FPGA Based Design And Simulation Of Extended Golay Decoder Design With Hardware Optimization

Mayanka Rai[1], Hema Singh[2]
[1] Dept of Electronics and Communication
[2]Associate Professor, Dept of Electronics and Communication
[1, 2] Technocrats Institute of Technology (Main), Bhopal, India

Abstract- *In wireless communication systems the ability of the receiver to detect and correct the error from the received information is the most important issue, so as to provide the correct information to the data processing system. To achieve this, there are numbers of methods explained by researchers to implement hardware and software. The selection of implementation method strongly depends on the ratio of error and information at the receiver unit of the communication system. The length of the communication link plays an important role as length increases the distance between the transmitter and the receiver and hence multiple bits of the transmitted information may change due to the effect of noise. This can cause extreme loss in many cases. This paper presents a brief on Field Programmable Gate Array (FPGA) based design and simulation of Golay Code (G23) and Extended Golay Code (G24). This paper presents a simulated design for implementation of Extended Golay Encoder and Decoder algorithm with optimization of the time in the operational circuit to encode and decode a data packet.*

Keywords- FPGA, Golay Code, Encoding, Extended Golay Code, Decoding, Hardware Optimization.

I. INTRODUCTION

Communication is become an important part of human life. Use of phones, satellites, computers and other devices has now become a basic need to solve our day to day problems in sharing messages through a channel. Unfortunately, errors in the messages that are being sending are caused by the noise. Particularly when sending messages is a complicated or expensive task, for example in satellite communication, it is important to find ways to restrained the amount of errors as much as possible. This is the central idea in coding theory: what we have received and what message was being sent. To make this problem simple and free of error we use error correcting codes. Addition of the redundancy bits to the messages is one of the best idea through which we enable to find out or correct the errors that may have occurred. The foremost idea is to add redundancy to the messages which enables us to both recognize and correct the errors that may have occurred. This paper proposed a specific type of error correcting codes, the extended Golay code G24. The extended Golay code was used for sending images of Jupiter and Saturn from the Voyager 1 and 2. There are three steps to transfer the information that a channel transmits and a receiver receives. At the time of transmission the information is changed, i.e., encoded, so to avoid the effect of error. In this condition we use error correction codes to detect and correct the error. Figure 1 shows that a message is encoded into a codeword and sent to the receiver through a channel. In this channel, the possibility exists that error may occur. The receiver tries to obtain the original message by decoding the word.

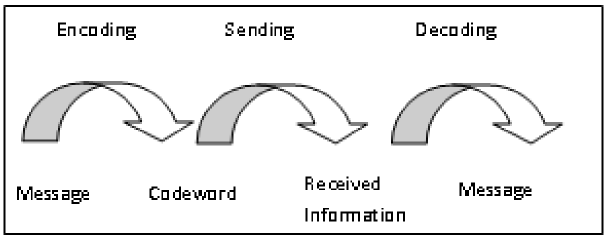


Figure 1. Process of Error correction

One of the best error correcting code that is proposed in this paper is Golay code which is an error correcting code which is used to specifies that what we have received and what is send. A detailed description of the extended Golay is described by some of the most important properties of such codes that is given by :

- Firstly a message m of length k is a sequence of k symbols out of some finite field F, so m = (m1 : : :mk) belongs to Fk. Then an n-code C over a finite field F is a set of vectors in Fn, where n ≤ k. Since we will be commerce with a binary code only, we will assume codes are binary from now on.
- Second property says that the error probability p is the probability that 0 is received when 1 was sent, or 1 is received when 0 was sent.
- Third property says that the hamming weight of a vector belongs to a function Fn is the number of its non zero elements.
- Fourth property says that the humming distance of two vectors belongs to a function Fn is the number of place where they differ. The idea is that an n-code C is a strict subset of Fn in which we want the

Hamming distance between any two vectors to be as large as possible. Therefore, the minimum Hamming distance is an important characteristic of the code.

- Fifth property says that the minimum Hamming distance d of a code C is defined as d = min {dist(x, y) I x, y belongs to C} where c is the code.

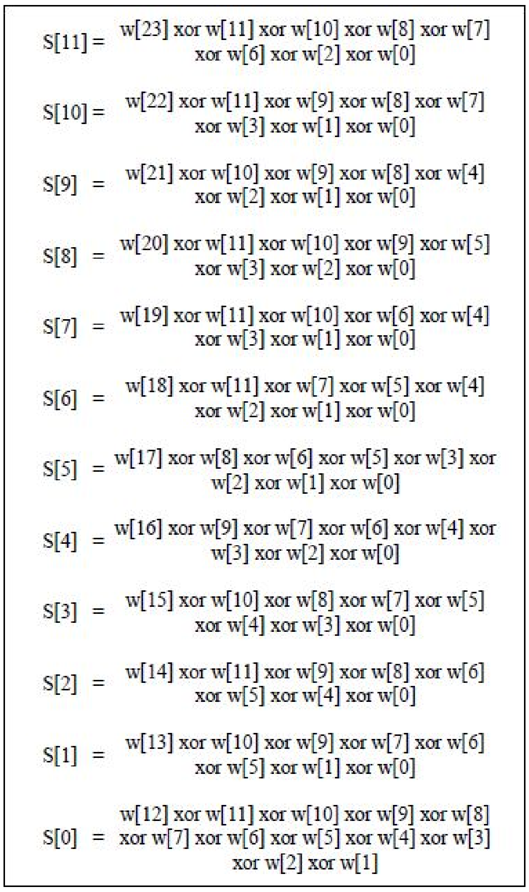
The description of work in this paper is arranged as follows: Section-II gives an overview on the work performed by other scholars in Golay Code implementation and applications. A brief of Golay code and its algorithm is described in Section-III. Section-IV presents the simulation and synthesis results of the performed work. The conclusion based on the proposed work and the future work scope is presented in Section-V. In the last the references are mentioned.

II. LITERATURE REVIEW

A In reference [1] the proposed paper presented error correcting phenomena using Golay code encoder. The algorithm of encoding data for error detection and correction was in the beginning proposed by Marcel J. E. Golay in 1949 [2]. A concise introduction and explanation of Golay coding scheme is presented in [3]. An implementation with complete verification of multiplication is simulated and an FPGA based 4-bit Golay Encoder and Decoder design in [4] using Xilinx ISE and ModelSim Tools. A soft algorithm based decoding orientation to hardware functioning of (24, 12, 8) Golay code with functioning of the algorithm on FPGA is presented by Reference [5]. In [6] it has been shown that (24, 12, 8) Golay code can be constructed with the help of the sum of two direct array codes that involve four component codes from which two are simple linear block codes. Assembly of Golay Code matching Sequences is presented in [7] for application of Golay Coding in the fields of surface acoustics, physics, combinatorial (orthogonal designs and Hadamard matrices), and tele-communication. Reference [8] represents a one on one mapping between the syndrome “S1” and correctable error-patterns scheme based adapted algorithm for decoding Binary Golay. In this proposed work the error location is determine by using look-up tables without the multiplication operation over a finite field. This algorithm has been established by the scholars on a C-language based software simulation platform. The work presented in [9] forced on Golay code decoding using symbol-by-symbol soft-in / soft-out APP (a posteriori probability) algorithm through co-set based technique. A study based on discussion on the error correction capability of BPSK modulation with Golay code and MSK modulation with Golay code is presented in [10], which concludes that MSK Golay code is comparatively more robust. In reference [11] a technique based upon reversing the conventional scheme of Golay code (24, 12, 8) that maps 24-bit vectors into 12-bit message words is focused to improve the search operation when multi-attribute objects are partially distorted. The work in [12] presents Golay code transformation for Ensemble Clustering in application to Medical Diagnostics. This clustering methodology is unique to outperform all other conventional techniques because of its linear time complexity. The work in [13] presents generation of Doppler Resilient waveforms using Golay Complementary sequence which have ideal ambiguity along the zero Doppler axis but are sensitive to non-zero Doppler shifts.

III. GOLAY CODE ALGORITHM

A binary Golay code is represented by (23, 12, 7) shows length of the codeword is 23-bits while message is of 12-bits and the minimum distance between two binary Golay code is 7. A matrix of binary sequence, Matrix-B, is used to generate extended Golay Code which is referred as (24, 12, 8). Where, length of the codeword is 24-bits while message is of 12-bits and the minimum distance between two binary Golay code is 8. The algorithm required to achieve the encoding procedure involves multiplication of the generator matrix by the encoding data (12-bit). The result of this matrix multiplication is a 24-bit Extended Golay Code, also referred as “codeword” in most of the literature works. The matrix B and Identity matrix of the same size (row –by- column) together are termed generator matrix ‘G’ when arranged as shown in Figure 2 and Figure 3. In verified G(24) codeword the weight is multiple of 4 and greater than equal to 8. If, the weight of the G(24) codeword is 12, so it is a valid codeword.



S[11] = w[23] xor w[11] xor w[10] xor w[8] xor w[7] xor w[6] xor w[2] xor w[0]

S[10] = w[22] xor w[11] xor w[9] xor w[8] xor w[7] xor w[3] xor w[1] xor w[0]

S[9] = w[21] xor w[10] xor w[9] xor w[8] xor w[4] xor w[2] xor w[1] xor w[0]

S[8] = w[20] xor w[11] xor w[10] xor w[9] xor w[5] xor w[3] xor w[2] xor w[0]

S[7] = w[19] xor w[11] xor w[10] xor w[6] xor w[4] xor w[3] xor w[1] xor w[0]

S[6] = w[18] xor w[11] xor w[7] xor w[5] xor w[4] xor w[2] xor w[1] xor w[0]

S[5] = w[17] xor w[8] xor w[6] xor w[5] xor w[3] xor w[2] xor w[1] xor w[0]

S[4] = w[16] xor w[9] xor w[7] xor w[6] xor w[4] xor w[3] xor w[2] xor w[0]

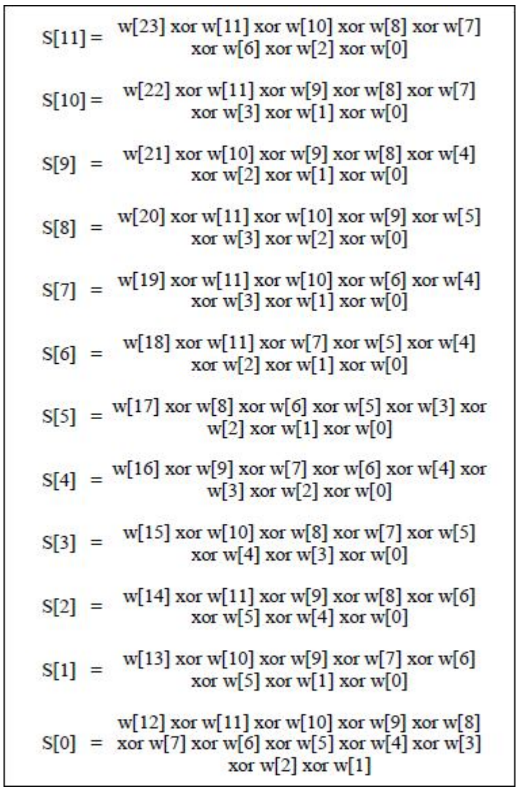
S[3] = w[15] xor w[10] xor w[8] xor w[7] xor w[5] xor w[4] xor w[3] xor w[0]

S[2] = w[14] xor w[11] xor w[9] xor w[8] xor w[6] xor w[5] xor w[4] xor w[0]

S[1] = w[13] xor w[10] xor w[9] xor w[7] xor w[6] xor w[5] xor w[1] xor w[0]

S[0] = w[12] xor w[11] xor w[10] xor w[9] xor w[8] xor w[7] xor w[6] xor w[5] xor w[4] xor w[3] xor w[2] xor w[1]

Figure 2. Matrix-B



S[11] = w[23] xor w[11] xor w[10] xor w[8] xor w[7] xor w[6] xor w[2] xor w[0]

S[10] = w[22] xor w[11] xor w[9] xor w[8] xor w[7] xor w[3] xor w[1] xor w[0]

S[9] = w[21] xor w[10] xor w[9] xor w[8] xor w[4] xor w[2] xor w[1] xor w[0]

S[8] = w[20] xor w[11] xor w[10] xor w[9] xor w[5] xor w[3] xor w[2] xor w[0]

S[7] = w[19] xor w[11] xor w[10] xor w[6] xor w[4] xor w[3] xor w[1] xor w[0]

S[6] = w[18] xor w[11] xor w[7] xor w[5] xor w[4] xor w[2] xor w[1] xor w[0]

S[5] = w[17] xor w[8] xor w[6] xor w[5] xor w[3] xor w[2] xor w[1] xor w[0]

S[4] = w[16] xor w[9] xor w[7] xor w[6] xor w[4] xor w[3] xor w[2] xor w[0]

S[3] = w[15] xor w[10] xor w[8] xor w[7] xor w[5] xor w[4] xor w[3] xor w[0]

S[2] = w[14] xor w[11] xor w[9] xor w[8] xor w[6] xor w[5] xor w[4] xor w[0]

S[1] = w[13] xor w[10] xor w[9] xor w[7] xor w[6] xor w[5] xor w[1] xor w[0]

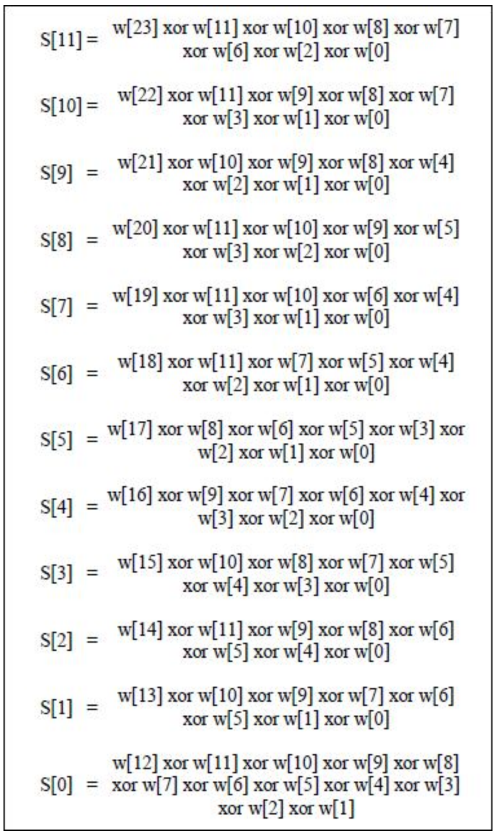
S[0] = w[12] xor w[11] xor w[10] xor w[9] xor w[8] xor w[7] xor w[6] xor w[5] xor w[4] xor w[3] xor w[2] xor w[1]

Figure 3. Generator Matrix

The steps of algorithm required to achieve the decoding process are as follows:

1) For the received codeword ‘W’ and matrix ‘H’, where H = [I / B] Compute the Syndrome ‘S’.
2) Error vector, E = [S, 0], If weight of ‘S’ is less than or equal to 3, i.e., $wt(S) \leq 3$.
3) If $wt(S+Bi) \leq 2$, then E = [S+Bi, Ii]. Where Ii represents ith row of the identity matrix I.
4) The second syndrome SB can be computed.
5) If $wt(SB) \leq 3$, then E = [0, SB].
6) If $wt(SB+Bi) \leq 2$, then E = [Ii, S+Bi].7) If E is still not determined then received data is required to be retransmitted.

The binary logical equations used to generate the syndrome S and S+B are represented in figure 4 and figure 5 respectively. The weight of the 12-bit syndrome is calculated using 3-bit adder (known as full adder), 3-bit decimal adder and 4-bit decimal adder. The block diagram of this is shown in figure 6.



S[11] = w[23] xor w[11] xor w[10] xor w[8] xor w[7] xor w[6] xor w[2] xor w[0]

S[10] = w[22] xor w[11] xor w[9] xor w[8] xor w[7] xor w[3] xor w[1] xor w[0]

S[9] = w[21] xor w[10] xor w[9] xor w[8] xor w[4] xor w[2] xor w[1] xor w[0]

S[8] = w[20] xor w[11] xor w[10] xor w[9] xor w[5] xor w[3] xor w[2] xor w[0]

S[7] = w[19] xor w[11] xor w[10] xor w[6] xor w[4] xor w[3] xor w[1] xor w[0]

S[6] = w[18] xor w[11] xor w[7] xor w[5] xor w[4] xor w[2] xor w[1] xor w[0]

S[5] = w[17] xor w[8] xor w[6] xor w[5] xor w[3] xor w[2] xor w[1] xor w[0]

S[4] = w[16] xor w[9] xor w[7] xor w[6] xor w[4] xor w[3] xor w[2] xor w[0]

S[3] = w[15] xor w[10] xor w[8] xor w[7] xor w[5] xor w[4] xor w[3] xor w[0]

S[2] = w[14] xor w[11] xor w[9] xor w[8] xor w[6] xor w[5] xor w[4] xor w[0]

S[1] = w[13] xor w[10] xor w[9] xor w[7] xor w[6] xor w[5] xor w[1] xor w[0]

S[0] = w[12] xor w[11] xor w[10] xor w[9] xor w[8] xor w[7] xor w[6] xor w[5] xor w[4] xor w[3] xor w[2] xor w[1]

Figure 4. Measurement of Syndrome ‘S’

$$S + b_1 = \{ \sim S[11], \sim S[10], S[9], \sim S[8], \sim S[7], \sim S[6], S[5], S[4], S[3], \sim S[2], S[1], \sim S[0] \}$$

$$S + b_2 = \{ \sim S[11], S[10], \sim S[9], \sim S[8], \sim S[7], S[6], S[5], S[4], \sim S[3], S[2], \sim S[1], \sim S[0] \}$$

$$S + b_3 = \{ S[11], \sim S[10], \sim S[9], \sim S[8], S[7], S[6], S[5], \sim S[4], S[3], \sim S[2], \sim S[1], \sim S[0] \}$$

$$S + b_4 = \{ \sim S[11], \sim S[10], \sim S[9], S[8], S[7], S[6], \sim S[5], S[4], \sim S[3], \sim S[2], S[1], \sim S[0] \}$$

$$S + b_5 = \{ \sim S[11], \sim S[10], S[9], S[8], S[7], \sim S[6], S[5], \sim S[4], \sim S[3], S[2], \sim S[1], \sim S[0] \}$$

$$S + b_6 = \{ \sim S[11], S[10], S[9], S[8], \sim S[7], S[6], \sim S[5], \sim S[4], S[3], \sim S[2], \sim S[1], \sim S[0] \}$$

$$S + b_7 = \{ S[11], S[10], S[9], \sim S[8], S[7], \sim S[6], \sim S[5], S[4], \sim S[3], \sim S[2], \sim S[1], \sim S[0] \}$$

$$S + b_8 = \{ S[11], S[10], \sim S[9], S[8], \sim S[7], \sim S[6], S[5], \sim S[4], \sim S[3], \sim S[2], S[1], \sim S[0] \}$$

$$S + b_9 = \{ S[11], \sim S[10], S[9], \sim S[8], \sim S[7], S[6], \sim S[5], \sim S[4], \sim S[3], S[2], S[1], \sim S[0] \}$$

$$S + b_{10} = \{ \sim S[11], S[10], \sim S[9], \sim S[8], S[7], \sim S[6], \sim S[5], \sim S[4], S[3], S[2], S[1], \sim S[0] \}$$

$$S + b_{11} = \{ S[11], \sim S[10], \sim S[9], S[8], \sim S[7], \sim S[6], \sim S[5], S[4], S[3], S[2], \sim S[1], \sim S[0] \}$$

$$S + b_{12} = \{ \sim S[11], \sim S[10], \sim S[9], \sim S[8], \sim S[7], \sim S[6], \sim S[5], \sim S[4], \sim S[3], \sim S[2], \sim S[1], S[0] \}$$

Figure 5. Measurement of Syndrome ‘S+B’

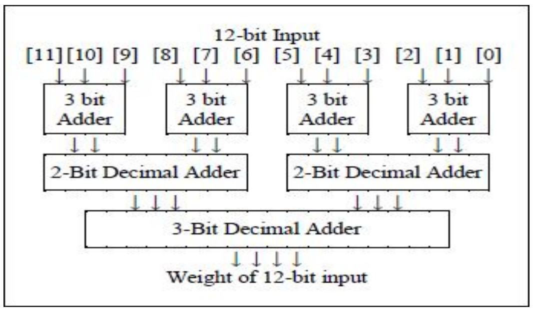
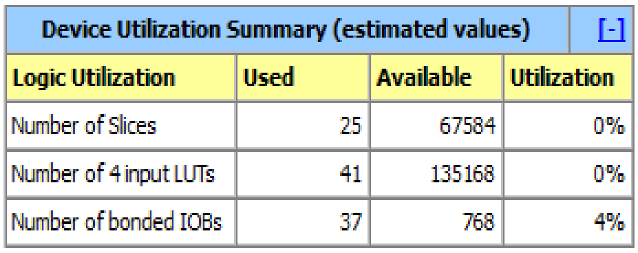


Figure 6. Calculation of Weight of Syndrome

IV. SIMULATION AND SYNTHESIS RESULTS

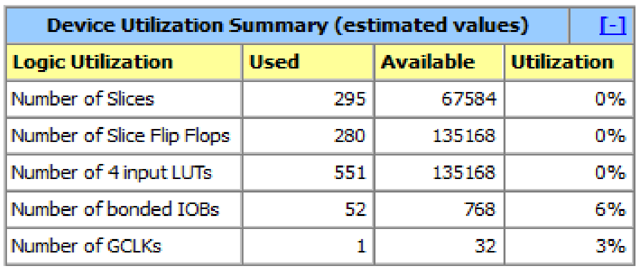
Xilinx software is used in the present work. Table–I and Table–II respectively represents the FPGA based hardware utilization summary of the proposed Encoder and Decoder designs for Xilinx Virtex-4 4VLX160FF1148-12 FPGA device. Maximum combinational path delay in the encoder design is 6.584ns. In the decoder design the Maximum Frequency is 226.164MHz.

TABLE I. HARDWARE UTILIZATION SUMMARY OF ENCODER



Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	25	67584	0%
Number of 4 input LUTs	41	135168	0%
Number of bonded IOBs	37	768	4%

TABLE II. HARDWARE UTILIZATION SUMMARY OF DECODER



Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	295	67584	0%
Number of Slice Flip Flops	280	135168	0%
Number of 4 input LUTs	551	135168	0%
Number of bonded IOBs	52	768	6%
Number of GCLKs	1	32	3%

The RTL Schematic diagrams of Encoder and Decoder designs are shown in figure 7 and figure 8 respectively.

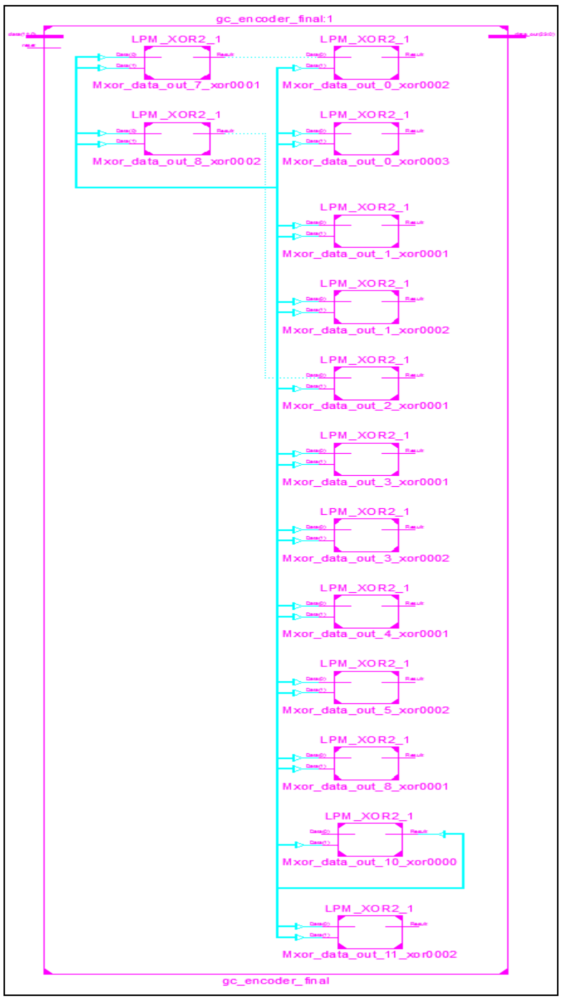
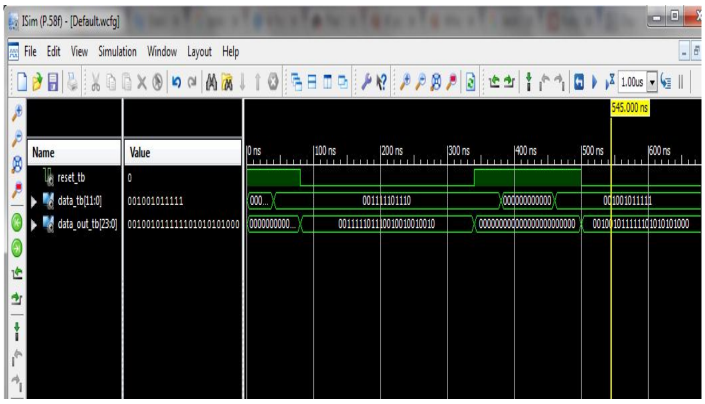


Figure 7. RTL Schematic Diagram of Proposed Extended Golay Code (24, 12, 8) Encoder

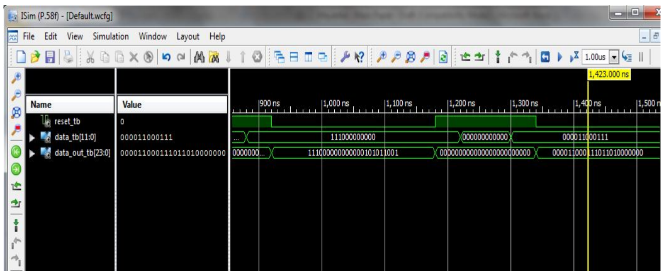
Figure 8. RTL Schematic of Proposed Extended Golay Code (24, 12, 8) Decoder

The Encoder and Decoder simulation waveforms are shown in figure 9 and figure 10 respectively. A 12-bit data is used to encode using the proposed encoder. The input data bits are followed by logic-'0' inputs.



Name	Value
reset_tb	0
data_tb[11:0]	001001011111
data_out_tb[23:0]	001001011111101010101000

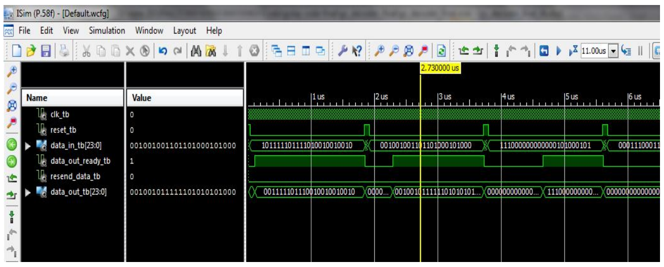
(A)


ISim (P.58f) - [Default.wcfg]
File Edit View Simulation Window Layout Help
1,423.000 ns

```
Name               Value
reset_tb           0
data_tb[11:0]      000011000111
data_out_tb[23:0]  000011000111011010000000
```

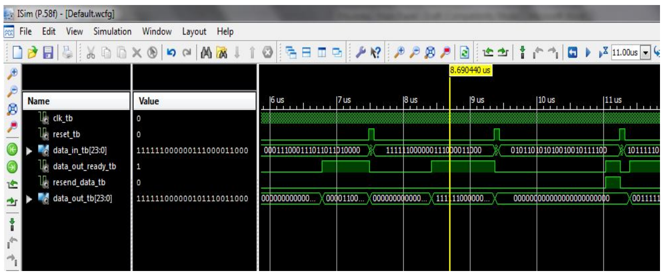
(B)

Figure 9. Encoder Simulation Waveform for Proposed Golay Code (24, 12, 8) Encoder


ISim (P.58f) - [Default.wcfg]
File Edit View Simulation Window Layout Help
2.730000 us

```
Name                Value
clk_tb              0
reset_tb            0
data_in_tb[23:0]    001001001101101000101000
data_out_ready_tb   1
resend_data_tb      0
data_out_tb[23:0]   001001011111101010101000
```

(A)


ISim (P.58f) - [Default.wcfg]
File Edit View Simulation Window Layout Help
8.690440 us

```
Name                Value
clk_tb              0
reset_tb            0
data_in_tb[23:0]    111111000000111000011000
data_out_ready_tb   1
resend_data_tb      0
data_out_tb[23:0]   111111000000101110011000
```

(B)

Figure 10. Encoder Simulation Waveform for Proposed Golay Code (24, 12, 8) Decoder

V. CONCLUSION

In the proposed work hardware optimized architecture of extended binary Golay encoder and decoder are designed and simulated. The consequences obtained from the design combination for encoder and decoder supersedes the reference schemes in term of the operational frequency. This makes the proposed design a good quality option to be used in the high rate application based configurable circuits. In future there is a enormous scope to further optimize the performance of the proposed algorithm. In future the scholars may assume the challenge to reduce the ratio of overhead bits versus data bits in the encoded codeword. Or the researchers might increase the length of the data word that can be encoded using the same algorithm with the same or better error detection and Correction capacity.

REFERENCES

[1] Satyabrata Sarangi and Swapna Banerjee, “Efficient Hardware Implementation of Encoder and Decoder for Golay Code”, IEEE Transaction on very large scale Integration (VLSI) system, Vol.23 Issue No.9, pg.1965-1968, September 2015.

[2] Marcel J. E. GOLAY, “Notes on Digital Coding”, Reprinted from proc. IRE, Vol.37, pg-657 June 1949.

[3] Mario de Boer and Ruud Pellikaan, “The Golay codes” Springer, pg.338-347, September 1995.

[4] Dr. Ravi Shankar Mishra, Prof Puran Gour and Mohd Abdullah, “Design and Implementation of 4 bits galois Encoder and Decoder in FPGA”, International Journal of Engineering Science and Technology (IJEST), Vol.3 No.7, pg.5724-5732, July 2011.

[5] Dongfu Xie, “Simplified algorithm and hardware implementation for the (24,12,8) Extended golay soft Decoder up to 4 Errors”, The International Arab Journal of Information Technology, Vol.11 No.2, pg.111-115, March 2014.

[6] Xiao-Hong Peng and Paddy G. Farrell” On Construction of the (24, 12, 8) Golay Codes”, December 2005.

[7] Matthew G. Parker, Kenneth G. Paterson and Chintha Tellambura, “Golay Complementary Sequences”, January 2004.

[8] Yan-Haw Chen, Chih-Hua Chine, Chine-Hsiang Huang, Trieu- Kien Truong And Ming-Haw Jing, “Efficient Decoding of schematic (24,12,7) and (41,21,9) Quadric Residue codes”,Journal of Information science And Engineering Vol.26, pg.1831-1843, December 2010.

[9] Li Ping and Kwan L. Yeung, “Symbol-by-Symbol APP Decoding of the Golay Code and Iterative Decoding of Concatenated Golay Codes”, IEEE Transaction on Informationtheory, Vol.45, No.7, pg.2558-2562, November 1999.

[10] Yihua Chen, Juehsuan Hsiao, PangFu Liu and Kunfeng Lin, “Simulation and Implementation of BPSK BPTC of MSK golay code in DSP chip”, Communications in Information Science and Management Engineering, Vol.1 No.4, pp.46-54, Nov.2011

[11] Eyas El-Qawasmeh, Maytham Safar and Talal Kanan, “Investigation of golay code (24,12,8) Structure in improving search techniques”, The International Arab Journal of Information Technology, Vol.8, No.3, pg.265-271, July 2011.

[12] Faisal Alsaby, Kholood Alnoowaiser and simon Berkovich, “Golay code Transformation for ensemble clustering in application of medical Diagnostics”, International Journal of Advanced Computer Science and Applications (IJACSA), Vol.6 No.1, pg.49-53, 2015.

[13] Ali Pezeshki, A. Robert Calderbank, William Moran andStephen D. Howard, “Doppler Resilient Golay Complementary Waveforms”, IEEE Transaction on Information Theory Vol.54, No.9 , pg.4254-4266, September 2008.