EFFICIENT MEMORY BUILT - IN SELF TEST FOR EMBEDDED SRAM USING PA ALGORITHM

G.PRAKASH[#1], S.SARAVANAN[#2]
[#1] M.Tech, School of Computing
[#2] Assistant Professor, SASTRA University, Thanjavur.
[#1]Prakashganesh.be@gmail.com, [#2] saran@core.sastra.edu

***Abstract*-Memory-Built In Self-Test (MBIST) is an very effectual and output enrichment for embedded RAMs. This paper presents effectual MBIST concepts of Built-In-Self Test (BIST) using Performance Accelerator Algorithm (PAA). This BIST concept very stretchable for embedded RAMs with suitable operation. PA algorithm efficiently detects probable number of fault models compare to other March test algorithms. This algorithm has been synthesized and implemented in Xilinx Virtex-V (XC5VLX50). The implementation results are tabulated for March C and PA algorithm.**

Keywords-MBIST, RAM, Fault Models, March algorithm

I. INTRODUCTION

In many number of applications the predefined blocks or models of complex functions are used as a core which are incorporated typically for System On Chips (SOC). The major advantage of FPGA lies in reconfigurability which plays a prominent role in system on chips [2]. Several FGPA has different vendors each vendor has sophisticated memories. Memories play a prominent role in SOCs and FPGAs. Currently, the area engaged by memories which are embedded is more than 90.0%, and estimated to increase up to 95% by 2014 [3]. Performance and output will lead chip technology in the case of embedded memories. However, memory production output is restricted more by random defects, processing over the gross and construct faults, processing for specific faults other defects and faults. To increase the consistency and output of memories, many algorithms and mechanism [4]. In both prolixity columns and rows are integrated into the array of memory. By using Prolixity mechanisms it is difficult to embedded memories and it leads to increase in area of design. This paper presents PA algorithm which is used for testing of BIST.

Fault models make out the target faults limits the range of the test generation and makes success measurable by experiment. Applying the condensed functional model of different RAM faults can be classified as follow. The details of various fault models refer to [1] and time is more consideration of covering the faults in memory test this are the basic phenomena. BIST has capability for testing memories with various test algorithms, like MATS (Modified Algorithmic Test Sequence), MSCAN, MARCH but most common and prevailing algorithm were used in BIST is March test algorithm. March test has benefits of more fault coverage efficient time consumption [5]. There are many March test algorithms were proposed in various papers [6-8], like March A, and so on. One of the basic algorithms for MBIST is March Y. It has 8n (number of operations), The steps involved are, here WD and RD denotes Write and Read operation respectively,

Down – w0

Up – r0, w1, w1

Down – r1, w0, r0

Up – r0

Transition Faults (TF), Stuck At Faults (SAF), Address Fault (AF) can be detected by March Y. Similarly, the March C- has 10n (number of operations). The steps involved are,

Up – w0

Up –r0, w1

Up – r1, w0

Down – r0, w1

Down – r1, w0

Down – r0

The following faults can be identified, Address Decoder Fault (AF), Stuck At Faults (SAF), Transition Faults (TF), Idempotent Coupling Faults (CFid), State Coupling Faults (Cfst) by March C. In above steps of algorithm ‘Up’ represents ascending order execution of SRAM addresses and ‘Down’ descending order

execution. Mentioned algorithms are not effective over fault detection. This paper presents PA algorithm which covers furthermost fault models compared to March-Y and March-C algorithms.

This paper is covered as follows. In segment II discusses our proposed PA algorithm. In segment III, the Implementation results are conveyed. Finally, segment IV concludes this work.

II. PROPOSED PA ALGORITHM

PA Algorithm works on group of memory and performs three terminal write and read process over group, which is not performed on other March test algorithms. PA algorithm involved following steps as shown below. The algorithm contains 20-3 Write Read operations (3WR). 3WR addresses of the memory based on the following steps.

a. Wd(0,0,0);
b. Rd(0,0,0),Wd(0,0,1);
c. Rd0,0,1),Wd(0,1,1);
d. Rd(0,1,1),Wd(0,1,0);
e. Rd(0,1,0),Wd(1,1,0);
f. Rd(1,1,0),Wd(1,1,1);
g. Rd(1,1,1),Wd(1,0,1);
h. Rd(1,0,1),Wd(1,0,0);
i. Rd(1,0,0),Wd(3{L1A});
j. Rd(3{L1A}),Wd(3{L0A});
k. Rd(3{L0A});

L1A-Leading '1' alternates L0A-Leading '0' alternates

PA algorithm used all possible patterns for 3WR operations. Instead of simple binary code, PA algorithm deployed with gray coding to grab the advantage of uncertainty. Generally gray code is used for error correction. The structure has no redundant and seems to be simple. Transition takes place from one digit to another digit like EX-OR operation of two binary bits. This can moderate power and optimize the transition for speed. It is easy to apply for any embedded memories. It can used to detect many faults using 3WR concept such as AF, SAF, TF, Cfid, Cfst, WDF, RDF etc,.

```
// pseudo-code for test mode 3WR concept
for(i=1;i<=cells;i=i+2)
    {
        if(write control )
        {
            write  prefix location Wr_up;
            write  desired location Wr;
            write suffix location Wr_d;
             alternate read control generation
        }
        if(read control )
        {
            read prefix location Rd_up;
            read desired location Rd;
            read suffix location Rd_d;
            alternate write control generation
        }
        loop exist after completion PA algorithm
    }
```

The pseudo-code appears that PA algorithms having three write and read pointers. Memory testing starts with initializing the write pointer onto three locations. After that successive read and write function runs over cells under test mode. Similarly all locations in memory get addressed.

TABLE I
COMPARISON OF VARIOUS FAULT MODELS
REGARDING WITH RESPECTIVE ALGORITHMS ARE LISTED BELOW

Algorithms	Faults									
	AF	SAF	TF	SF	CFid	CFst	CFir	WDF	RDF	IRF
March Y	+	+	+	x	x	x	x	x	x	x
March C	+	+	+	+	+	+	x	x	x	x
PAN	+	+	+	+	+	+	+	+	+	+

AF-Address decoder faults SAF – Stuck-at-faults IRF - Incorrect read fault + - detected x -not detected

TF – Transition faults CFid – Idempotent coupling faults CFst–State coupling faults

CFir –Incorrect read coupling faults WDF - Write disturb faults RDF-Read destructive fault

IV. IMPLEMENTATION RESULTS

In this subsection the paper discuss about the implementation of PA algorithm. The PA algorithm describes in HDL coding which undergone synthesize and implementation process by targeting to Xilinx Virtex-V (XC5VLX50).Figure 1 shows the RTL view of PA algorithm.

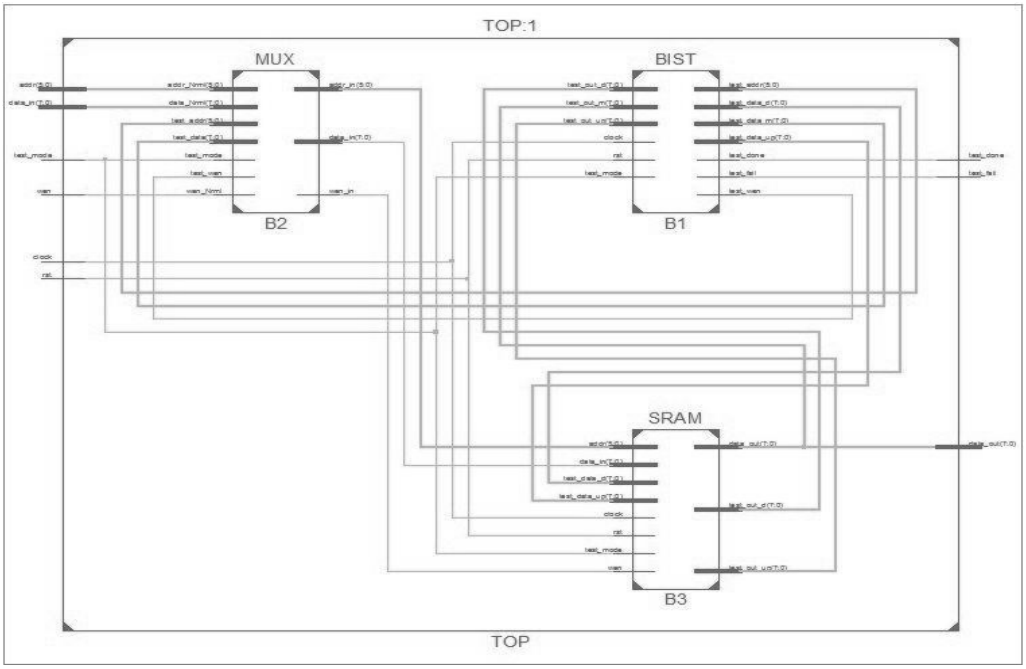

Figure 1: RTL View of MBIST Using PA Algorithm

TABLE II
COMPARISON OF TIMING AND AREA CONSTRAINTS
(I-MBIST USING PA ALGORITHM II-MBIST USING MARCH C ALGORITHM)

S.No	Components	PAA	MARCH C
1.	No of operation	20	10
2.	No of slice registers	580	595
3.	Unique control sets	67	72
4.	Bonded IO	21	23
5.	Asynchronous control signal	3	4
6.	Minimum clock period	3.548ns	3.559ns
7.	Maximum frequency	281.88mhz	380.950mhz

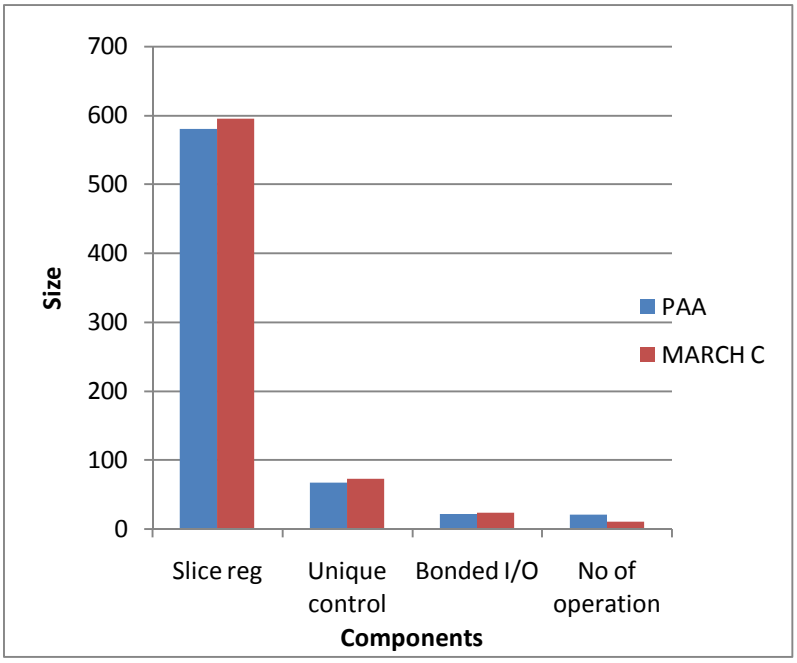

Figure 2: Comparative analysis of March C and PA algorithm

Table 2 summarizes area utilization on implementation process of March C and PA algorithm. The operation of MBIST algorithm increased in PA algorithm for exhaustive fault finding process. Above Figure 2 states that comparative analysis of March C and PA algorithm. PAA utilizes equal resources as March C and improved on fault detection than the March C.

V. CONCLUSION

Testing is one of the most important sections in VLSI. Testing a memory is very challenging role in today's life. Memory BIST became the best solution for embedded cores. This paper presented an efficient MBIST using PA algorithm. PA algorithm is multiport concept and effectively finding the faults that occur in memory. The hardware implementations of PA algorithm are targeted on FPGAs. The results are analyzed and tabulated.

REFERENCES

[1] Harutyunyan,Gurgen "A New Method for March Test Algorithm Generation and Its Application for Fault Detection in RAMs" IEEE Transaction on computer-Aided Design of Integrated Circuits and Systems, June 2012.
[2] Q. Zhao, Y. Ichinomiya, M. Amagasaki, M. Iida, T. Sueyoshi, "A novel soft error detection and correction circuit for embedded reconfigurable systems", IEEE Embedded Systems Letters, September 2011.
[3] Semiconductor Industry Association-International technology roadmap for semiconductors, Dec.2003.
[4] Shyue-Kung Lu, Chun-Lin Yang, and Han-Wen Lin, "Efficient BISR Techniques for Word-Oriented Embedded Memories with Hierarchical Redundancy," IEEE/ACIS International Conference on Computer and Information Science and International Workshop on Component-Based Software Engineering, Software Architecture and Reuse, July 2006.

[5] Gayathri, C.V "Generation of New March Tests with Low Test Power and High Fault Coverage by Test Sequence Reordering Using Genetic Algorithm" International Conference on Advances in Recent Technologies in Communication and Computing, Oct.2009
[6] Yeh, Jen-Chieh "Flash Memory Testing and Built-In Self-Diagnosis With March-Like Test Algorithms" IEEE Transaction on computer-Aided Design of Integrated Circuits and Systems, June 2007.
[7] Hasan, Wan Zuha Wan "A Realistic March-12N Test and Diagnosis Algorithm For SRAM Memories" IEEE Conference on Semiconductor Electronics, Dec 2006.
[8] Narayanan, Venkat "A built-in self-testing method for embedded multiport memory arrays" IEEE Transaction on Instrumentation and Measurement, , Oct 2005.