

# Implementation of Effective Error Correction Architecture for (24,12) Extended Golay Code

Nimmagadda Sravani<sup>1</sup>, Mulukutla Chaitanya<sup>2</sup>

PG Scholar, Department of ECE (VLSI&ES), GPREC (Autonomous), Kurnool, AP, India<sup>1</sup>

Assistant Professor, Department of ECE, GPREC (Autonomous), Kurnool, AP, India<sup>2</sup>

**ABSTRACT:** With the increase in the technological advancements we are now capable of designing and using high speed communication systems and memory systems with very huge capacities. All these systems work on binary data and there is a great need to protect this information from being corrupted for which we are now using Error Correcting Codes (ECCs). The (24,12) Extended Golay Code is one of the widely used half rate linear error correcting code with significant properties. This paper presents an efficient way of implementing the necessary architecture for the encoder and decoder required for the Extended Golay Code to rectify single and double bit faults. The architectures are described with the simulation and synthesis results.

**KEYWORDS:** Binary Golay Code, Extended Binary Golay Code, Long division algorithm

## I. INTRODUCTION

The various devices that are being used in numerous fields are predominantly working on binary data. In the present world, need for the huge storage devices and high speed communications is increasing rapidly. When these systems work under harsh environments there is a great chance for the data to get corrupted.

The errors which do not cause any damage to the hardware, but change the instruction or the data are known as soft errors, usually observed due to the material decay or noise phenomenon [1]. Error correction codes have been frequently used to improve the reliability of these systems.

Advances in semiconductor technology have fetched very high levels of integration, predominantly observed in VLSI memories. A review of error correcting codes for VLSI storage devices is presented in [2]. With the surge in the usage of highly compact memories the possibility of the no. of bits getting corrupted has also increased, leading to multiple bit errors.

The current developments in this field are analysed. The set of techniques in paper [3] are used for single-error rectification and double error identification. An interleaving scheme is proposed in [5] which is effective with multiple bit errors. A report on the geometric effect of multiple bit soft errors induced by the neutrons is presented in paper [6]. The concept of exploring the check bit design space and selective bit placement to design SEC-DED codes is presented in [7]. A mathematical model for bit-error-rate is discussed in [8].

There are many techniques used in the ECCs and one such method is the addition of the parity check bits to the message bit. This concept of parity bits is explained in the Fig1. The addition of parity bits introduces a trade-off between memory size and decoding time, which is to be balanced according to the requirement.

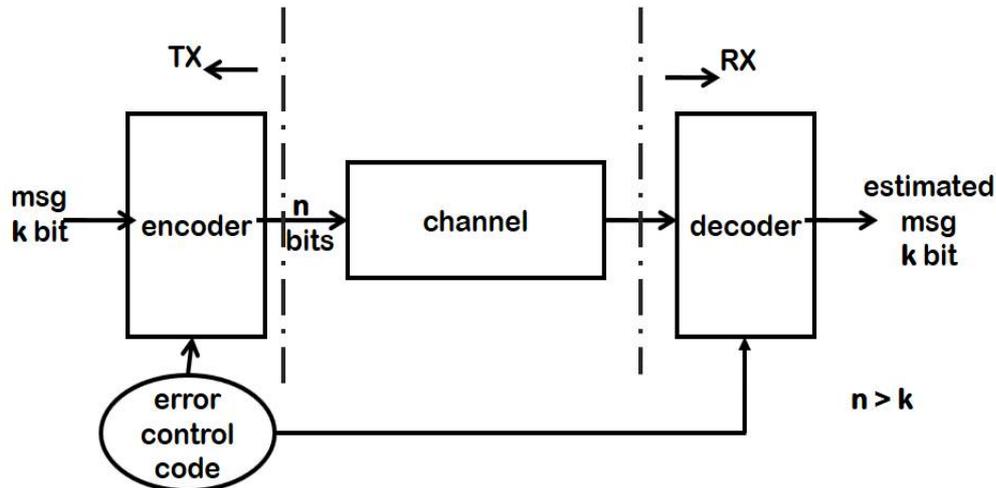


Fig 1: Concept of encoding and decoding using parity bits

## II. GOLAY CODE

The binary Golay Code is represented as (23, 12, 7) which indicates that the length of the code word is 23 bits, message being 12 bits and the minimum distance between two binary Golay Codes is seven.

The extended Golay Code (24, 12, 8) can be created by appending a parity bit to the binary Golay Code. The extended code has a minimum distance of eight, and therefore can rectify 3-bit errors and identify 4-bit errors.

The addition of parity bits is preferred despite the addition of overheads in memory and time as the no. of code words with  $n$ -bits are less than the possible words of  $n$ -bits, allowing to differentiate between the message word and erroneous words. This concept is explained in Fig 2.

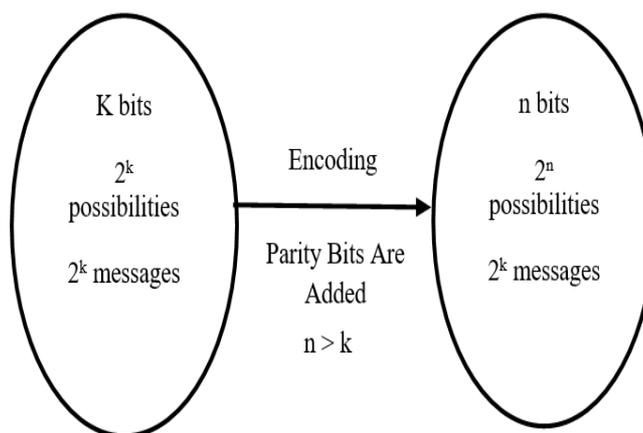


Fig 2: Necessity of parity bits

The generation of the coding sequence needs a generator polynomial. In this paper,  $AE3h$  is considered as the generator polynomial.



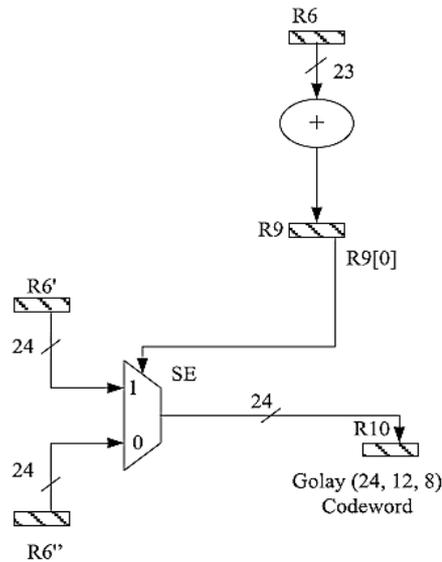


Fig 4: architecture for the conversion of binary Golay code to extended Golay code

#### IV. DECODING TECHNIQUE

The decoder designed here is used to identify and rectify single and double bit errors. The idea is illustrated in Fig 5. In this procedure, the first twelve bits of the received codeword are separated (these bits form the message bits) and are sent to the encoder to reconstruct the codeword. Now both the code words, received and reconstructed, are divided into three sets with 8-bits in each set. XOR gate is used to compare the respective bits of the received codeword and the reconstructed codeword of each set separately. If both the sets are equal, then no error has occurred, otherwise we flip the bit of the corresponding set in the respective position which is corrupted.

In this paper an erroneous input is given to the decoder as A27873h instead of A2786Bh. the decoder identified the error and rectified it to A2786Bh by flipping the corresponding bits. With this our requirement of identifying and rectifying errors in binary systems is achieved.

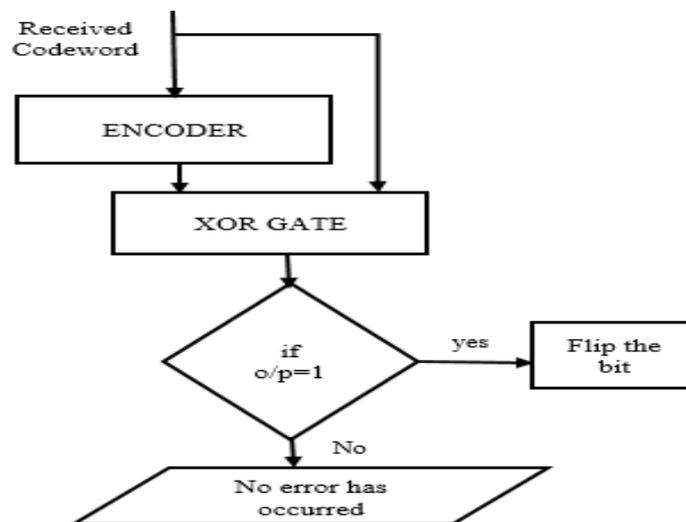


Fig 5: Illustrative Representation of the Decoding Technique

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

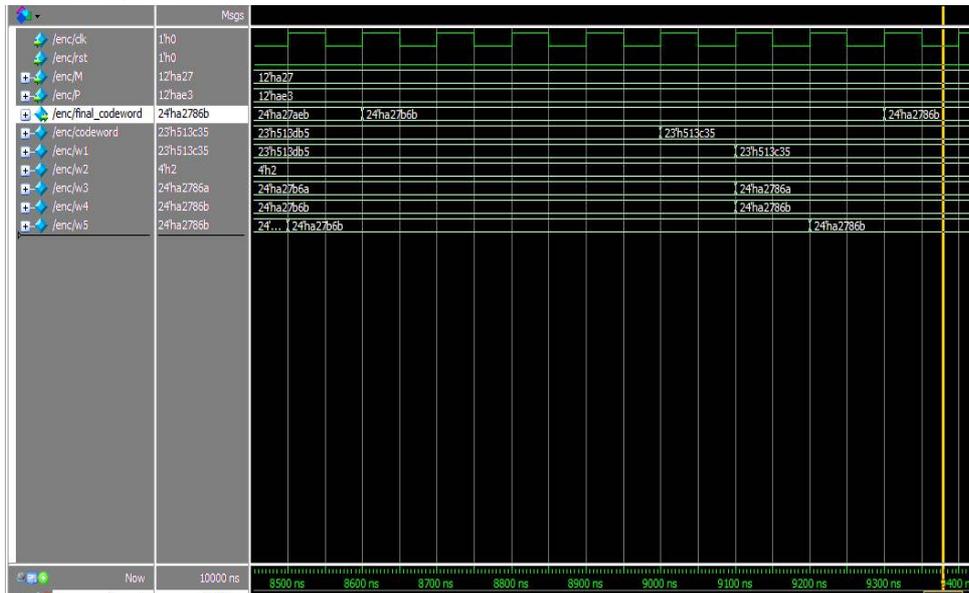
Vol. 6, Issue 8, August 2017

## V. RESULTS

The simulation and synthesis of the techniques described in the paper are performed using modelsim and Xilinx tools respectively. The following are the screenshots of the obtained results.

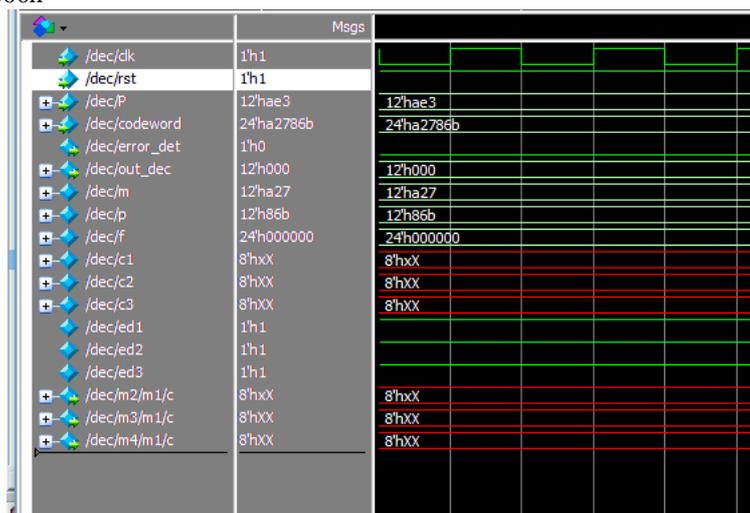
### Output of the encoder

A27h is encoded as A2786Bh



### When reset is triggered

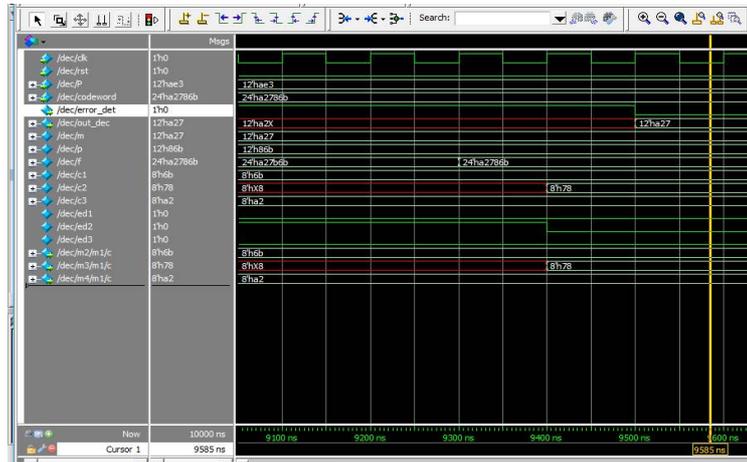
The decoder output is 000h



### Decoder output when no error has occurred

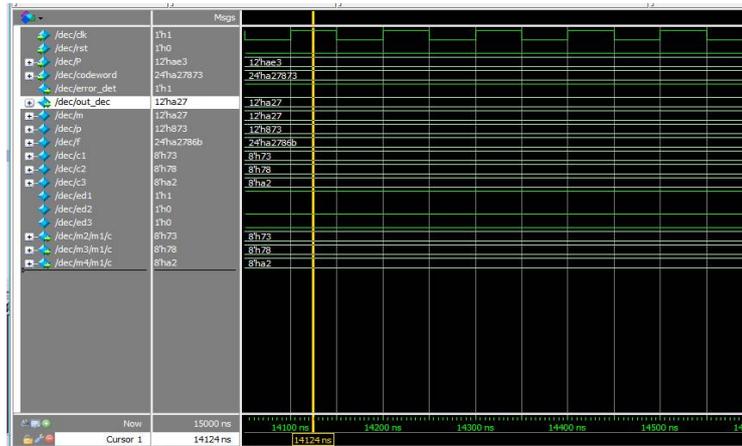
A27h is the encoder input, A2786Bh is decoder input. No error is detected. Value at pin error\_det is 0.

Decoder output is A27h

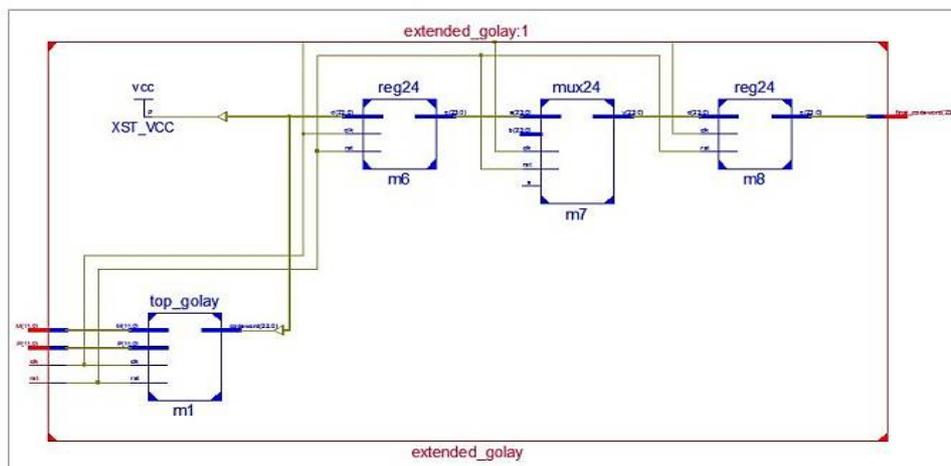


### Decoder output when error has occurred

A27h is the encoder input, A27873h is the decoder input. Error is detected. Value at pin error\_det is 1  
Decoder output is A27h



### RTL Schematic



# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 8, August 2017

## Design summary:

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	510	4656	10%
Number of Slice Flip Flops	684	9312	7%
Number of 4 input LUTs	665	9312	7%
Number of bonded IOBs	50	232	21%
Number of GCLKs	1	24	4%

## Timing report

```

Data Path: m8/q_23 to final_codeword<23>
      Gate      Net
Cell:in->out  fanout  Delay  Delay  Logical Name (Net Name)
-----
FDR:C->Q      1    0.514  0.357  m8/q_23 (m8/q_23)
OBUF:I->O      3.169      final_codeword_23_OBUF (final
-----
Total                                4.040ns (3.683ns logic, 0.357ns route)
                                       (91.2% logic, 8.8% route)

```

## VI. CONCLUSION

Effective hardware architecture for both encoder and decoder of the extended binary Golay code have been designed and implemented after verifying the proposed algorithm. The algorithm discussed in this paper is better than single error correcting algorithms as this scheme can correct double bit errors. By exploring the properties of the Golay code we can further extend this scheme to correct three bit errors or detect four bit errors.

## REFERENCES

- [1] R. C. Baumann, "Soft errors in advanced computer systems," IEEE Des. Test. Comput., vol. 22, no. 3, pp. 258– 266, May/Jun. 2005.
- [2] C. L. Chen and M. Y. Hsiao, "Error-correcting codes for semiconductor memory applications: A state-of-the-art review," IBM J. Res. Develop., vol. 28, no. 2, pp. 124–134, Mar. 1984.
- [3] M. Y. Hsiao, "A class of optimal minimum odd-weight-column SEC-DED codes," IBM J. Res. Develop., vol. 14, no. 4, pp. 301–395, Jul. 1970.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 8, August 2017

- [4] E. Ibe, H. Taniguchi, Y. Yahagi, K. Shimbo, and T. Toba, "Impact of scaling on neutron-induced soft error in SRAMs from a 250 nm to a 22 nm design rule," *IEEE Trans. Electron Devices*, vol. 57, no. 7, pp. 1527–1538, Jul. 2010.
- [5] P. Reviriego, J. A. Maestro, S. Baeg, S. Wen, and R. Wong, "Protection of memories suffering MCUs through the selection of the optimal interleaving distance," *IEEE Trans. Nucl. Sci.*, vol. 57, no. 4, pp. 2124–2128, Aug. 2010.
- [6] S. Satoh, Y. Tosaka, and S. A. Wender, "Geometric effect of multiple-bit soft errors induced by cosmic ray neutrons on DRAM's," *IEEE Electron Device Lett.*, vol. 21, no. 6, pp. 310–312, Jun. 2000.
- [7] A. Neale and M. Sachdev, "A new SEC-DED error correction code subclass for adjacent MBU tolerance in embedded memory," *IEEE Trans. Device Mater. Rel.*, vol. 13, no. 1, pp. 223–230, Mar. 2013.
- [8] M. A. Bajura et al., "Models and algorithmic limits for an ECC-based approach to hardening sub-100-nm SRAMs," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 4, pp. 935–945, Aug. 2007.
- [9] S. Sarangi and S. Banerjee, "Efficient hardware implementation of encoder and decoder for Golay code," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, No. 9, September 2015