International Journal of Engineering & Technology, 7 (4) (2018) 3793-3798

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet. v7i4.15685
Research paper

A novel hybrid error detection and correction method using VHDL

Adham Hadi Saleh [1] *, Omar A. Imran [1], Weaam Talaat Ali [1], Adnan M. Taha [1], Wisam Najm Al-Din Abed [1]

[1] University of Diyala/ College of Engineering/Electronic Department/ Iraq/ Diyala
**Corresponding author E-mail: adham.hadi@yahoo.com*

Abstract

In this paper, we proposed a novel hybrid technique to Error Detection and Correction (EDAC) which is based on merging of two types of linear block codes: Hamming code and CRC (Cyclic Redundancy Check) at the same system. This technique is corrected all types of error by retransmitted or by Forward error correction (FEC). This technique is simply and achieves higher reliability, accuracy and security as compared with other similar methods. The system algorithms is designed and simulation using VHDL ((VHSIC (Very High Speed Integrated Circuit Hardware Description Language) to be implemented on FPGA kit (Field Programmable Gate Arrays) with Xilinx ISE 10.1 software program. The proposed system circuits have been designed, implemented, and corrects any types of error successfully.

Keywords: *EDAC CRC; Hamming Coding; FEC and VHDL.*

1. Introduction

In any digital communication systems, data is represented as a stream of sequence data bits, whenever bits flow from transmitter to receiver via transmission media; they are subject to unpredictable corrupted because of interference, thermal noise or any other type of noise. Mainly there are two general types of error single and burst error, a single bit error means a one bit changed from 0 to 1 or from 1 to 0, burst error means a group of bits are changed [1].
In computer science and telecommunication applications, and according to a huge development in communication systems, and massive amount of data that transmitted, data protective from noise, error control (error detection and correction) is necessary to deliver data bits over noise channel. Error detection process is a first step to Error correction, which is depended on adding extra bits to the original data. These redundancy bits are added at the transmitter and removed by the receiver. Their presence is allowed to detect and correct corrupted bits as shown in Fig.1. [1-2].

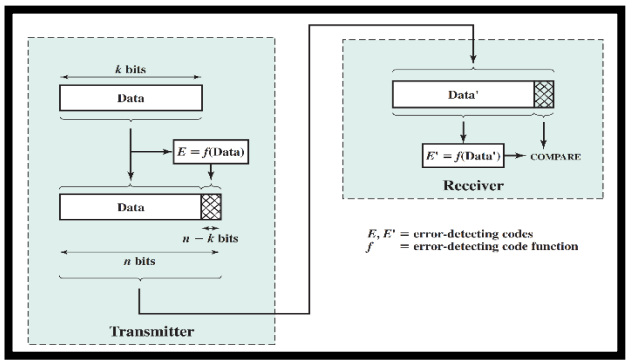


Fig. 1: Error-Detection Process [1].

Redundancy bits are achieved through two main coding schemes: convolution coding and block coding. In error correction, it is necessary to know the number of corrupted bits and their location in the received message. Error correction can be classified as an Automatic repeat request (ARQ) , Forward error correction (FEC),and Sometime ARQ and FEC can be combined, this method is called hybrid automatic repeat-request (HARQ) [1], [3], [4].
In this paper, we merged Hamming code and CRC technique, to ensure the original data arrived successfully, using hybrid automatic repeat-request (HARQ) by merged of Hamming Forward error correction (FEC) to single bits and Automatic repeat request (ARQ) by send an Acknowledgment to transmitter to resending data where system design using VHDL to implementation it at progressive steps.
As compared with other published papers in this field, there are many systems designed to detected and corrected errors ,but with a separated technique or with other correction code, such as error detection and correction using CRC technique only [5 - 9], or error Detection and Correction using Hamming technique only [10 - 16], and many other works. These papers designed CRC and Hamming as separate system. In this paper the system designed by merging of CRC code and Hamming code in one system using VHDL, with increasing data input to get high data rate and high reliability.

2. Proposed method

The proposed system circuits which shown in Fig. 2, have an input data with 32 bits (4 byte), these bits are coding at the transmitter circuit with a combination code based on (CRC and Hamming) coding which add and appending 8 bits at CRC circuit, then data passed to hamming circuit and produced an output data with 46 bits, at the receiver side the incoming data bits are decoding, and remove the redundancy bits added at the transmitter circuits. Hamming decoding circuit is correct all single error bits and cancelled all extra bits added at the Hamming encoder circuit, while CRC decoder circuit detects all other types of error and sending an acknowledgment to

Copyright © 2018 Adham Hadi Saleh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

transmitter to retransmitted data. This design will be increase the accuracy, performance and reliability of the system, the transmitter and receiver system designed using VHDL, so implemented using FPGA.

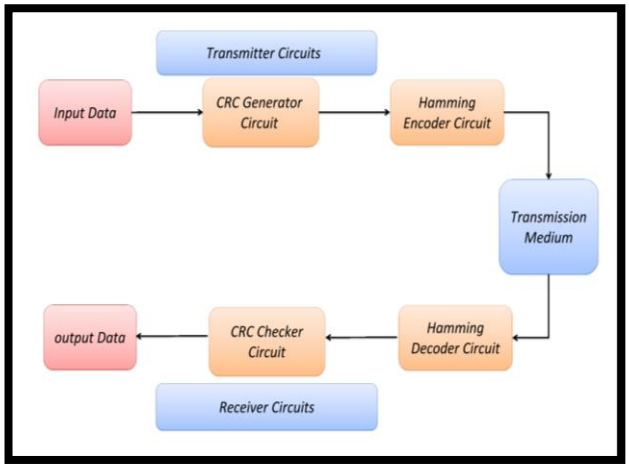


Fig. 2: A proposed System Circuits Components.

3. Transmitter circuit

3.1. CRC generator circuit

CRC technique is a simple and powerful technique to detected and corrected error, which is based on binary division and different from other methods that are based on parity check. The Cyclic Redundancy Check depends on the division remainder that is added to transmitted data. The basic block diagram and general design of CRC Generator -Checker is shown in Fig. 3[2].

In this paper the CRC Generator and Checker circuits are designed with CRC-8 divisor polynomial ($X^8 + X^2 + X + 1$) that equal to (107 HEX) that is used at Asynchronous Transfer Mode (ATM) header. The CRC method -Based framing re-using the header (CRC) is improving the efficiency of a pre-standard (ATM) protocol links, which is present in ATM and other similar protocols to provide framing with no overhead adding on the link. In ATM, this is known as the Header Error Control/Check (HEC) field [17], [18].

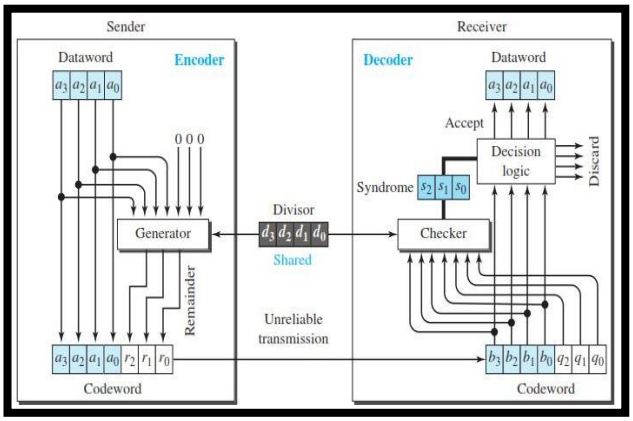


Fig. 3: CRC Generator and Checker [3], [8].

3.2. Hamming encoder circuit

Error-detection schemes can be either systematic or non-systematic: In a systematic scheme the transmitter sends the original data and attaches a fixed number of check bits. That is derived from the data bits by some deterministic algorithm. If only error detection is required a receiver can simply apply the same algorithm to the received data bits and compare its output with the received check bits if the values do not match an error has occurred at some point during the transmission. In a system that uses a non-systematic code the original message is transformed into an encoded message that has at least as many bits as the original message. Error correction & detection Hamming code may perform using even parity or odd parity [7], [8].

Hamming code is one of the most important and famous Binary Block coding method which is used to error detection and correction method based on add extra bits to the original data, that redundancy bits are used by the receiver to check consistency of incoming message, the redundancy bits is added to the original data according to an even or odd parity checking method to generated redundancy bit. The redundancy bits number are depended on information data bits, as shown in equation below [11], [12]:

$$2^r \geq m + r + 1 \quad (1)$$

Where r is redundancy bits and m is data bits, so to transmitted 40 bits (original data input (32) bits and CRC remainder (8) bits) according to equation (1) we need (6) redundancy bits (R1, R2, R4, R8, R16 and R32), these redundancy bits based on even parity check which are grouped as showing in equations below:

R(1)=datain(1) XOR datain(2) XOR datain(4) XOR datain(5) XOR datain(7) XOR datain(9) XOR datain(11) XOR datain(12) XOR datain(14) XOR datain(16) XOR datain(18) XOR datain(20) XOR datain(22) XOR datain(24) XOR datain(26) XOR datain(27) XORdatain(29) XOR datain(31) XORdatain(33) XOR datain(35) XOR datain(37) XOR datain(39) (2)

R(2)=datain(1) XOR datain(3) XOR datain(4) XOR datain(6) XOR datain(7) XOR datain(10) XOR datain(11) XOR datain(13) XOR datain(14) XOR datain(17) XOR datain(18) XOR datain(21) XOR datain(22) XOR datain(25) XOR datain(26) XOR datain(28) XOR datain(29) XOR datain(32) XOR datain(33) XOR datain(36) XOR datain(37) XOR datain(40) (3)

R(4)=datain(2) XOR datain(3) XOR datain(4) XOR datain(8) XOR datain(9) XOR datain(10) XOR datain(11) XOR datain(15) XOR datain(16)XOR datain(17) XOR datain(18)XOR datain(23) XOR datain(24)XOR datain(25)XOR datain(26) XOR datain(30) XOR datain(31) XOR datain(32) XOR datain(33) XOR datain(38) XOR datain(39) XOR datain(40) (4)

R(8) = datain(5) XOR datain(6) XOR datain(7) XOR datain(8) XOR datain(9) XOR datain(10) XOR datain(11) XOR datain(19) XOR datain(20) XOR datain(21) XOR datain(22) XOR datain(23) XOR datain(24) XOR datain(25) XOR datain(26) XOR datain(34) XOR datain(35)XOR datain(36) XOR datain(37) XOR datain(38) XOR datain(39) XOR datain(40) (5)

R(16) =datain(12)XOR datain(13)XOR datain(14)XOR datain(15) XOR datain(16)XOR datain(17)XOR datain(18)XOR datain(19) XOR datain(20)XOR datain(21) XOR datain(22)XOR datain(23) XOR datain(24)XOR datain(25)XOR datain(26) (6)

R(32) =datain(27)XORdatain(28)XOR datain(29) XOR datain(30) XOR datain(31) XOR datain(32) XOR datain(33)XOR datain(34) XORdatain(35) XORdatain(36) XORdatain(37) XORdatain(38) XOR datain(39) XOR datain(40) (7)

The transmitter block diagram is shown in Fig. 4, and it's pins desecration is explained in Table (1).

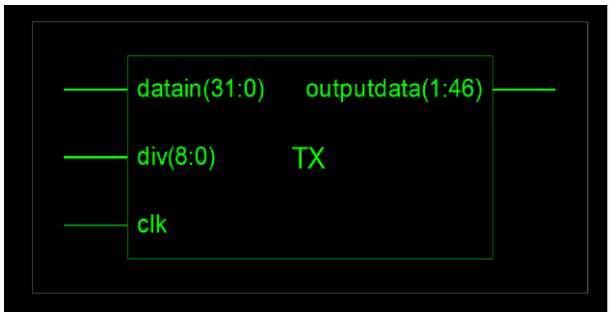


Fig. 4: Transmitter Block Diagram.

Table 1: Pins Description of the Transmitter Circuit

Signal	Direction	Size	Description
Clk	Input	1	Clock signal that clocks all internal component.
Div	Input	9	Input divisor data represented in binary bit stream.
Datain	Input	32	Input data represented in binary bit stream.
Output-data	Output	46	Output data of the transmitter circuit

The summary of the transmitter circuit using Spartan 3A and Spartan 3AN FPGA Kit is shown in Table (2) , and to testing the transmitter circuit operation assume a random data input is passed through it, as shown in time simulation(Fig. 5,6,7 and 8) , and explained in Table (3).

Table 2: Summary of the Transmitter Circuit Using Spartan 3A / 3AN FPGA Kit

Logic Utilization	Used	Available	Utilization
number of slices registers	32	19200	0%
Number of Slices LUTs	260	19200	1%
Number of fully used Bit Slices	32	260	12%
Number of bonded IOBs	79	220	35%
Number of BUFG/BUFGCTRLs	1	32	3%

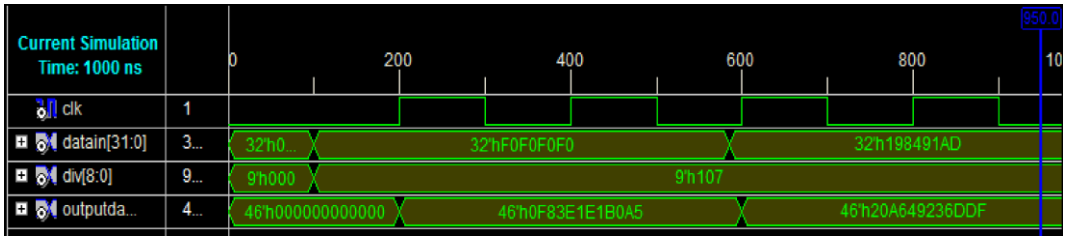


Fig. 5: Time Simulation with Input Data (Hex F0F0F0F0 and Hex 198491AD).

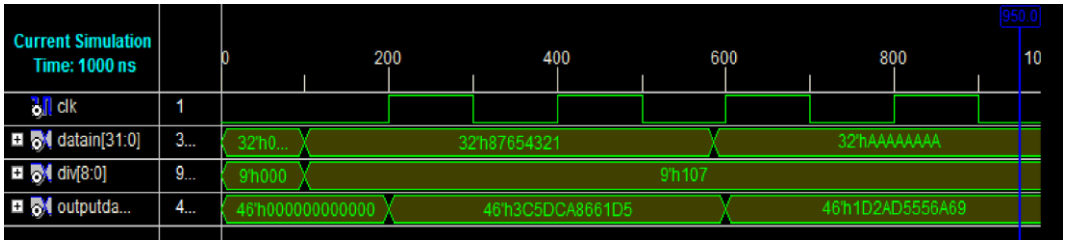


Fig. 6: Time Simulation with Input Data (Hex 87654321 and Hex AAAAAAAA).

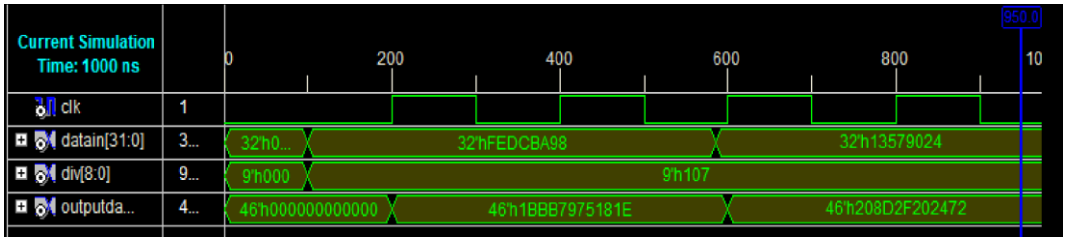


Fig. 7: Time Simulation with Input Data (Hex FEDCBA98 and Hex 13579024).

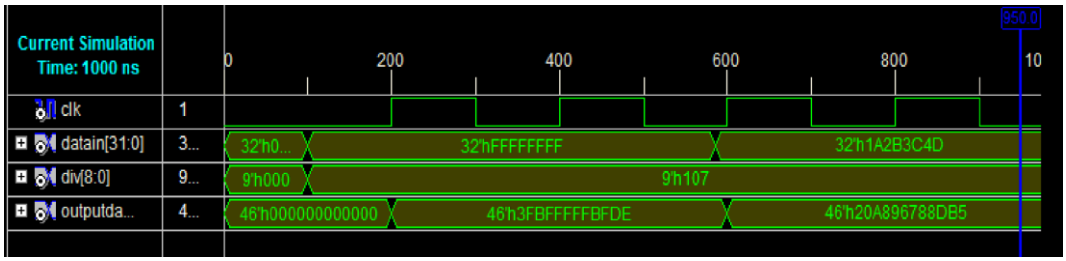


Fig. 8: Time Simulation with Input Data (Hex FFFFFFFF and Hex 1A2B3C4D).

Table 3: Transmitter Circuits Output Data

Random Encoding Data (32bits)	Transmitter Output Data (46 Bits)
Hex 87654321	Hex3C5DCA8661D5
Hex AAAAAAAA	Hex1D2AD5556A69
Hex FFFFFFFF	Hex3FBFFFFFBFDE
Hex1A2B3C4D	Hex20A896788DB5
HexF0F0F0F0	Hex0F83E1E1B0A5
Hex198491AD	Hex20A649236DDF
Hex FEDCBA98	Hex1BBB7975181E
Hex13579024	Hex208D2F202472

4. Receiver circuit

The receiver circuit is consisting of two merged circuit which is hamming decoder circuit and CRC checker circuit.

4.1. Hamming decoder circuit

Hamming decoder circuit is received message that consist of (46 bits) at receiver side, then it is detected and corrected all single error bits, and remove the redundancy bits that added at the encoder circuit.
Hamming decoder circuit detect the single error by Exorings of received data and determine a position of an error bit, so correct it using a NOT gate, the receiver is removed the redundancy bit and forward the data to the next circuit. The position of error bit is calculated as shown in equations below:

$$P(2^0)=DATA(1)\oplus DATA(3)\oplus DATA(5)\oplus DATA(7)\oplus DATA(9)\oplus DATA(11)\oplus DATA(13)\oplus DATA(15)\oplus DATA(17)\oplus DATA(19)\oplus DATA(21)\oplus DATA(23)\oplus DATA(25)\oplus DATA(27)\oplus DATA(29)\oplus DATA(31)\oplus DATA(33)\oplus DATA(35)\oplus DATA(37)\oplus DATA(39)\oplus DATA(41)\oplus DATA(43)\oplus DATA(45) \quad (8)$$

$$P(2^1)=DATA(2)\oplus DATA(3)\oplus DATA(6)\oplus DATA(7)\oplus DATA(10)\oplus DATA(11)\oplus DATA(14)\oplus DATA(15)\oplus DATA(18)\oplus DATA(19)\oplus DATA(22)\oplus DATA(23)\oplus DATA(26)\oplus DATA(27)\oplus DATA(30)\oplus DATA(31)\oplus DATA(34)\oplus DATA(35)\oplus DATA(38)\oplus DATA(39)\oplus DATA(42)\oplus DATA(43)\oplus DATA(46) \quad (9)$$

$$P(2^2)=DATA(4)\oplus DATA(5)\oplus DATA(6)\oplus DATA(7)\oplus DATA(12)\oplus DATA(13)\oplus DATA(14)\oplus DATA(15)\oplus DATA(20)\oplus DATA(21)\oplus DATA(22)\oplus DATA(23)\oplus DATA(28)\oplus DATA(29)\oplus DATA(30)\oplus DATA(31)\oplus DATA(36)\oplus DATA(37)\oplus DATA(38)\oplus DATA(39)\oplus DATA(44)\oplus DATA(45)\oplus DATA(46) \quad (10)$$

$$P(2^3) = DATA(8)\oplus DATA(9)\oplus DATA(10)\oplus DATA(11)\oplus DATA(12)\oplus DATA(13)\oplus DATA(14)\oplus DATA(15)\oplus DATA(24)\oplus DATA(25)\oplus DATA(26)\oplus DATA(27)\oplus DATA(28)\oplus DATA(29)\oplus DATA(30)\oplus DATA(31)\oplus DATA(40)\oplus DATA(41)\oplus DATA(42)\oplus DATA(43)\oplus DATA(44)\oplus DATA(45)\oplus DATA(46) \quad (11)$$

$$P(2^{4)} = DATA(16)\oplus DATA(17)\oplus DATA(18)\oplus DATA(19)\oplus DATA(20)\oplus DATA(21)\oplus DATA(22)\oplus DATA(23)\oplus DATA(24)\oplus DATA(25)\oplus DATA(26)\oplus DATA(27)\oplus DATA(28)\oplus DATA(29)\oplus DATA(30)\oplus DATA(31) \quad (12)$$

$$P(2^5) = DATA(32)\oplus DATA(33)\oplus DATA(34)\oplus DATA(35)\oplus DATA(36)\oplus DATA(37)\oplus DATA(38)\oplus DATA(39)\oplus DATA(40)\oplus DATA(41)\oplus DATA(42)\oplus DATA(43)\oplus DATA(44)\oplus DATA(45)\oplus DATA(46) \quad (13)$$

So after applying the last equations at the received data ,we can detected the error bit position by put the results of equations (8,9,10,11,12 and 13) with sequence ($2^5 2^4 2^3 2^2 2^1 2^0$) to get error position and corrected it.

4.2. CRC checker circuit

At CRC Checker circuit the incoming data (40 bits) is divided by divisor that is used at the transmitter system, to knows if the received data unit is correct or corrupted, if the remainder of division zero means arrived data unit is correct ,else it has been corrupted [2]. An error data frame is corrected by sending an acknowledgment to sender to retransmitted data frame. The block diagram of receiver circuit is shown in Fig .9, it's pins desecration is explained in Table (4), the summary of the receiver circuit using Spartan 3A and Spartan 3AN FPGA Kit is shown in Table (5).

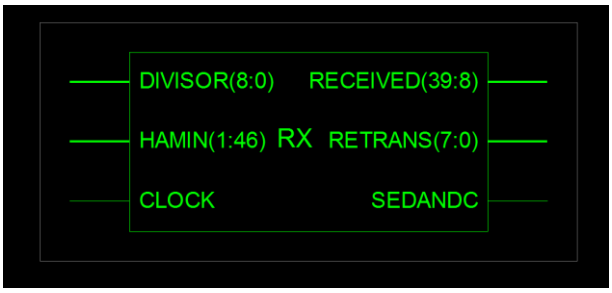


Fig. 9: Block Diagram of Received Circuit.

Table 4: Pins Description of the Transmitter Circuit

Signal	Direction	Size	Description
CLOCK	Input	1	Clock signal that triggered all internal components.
DIVISOR	Input	9	Input divisor (Hex107).
Hamin	Input	46	Received input data
RECEIVED	Output	32	Output data of error detection and correction system
RETRANS	Output	8	Acknowledgment signal sending to the transmitter system to retransmitted data, if it's corrupted, if output signal HEX00 the received data is correct otherwise it's corrupted and needs to retransmitted
SEDANDC	Output	1	Output signal means that a single error detected and corrected when it's state logic (1)

Table 5: Summary of the Receiver Circuit Using Spartan 3A/ 3AN FPGA Kit

Logic Utilization	Used	Available	Utilization
number of slices	292	704	41%
Number of Slices 4 inputs LUTs	534	1408	37%
Number of Slices flip flops	40	1408	2%
Number of bonded IOBs	95	108	87%
Number of BUFG/BUFGCTRLs	1	24	4%

So to check the system operation, we supposed a received random data input with different status single error, burst error and corrected data as explain in Table 6,and shown in time simulation in (Fig.10, Fig.11, Fig.12, Fig.13, Fig.14, Fig.15, Fig.16, and Fig.17).

Table 6: Receiver Circuits Output Data with Different Input States

Transmitter Circuit System		Receiver Circuit System		Single Error Detection And Correction (SEDANDC)	Burst Error Detection And Correction By Retransmitted (RETRANS)	Output Data Description
Input Data	Output Data	Input Data	Output Data			
Hex 87654321	Hex3C5DCA8661D5	Hex3C5DCA8661D5	Hex 87654321	0	Hex 00	Data Correct
Hex 87654321	Hex3C5DCA8661D5	Hex3C5DCA8661D4	Hex 87654321	1	Hex 00	Data Correct
Hex AAAAAAAA	Hex1D2AD5556A69	Hex1D2AD5556A69	Hex AAAAAAAA	0	Hex 00	Data Correct
Hex AAAAAAAA	Hex1D2AD5556A69	Hex1D2AD5556A65	Hex BAAAAAAA	1	Hex 16	Retransmitted
Hex FFFFFFFF	Hex3FBFFFFF-BFDE	Hex3FBFFFFF-BFDE	Hex FFFFFFFF	0	Hex 00	Data Correct
Hex FFFFFFFF	Hex3FBFFFFF-BFDE	Hex3FBFFFFFBFDF	Hex FFFFFFFF	1	Hex 00	Data Correct
Hex1A2B3C4D	Hex20A896788DB5	Hex20A896788DB5	Hex1A2B3C4D	0	Hex 00	Data Correct
Hex1A2B3C4D	Hex20A896788DB5	Hex20A896788DB1	Hex1A2B3C4D	1	Hex 00	Data Correct

HexF0F0F0F0	Hex0F83E1E1B0A5	Hex0F83E1E1B0A5	HexF0F0F0F0	0	Hex 00	Data Correct
Hex1A2B3C4D	Hex20A896788DB5	Hex0F83E101B0A5	HexF0F000F0	1	Hex 6C	Retransmitted
Hex198491AD	Hex20A649236DDF	Hex20A649236DDF	Hex198491AD	0	Hex 00	Data Correct
Hex198491AD	Hex20A649236DDF	Hex20A649A36DDF	Hex198491AD	1	Hex 00	Data Correct
Hex FEDCBA98	Hex1BBB7975181E	Hex1BBB7975181E	Hex FEDCBA98	0	Hex 00	Data Correct
Hex FEDCBA98	Hex1BBB7975181E	Hex1BBB79751B1E	Hex 7EDCBA98	1	Hex A8	Retransmitted
Hex13579024	Hex208D2F202472	Hex208D2F202472	Hex13579024	0	Hex 00	Data Correct
Hex13579024	Hex208D2F202472	Hex208D2F202471	Hex93579024	1	Hex 9E	Retransmitted

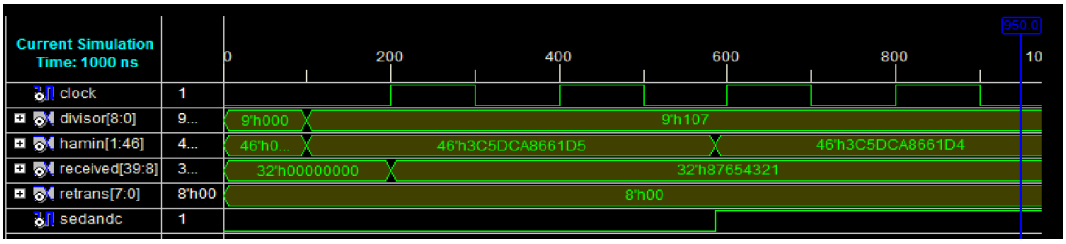


Fig. 10: Received Data (Hex3C5DCA8661D5 Correct and Hex3C5DCA8661D4 Corrupted).

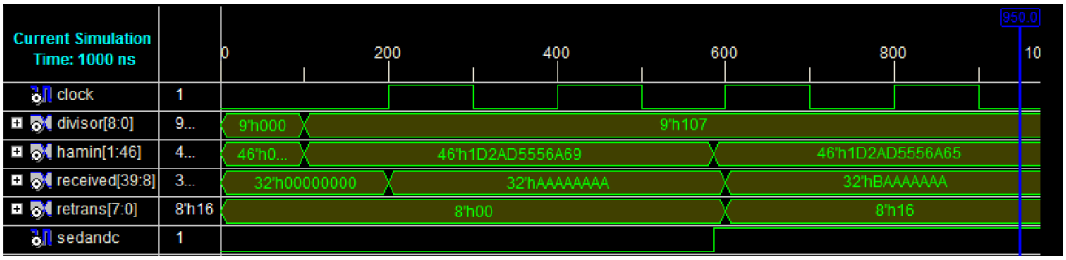


Fig. 11: Received Data (Hex1D2AD5556A69 Correct and Hex1D2AD5556A65 Corrupted).

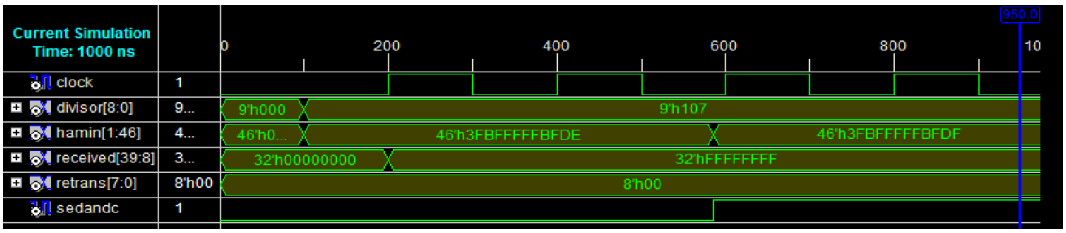


Fig. 12: Received Data (Hex3FBFFFFFBFDE Correct and Hex3FBFFFFFBFDF Corrupted).

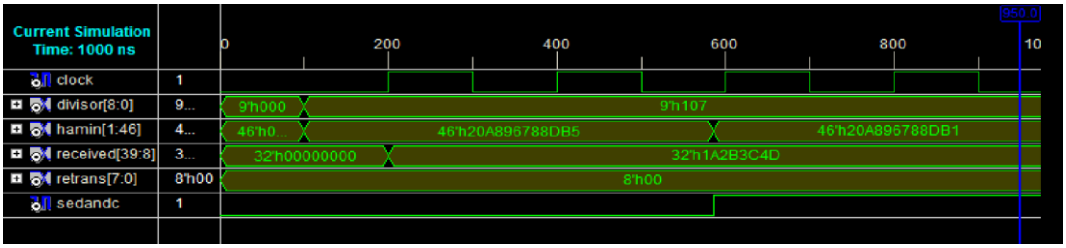


Fig. 13: Received Data (Hex20A896788DB5 Correct and Hex20A896788DB1 Corrupted).

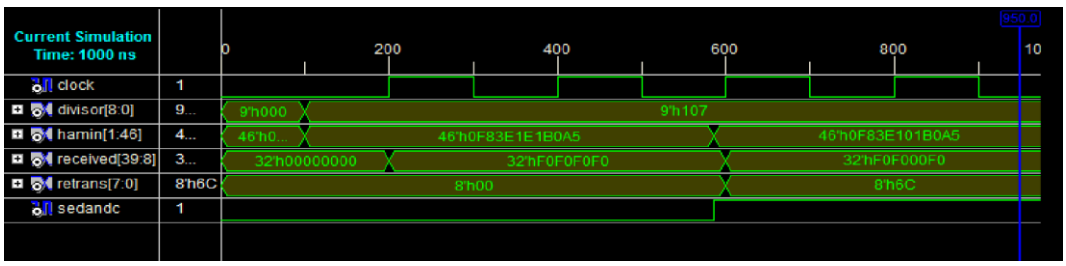


Fig. 14: Received Data (Hex0F83E1E1B0A5 Correct and Hex0F83E101B0A5 Corrupted).

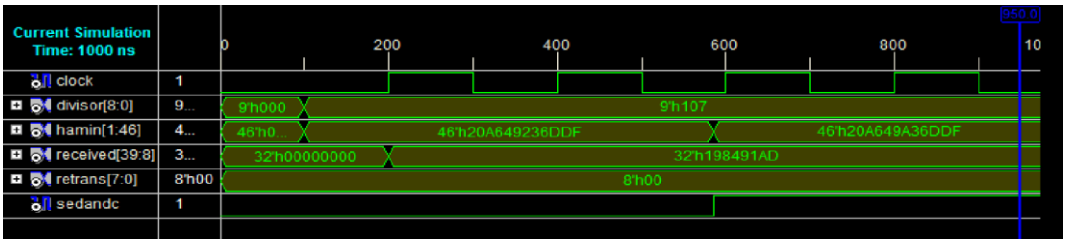


Fig. 15: Received Data (Hex20A649236DDF Correct and Hex20A649A36DDF Corrupted).

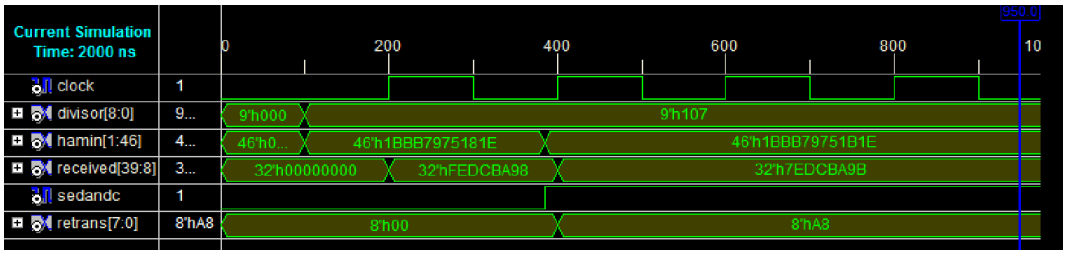


Fig. 16: Received Data (Hex1BBB7975181E Correct and Hex1BBB79751B1E Corrupted).

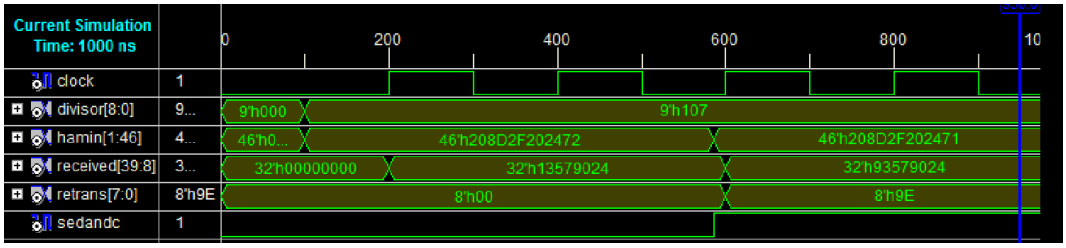


Fig. 17: Received Data (Hex208D2F202472 Correct and Hex208D2F202471 Corrupted).

5. Conclusion

This paper presents a hybrid error detection and correction method based on hamming coding and CRC. The encoding and decoding system are designed using VHDL, and implemented in Spartan 3A/3AN kit with Xilinx ISE 10.1 software program. This model is provide accurately, high reliability and ensure delivery as compared with other error detection and correction systems which detects all single error and corrected it using FEC, and all burst error using retransmitted of data by sending an acknowledgments to sender. The proposed circuits' show that a low hardware usage is required which make them suitable to be integrated with any other system with high date rate.

References

[1] William Stallings"Data and Computer Communications ", 10th edition, Pearson Education. Chapter 6, p.p 187-192.

[2] G.Shanthi , Dr.L.Padmasree"CHIPSCOPE Implementation of CRC circuit architecture"IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), Volume 2, Issue 1 (Mar. – Apr. 2013), PP 07-14.

[3] Forouzan, A. B. (2011)." Data Communications & Networking". McGraw-Hill Education. Chapter 10, p.p 282-285.

[4] Qusay Al-Doori and Omar Alani"A Multi Polynomial CRC Circuit for LTE-Advanced Communication Standard", 2015 seventh Computer Science and Electronic Engineering Conference (CEEC), University of Essex, UK.

[5] Shukla S. and Bergmann N. W., "Single bit error correction implementation in CRC-16 on FPGA", 1n...I EEE International Conference on Field- Programmable Technology. Brisbane, Australia, 2004.

[6] Wael M El-Medany,"FPGA Implementation of CRC with Error Correction", the Eighth International Conference on Wireless and Mobile Communications, 2012.

[7] M. Sprachmann, "Automatic generation of parallel CRC circuits", IEEE Des. Test Comput. vol. 18, no. 3, pp. 108–114, May/Jun. 2001. https://doi.org/10.1109/54.922807.

[8] Adham Hadi Saleh, Kamal Mohammed Saleh and Saad Al-Azawi "Design and Simulation of CRC Encoder and Decoder Using VHDL" first International Scientific Conference of Engineering, Sciences - 3rd Scientific Conference of Engineering Science (ISCES), IEEE, 2018.

[9] Rajesh Nair, Gerry Ryan and Farivar Farzaneh" A Symbol Based Algorithm for Hardware Implementation of Cyclic Redundancy Check (CRC)" Proceedings VHDL International Users' Forum. Fall Conference, IEEE, 1997.

[10] Tongsheng Zhang, Qun Ding" Design of (15, 11) Hamming Code Encoding and Decoding System Based on FPGA" International Conference on Instrumentation, Measurement, Computer, Communication and Control,IEEE, 2011 https://doi.org/10.1109/IMCCC.2011.179.

[11] Wirda Fitriani, Andysah Putera and Utama Siahaan " Single-Bit Parity Detection and Correction using Hamming Code 7-Bit Model",International Journal of Computer Applications, Volume 154 – No.2, November 2016.

[12] N. Shep and P. Bhagat, "Implementation of Hamming code using VLSI," International Journal of Engineering Trends and Technology, vol. 4, No. 2, pp. 186-190, 2013.

[13] Adham Hadi Saleh,"Design of Hamming code for 64 bit single error detection and correction using VHDL",Diyala Journal of Engineering Sciences, Vol. 08, No. 03, pp. 22-37, September 2015.

[14] Adham Hadi Saleh "Design of Hamming Encoder and Decoder Circuits For (64, 7) Code and (128, 8) Code Using VHDL" Journal of Scientific and Engineering Research,Vol(2),No(1), , pp. 1-15, 2015.

[15] Raymond Irudayaraj I., Abdul Lateef Haroon P.S, Ulaganathan J., Shridhar S. Bilagi "Design And Verification Of Improved Hamming Code (Ecc) Using Verilog",International Journal Of Electrical, Electronics And Data Communication, Volume-5, Issue-4, Aprl.-2017.

[16] Ravi Hosamani, Ashwini S. Karne"Design and Implementation of Hamming Code on FPGA using Verilog ", International Journal of Engineering and Advanced Technology (IJEAT), Volume-4 Issue-2, December 2014.

[17] M.Y. Rhee - "Error Correcting Coding Theory", McGraw-Hill, Singapore, 1989.

[18] Goresky, M. and Klapper, A.M. Fibonacci and Galois representations of feedback-with-carry shift registers, IEEE Transactions on Information Theory, Nov 2002, Volume: 48, On page(s): 2826 –2836.