SRAM RELIABILITY IMPROVEMENT USING ECC FOR MULTIPLE ADJACENT BIT ERRORS

AYESHA TABUSSUM[1], M. IMTHIAZ BASHA[2]
[1]PG SCHOLAR.DEPT OF ECE, GATES INSTITUTE OF TECHNOLOGY, AP, INDIA
[2] ASST. PROFESSOR, DEPT OF ECE, GATES INSTITUTE OF TECHNOLOGY, AP, INDIA

ABSTRACT: As memory bit cells of an IC get smaller and/or denser, the likelihood of an SEU impacting more than one of such memory bit cells simultaneously increases. However, increasing too is a demand for memory bandwidth, and thus the addition of more parity bits to resolve data corruption issues through use of an ECC would hamper efforts to satisfy such demand for memory bandwidth. As submicron technology scales, SRAM bit cell density increases on the chip. This results in an increase of soft errors due to radiation induced multiple-bits upsets (MBUs). SRAM uses SEC-DED [Single Error Correct – Double Error Detect] code along with word interleaving or column muxing to mitigate these soft errors. The probability of MBUs in the SRAM bitcells in 16nm/7nm technology has increased considerably. Even using word interleaving with 4:1 column mux is not sufficient to mitigate these soft errors. There are powerful ECC codes which can correct these soft errors but it comes at the cost of overhead in the parity bits which eventually translates into the SRAM size increase. So here we are proposing an area efficient code with no extra parity bit cost which can detect and correct adjacent 2-bit error and detect adjacent 3-bit error.

1.INTRODUCTION

As memory bit cells of an IC get smaller and/or denser, the likelihood of an SEU impacting more than one of such memory bit cells simultaneously increases. However, increasing too is a demand for memory bandwidth, and thus the addition of more parity bits to resolve data corruption issues through use of an ECC would hamper efforts to satisfy such demand for memory bandwidth. Accordingly, it would be desirable and useful to provide an ECC that addresses both of these conflicting issues. With the scaling of submicron technology, the size of the bit cell is continuously reducing and so is the bit cell internal nodes capacitance. So the probability of the bit cell storage node getting corrupted by the SEU event increases. Study has shown that 1-5% of the SEU can cause MBUs [1]. SEU can cause several types of MBUs. Analysis of the SRAM bit cell in 7nm FINFET further suggests that the probability of adjacent 3-bit in error has increased with respect to the planar 28nm SRAM bit cell and the 16nm FINFET SRAM bit cell.

All high-speed SRAM use SEC-DED Hamming Code to encode the input data into a code word using the H-Matrix and write into the memory. If any SEU upsets 1 bit of the data stored in the memory, then the ECC decode using the same H-matrix can correct the 1 bit of corrupted data. The syndromes generated using the H-matrix is used to detect which bit location is in error and correct it

Technology scaling has been leading to smaller and smaller device geometries over the years. This has given rise to a host of different problems with both the established memory technologies as well as the newer forms of emerging memory technologies. One such form of error is a burst error which is becoming more and more prevalent in several types of memories due to shrinking feature size. Consider the case of static random-access memory (SRAM). Soft errors caused by radiation poses a significant reliability concern for SRAMs [1]. With technology scaling, the susceptibility of SRAMs to soft-errors has significantly increased as well [2]. In current nano scale technology nodes, device geometries are small, and with technology scaling, devices are getting smaller. Thus, a particle strike might affect more than one cell causing a multiple bit upset (MBU) [3]. The smaller the device geometries, the larger the number of cells that are affected by a single particle strike. A *b*-bit burst error caused by such a particle strike can cause multiple bits to be flipped within the *b*-bit burst window.

Dynamic random-access memory (DRAM) also suffers from a similar problem [4]. The problem arises due to the small physical dimensions brought about by technology scaling. Although it enables to increase the memory capacity of a chip, it also enables the ease with which near-by or adjacent DRAM cells interact with each other. Thus, accessing a memory cell causes a disturbance in the neighboring memory cells causing their charge to leak into the cell or away from it. With enough accesses, it is possible to flip the neighboring cell's currently held value.

Thus, at any given point a memory access can potentially cause a single *b*-bit burst error in the vicinity of the cell being accessed, where *b*is the size of the burst.

The underlying principle of error correcting codes addressing these issues is that the codes should be able to correct a single burst of error. It is possible that not all bits or symbols change within the burst window *b*. Thus, codes aimed towards addressing these issues should be able to correct all possible error combinations within a *b*-bit burst window regardless of the position of burst error.

In this project, a new class of single burst error correcting codes is presented with a parallel decoding scheme. The proposed parallel decoding scheme enables high speed decoding. This is particularly useful for memories whose performance is sensitive to read or access latencies. A new construction methodology is presented which enables the proposed codes to be derived from already existing codes so that a single burst error can be corrected. Preliminary results for the proposed scheme were presented in [5] and [6]. A key feature of the proposed class of codes is that it leads to significant reduction in area of the decoding circuit specifically as the burst size being corrected *b* increases.

2.LITERATURE SURVEY

Geometric effect of multiple-bit soft errors induced by cosmic ray neutrons on DRAM's

Although it has been shown that cosmic ray neutrons play an important role in soft error (SE) phenomena, some important issues remain to be clarified in neutron-induced SE phenomena. This letter reports the geometric effect of multiple-bit SE's induced by neutrons. Multiple-bit SE's in 16 Mb DRAM's are investigated and their geometric effects on high reliability systems are discussed.

Analysis of single-ion multiple-bit upset in high-density DRAMs

Multiple-bit upsets were observed in two types of memories operating in the radiation environment of space. They have been categorized according to their orbital location, amount of shielding and upset multiplicity. The mechanisms responsible have been identified from ground testing of identical memories using both energetic ions and pulsed laser light. With the aid of bit-maps (generated with the pulsed laser) multiple-bit upsets could, in most cases, be attributed to one of three mechanisms, i.e., charge diffusion away from an ion strike, an ion strike to control circuitry, and an ion track intersecting a number of memory cells. Heavy-ion strikes to peripheral circuits on the memory chip generated multiple-bit upsets involving as many as twenty-one cells. Proton-induced multiple-bit upset rates have been calculated for the spacecraft orbit, and the results show good agreement with measured rates

Characterization of multibit soft error events in advanced SRAMs.

Error correction code schemes are being implemented in memories and microprocessor caches in response to SER increases which result from increasing bit counts and technology scaling. These methods can be rendered ineffective by multi-bit error events. An exhaustive characterization of multi-bit errors in 90/130 nm SRAMs is presented to support bit interleaving rules that make the impact of multi-bit errors negligible

Soft errors in advanced computer systems by R. 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.

3.EXISTING METHOD

When bits are transmitted over the computer network, they are subject to get corrupted due to interference and network problems. The corrupted bits leads to spurious data being received by the receiver and are called errors.

Error-correcting codes (ECC) are a sequence of numbers generated by specific algorithms for detecting and removing errors in data that has been transmitted over noisy channels. Error correcting codes ascertain the exact number of bits that has been corrupted and the location of the corrupted bits, within the limitations in algorithm.

4.IMPLEMENTATION OF PROPOSED ARCHITECTURE

In SRAM, the SEU event corrupting the adjacent bit has high probability [7]. We propose a new ECC(72,64), SEC-DEDADEC- ATED code with the optimized syndrome calculation for ECC encode and decode. ECC(72,64) uses the same number of parity bits as the SEC-DED Hamming code H(72,64). ECC(72,64) can correct adjacent 2-bit in error and detect adjacent 3-bit in error while reducing the number of 2 input XOR gates by 40% as compared to SEC-DED code. A new ECC(73,64), SEC-DED-ADEC-ATEC code is proposed to do adjacent 3-bit error correct with the optimized syndrome calculation and reducing the number of 2 input XOR gates by 30% as compared to SEC-DED code. This ECC(73,64) code uses only one extra parity bit compared to the code proposed.

One way to mitigate the SEU is to use word interleaving or column mux in the SRAM array design. Column mux of 2:1, 4:1 is used extensively. SRAM used in processors with no column mux could have multiple adjacent SRAM bits affected by the SEU. The probability of 5 physical adjacent bits corrupted by SEU is less.

ENCODING PROCEDURE

For a systematic code, the encoding procedure is a few XOR operations of the message bits or symbols which is then appended to the original message to form the code-word. If a code is non-systematic, then apart from the original encoding procedure of the base code used to construct the code, a few XOR operations are additionally required to compute the parity check bits or symbols related to the upper portion of the parity check matrix. These parity check bits or symbols can either be append-ed at the end of the codeword or they can be stored separately in a given word. If stored separately, they are here-by called as Separate Parity. The two different forms of parity storage have been shown in Fig. 2.

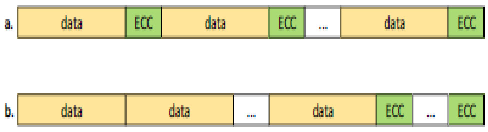


Fig. 1. (a) ECC bits stored alongside data bits. (b) ECC bits stored sepa-rately in memory (Separate Parity).

DECODING PROCEDURE

The general decoding procedure involves the two procedures of computing the error pattern and the error locations. For the case of the proposed scheme, the first step is to compute the syndromes using the parity check matrix. The syndrome is computed by multiplying the parity check matrix with the received codeword. This is a simple XOR operation between all the data bits or symbols which have a corresponding 1 in the parity check matrix on a per row basis. The structure of the parity check matrix of the proposed scheme is such that the error pattern is directly represented by the upper *b* syndrome bits or symbols, where *b* is the size of the burst error being corrected. The rest of the syndrome bits or symbols are then used to compute the location of the multi bit error.

The generation of the ECC code matrix also depends on how the data and the parity bits form a 72/73 bit word. Depending on how the parity is placed w.r.t to the data the ECC(73,64) or ECC(72,64) Code Matrix changes. Some of the examples of data and parity arrangement are shown in Fig. . Parity bits first followed by the data bits or vice versa. 8 bits of data followed by 2 bits parity, 8 bits of data and so on. 16 bits of data followed by 4 bits of parity, 16 bits of data and so on. ECC(72,64) or ECC(73,64) code Matrix will be separate for each of the case . Even with the code matrix change the algorithm to optimize the syndrome calculation holds true. The proposed ECC(72,64) or ECC(73,64) code Matrix algorithm has to satisfy certain conditions. The events 1 bit in error, 2 adjacent bits in error, 3 adjacent bits in error are mutually exclusive. This algorithm is true for all possible combinations of events which are mutually exclusive to each other. ECC(73,64) code matrix algorithm should satisfy the following conditions

1) The syndrome of each 1 bit error should be unique.

2) The syndrome of each 2 adjacent bit error should be unique.

3) The syndrome of each 3 adjacent bit error should be unique.

4) The 1 bit error syndrome can share the common space with the 2 adjacent bit error syndrome, p0 distinguishes between the even or odd number of bits in error.

5) The 3 adjacent bit error syndrome can share the common space with the 2 adjacent bit error syndrome, p0 distinguishes between the even or odd number of bits in error.

6) The 1 bit in error syndrome, 3 adjacent bit error syndrome should not share the common space.

The number of 2 input XOR gates for each syndrome calculation depend on the numbers of 1 in each row shown in the ECC matrix in Fig. 4 and Fig. 5. Optimizing the parity equation into separate minterms is shown in Fig. 8. The proposed algorithm uses the approach where there is overlap of minterms between syndromes. Term t1 is a part of computed syndrome S1. t1 is unique to S1. Term t2 is a part of computed syndrome S2 and so on. t1t2 means its common between S1 and S2. t1t2t3 means these terms are common to S1, S2, S3. Once all these t1, t2, t3, t4, t5, t6, t7 is broken in minterms, then S1 is computed by doing the XOR of all the terms which has t1 in common. All the common minterms are computed separately, and then used in the parity generation equation. It helps in reducing the number of 2 input XOR gates required in the proposed code as well as in the Hamming code. ECC(72,64) encode/decode compared to the Hamming Code H(72,64) and other codes. As the wordsize increases, the method shown in Fig. 8 helps in considerable reduction of 2 input XOR gates and hence the area and static power of ECC.

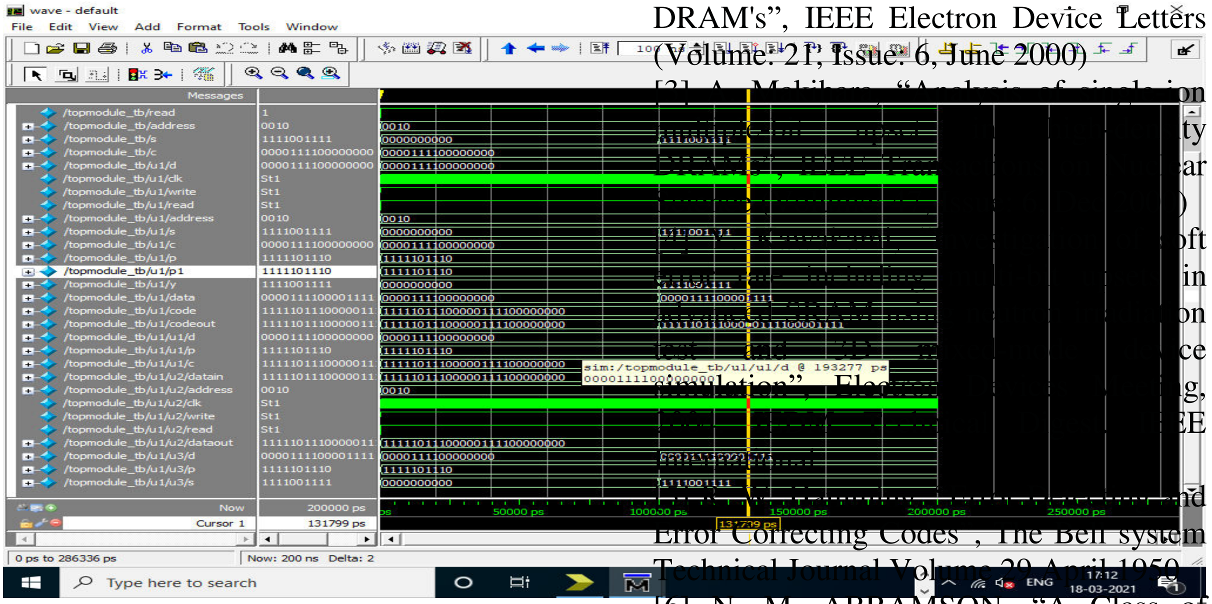
$$\left(\begin{array}{*{39}{c}}
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&1&1&1&1&1&0&1&1&0&1&1&0&1&1&0&1&0&1&0&1&0&0&0&0&1&1&0&0&0&1&1&1&1&0&0&1&1&0\\
0&0&1&1&1&0&1&1&1&1&1&0&1&0&0&1&1&1&1&0&1&0&1&0&0&0&1&0&1&0&0&0&1&1&0&0&0&0&1\\
0&1&1&1&0&1&1&1&0&1&0&1&0&0&1&0&0&0&0&0&1&0&1&0&0&0&0&1&1&1&1&1&1&1&1&1&0&0&0\\
0&1&0&1&1&1&1&0&1&1&0&1&1&0&0&0&1&1&0&1&0&0&1&1&0&0&0&0&0&0&0&0&1&1&1&0&0&1&1\\
0&0&1&0&1&1&1&0&0&0&1&0&1&0&0&0&0&0&1&0&0&0&0&0&1&0&0&1&0&1&0&1&1&1&0&0&1&1&0\\
0&0&0&0&0&0&0&1&1&1&1&1&1&0&0&0&0&0&0&0&0&1&0&1&0&1&1&0&1&1&1&1&1&0&1&0&1&1&1
\end{array}\right)$$

Figure 2. Proposed ECC(39,32) SEC-DED-ADEC-ATED

$$\left(\begin{array}{*{40}{c}}
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&0&0&0&0&0&0&0&1&0&1&1&0&1&0&0&1&1&1&1&0&0&1&1&0&1&1&0&0&0&1&1&1&1&0&0&1&0&1&0\\
0&1&0&0&0&0&0&0&0&1&1&0&1&0&1&1&1&1&0&1&0&1&0&1&1&0&0&0&1&1&1&0&1&0&0&0&1&1&1&0\\
0&0&1&0&0&0&0&0&1&0&0&0&1&1&0&1&1&1&0&0&1&0&1&0&0&0&1&0&0&0&1&0&0&0&0&1&0&1&1&1\\
0&0&0&1&0&0&0&0&0&1&0&0&0&1&0&0&0&1&0&0&0&1&0&1&0&0&0&1&0&0&0&0&0&1&0&0&1&0&0&1\\
0&0&0&0&1&0&0&0&0&0&0&1&0&0&1&1&1&1&0&0&0&0&1&0&0&0&0&0&1&0&0&1&0&0&1&0&0&1&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&0&0&1&0&1&1&1&0&1&1&0&0&1&0&1&1&0&1\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&1&1&0&1&0&0&1&0
\end{array}\right)$$

Figure 3. Proposed ECC(40,32) SEC-DED-ADEC-ATEC

5. SIMULATION RESULT:



```
/topmodule_tb/read            1
/topmodule_tb/address         0010
/topmodule_tb/s               1111001111
/topmodule_tb/c               0000111100000000
/topmodule_tb/u1/d            0000111100000000
/topmodule_tb/u1/clk          St1
/topmodule_tb/u1/write        St1
/topmodule_tb/u1/read         St1
/topmodule_tb/u1/address      0010
/topmodule_tb/u1/s            1111001111
/topmodule_tb/u1/c            0000111100000000
/topmodule_tb/u1/p            1111101110
/topmodule_tb/u1/p1           1111101110
/topmodule_tb/u1/y            1111001111
/topmodule_tb/u1/data         0000111100001111
/topmodule_tb/u1/code         1111101110000011
/topmodule_tb/u1/codeout      1111101110000011
/topmodule_tb/u1/u1/d         0000111100000000
/topmodule_tb/u1/u1/p         1111101110
/topmodule_tb/u1/u1/c         1111101110000011
/topmodule_tb/u1/u2/datain    1111101110000011
/topmodule_tb/u1/u2/address   0010
/topmodule_tb/u1/u2/clk       St1
/topmodule_tb/u1/u2/write     St1
/topmodule_tb/u1/u2/read      St1
/topmodule_tb/u1/u2/dataout   1111101110000011
/topmodule_tb/u1/u3/d         0000111100001111
/topmodule_tb/u1/u3/p         1111101110
/topmodule_tb/u1/u3/s         1111001111

Now       200000 ps
Cursor 1  131799 ps

sim:/topmodule_tb/ul/ul/d @ 193277 ps
0000111100000000

0 ps to 286336 ps     Now: 200 ns  Delta: 2
```

Fig 4 Simulation Result

Fig 8.1show the simulation result of proposed system. Where we can detect the error and correct the error

6.CONCLUSION AND FUTURE SCOPE

The ECC code proposed in this paper can correct multiple adjacent bit in error with minimal area and performance impact on the SRAM. The proposed code is easy to implement and verify. These codes are useful especially for submicron 16nm/7nm FINFET design where the probability of adjacent MBU due to SEU is high. These codes can be used with the word interleaving to mitigate the adjacent MBU. The need of these specially designed codes will increase with the scaling submicron technology. The proposed scalable ECC can be designed for SRAM with any kind of data bit arrangement in the word.

References

[1] J. Maiz, S. Hareland, K. Zhang, P. Armstrong"Characterization of multibit soft error events in advanced SRAMs", Electron Devices Meeting, 2003. IEDM '03 Technical Digest. IEEE International

[2] S. Satoh, Y. Tosaka, S.A. Wender, "Geometric effect of multiple-bit soft errors induced by cosmic ray neutrons on DRAM's", IEEE Electron Device Letters (Volume: 21, Issue: 6, June 2000)

[3] A. Makihara, "Analysis of single-ion multiple-bit upset in high-density DRAMs", IEEE Tran[illegible] on Nuclear Science [illegible] Issue: 6, Dec 2000)

[4] Y. Kawakami, "Investigation of soft error rate including multi-bit upsets in advanced SRAM using neutron irradiation test and 3D mixed-mode device simulation", Electron Devices Meeting, 2004. IEDM Technical Digest. IEEE International

[5] R. W. Hamming, "Error Detecting and Error Correcting Codes", The Bell system Technical Journal Volume 29 April 1950

[6] N. M. ABRAMSON, "A Class of Systematic Codes for Non-Independent Errors", IRE Transactions on Information Theory (Volume: 5, Issue: 4, December 1959)

[7] Eishi Ibe, Hitoshi Taniguchi, Yasuo Yahagi, Ken-ichi Shimbo, Tadanobu Toba,"Impact of Scaling on Neutron-Induced Soft Error in SRAMs from a 250 nm to a 22 nm Design Rule", IEEE

Transactions on Electron Devices (Volume: 57, Issue: 7, July 2010)

[8] A. Klockmann, G. Georgakos, M. Goessel, A new 3-bit burst-error correcting code", On-Line Testing and Robust System Design (IOLTS), 2017 IEEE 23rd International Symposium

[9] Sanghyeon Baeg, Ansan, ShiJie Wen, Richard Wong, "SRAM Interleaving Distance Selection With a Soft Error Failure Model", IEEE Transactions on Nuclear Science (Volume: 56, Issue: 4, Aug. 2009)

[10] Anand Dixit, Alan Wood, "The impact of new technology on soft error rates", Reliability Physics Symposium (IRPS), 2011 IEEE International

[11] Jorge Tonfat, "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 (Volume: 64, Issue:8,Aug.2017)