Power efficient technique for the removal of Soft Errors in Memories

Bautista | Pompili
Department of Electronics and Communication Engineering, University of waterloo, waterloo, Canada

Abstract— *Soft errors play a serious role in memory blocks. as a result of the high proportion of transient error rate in logic circuits, the encoder and decoder electronic equipment round the memory blocks area unit additional vulnerable to soft errors and therefore should be protected. thus to safeguard encoder and decoder electronic equipment against transient errors, error-detection methodology for difference-set cyclic codes with majority logic decryption was introduced. it's an acceptable technique to sight error however has giant decryption time and enormous access time. the same category of geometry tenuity check (EG-LDPC) codes that area unit one step Majority Logic decryption methodology is employed to scale back the decryption time. during this paper, Majority Logic Detector/Decoder is employed to any scale back the decryption time. Simulations area unit dole out and therefore the results area unit compared with the one step Majority Logic Decoder. Majority Logic Detector/Decoder is found to own less decryption time compared to Majority Logic Decoder.*

Keywords— Error correction codes (ECC), Euclidean Geometry Low Density Parity Check (EG-LDPC) codes, Majority Logic Decoding (MLD) and Difference-Set Low Density Parity Check (DS-LDPC) codes.

I. INTRODUCTION

In semiconductor devices, as a result of the impact of technology scaling, we've got smaller dimension, higher integration densities and low in operation voltage etc. [1], [2] to satisfy client wants, however their sensitivity to radiation will increase dramatically. Soft errors amendment the logical values of memory cells while not damaging the circuits. Soft errors also are referred to as as SINGLE EVENT UPSET (SEU) [3]. Memory could be a basic resource in each digital system. these days some sorts of soft error happens once bit that's flipped is in crucial system management register like found in FPGA or DRAM in order that errors cause product to malfunction. whereas retrieving info from memory, it ought to be uncorrupted as if that is encoded. thus it's vital to safeguard memory against error [4], [5]. Some normally used error identification techniques area unit

1. Triple standard Redundancy (TMR) and
2. Error Correction Codes (ECC).

Triple standard Redundancy, generally referred to as Triple-Mode Redundancy (TMR) could be a fault tolerant type of N-Modular Redundancy, therefore the complexness overhead would be 3 times and complexness of the bulk citizen and so increasing the facility consumption. For recollections, it clad that error correction code codes area unit best thanks to mitigate memory soft errors [1].

Error Correction Codes (ECCs) protects encoder and decoder against unobserved knowledge corruption, and is employed in recollections. error correction code memory maintains a memory system proof against single-bit errors: {the knowledge|the info|the information} that's scan from every word is usually constant because the data that had been written thereto [1]. Single Error Correction (SEC) codes which will correct one error in memory word area unit normally used.

Cyclic codes area unit linear block error-correcting codes that have convenient pure mathematics structures for economical error detection and correction. thus cyclic codes area unit additional appropriate among error correction code codes that meet the wants of high error correction capability and low decryption complexness owing to the bulk decodable logic. The decryption strategies include: one step majority-logic (MLG) decryption [8], Gallager's bit flipping (BF) decryption [7], weighted MLG decryption, weighted BF decryption, aposteriori chance (APP) [19] decryption and repetitious decryption supported belief propagation or sum-product formula (SPA) [10], [11]. geometry (EG) LDPC is constructed victimisation special structures of finite geometry.

numerous sorts of EG-LDPC codes have the various properties e.g., type-I codes area unit systematic, type-II codes have coding and check matrices with regular standard structure, Galleger codes have properties that alter their error correction rate to be modified dynamically [8]. geometry tenuity check (EG-LDPC) codes, a sub-group of the tenuity check (LDPC) codes, that belongs to the family of cubic centimeter decryption is taken into account owing to its low complexness and easy implementation.

A code is claimed to be cyclic provided that the rows of its parity-check matrix and generator matrix area unit the cyclic shifts of their initial rows. conjointly a code is cyclic code if for any codeword c, all the cyclic shifts of the codeword area unit still valid codewords. The minimum variety of code bits that area unit completely different in associatey 2 codewords is claimed to be minimum distance of an error correction code, d. the utmost variety of errors that associate error correction code will sight is d – one. the utmost variety of error that associate error correction code will correct is (d-1)/2. associate error correction code will be diagrammatical with a facilitate of code length n, info bit length k, minimum distance d and it's denoted as (n, k, d).

II. MAJORITY LOGIC DECODING TECHNIQUE

Majority logic decryption is that the economical error-correcting technique with low complexness [12]. MLD is predicated on the quantity of check equations that area unit orthogonal to every alternative, so that, every codeword gift in barely one check equation in every iteration, except the terribly initial bit which can gift all told equation.

Therefore call is taken by majority results of these check equations. the overall steps of memory schematic with MLD is that word is initial encoded and is written to the memory [13]. when memory reads the word from memory, it's passed to a majority logic detector block that detects and corrects the errors that occurred whereas reading the codeword. during this sort of decryption the information word is corrected from all bit-flips that it would have affected whereas being keep within the memory. this kind of decoder will be enforced in 2 ways that. the primary one is termed the TYPE-1 cubic centimeter Decoder, that determines the bit have to be compelled to be corrected from the XOR mixtures of the syndrome [8], the TYPE-II cubic centimeter Decoder that calculates the data of correctness of this bit underneath decryption, directly out of the codeword bits. each area unit quite similar, however once implementation is taken into account the TYPE-II uses less space, since it doesn't have syndrome calculation as associate intermediate step.

A. Existing ML Decoder

The present cubic centimeter decoder has the economical error management capability to sight and proper the errors that occur within the codeword whereas reading from the memory. In general, cubic centimeter decoder could be a easy and powerful decoder, capable of correcting multiple random bit-flips counting on the quantity of check equations. It consists of 4 major parts: (1) a cyclic shift register; (2) associate XOR matrix; (3) a majority gate; (4) associate XOR gate, for correcting the codeword bit underneath decryption as illustrated in Fig.1. The inputs signal x is ab initio keep into the cyclic shifter register and shifted through all the faucets. The intermediate values in every faucet area unit then accustomed calculate the result of the check add equation from the XOR matrix. within the ordinal cycle, the results have earned the ultimate register and create the output bits. The codeword area unit so littered with the soft error which ends within the wrong codeword. The codeword is passed to the shift register; the decryption method begins by conniving the check equation from the XOR matrix.

Thus the ensuing values area unit forwarded to the bulk gate module to outline whether or not the codeword is inaccurate or not. If {the variety|the amount|the quantity} of 0's is a smaller amount than number of 1's, then this little bit of the codeword is error. to beat this drawback, an indication to correct it'd be triggered.

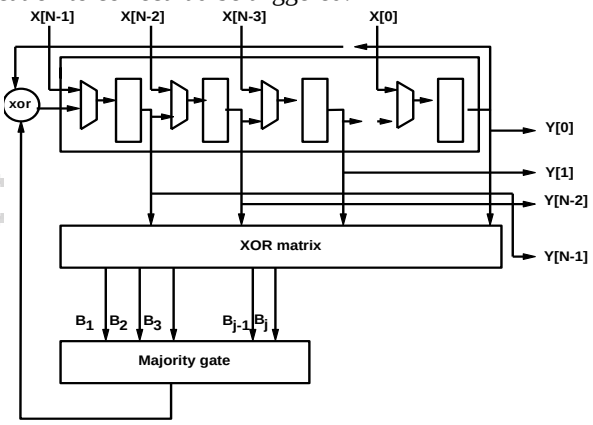

Fig.1. Schematic of ML Decoding method.

Otherwise, the bit underneath decryption would be correct and no additional operations would be required thereon. within the next step, the contents of the register area unit revolved and therefore the on top of procedure is recurrent till all codeword bits are processed. If the codeword is properly decoded then check sums ought to be zero. If there area unit N bits, the cubic centimeter decoder method takes N iteration. This formula wants N iteration for N bit codeword input. thus the most disadvantage of this method is especially supported the codeword size. it's overcome by the planned MLDD technique. Here if the codeword is error-free, then output are going to be processed within the 3 iteration, regardless of the codeword size.

III. DESIGN STRUCTURE OF PROPOSED MAJORITY LOGIC DECODER / DETECTOR

During this section, we offer the look structure of the encoder, corrector and detector units of the planned Majority Logic Decoder/Detector (MLDD). The planned MLDD has been enforced with the assistance of EG-LDPC [12]. The EG-LDPC area unit supported the structure of geometry. There area unit many categories of EG-LDPC codes out there, among that one utilized in this paper is one step majority logic decodable (MLD) [13]. The schematic of a memory system with MLDD is illustrated in Fig.2.

This methodology is most effective to come up with and check all attainable error mixtures for codes with tiny words and littered with alittle variety of bit flips. it's troublesome to check all attainable mixtures if the dimensions of the code will increase. to beat these drawbacks, the simulations area unit performed in 2 cases, in initial case the error mixtures area unit checked once it's possible and in next case the mixtures area unit checked haphazardly.

A. Encoder

THE ENCODER STRUCTURE CALCULATES THE PARITY OPERATE OF EVERY BIT SUPPORTED THE DATA BITS. THE ENCODER ELECTRONIC EQUIPMENT CONSISTS OF (N-K) XOR GATES WHEREVER EVERY PARITY OPERATE COULD BE A XOR GATE. ASSOCIATE N-BIT CODEWORD C, THAT ENCODES A K-BIT INFO VECTOR I IS GENERATED BY MULTIPLYING THE K-BIT INFO MEMORY. THE ENCODER VECTOR CONSISTS OF KNOWLEDGE BIT FOLLOWED BY PARITY BITS, WHEREVER EVERY BIT EASY ASSOCIATE REAL NUMBER OF KNOWLEDGE VECTOR AND COLUMN OF X, FROM G=[1:X] . THE GENERATOR MATRIX FOR (15, 7, 5) EG-LDPC CODE IS SHOWN IN FIG.3. THE ENCODER CIRCUIT IS MADE BY REPETITION THE DATA VECTOR (I0, I1…I6) TO ENCODER VECTOR (C0, C1…C6). THE REMAINDER OF ENCODED VECTOR (C7, C8…C14) IS THAT THE PARITY OPERATE EVERY BIT FASHIONED BY LINEAR SUMS OR XOR OF THE DATA BITS.

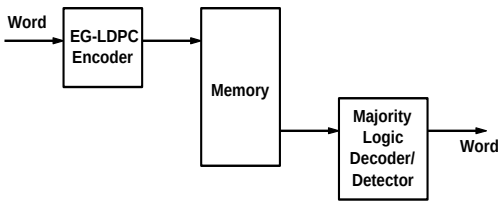

Fig.2. Schematic of a memory system with MLDD method.

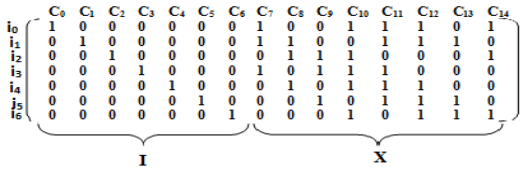

Fig.3. The generator matrix for (15, 7, 5) EG-LDPC code.

The encoder circuit to reason the parity bits of the (15, 7, 5) EG-LDPC code is shown in Fig.4. If the building block of XOR gate is two-input gate then the encoder electronic equipment consists of twenty-two two-input XOR gates. Table I shows the realm of the encoder circuits for every EG-LDPC codes into consideration supported their generator matrices.

To write the data vector, the data bits area unit fed into the encoder and it'll conjointly checks whether or not the encoded vector contains error or error-free with the assistance of detector.

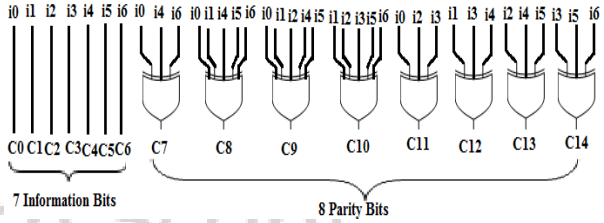

Fig.4. Structure of an encoder circuit for the (15, 7, 5) EG-LDPC code

If the detector detects any error, once more the coding operation should be redone to come up with the right codeword. The codeword is then keep within the memory, particularly location. throughout the access operation, the keep codewords are going to be accessed from the memory unit. whereas the codewords area unit keep within the memory, there's some risk of codewords liable to some transient faults. it's generally referred to as as soft errors.

TABLE I
ENCODER CIRCUIT AREA IN NUMBER OF 2-INPUT GATES

Codes	Number of two input gates
(15,7,5)	22
(63,37,9)	355
(255,175,17)	6577

B. Majority Logic Decoder and Detector Structure

MLDD is associate economical and powerful decoder, capable of correcting many random bit-flips that depends on variety of the check equation. MLDD methodology overcomes the disadvantages of MLD. In MLDD methodology, the N-bit codeword input is encoded and

decoded. If the codeword doesn't contain any error, then output are going to be processed in initial 3 iteration, otherwise it'll take N iterations.

Schematic of planned MLDD methodology for 15-bit codeword is shown in Fig.5. It consists of 4 modules as follows (1) cyclic shift register; (2) XOR-matrix; (3) majority gate and (4) management unit. A block of 'k' bit info vector area unit encoded to make a block of 'n' bit encoder vector together with bit is termed codeword. The codeword accessed from the memory is fed into the cyclic register. The output codeword of the foremost vital Bit (MSB) of the cyclic register is given as associate input codeword to Least vital little bit of the cyclic register.

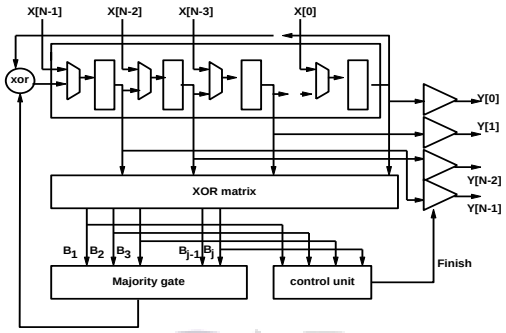

Fig.5. Schematic of the proposed MLDD for 15-bit codeword.

In the schematic for the planned MLDD for 15-bit codeword is illustrated in Fig.6, the codeword of fifteen bit is accessed from the memory and it's loaded into the cyclic register. it's shifted through all the faucets within the cyclic register. The results of the check add equations from the XOR matrix area unit calculated from the intermediate values in every faucet within the cyclic register. when the cyclic shifts, all the codeword bits of the register stay within the register however their bit position changes with none loss of knowledge. within the Nth-cycle, the result has reached the ultimate faucet, manufacturing the output. currently there's some probability of codeword liable to transient or soft errors, which might lead associate {input knowledge|input file|computer file} x as a wrong data. to beat this disadvantage, when the initial step wherever the codeword is loaded into the cyclic register, decryption method continues by conniving the check equations hardwired within the XOR matrix.

Now the ensuing sums area unit then forwarded to the bulk gate. This majority gate is employed to examine the validity of the codeword. If the quantity of 0's is a smaller amount than the quantity of 1's then this bit underneath decryption is wrong owing to soft errors. thus have to be compelled to trigger signal to correct it. If {the variety|the amount|the quantity} of 0's is bigger than number of 1's then this bit underneath decryption is error-free and no additional operations would be required thereon.

In the next step, the content of the registers area unit shifted and revolved. once more the on top of mentioned decryption method is recurrent. This procedure are going to be continued till all codeword bits are processed. during this step, if decryption method proceed with MLD decryption methodology mean it'll take N-cycles (here N=15). rather than decryption all codeword bits by MLD decryption methodology, the planned methodology intermediately stops in third cycle if the codeword is error-free. Finally the check sums should be zero if N-codeword has been properly decoded. Otherwise check sums should be one, if N-codeword has not decoded properly.

The additional hardware to perform the error detection is (1) the management unit; (2) the output tristate buffers. The output tristate buffers area unit forever in high ohmic resistance unless the management unit sends the end signal in order that this values of the register area unit forwarded to the output. The management unit within the schematic of planned MLDD methodology is especially used for detection method. The schematic for management unit is shown in Fig.6. It consists of (1) 2 OR gate; (2) a counter unit; (3) a FSM unit; and (4) a detection register.

Counter unit is employed to counts up to a few, that distinguishes {the initial|the primary} 3 iterations of cubic centimeter decryption and evaluates the output from XOR matrix Bj by giving it as input to first OR circuit.

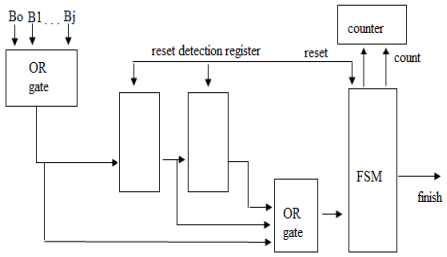

Fig.6. Schematic of MLDD control unit.

Output of this OR circuit is fed as input to three-stage shift registers that act as a detection register. This register already contains the results of the previous 3 stages keep in it. The results area unit shifted and therefore the output is directly given as associate input to the second OR circuit.

In the third cycle, the second OR circuit evaluates the content of the detection register. per the output of the second OR circuit, FSM can generate the validity of the

codeword. If the result's '0', FSM sends the end signal, that indicates that the processed word is error-free. If the result's '1', the decryption method can continue till the tip of N-cycles. this means that solely the codeword with error solely takes N-cycles.

C. Corrector

1) One-Step Majority-Logic Corrector: One–step majority logic correction is that the method that identifies the right price of associate every bit within the codeword directly from the accessed codeword. This method is distinction to the overall message-passing error correction strategy [14], which can demand multiple iterations.

This methodology consists of set of linear sums of received vector bits and majority price of computed linear sums to point the correctness of the codeword. the most aim of the ballroom dance majority-logic corrector is generating γ parity-check matrix.

The procedure for ballroom dance majority logic error correction is given in 2 steps:

1) Generate γ parity-check sums by computing the real number of the received vector and therefore the applicable rows of parity-check matrix.

2) The γ check sums area unit fed into a majority gate. If the output of the bulk gate is '1' then the output of majority gate corrects the bit Cn-1 by inverting the worth of Cn-1.

The circuit implementing a serial ballroom dance majority logic corrector for (15, 7, 5) EG-LDPC code is shown in Fig.1. This circuit computes the bulk price of parity-check sums by generating γ parity-check sums with γ XOR gates. every XOR gate produces a linear add victimisation ρ inputs. every check add is calculated employing a row of the parity-check matrix thus row density of the EG-LDPC codes is ρ. The XOR gate in Fig.1 corrects the code bit Cn-1 victimisation the output of the bulk gate.

Once the code bit Cn-1 is corrected the codeword is cyclic shifted and code bit Cn-2 is placed at Cn-1 position and can be corrected. the complete codeword is corrected in n cycles. The two-level majority gate is enforced by conniving all the merchandise terms that have [γ+1/2] ON inputs and one input OR-term [15]. as an example, the bulk of three inputs a, b, c is computed with three product terms and one 3-input OR-term as,

$$\text{Majority}(a,b,c)=ab+ac+bc$$

IV. RESULTS

Simulations are done using ModelSim SE 6.3f tool.

TABLE II
COMPARISON OF TWO DECODING TECHNIQUE FOR 15-BIT CODEWORD BASED ON FEW PARAMETERS

PARAMETER	15-BIT		30-BIT	
	MLD	MLDD	MLD	MLDD
Number of cycles (with error)	15	15	30	30
Number of cycles (without error)	15	3	30	6
Minimum period	13.006 ns	12.578 ns	10.890 ns	10.374 ns
Power consumption	162 mW	143 mW	232 mW	222 mW
Maximum frequency	76.888 MHz	79.504MHz	91.827 MHz	96.395 MHz
Minimum input arrival time before clock	6.126 ns	6.126 ns	6.263 ns	6.985 ns
Maximum output required time after clock	7.544 ns	7.544 ns	7.157 ns	7.067 ns

Majority Logic Detector detects the occurrences of any bit flip gift in codeword, when the process of N variety of steps for received codeword with N variety of bits. The presence of any bit flip within the received code word is effectively detected solely at the tip of ordinal cycle. thus it's necessary to bear N steps for N bit codeword even though there's single bit flip happens. Here the codeword thought-about is 15-bit codeword, thus cubic centimeter decryption takes fifteen cycles to decipher 15-bit codeword, even if there's no error in codeword, is shown in Table II.

In the planned MLDD methodology, if the 15-bit codeword is error-free, then output are going to be processed within the 3 iteration. It helps to get the lead to 3 cycles. so the decryption time is reduced in MLDD in comparison to MLD.

V. CONCLUSION

In this paper, associate error-detection mechanism, Majority Logic Decoder/Detector, cubic centimeterDD has been planned supported ML decryption victimisation EG-LDPC codes. Simulation results unconcealed that the planned technique is capable of sleuthing any bit errors within the initial 3 cycles of the decryption method. This improves the performance of the look with regard to the cubic centimeter decryption methodology. The performance of the planned cubic centimeterDD methodology is quicker in comparison to existing methodology ML decryption. conjointly the MLDD error detector module designed is freelance of the dimensions of the codeword. thus space overhead is additionally reduced compared with existing cubic centimeter decryption methodology. The planned methodology detects the fault in precisely 3 cycles thus variety of clock cycles are going to be saved.

In future, decryption/detection method are going to be enforced in parallel and its decryption time are going to be compared with the serial MLDD decoding methodology.

REFERENCES

[1]. R.C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *IEEE Trans. Device Mater. Reliabil.,* vol.5, no3, pp. 301-316, Sep.2005.

[2]. C.W. Slayman, "Cache and Memory error Detection, correction, and reduction techniques for terrestrial servers and workstations," *IEEE Trans. Device Mater. Reliabil.,* vol.5, no3, pp.301-316, Sep.2005.

[3]. R. Naseer and J. Draper, "DEC ECC design to improve memory reliability in sub-100 nm technologies," *Proc. IEEE ICECS*, pp.586-589, 2008.

[4]. J.A. Maestro and P. Reviriego, "Reliability of single-error correction protected memories," *IEEE Trans. on Reliability,* vol.58, no.1, pp. 193-201, Mar.2009.

[5]. T. Sasada, S. Ichikawa, and T. Kanai, "Measurement of single-event effects on a large number of commercial DRAMs," *IEEE Trans. on Nuclear Science*, vol.53, no. 4, pp. 1806-1812, Aug.2006.

[6]. M.A. Bajura et al., "Models and algorithmic limits for an ECC-based approach to hardening sub-100-nm SRAMs," *IEEE Trans. Nucl. Sci.,* vol. 54, no. 4, pp. 935-945, Aug.2007.

[7]. Yu Kou, Shu Lin, and Marc P.C. Fossorier, "Low-Density Parity-Check Codes Based on Finite Geometries," *IEEE Trans. on Information Theory,* vol. 47, no. 7, November 2001.

[8]. S.Lin and D. J. Costello, "Error Control Coding," 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2004.

[9]. T.Richardson, A. Shokrollahi, and R. Urbanke, " Design of Capacity-Approaching Irregular Codes," *IEEE Transactions on Information Theory*, vol. 47, no.2, pp.619-637, February 2001.

[10]. R. Lucas, M. Fossorier, Y. Kou, and S. Lin, "Iteration Decoding of One-Step Majority Logic Decodable Codes Based on Belief Propagation," *IEEE Transactions on communications,* vol. 48, pp. 931-937, June 2000.

[11]. H. Naeimi and A. DeHon, " Fault Secure Encoder and Decoder for Nano Memory applications," *IEEE Trans. Device Mater. Reliabil.,* vol.5, no.3, pp. 397-404, September 2005.

[12]. S. Liu, P. Reviriego, and J. Maestro, " Efficient Majority Logic Fault Detection with Difference-Set Codes for Memory Applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.,* vol.20, no.1, pp.148-156, Jan. 2012.

[13]. H. Naeimi and A. DeHon, "Fault Secure Encoder and Decoder for Memory Appication," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.,* vol. 17, no. 4, pp. 437-486, 2009.

[14]. S. Liu, G. Sorrenti, P. Reviriego, F. Casini, J.A.Maesro, M. Alderighi, and H. Mecha, "Comparison of the susceptibility to soft errors of SRAM-Based FPGA Error Correction Codes Implementations," *IEEE Trans. Nuclear science.*,vol.59, no.3, june 2012.