International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

IJIEMR Transactions, online available on 24th Mar 2020. Link :http://www.ijiemr.org/downloads.php?vol=Volume-09&issue=ISSUE-03

Title: HAMMING CODE FOR SINGLE BIT ERROR DETECTION & ERROR CORRECTION

Volume 09, Issue 03, Pages: 101-109.

Paper Authors

TELAGARAPU.PRABHAKAR, M.NARASIMHARAO

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

HAMMING CODE FOR SINGLE BIT ERROR DETECTION & ERROR CORRECTION

TELAGARAPU.PRABHAKAR[1], M.NARASIMHARAO[2]
[1,2]Dept of ECE, GMR Institute of Technology , Rajam -532127 , Andhra Pradesh, India.

Abstract—Error free transmission is a major issue in present Electronic devices. Fixed data transmission from sender to receiver occurs with greater issue, i.e. errors in information of data transmission from transmitter to receiver. Here Error correction codes and detect codes are used to protect the information In memories and logic registers in electronic circuits. Hamming codes are one of the best error correcting code using either, even parity or odd parity check method. In this paper Hamming code algorithm is used for Error detection and Error rectification. The Hamming codes will be used to detect and correct the one bit error. (A eleven bit message and four bit parity is used to form a 15-bit code word. Then this 15-bit code word is received at the receiver. If it is having one bit error it will be detected and corrected to get the original message code word. If the received information is having a single error , this hamming code detects the presence of a single error in the input code word). This process implemented in cadence tool.

Index Terms—Hamming codes, Error detection, Error correction, Cadence *tool*

1 INTRODUCTION

In Digital communication systems and the electronic devices errors are introduced during the transmission of data from sender to receiver. The input data from sender to receiver due to interference or noise. The error of fault is occurring circumstance when the output information does not correspond with the input information that means “0” bit invert to “1” or “1” bit invert to “0” Error change one to zero value (0 to 1). To improve the reliability and stability it is indispensable to find and even off the mistake. Hence, we have to employ some kind. Error detection and Error rectification codes. This type of codes, one or two than one other number is appended to bits at the time of sending the information [1] to be amended. To improve the reliability, it is indispensable to find and even off the mistake. Hence, we have to employ some kind. Error perception and error rectification codes. The data bits along with the parity bit forms a code word. Codes Which allow only error detection are called Error detection codes and are used to detect an error occurred during TX of the message. A simple example of error detection code is parity check method and both codes are error detection and error rectification also called error recognition
and correction codes. In error rectification codes, parity check has a childlike mode to detect error location bits.At one time the corrupted bit is located, At one time the corrupted bit is present, its value is a range (from 0 to 1 or 1 to 0) [2]. There are dissimilar cases of error controlling codes such as parity checking, checksum error detection, CRC. Comparing with other error identification codes, hamming code is one of the best methods for error detection as well as error rectification. at the location of bits first bit is (1,3,5,7,9,…..), second bit is (2,3,6,7,10,11,….) and each bits calculated as per same first and second bit parallel to third and fourth bit values.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

2 PROCEDURE FOR HAMMING CODE

2.1 Hamming Codes

Hamming code not just provides the detection of a bit error, but also identifies detect a double bit error. The number of numbers assigned as 'm' then the number of parity bits 'P' is determined by the following relationship

$2^P \geq P + m + 1$

Here m= number of message bits.

P= parity bits.

Eleven information bits, m=11, then the number of parity bits 'P' is obtained by trial and error using the above equation. Thus 4 parity bits are required for 11 information bits and these 4 parity bits are placed on the powers of 2. Like 2^0, 2^1, 2^2, 2^3, 2^4……[1].

Hamming code is one best efficiency for error detection as well as for error rectification and this code is likewise comfortable to implement. Fast-speed information transmission and high reliability becomes the two most important criteria in modern communication systems, speed of input data by the recipient, that is channel coding. Apparently, the two positions are at odds. Hamming code error perception and rectification with the odd &even parity method using Xilinx and Cadence Tool is a technique to determine the fault fix and that error . In this system information transmission of data from sender to receiver. Here to send 15 bit data from input to output, studied in details in the transmission section, data out means 15 bit encrypted output data is sent from sender to receiver. In transmission section and at receiver section, input means 15 bit data is picked up by the receiver. Different type of hamming codes like (7,4),(15,11),(31,26),(63,57),(127,120),(255,247)
In communication systems. These codes are communication error, i.e. from input dataTx (transmitter) to Rx (receiver) major issues formed. Sender to receiver data is corrupted, i.e. binary bit data Tx to Rx. data sending bit by bit, Ex Tx "01101111110" to Rx"01101111100" here one bit position occurs one error so input of the system is corrupted. This type of problem solves the different type of solutions, but the hamming code is one of the best and efficient error code. And also Hamming code used to burst errors(4bit burst error) double bit error detection and rectification.

2.2 PARITY BIT IDENTIFICATION PROCESS

The parity bit recognition method using either, even parity or odd parity checker method, the parity bit position calculating power of 2 i.e. 2^0, 2^1 2^2 ….2^n . The position of parity bits are P1,P2,P3,P4…..PN. Bit identity 1 bit checker is P1 and similarly the parity bits P2,P3,P4….PN, the 2bit checker is P2, 4bit checker is P3 and 8 bit checker is P4.

1.Parity bit1, check 1bit, skip 1 bit (P1, 0 , 0 ,1 ,0, 1, 0 ,1 = 1) The number of '1' is odd value is '1'
2.Parity bit2, check 2bit, skip 2bit (P2, 0, 1,1, 0, 1, 1, 0 = 0)
The number of '1'is even value is '0'
3.Parity bit3, check 4bit, skip 4bit (P3, 0, 1, 1, 1, 1, 0, 1 = 1)
The number of '1' is odd value is '1'
4.Parity bit4, check 8bit, skip 8bit (P4, 0, 0, 1, 1, 0, 1, 1 = 0)
The number of '1'is even value is '0'[1].

Single Error Detection and Error rectification and Single bit error detection (SER/SED) codes are experimentally used in cache memories and in main memory systems in information processing systems.
The general algorithm for the hamming code is as follows:
1The parity bits added the given input data depend on message bits i.e. p+m
2. Next, the parity bits location placed in given data
3. These positions are numbered in powers of 2 for the parity bits and the remaining bits are data bits.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

4. Parity bits are calculated by logic gate operation, i.e. EXOR Operation. The parity bits are calculated as follows:

P1 = EX-OR of bit positions (1, 3, 5, 7, 9, 11, 13,15).

P2 = EX-OR of bit positions (2, 3, 6, 7, 10, 11,14,15).

P3 = EX-OR of bit positions (4, 5, 6, 7, 12,13,14,15).

P4 = EX-OR of bit position (8, 9, 10, 11, 12, 13,14,15).

Calculate the parity bits with EXOR Operation

Here EXOR use bit wise operator, i.e. "^"

P1= d1^ d2 ^ d4 ^ d 5 ^ d 7 ^ d 9 ^ d11.

P2= d1 ^ d3 ^ d4 ^ d 6 ^ d 7 ^ d 10 ^ d11.

P3= d2 ^ d3 ^ d4 ^ d8 ^ d 9 ^ d 10 ^ d11.

P4= d5 ^ d6 ^ d7 ^ d8 ^ d 9 ^ d 10^ d11.

2.3 Figures

The parity bits location table

Parity Bits(p)	Input data(n)	Message bit(m)	Total data p=(n,m)
2	3	1	(3,1)
3	7	4	(7,4)
4	15	11	(15,11)
5	31	26	(31,26)
6	63	57	(63,57)
7	127	120	(127,120)
8	255	247	(255,247)

Table 1 Hamming code for parity bits location table.

The above fig shows how many parity bits possible given input data or message bits

2.4 PARITY BIT IDENTIFICATION IN 15 BIT DATA

An Input of 11-message bits, parity bits process

Input 11 message bits **"01101111110"** the parity bits become by formula,

$$2^P \geq P + m + 1$$

m=11 p=4

n=m+p , n=7

n=15 therefore input data is 15- bit data

Here m= message bits. P= parity bits.

CHECK THE PARITY BITS:

1.Parity bit1, check 1bit, skip 1 bit (P1, 0 , 0 ,1 ,1, 1, 1 ,1 = 1) The number of '1' is odd value is '1'

2.Parity bit2, check 2bit, skip 2bit (P2, 0, 1,1, 1, 1, 1, 0 = 1)

The number of '1'is odd value is '1'

3.Parity bit3, check 4bit, skip 4bit (P3, 0, 1, 1, 1, 1, 0, 1 = 1)

The number of '1' is odd value is '1'

4.Parity bit4, check 8bit, skip 8bit (P4, 0, 1, 1, 1, 0, 1, 1 = 1)

The number of '1'is odd value is '1'.

The general algorithm for the hamming code is as follows:

Parity bits are calculated by logic gate operation, i.e. EXOR Operation. The parity bits are calculated as follows:

P1 = EX-OR of bit positions (1, 3, 5, 7, 9, 11, 13,15).

P2 = EX-OR of bit positions (2, 3, 6, 7, 10, 11,14,15).

P3 = EX-OR of bit positions (4, 5, 6, 7, 12,13,14,15).

P4 = EX-OR of bit position (8, 9, 10, 11, 12, 13,14,15).

Calculate the parity bits with EX-OR Operation

Here EXOR use bit wise operator, i.e. **"^"**

P1= d1^ d2 ^ d4 ^ d 5 ^ d 7 ^ d 9 ^ d11.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

P2= d1 ^ d3 ^ d4 ^ d 6 ^ d 7 ^ d 10 ^ d11.
P3= d2 ^ d3 ^ d4 ^ d8 ^ d 9 ^ d 10 ^ d11.
P4= d5 ^ d6 ^ d7 ^ d8 ^ d 9 ^ d 10^ d11.

So now calculate input of 15-bit data

P1 = EX-OR of bit positions (1,3,5,7,9,11,13,15) (1bit skip)

P2 = EX-OR of bit positions (2,3,6,7,10,11,14,15) (2bit skip)

P3 = EX-OR of bit positions (4,5,6,7,12,13,14,15) (4bit skip)

P4 = EX-ORof bit position (8,9,10,11,12,13,14,15)(8bit skip)

Parity bits calculation given input code word location of binary bits, then values of parity bits P1,P2,P3,P4. Here using with EX-OR Operation then find parity bits.

Parity bit1, check 1bit, skip 1 bit (P1, 0 , 0 ,1 ,1, 1, 1 ,1 = 1) The number of '1' is odd value is '1'
Parity bit2, check 2bit, skip 2bit (P2, 0, 1,1, 1, 1, 1, 0 = 1)
The number of '1'is odd value is '1'
Parity bit3, check 4bit, skip 4bit (P3, 0, 1, 1, 1, 1, 0, 1 = 1)
The number of '1' is odd value is '1'
Parity bit4, check 8bit, skip 8bit (P4, 0, 1, 1, 1, 0, 1, 1 = 1)
The number of '1'is odd value is '1'.

So finally get the 4 parity bits

Table 2. parity bits table

P4	P3	P2	P1
1	1	1	1

An So finally get the 15-bit data is "**011011111111011**"

So input I.e. 11 bits are message bits and four bits parity bits get 15 bits
n=p+m n= 11+4 n=15
The bit position of parity bits 1,2,4,8 position place the P1=1,P2=1,P3=1,P4=1 and input of message bits d1=0,d2=1,d3=1,d4=1, d5=1, d6=1, d7=1, d8=0, d9=1, d10=1, d11=0. Place the input data. Both parity bits and message bits adding get the 15 bit data
So Finally get the 15 bit code word is"011011111111011" Input 11bit data is detected from sender data. Detect 4bit parity bits and these 4 parity bit positions of P4,P3,P2,P1 value like "1111" binary bit value the 4 bit parity bits binary bit value remove get the input of 15- bit data.

3 DETECTION OF ONE ERROR BIT IN 15-BIT DATA

3.1 Input of 15 bit data

As Given 15 bit data n=M+P ,n=11+4 n=15.
Given 15 bit code word is **"011011111111011"**
The error value of 15bit data **"011011111101011**" value of input code word one position value changes from 0 to 1. What position error occurs during input 15bit data, Now calculate the 4 bit redundancy value get the location of error value [2].

.

3.2 DETECTION OF ONE ERROR VALUE HAMMING CODE WITH15-BIT INFORMATION SIGNAL AND 11-MESSAGE BITS

Input 11 message bits data is detected from 15bit data. Detect 4bit redundancy bits and these 4 redundancies bit positions of error value like "0101" binary bit value the 4 bit redundancy binary

Table 3 . 15-Bit code word with one error value

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Code word	0	1	1	0	1	1	1	1	1	1	0	1	0	1	1
Data word	M11	M10	M9	M8	M7	M6	M5	P4	M4	M3	M2	P3	M1	P2	P1

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

bit value remove get the output of 11bit data[3]. The message 'm' bits are getting the input using even or odd parity check method in hamming code of n bits. These operation problem solve the redundant bits 'R' calculation. Hamming data are formed by adding the evidence and redundant bits, i.e. n= m+r are. Redundancies bit is calculated by the formula [3].

$$2^r = m + r + 1$$

Here r= redundant bits. m=data length. n=code word length.

Here are using the EX-OR operation to find the redundancy bit error position,

R1= P1 ^ M1 ^ M2 ^ M4 ^ M5 ^ M7 ^ M9 ^ M11
R2= P2 ^ M1 ^ M3 ^ M4 ^ M6 ^ M7 ^ M10 ^ M11.

R3= P3 ^ M2 ^ M3 ^ M4 ^ M8 ^ M9 ^ M10 ^ M11.

R4= P4 ^ M5 ^ M6 ^ M7 ^ M8 ^ M9 ^ M10 ^ M11.

Here calculate bits wise location

R1= XOR (1, 3, 5, 7, 9, 11, 13,15)

R1= XOR (1,0,0,1,1,1,1,0)

All 1's are odd numbers value is '1' and even numbers are '0' so value of R1 is '0' (all 1's are odd numbers value is '1')

The value of R1=1

R2= XOR (1,0,1,1,1,1,1,0)

Here All 1's are odd numbers value is '1' and even numbers are '0' (all 1's are odd numbers value is '1') and even numbers are '0'. These values be depend on total numbers like 1 or 0.

All 1's are odd numbers value is '1' and even numbers are '0' so the value of R2 is '0' (all 1's are even number's value is '0')

The value of R2=0

R3= XOR (0,0,0,1,1,1,1,1)

All 1's are odd numbers value is '1' and even numbers are '0' so value of R1 is '0' (all 1's are odd numbers value is '1')

The value of R3=1

R4= XOR (1,0,1,1,1,1,1,0)

All 1's are odd numbers value is '1' and even numbers are '0' so the value of R2 is '0' (all 1's are even number's value is '0')

The value of R4=0

So finally get the redundancy value bits

The value of"0101" is 5th in binary value

R4	R3	R2	R1
0	1	0	1

Table4 . Redundancy bits binary value

The value "0101" is a binary bit value 5th bit i.e. the 5th position bit is error. **"01101111110101 1"** 5th bit error means change the value '0' to '1' and we get input 15 bit data **"011011111111011".**

4 CORRECTION ONE ERROR BIT IN 15-BIT DATA

Given 15 bit code word is **"011011111111011"** One bit Error value in 15bit code word **"011011111101011"**

The value of input 15 bit data with parity bits and occur in
one bit error value, here observe the one bit error value of 15 bit data. Given 11 bit code word is "01101111110" and four parity bits "1111" and finally 15 bit code word is "011011111111011" the error value in 15 bit code word is "011011111101011" the value of input code word one position value change from 1 to 0.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Here which position error occur in input 15bit code word from sender to receive.The error value of 15 bit data where the error occur position. Then input 11bit message bits are ”01101111100” and parity bits are “1111” the error of bit 5th position error detection in message bits, that position is d2. The d2 position of error correction using decoder of (4 t0 16). That output of decoder values given to correct code word of input message bits, then finally get original message bits ”01101111110”.

The input 11 message bits “01101111110” and parity bits “1111” get 15 bit code word. And message bits d11=0, d10=1, d9=1, d8=0, d7=1, d6=1, d5=1, d4=1, d3=1, d2=1, d1=0, here parity data bits are p4=1, p3=1, p2=1, p1=1.

Error of input 11 message bits “01101111100” and input of parity bits “1111” and message bits d11=0, d10=1, d9=1, d8=0, d7=1, d6=1, d5=1, d4=1, d3=1, d2=0, d1=0, here d2 value change from 1 to 0, so error occur in receiver section.

Here solve the error detection using redundancy formula then get position of error detection that position of 5th location is done. Error correction is possible using Decoder

Input message bits 11 i.e. “01101111110” and parity bits “1111” and receiver of 11 message bits “01101111100” error correction value is d2. So here using 4 to 16 decoder in correction process.

Process of correction:

- Message bits i.e. data bits in receiver, that bits given to input of four XOR gates. And calculate the parity bits of receiver message bits, not for input of parity bits.
- The receiver of parity bits n1,n2,n3,n4 as for scematic diagram, these parity bits given to another XOR gates. one input is n1 parity and another input is orginal input parity bit p1, simillaly another n2,n3,n4 and p2,p3,p4 input of XOR gates.
- The XOR gates outputs s1,s2,s3,s4. These four outputs given to input of 4 to 16 deccoder as per scematic diagram. Decoder is input of value is 0101, and output is 5th value is high remain are 0. These decoder outputs given to another XOR gates, that XOR gates one input from decoder and one input data bits then get correct code of 11 message bits.
- The receiver 11 data bits one error is detection and correction is done given the 15-code word.

The error d2 bit position value 0 to 1 change that value of 0 this error calculates the redundancy bit formula here in the cadence input of 15 bit data With XORING i.e. using the XOR operation then find the output R1, R2, R3, R4 observe in CADENCE TOOL.

5 FIGURES

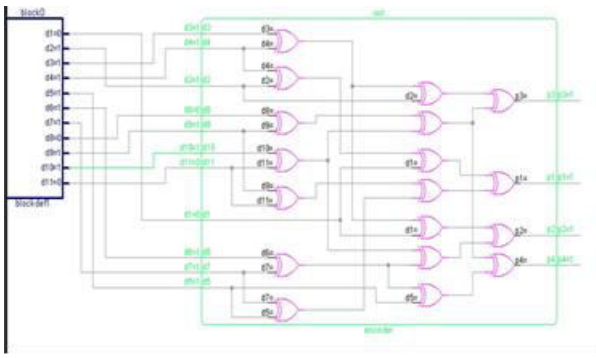

Fig1. Schematic diagram of parity bits identification

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

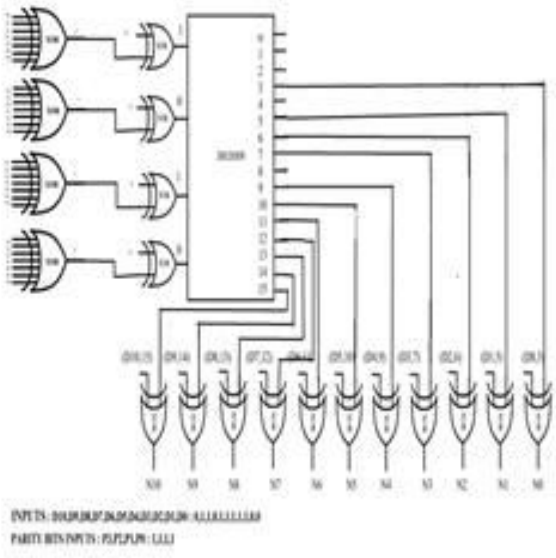

Fig 2. Schematic diagram for correction error value in 15-bit code word

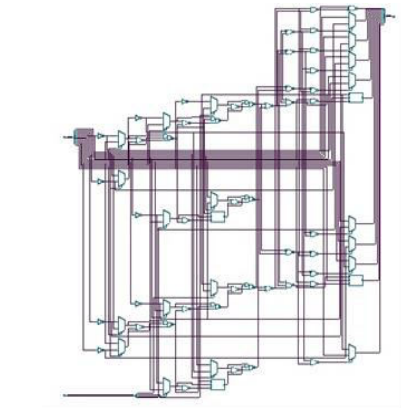

Fig 3. Schematic diagram for detection error value in 15-bit code word

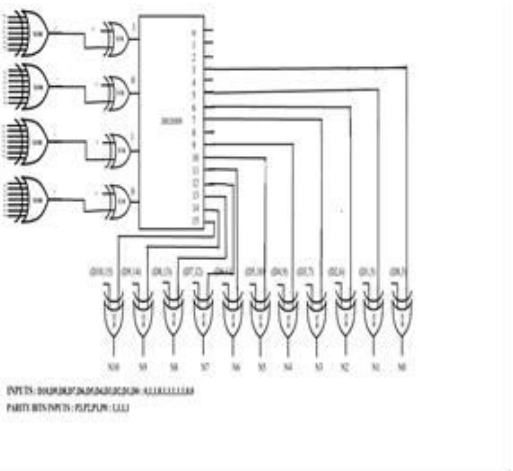

Fig 4. Architecture of correction decoder

6 ERROR DETECTION AND CORRECTION DESIGN SIMULATION AND VALIDATION

The simulation results of input 15 bit data area

Fig5. Simulation results for 11-input data signal & 4parity input signal (15 bit data)

The values of input 11 bit data with parity bits and occur in
one bit error value here observe the one bit error value.
Simulation results of 11 bit data The value of input 11 bit data with parity bits and occur in one bit error value, here observe the one bit error value in Simulation results of 11 bit data. Given 11 bit code word is **"11001010011"** Error value in 11 bit code word is"**11000010011"**the value of input code word one position value change from 0 to 1. What position

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Error occur in input 11bit data. Here one bit is error from sender to receive.The error value in 11 bit data where the error occur in given input datafunctionality verification and simulation

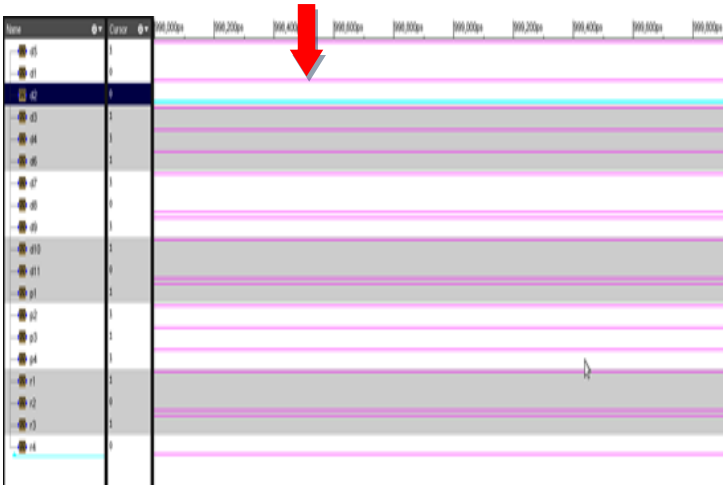

Fig 6. One bit error input of (15-bitdata)

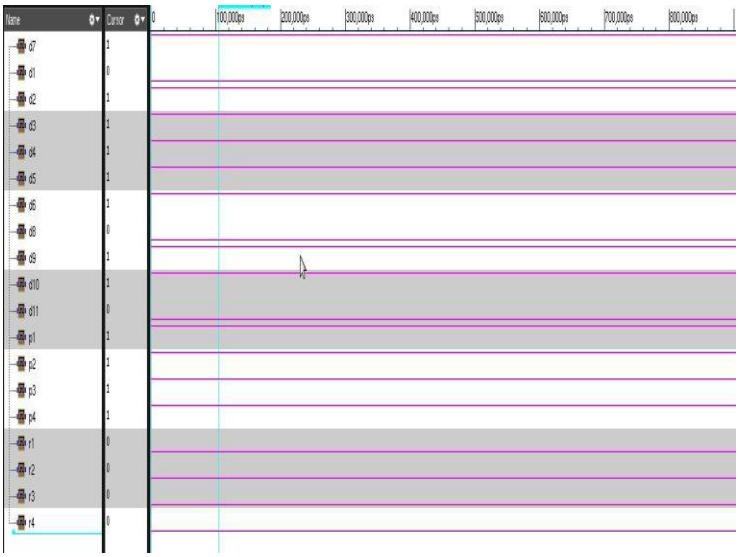

Fig 7. 15-data bit no error value simulation

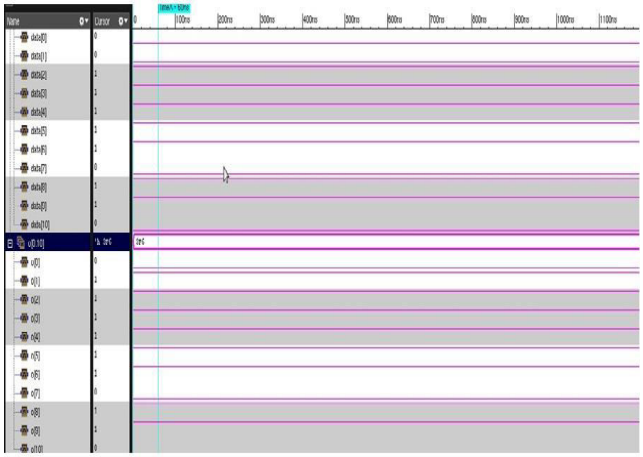

Fig 8. 11 message bits correction one error value simulation

7 CONCLUSION AND FUTURE SCOPE

Digital codes are being used in digital communication for sending data. In this Paper Hamming code is identify as the best method for detecting and correction of Single bit error. Even parity is used initially 11 bit data packet with 7 message bits and 4 parity bits are considered. The single bit error appeared in this 11 bit data packet is identified and corrected. A 15 bit data packet with 11 message bits and 4 parity bits are also considered for the work. The single bit error appeared in this 15 bit data packet is also identified and rectified. In future similar work can be extended using other codes such as cyclic codes. Future the work can be extended to detect and correct more error bits.

REFERENCES

1 Dr. Anil Kumar Singh,"Error detection and correction by hamming code", 978-1-5090-0467-6/16/$31.00 ©2016 IEEE.

2.PranjaliPothare,PrajaktaAmbatkar,PayalPatre,KarishmaPadole, ShilpaLende, AnkitaBelekar "Hamming Code For Single Bit Error Detection & Error Correction With Even Parity Using Vhdl" (IJARCET), Volume 4 Issue 1, January2015, pp. 262-265, 2278 – 1323.

3.Vivek Singh, RahulKumar, ManishKumarUpadhyay, "VHDL Code for Single Bit Error Detection and Correction with Even Parity Check Method Using Xilinx" IJARSE, Vol1, Issue1, 2013.

4. Tongsheng Zhang and Qun Ding, **"**Design of (15, 11) Hamming Code Encoding and Decoding System Based on FPGA",978-0-7695-4519-6/11 $26.00 © 2011 IEEE

5. Divya Mokara, "Design and Implementation of Hamming Code using VHDL"", International Journal of Latest Engineering Research and Applications (IJLERA) ISSN: 2455-7137 Volume – 02, Issue – 11, November – 2017, PP – 33-40.

6. Brajesh Kumar Gupta1, RajeshwarLal Dua2 and B. Surya Narayana Raju3 "30 Bit Hamming Code For Error Detection And Correction With Odd Parity Method By Using Vhdl" IJCSC Vol. 3, No. 1, June 2012.

7. Luis-J.Saiz-Adalid,et.al.,"Modified Hamming Codes to Enhance Short Burst Error Detection in Semiconductor Memories", 978-1-4799-3804-9/14 $31.00 © 2014 IEEE.

8. Lankesh and K.C.Narasimhamurthy, PhD "Hardware Implementation of Single Bit Error Correction and Double Bit Error Detection through Selective Bit Placement for Memory" International Journal of Computer Applications (0975 – 8887).

9. DeepikaSS,et.al.,"A Study on Error Coding Techniques", www.ijraset.com Volume 4 Issue VI, June 2016 IC Value: 13.98 ISSN: 2321-9653.

10. Achmad Fauzi, et.al.,"Bit Error Detection and Correction with Hamming Code Algorithm" IJSRSET | Volume 3 | Issue 1 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099.

11. Jiaqiang Li and Pedro Reviriego,"Extending 3-bit Burst Error-Correction Codes With Quadruple Adjacent Error Correction" IEEE transactions on very large scale integration (vlsi) systems, vol. 26, no. 2, february 2018