**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Title : Implementation of algorithm for decoding of BCH algorithm.**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**CODE:**

clc;clear all;close all;

disp('(n,k)BCH code Encoding Decoding');

n=input('Enter the value of n: ');

k=input('Enter the value of k: ');

gp=bchgenpoly(n,k); %genpoly = bchgenpoly(n,k) returns the narrow-sense generator %polynomial of a BCH code with codeword length n and message length k. The codeword %length n must have the form 2^m-1 for some integer m.

disp('The gp for BCH code is: ');

gp

dmsg=0:1:2^(k)-1;

bmsg=de2bi(dmsg,'left-msb'); %Convert decimal numbers to binary vectors.

disp('The messages to be encoded are as follows: ');

disp(bmsg);

gmsg=gf(bmsg)

code1=bchenc(gmsg,n,k,'end'); %code1=bchenc(msg,n,k) encodes the message in msg using an [n,k] BCH %encoder with the narrow-sense generator polynomial. msg is a Galois

%array of symbols over GF(2).

disp('Apending parity bit at the end of the codeword');

code1

t=bchnumerr(n,k)

disp('Error correcting capacity of BCH code is: ');

disp(t);

%(n,k)

%for all values & corrected codeword

decodemsh=bchdec(code1,n,k); %decoded = bchdec(code,n,k) attempts to decode the received %signal in code using an [n,k] BCH decoder with the narrow-sense generator polynomial.

%for error codeword

codeerr=input('Please enter bit long valid codeword: ');

gcodeerr=gf(codeerr);

%correction of error

codecomsg=bchdec(gcodeerr,n,k);

disp('Corrected codeword is: ');

codecomsg

**OUTPUT:**

(n,k)BCH code Encoding Decoding

Enter the value of n: 15

Enter the value of k: 5

The gp for BCH code is:

gp = GF(2) array.

Array elements =

1 0 1 0 0 1 1 0 1 1 1

The messages to be encoded are as follows:

0 0 0 0 0

0 0 0 0 1

0 0 0 1 0

0 0 0 1 1

0 0 1 0 0

0 0 1 0 1

0 0 1 1 0

0 0 1 1 1

0 1 0 0 0

0 1 0 0 1

0 1 0 1 0

0 1 0 1 1

0 1 1 0 0

0 1 1 0 1

0 1 1 1 0

0 1 1 1 1

1 0 0 0 0

1 0 0 0 1

1 0 0 1 0

1 0 0 1 1

1 0 1 0 0

1 0 1 0 1

1 0 1 1 0

1 0 1 1 1

1 1 0 0 0

1 1 0 0 1

1 1 0 1 0

1 1 0 1 1

1 1 1 0 0

1 1 1 0 1

1 1 1 1 0

1 1 1 1 1

gmsg = GF(2) array.

Array elements =

0 0 0 0 0

0 0 0 0 1

0 0 0 1 0

0 0 0 1 1

0 0 1 0 0

0 0 1 0 1

0 0 1 1 0

0 0 1 1 1

0 1 0 0 0

0 1 0 0 1

0 1 0 1 0

0 1 0 1 1

0 1 1 0 0

0 1 1 0 1

0 1 1 1 0

0 1 1 1 1

1 0 0 0 0

1 0 0 0 1

1 0 0 1 0

1 0 0 1 1

1 0 1 0 0

1 0 1 0 1

1 0 1 1 0

1 0 1 1 1

1 1 0 0 0

1 1 0 0 1

1 1 0 1 0

1 1 0 1 1

1 1 1 0 0

1 1 1 0 1

1 1 1 1 0

1 1 1 1 1

Apending parity bit at the end of the codeword

code1 = GF(2) array.

Array elements =

Columns 1 through 13

0 0 0 0 0 0 0 0 0 0 0 0 0

0 0 0 0 1 0 1 0 0 1 1 0 1

0 0 0 1 0 1 0 0 1 1 0 1 1

0 0 0 1 1 1 1 0 1 0 1 1 0

0 0 1 0 0 0 1 1 1 1 0 1 0

0 0 1 0 1 0 0 1 1 0 1 1 1

0 0 1 1 0 1 1 1 0 0 0 0 1

0 0 1 1 1 1 0 1 0 1 1 0 0

0 1 0 0 0 1 1 1 1 0 1 0 1

0 1 0 0 1 1 0 1 1 1 0 0 0

0 1 0 1 0 0 1 1 0 1 1 1 0

0 1 0 1 1 0 0 1 0 0 0 1 1

0 1 1 0 0 1 0 0 0 1 1 1 1

0 1 1 0 1 1 1 0 0 0 0 1 0

0 1 1 1 0 0 0 0 1 0 1 0 0

0 1 1 1 1 0 1 0 1 1 0 0 1

1 0 0 0 0 1 0 1 0 0 1 1 0

1 0 0 0 1 1 1 1 0 1 0 1 1

1 0 0 1 0 0 0 1 1 1 1 0 1

1 0 0 1 1 0 1 1 1 0 0 0 0

1 0 1 0 0 1 1 0 1 1 1 0 0

1 0 1 0 1 1 0 0 1 0 0 0 1

1 0 1 1 0 0 1 0 0 0 1 1 1

1 0 1 1 1 0 0 0 0 1 0 1 0

1 1 0 0 0 0 1 0 1 0 0 1 1

1 1 0 0 1 0 0 0 1 1 1 1 0

1 1 0 1 0 1 1 0 0 1 0 0 0

1 1 0 1 1 1 0 0 0 0 1 0 1

1 1 1 0 0 0 0 1 0 1 0 0 1

1 1 1 0 1 0 1 1 0 0 1 0 0

1 1 1 1 0 1 0 1 1 0 0 1 0

1 1 1 1 1 1 1 1 1 1 1 1 1

Columns 14 through 15

0 0

1 1

1 0

0 1

1 1

0 0

0 1

1 0

1 0

0 1

0 0

1 1

0 1

1 0

1 1

0 0

1 1

0 0

0 1

1 0

0 0

1 1

1 0

0 1

0 1

1 0

1 1

0 0

1 0

0 1

0 0

1 1

t = 3

Error correcting capacity of BCH code is: 3

Please enter bit long valid codeword: [1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 ]

Corrected codeword is:

codecomsg = GF(2) array.

Array elements = 1 1 1 1 1