LEARNING OUTCOMES

By the end of this topics, you will be able to:

1. describe fundamental concept of Boolean Algebra, Boolean Functions, Truth Table and Algebraic manipulation of Boolean expressions.
2. simplify Boolean functions.
3. use Boolean function for correspondence with electronic circuits, combinational logic and sequential logic.
4. use sequential and clocked logic mutually exclusive projects.

INTRODUCTION

This topic explains about the fundamental of Boolean Algebra, Boolean functions, truth table and Algebraic manipulation of Boolean expressions.

4.1 FUNDAMENTAL CONCEPTS OF BOOLEAN ALGEBRA

Boolean algebra is the branch of algebra in Mathematic which make the computers perform the simple to extremely complex instructions. The variables of Boolean Algebra are represented by the values of *true* which denoted by 1 and *false* which denoted by 0.

The main operations of Boolean algebra are the conjunction AND denoted as $\wedge$, the disjunction OR denoted as $\vee$, and the negation NOT denoted as $\neg$. It is a formalism for describing logical relations in the same way that ordinary algebra describes numeric relations.

SELF CHECK 4.1

1. Describe the Boolean algebra.
2. List THREE (3) main operation of Boolean algebra.

4.2 BOOLEAN FUNCTIONS AND TRUTH TABLES

4.2.1 BOOLEAN ALGEBRA

Boolean algebra operates with three functional operators; AND, OR and NOT which builds block of digital logic design. It is shown in Table 4.1:

Table 4.6

Operator Name	Alternate Name	Example	Alternative Representations
AND	Intersection logical multiplication	xy	x.y, x∩y, x&y, *x*∧*y*
OR	Union logical addition	x + y	x∪y, x∨y
NOT	Complement inversion	¬x	x’

4.2.1 TRUTH TABLE

A truth table is a breakdown of a logic function by listing all possible values the function can attain. the table contains several rows and columns. The top row representing the logical variables and combinations, in increasing complexity leading up to the final function.

For example, the values of *x*∧*y*, *x*∨*y*, and ¬*x* can be expressed by tabulating their values with the following truth table:

Table 4.7

x	y	x∧y	x∨y	¬x	¬y
0	0	0	0	1	1
1	0	0	1	0	1
0	1	0	1	1	0
1	1	1	1	0	0

SELF CHECK 4.2

1. Use a truth table to simplify the following expressions:
   - (x.y) + (¬x.y)
   - (¬x.y).(x.y)

2. Use a Boolean Algebra to simplify the following truth tables:

Table 4.8

x	y	x∨y
0	0	0
1	0	1
0	1	1
1	1	1

4.3 ALGEBRAIC MANIPULATION OF BOOLEAN EXPRESSIONS

This approach can transform one Boolean expression into an equivalent expression with the application of Boolean Theorems. There are no fixed rules that can be used to minimize a given expression. It is based on individual's ability to apply Boolean Theorems in order to minimise a function.

Boolean algebra satisfies many of the same laws as ordinary algebra when one matches up ∨ with addition and ∧ with multiplication. In particular the laws shown in Figure 4.3 are common to both kinds of algebra.

Associativity of $\vee$	$x \vee (y \vee z) = (x \vee y) \vee z$
Associativity of $\wedge$	$x \wedge (y \wedge z) = (x \wedge y) \wedge z$
Commutativity of $\vee$	$x \vee y = y \vee x$
Commutativity of $\wedge$	$x \wedge y = y \wedge x$
Distributivity of $\wedge$ over $\vee$	$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$
Identity for $\vee$	$x \vee 0 = x$
Identity for $\wedge$	$x \wedge 1 = x$
Annihilator for $\wedge$	$x \wedge 0 = 0$
Annihilator for $\vee$	$x \vee 1 = 1$
Idempotence of $\vee$	$x \vee x = x$
Idempotence of $\wedge$	$x \wedge x = x$
Absorption 1	$x \wedge (x \vee y) = x$
Absorption 2	$x \vee (x \wedge y) = x$

Figure 4.19

SELF CHECK 4.3

1. List any THREE (3) Boolean algebra rules.

4.4 SIMPLIFICATION OF BOOLEAN FUNCTIONS

Simplification of Boolean function is another approach to minimize the Boolean expression an equivalent expression. the Boolean expression can be minimized using any of law listed in Figure 4.3.

For example, given the Boolean expression $F(A,B,C)=(A+B)(A+C)$. It can be minimized through any one of the following Law:

- Distributive Rule: F(A,B,C)=A.A+A.C+B.A+B.C
- Idempotent Law: F(A,B,C)=A+A.C+B.A+B.C
- Distributive Law: F(A,B,C)=A(1+C)+B.A+B.C
- Dominance Law: F(A,B,C)=A+B.A+B.C
- Distributive Law: F(A,B,C)=(A+1).A+B.C
- Dominance Law: F(A,B,C)=1.A+B.C
- Dominance Law: F(A,B,C)=A+B.C

So, F(A,B,C)=A+BC is the minimized form.

SELF CHECK 4.4

1. Simplify the following Boolean expression using any appropriate law:
   - ab+ab
   - (x+¬y)(x+y)

4.5 CORRESPONDENCE BETWEEN ELECTRONIC CIRCUITS AND BOOLEAN FUNCTIONS

The electronic circuits have one-to-one correspondence with boolean functions. The boolean function can design an electronic circuit and vice versa. The boolean functions required the AND, OR, and NOT boolean operators to construct any electronic circuit exclusively.

The AND, OR and NOT Boolean operators are represented by the basic circuits; the AND, OR and inverter (NOT).

4.5.1 AND GATE

Figure 4.20 is showing the AND gate symbol, the representation using truth table and the Boolean expression.

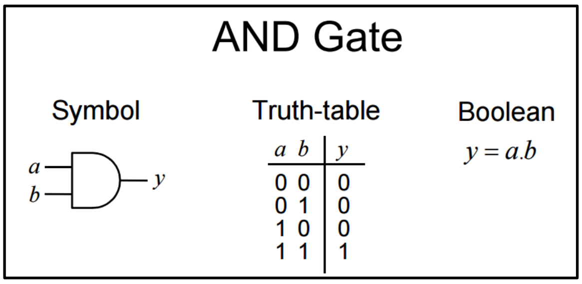


Figure 4.20

4.5.2 OR GATE

Figure 4.21 is showing the OR gate symbol, the representation using truth table and the Boolean expression.

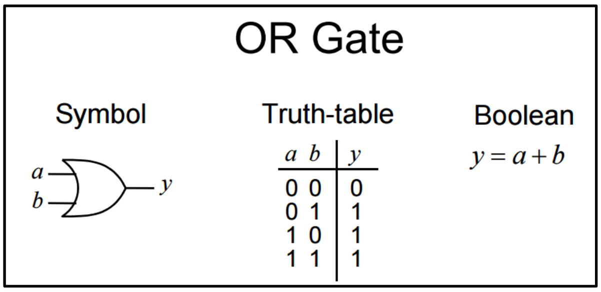


Figure 4.21

4.5.3 NOT GATE

Figure 4.22 is showing the NOT gate symbol, the representation using truth table and the Boolean expression.

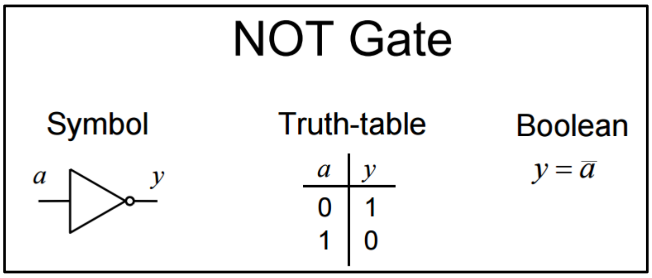


Figure 4.22

SELF CHECK 4.5

1. List THREE (3) basic logic gates.
2. Draw a logic gate for the AND, OR and NOT logic operation.

4.6 COMBINATIONAL LOGIC

The logic diagram also can be represented with combinational logic gates such as NOT AND (NAND), NOT OR (NOR) and EXCLUSIVE OR (XOR).

4.6.1 NOT AND GATE (NAND)

The NAND logic gate in a single gate symbol as shown in Figure 4.23. It can be represented using truth table and the Boolean expression.

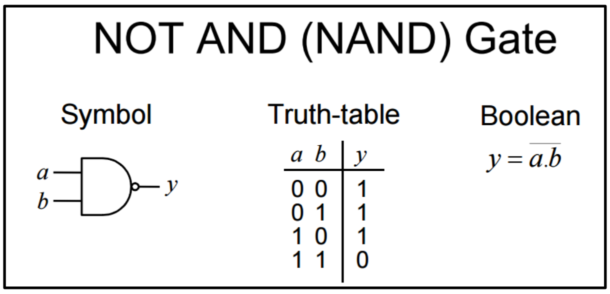


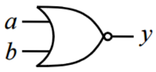
Figure 4.23

4.6.2 NOT OR GATE (NOR)

Next, the logic gate also can represent the NOR gate in a single gate symbol (Figure 4.24). It can be represented using truth table and the Boolean expression.

NOT OR (NOR) Gate

Symbol



Truth-table

a	b	y
0	0	1
0	1	0
1	0	0
1	1	0

Boolean

$y = \overline{a + b}$

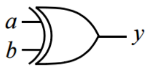
Figure 4.24

4.6.3 EXCLUSIVE OR GATE (XOR)

Another logic gate is XOR gate (Figure 4.25) which can be represented using truth table and the Boolean expression.

EXCLUSIVE OR (XOR) Gate

Symbol



Truth-table

a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

Boolean

$y = a \oplus b$

Figure 4.25

SELF CHECK 4.6

1. Draw a logic diagram for the following Boolean expressions:
   i. (A + ¬B + C) (¬A + B + ¬C)
   ii. (A + B + ¬C) (A + B + ¬C) (¬A + ¬B + ¬C)
   iii. ¬A +((B + C)(¬B + ¬C)

4.7 SEQUENTIAL LOGIC

Sequential logic has a memory to take the previous input state along with the actual one. it has three states; present input, past input and/or past output.

It can be represented into:

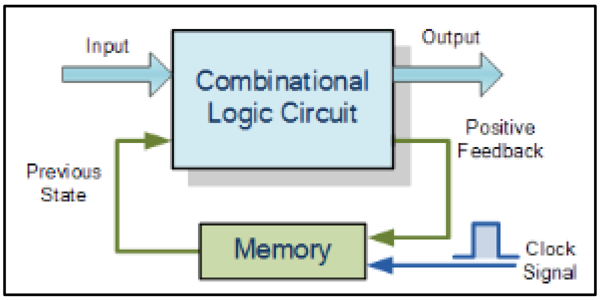


Figure 4.26

SELF CHECK 4.7

1. Draw the sequential logic diagram.

4.8 SEQUENTIAL AND CLOCKED LOGIC MUTUALLY EXCLUSIVE PROJECTS

The SR flip flop logic diagram (Figure 4.27) is the example of the sequential logic diagram.

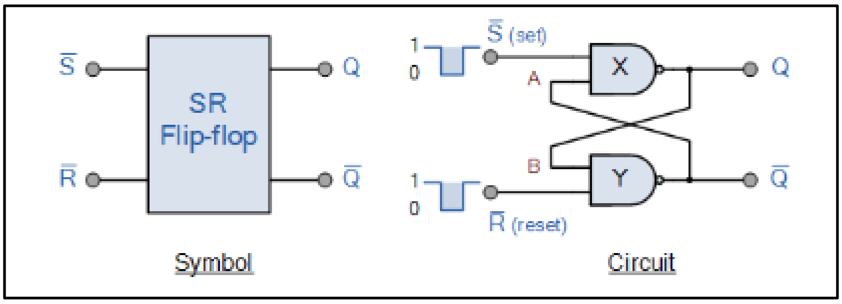


Figure 4.27

SELF CHECK 4.8

1. Draw a SR flip flop logic diagram.

SUMMARY

In this topic you have learnt that:

Fundamental concept of Boolean Algebra, Boolean Functions, Truth Table, Algebraic manipulation of Boolean expressions, Boolean functions, Boolean function for correspondence with electronic circuits, combinational logic, sequential logic, sequential and clocked logic mutually exclusive projects

KEY TERMS

Logic Diagram	Describes the arrangement of the circuit's logic gates.
Logic Circuit	An electronic circuit used in computers to perform a logical operation on its two or more input signals.

REFERENCES

Stallings, W., (2019). *Computer Organization and Architecture Designing for Performance.* 11th ed. New York: Pearson.