Simplify the following Boolean expressions:

   (a) $A + AB + A\overline{B}C$ (b) $(\overline{A} + B)C + ABC$ (c) $A\overline{B}C(BD + CDE) + A\overline{C}$

2. Implement each expression in Question 1 as originally stated with the appropriate logic gates. Then implement the simplified expression, and compare the number of gates.

References

1. Thomas L. Floyd, “Digital Fundamentals” 11th edition, Prentice Hall – Pearson Education.

Thank You

Lecture -2
Sum-of-Products, Product-of-Sum

Prepared By: Asif Mahfuz

Standardization

- All Boolean expression, regardless of their form, can be converted into either of the two standard forms.
- The two standard forms are: Sum-of-Products (SOP) and Product-of-Sum (POS).
- Standardization makes evaluation, simplification and implementation of Boolean expression much more systematic and easier.

Domain of a Boolean Expression

- The domain of a general Boolean expression is the set of variable contained in the Boolean expression in either complemented or uncomplemented form.
- For example, the domain of the expression $A\overline{B} + \overline{A}C$ is the set of variable A,B and C.
- And for example, the domain of the expression $A\overline{B}C + \overline{C}E + \overline{A}\overline{D}$ is the set of variables A,B,C,D and E.

Sum-of-Product (SOP)

- Sum of Products: A Product is defined as a term consisting of products of the literals. When two or more products are summed in a Boolean expression, it is called the Sum-of-Products (SOP).

$$\mathbf{A + \overline{A}B + B\overline{C}}$$

- Standard SOP: It is an expression where all the variables are present in each product terms. The products in a SSOP are called min terms ($\mathbf{m_i}$).

$$\mathbf{AB\overline{C} + \overline{A}BC + A\overline{B}C}$$

- Conversion of SOP to Standard SOP
  - Step1: Multiply each of the non-standard terms with a term made up of the sum of the missing variable and its complement. This do not change the function as we are just multiplying by 1.
  - Step2: Repeat step 1 until all the non-standard terms become standard terms.
- **Sum-of-Products is used to describe when the function is 1.**

Product-of-Sum (POS)

- Product of Sum: A Sum is defined as a term consisting of sum of the literals. When two or more sum terms are multiplied in a Boolean Expression, it is called the Product-of-Sum (POS).

$$(\mathbf{B} + \overline{\mathbf{C}})(\mathbf{A} + \overline{\mathbf{B}})$$

- Standard POS: It is an expression where all the variables are present in each sum terms. Each sum terms are called max terms ($\mathbf{M_i}$) .

$$(\overline{\mathbf{A}} + \mathbf{B} + \mathbf{C})(\mathbf{A} + \overline{\mathbf{B}} + \overline{\mathbf{C}})$$

- Conversion of POS to Standard POS
  - Step1: Add to each non-standard product terms a term made up of the product of the missing variable and its complement. This does not change the expression as we are just adding a 0.
  - Step2: Apply rule 12: $\mathbf{A} + \mathbf{BC} = (\mathbf{A} + \mathbf{B})(\mathbf{A} + \mathbf{C})$
  - Step3: Repeat step 1 until all the sum terms contain all the variable in the domain.
- **Product-of-Sum is used to describe when the function is 0.**

SOP & POS

Convert the following Boolean expression into standard SOP form:

$$A\overline{B}C + \overline{A}\,\overline{B} + AB\overline{C}D$$

Convert the expression $W\overline{X}Y + \overline{X}Y\overline{Z} + WX\overline{Y}$ to standard SOP form.

Convert the following Boolean expression into standard POS form:

$$(A + \overline{B} + C)(\overline{B} + C + \overline{D})(A + \overline{B} + \overline{C} + D)$$

Convert the expression $(A + \overline{B})(B + C)$ to standard POS form.

Min Terms and Max Terms

- Each variable in a Boolean expression is a literal.
- Boolean variable can appear in normal (A) or complemented ($\overline{A}$) form.
- Each product of all variables in the domain is called Min-Term.
- Each sum of all variables in the domain is called Max-Term.

A	B	C	Min-Terms	
0	0	0	$\bar{A}\bar{B}\bar{C}$	m_0
0	0	1	$\bar{A}\bar{B}C$	m_1
0	1	0	$\bar{A}B\bar{C}$	m_2
0	1	1	$\bar{A}BC$	m_3
1	0	0	$A\bar{B}\bar{C}$	m_4
1	0	1	$A\bar{B}C$	m_5
1	1	0	$AB\bar{C}$	m_6
1	1	1	ABC	m_7

A	B	C	Max-Terms	
0	0	0	$A + B + C$	M_0
0	0	1	$A + B + \bar{C}$	M_1
0	1	0	$A + \bar{B} + C$	M_2
0	1	1	$A + \bar{B} + \bar{C}$	M_3
1	0	0	$\bar{A} + B + C$	M_4
1	0	1	$\bar{A} + B + \bar{C}$	M_5
1	1	0	$\bar{A} + \bar{B} + C$	M_6
1	1	1	$\bar{A} + \bar{B} + \bar{C}$	M_7

For Min-Terms:
When 0→ Complemented Form
When 1→ Normal Form

For Max-Terms:
When 1→ Complemented Form
When 0→ Normal Form

Boolean Expression(SOP) to Truth-Table

- Truth-table can be formed for any Boolean expression.
- Converting a Boolean Expression to SSOP can make this task a lot easier.
- Find the truth-table for the following Boolean expression:

$$F(A,B,C) = AB + \bar{B}\bar{C}$$

$$F(A,B,C) = AB(C + \bar{C}) + (A + \bar{A})\bar{B}\bar{C}$$

$$F(A,B,C) = ABC + AB\bar{C} + A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}$$

A	B	C	Min-Terms
0	0	0	$\bar{A}\bar{B}\bar{C}$
0	0	1	$\bar{A}\bar{B}C$
0	1	0	$\bar{A}B\bar{C}$
0	1	1	$\bar{A}BC$
1	0	0	$A\bar{B}\bar{C}$
1	0	1	$A\bar{B}C$
1	1	0	$AB\bar{C}$
1	1	1	ABC

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

Truth-Table to Function Implementation

- Find and implement the function from the following truth-table.

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

A	B	C	F
0	0	0	$\bar{A}\bar{B}\bar{C}$
0	0	1	
0	1	0	$\bar{A}B\bar{C}$
0	1	1	
1	0	0	
1	0	1	
1	1	0	$AB\bar{C}$
1	1	1	ABC

$$\mathbf{F(A,B,C) = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + AB\bar{C} + ABC}$$
$$\mathbf{F(A,B,C) = \bar{A}\bar{C} + AB}$$

- For the function $F = \sum(2,3,5,7)$:
  a) Construct the truth table.
  b) Implement the function.

Boolean Expression(POS) to Truth-Table

- Form a truth-table for the following Boolean expression.

$$\mathbf{F(A, B, C) = (A + B) \cdot (\bar{B} + \bar{C})}$$

- In the first step, we will convert the POS to SPOS:

$$\mathbf{F(A, B, C) = (A + B + C\bar{C}) \cdot (\bar{B} + \bar{C} + A\bar{A})}$$

$$\mathbf{F(A, B, C) = (A + B + C) \cdot (A + B + \bar{C}) \cdot (\bar{B} + \bar{C} + A) \cdot (\bar{B} + \bar{C} + \bar{A})}$$

- Once we have converted the expression to SPOS, now we can directly form the truth-table.

A	B	C	Max-Terms
0	0	0	$A+B+C$
0	0	1	$A+B+\bar{C}$
0	1	0	$A+\bar{B}+C$
0	1	1	$A+\bar{B}+\bar{C}$
1	0	0	$\bar{A}+B+C$
1	0	1	$\bar{A}+B+\bar{C}$
1	1	0	$\bar{A}+\bar{B}+C$
1	1	1	$\bar{A}+\bar{B}+\bar{C}$

A	B	C	Max-Terms
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Function Implementation using POS

- For the function $F = \prod(1,2,5,6)$:
  a) Construct the truth table.
  b) Implement the function.

Solution:

Step 1

A	B	C	Max-Terms
0	0	0	
0	0	1	0
0	1	0	0
0	1	1	
1	0	0	
1	0	1	0
1	1	0	0
1	1	1	

Step 2

A	B	C	Max-Terms
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Step 3 $\mathbf{F(A,B,C) = \bar{A}\bar{B}\bar{C} + \bar{A}BC + A\bar{B}\bar{C} + ABC}$

- **Now that we have the Boolean expression, we can simplify it and implement it.**

Connecting the Dots between SOP and POS

- From the truth table determine the standard SOP expression and the equivalent POS expression.

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

Standard SOP expression:

- Write down the sum of min terms of the combinations for which the function is 1.

$$\mathbf{F = \overline{A}BC + A\overline{B}\overline{C} + A\overline{B}C + AB\overline{C} + ABC}$$

Equivalent POS expression:

- Write down the product of max terms of the combinations for which the function is 0.

$$\mathbf{F = (A + B + C) \cdot (A + B + \overline{C}) \cdot (A + \overline{B} + C)}$$

HOW DOES THIS EVEN WORK!!!!!!!!

Connecting the Dots between SOP and POS

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

A	B	C	F	$\bar{F}$
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

$$\mathbf{F = \bar{A}BC + A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC}$$

$$\mathbf{\bar{F} = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C}}$$

- Now we have the SOP expression of the complement of the function:

$$\mathbf{\bar{F} = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C}}$$

- This means we have converted the 0s of the function 1s and vice versa.
- What if we turn the 0s back to 1, that is we convert $\bar{F}$ to F

Connecting the Dots between SOP and POS

$$\bar{\mathbf{F}} = \bar{\mathbf{A}}\bar{\mathbf{B}}\bar{\mathbf{C}} + \bar{\mathbf{A}}\bar{\mathbf{B}}\mathbf{C} + \bar{\mathbf{A}}\mathbf{B}\bar{\mathbf{C}}$$

- Now to convert the 0s back to 1s we complement $\bar{\mathrm{F}}$ that is $\bar{\bar{\mathrm{F}}}$.

$$\bar{\bar{\mathbf{F}}} = \overline{\bar{\mathbf{A}}\bar{\mathbf{B}}\bar{\mathbf{C}} + \bar{\mathbf{A}}\bar{\mathbf{B}}\mathbf{C} + \bar{\mathbf{A}}\mathbf{B}\bar{\mathbf{C}}}$$

$$\mathbf{F} = \overline{\bar{\mathbf{A}}\bar{\mathbf{B}}\bar{\mathbf{C}}} \cdot \overline{\bar{\mathbf{A}}\bar{\mathbf{B}}\mathbf{C}} \cdot \overline{\bar{\mathbf{A}}\mathbf{B}\bar{\mathbf{C}}}$$

$$\mathbf{F} = (\mathbf{A} + \mathbf{B} + \mathbf{C}) \cdot (\mathbf{A} + \mathbf{B} + \bar{\mathbf{C}}) \cdot (\mathbf{A} + \bar{\mathbf{B}} + \mathbf{C})$$

- So we have reached the same expression of POS as we did earlier.
- SOP is the positive logic definition of the function.
- POS is the negative logic definition of the function.

What is positive logic and what is negative logic????

References

1. Thomas L. Floyd, "Digital Fundamentals" 11th edition, Prentice Hall – Pearson Education.

Thank You