



Decimal Matrix Code for Enhanced Memory Reliability against Multiple Cell Upsets

T.Dhivya¹, M.Mohankumar², Swaminathan Veerapandian³

PG Student, ECE Dept, Sri Eshwar college of Engineering, Kinathukadavu, Coimbatore¹
Assistant professor, ECE Dept, Sri Eshwar college of Engineering, Kinathukadavu, Coimbatore²
Employee, Network executive in Tata communications Ltd., Coimbatore³

Abstract: Transient multiple cell upsets MCUs are becoming major issues in their liability of memories exposed to radiation environment. To prevent MCUs 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 over head. Moreover, the encoder-reuse technique (ERT) is proposed to minimize the area over head of extra circuits without disturbing the whole encoding and decoding processes. The main issue is that they are double error correction codes and the error correction capabilities are not improved in all cases. In this paper, novel decimal matrix code (DMC) based on divide-symbol is proposed to enhance memory reliability with lower delay over head. The proposed DMC is compared to well known codes such as existing Hamming codes, Punctured different set PDS codes. The only drawback is that, it requires more redundant bits for memory protection. The obtained results show that the mean time to failure (MTTF) of the proposed scheme is 452.9%, 154.6% and 122.6% of Hamming, MC, and PDS, respectively. Recently, matrix codes (MCs) based on Hamming codes have been proposed for memory protection. The proposed DMC utilizes decimal algorithm to obtain the maximum error detection capability. ERT uses DMC encoder itself to be part of the decoder. At the same time, the delay overhead of the proposed scheme is 73.1%, 69.0% and 26.2% of Hamming, MC and PDS respectively.

Keywords: Decimal algorithm, error correction codes (ECCs), mean time to failure (MTTF), memory, multiple cells upsets(MCUs).

I. INTRODUCTION

As CMOS technology scales down to nano scale 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 cell upsets (MCUs) 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 of terrors for years. For example, the Bose–Chaudhuri–Hocquenghem codes, Reed–Solomon codes and punctured difference set (PDS) codes have been used to deal with MCUs 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. Interleaving technique has been used to restrain MCUs, Which rearrange cells 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 cells and comparison circuit structures.

Built-in current sensors (BICS) are proposed to assist with single-error correction and double-error detection codes to provide protection against MCUs. However, this technique can only correct two errors in a word. More recently in 2-D matrix codes (MCs) are proposed to efficiently correct MCUs 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 decimal 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. In this paper, novel decimal matrix code (DMC) based on divide-symbol is proposed to provide enhanced memory reliability. The proposed DMC utilizes decimal algorithm (decimal integer addition and decimal integer subtraction) to detect errors. The advantage of using decimal algorithm is

that the error detection capability is maximized so that the reliability of memory is enhanced. Besides the encoder-reuse technique (ERT) is proposed to minimize the area overhead of extra circuits (encoder and decoder) without disturbing the whole encoding and decoding processes, because ERT uses DMC encoder itself to be part of the decoder.

This paper is divided into the following sections. The proposed DMC is introduced and its encoder and decoder circuits are present in Section II. This sectional so illustrates the limits of simple binary error detection and the advantage of decimal error detection with some examples. The reliability and over heads analysis of the proposed code are analyzed in Section III. In Section IV the implementation of decimal error detection together with BICS for error correction in CAM is provided. Finally some conclusions of this paper are discussed and shared in Section V.

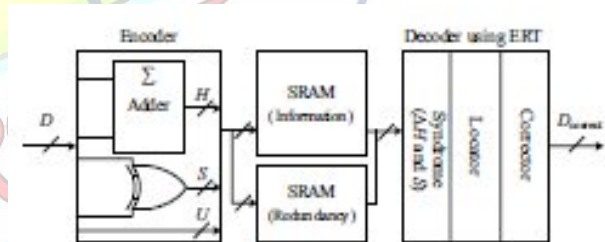


Fig. 1. Proposed schematic of fault-tolerant memory protected with DMC.

II. PROPOSED DMC

In this section, DMC is proposed to assure reliability in the presence of MCUs with reduced performance over heads, and a 32-bit word is encoded and decoded as an example based on the proposed techniques.

A. Proposed Schematic of Fault-Tolerant Memory

The proposed schematic of fault-tolerant memory is depicted in Fig.1. First during the encoding (write) process, information bits are fed to the DMC encoder and then the horizontal redundant



bits H and vertical redundant bits V are obtained from the DMC encoder. When the encoding process is completed the obtained DMC code word is stored in the memory. If MCUs occur in the memory. These errors can be corrected in the decoding (read) process. Due to the advantage of decimal algorithm, the proposed DMC has higher fault-tolerant capability with lower performance overheads. In the fault-tolerant memory the ERT technique is proposed to reduce the area overhead of extra circuits and will be introduced in the following sections.

B. Proposed DMC Encoder

In the proposed DMC, first, the divide-symbol and arrange-matrix ideas are performed. i.e., the N -bit word is divided into k symbol so fm bits ($N = k \times m$), and these symbols are arranged in a $k_1 \times k_2$ 2-D matrix ($k = k_1 \times k_2$), where the values of k_1 and k_2 represent the numbers of the horizontal redundant bits H are produced by performing decimal integer addition of selected symbols per row. Here, each symbol is regarded as a decimal integer. Third, the vertical redundant bits V are obtained by binary operation among the bits per column. It should be noted that both divide-symbol and arrange-matrix are implemented in logical instead of in physical. Therefore, the proposed DMC does not require changing the physical structure of the memory. To explain the proposed DMC scheme, we take a 32-bit word as an example as shown in Fig. 2. The cells from D_0 to D_{31} are information bits. This 32-bit word has been divided into eight symbols of 4-bit. $k_1=2$ and $k_2=4$ have been chosen simultaneously. H_0-H_{19} are horizontal check bits. V_0 through V_{15} are vertical check bits. However, it should be mentioned that the maximum correction capability (i.e., the maximum size of MCUs can be corrected) and the number of redundant bits are different when the different values for k and m are chosen. Therefore, 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 $k=2 \times 2$ and $m=8$, only one bit error can be corrected and the number of redundant bits is 40. When $k=4 \times 4$ and $m=2$, 3-bit errors can be corrected and the number of redundant bits is reduced to 32. However, when $k=2 \times 4$ and $m=4$, the maximum correction capability is upto 5 bits and the number of redundant bits is 36. In this paper, in order to enhance the reliability of memory, the error correction capability is first considered, so $k = 2 \times 4$ and $m = 4$ are utilized to construct DMC.

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

$$H_4H_3H_2H_1H_0 = D_3D_2D_1D_0 + D_{11}D_{10}D_9D_8 \quad (1)$$

$$H_9H_8H_7H_6H_5 = D_7D_6D_5D_4 + D_{15}D_{14}D_{13}D_{12} \quad (2)$$

And similarly for the horizontal redundant bits $H_{14}H_{13}H_{12}H_{11}H_{10}$ and $H_{19}H_{18}H_{17}H_{16}H_{15}$ where “+” represents decimal integer addition.

For the vertical redundant bits V , we have

$$V_0 = D_0 \text{ XOR } D_{16} \quad (3)$$

$$V_1 = D_1 \text{ XOR } D_{17} \quad (4)$$

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 multibit adders and XOR gates is shown in Fig. 3. In this figure, $H_{19}-H_0$ are horizontal redundant bits, $V_{15}-V_0$ are vertical redundant bits, and the remaining bits $U_{31}-U_0$ are the information bits which are directly copied from D_{31} to D_0 . The



enable signal En will be explained in the next section.

II. Proposed DMC Decoder

To obtain a word being corrected, the decoding process is required. For example, first, the received redundant information bits D^* . Second, the horizontal syndrome bits $H_4H_3H_2H_1H_0$ and the vertical syndrome bits S_3-S_0 can be calculated as follows:

$$H_4H_3H_2H_1H_0 = H_4H_3H_2H_1H^* - H_4H_3H_2H_1H_0 \quad (5)$$

$$S_0 = V_0 \oplus V_0^{(6)}$$

And similarly for the rest vertical syndrome bits, where “-” represents decimal integer subtraction.

When $H_4H_3H_2H_1H_0$ and S_3-S_0 are equal to zero,

the stored codeword has original information bits in symbol 0 where no errors occur. When $H_4H_3H_2H_1H_0$ and S_3-S_0 are non zero, 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 $D_{0\text{correct}} = D_0 \oplus S_0$ (7) The proposed DMC decoder is depicted in Fig.4, which is made up of the following sub-modules, and each executes a specific task in the decoding process: Syndrome calculator, 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 H and S. The error locator uses H and S to detect and locate which bits some error occur in. Finally the error can be corrected by inverting values of error bits.

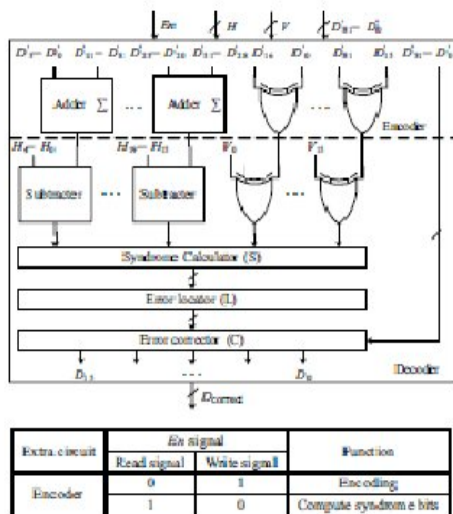


Fig. 4. 32-bit DMC decoder structure using ERT.

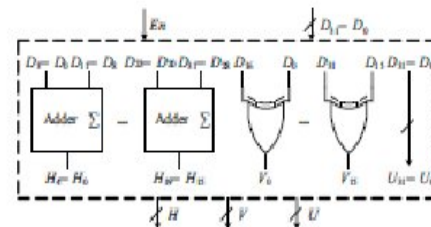


Fig. 3. 32-bit DMC encoder structure using multibit adders and xor gates.



32-bits DMC logical organization ($k=2 \times 4$ and $m=4$). Here, each symbol is regarded as a decimal integer.

D. Limits of simple binary detection:

For the proposed binary error detection technique in [13], although it requires low redundant bits, its error detection capability is



limited. The main reason for this is that its error detection mechanism is based on binary.

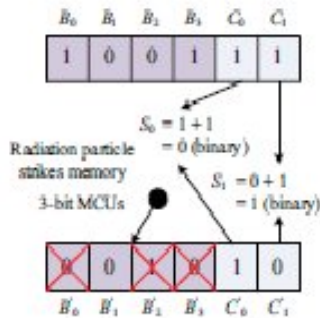


Fig. 5. Limits of binary error detection in simple binary operations.

We illustrate the limits of this simple binary error detection [13] using a simple example. Let us suppose that the bits B_3 , B_2 , B_1 , and B_0 are original information bits and the bits C_0 and C_1 are redundant bits shown in Fig.5. The bits C_0 and C_1 are obtained using the binary algorithm.

E. Advantage of Decimal Error Detection

In the previous discussion, it has been shown that error detection [13] based on binary algorithm can only detect a finite number of errors. However, when the decimal algorithm is used to detect errors, these errors can be detected so that the decoding error can be avoided. The reason is that the operation mechanism of decimal algorithm is different from that of binary. The detection procedure of decimal error detection using the proposed structure shown in Fig.2 is fully described in Fig.6. First of all, the horizontal redundant bits $H_4H_3H_2H_1H_0$ are obtained from the original information bits in symbols 0 and 2 according to (1)

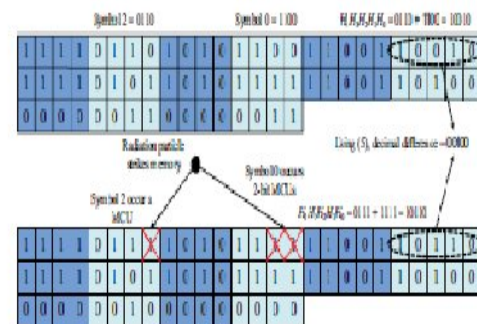
$$H_4 H_3 H_2 H_1 H_0 = D_3 D_2 D_1 D_0 + D_{11} D_{10} D_9 D_8 = 1100 + 0110 = 10010$$

$$H_4 H_3 H_2 H_1 H_0' = D_3 D_2 D_1 D_0' + D_{11} D_{10} D_9 D_8' = 0111 + 1111 = 10110$$

Then horizontal syndrome bits are

$$H_4 H_3 H_2 H_1 H_0 = H_4 H_3 H_2 H_1 H_0' - H_4 H_3 H_2 H_1 H_0 = 10110 - 10010 = 00100.$$

The decimal value of $H_4H_3H_2H_1H_0$ is not “0,” which represents that errors are detected and located in symbol 0 or symbol 2. Subsequently, the precise location of the bits that were flipped can be located by using the vertical syndrome bits S_3-S_0 and $S_{11}-S_8$. Finally, all these errors can be corrected by (7). Therefore, based on decimal algorithm, the proposed technique has higher tolerance capability for protecting memory against MCUs.



The proposed DMC can easily correct upset softtype1, 2, and 3, because these are the essential property of DMC: any types of single-error and multiple-error corrections in two consecutive symbols. Upsets of types 4 and 5 introduced in Fig.7 are also corrected because the multiple errors per row can be detected by the horizontal syndrome bits (see Fig. 6). These show that the proposed technique is an attractive option to protect memories from large MCUs. However, for the upsets of type4 and 5, It is important to recognize that it can result in decoding error when the following pre requisite factors are achieved simultaneously (this error is typical of its kind).

- 1) The decimal integer sum of information bits in symbols 0 and 2 is equal to 2^m-1 .



2) All the bits in symbols 0 and 2 are upset.

The more detailed explanation is shown in Fig.8. Assuming that these two factors have been achieved, according to the encoding and decoding processes of DMC, $H_4H_3H_2H_1H_0$, and

$H_4H_3H_2H_1H^*$ are computed, as follows:

$$\begin{aligned} H_4H_3H_2H_1H_0 &= D_3D_2D_1D_0 + D_{11}D_{10}D_9D_8 \\ &= 0110 + 1001 \\ &= 01111 \quad (17) \end{aligned}$$

$$\begin{aligned} H_4H_3H_2H_1H^* &= D_3D_2D_1D_0^* + D_{11}D_{10}D_9D_8 \\ g^* &= 1001 + 0110 \\ &= 01111. \quad (18) \end{aligned}$$

Then the horizontal syndrome bits $H_4H_3H_2H_1H_0$ can be obtained

$$\begin{aligned} H_4H_3H_2H_1H_0 &= H_4H_3H_2H_1H^* - H_4H_3H_2H_1H_0 \\ &= 01111 - 01111 \\ &= 00000 \end{aligned}$$

This result means that no errors occur in symbols 0 and 2 and memory will suffer a failure. However, this case is rare.

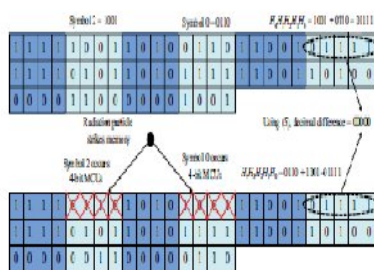


Fig. 8. Error type cannot be corrected by our proposed DMC. The main reason is that $H_4H_3H_2H_1H_0$ will be "00000". Note that even though 7-bit errors occur in symbols 0 and 2 simultaneously, the decoding error can be reduced.

III. RELIABILITY AND OVERHEADS ANALYSIS

In this section, the proposed DMC has been implemented in HDL, simulated with Model Sim and tested for functionality by given various inputs.

The area, power, and critical path delay of extra circuits have been obtained. For fair comparisons, Hamming, PDS [9], and MC [15] are used for references. Here, the usage of (64, 45) PDS is a triple-error correction code [9] and its information bits is shorted to 32 from 45 bits.

Fault Injection:

The correction coverage of PDS [9], MC [15], Hamming, and the proposed DMC codes are obtained from one million experiments.

TABLE I
CORRECTION FOR COVERAGE (32-bit)

ECC Codes	The Number of Errors in a Word															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DMC (9)	100	100	100	100	100	92.5	84.7	76.0	66.7	60.9	54.5	47.7	40.0	31.6	22.2	11.8
PDS [9] (%)	100	100	100	0.8	0	0	0	0	0	0	0	0	0	0	0	0
MC [15] (%)	100	100	76.4	54.3	35.1	14.2	6.7	0.6	0	0	0	0	0	0	0	0
Hamming (%)	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

These results show how our proposed technique provides single and double-error correction, but can also provide effective tolerance capabilities against large MCUs that exceed the performance of other codes.

TABLE II
MTTF ($M = 32$)

λ (Upsets/bit per Day)	DMC	PDS [9]	MC [15]	Hamming
10^{-4}	1121.9	915.0	725.6	247.7
10^{-5}	11218.8	9150.3	7256.5	2477.4



TABLE III
AREA, POWER, AND DELAY ANALYSIS OF ENCODER AND DECODER

ECC Codes	Area		Power		Delay	
	μm^2	%	mw	%	ns	%
DMC	41572.6	100	10.8	100	4.9	100
PDS* [9]	486778.1	1170.9	221.1	2047.2	18.7	381.6
MC [15]	77933.7	187.5	24.7	228.7	7.1	144.9
Hamming	58409.4	140.5	20.5	189.8	6.7	136.7

*Using parallel decoder instead of serial decoder for fair comparisons

TABLE IV
REDUNDANT BITS (32-bit)

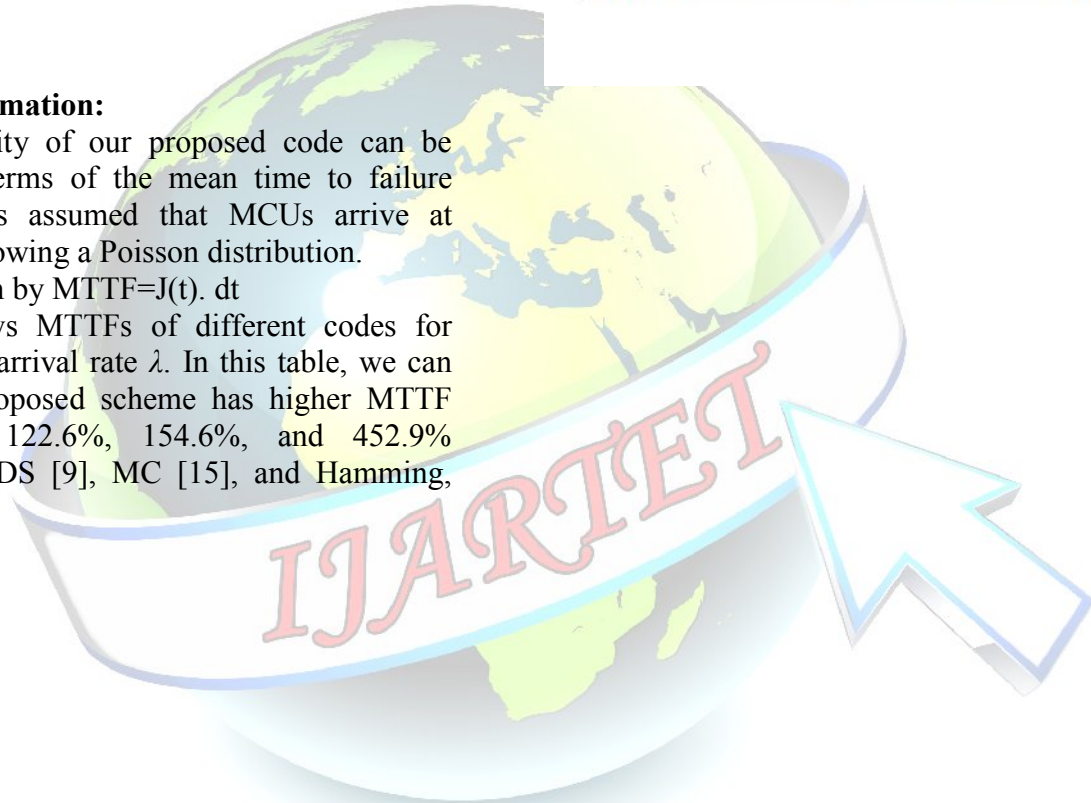
ECC	Information Bits	Redundant Bits	β	Note
DMC	32	36	52.9%	$k = 2 \times 4, m = 4$
DMC	32	32	50.0%	$k = 4 \times 4, m = 2$
PDS [9]	32	19	37.3%	Shorting and puncturing
MC [15]	32	28	46.7%	Correction capability is 2
Hamming	32	7	17.9%	Correction capability is 1

Reliability Estimation:

The reliability of our proposed code can be analyzed in terms of the mean time to failure (MTTF). It is assumed that MCUs arrive at memories following a Poisson distribution.

MTTF is given by $\text{MTTF} = \int_0^\infty J(t) dt$

Table II shows MTTFs of different codes for different event arrival rate λ . In this table, we can see that the proposed scheme has higher MTTF by more than 122.6%, 154.6%, and 452.9% compared to PDS [9], MC [15], and Hamming, respectively





In general cases, for proposed technique it can be inferred that the larger the word widths, the higher the tolerance capabilities and better the reliabilities.

C. Overheads Analysis:

The area and power overheads of PDS are 1170.9% and 2047.2% of the proposed scheme, respectively. The delay overhead of DMC is 26.2%, 69.0%, and 73.1% of PDS [9], MC [15], and Hamming respectively. This indicates that the memory with the proposed scheme performs faster than other codes. Different decoding algorithms could result in different overheads.

The decoding algorithm of PDS [9] is more complex than that of other codes. Thus, it has maximum area, power, and delay overheads. However, for the proposed DMC, its decoding algorithm is simple with minimal overhead. The issue is that the proposed technique requires more redundant bits compared with other codes.

IV. CONCLUSION

In this paper, novel per-word DMC was proposed to assure the reliability of memory. The proposed protection code utilized decimal 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 MCUs in memory. Besides, the proposed decimal error detection technique is an attractive opinion to detect MCUs in CAM because it can be combined with BICS to provide an adequate level of immunity. The only drawback of the proposed DMC is that more redundant bits are required to maintain higher reliability of memory, so that a reasonable combination of k and m should be chosen to

maximize memory reliability and minimize the number of redundant bits based on radiation experiments in actual implementation. Therefore, future work will be conducted for the reduction of the redundant bits and the maintenance of the reliability of the proposed technique

REFERENCES

- [1] D. Radaelli, H. Puchner, S. Wong, and S. Daniel, "Investigation of multi-bit up sets in a 150nm technology SRAM device," *IEEE Trans. Nucl. Sci.*, vol. 52, no. 6, pp. 2433–2437, Dec. 2005.
- [2] E. Ibe, H. Taniguchi, Y. Yahagi, K. Shimbo, and T. Toba, "Impact of scaling on neutron induced soft error in SRAM from 250nm to a 22nm design rule," *IEEE Trans. Electron Devices*, vol. 57, no. 7, pp. 1527–1538, Jul. 2010.
- [3] 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.
- [4] 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.
- [5] S. Liu, P. Reviriego, and J. A. Maestro, "Efficient majority logic fault detection with difference-set codes for memory applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 1, pp. 148–156, Jan. 2012.
- [6] M. Zhu, L. Y. Xiao, L. L. Song, Y. J. Zhang, and H. W. Luo, "New mix codes for multiple bit up sets 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 up sets 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, “Inter leaving 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 address able 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 bitup set correction,” *IEEE Trans. Rel.*, vol. 60, no. 3, pp. 528–537, Sep. 2011.

Authors Profile



Miss. T. DHIVYA pursuing Master degree in VLSI Design from Sri Eshwar College of Engineering, Coimbatore. Area of interests are Low Power VLSI and Testing of VLSI.



Mr. M. MOHANKUMAR is Assistant professor at Sri Eshwar College of Engineering, Coimbatore. His Specialization area is VLSI Design.



Mr. SWAMINATHAN VEERAPANDIAN is working as network executive in Tata Communications Ltd. His specialization area is Networking Systems.