IJRRECS/April 2016/Volume-4/Issue-4/5409-5413 **ISSN 2321-5461**

INTERNATIONAL JOURNAL OF REVIEWS ON RECENT ELECTRONICS AND COMPUTER SCIENCE

DISCOVERING OF FAULTS FOR MANAGING APPLICATIONS OF MEMORY

Manchala Vinaykumar[1], P.Osman[2]

[1]M.Tech Student, Dept of ECE, Nimra Institute of Engineering &Technology, Ongole, A.P, India

[2]Associate Professor, Dept of ECE, Nimra Institute of Engineering &Technology, Ongole, A.P, India

ABSTRACT:

Error Correction Codes have become the best means to moderate soft errors within memory. Due to result of increasing densities of integration, there is an increase in soft errors which point prerequisite for abilities of higher error correction. A new version of Majority Logic decoder for improvisation of performance is presented in our work. We focus on designing of Majority Logic Detector circuitry itself as the module of error detection all along with the correction of fault in support of memory applications by means of decreasing the time of fault detection. As much as memory applications are concerned soft errors as well as single event upsets are constantly a matter of problem. The proposed system keeps area overhead smallest and power consumption low for big code word sizes. Proposed decoder was implemented by means of Euclidean geometry low-density parity check codes.

Keywords: Error Correction Codes, Majority Logic decoder, Euclidean geometry low-density parity check codes, Fault detection, Memory, Error detection.

1. INTRODUCTION:

Memories are the majority common component these days and for several years, memory cells were protected from soft errors. In the recent times rates of memory failure are increasing because of impact of scaling-smaller dimensions and so on [1]. The capability to speedily find out that a bit has flipped is important to high consistency

as well as high availability applications. Some of the generally used methods are Triple Modular Redundancy as well as Error Correction Codes. The common Error Correction Codes are Single Error Correction codes that correct one bit error within a memory word. Because of result of increasing densities of integration, there is an increase in soft errors which point prerequisite for abilities of higher error correction. Improved Error Correction codes were projected for memory applications but yet Double Error Correction codes with parallel functioning incur in important power consumption consequence. Cyclic block codes contain property of being majority logic decidable hence cyclic block codes were identified as more appropriate among Error Correction codes that meet needs of high error correction ability as well as low decoding difficulty. An additional option is to first distinguish when there are errors in word and only carry out rest of decoding process when there are errors. This reduces average power expenditure since most words will contain no errors [2][3]. We can apply a fault detector for Error Correction codes are but this moreover add an added complex functional unit. Our work spotlight on usage of Majority Logic Detector circuitry itself as the module of error detection hence with no extra hardware read operations might be accelerated.

2. METHODOLOGY:

A new version of Majority Logic decoder for improvisation of performance is presented in our work. Regarding the original Majority Logic decoder, proposed decoder was implemented by means of Euclidean geometry low-density parity check codes. Although majority of decodable codes can correct great number of errors, they require high decoding time for finding errors and Majority Logic Decoding process might consider similar fault detecting time for invalid as well as error free code words, which in turn delays the performance of memory performance. Euclidean geometry low-density parity check codes are on basis of Euclidean geometries over Galois field. Among Euclidean geometry low-density parity check codes there is a subclass of codes that is single step majority logic decodable. The error detection as well as correction method is done by single step majority logic decoding and is effective for Euclidean Geometry Low Density Parity Check Codes. The memory system of proposed scheme is

shown in the fig. The proof of hypothesis that all the errors are detectable in three cycles is extremely difficult from mathematical viewpoint. It is practical to make and check the entire achievable error combinations for codes by small words and affected by little number of bit flips. The simulations are performed in two ways, error combinations are carefully checked when it is practicable and in rest of cases combinations are checked at random. As it is suitable to first describe select design and moreover for simplicity, we assume that hypothesis is true, that only three cycles are essential to notice the entire errors affecting up to four bits in Euclidean geometry low density parity check codes. Our work focuses on designing of Majority Logic Decoder for detection of fault all along with correction of fault, appropriate for memory applications, with decreased fault detection time. For error detection as well as correction, when comparing to traditional system, a speed up is obtained when there is no error within data read access [4]. It is because fault recognition needs simply three cycles and after detection of error free condition, codeword is passed to output devoid of further corrections. This is huge saving of time as most of situations memory read access does not build errors thus there is a significant reduction within memory access time. The proposed system is considered as it is independent of code word size and inference about area is that for huge values of code word size, area transparency of proposed system actual decreases regarding plain technique. Hence the proposed system will be a resourceful design in support of fault detection as well as correction.

3. OVERVIEW OF MAJORITY LOGIC DETECTOR:

In our proposed work we spotlight on usage of Majority Logic Detector circuitry as the module of error detection, thus no additional hardware read operations might be accelerated. Generally proposed version makes use of same decoding algorithm which is used in plain Majority Logic decoder version. The advantage is that, proposed technique stops intermediately in third cycle when there is no error within data read, rather than decoding it for complete codeword size. The xor matrix is evaluated for initial three cycles of decoding process, and when the entire outputs is zero, codeword is determined to be error-free and forwarded directly to output. The proposed technique would carry on whole decoding

procedure to remove the errors when the entire outputs are 1 in any of three cycles. In the proposed system, there is an additional circuitry of control logic which includes three shift registers as well as a xor gate. Thus there will be a slight area transparency when compared to traditional system due to this detection logic. For error detection as well as correction, when comparing to traditional system, a speed up is obtained when there is no error within data read access. The design for 15 bit codeword shows basic ML decoder with 15-tap shift register, XOR array to compute orthogonal parity check sums and majority logic circuit which decide whether current bit in decoding is incorrect and requirement for its inversion. The plain Majority Logic also has the same schematic arrangement up to this stage. The extra hardware proposed for fault detection is control logic unit and output tri-state buffers. The control unit set off a finish flag when there is no fault is distinguished in data read [5]. Output tristate buffers are constantly in high impedance state until control unit sends finish signal in order that current values are forwarded towards output from shift register. The detection procedure is managed by control unit. For distinguishing the initial three iterations of Majority Logic decoding, counter is used that counts up to three cycles. The output value is fed towards two shift registers which contains results of earlier stages stored in it. The values are moved. When during memory read access an error is noticed, xor gate will correct it, by means of inverting current bit under decoding which clearly provides performance enhancement regarding traditional method which existing Majority Logic decoding. The proposed technique mainly take three cycles for decoding as the most of words will be error free and would require to complete decoding procedure only for those words with errors [6].

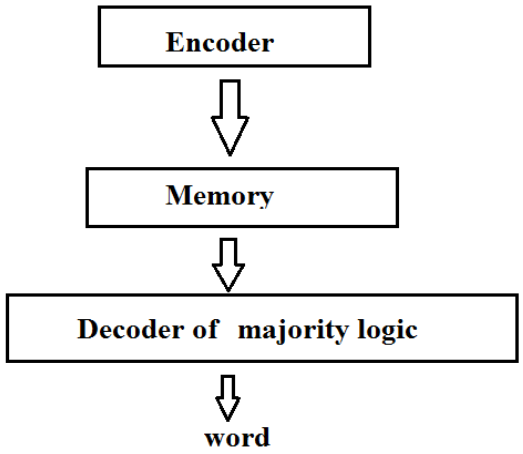

Fig1: An overview of memory system with proposed scheme.

4. CONCLUSION

Our work limelight on usage of majority logic detector circuitry itself as the module of error detection hence with no extra

hardware read operations might be accelerated. Although most of codes can correct great number of errors, they require high decoding time for finding errors. Majority Logic Decoding process might consider similar fault detecting time for invalid as well as error-free code words, which in turn delays the performance of memory performance. The projected fault-detection system can distinguish fault in less decoding cycles. When data read is error free, it noticeably reduces memory access time. Proposed decoder was implemented by means of Euclidean geometry low-density parity check codes. The projected system will be a resourceful design in support of fault detection as well as correction. Error detection as well as correction method is done by single step majority logic decoding and is effective for Euclidean Geometry Low Density Parity Check Codes. The proposed technique mainly take three cycles for decoding as the most of words will be error free and would require to complete decoding procedure only for those words with errors. The proposed method keeps area overhead smallest and power consumption low for big code word sizes.

REFERENCES

[1] C. W. Slayman, "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations," IEEE Trans. Device Mater. Reliabil., vol. 5, no. 3, pp. 397–404, Sep. 2005.

[2] H. Naeimi and A. DeHon, "Fault secure encoder and decoder for NanoMemory applications," IEEE Trans. Very Large Scale Integr.(VLSI) Syst., vol. 17, no. 4, pp. 473–486, Apr. 2009.

[3] S. Ghosh and P. D. Lincoln, "Low-density parity check codes for error correction in nanoscale memory," SRI Comput. Sci. Lab. Tech. Rep. CSL-0703, 2007.

[4] Pedro Reviriego,Juan A. Maestro, and Mark F. Flanagan, "Error Detection in Majority Logic Decoding of Euclidean Geometry Low Density Parity Check (EG-LDPC) Codes," IEEE Transactions On Very Large Scale Integration (Vlsi) Systems 1 (will be published).

[5] C.Tjhai, M. Tomlinson, M. Ambroze, and M. Ahmed, "Cyclotomic idempotent- based binary cyclic codes," Electron. Lett., vol. 41, no. 6, Mar. 2005.

[6] T Shibuya and K. Sakaniwa, "Construction of cyclic codes suitable for iterative decoding via generating idempotents," IEICE Trans. Fundamentals, vol. E86-A, no. 4, pp. 928–939, 2003.