DAYANANDA SAGAR COLLEGE OF ENGINEERING

(An Autonomous Institute affiliated to Visvesvaraya Technological University (VTU), Belagavi,
Approved by AICTE and UGC, Accredited by NAAC with 'A' grade & ISO 9001 – 2015 Certified Institution)
Shavige Malleshwara Hills, Kumaraswamy Layout, Bengaluru-560 111, India

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

Open ended experiment Report submitted for the subject

Digital Communication Systems – 22EC53

Submitted by

Nagalakshmi S (1DS22EC138)
Naina R Bhandari (1DS22EC140)
Nandani Kumari (1DS22EC141)

Under the Guidance of
Prof. Navya Holla K
Prof. Aishwarya N

Department of Electronics & Communication Engineering
DSCE, Bengaluru

Evaluation

USN	Name		Simulation & Analysis -10 Marks	Presentation &Report -10 Marks
1DS22EC138	Nagalakshmi S			
1DS22EC140	Naina R Bhandari			
1DS22EC141	Nandani Kumari			

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
JNANASANGAMA, BELAGAVI-590018, KARNATAKA, INDIA
2024-25

DAYANANDA SAGAR COLLEGE OF ENGINEERING

(An Autonomous Institute affiliated to Visvesvaraya Technological University (VTU), Belagavi,
Approved by AICTE and UGC, Accredited by NAAC with 'A' grade & ISO 9001 – 2015 Certified Institution)
Shavige Malleshwara Hills, Kumaraswamy Layout, Bengaluru-560 111, India

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that open ended experiment entitled **"Linear block code encoder and decoder"** as part of **Digital Communication Systems – 22EC53** is a bonafide work carried out by Nagalakshmi S (1DS22EC138), Naina R Bhandari (1DS22EC140), Nandani Kumari(1DS22EC141) ,as 30-marks component in partial fulfillment for the 5th semester of Bachelor of Engineering in Electronics and Communication Engineering of the Visvesvaraya Technological University, Belagavi during the year 2024-2025. The open ended report report has been approved as it satisfies the academic requirements prescribed for the Bachelor of Engineering degree.

Signature of Faculty	**Signature of HOD**
Prof. Navya Holla K	**Dr. Shobha K R**
Prof. Aishwarya N	

DAYANANDA SAGAR COLLEGE OF ENGINEERING

(An Autonomous Institute affiliated to Visvesvaraya Technological University (VTU), Belagavi,
Approved by AICTE and UGC, Accredited by NAAC with ‘A’ grade & ISO 9001 – 2015 Certified Institution)
Shavige Malleshwara Hills, Kumaraswamy Layout, Bengaluru-560 111

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

DECLARATION

We declare that we abide by the ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. The work submitted in this report of **Digital Communication Systems – 22EC53**, V Semester BE, ECE has been compiled by referring to the relevant online and offline resources to the best of our understanding and in partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Electronics and Communication Engineering, at Dayananda Sagar College of Engineering, an autonomous institution affiliated to VTU, Belagavi during the academic year 2024-2025.We hereby declare that the same has not been submitted in part or full for other academic purposes.

(Nagalakshmi S -1DS22EC138)
(Naina R Bhandari -1DS22EC140)
(Nandani Kumari -1DS22EC141)

Place: Bengaluru

Date:29/11/2024

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of this project.

We take this opportunity to express our sincere gratitude to Dayananda Sagar College of Engineering for giving us the opportunity to pursue our Bachelor's Degrees in this institution.

In particular, we would like to thank Dr. B G Prasad, Principal of Dayananda Sagar College of Engineering, for his constant encouragement and advice.

We would like to express my gratitude to Dr. Shobha K R, Professor and HoD, Department of Electronics and Communication Engineering, Dayananda Sagar College of Engineering, for her motivation and invaluable support throughout the development of this project.

Nagalakshmi S-1DS22EC138

Naina R Bhandari-1DS22EC140

Nandani Kumari-1DS22EC141

1.**AIM**: Designing a " linear block encoder and decoder" using c programming .

2. **INTRODUCTION**:

A block code is an error-correcting code where a fixed-length block of message bits is transformed into a longer block of codeword bits. The primary goal of this transformation is to introduce redundancy into the original message so that errors introduced during transmission or storage can be detected and corrected.

For a given , a block code maps a sequence of k-bit message vectors into n-bit codeword vectors, where n≥kn \geq kn≥k. The difference n−kn - kn−k represents the redundant bits that allow for error detection and correction.

A linear block code is a special type of block code where the encoding and decoding operations are based on linear algebra over a finite field. The defining feature of linear block codes is that their codewords form a vector space, meaning that the sum of any two codewords is also a valid codeword.

The linearity property ensures that the encoding process can be performed using matrix operations, and the decoding process relies on algebraic methods for detecting and correcting errors.

Linear block codes are defined by two key parameters:

- **n:**The length of the codeword (number of bits in the codeword).
- **k:**The length of the message (number of information bits in the message).

3. **CODE :**

```
#include <stdio.h>

#define K 4

#define N 7

#define M (N - K)

void matrix_multiply_mod2(int message[], int matrix[][N], int result[]) {

    for (int i = 0; i < N; i++) {

        result[i] = 0; // Initialize result to 0

        for (int j = 0; j < K; j++) {

            result[i] ^= (message[j] & matrix[j][i]); // XOR operation   }}}

    void calculate_syndrome(int received[], int H[][N], int syndrome[]) {

    for (int i = 0; i < M; i++) {

        syndrome[i] = 0; // Initialize syndrome to 0

        for (int j = 0; j < N; j++) {

            syndrome[i] ^= (received[j] & H[i][j]); // XOR operation}}}

int main() {

    int G[4][7] = {

        {1, 0, 0, 0, 1, 1, 1}, // Row 1

        {0, 1, 0, 0, 1, 1, 0}, // Row 2

        {0, 0, 1, 0, 1, 0, 1}, // Row 3

        {0, 0, 0, 1, 0, 1, 1}  // Row 4

    };



    int message[K] = {1, 0, 1, 0};
```

```
int codeword[N] = {0}; // To store the encoded codeword
matrix_multiply_mod2(message, G, codeword);
printf("Generated Codeword: ");
for (int i = 0; i < N; i++) {
    printf("%d ", codeword[i]);
}
printf("\n");


int error_vector[N] = {0, 0, 1, 0, 0, 0, 0}; // Error in the 3rd bit
int received_codeword[N]; // Received codeword with error


for (int i = 0; i < N; i++) {
    received_codeword[i] = (codeword[i] ^ error_vector[i]) % 2; // Add error to the codeword
}
printf("Received Codeword (with Error): ");
for (int i = 0; i < N; i++) {
    printf("%d ", received_codeword[i]);
}
printf("\n");
int H[3][7] = {
    {1, 1, 1, 0, 1, 0, 0}, // Row 1
    {1, 1, 0, 1, 0, 1, 0}, // Row 2
    {1, 0, 1, 1, 0, 0, 1}  // Row 3
};
```

```
    int syndrome[M]; // To store the syndrome

    calculate_syndrome(received_codeword, H, syndrome);

    printf("Syndrome: ");

    for (int i = 0; i < M; i++) {

        printf("%d ", syndrome[i]);

    }

    printf("\n");

    int error_detected = 0;

    for (int i = 0; i < M; i++) {

        if (syndrome[i] != 0) {

            error_detected = 1;

            break;

        }}

   if (error_detected) {

        printf("Error detected in the received codeword.\n");{ else {

        printf("No error detected.\n"); }

    }return 0; }
```

4. **SIMULATION OUTPUT**:

Generated Codeword: 1 0 1 0 1 1 1

Received Codeword (with Error): 1 0 0 0 1 1 1

Syndrome: 1 0 1

Error detected in the received codeword.

$C = DG$

$m = [1\ 0\ 1\ 0]$

$$G = \left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

$$= [1\ 0\ 1\ 0]\left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

$$= [d_1, d_2, d_3, d_4, d_1 \oplus d_2 \oplus d_3, d_1 \oplus d_2 \oplus d_4, d_1 \oplus d_3 \oplus d_4]$$

$C = [1\,0\,1\,0\,0\,1\,0]$

$S = RH^T$

$$H = \left[\begin{array}{cccc|ccc} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$

$$S = [1\,0\,0\,0\,0\,1\,0]\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$= [1\,0\,1]$

error in 3^{rd} bit

$\therefore$ corrected vector

$C = 1010010$

6. **ADVANTAGES AND APPLICATIONS**:

Advantages of Linear Block Code Encoder and Decoder

1. **Error Detection and Correction**: Linear block codes are widely used for detecting and correcting errors in transmitted data. By adding redundant bits, they enable the receiver to detect and correct errors in the received message, improving data integrity.
2. **Efficiency**: Linear block codes are computationally efficient to implement. The encoding and decoding processes generally require fewer operations compared to other types of error-correcting codes, such as turbo codes or LDPC (Low-Density Parity-Check) codes.
3. **Simple Structure**: The structure of a linear block code is relatively simple. It is based on matrix operations, such as multiplication of message vectors with a generator matrix, making it easy to implement and understand.
4. **Error-correcting Capability**: These codes can correct errors in data transmission. The number of errors that can be corrected depends on the minimum distance of the code, which is a key parameter.

Applications of Linear Block Code Encoder and Decoder

1. **Communication Systems**: Linear block codes are commonly used in communication systems (e.g., satellite communication, wireless networks, optical fiber, and cellular networks) to ensure reliable data transmission. They help to detect and correct errors caused by noise, interference, or signal degradation.
2. **Data Storage**: In data storage systems, such as hard drives, CDs, DVDs, and solid-state drives (SSDs), linear block codes are used for error correction during data reading and writing. This ensures that data can be accurately retrieved even if some bits are corrupted.
3. **Deep Space Communication**: Space missions often use linear block codes (such as Reed-Solomon codes) to correct errors that occur during the transmission of data over long distances, where the signal is weak and prone to interference.
4. I**nternet of Things (IoT)**: IoT systems use linear block codes in their communication protocols to ensure that the data sent between devices is error-free, even under adverse conditions like weak signals or network congestion.

7.**CONCLUSION**:

Linear block codes are essential for error detection and correction, offering simplicity, efficiency, and strong error-correcting capabilities. They ensure reliable data transmission and storage by adding redundancy to handle noise and disturbances. Widely used in communication systems, data storage, and broadcasting, they maintain data integrity. The encoder and decoder processes are straightforward, making them easy to implement. Despite newer coding schemes, linear block codes remain vital due to their balance of performance and simplicity.