© 2017 IJSRSET | Volume 3 | Issue 1 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Bit Error Detection and Correction with Hamming Code Algorithm

Achmad Fauzi[1], Nurhayati[2], Robbi Rahim[3*]

[1,2]Lecturer, STMIK Kaputama, Jl. Veteran No. 4A-9A, Binjai, Indonesia
[3]Departement of Computer Engineering, Medan Institute of Technology, Jl. Gedung Arca No.52, Medan, Indonesia

ABSTRACT

Data communication performed at any time does not always go well, sometimes also happens that his name error when transmitting data. The detection and correction of errors in data transmission requires special algorithms in this study using the algorithm Hamming Code, the use of this algorithm due to ease in the detection and correction of bit damaged and is expected by this research easier for other researchers to develop algorithms Hamming Code for a variety of purposes in data transmission.

Keywords*:* Error Correction, Hamming Code, Transmission Data, Bit Error, Bit Detection

I. INTRODUCTION

Process digital data transmission in communication can run safe premises but also there is an error [1] [2] [3]. Error result in changes to the content of data transferred. There is a wide - range logic to detect and correct the error. One way to detect errors that simply is by using Hamming Code single error correction [3] [4], this technique is a most convenient to find errors in bit data transmission [5] [6] [7] [8].

Hamming code is an error-detection method that can detect some errors, but it is only capable of single error correction [6]. This error detection method is suitable for use in situations where there is few randomized mistake [4]. Method Hamming Code inserts (n + 1) check bits into 2n data bits. This method uses XOR (Exclusive - OR) in the error detection process [1] [3] [7]. Input and output data from the method of Hamming Code in the form of binary numbers, This study tried to detecting and correcting the error with a length of 8 bits of data.

II. METHODS AND MATERIAL

Communication

Digital communication is something that is important in a communications infrastructure that is being built [9], a simple basic communication model is as follows :

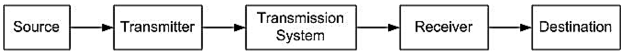

Figure 1. Basic Communication

The following is a description of Figure 1 [9]:

a. The source is a tool to generate the data so that it can be transmitted.
b. Transmitter serves to convert the data into the form of an analog signal to be transmitted. Often the data generated from the source system is not transmitted directly in its original form. A transmitter is quite moving, and marking information in the same way as to produce a signal - a signal electro - magnetic that can transmit over multiple sequential transmission systems.
c. Source and Transmitter are part of Source System.

d. Transmission system, in the form of a single transmission line or a complex network that connects the source to the destination.
e. Receiver serves to convert an analog signal that receive in the form of digital data. The receiver receives signals from the transmission system and incorporates it into a specific shape that can be captured by the goal. For example, a modem will receive an analog signal coming from the network or the transmission path and converting it into a digital bit streams.
f. Destination serves to capture data generated by the receiver.
g. Receiver and Destination are part of the Destination System.

The data transmission process in detail [9] can be seen in the figure below:

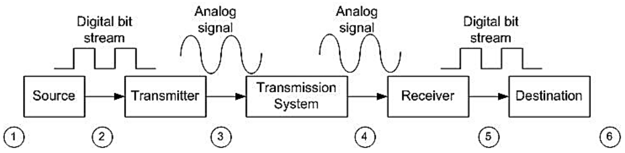

Figure 2. Details of the data transmission process

The following is a description of Figure 1 [9]:
a. Information in the input.
b. Data which is input.
c. The signal will be transmitted in the form of an analog signal.
d. Signal received in the shape of an analog signal.
e. Data output.
f. Information output.

Error Correction

At the time of data transmission system, there is the possibility of corrupted data (data error) [1] [4] [6] [7] [9]. The receiver will correct data error through a process of error detection and error correction. Error detection process performed by the transmitter by adding a few extra bits into data to be transmitted. Error detection and correction process often used in CD Players, High-speed modems, and cellular phones.

Hamming Code

Method Hamming Code is one method of error detection and error correction the most simple. This method uses logic operation XOR (Exclusive - OR) in the process of error detection, and the process of error correction, while input and output of data from the method of Hamming Code in the form of binary numbers [1] [3] [6] [7] [8].

Hamming Code method invented by Richard W. Hamming in the 1940s [3]. Hamming Code Method is one of the error detection methods that can detect some errors, but it is only capable of correcting one error [1] [6] [7] [8]. This error detection method is suitable for use in situations where errors are occuring randomly [3] [6].

Hamming Code Method inserts multiple pieces of check bits into data. The number of check bits inserted depending on the length of the data [2] [6] [7]. Calculation formula for calculating the number of check bits that must insert into the data such as follows:

For 2n bits of evidence, the number of check bits that are inserted there as much as c = (n + 1) bits.

So that got a table increment data bits and checked bits [5] [8] as follows:

TABLE I
Table Increase data bits and check bits

Data Bit	Check Bit
2	2
4	3
8	4
16	5
32	6
64	7
128	8
256	9

Check bits are insert into the data placed in a position that is calculated based on the formula for calculating the position following the check bits [3] [4] [8]:

$$C_i = 2^{i-1}$$

So that got the check bit position table as follows:

TABLE II
Position Check Bit

Check Bit	Position
C_1	1
C_2	2
C_3	4
C_4	8
C_5	16
C_6	32
C_7	64
C_8	128
C_9	256

Check bits are used to make the process of error detection and error correction. Algorithms process error detection and error correction of the method of Hamming Code [1] [3] [6] [7] is as follows:

a. Calculate the length of the data input method of Hamming Code that represents the sum of the duration of the input data and check bits long. The term of the output data from the method of Hamming Code equal to the length of the data input method of Hamming Code.
b. Mark all the bit position which is the position of the check bits. The position other than the post of the check bit is the post of the data bits.
c. Determine the calculation formula of each - each check bit. For n = 1 to the number of check bits, do the following:
   - ✓ Record all positions where bits n of the member position is 1, except for the bit position itself. Member position is a binary form of the bit position.
   - ✓ The formula of check bits n equals XOR operation of position - a position he noted.
d. Calculate the value of the check bits for data input and data output.
e. If the value of the check bit input is not equal to the value of the check bit output means there is an error.
f. Perform XOR operation to test the input bits and check bit output.
g. Convert the XOR operation results in the form of a decimal number.
h. If the value of the results of the XOR operation is greater than the length of the data input or the value of the product of the XOR operation similar to the position of the check bits, then there is more than one error.

If not, then the number of errors is only one position, and results of operations are the data contained errors.

III. RESULTS AND DISCUSSION

A To understand the process of completion method of Hamming Code can be seen in the following example.

Suppose the input data is transmitted 10111010. Data received by the receiver in the form of 10111110, the error detection and error correction of the method of Hamming Code is as follows:

1. Calculate the length of data input and output of the method of Hamming Code. For 2n bits of evidence, the number of check bits that are inserted there as much as c = (n + 1) bits. The length of the input and output data = 8 bits = 23, so the number of check bits: 3 + 1 = 4 and the data duration of the Take method Hamming Code = 12 bit
2. Mark all the bit position which is the position of the check bits. The position other than the post of the check bit is the position of the data bits. The formula for computing the position of the check bits:

$$\boxed{C_i = 2^{i-1}}$$

$$C_1 = 2^{1-1} = 2^0 = 1$$
$$C_2 = 2^{2-1} = 2^1 = 2$$
$$C_3 = 2^{3-1} = 2^2 = 4$$
$$C_4 = 2^{4-1} = 2^3 = 8$$

These steps do all the bits in the process, and the results are as shown in the following table:

TABLE III
Position check bits and the data bits after process

Bit Position	Member Position	Check Bit	Data Bit
12	1100		M_8
11	1011		M_7
10	1010		M_6
9	1001		M_5
8	1000	C_4	
7	0111		M_4
6	0110		M_3
5	0101		M_2
4	0100	C_3	
3	0011		M_1

Bit Position	Member Position	Check Bit	Data Bit
2	0010	C_2	
1	0001	C_1	

3. Determine the calculation formula of check bits C1, C2, C3, and C4. C1 = see the right-most position of the member position where the bit value is 1, unless the post of the check bits. Take all data bits in that position

TABLE IV
Calculation check bits C1

Bit Position	Member Position	Check Bit	Data Bit
12	1100		M_8
11	101**1**		M_7
10	1010		M_6
9	100**1**		M_5
8	1000	C_4	
7	011**1**		M_4
6	0110		M_3
5	010**1**		M_2
4	0100	C_3	
3	001**1**		M_1
2	0010	C_2	
1	0001	C_1	

$C_1 = M_1 \oplus M_2 \oplus M_4 \oplus M_5 \oplus M_7$.

C2 = see the position of second bit from the right of the member position where the bit value is 1, unless the post of the check bits. Take all data bits in that position.

TABLE V
Calculation check bits C2

Bit Position	Member Position	Check Bit	Data Bit
12	1100		M_8
11	10**1**1		M_7
10	10**1**0		M_6
9	1001		M_5
8	1000	C_4	
7	01**1**1		M_4
6	01**1**0		M_3
5	0101		M_2
4	0100	C_3	
3	00**1**1		M_1
2	0010	C_2	
1	0001	C_1	

$C_2 = M_1 \oplus M_3 \oplus M_4 \oplus M_6 \oplus M_7$.

C3 = see the position of the third bit from the right of the member position where the bit value is 1 unless the post of the check bits. Take all data bits in that position

TABLE VI
Calculation check bits C3

Bit Position	Member Position	Check Bit	Data Bit
12	1**1**00		M_8
11	1011		M_7
10	1010		M_6
9	1001		M_5
8	1000	C_4	
7	0**1**11		M_4
6	0**1**10		M_3
5	0**1**01		M_2
4	0100	C_3	
3	0011		M_1
2	0010	C_2	
1	0001	C_1	

$C_3 = M_2 \oplus M_3 \oplus M_4 \oplus M_8$.

C4 = see the fourth bit position from the right of the member position where the bit value is 1, unless the post of the check bits. take all data bits in that position.

TABLE VII
Calculation check bits C4

Bit Position	Member Position	Check Bit	Data Bit
12	**1**100		M_8
11	**1**011		M_7
10	**1**010		M_6
9	**1**001		M_5
8	1000	C_4	
7	0111		M_4
6	0110		M_3
5	0101		M_2

Bit Position	Member Position	Check Bit	Data Bit
4	0100	C_3	
3	0011		M_1
2	0010	C_2	
1	0001	C_1	

$C_4 = M_5 \oplus M_6 \oplus M_7 \oplus M_8$.

4. Calculate the value of the check bits for data input and data output.

Data *input* :

1	0	1	1	1	0	1	0
M1	M2	M3	M4	M5	M6	M7	M8

$C_1 = M_1 \oplus M_2 \oplus M_4 \oplus M_5 \oplus M_7$
$= 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0$

$C_2 = M_1 \oplus M_3 \oplus M_4 \oplus M_6 \oplus M_7$
$= 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 0$

$C_3 = M_2 \oplus M_3 \oplus M_4 \oplus M_8$
$= 0 \oplus 1 \oplus 1 \oplus 0 = 0$

$C_4 = M_5 \oplus M_6 \oplus M_7 \oplus M_8$
$= 1 \oplus 0 \oplus 1 \oplus 0 = 0$

Values check bits for data input,

0	0	0	0
C4	C3	C2	C1

Once processed, the data input into,

0	1	0	1	**0**	1	1	0	**0**	1	**0**	**0**
M8	M7	M6	M5	**C4**	M4	M3	M2	**C3**	M1	**C2**	**C1**

Then followed the process of calculation to output data as follows,

1	0	1	1	1	1	1	0
M1	M2	M3	M4	M5	M6	M7	M8

$C_1 = M_1 \oplus M_2 \oplus M_4 \oplus M_5 \oplus M_7$
$= 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0$

$C_2 = M_1 \oplus M_3 \oplus M_4 \oplus M_6 \oplus M_7$
$= 1 \oplus 1 \oplus 1 \oplus 1 \oplus 1 = 1$

$C_3 = M_2 \oplus M_3 \oplus M_4 \oplus M_8$
$= 0 \oplus 1 \oplus 1 \oplus 0 = 0$

$C_4 = M_5 \oplus M_6 \oplus M_7 \oplus M_8$
$= 1 \oplus 1 \oplus 1 \oplus 0 = 1$

Values check bits for data output,

1	0	1	0
C4	C3	C2	C1

Once processed, the data output becomes,

0	1	0	1	**1**	1	1	0	**0**	1	**1**	**0**
M8	M7	M6	M5	**C4**	M4	M3	M2	**C3**	M1	**C2**	**C1**

5. Check the value of input bits is not equal to the value of the check bit output means there is an error.
6. Perform XOR operation to test the input bits and check bit output.

	C4	C3	C2	C1
Input	0	0	0	0
Output	1	0	1	0
	---------------------------------- ⊕			
	1	0	1	0

7. Convert XOR operation results into the shape of a decimal number.
$(1010)_2 = 1 \text{ x } 2^3 + 0 \text{ x } 2^2 + 1 \text{ x } 2^1 + 0 \text{ x } 2^0 = 8 + 0 + 2 + 0 = 10$.
8. The value of XOR operation is smaller than the length of the input data, and the value of the XOR operation is not the position of the check bits, then there is only one mistake (error), and the operating result is bad bit position.

TABLE VIII
bad bit position

Bit Position	Member Position	Check Bit	Data Bit
12	1100		M_8
11	1011		M_7
10	**1010**		$\mathbf{M_6}$
9	1001		M_5
8	1000	C_4	
7	0111		M_4
6	0110		M_3
5	0101		M_2
4	0100	C_3	
3	0011		M_1
2	0010	C_2	
1	0001	C_1	

Position 10 is the position of the data M6, mean scores on the 6th position in the output data contains errors, data output shown as bellow:

1	0	1	1	1	**1**	1	0
M1	M2	M3	M4	M5	**M6**	M7	M8

Data value at position 6 (M6) on the data output is supposed to be,

M6 = ~ (M6) = ~ (1) = 0

Therefore, the data output should be:

1	0	1	1	1	**0**	1	0
M1	M2	M3	M4	M5	**M6**	M7	M8

IV. CONCLUSION

Research by the author using Hamming Code is only capable of correcting single error correction, then data input and output on the method of Hamming Code must be a result of the powers of 2n with *n* must be greater than one with a length of data input and output of Hamming Code must be at least equal to 4 bits, and also Hamming Code unable to check the position of the data error (bad bit) more than one.

V. REFERENCES

[1] Deepika, A. Kumar, and Gurusiddayya, "A Study on Error Coding Techniques," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 4, no. 4, pp. 825-828, 2016.

[2] V. Singh and N. Sharma, "Improving Performance Parameters of Error Detection and Correction in HDLC Protocol by using Hamming Method," International Journal of Computer Applications, vol. 126, no. 1, pp. 1-7, 2015.

[3] R. Hamming, "Error Detecting and Error Correcting Codes," The Bell System Technical Journal, vol. 29, no. 2, 1950.

[4] R. Ullah, J. Khan, S. Latif and I. Ullah, "Indication of Efficient Technique for Detection of Check Bits in Hamming Code," International Journal of Computer Science (IJCSI), vol. 8, no. 5, pp. 241-246, 2011.

[5] V. Badole and A. Udawat, "Implementation of Multidirectional Parity Check Code Using Hamming Code for Error Detection and Correction," International Journal of Research in Advent Technology, vol. 2, no. 5, pp. 1-6, 2014.

[6] W. Fitriani and A. P. U. Siahaan, "Single-Bit Parity Detection and Correction using Hamming Code 7-Bit Model," International Journal of Computer Applications, vol. 154, no. 2, pp. 12-16, 2016.

[7] V. Singh and N. Sharma, "A Review on Various Error Detection and Correction Methods Used in Communication," American International Journal of Research in Science, Technology, Engineering & Mathematics, vol. 9, no. 3, pp. 252-257, 2014.

[8] B. Antony and Divya S, "Modified Hamming Codes with Double Adjacent Error Correction along with Enhanced Adjacent Error Detection," International Journal of Innovative Research in Computer and Communication Engineering, vol. 3, no. 8, pp. 7706-7713, 2015.

[9] B. Maulana and R. Rahim, "GO-BACK-N ARQ APPROACH FOR IDENTIFICATION AND REPAIRING FRAME IN TRANSMISSION DATA," International Journal of Research In Science & Engineering, vol. 2, no. 6, pp. 208-212, 2016.