Error Detection and Correction using Fast Coding

Khalid. Faraj
Computer Science
Wajdi Institute of Technology
Jerusalem, Mount of Olives. P.O.Box 19014
Palestine
kfaraj@wit.edu.ps http://www.wit.edu.ps

Abstract:-This paper presents new methods to encode the message, detects the error, and corrects the message in the communication processes. These methods have been developed based on Reed Muller basic matrix. The key point for the implementation of error-free communication is the encoding of the information to be transmitted in such a way that some extent of redundancy is included in the encoded data, and a method for efficient decoding at the receiver is available. These two requirements have been achieved in the new method in an efficient and simple way. The new methods are demonstrated using some examples, and have give a good result.

Keywords:- communication, coding, encoding messages, corrects the message.

1. Introduction

A central problem of coding theory is reliable communication over an unreliable channel. All solutions to this problem, in some form or another, depend on the basic idea of encoding messages with some redundancy, allowing the receiver to detect and correct whatever errors may arise during transmission through the channel. The main goal is to minimize the amount of redundancy while maximizing the quantity of errors that can be corrected. Networks and other communication systems must be able to transfer data from the source to the receiver with complete accuracy. A system that cannot guarantee that the data received by one device are identical to the data transmitted by another device is essentially useless.

The theory of error correction is concerned with sending reliably information over a noisy channel that introduces errors into the transmitted data. The goal of this research is to design coding schemes which are capable of detecting and correcting such errors. The setting is usually modeled as follows: a transmitter starts with some message, which is represented as a string of symbols over some alphabet. The transmitter encodes the message into a longer string over the same alphabet, and transmits the block of data over a channel. The channel introduces errors (or noise) by changing some of the symbols of the transmitted block, and then delivers the corrupted block to the receiver. Finally, the receiver attempts to decode the block, hopefully to the intended message. Whenever the transmitter wants to transmit a new message, the process is repeated. Two factors are of special interest in this setting. The first is the information rate, which is the ratio of the message length to the encoded block length. This is a measure of how much "actual message data" is carried by each transmitted symbol. The second is the error rate, which is the ratio of the number of errors to the block length. This is a measure of how "noisy" the channel is, i.e. how much data it corrupts. Of course, we desire coding schemes that tolerate high error rates while simultaneously having large information rates. In practice, smaller alphabets are desirable too, as most digital communication devices are, at their lowest levels, capable of interpreting only binary digits (bits) [1,2].

Shannon demonstrated how information can be encoded to withstand such noise with probability arbitrarily close to 1 [3]. Two years later on error-correcting codes [4], Hamming proposed an adversarial channel that perturbs symbols in a worst-case fashion. This model of a channel is much more "pessimistic" than Shannon's, as it encompasses any arbitrarily-complex source of noise. As such, it gives much stronger guarantees on the robustness of the resulting coding scheme.

2. Coding Algorithm

The basic structure of a communication system is presented in the following diagram:

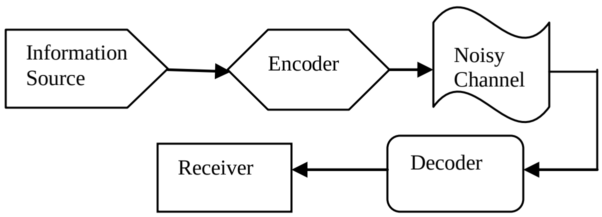

Figure 1. Communication system

The information source can be anything representable in symbols. In our research the information source is string of binary digits. The purpose of the communications system is to convey the message from one point to another with no degradation.
The noisy channel adds noise without our consent, therefore, corrupting some of the bits in our message. In the communication system, we are concerned mostly with the encoder in the transmitter part. The encoder will adds some extra bits to the original message, which is used in the receiver part in order to detect the error which has happened either in the message or in the check bits themselves. Hence, the main function of the decoder is to detect the errors, corrects errors, or the combination of both.
This section describes a new theory for encoding the message "the transmitted data" by using Reed-Muller basic matrix [5,6]. Reed-Muller matrix is operated on the data bits using the XOR and the AND operations to produce the necessary redundancy bits that are needed in the system to detect the error in the received message. The basic Reed-Muller matrix for the zero polarity is given in equation (1). From equation (1) we can generate the Reed-Muller matrix for *n* equals 3, where *n* is the number of variable in the binary form as shown in equation (2). Therefore, equation (2) will be the essential matrix to generate the redundancy bits.

$$RM = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad (1)$$

For *n* equals to three, the RM matrix is constructed as follows:

$$RM\,(3) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

This yields the following result:

$$RM\,(3) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad (2)$$

Where '*' is Kronecker operator [7,8].

The following notation is used throughout this paper. The terms are defined as follows.

k Number of "information" or "message" bits.
m Number of parity-check bits (check bits).
n Code length, $n = m + k$.
u Information bit vector, ***u***0, ***u***1, … ***u****k*–1.
p Parity check bit vector, ***p***0, ***p***1, …, ***p****m*–1.

The algorithm development is a direct construction of a code that permits correcting single-bit errors. We assume that the data to be transmitted consists of a certain number of information bits ***u***, and the algorithm adds to these a number of check bits ***p*** such that if a block is received that has at most one bit in error, then ***p*** identifies the bit that is in error (which may be one of the check bits). Specifically, in our code ***p*** is interpreted as an integer which is 0 if no error occurred, otherwise there is error either in the data bits or in the check bits. Let *k* be the number of information bits, and *m* the number of check bits used. Because the *m* check bits must check themselves as well as the information bits, the value of ***p***, interpreted as an integer, must range from 0 to which are distinct values. Because *m* bits can distinguish cases, we must have The check bits will be placed at the end of the information bits in a manner described below.

2.1 **Transmitter**

The transmitter consists mainly from the source and the encoder. The encoder plays most important part, its main function is to generate the proper code, which is needed to recover the corrupted bit in the receiver. The new algorithm will be is illustrated as shown in example one.

Example one:
For seven bits:
Let the number of the transmitted data (k) is seven (D7D6,……D0) = (00110111), where we assumed that the most significant bit(D7) is zero.
The encoder will generate the required code, which consists of the check bits (p). Therefore, the transmitter will send the information bits (k), and the check bits vector.
The generated check bits are constructed as follows:
Using the basic Reed Muller matrix for one bit using polarity zero is

$$RM = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

For two bits

$$RM = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

This is can be achieved by using the (AND) operator as follows:
1 AND $\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
This will yield the following:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} D0 \\ D1 \\ D2 \\ D3 \\ D4 \\ D5 \\ D6 \\ D7 \end{bmatrix}$$

C0 = 1•D0 ⊕ 0•D1 ⊕ 0•D2 ⊕ 0•D3 ⊕ 0•D4 ⊕ 0•D5 ⊕ 0•D6 ⊕ 0•D7
C0 = 1•1 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•0 = 1
Similarly for C1, C2, …., C7
C1 = 1•D0 ⊕ 1•D1 ⊕ 0•D2 ⊕ 0•D3 ⊕ 0•D4 ⊕ 0•D5 ⊕ 0•D6 ⊕ 0•D7
C1 = 1•1 ⊕ 1•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•0 = 0
C2 = 1•1 ⊕ 0•1 ⊕ 1•1 ⊕ 0•0 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•0 = 0
C3 = 1•1 ⊕ 1•1 ⊕ 1•1 ⊕ 1•0 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•0 = 1
C4 = 1•1 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 1•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•0 = 0
C5 = 1•1 ⊕ 1•1 ⊕ 0•1 ⊕ 0•0 ⊕ 1•1 ⊕ 1•1 ⊕ 0•0 ⊕ 0•0 = 0
C6 = 1•1 ⊕ 0•1 ⊕ 1•1 ⊕ 0•0 ⊕ 1•1 ⊕ 0•1 ⊕ 1•0 ⊕ 0•0 = 1
C7 = 1•1 ⊕ 1•1 ⊕ 1•1 ⊕ 1•0 ⊕ 1•1 ⊕ 1•1 ⊕ 1•0 ⊕ 1•0 = 1

Where '⊕' is XOR operator
Choosing the coding bits which cover all the data bits as follows:
C3 = D0 ⊕ D1 ⊕ D2 ⊕ D3
C5 = D0 ⊕ D1 ⊕ D4 ⊕ D5
C6 = D0 ⊕ D2 ⊕ D4 ⊕ D6
C7 = D0 ⊕ D1 ⊕ D2 ⊕ D3 ⊕ D4 ⊕ D5 ⊕ D6 ⊕ D7

Therefore, the parity check bit vector, $\boldsymbol{p}0$, $\boldsymbol{p}1$, …, $\boldsymbol{p}m{-}1$ is (c3c5c6c7) which is (1011).

Hence, the transmitted code is given as follows:

0D6D5D4D3D2D1D0C7C5C6C7
Which is: (00110111 1011) in the binary form.

Note we are transmitting seven bits only from (D6 to D0) where the D7 is set to zero, plus the four check bits.

2.2 Receiver

In the receiver side, we will recalculate the chick bits again using equation (2) in the same way as in the transmitter.
Then the new chick bits are compared with the received chick bits as the following:

1. Suppose now that data D0 sustains an error and is changed its value from 1 to 0.

Then the new chick bits according to this data are recalculated as before to yield the following:

C3 = 1•0 ⊕ 1•1 ⊕ 1•1 ⊕ 1•0 ⊕ 0•1 ⊕ 0•1 ⊕ 0•0 ⊕ 0•0 = 0
C5 = D0 ⊕ D1 ⊕ D4 ⊕ D5= 0⊕ 1 ⊕ 1 ⊕ 1= 1
C6 = D0 ⊕ D2 ⊕ D4 ⊕ D6 = 0⊕ 1 ⊕ 1 ⊕ 0= 0
C7 = D0 ⊕ D1 ⊕ D2 ⊕ D3 ⊕ D4 ⊕ D5 ⊕ D6 ⊕ D7
C7 = 0 ⊕ 1 ⊕ 1 ⊕ 0 ⊕ 1 ⊕ 1 ⊕ 0 ⊕ 0 = 0

To determine whether the error has occurred in one of data bits or in the check bits or none of that, the new check bits are compared with the old check bits. The syndrome word is formed:

$$\begin{array}{c} \text{C7C6C5C3} \\ 1101 \\ \underline{\oplus 0010} \\ 1111 \end{array}$$

The result is 1111, indicating that data bit 0, is in error.

ISSN: 1790-5117

ISBN: 978-960-474-155-7

2. Suppose now that data D1 sustains an error and is changed its value from 1 to 0.

Then the new chick bits according to this data are recalculated as before to yield the following:

$C3 = 1\bullet1 \oplus 1\bullet0 \oplus 1\bullet1 \oplus 1\bullet0 \oplus 0\bullet1 \oplus 0\bullet1 \oplus 0\bullet0 \oplus 0\bullet0 = 0$
$C5 = D0 \oplus D1 \oplus D4 \oplus D5 = 1 \otimes 0 \otimes 1 \otimes 1 = 1$
$C6 = D0 \oplus D2 \oplus D4 \oplus D6 = 1 \otimes 1 \otimes 1 \otimes 0 = 1$
$C7 = D0 \oplus D1 \oplus D2 \oplus D3 \oplus D4 \oplus D5 \oplus D6 \oplus D7$
$C7 = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 0$

The new check bits are compared with the old check bits. The syndrome word is formed:

$$\begin{array}{r} C7C6C5C3 \\ 1101 \\ \underline{\oplus 0110} \\ 1011 \end{array}$$

The syndrome word is $(p_0p_1p_2p_3) = 1011$,
Hence, the weight of each bit from left to right in the decimal value is (0,1,2,4). Therefore, the value of the corrupted bit reading just the zero bits in the syndrome is one.

3. Suppose now that data D2 sustains an error and is changed its value from 1 to 0.

Then the new chick bits according to this data are recalculated as before to yield the following:

$C3 = D0 \oplus D1 \oplus D2 \oplus D3 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$
$C5 = D0 \oplus D1 \oplus D4 \oplus D5 = 1 \oplus 1 \oplus 1 \oplus 1 = 0$
$C6 = D0 \oplus D2 \oplus D4 \oplus D6 = 1 \oplus 0 \oplus 1 \oplus 0 = 0$
$C7 = D0 \oplus D1 \oplus D2 \oplus D3 \oplus D4 \oplus D5 \oplus D6 \oplus D7$
$C7 = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 0$

The new check bits are compared with the old check bits. The syndrome word is formed:

$$\begin{array}{r} C7C6C5C3 \\ 1101 \\ \underline{\oplus 0000} \\ 1101 \end{array}$$

The syndrome word is $(p_0p_1p_2p_3) = 1101$,
Hence, the weight of each bit from left to right in the decimal value is (0,1,2,4). Therefore, the value of the corrupted bit reading just the zero bits in the syndrome is two.

Similarly for D3 the syndrome is:

$$\begin{array}{r} C7C6C5C3 \\ 1101 \\ \underline{\oplus 0100} \\ 1001 \end{array}$$

Therefore, the value of the corrupted bit is (one plus two) which gives three.

The following table (1) summarize the process if one of the transmitted data has been corrupted, gives the corresponding syndrome, and the data bit number or location.

Table 1: Syndromes for corrupted bits

Data	Check bits received	Recalculated Check bits	syndrome	Corrupted bit
D0	1101	0010	1111	0
D1	1101	0110	1011	1
D2	1101	0000	1101	2
D3	1101	0100	1001	1+2=3
D4	1101	0011	1110	4
D5	1101	0111	1010	1+4=5
D6	1101	0001	1100	2+4=6

To find the error occurred in the check bits themselves.
Let the error in C1, the syndrome is calculated as follows:

$$\begin{array}{r} C7C6C5C3 \\ 1101 \\ \underline{\oplus 1100} \\ 0001 \end{array}$$

The result reflects an error occurred in the first check point, and only one of the syndrome bits is set to one, and the rest of the bits are zeros.
Suppose the error is the second check bit, then the syndrome is:

$$\begin{array}{r} C7C6C5C3 \\ 1101 \\ \underline{\oplus 1111} \\ 0010 \end{array}$$

Therefore, the second bit is set to one, which reflects that the error is in the second check bit.

The following table (2) summarizes the process if one of the check bits has been corrupted, gives the corresponding syndrome, and the check bit number or location.

Table 2: Syndromes for corrupted check bits

Error in check bit	Check bits received	Recalculated Check bits	syndrome	Corrupted bit
C0	1100	1101	0001	C1
C1	1111	1101	0010	C2
C2	1001	1101	0100	C3
C3	0101	1101	1000	C4

We summarize the final characteristics for the four bit syndromes which were generated by XORing the received check bits with the recalculated check bits as follows:

1. If the syndrome contains one and only one bit set to one, then there is an error in one of the check bits.
2. If the syndrome contains all zeroes, then there is no error, and the transmitted data is accurate.
3. If the syndrome contains more than one bit set to one, then there is an error has occurred in one of the data bits. The data bit is identified according to the results, which have been given in table (1).

3. Conclusion

In this paper we introduced a simple algorithm, which can be used to detect, and correct the errors in the transmitted message based on Reed-Muller matrix. The algorithm was tested on some examples, and has given correct results. The algorithm can be extended to *n* bits messages, and can be implemented on integrated circuits based on XOR gates.

References

[1] B. P. Lathi., 1983, Modern Digital and Analog Communication Systems. CBS College Publishing.
[2] Simon Haykin, 1994, Communication Systems.John Wiley & Sons, INC.
[3] Claude E. Shannon. A mathematical theory of communication. Bell System Technical Journal, 27:379–423, July 1948.
[4] Richard W. Hamming. Error detecting and error correcting codes. Bell System Technical Journal, 29:147–160, 1950.
[5] Reed, I. S., 'Class of multiple error correcting codes and their decoding scheme,' Institute of Radio Engineers Transaction on Information Theory, PGIT-4: pp. 38- 49, 1954.
[6] Muller, D. E., 'Application of Boolean algebra to switching circuit design and to error detection,' Institute of Radio Engineers Transaction on Electronic Computers, EC-3: pp. 6-12, September 1954.
[7] Faraj, K., Almaini, A.E.A.,Minimization of Dual Reed-Muller Forms using Dual Property. WSEAS TRANSACTIONS on CIRCUITS and SYSTEMS, Issue 1, Vol. 6, pp.9-15, January 2007.
[8] Faraj, K., Almaini, A.E.A., Optimal Expression for Fixed Polarity Dual Reed-Muller Forms. WSEAS TRANSACTIONS on CIRCUITS and SYSTEMS, Issue 3, Vol. 6, pp.9-15, March 2007.