e-ISSN: 2582-5208
International Research Journal of Modernisation in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

CUSTOM IP DESIGN FOR BCH ENCODER

Dr. D. Asha Devi*1, Divya Katta*2, M. Manasa*3, V. Nikitha*4

*1Professor, Dept. Of Electronics And Communications Engineering, Sreenidhi Institute Of Science And Technology, Hyderabad, Telangana, India.

*2,3,4Student, Dept. Of Electronics And Communications Engineering, Sreenidhi Institute Of Science And Technology, Hyderabad, Telangana, India.

ABSTRACT

In a communication channel, noise and interferences are the two main sources of errors occur during the transmission of the message. Debug Codes are required for reliable communication using a device with an unacceptable bit rate and low audio signal. In this project we designed and used code (15, k) BCH using VHDL to transmit reliable data with a multimedia control system using FPGA. Reasonable digital implementation of binary code coding for multiple BCH (15, k) code editing lengths n = 15 over GF (2^4) with incredible old polynomial x^4 + x + 1 is arranged into shift register circuits. By using circuit codes, a reminder b (x) can be obtained from a stage shift register (15-k) with a line and a response link corresponding to the generated polynomial coefficients. Designing in FPGA results in a higher calculation using compatibility (performance is faster), and is easier to change. Here we have created (15,4) BCH code encoder on Vivado Artix7 FPGA using VHDL and simulation and integration is done using Xilinx ISE 10.1. And comparative performance based on integration and simulation in the FPGA is presented.

Keywords: Vivado, FPGA, Artix7, BCH Code.

I. INTRODUCTION

With the rapid growth of the internet and mobile technology, information exchange is a common practice. Information will mostly be text, audio, video etc. The connections that occur in our daily lives are in a state of symptoms. These signals, like sound signals, are generally, naturally analogous. When the connection needs to be established at a distance, then analog signals are sent by telephone, using various effective transmission techniques. The Need to Make Digital communication is that, Conventional communication methods use analog signals to connect long distances, which deal with many losses such as distortion, disruption, and other losses including security breaches. To overcome these problems, signals are digitized using a variety of techniques. Digital signals allow communication to be clear and precise without loss. In current Digital Communication systems, it is very likely that data or message was corrupted during the transmission and reception via a noisy channel. Environmental interference and moderate physical impairment are the main causes of data or message corruption in the communication area, leading to the injection of random fragments into the original message and damage the original message. In order to have reliable audio communication with unacceptable bit error rate (BER) and low signal-to-audio signal (SNR), we need to have Error Codes (ECC). Error correction is based on mathematical formulas, which are used for error correction codes (ECC). Error correction occurs by adding measurement bits to the actual message bits during data transmission. Due to the addition of measurement bits to message bits it makes the original message size longer. Now these pieces of long messages are called "Codeword". This codeword is received by the recipient on the go, and may be recorded to receive fragments of the actual message. Debug codes are used in many digital applications, space and satellite communications and mobile networks. There are many types of error correction codes used in the current digital communication system based on the type of error, expectations, the expected level of communication error, and the weather relay may or may not occur. Other bug fix codes, most commonly used these days, are BCH, Turbo, Reed Solomon, and LDPC. These codes differ from each other in their complexity and functionality.

Transferring information using a portable device or wireless medium, which data can be corrupted, leads to error. In a noisy channel where data is transferred, on the receiver side it is very difficult to retrieve the actual data. It is common for a digital system to be fully trusted, as a single error can shut down an entire system, or cause unacceptable data corruption. According to Channel Coding, "The error rate of data transmitted through a limited audio channel can be reduced to a minimum if the information level is below channel capacity" . Errors that may be introduced to digital data as a result of a channel communication transmission can be corrected based on the received data [1,2]. A few of them are Hamming code , Low Density Parity Check (LDPC) code , Bose-Chaudhuri Hocquenghem (BCH) code , Reed Solomon code , and code -Turbo. These codes differ from each other in their complexity and functionality. They can fix only one random error so they can be used, unless a bug control circle is required. The BCH codes are one of the most powerful random correction cycle codes. BCH codes can be defined by two parameters of code size n and the number of errors to be corrected t. In the 1960s Bose, Ray - Chaudhuri, Hocquenghem, developed BCH codes independently .Bose, Chaudhuri and Hocquenghem (BCH)) coding is the most common form of Haming coding to correct many errors. (Bose-Chaudhuri-Hocquenghem) BCH codes form a large category of random random error that corrects cycle codes [7] - [9] capable of correcting many errors . They are a powerful class of cyclic codes with the ability to correct many errors and well-defined mathematical features. Galois Field or Finite Field Theory describes the mathematical features of BCH codes. BCH codes are widely used in mobile communications, computer networks, satellite communications, and storage systems such as computer memory or compact disc . In recent years there has been a growing need for a digital transfer and storage system and accelerated development and acquisition of VLSI technology and digital processing. The Programmable Logic Device (PLD) and Field Programmable Gate Arrays (FPGAs) , have transformed computer infrastructure and its implementation benefits offer a variety of solutions as FPGA is fully redesigned and remodeled. - (15, k) BCH code edited by LFSR control to correct single, double and triple error using VHDL, in FPGA presentation and performance comparisons based on combination and simulation results to understand device usage and time simulation by pointing to Xilinx Artix7 FPGA.

Block code is a set of words, called "Codeword". In Block Codeword Codes are a combination of information pieces and measurement pieces. The bits of information are those bits that carry the message while the bits of measurement provide security and ensure that the code name has the correct structure required for Block codes. Encoder_LFSR generates bit bits and integrates them into pieces of information. In k - bits of information and bits of r-parity the code name n produced will be the sum of the bits of information and equals bits, given as n = k + r. This type of code is called (n, k) blocking code. The codeword position is not adjusted, it can be set at the beginning of the code (MSB) or at the end of the information fragment (LSB). The codeword may be scattered throughout the codeword. There are two types of codewords called, standard codes and non-systematic. The codeword where the pieces of information are stored together is called the systematic codeword and when the pieces of information are scattered are called the non-systematic codeword. In (n, k) block codes, k-bits give 2 ^ k a unique code name, so there is a 2 ^ k code name in the code (n, k).

II. METHODOLOGY

The Galois Garden was founded by Everest Galois. The Galois field has a limited number of elements in it. Finite field theory was introduced around the 18th century, and its value and widespread use were widely recognized in recent years. Galois field is widely used in numerical theory, coding theory and cryptography. The mathematical structures in which the BCH codes are defined also represent the Galois field. Mathematical operations such as Addition, Subtraction, Multiplication and Division are performed using Finite field theory. The most basic axioms of the limited field are:

[1] All the elements in the field form an Abeliana group with an additional “+” operator.

[2] Non-zero elements in the field form a group with a multiplication operator “. ”.

[3] Multiplication by any non-zero factor is the default of the Additive group.

BCH architecture codes use field theory and polynomial over finite filed. To detect any errors that occurred during the transfer, a polynomial test was created. BCH δ code with a δ over-field GF (q ^ m) is formed by finding a polynomial over GF (q), its roots comprising δ the successive power of y.

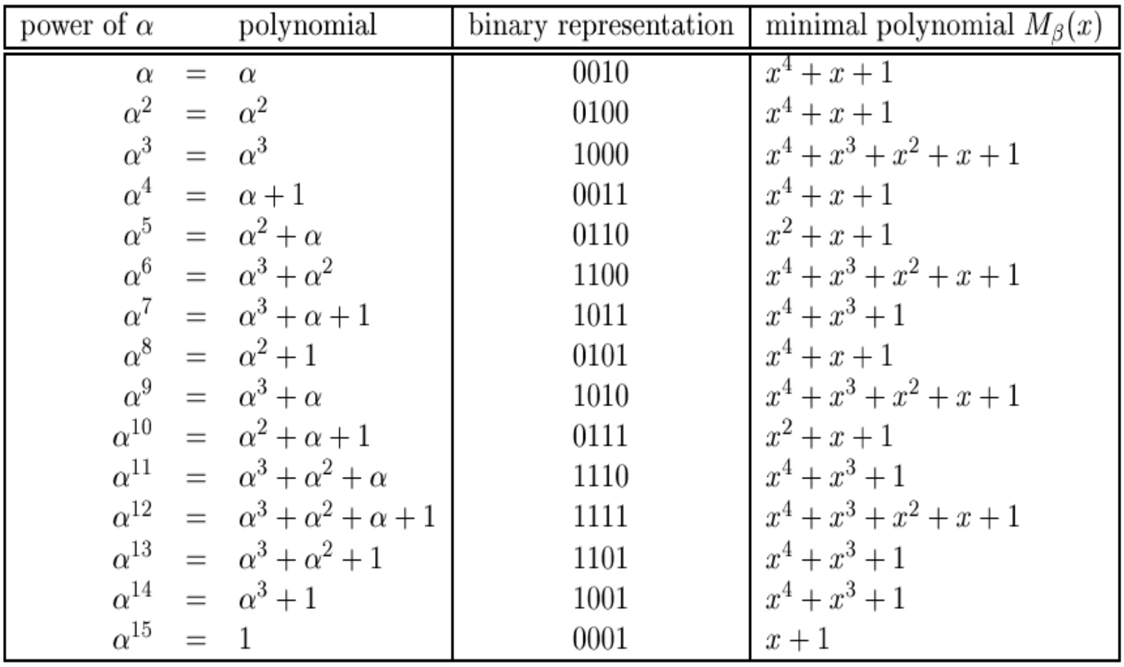
Roots of Equation

Limited field can be calculated using the roots of the number α. This may be due to the property of the cyclic codes. The cyclic code has a feature where all codeword polynomials c (x) have a g (x) generator as a feature. In other words, any root of g (x) = 0, also gives the root of c (x) = 0.

Primitive Polynomials

Non-reduction polynomials are defined as those polynomials that are self-limiting and single and can not be specified. Non-standard polynomials are used to produce GF (2 ^ m). The unstable polynomial root is called primitive polynomial. A is first in GF (2 ^ m); primitive is defined as part of a field that can produce all non-zero field features. Here, as the old α element became the unbreakable source of polynomials, so did the polynomials. In any positive number m we can find at least one unstoppable polynomial of degree m. It can be shown that the non-reduced polynomial of degree m is divided by x ^ r +1 (where r = 2 ^ m -1), which can be used to determine whether polynomial is non-reduced. In order to construct BCH codes with GF (24), we need to detect small polynomials of α power. Small polynomials of all GF (2^4) features are given in the table below

Table 1: Polynomials for GF(2^4)



power of α		polynomial	binary representation	minimal polynomial $M_\beta(x)$
α	$=$	α	0010	x^4+x+1
α^2	$=$	α^2	0100	x^4+x+1
α^3	$=$	α^3	1000	$x^4+x^3+x^2+x+1$
α^4	$=$	$\alpha+1$	0011	x^4+x+1
α^5	$=$	$\alpha^2+\alpha$	0110	x^2+x+1
α^6	$=$	$\alpha^3+\alpha^2$	1100	$x^4+x^3+x^2+x+1$
α^7	$=$	$\alpha^3+\alpha+1$	1011	x^4+x^3+1
α^8	$=$	α^2+1	0101	x^4+x+1
α^9	$=$	$\alpha^3+\alpha$	1010	$x^4+x^3+x^2+x+1$
α^{10}	$=$	$\alpha^2+\alpha+1$	0111	x^2+x+1
α^{11}	$=$	$\alpha^3+\alpha^2+\alpha$	1110	x^4+x^3+1
α^{12}	$=$	$\alpha^3+\alpha^2+\alpha+1$	1111	$x^4+x^3+x^2+x+1$
α^{13}	$=$	$\alpha^3+\alpha^2+1$	1101	x^4+x^3+1
α^{14}	$=$	α^3+1	1001	x^4+x^3+1
α^{15}	$=$	1	0001	$x+1$

The main reason for building GF (2 ^ m) is that they do not have both 0 and 1 as their roots. This section is given a detailed description of the Galois field (2 ^ 4). Consider the equation below,

$$P(x) = x^4 + x + 1.$$

From the figures above, it is clear that no 0 or 1 is the root of the figure. Therefore, we can say that the number 4 is outside the GF field (2 ^ 4). By taking α as one of the numerical root, p (α) should be equal to zero. This can be explained by the equation below,

$$P (\alpha) = \alpha^4 + \alpha + 1 = 0$$

The above figure can be used to create a GF (2^4). Redesigning the above equation provides,

$$\alpha4 = \alpha + 1$$

But the multiplication of α in the above equation gives,

$$\alpha^4 = \alpha + 1$$

$$\alpha^5 = \alpha^4. \alpha = \alpha^2 + \alpha$$

$$\alpha^6 = \alpha^5. \alpha = \alpha^3 + \alpha^2.$$

$$\alpha^7 = \alpha^6. \alpha = \alpha^4 + \alpha^3$$

$$\alpha^8 = \alpha^7. \alpha = \alpha^5 + \alpha^4 = \alpha^2 + 1$$

$$\alpha^9 = \alpha^8. \alpha = \alpha^3 + \alpha$$

$$\alpha^{10} = \alpha^9. \alpha = \alpha^2 + \alpha + 1$$

$$\alpha^{11}= \alpha^{10}. \alpha = \alpha^3 + \alpha^2 + \alpha$$

$$\alpha^{12}= \alpha^{11}. \alpha = \alpha^3 + \alpha^2 + \alpha + 1$$

$$\alpha^{13} = \alpha^{12}. \alpha = \alpha^3 + \alpha^2 + 1$$

$$\alpha^{14} = \alpha^{13}. \alpha = \alpha^3 +1$$

Higher order field elements can be similarly produced by multiplying α from its previous strength. The fifteenth strength of α can be calculated as follows:

$$\alpha^{15} = \alpha^{14}. \alpha = 1.$$

Here, the simplification of the fifteenth system gives 1, which is a factor present in; the extra strength of α will always provide the available features. The GF field (2^4) therefore has the following 16 characteristics:

$$0, 1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6, \alpha^7, \alpha^8, \alpha^9, \alpha^{10}, \alpha^{11}, \alpha^{12}, \alpha^{13}, \alpha^{14}.$$

The BCH code has carefully defined roots to provide excellent error correction. The t-error correction code with generator polynomial g (x) is a BCH binary code if and only if g (x) is a polynomial level lower than GF (2),

β, β2, β3,… β2t

B is a GF element (2m). From the above it is clear that with this root selection, the result codes will be able to correct the t error. The polynomial g (x) generator of t error in binary repair BCH code is provided by,

g (x) = LCM [m1 (x), m2 (x), m3 (x),…., m2t (x)] (2.14)

According to the above equation (2.14), BCH code with error code 3 (15, 5) is considered. A polynomial generator with 2,3, 6 as the roots are obtained by multiplying the following small polynomials:

Small polynomial roots

$$g(x) = f_1(x) * f_3(x) * f_5(x) = 1 + x + x^2 + x^4 + x^5 + x^8 + x^{10}.$$

This chapter includes a detailed description of the Galois field and Block codes. It also incorporates the old polynomial concept associated with the Galois camp and also explains the construction of (15, 5) three BCH code correction errors. The next chapter describes the structure of BCH Encoder_LFSR and architecture.

Since the BCH code applies to Galois Field , it can be defined by two parameters the length of the code names (n) and the error number to be corrected t. BCH binary t-error correction code can fix any combination of t or fewer errors in n = 2^m -1 digit block. For any positive number m ≥3 and t <2^m-1, there is a BCH binary code with the following parameters:

Block length: n =2^m - 1

Number of bits of information: k≥ n-m * t

BCH codes for polynomial codes apply to Galois (or limited) fields. The polynomial generator of this code is specified according to its roots from the Galois field. Let α be the old thing

in. The polynomial code generator is specified by its roots over the Galois GF (2^m) field described in . Let us be old for GF (2^m). The polynomial g (x) code generator is a much lower degree of polynomial than GF (2), with α, α2, α3……… α^{2t}as its roots. α_i [g(α_i)= 0 for 1≤ i ≤ 2t].

Let ϕ_i (x) be the minimum polynomials of then g(x) must be the,

$$g(x)= LCM\{\phi_1 (x), \phi_2 (x),………, \phi_{2t} (x)\} (1)$$

Since the minimal polynomial of conjugate roots is the same α_i = (α_i ') 2l , ϕ_i (x) = $\phi_i{}'$ (x), where = i * 2l of l ≥1, thus polynomial g (x) is made The binary t error corrects the BCH code length given to eqn. (1) can be reduced to

$$g(x) = LCM\{\phi_1 (x), \phi_2 (x),………, \phi_{2t} -1(x)\}$$

The BCH code generated by the genetic elements is given. The unmodified polynomial g (x) of degree m is said to be old only if it divides the polynomial type of degree n, x^{n+1} into n = 2^m -1. In fact, every first polynomial binary g (x) of degree m is a factor of 2^{m-1} + 1. The old polynomial range of degree m and the unstoppable polynomial finding is given. In (15, k) BCH code, α should be the old GF (24) component given so that 1 + α + α^4 is the old polynomial. we find that small polynomials of α, α^3, α^5

$$\phi_1 (x) = 1+ x + x^4$$

$$\phi_3 (x) = 1+ x + \alpha^2 + \alpha^3 + \alpha^4$$

$$\phi_5 (x) = 1+ x + \alpha^3$$

To fix one error, BCH code length n = 24 -1 = 15 is generated by

$$g (x) = \phi_1 (x) = 1+ x + x^4 \qquad (3)$$

Here the highest degree is 4 i.e. (n-k = 4), so the code is code (15, 11) of cyclic code and d_{min}-3 Since a polynomial generator is a polynomial code of weighted 5, the minimum distance of this code is 3.

To fix the double error, BCH code length n = 15 is generated by

$$g(x) = \text{LCM}\{\phi_1\,(x),\ \phi_3\,(x)\} = 1 + x^4 + x^6 + x^7 + x^8 \qquad (3)$$

Here highest degree is 8 i.e (n-k = 8), thus the code is a (15, 7) cyclic code with $d_{min} \geq 5$.

For triple error correcting, BCH code of length n = 15 is generated by

$$g(x) = \text{LCM}\{(\phi_1(x),\ \phi_3(x),\ \phi_5(x))\}$$

$$= 1 + x + x^2 + x^4 + x^5 + x^8 + x^{10} \qquad (4)$$

Here highest degree is exactly 10 i.e (n-k = 10), thus the code is a (15, 5) cyclic code with d_{min} greater than or equal to 7.

The polynomial code generator is specified by its roots over the Galois GF (2m) field. A should be an old element in GF (2m) .The polynomial g (x) code generator is a much lower polynomial degree than GF (2). Let mi (x) be the minimum of α_i polynomials and the polynomial G (x) generator can be calculated electronically G (x) = LCM [m_1 (x), m_3 (x),...., m_{2t} (x)] (1) In this function consider n = 15, k = 4 and t = 4. To create BCH codes over GF (4), we need to find the GF (4) features generated by p (x) = 1 + x + x^4 given in the Table below.

Table 2: The elements of GF (2^4) generated byp(x) =1+x+x^4

Powers of α	Elements	Minimal polynomial
α^1	z	x^2+x+2
α^2	z+2	x^2+x+3
α^3	3z+2	x^2+3x+1
α^4	z+1	x^2+x+2
α^5	2	x+2
α^6	2z	x^2+2x+1
α^7	2z+3	x^2+2x+2
α^8	z+3	x^2+x+3
α^9	2z+2	x^2+2x+1
α^{10}	3	x+3
α^{11}	3z	x^2+3x+3
α^{12}	3z+1	x^2+3x+1
α^{13}	2z+1	x^2+2x+2
α^{14}	3z+3	x^2+3x+3
α^{15}	1	1

The polynomial g (x) code generator is a much lower degree of polynomial than GF (2). Let mi (x) be minimum αi polynomials and then polynomial G (x) generator can be calculated electronically

$$G\,(x) = \text{LCM}\,[m_1\,(x),\ m_3\,(x),\ldots,\ m_{2t}\,(x)]\ (1)$$

In this function n = 15, k = 4 and t = 4 are considered. Thus a Polynomial generator with α, α^2, ... α^4 as the roots are obtained by multiplying the following small polynomials:

$$m_1\,(x) = x^2+x+2$$

$$m_2\,(x) = x^2+x+3$$

$$m_3\,(x) = x^3+3x+1$$

$$m_4\,(x) = x^2+x+2$$

$$m_5\,(x) = x+2$$

$$m_6\,(x) = x^2+2x+1$$

$$m_7\,(x) = x^2+2x+2$$

$$m_8\ (x)=x^2+x+3$$

Converting m_1(x), m_2(x), m_3(x), m_4(x), m_5(x), m_6 (x), m_7 (x) and m_8 (x) in the equation (1) to -generator polynomial.

$$G\ (x) = LCM\ \{m_1\ (x), m_2\ (x), m_3\ (x), m_4\ (x), m_5\ (x), m_6\ (x), m_7\ (x), m_8\ (x)\}$$

$$G(x)=\{(x^2+x+2\ ),(x^2+x+3),(x^3+3x+1),(x^2+x+2),(x+2),(x^2+2x+1),(x^2+2x+2),(x^2+x+3)\}$$

$$G(x)\ =x^{11}+x^{10}+2x^8+3x^7+3x^6+x^5+3x^4+x^3+x+3\ (2)$$

III. MODELING AND ANALYSIS

(15, 4) BCH Encoder is used with Linear Feedback Shift Register (LFSR). (15, 4) BCH codeword is written as follows.

Let c (x) be the polynomial codeword and e (x) be the polynomial error. Then the approved polynomial can be labeled as

$$v\ (x) = c\ (x) + e\ (x).$$

When the polynomial coefficients are in GF (q^m). If g_1, g_2 ,, ……, g_p then those GF (q^m) elements are zero (g), i.e., g (g_i) = 0 to I = 1,…, p . As u_c(x) = a (x) g(x) of polynomial a (x), we also have c (g_i) = 0 of I = 1, …… , p.

Thus, v (g_i) = c (g_i) + e (g_i)

= e (g_i) of I = 1,…., p.

So we have a set of p-calculations that include only parts of the error pattern. If it is possible to solve this e_i mathematical set, the error pattern can be determined accurately. Whether this set of statistics can be solved depends on the value. of p, zero number g (x).In order to solve the error pattern, we must select a set of p numbers correctly. a set of calculations that I can solve at least t non-zero in J. Let's define the syndromes of S_i = e (g_i) in I = I,…., p. errors can be calculated in S_1, S_2, …….,S_p. If α is an old factor it means a set of g_i that allows for correction of t

$$\{\alpha^1, \alpha^2, \alpha^3, \ldots\ldots\ldots, \alpha^{2t}\}.$$

So we have a simple machine to determine the polynomial generator of BCH code that can fix t errors.

Figure 1 shows the block diagram of the (15, 4) BCH Encoder module. The seven message bits (M0, M1….M3) are sent to the corresponding serial shift register. The shift register being used is parallel to serial shift register which has multiple input lines in this case we have 15 input lines and single output line. The result of the parallel to serial shift register will be sent to (15, 4) BCH Encoder module as shown in the figure. Using these parity bits are computer generated and sent to the serial to the corresponding shift register. These measurement pieces have been added to the actual message bits for 15-bit encrypted data. We also use a synchronous clock for the BCH Encoder. The clock signal synchronizes all the operations of various parallel to shift registers and ex-or gates of the BCH encoder. The ex-or operation is performed among the various input bits.

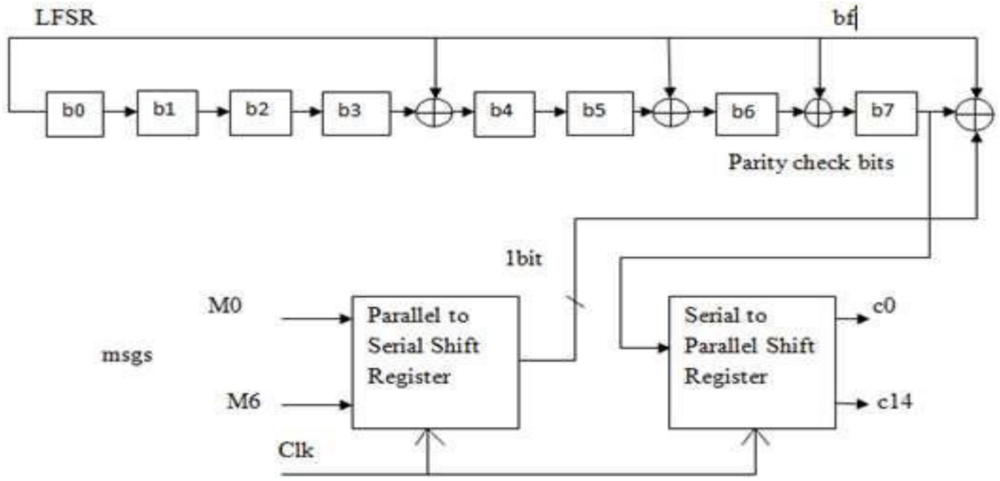
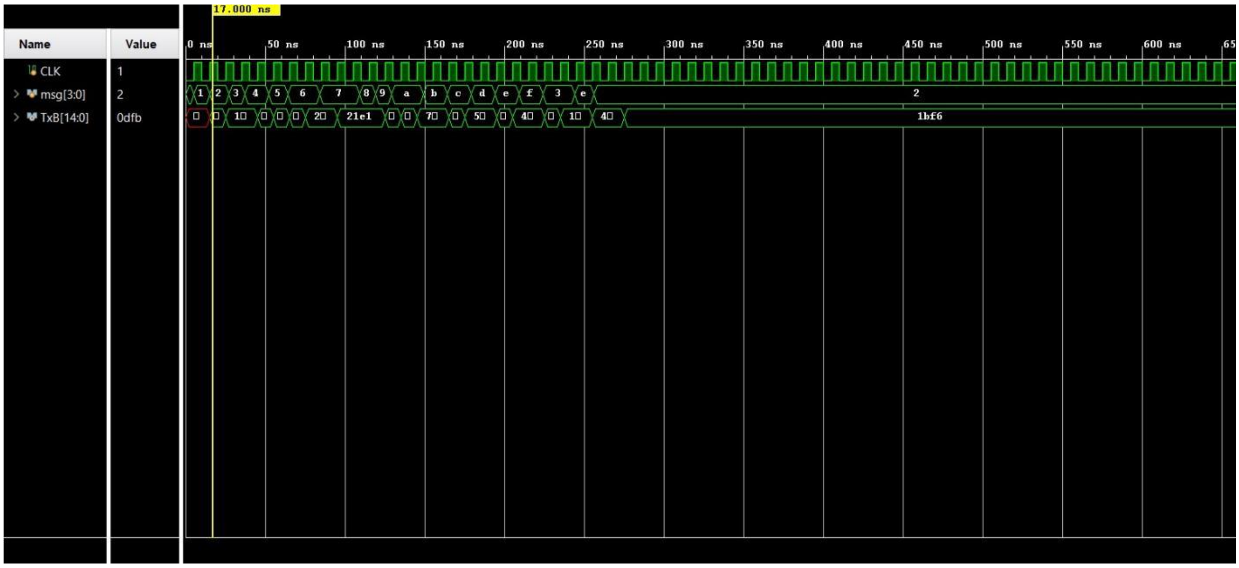

LFSR
bf
b0
b1
b2
b3
b4
b5
b6
b7
Parity check bits
1bit
M0
msgs
M6
Parallel to Serial Shift Register
Serial to Parallel Shift Register
c0
c14
Clk

Fig 1: Block diagram of (15,4) BCH Encoder

Four message bits can be sent. Total 15 bits sent 11 check bits are added. To know the error position codeword is done ex-or with received codeword. The position where the error is present turns out to exist in the result of the ex-or operation. The output of the serial to parallel shift register generates the 15 bits say from c0 to c14.These 15 bits are used to correct the four error bits that are present in the sent message bits using the converse operation.

IV. RESULTS AND DISCUSSION



Name	Value
CLK	1
msg[3:0]	2
TxB[14:0]	0dfb

Fig 2: Timing diagram

RTL Schematic:

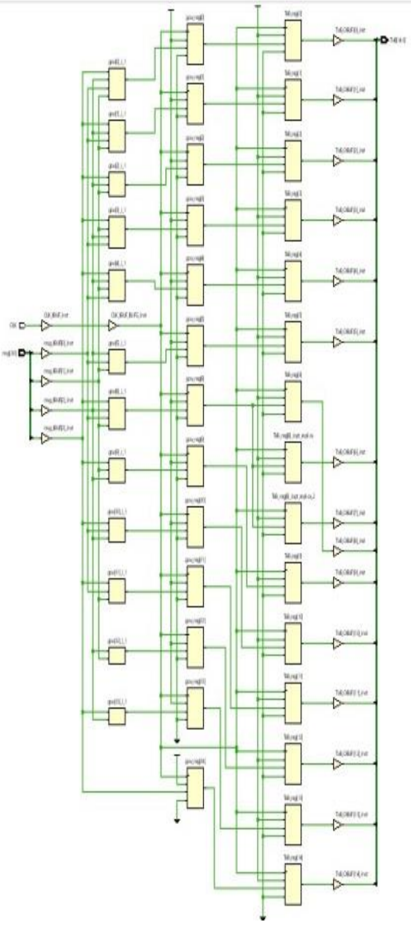
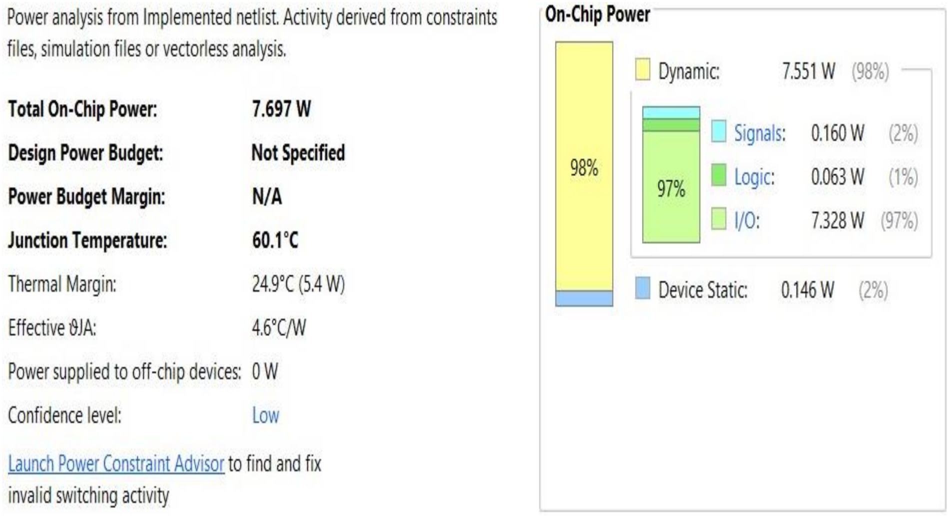


Fig 3: Schematic diagram

Power report:



Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power: **7.697 W**
Design Power Budget: **Not Specified**
Power Budget Margin: **N/A**
Junction Temperature: **60.1°C**
Thermal Margin: 24.9°C (5.4 W)
Effective ϑJA: 4.6°C/W
Power supplied to off-chip devices: 0 W
Confidence level: Low

Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power

Dynamic:	7.551 W	(98%)
Signals:	0.160 W	(2%)
Logic:	0.063 W	(1%)
I/O:	7.328 W	(97%)
Device Static:	0.146 W	(2%)

Fig 4: Power report

V. CONCLUSION

To ensure the reliable transmission of information using a mobile or wireless medium, error control codes applied to digital information and communication systems.

In this paper we have introduced the implementation (15,4, t = 4) BCH connector. Synthesis was successfully performed using Xilinx ISE 14.2 and this design was successfully performed on Artix7 FPGA. Here are 4 message bits embedded 15 bit code. If there is any 4 bit error in any 15 bit codeword area, it can be detected and corrected. The encoder is used using the LFSR. The proposed polynomial field of the Galois field is used to calculate the syndrome and to detect the error of finding polynomial coefficients. Allows for faster field duplication. The BCH code forms a large section of cycle codes to correct random errors. They are relatively easy to code. In addition, performance can be improved by using similar method.

VI. REFERENCES

[1] Neubauer, J. Freudenberger and V. Kuhn “Coding Theory Algorithms, Architectures and Applications” John Wiley & Sons, 2007.

[2] T. K. Moon, “Error Correction Coding”, John Wiley & Sons, 2005.

[3] Shyue-Win Wei, Che-Ho Wei, “High-speed hardware decoder for double-error-correcting binaryBCH codes”, IEEE Proceedings, vol 136, no. 3, pp. 227-231, June 1989.

[4] Amit Kumar Panda, “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.

[5] Warren J. Gross, Frank R. Kschischang, “Applications of Algebraic Soft-Decision Decoding of Reed–Solomon Codes”, IEEE transactions on communications, Vol. 54, No. 7, July 2006.

[6] Claude Berrou, Alain Glavieeux, “Near optimum error correcting coding and decoding Turbo codes”, IEEE transactions on communications Vol.44, No.10, October 1996.

[7] R.T. Chien, “Cyclic decoding procedure for Bose Chaudhuri Hocquenghem Codes”, IEEE Trans. on Information Theory, vol. IT-IO, pp. 357-363, October 1964.

[8] Da-Chun Wu, AND Ming-Kao Hsu, “Authentication of Binary Document Images Based on Embedding the BCH Codes of Watermarks”, Asian Journal of Health and Information Sciences, Vol. 1, No. 4, pp. 446-455, 2007.

[9] Shu Lin, Daniel J. Castello, “Error control coding, Fundamentals and applications”, Premtice-Hall, New Jersey, 1983, Pages 15-50.

[10] R. Lidl and H. Niederreiter, “Finite Fields”, Cambridge University Press, Cambridge, 1966.

[11] John Gill, “Finite Fields”, Stanford University

[12] D.Muthiah, A. Arockia Bazil Raj, “Implementation of High-Speed LFSR Design with Parallel Architectures”, International Conference on Computing Communication and Applications (ICCCA),pp. 1-6, Feb. 2012.

[13] Rohith S, Pavithra S “FPGA Implementation of (15,7) BCH Encoder and Decoder for Text Message”, International Journal of Research in Engineering and Technology, vol. 2, pp. 209-214,Sep 2013

[14] J J.Rose S.D. Brown, R.J. Francis – “Field Programmable Gate Arrays”, Kluwer Academic Publishers, 1992

[15] Brown S., Vranesic Z “Fundamental of Digital Logic Design with VHDL” McGraw Hill, 2nd Edition.