ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 3, Issue 1, January 2018

A Novel Multibit Errors Prediction And 32-Bit Correction In Memories Using DMC

[1] B.Venukumari, [2]B.Srinivas
[1] M.Tech Student, Department of ECE, Aditya engineering college,surampalem, India,
[2] Associate Professor, Department of ECE, Aditya engineering college, surampalem, India.

Abstract: ***This paper presents a cost efficient technique to correct Multiple Bit Upsets (MBUs) to protect memories against radiation. To protect memories from MBUs, many complex Error Correction Codes (ECCs) were used previously, but the major issue is higher redundant memory overhead. In this project 64-bit Decimal Matrix Code was implemented to assure the reliability of memory. The modified protection code utilized procedure to detect errors, so that more errors were detected and corrected. The results showed that the modified scheme has a protection level against large MBUs in memory. Transient MBUs are suitable major problems in the reliability of memories exposed to radiation environment. In this modification method 64-bit matrix code implemented for error correction in memories. To prevent MBUs from causing data corruption, more complex Error Correction Codes (ECCs) are widely used to protect memory, but the main problem is that they would require higher delay overhead. Decimal Matrix Codes (DMCs) based on Hamming codes are being proposed for memory protection. The main issue is that they are double error correction codes and the error correction capabilities are not enhanced in all cases. Moreover, the erasure codes is introduced to reduce the area overhead of extra circuits exclusive of disturbing the total encoding and decoding processes. Now a days to maintain good level of reliability, it is necessary to protect memory bits using protection codes, for this purpose, various error detection and correction methods are being used. The only drawback of the existing DMC is that more redundant bits are required to maintain higher reliability of memory. The modified technique used DMC to assure reliability in presence of multiple bit upset and reduce more redundant bit and its correct more error compare to existing system.***

Keywords: FPGA, Multiple Bit Upsets, Matrix Codes, Soft Error.

I. INTRODUCTION

AS CMOS technology scales down to nanoscale and memories are combined with an increasing number of electronic systems, the soft error rate in memory cells is rapidly increasing, especially when memories operate in space environments due to ionizing effects of atmospheric neutron, alpha-particle, and cosmic rays. Although single bit upset is a major concern about memory reliability, Multiple Bell Upsets (MBUs) have become a serious reliability concern in some memory applications. In order to make memory cells as fault-tolerant as possible, some Error Correction Codes (ECCs) have been widely used to protect memories against soft errors for years. For example, the BCH codes,

Reed–Solomon codes, and Punctured Difference Set (PDS) codes have been used to deal with MBUs in memories. But these codes require more area, power, and delay overheads since the encoding and decoding circuits are more complex in these complicated codes. The general idea for achieving error detection and correction is to add some redundancy (i.e., some extra data) to a message, which receiver can use to check consistency of the delivered message, and to pick up data determined to be corrupt. Error-detection and correction scheme can be either systematic or non-systematic: In a systematic scheme, the transmitter sends the unique data, and attaches a fixed number of check bits (or parity data), which are derived from the data bits by some

deterministic algorithm. If only the error detection is required, a receiver can simple apply the same algorithm to the received data bits and compare its output with the receive check bits; if the values do not match, an error has occurred at some point throughout the transmission. ECC codes are regularly used in lower-layer communication, As well as for reliable storage in media such as CDs, DVDs, hard disks and RAM.

Static RAM based Field-Programmable Gate Arrays (FPGAs) are most widely used in variety of applications mainly due to short time-to-market time, flexibility, high density, and cost-efficiency. SRAM-based FPGA stores logic cells configuration data in the static memory organized as an array of latches. FPGA is used for designing complex digital circuits. Power consumption is also reduced by using SRAM. The power consumption of SRAM varies widely depending on how frequently it is accessed, it can be as power-hungry as dynamic RAM, when used at high frequencies, and some ICs can consume many watts at full bandwidth. On the other hand, static RAM used at a somewhat slower pace, such as in

applications with Moderately clocked microprocessors, draws very little power and can have a nearly negligible power consumption when sitting idle in the region of a few micro watts. Several techniques have been proposed to manage power consumption of SRAM-based memory structures. FPGA device customizable by SRAM consists of an array of programmable logic blocks interconnected by a programmable routing network and I/O blocks. SRAM-based FPGA devices are becoming popular because of their high performance, reduced development cost and re-programmability. FPGAs based on a nanometer technology with denser integration schemes. Memories are one of the most widely used elements in electronic systems. Radiation in the environment seriously affect the functionality of a circuit. A Single-Event Upset (SEU) occurs when a charged particle, present in the environment, hits the silicon of a circuit introducing an error in the system. Such errors in FPGA device affects the functionality of the mapped design also called as Soft errors. A soft error will not damage a system's hardware, the only damage is to the data that is being processed in the memory. To address this issue, Built-in Current Sensors (BICS) have recently been applied in conjunction with Single Error Correction/Double Error Detection (SEC-DED) codes to protect memories from MBUs. But by using those methods only SEU could be corrected. For both the detection and correction of errors, a generic scrubbing scheme to reconstruct the erroneous configuration frame based on the concept of Erasure coding algorithm is introduced in this project. In this type of Erasure coding algorithm, MBUs are detected by using the interleaving distance which is further classified into horizontal and vertical parity.

II. MBU PATTERNS

Interleaving technique has been used to restrain MBUs, which rearrange bits in the physical arrangement to separate the bits in the same logical word into different physical words. However, interleaving technique may not be practically used in Content-Addressable Memory (CAM), because of the tight coupling of hardware structures from both bits and comparison circuit structures. More recently, in 2-D Matrix Codes (MCs) are proposed to efficiently correct MBUs per word with a low decoding delay, in which one word is divided into multiple rows and multiple columns in logical. The bits per row are protected by Hamming code, while parity code is added in each column. For the MC based on Hamming, when two errors are detected by Hamming, the vertical syndrome bits are activated so that these two errors can be corrected. As a result, MC is capable of correcting only two errors in all cases. In an approach that combines algorithm with Hamming code has been conceived to be applied at software level. It uses addition of integer values to detect and correct soft errors. The results obtained have shown that this approach have a lower delay overhead over other codes. Built-In Current Sensors (BICS) are proposed to assist with single-error correction and double-error detection codes to provide protection against MBUs. However, this technique can only correct two errors in a word.In this paper, novel matrix code based on divide-symbol is proposed to provide enhanced memory reliability. The proposed matrix code utilize algorithm (integer addition and integer subtraction) to identify errors. The advantage of using algorithm is that the error detection capability is maximize so that the reliability of memory is enhanced. Besides, the erasure codes is proposed to minimize the area overhead of extra circuits (encoder and decoder) without disturbing the whole encoding and decodingprocess.

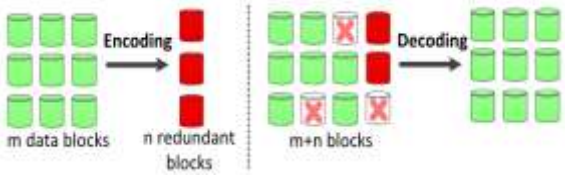

Fig. 1 Encoding and decoding of erasure codes.

III. MODIFIED DMC

In this section, DMC is proposed to assure reliability in the presence of MCUs with reduced performance overheads, and a 64-bit word is encoded and decoded as an example based on the proposed techniques.

A. Modified Schematic of Fault-Tolerant Memory:

The schematic of fault-tolerant memory is as shown in Fig. 2. The DMC encoder is fed with information bits D, during the encoding (write) process, and then the DMC encoder produces the vertical redundant bits V and horizontal redundant bits H. The obtained DMC code word is stored in the memory, once encoding process is completed. If the memory is affected by MBUs, in the decoding (read) process these errors can be corrected. The modified DMC has higher fault-tolerant capability with higher performance because of decimal algorithm. The fault-tolerant memory uses ERT technique, to reduce extra circuit's area overhead and will be introduced in the following sections.

ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 3, Issue 1, January 2018

Fig. 2 Proposed schematic of fault-tolerant memory protected with DMC

B. Implementation DMC Encoder

The DMC implement, first it performs the ideas of divide-symbol and arrange-matrix, i.e. symbols of m bits (N = k × m) is obtained by dividing the N-bit word, and these symbols are arranged in a k1 × k2 2-D matrix (k = k1 × k2, where k1 and k2 values represent the numbers of rows and columns in the logical matrix respectively). Second by decimal integer addition of selected symbols per row the horizontal redundant bits H are obtained. Here, each symbol is regarded as a decimal integer. Third, by binary operation among the bits per column the vertical redundant bits V are obtained. It should be noted that instead of in physical both divide-symbol and arrange-matrix are implemented in logical. Therefore, the physical structure of the memory is not required to be changed as according to the modified DMC. We considered a 64-bit word as an example, to explain the proposed DMC scheme, as shown in Fig. 2. From D0 to D63 cells are information bits. Eight symbols of 4-bit are obtained by dividing 64-bit word. By choosing k1 = 2 and k2 = 4 simultaneously. Horizontal check bits are H0–H39; vertical check bits are V0 through V31 are. However, it should be mentioned that the number of redundant bits and the maximum correction capability (i.e., the maximum size of MBUs can be corrected) are different when values for k and m are chosen different. Therefore, to maximize the correction capability and minimize the number of redundant bits k and m should be carefully adjusted to maximize the correction capability and minimize the number of redundant bits. For example, in this case, when k = 2×2 and m = 8, only 1-bit error can be corrected and the number of redundant bits is 80. When k = 4 × 4 and m = 2, 3-bit errors can be corrected and the number of redundant bits is reduced to 32. However, when k = 2 × 4 and m = 4, the maximum correction capability is up to 5 bits and the number of redundant bits is 72. In this project, in order to enhance the reliability of memory, the error correction capability is first considered, so k = 2 × 8 and m=4 are utilized to construct DMC.

The horizontal redundant bits H can be obtained by decimal integer addition as follows

H4H3H2H1H0=D3D2D1D0+ D19D18D17D16(1)

H9H8H7H6H5 = D7D6D5D4+D23D22D21D20(2)

and similarly for the horizontal redundant bits H14H13H12H11H10,H19H18H17H16H15H16, H24H23H22H21H20,H29H28H27H26H25, H34H33H32H31H30andH39H38H37H36H35 where "+" represents decimal integer addition.

For the vertical redundant bits V, we have

$V_0=D_0 \oplus D_{31}$ (3)

$V_1=D_1 \oplus D_{32}$ (4)

and similarly for the rest vertical redundant bits. The encoding can be performed by decimal and binary addition operations from (1) to (4). The encoder that computes the redundant bits using multi bit adders and XOR gates is shown in Fig. In this figure, H39 − H0 are horizontal redundant bits, $V_{31} - V_0$ are vertical redundant bits, and the remaining bits U63 − U0 are the information bits which are directly copied from D_{31} to D_0.

C. Implementation of DMC Decoder

To obtain a word being corrected, the decoding process is required. For example, first, the received redundant bits H4H3H2H1H0' and V0'-V3' are generated by the received information bits D'. Second,the horizontal syndrome bits ΔH4H3H2H1H0 and the vertical syndrome bits S3 − S0 can be calculated as follows:

ΔH4H3H2H1H0 = H4H3H2H1H0' − H4H3H2H1H0 (5)

S0=V0⊕V0 (6)

and similarly for the rest vertical syndrome bits, where "–" represents decimal integer subtraction. When ΔH4H3H2H1H0 and S3 − S0 are equal to zero, the stored code word has original information bits in symbol 0 where no errors occur. When ΔH4H3H2H1H0 and S3 − S0 are nonzero, the induced errors (the number of errors is 4 in this case) are detected and located in symbol 0, and then these errors can be corrected by

D0correct=D0⊕S0 (7)

ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 3, Issue 1, January 2018

The modified DMC decoder is depicted in Fig, which is made up of the following sub modules, and each executes a specific task in the decoding process: syndrome calculator, error locator, and error corrector. It can be observed from this figure that the redundant bits must be recomputed from the received information bits D' and compared to the original set of redundant bits in order to obtain the syndrome bits ΔH and S. Then error locator uses ΔH and S to detect and locate which bits some errors occur in. Finally, in the error corrector, these errors can be corrected by inverting the values of error bits. Recovery based on erasure codes uses for error detection, to function in drive a scrubber unit that periodically assessments the parity bits of the configuration frames for practicable error. Upon a detection of error, with the aid of assuming that the inaccurate physique is erased, its contents are recovered utilizing an erasure code.

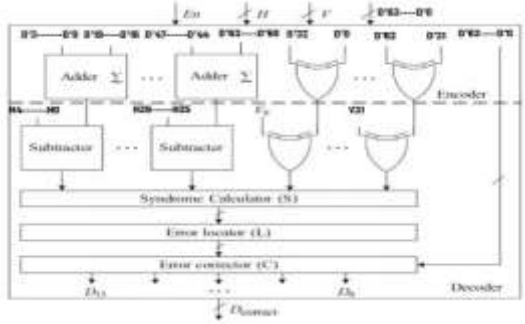

Fig 3: 64-bit DMC decoder structure using ERT

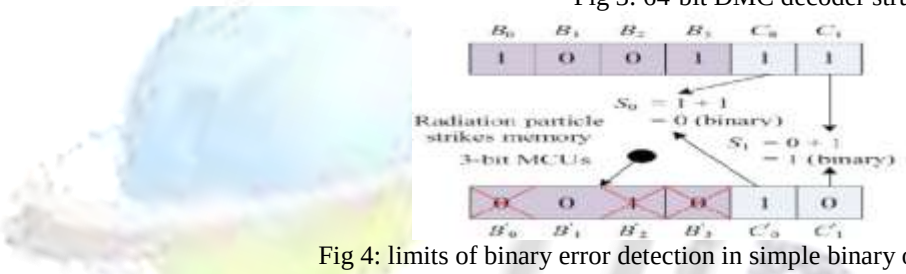

Fig 4: limits of binary error detection in simple binary operations

In the modified scheme, the circuit area of DMC is minimized by reusing its encoder. This is called the ERT. The ERT can reduce the area overhead of DMC without disturbing the whole encoding and decoding processes. From Fig, it can be observed that the DMC encoder is also reused for obtaining the syndrome bits in DMC decoder. Therefore, the whole circuit area of DMC can be minimized as a result of using the existent circuits of encoder. Besides, this figure also shows the proposed decoder with an enable signal En for deciding whether the encoder needs to be a part of the decoder. In other words, the En signal is used for distinguishing the encoder from the decoder, and it is under the control of the write and read signals in memory. Therefore, in the encoding (write) process, the DMC encoder is only an encoder to execute the encoding operations. However, in the decoding (read) process, this encoder is employed for computing the syndrome bits in the decoder. These clearly show how the area overhead of extra circuits can be substantially reduced.

In the semiconductor and electronic outline industry, Verilog is an predefined language used to show electronic framework. Verilog HDL, not to be mistaken for VHDL is most generally utilized as a part of the outline, confirmation, and usage of digital system.

Table: Read/Write Encode signal

Extra Circuit	En Signal		Function
	Read signal	Write signal	Encoding
Encoder	0	1	Computer Syndrome bits
	1	0	

ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 3, Issue 1, January 2018

IV. SIMULATION RESULTS

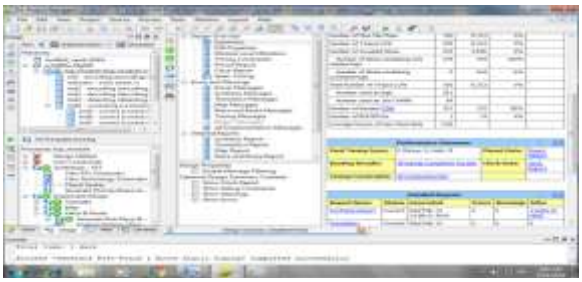

Fig5: Synthesis report

Block diagram

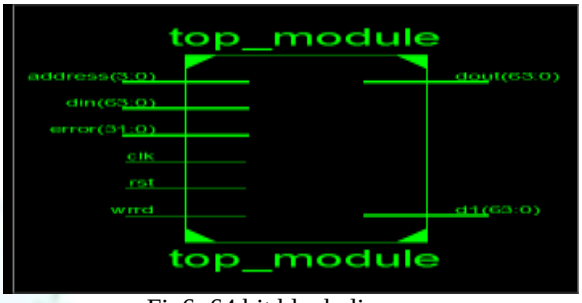

Fig6: 64 bit block diagram

RTL Schematic

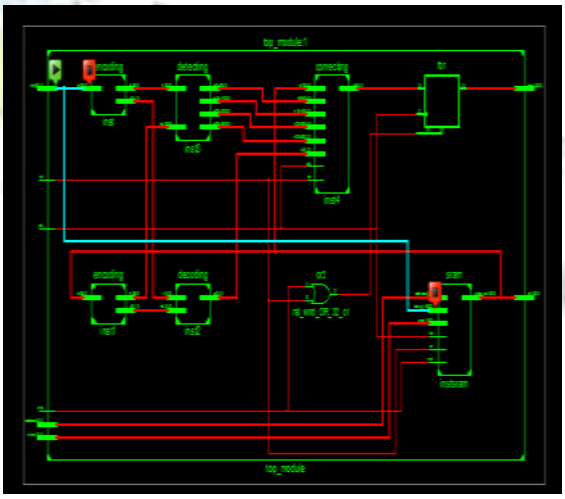

Fig7: RTL schematic

Technology Schematic

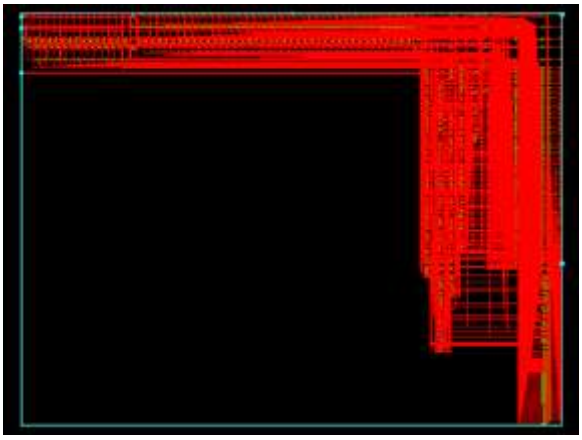

Fig8: Technology schematic

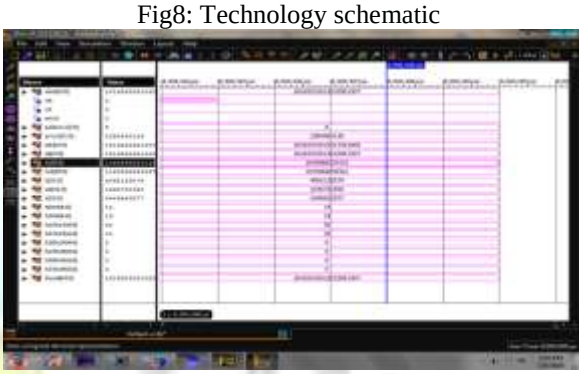

Fig9: Modified DMC Results

V. CONCLUSION AND FUTURE SCOPE

To prevent MBUs from causing data corruption, more complex error correction codes (ECCs) are widely used to protect memory, but the main problem is that they would require higher delay overhead. Recently, matrix codes (MCs) based on hamming codes have been proposed for memory protection. In this paper, novel per-word was proposed to assure the reliability of memory. The proposed protection code utilized algorithm to detect errors, so that more errors were detected and corrected. The obtained results showed that the proposed scheme has a superior protection level against large MBUs in memory. Besides, the proposed error detection technique is an attractive opinion to detect MBUs in CAM because it can be combined with BICS to provide an adequate level of immunity.

The study carried out in this project can be extended to many other potential fields. Major possibility is to develop an error correcting system which can provide better performance with less delay overhead, lower power requirements and less are consumption. The study can be carried out the carried out the pipelining the existing codes into an efficient form. So that the delay is reduced. Also by changing the address and other elements used in realization, the delay can be considerably reduced.

REFERENCES

[1] Jing Guo, Liyi Xiao, Member, IEEE, Zhigang Mao, Member, IEEE, and QiangZhao,"Enhanced memory reliability against multiple cell upsets using Decimal Matrix Code" IEEE Trans. Very Large ScaleIntegr. (VLSI) Syst., vol. 22, no. 1, pp.127-135, Mar 2013.

[2] D. Radaelli, H. Puchner, S. Wong, and S. Daniel, "Investigation of multi-bit upsets in a 150 nm technology SRAM device," IEEE Trans.Nucl. Sci., vol. 52, no. 6, pp. 2433–2437, Dec. 2005.

[3] E. Ibe, H. Taniguchi, Y. Yahagi, K. Shimbo, and T. Toba, "Impact of scaling on neutron induced soft error in SRAMs from an 250 nm to a 22 nm design rule," IEEE Trans. Electron Devices, vol. 57, no. 7, pp. 1527–1538, Jul. 2010.

[4] C. Argyrides and D. K. Pradhan, "Improved decoding algorithm for high reliable reed muller coding," in Proc. IEEE Int. Syst. On Chip Conf., Sep. 2007, pp. 95–98.

[5] A. Sanchez-Macian, P. Reviriego, and J. A. Maestro, "Hamming SEC-DAED and extended hamming SEC-DED-TAED codes through selective shortening and bit placement," IEEE Trans. Device Mater. Rel., to be published.

[6] S. Liu, P. Reviriego, and J. A. Maestro, "Efficient majority logic fault detection with difference-set codes for memory applications," IEEETrans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 1, pp. 148–156, Jan. 2012.

[7] M. Zhu, L. Y. Xiao, L. L. Song, Y. J. Zhang, and H. W. Luo, "New mix codes for multiple bit upsets mitigation in fault-secure memories," Microelectron. J., vol. 42, no. 3, pp. 553–561, Mar. 2011.
[7] R. Naseer and J. Draper, "Parallel double error correcting code design to mitigate multi-bit upsets in SRAMs," in Proc. 34th Eur. Solid-State Circuits, Sep. 2008, pp. 222–225.
[8] G. Neuberger, D. L. Kastensmidt, and R. Reis, "An automatic technique for optimizing Reed-Solomon codes to improve fault tolerance in memories," IEEE Design Test Comput., vol. 22, no. 1, pp. 50–58, Jan.–Feb. 2005.
[9] P. Reviriego, M. Flanagan, and J. A. Maestro, "A (64,45) triple error correction code for memory applications," IEEE Trans. Device Mater. Rel., vol. 12, no. 1, pp. 101–106, Mar. 2012.
[10] S. Baeg, S. Wen, and R. Wong, "Interleaving distance selection with a soft error failure model," IEEE Trans. Nucl. Sci., vol. 56, no. 4, pp. 2111–2118, Aug. 2009.
[11] K. Pagiamtzis and A. Sheikholeslami, "Content addressable memory (CAM) circuits and architectures: A tutorial and survey," IEEE J. Solid-State Circuits, vol. 41, no. 3, pp. 712–727, Mar. 2003.
[12] S. Baeg, S. Wen, and R. Wong, "Minimizing soft errors in TCAM devices: A probabilistic approach to determining scrubbing intervals," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 4, pp. 814–822, Apr. 2010.
[13] P. Reviriego and J. A. Maestro, "Efficient error detection codes for multiple-bit upset correction in SRAMs with BICS," ACM Trans. Design Autom. Electron. Syst., vol. 14, no. 1, pp. 18:1–18:10, Jan. 2009.
[14] C. Argyrides, R. Chipana, F. Vargas, and D. K. Pradhan, "Reliability analysis of H-tree random access memories implemented with built in current sensors and parity codes for multiple bit upset correction," IEEE Trans. Rel., vol. 60, no. 3, pp. 528–537, Sep. 2011.
[15] C. Argyrides, D. K. Pradhan, and T. Kocak, "Matrix codes for reliable and cost efficient memory chips," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 3, pp. 420–428, Mar. 2011.