IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 9, Issue 10, October 2020

DOI 10.17148/IJARCCE.2020.91020

A Survey on Error Detection Techniques

Nishanth S[1]

B.E., Department of ECE, BIT, Bangalore, India[1]

Abstract: During this paper, numerous error detection techniques are analyzed. To attain high performance and higher quality of data transmission, error detecting and correcting techniques are used. Many various errors are detecting and correcting techniques are available. It will improve the performance similar to the quality of the data transmitted. When a digital signal is transmitted between two systems, the signal gets distorted owing to the addition of noise to the transmitted signal. The noise may introduce an error is the binary train of bit travelling from one system to the other. This suggests that a 0 may amend to 1 or a 1 may amend to 0. This Error may become a severe issue with the accuracy and performance of the digital system. Therefore, it is essential to detect and correct errors. In data communication errors are introduced during the transmission of data from the transmitter to the receiver because of noise or some other reasons. The reliability of data transmission gets severely affected and distorted owing to these added errors on data. The detection and correction of transmission error another one or more than one extra bit during the time of transmitting the signal. So we found some extra bits which are called parity bits. These parity bits generally detect or sometimes correct the errors.

Keywords: Error detection techniques, Hamming Code, Even Parity generator, Odd Parity generator, Counting technique, FPGA [Field Programmable Gate Array], HVD [Horizontal Vertical Detection], CRC [Cyclic Redundancy Check], Matrix codes, Power, Delay.

I. INTRODUCTION

For booming data communication, to use bound techniques for data communication & need at least two devices or two machines, one is sender another one is received. They require an excellent deal of coordination between these two devices or machines for economical and efficient data communication. Throughout data communication, because of attenuation, distortion, noise interferences, some errors can occur, and this can result in corruption of the number of the bits. The transmission media are noisy; the probability of single-bit Error is going to be more when sending data in the form of a frame. The dimensions of the frame will also detect and choose the probability of Error of a frame. Though many techniques and approaches are planned and applied within the last decade, data reliability in transmission continues to be a retardant. Error detection and correction techniques are thus needed. A number of those techniques can solely detect errors; others are designed to detect as well as correct errors. At Error correction methodology automatic repeat request (ARQ), once an error is detected; the receiver sends a request to the transmitter for repeat transmission when that happens, the transmitter retransmits. The ARQ method wants duplex arrangement as apart from the traditional transmitter to receiver signal, the request signal is to travel from receiver to transmitter.

On the other hand, the forward error correction (FEC) methodology wants simplex arrangement as the signal should hassle solely from the transmitter to the receiver. Therefore, FEC methodology is a lot more fashionable than ARQ methodology. In error detection methodology CRC is predicated on binary division. Within the parity checking strategies, an additional bit call as parity bits is added to every date word. Even parity or odd parity bit is the methodology to detect the errors. When even parity is used, to make the total number of 1 bit even, A parity bit is added to it.

Similarly, for odd parity, the total number of 1 bit is created out by adding the parity bits. Suppose the number of errors introduced within the transmitted code is 2 or any even number. In that case, the parity of the received codeword will not amend; it will remain even on that case parity checking methodology not detect the error or error detection method may fail. So, this is the demerit of the parity checking methodology.

II. TYPES OF ERROR DETECTION TECHNIQUES

A. Hamming Code: As we tend to undergo the example, the primary step is to spot the bit position of the data & all the bit positions that are powers of two are marked as parity bits (e.g. 1, 2, 4, 8, etc.). The following image can facilitate in visualizing the received hamming code of seven bits.

D7	D6	D5	P4	D3	P2	P1
1	0	1	1	0	1	1

First, we would like to detect whether or not there are any errors in this received hamming code.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 9, Issue 10, October 2020
DOI 10.17148/IJARCCE.2020.91020

Step 1**:** For checking parity bit P1, use check one and skip one methodology, which suggests, starting from P1 so skip P2, take D3 then skip P4 then take D5, and so skip D6 and take D7, this manner we will have the following bits,

D7	D5	D3	P1
1	1	0	1

As we can observe the whole range of bits is odd. Therefore, we will write the value of the parity bit as P1 = 1. This suggests that Error is there.
Step 2**:** Check for P2, but while we are checking for P2, we will use check two and skip two methods, which will offer us the following data bits. However, keep in mind since we are checking for P2, so we have to start our count from P2 (P1 must not be considered).

D7	D6	D3	P2
1	0	0	1

As we can see that the number of 1's are even, then we will write the value of P2 = 0. This suggests that there is no error.
Step 3**:** Check for P4, but while we are checking for P4, we will use check four and skip four methodologies, which will offer us the following data bits. However, remember since we are checking for P4, therefore started our count from P4(P1 & P2 must not be considered).

D7	D6	D5	P4
1	0	1	1

As we can see that the number of 1's are odd, then we will write the value of P4 = 1. This suggests the Error is there.
So, from the above parity analysis, P1 & P4 are not equal to 0, so we can clearly say that the received hamming code has errors.

B. Even Parity Generator

Even a parity generator is a form of parity code shown in Fig 1. Here there are seven-bit inputs that are fed to the XOR gate. It checks the number of ones; if there is an even number of ones, the output is zero, and if there is an odd number of ones, the output is going to be one. This extra line goes onto the bus at the side of the original data. Through the extra line info, the receiver will recognize the Error present in the info.

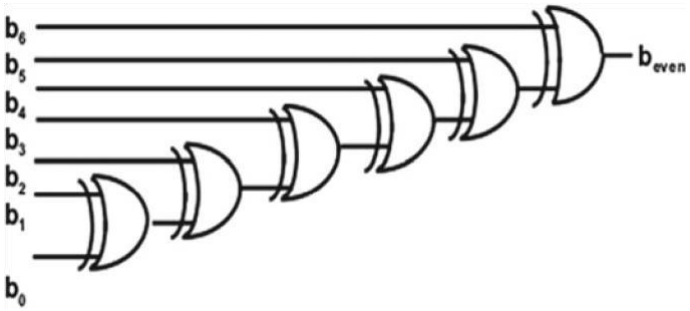

Fig 1 even parity generator

C. Odd Parity Generator

An odd parity generator is a form of parity code shown in Fig 2. Here there are three-bit inputs that are fed to the XOR gate and XNOR gate. It checks the number of ones; if there is an odd number of ones, the output is zero, and if there is an even number of ones, the output is going to be one.
This extra line goes onto the bus along with the initial data. Through the extra line information, the receiver will recognize the Error present in the info.
Apart from these techniques, we have blocked the parity codes technique. During this technique, the transmitter transmits the data as a block; every block consists of many binary words. Parity bits can be allocated to both rows and columns.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 9, Issue 10, October 2020
DOI 10.17148/IJARCCE.2020.91020

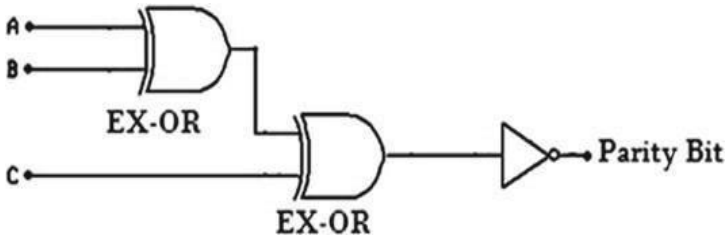

Fig 2 odd parity generator

D. Counting Technique

One way of determining whether or not a combinational circuit operates properly is by applying to the circuit all doable input combos and scrutinizing the resultant outputs with either the corresponding truth table or a perfect version of the same circuit. It shows the presence of a fault if it indicates deviation. Moreover, suppose a better-known relationship exists between the various doable faults and also the deviations of output patterns. In that case, it is possible to diagnose the Error and to classify it at least within a subset of faults whose effects on the circuit outputs are same.

The planned technique has been designed for seven-bit input. The planning can vary looking on the need. Here a0,a1,a2 show the number of ones out there at the inputs excluding generating the parity bits. a0, a1, and a2 display the weighted values. If we have input 0001111, then the output is going to be generated as 100, for 1001000, the output is going to be 010, which means 2. Here 2 say that there are two ones within the input lines. At the transmitting time, three extra lines are needed. At the receiver end, original data will reach along with these other lines. It informs the receiver regarding the number of ones/zeroes out there in the input line excluding generating parity bits. This method makes the analysis easier compared to even and odd parity generators. If there are any faults on the bus because of stuck-one or stuck-zero, we can analyze the input data quickly. The design will be modified with a different number of inputs.

Figure 3 shows the diagram of "error detection using a counting technique." b0, b1, …, b6 are seven-bit inputs, whereas a0, a1, and a2 are output.

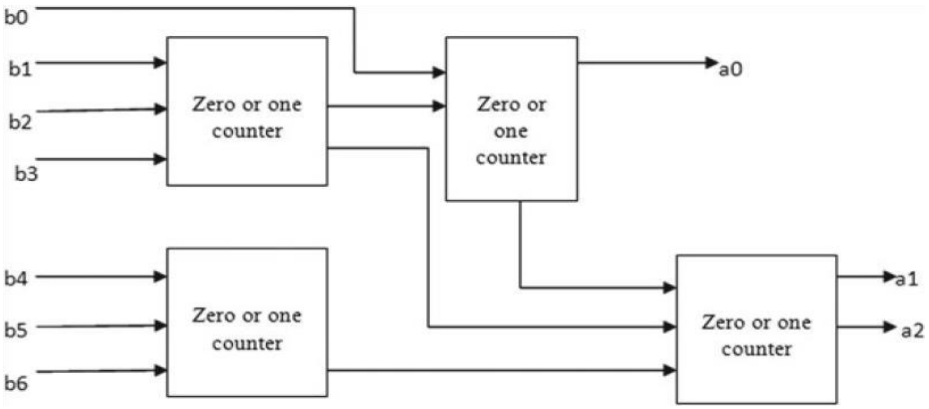

Fig 3. Block diagram of the counting technique

E. FPGA Based Error Detection

Orthogonal codes have equal numbers of 1's and 0's, and they are binary-valued. An n-bit orthogonal code has n/2 1's and n/2 0's; i.e., there are n/2 positions wherever 1's and 0's differ. Therefore, all orthogonal codes can generate zero parity bits. The thought is illustrated by 8-bit orthogonal codes, as shown in Fig. 4.1. It has eight orthogonal codes and eight antipodal codes for a complete of sixteen bi-orthogonal codes. Antipodal codes are simply the inverse of orthogonal codes; they have the same properties.

A notable distinction in this methodology is that the transmitter does not have to send the parity bit for the code, since it is better-known to be always zero. Therefore, the receiver will be able to detect it by generating a parity bit at the receiving end, if there is a transmission error.

Before transmission, an m-bit information set is mapped into a novel n-bit orthogonal code. As an example, a 4-bit information set is described by a novel 8-bit orthogonal code, that is transmitted without the parity bit. When received, the information is decoded based on code correlation. It will be done by setting a threshold between 2 orthogonal codes. This is set by the following equation:

$$d_{th} = n/4 \text{ ---------(1)}$$

Where d_{th} is the threshold, that is midway between 2 orthogonal codes and n is the code length. Therefore, for the 8bit orthogonal code (Fig. 4.2), we've $d_{th} = 8/4 = 2$.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 9, Issue 10, October 2020

DOI 10.17148/IJARCCE.2020.91020

This method offers a decision process in error correction, wherever the incoming impaired orthogonal code is examined for correlation with the codes present in a look-up table, for a possible match. The acceptance criterion for a valid code is that an n-bit comparison should yield a decent cross-correlation value; otherwise, a false detection can occur. A pair of n-bit codes x1x2…xn and y1y2…yn is compared, where it is governed by the following correlation method to give,

Orthogonal Code									P				Antipodal Code									P
0	0	0	0	0	0	0	0		0				1	1	1	1	1	1	1	1		0
0	1	0	1	0	1	0	1		0				1	0	1	0	1	0	1	0		0
0	0	1	1	0	0	1	1		0				1	1	0	0	1	1	0	0		0
0	1	1	0	0	1	1	0		0				1	0	0	1	1	0	0	1		0
0	0	0	0	1	1	1	1		0				1	1	1	1	0	0	0	0		0
0	1	0	1	1	0	1	0		0				1	0	1	0	0	1	0	1		0
0	0	1	1	1	1	0	0		0				1	1	0	0	0	0	1	1		0
0	1	1	0	1	0	0	1		0				1	0	0	1	0	1	1	0		0

Fig 4.1. Illustrations of 8-bit orthogonal codes.

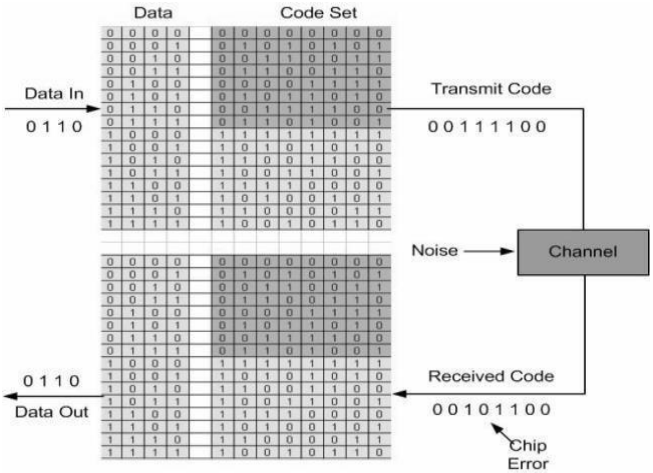

Fig 4.2. Illustration of OCC Encoding and Decoding.

$$R(x, y) = x_i y_i \leq d_{th} \quad \text{----------(2)} \quad \{ i = 1 \text{ to } n \}$$

here dth is the threshold, and cross-correlation function is R(x, y). an additional 1-bit offset is added to (2) for reliable detection, because the threshold (dth) is between 2 valid codes The number of errors that can be corrected employing this process can be calculated by combining (1) and (2), yielding,

Table I Orthogonal Codes and Their Corresponding Error Correction Capacities.

Length of code *(n)*	**Correction Capacities *(nc)***
8	1
16	3
32	7
64	15

Table II: Orthogonal Codes and Their Corresponding Detection Rates.

Length of code *(n)*	**Detection Rate (%)**
8	71.880
16	80.349
32	86.010
64	90.07

A counter is employed to count the number of 1's within the resulting signal. As an example, for 8-bit orthogonal code, the operation can result in sixteen counter results. If one among the results is zero, it suggests that there is no error.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 9, Issue 10, October 2020
DOI 10.17148/IJARCCE.2020.91020

Otherwise, the code is corrupted. The minimum count is associated with corrected code. The received and corrected code will be in the same combination, If the minimum count is associated with one combination. However, it is impossible to correct the corrupted code, if the minimum count is associated with more than one combination of the orthogonal codes.

F. HVD Based Error Detection

A t the receiver end, the entire array is calculated as parities in all directions(example, we can consider in a horizontal direction from hi to hs in fig 5). These calculated parities are compared with the received parties. If the results of comparison do not show any difference, it suggests that the received information at the receiver is correct therefore no correction is required; however if there is a distinction between the received and calculated parties, the inaccurate parity lines are identified, and then the correction process starts.

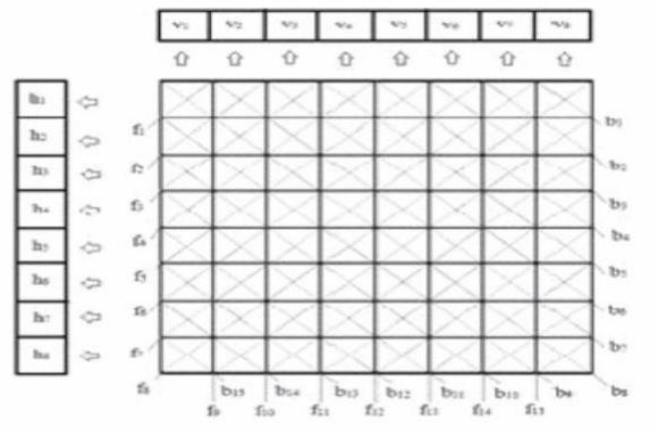

Fig 5 array of bits

G. Cyclic Redundancy Check

The incoming frame is divided by the receiver by that number, and if there is no remainder, it will be assumed as no error. The information of m bits and then additional zeros are concatenated with it; therefore, this can be divided by n + 1-bit divisors that are carefully chosen within the bit sequence. Which is employed as a divisor to divide the M + n bit sequence and it provides n bit information. Then divide a number by n + 1 bit and get the remainder of n bits and that n bit is used as CRC bits. These zeroes and zeros are replaced by CRC, so we get the data + CRC. This is transmitted through the transmission media; therefore, through the transmission media, it is sent at the receiving end. The receiver receives m + n bits, data + CRC and here it is once more divided by the same divisor. The divisor should be the same in both cases at the transmitting and receiving end, and after division, it generates a reminder. If the remainder is zero, then it is accepted as correct data if it is not zero then it is rejected. Therefore, find that by division, at both receiving end and the transmitting end, a division is performed to generate CRC and also to check whether or not the received information including a CRC is correct or not. Then it is accepted or rejected based on whether the remainder is zero or not. This type is based on modulo 2, which is arithmetic.

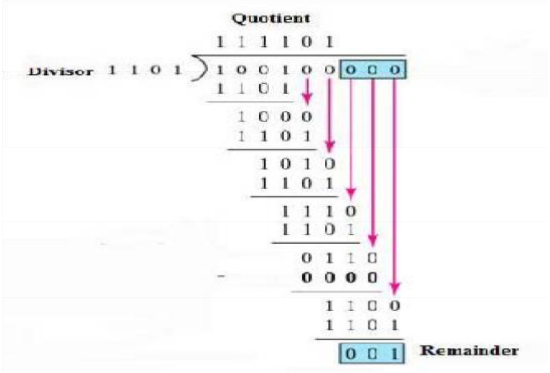

Fig 6.1. CRC at receiver side

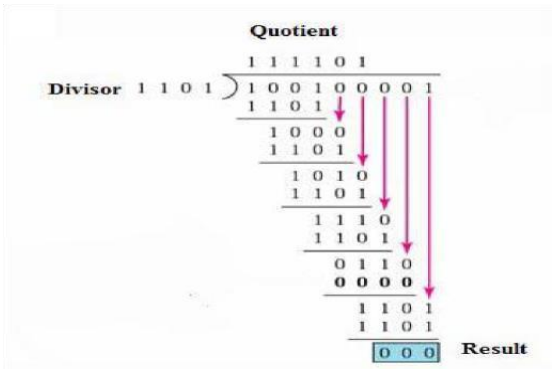

Fig 6.2. CRC at the sender side

H. Matrix Codes

Matrix Code is based on a combination of Hamming codes and Parity codes in a matrix format, so the detection and correction of multiple errors are achieved. The protection bits are utilized in a matrix format. The n-bit codeword is split into k_1 subwords of width k_2. A (k_1, k_2) matrix is created where and represents the numbers of rows and columns,

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 9, Issue 10, October 2020
DOI 10.17148/IJARCCE.2020.91020

respectively. For each of the k_1 rows, the check bits are appended for single error correction/double error detection. Other k_2 bits are appended as vertical parity bits.

III. COMPARISON ANALYSIS OF DIFFERENT TECHNIQUES OF ERROR DETECTION

Parameters	Power (mW)	Delay (nS)
Hamming code	163×10^{-3}	17.133
Even parity generator	550.69×10^{-6}	18.31×10^{-3}
Odd parity generator	88.7×10^{-3}	313.52×10^{-3}
Counting technique	141.91×10^{-3}	1.0570
FPGA based error detection	121×10^{-3}	17.587
HVD based error detection	0.12	20.958
Cyclic Redundancy Check	21.7×10^{-6}	10.33×10^{-3}
Matrix Codes	105×10^{-3}	14.548

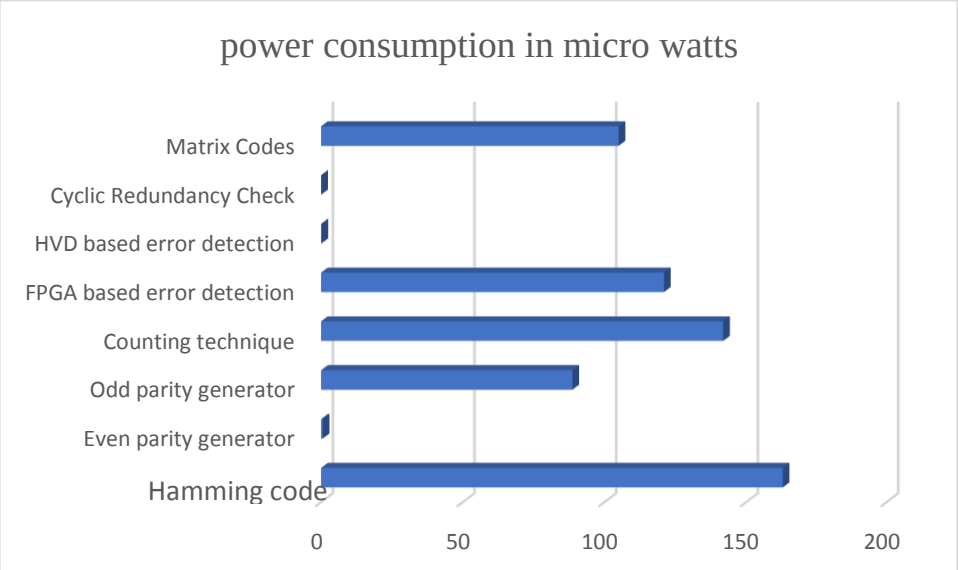

Fig7.1 Power consumption

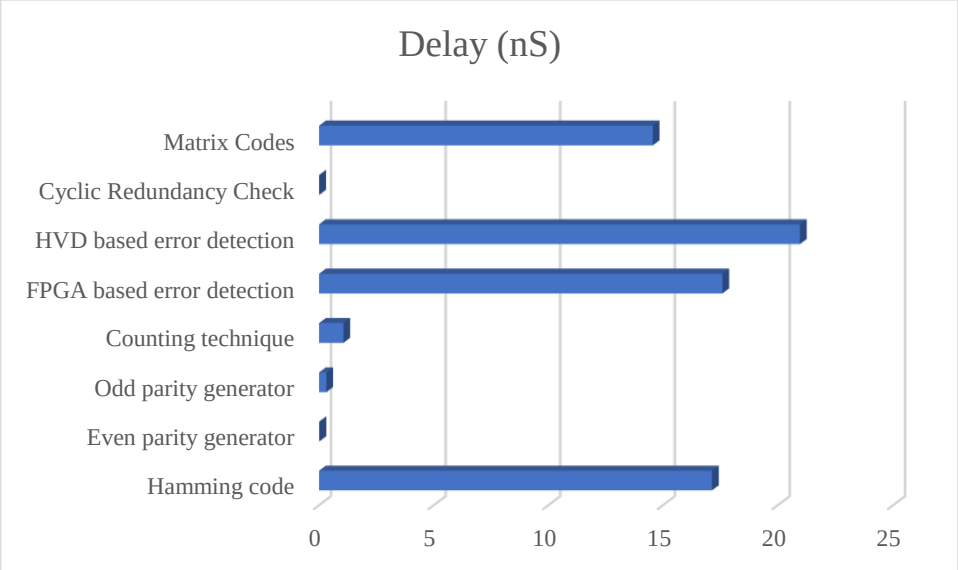

Fig7.2 Delay

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 9, Issue 10, October 2020

DOI 10.17148/IJARCCE.2020.91020

a. Power Comparison

The Hamming Code method consumes more power when compared to the other methods as observed from the above graph of power consumption (fig 7.1), the Matrix Codes, FPGA, counting technique and Odd Parity generator technique nearly have similar microwatts of power consumption. Whereas the Cyclic Redundancy technique has the least power consumption value, thus making it the most efficient error correction technique among the others.

b. Delay Comparison

The Matrix Codes technique shows a much lesser delay than the FPGA, HVD techniques. Hamming code shows similar Delay levels when compared to FPGA technique. Cyclic Redundancy technique shows the least delay among the other techniques according to the above Delay graph (fig 7.2)

IV. CONCLUSION

Thus, the paper presents different types of error detection techniques which are employed for the detection of errors during data transmission. This type of improvement will increase the reliability and efficiency of data transmission. In the future, more improvements will be made for more efficient transmission of data. The paper also compares and analyses the different techniques used for error detection based on this comparison The Cyclic Redundancy technique consumes the least power out of the rest, similarly has a delay value much lesser than that of the Even Parity generator technique. Thus, making it the most efficient and reliable technique to be used for error detection.

REFERENCES

[1]. S. Sharma, *Digital Communication*, 2nd edition. (S.K. Kataria & Sons Publication)
[2]. C.E. Shannon, A mathematical theory of communication. Bell Syst.
[3]. S. Lin, D.J. Jr. Costello, *Error Control Coding: Fundamentals and Applications* (PrenticeHall, 1983). ISBN 0-13-283796-X
[4]. W.C. Huffman, V.S. Pless, *Fundamentals of Error-Correcting Codes* ISBN 978-0-52178280-7
[5]. F.J. Mac Williams, N.J.A. Sloane, *The Theory of Error-Correcting Codes* (North-Holland, 1977), p. 35. ISBN 0-444-85193-3
[6]. J.H. van Lint, *Introduction to Coding Theory*. *GTM*. 86, 2nd edition. p. 31. ISBN 3-54054894-7
[7]. W.E.Ryan, S.Lin,*ChannelCodes:ClassicalandModern*(CambridgeUniversityPress,2009),
[8]. M. Greferath, An introduction to ring-linear coding theory, in *Gröbner Bases, Coding, and Cryptography* ed. by M. Sala, T. Mora, L. Perret, S. Sakata, C. Traverso (Springer Science & Business Media,2009). ISBN 978-3-540-93806-4
[9]. Anlei Wang, 'FPGA Based Design of a Novel Enhanced Error detection and correction technique' IEEE proceedings
[10]. Kumud Kumar Bharadwaj and T.Swapna Rani 'Error Detection Using Counting Technique in Low Power VLSI'
[11]. Shubham Fadnavis 'An HVD Based error detection and correction code in HDLC protocol used for communication' international journal of advanced research in computer and communication engineering vol.2, issue 6, June 2013
[12]. Sandeep Singh, Neeraj Gupta, Rashmi Gupta 'Implementation of various error detection and correction techniques in communication' international journal of Innovation research in science, engineering and technology, vol.7, issue 10, October 2018
[13]. E.Abinaya, Dr D.Somasundareshwari, S.Mathan Pradad 'Error Detection and correction in 32-Bit SRAM based FPGA memory using decimal Matrix code', international journal of Innovation research in science, engineering and technology, vol.4, special issue 01, February 2015
[14]. B.Balagi, N Ajaynagendra, Erigela Radhamma, A Krishna Murthy, M Lakshana Kumar 'Design of efficient 16 Bit Crc with optimized power and area in Vlsi Circuits', international journal of innovative technology and exploring engineering, vol.8, issue 8, June 2019 ISSN: 2278-3075
[15]. C. A. Argyrides, P. Reviriego, D. K. Pradhan, and J. A. Maestro, "Matrix-based codes for adjacent error correction," IEEE Trans. Nucl. Sci., vol. 57, no. 4, pp. 2106–2111, Aug. 2010.
[16]. F. Alzahrani, and T. Chen, "On-chip TEC-QED ECC for ultra-large, single-chip memory systems," in Proc. IEEE Int. Conf. Comput. Design Design, Very-Large-Scale Integr. (VLSI) Syst. Comput. Process., Oct. 1994, pp. 132–137.
[17]. T. Baicheva, S. Dodunekov, and P. Kazakov, "Undetected error probability performance of cyclic redundancy-check codes of 16-bit redundancy", IEEE Proceedings. Communications, Vol. 147, No. 5, pp. 253-256, Oct. 2000.
[18]. Stylianakis, S. Toptchiyski, "A Reed-Solomon coding/decoding structure for an ADS modem", in Conf. Rec. 1999 IEEE Int. Conf. on Electronics, Circuits and Systems, pp. 473 – 476.
[19]. P. Reviriego, M. Flanagan, and J. A. Maestro, "A (64,45) triple error correction code for memory applications," IEEE Trans. Device Mater. Rel., vol. 12, no. 1, pp. 101–106, Mar. 2012
[20]. Mathukiya and M. P. Naresh. "A Novel Approach for Parallel CRC generation for high-speed application." Communication Systems and Network Technologies
[21]. Fadnavis, An HVD based error detection and correction code in HDLC protocol used for communication. Int. J. Adv. Res. Comput. Commun. Eng. 2(6), 2349–2353 (2013)
[22]. U K Kumar and BS Umashankar, 'Improved hamming code for error detection and correction', 2007 2nd edition International Symposium on Pervasive wireless computing, pp.498-500