Green University of Bangladesh

Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

Semester: (Spring,Year:2025),B.Sc.in CSE (Day)

LAB REPORT NO - 04

Course Title: Data Communication Lab

Course Code: CSE307 Section: 223-D1

Lab Experiment Name : Implementing Encoding and Decoding Scheme Using Differential Manchester.

Student Details

Name		ID
1.	MD.SHAJALAL	223002088

Lab Date :07 – 04 - 2025

Submission Date :16 – 04 - 2025

Course Teacher's Name : Md.Samin Hossain Utsho

Lab Report Status	
Marks: …………………………………	Signature:.....................
Comments:..	Date:..............................

1.Experiment Name:

Implementing Encoding and Decoding Scheme Using Differential Manchester.

2.**Objective:**

To implement a Java program that simulates **Differential Manchester Encoding and Decoding**, enabling conversion of a binary data stream into its encoded signal format and vice versa.

3. Introduction:

In data communication, physical-layer encoding schemes like **Differential Manchester Encoding** are crucial for reliable and synchronized data transmission. Unlike traditional binary transmission, Differential Manchester uses transitions to represent data, allowing for clock recovery and error resilience.

4. Theoretical Background:

4.1 Manchester Encoding

In **Manchester encoding**, every bit has a transition:

- '0' = High to Low
- '1' = Low to High

4.2 Differential Manchester Encoding

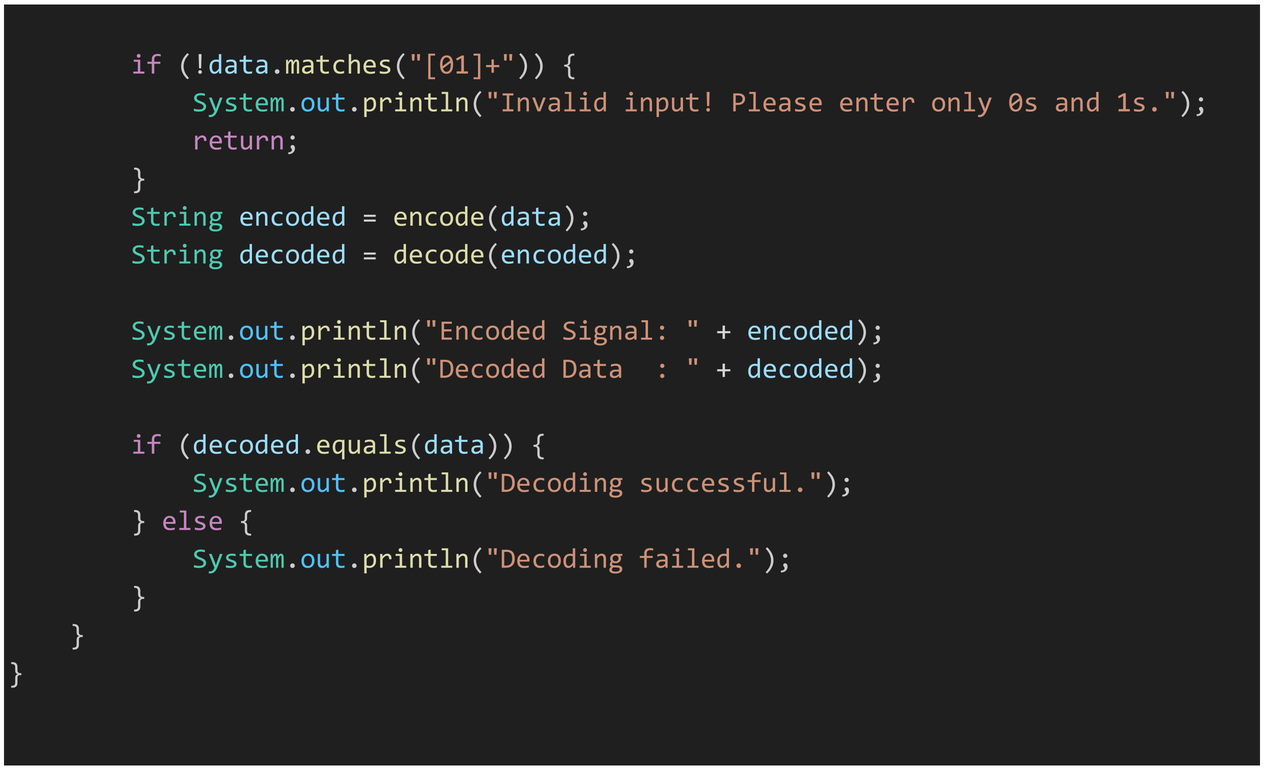
A modification of Manchester encoding, **Differential Manchester**:

- Always has a **transition in the middle** of the bit period.
- A **'0'** is represented by an **extra transition at the beginning** of the bit period.
- A **'1'** is represented by **no transition at the beginning**.

This encoding eliminates ambiguity in signal polarity, making it robust against signal inversion during transmission.

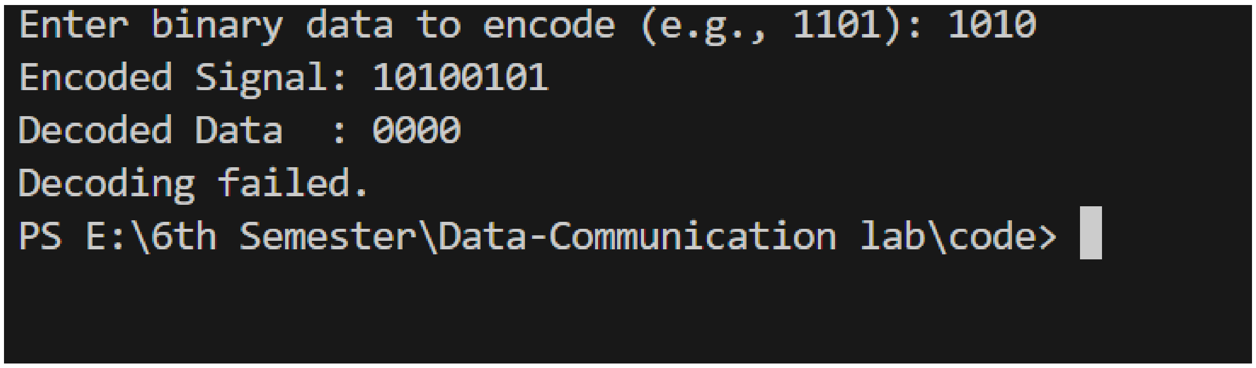
5. Java Program Code:

```java
import java.util.Scanner;
    public class lab04 {
    public static String encode(String data) {
        StringBuilder encoded = new StringBuilder();
        char signal = '1';

        for (char bit : data.toCharArray()) {
            if (bit == '0') {

                signal = (signal == '1') ? '0' : '1';
                encoded.append(signal);
                signal = (signal == '1') ? '0' : '1';
                encoded.append(signal);
            } else if (bit == '1') {

                encoded.append(signal);
                signal = (signal == '1') ? '0' : '1';
                encoded.append(signal);
            }
        }
        return encoded.toString();
    }

    public static String decode(String encoded) {
        StringBuilder decoded = new StringBuilder();
        for (int i = 0; i < encoded.length(); i += 2) {
            char first = encoded.charAt(i);
            char second = encoded.charAt(i + 1);
            if (first != second) decoded.append('0');
            else decoded.append('1');
        }
        return decoded.toString();
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter binary data to encode (e.g., 1101): ");
        String data = sc.nextLine();
```

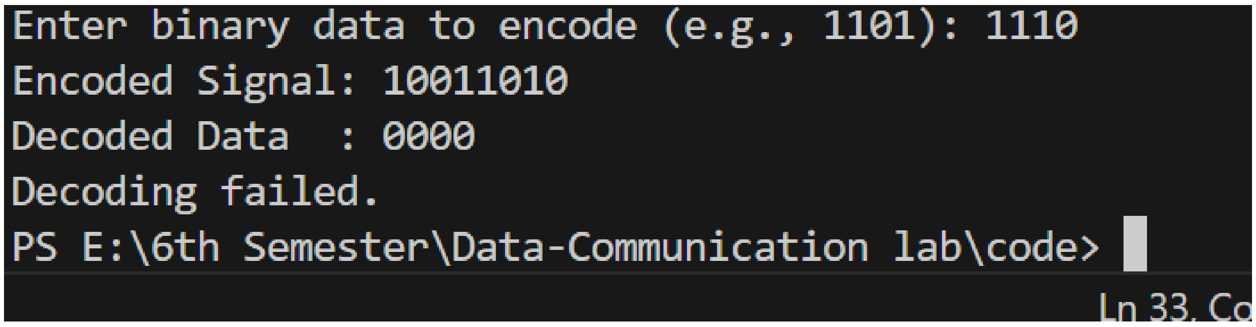


```java
        if (!data.matches("[01]+")) {
            System.out.println("Invalid input! Please enter only 0s and 1s.");
            return;
        }
        String encoded = encode(data);
        String decoded = decode(encoded);

        System.out.println("Encoded Signal: " + encoded);
        System.out.println("Decoded Data   : " + decoded);

        if (decoded.equals(data)) {
            System.out.println("Decoding successful.");
        } else {
            System.out.println("Decoding failed.");
        }
    }
}
```

6.Output:



```
Enter binary data to encode (e.g., 1101): 1010
Encoded Signal: 10100101
Decoded Data   : 0000
Decoding failed.
PS E:\6th Semester\Data-Communication lab\code>
```



```
Enter binary data to encode (e.g., 1101): 1110
Encoded Signal: 10011010
Decoded Data   : 0000
Decoding failed.
PS E:\6th Semester\Data-Communication lab\code>
                                                              Ln 33, Co
```

7.Conclusion:

This lab explored the implementation of **Differential Manchester Encoding**, a fundamental concept in digital communication. The approach enhances synchronization and is widely used in standards like IEEE 802.5 (Token Ring). The Java implementation reinforced understanding of signal transitions, timing, and binary logic.