FAULT ENHANCEMENT FOR MEMORY TESTING

Mr.G.KARTHY, M.SRIRAMPRIYA, S.SOWMIYA, M.YOGESWARI,IV Year ECE

1-Assistant Professor II &2,3.4- Students

Kalasalingam University,Krishnan koil

Abstract— **Memories are an important aspect as it's technology growth increases. RAM, ROM, DRAM etc are the different types of memory and it becomes difficult to test the memory as the complexity of system increases. The embedded memories are on SOCs in which the embedded RAM memory is very hard to test as it's testing needs a large number of pattern stimuli to be delivered to memory and retrieving a huge data. In this project a new test architecture is designed using response analyzer and checker to detect fault on a chip. And decided to implement March test algorithm because of its regularity in achieving high fault coverage .March test algorithms are used to detect the maximum faults. It has number of operations per element according to the today's growing needs of embedded memory testing with enhanced fault using programming languages.**

Keywords—RAM, SOC, response analyzer, checker, March algorithm.

I . INTRODUCTION

Testing Random Access Memories for all possible failures is not feasible. The fault that occurs in memories has to be restricted as different types of class and it is called as fault model. Many fault models for RAMs have been proposed. In this paper we consider traditional fault models and tests which allow to detect not only single but also multiple faults. We choose only march test algorithms which require test times on the order of n (where n is the number of bits on the memory). Algorithms which require test time on order of n/2 or n3/2 are impractical for modern high density RAM chips (16 megabits and more).In this paper we proposed a new architecture which consist of checker, response analyzer, memory unit and a BIST controller. By using checker we can get more précised output.

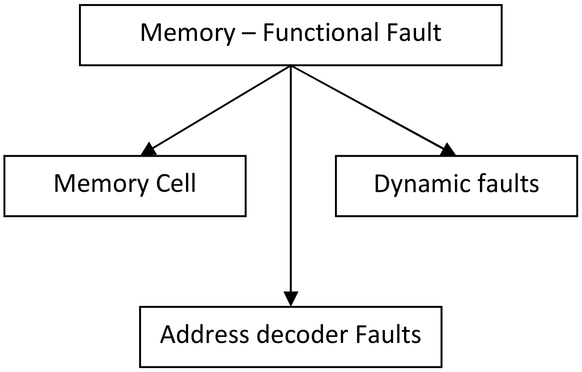
II PROBLEM STATEMENT:

Memory tests are used to confirm that each location in memory device is working.This involves writing a set of data to each memory address and verifying this data by reading it back.If all the values read back are the same as that were written,then memory device is said to pass the test otherwise device fails. March based algorithms were capable of locating and identifying the fault types which can help to catch design and manufacturing errors.The quality of test is strongly dependent on the fault model in terms of its fault coverage,its test length as well as the test time required.

MBIST usually use the deterministic pattern such as March test algorithm to test memories . The importance of developing new fault model increases with the new memory technologies.Traditional test like MARCH C- are thus becoming insufficient ,for today 's and the future high speed memories .Therefore it is essential to design new test algorithms which fullfill the need of detecting new faults.

In this project MARCH C algorithm is implemented to detects maximum fault .In addition to that Response analyser and Checker are include in this architecture to identify more faults with high précision.

III. FAULT OCCURS IN MEMORY



Memory cell faults [1]

1. Stuck-at fault (SAF): cell or line s-a-0 or s-a-1.

2. Stuck-open fault (SOF): open cell or broken line.

3. Transition fault (TF): cell fails to transit.

4. Data retention fault (DRF): cell fails to retain its logic value after some specified time due to leakage, resistor or feedback path opens.

5. Coupling fault (CF): Coupling faults are of three types.

- Inversion coupling fault (CFin): a transition in one cell (aggressor) inverts the content of another cell (victim).
- Idempotent coupling fault (CFid): a transition in one cell forces a fixed logic value into another cell.
- State coupling fault (CFst): a cell/line is forced to a fixed state only if the coupling cell/line is in a given state (a.k.a. pattern sensitivity fault (PSF)).

6. Bridging fault (BF): short between cells (can be

AND type or OR type)

7. Neighborhood Pattern Sensitive Fault (NPSF)

8. Active (Dynamic) NPSF

9. Passive NPSF

10. Static NPSF

Address decoder faults (AFs)

3. Data Retention faults: Data cannot be retrieved because memory loses its content spontaneously.

- DRAM refresh fault: stuck-at fault occurs in Refresh-line.
- DRAM leakage fault:
  Sleeping sickness—it usually affects a row or a column. When charge leakage or environment sensitivity occurs the loss of data is less than specified hold time causes sleeping sickness.

  Static data losses— the state of a cell Checkerboard pattern changes and triggers max leakage due to defective pull-up device inducing excessive leakage currents.

VI. MARCH ALGORITHM:

Finite sequence of read and write operations are carried out to verify whether the memory cell is good. The targeted fault model decides the actual number of read/write operations and the order of the operations. March tests are the most commonly used memory test algorithms, in which there are finite sequences of March elements. Then March element is applied to a cell in memory one by one. The operation can be in either ascending or descending address order. The notations of the March algorithm are summarized below The response will be 0 or 1, if test algorithm reads a cell, and they are specified as r0 and r1, respectively. Similarly, writing a 1 into a cell is denoted as w1 and writing 0 as w0. In Table below we summarized various March test algorithms. This table summarizes which faults and linkages are covered by all of the March tests [6], [7].A March-based test algorithm is a finite sequences of read and write operation called March element. It is specified by a number of read and write operations and n address order.

1. No cell accessed by certain address.

2. Multiple cells accessed by certain address.

3. Certain cell not accessed by any address.

4. Certain cell accessed by multiple addresses.

Dynamic Faults

1. Recovery faults: when some part of the memory cannot recover fast enough from a previous state [2].

- Sense amplifier recovery: After reading/writing a long string of 0s or 1s
  Sense amplifier becomes saturation.
- Write recovery: write at a different location or a write followed by a read resulting in reading or writing at the same location due to slow address decoder.

2. Disturb faults*:* victim cell forced to 0 or 1 if we read or write aggressor cell (may be the same cell).

MATS, MATS+, March-C, March-Y, March-A and March-B are different types of March-based tests. In most modern memory BIST, March based test algorithm is implemented because of its simplicity and high fault coverage.

MARCHING 1/0 Test [*Breuer & Friedman, 1976*]:

It is a complete test for AF's, SAF's and TF's but has the ability to detect only a part of CF's [2]. The test sequence is given. It is a test of 14n complexity.

MATS Test [*Nair]*:

MATS stands for Modified Algorithmic Test Sequence. It is the shortest MARCH test for unlinked SAF's in memory cell array and read/write logic circuitry [3]. The algorithm can detect all faults for OR type technology since the result of reading multiple cells is considered as an OR function of the contents of those cells. It can also be used for Address faults of AND type technology using the MATS-AND test sequence given below [2]. The MATS Algorithm has a complexity of 4n with a better fault coverage compared to equivalent zero-one and checkerboard tests .

MATS+ Test [*Abadir & Reghbati, 1983*]:

It is often used instead of MATS when the technology used under test is unknown and detects all SAF's and AF's. It has a test complexity of 5n.

MATS++ [*Goor, 1991*]:

It is similar to the MATS+ test but allows fault coverage for TF's. Its test sequence is a complete, irredundant, & optimized test sequence It has a test complexity of 6n.

MARCH X [*unpublished*]

The MARCH X test is called so since it has been used without being published. It detects all unlinked SAF's, AF's, TF's and CFin's. It has a test complexity of 6n.[4]

MARCH C [*Marinescu, 1982*][10]:

The MARCH C test is suited for AF's, SAF's, TF's and all CF's . It is a test of 11n complexity.

MARCH C- [*Goor, 1991*]

This test sequence is a modification to MARCH C test implemented in order to remove redundancy present in it. It detects unlinked AF's, SAF's, TF's and all types of CF's. This test has the complexity 10n.

MARCH A [*Suk & Reddy, 1981*]

The MARCH A test is the shortest test. It detects AF's, SAF's, linked CFid's, TF's and certain CFin's linked with CFid's [2]. It is a complete and irredundant test. It has a complexity 15n.

MARCH Y [*unpublished*]

MARCH X test is extended and it is called as MARCH Y. It can detect all faults detectable by MARCH X. This test is of complexity 8n.[4]

MARCH B [Suk & Reddy, 1981];

The MARCH A test is an extended and it is called MARCH B test. It is a complete and irredundant test. It is capable of detecting AF's, SAF's, linked CFid's or CFin's. It has a complexity 17n.

Algorithm	**Notations**
MATS	{↕(w0);↕(r0,w1);↕(r1); }
MATS+	{↕(w0);↑(r0,w1);↓(r1,w0); }
MATS++	{↕(w0);↑(r0,w1);↓(r1,w0,r0); }
March X	{↕(w0);↑(r0,w1);↓(r1,w0);↕(r0); }

March C	{↕(w0);↑(r0,w1);↑(r1,w0);↕(r0);
	↓(r0,w1);↓(r1,w0);↕(r0); }

March A	{↕(w0);↑(r0,w1,w0,w1);↑(r1,w0,w1);
	↓(r1,w0,w1,w0);↓(r0,w1,w0); }

March Y	{↕(w0);↑(r0,w1,r1);↓(r1,w0,r0);↕(r0); }

March B	{↕(w0);↑(r0,w1,r1,w0,r0,w1);
	↑(r1,w0,w1);↓(r1,w0,w1,w0);
	↓(r0,w1,w0); }

ANALYSIS & RESULTS:[5]

The summary of the march algorithm and its complexity is given in this table.

MARCH TEST DETECTION & COMPLEXITY:

Faults	AF	SAF	TF	CF	Other	TC
MARCHING 1/0	DS	D	N	N		14n
MATS	D	D	N	N		4n
MATS+	D	D	D	N		5n
MATS++	D	D	D	N		6n
MARCH X	D	D	D	D	Ul-CFin	6n
MARCH C	D	D	D	D	Ul-CFin	11n
MARCH C-	D	D	D	D	Ul-CF	10n
MARCH A	D	D	D	D	l-TF	15n
MARCH Y	D	D	D	D	l-CF	8n

V. BIST:

Built-In Self-Test (BIST), test generation and response evaluation hardware are included on-chip so that in-circuit tests can be performed with minimal need of external test equipment, if any. The standard BIST set-up is shown in figure 1. Under operation conditions, the additional test is transparent to the functionality of the -in self-test (BIST) is common used as a technique to test embedded arrays such as RAMs and ROMs. The primary use of BIST is for manufacturing or production testing but additional features can be added for diagnostics and debug. Conventional memory BIST implementations provide Pass/Fail information, which is usually sufficient for manufacturing test.
There are two types of BIST; On-line and Off-line.

On-line BIST: It has tests implemented on-chip. It has shorter test time. But it has an area overhead of one to three percent.

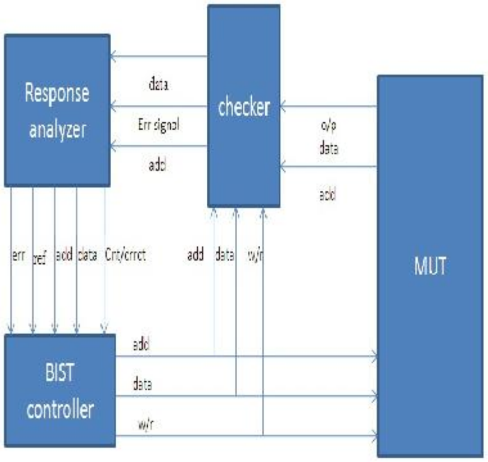
Off-line BIST: It has tests implemented off-chip. It has longer test time but no area overhead
.

Built-in self-test (BIST) has been shown to be one of the most cost-effective and widely used solutions for memory testing for the following reasons:

1.No external test equipment.
2.Reduced development efforts.
3.Tests can run at circuit speed to generate a more realistic test time.
4.On-chip test pattern generation to deliver higher controllability and observability.
5.On-chip response analysis.
6. Test can be on-line or off-line

TEST ARCHITECTURE EXPLANATION:

This architecture consist of BIST Controller,MUT,Checker, and Response Analyser .As Por and Clock signal becomes enable the BIST controller starts working.The BIST controller gives the control signal to the memory.then the memory undergoes read or/write operation according to the March algorithm. Then the output from the memory is given to the checker.The checker compares the output from the memory to the data stored inside it. Whenever the fault occurs the checker gives the error signal, the original data along with the address to the response analyser .



The response analyzer is used to switch the controller from normal to repair mode.Whenever repair mode becomes enable the controller automatically enables the write signal to repair the fault according to the address and data given by the response analyzer. After the repair operation gets completed the ref signal becomes enable in order to indicate that fault is repaired .As the continue signal becomes enable the controller switches to normal mode.This process is continued till the end of the operation.The export mask address signal is used to indicate whether the fault is repairable or not.By this way this test architecture is used to test and repair the fault in memory with maximum accuracy.

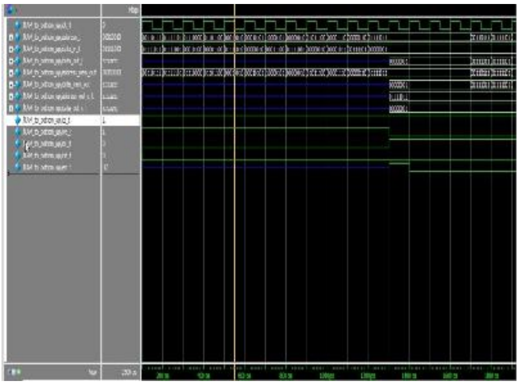


Figure.1-Writing data into the memory and as well as into the checker

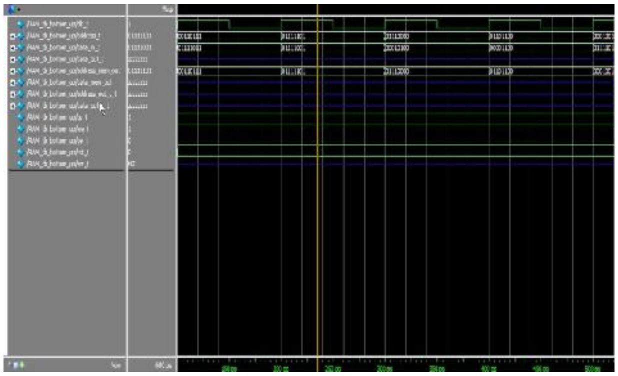


Figure .2- Reading the data from the memory and it is compared inside the checker for error

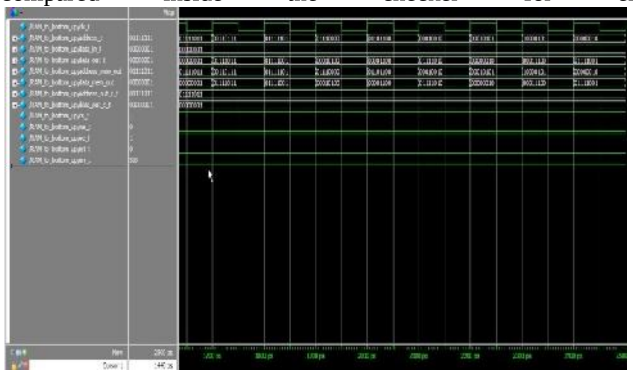


Figure.3- Reading,Writing and comparision of the data inside the checker for detecting the error

CONCLUSION:

The crucial part in testing is how well the test can be completed in minimum time with minimal test length. This can be enhanced by MARCH algorithm.

MARCH tests are extensively being used today for functional testing of memory technologies. They are more efficient with better fault coverage than older classical pattern. In this project modified BIST architecture is used proving to be an effective testing method to test embedded memories as it provides a flexible approach and better fault coverage.

VII. ACKNOWLEDGMENT

We thank the Department of Electronics and Communication Engineering of Kalasalingam University, (Kalasalingam Academy of Research and Education), Tamil Nadu, India for permitting to use the computational facilities available in Centre for Research in Signal Processing and VLSI Design which was setup with the support of the Department of Science and Technology (DST), New Delhi under FIST Program in 2013.

REFERENCES:

[1] Swati Singh , Built-In-Self Test for Embedded Memories by Finite State Machine , International Journal of Digital Application & Contemporary research , September 2013.

[2] Imthiazunnisa Begum M.Tech, A Built-In Self Test By Enhanced Faults Coverage with Microcode Optimization for Embedded Memory, VIFCET, JNTU Hyderabad,july 2013.

[3] NOR AZURA ZAKARIA, W.Z.W. HASSAN, HALIN, M. SIDEK, QING WEN , FAULT DETECTION WITH OPTIMUM MARCH TEST ALGORITHM , E-ISSN: 1817-3195 , University Putra Malaysia , 10thJanuary2013

[4] Dr. R.K. Sharma Aditi Sood, Modeling and Simulation of Microcode-based Built-In Self Test for Multi-Operation Memory Test AlgorithmNational Institute of Technology, Kurukshetra, 2012.

[5] Muddapu Parvathi1, N. Vasantha, K. Satya Parasad,Modified March C - Algorithm for Embedded Memory Testing, J.N.T.U.K, Kakinada, May 2010 .

[6] Fakhraie, M. R. Movahedin, Embedded Test for Processor and Memory Cores in System-on-Chips, University of Texas at Dallas.

[7] Da Wang, Yu Hu, Huawei Li, and Xiaowei Li,The Design-for-Testability Features and Test Implementation of a Giga Hertz General Purpose Microprocessor, University of Chinese Academy of Sciences (GUCAS), Beijing.

[8] MOHAMMED ARIF,Optimization of Microcode Built-In Self Test By Enhanced Faults Coverage for Embedded Memory, Gyan Ganga Institute of Technologyand Sciences Jabalpur, 2012 IEEE Students' Conference on Electrical, Electronics and ComputerScience.