International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com

BCH Code Error Detection and Correction for Variable Block Length and Message Length using VHDL

Shreya D. Potey[1], Piyush M. Dhande[2]
[1]*PG (Student, M. Tech.) Electronics & Communication, Department of Electronics & Telecommunication, D.M.I.E.T.R. Sawangi (Meghe), Wardha, India*
[2]*Asst. Professor, Department of Electronics & Telecommunication, D.M.I.E.T.R. Sawangi (Meghe), Wardha, India*

Abstract: ***The BCH code is the essential class of multiple-error-correcting linear cyclic code. In actuality, Bose-Chaudhuri-Hocqunghem code is an abstraction of the cyclic hamming code for multiple-error correction. The finding of errors made by noise during transmission from the sender to the recipient is called as Error Detection. The finding of errors and renewal of the original data is called as Error Correction. The error can be detected and corrected for the fixed value of n and k. So that, the system efficiency and correcting capability are lower. In this paper, we are implementing a system in which we can detect and correct the flexible block length and message length of BCH code on Xilinx using VHDL. The simulation process is done by using Xilinx ISE 13.2.***

Keywords: BCH (Bose-Chaudhari-Hocqenghem) code, BCH Encoder, BCH Decoder, VHDL, LFSR, Galois Field, Syndrome, BER (Bit Error Rate)

I. INTRODUCTION

Nowadays, the fastest growing area in the field of communication is Wireless Communication. Whenever the communication takes place, the data need to be transmitted through the sender to the recipient.

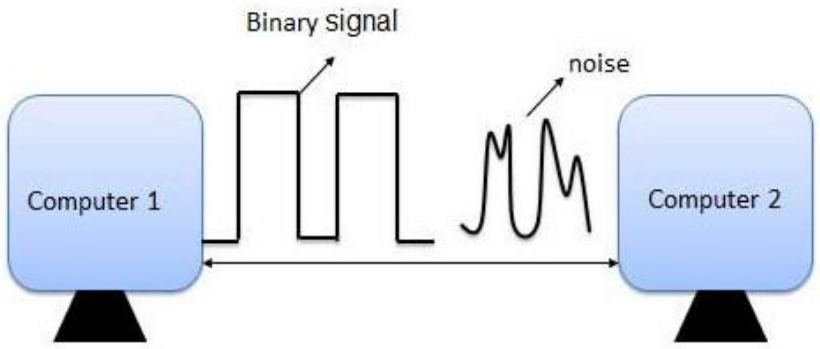

Fig. 1: Basic block diagram of error detection and correction

The error is such a situation in which output information is not similar to the input information. When digital signals are transmitted from the sender to the recipient, some errors can introduce in the signals because of noise interference. That means a 0 bit may change to 1 or vice versa [16]. The significant data will be lost because of the data error.

In a communication system, detecting the error during the transmission process of data through the sender to the recipient is called error detection. For the detection of error, we can use some redundancy codes. These redundancy codes are then added to the data during transmission, such codes are termed as “Error detecting codes” [17].

Error Correction Code is a procedure of put in parity bits into real data in order to recover the message by the receiver. We need error correcting code for good conveying message over a noisy way that has an improper BER as well as SNR is lower. Reed-Solomon codes, Golay codes, Cyclic Hamming code, BCH codes, Goppa codes are used in error correction code [5]. The computational complexity of Reed-Solomon encoder and decoder is high, but the complexity of BCH code is low.

II. BASIC OF BCH CODE

The BCH code is the essential part of the multiple - error-correcting linear cyclic code. In 1959, A. Hocquenghem and in 1960, R. C. Bose and D. K. Ray-Chaudhari invented Binary codes which are called as BCH code. An accurate control above the multiple symbol errors correctable in the code is a central feature of Bose-Chaudhuri-Hocqunghem code. Especially, multiple bit errors can be corrected by designing binary Bose-Chaudhuri-Hocqunghem codes [18]. BCH code depends on Galois field arithmetic which includes describing the binary actions over finite collections of an element [5].

Bose-Chaudhuri-Hocqunghem code is the subclass of cyclic codes. In order to give good error correction capability, we have to specify the roots of the generator polynomials. BCH code can identify and overcome the error up to 't' arbitrary errors per code words because every BCH code has 't' error correcting code. BCH code has the following factors:

Block length: $n=2^m-1$

Number of message bits: $k \geq n-mt$

Minimum distance: $d_{min} \geq 2t+1$

Here 'n' is the block length and 'k' is the message length. 'm' is the positive integer, ranges to $m \geq 3$. The above factor can correct the 't' errors or fewer error. The generator polynomial of this code is identified in respect of its roots from the Galois field GF(2m) [11].

III. BCH ENCODER

The BCH encoder has three parts which perform the operation in it, i.e.,

A. Galois Field

A field having just a finite number of components is known as Galois Field. Another name of the Galois field is the Finite field. Galois field is a special case of a finite field. This field can perform many operations like addition, subtraction, multiplication, division and satisfy certain basic rules.

A field in which the number of the element is the type p^n. In the p^n, 'p' refer as prime number and 'n' refer as a positive integer, is termed as the Galois field, such a field is represented by $GF(p^n)$.

B. Generator Polynomial

BCH code is a subset of cyclic codes, the organized method of encoding is related to the method of binary coding. For the BCH code, the generating polynomial is:

$g(X)=LCM(m_1(x)+m_2(x)+\ldots\ldots\ldots+m_{2t}(x))$

The number of parity bits is equivalent to the degree of generator polynomial. Since the degree 2t is equal to the generator polynomial, there must be a precisely 2t successive powers of α that is the roots of the polynomial. Message polynomial m(x) can be transferred into the leftmost n-k stages of a code word register and then tacking on a parity polynomial p(x) [5].

C. LFSR (Linear-Feedback Shift Register)

The LFSR is used in the innovative digital communication scheme and it is similar to all the BCH code design. Linear-Feedback Shift Register structures are broadly used in communication system and digital signal processing.

The linear function of individual bits is commonly used exclusive-or (XOR). LFSR is a shift register whose input bit is determined through the XOR of some bits. The shift register is a device whose recognizing function is to move the contents into adjacent position.

Following is the design operation of Encoder Linear Feedback Shift Register:

1) For the clock cycle 1 to k, the original message bits are transferred without changing its form, and the linear feedback shift register calculates the parity.
2) For cycle k+1 to n, the generated parity bits in the linear feedback shift register are transmitted and the feedback in the LFSR is switch off.

IV. BCH DECODER

The BCH decoder has four parts which perform the operation in it, i.e.,

1) Solving syndrome
2) Finding the error locating polynomial
3) Finding the error locating number
4) Error correction

V. LITERATURE REVIEW

In [6] and [8] paper authors proposed method designed the BCH encoder on FPGA for (15, k) using an AWGN channel with the numerous error correcting code. Considered (15, 11, 1), (15, 7, 2), and (15, 5, 3) BCH encoder for error correction according to the highest degree of a polynomial. Here encoder of (15, 5, 3) was more beneficial than the remaining, because of the necessity of speed. When the noise corrupted the original data the BCH encoder was corrected 3 errors at the receiver side [6], [8].

Sahana C*, V Anandi [2] presented a binary encoding of (255, 215, 5) for the simulation of encoder also Syndrome computation. The message bits of 215 were encrypted and any 5-bit error was detected from the 255-bit code word. LFSR was used for the encoder process and Finite field polynomial multiplication was used for the calculation of syndrome [2].

Samir Jasim Mohammed [10] designed and simulated BCH encoder codes of (31, 26, 1), (31, 21, 2), (31, 16, 3), (31, 11, 5) and (31, 6, 7) utilizing Xilinx-ISE 10.1 and enforced in FPGA. Here, a 31-bit size code word was used in this implementation. The (31, k) BCH encoder was properly examined and compared. The BCH encoder code (31, 6) occupied a greater area than the other codes. The results showed the system works well [10].

In [11] at the time of corrupting the actual data, the binary BCH code (15, 11, 3) and (63, 39, 4) were corrected 3-bit error and four-error on the output side respectively. Here, iterative algorithm was used to identify the location of the error. Due to this, the speed and utilization of the device were improved [11].

In [12] author has designed a (15, 7) BCH encoder and decoder. By utilizing the algorithms, two errors have been acknowledged and adjusted in the BCH code which worked at the single cycle. The result showed the area and delay were reduced [12].

The author in [1] proposed a new method for compressing the LUT and reduces the requirement that was compared to prior design. The throughput of the decoder was greater and the area has been reduced by 19% for (1023, 993) BCH code over GF(2^{10}) [1].

VI. PROBLEM FORMULATION AND PROPOSED WORK

In the existing work, BCH and Reed-Solomon encoder and decoder has been developed. In the current work fixed block length and message length code are decoded as well as the fixed error correcting capability (t) are calculated.

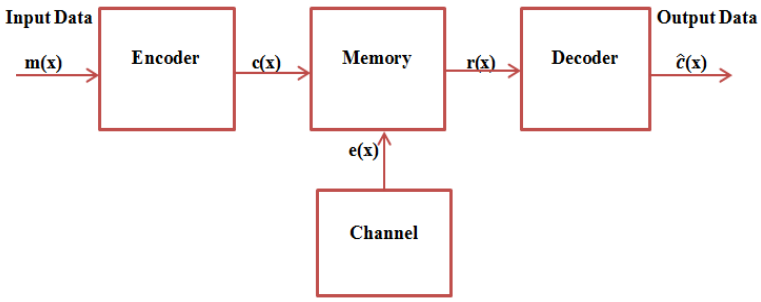

Fig. 2: Block Diagram of the BCH Code

The proposed work is to implement the variable length BCH encoder and decoder using VHDL. At the sender side, the data streams are encrypted by adding some additional bits with message bits called as parity bits. The message bits composed with parity bits named as a 'Codeword'.

In our work, we have designed a BCH system to detect and correct the error for (n, k) code. On the input side, when the input message bit, i.e., 'k' is applied to the encoder, it performs the operations like Galois field, Generator polynomial (H), LFSR. This message bits of 'k' then converted into the block length or code word i.e., 'n'.

For example, if we are taken (16, 8) BCH code, then 8-bit of message bit, i.e., m(x) is converted due to the generator polynomial, and LFSR, we get the 16-bit code word or block length i.e., c(x). Here, LFSR performs the shifting operation whose input bit is a linear function of the previous bit. For polynomial division, we can use Linear Feedback Shift Register. Based on this H matrix, we XOR the bits of the input register and produced the output.

Block length (n): 16

Message length (k): 8

Consider double error correction capability, i.e., t=2

Then, $d_{min}=2t+1$

=2*2+1=5

Minimal polynomial= 2t-1

=2*2-1=3

The generator polynomial is,

$g(X)=LCM(m_1(x)+m_2(x)+\ldots\ldots\ldots+m_{2t}(x))$

due to the generator polynomial, and exclusive-OR (X-OR) operation with input data and H matrix, we get the output of encoder part i.e., c(x). The output c(x) is stored into the memory. Here, memory is storage system and the AWGN channel is added noise or error, i.e., e(x) into the memory. So that, the output of the encoder gets corrupted and we got the r(x) which is received code word.

The received code word is,

$r(x)= c(x)+ e(x)$

Where, the received codeword is

$r(x) =r_0+r_1x+\ldots..+r_{n-1}x^{n-1}$

Transmitted codeword is,

$c(x) =c_0+c_1+\ldots..+c_{n-1}x^{n-1}$

The error pattern is,

$e(x) =e_0+e_1+\ldots..+e_{n-1}x^{n-1}$

The (n, k) BCH decoder, is based on the majority logic decoder. In this decoder we can detect and correct more than one-bit error. In the decoder part, syndrome plays the important role in it. The Berlekamp-Massey algorithm and chien search algorithm is used.

Syndrome Si can be computed by:

$S_i=r(\alpha^i)= r_0+r_1\,\alpha^i + r_2\,\alpha^{2i}+ r_3\alpha^{3i} \ldots..+r_{n-1}\,\alpha^{(n-1)i}$

where $1 \le i \le 2t - 1$.

If the syndromes are zero, then there is no error found and if the syndromes are non-zero, then there are errors in the received code word. Then the next part is to find out the error locating polynomial.

The error locating polynomial is,

$\lambda(\alpha^j) = \lambda_0 + \lambda_1\alpha^j + \lambda_2\alpha^{2j}$

to find out the coefficient of error locating polynomial, Berlekamp-Massey algorithm (BMA) is used [9].

After finding the location polynomial we can find out the root of the polynomial with the help of chien search algorithm. The reciprocal of the root is an error locator number.

The last step is the error correction process. The received code word is stored in the memory until it is corrected successfully. In this, the error bits are changed, that is 0 bit may change to 1 and 1 bit may change to 0. In this method the received code word is corrected.

VII. SIMULATION AND RESULTS

The design waveform of BCH code is depicted as below:

The below depicts waveform 3 shows (16, 8) BCH code. The input r(x)= “1010101011001100” and “1010101010001100”, and at the output side, we get the corrected error free data as well as Fig. 4 shows RTL view of (16, 8) BCH code.

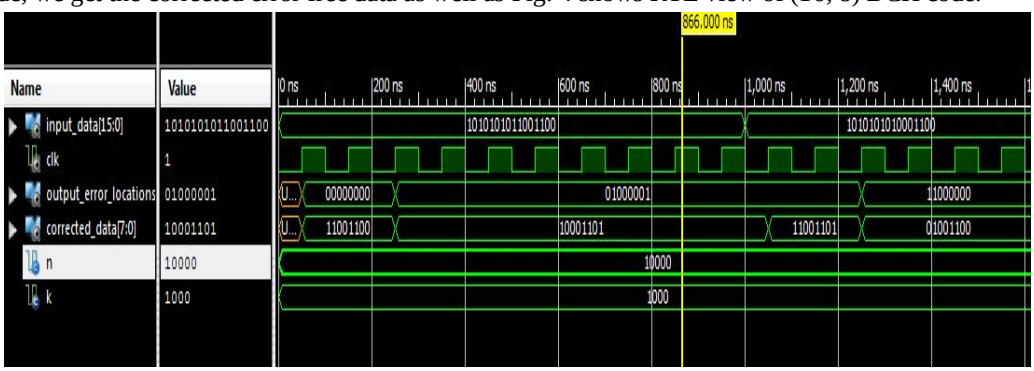

Fig. 3: Simulation Result of (16, 8) BCH code

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com

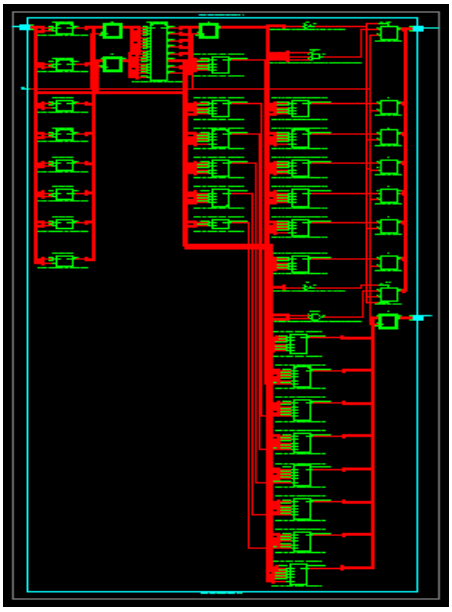

Fig. 4: Technology schematic of (16, 8) BCH code

The below illustrates waveform 5 shows the (18, 9) BCH code. We are given an input as "110101110110100111" and "011010010101100010" and we get the correct output waveform. RTL view of (18, 9) BCH code is shown in fig. 6.

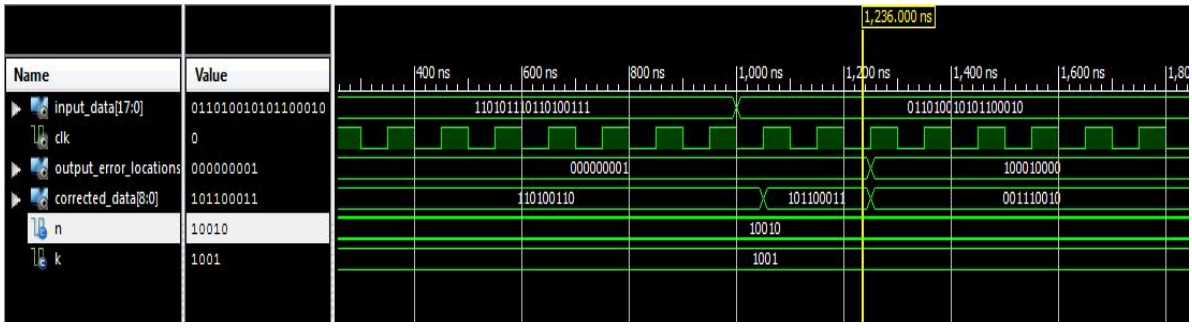

Fig. 5: Simulation Result of (18, 9) BCH code

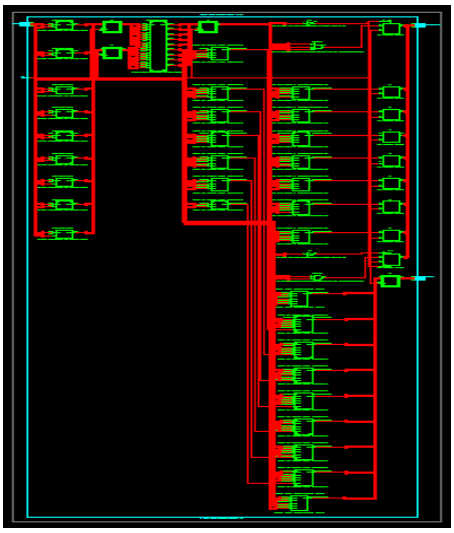

Fig. 6: Technology Schematic of (18, 9) BCH code

The below waveform 7 shows the corrected output of (20, 10) BCH code. The input of r(x) is "10010110111011011001" and "01011001110101010011". In the output waveform we get an error free output. Fig. 8 shows Technology Schematic of (20, 10) BCH code.

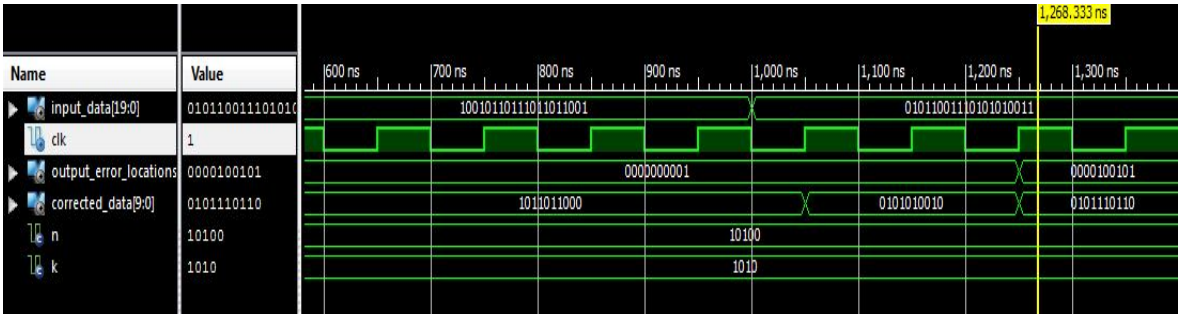

Fig.7: Simulation Result of (20, 10) BCH code

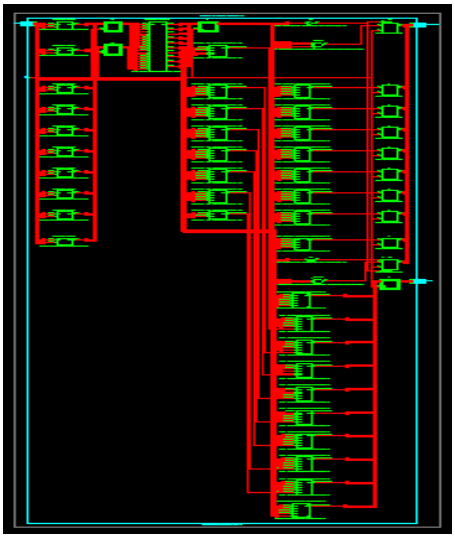

Fig. 8: Technology Schematic of (20, 10) BCH code

VIII. COMPARISON OF THE RESULTS

The below table 1 shows the comparison of different block lengths and the message lengths of the BCH code.

n	k	Input	Error Location	Corrected Data
16	8	1010101011001100	01000001	10001101
		1010101010001100	11000000	01001100
18	9	110101110110100111	000000001	110100110
		011010010101100010	100010000	001110010
20	10	10010110111011011001	0000000001	1011011000
		01011001110101010011	0000100101	0101110110

Table 1: Tabular comparison of the results

IX. CONCLUSION

In a communication system, error detection is the operation of finding the faults that are present in the data broadcast from the sender to the receiver. To improve the reliability of the binary transmission system, error Correction Codes are vital. The proposed work is to implement the variable length BCH encoder and decoder using VHDL. When the input data is applied to Galois field, the generator matrix H has performed the XOR operation with bits of the input register and formed the output. If we are given n-bits of input to the encoder then we get doubles of input-bits at the output due to the generator polynomial. This output data is stored in the memory and passed it to the decoder side. The decoder removes the errors from the received code word with the help of the syndrome. After that, the error location polynomial, and numbers are found. Due to the root we get the error location number and we just flipped that bit such as 1 bit may change to 0 and 0 bit may change to 1. In this way we get the error free output. We can correct an error for various (n, k) values of BCH code. It improves the throughput of the system and the system gets efficient.

REFERENCES

[1] Xinmiao Zhang, Michael O'Sullivan, "Ultra-Compressed Three-Error-Correcting BCH Decoder", 978-1-5386-4881-0/18/$31.00 ©2018 IEEE.
[2] Sahana C*, V Anandi, "Error Detection USING Binary BCH (255, 215, 5) codes", International Journal of Engineering Science and Research -Technology, [Sahana, 4(6): June, 2015], ISSN: 2277-9655, (I2OR), Publication Impact Factor: 3.785.
[3] Priyanka Shrivastava, Uday Pratap Singh, "Error Detection and Correction Using Reed Solomon Codes", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8, August 2013, ISSN: 2277 128X.
[4] Shital M. Mahajan, Piyush M. Dhande, "Design of Reed Solomon Encoder and Decode", International Journal on Recent and Innovation Trends in Computing and Communication, Volume: 4, Issue: 5, ISSN: 2321-8169 306 – 310 IJRITCC, May 2016.
[5] Saeideh Nabipour, Javad Javidan, Gholamreza Zare Fatin, "Error Detection Mechanism based on BCH Decoder and Root Finding of Polynomial over Finite Fields", Journal of mathematics and computer science 12 (2014), 271-281.
[6] Amit Kumar Panda, Shahbazsarik, AbhishekAwasthi, "FPGA Implementation of Encoder for (15, k) Binary BCH Code Using VHDL and Performance Comparison for Multiple Error Correction Control", 2012 International Conference on Communication Systems and Network Technologies, 978-0-7695-4692-6/12 $26.00 ©2012 IEEE DOI 10.1109/CSNT.2012.170.
[7] Faisal Rasheed Lone, ArjunPuri, Sudesh Kumar, "Performance Comparison of Reed Solomon Code and BCH Code over Rayleigh Fading Channel", International Journal of Computer Applications (0975 – 8887) , Volume 71– No.20, June 2013.
[8] Yathiraj H U, Mahasiddayya R Hiremath, "Implementation of BCH Code (n, k) Encoder and Decoder for Multiple Error Correction Control", International Journal of Computer Science and Mobile Applications, Vol.2 Issue. 5, May- 2014, pg. 45-54 ISSN: 2321-8363.
[9] Priya Mathew, Lismi Augustine, Sabarinath G., Tomson Devis, "Hardware Implementation OF (63, 51) BCH Encoder and Decoder for WBAN using LFSR and BMA", Department of ECE, St. Joseph's College of Engineering & Technology, Palai, Kerala.
[10] Samir Jasim Mohammed, "Implementation of Encoder for (31, k) Binary BCH Code based on FPGA for Multiple Error Correction Control", International Journal of Computer Applications (0975 – 8887), Volume 76 – No.11, August 2013.
[11] R.Elumalai, A.Ramachandran, J.V.Alamelu, Vibha B Raj, "Encoder And Decoder For (15,11,3) and (63,39,4) Binary BCH Code With Multiple Error Correction", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 3, Issue 3, March 2014, ISSN (Print) : 2320 – 3765, ISSN (Online): 2278 – 8875.
[12] Gnana Prakash, M.Muthamizhan, "FPGA Implementation of Bose Chaudhuri Hocquenghem Code (BCH) Encoder and Decoder for Multiple Error Correction Control", International Journal of Innovative Research in Science, Engineering and Technology, Vol. 5, Issue 3, March 2016, ISSN(Online) : 2319-8753, ISSN (Print) : 2347-6710.
[13] DejanAzinović, Klaus Tittelbach-Helmrich and ZoranStamenković, "Performance Investigation on BCH Codec Implementations", International Symposium on Signal Processing and Information Technology (ISSPIT), 978-1-5090-5844-0/16/$31.00 ©2016 IEEE.
[14] Vishal Chandale, Pallavi Gavali, Payal Giri, Mrs. Anjali Shrivastav, "FPGA Based Error Detection And Correction System Using Reed-Solomon Code", International Research Journal of Engineering and Technology (IRJET), Volume: 03 Issue: 05 | May-2016, e-ISSN: 2395 -0056, p-ISSN: 2395-0072.
[15] Sweta Thakur, Tabassum Nasrat and Soumyasree Bera, "Comparison of Channel Encoding Technique", International Journal Series in Engineering Science (IJSES), Vol. 2, No. 2, 2016, 11-17, ISSN: 2455-3328.
[16] https://www.tutorialspoint.com/computer_logical_organization/error_codes.htm
[17] https://www.electronicshub.org/error-correction-and-detection-codes/
[18] https://en.wikipedia.org/wiki/BCH_code