taleemzone.com

CLASS 9TH • COMPUTER SCIENCE

3

Digital Systems and Logic Design

Comprehensive Study Material

SECTION 1: ANALOG AND DIGITAL SIGNALS

1. Analog Signals

Analog signals are signals that change with time smoothly and continuously. They have any value within a given range.

EXAMPLE

- Voice signal (speaking)
- Body's temperature
- Radio-wave signals

2. Digital Signals

Digital signals are the signals which have only two values that are in the form of '0' and '1'. These are utilized in digital electronics and computing systems. Digital logic circuits use binary values to perform various operations, and they are essential to the function and operation of computers and many other electronic devices.

Comparison: Analog vs Digital Signals

Aspect	Analog Signal	Digital Signal
Nature	**Continuous**	**Discrete or Discontinuous**
Possible Values	Infinite possible values	Finite (0 or 1)
Example	Sound waves, Pulse rate	Binary data in computers

Analog to Digital Converter (ADC) and Digital to Analog Converter (DAC)

ADC and DAC are important operations in today's technological developments, enabling the transmission and control of signals.

Why is DAC/ADC Conversion Needed?

Digital to analog conversion and analog to digital conversion are critical since they enable processing, storage, and transmission. Digital signals are much less affected by noise and interference.

SECTION 2: BOOLEAN ALGEBRA AND LOGIC GATES

1. Boolean Algebra

Boolean algebra is a branch of mathematics related to logic and symbolic computation, using two values namely True and False. It is an essential branch of digital circuits since it is the basis for the analysis and design of circuits.

2. Boolean Functions and Expressions

Binary values are used to describe the relationship between variables in Boolean functions and expressions. The expressions are built using AND, OR, and other logic operations and can be reduced to optimize digital circuits in several ways.

3. Binary Variables and Logic Operations

Binary variables can have only two values: 0 and 1. Logic operations are basic operations implemented in Boolean algebra for processing these binary variables. The primary operations are AND, OR, and NOT.

SECTION 3: BASIC BOOLEAN LOGIC OPERATIONS c&W

1. AND Operation (·)

The AND operation is the basic logical operator used in Boolean algebra. It requires two inputs which will give a single binary output. The symbol '·' is used for the AND operation. The output of the AND operation is **"1" only when both inputs are "1"**. Otherwise, the output is "0".

Mathematical Representation

text

$P = A \cdot B$

EXAMPLE

Consider two binary variables:

- A = 1 (True)

- B = 0 (False)

The result P of the AND operation is 0 (false).

Truth Table for AND Operation

A	B	A AND B (P) = A·B
0	**0**	**0**
0	1	0
1	0	0
1	1	1

2. OR Operation (+)

The OR operation outputs "1" when **at least one** of the inputs is "1".

Mathematical Representation

text

P = A + B

Truth Table for OR Operation

A	B	A OR B (P) = A+B
0	**0**	**0**
0	1	1
1	0	1
1	1	1

3. NOT Operation (¬)

The NOT operation inverts the input: if input is 0, output is 1; if input is 1, output is 0.

Mathematical Representation

text

P = $\bar{A}$ (or ¬A)

Truth Table for NOT Operation

A	NOT A (Ā)
0	**1**
1	0

SECTION 4: SPECIAL LOGIC GATES

XOR Gate (Exclusive OR)

The XOR (Exclusive OR) gate outputs true only when **exactly one** of the inputs is true. It differs from the OR gate because it does not output true when both inputs are true.

EXAMPLE

Imagine a scenario where you can either play video games or do homework, but not both at the same time.

- Play video games: Yes (True)
- Do homework: No (False)
- Allowed? Yes (True) because only one activity is being done.

Truth Table for XOR Gate

A	B	A XOR B
0	**0**	**0**
0	1	1
1	0	1
1	1	0

SECTION 5: SIMPLIFICATION OF BOOLEAN FUNCTIONS

Simplification of Boolean functions is a particularly important process in designing an efficient digital circuit. Such simplified functions require fewer gates making them compact in size, energy efficient, and faster than complicated ones. Simplification means applying some Boolean algebra rules to make the functions less complicated.

Basic Boolean Algebra Rules

1. Identity Laws

text

$A + 0 = A$

$A \cdot 1 = A$

2. Null Laws

text

$A + 1 = 1$

$A \cdot 0 = 0$

3. Idempotent Laws

text

$A + A = A$

$A \cdot A = A$

4. Complement Laws

text

$A + \bar{A} = 1$

$A \cdot \bar{A} = 0$

5. Commutative Laws

text

$A + B = B + A$

$A \cdot B = B \cdot A$

6. Associative Laws

text

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

$(A \cdot B) \cdot C = A \cdot (B \cdot C)$

7. Distributive Laws

text

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

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

8. Absorption Laws

text

$A + (A \cdot B) = A$

$A \cdot (A + B) = A$

9. De Morgan's Theorems

text

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

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

10. Double Negation Law

text

$\bar{A} = A$

SECTION 6: BOOLEAN FUNCTION SIMPLIFICATION EXAMPLES

Example 1

Simplify the expression: **$A + \bar{A} \cdot B$**

Solution:

text

$A + \bar{A} \cdot B = (A + \bar{A}) \cdot (A + B)$ [Distributive Law]

$= 1 \cdot (A + B)$ [Complement Law]

$= A + B$ [Identity Law]

Answer: $A + B$

Example 2

Simplify the expression: **$\bar{A} \cdot \bar{B} + \bar{A} \cdot B$**

Solution:

text

$\bar{A} \cdot \bar{B} + \bar{A} \cdot B = \bar{A} \cdot (\bar{B} + B)$ [Distributive Law]

$= \bar{A} \cdot 1$ [Complement Law]

$= \bar{A}$ [Identity Law]

Answer: $\bar{A}$

Example 3

Simplify the expression: **$(A + B) \cdot (A + \bar{B})$**

Solution:

text

$(A + B) \cdot (A + \bar{B}) = A + (B \cdot \bar{B})$ [Distributive Law]

$= A + 0$ [Complement Law]

$= A$ [Identity Law]

Answer: A

Example 4

Simplify the expression: **$\bar{A} + \bar{B} \cdot (A + \bar{B})$**

Solution:

text

$\bar{A} + \bar{B} \cdot (A + \bar{B}) = (\bar{A} \cdot \bar{B}) \cdot (A + \bar{B})$ [De Morgan's Theorem]

$= \bar{A} \cdot \bar{B} \cdot A + \bar{A} \cdot \bar{B} \cdot \bar{B}$ [Distributive Law]

$= \bar{A} \cdot \bar{B} + \bar{A} \cdot \bar{B}$ [Idempotent Law]

$= \bar{A} \cdot \bar{B}$ [Idempotent Law]

Answer: $\bar{A} \cdot \bar{B}$

SECTION 7: CREATING LOGIC DIAGRAMS

What are Logic Diagrams?

Logic diagrams depict the working of a digital circuit through symbols that represent individual logic gates.

Steps to Create a Logic Diagram

1. **Identify Logic Gates:** Find out the logic gates needed for the Boolean function.
2. **Arrange the Gates:** Arrange the gates to perform the operations as defined by the function of the circuit.
3. **Connect Components:** Connect the inputs and the output of the gates correctly.

Summary

Knowledge of Boolean algebra and logic gates is crucial when it comes to the creation and study of digital circuits. If students understand these concepts, they can build efficient and effective digital systems.

SECTION 8: SYSTEM TROUBLESHOOTING

What is System Troubleshooting?

Troubleshooting is essential for maintaining the smooth operation of systems, whether they are computers, machines, or other types of equipment. When something goes wrong, troubleshooting helps identify the problem and find a solution quickly.

EXAMPLE

If your computer suddenly stops working, knowing how to troubleshoot can help you get it running again without needing to call for expensive professional help.

Systematic Process of Troubleshooting

The troubleshooting process involves several steps that help you systematically identify and fix problems. These steps ensure that you don't overlook any potential issues and that you solve the problem efficiently.

Step 1: Identify Problem

The first step in troubleshooting is to identify the problem. This means recognizing that something is not working as it should.

Example: If you press the power button and your laptop does not turn ON, the problem is clear - it won't start.

Step 2: Establish a Theory of Probable Cause

Once you have identified the problem, the next step is to come up with a theory about what might be causing it. This involves thinking about what could have gone wrong.

Example: If your laptop does not turn ON, possible causes might be:

- A dead battery
- A faulty power cord
- An internal hardware issue

Step 3: Test the Theory to Determine the Cause

After establishing a theory, you need to test it to see if it is correct. This involves checking if the suspected cause is actually the reason for the problem.

Example: If you think the laptop's battery is dead, you can test this theory by plugging in the power cord and seeing if the laptop turns ON.

Step 4: Establish a Plan of Action to Resolve the Problem

If your test confirms the cause of the problem, the next step is to come up with a plan to fix it. This means deciding what steps you need to take to resolve the issue.

Example: If the problem is a dead battery, your plan of action might be to replace the battery or keep the laptop plugged in until you can get a new one.

Step 5: Implement the Solution

Once you have a plan, you need to put it into action. This means doing whatever is needed to fix the problem.

Example: If your plan is to replace the battery, you would buy a new battery and install it in your laptop.

Step 6: Verify Full System Functionality

After implementing the solution, verify that the system is working properly.

Step 7: Document Findings, Actions, and Outcomes

Record what you found, what you did, and what the results were for future reference.

SECTION 9: KEY FORMULAS AND EQUATIONS

Boolean Algebra Laws Summary

Identity Laws

text

$A + 0 = A$

$A \cdot 1 = A$

Null Laws

text

$A + 1 = 1$

$A \cdot 0 = 0$

Idempotent Laws

text

$A + A = A$

$A \cdot A = A$

Complement Laws

text

$A + \bar{A} = 1$

$A \cdot \bar{A} = 0$

Commutative Laws

text

$A + B = B + A$

$A \cdot B = B \cdot A$

Associative Laws

text

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

$(A \cdot B) \cdot C = A \cdot (B \cdot C)$

Distributive Laws

text

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

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

Absorption Laws

text

$A + (A \cdot B) = A$

$A \cdot (A + B) = A$

De Morgan's Theorems

text

$\overline{A + B} = \bar{A} \cdot \bar{B}$

$\overline{A \cdot B} = \bar{A} + \bar{B}$

Double Negation Law

text

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

Truth Tables Summary

AND Operation (·)

A	B	A·B
0	**0**	**0**
0	1	0
1	0	0
1	1	1

OR Operation (+)

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

NOT Operation (¬)

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

XOR Operation (⊕)

A	B	A⊕B
0	**0**	**0**
0	1	1
1	0	1
1	1	0

SECTION 10: KEY DEFINITIONS

Signals

- **Analog Signal:** A continuous signal that changes smoothly over time and can have any value within a given range.
- **Digital Signal:** A discrete signal that has only two values: 0 and 1.

Logic Operations

- **AND Operation:** Outputs 1 only when all inputs are 1.
- **OR Operation:** Outputs 1 when at least one input is 1.
- **NOT Operation:** Inverts the input (0 becomes 1, 1 becomes 0).

- **XOR Operation:** Outputs 1 when exactly one input is 1.

Boolean Algebra

- **Boolean Algebra:** A branch of mathematics dealing with logic and symbolic computation using two values (True/False or 0/1).
- **Boolean Function:** A function that uses binary variables and logic operations.
- **Truth Table:** A table showing all possible input combinations and their corresponding outputs.

Converters

- **ADC (Analog to Digital Converter):** Converts analog signals to digital signals.
- **DAC (Digital to Analog Converter):** Converts digital signals to analog signals.

Troubleshooting

- **Troubleshooting:** The systematic process of identifying, diagnosing, and solving problems in a system.
- **Systematic Approach:** A step-by-step method for solving problems efficiently.