IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657
Volume2,Issue4,Dec2014

AN EFFICIENT MULTIBIT UPSET DETECTION AND CORRECTION IN 64-BIT SRAM BASED FPGA MEMORY USING DECIMAL MATRIX CODE

Sekhar Reddy

Abstract:
In order to safeguard memory against radiation damage, this work provides a low-cost method of correcting Multiple Bit Upsets (MBUs). Many complicated error correction codes (ECCs) have been employed in the past to safeguard memories against MBUs, but the main problem is greater redundant memory costs.In this study, we suggest using 64-bit Matrix Code to guarantee memory's dependability. In order to catch more mistakes and fix them, the suggested protection code made use of a mechanism to do so. The results demonstrated that the suggested technique offers some security against memory-intensive MBUs of significant size. The reliability of memory in a radiation environment is severely compromised by transient multiple bit upsets (MBUs). The suggested technique uses a 64-bit matrix for memory error correction. Complex error correction codes (ECCs) are frequently employed to safeguard memory against MBUs that might cause data corruption, but the primary issue is that they would demand greater delay overhead. Recently, matrix codes (MCs) based on Hamming codes has been suggested for memory protection. The primary difficulty is that these codes are double error correction codes, and their error correction capabilities are not always improved. In addition, erasure codes are suggested to cut down on the space overhead of additional circuits without affecting the overall encoding and decoding operations in any way. Modern error correction techniques need the use of protection codes to keep memory bits secure. There are techniques of detection and rectification in use. The only real problem with the current MC is that more redundant bits are needed to maintain the same level of memory dependability. In order to guarantee dependability in the face of multiple bit upset, the suggested method made use of a matrix code to cut down on redundancy and rectify more errors than the current approach.

Keywords: FPGA, Multiple Bit Upsets, Reliability,Soft Errors

I. INTRODUCTION

Due to the ionizing effects of atmospheric neutrons, alpha-particles, and cosmic rays, the soft error rate in memory cells is rapidly increasing as CMOS technology scales down to the nano scale and memories are combined with an increasing number of electronic systems. Although single bit upset is a key problem concerning memory reliability, multiple cell upsets (MCUs) have become a critical reliability risk in several memory applications. For many years, error correction codes (ECCs) have been extensively utilized to safeguard memories against soft faults, with the goal of making memory cells as fault- tolerant as feasible. The Bose, Chaudhuri, Hocquenghem, Reed-Solomon, and punctured difference set (PDS) codes are only some of the encoding schemes that have been used to MCUs in storage. However, there is a cost in terms of space, power, and latency when using these codes.

Department of ECE, ANU college of Engineering and Technology, Guntur, India

ISSN 2347–3657
Volume2,Issue4,Dec2014

since the encoding and decoding circuits are more complex in these complicated codes.The general idea for achieving error detection and correction is to add some redundancy (i.e., some extra data) to a message, which receiver can use to check consistency of the delivered message, and to pick up data determined to be corrupt. Error-detection and correction scheme can be either systematic or non- systematic: In a systematic scheme, the transmitter sends the unique data, and attaches a fixed number of check bits (or parity data), which are derived from the data bits by some deterministic algorithm. If only the error detection is required, a receiver can simple apply the same algorithm to the received data bits and compare its output with the receive check bits; if the values do not match, an error has occurred at some point throughout the transmission. Error-correcting codes are regularly used in lower-layer communication,As well as for reliable storage in media such as CDs, DVDs, hard disks and RAM.Static RAM based Field-Programmable Gate Arrays (FPGAs) are most widely used in variety ofapplications mainly due to short time-to-market time, flexibility, high density, and cost-efficiency. SRAM-based FPGA stores logic cells configuration data in the static memory organized as an array of latches. FPGAis used for designing complex digital circuits. Power consumption is also reduced by using SRAM. The power consumption of SRAM varies widely dependingon how frequently it is accessed; it can be as power-hungry as dynamic RAM, when used at highfrequencies, and some ICs can consume many watts atfull bandwidth. On the other hand, static RAM used at a somewhat slower pace, such as in applications with Moderately clocked microprocessors, draws very little power and can have a nearly negligible power idle power usage on the order of microwatts. There are a number of methods offered for controlling the power consumption of SRAM-based memory systems.

The programmable logic and input/output (I/O) blocks of an FPGA device are linked via a programmable routing network. The excellent performance, low development cost, and re-programmability of SRAM-based FPGA devices are increasing their popularity.Denser integration methods and FPGAs built on nanoscale technology. Memories are among the most common components of modern electronic devices. Environmental radiation has a devastating effect on the performance of an electronic circuit. When a charged particle from the environment collides with the silicon of a circuit and causes an error, this is called a single-event upset (SEU). Soft errors are a kind of FPGA device mistake that negatively impacts the mapped design. A soft error will not destroy a system's hardware, the only damage is to the data that is being processed in the memory. Single Error Correction/Double Error Detection (SEC-DED) codes and Built-in Current Sensors (BICS) have lately been employed to safeguard memory from MBUs. However, such techniques could only be used to rectify SEU. A general scrubbing approach is given in this study to recreate the incorrect configuration frame using the idea of Erasure coding algorithm, which may be used for mistake detection and rectification. Using the interleaving distance, which is further categorized into horizontal and vertical parity, MBUs are recognized in this Erasure coding process.

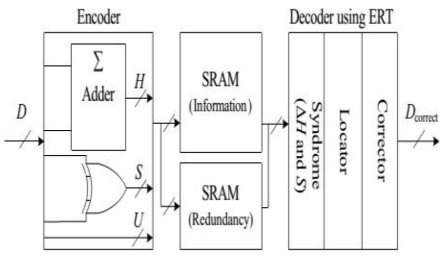
MBU DESIGNS, PART II

Restricting MBUs, which rearrange bits in the physical arrangement to split the bits in the same logical word into various physical words, has been accomplished by the use of an interleaving approach. Nonetheless, the interleaving method not be

ISSN 2347–3657
Volume2,Issue4,Dec2014

practically used in content-addressable memory (CAM), because of the tight coupling of hardware structures from both bits and comparison circuit structures.

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



advantage of using algorithm is that the error detection capability is maximize so that the reliability of memory is enhanced. Besides, the erasure codes is proposed to minimize the area overhead of extra circuits (encoder and decoder) without disturbing the whole encoding and decoding processes.

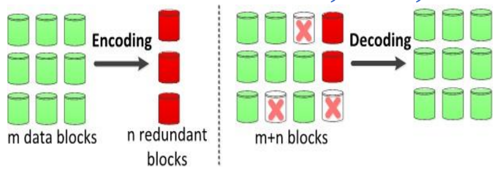


Fig. 1 Encoding and decoding of erasure codes.

I. PROPOSED DMC

In this section, DMC is proposed to assure reliability in the presence of MCUs with reduced performance overheads, and a 64-bit word is encoded and decoded asan example based on the proposed techniques.

A. Proposed Schematic of Fault-Tolerant Memory:

The schematic of fault-tolerant memory is as shown in Fig. 2. The DMC encoder is fed with information bits D, during the encoding (write) process, and then the DMC encoder produces the vertical redundant bits V and horizontal redundant bits H. The obtained DMC code word is stored in the memory, once encoding process is completed. If the memory is affected by MCUs, in the decoding (read) process these errors can be corrected. The proposed DMC has higher fault- tolerant capability with higher performance because of decimal algorithm. The fault-tolerant memory uses ERT technique, to reduce extra circuit's area overhead and will be introduced in the following sections.

Fig. 2 Proposed schematic of fault-tolerant memoryprotected with DMC

The DMC Encoder Proposal B.

After dividing the N-bit word into m-bit symbols (N = k m), the DMC proposes arranging these symbols in a k1 k2 2-dimensional matrix (k = k1 k2, where k1 and k2 values represent thenumber of rows and columns in the logical matrix, respectively). Second, the H bits of horizontal redundancy are produced using decimal integer addition of certain symbols inside each row. In this context, each icon is treated as a separate decimal number. Finally, we get the V redundant bits vertically by performing a binary operation on the bits in each column. It is important to note

ISSN 2347–3657
Volume2,Issue4,Dec2014

that the logical implementation of divide-symbol and arrange-matrix replaces their physical counterparts. As a result, the proposed DMC eliminates the need to alter the hardware of the memory itself.

To illustrate the suggested DMC system, we used a 64-bit word as shown in Fig. 2. The information bits range from cell D0 to cell D63. When a 64-bit word is divided into eight 4-bit symbols, we get eight such symbols. By concurrently setting k1 = 2 and k2 = 4, we get this. H0-H39 represent the horizontal check bits, while V0-V31 represent the vertical check bits. It should be noted, however, that various choices for k and m result in varying numbers of redundant bits and maximum correction capabilities (i.e., maximum size of MCUs that can be repaired). As a result, k and m should be fine-tuned so as to optimize the correction capacity while minimizing the amount of duplicated bits. In this specific circumstance, for instance, when k = 22, The DMC Encoder Proposal B.After dividing the N-bit word into m-bit symbols (N = k m), the DMC proposes arranging these symbols in a k1 k2 2-dimensional matrix (k = k1 k2, where k1 and k2 values represent thenumber of rows and columns in the logical matrix, respectively). Second, the H bits of horizontal redundancy are produced using decimal integer addition of certain symbols inside each row. In this context, each icon is treated as a separate decimal number. Finally, we get the V redundant bits vertically by performing a binary operation on the bits in each column. It is important to note that the logical implementation of divide-symbol and arrange-matrix replaces their physical counterparts. As a result, the proposed DMC eliminates the need to alter the hardware of the memory itself.To illustrate the suggested DMC system, we used a 64-bit word as shown in Fig. 2. The information bits range from cell D0 to cell D63.

When a 64-bit word is divided into eight 4-bit symbols, we get eight such symbols. By concurrently setting k1 = 2 and k2 = 4, we get this. H0-H39 represent the horizontal check bits, while V0-V31 represent the vertical check bits. It should be noted, however, that various choices for k and m result in varying numbers of redundant bits and maximum correction capabilities (i.e., maximum size of MCUs that can be repaired). As a result, k and m should be fine-tuned so as to optimize the correction capacity while minimizing the amount of duplicated bits. In this specific circumstance, for instance, when k = 22, and m = 8, only 1-bit error can be corrected and the number of redundant bits is 80. When k = 4 × 4 and m
= 2, 3-bit errors can be corrected and the number of redundant bits is reduced to 32. However, when k = 2 ×4 and m = 4, the maximum correction capability is upto 5 bits and the number of redundant bits is 72. In this paper, in order to enhance the reliability of memory, the error correction capability is first considered, so k = 2 ×8 and m = 4 are utilized to construct DMC.

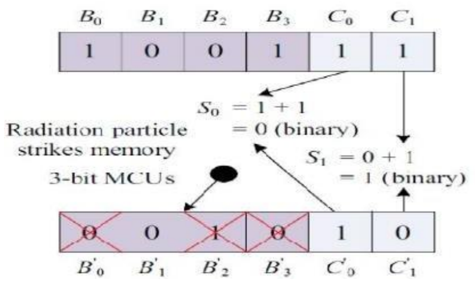
The horizontal redundant bits H can be obtained by decimal integer addition as follows

H4H3H2H1H0 = D3D2D1D0 + D19D18D17D16 (1)H9H8H7H6H5 = D7D6D5D4 + D23D22D21D20 (2)
and similarly for the horizontal redundant bits H14H13H12H11H10,
H19H18H17H16H15
H16,H24H23H22H21H20,
H29H28H27H26H25,
H34H33H32H31H30 and H39H38H37H36H35 where
"+" represents decimal integer

addition. For the vertical redundant

B. Proposed DMC Decoder

To obtain a word being corrected, the decoding process is required. For example, first, the received



B_0 B_1 B_2 B_3 C_0 C_1
1 0 0 1 1 1
$S_0 = 1 + 1$
$= 0$ (binary)
Radiation particle strikes memory
3-bit MCUs
$S_1 = 0 + 1$
$= 1$ (binary)
0 0 1 0 1 0
B'_0 B'_1 B'_2 B'_3 C'_0 C'_1

redundant bits H4H3H2H1H0' and V0'-V3' are generated by the received information bits D'. Second, the horizontal syndrome bits ΔH4H3H2H1H0 and the vertical syndrome bits S3 − S0 can be calculated as follows:

ΔH4H3H2H1H0 = H4H3H2H1H0' − H4H3H2H1H0(5)

information bits D' and compared to the original set of redundant bits in order to obtain the syndrome bits

ΔH and S. Then error locator uses ΔH and S to detect and locate which bits some errors occur in. Finally, inthe error corrector, these errors can be corrected by inverting the values of error bits.

$$S0 = V0'^{\wedge}V0 \qquad (6)$$

and similarly for the rest vertical syndrome bits, where "–" represents decimal integer subtraction. When ΔH4H3H2H1H0 and S3 − S0 are equal to zero, the stored code word has original information bits in symbol 0 where no errors occur. When ΔH4H3H2H1H0 and S3 − S0 are nonzero, theinduced 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

$$D0correct = D0 \wedge S0 \qquad (7)$$

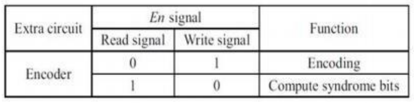
The proposed DMC decoder is depicted in Fig, which is made up of the following sub modules, and each executes a specific task in the decoding process: syndrome calculator, error locator, and error corrector. It can be observed from this figure that the redundant bits must be recomputed from the received
Fig 5: 64-bit DMC decoder structure using ERT

Fig 6: limits of binary error detection in simple binary operations

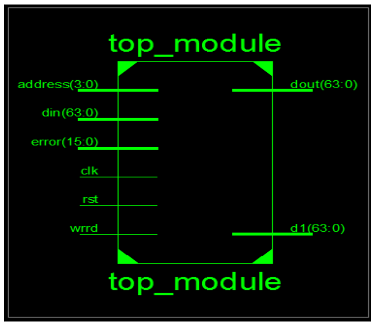
In the proposed scheme, the circuit area of DMC is minimized by reusing its encoder. This is called the ERT. The ERT can reduce the area overhead of DMC without disturbing the whole encoding and decoding processes. From Fig, it can be observed that the DMC encoder is also reused for obtaining the syndrome bits in DMC decoder. Therefore, the whole circuit area of DMC can be minimized as a result of using the existent circuits of encoder. Besides, this figure also shows the proposed decoder with an enable signal En for deciding whether the encoder needs to be a part of the decoder. In other words, the En signalis used for distinguishing the encoder from the decoder, and it is under the control of the write and read signals in memory. Therefore, in the encoding (write) process, the DMC encoder is only an encoder to execute the encoding operations. However, in the decoding (read) process, this encoder is employed for computing the syndrome bits in the decoder. These clearly show how the area overhead of extra circuits can be substantially reduced.

II. RESULTS

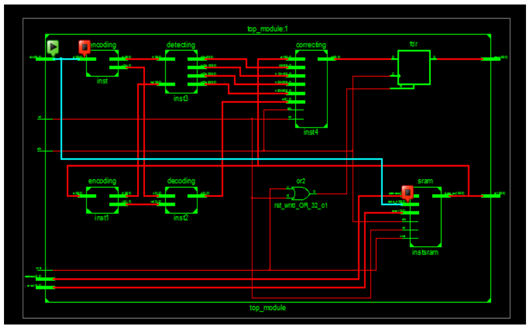
Block diagram



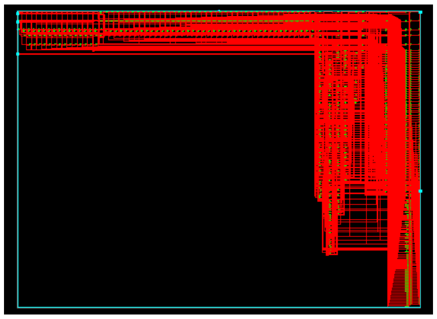
Extra circuit	*En* signal		Function
	Read signal	Write signal	
Encoder	0	1	Encoding
	1	0	Compute syndrome bits

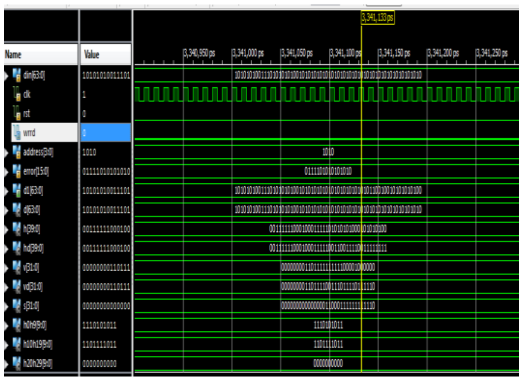


RTL Schematic



Technology Schematic





Simulation Results

CONCLUSION

DMC was introduced in this study as a means of guaranteeing memory's dependability. In order to improve mistake detection and correction, the suggested protection code made use of a decimal method. Based on the findings, the suggested technique was shown to provide a higher degree of security against memory-intensive, big MCUs. More redundant bits are needed to maintain higher reliability of memory with the proposed DMC, so it's important to pick a value for k and m that strikes a balance between the two goals, based on radiation experiments. As a result, more research will be done to lessen the number of unnecessary bits while keeping the suggested method's dependability intact.

REFERENCES

[1] Jing Guo, Liyi Xiao, Member, IEEE, Zhigang Mao, Member, IEEE, and QiangZhao,”Enhanced memory reliability against multiple cell upsets using Decimal Matrix Code” IEEE Trans. Very Large ScaleIntegr. (VLSI) Syst., vol. 22, no. 1, pp.127-135, Mar 2013.

[2] D. Radaelli, H. Puchner, S. Wong, and S. Daniel, “Investigation of multi-bit upsets in a 150 nm technology SRAM device,” IEEE Trans.Nucl. Sci., vol. 52, no. 6, pp. 2433–2437, Dec. 2005.

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

[4] 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.

[5] 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.

[6] S. Liu, P. Reviriego, and J. A. Maestro, “Efficient majority logic fault detection with difference-set codes for memory applications,” IEEETrans. VeryLarge Scale Integr. (VLSI) Syst., vol. 20, no. 1, pp. 148–156, Jan. 2012.

[7] M. Zhu, L. Y. Xiao, L. L. Song, Y. J. Zhang, and H. W. Luo, “New mix codes for multiple bit upsets mitigation in fault-secure memories,” Micro electron.J., vol. 42, no. 3, pp. 553–561, Mar. 2011.

[8] R. Naseer and J. Draper, “Parallel double error correcting code design to mitigate multi-bit upsets in