A Survey on Built in Self-Test and Repair Analyzer for Embedded Memories

Jyothi M
Department of Electronics and Communication Engineering
CMR Institute of Technology, Bangalore, India

Abstract— This paper presents a survey of different algorithms to test and repair a memory. This paper explains about different built in self-test and built in self repair analysis of embedded memories. The improvement in the yield of memories plays an important role in SOC. The use of spare rows and spare columns with redundancy allocation is proven to be more promising since it gives an optimal repair rate in a single test.

Key words: BISR, SOC technology, memory

I. Introduction

In recent days, much advancement has made to put a millions of transistors on a single die in a VLSI manufacturing technology. In the SOC technology, for the proper and successful implementation of a system the embedded memories should be of high density and high capacity. In modern SOCs, the half of the total chip Area is occupied by these memories and the ratio of these embedded memories may keep on increasing in the future years. If any defect found in a memory that will affect the entire chip design and the whole chip must be discarded. This advancements leads to an integration of many components into a single chip. System on Chip (SOC) is defined as an IC that integrates the major functional elements of a end product into a single chip. This SOC enables the designer to move everything from board to chip. Memories are the widely used cores in SOC. The overall yield of SOC is dominated by the yield of memory, so optimizing the yield of memories plays a crucial role in SOC environment.

Before the start-up of the circuit, testing of the circuit has been done every time is called a Built in Self Test (BIST) and then repairing of the circuit has been done using Built in Self Repair (BISR) technique to make the memory a fault free one.

The review of various techniques of BISR is as follows. The use of spare rows and spare columns, and also the total area of the system is dominated by number of Content Addressable Memory (CAM) entries to store faulty addresses [1]. But here the use of CAM is of cost disadvantage. For repairing RAM with different sizes and redundancy allocations, the Reconfigurable BISR (ReBISR) scheme is used [2]. However in this the use of repair register causes a memory overhead. In [3], the authors have proposed a technique of using bit swapping linear feedback shift register (BS-LFSR), this technique is mostly used to reduce the test time.

BISR is a technique used to repair the faulty addresses in a memory and this method is used in many of the applications in SOC environment. The BISR technique is the most promising and popular one for the repairing of memories.

This paper is a survey about different techniques of BISR. This paper is organized as follows. This paper is organized as follows. In section II, literature review about different algorithms based on BISR is described. In section III, conclusion of the above survey is given. In section IV, consists of tabulation of the above survey.

II. Literature review

A. Built In Self Repair Using Content Addressable Memory

An overview on built in self repair using block level replacement technique for content addressable memories [1] is presented here. This is an on chip infrastructure for bit oriented memories and in later sections it is implemented for word oriented memories also. KUO-FUCHS ALGORITHM framework is adopted for BIST and BISR contains an must repair analysis (MRA) and final analysis. Kuo and Fuchs proposed a kind of Branch-and-Bound algorithm that is based on a binary search tree. The must repair rows and columns is indentified by must repair analysis and the repair solution is searched by final analysis. This MRA has fault list that consists of pair of CAM for fault addresses and a pair of CAM of solution record to repair. If any fault is detected by a BIST engine it is reported to an MRA and then those affected rows and columns are compared against number of CAM entries and it is counted by parallel counter. Then the MRA writes the row and column address into the solution record. The SOLVER generates the first repair strategy and MRA reads each fault address in the fault-list in order until there exists no more fault address.

B. Reconfigurable Scheme (Rebisr) For Repairing Rams

The reconfigurable scheme of BISR [2] uses different sizes of RAMs and also an redundancy organization. A suitable and efficient redundancy algorithm is proposed for defective RAMs. Yield of memories can also be recovered by the use of redundancies. When failures are identified and located during testing, a redundancy analysis procedure determines which failures can be repaired using redundant rows and columns. If the BIST circuit detects a fault, then the fault information is given to the ReBIRA circuit, and then the ReBIRA performs redundancy allocation using the rules of the redundancy algorithm. The redundancy algorithm implemented in this paper is RANGE CHECKING FIRST ALGORITHM (RCFA). Here the repair registers are used for storing defective rows and columns addresses, once it writes a single high, it ensures that there are no faults.

C. Syndrome Storing Based Detection (SSD) Technique To Reduce More Power Consumption

In this paper [3], the faults in the memory are detected using a SYNDROME STORING BASED DETECTION (SSD) method that involves the consecutive code syndromes at the receiver. In the test mode the power consumption is more compare to normal mode. In order to reduce power, transitions of bits in pattern generator are reduced by use of Bit swapping linear feedback shift register (BS-LFSR) leading to power reduction. The BS-LFSR reduces the average and instantaneous switching activity during test operation by reducing the number of transitions in the scan input of the circuit under test.

D. Comprehensive Real-Time Exhaustive Search Test (CRESTA) Algorithm

In this paper [6], the CRESTA algorithm tells by taking an example, that there are m spare rows and n spare columns in a memory. Then a CRESTA repair analyzer contains C(m+n, m) sub-analyzers. Each sub-analyzer analyzes incoming row/column addresses of faulty memory cells in parallel in a different repair strategy. Because CRESTA tries all the possible repair strategies of spare resources, it guarantees finding a solution for a repairable memory. Since CRESTA needs row address and column address of a faulty memory cell in order to check if the current faulty memory cell can be repaired by previously allocated spare resources, it is unable to handle at-speed multiple-bit failure occurring in a word-oriented memory and to determine the number of extra columns required for all fault bits in a word cannot be found in one cycle.

III. CONCLUSION

In this paper, different built in repair techniques for memories are as seen in [1], [2], [3], [4], [5] and are over viewed. The memory BISR based approaches are used for repairing of embedded memories and to improve the yield and efficiency of the memories these techniques are used. The algorithms concentrating here are mainly based to reduce power, cost, test time and area. Refer below Table for the summary of various algorithms in system on chip (SOC) proposed for memories by different authors.

ACKNOWLEDGMENT

The author would thank the referees for their suggestions to improve the content of this paper. The authors would thank the Principal for providing necessary facilities to carry out the work and Guide for guiding towards the implementation.

REFERENCES

[1] Jaeyong Chung, Member, Joonsung Park, Member, and Jacob A. Abraham, Fellow, IEEE "A Built-In Repair Analyzer With Optimal Repair Rate for Word-Oriented Memories" vol 21,no 2, 2013.

[2] ReBISR: A Reconfigurable Built-In Self-Repair technique for Random Access Memories in SOCs Tsu-Wei Tseng, Jin-Fu Li, Member, IEEE, and Chih-Chiang Hsu, no 2, 2012

[3] Optimization and Power Reduction of Built-In Repair Analyzer for Memories S.Jeevith1, T.Angala Parameshwari, R.Yamini, IJAREEIE, vol 3, issue 4, 2013

[4] T. Kawagoe, J. Ohtani, M. Niiro, and T. Ooishi, "A built-in self-repair analyzer (CRESTA) for embedded rams," in Proc. Int. Test Conf., 2000, pp. 567–574.

[5] P. Oehler, S. Hellebrand, and H.-H. Wunderlich, "An integrated built-in test and repair approach for memories with 2D redundancy," in Proc. Eur. Test Symp., 2007, pp. 91–96.

[6] Built-In Self-Repair Techniques for Content Addressable Memories ,Guan-Quan Lin, Zhen-Yu Wang, and Shyue-Kung Lu, 2009

[7] C.-T. Huang, C.-F. Wu, J.-F. Li, and C.-W. Wu, "Built-in redundancy analysis formemory yield improvement," IEEE Trans. Reliab., vol. 52, no. 4, pp. 386–399, Dec. 2003.

IV. TABLE : SUMMARY OF VARIOUS BISR ALGORITHMS

TITLE	AUTHOR	ALGORITHM	POWER CONSUMPTION(mw)	TIME DELAY (ns)
Built in self repair technique using content addressable memories	Jaeyong Chung	kuo-fuchs algorithm	373	8.31
Reconfigurable scheme (ReBISR) for repairing RAMs	Tsu-Wei Tseng	Range-checking first algorithm	220	5.3
Syndrome storing based detection(SSD) technique to reduce more-power consumption	S.Jeevith	syndrome storing based detection	113	2.9
A built-in self-repair analyzer (CRESTA) for embedded rams	T. Kawagoe	Comprehensive real-time exhaustive search test(CRESTA) algorithm	370	8.3