

VLSI IMPLEMENTATION OF ERROR DETECTION AND CORRECTION CODES FOR SPACE ENGINEERING

J.VASUNDHARA

ASSISTANT PROFESSOR, ECE DEPARTMENT, SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY, ANANTAPUR

ABSTRACT: On behalf of technology scaling, on-chip memories in a die undergoes bit errors because of single events or multiple cell upsets by the ecological factors such as cosmic radiation, alpha, neutron particles or due to maximum temperature in space, leads to data corruption. Error detection and correction techniques (ECC) recognize and rectify the corrupted data over communication channel. In this paper, an advanced error correction 2-dimensional code based on divide-symbol is proposed to weaken radiation-induced MCUs in memory for space applications. For encoding data bits, diagonal bits, parity bits and check bits were analyzed by XOR operation. To recover the data, again XOR operation was performed between the encoded bits and the recalculated encoded bits. After analyzing, verification, selection and correction process takes place. The proposed scheme was simulated and synthesized using Xilinx ISE implemented in Verilog HDL. Compared with the well known existing methods, this encoding-decoding process consumes low power and occupies minimum area and delay.

1. INTRODUCTION

Artificial intelligence (AI) has become the focus of social attention. At present, the AI semiconductors field is in full bloom and storage is a major basic function of chips. If AI chips pursue performance, the memory itself must be fast enough and the bandwidth be large enough to exchange data quickly, which is inseparable with the progress of SRAM [1]. At the same time, the manufacturing process of integrated circuits (IC) has entered the nanoscale stage, and the small size and low voltage make circuit

nodes more and more sensitive to the impact of space high-energy particles. Data security and reliability have been of concern. With the exploration of space, integrated circuit devices have been gradually applied to aerospace devices. The aerospace microprocessors widely use static random access memory (SRAM) to store data and instructions, because of the advantages of high read/write rate and low power consumption [2]. The rapidly changing radiation environment in space causes errors in memory [3], which have different degrees

of impact, and seriously affect the performance and life of SRAM, which seriously threatens the regular operation of various spacecraft [4]. Therefore, it becomes more and more essential to improve the radiation resistance of SRAM.

SRAM usually adds ECC for error correction and error detection in anti-radiation reinforcement. In the process of rectifying soft mistakes, ECC technology does not require unique and sophisticated design in layout and storage unit circuit structure, and it is compatible with commercial memory. As a result, ECC is frequently utilized at this level for error repair and detection. However, the ECC reinforcement design usually brings new problems: timing violations, area increase, and power consumption increase [5]. The constraints on timing, area, and power consumption are becoming larger and larger with the continued increase of chip-scale and operating frequency. Taking into account various indicators and improving the energy-efficiency and hardware-cost ratio is a fundamental challenge for the current SRAM reinforcement technology.

Although the frequency and range of memory increase, the system still spends most of its time running without errors. However, each memory access must suffer

performance, power, and area cost for error coverage in traditional error detection and correction techniques. For instance, suppose we wish to prevent multi-bit errors using robust ECC, each word must hold additional bits for ECC detection, and each access must suffer the delay and power cost of accessing, computing, and comparing the codewords. So to a certain extent, traditional memory protection techniques are not suitable for detecting and recovering from failures because of the high overhead involved. Suppose we can decouple the none-error operation in the normal situation from the error-occurred operation in the uncommon case, and only pay the error protection expense when an error is detected. In that case, we can accomplish both minimal overhead and error prevention.

2. LITERATURE SURVEY

Soft errors in advanced computer systems by
R. C. Baumann

As the dimensions and operating voltages of computer electronics shrink to satisfy consumers' insatiable demand for higher density, greater functionality, and lower power consumption, sensitivity to radiation increases dramatically. In terrestrial applications, the predominant radiation issue is the soft error, whereby a single radiation event causes a data bit stored in a device to

be corrupted until new data is written to that device. This article comprehensively analyzes soft-error sensitivity in modern systems and shows it to be application dependent. The discussion covers ground-level radiation mechanisms that have the most serious impact on circuit operation along with the effect of technology scaling on soft-error rates in memory and logic.

Flipping bits in memory without accessing them: An experimental study of DRAM disturbance errors by Yoongu Kim; Ross Daly; Jeremie Kim; Chris Fallin; Ji Hye Lee Donghyuk Lee

Memory isolation is a key property of a reliable and secure computing system—an access to one memory address should not have unintended side effects on data stored in other addresses. However, as DRAM process technology scales down to smaller dimensions, it becomes more difficult to prevent DRAM cells from electrically interacting with each other. In this paper, we expose the vulnerability of commodity DRAM chips to disturbance errors. By reading from the same address in DRAM, we show that it is possible to corrupt data in nearby addresses. More specifically, activating the same row in DRAM corrupts data in nearby rows. We demonstrate this phenomenon on Intel and AMD systems

using a malicious program that generates many DRAM accesses. We induce errors in most DRAM modules (110 out of 129) from three major DRAM manufacturers. From this we conclude that many deployed systems are likely to be at risk. We identify the root cause of disturbance errors as the repeated toggling of a DRAM row's wordline, which stresses inter-cell coupling effects that accelerate charge leakage from nearby rows. We provide an extensive characterization study of disturbance errors and their behavior using an FPGA-based testing platform. Among our key findings, we show that (i) it takes as few as 139K accesses to induce an error and (ii) up to one in every 1.7K cells is susceptible to errors. After examining various potential ways of addressing the problem, we propose a low-overhead solution to prevent the errors.

Systematic b-adjacent symbol error correcting reed-solomon codes with parallel decoding by Abhishek Das; Nur A. Touba

With technology scaling, the probability of write disturbances affecting neighboring memory cells in nonvolatile memories is increasing. Multilevel cell (MLC) phase change memories (PCM) specifically suffer from such errors which affects multiple adjacent memory cells. Reed Solomon (RS) codes offer good error protection since they

can correct multi-bit symbols at a time. But beyond single symbol error correction, the decoding complexity as well as the decoding latency is very high. This paper proposes a systematic b-adjacent symbol error correcting code based on Reed-Solomon codes with a low latency and low complexity parallel one step decoding scheme. A general code construction methodology is presented which can correct any errors within b-adjacent symbols. The proposed codes are compared to existing adjacent symbol error correcting Reed-Solomon codes, and it is shown that the proposed codes achieve better decoder latency. The proposed codes are also shown to achieve much better redundancy compared to symbol error correcting orthogonal Latin square (OLS) codes.

Siva Sreeramdas, S.Asif Hussain and “Dr.M.N.Giri Prasad proposed on Secure Transmission for Nano-Memories using EG-LDPC”

Memory cells have been protected from soft errors for more than a decade; due to the increase in soft error rate in logic circuits, the encoder and decoder circuitry around the memory blocks have become susceptible to soft errors as well and must also be protected. Here introducing a new approach to design faultsecure encoder and decoder

circuitry for memory designs. The key novel contribution of this paper is identifying and defining a new class of error-correcting codes whose redundancy makes the design of fault-secure detectors (FSD) particularly simple and further quantify the importance of protecting encoder and decoder circuitry against transient errors. By using that Euclidean Geometry Low-Density Parity-Check (EG-LDPC) codes have the fault-secure detector capability. Using some of the smaller EG LDPC codes, can tolerate bit or nanowire defect rates of 10% and fault rates of 10-18 upsets/device/cycle, achieving a FIT rate at or below one for the entire memory system and a memory density of 1011 bit/cm² with nanowire pitch of 10nm for memory blocks of 10 Mb or larger. Larger EG-LDPC codes can achieve even higher reliability and lower area overhead.

Error-correcting codes for semiconductor memory applications: A state-of-the-art review by C. L. Chen and M. Y. Hsiao

This paper presents a state-of-the-art review of error-correcting codes for computer semiconductor memory applications. The construction of four classes of error-correcting codes appropriate for semiconductor memory designs is described, and for each class of codes the number of check bits required for commonly used data

lengths is provided. The implementation aspects of error correction and error detection are also discussed, and certain algorithms useful in extending the error-correcting capability for the correction of soft errors such as α -particle-induced errors are examined in some detail.

FPGA

implementation of FEC Encoder with BCH and LDPC codes for DVB S2 system by D. Digdharsini, D. Mishra, S. Mehta and TVS Ram

This paper gives the design and implementation of Xilinx FPGA based Forward Error Correction (FEC) encoder for DVB S2 system which includes BCH code followed by LDPC code and finally bit mapped to constellation for QPSK modulation. DVB-S2 FEC: ($n=64800, k=32400$) rate 1/2 code, with QPSK modulation scheme is considered as target for FPGA implementation. The architecture in this design efficiently uses pipeline technique along with parallel processing to optimize the hardware resources and overall latency, to accomplish FEC encoding for DVB S2 system. Coding is completed in Verilog HDL with Xilinx Virtex6 XC6VLX240T FPGA as target for hardware realization and QuestaSim

simulator is used to complete the functional simulation.

Flexible unequal error control codes with selectable error detection and correction levels Luis-J. Saiz-Adalid, P. Gil-Vicente, Unequal Error Control (UEC) codes provide means for handling errors where the codeword digits may be exposed to different error rates, like in two-dimensional optical storage media, or VLSI circuits affected by intermittent faults or different noise sources. However, existing UEC codes are quite rigid in their definition. They split codewords in only two areas, applying different (but limited) error correction functions in each area. This paper introduces Flexible UEC (FUEC) codes, which can divide codewords into any required number of areas, establishing for each one the adequate error detection and/or correction levels. At design time, an algorithm automates the code generation process. Among all the codes meeting the requirements, different selection criteria can be applied. The code generated is implemented using simple logic operations, allowing fast encoding and decoding. Reported examples show their feasibility and potentials.

3. EXISTING METHOD

Based on the structure of the parity check matrix, the check bits are calculated by the

corresponding data bits. The new encoded codeword, the combination of check bits and data bits is stored in the memory. When the particles hit the memory resulting in MBUs, the contents of affected memory cells are flipped. Here, to elaborate on the correction ability of QAEC codes, quadruple adjacent bits are flipped on D_2 , D_3 , D_4 , and D_5 . In the decoding process, the syndrome is calculated using the stored check bits and data bits and the structure of the parity check matrix. Through the corresponding relationship between the syndrome and the XOR result of the columns mentioned in Section II, the flipped bits can be located. With the flipped bits inverted, the errors from the storage stage in the memory are effectively corrected. This is the whole procedure of encoding and decoding for the proposed QAEC codes

4. PROPOSED ARCHITECTURE

We propose using two-dimensional error correction code techniques on SRAM for the fast, error-free operation of common errors. The key to 2D error coding is the combination of lightweight horizontal per-word error coding and vertical column error coding. The horizontal and vertical coding can be an error detection code (EDC) or an error correction code (ECC). So we only use vertical codes for correcting errors, and keep

them in the background, so they have a minimal overhead impact in the absence of errors.

In order to demonstrate how 2D error code works, 2D error coding and Hamming code are compared and described below. The first is to compare two protection methods' error covering and memory requirements for 8×8 memory arrays

To enhance memory reliability, a new error correction 2-dimensional code (2D-ECC) is proposed. This algorithm detects and corrects errors effectively when relates with other existing error correction techniques. This performs data region division, redundancy and syndrome calculation, verification and region selection one by one to recover the original data. Boolean XOR operation is performed which is most widely used in cryptography and also in generating parity bits for error checking and fault tolerance. The block diagram of the proposed ECC methodology is shown in fig. 1.

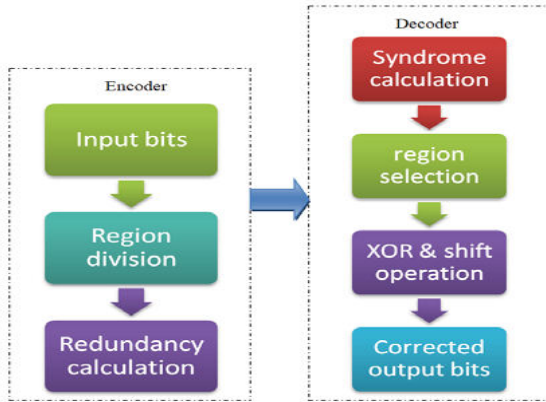


Fig. 1 ECC methodology

This 2-dimensional algorithm performs encoding-decoding process which codifies 16 bit input data into 32 bits in encoding and while decoding again the original 16 bit data is recovered.

Proposed Algorithm

STEP 1: Read the input 16 bit data (A16 – A0)

STEP 2: Divide the input data into 4 groups

X ₁	Y ₁	Z ₁	W ₁
X ₂	Y ₂	Z ₂	W ₂
X ₃	Y ₃	Z ₃	W ₃
X ₄	Y ₄	Z ₄	W ₄

Process of encoding

First, divide the 16 input bits into four groups (Xi, Yi, Zi, Wi). The diagonal bits (Di), parity bits (Pi) and check bits (Ci) are determined using XOR operation. In the process of encoding, the input 16 bits gets converted into 32 bits (redundancy bits).

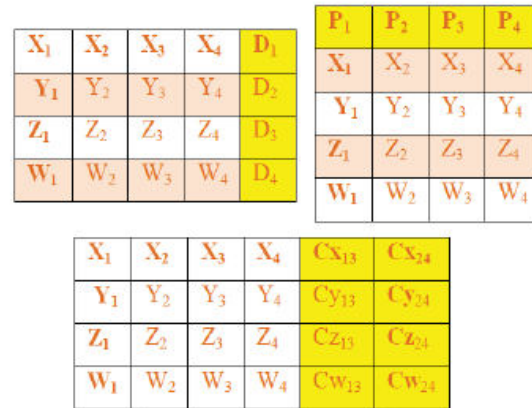


Fig.2 Encoding model

STEP 3: Analyze diagonal bits, parity bits and check bits using XOR operation.

i) Diagonal bits (D1, D2, D3, D4) using XOR operation as the 2x2 matrix,

$$D_1 = X_1 \oplus Y_2 \oplus Z_1 \oplus W_2$$

$$D_2 = X_2 \oplus Y_1 \oplus Z_2 \oplus W_1$$

ii) Parity bits (P1, P2, P3, P4) using XOR operation taking the first bits, second bits, third bits and the fourth bits from four groups

$$P_1 = X_1 \oplus Y_1 \oplus Z_1 \oplus W_1$$

$$P_2 = X_2 \oplus Y_2 \oplus Z_2 \oplus W_2$$

iii) Check bits (Cx, Cy, Cz, Cw) using XOR operation by taking the alternative bits

$$CX_{13} = X_1 \oplus X_3$$

$$CX_{24} = X_2 \oplus X_4$$

$$CY_{13} = Y_1 \oplus Y_3$$

$$CY_{24} = Y_2 \oplus Y_4$$

Process of decoding

In decoding, the syndrome calculation has been analyzed with the encoded data and the recalculated encoded bits (SDi, SPi and

SC_i). After that, verification, region selection and correction can be performed.



Fig.3 Different regions of data bits

STEP 4: Calculate the syndrome values for diagonal, parity and check bits by performing XOR operation between the redundancy data stored and the recalculated redundancy bits

$$SD_i = D_i \oplus RD_i$$

$$SP_i = P_i \oplus RP_i$$

$$SC_i = C_i \oplus RC_i$$

where i = 1, 2, 3, 4

STEP 5: Check the following conditions to identify the error that to be satisfied

- i) SD_i and SP_i bits have atleast one value similar to 1
- ii) More than one SC_i value was similar to 1

STEP 6: Perform region selection and change the erroneous data to get the corrected output.

Divide the data bits into regions 1, 2 and 3. This is formed by dividing the data bits by columns (1&2, 3&4, 2&3) to get efficient results.

5. SIMULATION RESULT

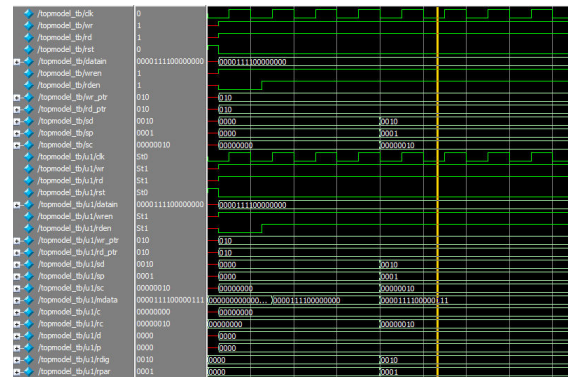


Fig 4 Simulation Result of proposed method

6. CONCLUSION AND FUTURE SCOPE

In this project, a new error correction code (ECC) is proposed to reduce data corruption in volatile memories. The proposed scheme was simulated and synthesized using Xilinx ISE implemented in Verilog HDL. Compared with the well known existing methods, this encoding-decoding process consumes low power and occupies minimum area and delay. Further, this algorithm is to be extended to reduce the area, delay and power consumption. As the regions were selected particularly, the decoder area increases compared to the other existing methods. Further, this is reduced by using the advanced region selection criteria.

REFERENCES

[1] R. Baumann, "Soft errors in advanced computer systems," in IEEE Design & Test

of Computers, vol. 22, no. 3, pp. 258-266, May-Jun 2005.

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

[3] D. Radaelli, H. Puchner, S. Wong and S. Daniel, "Investigation of multi-bit upsets in a 150 nm technology SRAM device," in IEEE Transactions on Nuclear Science, vol. 52, no. 6, pp. 2433-2437, Dec. 2005.

[4] Y. Kim, R. Daly, J. Kim, C. Fallin, J. H. Lee, D. Lee, C. Wilkerson, K. Lai and O. Mutlu, "Flipping bits in memory without accessing them: An experimental study of dram disturbance errors", in Proc. of ACM/IEEE International Symposium on Computer Architecture (ISCA), pp. 361-372, 2014.

[5] A. Das and N.A. Touba, "Systematic b-Adjacent Symbol Error Correcting Reed-Solomon Codes with Parallel Decoding" in Proc. of IEEE VLSI Test Symposium, pp. 1-6, 2018.

[6] A. Das and N.A. Touba, "Low Complexity Burst Error Correcting Codes to Correct MBUs in SRAMs" in Proc. of ACM Great Lakes Symposium on VLSI (GLSVLSI), pp. 219-224, 2018.

[7] H. O. Burton, "Some asymptotically optimal burst-correction codes and their relation to single-error-correcting reed-solomon codes," in IEEE Transactions on Information Theory, vol. 17, no. 1, pp. 92-95, Jan. 1971.

[8] S. Baeg, S. Wen and R. Wong, "SRAM Interleaving Distance Selection with a Soft Error Failure Model," in IEEE Transactions on Nuclear Science, vol. 56, no. 4, pp. 2111-2118, Aug. 2009.

[9] R. Datta and N.A. Touba, "Generating Burst-Error Correcting Codes from Orthogonal Latin Square Codes - A Graph Theoretic Approach," in Proc. of IEEE Symposium on Defect and Fault Tolerance, pp. 367-373, 2011.

[10] P. Reviriego, S. Liu, J.A. Maestro, S. Lee, N.A. Touba and R. Datta, "Implementing Triple Adjacent Error Correction in Double Error Correction Orthogonal Latin Square Codes," in Proc. of IEEE Symposium on Defect and Fault Tolerance, pp. 167-171, 2013.

[11] P. Reviriego, M. Flanagan, S.-F. Liu and J. Maestro, "Multiple cell upset correction in memories using difference set codes," in IEEE Transactions on Circuits and Systems-I: Regular Papers, vol. 59, no. 11, pp. 2592-2599, Nov. 2012.

- [12] P. Reviriego, S. Pontarelli, A. Evans and J. A. Maestro, "A Class of SEC-DED-DAEC Codes Derived from Orthogonal Latin Square Codes", in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 5, pp. 968-972, May 2015.
- [13] J. Kim, N. Hardavellas, K. Mai, B. Falsafi and J. Hoe, "Multi-bit error tolerant caches using two-dimensional error coding," in *Proc. of IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 197–209, 2007.
- [14] C. Argyrides, D. Pradhan and T. Kocak, "Matrix codes for reliable and cost efficient memory chips," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 3, pp. 420-428, Mar. 2011.
- [15] A. Dutta and N.A. Touba "Multiple Bit Upset Tolerant Memory Using a Selective Cycle Avoidance Based SEC-DED-DAEC Code," in *Proc. of IEEE VLSI Test Symposium*, pp. 349-354, 2007.
- [16] S. Shamshiri and K. T. Cheng, "Error-locality-aware linear coding to correct multi-bit upsets in SRAMs," in *Proc. of IEEE International Test Conference (ITC)*, Paper 7.1, 2010.
- [17] A. Neale and M. Sachdev, "A new SEC-DED error correction code subclass for adjacent MBU tolerance in embedded memory," in *IEEE Transactions on Device and Materials Reliability*, vol. 13, no. 1, pp. 223–230, Mar. 2013.
- [18] L. S. Adalid, P. Reviriego, P. Gil, S. Pontarelli and J. A. Maestro, "MCU Tolerance in SRAMs Through Low-Redundancy Triple Adjacent Error Correction," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2332-2336, Oct. 2015.
- [19] J. Li, P. Reviriego, L. Xiao and R. Zhang, "Efficient Implementations of 4-Bit Burst Error Correction for Memories", in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 12, pp. 2037-2041, Dec. 2018.
- [20] C. Wilkerson, A. R. Alameldeen, Z. Chishti, W. Wu, D. Somasekhar and S. Lu, "Reducing cache power with low-cost, multi-bit error-correcting codes," in *Proc. of ACM/IEEE International Symposium on Computer Architecture (ISCA)*, pp. 83–93, 2010.
- [21] K. Namba, S. Pontarelli, M. Ottavi and F. Lombardi, "A Single-Bit and Double-Adjacent Error Correcting Parallel Decoder for Multiple-Bit Error Correcting BCH Codes," in *IEEE Transactions On Device And Materials Reliability*, vol. 14, no. 2, pp. 664-671, Jun. 2014.