International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V71I3P227

Volume 71 Issue 3, 261-275, March 2023
© 2023 Seventh Sense Research Group®

Original Article

FPGA Implementation of Reconfigurable Modulation Scheme and Hamming Encoder for Cognitive Radio

M. A. Usha Rani[1], C. R. Prashanth[2]

[1, 2]Department of Electronics & Telecommunication Engineering, Dr.Ambedkar Institute of Technology, Karnataka, India.

[1]Corresponding Author : Usharanima.et@drait.edu.in

Received: 19 December 2022 Revised: 07 February 2023 Accepted: 12 March 2023 Published: 25 March 2023

Abstract *- Cognitive Radio (CR) is broadly accepted as a wireless communication technology because the CR minimizes the problem of bandwidth shortages by using idle spectrum. However, the development of the communication application of CR is vulnerable to noise and high resource consumption. This paper proposes the Reconfigurable Modulation Scheme (RMS) approach and the Hamming Encoder (HE) to improve communication over the CR. Instead of using Digital modulation techniques like QPSK, QAM and BPSK in the spectrum sensing, the RMS technique is used for less hardware utilization. These three modulation techniques (QPSK, QAM, and BPSK) are performed in a single RMS, whereas the HE is used to perform the error correction process. Here, the Field Programmable Gate Array (FPGA) based CR communication is developed RMS, and HE is developed by using the Xilinx ISE 14.2 software. The developed RMS-HE-FPGA architecture was also analysed by using a hardware development board, i.e., Spartan 6 FPGA. The performance of the RMS-HE-FPGA architecture is analyzed in terms of all aspects related to FPGA.*

Keywords *- Cognitive radio, Error correction process, Field programmable gate array, Hamming encoder, Hardware utilization, Reconfigurable modulation scheme.*

1. Introduction

CR is generally an advanced technology with extraordinary intelligence which improves the system size and spectrum flexibility of the wireless system [1]. The spectrum resources are effectively utilized by developing an effective approach, namely CR. The licensed bands are dynamically accessed by the unlicensed wireless users, i.e., Secondary Users (SUs). Since this dynamic access of the licensed bands is accomplished under the restriction of limited intrusion to the Primary Users (PUs) [2] [3] [4]. But CR is not permitted to create any disturbance to the PU, which has the priority for using the spectrum. Therefore, it is required to detect the PU absence before accomplishing the available spectrum access [5] [6]. The CR triggers the growth of next-generation wireless networks and is realized in the form of a self-motivated spectrum access radio. The cognitive cycle of the CR has three primary components such as spectrum sensing, cognition/management, and control action [7]. In that spectrum, sensing is considered a basic process of CR systems. This spectrum sensing is essential for spectrum sharing and dynamic spectrum management [8] [9]. The spectrum is effectively utilized by sensing the unoccupied wireless channels and allocating the transmission features to the unoccupied wireless channels [35,36]. An essential information is provided by applying the modulation technique to the spectrum sensing of the receiver at the electronic equipment [12]. The error detection of the PU signal makes the SU waste its own spectrum opportunity. Additionally, the shadow and multipath fading affect the signal features in typical wireless communication systems [13]. Specifically, the spectrum deficiency is caused because of the inappropriate usage of the fixed bandwidth. This kind of bandwidth utilization denoted the less usage of the licensed spectrum [14]. The conventional research fields in spectrum sensing face some issues with the arrival of CR. A few instances of these issues comprise noise robustness, real-time processing, lesser power consumption and detection efficiency. Hence, the conventional spectrum sensing methods do not satisfy the current requirements of CR [34].

The design methodology includes Reconfigurable modulation and flexible software/hardware system for implementing Cognitive radio on different FPGAs like Spartan 3, Artix 7 and Spartan 6. Wireless communication engineers can utilize the proposed system to implement it with available resources. It requires no additional hardware design for real-time transmission of information from the CR RMS-HE-FPGA transmitter to the CR RMS-HE-FPGA receiver. The software and hardware performance analysis shows that the expected result is obtained at the receiver without any error.

The main contributions of this research are given as follows:

- Reconfigurable modulation is developed instead of conventional modulation techniques for reducing the hardware required for communication.
- An encoding scheme, namely hamming encoder/decoder, is developed to lessen the errors occurred while transmitting the data bits.
- Further, the developed RMS-HE-FPGA is investigated in various FPGA devices such as Spartan-3A, Spartan -6 and Artix-7.

The rest of the paper is organized as follows: The related work about the existing FPGA-based modulation schemes and communications over the CR is given in section 2. The problem statement of this research is given in section 3. A detailed explanation of the RMS-HE-FPGA architecture is given in section 4. Further, the conclusion is made in section 5.

2. Literature Review

This section provides information about the existing FPGA-based modulation schemes and communications performed over the CR. Hemanth Kumar [16] presented the digital modulation approaches using FPGA for high-resolution communication applications. This work implemented the extensively used modulation techniques such as BPSK, QPSK, BFSK, and BASK—the developed modulation techniques used fewer digital blocks. However, the designed modulation techniques were not analyzed under any data transmission.

Rajalakshmi *et al*. [17] developed DSSS and FHSS modulation approaches to accomplish the execution of 3-phase Voltage-Source Rectifiers (VSR). These modulation approaches used the BPSK and QPSK techniques to avoid radio frequency interference. However, this work failed to investigate the hardware resource usage of the modulation techniques.

Sadiq *et al*. [18] presented the FPGA-based ASK and FSK techniques. The modulation has a simple architecture for generating sinusoidal waves with different phrases and frequencies. Subsequently, the frequency was used to identify whether it was an ASK or FSK signal on the demodulation side. The ASK and FSK were easily used in any condition because the designed modulation technique was reconfigurable. The LUT's index was varied in the modulation techniques due to the low frequency.

Shanigarapunareshkumar *et al.* [19] developed an Adaptive Absolute SCORE (AAS) approach to accomplish spectrum sensing in CR. Moreover, the Radix-8 and Carry Select Adder (CSLA) were used to implement the FIR filter, resulting in less complexity for the filter. The developed AAS-R8-CSLA has reduced the area in the overall design used to increase the spectrum sensing of CR. However, the QAM, BPSK, and QPSK modulation failed to design and operate simultaneously in the FPGA platform.

Rohit B. Chaurasiya *et al.* [20] designed a Maximum-Minimum-Eigenvalue (MME)-based spectrum sensing approach for the CR network. A hardware efficiency of this spectrum sensor structure was developed by sharing the resources. Moreover, the designed MM offered better performances under noise constraints. However, the communication performance was subjected to affect due by the noise.

Mohammad khayyeri *et al.* [21] developed a sparse spread spectrum sensing algorithm (CR4S) according to the sub-sampling solution. The free bandwidths with minimum calculations were detected by using the RF signal's real-valued features and sparsity of the frequency spectrum. Here, the Sparse Fast Fourier Transform (SFFT) was used to transform the time domain to the frequency domain, which enhanced the spectrum sensing and minimized the amount of computations. However, the operating frequency of the CR4S was less while accomplishing the spectrum sensing.

Sanker *et al.* [22] designed a Non-Contiguous Orthogonal Frequency Division Multiplexing (NC-OFDM) in CR to offer a higher data rate. Here, the QPSK was used to modulate the high-speed input data. An unused licensed band of RF spectrum was stored by simulating the FFT-based spectrum sensing approach. Subsequently, the sensed results acquired from the CR's dynamic operation were used to shape the RF spectrum. Therefore, the spectrum shaping of NC-OFDM was done according to the jamming threats, interference threats and channel impairments.

Murthy *et al.* [23] developed the Distributed Arithmetic (DA) based Residue Number System (RNS) for the filter. The developed DA-based residue calculation unit decreased the complexity in a wide range of modules. Moreover, the FIR filter with DA & RNS solved the tradeoff constraints according to the typical RNS system. However, this work's booth algorithm was inefficient for a smaller dynamic range.

Rahil Sharma *et al.* [24] designed VLSI architecture multicoset sampling on wideband spectrum sensing. The detection ranges obtained were from 0.9 to-5dB of SNR. The WSSR achieved a 90% shorter sensing time and 38.5% higher sensing bandwidth.

Tolga *et al.* [25] developed the model-based design with MATLAB & Simulink. The design was implemented on FPGA. The sampling frequency was 1MHz, broadband spectrum sensing was performed, and the band for communication was selected to obtain maximum spectrum efficiency.

Abderezzaq Bouhdieur *et al*. [26] presented Energy detection based on Autonomous global threshold adjustment in self-reconfigurable CR systems. The results obtained met the IEEE 802.22 standard with a probability of false alarm of less than 10% and a detection rate of more than 90%.

Qiang Li *et al*. [27] designed Reconfigurable Intelligent surface-based transmission and spatial modulation. To obtain high energy efficiency and high spectral efficiency, the cognitive radio, relay networks and WSNs carry dual information of both RIS and SM.

Rinu C Varghese [28] presented FIR bank of filter multiple non-uniform sharp transition width with less complexity for cognitive radio spectrum sensing for the Internet of Things Network. The proposed system incorporated less number of multipliers compared to others.

Mario Lopes Ferreira *et al*. [29] designed a baseband modulator, which was a reconfigurable FPGA-based Filter bank Multicarrier Modulation. The system consists of multiple modules and dynamic partial configuration to obtain a flexible and evolvable system. The hardware virtualization shows that the design was resource efficient.

Mahesh S. Murthy *et al*. [30] developed spectrum sensing on, Reconfigurable system of time domain-based cyclostationary detection. The simulation was in a 90nm CMOS process, and FPGA was implemented. Using the software radio peripheral transceivers, real-world signals were received. The system was high memory efficient by 99% and hardware efficient by 33%.

D.Tequiq *et al*. [31] presented Energy Detection and Goodness of Fit spectrum sensing in Cognitive Radio. The system was highly efficient in terms of performance and hardware design, based on blind detection methods analyzed in VHDL and FPGA technology.

K A Arun kumar [32] proposed a Reconfigurable spectrum sensing approach. The system addresses Matched filter detection, Cyclostationary detection and Energy detection techniques for spectrum sensing. The system was implemented in FPGA Xilinx Artix 7.

3. Problem Statement

The following problems were encountered when the spectrum sensing was designed in the FPGA platform.

- Different kinds of modulation techniques were used for different glitches. The hardware utilization was increased for designing each FPGA platform modulation technique.
- In the receiver end, a normal filter was used to reduce the noises which present in the receiver data. But, channel path identification was not discussed.
- In the encoding scheme, the encoding compression performed all the input data. Due to this process, a time delay has occurred in the cognitive radio networks.

3.1. Solutions

To overcome the above-mentioned problems, RMS-HE-FPGA architecture is used in this research work. RMS technique is used to design all the modulation techniques with a single architecture. The modulation will be chosen based on the glitches' density in the transmitter and receiver section. In the encoding model, HE encodes the data without affecting the system's performance. Therefore, the area, power, and speed of the RMS-HE-FPGA architecture are improved because of the RMS and HE.

4. RMS-HE-FPGA Architecture

In this research, a combination of RMS and HE is designed using the FPGA to perform the communication over the CR. Nowadays, communication systems are deployed in various scenarios, such as radar, radio, internet, mobile communication, etc. Each communication system requires different modulators and demodulators for operation, creating an issue in various situations.

Therefore, the reconfigurable modulation scheme is developed, where the same hardware is reconfigured for executing different modulations according to the noise constraints. This helps minimize the hardware resources required to develop the RMS-HE-FPGA architecture. On the other hand, the designed modulation and encoding techniques are used to minimize the error occurred during the transmission. The architecture of the transmitter and receiver for RMS-HE-FPGA are shown in Figure 1 and Figure 2.

The steps processed in the RMS-HE-FPGA architecture are given as follows:

- At first, the 8-bit input is stored in the register, followed by the stored 8-bit input data given as input to the glitter. Here, the control signals are used to define the time to process of each module. The glitter provides the 2-bit output according to the noise level of input data.
- The 2-bit output received from the glitter (i.e., control signal) is given as input to the RMS, which defines the modulation type required to process in the RMS. This RMS receives one more input, i.e., serial to parallel conversion of 8-bit input data. Further, this 8-bit data is modulated, and a 3-bit MSB of 0 is added to create an 11-bit of data.
- The generated 11-bit data is input to the Hamming Encoder, resulting in 16-bit encoded data. Moreover, the encoded 16-bit of data processed under up sampler to add the zeroes in certain clock cycles.
- Next, 16-bit of the data is transmitted, and the down sampler module of the receiver receives it. This down sampler is used to remove the zeroes where the up sampler added zeroes in the encoded data.
- Further, the process of hamming the decoder and demodulation is accomplished to obtain the given 8-bit input data.

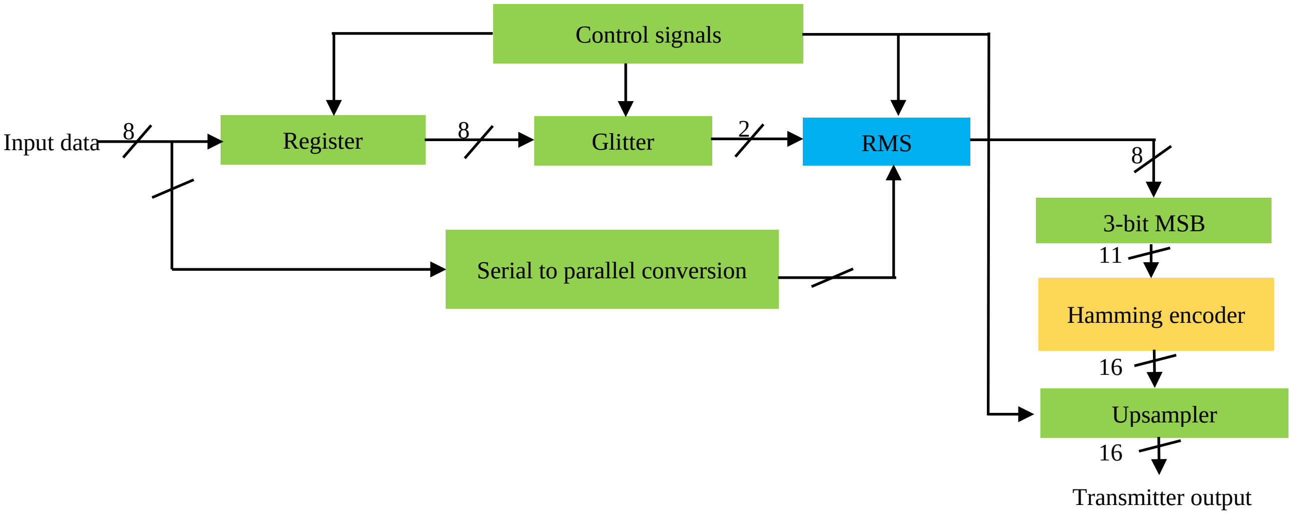


Fig. 1 Architecture of CR transmitter for RMS-HE-FPGA

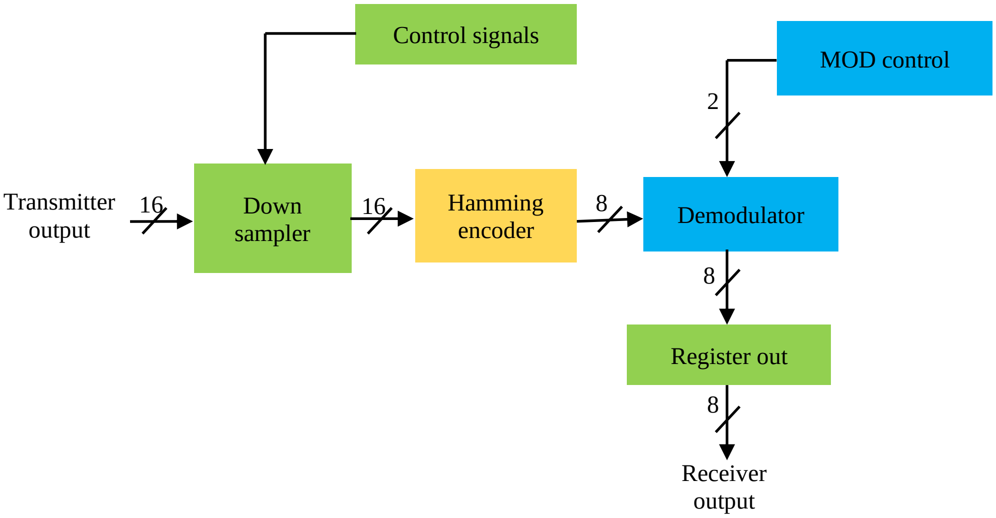


Fig. 2 Architecture of CR receiver for RMS-HE-FPGA

4.1. Process of Glitter

The glitter used in the RMS-HE-FPGA architecture is used to define the modulation type according to the noise level. The user-defined values considered for enabling the modulations are specified in the following Table 1.

Table 1. Control signals of glitter

Noise level	Control signal	Modulation type
Between 1010110 and 10101101	00	BPSK
Greater than 10101101	01	QPSK
Less than 1010110	10	QAM

4.2. Reconfigurable Modulation Scheme

In this RMS-HE-FPGA architecture, three different modulation schemes, such as BPSK, QPSK and QAM, designed using FPGA, are used in a reconfigurable way to achieve effective communication over the CR. These FPGA-based modulation techniques select a certain technique according to the signal received from the glitter. The process of each modulation technique is detailed as follows:

4.2.1. BPSK

The sinusoidal carrier signal's phase is varied based on the message level (i.e., 1 or 0) with the amplitude and frequency constant. In general, the BPSK is one of the easiest PSK modulation approaches. This BPSK uses two different phases such as $0°$ and $180°$, whereas the BPSK modulation is shown in Figure 3.

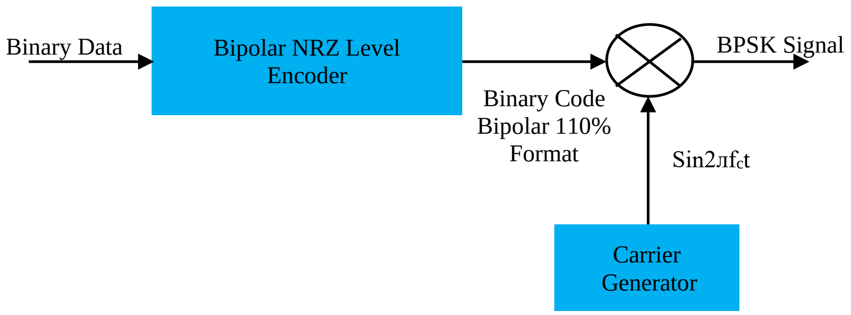


Fig. 3 BPSK modulation

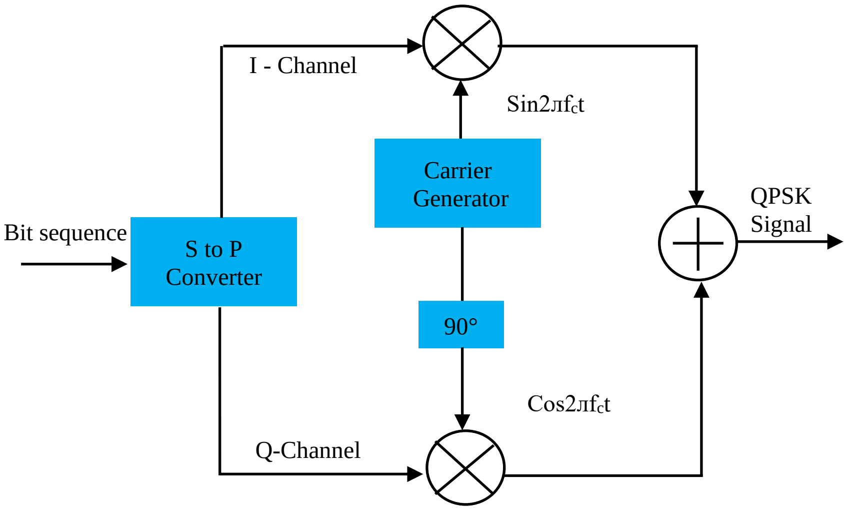


Fig. 4 QPSK modulation

Equation (1) and (2) shows the transmitted signal of BPSK.

$$S_{BPSK} = \sqrt{\frac{2E_b}{T_b}}\cos(2\pi fct + \pi + \Theta_c) \tag{1}$$

$$S_{BPSK} = -\sqrt{\frac{2E_b}{T_b}}\cos(2\pi fct + \pi + \Theta_c) \quad 0 \le t \le T_b(binary\ 0) \tag{2}$$

Where, the energy per bit is represented as E_b, $E_b = 1/2(A_c)^2T_b$; bit duration is denoted as T_b, and A_c defines the amplitude of the sinusoidal signal.

4.2.2. QPSK Modulation

QPSK modulation is phase modulation; accordingly, different phase data is forwarded to the channel for each symbol. Here, each symbol has two bits which are modulated in the I channel and the Q channel. The sinus and cosines are used for the channel I carrier signal and channel Q carrier signal. Four distinct cases are used for QPSK modulation, such as 00, 01, 10 and 11. A different phase signal is used for generating the QPSK signal in each case. The different phases are $45°,135°,225°$ and 315° whereas the architecture of QPSK modulation is shown in Figure 4.

The QPSK signal is expressed in equation (3).

$$\mathrm{S_{QPSK}(t)} = \sqrt{\frac{2\mathrm{E_s}}{\mathrm{T_s}}}\cos\left[2\pi \mathrm{f_c t} + \frac{(\mathrm{i\text{-}1})\pi}{2}\right] \quad 0 \le \mathrm{t} \le \mathrm{T_s}\,\mathrm{for\ i} = 1,2,..\mathrm{M} \tag{3}$$

Where, E_s defines the energy per symbol and T_s specifies the symbol duration. The general two functions for the QPSK are shown in equations (4) and (5).

$$\varphi_1(t) = \sqrt{\frac{2}{T_s}}\cos(2\pi f_c t) \tag{4}$$

$$\varphi_2(t) = \sqrt{\frac{2}{T_s}}\sin(2\pi f_c t)\ for\ 0 \le t \le T_s \tag{5}$$

Hence, the equation (3) is rewritten as shown in equation (6).

$$S_{QPSK}(t) = \sqrt{E_s}\cos\left[\frac{(i-1)\pi}{2}\right]\varphi_1(t) + \sqrt{E_s}\sin\left[\frac{(i-1)\pi}{2}\right]\varphi_2(t)\ \ for\ i = 1,2,..M \tag{6}$$

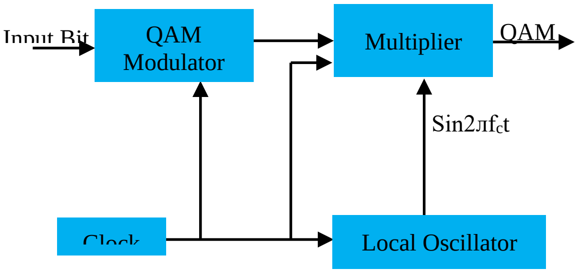


Fig. 5 QAM modulation

4.2.3. QAM Modulation

In QAM, 2 sinusoidal carrier signals are independently modulated based on 90° phase shift, and it is demodulated in the receiver section. The two carried signals are In-phase (I) and the Quadrature phase (Q). The developed QAM is generally a bandpass digital modulation that modulates both the phase and amplitude of the carrier signal, resulting in improved error performance in the receiver. Figure 5 shows the architecture of QAM modulation.

$$S_{QAM}(t) = \sqrt{\frac{2E_{min}}{T_s}} a_i \cos(2\pi f c t) + \sqrt{\frac{2E_{min}}{T_s}} b_i \cos(2\pi f c t) \quad 0 \le t \le T, i = 1,2, \ldots M \tag{7}$$

Where, the signal's energy with less amplitude is denoted as E_{min}; two independent integer pairs selected according to the certain location of signal points are represented as a_i and b_i.

Therefore, the RMS generates 8-bit modulated data, and then 3-bit MSB of 0s is added to the modulated data to generate 11-bit data, as shown in equation (8).

$$d = \{0,0,0,8 - \text{bit modulated data}\} \tag{8}$$

4.3. Process of Hamming Encoder and Upsampler

The hamming encoder is used to perform the bit error minimization that occurred during the communication. The 11-bit generated after modulation and 3-bit MSB addition are given as input to this hamming encoder to generate 16-bit encoded output data. The hamming encoder is one of the forward error correction coding generally used in the communication system (i.e., CR). The binary information source broadcasts the data or a bit sequence to the HE followed by redundant or parity bits injected in the encoder, resulting in the lengthier sequence of code bits named code word. Next, these code words are broadcasted using the communication channel.

The hamming code used in this RMS-HE-FPGA architecture is a common linear block code. The developed hamming code identifies and corrects a single-bit error in a data block. Each bit is inserted in a distinctive set of parity bits in this HE. The combinations of the received bit's parties must be analysed to determine the existence and position of a single parity bit error. Moreover, the combinations of received bit's parities produced the table of parities where each parity is related to a certain bit-error combination. This table of errors is referred to as error syndrome. If the parties are corrected based on the pattern, then it is determined that the message does not have any single-bit error. The location of erroneous parities is added to find the erroneous data bit when a single-bit error creates the errors in the parties. The architecture HE used in the RMS-HE-FPGA architecture is shown in Figure 6.

In encoding, the code of (x, d, t) represents the d data bits of the block which is mapped in the code word of x (where $d < x$) and error control bits are represented as b_r which is expressed in equation (9).

$$b_r = x - d \tag{9}$$

The error control bit b_r along with the code, has the capacity to adjust t amount of bits in error. The b_r is capable of denoting the $x + 1$ various states when the total amount of bits in the transmitter section is the code word x. Equation (10) and (11) expresses the x and $x + 1$.

$$x = d + b_r \tag{10}$$

$$x + 1 = d + b_r + 1 \tag{11}$$

Here, one state represents no error, and the state x represents the error's position. Therefore, the bits b_r is used to identify the $n + 1$ states and the different states $2b_r$ is represented by using bits b_r. Hence, the $2b_r$ is either equal or greater than the $n + 1$. Since the transmitted length is used to determine the b_r value. The 16-bit encoded data is processed through up sampler to generate the 16-bit of data. This 16-data is transmitted over the CR, and it is received by the down sampler used in the receiver unit.

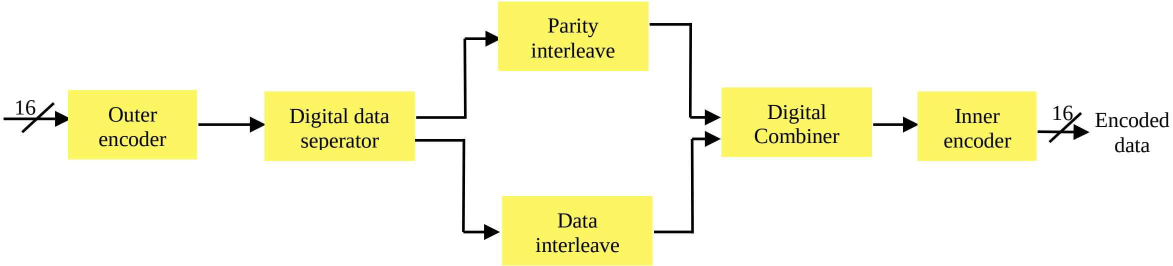


Fig. 6 Architecture of HE circuits

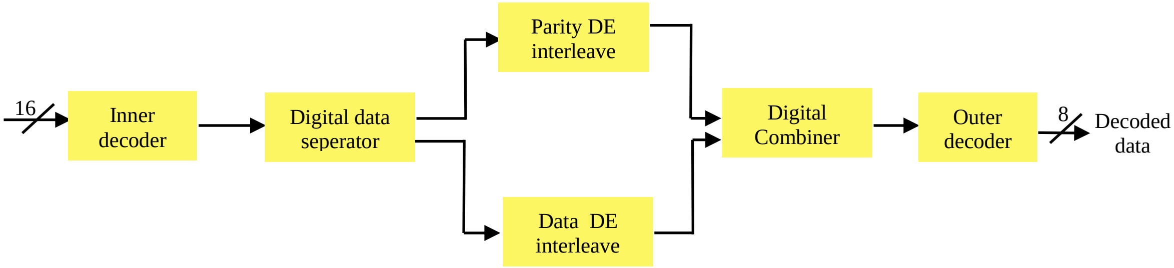


Fig. 7 Architecture of hamming decoder

4.4. Process of Down Sampler and Hamming Decoder

The 48-bits of data obtained from the down sampler are given as input to this hamming decoder. The architecture of hamming decoder is shown in Figure 7, represented as k: n where n and k define the amount of encoded and message bits. The input data given to the decoder is separated into 4 sets of 12 bits, followed by each set being transferred to the inner decoder. The error is corrected using a correcting mechanism when a single-bit error is identified in the encoded data. If the double-bit error is identified, then an error flag is created during the decoding process. Here, each 8-bit data has 4-bit data and 4-bit parity.

Moreover, this data and parities are divided and provided to the data and parity de-interleaver block for achieving the original form of data given before the encoding process. The outer decoder receives each 8-bit data and then decodes it into a 4-bit data stream with 1-bit error correction and 2-bit error detection. Hence, the outer decoder offers the original data

The hamming decoder produces 8-bit of decoded data, then demodulates according to the control signals, which returns demodulated data of 8-bit. This demodulated data is stored in the register, and this register provides the output data of 8-bit.

5. Results and Discussions

This section shows the outcomes of the RMS-HE-FPGA architecture along with its comparative and hardware analysis. The RMS-HE-FPGA architecture is developed for the CR communication application using the Xilinx ISE 14.2 software. The design and functional simulations of the designed RMS-HE are performed using Verilog Description Language (VHDL) and ModelSim simulator, respectively. The design specifications of the RMS-HE-FPGA architecture are given in Table 2.

Table 2. Design specifications

Parameter	Value
Clock frequency	50MHz
Clock period	20ns
Pulse width	10ns
Duty cycle	50%

5.1. Performance Analysis

The performance of the RMS-HE-FPGA architecture is investigated in two different FPGA devices such as Spartan-3A, Spartan -6 and Artix-7 FPGA devices. The RMS-HE-FPGA architecture is evaluated by means of the number of slice registers, flip flops, number of slice LUTs, number of logical elements, slices, bonded IOB, delay, power and operating frequency.

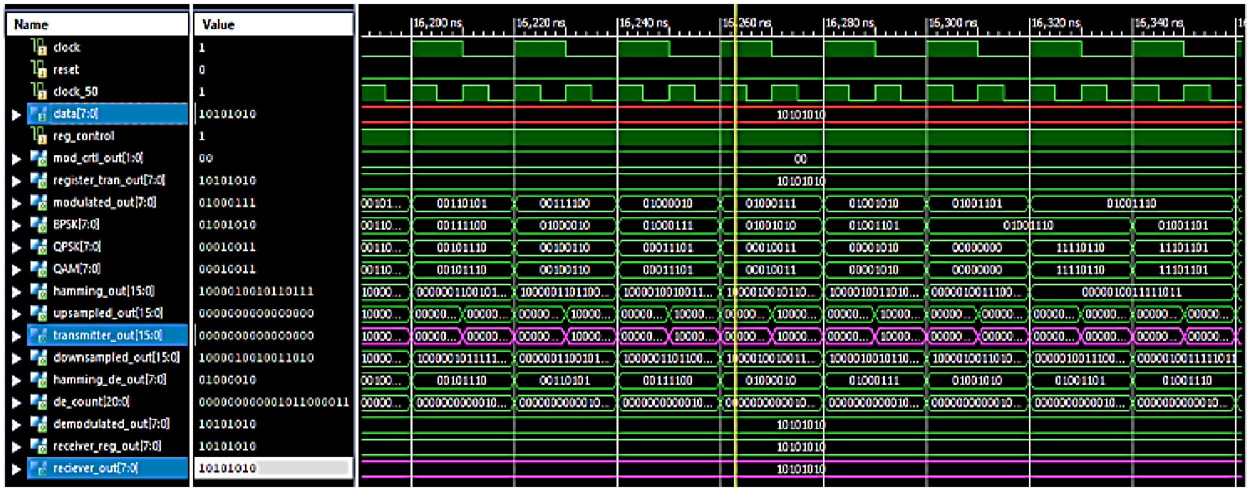
The hardware resource usage of RMS-HE-FPGA architecture for Artix-7, Spartan 3A and Spartan 6 FPGA devices are shown in Tables 3, 4 and 5, respectively. Moreover, the analysis of delay, power and operating frequency for RMS-HE-FPGA are given in Table 6. These hardware resource usage results provide information about the amount of logical elements used during the data communication. The important resources, namely LUT, slices and flip flops, are used the 1%, 1% and 1% of the resources when RMS-HE-FPGA architecture is designed in the Artix-7. Besides, the RMS-HE-FPGA designed in the Artix-7 achieves a higher operating frequency of 458.288 MHz, whereas the RMS-HE-FPGA designed in the Spartan 3A obtains 266.902 MHz and Spartan 6 obtains 369.045 MHz

Table 3. Hardware utilization of RMS-HE-FPGA architecture for Artix-7 FPGA

FPGA performances	Available	Used	Utilization in %
Slice registers	28800	38	1%
Flip Flops	28800	38	1%
Slice LUTs	28800	174	1%
Logical elements	28800	166	1%
Slices	7200	62	1%
Bonded IOB	480	26	5%

Table 4. Hardware utilization of RMS-HE-FPGA architecture for Spartan 3A FPGA

FPGA performances	Available	Used	Utilization in %
Slice registers	7168	38	1%
Flip Flops	7168	37	1%
Slice LUTs	7168	229	3%
Logical elements	7168	224	3%
Slices	3584	122	3%
Bonded IOB	140	26	18%



Name	Value
clock	1
reset	0
clock_50	1
data[7:0]	10101010
reg_control	1
mod_crtl_out[1:0]	00
register_tran_out[7:0]	10101010
modulated_out[7:0]	01000111
BPSK[7:0]	01001010
QPSK[7:0]	00010011
QAM[7:0]	00010011
hamming_out[15:0]	1000010010110111
upsampled_out[15:0]	0000000000000000
transmitter_out[15:0]	0000000000000000
downsampled_out[15:0]	1000010010011010
hamming_de_out[7:0]	01000010
de_count[20:0]	000000000001011000011
demodulated_out[7:0]	10101010
reciever_reg_out[7:0]	10101010
reciever_out[7:0]	10101010

Fig. 8 Overall simulation waveform

Table 5. Hardware utilization of RMS-HE-FPGA architecture for Spartan 6 FPGA

FPGA performances	Available	Used	Utilization in %
Slice registers	11440	45	1%
Flip Flops	11440	47	1%
Slice LUTs	5720	201	3%
Logical elements	9112	180	1%
Slices	1430	49	3%
Bonded IOB	102	19	18%

Table 6. Analysis of delay, power and operating frequency

FPGA devices	Delay (ns)	Power (W)	Operating frequency (MHz)
Artix-7	1.094	0.010	458.288
Spartan 3	2.981	0.022	266.902
Spartan 6	1.994	0.018	369.045

The overall simulation waveform for the RMS-HE-FPGA architecture is shown in Figure 8. The control signals generally used for this RMS-HE-FPGA are $clock, reset, clock\_50$ and $reg\_control$.

The process that occurred between the CR transmitter and receiver using RMS-HE-FPGA is given as follows:

The 8-bit input data given to the CR transmitter is mentioned in equation (12). Subsequently, the given input is stored in the register, which further provides input to process in the transmitter section. The output from the register of the CR transmitter ($register\_tran\_out$) is shown in equation (13).

$$data = \{10101010\} \qquad (12)$$

$$register\_tran\_out = \{10101010\} \qquad (13)$$

Next, this $data$ is given as input to the glitter, which returns the 2-bit control signal for the modulation (mod_ctrl_out). Since the mod_ctrl_out is generated according to the noise density of the input data, whereas the generated mod_ctrl_out is expressed in equation (14).

$$mod_{ctrl_{out}} = \{00\} \qquad (14)$$

Equation (14) specifies that the BPSK modulation of RMS is going to be carried out in the $register\_tran\_out$ and the modulation output ($modulated\_out$) is expressed in equation (15). Subsequently, a 3-bit MSB of zeros are added to the 8-bit output of RMS and given to the HE to encode the data, as shown in equation (16).

$$modulated_{out} = \{01001010\} \qquad (15)$$

$$hamming_{out} = \{1000010010110111\} \qquad (16)$$

The encoded data is processed under upsampling to generate a 16-bit of upsampled output ($upsampled\_out$) as shown in equation (17). This $upsampled\_out$ is the transmitter output ($transmitter\_out$) which is transmitted over the CR.

$$transmitter_{out} = \{0000000000000000\} \qquad (17)$$

On the receiver side, the inverse processes such as downsampling, decoding, and demodulation are carried out to generate the 8-bit output data ($reciever\_out$), which is shown in equation (18).

$$reciever_{out} = \{10101010\} \qquad (18)$$

The output obtained from the receiver shows that the developed CR communication application using RMS-HE-FPGA architecture provides errorless data transmission while broadcasting the data.

5.2. Comparative Analysis

This section provides a comparison of the RMS-HE-

FPGA architecture with recent years of research. The existing researches considered for the comparison are NC-OFDM-QPSK [22] and DA-RNS [23]. Here, the comparison is made for two different FPGA devices such as Spartan-3A and Artix-7 FPGA.

Table 7. Comparison of RMS-HE-FPGA for Spartan-3A

FPGA performances	NC-OFDM-QPSK [22]	RMS-HE-FPGA
LUT	2898	229
FF	2817	37

Table 8. Comparison of RMS-HE-FPGA for Artix-7

FPGA performances	DA-RNS [23]	RMS-HE-FPGA
LUT	249	174
Slices	174	62
Delay (ns)	4.771	1.094
Power (mW)	0.088	0.010

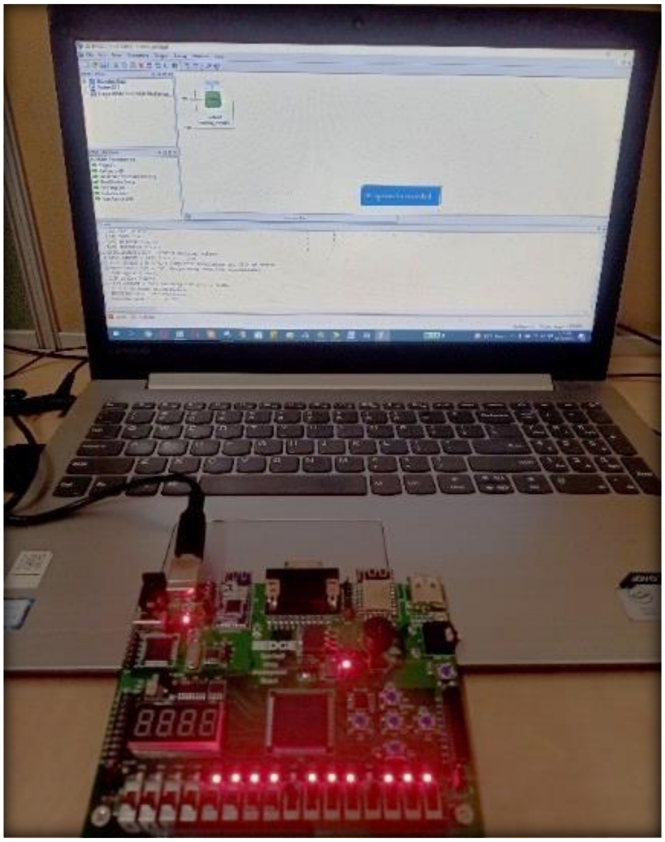


Fig. 9 Hardware setup

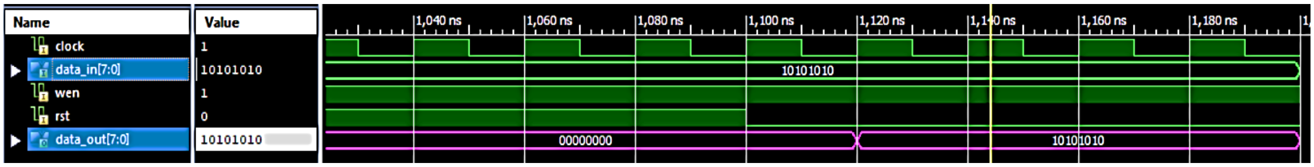
Table 7 and 8 shows the performance analysis of RMS-HE-FPGA for Spartan-3A and Artix-7 FPGA devices, respectively. For Spartan-3A, the RMS-HE-FPGA is compared with the NC-OFDM-QPSK [22], whereas the DA-RNS [23] is used for Artix-7. From the analysis, it is concluded that the RMS-HE-FPGA provides better performance than the NC-OFDM-QPSK [22] and DA-RNS [23]. For example, the LUT of the RMS-HE-FPGA designed in the Artix-7 is 174, which is less when compared to the DA-RNS [23]. The reconfigurable modulation scheme used in the RMS-HE-FPGA minimizes the hardware resources.

5.3. Hardware Analysis

The hardware analysis of the RMS-HE-FPGA architecture is shown in this section. The hardware setup of the RMS-HE-FPGA is shown in Figure 9, where the type of hardware development board considered for this RMS-HE-FPGA is Spartan 6 FPGA.

As mentioned in Table 2, the overall clock frequency for the RMS-HE-FPGA architecture is 50MHz. The VHDL code is dumped into the hardware to perform the hardware analysis of the RMS-HE-FPGA architecture. The process of dumping software code into hardware is specified as follows: 1) Code synthesize, 2) Implementation that has different steps including translation, map & routing, 3) User Constrained File (UCF) generation to accomplish the interconnection among the software and hardware, 4) Insertion of 8-bit input data, 6) Allocate the switch pin to the input data, 6) Output acquired from LED, 7) Programming file generation that creates a bit file for the code, 8) The manage configuration project is clicked to load the bit file in Spartan 6 memory and finally 9) Code dumbed successfully.

The simulation waveform for the register used in the transmitter is shown in Figure 10, whereas the hardware output of the register is shown in Figure 11. The 8-bit input given to the register of the CR transmitter is 10101010, which is equal to the decimal number 170. Since the control signals, i.e., clock, enable (wen), and reset (rst), given to the registers are 1, 1 and 0, respectively.The same 8-bit input data is taken as output when the $wen = 1$ and $rst = 0$; Otherwise, the output is zero when the $wen = 1$ and $rst = 1$. In Figure 11, the first two switches are the clock signals (rst and wen), and 8 switches from the right to left are the switches for 8-bit input. The register output $(data\_out)$ is taken from the LED. From the LED output of Figure 11, it is known that the hardware register module works similarly to the simulation waveform shown in Figure 10.



Name	Value
clock	1
data_in[7:0]	10101010
wen	1
rst	0
data_out[7:0]	10101010

Fig. 10 Register simulation waveform

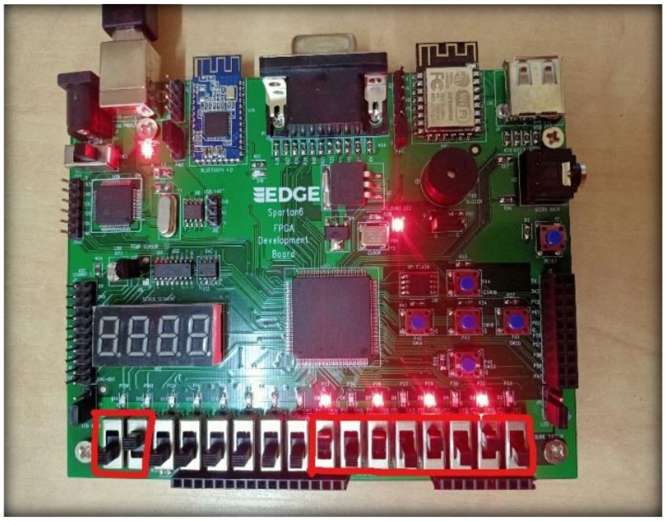
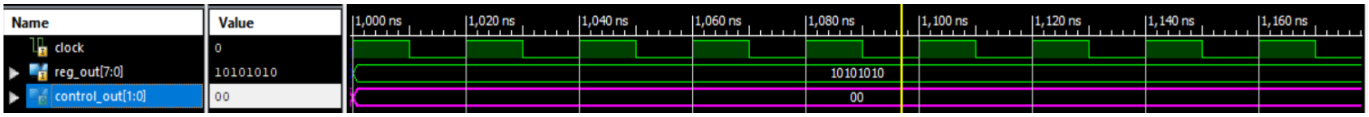


Fig. 11 Register hardware output



Name	Value
clock	0
reg_out[7:0]	10101010
control_out[1:0]	00

Fig. 12 Glitter simulation waveform

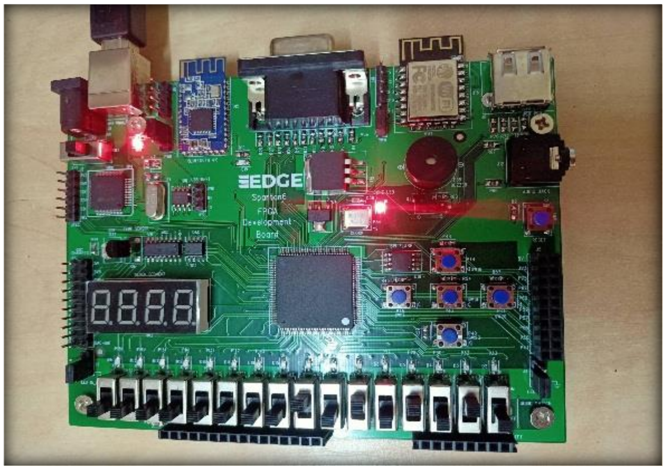
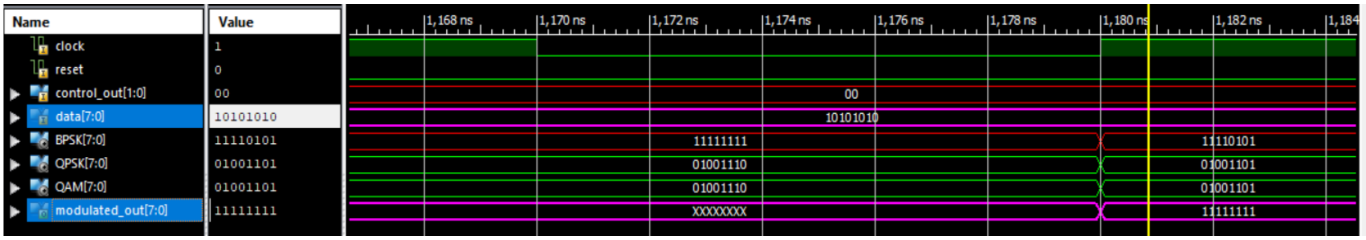


Fig. 13 Glitter hardware output



Name	Value
clock	1
reset	0
control_out[1:0]	00
data[7:0]	10101010
BPSK[7:0]	11110101
QPSK[7:0]	01001101
QAM[7:0]	01001101
modulated_out[7:0]	11111111

Fig. 14 Modulation simulation waveform

The 8-bit register output (10101010) is given as input to the glitter where the clock signal value is 0. The simulation waveform for the glitter is shown in Figure 12, and the hardware output of the glitter is shown in Figure 13. Moreover, the designed glitter provides the control signal $(control\_out)$ for RMS according to the reg_out. According to the condition mentioned in Table 1, the glitter is required to provide a control signal of 00. From Figure 12, it is known that the control signal obtained from the glitter is 00, which is used to define the type of modulation technique required to process in the RMS.

Table 9. Input UCF for modulation

Input	Pin name of switches
clock	p84
control_out [0]	p2
control_out [1]	p1
data[0]	p22
data[1]	p21
data[2]	p17
data[3]	p16
data[4]	p15
data[5]	p14
data[6]	p12
data[7]	p11

Table 10. Output UCF for modulation

Input	Pin name of switches
modulated_out[0]	p33
modulated_out [1]	p32
modulated_out [2]	p30
modulated_out [3]	p29
modulated_out [4]	p27
modulated_out [5]	p26
modulated_out [6]	p24
modulated_out [7]	p23

On the other hand, the reg_out is given as input for the hardware glitter module through the 8 switches from right to left. Accordingly, this hardware module returns the output of 00. Therefore, the hardware module of the glitter works similarly to the simulation design.

Figures 14 and 15 show the simulation waveform and hardware output of the modulation. The input and output UCF are shown in Tables 9 and 10. The 8-bit input data given to the modulation is 10101010, followed by BPSK modulated value is taken as the output according to the control signal 00. Therefore, the BPSK modulated value of 8-bit input data is 11111111. On the other hand, the hardware modulator also provides the same output of 11111111. The simulation waveform and hardware output of hamming encoder are shown in Figures 16 and 17.

After modulating the input data, 3-bit of 0's are inserted in the MSB of modulated data. Accordingly, the 11-bit data of 00011111111 is given as input to the hamming encoder, and it returns the 16-bit output $(hamming\_out)$ of 0000111101110111. The hardware module of the modulator also provides the same output when compared to the simulation, which proves that both the simulation and hardware design of the HE is the same.

Figures 18 and 19 show the simulation waveform and hardware output of the hamming decoder. The 16-bit input data (in) collected in the hamming decoder of the receiver is 0000111101110111, and it decodes the given 16-bit data into 11-bit data (out), i.e., 00011111111. Further, the hamming decoder resulted in the 8-bit data $(hamming\_decoder\_out)$ of 11111111 which is a decoded data. This simulation and hardware output show that it provides the same output value.

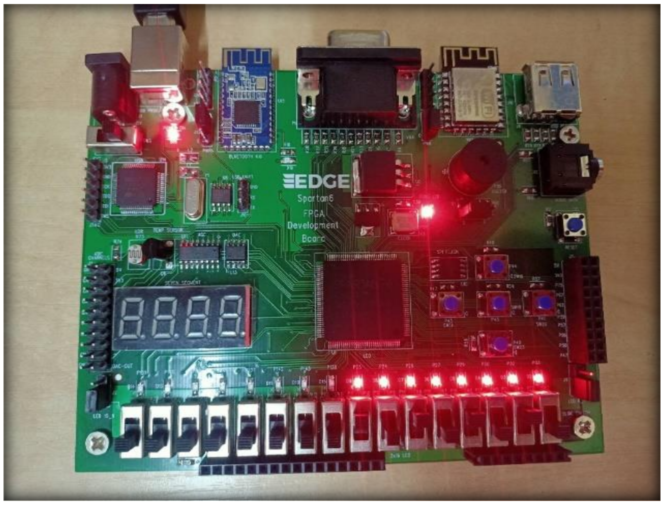
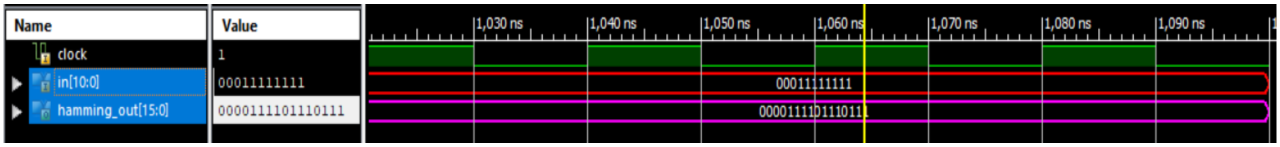


Fig. 15 Modulation hardware output



Name	Value
clock	1
in[10:0]	00011111111
hamming_out[15:0]	0000111101110111

Fig. 16 Hamming encoder simulation waveform

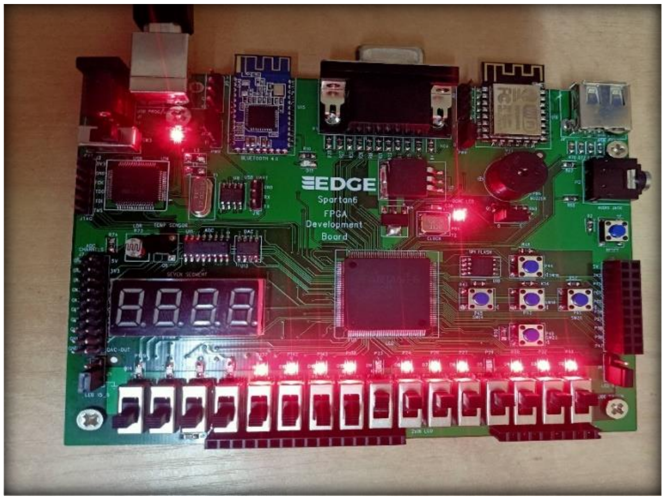
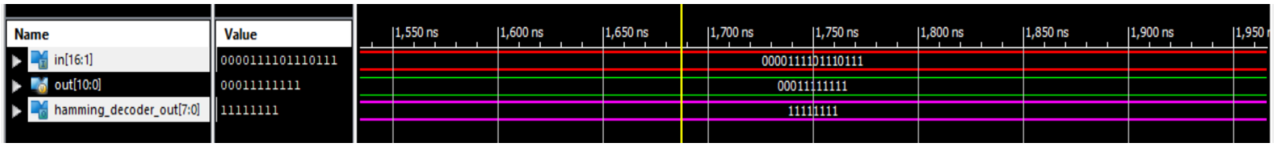


Fig. 17 Hardware output of the hamming encoder



Name	Value
in[16:1]	0000111101110111
out[10:0]	00011111111
hamming_decoder_out[7:0]	11111111

Fig. 18 Simulation waveform of hamming decoder

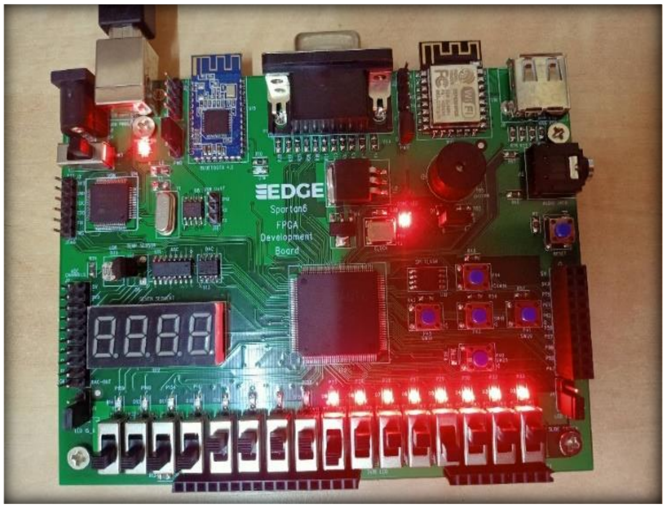
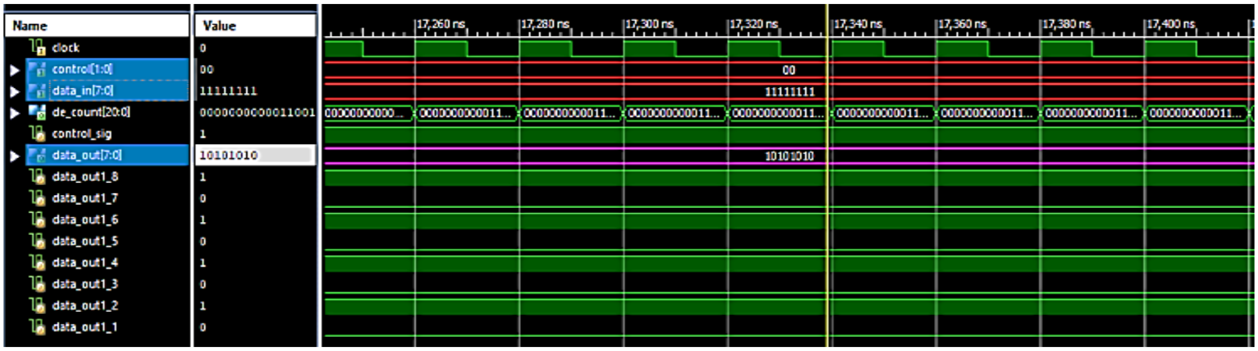


Fig. 19 Hardware output of the hamming decoder



Name	Value
clock	0
control[1:0]	00
data_in[7:0]	11111111
de_count[20:0]	000000000000011001
control_sig	1
data_out[7:0]	10101010
data_out1_8	1
data_out1_7	0
data_out1_6	1
data_out1_5	0
data_out1_4	1
data_out1_3	0
data_out1_2	1
data_out1_1	0

Fig. 20 Simulation waveform of demodulation

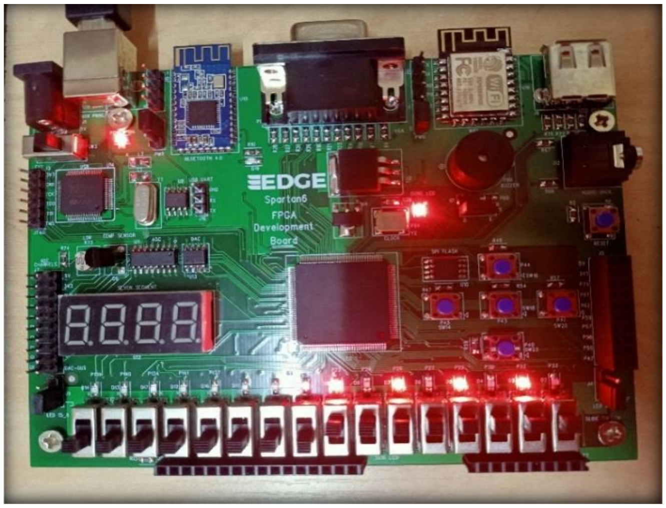


Fig. 21 Hardware output of the demodulation

Further, the simulation waveform and hardware output of BPSK demodulation are shown in Figures 20 and 21, respectively. The 8-bit decoded data $(data\_in)$ from the hamming decoder is given as input to the demodulator, which provides the 8-bit output data $(data\_out)$ of 10101010 which is stored and taken as final output at the receiver. The hardware module also provides the same output as 10101010.

4. Conclusion

In this proposed work, the RMS and HE are developed using FPGA to perform effective communication over the CR. In the RMS approach, the same hardware module is used for three different modulation techniques such as QPSK, BPSK and QAM, which are used to minimize the overall RMS-HE-FPGA architecture's hardware resources. Moreover, the HE used in the CR is used to obtain error-free data transmission. Therefore, the combination of RMS and HE is used to obtain error-free communication over the CR while minimizing the hardware resources. From the performance analysis, it is known that the RMS-HE-FPGA architecture outperforms well than the OFDM-QPSK and DA-RNS. Moreover, the hardware analysis using the Spartan 6 FPGA device shows that the developed RMS-HE-FPGA architecture provides the same results in both the software and hardware analysis. The LUT of the RMS-HE-FPGA designed in the Artix-7 is 174, which is less when compared to the DA-RNS.

Acknowledgments

We obliged all the teaching and non-teaching staff of the Department of Electronics & Telecommunication Engineering, Dr. Ambedkar Institute of Technology, Bangalore, for their cooperation and for the support given for completing the work.

References

[1] Mahesh S. Murty, and Rahul Shrestha "Hardware Implementation and VLSI Design of Spectrum Sensor for Next-Generation LTE-A Cognitive-Radio Wireless Network," *IET Circuits, Devices & Systems*, vol. 12, no. 5, pp. 542-550, 2018. [CrossRef] [Google Scholar] [Publisher link]

[2] Ardalan Alizadeh, Hamid Reza Bahrami, and Mehdi Maleki, "Performance Analysis of Spatial Modulation in Overlay Cognitive Radio Communications," *IEEE Transactions on Communications*, vol. 64, no. 8, pp. 3220-3232, 2016. [CrossRef] [Google Scholar] [Publisher link]

[3] Rohit B. Chaurasiya, and Rahul Shrestha, "A New Hardware-Efficient Spectrum-Sensor VLSI Architecture for Data-Fusion-Based Cooperative Cognitive-Radio Network," *IEEE Transactions on Very Large-Scale Integration (VLSI) Systems*, vol. 29, no. 4, pp. 760-773, 2021. [CrossRef] [Google Scholar] [Publisher link]

[4] Bin Tang et al., "Digital Signal Modulation Classification with Data Augmentation Using Generative Adversarial Nets in Cognitive Radio Networks," *IEEE Access*, vol. 6, pp. 15713-15722, 2018. [CrossRef] [Google Scholar] [Publisher link]

[5] Xin Liu et al., "A Multichannel Cognitive Radio System Design and Its Performance Optimization," *IEEE Access*, vol. 6, pp.12327-12335, 2018. [CrossRef] [Google Scholar] [Publisher link]

[6] D. Damodaram, and T. Venkateswarlu, "FPGA Implementation of Genetic Algorithm to Detect Optimal User By Cooperative Spectrum Sensing," *ICT Express*, vol. 5, no. 4, pp. 245-249, 2019. [CrossRef] [Google Scholar] [Publisher link]

[7] C. S. Karthikeyan, and M. Suganthi, "Optimized Spectrum Sensing Algorithm for Cognitive Radio," *Wireless Personal Communications*, vol. 94, no. 4, pp. 2533-2547, 2017. [CrossRef] [Google Scholar] [Publisher link]

[8] Mingxuan Li et al., “Generative Adversarial Networks-Based Semi-Supervised Automatic Modulation Recognition for Cognitive Radio Networks,” *Sensors*, vol. 18, no. 11, p. 3913, 2018. [CrossRef] [Google Scholar] [Publisher link]

[9] Rohit B. Chaurasiya, and Rahul Shrestha, “Fast Sensing-Time and Hardware-Efficient Eigenvalue-Based Blind Spectrum Sensors for Cognitive Radio Network*,” IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 4, pp.1296-1308, 2019. [CrossRef] [Google Scholar] [Publisher link]

[10] E.Ganesan, and V.Sakthivel, "A Novel FPGA Design with Hybrid LUT / MUX Architecture," *SSRG International Journal of Electronics and Communication Engineering*, vol. 3, no. 11, pp. 6-8, 2016. [CrossRef] [Google Scholar] [Publisher link]

[11] R.Belly Ballot, T.Anisley, and N.Addison, "Advanced Signal Recognition Method for Path Using FPGA," *SSRG International Journal of VLSI & Signal Processing*, vol. 4, no. 3, pp. 26-30, 2017. [CrossRef] [Publisher link]

[12] Rugui Yao et al., “Intelligent Modulation Pattern Recognition Based on Wavelet Approximate Coefficient Entropy in Cognitive Radio Networks,” *IEEE Access*, vol. 8, pp. 226176-226187, 2020. [CrossRef] [Google Scholar] [Publisher link]

[13] Xiaoyong Sun et al., “Modulation Classification Using Compressed Sensing and Decision Tree–Support Vector Machine in Cognitive Radio System,” *Sensors*, vol. 20, no. 5, p. 1438, 2020. [CrossRef] [Google Scholar] [Publisher link]

[14] E.Gnana Manoharan, and K.Vadivelu, “CFD Technique-Based Spectrum Sensor for Hardware-Efficient and Fast Sensing of Cognitive Radio Network,” *European Journal of Molecular & Clinical Medicine*, vol. 7, no. 9, pp. 1770-1776, 2020. [Google Scholar] [Publisher link]

[15] Ms.Manjula B.M, and Dr.Chirag Sharma, "FPGA Implementation of BCG Signal Filtering Scheme by Using Weight Update Process," *SSRG International Journal of VLSI & Signal Processing*, vol. 3, no. 3, pp. 1-7, 2016. [CrossRef] [Google Scholar] [Publisher link]

[16] S. N. Hemanth Kumar et al., “FPGA Implementation of Digital Modulation Schemes Using Verilog Hdl,” *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol.12, no. 12, pp. 2442-2451, 2021. [Google Scholar] [Publisher link]

[17] A. Rajalakshmi, and A. Kavitha, “Suppression of EMI Using Cost-Effective FPGA-Based Digital Communication Modulation Techniques in Power Converters,” *IETE Journal of Research*, pp. 1-12, 2021. [CrossRef] [Google Scholar] [Publisher link]

[18] Md Wahid Sadiq, and Md Ahasan Kabir., “Design and Implementation of Reconfigurable ASK and FSK Modulation and Demodulation Algorithm on FPGA (Field Programmable Gate Array),” *Sensors International*, vol. 3, pp. 100155, 2022. [CrossRef] [Google Scholar] [Publisher link]

[19] Shanigarapu Nareshkumar, and Kalagadda Bikshalu, “Adaptive Absolute SCORE Algorithm for Spectrum Sensing in Cognitive Radio,” *Microprocessors and Microsystems*, vol. 69, pp. 43-53, 2019. [CrossRef] [Google Scholar] [Publisher link]

[20] Rohit B. Chaurasiya, and Rahul Shrestha, “Hardware-Efficient and Fast Sensing-Time Maximum-Minimum-Eigenvalue-Based Spectrum Sensor for Cognitive Radio Network,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 11, pp. 4448-4461, 2019. [CrossRef] [Google Scholar] [Publisher link]

[21] Mohammad Khayyeri, and Karim Mohammadi, “Design and Implementation of a High-Performance and High-Speed Architecture for Wideband Spectrum Sensing in Cognitive Radio Networks,” *Circuits, Systems, and Signal Processing*, vol. 39, no. 4, pp. 2151-2177, 2020. [CrossRef] [Google Scholar] [Publisher link]

[22] Penilop Parukutty Sanker et al., “Spectrum Shaping Using NC-OFDM for Cognitive Radio Applications,” *IET Communications*, vol. 14, no. 7, pp. 1120-1128, 2020. [CrossRef] [Google Scholar] [Publisher link]

[23] C. Srinivasa Murthy, and K. Sridevi, “FPGA Implementation of High Speed-Low Energy RNS Based Reconfigurable-FIR Filter for Cognitive Radio Applications,” *WSEAS Transactions on Systems and Control*, vol. 16, pp. 278-293, 2021. [CrossRef] [Google Scholar] [Publisher link]

[24] Rahul Sharma, Rahul Shrestha, and Satinder K. Sharma, “Hardware-Efficient and Short Sensing-Time Multicoset-Sampling Based Wideband Spectrum Sensor for Cognitrive Radio Network,” *IEEE Transactions on Circuits and Systems*, vol. 70, no. 3, pp. 1298-1310, 2023. [CrossRef] [Google Scholar] [Publisher link]

[25] Tolga Keleş, Sıddıka Berna Örs Yalçın, and Yaşar Kemal Alp, “Model Based Design of Software Defined and Cognitive Radio and Implementation on FPGA,” *Signal Processing and Communication Applications Conference (SIU)*, pp. 1-4, 2021. [CrossRef] [Google Scholar] [Publisher link]

[26] Abderrezzaq Bouhdjeur, “Autonomous Global Threshold Adjustment Algorithm for Energy Detection in Self-Reconfigurable Cognitive Radio Systems,” *International Conference on Advanced Electrical Engineering (ICAEE)*, pp. 1-5, 2022. [CrossRef] [Google Scholar] [Publisher link]

[27] Qiang Li et al., “Interplay Between Reconfigurable Intelligent Surfaces and Spatial Modulation: New Applіocation Paradigms,” *IEEE Wireless Communications*, pp. 1-18, 2022. [CrossRef] [Google Scholar] [Publisher link]

[28] Rinu C Varghese, and A. Amir, “Design of Computationally Efficient FRM Based Reconfigurable Filter Structure for Spectrum Sensing in Cognitive Radio,” *IEEE Humanitarian Technology Conference*, pp. 281-287, 2022. [CrossRef] [Google Scholar] [Publisher link]

[29] Mário Lopes Ferreira, and João Canas Ferreira, “Flexible and Dynamically Reconfigurable FPGA-Based FS-FBMC Baseband Modulator,” *IEEE International Symposium on Circuits and Systems*, pp. 1-5, 2018. [CrossRef] [Google Scholar] [Publisher link]

[30] Mahesh S. Murty, and Rahul Shrestha, “Reconfigurable and Memory-Efficient Cyclostationary Spectrum Sensor for Cognitive Radio Wireless Networks,” *IEEE Transactions on Circuits and Systems*, vol.65, no.8, pp. 1039-1043, 2018. [CrossRef] [Google Scholar] [Publisher link]

[31] D. Teguig, and M.S. Azzaz., “FPGA Implementation of Spectrum Sensing Methods for Cognitive Radio,” *International Symposium on Networks Computer and Communications*, pp. 1-5, 2018. [CrossRef] [Google Scholar] [Publisher link]

[32] K.A. Arun Kumar., “FPGA Implementation of Spectrum Sensing Engine for Cognitive Radios,” *International Conference on Networks and Advances in Computational Technologies*, pp. 116-119, 2017. [CrossRef] [Google Scholar] [Publisher link]

[33] Ramya R, Madhura R, "FPGA Implementation of Optimized BIST Architecture for Testing of Logic Circuits," *SSRG International Journal of VLSI & Signal Processing*, vol. 7, no. 2, pp. 36-42, 2020. [CrossRef] [Publisher link]

[34] Mohammad Khayyeri, and Karim Mohammadi, “Cooperative Wideband Spectrum Sensing in Cognitive Radio Based on Sparse Real-Valued Fast Fourier Transform,” *IET Communications*, vol. 14, no. 8, pp. 1340-1348, 2020. [CrossRef] [Google Scholar] [Publisher link]

[35] Rohit B. Chaurasiya, and Rahul Shrestha, “Area-Efficient and Scalable Data-Fusion Based Cooperative Spectrum Sensor for Cognitive Radio,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 4, pp. 1198-1202, 2020. [CrossRef] [Google Scholar] [Publisher link]

[36] Sumin D. Joseph et al., “UWB Sensing Antenna, Reconfigurable Transceiver and Reconfigurable Antenna Based Cognitive Radio Test Bed,” *Wireless Personal Communications*, vol. 96, no. 3, pp. 3435-3462, 2017. [CrossRef] [Google Scholar] [Publisher link]