

# DOUBLE ERROR CORRECTING LONG CODE

Joanne Gomes<sup>1</sup> and B K Mishra<sup>2</sup>

<sup>1</sup>Research Scholar, SNTD University, Mumbai

*jngomes@ieee.org*

<sup>2</sup>Thakur College of Engineering, Kandivali, Mumbai

*drbk.mishra@thakureducation.org*

## ABSTRACT

*This paper introduces a novel binary, long double error correcting, systematic code (8 2 5) that can detect and correct errors up to two bits in the received vector using simple concept of syndrome decoding. The motivation behind the construction of this code is the idea to achieve 100% error correction on the receiver side and to use the encoder/decoder that is simpler than the existing double error correcting codes. By 100% correction we mean that when the two bits of information ( $k=2$ ) is transmitted simultaneously over the noisy channel and if the two bits are in the error, in the received vector, then this code can detect and correct errors up to two bits thus recovering both the two bits of information. We show that to achieve this we need to choose long code length, maximum of 8 bits ( $n=8$ ). We present a generator matrix and a parity check matrix to achieve required Hamming distance by using the concept of long code. The paper also presents the performance bounds satisfied by the said code.*

## KEYWORDS

*Double-Error-Correction, Long code, generator matrix, parity check matrix, Encoder, Decoder, Syndrome, Performance bounds.*

## 1. INTRODUCTION

The theory of linear block codes is well established since many years. In 1948 Shannon's work showed that any communication channel could be characterized by a capacity at which information could be reliably transmitted [1]. In 1950, Hamming introduced a single error correcting and double error detecting codes with its geometrical model [2] whereas just before Hamming, Golay had introduced (23, 12) triple error correcting perfect code. MacDonald in [3] derives an upper bound on minimum distance of a linear block code. Cyclic codes using polynomials were redefined and described by Peterson for error detection in [4]. The papers [5][6] review and highlight the important contributions of different scientist to the coding theory for the period of almost fifty years. Paper [4] includes the table indicating Shannon limit on  $E_b/N_0$  for the AWGN channel. According to Shannon to achieve reasonable BER, at  $1/2$  code rate  $E_b/N_0$  required is 0.2 dB and as the code rate approaches to zero minimum  $E_b/N_0$  required is -1.6 dB. So far many different and more effective error correcting codes have been invented by researchers. In [7] author derives the necessary conditions for existence of  $e$ -error correcting code over  $GF(q)$  with word length  $n$  and concludes that Golay (23,12) is the only nontrivial binary perfect 3-error correcting code over any  $GF(q)$  and no other perfect codes exists except those invented by Hamming and Golay. Number of double error correcting BCH codes are listed and permutation decoding method for codes with code rates  $(k/n) \geq 1/2$  is presented in [8]. Computer results on the minimum distance of some BCH codes are listed down in [9]. Construction of long codes, a class of codes derived from algebraic plane curves and its decoding is presented by Justesen in [10]. Similarly updated table of collection of lower and upper bounds for  $d_{max}(n, k)$ , i.e. maximum possible minimum distance of a binary linear code, of code word length  $n$  and dimension  $k$ , has been given in [11]. The utility of different types of linear codes in communication over power limited and bandwidth limited channels is discussed in [12]. For power limited channels concatenated codes are more suitable with inner code as short length convolution code and outer code as long length, simple, high rate Reed Solomon code. A systematic quasi-cyclic code (16, 8) for computer memory system which corrects double errors and detects triple errors has been given in [13]. It also gives encoding and

decoding method for the same and also presents a quasi-cyclic code (32, 16) for triple error correction. The article given in [14] reformulates Shannon's sphere-packing bound in a normalized form conducive to analyzing the relative performance of families of turbo codes. In [15] Author shows that the longer the information block size, better the performance of long turbo codes. It shows that long practical codes can closely approach the theoretical limits on code performance and presents error bounds on long turbo code. In [16] the decoding method for the linear block codes based on multiple information sets and ordered statistics has been presented by Shu lin. It is more effective for long codes with code rate less than  $\frac{1}{2}$ . In [17] Berrou introduces long codes, a new family of convolution code called turbo codes which are derived from a particular concatenation of two recursive systematic codes linked together by non uniform interleaving to achieve near optimum performance. All these and many other existing codes give the efficient coding for partial correction of information bits and most of them assume that the minimum number of information bits to be transmitted should be at least three ( $k=3$ ) for double error correction. E.g. Golay (23, 12) code corrects 3 bit errors in 12 information bits or (16, 8) code in [13] corrects 2 errors in 8 information bits. Such types of codes have code rate  $\geq \frac{1}{2}$  to get optimum transmission rates. The table given in [11] indicates, for  $k = 2$ , code word length  $n$  should be 8 to achieve min Hamming distance of 5. Practically (8 2 5) code is not available. Most of the (double error correcting codes) have complicated decoding procedures.

Consider a case wherein we transmit  $k$  information bits simultaneously. If all the  $k$  bits are in error on the receiver side and if we want to correct them, transmitted code word should have more redundant bits. This paper presents a simple long double error correcting (8 2 5) code based on syndrome decoding. The long code will consume more transmission bandwidth and the will be less efficient. However in power limited system such as *Ultra wideband communication, bandwidth is abundant*. UWB has wide applications in radar, sensor networks and indoor multimedia communication. UWB can also be used to communicate with sensors placed inside the human body for cure of certain diseases. In all these UWB applications, to improve the quality of wireless communication, we can take benefit of the long code presented here, for *100% error correction* in wideband communication because the accuracy is an equal important criterion while assuring good quality of service in communication. The design of double error correcting long binary code presented here is for 2 bits of information ( $k=2$ ). According to Hamming, the minimum distance required for double error correction is '5'. In this paper we evaluate the code length required to achieve a Hamming distance of '5', when two information bits have to be transmitted. We show that a long codeword of 8 bits is needed to be transmitted in order to correct double error in two information bits and to achieve 100% error correction.

## 2. EXISTENCE OF A LONG CODE AND RELATED WORK

In an error correcting code the central concept is the notion of minimum distance. If a code can be constructed with the minimum distance of  $2t+1$  between two code words, then any number of errors per codeword which does not exceed  $t$  can be corrected. A linear block code  $C$  is generally specified as  $(n, k, d)$  code, where  $n$ = length of the code word,  $k$ = length of information bits and  $d$  is the minimum Hamming distance between any two code words [18][19]. Shannon showed that at any rate of information transmission up to the channel capacity, it should be possible to transfer information at error rates that can be reduced to any desired level. In [15] author shows that the practical codes such as, long turbo codes with code rates  $< \frac{1}{2}$  can approach the theoretical limits on code (near Shannon limit) performance. The following figure 1 indicates that the long codes can improve the performance of error correcting system [20].

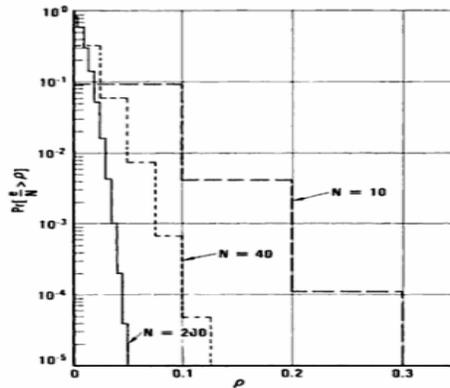


Figure 1. Probability that the fraction of symbol is in error  $e/N$  in a block of length  $N$  exceeds  $p$  for  $P_e = 0.01$

The form of the curve in fig 1 suggested that if one has a scheme for correcting a fixed fraction  $t/n$ , then the error rate could be made arbitrarily small by simply choosing the block to be long enough [20]. Thus the results in fig 1 indicate the potential for performance improvement by increasing the block length. Here the loss of efficiency occurs because the relative number of useful messages that these schemes convey becomes vanishingly small. This might be tolerated in medical applications where accuracy plays important role.

The development of turbo codes created a new interest in knowing how closely practical codes can approach the theoretical limits on code performance. Simulations performed in [23] show that turbo codes approach perfect ness closely (within 0.7 dB) and almost uniformly over a wide range of code rates and block sizes. Inspired by this near Shannon performance of some long codes, we came across an idea of correcting all the transmitted bits. In this paper we assume that two bits are transmitted simultaneously and corrected at the receiver.. The work presented in this paper is directly related to the Hamming codes of earlier days. For 1 bit correction the concept of repeated bit sequence was used then, where for “1” data bit “111” and for “0” data bit “000” was transmitted as repeated sequence. When we want to transmit two data bits simultaneously (e.g. as in 2-dimensional modulation scheme) and want to correct them all, there is no error correcting code existing for such a scheme as far as our knowledge is concerned. This paper attempts to devise such a code and is based on the concept of Hamming codes.

### 2.1. Maximum Code Length

In this section we find the maximum length required for double error correction long code, with 100% error correcting capability. Consider  $(n, k, d_{\min})$  is a binary linear cyclic code  $C$  where  $n$  is the length of each codeword in the code,  $k$  is the dimension of the code and  $d_{\min}$  is the minimum Hamming distance between any two code words.  $R = k/n$ , is the code rate.  $(n-k)$  indicates parity or redundant bits added to the information bits  $k$ . If  $t$  indicates the number of errors to be corrected then the minimum distance criteria is  $d_{\min} \geq 2t+1$ . According to theory, to design any linear block code, the code must satisfy Hamming bound given by the following eq. 1 along with the minimum distance criteria [18][19].

$$\sum_{i=0}^t \binom{n}{i} \leq 2^{n-k} \tag{1}$$

Where  $\sum_{i=0}^t \binom{n}{i}$  indicates number of correctable error patterns in a given (n, k) code for t correctable errors.

Suppose we have the problem defined as: Given that k=2 and t=2 determine code length n then consider the following cases.

*Case I:* With t =2, required  $d_{\min} \geq 5$ , assuming code rate =  $R = 1/2$  and k=2, n will be 4 and n-k will be 2. Now correctable error patterns will be

$$\sum_{i=0}^t \binom{n}{i} = {}^n c_0 + {}^n c_1 + {}^n c_2, \text{ where, } {}^n c_i = \frac{n!}{i!(n-i)!}$$

i.e.  $i=0; {}^4 c_0 = 1, \quad i=1; {}^4 c_1 = 4, \quad i=2; {}^4 c_2 = 6$ , therefore

$$\sum_{i=0}^t \binom{n}{i} = 1+4+6= 11, \text{ where as } 2^{n-k} = 2^2=4.$$

Thus at  $1/2$  code rate, with k = 2 and n = 4, it does not satisfy this Hamming bound.

*Case II:* Decreasing the code rate  $R \leq 1/3$ , k=2 and n = 7, n-k will be 5. Correctable error patterns will be

$$\sum_{i=0}^t \binom{n}{i} = 1+7+21= 29, \text{ and } 2^{n-k} = 2^5 = 32.$$

This case does satisfy hamming bound but does not satisfy min distance criteria.

*Case III:* Now if we choose  $R = 1/4$ , n will be 8 and for t=2, Hamming bound will be satisfied as follows

$$\begin{aligned} \sum_{i=0}^t \binom{n}{i} &\leq 2^{n-k} \\ \sum_{i=0}^2 \binom{8}{i} &\leq 2^{8-2} \\ 1 + 8 + 28 &\leq 2^6 \\ 37 &\leq 2^6 \end{aligned}$$

Thus we decide that the maximum code length of ‘8’ is required to achieve Hamming distance of ‘5’, in order to correct two bit errors while transmitting two information bits (k=2). This code length n = 8 will guarantee us 100% error correction.

### 3. DESIGN OF A LONG CODE

#### 3.1. Generator and Parity Matrix

With length n=8, we cannot choose an nth root primitive polynomial of Galois field GF(7) to obtain generator polynomial g(x) for (8 2 5) code that we wish to design, but we can use the polynomials of field GF(15) to form g(x). Consider the generator polynomial g(x) of degree (n-k) = 6 is derived from the primitive polynomials  $(x^2+ x+1)$  and  $(x^4+x^3+x^2+ x+1)$  of GF(15) as follows [18][19].

$$\begin{aligned} g(x) &= (x^2+ x+1) (x^4+x^3+x^2+ x+1) = (x^6+x^4+x^3+x^2+1) \quad (2) \\ g(x) &= (g_6x^6+g_5x^5+g_4x^4+g_3x^3+g_2x^2+g_1x+g_0) \end{aligned}$$

$$G_{(k,n)} = [I_k \ : P_{(k,n-k)}] = \begin{pmatrix} g_0 g_1 g_2 g_3 g_4 g_5 g_6 g_7 \\ 1 \ 0 \ : 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 1 \ : 0 \ 1 \ 1 \ 1 \ 0 \ 1 \end{pmatrix} \quad (3)$$

We know the parity matrix of above code is

$$P_{(k,n-k)} = \begin{pmatrix} 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \end{pmatrix} \quad (4)$$

Now parity check matrix =  $H_{(n-k,n)} = [P^T_{(n-k,k)} \ : I_{n-k}]$

$$H_{(n-k,n)} = \begin{pmatrix} 10 \ : 100000 \\ 11 \ : 010000 \\ 11 \ : 001000 \\ 01 \ : 000100 \\ 10 \ : 000010 \\ 01 \ : 000001 \end{pmatrix} \quad (5)$$

The code (8 2 5) presented here is a systematic cyclic code and its hardware part, encoder and decoder can be implemented using shift registers.

### 3.2. Encoding of a long code

With code dimension k=2 we can transmit four types of messages (m = 00, 01, 10, 11) and corresponding code words are obtained using formula  $c = m \times g(x)$  [18] [19] as shown below

$$c1 = [0 \ 0] \begin{pmatrix} 1 \ 0 \ : 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 1 \ : 0 \ 1 \ 1 \ 1 \ 0 \ 1 \end{pmatrix} = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0], \quad c2 = [0 \ 1] \begin{pmatrix} 1 \ 0 \ : 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 1 \ : 0 \ 1 \ 1 \ 1 \ 0 \ 1 \end{pmatrix} = [0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1]$$

$$c3 = [1 \ 0] \begin{pmatrix} 1 \ 0 \ : 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 1 \ : 0 \ 1 \ 1 \ 1 \ 0 \ 1 \end{pmatrix} = [1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0], \quad c4 = [1 \ 1] \begin{pmatrix} 1 \ 0 \ : 1 \ 1 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 1 \ : 0 \ 1 \ 1 \ 1 \ 0 \ 1 \end{pmatrix} = [1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1]$$

#### 3.2.1 Software implementation of encoder:

MATLAB program for implementation of (8 2 5) code is given below.

```
%Transmitter/encoder
%-----
m=[0 1];
g=[1 0 1 1 1 0 1 0 ; 0 1 0 1 1 1 0 1]; %generator matrix
c=m*g; %codeword to be transmitted
%-----
```

#### 3.2.2 Hardware implementation of encoder:

The code presented here is a systematic cyclic code, its hardware part; encoder and decoder can be implemented using shift registers. We know that the generator matrix of the said code is given by

$$g(x) = (1+x^2+x^3+x^4+x^6), \text{ where } g_0=1, g_1=0, g_2=1, g_3=1, g_4=1, g_5=0, g_6=1$$

The encoder as shown in figure 2 uses six shift registers. The encoded message enters the encoder with MSB bit first.

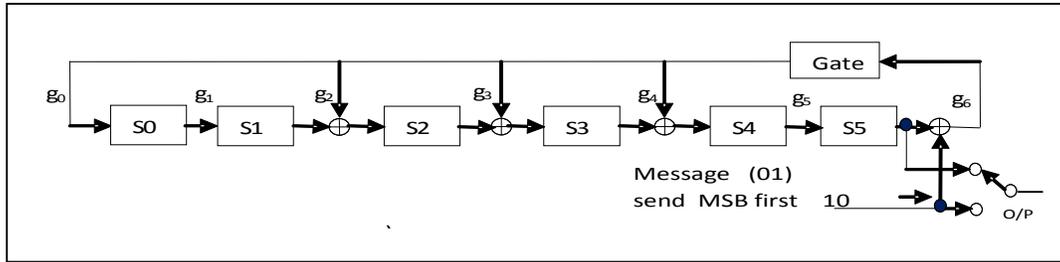


Figure 2. Encoder for (8 2 5) long binary code

### 3.3 Decoding of a Long Code:

If  $c$  is the transmitted code vector,  $r$  is a received code vector with error,  $e$  is an error vector by which  $r$  differs from  $c$ ,  $H$  is a parity check matrix and  $S$  is syndrome indicating the presence of error then the syndrome  $S$  is given as [19] [20],

$$S = r \times H^T = (c + e) \times H^T = c H^T + e H^T = m \times G \times H^T + e H^T = 0 + e H^T \text{ thus,}$$

$$S = e \times H^T \quad (6)$$

For this code, total of 36 error patterns per message (single bit error-8 and double bit error-28) can be received on the receiver side. This will give us 36 unique syndromes of six bits ( $n-k$ ) long and will help us to find which one or two bits in the received vector are in error. Complimenting these located error bits will give us the correct transmitted code word which will be in systematic form.

#### 3.3.1 Software implementation of Syndrome Decoder:

Matlab program for decoder:

```
%Receiver/decoder
%-----
r = [1 0 0 1 1 1 0 1];           % suppose the received vector with first two bits in error
H_tr = [1 1 1 0 1 0; 0 1 1 1 0 1; 1 0 0 0 0 0; 0 1 0 0 0 0; 0 0 1 0 0 0; 0 0 0 1 0 0; 0 0 0 0 1 0; 0 0 0 0 0 1];
S = r*H_tr;                       % H_tr is transpose of parity check matrix
if S == [1 0 0 1 1 1]              % syndrome for first two bits in error
r(1) = not(r(1));                  % compliment first bit
r(2) = not(r(2));                  % compliment second bit
end
r_corrected = r;                   % received vector after correction
```

#### 3.3.2 Hardware implementation of decoder:

The (8 2 5) code presented here has  $n-k = 6$  parity bits, so its decoder will have 6 stages of shift registers as shown in figure 3.

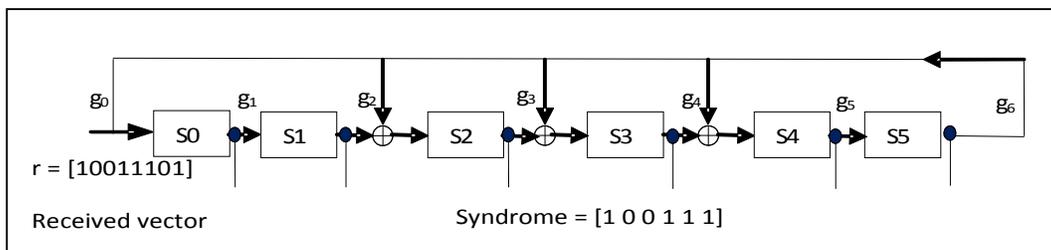


Figure 3. Syndrome decoder

Here the received codeword is fed at the left side (LSB bit entering first) of the decoder as shown in figure 3. The output of each shift register will contribute to syndrome.

## 4. PERFORMANCE ANALYSIS OF (8, 2, 5) CODE

### 4.1. Test of Linearity:

The Code (8 2 5) presented here satisfies the following conditions of linearity [18][19] hence it is a linear code.

- All zero word is always a codeword
- Minimum weight of non-zero vector = minimum distance of the code = 5
- Addition of two code words is a valid code word

### 4.2. Performance Bounds:

The Code (8 2 5) presented here satisfies the following performance bounds [19] [20]

#### 4.2.1. Hamming Bound:

The Hamming bound being tighter for higher rates.

$$\sum_{i=0}^t \binom{n}{i} \leq 2^{n-k}$$

$$\sum_{i=0}^t \binom{n}{i} = 1+8+28 \leq 2^6 \tag{7}$$

$$37 \leq 2^6$$

#### 4.2.2. Gilbert-Varsharmov Bound:

Gilbert bound gives an upper bound on selecting n-k. In this case the Gilbert bound is satisfied as follows.

$$2^{n-k} \geq \sum_{i=0}^{d-2} \binom{n-1}{i}$$

$$\geq \sum_{i=0}^3 \binom{7}{i} \tag{8}$$

$$2^6 \geq 1+7+21+35$$

$$2^6 = 64$$

#### 4.2.3. Singleton Bound:

This bound sets an upper bound to minimum distance between two code words.

$$d_{\min} \leq n - k + 1 \tag{9}$$

with  $d_{\min} = 5$  and  $n-k = 6$ , singleton bound is satisfied.

#### 4.2.4. Plotkin Bound:

It sets an upper limit to  $d_{\min}$  for fixed values of  $n$  and  $k$ . It tends however to set a tighter bound for the codes with lower code rate, the Hamming bound being tighter for higher rates. The Plotkin bound applies to linear codes and states that the minimum distance is at most equal to the average weight of all nonzero codewords

$$\begin{aligned}
 d_{\min} &\leq \frac{n 2^{k-1}}{2^k - 1} \\
 &\leq \frac{8 \times 2^{2-1}}{2^2 - 1} \\
 &\leq \frac{16}{3} \\
 d_{\min} &\leq 5.333
 \end{aligned}
 \tag{10}$$

In a given code  $n = 8, k = 2, d_{\min} = 5$  so it satisfies the Plotkin bound as shown above.

**4.2.5. Griesmer Bound:**

The Griesmer bound is often tighter than the Plotkin bound, and its derivation leads to methods of constructing good codes. Let  $(n, k, d)$  represent the lowest possible value of length  $n$  for a linear code  $C$  of dimension  $k$  and minimum distance  $d$ .

$$\begin{aligned}
 n &\geq \sum_{i=0}^{k-1} \frac{\lceil d \rceil}{2^i} \\
 &\geq \sum_{i=0}^{k-1} \frac{\lceil d \rceil}{2^i} \\
 &\geq 5 + \frac{5}{2} \\
 &\geq 7.5
 \end{aligned}
 \tag{11}$$

Here  $n = 8$  and hence it satisfies the Griesmer bound.

**5. ERROR DETECTING AND CORRECTING CAPABILITY:**

The linear binary code  $(8, 2, 5)$  can detect and correct all one bit and 2 bit errors.

**5.1. Minimum distance criteria:**

Given a block code  $C$ , its minimum Hamming distance,  $d_{\min}$ , is defined as the minimum Hamming distance among all possible distinct pairs of code words (e.g.  $v_1, v_2$ ) in  $C$ , [18][19].

$$d_{\min} = \min_{v_1, v_2 \in C} \{d_H(\overline{v_1}, \overline{v_2}) \mid \overline{v_1} \neq \overline{v_2}\}
 \tag{12}$$

In order to compute the minimum distance  $d_{\min}$  of a block code  $C$ , in accordance with above equation 12, a total of  $2^{k-1}(2^k - 1)$  distances between distinct pairs of code words are needed.

The following table-1 shows the hamming distance between different code words of  $(8, 2, 5)$  code. Here minimum weight is equal to minimum distance of code  $C$ .

Table 1. Hamming distance between different code vectors

1011 1010 0000 0000 ----- 1011 1010	1011 1010 0101 1101 ----- 111 0 0111	1011 1010 111 0 0111 ----- 0101 1101	0101 1101 111 0 0111 ----- 1011 1010	111 0 0111 0000 0000 ----- 111 0 0111	0101 1101 0000 0000 ----- 0101 1101
$d_{\min} = 5$	$d_{\min} = 6$	$d_{\min} = 5$	$d_{\min} = 5$	$d_{\min} = 6$	$d_{\min} = 5$

**5.2. Triangle inequality:**

It states that the code  $C$  is capable of correcting all error patterns of  $t$  or fewer errors. Let  $v$  and  $r$  be the transmitted and received vectors respectively and let  $w$  be any other code vector in  $C$  then The Hamming distances among  $v, r$  and  $w$  satisfy the triangle inequality:

$$d(v, r) + d(w, r) \geq d(v, w) \tag{13}$$

For a given block code, designed with generator matrix of equation 14, considering  $v = [10\ 11\ 1010]$ ,  $r = [00\ 11\ 1010]$  and  $w = [01\ 01\ 1101]$ ,  $d(v, r) = 1$ ,  $d(w, r) = 5$  and  $d(v, w) = 6$ .

Here  $d(v, r) + d(w, r) = d(v, w)$ . Thus it satisfies the triangle inequality.

### 5.3. Weight distribution W(C):

The Weight distribution  $W(C)$  of an error correcting code  $C$ , is defined as the set of  $n + 1$  integers  $W(C) = \{A_i, 0 \leq i \leq n\}$  such that there are  $A_i$  code words of Hamming weight  $i$  in  $C$ , for  $i = 0, 1, \dots, n$ .

For this code, the weight distribution will be as per table 2

Table 2. Weight distribution of a new code

i	0	1	2	3	4	5	6	7	8
A <sub>i</sub>	1	8	28	18	27	36	21	8	1

### 5.4. Asymptotic Coding Gain (G<sub>a</sub>):

It is the gain that would be delivered if vanishingly small decoded error rates were required [19][20]. It is given by  $G_a = 10 \log[R(t+1)]$  Or  $G_a = 10 \log[R \cdot d_{min}]$

If  $R = \text{Coding gain} = 1/4$ ,  $t=2$ ,  $d_{min}=5$ , then

$$G_a = 10 \log [3/4] = -1.249 \text{ dB}$$

$$\text{Or } G_a = 10 \log[Rd] = 10 \log [5/4] = 0.969 \text{ dB}$$

Thus asymptotic coding gain will be between -1.249 to 0.969 dB.

## 6. Probabilities over BSC channel:

### 6.1. Probability of Undetected error (Exact Value) over BSC channel:

The probability of an undetected error over a BSC, denoted  $P_u(C)$ , is the probability that the received word differs from the transmitted code word but the syndrome equals zero [18][19]. This probability is given by eq. 14.

$$P_u(C) = \sum_{i=d_{min}}^n A_i P^i (1 - P)^{n-i} \tag{14}$$

Inserting values of  $A_i$  from weight distribution table 2 in eq. 14 we get,

$$P_u(C) = 34 \cdot p^5 (1 - p)^3 + 15 \cdot p^6 (1 - p)^2 + 3 \cdot p^7 (1 - p) \tag{15}$$

The following table 3 gives exact values of probability of undetected error according to eq. 15.

Table 3. Probability of undetected error-Exact Value

P	0.1	0.01	1.0e-003	1.0e-004	1.0e-005
P <sub>u</sub> (C)	2.60e-004	3.31e-009	3.39e-014	3.39e-019	3.39e-024

### 6.2. Probability of Undetected error (Upper Bound) over BSC channel:

The following eq. 16 gives upper bound on probability of undetected error over BSC channel.

$$P_u(C) \leq \sum_{i=d_{min}}^n \binom{n}{i} P^i (1 - P)^{n-i} \tag{16}$$

$$P_u(C) \leq \binom{8}{5} P^5 (1-P)^3 + \binom{8}{6} P^6 (1-P)^2 + \binom{8}{7} P^7 (1-P) \tag{17}$$

$$\leq 56 P^5 (1-P)^3 + 28 P^6 (1-P)^2 + 8 P^7 (1-P)$$

The following table 4 gives Upper Bound values of probability of undetected error.

Table 4. Probability of undetected error-Upper Bound

P	0.1	0.01	1.0e-003	1.0e-004	1.0e-005
P <sub>u</sub> (C)	4.31e-004	5.46e-009	5.58e-014	5.59e-019	5.59e-024

The following figure 4 plots the exact values and upper bound values of probability of undetected error over BSC channel.

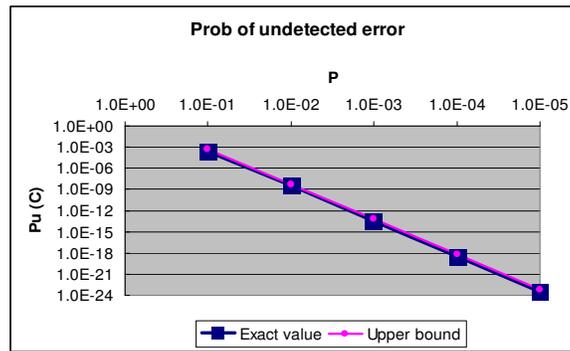


Figure 4. Exact value and upper bound on the prob of undetected error Pu(C) for a (8 2 5) code

Figure 4 indicates that the exact and upper bound values on the prob. of an undetected error over a BSC channel are same and negligible for the dual error correcting long code (8 2 5).

**6.3. Probability of Correct Decoding Pc(c) (Lower bound) over BSC channel:**

$$P_c(C) = \sum_{i=0}^t \binom{n}{i} P^i (1-P)^{n-i} \tag{18}$$

$$P_c(C) = 1 \times P^0 (1-P)^8 + 8 \times P^1 (1-P)^7 + 28 \times P^2 (1-P)^6 \tag{19}$$

The following table 5 gives Lower Bound values of probability of correct decoding.

Table 5. Probability of correct decoding-Lower Bound

P	0.1	0.01	1.0e-003	1.0e-004	1.0e-005
P <sub>c</sub> (C)	9.61e-01	9.99e-01	1.00e-00	1.00e-00	1.00e-00

**6.4. Probability of incorrect Decoding or Decoding error (Upper bound) over BSC channel:**

It is given as Pe(C) = 1 – Pc(C) [19] [20].

$$P_e(C) \leq 1 - \sum_{i=0}^t \binom{n}{i} P^i (1-P)^{n-i} \tag{20}$$

The following table 6 gives Upper Bound values of probability of decoding error.

Table 6. Probability of decoding error-Upper Bound

P	0.1	0.01	1.0e-003	1.0e-004	1.0e-005
$P_e(C)$	3.81e-02	5.39e-05	5.58e-08	5.59e-11	5.55e-14

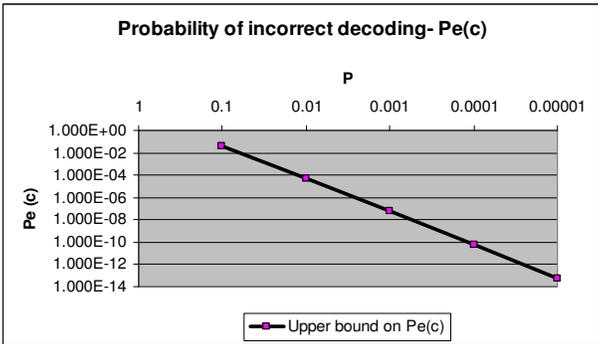


Figure 5. Upper bound on the probability of decoding error  $P_e(C)$  for a binary code (8 2 5)

Figure 5 indicates that the  $P_e(C)$ , is negligible for the dual error correcting long code (8 2 5).

### 7. CONCLUSIONS

This paper has presented a new double error correcting long binary linear cyclic code (8 2 5) with code rate  $\frac{1}{4}$ . It can detect and correct single as well as double bit errors in the received codeword. Since transmitted codeword contains two information bits, double error correction by this code achieves 100% error correction. Near optimum performance of theoretical code can be achieved with this code using practically simple encoding and decoding procedure. This is done at the cost of transmission bandwidth but it could be a suitable method for sensitive applications in medical science where accuracy is important. Wideband technologies such as UWB have abundant bandwidth; by using this error correcting method along with the wideband technologies can help us achieve reasonably good transmission rates.

### REFERENCES

- [1] Shannon C E (1948) , “Mathematical theory of communication”, Bell Sys. Tech. Journal, pp 379-423 and 623-656.
- [2] Hamming, R W, (1950),”Error Detecting and Error Correcting Codes”, The Bell System Technical Journal, Soc, Industrial Appl. Math., Vol. 26, No. 2.
- [3] J. E. MacDonald, (1960), “Design Methods for Maximum Minimum-Distance Error-Correcting Codes”, IBM Journal, pp 43-57.
- [4] W W Peterson and D T Brown, (1961) “Cyclic Codes for Error Detection” Proceedings of the IRE, pp 228-235.
- [5] Madhu Sudan, (2002), “Coding Theory: Tutorial & Survey”, Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science (FOCS.01), pp 1-18.
- [6] Michelle Effros, (1998), “1948-1998 Information Theory: The First Fifty Years” IEEE Information Theory Society Newsletter, Special Golden Jubilee Summer, Editorial.
- [7] J. H. Van Lint (1970), “On the nonexistence of perfect 2- and 3- Hamming-error-correcting-codes over  $GF(q)$ ”, IEEE Transaction on Information and Control, Vol. 16, No. 4, pp 396-401.
- [8] S G S Shiva K C Fung and H S Y Tan, (1970), “On Permutation Decoding of Binary Cyclic Double-Error-Correcting Codes of Certain Lengths”, IEEE Transaction on Information Theory, pp 641-643.

- [9] C. L. Chenz, (1970), "Computer Results on the Minimum Distance of Some with Binary Cyclic Codes" IEEE Transaction on Information Theory, Vol. IT-18, pp 359-360.
- [10] Jorn Justesen, Knud J Larsen and H. Elbrond Jensen, (1989), "Construction and Decoding of a Class of Algebraic Geometry Codes ", IEEE Transactions on Information Theory, Vol. 35, No. 4, pp. 811-821.
- [11] James Massey (1989), "A short introduction to coding theory and practice" Proceeding of International symposium on signal, system and electronics, Germany, pp 629-633.
- [12] A E Brouwer and Tom Verhoeff,, (1993), "An Updated table of minimum distance bounds for binary linear code", IEEE Transaction on Information Theory, Vol. 39. No. 2, pp 662-677.
- [13] T. Aaron Gulliver and Vijay K. Bhargava, (1993), "A Systematic (16,8) Code for Correcting Double Errors and Detecting Triple Adjacent Errors" IEEE Transaction on Computers, Vol. 42, No. 1, pp. 109-112.
- [14] S Dolinar, D Divsalar, and F Pollara, (1998), "Code Performance as a Function of Block Size", TMO Progress Report , California, 42-133, pp 1-23.
- [15] Christian Schlegel and Lance Per'ez, (1999), "On Error Bounds and Turbo-Codes", IEEE Communications Letters, Vol. 3, No. 7, pp 205-207.
- [16] M P C Fossorier and Shu LIN, (1999), "Reliability based Information Set Decoding for Linear Block Codes", IEEE transaction on Fundamentals, Vol. E82-A, No. 10, pp 2034-2042.
- [17] Claude Berrou and A Glavieux, (1999), "Near Optimum Error Correcting Coding and Decoding: Turbo Codes" IEEE Transactions on Communications, Vol. 44, NO.10, pp 1361-1389,
- [18] S. Lin and D.J. Costello, (1983), "Error Control Coding: fundamentals and applications, Prentice Hall Publishers.
- [19] S B Wicker, (1994), Error Control Systems for Digital Communication and Storage, Prentice Hall Publishers.
- [20] George Clark and J Cain, "Error correcting code for digital communications" John Willey Publishers.
- [21] D J C Mackay, "Near Shannon limit performance of low density parity check codes", Electronics Letters, Vol. 33 NO. 6, 1997, pp. 457-458.

**Authors**

B.K.Mishra was awarded PhD degree from Birla institute of technology in 1998.He has 22years of teaching experience. His present research interest focuses on device modeling of optical sensors.



Joanne Gomes completed ME in Electronics and Telecom engineering, from Mumbai University of, India, in 2005. She has 13 years of industrial and 9 years of teaching experience. She is presently pursuing her PhD. Her present research area is UWB wireless communication for home networking.

