A novel error correction technique for adjacent errors

C. Argyrides, P. Reviriego, D. K. Pradhan and J. A. Maestro

Abstract—Memories are one of the most widely used elements in electronic systems, and their reliability when exposed to Single Events Upsets (SEUs) has been studied extensively. As transistor sizes shrink, Multiple Bits Upsets (MBUs) are becoming an increasingly important factor in the reliability of memories exposed to radiation effects. To address this issue, Built-in Current Sensors (BICS) or Parity codes have recently been applied in conjunction with Single Error Correction/Double Error Detection (SEC-DED) codes to protect memories from MBUs. In this paper, this approach is taken one step further, proposing specific codes optimized to provide protection against errors in adjacent bits in memories. By exploiting the locality of errors within an MBU and the error detection and location capabilities of parity codes, the proposed codes result in both a better protection level and a reduced cost. This techniques improves memory's reliability by 40X compared to Hamming Codes (HC) and 2X the MTTF compared to Reed Muller Codes (RMC) for clustered MBUs

***Index Terms*—Reliability, Testing, and Fault-Tolerance**

I INTRODUCTION

As process technology scales to small nanometers, high-density, low cost, high performance integrated circuits, characterized by high operating frequencies, low voltage levels and small noise margins will be increasingly susceptible to temporary faults [11]. In very deep sub-micron technologies single-event upsets like atmospheric neutrons and alpha particles severely impact field-level product reliability, not only for memory, but for logic as well. When these particles hit the silicon bulk, they create minority carriers which if collected by the source/drain diffusions, could change the voltage level of the node.

Transient faults are also a major concern in space applications, with potentially serious consequences for the spacecraft, including loss of information, functional failure or loss of control [3]. Although SEU is the major concern in space and terrestrial applications, multiple bit upsets (MBU) are also became important problem in designing memories because of these points:

1) The error rate of memories increased due to the continuing technology shrinkage [9][6]. Therefore the probability of having multiple errors increases.
2) MBUs can be induced by direct ionization or nuclear recoil after passing a high-energy ion [10] ,
3) The experiments in memories under proton and heavy ions fluxes in [12], [16] show the probability of having multiple errors is increased when the size of memory is increased.

Unfortunately, packaging and shielding cannot effectively be used to shield against SEUs and MBUs since they may be caused by neutrons which can be easily penetrate through packages [20][9].

The most common approach to deal with multiple errors has been the use of interleaving in the physical arrangement of the memory cells, so that cells that belong to the same logical word are separated. As the errors in an MBU are physically close as discussed in [19] they will cause single errors in different words that can be corrected by the SEC-DED codes.

However, interleaving cannot be used, for example, in small memories or register files, and in other cases, its use may have an impact on floor-planning, access time and power consumption, as discussed in [5]. For those reasons, the use of more sophisticated codes or the combination of different codes has been proposed in [14] to deal with MBUs when the use of interleaving is not a valid option. More recently, codes that can correct multiple errors only when they are adjacent have also been proposed in [5]. These codes are tailored to the specific patterns of errors in an MBU (again the errors will tend to be physically close) and therefore can achieve effective protection at a reduced cost.

An alternative approach to protect memories is the use of Built in Current Sensors (BICS) that are able to detect the occurrence of errors by detecting changes in the current, as proposed in [21] and [13] . The sensors are placed in the columns of the memory block and they detect unexpected current variations on each of the memory bit positions.

The protection can be optimized with protection codes that are tailored to the specific problem of a memory protected that suffers MBUs. This is the objective of this paper in which such codes are proposed.

The article is organized as follows. The related work covering techniques to cope with MBUs is presented in section

This work was supported by the Spanish Ministry of Science and Education under Grant ESP-2006-04163, the Regional Government of Madrid and the European Union FEDER programme.

C. Argyrides is with the Department of Computer Science at the University of Bristol, Bristol, UK. (costas@cs.bris.ac.uk)
P. Reviriego Departmento de Ingenieria Informatica, Universidad Antonio de Nebrija, Calle Pirineos 55, 28040 Madrid, Spain; (previrie@nebrija.es)
D. K. Pradhan is with the Department of Computer Science at the University of Bristol, Bristol, UK. (pradhan@cs.bris.ac.uk)
J. A. Maestro Departmento de Ingenieria Informatica, Universidad Antonio de Nebrija, Calle Pirineos 55, 28040 Madrid, Spain; (jmaestro@nebrija.es)

II. In section III, the proposed codes are introduced and their correction capabilities. Then, in section IV, the reliability and the cost analysis of the proposed codes analyzed in detail and compared with the traditional coding codes. Finally the conclusions of the work are presented in section V.

II Related Work

In [7], [21] and in [4] authors proposed the BICS in conjunction with codes to deal with MBUs in memories as an alternative to the traditional combination of interleaving, parity and SEC-DED codes. This new application of BICS would be useful in situations where the use of interleaving is not possible or appropriate as discussed before. The combination of BICS with error protection codes can correct one or two errors per word. In particular the codes can detect that there have been an error in the words since there are only one or two errors per-word in the worst case for parity and SEC-DED respectively. Using the additional information from the BICS about the faulty columns, the errors could be located, and combining that information with the one from the codes, the errors could be corrected in most cases.

Most recently [18], authors combine vertical BICS with multiple parity bits per word to guarantee the correction of multiple errors cause by an MBU (bounded). The number of extra parities depends on the maximum distance between errors in an MBU as shown in [15], [17] and [19].

Another way to cope with MBU was reported in [1]. In this paper authors avoid the use of BICS and combine the SEC-DED and Parity codes to protect SRAM memories against MBUs. In this work, they guarantee the correction of any double errors even if the errors are not bounded. A new approach that combines the work proposed in [1] and in [18] is presented in this paper.

III Proposed Technique

The traditional assumptions when dealing with MBUs in memories, it can be assumed that the maximum distance between errors in an MBU is bounded by a certain value, as shown in [15], [17] and [19]. In [18] authors considered only the maximum horizontal distance and they add BICS vertically. With this technique they are able to correct all single errors and multiple errors with distance equal or less than L.

The proposed approach is shown in Figure 1.

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	C_1	C_2
X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}	C_3	C_4
X_{17}	X_{18}	X_{19}	X_{20}	X_{21}	X_{22}	X_{23}	X_{24}	C_5	C_6
X_{25}	X_{26}	X_{27}	X_{28}	X_{29}	X_{30}	X_{31}	X_{32}	C_7	C_8
D_1	D_3	D_5	D_7	D_9	D_{11}	D_{13}	D_{15}		
D_2	D_4	D_6	D_8	D_{10}	D_{12}	D_{14}	D_{16}		

Figure 1: Logical organization of 32bits word

where C_i and D_j are parity bits.

$$C_1 = X_1 \oplus X_3 \oplus X_5 \oplus X_7 \quad (1)$$

$$C_2 = X_2 \oplus X_4 \oplus X_6 \oplus X_8 \quad (2)$$

and similar for the rest of the other C_i.

$$D_1 = X_1 \oplus X_{17} \quad (3)$$

$$D_2 = X_9 \oplus X_{25} \quad (4)$$

and similar for the rest of the other D_j.

In the proposed technique, we arrange the bits in a matrix way and we add parity bits horizontally and vertically. The logical organization of the bits is illustrated in Figure 1. By rearranging the word into a matrix way and following the technique from [18], we can have correction up to four distance-1 or adjacent errors.

Note that, the proposed technique may be extended for larger code words, like 64 bits words and improve the protection against larger MBUs by setting the word as an 8x8 matrix with 4 parity bits per row and column. Such that the parity bits are for example for the first row are:

$$C_1 = X_1 \oplus X_5 \quad (5)$$

$$C_2 = X_2 \oplus X_6 \quad (6)$$

$$C_3 = X_3 \oplus X_7 \quad (7)$$

$$C_4 = X_4 \oplus X_8 \quad (8)$$

and for the first row

$$D_1 = X_1 \oplus X_{33} \quad (9)$$

$$D_2 = X_9 \oplus X_{41} \quad (10)$$

$$D_3 = X_{17} \oplus X_{49} \quad (11)$$

$$D_4 = X_{25} \oplus X_{57} \quad (12)$$

This could correct MBUs whose errors are clustered up to a distance of three.

IV Reliability And Cost Analysis

In order to estimate the error detection/ correction coverage of the proposed technique and previous one, we used fault injection method. Fault injection is one of the key methods to estimate the error study of the circuits which utilized error detection and correction codes [1]. Using fault injection method, the coverage of the proposed method was estimated for two scenarios: independent errors and clustered errors. The second one is closer to the errors caused by real MBUs.

IV.1 Fault Injection Experiments

Without loss of generality, we considered the coverage of the proposed technique for a 32 bit data word since the protection code can be applied on each data word of a given memory. We assume two different word sizes of 16 and 32 bits. Both single and multiple faults were injected. For each number of errors, we used random multiple fault injection and about 10.000 experiments for each case were conducted. The obtained values are portrayed in Table 1 for RMC [2], HC [8]. For each protection method we illustrate the protection

coverage. The first column shows the number of faulty bits in a word.

Table 1: Correction Coverage, using random faults

Number of faults	RMC (2,5)	HC (16,26)	HC (32,39)	Prop 16	Prop 32
1	100%	100%	100%	100%	100%
2	100%	0%	0%	0%	0%
3	100%	0%	0%	0%	0%
4	0%	0%	0%	0%	0%

The errors caused by an MBU are physically close as discussed in [19], we run a second set of fault injection experiments based on this assumption. In the next fault injection experiment we inject up to four errors with distance one of each other and report the correction coverage of the proposed technique. As we can observe from this table, all errors were corrected and the correction coverage of the proposed technique is better than HC, and even better than RMC. Examples of the patterns used for this analysis are illustrated in Figure 2.

These results show how the proposed technique provides single error correction as HC but can also provide effective protection against errors that are physically close exceeding in this case the performance of more sophisticated codes like RMC.

Table 2: Correction Coverage, using distance-1 faults

Number of faults	RMC (2,5)	HC (22,16)	HC (39,22)	Prop 16	Prop 32
1	100%	100%	100%	100%	100%
2	100%	0%	0%	100%	100%
3	100%	0%	0%	100%	100%
4	0%	0%	0%	100%	100%
5	0%	0%	0%	0%	0%

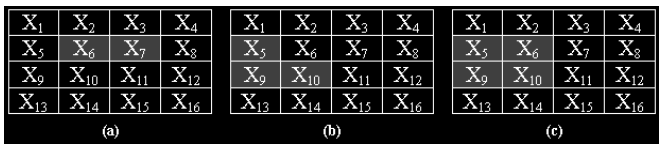

Figure 2: Example of patterns used in the distance-1 faults

IV.2 Reliability Analysis

In order to analyze the capability of fault detection and correction of the mentioned protection codes, it is required to use the values in Tables 1 and 2 with their corresponding probabilities. For this purpose, we assume the following statements which were also assumed in [1]:

1. Transient faults occur with a Poisson distribution.
2. Bit failures are independent for the first type of experiments and clustered with distance 1 for the second type.

In Figure 3 and Figure 4 we can see how the reliability improves using our technique in the scenario that the errors caused by an MBU are physically closed. In Figure 3 the reliability of a 16bits word protected HC, RMC and the proposed technique is illustrated. In Figure 4 the memory's reliability while protected using the proposed technique, RMC and HC is also illustrated. In Figure 4 we assume that the fault rate $\lambda=10^{-4}$, and we can see that our technique improves memory reliability compared to the RMC technique by 2X while improves reliability compared to HC by more than 40X.

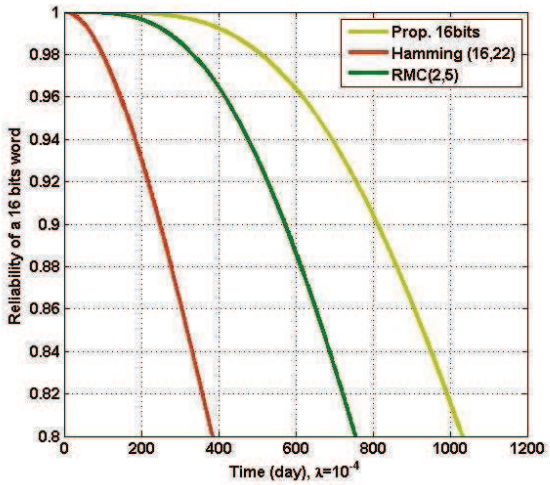

Figure 3: Reliability of the word, $\lambda=10^{-4}$ (distance-1 faults)

Table 3: MTTF in days for random fault injection

Fault Rate	Prop 16bits	Prop 32bits	HC 16bits	HC 32bits
$\lambda=10^{-1}$	0.02	0.01	0.003	0.0026
$\lambda=10^{-2}$	0.17	0.12	0.033	0.0261
$\lambda=10^{-3}$	1.67	1.19	0.329	0.2609
$\lambda=10^{-4}$	16.68	11.90	3.290	2.6092
$\lambda=10^{-5}$	166.76	118.99	32.870	26.0923
$\lambda=10^{-6}$	1.676e+003	1.19e+003	328.720	260.9230

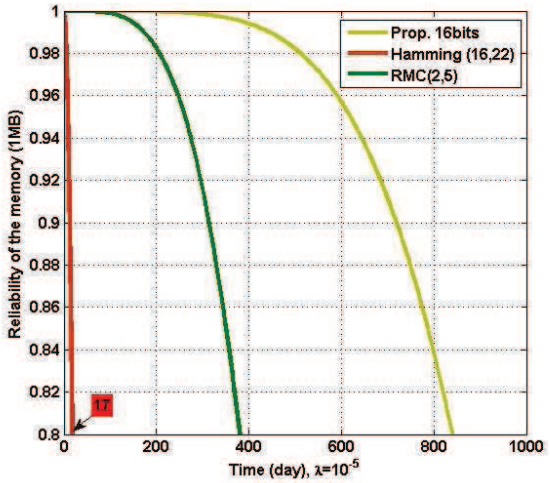

Figure 4: Reliability of the 1Mbit memory, $\lambda=10^{-5}$ (distance-1 faults)

In Table 3 we portray the mean time to failure of the proposed technique and HC for the 1MBit memory in different fault rates in the random faults.

Table 4 portrays the mean time to failure of the proposed technique, RMC and HC codes with 16bits code word for the 1MBit memory in different fault rates in the scenario of

distance-1 faults. In this table we can see that the proposed technique improves the memory's MTTF by more than 32X compared to HC and more than 2X compared to RMC.

Table 4: MTTF in days for distance-1 faults

Fault Rate	Prop 16bits	HC. (22,16)	Impro Over HC	RMC (2,5)	Impro Over RMC
$\lambda=10^{-4}$	105.61	3.29	32.13X	50.68	2.08X
$\lambda=10^{-5}$	1.06E+03	32.87	32.12X	506.80	2.08X
$\lambda=10^{-6}$	1.06E+04	328.72	32.12X	5.07E+03	2.08X

IV.3 Cost of the Technique

In Table 5 we can see the number of bits required implementing this coding technique and the number of bits required to implement classic coding techniques. From this table we can see that HC has the least overhead as well as the worst coverage, reliability and MTTF performance as shown in Table 1 to Table 4 and Figure 3 and Figure 4. The shortened version of RMC requires slightly less number of redundant bits but it has a more complicated decoding circuitry [2] compared the decoding circuitry of simple parity codes. Finally the Matrix coding requires more redundant bits than the proposed technique. Therefore the proposed codes are an interesting option to protect memories from clustered MBUs when interleaving can not be used as they achieve larger MTTFs with reduced cost.

Table 5: Cost of the technique (Extra bits) for L=2

N	Proposed	HC	Matrix [1]	RMC [2]
16	16	6	20	16
32	24	7	28	32 (22*)
64	32	8	48	64 (29*)

* Based on shortened version of Reed Muller coding.

V CONCLUSIONS

In this paper we proposed a novel technique to cope with four errors cause by MBUs. We have shown that our technique improves the reliability and the MTTF of the memory by more than 40X compared to traditional HC technique and more than 2X compared to RMC when clustered MBU are considered. The cost of the technique is less than traditional Reed Muller codes. At the same time the proposed technique can also correct all single errors and therefore can provide effective protection against single errors and MBUs.

REFERENCES

[1] Argyrides C., Zarandi H., Pradhan D.K. "Matrix Codes: Multiple Bit Upsets Tolerant Method for SRAM Memories" DFT 2007, September 2007. 340-348

[2] Argyrides, C., Loizidou, S., and Pradhan, D. K.. Area Reliability Trade-Off in Improved Reed Muller Coding. In *Proceedings of the 8th international Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation* (Samos 08), Greece, July 21 - 24, 2008. 116-125. 2008

[3] Barth J.L., Dyer C.S., Stasinopoulos E.G., "Space, atmospheric, and terrestrial radiation environments", *IEEE Trans. Nuclear Science*, Vol 50, pp 466-482,June 2003

[4] Calin, T., Vargas F. L., and Nicolaidis, M. Upset-tolerant CMOS SRAM using current monitoring: prototype and test experiments. In *Proceedings of the International Test Conference*. 45–53. 1995.

[5] Dutta, A. and Touba, N. A. Multiple bit upset tolerant memory using a selective cycle avoidance based SEC-DED-DAEC code. In *Proceedings of the IEEE VLSI Test Symposium,* 349–354. 2007

[6] Ferreyra P. A., Marques C. A., Ferreyra R. T., and Gaspar J. P., "Failure map functions and accelerated mean time to failure tests: New approaches for improving the reliability estimation in systems exposed to single event upsets," *IEEE Trans. Nucl. Sci.*, Vol. 52, No. 1, pp. 494-500, 2005.

[7] Gill, B., Nicolaidis , M., and Papachristou, C.. Radiation induced single-word multiple-bit upsets correction in SRAM. In *Proceedings of the IEEE International OnLine Testing Symposium*. 266–271. 2005

[8] Hamming R.W. "Error Detecting and Error Correcting Codes" The Bell Systems Technical Journal, Vol XXVI (2), April, 147-161. 1950,

[9] Hazucha P., Svensson C., "Impact of CMOS technology scaling on the atmospheric neutron soft error rate," *IEEE Trans. Nucl. Sci.*, Vol. 47, No. 6, pp. 2586-2594, Dec. 2000.

[10] Hentschke R., Marques R., Lima F., Carro L., A. Susin, R. Reis, "Analyzing Area and Performance Penalty of Protecting Different Digital Modules with Hamming Code and Triple Modular Redundancy", *Symposium on Integrated Circuits and Systems Design*, pp. 95-100, 2002.

[11] International Technology Road map for Semiconductors, *http://public.itrs.net/* , 2002.

[12] Karlsson J., Liden P., Dahlgren P., Johansson R., Gunneflo U.,"Using Heavy-Ion Radiation to Validate Fault-Handling Mechanisms," *IEEE Micro*. Vol. 14, pp. 8-23, 1994.

[13] Lo, J. 2002. Analysis of a BICS-only concurrent error detection method. *IEEE Trans. Comput.51*, 3, 241–253.

[14] Neuberger, G., De Lima, F., Carro, L., and Reis, R. A multiple bit upset tolerant SRAM memory. *ACM Trans. Desi. Automat. Electr. Syst. 8*, 4, 577–590. 2003.

[15] Radaelli D., Puchner H., Wong S., and Daniel S. Investigation of multi-bit upsets in a 150 nm technology SRAM device. *IEEE Trans. Nucl. Sci. 52*, 6, 2433–2437. 2005.

[16] Reed R., "Heavy Ion and Proton Induced Single Event Multiple Upsets", *IEEE Nuclear and Space Radiation Effects Conference*, pp. 2224-2229, 1997.

[17] Reviriego, P. and Maestro, J. A. and Cervantes C. Reliability analysis of memories suffering multiple bit upsets. *IEEE Trans. Device Materials Reliabil. 7*, 4, 592–601. 2007.

[18] Reviriego, P. and Maestro, J. A. Efficient error detection codes for multiple-bit upset correction in SRAMs with BICS.*ACM Trans. Des. Autom. Electron. Syst.* 14, 1 (Jan. 2009), 1-10 2009.

[19] Satoh S., Tosaka Y., and Wender S. A. Geometric effect of multiple-bit soft errors induced by cosmic ray neutrons on DRAM's. *IEEE Electron Device Lett. 21*, 6, 310–312. 2000.

[20] Seifert N., Moyer D., Leland N., Hokinson R., "Historical trend in alpha-particle induced soft error rates of the Alpha microprocessor," *Proc. 39th Annu. IEEE Int. Reliab. Phys. Symp.*, pp. 259-265, 2001.

[21] Vargas F., Nicolaidis M, and Courtois B. Quiescent current monitoring to improve the reliability of electronic systems in space radiation environments. In *Proceedings of the IEEE International Conference on Computer Design (ICCD),* 596–600. 1993.