Decoding Algorithm for Ternary RM Codes

B.S.Brar[1]
[1]*Department of Applied Sciences,*
Baba Farid College of Engineering and Technology, Bathinda, Punjab (India)

Abstract: RM codes are familiar and important codes. Ternary RM codes are interpreted in terms of super-imposition. A new decoding algorithm for a class of Simple Iterated codes is proposed. It plays central role in decoding algorithm. In this paper, a new decoding algorithm for Ternary RM codes is presented along with examples. As compared to binary RM code, ternary RM code has stronger role of securing the transmission of the messages, has enhanced utility, and has increased detection and correcting capability.

Keywords: **Constituent codes, Decoding algorithm, Decomposition of RM codes, SI codes, super-imposition of RM codes, Ternary RM codes.**

I. INTRODUCTION

Muller first put forward these codes. A decoding algorithm for these codes was devised by Reed. In the decoding algorithm, Majority-logic was used which is based upon the concept of finite geometry. A majority-logic decoding algorithm can decode Finite geometry codes including Euclidean geometry (EG) and Projective geometry (PG) codes [Peterson and Weldon Jr. (1972), Goethals and Delsarte (1968)]. Peterson and Weldon Jr.(1972), Welden Jr. (1969), and Chen (1971, 1972) showed that in a decoder the number of majority-logic gates used can be reduced. Rodolph and Hartmann (1973) showed that complexity of decoder may be reduced to a great extent. But it can be done at the expense of decoding-delay. For first-order binary RM codes, MacWilliams and Sloane (1977) proposed a decoding algorithm. Tokiwa, Sugimura, Namekawa and Kasahara (1982) interpreted binary RM codes in terms of the concept of superimposition and presented new decoding algorithm. They compared their own decoding algorithm with conventional algorithm which is there for cyclic binary RM codes in relation with problem of the decoding-delay.

II. BINARY RM CODES

Def.1. [Peterson (1961)] Let there be two integers r and m such that $0 \leq r \leq m$. Then there exists a binary code having length $n = 2^m$, min. distance as $d = 2^{m-r}$, and information-symbols $k = 1 + {}^mC_1 + {}^mC_2 + \quad . \quad . \quad . \quad + {}^mC_r$. It is called rth order binary RM code, written as r-RM binary code or $(r, n = 2^m)$RM binary code.

Def.2. [Peterson (1961), and MacWilliams and Sloane (1977)] Let v_0 is a vector, whose $n = 2^m$ components are all 1s. Let $v_1, v_2, \quad . \quad . \quad . \quad , v_m$ be row-vectors of an $m \times 2^m$ matrix, having its ith column as binary representation of integer i, where $i = 0, 1, \quad . \quad . \quad . \quad , 2^m - 1$. Clearly, each of $v_1, v_2, ,v_m$ has $n = 2^m$ components. As a result, the number of columns of the matrix will be $n = 2^m$. Then the $(r, n = 2^m)$ binary Reed-Muller code is k-dimensional vector-space. The vectors $v_0, v_1, v_2, \quad . \quad . \quad . \quad , v_m$ and also all vector-products of these vectors taken r or fewer at a time, are basis vectors of this k-dimensional vector-space, where k is given by: $k = 1 + {}^mC_1 + {}^mC_2 + \quad . \quad . \quad . \quad + {}^mC_r$. Note that vector-product of vectors **u** and **v**, where $\mathbf{u} = (u_1, u_2, \quad . \quad . \quad . \quad , u_n)$ and $\mathbf{v} = (v_1, v_2, \quad . \quad . \quad . \quad ,v_n)$, is given by $\mathbf{u}.\,\mathbf{v} = (u_1v_1, u_2v_2, \quad . \quad . \quad . \quad , u_nv_n)$.

If m = 3, then the length of $(r, n = 2^m)$ binary RM code will be $n = 2^m = 2^3 = 8$. Therefore, we will have basis vectors as follows:

$$v_0 = 11111111] \ 0^{th} \text{ order} \qquad \left.\begin{array}{l} v_0 = 11111111 \\ v_1 = 01010101 \\ v_2 = 00110011 \\ v_3 = 00001111 \end{array}\right] 1^{st} \text{ order}$$

$$\left.\begin{array}{r} v_0 = 11111111 \\ v_1 = 01010101 \\ v_2 = 00110011 \\ v_3 = 00001111 \\ v_1v_2 = 00010001 \\ v_1v_3 = 00000101 \\ v_2v_3 = 00000011 \end{array}\right] 2^{nd} \text{ order} \qquad \left.\begin{array}{r} v_0 = 11111111 \\ v_1 = 01010101 \\ v_2 = 00110011 \\ v_3 = 00001111 \\ v_1v_2 = 00010001 \\ v_1v_3 = 00000101 \\ v_2v_3 = 00000011 \\ v_1v_2\,v_3 = 00000001 \end{array}\right] \text{3rd order}$$

(arithmetic operations are modulo 2)

Fig.1.Basis Vectors for (r, n = 2^m) Binary RM Code of Length n= $2^m = 2^3 = 8$

So, these are basis vectors of 3-RM, each of length 8. The linear combinations of all these vectors will give all the codewords present in rth order binary RM code.

III. TERNARY RM CODES

Def.3. For r and m, $0 \le r \le m$, r and m being any integers, there exists code known as rth order ternary RM code having length $n = 3^m$, information-symbols $k = 1 + {}^mC_1 + {}^mC_2 + \ . \ . \ . \ + {}^mC_r$, and minimum distance $d = 3^{m-r}$. It will be referred to as (r, n = 3^m) ternary Reed-Muller code or r-Reed-Muller ternary code.

Def.4. Let v_0 is a vector, whose $n = 3^m$ components are all 2's. Let there be an $m \times 3^m$ matrix having v_1 , v_2 , ... , v_m as row vectors, where its ith column describes the ternary representation of integer i, where i = 0, 1, ... , 3^m-1. Clearly, each of v_1, v_2, ... ,v_m has $n = 3^m$ components. As a result, the number of columns of the matrix will be $n = 3^m$. Then the (r, n = 3^m) ternary Reed-Muller code is k-dimensional vector-space having v_0, v_1, v_2, . . . ,v_m and also all vector products of these vectors taken r or fewer at a time as basis-vectors , k being $k = 1 + {}^mC_1 + {}^mC_2 + \ . \ . \ . \ + {}^mC_r$. It should again be noted that vector-product of the two vectors **u** and **v**, where **u** = (u_1, u_2, ... , u_n) and **v** = (v_1, v_2, ... ,v_n), is given by **u. v** = (u_1v_1, u_2v_2, ... , u_nv_n).

Let m = 2. Therefore, required variables will be v_0, v_1, and v_2. Length of ternary RM code will be $n = 3^m = 3^2 = 9$. Because $0 \le r \le m$ implies that $0 \le r \le 2$ which means that maximum value of r will be 2. So, r = 0, r = 1, and r = 2. So, ternary RM code will be of 0^{th} order, 1^{st} order, and 2^{nd} order. The basis vectors of this ternary RM code will be as shown below:

$$v_0 = 222222222] \ 0^{th} \text{ order}$$

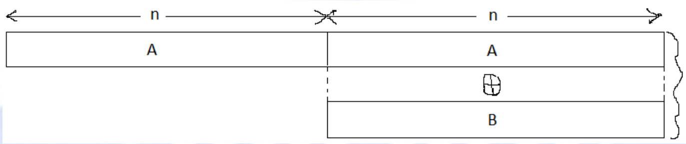
$$\left.\begin{array}{l} v_0 = 222222222 \\ v_1 = 012012012 \\ v_2 = 000111222 \end{array}\right] 1^{st} \text{ order} \qquad \left.\begin{array}{r} v_0 = 222222222 \\ v_1 = 012012012 \\ v_2 = 000111222 \\ v_1v_2 = 000012021 \end{array}\right] \text{2nd order}$$

(arithmetic operations are modulo 3)

Fig.2.Basis Vectors for (r, n = 3^m)Ternary RM Code of Length n=3^m = 3^2 = 9

The linear combinations of all these basis vectors will give all the codewords present in the ternary (r, n = 3^m, m = 2) RM code.

The rth order ternary RM codes can be interpreted in terms of the superposition as shown in the fig. 3., where two codes of length n superimpose to give a new code of length 2n.



BLK(1) BLK(2)

A: a codeword in an [n, k, d] code, B: a codeword in [n, k$'$, 2d] code

Fig.3.Construction of Super-imposed Code

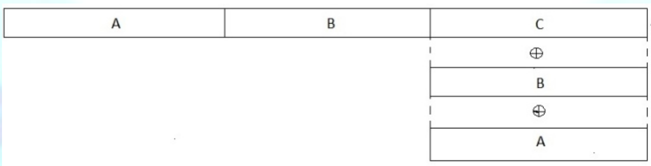
Constituent codes, which contain codewords A and B respectively, are called sub-codes. Therefore, super-imposed code can be decomposed into sub-codes which have codewords A, B respectively. Hence, inverse operation of super-imposition is decomposition.

Theorem 1. [MacWilliams-Sloane (1977)]: For r = 1, 2, . . . , m - 1, r being any integer, (r, 2^m) binary RM code can be split or decomposed into sub-codes: (i) (r, 2^{m-1}) binary RM code, (ii) (r - 1, 2^{m-1}) binary RM code.

This theorem can be generalised for ternary RM code. So, we have the following proposition:

Proposition 1. For r = 1, 2, ... , m - 1, r being any integer, the (r, 3^m) ternary RM code can be split or decomposed into sub-codes: (i) (r, 3^{m-1}) ternary RM code, (ii) (r - 1, 3^{m-1}) ternary RM code, (iii) (r - 2, 3^{m-1}) ternary RM code.

This proposition can be illustrated as in fig. 4:



BLK(1) BLK(2) BLK(3)

A: a codeword in ternary (r, 3^{m-1})RM code, B: a codeword in ternary (r - 1, 3^{m-1})RM code,

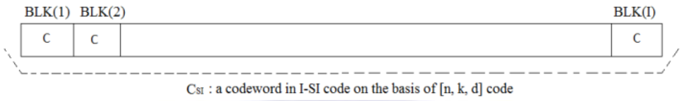
and C: a codeword in ternary (r - 2, 3^{m-1})RM code

Fig.4.Construction of Ternary (r, 3^m) Reed-Muller Code

So, construction of ternary (r, 3^m) RM code from sub-codes: (i) ternary (r, 3^{m-1})RM code, (ii) ternary (r - 1, 3^{m-1})RM code, (iii) ternary (r - 2, 3^{m-1})RM code, is like | u | v | u + v + w | construction. Hence ternary RM codes can be interpreted in terms of super-imposition.

IV. SIMPLE ITERATED CODES

Let [n, k, d] be the given ternary RM code. If a codeword in the [n, k, d] code is simply repeated I times, a new code is formed, which will be [n I, k, d I] code. It is called simple iterated code (SI). It is as shown in fig.5. as follows:



C_{SI}: a codeword in the I-SI code on the basis of the [n, k, d]code,

C: a codeword in the [n, k, d]code

Fig.5.Construction of SI Code

Here BLK (i) means ith block. The new super-imposed code is written as I-SI code on basis of the [n, k, d] code. The I-SI code is the [n I, k, d I]code.

Now let I-SI be simple iterated code on basis of the [n, k, d]code, which is ternary (r, n = 3^m) RM code. Let corresponding to the codeword C_{SI}, R be the received codeword. Let e_i, where i = 1, 2, ... , I be error-vectors in the BLK (i). It is assumed that these e_i satisfy following formulation:

$$\sum_{i=1}^{I} \text{wt.}(e_i) < d\, I / 3, \text{ where d is assumed to be a multiple of 3.}$$

Therefore, $C_{SI} = |C|C| \;.\;.\;.\; |C|$

and $R = |C \oplus e_1|C \oplus e_2| \;.\;.\;.\; |C \oplus e_I|$

Then we have following algorithm of decoding for the SI codes:

Step 1: Let i =1.

Step 2: Decode the $C \oplus e_i$ as $\hat{C}_i$ of received word R. It can be done by utilising any appropriate method. (Because length n of every block may be selected much shorter as compared to length n I of original I-SIcode, so it may be decoded easily utilising syndrome decoding etc.).

Step 3: If the error-correction be made in the step 2,

Then find out value of N_i using equation:

$$N_i = \sum_{j=1}^{I} \text{wt.}\left(C \oplus e_j \oplus \hat{C}_i \right) \tag{4.1}$$

If the error-detection is complete in the step 2,

then go to the step 5.

Step 4: Do Comparison of N_i with threshold-value d I / 3.

(i) If $N_i < d\, I / 3$, go to step 6.

(ii) If $N_i \geq d\, I / 3$, then go to the step 5.

Step 5: Now if i < I, then let i = i + 1 and go to the step 2.

If i = I, then error-detection gets completed.

Step 6: Now let C = $\hat{C}_i$. Error-correction is completed.

Using this decoding algorithm, the I-SI code on basis of [n, k, d]code can be decoded.

In the step 2, all blocks of received-word R cannot be erroneously corrected. It means either at least one of blocks is successfully corrected or all blocks are error-detected.

Now consider former case, i.e. when at least one of blocks is successfully corrected. Let $\hat{C}_k$ be correct-version of C.

If the total number of errors is < d I / 3,

then we will have: $N_k = \sum_{j=1}^{I} \text{wt.}(e_j) < dI/3$.

If the total number of errors = d I / 3,

then $N_k = \sum_{j=1}^{I} \text{wt.}(e_j) = d\,I/3$, where d I is assumed to a multiple of 3.

If $C \oplus e_h$ gets decoded as $\hat{C}_h \neq C$, then:

$$N_h = \sum_{j=1}^{I} \text{wt.}\left(C \oplus e_j \oplus \hat{C}_h\right)$$

$$\geq \sum_{j=1}^{I}[\text{wt.}\left(C \oplus \hat{C}_h\right) - \text{wt.}(e_j)]$$

$$\geq d\,I - d\,I / 3.$$

$$= (2/3)\, d\,I.$$

$$\geq d\,I / 3.$$

Therefore, $N_h \geq d\,I / 3$.

In the latter case, i.e. when all blocks are error-detected, then it is clear that error-detection is completed in the step 5. In this latter case, d will be a multiple of 3, i.e. when all blocks are error-detected, this happens only if d is a multiple of 3.

V. SUPER-IMPOSITION AND DECOMPOSITION OF TERNARY RM CODES

By using Proposition 1., original ternary (r, 3^m)Reed-Muller code, where $1 \leq r \leq m - 1$, can be split or decomposed into three sub-codes: (i) the ternary (r, 3^{m-1})Reed-Muller code, (ii) the ternary (r - 1, 3^{m-1})Reed-Muller code, and (iii) the ternary (r - 2, 3^{m-1})Reed-Muller code.

The ternary (r, 3^{m-1})RM subcode will further be decomposed into three ternary RM subcodes: ternary (r, 3^{m-2}) Reed-Muller code, ternary (r - 1, 3^{m-2})Reed-Muller code, and ternary (r - 2, 3^{m-2}) Reed-Muller code; the ternary (r - 1, 3^{m-1}) Reed-Muller subcode will further be decomposed into three ternary RM subcodes: ternary (r - 1, 3^{m-2})Reed-Muller code, ternary (r - 2, 3^{m-2})Reed-Muller code, and ternary (r - 3, 3^{m-2}) Reed-Muller code; the ternary (r - 2, 3^{m-1})Reed-Muller subcode will further be decomposed into three ternary RM subcodes: ternary (r - 2, 3^{m-2})Reed-Muller code, ternary (r - 3, 3^{m-2})Reed-Muller code, and ternary (r - 4, 3^{m-2})Reed-Muller code. And so on.

This operation is repeated until and unless subcodes of the lowest order become ternary 0^{th} order RM code.

***Example 1*:** Consider ternary (r, 3^m) RM code, in which m = 3 and r = 2 so that $0 \le r \le m$. So, this code will be ternary (2, 3^3) i.e. (2, 27) RM code. Its three ternary RM sub-codes will be (r, 3^{m-1}), (r - 1, 3^{m-1}) (r - 2, 3^{m-1}) i.e. (2, 3^{3-1}), (2 - 1, 3^{3-1}), (2 - 2, 3^{3-1}) i.e. (2, 9), (1, 9) (0, 9) RM codes. The basis vectors of ternary (2, 9) RM code will be v_0 = 222222222, v_1= 012012012, v_2 = 000111222, v_1v_2 = 000012021; that of ternary (1, 9) RM code will be v_0 = 222222222, v_1= 012012012, v_2 = 000111222; and that of ternary (0, 9) RM code will be v_0 = 222222222. Linear combinations of these basis-vectors will give all codewords of the respective codes. Therefore, these three sub-codes will be:

(r, 3^{m-1}) = (2, 9) = {222222222, 012012012, 000111222, 000012021, 201201201, 222000111, 222201210, 012120201, 012021000, 000120210, 000000000, 111111111, 120210012, . . .};

(r - 1, 3^{m-1}) = (1, 9) = {222222222, 012012012, 000111222, 201201201, 222000111, 012120201, 201012120, 000000000, 111111111, 120201012, . . .}

(r - 2, 3^{m-1}) = (0, 9) = {222222222, 000000000, 111111111}

Let u ∈ (r, 3^{m-1}) = (2, 9) be u = 000000000,

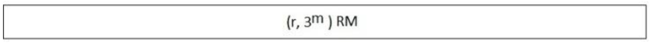
and v ∈ (r - 1, 3^{m-1}) = (1, 9) be v = 111111111,

and w ∈ (r - 2, 3^{m-1}) = (0, 9) be w = 111111111.

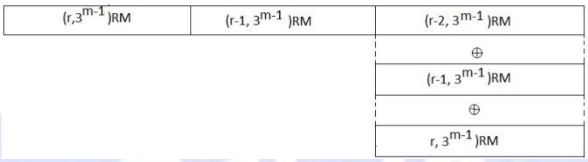
Therefore, | u | v | u + v + w | = | 000000000 | 111111111| (000000000 + 11111111 + 111111111 | = | 000000000 | 111111111 | 222222222 |. All these three blocks are present in (r, 3^{m-1}) = (2, 9), (r - 1, 3^{m-1}) = (1, 9), and (r - 2, 3^{m-1}) = (0, 9) sub-codes. So,| u | v | u + v + w | = 000000000 111111111222222222. This codeword will belong to (r, 3^m) = (2, 3^3) = (2, 27) ternary RM code. For this, order will be r = 2, and block-length will be equal to 3^3 = 27, and value of m will be 3, and hence its basis-vectors will be: v_0=222222222222222222222222222,v_1=012012012012012012012012012,v_2=00011122200 0111222000111222,v_3=000000000111111111222222222,v_1v_2=00001202100001202100001 2021,v_1v_3=000000000012012012021021021,v_2v_3=000000000000111222000222111.

Also we note that in code (2, 27) = (2, 3^3), r = 2, m = 3, so for this code, value of k will be: k = 1 + ${}^{m}C_1$ + ${}^{m}C_2$ + . . . + ${}^{m}C_r$ = 1 + ${}^{3}C_1$ + ${}^{3}C_2$ = 1 + 3 + 3 = 7. Hence number of codewords in code (r, 3^m) = (2, 3^3) will be = 3^k = 3^7 = 2187. Every | u | v | u + v + w |, for values u ∈ (r, 3^{m-1}), v ∈ (r - 1, 3^{m-1}), w ∈ (r - 2, 3^{m-1}) will be one of these 2187 codewords of the code (r, 3^m) = (2, 3^3) = (2, 27).

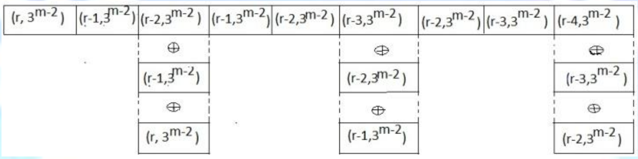
If decomposition is repeatedly performed on each sub-code of the original ternary $(r, 3^m)$ RM code, and this process is repeated μ times, then this is called μ-decomposition. It is as shown in fig 6 as follows:



0-Decomposition



1-Decomposition



2-Decomposition

Fig.6. Displaying the μ-Decomposition of Ternary $(r, 3^m)$ RM Code:

(i) 0-Decomposition, (ii) 1-Decomposition, (iii) 2-Decomposition.

This discussion is generalised in the form of theorem as follows:

Theorem 2. Given $\mu \in \{1, 2, \ . \ . \ . \ , m - r\}$, where μ is an integer, the ternary $(r, 3^m)$ Reed-Muller code can be split or decomposed into ternary $(r - j, 3^{m-\mu})$ Reed-Muller codes, where $j = 0, 1, \ . \ . \ . \ , r$ with same minimum distance $3^{m-r-\mu}$, each of which constitutes the 3^μ - code.

Therefore, if we take μ = 1, then subcodes will be ternary $(r - j, 3^{m-\mu})$ RM codes, i. e. ternary $(r - j, 3^{m-1})$ RM codes. Since $j = 0, 1, \ . \ . \ . \ , r$, therefore, the corresponding subcodes will be: ternary $(r, 3^{m-1})$ RM subcode, ternary $(r - 1, 3^{m-1})$ RM code, ternary $(r - 2, 3^{m-1})$ RM code, ... , ternary $(0, 3^{m-1})$ RM code.

If we take μ = 2, then subcodes will be ternary $(r - j, 3^{m-\mu})$ RM codes, i. e. ternary $(r - j, 3^{m-2})$ RM codes. Since $j = 0, 1, \ . \ . \ . \ , r$, therefore, the corresponding subcodes will be: ternary $(r, 3^{m-2})$ RM subcode, ternary $(r - 1, 3^{m-2})$ RM code, ternary $(r - 2, 3^{m-2})$ RM code, . . . , ternary $(0, 3^{m-2})$ RM code.

If we take μ = 3, then subcodes will be ternary $(r - j, 3^{m-\mu})$ RM codes, i. e. ternary $(r - j, 3^{m-3})$ RM codes. Since $j = 0, 1, \ . \ . \ . \ , r$, therefore, the corresponding subcodes will be: ternary $(r, 3^{m-3})$ RM subcode, ternary $(r - 1, 3^{m-3})$RM code, ternary $(r-2, 3^{m-3})$RM code, . . . , ternary $(0, 3^{m-3})$ RM code.

. . . .

. . . .

Lastly, we take μ = m - r, then subcodes will be ternary $(r - j, 3^{m-\mu})$ RM codes, i. e. ternary $(r - j, 3^r)$ RM codes. Since $j = 0, 1, \ . \ . \ . \ , r$, therefore, the corresponding subcodes will be: ternary (r, 3^r) RM subcode, ternary (r - 1, 3^r) RM code, ternary (r - 2, 3^r) RM code, . . . , ternary (0, 3^r) RM code.

On the other hand, if we take j = 0, then subcodes will be ternary (r - j, $3^{m-\mu}$) RM codes, i. e. ternary (r , $3^{m-\mu}$) RM codes. So, since $\mu \in \{1, 2, \ldots, m - r\}$, therefore, the corresponding subcodes will be: ternary (r, 3^{m-1}) RM subcode, ternary (r, 3^{m-2}) RM code, ternary (r, 3^{m-3}) RM code, . . . , ternary (r, 3^r) RM code.

If we take j = 1, then subcodes will be ternary (r - j, $3^{m-\mu}$) RM codes, i. e. ternary (r - 1, $3^{m-\mu}$) RM codes. So, because $\mu \in \{1, 2, \ldots, m - r\}$, therefore, the corresponding subcodes will be: ternary (r - 1, 3^{m-1}) RM subcode, ternary (r - 1, 3^{m-2}) RM code, ternary (r - 1, 3^{m-3}) RM code, . . . , ternary (r - 1, 3^r) RM code.

If we take j = 2, then subcodes will be ternary (r - j, $3^{m-\mu}$) RM codes, i. e. ternary (r - 2, $3^{m-\mu}$) RM codes. So, because $\mu \in \{1, 2, \ldots, m - r\}$, therefore, the corresponding subcodes will be: ternary (r - 2, 3^{m-1}) RM subcode, ternary (r - 2, 3^{m-2}) RM code, ternary (r - 2, 3^{m-3})RM code, . . . , ternary (r - 2, 3^r) RM code.

. . . .

. . . .

Lastly, if we take j = r, then subcodes will be ternary (r - j, $3^{m-\mu}$) RM codes, i. e. ternary (0, $3^{m-\mu}$) RM codes. So, because $\mu \in \{1, 2, \ldots, m - r\}$, therefore, the corresponding subcodes will be: ternary (0, 3^{m-1}) RM subcode, ternary (0, 3^{m-2}) RM code, ternary (0, 3^{m-3}) RM code, . . . , ternary (0, 3^r) RM code.

Example 2:

Consider codeword in original ternary (r, 3^m) Reed-Muller code as shown below:

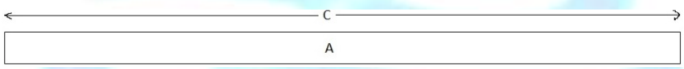


Fig.7. A Codeword in Original Ternary (r, 3^m)RM Code

Consider three equal blocks of the above codeword as under:

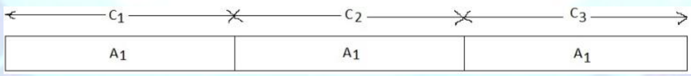


Fig.8.Three Equal Blocks of Codeword in Original Ternary(r,3^m) RM Code

To obtain B_1, Add $|C_1|$ and $|C_2|$ to $|C_3|$.

Therefore: $B_1 = |C_1 \oplus C_2 \oplus C_3|$.

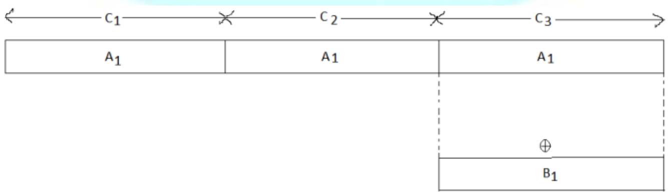


Fig.9. Obtaining B_1

To obtain B_2's:

Add above B_1 to $|C_3|$ to get $|C_3^{/}|$, so $|C_3^{/}| = |C_3| \oplus B_1$,

and $|C_1|$ gives $|C_1^{/}|$ as such, i.e. $|C_1^{/}| = |C_1|$,

and $|C_2|$ gives $|C_2^{/}|$ as such, i.e. $|C_2^{/}| = |C_2|$.

So, the fig. is as follows:

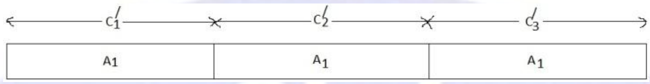


Fig.10. Obtaining B_2's

Therefore, first $B_2 = |C_1^{/}|$, second $B_2 = |C_2^{/}|$, third $B_2 = |C_3^{/}|$.

Hence, all A_1's are obtained as follows:

first $A_1 = |C_1^{/}|$, second $A_1 = |C_2^{/}|$, third $A_1 = |C_3^{/}|$.

Here, first A_1 represents a codeword in sub-code $(r, 3^{m-1})$, second A_1 represents a codeword in sub-code $(r - 1, 3^{m-1})$, third A_1 represents a codeword in sub-code $(r - 2, 3^{m-1})$, where $(r, 3^{m-1})$, and $(r - 1, 3^{m-1})$, and $(r - 2, 3^{m-1})$ are the sub-codes of the original ternary $(r, 3^m)$ RM code.

Further, consider three equal blocks of each of A_1. So, we have:

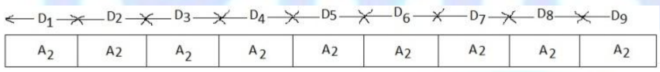


Fig.11. Considering Three Equal Blocks of Each of the A_1's

To obtain B_3: Add $|D_1|D_2|D_3|$ and $|D_4|D_5|D_6|$ to $|D_7|D_8|D_9|$ to get B_3.

Therefore: $B_3 = |D_1 \oplus D_4 \oplus D_7| D_2 \oplus D_5 \oplus D_8| D_3 \oplus D_6 \oplus D_9|$.

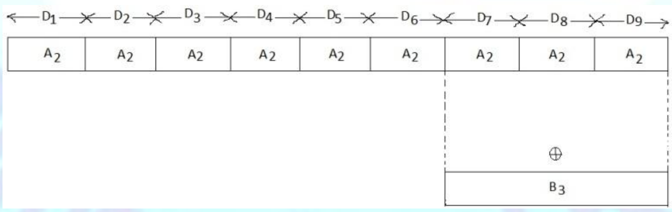


Fig.12. Obtaining B_3

To obtain B_4's:

Firstly: Add above B_3 to $|D_7|D_8|D_9|$ to get $|D_7^{/}| D_8^{/}| D_9^{/}|$,

so $|D_7^{/}| D_8^{/}| D_9^{/}| = |D_7|D_8|D_9| \oplus B_3$,

and $|D_1|D_2|D_3|$ gives $|D_1^{/}| D_2^{/}| D_3^{/}|$ as such,

i.e. $|D_1^{/}| D_2^{/}| D_3^{/}| = |D_1|D_2|D_3|$,

and $|D_4|D_5|D_6|$ gives $|D_4^{/}| D_5^{/}| D_6^{/}|$ as such, i.e. $|D_4{}'|\,D_5{}'|\,D_6{}'| = |D_4|D_5|D_6|$.

Secondly: Add $|D_1{}'|$ and $|D_2{}'|$ to $|D_3{}'|$ to get first B_4,

i.e. first $B_4 = |D_1{}'| \oplus |D_2{}'| \oplus |D_3{}'|$,

Add $|D_4{}'|$ and $|D_5{}'|$ to $|D_6{}'|$ to get second B_4,

i.e. second $B_4 = |D_4{}'| \oplus |D_5{}'| \oplus |D_6{}'|$,

Add $|D_7{}'|$ and $|D_8{}'|$ to $|D_9{}'|$ to get third B_4,

i.e. third $B_4 = |D_7{}'| \oplus |D_8{}'| \oplus |D_9{}'|$.

Therefore, all the A_2's are obtained as follows:

First $A_2 = |D_1{}'|$ **,** Second $A_2 = |D_2{}'|$, Third $A_2 = |D_3{}^{(1)} \oplus$ first $B_4|$

Fourth $A_2 = |D_4{}'|$, Fifth $A_2 = |D_5{}'|$, Sixth $A_2 = |D_6{}' \oplus$ second $B_4|$

Seventh $A_2 = |D_7{}'|$, Eighth $A_2 = |D_8{}'|$, Ninth $A_2 = |D_9{}' \oplus$third $B_4|$

All this is shown in fig.13 as follows:

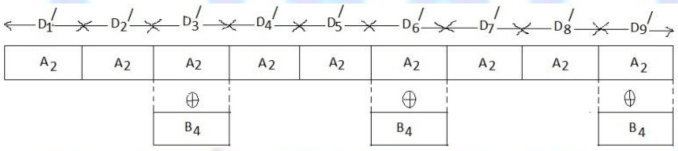


Fig.13. Obtaining B_4's

Here, first A_2 represents a codeword in sub-code (r, 3^{m-2}), second A_2 represents a codeword in sub-code (r - 1, 3^{m-2}), third A_2 represents a codeword in sub-code (r - 2, 3^{m-2}), where (r, 3^{m-2}), (r - 1, 3^{m-2}), and (r - 2, 3^{m-2}) are the sub-codes of sub-code (r, 3^{m-1}). Fourth A_2 represents a codeword in sub-code (r - 1, 3^{m-2}), fifth A_2 represents a codeword in sub-code (r - 2, 3^{m-2}), sixth A_2 represents a codeword in sub-code (r - 3, 3^{m-2}), where (r - 1, 3^{m-2}), and (r - 2, 3^{m-2}), and (r - 3, 3^{m-2}) are the sub-codes of sub-code (r - 1, 3^{m-1}). Seventh A_2 represents a codeword in sub-code (r - 2, 3^{m-2}), eighth A_2 represents a codeword in sub-code (r - 3, 3^{m-2}), ninth A_2 represents a codeword in sub-code (r - 4, 3^{m-2}), where (r - 2, 3^{m-2}), and (r -3, 3^{m-2}), and (r- 4, 3^{m-2}) are the sub-codes of sub-code (r - 2, 3^{m-1}).

Further, consider three equal blocks of each of A_2's. So, we have:

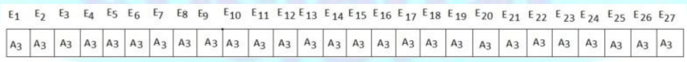


Fig.14.Considering Three Equal Blocks of Each of the A_2's

To obtain B_5's :

Add $|E_1|E_2|E_3|$ and $|E_4|E_5|E_6|$ to $|E_7|E_8|E_9|$ to get first B_5.

Add $|E_{10}|E_{11}|E_{12}|$ and $|E_{13}|E_{14}|E_{15}|$ to $|E_{16}|E_{17}|E_{18}|$ to get second B_5.

Add $|E_{19}|E_{20}|E_{21}|$ and $|E_{22}|E_{23}|E_{24}|$ to $|E_{25}|E_{26}|E_{27}|$ to get third B

Therefore:

First $B_5 = |E_1 \oplus E_4 \oplus E_7| E_2 \oplus E_5 \oplus E_8| E_3 \oplus E_6 \oplus E_9|$.

Second $B_5 = |E_{10} \oplus E_{13} \oplus E_{16}| E_{11} \oplus E_{14} \oplus E_{17}| E_{12} \oplus E_{15} \oplus E_{18}|$.

Third $B_5 = |E_{19} \oplus E_{22} \oplus E_{25}| E_{20} \oplus E_{23} \oplus E_{26}| E_{21} \oplus E_{24} \oplus E_{27}|$.

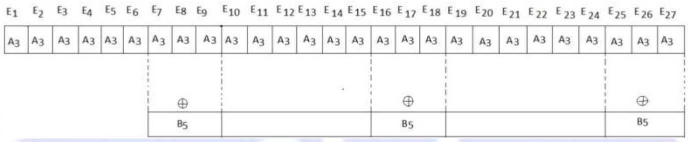


Fig.15.Obtaining B_5's

To obtain B_6's:

Firstly:

Add above first B_5 to $|E_7|E_8|E_9|$ to get $|E_7^{/}| E_8^{/}| E_9^{/}|$,

so $|E_7^{/}| E_8^{/}| E_9^{/}| = |E_7|E_8|E_9| \oplus$ first B_5,

and $|E_1|E_2|E_3|$ gives $|E_1^{/}| E_2^{/}| E_3^{/}|$ as such,

i.e. $|E_1^{/}| E_2^{/}| E_3^{/}| = |E_1|E_2|E_3|$,

and $|E_4|E_5|E_6|$ gives $|E_4^{/}| E_5^{/}| E_6^{/}|$ as such,

i.e. $|E_4^{/}| E_5^{/}| E_6^{/}| = |E_4|E_5|E_6|$.

Add above second B_5 to $|E_{16}|E_{17}|E_{18}|$ to get $|E_{16}^{/}| E_{17}^{/}| E_{18}^{/}|$,

so $|E_{16}^{/}| E_{17}^{/}| E_{18}^{/}| = |E_{16}|E_{17}|E_{18}| \oplus$ second B_5,

and $|E_{10}|E_{11}|E_{12}|$ gives $|E_{10}^{/}| E_{11}^{/}| E_{12}^{/}|$ as such,

i.e. $|E_{10}^{/}| E_{11}^{/}| E_{12}^{/}| = |E_{10}|E_{11}|E_{12}|$,

and $|E_{13}|E_{14}|E_{15}|$ gives $|E_{13}^{/}| E_{14}^{/}| E_{15}^{/}|$ as such,

i.e. $|E_{13}^{/}| E_{14}^{/}| E_{15}^{/}| = |E_{13}|E_{14}|E_{15}|$.

Add above third B_5 to $|E_{25}|E_{26}|E_{27}|$ to get $|E_{25}^{/}| E_{26}^{/}| E_{27}^{/}|$,

so $|E_{25}^{/}| E_{26}^{/}| E_{27}^{/}| = |E_{25}|E_{26}|E_{27}| \oplus$ third B_5,

and $|E_{19}|E_{20}|E_{21}|$ gives $|E_{19}^{/}| E_{20}^{/}| E_{21}^{/}|$ as such,

i.e. $|E_{19}^{/}| E_{20}^{/}| E_{21}^{/}| = |E_{19}|E_{20}|E_{21}|$,

and $|E_{22}|E_{23}|E_{24}|$ gives $|E_{22}^{/}| E_{23}^{/}| E_{24}^{/}|$ as such,

i.e. $|E_{22}^{/}| E_{23}^{/}| E_{24}^{/}| = |E_{22}|E_{23}|E_{24}|$.

Secondly:

Add $|E_1^{/}|$ and $|E_2^{/}|$ to $|E_3^{/}|$ to get first B_6, i.e. first $B_6 = |E_1^{/}| \oplus |E_2^{/}| \oplus |E_3^{/}|$,

Add $|E_4^{/}|$ and $|E_5^{/}|$ to $|E_6^{/}|$ to get second B_6,

i.e. second $B_6 = |E_4^{/}| \oplus |E_5^{/}| \oplus |E_6^{/}|$,

Add $|E_7^{/}|$ and $|E_8^{/}|$ to $|E_9^{/}|$ to get third B_6,

i.e. third $B_6 = |E_7^{/}| \oplus |E_8^{/}| \oplus |E_9^{/}|$.

Add $|E_{10}^{/}|$ and $|E_{11}^{/}|$ to $|E_{12}^{/}|$ to get fourth B_6,

i.e. fourth $B_6 = |E_{10}^{/}| \oplus |E_{11}^{/}| \oplus |E_{12}^{/}|$,

Add $|E_{13}^{/}|$ and $|E_{14}^{/}|$ to $|E_{15}^{/}|$ to get fifth B_6,

i.e. fifth $B_6 = |E_{13}^{/}| \oplus |E_{14}^{/}| \oplus |E_{15}^{/}|$,

Add $|E_{16}^{/}|$ and $|E_{17}^{/}|$ to $|E_{18}^{/}|$ to get sixth B_6,

i.e. sixth $B_6 = |E_{16}^{/}| \oplus |E_{17}^{/}| \oplus |E_{18}^{/}|$,

Add $|E_{19}^{/}|$ and $|E_{20}^{/}|$ to $|E_{21}^{/}|$ to get seventh B_6,

i.e. seventh $B_6 = |E_{19}^{/}| \oplus |E_{20}^{/}| \oplus |E_{21}^{/}|$,

Add $|E_{22}^{/}|$ and $|E_{23}^{/}|$ to $|E_{24}^{/}|$ to get eighth B_6,

i.e. eighth $B_6 = |E_{22}^{/}| \oplus |E_{23}^{/}| \oplus |E_{24}^{/}|$,

Add $|E_{25}^{/}|$ and $|E_{26}^{/}|$ to $|E_{27}^{/}|$ to get ninth B_6,

i.e. ninth $B_6 = |E_{25}^{/}| \oplus |E_{26}^{/}| \oplus |E_{27}^{/}|$.

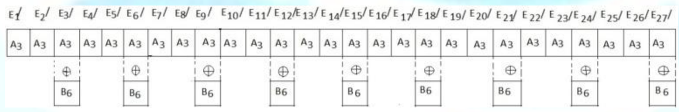


Fig.16.Obtaining B_6's

Therefore, all the A_3's are obtained as follows:

First $A_3 = |E_1^{/}|$, Second $A_3 = |E_2^{/}|$, Third $A_3 = |E_3^{(1)} \oplus$ first $B_6|$

Fourth $A_3 = |E_4^{/}|$, Fifth $A_3 = |E_5^{/}|$, Sixth $A_3 = |E_6^{/} \oplus$ second $B_6|$

Seventh $A_3 = |E_7^{/}|$, Eighth $A_3 = |E_8^{/}|$, Ninth $A_3 = |E_9^{/} \oplus$ third $B_6|$

Tenth $A_3 = |E_{10}^{/}|$, Eleventh $A_3 = |E_{11}^{/}|$, Twelfth $A_3 = |E_{12}^{/} \oplus$ fourth $B_6|$

Thirteenth $A_3 = |E_{13}^{/}|$, Fourteenth $A_3 = |E_{14}^{/}|$, Fifteenth $A_3 = |E_{15}^{/} \oplus$ fifth $B_6|$

Sixteenth $A_3 = |E_{16}^{/}|$, Seventeenth $A_3 = |E_{17}^{/}|$, Eighteenth $A_3 = |E_{18}^{/} \oplus$ sixth $B_6|$

Nineteenth $A_3 = |E_{19}^{/}|$, Twentieth $A_3 = |E_{20}^{/}|$, Twenty-first $A_3 = |E_{21}^{/} \oplus$ seventh $B_6|$

Twenty-second $A_3 = |E_{22}^{/}|$, Twenty-third $A_3 = |E_{23}^{/}|$, Twenty-fourth $A_3 = |E_{24}^{/} \oplus$ eighth $B_6|$

Twenty-fifth $A_3 = |E_{25}{}^{/}|$, Twenty-sixth $A_3 = |E_{26}{}^{/}|$, Twenty-seventh $A_3 = |E_{27}{}^{/} \oplus$ ninth $B_6|$

Here, first A_3 represents a codeword in sub-code (r, 3^{m-3}), second A_3 represents a codeword in sub-code (r - 1, 3^{m-3}), third A_3 represents a codeword in sub-code (r - 2, 3^{m-3}); (r, 3^{m-3}), (r - 1, 3^{m-3}), (r - 2, 3^{m-3}) being the sub-codes of sub-code (r, 3^{m-2}). Fourth A_3 represents a codeword in sub-code (r - 1, 3^{m-3}), fifth A_2 represents a codeword in sub-code (r - 2, 3^{m-3}), sixth A_3 represents a codeword in sub-code (r - 3, 3^{m-3}), where (r - 1, 3^{m-3}), (r - 2, 3^{m-3}), (r - 3, 3^{m-3}) are the sub-codes of sub-code (r - 1, 3^{m-2}). Seventh A_3 represents a codeword in sub-code (r - 2, 3^{m-3}), eighth A_3 represents a codeword in sub-code (r - 3, 3^{m-3}), ninth A_3 represents a codeword in sub-code (r - 4, 3^{m-3}), where (r - 2, 3^{m-3}), (r - 3, 3^{m-3}), (r - 4, 3^{m-3}) are the sub-codes of sub-code (r-2, 3^{m-2}). Tenth A_3 represents a codeword in sub-code (r - 1, 3^{m-3}), eleventh A_3 represents a codeword in sub-code (r - 2, 3^{m-3}), twelfth A_3 represents a codeword in sub-code (r - 3, 3^{m-3}), where (r - 1, 3^{m-3}), (r - 2, 3^{m-3}), (r - 3, 3^{m-3}) are the sub-codes of sub-code (r - 1, 3^{m-2}). Thirteenth A_3 represents a codeword in sub-code (r - 2, 3^{m-3}), fourteenth A_2 represents a codeword in sub-code (r - 3, 3^{m-3}), fifteenth A_3 represents a codeword in sub-code (r - 4, 3^{m-3}), where (r - 2, 3^{m-3}), (r - 3, 3^{m-3}), (r - 4, 3^{m-3}) are the sub-codes of sub-code (r - 2, 3^{m-2}). Sixteenth A_3 represents a codeword in sub-code (r - 3, 3^{m-3}), seventeenth A_3 represents a codeword in sub-code (r - 4, 3^{m-3}), eighteenth A_3 represents a codeword in sub-code (r-5, 3^{m-3}), where (r - 3, 3^{m-3}), and (r - 4, 3^{m-3}), and (r - 5, 3^{m-3}) are the sub-codes of sub-code (r - 3, 3^{m-2}). Nineteenth A_3 represents a codeword in sub-code (r - 2, 3^{m-3}), twentieth A_3 represents a codeword in sub-code (r - 3, 3^{m-3}), twenty-first A_3 represents a codeword in sub-code (r - 4, 3^{m-3}), where (r - 2, 3^{m-3}), (r - 3, 3^{m-3}), (r - 4, 3^{m-3}) are the sub-codes of sub-code (r - 2, 3^{m-2}). Twenty-second A_3 represents a codeword in sub-code (r - 3, 3^{m-3}), twenty-third A_2 represents a codeword in sub-code (r - 4, 3^{m-3}), twenty-fourth A_3 represents a codeword in sub-code (r - 5, 3^{m-3}), where (r - 3, 3^{m-3}), and (r - 4, 3^{m-3}), and (r - 5, 3^{m-3}) are the sub-codes of sub-code (r - 3, 3^{m-2}). Twenty-fifth A_3 represents a codeword in sub-code (r - 4, 3^{m-3}), twenty-sixth A_3 represents a codeword in sub-code (r - 5, 3^{m-3}), twenty-seventh A_3 represents a codeword in sub-code (r - 6, 3^{m-3}), where (r - 4, 3^{m-3}), and (r - 5, 3^{m-3}), and (r - 6, 3^{m-3}) are the sub-codes of sub-code (r - 4, 3^{m-2}). And so on.

So, original (r, 3^m) Ternary RM code is decomposed into sub-codes of lower orders which are also Ternary RM codes. In a reversal way, it can be said that the (r, 3^m) Ternary RM code is obtained from Ternary RM codes of lower orders. One more thing is clear, which is that the Ternary RM codes of lower orders and the (r, 3^m) Ternary RM code, are all SI codes. Hence all these Ternary RM codes can be decoded with the help of decoding algorithm for SI codes.

The direct-sum construction | u | v | means set of all vectors of the type | u | v |, u ∈ C_1 and v ∈ C_2, C_1, C_2 being [n_1, M_1, d_1], and [n_2, M_2, d_2] codes respectively, resulting in a new code [$n_1 + n_2$, $M_1 + M_2$, d = min.{d_1, d_2}]. Then we have | u | u + v | construction, which means set of all vectors of the type | u | u + v |, u ∈ C_1 and v ∈ C_2, C_1, C_2 being [n, M_1, d_1] and [n, M_2, d_2] codes respectively, resulting in a new code [2n, $M_1 + M_2$, d = min.{$2d_1$, d_2}]. As compared to direct-sum construction | u | v |, the | u | u + v | construction gives us a new code of increased block-length, and minimum distance may also be more. We have used | u | u + v + w | construction, u ∈ C_1, v ∈ C_2, and w ∈ C_3, which gives us a new code of further increased block-length, and minimum distance may also be furthermore, new code being [3n, $M_1 + M_2 + M_3$, d = min.{$2d_1$, $2d_2$, d_2}], where code C_1 is [n, M_1, d_1] code, code C_2 is [n, M_2, d_2] code, and code C_3 is [n, M_3, d_3] code.

Simple Iterated (SI) code is defined as a new code which is formed when a codeword u in given code C_1 is simply repeated, that is, it is simply | u | u | u | . . . | u |. If a codeword u is repeated two times, then this construction will become as | u | u | construction. It leads to SI code. Now if in | u | u | construction, we replace second u by codeword v belonging to another code C_2, then it becomes | u | v | construction. It is a new form of SI code. Similarly, if in | u | v | construction, the second codeword v belonging to code C_2, is replaced by u + v, then it becomes | u | u + v | construction. It is still another new form of SI code. Similarly, | u | v | u + v + w | construction, u ∈ code C_1, v ∈ code C_2, and w ∈ code C_3, is still another new form of SI code. In all these constructions: | u | u |, and | u | v |, and | u | u + v |, and | u | v | u + v + w |, the common feature is that the codewords are placed side-ways to form codewords of a new code, and this feature is the content of SI codes. Hence all these constructions: | u | u |, and | u | v |, and | u | u + v |, and | u | v | u + v + w |, give rise to the SI codes on basis of code C_1, C_1 and C_2, C_1 and $C_1 \oplus C_2$, C_1 and C_2 and $C_1 \oplus C_2 \oplus C_3$ respectively.

In these new codes given by constructions: | u | u |, | u | v |, | u | u + v |, and | u | v | u + v + w |; u, v, u + v, u + v + w, etc. denote the blocks of codeword of the new code. Hence every codeword of the new code generated by these constructions is composed of blocks, these being of equal length. So, these new codes can be decoded by algorithm which is there for the SI codes.

VI. GENERAL DECODING ALGORITHM FOR THE TERNARY RM CODES

So, decoding algorithm (general) for Ternary RM code (r, 3^m) and sub-codes of lower orders will be as follows:

Step 1: Get the Ternary RM code (r, 3^m) and sub-codes of lower orders in SI form.

Step 2: Decode all these SI codes with the help of algorithm for SI codes.

VII. CONCLUSION

Ternary RM codes are interpreted in terms of super-imposition. A new algorithm of decoding for class of Simple Iterated codes is proposed. It plays central role in the decoding algorithm for the Ternary RM codes. For same value of m, the (r, n = 3^m) ternary RM code will have larger block-length n, as compared to block-length n of (r, n = 2^m) binary RM code. The larger value of length n will help to strengthen the role of safeguarding the transmission of the message. Also we shall have more number of codewords in (r, n = 3^m) ternary RM code, i.e. 3^k as compared to 2^k in (r, n = 3^m) binary RM code. This will enhance the utility of ternary RM code. The detection and correction capability of a code depends upon the value of d, it is directly proportional to the value of d. Hence larger value of minimum distance d = 3^{m-r} of ternary RM code as compared to value of d = 2^{m-r} of binary RM code, will increase the detection and correction capability of the ternary RM code. So, as compared to binary RM code, ternary RM code has stronger role of securing the transmission of the messages, has enhanced utility, and has increased detection and correcting capability.

REFERENCES

1. **Weldon Jr., E.J. (**1969**),** *Some Results On Majority-Logic Decoding* in *Error-Correcting Codes*, H.B. Mann, Ed. New York: John Wiley, pp.149-162.

2. **Chen, C.L. (1972),** *Note on Majority-Logic Decoding of Finite Geometry Codes*, IEEE Transactions on Information Theory, Vol. IT-18, pp.539-541.

3. **MacWilliams, F.J. and Sloane, N.J. (1977),** *The Theory of Error Correcting Codes.* Amsterdam: North Holland.

4. **Tokiwa, Sugimura, Kasahara, and Namekawa (1982),** *New Decoding Algorithm for Reed-Muller Codes*, IEEE Transactions on Information Theory, Vol. IT-28, No. 5, pp.779-787.

5. **Vinocha, O.P., Bhullar, J.S., Brar, B.S., (2013),** *Covering Radius of RM Binary Codes*, ARPN Journal of Science and Technology (USA), **3**(6), pp. 223-233.