Symmetric Transparent Online Bist and Repair of Memories

L. Haribhanu Priya, Prasad Janga, K. Niranjan Reddy

Abstract— During regular testing Symmetric transparent BIST and Repair programme of RAM modules satisfy the memory contents conservation during similar time bouncing and signature prediction part is needed in transparent BIST programme which achieves considerable limiting in test time. In this study adders incorporated with binary addition is suggested for the utilization of accumulator modules. A symmetric Transparency march c primarily based algorithmic rule for constitutional self-repair (BISR) programme is recommended for multiple embedded reminiscences to realize best purpose of the performance of BISR for multiple embedded memories.

I. INTRODUCTION

Algorithm A March is composed of n march parts, denoted by Mi, with $0 \leq i < n$, all march part consists of zero (or more) write operations, denoted by w0 / w1 (written as 0 / 1 in RAM cell) and nil (or more) read operations denoted by r0/r1, (read as cell 0 / 1 from memory). Figure 1 shows the C-algorithm which consists of six elements of march i.e from M0 to M5, where ⇑ represents associate degree increasing addressing order (Any arbitrary order of addressing) and refers to a declining order of addressing (reverse order of addressing).

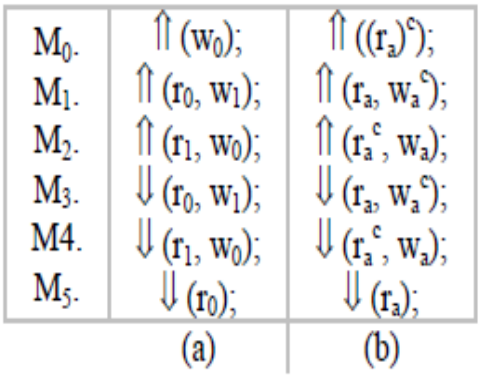


Fig. 1. C- march algorithm (a) Version-original (b) Version-symmetric transparent

In transparent BIST signature prediction part is used instead of w0 part which consists of all read operations identical to march algorithm operations used to compute a signature which can be differentiated with compacted signature studied through the march test.

Transparent BIST has a drawback in signature prediction portion that sum-up the whole testing duration with a share of (greater than) half-hour. The overcome the drawback an idea of symmetric transparent BIST algorithm is introduced [6-7].

Symmetrical transparent algorithm is derived and the march series is altered for the equivalent value of anticipated output reaction [8]. Figure 1(b) shows the C algorithm (symmetric transparent form)

The symbols for algorithms ' transparent variants varies with respect to standard march algorithms. Notations of r0, r1, w0, w1is replaced by ra, rac, wa, wac and (ra) c and their significance is presented in the below table.

Notation	Meaning
ra	RAM word studies the opening contents.
rac	RAM word studies the complement of the opening contents.
(ra)c	feed the complement value to the compactor from RAM
wa	Write to the memory word that is stored at the beginning of test.
wac	Write to the memory term; the opposite of the value saved at the start of the test during this memory term is written to th e term (assumed to be).

II. PERFORMANCE OF BISR

The execution of BISR is assessed by BIST and BIRA. The norms used in BIST performance are test time, fault coverage, and area overhead.

Three major norms for BIRA are a) repair rate, b) repair time and c) area overhead. Due to the proportion of the quantity of restored memories t o the quantity of defective memories, the repair rate is descri bed. Low repair rate predicts a decrease in output due to fail ure to repair repairable memories. It is essential to optimize this rate because the complexity of the RA algorithm for memory fix utilizing 2-D save cells is non-polynomial-complete [19] and a comprehensive hunt is utilized to accomplish an ideal fix rate.

Revised Manuscript Received on September 10, 2019.

Ms. L. Haribhanu Priya, PG Scholar, Department of ECE, CMR Institute of Technology, Hyderabad, Telengana, India.
(Email: bhanu.lellela@gmail.com)
Dr. Prasad Janga, Professor, Department of ECE, CMR Institute of Technology, Hyderabad. Telengana, India.
(Email: prasadjanga85@gmail.com)
Mr. K. Niranjan Reddy, HOD & Professor, Department of ECE, CMR Institute of Technology, Hyderabad, Telengana, India.
(Email: ecehod.cmrit@gmail.com)

The repair time rely on count of faults, the number of spare reminiscences, and also the RA algorithm. as a result of the {number| the amount| the quantity} of deficiencies and extra recollections are much lower than the measure of memory cells, BIST time commands testing and fix time, despite the fact that BIST time rules testing and fix time, a few preliminaries have endeavored to lessen fix time utilizing the best RA calculation.

A module of BIRA requires more space than a module of BIST. Most of the BIRA module is occupied by storage parts such as bitmaps, registers, and content-addressable memories (CAMs) [20]. Therefore, by decreasing the storage parts, overhead area of BISR is to be reduced.

All flaw data is put away to accomplish an ideal fix rate utilizing a careful hunt strategy, be that as it may, on the grounds that there is an exchange off between these choices, it is hard to improve each of the three alternatives at a comparative time.

2.1 Test and Repair procedure for multiple memories

Three strategies for numerous installed recollections are a) Parallel test and parallel fix technique, b) Parallel test and sequential fix strategy and c) Serial test and sequential fix strategy.

The parallel test and parallel fix system needs n BIST modules and n BIRA modules any place there are n recollections. The BIST and BIRA modules are corresponding to the measure of implanted recollections, in light of the fact that each memory has committed BIST and BIRA modules. Every BISR module simultaneously tests and repairs all the memories, the region overhead is the biggest of the BISR techniques. However, due to the simultaneous test and repair method, this technique provides the shortest test and repair time among the BISR techniques.

The parallel testing and sequential fix strategy requires n BIST modules and one BIRA module, after every one of the recollections are tried at the same time, the fix is performed sequentially, in spite of the fact that there is only one BIRA module for the sequential fix approach, the { number amount } of the issue stockpiling parts is should have been equivalent to the quantity of recollections because of the flaws be distinguished from

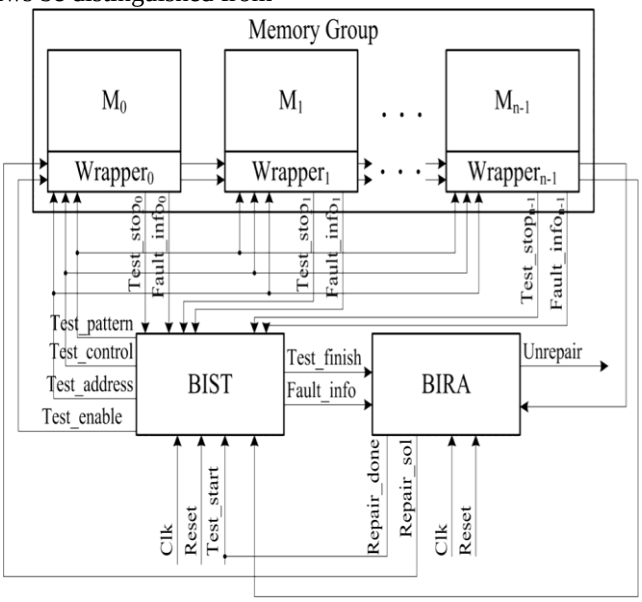


Fig. 2.Diagram of the planned BISR design.

alternative memories at the same time, they must be kept individually. The repair solutions can not be accurately obtained by the fault info is mixed without differentiating the memory identification. The faults are stored individually for each memory to prevent this mistake.

The serial test and serial repair method test and repair the memories. It requires space overhead token, as a consequence of various reminiscences having one BISR. Because of the sequential methods, the check and fix time is nearly as long as the BIST time is because of the check and fix time, it is important to diminish the BIST time.

The scheduled BISR theme is based on two phases of a check and repair method to decrease overhead room and check and repair time. Only defective reminiscences will be evaluated serially and the reminiscences will be evaluated at the same moment.

2.2 BISR scheme

Since there are a few recollections inside the SoC, the recollections are part into gatherings that consider steering, timing, and power utilization. Each gathering has the recommended circuit of the BISR. The quantity of required BISR circuits, in various words, is equivalent to the quantity of memory bunches inside the SoC. At that point, in each gathering, the proposed BISR targets different recollections.

All recollections are assessed and sorted by a parallel test activity as imperfect memory or deficiency free memory. Because of the way that it isn't important to fix the issue free recollections, just defective recollections are chosen and sequentially tried and fixed by the memory scale in diminishing request.

Fig. 2 Displays the recommended BISR configuration outline. It comprises fundamentally of BIST and BIRA modules, the primary reason for existing is to order the recollections as defective or not, and furthermore to store the quantity of deficiencies in every memory inside the wrapper.

At the point when BIST distinguishes the shortcoming, the issue data is moved to BIRA through the Fault information port. When the test is finished or ceased, the finish of the sign test is actuated and BIRA plays out the RA strategy to perform answers for fix.

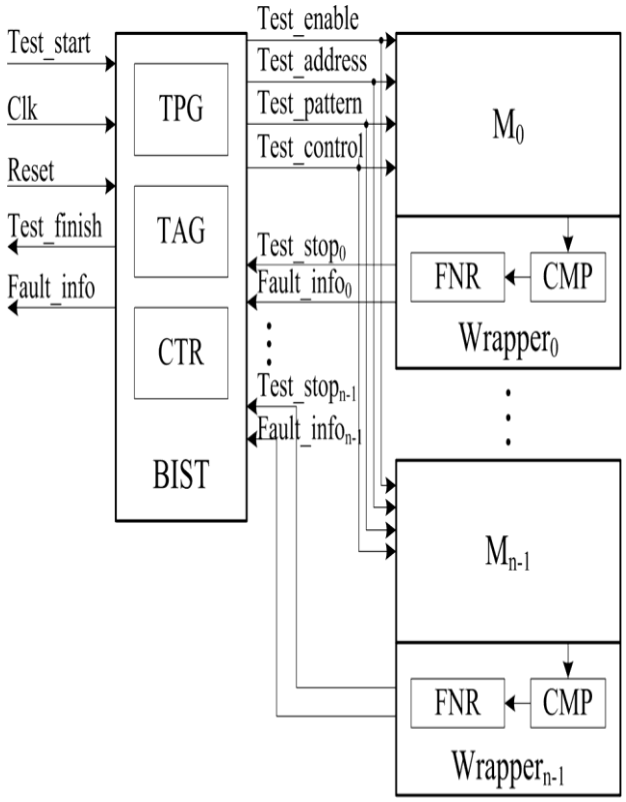


Fig. 3. Block diagram of the BIST and wrapper modules.

2.3 Symmetric transparent BIST with accumulators using carry adders:

Consider a 3-stage accumulator consisting of a binary adder as shown in Figure 4(a). The adder's ' B ' input is guided by the accumulator's register (R) output. The information output of the RAM drives the ' A ' input as shown in Figure 4(c). The adder's performance is pushed into a flip flop's inputs whose output drives the accumulator's low-stage carry input.

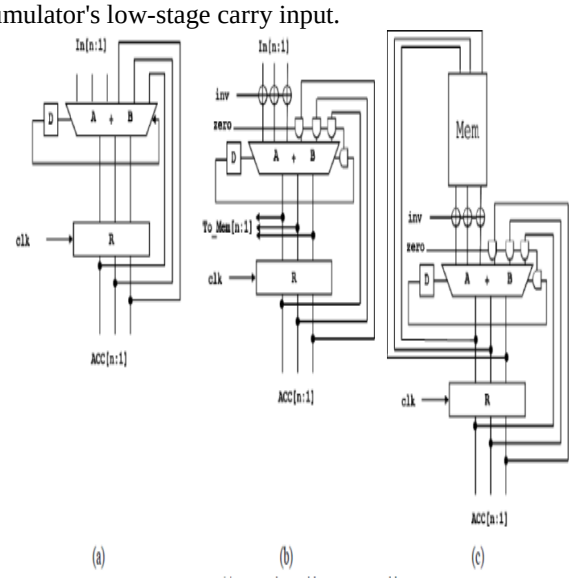


Figure 4: (a) Accumulator with Rotate carry adder (b) proposed plot for compaction of the memory yields (c) complete engineering

The suggested system is granted in Figure 4(b). A series of XOR gates at the adder's' A' input outputs will invert the' A' input calculation on the' zero' signal value; the' B' inputs are guided by a sequence of AND doors gated by the' zero' signal; zero is also pushed to the adder's carry button.

Figure 4(b) system procedure is provided in Table 1 for the execution of the different march activities. In Table 1, the inputs (inv = 1) and (since zero = 1) are added to the ' B ' input, i.e. the previous value of R, to implement operation (ra)c. In v is disabled in order to execute the ra procedure and therefore the memory word content is added to the R value. Similarly, in order to execute wac (this way clk is disabled), the term is reversed and thus added to the all-zero vector (because zero = 0).

March operation	Clk	Inv	Zero	A	Input to Mem
(ra)c	En	1	1	(M[A])c	
ra	En	0	1	(M[A])	
wac	Dis	1	0	(M[A])c	(M[A])c
rac	En	0	1	M[A]	
wa	Dis	1	0	M[A]	M[A]

Table 1: Operation of the proposed design

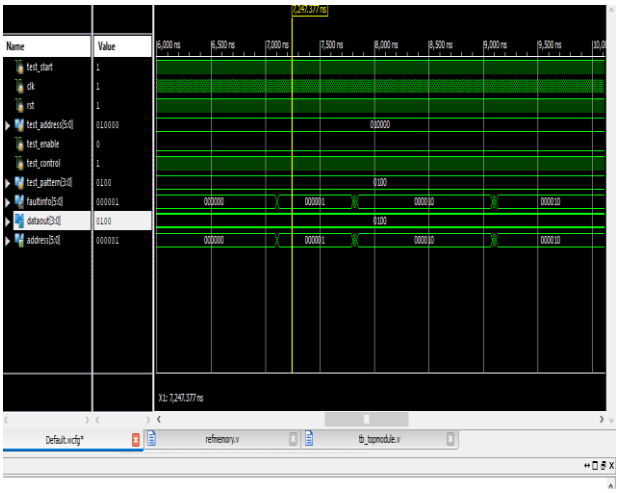
For the description of the suggested system, we shall indicate with n the amount of phases of the ALU that can be added and with k the amount of pieces of the RAM phrase (thus k < n). Suggested system objective is to register content beyond examfor a definite point (i.e. 11... 1). So as to guarantee that the ALU's (n-k) high-request information sources are roughly determined by an every one of the 1 or each of the—0 esteem, it ought to be noticed that if the walk calculation is symmetrical, the information sources driven by the reaction verifier and the information generator are integral during back to back walk components. To broaden this for situations where the memory width is not exactly the quantity of phases of the ALU

In Figure 4 the scenario of 3-bit memory with a 5-stage ALU is tested transparently is shown. The signal material drives the ALU's 5-3=2 high-order inputs, here we demonstrate the module procedure of the figure for the case where the RAM has 4 words. We suppose the original RAM word content is{ 010,111,011,100}.

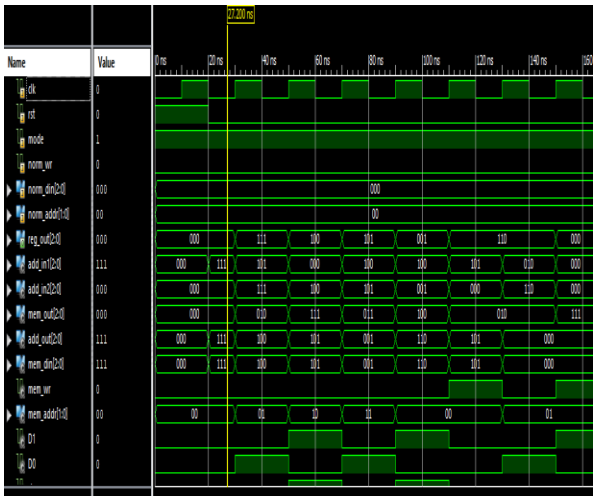
Transparent online BISR for an array of RAM Modules:

Past debate sub section are used to test more than one memory modules in a transparent manner with distinct word widths. We exemplify the concept in the figure where three RAM modules, each with phrases of 3, 4 and 5 bits, tested online by 5-stage ALU. The RAM to be tested is allowed respectively by selecting the CS1, CS2 or CS3 chip signal. When RAM 1 is tested, the ALU inputs are driven by the RAM1 outputs; when RAM2 is tested, the high-order ALU input is driven by things 2; when RAM 3 is tested, both high-order inputs are powered by things 1 and things 2, as shown in figure 2.

III. RESULTS



Simulation Results of Existing Method



Simulation Results of Proposed Symmetric Transparent Online BISR

IV. CONCLUSION

The suggested strategy utilizes the likelihood of distribution of spares based on the amount of spares in the sparse defective row. Too much overhead hardware is required for the earlier BIST methods, which have an ideal repair rate. It can be considered a waste because it is necessary to save all faults in order to obtain an ideal rate of repair.

Also, as the amount of faults rises, the past heuristic BIST methods demonstrate the fast reduction in repair rate. So the suggested BIST utilizes the probability approach to use a partial exhaustive search. It has both exhaustive search and heuristic spare allocation features. Experimental findings indicate that the suggested BIST has an almost ideal overhead repair rate and a tiny overhead hardware. Therefore, supporting the integrated memory in system-on-chip can be a suitable answer.

V. REFERENCES

1. "Flexible Architecture of Memory BISTs"ReinaldoSilveira, QadeerQureshi and Rodrigo Zeli Memory Test and Repair Team at NXP Semiconductors Corp. Campinas, Brazil and Austin, USA (emails: reinaldo.silveira@nxp.com, qadeer.qureshi@nxp.com, rodrigo.zeli@nxp.com)
2. AJ. van de Goor, "Using March Tests to Test SRAMs", IEEE Design and Test of Computers, 1993.
3. S. Hamdioui, J. E. Q. D. Reyes", Newdatabackground sequences and their industrial evaluation for word-oriented random-access memories", IEEE Trans. on CAD of Integrated Circuits and Systems 24(6): 892-904 (2005).
4. S. Hamdioui, Z. Al-Ars, Ad J. van de Goor, "Opens and Delay Faults in CMOS RAM Address Decoders", IEEE Transactions on Computers, vol. 55, no. 12, December 2006, pp. 1630-1639.
5. S. Hamdioui, R. Wadsworth, J. D. Reyes, A. J. van de Goor, "Memory Fault Modeling Trends: A Case Study". J. Electronic Testing 20(3): 245-255 (2004)