Lecture 04 Boolean Algebra

ECE09 – Digital Electronics: Logic Circuits and Switching Theory

Engr. Zoren P. Mabunga, M.Sc

Introduction

- Boolean Algebra, named after its pioneer **George Boole** (1815-1864) is the algebra of logic presently applied to computer systems and devices.
- The rule of this algebra is based on human reasoning.
- Boolean algebra remained in the realm of philosophy till 1938 when Claude E. Shannon used it to solve relay logic problems.
- As compared to other mathematical tools of analysis and design, Boolean algebra has the advantages of simplicity, speed and accuracy.

Laws of Boolean Algebra

- OR LAWS

Law 1. $A + 0 = A$

Law 2. $A + 1 = 1$

Law 3. $A + A = A$

Law 4. $A + \overline{A} = 1$

- AND LAWS

Law 5. $A \,.\, 0 = 0$

Law 6. $A \,.\, 1 = A$

Law 7. $A \,.\, A = A$

Law 8. $A \,.\, \overline{A} = 0$

Laws of Boolean Algebra

- LAWS OF COMPLEMENTATION

Law 9. $\overline{0} = 1$

Law 10. $\overline{1} = 0$

Law 11. if $A = 0$, then $\overline{A} = 1$

Law 12. if $A = 1$, then $\overline{A} = 0$

Law 13. $A = A$

- COMMUTATIVE LAWS

These laws allow ***change in the position*** of variables in *OR* and *AND* expressions.

Law 14. $A + B = B + A$

Law 15. $A \, . \, B = B \, . \, A$

Laws of Boolean Algebra

- ASSOCIATIVE LAWS

These laws allow removal of brackets from logical expression and regrouping of variables.

Law 16. $A+(B+C)=(A+B)+C$

Law 17. $(A+B)+(C+D)=A+B+C+D$

Law 18. $A\,.\,(B\,.\,C)=(A\,.\,B)\,.\,C$

- DISTRIBUTIVE LAWS

These laws permit factoring or multiplying out of an expression.

Law 19. $A\,(B+C)=A\,B+A\,C$

Law 20. $A+BC=(A+B)(A+C)$

Law 21. $A+\bar{A}\,.\,B=A+B$

Laws of Boolean Algebra

- ABSORPTIVE LAWS

These enable us to reduce a complicated logic expression to a simpler form by absorbing some of the terms into existing terms.

Law 22. $A + AB = A$

Law 23. $A.(A + B) = A$

Law 24. $A.(\bar{A} + B) = AB$

- DE MORGAN'S THEOREM

Law 25. $\overline{A + B} = \bar{A}\,.\,\bar{B}$

Law 26. $\overline{A.B} = \bar{A} + \bar{B}$

Examples

1. Prove the following Boolean identity: $AC + ABC = AC$

Examples

2. Prove the following Boolean identity: $(\mathrm{A} + \mathrm{B})(A + C) = A + BC$

Examples

3. Prove the following Boolean identity: $A + \bar{A}B = A + B$

Examples

4. Prove the following Boolean identity: $\mathrm{ABC} + \mathrm{A}\bar{B}C + AB\bar{C} = A(B + C)$

Examples

5. Demorganize the expression: $\overline{(A+B)(C+D)}$

Examples

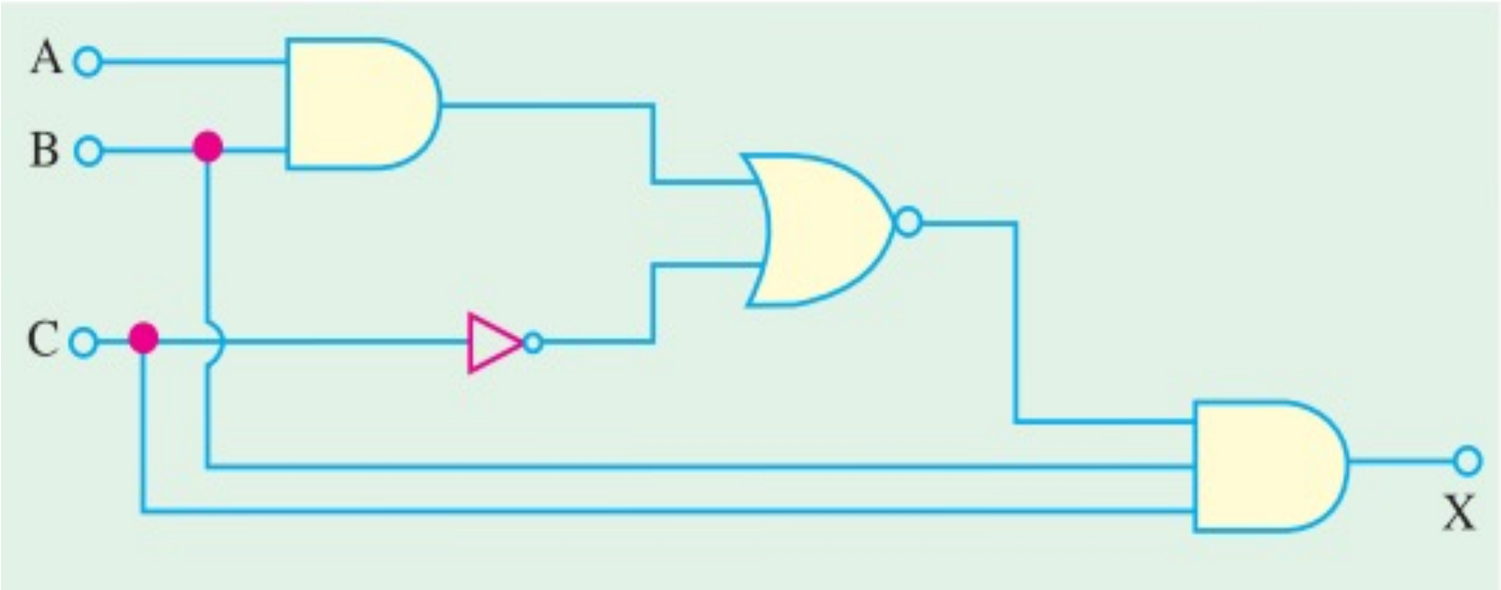
6. Simplify each of the following using De Morgan’s Theorem

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

   b. $\overline{\overline{\overline{AB}\,\overline{C}}D}$

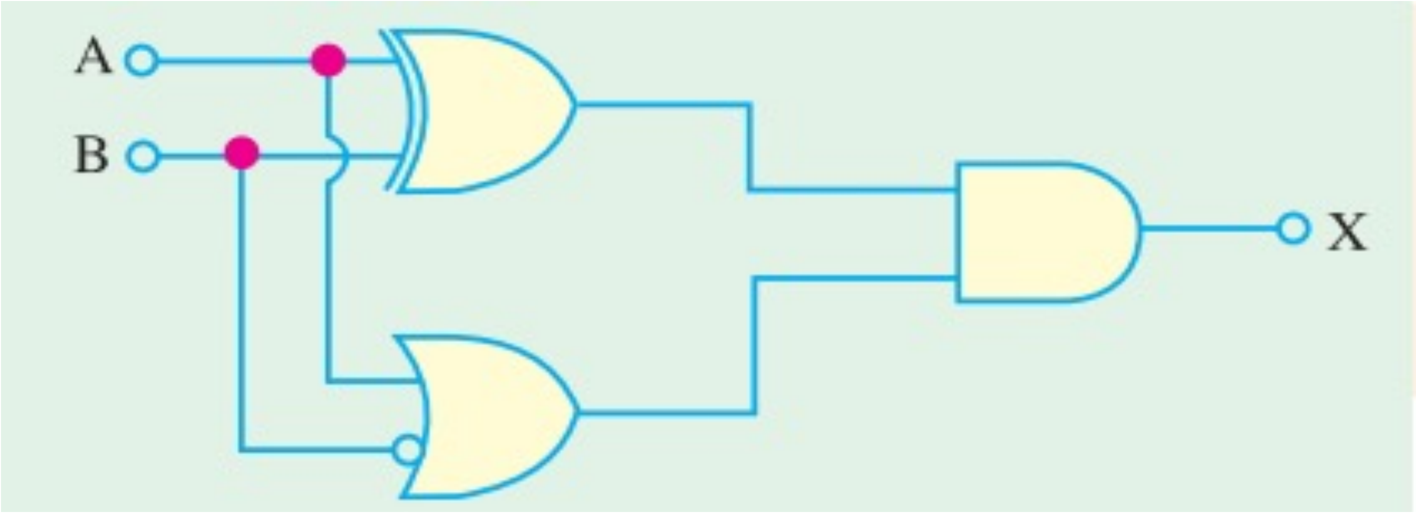
Examples

- Determine the Boolean expression for the logic circuit below. Simplify the Boolean expression and redraw the logic circuit using the simplified Boolean expression.



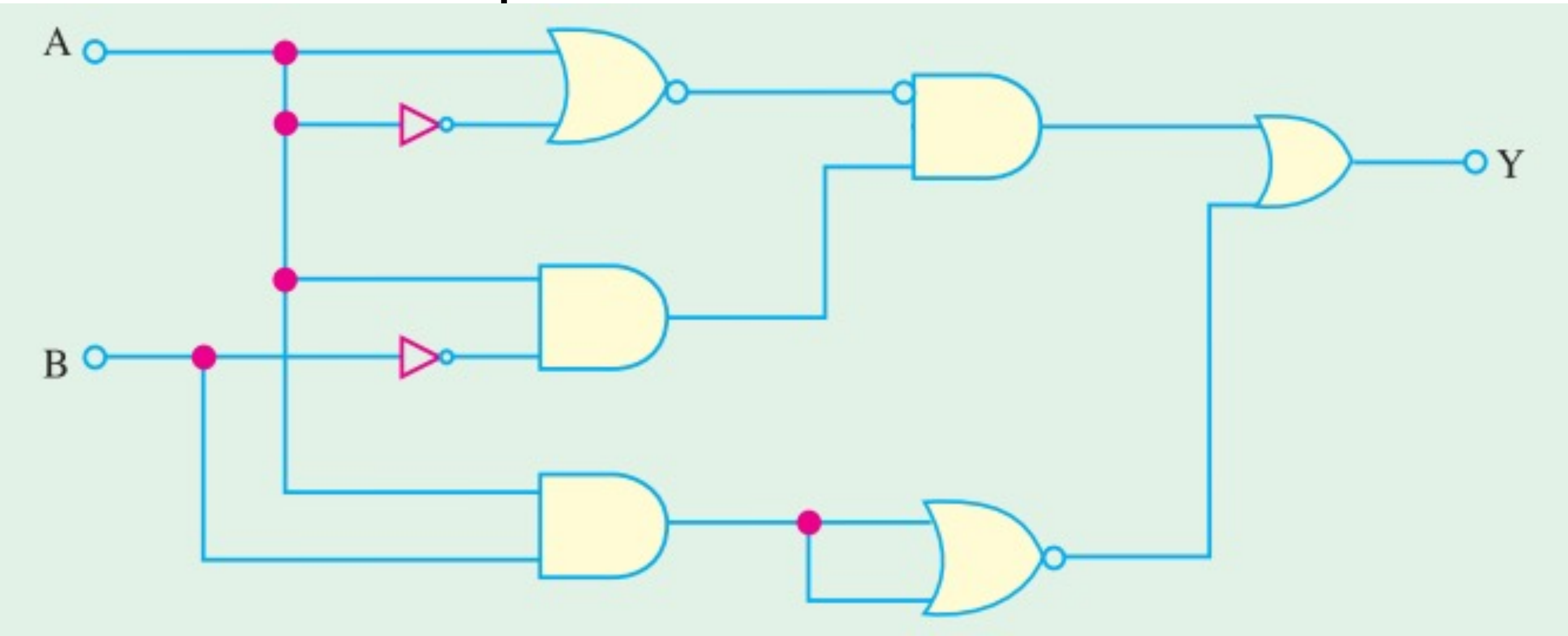
Example

- Determine the output X of a logic circuit shown below. Simplify the output expression using Boolean laws and theorems. Redraw the logic circuit with the simplified version.



Example

- Determine the output X of a logic circuit shown below. Simplify the output expression using Boolean laws and theorems. Redraw the logic circuit with the simplified version.



Design of Logic Circuit Using Boolean Algebra

1. SUM OF PRODUCT (SOP)

$$A\overline{B} + \overline{A}BC \qquad \overline{A}BC + A\overline{C}D + A\overline{B}CD, \qquad \overline{A}B + AC + \overline{A}\,\overline{B}\,C$$

2. PRODUCTS OF SUM (POS)

$$(A + \overline{B})(\overline{A} + B + C), \qquad (\overline{A} + \overline{B})(A + C)(\overline{A} + \overline{B} + C)$$

Conversion of Truth Table to Equivalent Boolean Expression

A	B	C	X	MINTERM	MAXTERM
0	0	0	0		
0	0	1	1		
0	1	0	0		
0	1	1	0		
1	0	0	1		
1	0	1	0		
1	1	0	0		
1	1	1	1		

Examples

1. Obtain the simplified Boolean expression based on the given truth table.

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

Examples:

1. Design a logic circuit that has three inputs A, B and C whose output will be high only when the majority of the inputs are high.
2. Jon Snow’s castle has three doors. He wants to know when a white walker opens one of his door. Design a logic circuit that will be used to trigger an alarm when only one door is left open.