Enhancing Memory Reliability through BIST-Based Fault Diagnosis

Divya Reddy*, Dr. K. Suresh
PG Scholar, Department of ECE, G. Pullaiah College of Engineering and Technology, Kurnool, Andhra Pradesh, India
Assistant Professor, Department of ECE, G. Pullaiah College of Engineering and Technology, Kurnool, Andhra Pradesh, India

ABSTRACT

New memory technologies and processes introduce new defects that cause previously unknown faults. Dynamic faults are among these new faults; they can take place in the absence of the traditional static faults. Further shows based on industrial test results, the importance of such faults for the new memory technologies, and introduce a systematic way for modeling them. It concludes that current and future SRAM products need to consider testability for dynamic faults or leave substantial DPM and it sets a direction for further research. It therefore introduces a new test (March SS), with a test length of 22n that detects all realistic simple static faults and some of the Dynamic Faults in RAMs. This project not only describes detection of faults but also to repair the detected faults, simply evaluates the improvement of RAM reliability in terms of the resistance value of the resistive-open defect and proposes a simple method for enhancing the reliability of static random access memories (SRAMs) with resistive open defects. This method prevents a SRAM from executing successive multiple read operations on the same position, such that the hard-to-detect defects cannot manifest as functional faults. This can prolong the lifetime of the SRAM with latent hard-to-detect defects.

KEYWORDS: Memory Built In Self-Test (MBIST), State Machine Controller (SMC), Reliability Enhancement Circuit (REC), Successive Read Detector (SRD), Defect-per Million (DPM).

INTRODUCTION

Random Logic and embedded memories have become inseparable parts of many system-on-chip solutions. Following a significant increase in the chip area occupied by memory arrays, the International Technology Roadmap for Semiconductors predicts memories to take up more than 90% of the silicon area within the decade. Due to their extremely large scale of integration, memories have already started introducing new yield loss mechanisms at a rate, magnitude, and complexity large enough to demand major changes in test strategies Today's demanding applications require SoCs that are bigger and faster those are more area, timing, and power sensitive than ever before, resulting in a shift from the logic-dominant chips of the past to memory-dominant ones. Figure 1 shows embedded memory projections from Semico Research Corporation. In 2008, embedded memories accounted for more than half of the die area in a typical SoC. It's predicted that the amount of space they occupy on the die will continue to increase, reaching up to 70% by 2017.

These shrinking technologies give rise to new defects and new fault models have to be defined to detect and eliminate these new defects. These new fault models are used to develop new high coverage test and diagnostic algorithms. The greater the fault detection and localization coverage, the higher the repair efficiency; hence higher the obtained yield. Memory repair is the necessary, since just detecting the faults is no longer sufficient for SoC's, hence both diagnosis and repair algorithms are required. Predictions are that this will increase to 70% of the die area by 2017.

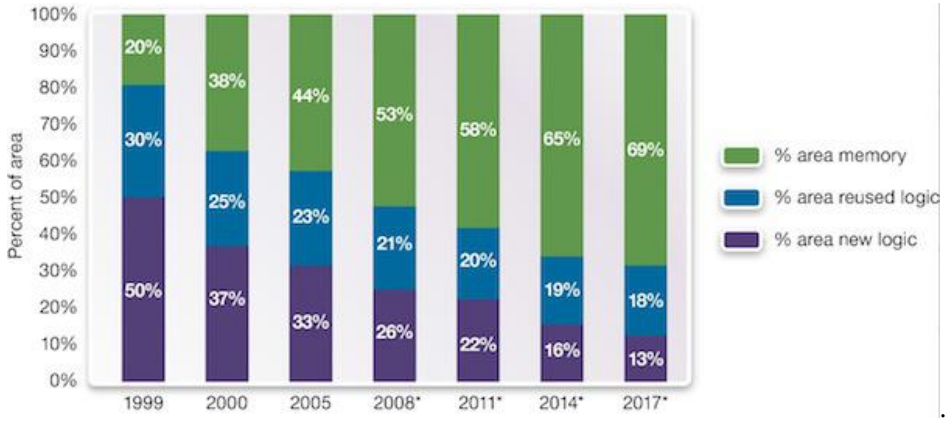

Fig 1. Embedded memories account for half the die area of a typical SoC today. Predictions are that this will increase to 70% of the die area by 2017.

BUILT IN SELF TEST

BIST is a Design-for-Testability (DFT) technique, because it makes the electrical testing of a chip easier, faster, more efficient, and less costly. The concept of BIST is applicable to just about any kind of circuit, so its implementation can vary as widely as the product diversity that it caters to

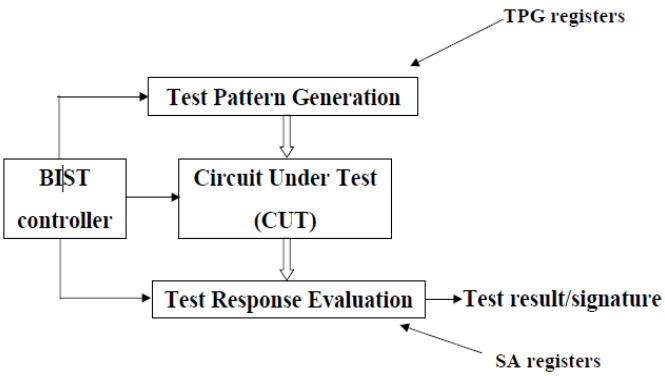

Fig 2.BIST Configuration& Architecture

Memory Built In Self-Test

In order to perform a high quality test, the memory BIST has emerged as a valuable choice due to:

1. Very regular test patterns and expected responses that can be generated, compressed, and stored by using a relatively simple testing circuitry;
2. The reduced number of input/output channels—they usually suffice to control BIST operations such as activation, scan-in, scan-out, and others;
3. on-chip location of the entire test logic; in particular it facilitates at-speed testing allowing detection of time related faults.

A memory BIST controller could be designed as a Finite State Machine (FSM)-based or microcode-based controller. The FSM-based controllers are the hardware realization of a selected memory test algorithm. This type of memory BIST architecture has optimum logic overhead, however, lacks the flexibility to accommodate any changes in the selected memory test algorithm. This results in redesign and re-implementation of the FSM-based controller for any minor changes in the selected memory test algorithm. In microcode-based controllers, a selected memory test algorithm is written in terms of a set of supported instructions and loaded in the memory BIST controller. This type of memory BIST architecture allows changes in the selected test algorithm with no impact on the hardware of the controller. This flexibility might result in higher logic overhead for the memory BIST controller. However, the additional logic overhead, if kept in reasonable levels, could be justified by using the memory BIST to reduce the cost of diagnostics.

The new memory technologies involving high density of shrinking devices lead to newer faults. Some such newly defined fault models are Write Disturb Fault (WDF), Transition Coupling Fault (CFT), Deceptive Read Disturb Coupling Fault (CFDRD).These new faults cannot be easily detected by established tests like March C-, rendering it insufficient/ inadequate for today's and the future high speed memories. More appropriate test algorithms like March SS and March RAW have been developed to deal with these new fault models. While March SS covers some of the new fault models like DRDF, WDF. Thus architectures which have been developed to implement earlier tests like March C- may not be able to easily implement these newer test algorithms. The reason is that most of the newly developed algorithms have up to six or seven (or even more) number of test operations per test element. Thus some of the recently developed architectures that had been specifically designed to implement these older algorithms can only implement up to two March operations per march element, rendering them incapable of easily implementing the new test algorithms.

BISR WITH RELIABILITY

The architecture of the micro based built in self-repair is shown In the Fig 3. It consists of the Instruction Pointer, Instruction storage, Instruction register, Input multiplexer, Address generator, Data generator, RW Control, memory, redundant logic array, output multiplexer, fault diagnosis and SMC controller blocks.

Instruction Pointer is used to point which instruction we have to fetch from the instruction storage. It works for every rising edge of the Clk depending on the enabling signals and Rst values. If Rst is active low InstAddr is reset to zero.

Instruction Storage is used to store the instructions. 22 instructions are used to find the faults in the memory. These instructions are stored in the instruction register. If Rst is active high, for every rising edge of the Clk if InstEna is active high one instruction will be fetched from the memory from the location instruction address pointed by instruction pointer.

Instruction Register is used to store the instruction which is coming out from the instruction storage. The instruction is decoded and given as input to the required modules. It is a 7-bit register.IfRstisactivelowinstructionregistervalueisresettozero.If Rst is active high and for every rising edge of the Clk, if IREna is active high instruction is stored in the instruction register and decoded

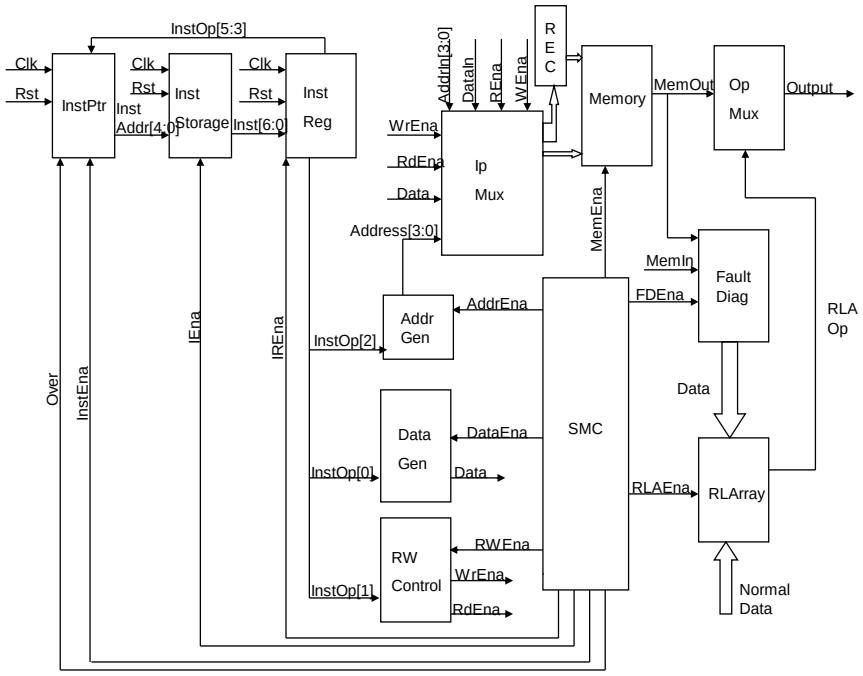

Fig 3.Architecture of BISR with Reliability Enhancement Circuit

AddrGen is used to generate the address. If Rst is active low, address will be reset to zero, otherwise for every rising edge of the Clk, if AddrEna is active high address will be incremented or decremented according to the input signals.

DataGen is used to generate the data, which is given as input to the memory.. If Rst is active low, data will be reset to zero, otherwise for every rising edge of the Clk, if DataEna is active high data will be according to the input signals.

RWControl is used to generate the RdEna and WrEna signals, which are given as inputs to the memory. If Rst is active low, then RdEna and WrEna signals will be reset to zero, otherwise for every rising edge of the Clk, if RWEna is active high then RdEna and WrEna signals will be set according to the input signals.

Input Multiplexer is used to select one group of signals depending on whether it is working in Normal/Test mode. Multiplexer output is given as input to the memory.

Memory is used as a unit under test. If MemEna, WrEna both are active high and RdEna is active low, the data is written into the memory location specified on the address signal. If MemEna, RdEna both are active high and WrEna is active low, the data is from the memory location specified on the address signal.

Fault Diagnosis is used to compare the expected data with the original data. If any change is there, it gives that location address and actual data as input to the Redundant Logic Array.

Redundant Logic Array acts as the redundant memory. In this we will store the memory faulty locations address and data. In Normal Mode it compares normal input address with the existing faulty locations; if it matches it uses redundant logic memory for read and write operations. If it doesn't match it will use the original memory for read

and write operations. The fault pulse acts as an activation signal for programming the array. The redundancy word is divided into three fields. The FA (fault asserted) indicates that a fault has been detected. The address field of a word contains the faulty address, here as the data field is programmed to contain the correct data which is compared with the memory output.

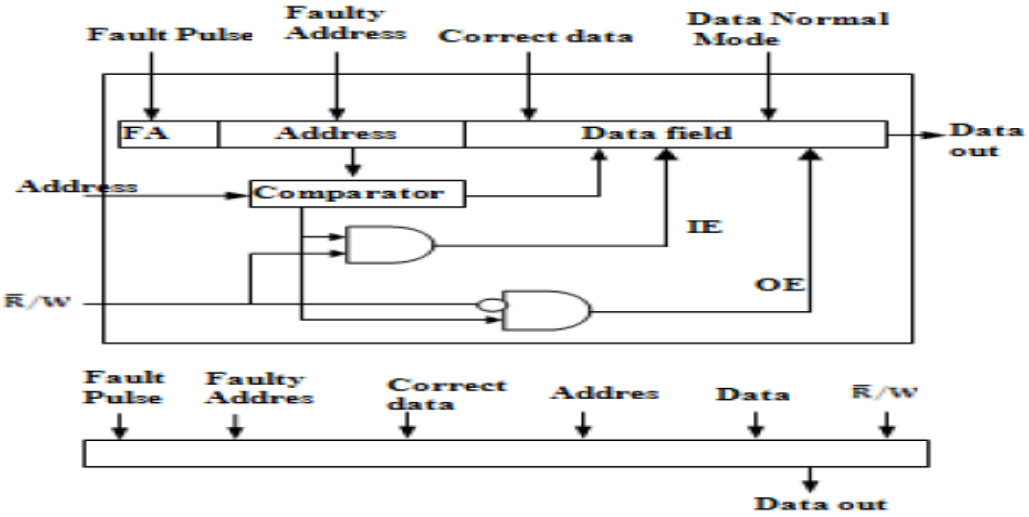

Fig 4.Redundancy word line

The IE and OE signals respectively act as control signals for writing into and reading from the data field of the redundant word. An overflow signal indicates that memory can no longer be repaired if all the redundancy words have been programmed.

Output multiplexer is used to select one value from the Redundant memory and Memory depending whether it is faulty or not. SMC works under 3 different modes namely, idle, test and normal modes. Redundant memory is used to store the faulty location detected by fault diagnosis module. After that it enters into normal mode. The outputs are the Enable signals of all the modules .if SMCEna=1, then depending upon Mode type the respective output Modules is enabled. The microcode is a binary code that consists of a fixed number of bits, each bit specifying a particular data or operation value. As there is no standard in developing a microcode MBIST instruction, the microcode instruction fields can be structured by the designer depending on the test pattern algorithm to be used. The microcode instruction developed in this work is coded to denote one operation in a single micro word. Thus a five operation March element is made up by five micro-code words. The format of 7-bit microcode MBIST instruction word is as shown in Table 1. Its various fields are explained as follows: Bit #1 (=1) indicates a valid microcode instruction, otherwise, it indicates the end of test for BIST Controller. Bits #2, #3 and #4 are used to specify first operation, in-between operation and last operation of a multi-operation March element, interpreted as shown in Table 1. Bit #5 (=1) notifies that the memory under test (MUT) is to be addressed in decreasing order; else it is accessed in increasing order. Bit #6 (=1) indicates that the test pattern data is to be written into the MUT; else, it is retrieved from the memory under test. Bit #7 (=1) signifies that a byte of 1s is to be generated (written to MUT or expected to be read out from the MUT); else byte containing all zeroes are generated. The instruction word is so designed so that it can accommodate any existing or future March algorithm. The contents of Instruction storage unit for March SS algorithm the First march element M0 is a single operation element, which writes zero to all memory cells in any order, whereas the second march element M1 is a multi-operation element, which consists of five operations: i) R0, ii) R0, iii) W0, iv) R1 and v) W1. MUT is addressed in increasing order as each of these five operations is performed on each memory location before moving on to the next location.

Table 1: Format of Microcode Instruction word

#1	#2	#3	#4	#5	#6	#7
Valid	Fo	Io	Lo	I/D	R/W	Data

Fo	Io	Lo	Description
0	0	0	A single operation element
1	0	0	First operation of a Multi-operation element
0	1	0	In-between Operation of a Multi-operation element
0	0	1	Last Operation of a Multi-operation element

March SS Algorithm

The March SS algorithm consists of the following steps:

1. Write 0 to all locations either increasing or decreasing order of address locations
2. Read 0 at lowest address location twice, write 0 at same address and read it and then write 1 into the same location
3. Read 1 at lowest address location twice, write 1 at same address and read it and then write 0 into the same location
4. Read 0 at highest address location twice, write 0 at same address and read it and then write 1 into the same location.
5. Read 1 at highest address location twice, write 1 at same address and read it and then write 0 into the same location.
   Repeat the steps 4 and 5 for all the address locations of the memory in decreasing order.
6. Read 0 from the lowest address to the highest address.

In this type testing occurs simultaneously with normal functional operation. This form of testing is usually accomplished using coding technique or duplication or comparison.

Table 2: Content of Instruction Storage Unit

	#1 Valid	#2 Fo	#3 Io	#4 Lo	#5 I/D (0/1)	#6 R/W (0/1)	#7 Data (0/1)
M0: χ W0	1	0	0	0	0	1	0
M1: ↑{R0	1	0	0	0	0	0	0
R0	1	0	1	0	0	0	0
W0	1	0	1	0	0	1	0
R0	1	0	1	0	0	0	0
W1}	1	0	0	1	0	1	1
M2: ↑{R1	1	1	0	0	0	0	1
R1	1	0	1	0	0	0	1
W1	1	0	1	0	0	1	1
R1	1	0	1	0	0	0	1
W0}	1	0	0	1	0	1	0
M3: ↓{R0	1	1	0	0	1	0	0
R0	1	0	1	0	1	0	0
W0	1	0	1	0	1	1	0
R0	1	0	1	0	1	0	0
W1}	1	0	0	1	1	1	1

M4: ↓{R1	1	1	0	0	1	0	1
R1	1	0	1	0	1	0	1
W1	1	0	1	0	1	1	1
R1	1	0	1	0	1	0	1
W0}	1	0	0	1	1	1	0
M5: χ R0	1	0	0	0	1	0	0
	0	X	X	X	X	X	X

Reliability

Simple approach to prolong the lifetime of a RAM cell with this type of latent defects is to prevent the RAM cell from successive multiple read operations. Thus, if no successive multiple read operations on a RAM cell is allowed, then the larger defect size can be tolerated for the RAM cell. For example, consider a SRAM cell with 25fF bit-line load is operated at 1.4ns. Assume that no successive multiple read operations on the SRAM cell are allowed. Then the SRAM cell still can work correctly when the defect size is smaller than 1.13M. However, if successive multiple read operations on the SRAM cell are allowed, then the SRAM cell only can work correctly when the defect size is smaller than 0.979M. Therefore, the time of resistance degradation from 0.979M to 1.13M is the prolonged lifetime of the SRAM.

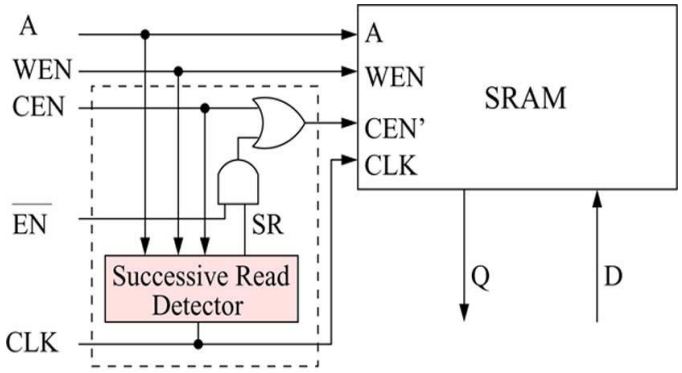

Fig 5.Reliability Enhancement Circuit

SRD (Successive Read Detector)

The SRD checks whether successive multiple read operations on the same position are executed. The enable signal (EN) determines if the SRAM is protected by the REC. If EN is logic1, then the successive read (SR) output is blocked. So, CEN' is equal to CEN. If EN is logic 0, then CEN'=CEN|SR, where | denotes a bit-wise OR operation. Therefore, if SR=1 then CEN' is logic1 regardless of the logic value of CEN and the SRAM is in idle state.

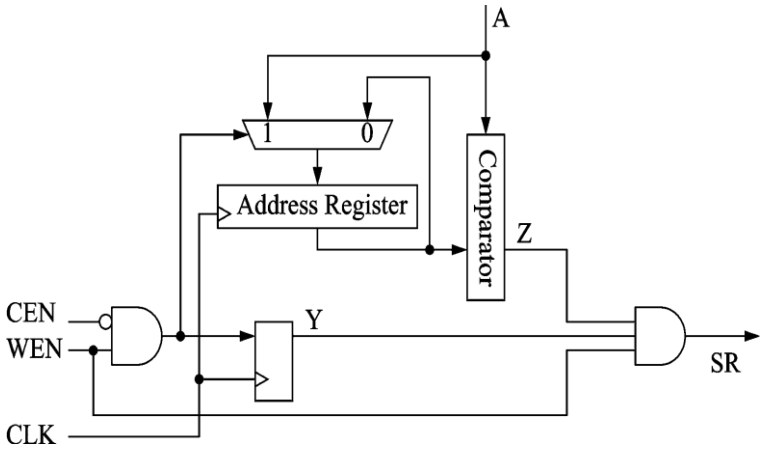

Fig 6.Detail Implementation of the Successive Read Detector

Resistive-Open Defects

Fault behavior of a dynamic fault caused by a resistive-open defect may vary with the defect size (the resistance value of the defect).For example, if a resistive-open defect existing between the Vdd and the pull-up transistor of a 6TSRAM cell, then the defect may cause a data retention fault (DRF) or a DRDF (also called dynamic destructive read fault). The testing of DRF in SRAM is difficult. To cope with this problem, some efficient design-for-testability techniques were proposed.

If a resistive-open defect in the pull-up transistor of a SRAM cell causes a DRDF, then it is detectable by a Read after Write operation. But if the size of the resistive-open defect is small, the n multiple Read operations after one Write operation are needed to detect it. This fault is regarded as a deceptive multiple read destructive fault(DMRDF).The Sensitizing sequence for DMRDF is 1w0(r0)k or 0w1(r1)k ,which means that k Read operations after one Write operation are executed.

SIMULATION AND SCHEMATIC BEHAVIOR

Verilog HDL is used to write the code for the above architecture and synthesized using Xilinx ISE 12.1 tool.

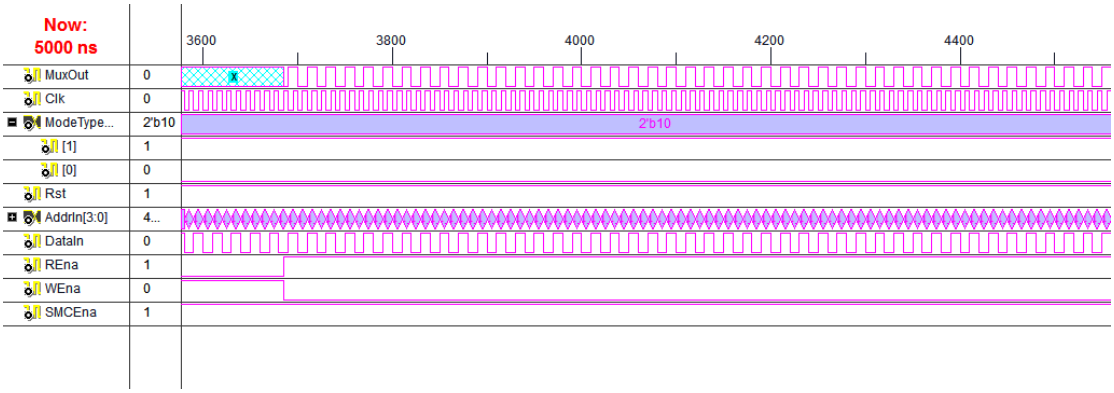

Fig 7.Simulation result for BISR module

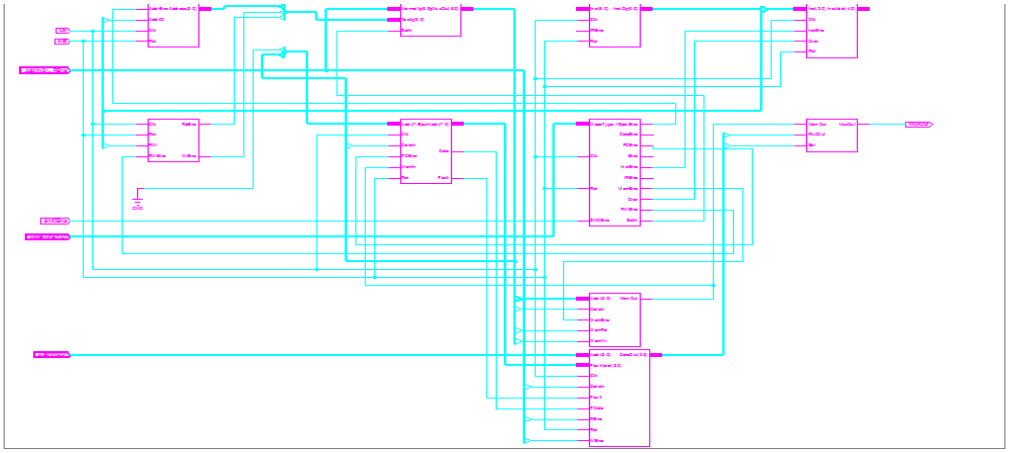

Fig 8.RTL schematic for BISR module

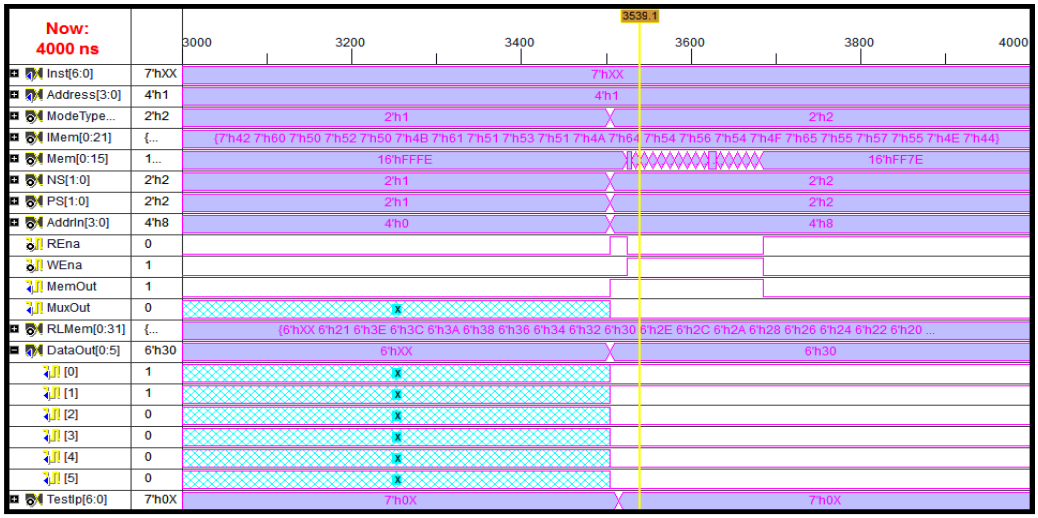

Fig 7.Simulation result for BISR module with reliability

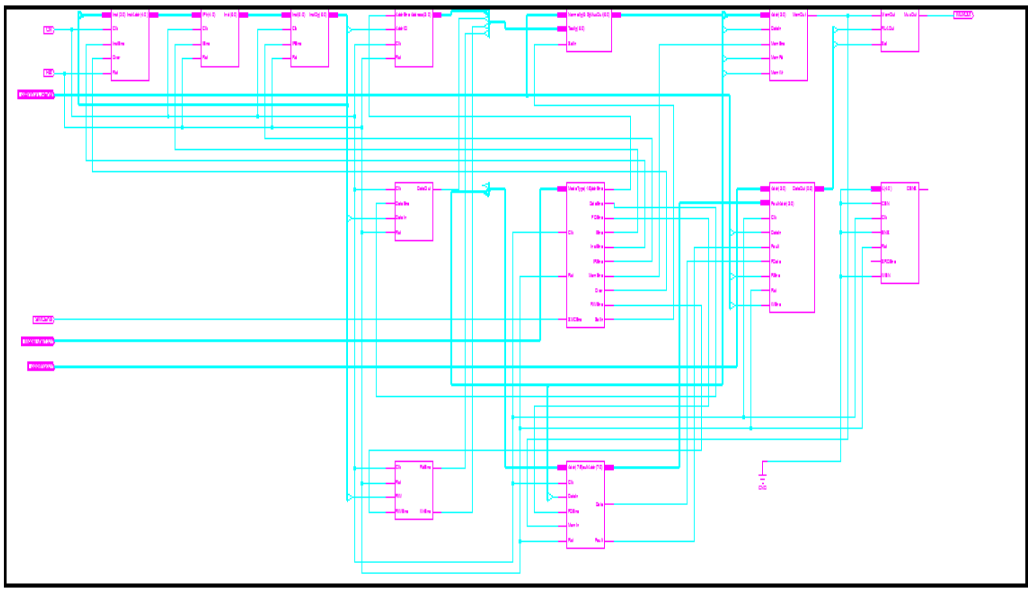

Fig 8.RTL schematic for BISR module with Reliability

CONCLUSION

The Simulated Waveforms shows that microcode-based MBIST architecture is an effective testing method to test Memories as it offers flexible and better fault coverage. Testing can be done with different algorithms without the need to redesign the whole circuit. Only the content of the instruction register need to be modified when designing with other test algorithms. Therefore design time and verification process can be reduced.

The Reliability concept is used to increase the life time of the RAM and March SS algorithm is used for detecting the faults and Redundant logic array for the Repairing the fault memory locations. As the number of operations on memory is reduced with the reliability enhancement circuit, the Life time will be increased

REFERENCES

[1] International SEMATECH, “International Technology Roadmap for Semiconductors (ITRS): Edition 2001”

[2] S. Hamdioui, G.N. Gaydadjiev, A.J .van de Goor, “State-of-art and Future Trends in Testing Embedded Memories”, International Workshop on Memory Technology, Design and Testing (MTDT’04), 2004.

[3]] S. Hamdioui, Z. Al-Ars, A.J. van de Goor, “Testing Static and Dynamic Faults in Random Access Memories”, In Proc. of IEEE VLSI Test Symposium, pp. 395-400, 2002

[4] S. Hamdioui, et. al, “Importance of Dynamic Faults for New SRAM Technologies”, In IEEE Proc. Of European Test Workshop, pp. 29-34, 2003.

[5] S. Hamdioui, A.J. van de Goor and M. Rodgers, ”March SS: A Test for All Static Simple RAM Faults”, In Proc. of IEEE International Workshop on Memory Technology, Design, and Testing, pp. 95-100, Bendor, France, 2002.
[6] N. Z. Haron, S.A.M. Junos, A.S.A. Aziz, “Modelling and Simulation of Microcode Built-In Self-test Architecture for Embedded Memories”, In Proc. of IEEE International Symposium on Communications and Information Technologies pp. 136-139, 2007.
[7] R. Dean Adams, “High Performance Memory Testing: Design Principles, Fault Modeling and Self-Test”, Springer US, 2003.
[8] . Ferguson, J., and J. Shen, “Extraction and simulation of realistic CMOS faults using inductive fault analysis,” Proc. Intl. Test Conf., 475−84 (1988). doi:10.1109/TEST.1988.207759.

[9] Malaiya, Y. K., and S. Y. H. Su, “A survey of methods for intermittent fault analysis,” Proc. Nut Comput Conf., 577−84 (1979).
[10] an introduction to logic circuit testing by parag k. Lala(morgan & claypool publishers).Synthesis lectures on digital circuits and systems
[11] Dear, I.D. et al., ‘‘Economic effects in design and test,’’ IEEE Design & Test of Computers, December 1991, pp. 64–77.