TMR Group Coding Method for Optimized SEU and MBU Tolerant Memory Design

Yi Jin[1], Yuxiang Huan[1,2], Haoming Chu[1], Zhuo Zou[1], Lirong Zheng[1,2]
[1]Fudan University, Shanghai, China;
[2]Royal Institute of Technology, Stockholm, Sweden;
Email: {jin_y16, yhuan13, hmchu16, zhuo, lrzheng}@fudan.edu.cn

***Abstract*—This work proposes a fault tolerant memory design using the method of Triple Module Redundancy (TMR) group coding to tolerant the Single-Event Upset (SEU) and Multi-Bit Upset (MBU) influence on memory devices in space environment. The group coding method uses different models to partition and code each word line in memory with Hamming code to achieve best performance. TMR group coding method further increases the capability of self-correction for the errors occurred in parity bits. The evaluation results show that the suggested approach can obtain improved correctness for the memory output with optimized tradeoff between reliability and cost. At 5% error rate, the probability of correct output reaches 70.78% with small cost increment. To achieve 90% reliability, the accuracy improvement is 31.9% compared to TMR with 9% increased area. This solution proposed is evaluated on the memory rich micro-coded processor, but can be further extended to other memory-based processors that need high reliability for the SEU and MBU influence in aerospace applications.**

I. Introduction

There is a growing demand for high reliable smart chips consisting of more Commercial Off-The-Shelf (COTS) components for aerospace applications nowadays. As the requirements for processing performance continuously increasing, many processors with more memories or based on processing-in-memory are proposed [1], [2], such as the Field-Programmable Gate Array (FPGA) and the micro-coded processor for IoT applications [3].

Single-Event Effects (SEE) have become the main inducing factor for soft errors in nanoscale electronic devices in space, including Single-Event Upset (SEU), and Multi-Bit Upset (MBU) [4], [5]. An ionizing energetic particle may strike electronic components and cause the bit flips in CPU or memory, resulting in system failures [6], [7]. In electronic devices, memory is one of the most vulnerable parts to SEU and MBU. As transistor sizes shrink, they have become two crucial factors influencing the reliability of memories exposed to space radiations.

The reliability of memory can to a great extent affect the processing performance of the chip, including the accesses and executions of the instructions. Therefore, a highly reliable memory is very crucial for the design of aerospace applications processors. The improvement of the memory reliability can greatly reduce the influence of radiation interference such as SEU or MBU, meanwhile save the costs. To solve this problem, in gate level, many researches have focused on various error detection and correction codes to rectify single bit error.

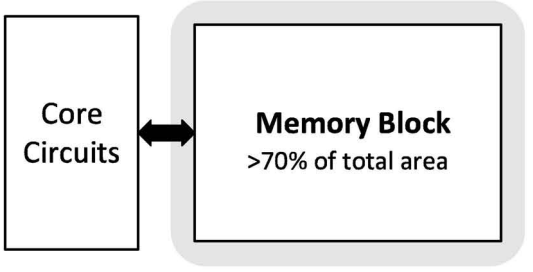

Fig. 1. Diagram for the memory rich processor

While in circuit level, the goal is to ensure the bits error does not influence the final output value. The conventional software solution that uses the off-chip backup results in large time delay and less reliability while the hardware on-chip backup will take up more area and increase the costs.

Triple Module Redundancy (TMR) and Error Correcting Code (ECC) are two commonly used fault tolerant approaches. TMR can ensure the right output in the case of single module fault, but it requires high hardware redundancy. The hardware cost for ECC is less while its ability to detect and correct soft errors is limited [8]. The group coding method in this work uses Hamming code in view of its ability to correct single error with low redundancy and simple hardware implementation with small delay [9].

This work combines TMR with the group coding methods to improve the accuracy of the entire memory block at a relatively small overhead. Experimental results show that the TMR-G3 model with bit voter can obtain the output accuracy of 70.78% at 5% error rate. The optimized fault tolerant design demonstrated outperforms the simple implementation of TMR or Hamming coding method, with an optimal tradeoff between cost and reliability.

II. System design

A. Proposed TMR Group Coding Method

The goal of this work is to improve the reliability compared to existing fault tolerant methods at a small overhead. Experiments are simulated and verified on the memory rich micro-coded processor whose 120KB memory block covers over 70% of the total chip area as shown in Fig. 1. It can also be implemented on other processors with big memories.

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

TABLE I
GROUP CODING MODELS

Hamming Model	Data word length	Check word length	Segment number	Total bits (considering zero filling)	Maximum number of errors can be corrected (in different segments)
Hamming-3	4	3	20	140	20
Hamming-4	11	4	8	120	8
Hamming-5	26	5	4	124	4
Hamming-k	2^k-k-1	k	[Total bits/(2^k-1)]	[Total bits/(2^k-1)] * (2^k-1)	[Total bits/(2^k-1)]

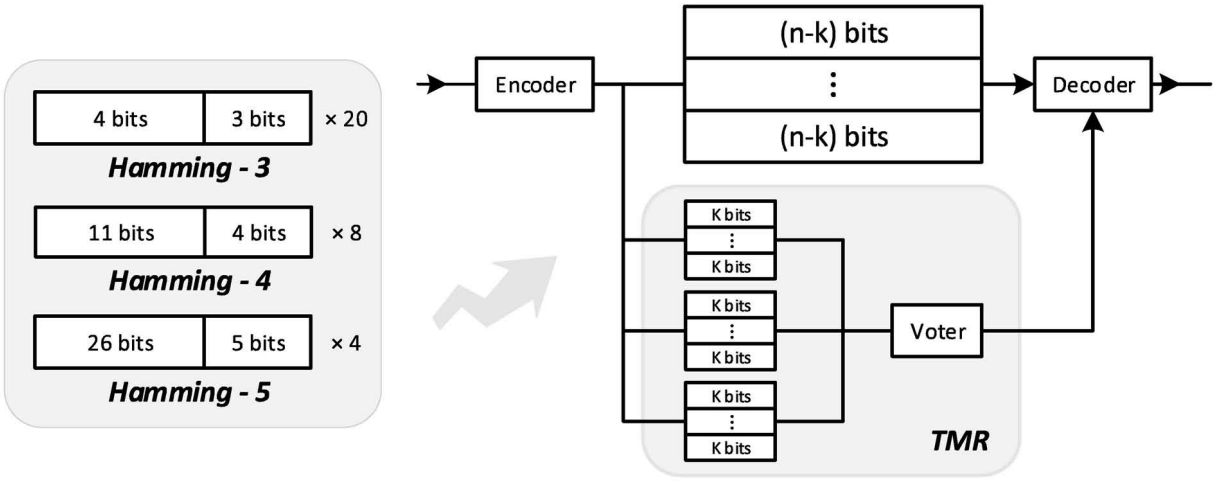

Fig. 2. Architecture of TMR group coding

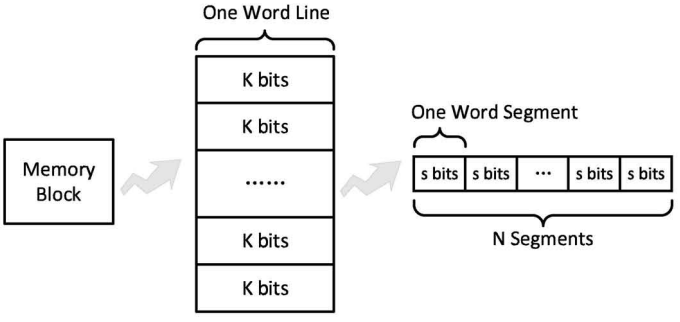

Fig. 3. Block partitioning for group coding

Fig. 2 demonstrates the architecture of the TMR group coding method proposed in this work.

Because of the need to tolerant the SEU and MBU effects, word lines should be able to self-detect and correct multiple bits errors. Therefore, the group coding method is proposed to divide each word line into N small segments, with each segment consisting of s bits for separate coding, as demonstrated in Fig. 3. Only the parity bits here are transferred to the TMR block, therefore the original code word remains unchanged after encoding.

Hamming code and Reed-Solomon (RS) code are two of the most commonly used ECC methods whose code word remains unchanged after encoding. Hamming code has the advantages of easy circuit implementation and low cost, while it can only detect up to two errors and correct one error. RS code has the correcting capability of $(n-k)/2$, where n is the code word length and k is the check word length [10]. But the encoder and the decoder of RS code are much more complex than Hamming code. Many multipliers and registers are used, causing unwished delay and increasing cost. As a result, considering the need to keep the code word unchanged and the need for less delay, Hamming code is used in this work for group coding.

The common (n,k) Hamming code has n bits in total, consisting of k parity bits and $(n-k)$ data bits. The parity bits increases the overhead of overall system. Since the Hamming code can correct no more than one bit error, it can only tolerant the error rate of 1.15% if the total 80 bits in one word line are encoded together. Therefore MBU faults cannot be corrected.

In the verification, each word line in the memory block consists of 80 bits data [11]. The proposed TMR group coding method can also be applied to various memory sizes using different partitioning models. There are three group coding models applicable for the micro-coded processors used in this work as shown in Table I. For other applications whose word line consists of more bits, there are more usable group coding models. When k increases, the redundancy and expenses decreases. However, the probability of error happening in one segment also becomes larger since the vulnerable area for each segment exlarges [12]. With the group coding, the whole word

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

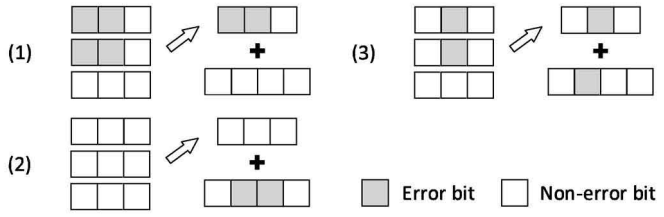

Fig. 4. Error pattern for TMR-G3

line can tolerant no more than N bits errors on condition that they distributes in different segments.

The group coding and TMR are combined to further improve the reliability of the memory. The basic TMR model uses a voter and three modules with same size to execute one task at the same time, and gives the final output according to the result of the two-of-three voter. TMR has two types of voters, which are the bit voter that takes each bit of the three modules for comparison and the word voter that compares each word line for output, named as TMR-B and TMR-W respectively. Both of them are testes for the overall performance analysis.

When the whole memory block is triplicated for TMR, the total cost is at least three times of the original size. Instead of triplicating the whole block, this work uses three redundancy modules only for the parity parts to save cost. As shown in Fig. 2, the input data is encoded with Hamming code. The encoded data is separated into two parts as the original bits and the parity bits. TMR is applied on the latter to get the triplicated redundancy for two-of-three voting, therefore achieving the improved reliability with less cost. The cost can be reduced if the parity part is transmitted to TMR instead of the whole word line. In this paper, TMR group coding methods with Hamming-3, Hamming-4, and Hamming-5 are named as TMR-G3, TMR-G4, and TMR-G5 respectively.

B. Error Correction Pattern

TMR group coding can tolerate more than N errors according to the error pattern. When two or more errors happened in one segment, the reliability can be varied according to how those bit upsets distribute. TMR block can tolerant some errors occurred in the parity bits as shown in Fig. 4, taking TMR-G3 as example. In each segment, only three cases can cause a fault output with the TMR group coding.

1) More than one errors in the voter output
2) More than one errors in the data bits
3) No less than one error in the data bits and no less than one error in the voter output

The probabilities of error patterns vary with different group coding models and word length. Therefore the reliabilities and costs for models are verified to obtain the best tradeoff.

III. Evaluation Results

The study for the reliability of 28-nm SRAM cells from Artix-7 FPGA in [13] shows that below 10 MeV $\cdot$ cm^2 $\cdot$ mg^{-1}, the average number of upsets/total bits is less than 10^-3 level. One bit upset, two bits upsets, and three bits upsets sum up to over 90 percent of all the upsets occurred. It was verified in 24 bits frames employed in 45nm SRAM memory [14]. In real aerospace circumstance, the radiation energy may differ according to the working attitude and angles. According to the effects pattern of SEU and MBU, this work tests the reliabilities and the hardware costs for different models under the error rate up to 5 percent.

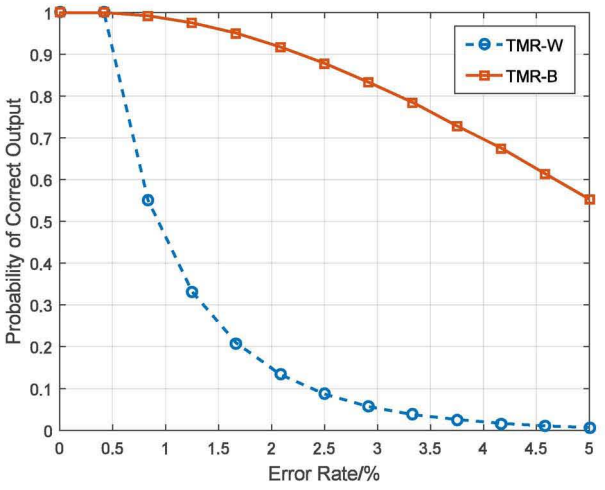

Fig. 5. Probability of correct output with TMR-B and TMR-W

A. Reliability Analysis

The probability of correct output decreases as the error rate increases in all models.

For TMR, the bit voter shows better performance compared to the word voter in the terms of the output correctness, but worse behavior for the output corruption because it may recognize the several same errors happened in one bit as correct [15]. According to the Fig. 5, the probability for word-voter declines sharply when the error rate increases from zero to five percent, while for bit voter it can maintain 55.6% accuracy.

With the check bits/data bits ratio changing for different Hamming-k models, the total size of each word section vary. The total bits in one word line increases with the word length, causing more bits error in the memory block at the same error rate. Larger k leads to lower accuracy of the output results as shown in Fig. 6. Hamming-3 has the best reliability compared to Hamming-4 and Hamming-5. Yet its area cost is the highest.

With same group coding method, the TMR-B model still shows better reliability than the TMR-W model. The TMR-G3 shows the best output correctness as shown in Fig. 7. It achieves the accuracy of 70.78% at 5% error rate. For output reliability, the TMR-G3 shows an improvement of 27.3% compared with the TMR-B model and 97.6% compared with the Hamming-3 model.

978-1-5386-4881-0/18/$31.00 ©2018 IEEE

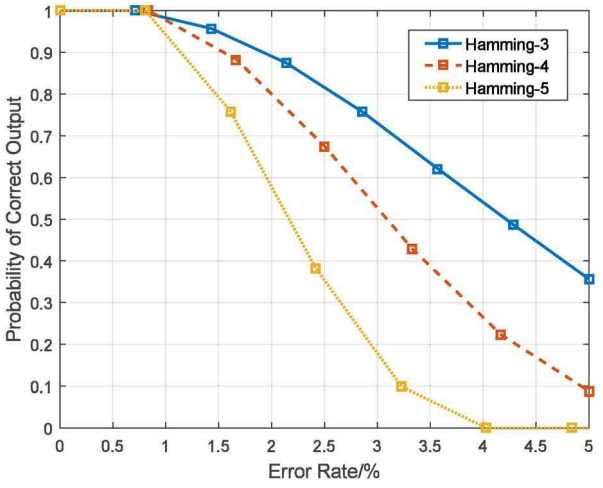

Fig. 6. Probability of correct output with Hamming code

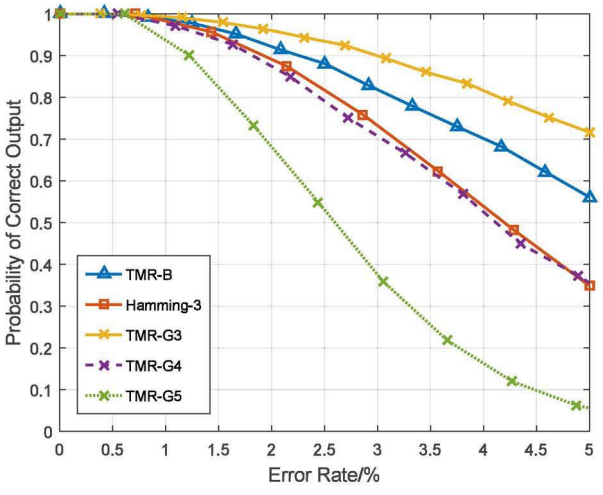

Fig. 7. Reliability comparison for TMR group coding and other methods

B. Tradeoff between Reliability and Cost

The costs are simulated by the Design Compiler, using the micro-coded processor with 120KB memory.

Table II and Fig. 8 summarize corresponding reliabilities and costs for models verified. TMR-G3 shows highest error tolerance with 2.98% error rate to achieve 90% reliability, where the improvement is 31.9% compared to TMR-B and 56.5% to Hamming-3. Yet the cost for TMR-G3 increases by 9% compared with TMR-B. Considering the improvement for accuracy, the cost increment is acceptable for aerospace applications that require high reliability.

IV. Conclusion

Radiation effects on electronic devices have become one of the biggest concerns for aerospace chip design, especially for memory that is very vulnerable to SEU and MBU effects. Because of the requirement to design high reliability memory, this work proposes the TMR group coding method to improve the output accuracy. Group coding used in this design is applied with Hamming coding. With the feature of easy implementation and data word unchanged after encoding, Hamming coding enables the separation of the original data and the parity bits. TMR and the group coding are combined to further increase the ability of self-correction for the errors happened within the parity bits. The evaluation results show that compared to TMR, this approach can obtain improved correctness of 70.78% accuracy at 5% error rate. The improvement for reliability is 27.3%. To achieve 90% reliability, the improvement of reliability is 31.9% with 9% increased area. The method proposed optimizes the reliability and cost performance with the micro-coded processor, and can be used for future memory applications that require high reliability in space environment.

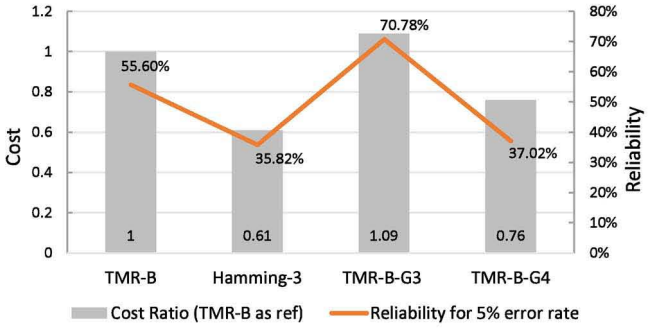

Fig. 8. Tradeoff between reliability and cost

TABLE II
SUMMARY FOR COMPARISON

Model	Reliability[a] /%	Error Tolerance[b] (/% rate)	Cost Area	Cost Ratio[c]
TMR-B	55.60	2.259	661506	1.00
TMR-W	0.72	0.5088	662539	1.00
Hamming-3	35.82	1.904	402007	0.61
Hamming-4	8.52	1.546	340846	0.52
Hamming-5	0	1.138	361929	0.55
TMR-G3	70.78	2.98	720583	1.09
TMR-G4	37.02	1.806	500132	0.76
TMR-G5	5.79	1.228	462627	0.70

[a]at 5% error rate [b]with 90% reliability [c]take TMR-B as reference

Acknowledgment

This work was supported by Shanghai Pujiang Program (No. 17PJ1400800), Shanghai Science and Technology Innovation Program (No. 17JC1401400), NSFC Grant (No. 61571137), the key project of Shanghai Science and Technology Innovation Plan (No. 15JC1400101), and JSSF project (BK20150164).

References

[1] F. Su, W. H. Chen, L. Xia, C. P. Lo, T. Tang, Z. Wang, K. H. Hsu, M. Cheng, J. Y. Li, and Y. Xie, "A 462gops/j rram-based nonvolatile in-

telligent processor for energy harvesting ioe system featuring nonvolatile logics and processing-in-memory," pp. T260–T261, 2017.

[2] M. Imani, S. Gupta, and T. Rosing, "Ultra-efficient processing in-memory for data intensive applications," in *Design Automation Conference*, 2017, p. 6.

[3] N. Ma, Z. Zou, Z. Lu, L. Zheng, and S. Blixt, "A hierarchical reconfigurable micro-coded multi-core processor for iot applications," in *International Symposium on Reconfigurable and Communication-Centric Systems-On-Chip*, 2014, pp. 1–4.

[4] M. Raine, G. Hubert, M. Gaillardin, P. Paillet, and A. Bournel, "Monte carlo prediction of heavy ion induced mbu sensitivity for soi srams using radial ionization profile," *IEEE Transactions on Nuclear Science*, vol. 58, no. 6, pp. 2607–2613, 2011.

[5] E. Petersen, *Single Event Effects in Aerospace*. Wiley-IEEE Press, 2011.

[6] W. He, Y. Wang, K. Xing, and W. Deng, "Single event effect vulnerability analysis and on-orbit error rate prediction," in *IEEE International Conference on Signal and Image Processing*, 2017, pp. 471–477.

[7] R. Rajaei, B. Asgari, M. Tabandeh, and M. Fazeli, "Design of robust sram cells against single-event multiple effects for nanometer technologies," *IEEE Transactions on Device & Materials Reliability*, vol. 15, no. 3, pp. 429–436, 2015.

[8] J. Guo, L. Xiao, and Z. Mao, "Novel low-power and highly reliable radiation hardened memory cell for 65 nm cmos technology," *IEEE Transactions on Circuits & Systems I Regular Papers*, vol. 61, no. 7, pp. 1994–2001, 2017.

[9] L. J. Saiz-Adalid, P. Gil, J. C. Baraza-Calvo, J. C. Ruiz, D. Gil-Tomas, and J. Gracia-Moran, "Modified hamming codes to enhance short burst error detection in semiconductor memories (short paper)," in *Tenth European Dependable Computing Conference*, 2014, pp. 62–65.

[10] P. Parvathi and P. R. Prasad, "Fpga based design and implementation of reed-solomon encoder & decoder for error detection and correction," in *Power, Control, Communication and Computational Technologies for Sustainable Growth*, 2016, pp. 261–266.

[11] Y. Huan, N. Ma, S. Blixt, and Z. Zou, "A 61 *uma*/mhz reconfigurable application-specific processor and system-on-chip for internet-of-things," in *IEEE International System-On-Chip Conference*, 2015, pp. 235–239.

[12] A. Neale and M. Sachdev, "Neutron radiation induced soft error rates for an adjacent-ecc protected sram in 28 nm cmos," *IEEE Transactions on Nuclear Science*, vol. 63, no. 3, pp. 1912–1917, 2016.

[13] J. Tonfat, F. L. Kastensmidt, L. Artola, G. Hubert, N. H. Medina, N. Added, V. A. P. Aguiar, F. Aguirre, E. L. A. Macchione, and M. A. G. Silveira, "Analyzing the influence of the angles of incidence and rotation on mbu events induced by low let heavy ions in a 28-nm sram-based fpga," *IEEE Transactions on Nuclear Science*, vol. PP, no. 99, pp. 1–1, 2017.

[14] M. Ebrahimi and M. B. Tahoori, "Stepped parity: A low-cost multiple bit upset detection technique," in *Test Conference*, 2015, pp. 1–8.

[15] T. B. Lo, F. L. Kastensmidt, and A. C. S. Beck, "Using configurable bit-width voters to mask multiple errors in integrated circuits," in *Vlsi*, 2015, pp. 533–538.

978-1-5386-4881-0/18/$31.00 ©2018 IEEE