RESEARCH ARTICLE OPEN ACCESS

An Efficient Fault Tolerance System Design for Cmos/Nanodevice Digital Memories

D. Kavitha[1], R. Anandh[2], D. Karthikeyan[3]
[1]Assistant professor, Department of ECE, Arasu Engineering College, Kumbakonam.
[2]Assistant professor, Department of ECE, Arasu Engineering College,kumbakonam.
[3]PG Scholar, Department of ECE, Arasu Engineering College,kumbakonam,

ABSTRACT
Targeting on the future fault-prone hybrid CMOS/Nanodevice digital memories, this paper present two fault-tolerance design approaches the integrally address the tolerance for defect and transient faults. These two approaches share several key features, including the use of a group of Bose-Chaudhuri- Hocquenghem (BCH) codes for both defect tolerance and transient fault tolerance, and integration of BCH code selection and dynamic logical-to-physical address mapping. Thus, a new model of BCH decoder is proposed to reduce the area and simplify the computational scheduling of both syndrome and chien search blocks without parallelism leading to high throughput.The goal of fault tolerant computing is improve the dependability of systems where dependability can be defined as the ability of a system to deliver service at an acceptable level of confidence in either presence or absence falult.ss The results of the simulation and implementation using Xilinx ISE software and the LCD screen on the FPGA's Board will be shown at last.

Keywords—Bose- Chaudhuri – Hocquenghem (BCH)codes, complementary metal oxide semiconductor (CMOS), Defect/fault tolerance, error correcting code (ECC), very large scale integration circuits(VLSI).

I. INTRODUCTION

This past few years experienced spectancular advances in the fabrication and manipulation of molecular and other nanoscale device [1]. Although these new devices show significant future promise to sustain Moore's law beyond the CMOS scaling limit, there is a growing consensus [2], that at least in the short term, they cannot completely replace CMOS technology.

As a result, there is a substantial demand to explore the opportunities for CMOS and molecular/nanotechnologies to enhance and complement each other. This naturally leads to a paradigm of hybrid CMOS/nanodevice nanoelectronics, where any of array of nanowire crossbars, with wires connected by simple nanodevices at each cross point site on the top of a bulk information processing and/or storage, while the CMOS circuit may some perform testing and fault tolerance, global interconnect, and some other critical functions. It is almost evident that, compared with the current CMOS technology, any emerging nanodevices will have (much) worse reliability characteristics (such at the probabilities of permanent defect and transient fault).

Hence, fault tolerances have been well recognized as one of the biggest challenges in the emerging hybrid nanoelectronics.

II. BCH CODE

This work concerns the fault-tolerant system design for hybrid nanoelectronic digital memories. Conventionally, defect and transient faults in CMOS digital memories are treated separately, i.e., defects are compensated by using spare rows, columns, and/or words to repair (i.e., replace) the defective ones, while transient faults are compensated by error correcting codes (ECC) such Hamming and Bose-Chaudhuri-Hocquenghem (BCH) codes.

The Bose-Chaudhuri- Hocquenghem (BCH) codes form a large class of powerful **random error-correcting cyclic** codes. This class of codes is a remarkable generalization of the Hamming codes for multiple-error correction. For any positive integers m ($m \geq 3$) and t ($t < 2^m-1$), there exists a binary BCH code with the following parameters:

- Block length: $n=2^m-1$
- Number of parity check digits : $n- k \leq mt$
- Minimum distance: $d_{min} \geq 2t$

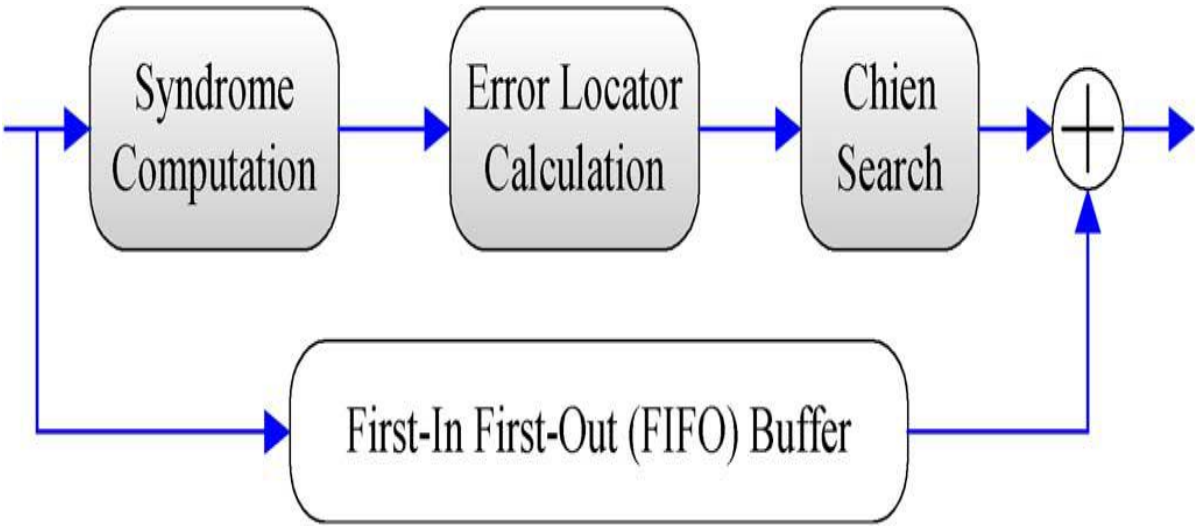

Figure.2 Binary BCH code decoder structure.

Clearly, this code is capable of correcting any combination of t or fewer errors in a block of $n = 2^m-1$ digits. We call this code a t- error-correcting BCH code. The generator polynomial of this code is specified in terms of its roots from the Galois field GF (2^m) using binary BCH code decoder structure figure(1).

BCH decoding

The basic idea of the BCH code decoder is to detect an erroneous sequence with few words, who summoned the received data, gives rise to a valid code word.

Several steps are required for decoding these codes:

- Calculation of syndrome.
- Calculation of polynomials error localization and amplitude.
- Calculation of roots and evaluation of two polynomials.
- Sum of the polynomial consists of the polynomial and to reconstruct the received information.
- Start without error.

This can be summed in the upcoming figure for easier conception of the VHDL source-code that we will be using in the conception of our BCH decoder. The BCH codes are implemented as cyclic codes, that is, the digital logic implementing the encoding and decoding algorithms is organized into shift-register circuits that mimic the cyclic shifts and polynomial arithmetic required in the description of cyclic codes. Using the properties of cyclic codes, the remainder can be obtained in a linear stage shift register with feedback connections corresponding to the coefficients of the generator polynomial as shown in the following figure(2),

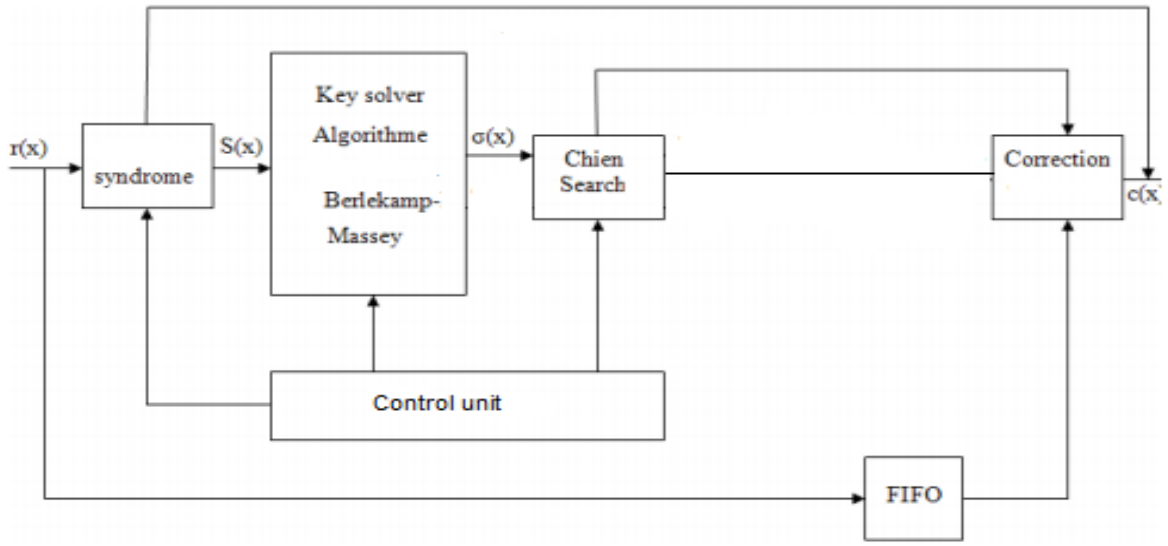

Figure.2 Digital circuit for BCH decoder

Where:

R(x): received code word
S(x): the calculated syndrome
σ(x): The eror locating polynomial
C(x): Codeword after decoding

.

In this paper we used the algorithm of BERKLAMP-MASSEY from the fact that it was specially made for the decoding of this type of codes. The logic chart of this algorithm is:

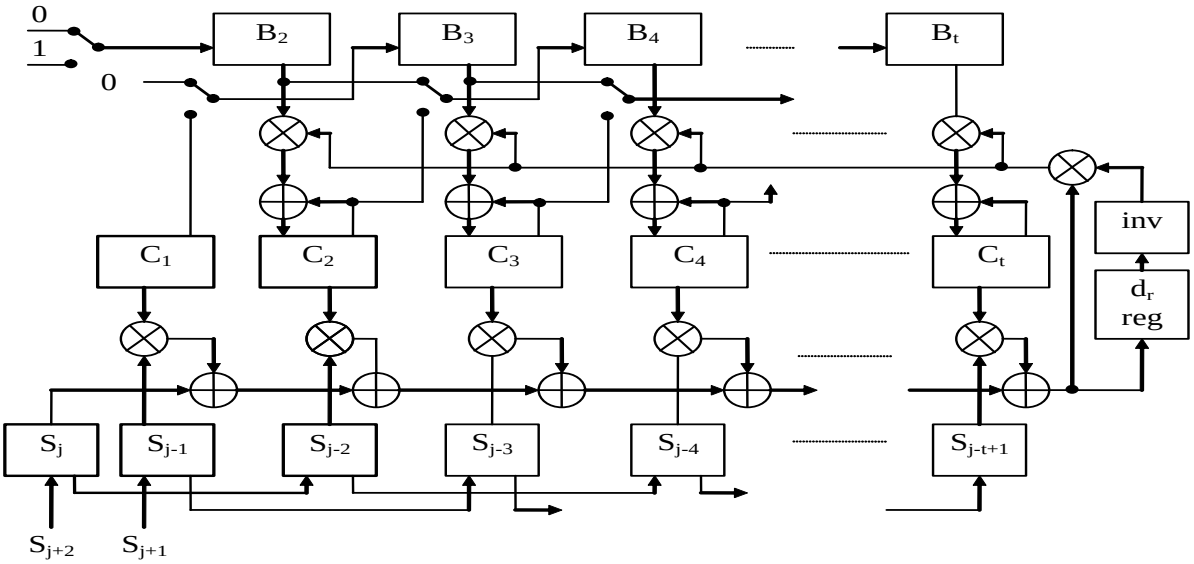

Figure 3. Berlekamp Massey Algorithm with inversion.

Note that the above algorithm is slightly modified in comparison with the previously presented BMA [2]. Due to more complicated initial states, the number of iterations is decreased by one. In practice, this causes only a slight increase in the hardware requirements but the BMA calculation time is significantly reduced. A circuit implementing the BMA is given in Figure2. The error location polynomial $\sigma(x)$ is obtained in the *C* registers after *t-1* iterations. In some applications it may be beneficial to implement the BMA without inversion.

III. BINARY BCH CODES AND DECODER IMPLEMENTATION

1. Background

Because of their strong random error correction capability, binary BCH codes[3] are among the best ECC candidates for realizing fault tolerance in hybrid nanoelectronic digital memories where the faults(both defects and transient faults) are most likely random and statistically independent. Binary BCH code construction and encoding/decoding are based on binary Galois fields. A binary Galois field with degree of m is represented as GF (2^m).

For any $m \geq 3$ and $t \leq 2^{m-1}$, there exists a primitive binary BCH code over GF (2^m), denoted as C^m (t), that has the code length $n=2^m-1$ and information bit length $k \geq 2^m-m.t$ and can correct up to (or slightly more than) t error.

For most values of t, $C^m(t+1)$ requires m more redundant bits than $C^m(t)$. A primitive t-error-correcting (n, k, t) BCH code can be shortened (i.e., eliminate a certain number, say s, of information bits) to construct a t-error-correcting (n-s, k-s, t) BCH code with less information bits and code length but the same redundancy[3].

Although BCH encoding is very simple and only involves a Galois field polynomial multiplication, BCH code decoding algorithms may lead to (slightly) different decoding computational result, (n, k, t) binary BCH code under GF(2^m) , the product of the decoder silicon area and decoding latency is approximately proportional to $n \cdot t \cdot m^2$.

Moreover, a group of binary BCH codes under the same GF (2^m) can share the same hardware encoder and decoder that are designed to accommodate the maximum code length, maximum information bit length, and maximum number of correctable errors among all the codes within the group.

For a detailed discussion on BCH codes and their encoding/decoding, readers are referred to [4] and [5].

In order to realize satisfactory defect tolerance efficiency, the repair-only approach requires very low defect densities that can be readily met by current CMOS technology. Nevertheless, the much higher defect densities of nanodevice make the repair-only approach not sufficient.

Which naturally demands extending the use of ECC for both defect tolerance and transient fault tolerance, Because of the dual role of ECC, defect tolerance transient fault tolerance should be addressed integrally.

More importantly, realization of fault tolerance and transient fault tolerance in hybrid nanoelectronic memory will incur area, energy and operational latency overhead in CMOS domain, e.g., the overhead incurred by the implementation of ECC decoder and reliable storage of certain nanodevice memory configuration information in CMOS memory.

Such overhead in CMOS domain must be taken into account when investigating and evaluating hybrid nanoelectroic digital memory fault-tolerant system design solution. Defect tolerances in hybrid nanoelectronic digital memory have been addressed[6].

In the authors analysed the effectiveness of integrating Hamming codes with spare tow/column repair only defect tolerance. The ECC-only defect tolerance has been used to estimate the hybrid nanoelectronic memory storage capacity.

This paper presents in hybrid nanoelectronic digital memory fault-tolerant system design approaches using strong BCH codes, and evaluates the BCH coding system implementation overhead in CMOS domain based on practical IC design.

We understand that, at this early stage of nanoelectronic when few preliminary experimental data under laboratory environments have been ever reported, there is a large uncertainty of the defect and transient fault statistical characteristics (such as their probabilities and temporal/spatial variations) in the future real-life hybrid CMOS/nanodevice digital memories.

Table :1

Port Name	Type	Description
Data	Input	16-bit data input to digital memory
Clock	Input	Clock
read address	Input	4-bit read address input
write address	Input	4-bit write address input
We	Input	Write enable input
Q	Output	8-bit data output of digital memory

Table.1 Binary memory port pin allocation

Therefore, instead of attempting to provide a definite and complete fault-tolerant system design solution, this work mainly concerns the feasibility and effectiveness of realizing memory fault tolerance under as-worse-as-possible scenarios.

In particular, we are interested in the fault-tolerant strategies with two features: 1) they should be high as possible of the defect probabilities and transient fault rates and 2) they can automatically adapt to the variations of the defect statistics in digital memories (i.e., the on-chip fault-tolerant system can automatically provide just enough defect tolerance capability for a wide range of defect[7].

2. Code Construction And Decoder Implementation

In nanodevice memory, due to the high defect probabilities and their possibly large temporal/spatial variations, different physical memory portions may have (largely) different physical memory cell hence demand (largely) different error correcting capability. Therefore, other than using a single BCH code, we propose to use a group of BCH codes in the group should be constructed under the same binary Galois

field. In this work, to demonstrate and evaluate the proposed fault- tolerance design approaches, we constructed four BCH code groups as list[8].

A binary BCH code decoder consists of three computational blocks and one first in first out (FIFO) buffer, while implementation of syndrome computational and Chine search blocks are straightforward, the realization of error locator calculation is nontrivial and several algorithms have been proposed in the regard.

In this work, we use the inversion – free Brelekamp – Massey to realize the error locator calculation. To minimize the decoders are fully serial, i.e., it receives 1 – bit input and generates 1- bit output per clock cycle[9].

As mentioned in the above, multiple **BCH codes(e**ight BCH per group in this work**), wh**ich share the same encoding and decoding circuit, have been used for error correction. Although the use of multiple BCH codes may potentially improve the effective storage capacity in the nano domain, it will incur storage overhead n cmos doamn, leading to a design tradeoff.

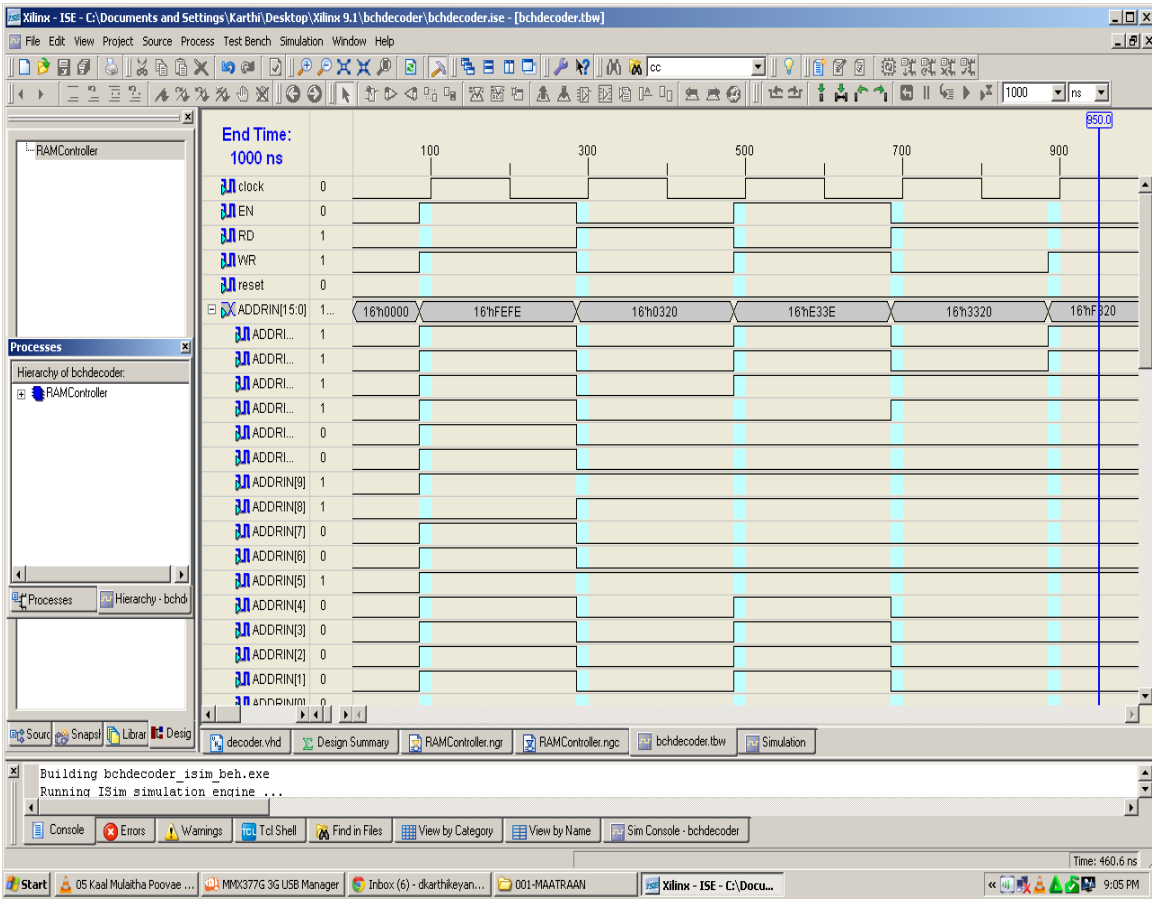

Figure.4 Binary BCH code and test bench wave form

IV. PROPOSED FAULT – TOLERANT DESIGN APPROACHHES

In this work, we assume the following fault model for nanodevice memory. In terms of defect, we only consider static defects of nanowires and nanodevice memory cells. We assume a defective nanowire (irrelevant to defect type) will make all the connected nanodevice memory cells unfunctional. A memory cell may be subject to open or short two orthogonal nanowire defects. An open memory cell defect does not affect the operation of any other are memory cells and any nanowires.

We assume these static defects are random and statically independent, which are characterized by two defect probabilities, including:

1) Bit defect probability p_{bit} that represents the probability of the open memory cell defect. Given the BCH code group and memory defect map, a fault-tolerant system should determine.

2) Which BCH codes should be used for protecting each bit user data block and 2) how to physically map each BCH coded data block onto the nanodevice memory cell.

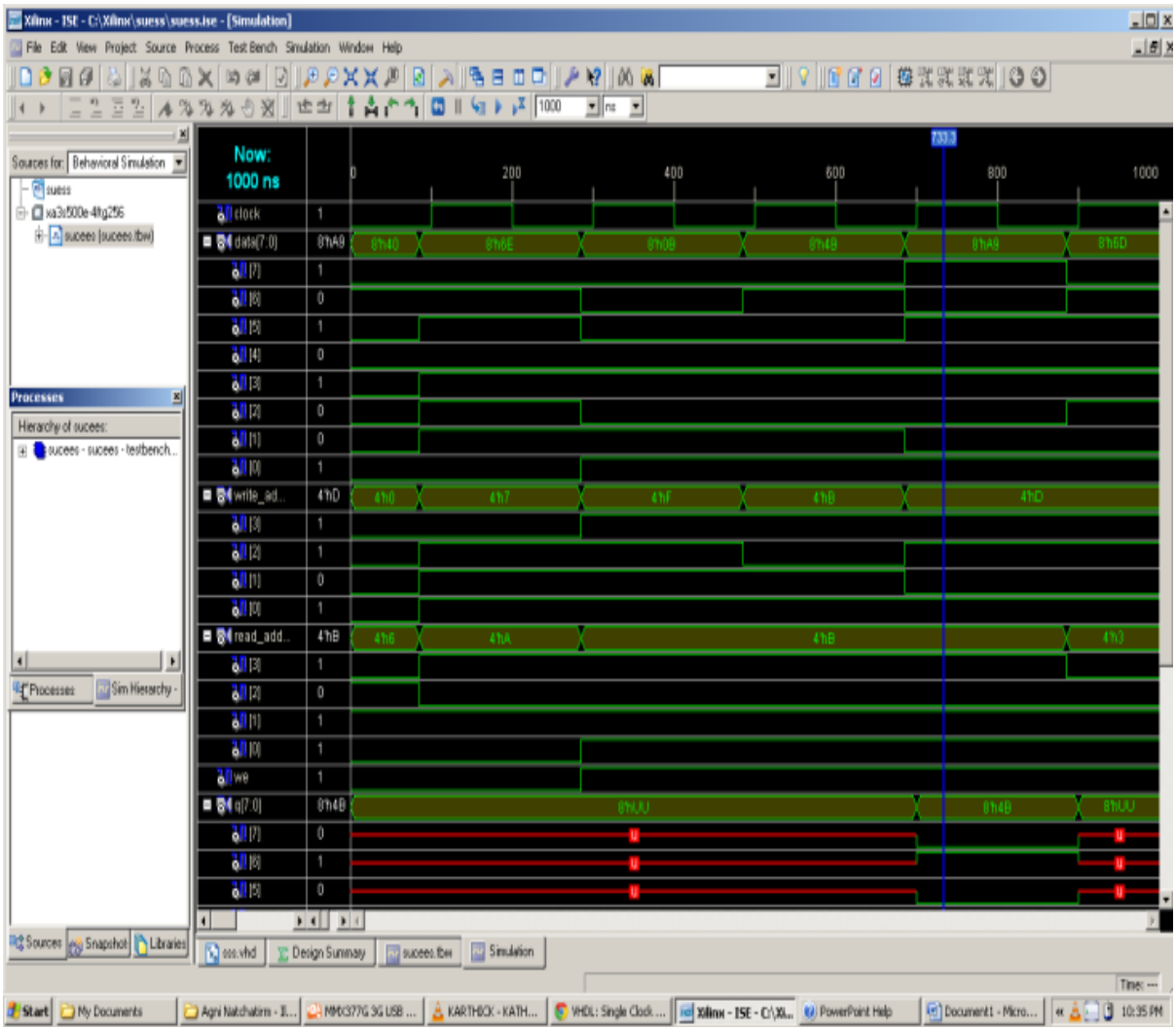

Figure.5 simulation wave form fault analysis

A. Approach I: Two Level Hierarchical Fault Tolerance

The basic idea of this design approach can be described as follows: we partition each nanodevice memory cell array into a certain number of memory cell segments; each segment contains consecutive memory cells and can store one BCH codeword that provide just enough coding redundancy to compensate all the defects in present segment and ensure a target block error rate under a given transient fault rate in figure(4).

This first approach is simple and works well under relatively low and modest bit probabilities and/or transient fault rate.

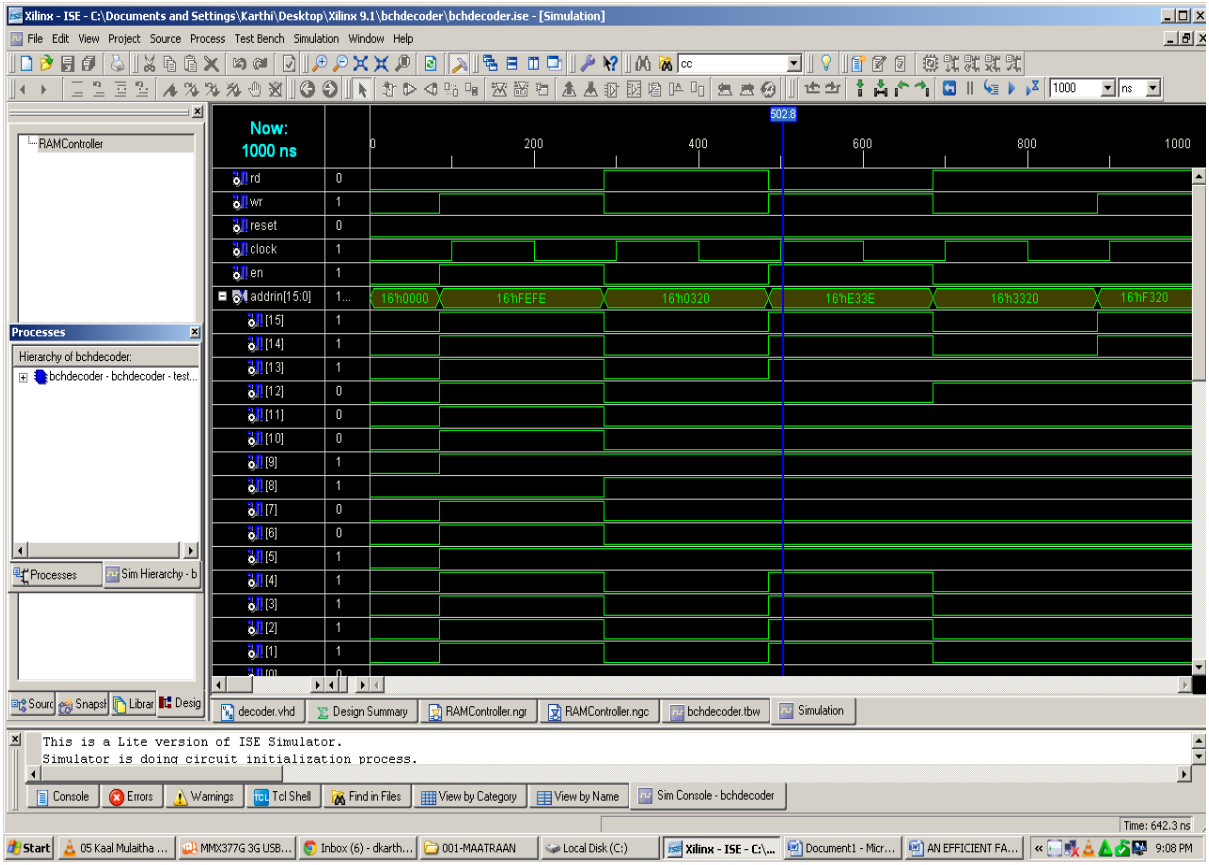

Figure.6 Simulation result for first level approaches

B. Approach II: Three- Level Hierarchical Fault Tolerance

In the above two- level hierarchical design approach, we always attempt to locate a continuous memory cell segment to store each coded data block. Hence, with high bit defect probabilities, the total number of defective memory cells within a segment may accumulate very quickly and exceed the maximum error correcting capability.

V. CONCLUTION

In this paper, we presented two fault-tolerance design approaches that integrally address the defect tolerance and transient fault tolerance for hybrid CMOS/nanodevice digital memories. To accommodate the high defect probabilities and transient fault rates, the developed approaches have several key features that have not been used in conventional digital memories, including the use of a group of BCH codes for both defect tolerance and transient fault tolerance, and integration of BCH codes selection and dynamic logical-to-physical address mapping. These two fault-tolerance design approaches seek different tradeoffs among the achievable storage capacity, robustness to defect statistics variations, implementation complexity, and operational latency and CMOS storage overhead. Simulation results demonstrated that the developed approaches can achieve good storage capacity, while taking into account of the storage overhead in CMOS domain, under high defect probabilities and transient fault rate, and can readily adapt to large defect statistics variations. To evaluate the BCH code coding system implementation overhead, we designed the corresponding BCH decoders at CMOS technology node.

REFERENCES

[1] Y. Chen, G. Y. Jung, D. A. A. Ohlberg, X. Li, D. R. Stewart, J. O. jeppesen, K. A. Nielsen, J. F. Stoddart, and R. S. Williams, "*Nanoscale molecular-switch crossbar circuits*," Nanotechnology, vol.14, pp. 462-468, Apr.2003.

[2] M. A. Reed, "*Molecular- scale electronics,*" *Proc. IEEE*, vol. 87, no. 4, pp. 652-658, Apr. 1999.

[3] T. Rueckes et al., "*Carbon nanotube-based nonvolatile random access memory for molecular computing,"* Science, vol. 289, pp.94-97, 2000.

[4] N. A. Melosh et al., "*Ultra high-density nanowire lattices and circuits*," Science, vol. 300, pp. 112-115, 2003.

[5] M. R. Stan, P. D. Franzon, S. C. Goldstein, J. C. Lach, and M. M. Ziegler,

"*Molecular electronics: From devices and interconnect to circuits and architecture*," *Proc. IEEE*, vol. 91, no. 11, pp. 1940-1957, Nov. 2003 M. J. 1994.

[6] M. M. Ziegler and M. R. Stan, "*CMOS/nano co-design for crossbar- based molecular electronic systems*," *IEEE Trans. Nanotechnol.*, vol. 2, no. 4, pp.317-230, Dec. 2003

[7] A. Dehon, S. C. Goldstein, P. J. Kuekes, and P. Lincoln, "*Nonphotolothographic nanoscale memory density prospects*," *IEEE Trans. Nanotehnol.*, vol. 4, no. 2, pp. 215-228, Mar. 2005.

[8] D. B. Strukov and K. K. Likharev, "*Defect-tolerant arhitechtures for nanoelectronic crossbar memories*," J. Nanosci. Nanotechnol., vol. 7, pp. 151-167, Jan. 2007.

[9] H. O. Burton, "*Inversionless decoding of binary BCH codes," IEEE Trans. Inf. Theory*, vol. IT-17, no. 4, pp.464-466, Jul. 1971.

Journal Papers:

E.H.Anas, ; R.El gouri, L. Hlou « *A low power error detection in the Chien Search Block for Reed-Solomon code* », IEEE conference publications Complex Systems (ICCS), 2012 International Conference, (2012) , pp: 1 - 3

BOOKS: BEYOND CMOS NANODEVICES 2, THIS BOOK OFFERS A COMPREHENSIVE REVIEW OF THE STATE-OF-THE-ART IN INNOVATIVE BEYOND-CMOS NANODEVICES FOR DEVELOPING NOVEL FUNCTIONALITIES, LOGIC AND MEMORIES DEDICATED TO RESEARCHERS, ENGINEERS AND STUDENTS. THE BOOK WILL PARTICULARLY FOCUS ON THE INTEREST OF NANOSTRUCTURES AND NANODEVICES (NANOWIRES, SMALL SLOPE SWITCHES, 2D LAYERS, NANOSTRUCTURED MATERIALS, ETC.) FOR ADVANCED MORE THAN MOORE (RF-NANOSENSORS-ENERGY HARVESTERS, ON-CHIP ELECTRONIC COOLING, ETC.) AND BEYOND-CMOS LOGIC AND MEMORIES APPLICATIONS.

CHAPTER2. NANOWIRE DEVICES 25
Gérard GHIBAUDO, Sylvain BARRAUD, Mikaël CASSÉ, Xin Peng WANG, Guo Qiang LO, Dim-Lee KWONG, Marco PALA and Zheng FANG

Theses:

Raghav Bhaskar, Pradeep K. Dubey, Vijay Kumar, Atri Rudra, "*Efficient Galois Field Arithmetic On SIMD Architectures*", ACM Symp. On ParallelAlgorithms and Architectures, pp 256-257, 2003.

Proceedings Papers:

H.Hussien, K.A.Shehata,; M.Khedr, S.Hareth, "*Performance study on implementation of DVB-S2 low density parity check codes on additive white Gaussian noise channel and Rayleigh fading channel* » Electronics Design, Systems and Applications (ICEDSA) 2012 IEEE International Conference on, (2012) , pp: 179 – 182

ABOUT THE AUTHORS

D.KARTHIKYAN is doing ME degree in VLS**I Des**ign from ARASU ENG**INEERING COLLEGE ,Kumbakonam** in knowledge institute of Technology, He received his BE degree in Electroincs and Communication Engineering from Arasu Engineering College kumbakonam in 2012. Email. id:- dkarthikeyand6@gmail.com.

D.KAVTIHA has completed ME Network Engineering in **Arulmigu Kalasalingam college of Engineering, Krishnan koil.** She is having 5(five) years teaching experience. She has published 5(five) papers in National conference. Her research areas of interest are Network Security especially in Network processor, cloud computing. She is currently working as a Asst.Professor in the department of Electronics and Communication Engineering at ARASU ENG**INEERING COLLEGE ,Kumbakonam.** Email. id:- kavithadurairaj1984@gmail.com.

R. ANANDH, completed his B.Tech (Inforamation Technology) from Annai Teresa college of Engineering,Panruti in 2010. He received his ME degree in (Embedded systems technology) from karpaga vinayaga college of engineering and technology Chennai, in 2012. His research areas of interest are Network Security especially in **Wireless and communcation** Network, **He** is currently working as a Asst.Professor in the department of Electronics and Communication Engineering at ARASU ENG**INEERING COLLEGE ,Kumbakonam. Email. id:- anandhraj3@gmail.com.**