

**INDIAN INSTITUTE OF INFORMATION TECHNOLOGY NAGPUR**

Name: Uma

Roll Number: BT20ECE044

Semester: 6

Course: Coding Techniques

Submitted To: Dr.Rashmi Pandhare

Assistant Professor

Department of Electronics and Communication Engineering

**Lab Report**

Data Encoding and Decoding by Huffman Coding Technique

**Aim**: To Implement Encoding and Decoding of Data using Huffman Coding in Matlab.

**Software used**: Matlab

**Theory:**

* Error-correcting codes are the way to deal with the errors introduced in the actual message signal at the time of transmission in data communication. It is regarded as an error detection and correction technique by which the information signal is encoded using redundant bits. These are categorized as Block Code and Convolutional Code.
* The major elements of the convolutional coding technique include the shift register that acts as temporary storage and whose stored bits undergo shifting using a sliding window and a logic circuit that performs modulo-2 addition incorporating the X-OR function.

* Basically, there are mainly two parameters that define the convolutional coding which is as follows:
* **Constraint length**: The constraint length corresponds to the length of the convolutional encoder i.e., the overall window size in bits, within the shift register. It is denoted by K (uppercase). Sometimes also denoted by L as it might cause confusion with k (lowercase). There is another parameter ‘m’ which corresponds to the number of input bits retained within the shift register once it is entered in the encoder.
* **Code rate:** Code rate is the ratio of the number of bits shifted at once within the shift register (denoted by k) to the total number of bits in an encoded (generated)

bitstream (denoted by n). Thus, it is given as k/n

Implementation of Huffman Coding:

1. List the symbols and sort probabilities per symbol

2. Combine the lowest two probabilities of symbols and label the new code with it

3. A newly created item is given priority and placed at the highest position in the sorted list

4. Repeat step 2 until only one node remains

5. Assign code 0 to higher up symbol and 1 to lower down symbol

6. Now trace the code symbols going backwards.

**Code:**

K =3;

G1=7;

G2=5;

msg = [1 1 0 0 1 0];

trellis = poly2trellis(K,[G1,G2]);

coded = convenc(msg,trellis);

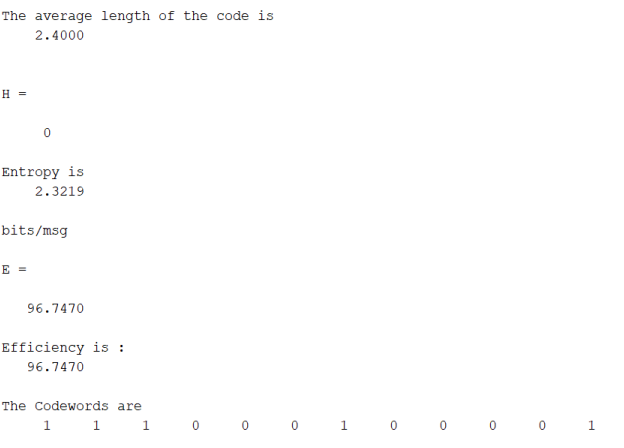
tblen=length(msg);

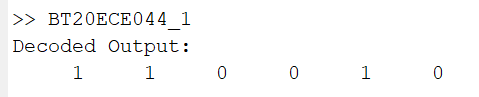
decoded = vitdec(coded,trellis,tblen,'trunc','hard');

disp('Decoded Output:');

disp(decoded);

**Output:**





**Conclusion** :The Huffman coding algorithm is succesfully implemented in MATLAB.It is observed that the compression ratio achieved by Huffman coding depends on the frequency distribution of the symbols in the data. Overall, Huffman coding is a powerful tool for reducing the size of data and has practical applications in data compression and transmission over networks.

**Lab Report**

Encoding and Decoding on the data sample by Linear Block Coding using Matlab

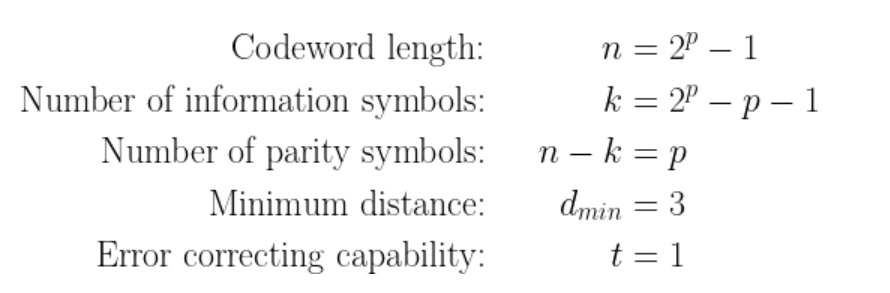
**Aim**: To perform Encoding and Decoding on a by Linear Block Coding using Matlab.

**Software used**: Matlab

**Theory:**

A linear block code is generally an (n,k) linear block code in which k bits are identical to the message sequence and the remaining n-k number of bits are called generalized parity check bits or simply parity bits.

The characteristics of a generic (n,k) Hamming code are given below.

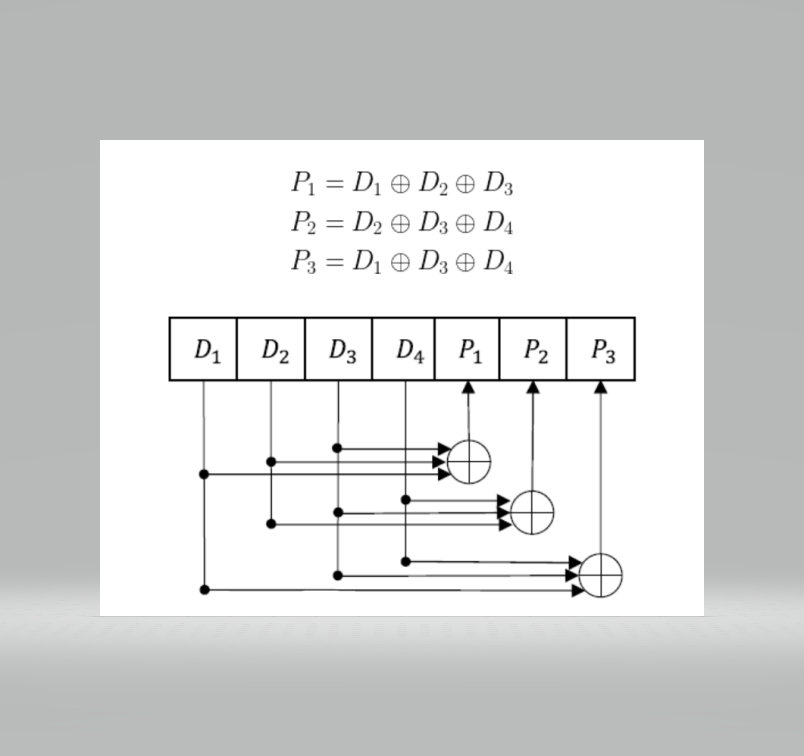


The (7,4) binary Hamming block encoder accepts blocks of 4-bit of information, adds 3 parity bits to each such block and produces 7-bits wide Hamming coded blocks.Let a codeword belonging to (7, 4) Hamming code be represented by [D7,D6,D5,P4,D3,P2,P1], where D represents information bits and P represents parity bits at respective bit positions. The subscripts indicate the left to right position taken by the data and the parity bits.

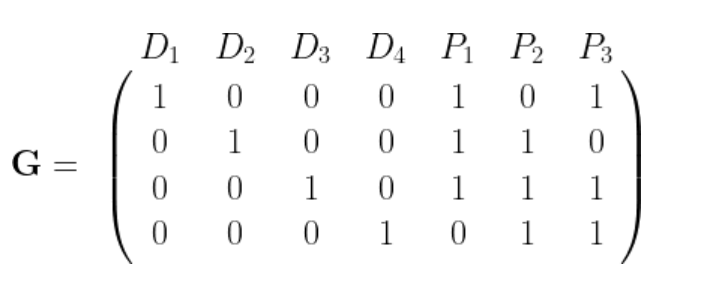
**Encoding process:**

the parity bits are calculated from the following linearly independent equations using modulo-2 additions.

For example,



At the transmitter side, a Hamming encoder implements a generator matrix G. It is easier to construct the generator matrix from the linear equations listed in the equation above. The linear equations show that the information bit D1 influences the calculation of parities at P1 and P3. Similarly, the information bit D2 influences P1 and P2, D3 influences P1, P2 & P3 and D4 influences P2 & P3.



Represented as matrix operations, the encoder accepts a ***4-bit*** message block \mathbf{m}, multiplies it with the generator matrix, and generates ***7-bit*** codewords \mathbf{c}. Note that all the operations (addition, multiplication, etc.,) are in the modulo-2 domain.

Given a generator matrix, the Matlab code snippet for generating a codebook containing all possible codewords is given below. The resulting codebook \mathbf{C} can be used as a Look-Up-Table (LUT) when implementing the encoder. This implementation will avoid repeated multiplication of the input blocks and the generator matrix.

**Code:**

**% G Matrix**

**G = [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]**

**k = 4;n = 7;**

**for i = 1 : 2^k**

**for j = k: -1: 1**

**if rem(i - 1, 2 ^ (-j + k + 1)) >= 2 ^ (-j + k)**

**u(i, j) = 1;**

**else**

**u(i, j) = 0;**

**end**

**echo off;**

**end**

**end**

**echo on**

**% CodeWords**

**c = rem(u \* G, 2)**

**w\_min = min(sum((c(1 : 3, :))'))**

**r = [0 0 0 1 0 0 0];**

**% G Matrix**

**H= [1 0 1 1 1 0 0;**

**1 1 0 1 0 1 0;**

**0 1 1 1 0 0 1]**

**ht = transpose(H)**

**s = rem(r \* ht, 2)**

**for i = 1 : 1 : size(ht)**

**if(ht(i,1:3)==s)**

**r(i) = 1-r(i);**

**break;**

**end**

**end**

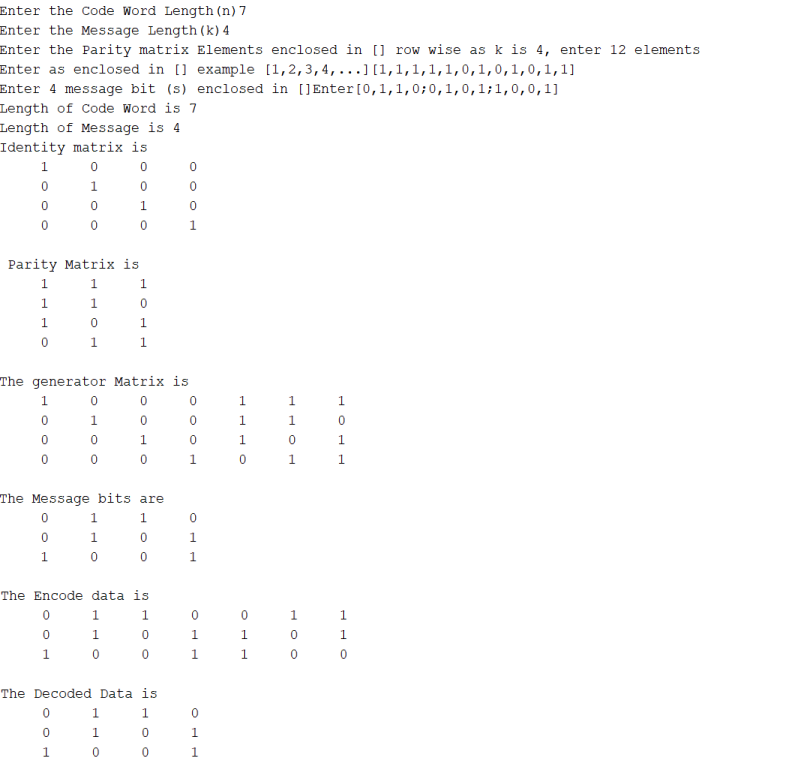
**disp('The Error is in bit:')**

**disp(i)**

**disp('The Corrected Codeword is :')**

**disp(r)**

**Output:**



**Conclusion** : Linear Block Encoding and Decoding is a fundamental concept in Information Theory that allows reliable and efficient transmission of information. In this lab, we have successfully implemented a Matlab code for Linear Block Encoding and Decoding, which can be applied to various communication systems. By using this code, we have demonstrated the effectiveness of Linear Block Encoding and Decoding in correcting errors in transmitted data, ensuring that the receiver obtains the correct message even in the presence of noise or other disturbances.

**Case Study**

JPEG Image Compression and Decompression by Huffman Coding

Authors:

* Surbhi Singh

M.Tech Scholar Department of Electronics and communication Suresh Gyan Vihar University, Jaipur, Raj, India

* Vipin Kumar Gupta

Assistant Professor of Electronics and communication Suresh Gyan Vihar University Jaipur, Raj, India

Year of Publication: August 2016

Source: International Journal of Innovative Science and Research Technology

Volume 1, Issue 5, August – 2016

ISSN No: - 2456- 2165

Abstract:

Image compression is considered an application performed for the compression of data in digital format images. Digital images are comprised of a large amount of information that requires bigger bandwidth. Image compression techniques can be generally categorized into two types: lossless & lossy techniques. DCT (discrete cosine transform) can also be used for compressing an image and also approaches like Huffman encoding, and quantization & such steps are required for the compression of images with JPEG format. The format JPEG can be used for both RGB (colored) & YUV (grayscale) images.

The DCT is considered to be a mathematical function that will transform an image of digital format from spatial to the frequency domain. It is very much easy to implement Huffman encoding & decoding to minimize memory complexity. In this technique, the analog image pixels are transformed into discrete image pixels, and therefore compression is performed. On the receiving side, the pixels are decompressed to obtain the actual image. The PSNR(PSNR is considered a majorly used quality measuring parameter in sections of image compression. ) is computed for analyzing the quality of the image.

The general mode using DCT is the JPEG baseline coding system. JPEG compression will minimize the size of the file through minimal image degradation by eliminating the least required information. This method eliminates the information having higher frequency i.e. sharp transitions of intensity & color hue. The process for minimizing the information in the transform domain is termed quantization. Though, it is referred to as a lossy image compression method as the final image & actual image are not similar. The information of lossy compression will be lost while missed is affordable. Sequential steps will be performed for JPEG compression.

Huffman coding is considered a lossless data compression algorithm. The motive behind this is allocating variable-length codes for inputting characters; the length of allocated codes is constituted over the frequency of associated characters. The most frequently used character will be having smallest code & least frequent character will be having biggest code. Huffman coding is considered some of the most prominent techniques for the elimination of redundancy in coding. It is been implemented in several compression algorithms, incorporating image compression. It makes use of statistical characteristics of alphabets in a source stream & further generates associated codes for such alphabets. These codes have a variable code length while making use of an integral number of bits. The alphabetical codes processing higher probability for occurrence has short length than the codes for alphabets possessing lesser probability. Hence it is considered over the frequency of occurrence of a data item (pixels or small blocks of pixels in images).

It requires a lesser number of bits for encoding frequency-used information. The codes will be accumulated in a code book. A code book will be made for every image or set of images. Huffman coding is considered the most optimal lossless schema for the compression of a bit stream. It operates by firstly making calculations of probabilities. Defining permutations {0,1} n by allocating symbols, termed as A, B, C, D. The bit stream may seem as AADAC.

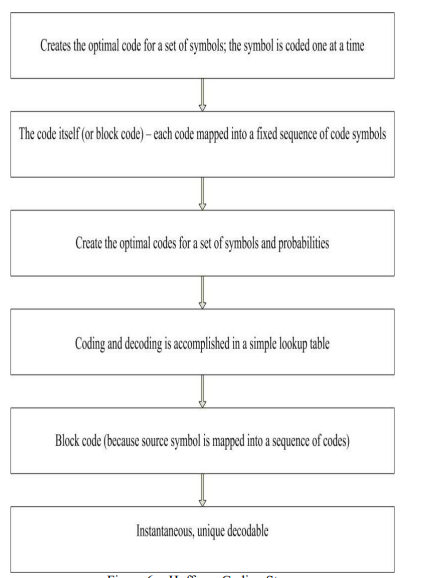
Now the symbols are allocated newer codes, the higher will be the probability, the lower will be the number of bits in the code. These codes serve as an outcome of the Huffman coder in the form of the bit stream.

Now stopping point of the code must be known & point for starting a new code. This problem is solved through the enforcement of a unique prefix condition: no code is a prefix of any other code. The initial codes are referred to as 01; 11; 001; 101; 0000; 1000l; 1001. In the Huffman coding schema, shorter codes are allotted to the symbols that are incorporated on a frequent basis & longer codes to those which seem to occur less frequently.

Huffman Coding Algorithm works as a bottom-up approach. Algorithm Steps of the Huffman Coding algorithm are shown below:

Generating a series of source reductions: combining two minimal probability symbols to a single symbol; it is repeated till a minimized source having two symbols is obtained.

Coding every reduced symbol: starting from the smallest source & coming back to the actual source.



**Conclusion:**

Image compression is an important technique for reducing the size of the image and for sending it in low size. Image has many types of this lossy compression that is widely used in network-related applications. The applications of lossless compression are file image/video/audio compression, big-size file data zip, and much more.

References:

[1] L. Agostini, S. Bampi,

“Pipelined Fast 2

-D DCT Architecture for

JPEG Image Compression” Proceedings of the 14th Annual

 Symposium on Integrated Circuits and Systems Design, Pirenopolis,Brazil. IEEE Computer Society 2001. pp 226-231.

[2] Y. Arai, T. Agui, M. Nakajima. “

A Fast DCT-SQ Scheme for

Images”. Transactions of IEICE, vol. E71, nÂ. 11, 1988, pp.

1095-1097.

[ 3 ] D. Trang, N. Bihn, “A High

-Accuracy and High-Speed 2-D 8x8

Discrete Cosine Transform Design”. Proceedings of ICGCRCICT

2010, vol. 1, 2010, pp. 135-138

[4] I. Basri,

B. Sutopo, “Implementasi 1D

-DCT Algoritma Feig-Winograd di FPGA Spartan-

3E (Indonesian)”. Proceedings of

CITEE2009, vol. 1, 2009, pp. 198-203

[5] E. Magli, “The JPEG Family of Coding Standard,”

Part of

“Document and Image Compression”, New York: Taylor

and Francis,2004.

[6] Wallace, G. K. ,''The JPEG Still Picture Compression Standard'',Communications of the ACM, Vol. 34, Issue 4, pp.30-44. 1991.

[7] Sun, M., Ting C., and Albert M., ‘‘

VLSI Implementation of a 16 X

16 Discrete Cosine Transform’’, IEEE T

ransactions on Circuits andSystems, Vol. 36, No. 4, April 1989.

[8] Xilinx, Inc., “Spartan

3E FPGA Family : Data Sheet ”, Xilinx

Corporation, 2009.

[9] Omnivision, Inc., “OV9620/9120 Camera Chip Data Sheet ”, Xilinx

Corporation, 2002.

[10] Xilinx, Inc., “2D Discrete Cosine Transform (DCT) V2.0 ”

,Logicore Product Specification, Xilinx Corporation, 2002. InternationalJournal of Innovations in Engineering and Technology (IJIET) Vol.

[11] K. Deepthi: “Design and Implementation of JPEG Image

Compression and D

ecompression”. International Journal of Innovations

in Engineering and Technology (IJIET) CVSR College of Engineering,Hyderabad, Andhra Pradesh, India Vol. 2 Issue 1 February (2013).

[12] Deepak Singla and Rupali Syal: “Data Security Using LSB &DCT Steganography in Images”. Deepak Singla, Rupali Syal

/International Journal Of Computational Engineering Research/ ISSN:2250

–

3005. IJCER | Mar-Apr 2012 | Vol. 2 | Issue No.2 |359-364.

[13] Lei Wang, Jiaji Wu, Licheng Jiao, Li Zhang and Guangming Shi:

“Lossy to Lossless Image Compression based on Reversible IntegerDCT”. Institute of Intelligent Information Processing, Xidian

University, Xi'an 710071, P.R.China 978-1-4244-1764-3/08/$25.00©(2008) IEEE.

[14] Dipti Bhatnagar and Sumit Budhiraja proposed that ImageCompression using DCT based Compressive Sensing and VectorQuantization. International Journal of Computer Applications (0975 8887) Volume 50

No.20, July (2012).

[15] Kiran Bindu,Anita Ganpati and Aman Kumar Sharma:”AComparative Study of Image Compression Algorithms”. InternationalJournal of Research in Computer Science eISSN 2249-8265 Volume 2Issue 5 (2012) pp. 37-42.

[16] Radha Krishna A.N and G.Shruthi: “ JPEG Encoder using

DiscreteCosine Transform & Inverse Discrete Cosine Transform”. IOSR

Journal of Electronics and Communication Engineering (IOSR-JECE)e-ISSN: 2278-2834,p- ISSN: 2278-8735. Volume 5, Issue 4 (Mar. -Apr. 2013), PP 51-56

[17] Ryuji Matsuoka, Mitsuo Sone, Kiyonari Fukue, Kohei Cho andHaruhisa Shimoda proposed that Quantitative Analysis of ImageQuality of Lossy Compression Images. Tokai University Research &Information Center 2-28-4 Tomigaya, Shibuya-ku, Tokyo 151-0063,JAPAN

**Case Study**

A Brief Introduction on Shannon’s

Information Theory

Author:

* Ricky Xiaofeng Chen

Biocomplexity Institute and Dept. of Mathematics, Virginia Tech, 1015 Life Science Circle, Blacks-burg, VA 24061, USA.

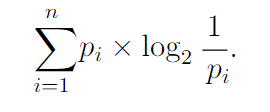
Year of Publication: July and October of 1948

Abstract:

Shannon's Information Theory is a mathematical framework for understanding the fundamental concepts of information, communication, and data transmission. It is based on the idea that information can be quantified and measured in terms of its entropy or uncertainty.

Communication is the exchange of information. A sentence or sentences in English can be viewed as a sequence of letters (‘a’, ‘b’, ‘c’,.. . )and symbols (‘,’,‘.’,‘ ’,. . . ). So, we can just think of sentences conveying diﬀerent meaning as diﬀerent sequences.

Consider K sequences() in total, and all sequences appear equally likely. Assume they encode diﬀerent messages. Regardless of speciﬁc messages they encode, we regard them as having the same amount of information. Let’s just employ the number of bits needed to encode a sequence to count the amount of information a sequence encodes (or can provide). It only depends on the probability distribution of these letters/information present in the sequence which is given by entropy. The entropy could be given by the formula :

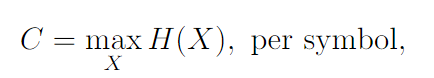


It is clearly the expectation (i.e., average in the sense of probability) of the quantity log2(1/pi) associated with the letter xi, for 1 ≤i≤n. This matches the term “average” so that we can deﬁne the amount of information a letter xi with probability pi has to be log2(1/pi) bits. In this deﬁnition of information, we observe that if a letter has a higher probability it has less information and vice versa.

In a communication system, we have three basic ingredients: the source, the destination, and the media between them. We call the media the (communication) channel. A channel could be in any form. It could be physical wires, cables, open environments in wireless communication, antennas, and certain combinations of them. This theory also introduces the concept of channel capacity, which defines the maximum rate at which information can be transmitted through a communication channel with a given level of noise or interference

in a ﬁxed time duration, e.g., per second, or per symbol(time).

Given a channel and a set A of letters (or symbols) that can be transmitted via the channel.

For an errorless channel, the maximum amount of information that can be received at the destination equals the maximum amount of information that can be generated at the source.

where X ranges over all possible distributions on A.

A channel with error means that the source generated a letter Xi∈A and transmitted it to the destination via the channel, with some unpredictable error, the received letter at the destination may be Xj. Assume statistically, Xj is received with probability p(Xj|Xi) when Xi is transmitted probabilities are called transit probabilities of the channel.

One of the key insights of Shannon's Information Theory is that redundancy in communication can actually improve the reliability of the transmission. This is because redundant information can be used to correct errors that may occur during transmission.

**Conclusion:**

The theory has had a significant impact on a wide range of fields, including digital communication, computer science, cryptography, and statistics. It has also played a major role in the development of modern technologies such as the internet, wireless communication, and digital data storage.

Overall, Shannon's Information Theory provides a powerful and flexible framework for understanding and optimizing the transmission of information in a wide range of contexts.

References:

[1] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J., vol. 27, pp. 379-423, 623-656, July-Oct. 1948.

[2] C.E. Shannon, Communication in the presence of noise, Proc. IRE, 37 (1949), 10-21.

[3] S. Verd´u, Fifty years of Shannon theory, IEEE Transactions On Information Theory,

44 (1998), 2057-2078

**Lab Report**

Syndrome Decoding for Linear Block Coding using Matlab

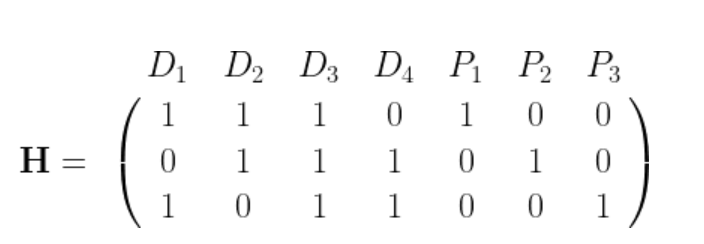
**Aim**: To Implement Syndrome Decoding for Linear Block Coding using Matlab.

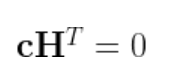
**Software used**: Matlab

**Theory:**

The decoding of linear block codes is done by using a special technique called syndrome decoding. This kind reduces the memory requirement of the decoder to a great extent. Syndrome decoding is a powerful and effective technique for correcting errors in linear block codes, and is widely used in various communication systems

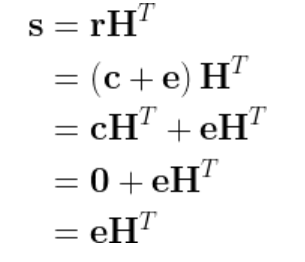
The three check equations for the given generator matrix G for the sample (7,4) Hamming code, can be expressed collectively as a parity check matrix – H. The parity check matrix finds its usefulness on the receiver side for error detection and error-correction.



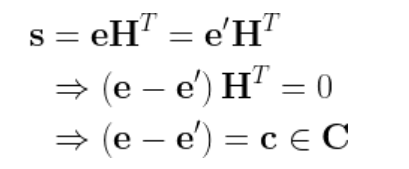
According to the parity-check theorem, for every generator matrix G, there exists a parity-check matrix H, that spans the null space of G. Therefore, if c is a valid codeword, then it will be orthogonal to each row of H.

Therefore, if H is the parity-check matrix for a codebook C, then a vector c in the received code space is a valid codeword if and only if it satisfies cH^T=0

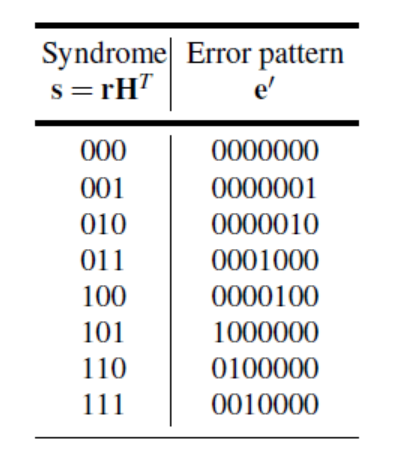
Therefore, if H is the parity-check matrix for a codebook C, then a vector c in the received code space is a valid codeword if and only if it satisfies cH^T=0

Consider a vector of the received word r=c+e, where c is a valid codeword transmitted and e is the error introduced by the channel. The matrix product rH^T is defined as the syndrome for the received vector r, which can be thought of as a linear transformation whose null 

space is C [2].

Thus, the syndrome is independent of the transmitted codeword \mathbf{c} and is solely a function of the error pattern \mathbf{e}. It can be determined that if two error vectors \mathbf{e} and \mathbf{e}' have the same syndrome, then the error vectors must differ by a nonzero codeword.

It follows from the equation above, that decoding can be performed by computing the syndrome of the received word, finding the corresponding error pattern and subtracting (equivalent to addition in GF(2) domain) the error pattern from the received word. This obviates the need to store all the vectors as in a standard array decoding and greatly reduces the memory requirements for implementing the decoder.

Following is the syndrome table for the ***(7,4)*** Hamming code example, illustrated here.

**Code:**

n = i n p u t ( ’ E n t e r t h e CodeWord Length ’ ) ;

%Message Length

k = i n p u t ( ’ E n t e r Number of Message B i t s ’ ) ;

%G e n e r a t o r Ma t r i x

G = i n p u t ( ’ E n t e r G e n e r a t o r Ma t r i x ’ ) ; f o r i = 1 : 2^ k

f o r j = k : −1 : 1

i f rem ( i −1 , 2^( − j +k + 1 ) ) >= 2^( − j +k ) M( i , j ) = 1 ;

e l s e

M( i , j ) = 0 ;

end end

end

d i s p ( ’ Message Ma t r i x ’ ) ; d i s p (M) ;

EncodeMsg = encode (M, n , k , ’ l i n e a r ’ , G) ; d i s p ( ’ Encoded Messages ’ ) ;

d i s p ( EncodeMsg ) ; P = hammgen ( n−k ) ; E = s y n d t a b l e ( P ) ;

d i s p ( ’ E r r o r Ma t r i x ’ ) ; d i s p ( E ) ;

Par = G ( : , k + 1 : n ) ; P = ( Par ) ’ ;

d i s p ( ’ P a r i t y Ma t r i x ’ ) ; d i s p ( P ) ;

I = eye ( k − 1 ) ; H = [ P I ] ;

h = H’ ;

S = mod ( E\*h , 2 ) ;

d i s p ( ’ Syndrome Table ’ ) ;

d i s p ( S ) ;

decodeMsg = decode ( EncodeMsg , n , k , ’ l i n e a r ’ , G) ; d i s p ( ’ Decoding Messages ’ ) ;

d i s p ( decodeMsg ) ; HammDis = z e r o s ( 2 ^ k , 1 ) ; f o r i = 1 : 2 ^ k

f o r j = 1 : n

i f EncodeMsg ( i , j ) == 1

HammDis ( i ) = HammDis ( i ) + 1 ; end

end end

s o r t ( HammDis ) ;

i f HammDis ( 1 ) == 0 dmin = HammDis ( 2 ) ; e l s e

dmin = HammDis ( 1 ) ; e n d i f

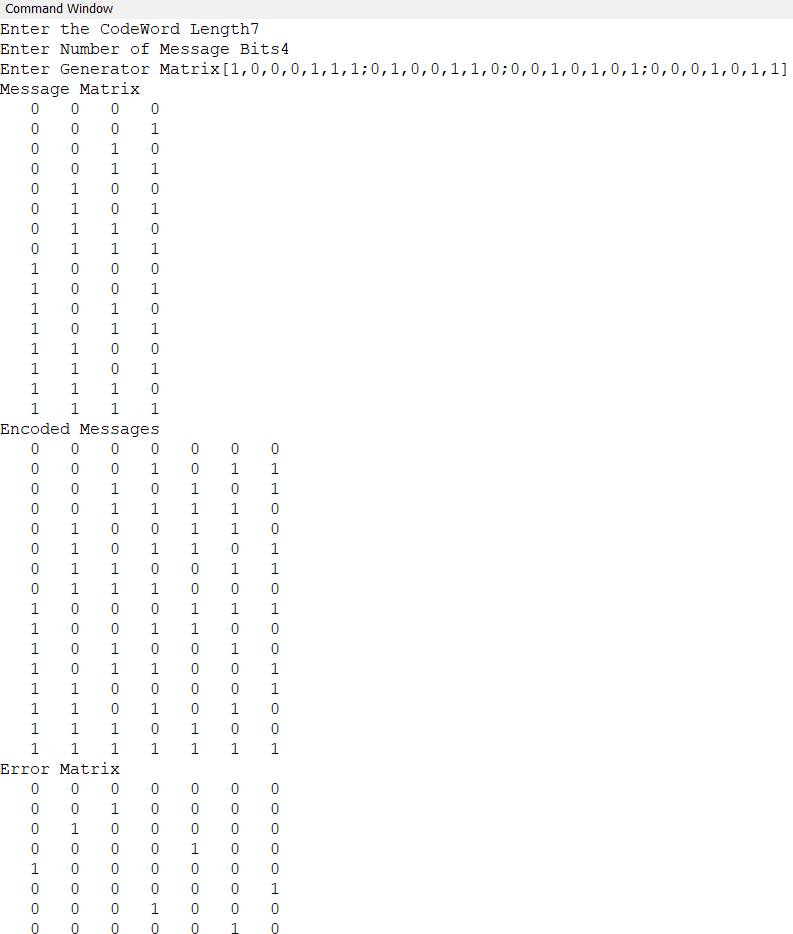
d i s p ( ” Minimum Hamming D i s t a n c e i s ” ) ; d i s p ( dmin ) ;

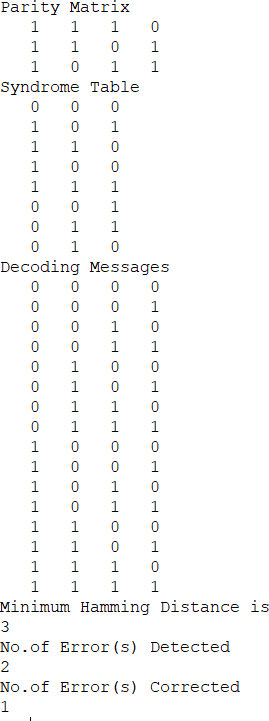
E r r D e t = dmin − 1 ; Err Cor = E r r De t / 2 ;

d i s p ( ” No . Of E r r o r ( s ) D e t e c t e d ” ) ; d i s p ( E r r D e t )

d i s p ( ” No . of E r r o r ( s ) C o r r e c t e d ” ) ; d i s p ( Err Cor ) ;

**Output:**





**Conclusion:**

The Syndrome Decoding for Linear Block Codesis succesfully implemented in MATLAB and its effectiveness on a sample data set is demonstrated. Syndrome decoding is a powerful technique for cor recting errors in Linear Block codes. It works by identifying the error pattern in the received message and using this information to correct the errors. Syndrome decoding is an important technique in the field of error correction coding, with many practical applications in communications and data storage.

**Lab Report**

# Cyclic block encoding using Matlab

**Aim**: To Implement cyclic Block Encoding using Matlab.

**Software used**: Matlab

**Theory:** In coding theory, a **cyclic code** is a block code, where the [circular](https://en.wikipedia.org/wiki/Circular_shift) shifts of each codeword gives another word that belongs to the code. They are [error-correcting codes](https://en.wikipedia.org/wiki/Error-correcting_codes) that have algebraic properties that are convenient for efficient [error detection and correction](https://en.wikipedia.org/wiki/Error_detection_and_correction).

**Cyclic Code** is known to be a subclass of linear block codes where cyclic shift in the bits of the codeword results in another codeword. It is quite important as it offers easy implementation and thus finds applications in various systems.

Cyclic codes are widely used in satellite communication as the information sent digitally is encoded and decoded using cyclic coding. These are error-correcting codes where the actual information is sent over the channel by combining with the parity bits.

It has two properties:

**Property 1**: Property of Linearity

According to this property, a linear combination of two codewords must be another codeword.

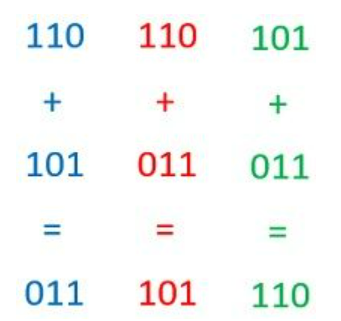
Suppose we have two codewords Ci and Cj. So, on adding

property of linearity

where this Cp must also be a codeword.

For example, suppose we have given 3 codewords (110, 101, 011).

So, according to linearity property, the addition of any of the two given codewords must produce the third codeword. Let’s check



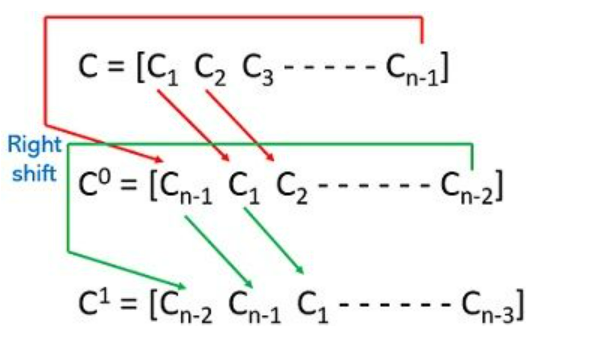
Hence, the given code is linear.

**Property 2**: Property of Cyclic Shifting

According to this property, after a right or left shift in the bits of codewords the resultant code generated must be another codeword.

Suppose, C is a codeword given as:

codeword representation

Then after cyclic shifts

**1**

Here we have performed the right cyclic shift that has produced these codewords.

For example, consider again those 3 codewords (110, 101, 011) which we considered for linearity property.

So, according to cyclic shifting property, an either right or left shift in the bits of a codeword must generate another codeword.

110: shifting the bits towards the right will provide 011.

101: a right shift in the bits of this codeword will give 110.

011: right shifting of these bits will provide 101.

Hence, it is clear that shifting the bits of the codewords gives rise to another codeword thus cyclic shifting property is verified.

Codes that follow both linearity, as well as cyclic shifting, are called **cyclic codes**.

It is to be noted here that if a codeword has all 0’s then it is called a valid codeword. But at the same time, 0 is regarded as a necessary condition and not a sufficient condition.

**Encoding**

**Non-Systematic Cyclic Encoding:**

With message polynomial

m(x) = m0 + m1x + · · · + mk−1xk−1

and generator polynomial g(x), the codeword polynomial is

c(x) = m(x)g(x) = m0g(x) + m1xg(x) + · · · + mk−1xk−1g(x)

**Systematic Cyclic Encoding:**

The equation for determining codeword for systematic code is given as:

codeword equation for systematic code

P(X) represents the parity polynomial and is given by:

parity polynomial equation

So, to construct the systematic codeword first we have to determined P(X).

## **Functions Used:**

## dec2bin(D) returns the binary, or base-2, representation of the decimal integer D. The output argument binStr is a character vector that represents binary digits using the characters 0 and 1.

## cyclpoly(n,k) returns the row vector representing one nontrivial generator polynomial for a cyclic code having codeword length n and message length k.

## Code:

%Code for Cyclic block encoding

clc; clear all; close all;

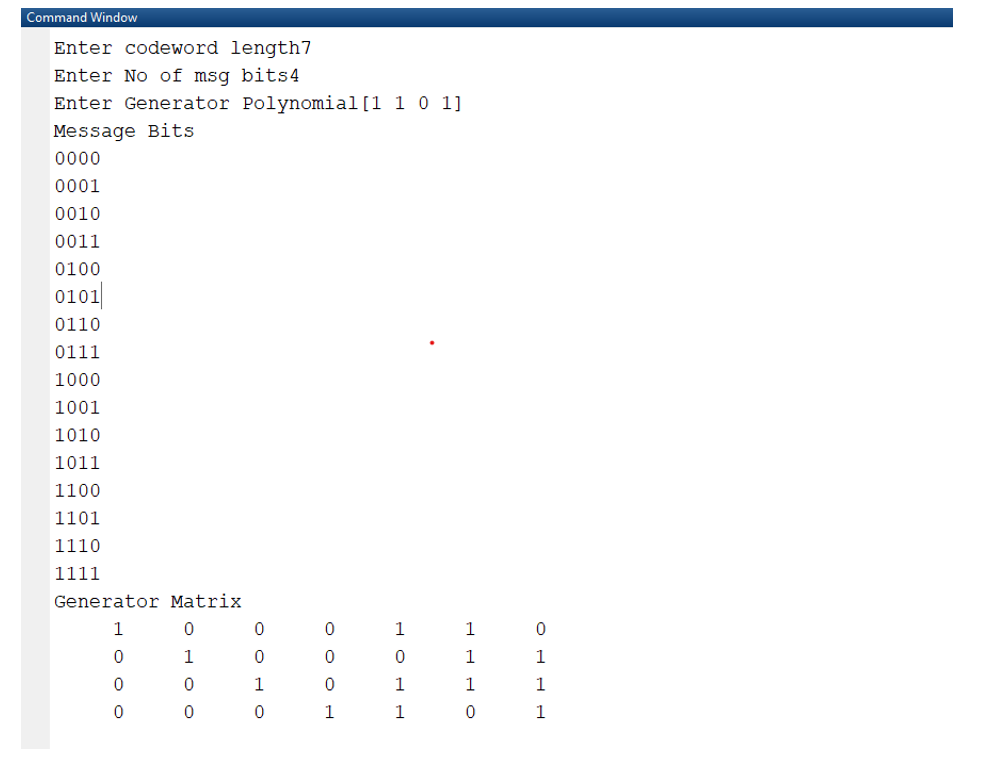
n=input('Enter codeword length'); k=input('Enter No of msg bits'); G=input('Enter Generator Polynomial');

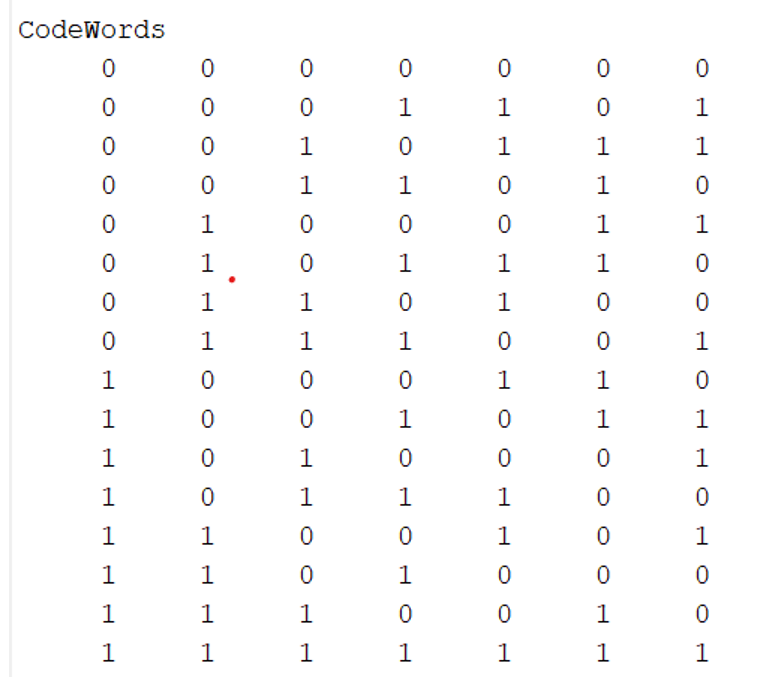
msg=dec2bin(0:2^k-1); %Generating all the Message Bits [i,px]=cyclgen(n,G); %Converting Generator Polynomial to Generator Matrix g=circshift(px,[0,k]); %Circular Shifting the matrix to get correct

%generator Matrix disp("Message Bits"); disp(msg); disp("Generator Matrix"); disp(g);

c=rem(msg\*g,2); disp("CodeWords") disp(c);

## Output:





**Conclusion:**

Cyclic block encoding is performed using in-built functions in Matlab.