Built In Self Test Based Memory Testing Using FPGA

[1]Mayank Kumar, [2]Archit Sharma, [3]Ashwani Sharma, [4]Abhay Singh

[1]Assistant Professor, [2]PG Scholar, [3]PG Scholar, [4]Assistant Professor
[1]School of Electrical & Electronics and Communication Engineering,
[1]Galgotias University, Greater Noida, India

Abstract : ***This The Design and implementation of Built In self test (BIST) for RAM testing in VHDL(VHSIC hardware description language) mainly describes how the equipment can be tested itself in the circuit. In this , additional software and hardware components are integrated on the board itself and there is no need of using the additional equipment to check the functionality and performance of the equipment, So by integrating hardware and software will reduce the complexity of using the external components for testing. The circuit will itself observe the performance. BIST is used to test the complex circuits that are not having any external connections. It is the low cost technique to implement the self test. BIST implements most ATE functions on chip so the cost for running the test could be reduced through lesser time, and very less tester memory requirement, or a cheaper tester. Logic BIST applies a large number of test patterns so that more defects can be detected. In addition, logic BIST makes it easy to conduct the at-speed test for detecting timing-related defects. In this we will summarize a Design and Implementation of BIST (Built in Self Test) for Memory Testing scheme that implements the structures using VHDL.***

Index Terms— BIST, VHDL, MEMORY, MISR.

I. INTRODUCTION

Logic built-in self-test (L-BIST) is an a design for testability (DFT) technique in which a portion of a circuit on a chip, board, or system is used to test the digital logic circuit itself. With logic BIST, circuits that generate test patterns and analyse the output responses of the functional circuitry are embedded in the chip or elsewhere on the same board where the chip resides[1]. There are two kinds of memory test methods: electrical (technology-dependent) and functional (technology-independent). Electrical memory testing consists of parametric testing, which includes testing DC and AC parameters, IDDQ and dynamic testing for recovery, retention and imbalance faults[2]. RAM modules are tested both after manufacturing and periodically in the field. During testing, some tests are applied to check that the RAM operates normally[3]. With a built-in test system positioned inside the circuit, the analysis of every single part of the circuit could become a simple task as the period and cost for testing are cut down. This BIST technology is capable of saving the time and cost of maintenance that also allow online diagnosis, which can deal with the even greater advance of embedded systems in future[4]. The objective of this project is to develop a BIST for RAM by using VHSIC Hardware Description Language (VHDL).

II. LITERATURE REVIEW

Andreas Steininger proposed bit This Built-in Self-Test (BIST) approach offered not only economic benefits but also interesting technical opportunities on hierarchical testing and the reuse of test logic during the application of the circuit. It was proposed method only.

U. Siva, Dr. Nisha, implemented Diagnosis and Fault Tolerance for Embedded Bist Rams. In this Due to poor controllability and observability of address, control, and data lines embedded RAMs are extremely very difficult to test. The memory cell in embedded RAM have faults like stuck-at faults, coupling faults, data retention faults, and bridging faults. To address this concern we developed a method which is used to detect the fault, isolate the location of the fault and can predict the remaining useful performance of embedded BIST RAMs. For on-chip memory only.

Mrs. S. Ellammal M.E1, T.Saranya2, Transparent Built in Self Test for word oriented RAMs produces test data with high degree of symmetry. Transparent test approach is applied to the idle state of systems. Reducing the test time is very important for avoiding the interrupt of testing. Transparent word oriented March tests are directly obtained by repeatedly executing the corresponding bit oriented March test on each bit of word. In this system RAMs are of different size are used and each has different width structure.

Mr. "Tsu-Wei Tseng, and Jin-Fu Li, Member of IEEE, with Chih-Chiang Hsu" worked on this field. Built-in self-Test or repair (BISR) technique has been widely used to repair embedded memories (RAMs). This paper presents a

reconfigurable BISR (Re BISR) scheme for repairing different sizes and redundancy organizations with RAM. An efficient redundancy analysis algorithm is proposed to allocate redundancies of defective RAMs.

III. PROPOSED WORK

A. General BIST architecture

BIST technique can be categorized into several general architectures and often classified to be centralized or distributed. These two basic BIST architectures are illustrated in Fig. 1.

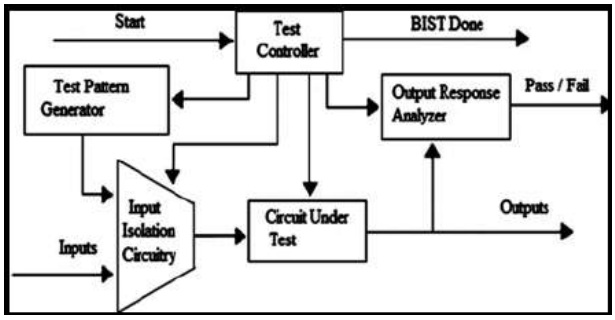


Fig.1.

B. Block diagram for BIST.

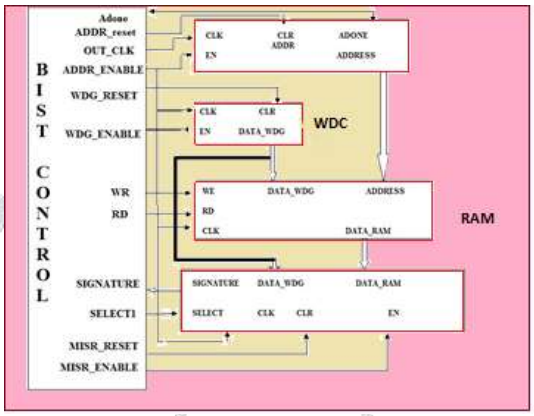
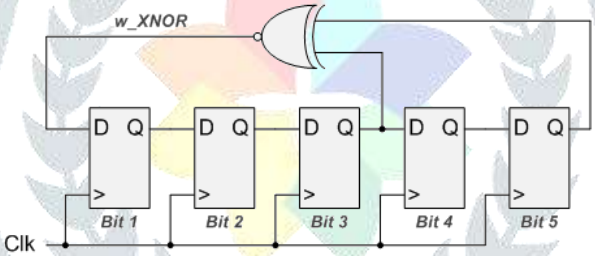


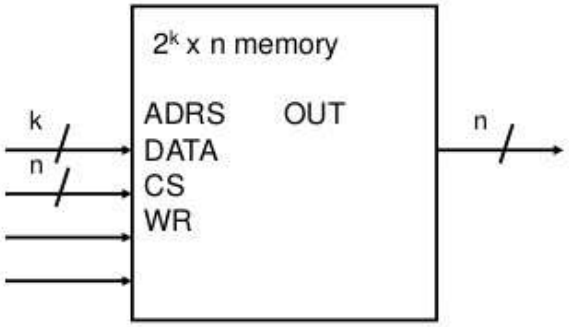
Fig.2.

B.1 . Write data Generation: **-** There are various methods and approaches have been used to generate test patterns during BIST, here we have used **LFSR-** Linear Feedback Shift Register. From fig.2.



LFSR

B.2 Circuit Under Test (CUT): **-** For testing purpose we have used RAM circuit under test shown in ***Figure.*** The system that is to be tested is termed as CUT**.** It is the circuit of the IC that is going to be checked for any defects after its manufacturing. Any digital design represented in VHDL is used as a CUT. From fig.2.



RAM

B.3 BIST Controller: **-** BIST controller coordinates the operations of different blocks of the BIST. Based on the test mode input to the controller, the system either operates in the normal mode or the test mode. When the TM is 1, the system enters the test mode; it gives enable signal to the LFSR which generates the patterns and then it gives enable signal to MISR for the compression of patterns from the RAM. It is the controller which decides for how many cycles the enable need to be made 1 based on the length of the scan chains and the input-output size of the RAM.

B.4 Multiple Input Shift register:-

Thus, it is necessary to reduce the enormous of circuit responses to a manageable size that can be either stored on the chip or can easily compared with the golden response values. For example, a BIST pattern generator in a chip can produce 1 million test patterns. If the chip has a total of 100 primary output, at the end of the BIST process, it will generate a total of 1 million output values or 1000000 x100 = 100 million bits of output values. With such a huge amount of data, it is very costly and almost impossible to store in the storage or ROM inside a chip. There are several approaches and method can be used for response compaction, such as *transition count* response compaction, *LFSR* for response compaction, *Modular LFSR* response compaction, *single-input signature register* (SISR) and *multiple input signature register* (MISR). In this project, multiple input signature register as shown in ***Figure***-2 will be used as response compactor

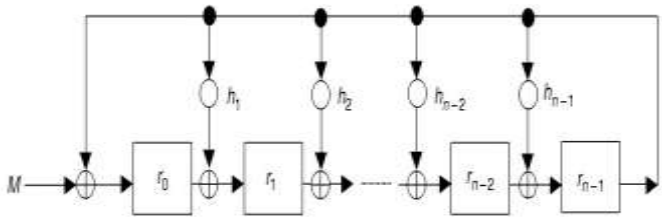


Figure-3 an n-stage multiple-input signature register (MISR)

IV. SIMULATION AND RESULT

A.Bist Controller:-

By using Bist controller we run write data generation , circuit under test, multiple input shift register . Simulation result result is shown below.

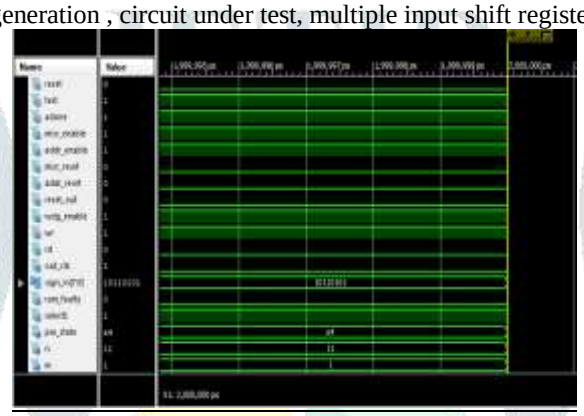


Figure-4

This simulation takes **0.167ns** time delay to give output. Power consumed in this circuit is **33mW.**

B.RAM:-

In this module we give input at some address of RAM & check that same output at the same address show that we can ensure that RAM is working correctly and when we call it in the test bench it work properly .

Figure-5

On the simulation of above block the delay in the circuit is found in the synthesis report that is **0.158ns .** Power consumed in the circuit is calculated using **Xilinx Sparten3E spreadsheet** that is **31mW.**

C-Write Data Generator:-

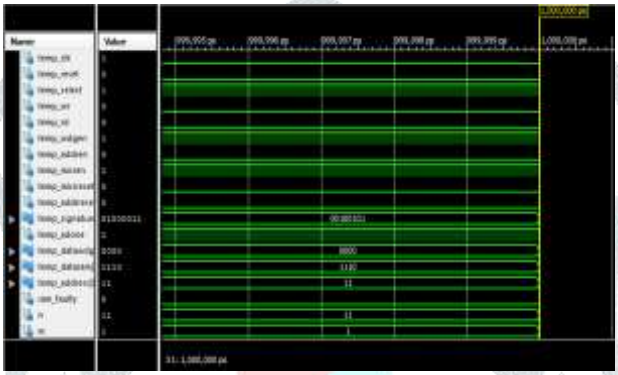
In this module we generate test pattern using data generator which is Module LFSR . We use Module LFSR because it has small delay than other test pattern generator. Here we use this block to generate a number of test pattern generator to save in the selected RAM address. After writing the test pattern in the RAM we read it at the same address using the BIST controller here we get the same address which we write at there . We found that our RAM fault free. The simulation result is shown in the figure.6.

Figure-6

After doing the simulation process we got synthesis report in which the delay resut is **0.157ns.** Power consumed in the circuit is **31mW.**

D-Final Result test bench and Chip Utilization:-

This block call all the component which is shown in the BIST architecture block diagram and make them run together this simulation process gives us whether the RAM faulty or not. It depend on the signature value and the output value which comes from the circuit under test here it is RAM . In the simulation process we found the RAM fault free.



Final Simulation Result

FPGA

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	8	9,312	1%	
Number of 4 input LUTs	7	9,312	1%	
Number of occupied Slices	5	4,656	1%	
Number of Slices containing only related logic	5	5	100%	
Number of Slices containing unrelated logic	0	5	0%	
Total Number of 4 input LUTs	7	9,312	1%	
Number of bonded IOBs	20	232	8%	
Number of BUFG/BUFGCTRLs	1	24	4%	
Average Fanout of Non-Clock Nets	2.33			

Chip Utilization FPGA Report

V. CONCLUSION AND FUTURE WORK

Each & every component is simulated individually, also BIST is simulated using components. All the components and BIST shows good results. Waveforms are shown in the report, which shows correct results.

There were a lot of work remain in this field of memory testing like cross coupling fault and many other fault in the memory which should be removed for the betterment of the memory. Because in the future memory plays a very important role in the era of the integrated circuits .

.

REFERENCES

[1] Pushpraj Singh Tanwar, Priyanka Shrivastava "VHDL Implementation of Logic BIST (BuiltIn Self Test) Architecture for Multiplier Circuit for High Test Coverage in VLSI Chips."

[2] Shaik Karimullah G1,, Abdul Rahim b C.Ushasree2 " DESIGN AND VERIFICATION OF ONLINE BIST FOR DIFFERENT WORD SIZES OF MEMORIES"

[3] Mrs. S. Ellammal M.E1, T.Saranya2 "Transparent Test Scheme in Online BIST For Word-Oriented Memories"

[4] M.H. Husin, S.Y. Leong, M.F.M. Sabri, R. Nordiana "Built in self test for RAM Using VHDL"

[5] V.Kirthi1, Dr.G.Mamatha Samson2,"Design of BIST with Low Power Test Pattern Generator"

[6] Tseng, T. W., L. Jin Fu and C.C. Hsu 2010. Re BISR: A Reconfigurable bilt in self repair scheme for random access memories in SOCs. IEEE trans. Very Large scale integration (VLSI) Syst., 18(6): 921-932.

[7] Lusco, M.A., J.L. Dailey and C.E. stourd, 2011 BIST for multipliers in altera cyclone II field Programmable gate arrays. IEEE 43rd system theory (SSST), Mar, 14-16, pp: 214-219.

[8] Yuejain, W., S. Thomson, D Mutcher and E. Hall, 2011. Built-In Functional test for silicon validation and system integration of telecom SOC designs. IEEE trans. Very large scale integration (VLSI) Syst., 19(4): 629-637.

[9] Peterson, W. W. and Weldon, E. J. Jr. Error-Correcting Codes, 2nd ed. Cambridge, MA: MIT Press, p. 476, 1972.

[10] P.H. Bardell, W.H. McAnney, Parallel Pseudo-random Sequences for Built-In Test, Proc. Int. Test Conf., IEEE, 1984, pp. 302-308.