An Efficient Double and Tripple-Adjacent Error Correcting Parallel Decoder for the (24, 12) Extended Golay Code

Angela Prasanna Raj and R. Dharmalingam

Abstract--- As multiple cell upsets (MCUs) become more frequent on SRAM storage devices, there is a growing interest on error correction codes that can correct multibit errors. MCUs affect cells that are close together, and hence the numbers of codes that can correct double-adjacent or triple-adjacent errors have been proposed. These codes, in many cases, do not require additional parity check bits and in the rest require only one or two additional bits. The decoding complexity improves but in many cases can still be invoked with limited impact on the memory speed. These codes are useful for applications in which the error rate is low, however, when the error rate is high, codes that can correct errors on multiple independent bits are needed. The proposed decoder is also able to correct triple-adjacent errors, thus covering the most common error patterns.

Keywords--- Double Adjacent Error Correction (DAEC), Error Correction Codes (ECCs), Golay Code, Memory, Single Error Correction (SEC), Triple-adjacent Error Correction.

I. INTRODUCTION

In recent years, the number of errors that affect more than one memory cell has increased significantly. This is due to the scaling of the memory cells and is projected to grow further. These errors, known as multiple cell upsets (MCUs), pose a challenge for SEC–DED codes. One solution to ensure that the MCU errors can be corrected is to interleave the bits of different logical words so that an MCU affects one bit per word. This is based on the observation that the cells affected by an MCU are physically close. Interleaving, however, has a cost as it complicates the memory design. In some space applications, there is an additional issue as the number of errors is high, and SEC–DED codes may not be sufficient when errors accumulate over time. These issues have led to an increased interest on the use of more advanced ECCs to protect SRAM memories. As MCUs affect cells that are close together, a number of codes that can correct double-adjacent or triple-adjacent errors have been recently proposed. These codes, in many cases, do not require additional parity check bits and in the rest require only one or two additional bits. The decoding complexity increases but in many cases can still be implemented with limited impact on the memory speed. These codes are useful for applications in which the error rate is low, however, when the error rate is large, codes that can correct errors on multiple independent bits are needed. Research for multibit ECCs has focused on reducing the decoding latency as in many cases, the traditional decoders are serial and require several clock cycles. To some extent this can be done for some traditional ECCs by using a parallel syndrome decoder but the decoder complexity explodes as the error correction capability or the word size increases. Another approach is to use codes that can be decoded with low delay, such as orthogonal Latin squares (OLSs) or difference set (DS) codes. In the case of OLS codes, the main issue is that they are not optimal in terms of the number of parity check bits and thus require more memory overhead.

Angela Prasanna Raj, M.E, Department of VLSI, Maharaja Institute of Technology, Coimbatore.
R. Dharmalingam, HOD, Department of VLSI, Maharaja Institute of Technology, Coimbatore.

With the rapid growth of digital communications, such as Digital Audio Broadcasting (DAB) and ATM methods, increased data rate and advanced error control coding techniques are required. Thus, the parallelism inherent in the decoding algorithm and the area-adequate high-speed VLSI architectures must be exploited. The (24,12,8) extended Golay code is a well-known error-correcting code, which has been successfully applied in several existing transactions systems to improve the system bit-error-rate (BER) performance. One goal of this research was to provide a strong error protection for the important head data in the transmission of the high quality compressed music signal of the DAB system.

The parallel Golay decoder can be, of course, used generally to protect the data communication or memory space against channel errors for high speed data processing. A number of soft-decision decoding of the (24,12) binary Golay code were intensively investigated in the last few years and detailed search of computational complexity were discussed. However, none of these algorithms have been realized efficiently with parallel VLSI circuits. This paper introduces a full parallel permutation decoding algorithm with look-ahead error-correction and a fast soft-decision decoding for (24, 12, 8) extended Golay code. The area-efficient parallel VLSI architectures and the computer simulation results are also presented

II. APPROACHES

Multiple Cell Upsets (MCUS)

The security code utilizes decimal method to detect errors, so that more errors were detected and corrected. At the present days to maintain best level of acceptability, it is required to protect memory cells using protection codes, for this purpose, various error detection and correction methods are being used. In the paper 64-bit Decimal Matrix Code was exposed to assure the dependability of storage. Here to detect and correct up to 50% errors. The results showed that the proposed scheme has a protection level against large MCUs in memory. To avoid MCUs from causing information corruption, more complex error correction codes (ECCs) are widely used to protect memory, but the main problem is that they would wanted a higher delay overhead. Previously, matrix codes (MCs) based on Hamming codes include been proposed for memory safety. The main problem is that they are double error correction codes and the error correction capabilities are not enriched in each case. Transient multiple cell upsets (MCUs) are appropriate major problems in the reliability of memories exposed to energy environment.

III. EXISTING SEC-DAEC PARALLEL DECODER

The existing system consist of an encoding-decoding scheme which encodes 12 bit data into 24 bit data by adding 12 check bits to the input. Single and double adjacent (DAEC) errors can be detected and corrected by the use of these extra check bits. The existing SEC-DAEC decoders are similar to SEC decoders but they need to check also the syndrome values that correspond double adjacent errors. This requires roughly doubling the number of comparisons. Then, the correction of each bit is triggered by three syndrome values (the single bit and the two double adjacent). This results in a decoder that is significantly more complex than a simple SEC decoder. Each error has a unique syndrome value, which leads to detection and correction of errors within the range.

The parallel decoder as discussed before has the objective of correcting single and double-adjacent bit errors. The first step is to place the bits in the memory such that data and parity bits are interleaved. This interleaving has no impact on memory performance, as it is a simple remapping of the bits when they are read from or written to the memory.

The requirement for SEC is that the columns must be different. Therefore, it would seem possible to use a subset of the parity bits to decode single errors. However, since the code can correct 3 errors, we want to assure that the single-error parallel decoder does not introduce erroneous

corrections in the presence of multiple bit errors. For example, if we use an SEC-DAEC code with a minimum length of four, a triple error can cause a miss-correction in the SEC-DAEC decoding phase. A 4-bit error may not be even detected by the SEC-DAEC decoder. Therefore, the full syndrome is used for comparisons in all the situations to ensure that triple errors do not trigger mis-corrections and 4-bit errors are detected.

IV. GOLAY CODE

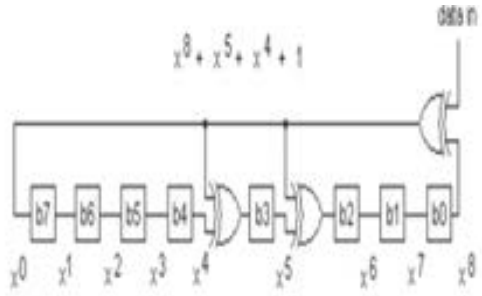
The Binary Golay code is represented as (23, 12, 7) that depicts that length of codeword is 23 bits, while message is of 12 bits and the minimum distance between two binary Golay codes is 7.

$$A=\begin{bmatrix}0&1&1&0&1&1&1&1&1&1&1&1\\1&1&0&1&1&1&1&1&1&1&1&0\\1&0&1&1&1&1&1&1&1&1&0&1\\1&0&1&1&1&0&0&0&1&0&1&1\\1&1&1&1&0&0&0&1&0&1&1&0\\1&1&1&0&0&0&1&0&1&1&0&1\\1&1&0&0&0&1&0&1&1&0&1&1\\1&0&0&0&1&0&1&1&0&1&1&1\\1&0&0&1&0&1&1&0&1&1&1&0\\1&0&1&0&1&1&0&1&1&1&0&0\\1&1&0&1&1&0&1&1&1&0&0&0\\1&0&1&1&0&1&1&1&0&0&0&1\end{bmatrix}$$

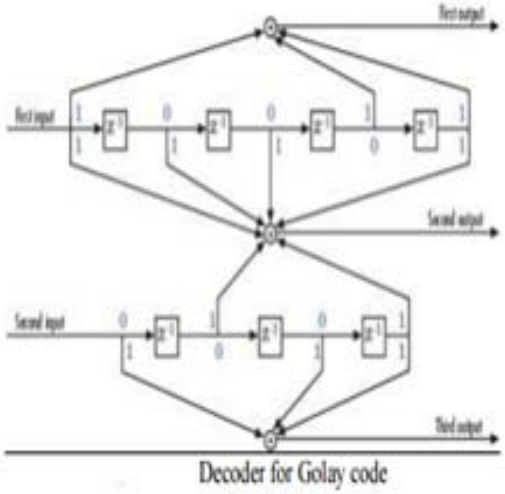
A Galois field (GF) is necessary to construct binary codes. In general, binary field is denoted by GF (2), which supports different binary arithmetic operations. The generation of coding flow needs a generator polynomial. The possible generator polynomials [13] over GF (2) for Golay (23, 12, 7) code are x11 + x10 + x6 + x5 + x4 + x2 + x1 and x11+x9+x7+x6+x5+x1+1. In this brief, AE3h is considered as the characteristic polynomial. The carry-over of the long division gives the required check bits. Finally, appending the generated check bits with the message gives us the extended Golaycodeword. The extended Golay code (24, 12, 8) can be executed by appending a parity bit with the binary Golay code or using a generator matrix G, which is identified as [I, B], where I represents an identity matrix of order 12.

Golay Code of encoder and Decoder

A Galois field (GF) is necessary to construct binary codes. In general, binary field is represented by GF (2), which supports different binary arithmetic operations. The execution of coding sequence needs a generator polynomial. The possible generator polynomials over GF (2) for Golay (23, 12, 7) code are x11 + x10 + x6 + x5 + x4 + x2 + x1 and x11+x9+x7+x6+x5+x1+1.



Block Diagram of Encoder for Golay Code



Decoder for Golay code

In this brief, AE3h is considered as the characteristic polynomial. The carry-over of the long division gives the required check bits. Finally, appending the generated check bits with the message gives us the extended Golaycodeword. The extended Golay code (24, 12, 8) can be executed by appending a parity bit with the binary Golay code or using a generator matrix G, which is defined as [I, B], where I denotes an identity matrix.

The Steps required to Accomplish the Encoding Procedure are Enlisted as Follows

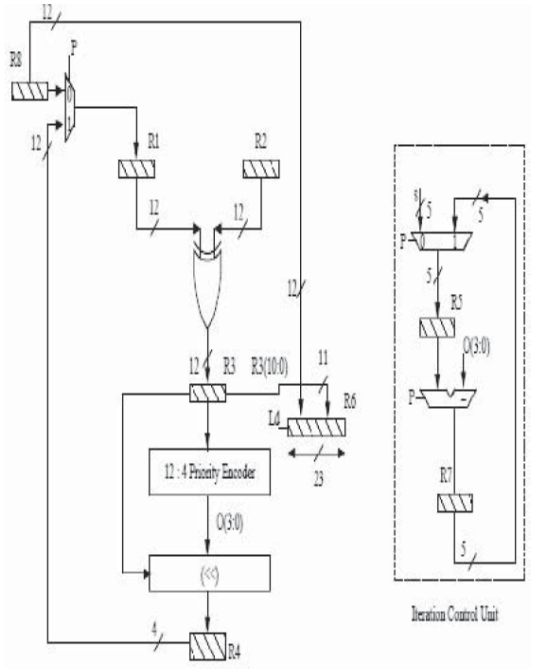
1. A characteristic polynomial $G(x)$ is chosen for check bits execution.
2. 11 zeros are appended to the right of message $M(x)$, like that resultant polynomial $P(x)$ participates in long division process with $G(x)$.
3. The remainder bits except the most significant bit (MSB) resulted at the end of the division execution are the check bits for $G23$.Appending check bits with the message gives us the encoded Golay (23, 12, 7) Codeword
4. A parity bit is added to convert the binary Golay Code into showed as a binary Golay Code (24, 12, 8). If the weight of binary Golay Code is even, then parity bit 0 is appended, otherwise 1 is appended.

Example of check bits generation.

An example of GolayCodeword generation based on the encoding algorithm is shown in Fig. 1. Let us say, the data to be encoded is A27h. Hence, $M(x)$ = A27h and $P(x)$ in binary format is showed as 1010 0010 0111 0000 0000 000. Finally, the generated check bits in hexadecimal format are 435h. Hence, the encoded codeword for the data bits (A27h) is A27435h. This is a binary GolayCodeword. To convert it into an extended Golay Code, a parity bit 1 is appended, as weight of A27435h is 11 (odd). Finally, the generated Golay (24, 12, 8) Codeword is (1010 0010 0111 1000 0110 101 1). The validity of the executed Golay Code can be tested by measuring the weight of the code.

V. PROPOSED METHODOLOGY

In each step during polynomial division, simple binary XOR operation occurs for modulo-2 subtraction. The residual result got at each step during the division process is circularly left shifted by number of leading zeros present in the result. A 12:4 priority encoder is used to detect efficiently the number of leading zeros before first 1 bit in the residual result in each step. A circular shift register is used to shift the intermediate result by the output of priority encoder 2:1 multiplexer is used to select the initial message or the circularly shifted intermediate result. The control signal used for the multiplexer and the controlled subtractor is represented as *p*, which is bit wise OR operation of priority encoder output. A controlled subtractor is used for loop control mechanism. Initially, one input of subtractor is intimated with 11, which is the number of zeros appended in the first step of the long division process and it gets updated with the content of *R*7 register due to multiplexer chosen after each iteration. The output of the priority encoder is the other input to the subtractor. After the final iteration, the result of subtractor is zero, which is stored in register *R*7. The register *R*6 is loaded when the content of register *R*7 becomes zero, which depicts the end of the division process and hence the check bits generation process. Architecture for decoding extended Golay Code consist of syndrome measurement, weight measurement, priority encoder and multiplexer to select the register. Satyabrata Sarangi and Swapna Banerjee proposed the structure of weight measurement unit that consists of 2-bit and 3-bit ripple carry adder. Ripple Carry Adder consumes large area and induces more delay as compared to Common Boolean Logic (CBL) adder and Kogge-Stone Logic (KSL) adder. Thus to overcome the mentioned problems we will use CBL and KSL in our model.



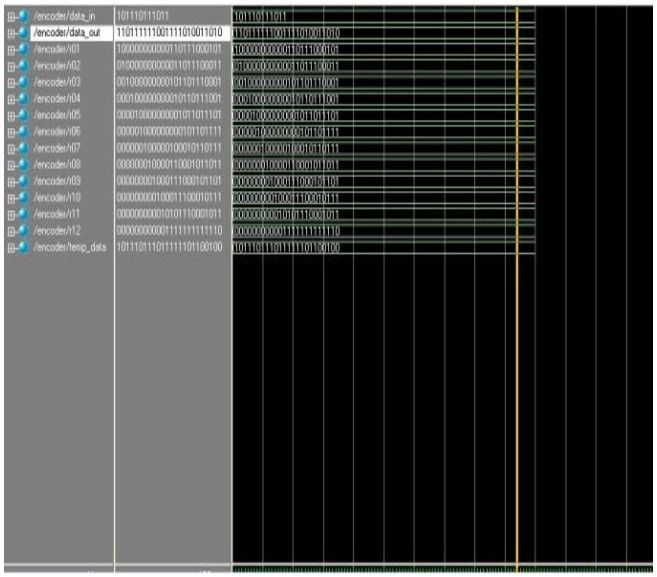
In this method each error can be identified by finding a value called Syndrome at the decoder. Every error has a unique Syndrome value, which leads to detecting and correcting of errors within the range. By finding the Syndrome value of Tripple adjacent errors (TAEC), we can extend our correctable range to 3 from two. These syndromes values can be calculated by making small changes at the decoder. In the proposed system check bits and syndrome values can be calculated by the use of various circuits. XOR gate is the major gate used in both encoder and decoder. XOR is a circuit made from the basic gates (AND, OR, NOT), which has 5 gates. we can replace these XOR gate by a circuit arrangement which has 4 gates. By using these gate in the system area, power and delay can be upgrade.

VI. RESULTS AND DISCUSSION

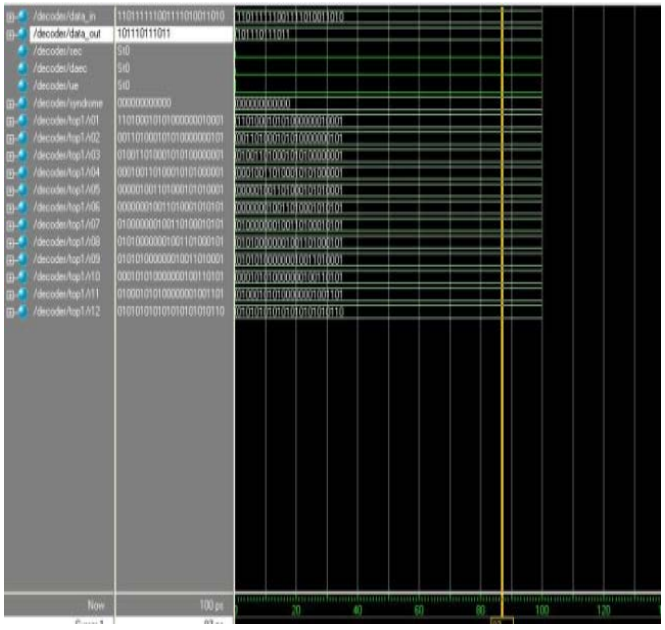
Error detection and correction helps in transmitting errorless data in a noisy channel. Error detection refers to find errors if any received by the receiver and correction is to correct errors got by the receiver. Different errors correcting codes are there and can be used depending on the properties of the method and the application in which the error correcting is to be intialized. Generally error correcting codes have been classified into Block Codes, Convolutional Codes, Low Density Parity Check Code (LDPC) and Golay Code. The purpose of this thesis is to review the published encoding and decoding models in the literature and to critique their reliability effects. We will try to reduce the area, Maximum Combinational Path Delay (MCPD) of decoding algorithm of Golay Code

Existing-Single and Double Adjacent Error Correction

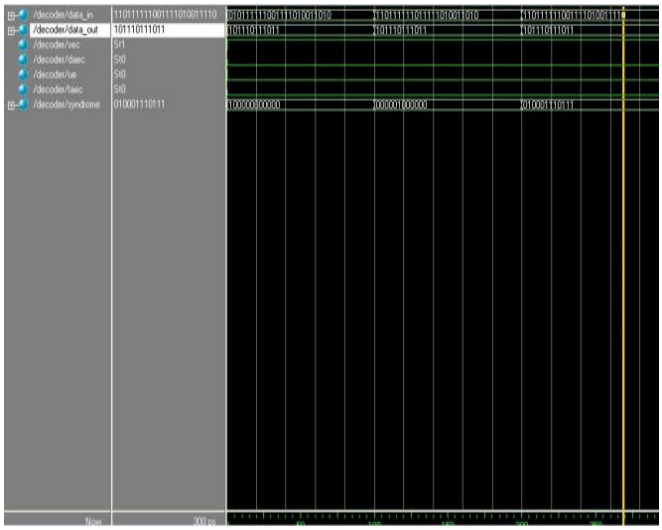
A. Encoder



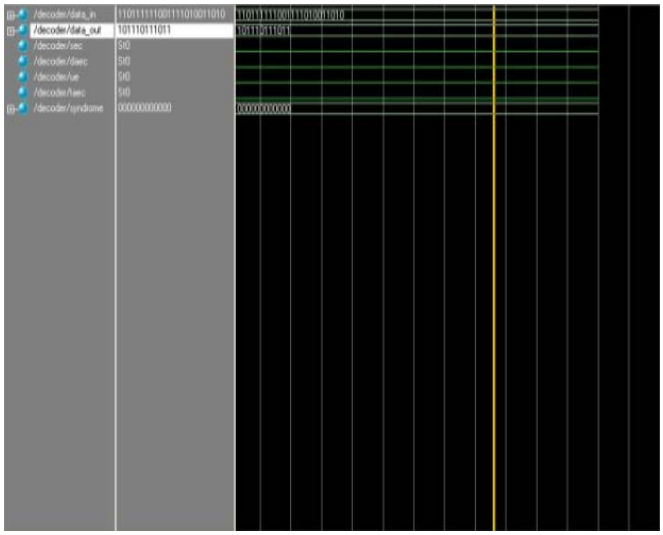
B. Decoder - No Error



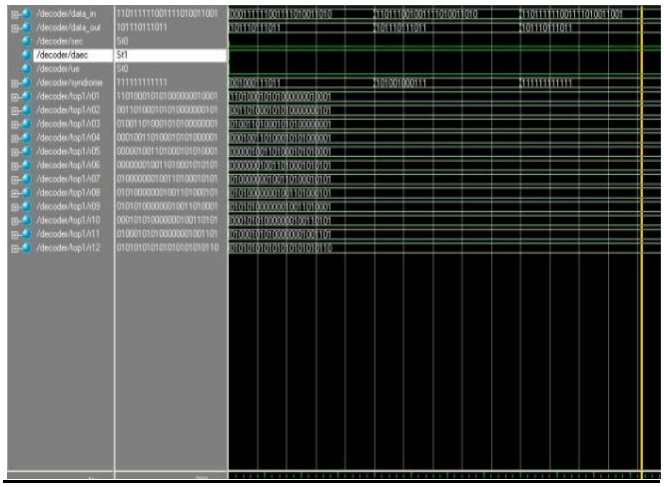
C. Decoder - Single Error Correction



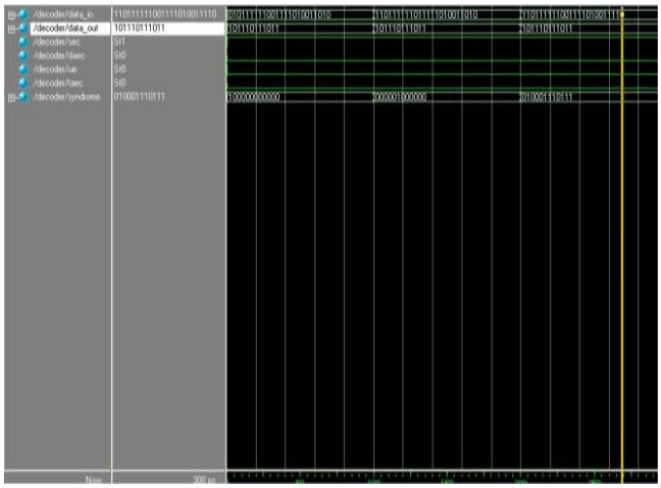
B. Decoder - No Error



D. Decoder - Double Adjacent Error Correction

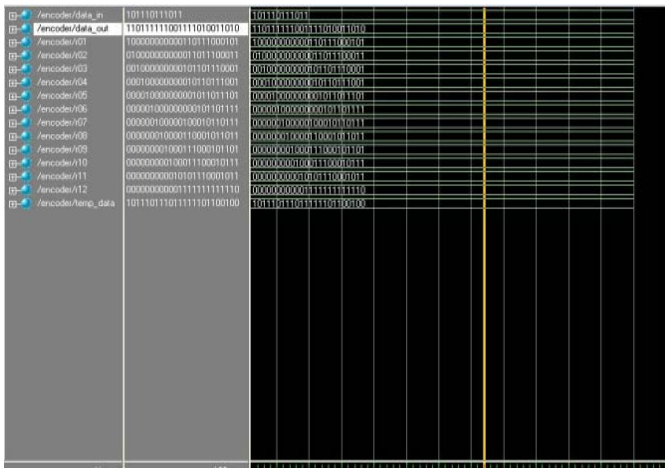


C. Decoder - Single Error

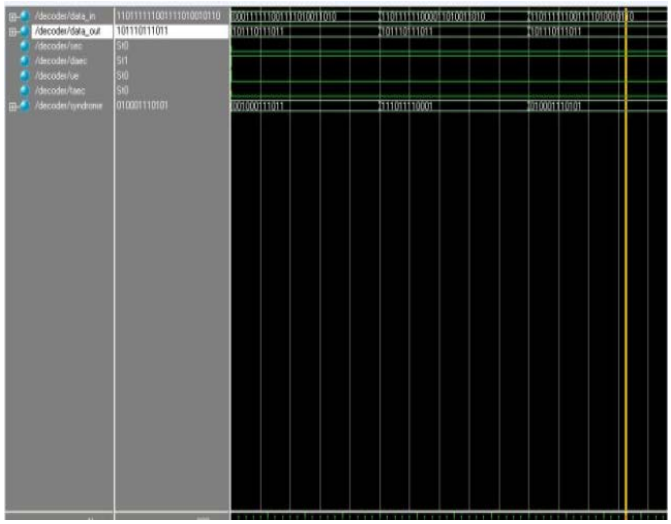


Proposed –Tripple Adjacent Error Correction

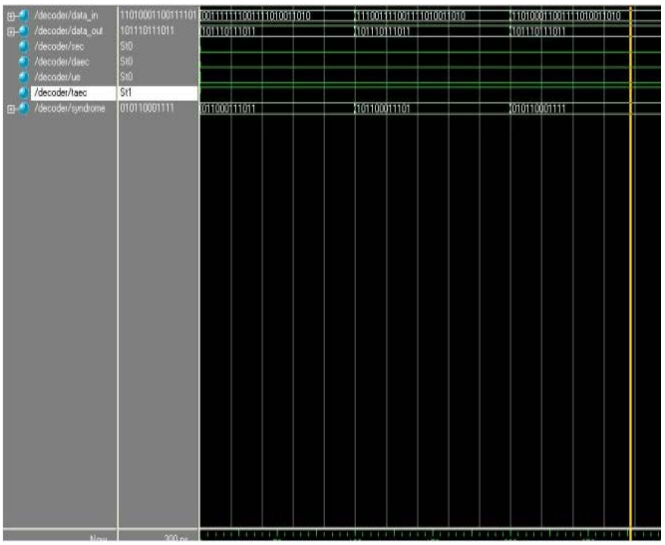
A. Encoder



D. Double Adjacent Error Correcting Decoder



E. Decoder-Tripple Adjacent Error Correcting Decoder



VII. CONCLUSION

The Golay Code and operation for various encoder and decoder is discussed. This encoding and decoding algorithm have been successfully applied to short block codes such as Golay Code in modified proposed methodology. Decoding algorithm consists of syndrome measurement unit, weight measurement unit and weight constraint. the proposed decoder is not only much simpler than a traditional SEC-DAEC decoder, but also simpler than a standard SEC decoder for the Golay code. To evaluate the benefits of the new decoder, it has been implemented in HDL and mapped to a65-nm library. The results confirm that significant reductions in area, delay, and power consumption can be obtained compared with the traditional SEC-DAEC decoder.

REFERENCES

[1] R.D. Schrimpf and D.M. Fleetwood, "Radiation Effects and Soft Errors in Integrated Circuits and Electronic Devices", Singapore: World Scientific, 2004.

[2] R.C. Baumann, "Soft errors in advanced computer systems", IEEE Des.Test. Comput., Vol. 22, No. 3, Pp. 258–266, 2005.

[3] 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, 1984.

[4] 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, 1970.

[5] E. Ibe, H. Taniguchi, Y. Yahagi, K. Shimbo and T. Toba, "Impact ofscaling 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, 2010.

[6] 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, 2010.

[7] S. Satoh, Y. Tosaka and S.A. Wender, "Geometric effect of multiple-bits of errors induced by cosmic ray neutrons on DRAM's", IEEE Electron Device Lett., Vol. 21, No. 6, Pp. 310–312, 2000.

[8] 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, 2013.

[9] M.A. Bajuraet, Y. Boulghassoul, R. Naseer, S. Das Gupta, A.F. Witulski, J. Sondeen and J.N. Damoulakis, "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, 2007.

[10] A. Dutta and N.A. Touba, "Multiple bit upset tolerant memory using a selective cycle avoidance based SEC-DED-DAEC code", Proc. IEEE VLSI Test Symp., Pp. 349–354, 2007.

[11] Z. Ming, X.L. Yi and L.H. Wei, "New SEC-DED-DAEC codes for multiple bit upsets mitigation in memory", Proc. IEEE/IFIP 20th Int.Conf. VLSI Syst.-Chip, Pp. 254–259, 2011.

[12] L.J. Saiz-Adalid, P. Reviriego, P. Gil, S. Pontarelli, and J.A. Maestro, "MCU tolerance in SRAMs through low-redundancy triple adjacent error correction", IEEE Trans. Very Large Scale Integr. (VLSI) Syst., Vol. 23, No. 10, Pp. 2332-2336, 2015.

[13] R. Naseer and J. Draper, "DEC ECC design to improve memory reliability in sub-100 nm technologies", Proc. IEEE ICECS, Pp. 586–589, 2008.

[14] M.Y. Hsiao, D.C. Bossen and R.T. Chien, "Orthogonal Latin squarecodes", IBM J. Res. Develop., Vol. 14, No. 4, Pp. 390–394, Jul. 1970.

[15] S.F. Liu, P. Reviriego, and J.A. Maestro, "Efficient majority logicfault detection with difference-set codes for memory applications", IEEE Trans. Very Large Scale Integr. (VLSI) Syst., Vol. 20, No. 1, Pp. 148–156, 2012.

[16] M.J.E. Golay, "Notes on digital coding", Proc. IEEE, Vol. 37, No. 6, Pp. 657, 1949.

[17] S. Sarangi and S. Banerjee, "Efficient hardware implementation of encoder and decoder for Golay

code", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 23, No. 9, Pp.1965-1968, 2015.

[18] K. Namba, S. Pontarelli, M. Ottavi and F. Lombardi, "A single-bit anddouble-adjacent error correcting parallel decoder for multiple-bit error correcting BCH codes", IEEE Trans. Device Mater. Rel., Vol. 14, No. 2, Pp. 664–671, 2014.

[19] S. Lin and D.J. Costello, Error Control Coding, 2nd ed.Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.

[20] E. R. Berlekamp, "Decoding the Golay code", Jet Propulsion Lab.,Pasadena, CA, USA, Tech. Rep. Pp. 81–84, 1971.