Bulletin of Electrical Engineering and Informatics
Vol. 11, No. 3, June 2022, pp. 1460~1470
ISSN: 2302-9285, DOI: 10.11591/eei.v11i3.3778

FPGA-based experimental board for error control codes

Ahmed Abdulkadhim Hamad, Haider Al-Mumen, Mohammed Taih Gatte
Department of Electrical Engineering, College of Engineering, University of Babylon, Babylon, Iraq

Article Info

Article history:

Received Mar 9, 2022
Revised Apr 21, 2022
Accepted May 12, 2022

Keywords:

Cyclic redundancy check code
Experimental board
Field programmable gate array
Hamming code
Vivado high-level synthesis

ABSTRACT

This paper proposed an experimental device that emulates and facilitates teaching the theoretical concepts of error-control coding (ECC) techniques. Two prototypes laboratory boards were designed and implemented. The first board simulates the transmitter side of a typical digital communication system. It mainly contains a data source, (7,4) Hamming encoder, cyclic redundancy check encoder (CRC), and Gaussian noise generator. The second board has two types of Hamming decoders, a syndrome decoder, and a maximum likelihood (ML) decoder that accepts the received soft signals at the channel output. Each board has several control switches so that the trainee can change the code variables, noise power, user ID, in addition to display tools such as light-emitting diodes (LEDs), and test points for the oscilloscope that help the user to observe the results. Moreover, results and setting variables can also be displayed on a PC's screen connected through a USB port and organic light emitting diodes (OLED) display that is attached to the receiver board. The pipelining architecture of the field programmable gate array (FPGA) device was exploited in this proposed system to reduce the processing delay and hence increase the data throughput. Furthermore, we proved that both theoretical and experimental tests are identical.

Corresponding Author:

Ahmed Abdulkadhim Hamad
Department of Electrical Engineering, College of Engineering, University of Babylon
Hillah-Najaf Road, Babylon, Iraq
Email: eng.ahmed.ak@uobabylon.edu.iq

1. INTRODUCTION

Error control codes are crucial for detecting and correcting data errors in unreliable or noisy digital data transmission and storage systems [1]. The fundamental principle of error-control coding (ECC) is to add check digits to the message on the encoder, which may be utilized to detect and correct errors on the decoder. An encoding mechanism based on modern algebra is utilized to find the corresponding check digits for each input message block. For the code to be useful, it does not only have to cover the loss in energy as a result of adding check digits but it also must add gains to the uncoded system [1], [2]. Due to its emphasis on abstract algebra, it is not convincing to introduce ECC to undergraduate students without accompanying laboratory experiments. The experiments should present the aforementioned coding gain using various code parameters and noise levels. Currently, there is no experimental board that achieves these requirements. However, if exist, most universities in developing countries are unable to afford these laboratories and resort to virtual laboratories as proposed, for instance, by the authors of [3] and [4]. A java based virtual laboratory for data communication including (8, 4) extended Hamming is simulated [3]. Gilbert *et al*. [5] present the design and implementation of (7, 4) Hamming code, rate ½ convolutional code, and cyclic redundancy check code (CRC) code based on very high speed integrated circuits (VHSIC) hardware description language (VHDL) using Xilinx ISE 14.7 and Spartan- 6 field programmable gate array (FPGA) device. A real-time Wi-Fi laboratory platform for wireless communications education based on the LabVIEW platform is presented in [6]. One of

the experiments provided by this platform is the 64 state convolutional encoder with hard and soft Viterbi algorithms [7]. Garcia and Guzmán- Ramírez [8] introduce an FPGA-based experimental system for designing, implementing, and evaluating a real-time video processing scheme. Samy *et al*. [9] proposed a concatenation of bose-chaudhuri hocquenghem (BCH) [10] and convolutional code based on the Xilinx Spartan-3A/3AN FPGA starter kit. The study aims to reduce the error floor at a high signal-to-noise power ratio. To reduce the time to market, vivado high-level synthesis (HLS) is utilized by [11], [12] in the implementation of turbo codes [13] with different stopping criteria.

Despite the existence of researches literature on various implementations of digital communication systems in high descriptive language (HDL) [14]–[21] and LabVIEW [6] on FPGA devices, there is a dearth of such research in vivado HLS software [22]. This work introduces undergraduate students to a digital communication system using HLS software and FPGA platforms to give a hands-on experience for testing ECC techniques over practical channels. This work aims to: i) design of two experimental boards to evaluate the bit error rate (BER) and frame error rate (FER) of (7, 4) Hamming code and the CRC code. The first board represents the transmitter side, while the second represents the receiver side in a typical digital communication system, ii) design a prototype control panel for both transmitter and receiver to allow the trainee to adjust various codes and channel parameters, and iii) design of all digital electronic circuits that act as an interface between the FPGA and the control panel.

The rest of this paper is organized as follows; section 2 introduces the methodology of designing and implementing the required intellectual property (IP) for both transmitter and receiver boards. Sections 3 and 4 present the design concepts of constituent components comprising the transmitter and receiver boards. The assembly of the designed IPs in the previous two sections with Microblaze is described in section 5. The practical implementation of the prototype interface circuits and control panels was introduced in section 6. Section 7 shows the ability of HLS to test different directives like pipelining and loop unrolling over different solutions for the same project. Moreover, the BER and FER performance for uncoded and coded systems are tested practically to ensure that the experimental boards work properly. Section 8 concludes this paper.

2. IMPLEMENTATION METHODOLOGY

The implementation of both transmitter and receiver boards obey the following steps:

- Design of all required functional blocks like the pseudo-random binary source, Hamming encoder and decoder, CRC encoder and decoder, and Gaussian noise channel in a form of IP using C^{++} language via vivado HLS platform and store them at the repository of the vivado IP integrator.
- The IPs generated from step (a) are assembled and banded to the Microblaze processor through the advanced extensible interface (AXI) [23] in the vivado IP integrator. At this stage, other blocks should be added like memory interface generator (MIG), AXI uartlite, AXI quad SPI, and Pmod general-purpose input/output (GPIO) to make the board work properly.
- In the vivado IP integrator, one can perform simulation, RTL analysis, synthesis, implementation, and then generate a bitstream file.
- The generated bitstream file and all other required package files are exported from the vivado IP integrator to the SDK platform to program the Microblaze and FPGA.

2.1. Transmitter board

The proposed experimental panel for the transmitter board is shown in Figure 1. The board was designed to make sure that the trainee can adjust different code and channel parameters and can measure different signals using, for example, the oscilloscope or logic analyzer. The transmitter board mainly consists of the following parts:

2.1.1. Pseudo-random binary source

This part is responsible for generating four message bits $M = [M1 - M4]$ in each transmission session. This simulates the data generated from any physical source such as digital video, audio, or the output signal from digital sensors. To generate the message, a linear feedback shift register (LFSR) circuit depicted in Figure 2 was used. It can be represented by the following generator polynomial;

$$Gp(X) = X^{31} + X^{22} + X^2 + X^1 + 1 \quad (1)$$

2.1.2. Hamming encoder

The Hamming code family is a typical type of linear block code [1], [2], [24]–[26]. For this project, the number of parity check bits m is equal to 3, and the Hamming code has the following characteristics: i) codeword length $n = 2^m - 1 = 7$, ii) message length $k = 2^m - m - 1 = 4$, iii) error-correction capability

$t = 1$, and iv) the minimum distance between any two codewords $d_{min} = 3$. The parity check matrix H is formed in systematic form as $H = [P\ I_m]$, where the identity submatrix I_m is $m \times m$ square matrix and P is the coefficient submatrix that consists of k columns and is given by:

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{2,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \end{bmatrix} \tag{2}$$

where the values of $P_{i,j} \in \{0,1\}$ can be set by the trainee using switches as shown in Figure 1. This will define various code sets. The generator matrix for systematic linear block codes can be constructed using the expression, $G = [I_k\ P^T]$. The codeword vector C is generated from the message vector M as follows:

$$C = MG = [M1\ M2\ M3\ M4] \begin{bmatrix} 1 & 0 & 0 & 0 & P_{1,1} & P_{2,1} & P_{3,1} \\ 0 & 1 & 0 & 0 & P_{1,2} & P_{2,2} & P_{3,2} \\ 0 & 0 & 1 & 0 & P_{1,3} & P_{2,3} & P_{3,3} \\ 0 & 0 & 0 & 1 & P_{1,4} & P_{2,4} & P_{3,4} \end{bmatrix} \tag{3}$$

which gives,

$$\left.\begin{aligned} C1 &= M1 \\ C2 &= M2 \\ C3 &= M3 \\ C4 &= M4 \\ C5 &= M1 \cdot P_{1,1} + M2 \cdot P_{1,2} + M3 \cdot P_{1,3} + M4 \cdot P_{1,4} \\ C6 &= M1 \cdot P_{2,1} + M2 \cdot P_{2,2} + M3 \cdot P_{2,3} + M4 \cdot P_{2,4} \\ C7 &= M1 \cdot P_{3,1} + M2 \cdot P_{3,2} + M3 \cdot P_{3,3} + M4 \cdot P_{3,4} \end{aligned}\right\} \tag{4}$$

The dot (·) and plus (+) signs signify the AND and XOR logical operation respectively.

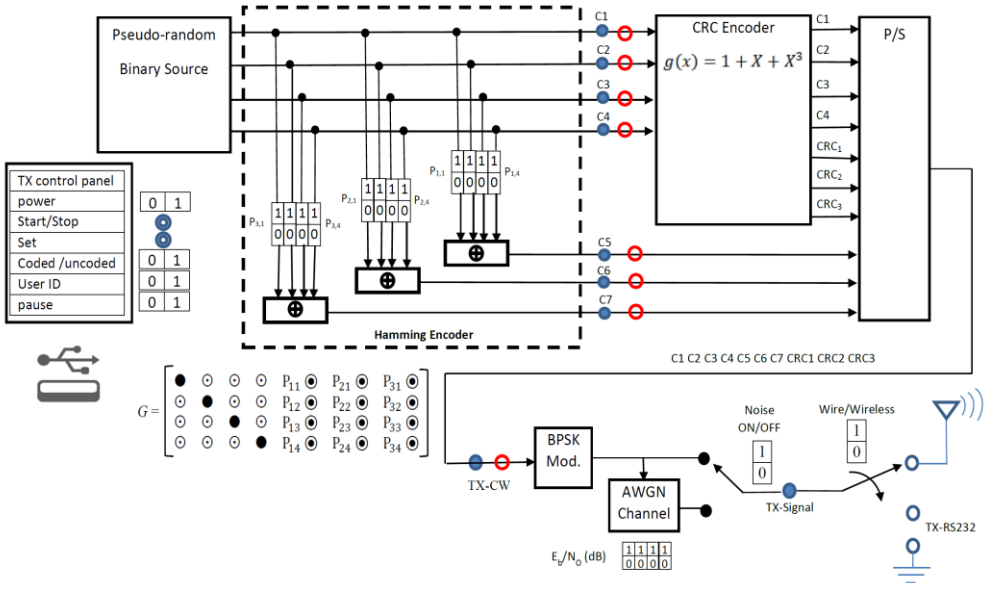


Figure 1. Transmitter experimental panel

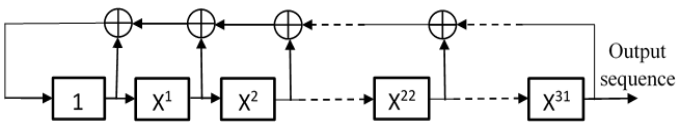


Figure 2. Pseudo-random binary generator

2.1.3. CRC encoder

A CRC is a cyclic code [27]–[29] where, a certain generator polynomial $g(x)$ is a factor in all the codeword polynomials. The cyclic code has two essential properties: i) if any two codewords are added together, the output is also a codeword in the code set (linearity property) and ii) any codeword in the code set can be generated by a cyclic shift of a certain codeword in the same set (cyclic property). The message sequence $M(X)$ is encoded into an (n, k) systematic cyclic code. That is, the bits of the message are sent in their original form, and the code polynomial $C(X)$ can be expressed as:

$$C(X) = B(X) + X^{n-k}M(X) \tag{5}$$

where $B(X)$ is the parity polynomial ($CRC1 + CRC2 \cdot X + CRC3 \cdot X^2$) and is given by:

$$B(X) = Rem\frac{X^{n-k}M(X)}{g(X)} \tag{6}$$

where Rem means the division reminder. The LFSR circuit with generator polynomial $g(X) = 1 + X + X^3$ that utilized in the encoding operations for generating a cyclic code as depicted in Figure 3.

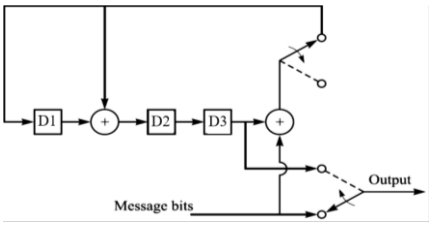


Figure 3. CRC encoder with $g(X) = 1 + X + X^3$

2.2. Receiver board

As in the transmitter board, the trainee can also adjust different code and channel parameters in the receiver board such as the decoding type (hard or soft decoding), the coefficient matrix, wire/wireless transmission, and adding deliberate errors. Figure 4 depicts the proposed receiver experimental panel which consists of several parts that will be exIn the same way that followed in the design of the transmitter was adopted in the receiver. plained in the following sections.

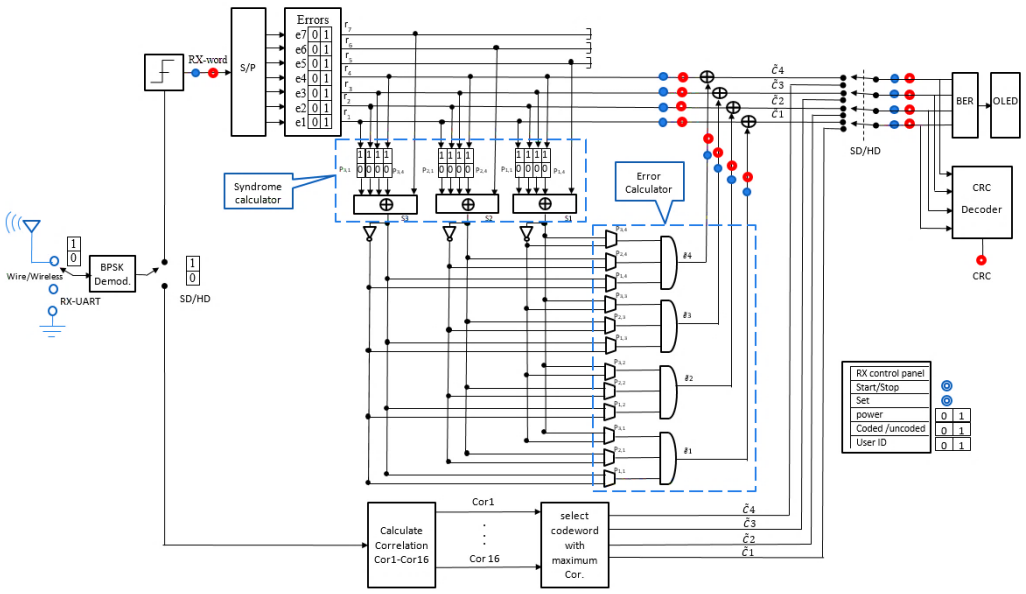


Figure 4. Receiver experimental panel

2.2.1. Syndrome decoder (hard decoder)

If a code vector C of size $1 \times n$ is sent over a noisy channel, then the received vector R is given by:

$$R = C + e \tag{7}$$

where e is the error vector $e = [e_1\ e_2 \ldots .\ e_n]$, and $e_i \in \{0,1\}$. The receiver board is equipped with a set of switches $e1 - e7$ (Figure 4) to allow the trainee to intentionally add errors to the received word. The decoding procedure for single errors consists of the following steps;

a. Compute the syndrome $S = RH^T$.
b. if $S = 0$ then no error and $R = C$
c. if $S \neq 0$ then:

– if $S = i^{th}$ row of H^T, then the i^{th} digit of R is in error and $C = R$ with the i^{th} digit changed.
– If S is not equal to any row of H^T then two or more errors have occurred and the procedure fails.

2.2.2. Maximum-likelihood decoder (soft decoder)

To demonstrate the benefits of adopting soft decoding compared to the hard decoding technique, the receiver board offers the maximum-likelihood algorithm [30] as soft decoding for Hamming code. Assume that the received coded signal is:

$$R = X + n \tag{8}$$

where $X = 2 \times C - 1$ is the modulated coded signal, and n is i.i.d Gaussian noise with zero mean and variance $\sigma^2 = N_o/2$. Where N_o is the two-sided power spectral density. The decoding principle is based on finding the correlation Cor_i between the received word R and all the 2^k possible codewords $X_i, i = 1\ to\ 2^k$, in the code set. The codeword which has maximum correlation is selected as the possible transmitted codeword.

2.2.3. CRC decoder

The CRC decoder ensures that the decoded message word is error-free after it leaves the Hamming decoder. If the Hamming decoder fails to decode correctly, a CRC light-emitting diode (LED) indicator is lit up. Figure 5 represents the digital circuit of the CRC decoder that determines the syndrome;

$$S(X) = r(X)\ mod\ g(X) \tag{9}$$

where, $r(X) = CRC1 + CRC2 \cdot X + CRC3 \cdot X^2 + \tilde{C}1 \cdot X^3 + \tilde{C}2 \cdot X^4 + \tilde{C}3 \cdot X^5 + \tilde{C}4 \cdot X^6$, and $\tilde{C}1 - \tilde{C}4$ are the decoded message bits. If $S(X) = 0$, then it is declared that the decoded message is error-free, else it is in error.

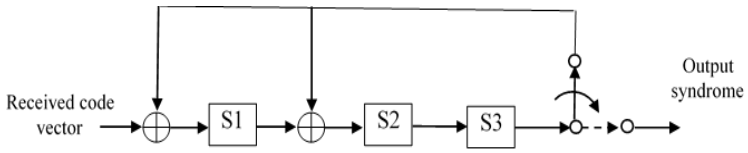


Figure 5. CRC decoder circuit

3. THE INTEGRATED IP BLOCKS

The IPs generated in vivado HLS are sent to the vivado IP integrator repository after being tested using testbench code, which is also written in C++. The IPs that comprise the transmitter circuit are assembled and linked together with the Microblaze processor through the AXI interface as in Figure 6. Several IP blocks provided by Xilinx or other partners like memory interface generator (MIG), processor system reset, Microblaze debug module (MDM), AXI interconnect, clocking wizard, quad SPI flash memory, PmodGPIO, and AXI UARTlite are essential to ensure that the system is working properly. Figure 7 shows the schematic diagram of the receiver circuit.

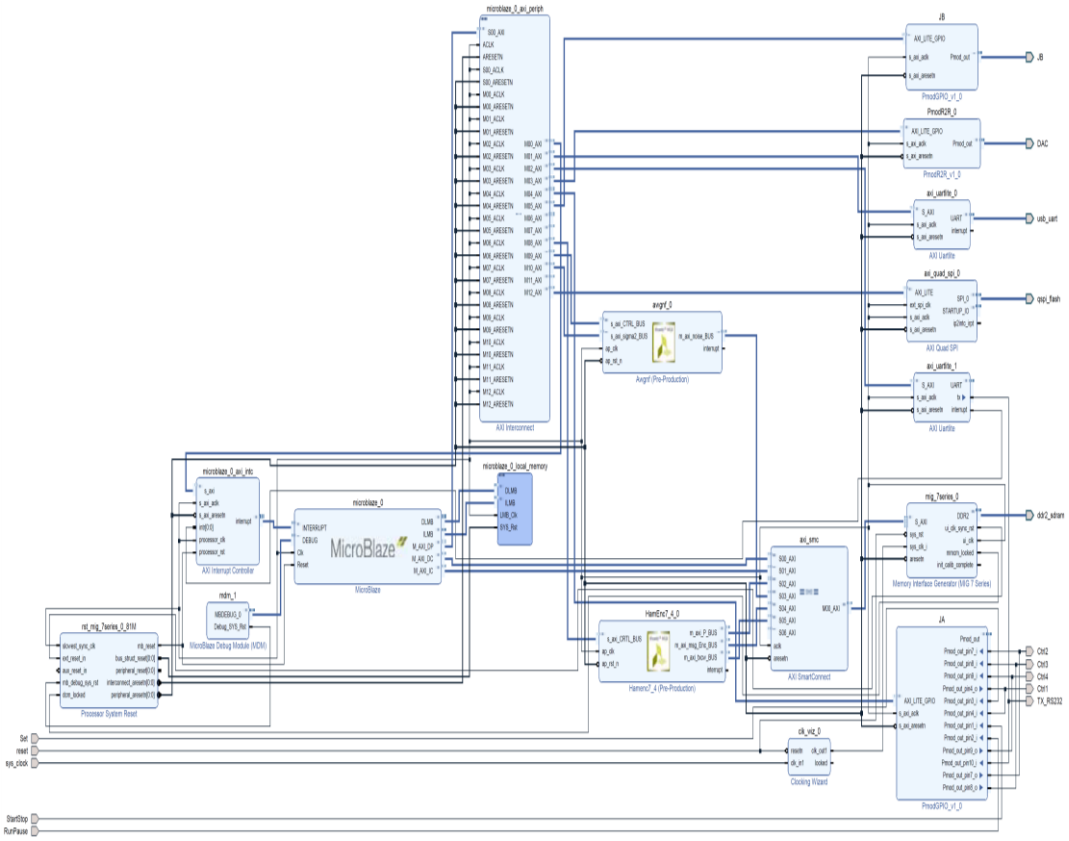


Figure 6. Transmitter IP schematic diagram

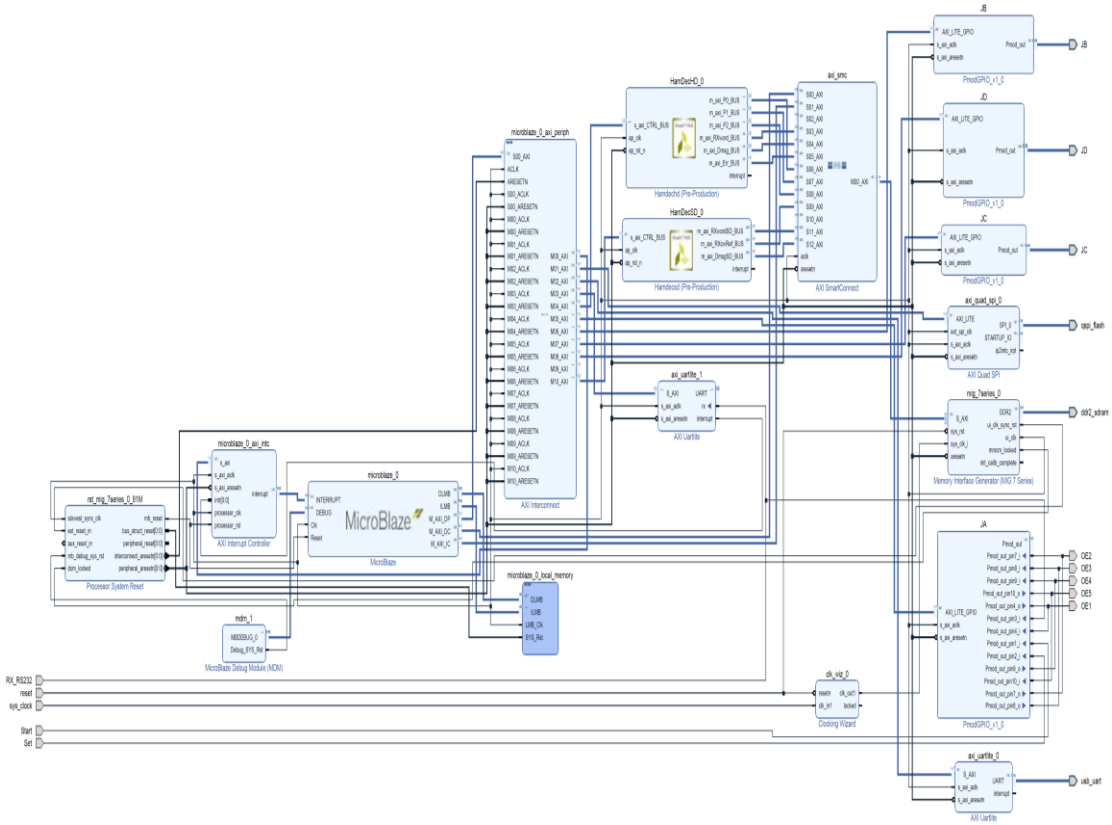
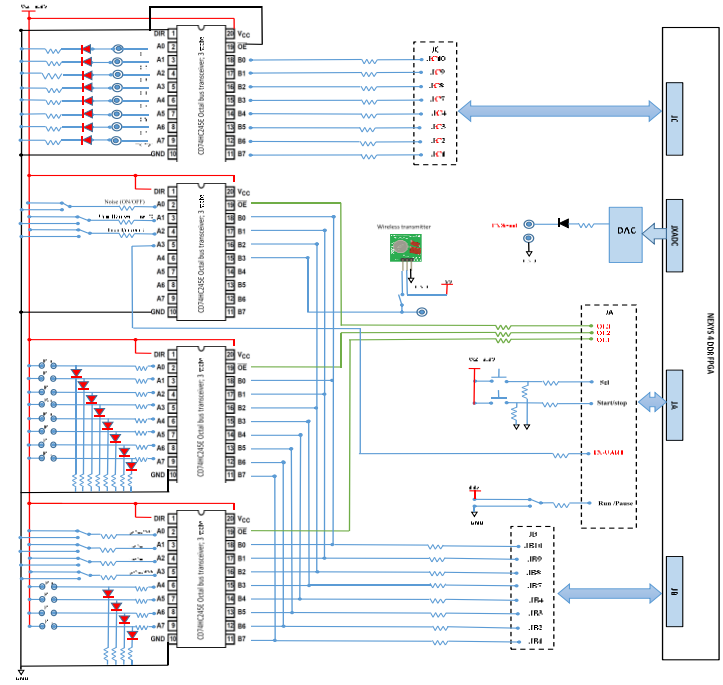


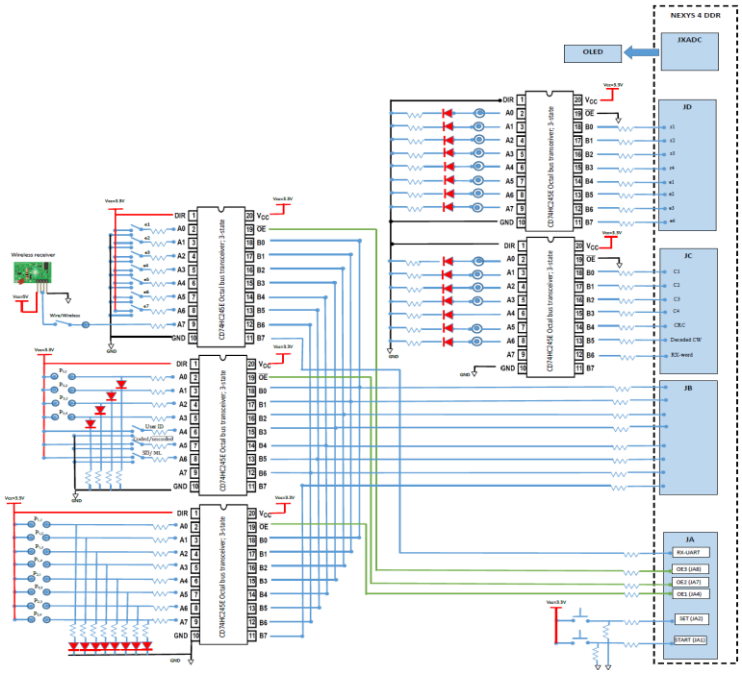
Figure 7. Receiver IP schematic diagram

3.1. Practical implementation

Two interfacing circuits were implemented to connect the FPGA board with the control panel of the transmitter and receiver. The aforementioned circuits address the shortage of the available GPIO ports in the FPGA board as well as serve to protect the FPGA from accident trainee mistakes. This can be done by using a common bus for various input/output signals. The 74HC245 IC, which is a high-speed octal tri-state bidirectional transceiver is utilized for signal multiplexing. Figure 8(a) and (b) show the interface circuits for the transmitter and receiver, respectively. To make sure that our design works properly, a prototype circuit for the transmitter and receiver boards was implemented and evaluated. Figure 9(a) and (b) show the photographic images of the transmitter and receiver boards, respectively.

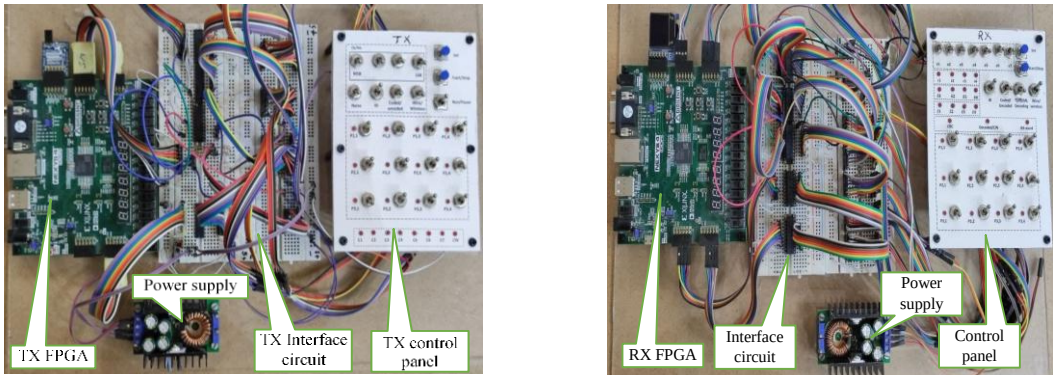


(a)



(b)

Figure 8. Interface circuits for the (a) transmitter and (b) receiver



(a) (b)

Figure 9. Prototype circuit for the (a) transmitter and (b) receiver boards

4. RESULTS AND DISCUSSION

This project was implemented using Nexys 4 demobilization and reintegration (DDR) development platform based on Artix-7 (Xilinx part number XC7A100T-1CSG324C) FPGA device.

4.1. Latency and utilization estimate

The HLS provides the ability to test different directives like Pipelining, data flow, and loop unrolling over different solutions. This can be done for the same project and one can compromise between the achieved throughput and resource utilization. Table 1 shows the latency and utilization estimate for the Hamming encoder. It is noted from Table 1 that using the aforementioned directives in solution 2 reduces the latency and utilization area significantly. The utilization estimate for the transmitter circuit presented in Figure 6 and provided by the vivado IP integrator is given in Figure 10.

Table 1. Latency and utilization estimate for the Hamming encoder

Latency (clock cycles)				Utilization estimation		
		Solution1	Solution2		Solution1	Solution2
latency	min	72	25	BRAM_18K	6	6
	max	72	25	DSP48E	0	0
interval	min	72	12	FF	3130	2908
	max	72	12	LUT	5096	3643

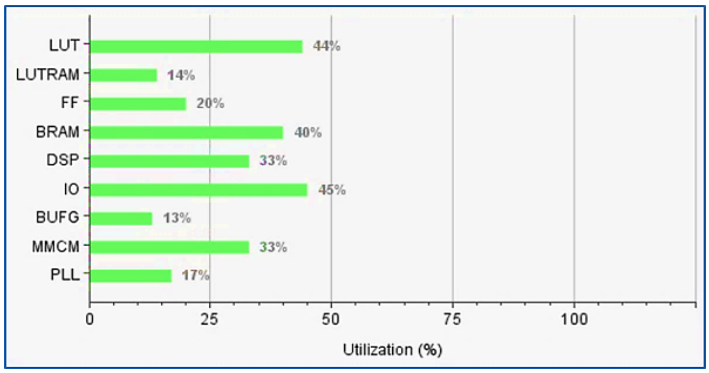


Figure 10. The utilization estimate for the transmitter circuit

4.2. BER performance estimation

The transmitter board can be connected to the receiver board using a wire/wireless channel. The trainee can adjust the boards to work in coded and uncoded schemes. For each scheme, the value of E_o/N_0 can be adjusted between 0-15 dB in a step of 1 dB using four switches (Figure 1), where E_o is the energy per bit. Figure 11 shows the BER performance comparison between uncoded, ML decoding, and syndrome decoding against E_o/N_0 over wire channel. The performance of the ML decoder outperforms the uncoded system as well

as the system that utilized syndrome decoding. The BER and FER results of the experiment can be noted on the computer monitor and the OLED display as shown in Figure 12(a) and (b), respectively.

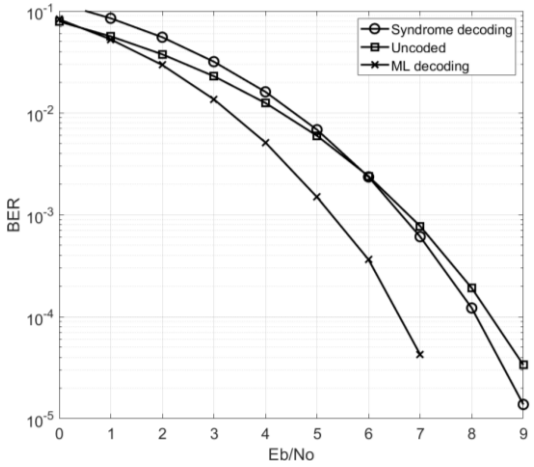


Figure 11. BER performance comparison between uncoded, ML decoding, and syndrome decoding

Frame# 16	ID# 1	BER=0.0312500	FER=0.1250000
Frame# 17	ID# 1	BER=0.0294117	FER=0.1176470
Frame# 18	ID# 1	BER=0.0277777	FER=0.1111111
Frame# 19	ID# 1	BER=0.0263157	FER=0.1052631
Frame# 20	ID# 1	BER=0.0250000	FER=0.1000000

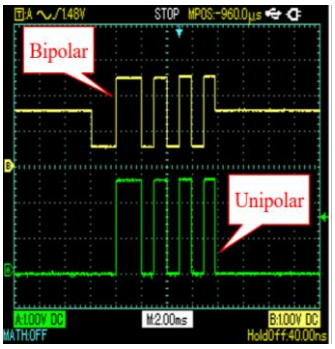
(a)

(b)

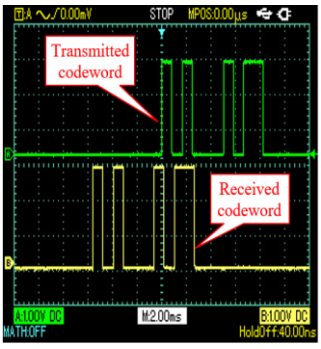
Figure 12. The BER and FER performance displayed on (a) computer monitor and (b) OLED display

4.3. Waveforms observation

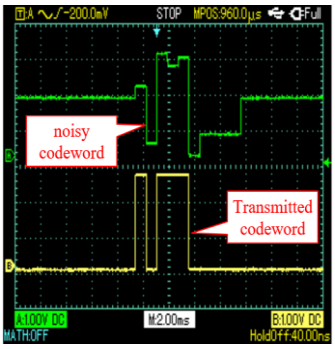
The experimental boards allow the user to test different digital and analog signals through a banana socket and LEDs indicator. Figure 13 shows the observation of different waveforms. The same transmitted codeword in unipolar and bipolar format is shown in Figure 13(a). Figure 13(b) depicts the similarity of the transmitted and received codeword when no noise is added. The same figure also shows the delay between the two signals as a result of processing delay. The waveforms of transmitted and received codeword with noise are shown in Figure 13(c).



(a)



(b)



(c)

Figure 13. Waveforms for (a) unipolar and bipolar, (b) transmitted and received codeword without noise, and (c) transmitted and received codeword with noise

5. CONCLUSION

This work introduces a simple methodology to design and implement two experimental laboratory boards to emulate a transmitter and receiver in a typical digital communication system. The objective of this project is to provide students with a hands-on experience with ECC by implementing Hamming and CRC encoders and decoders. To reduce the time to market, all the essential IPs comprising the proposed system were designed in C^{++} language using vivado HLS software instead of HDL. The system implementation was done using Nexys 4 DDR FPGA platform. The proposed experimental boards assist to confirm the interconnection between theoretical concepts and practical aspects for undergraduate and graduate students taking courses in information theory and error control coding. This project could be extended in the future to include other codes, for instance, convolutional, BCH, Reed-Solomon, turbo codes, and which all have superior performance to Hamming codes but with higher decoder complexity.

REFERENCES

[1] S. Lin and D. J. Costello, *Error Control Coding, Second Edition*. USA: Prentice-Hall, Inc., 2004.

[2] T. K. Moon, *Error correction coding: mathematical methods and algorithms*. John Wiley & Sons, 2020.

[3] O. Okoyeigbo, E. Agboje, E. Omuabor, A. Uyi, and A. Orimogunje, "Design and implementation of a java based virtual laboratory for data communication simulation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 6, pp. 5883–5890, Dec. 2020, doi: 10.11591/ijece.v10i6.pp5883-5890.

[4] R. Hosamani and A. S. Karne, "Design and implementation of hamming code on FPGA using Verilog," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 4, no. 2, pp. 180–184, Dec. 2014,

[5] J. M. Gilbert, C. Robbins, and W. Sheikh, "FPGA implementation of error control codes in VHDL: An undergraduate research project," *Computer Applications in Engineering Education*, vol. 27, no. 5, pp. 1073–1086, Jul. 2019, doi: 10.1002/cae.22137.

[6] K. Küçük, "RTWiFi- Lab: A real- time Wi- Fi laboratory platform on USRP and LabVIEW for wireless communications education and research," *Computer Applications in Engineering Education*, vol. 26, no. 1, pp. 111–124, 2018, doi: 10.1002/cae.21865.

[7] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," in *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967, doi: 10.1109/TIT.1967.1054010.

[8] I. Garcia and E. Guzmán- Ramírez, "A FPGA- based experimentation system for designing, implementing, and evaluating real-time video processing and analysis algorithms at undergraduate level," *Computer Applications in Engineering Education*, vol. 27, no. 2, pp. 387–405, 2019, doi: 10.1002/cae.22083.

[9] A. Samy, A. Y. Hassan, and H. M. Zakaria, "Improving bit error-rate based on adaptive Bose-Chaudhuri Hocquenghem concatenated with convolutional codes," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, no. 2, pp. 890–901, Aug. 2021, doi: 10.11591/ijeecs.v23.i2.pp890-901.

[10] A. Hocquenghem, "Codes correcteurs d'erreurs," *Chiffers*, vol. 2, pp. 147–156, 1959.

[11] R. Mahdi and A. A. Hamad, "Implementation of Efficient Stopping Criteria for Turbo Decoding," in *Journal of Physics: Conference Series*, vol. 1804, no. 1, p. 12015, Feb. 2021, doi: 10.1088/1742-6596/1804/1/012015.

[12] A. A. Hamad, H. F. Jaafar, and H. Al-libawy, "Efficient SOVA decoding and enhanced early termination mechanism based on new attenuation factors," *Telkomnika*, vol. 20, no. 2, pp. 268–278, Apr. 2022, doi: 10.12928/TELKOMNIKA.v20i2.23164.

[13] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," *Proceedings of ICC '93-IEEE International Conference on Communications*, vol. 2, pp. 1064–1070, 1993, doi: 10.1109/ICC.1993.397441.

[14] Z. Salcic and A. Smailagic, *Digital systems design and prototyping: using field programmable logic and hardware description languages*. Springer Science & Business Media, 2007.

[15] V. A. Pedroni, *Circuit design with VHDL*. MIT press, 2020.

[16] P. P. Chu, *FPGA prototyping by VHDL examples: Xilinx Spartan-3 version*. John Wiley & Sons, 2011.

[17] T. Zhang and Q. Ding, "Design of (15, 11) Hamming Code Encoding and Decoding System Based on FPGA," *2011 First International Conference on Instrumentation, Measurement, Computer, Communication and Control*, 2011, pp. 704–707, doi: 10.1109/IMCCC.2011.179.

[18] S. Y. Ameen, M. H. Al-Jammas, and A. S. Alenezi, "FPGA implementation of modified architecture for adaptive Viterbi decoder," *2011 Saudi International Electronics, Communications and Photonics Conference (SIECPC)*, 2011, pp. 1–9, doi: 10.1109/SIECPC.2011.5876958.

[19] S. Mishra and R. R. Tripathi, "VDHL Implementation of Viterbi Algorithm for Decoding of Convolutional Code," *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, 2015, pp. 1367–1370, doi: 10.1109/CICN.2015.265.

[20] W. Andre and O. Couillard, "Design and Implementation of a New Architecture of a Real-Time Reconfigurable Digital Modulator (DM) Into QPSK, 8-PSK, and 16-PSK on FPGA," *International Journal of Reconfigurable and Embedded Systems*, vol. 7, no. 3, pp. 173–185, Nov. 2018, doi: 10.11591/ijres.v7.i3.pp173-185.

[21] A. Hebibi, A. Bartil, and L. Ziet, "Comparison of two new methods for implementa BPSK modulator using FPGA," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 2, pp. 819–827, Aug. 2020, doi: 10.11591/ijeecs.v19.i2.pp819-827.

[22] Xilinx, "Vivado Design Suite User Guide: High-Level Synthesis (UG902)," *Xilinx, Inc*., [Online]. Available: 2019. https://docs.xilinx.com/v/u/2014.3-English/ug902-vivado-high-level-synthesis.

[23] Xilinx, "Vivado Design Suite: AXI reference guide," [Online]. Available: https://www.xilinx.com/content/dam/xilinx/support/documentation/ip_documentation/axi_ref_guide/latest/ug1037-vivado-axi-reference-guide.pdf.

[24] R. W. Hamming, "Error detecting and error correcting codes," in *The Bell System Technical Journal*, vol. 29, no. 2, pp. 147-160, April 1950, doi: 10.1002/j.1538-7305.1950.tb00463.x..

[25] I. F. Blake, "Error Control Coding (S. Lin and D. J. Costello; 2004) [book review]," in *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1616–1617, Apr. 2005, doi: 10.1109/TIT.2005.844056.

[26] S. Haykin, *Communication systems*. John Wiley & Sons, 2008.

[27] E. Prange, *Cyclic error-correcting codes in two symbols*. Air force Cambridge research center, 1957.
[28] E. Prange, "The Use of Coset Equivalene in the Analysis and Decoding of Group Codes," *AIR FORCE CAMBRIDGE RESEARCH LABS HANSCOM AFB MA*, 1959.
[29] J. Meggitt, "Error correcting codes and their implementation for data transmission systems," in *IRE Transactions on Information Theory*, vol. 7, no. 4, pp. 234–244, Oct. 1961, doi: 10.1109/TIT.1961.1057659.
[30] J. G. Proakis and M. Salehi, *Digital communications*, McGraw-hill New York, vol. 4, pp. 593–620, 2001.

BIOGRAPHIES OF AUTHORS

Ahmed Abdulkadhim Hamad received his B.Sc degree in Electrical and Electronic Engineering from the University of Technology, Baghdad, Iraq in 1991. The M.Sc degree in Communication engineering from the University of Technology, Baghdad, Iraq in 1995. Hamad received his Ph.D in communication engineering from the University of Technology, Baghdad, Iraq, in 2007. He is a faculty member at the University of Babylon, Babylon, Iraq since 2006. His research interests include digital communication, information theory, cryptography, and FPGA applications. He can be contacted at email: eng.ahmed.ak@uobabylon.edu.iq.

Haider Al-Mumen received his Ph.D degree in Electrical Engineering from Michigan State University, East Lansing, Michigan, USA in 2014. Before that, he received his M.Sc and B.Sc degrees in control and systems engineering from the University of Technology, Baghdad, Iraq in 1992 and 2000 respectively. Currently, he is a faculty member in the Department of Electrical Engineering at the University of Babylon, Babylon. Iraq. His research interests include, micro/nano devices fabrication, MEMS/NEMS, sensors technology, and graphene. He can be contacted on email: eng.almumenh@uobabylon.edu.iq.

Mohammed Taih Gatte received the BSc in Electrical and Electronic Engineering in 1993 and worked in the center of radar and communication researches/state of military research and development between 1993 and 2002 and then got the degree of M.Sc. in Computer engineering in 2005 and worked in the center of communication researches/ministry of science and technology, from 2005 to 2013. Then he got his PhD degree in Communication Engineering in 2017 and worked at the Babylon department of Science and Technology from 2017 to 2018. Currently, he is a teaching staff in the department of electrical engineering/University of Babylon. His research of interest is wireless communication, antenna, nanomaterials, nanotechnology, cryptography, FPGA application power & energy, and energy conservation. He can be contacted at email: mohtaih@gmail.com.