www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 1 January 2015, Page No. 9809-9813

An Advanced BIST Architecture with Low Power LBIST and BDS Oriented March Algorithm for Intra Word Coupling Faults

P.Ananda Babu[1], Jagadeesh Samudrala[2]

[1] Aditya Engineering College,
Andhra Pradesh, India
perlianand@gmail.com

[2]Aditya Engineering College,
Andhra Pradesh, India
samudrala.naren@gmail.com

Abstract: ***Today's System-on-Chips (SoCs) design and test confront several problems, especially power dissipation. Generally, power dissipation of a system in test mode is more than in normal mode. This is because a significant correlation exists between consecutive vectors applied during the circuit's normal mode of operation, whereas this may not be necessarily true for applied test vectors in the test mode. Reduced correlation between the consecutive test vectors increases the switching activity and eventually the power dissipation in the circuit. The second reason of increasing the power dissipation during test is because the test engineers may test cores in parallel to reduce the test application time. This extra power (average or peak) can cause problems such as instantaneous power surge that causes circuit damage, difficulty in performance verification and decreased overall product yield and cost. Low power test application has become important in today's VLSI design and test.***

Keywords: System-on-Chip (SoC), Built in Self Test, Design for testability, VLSI.

1. Introduction

Built-In Self-Test (BIST) has emerged as a promising solution to the VLSI testing problems. BIST is a DFT methodology aimed at detecting faulty components in a system by incorporating the test logic on chip. BIST is well known for its numerous advantages such as improved testability, at-speed testing and reduced need for automatic test equipment (ATE). In BIST, a linear feedback shift register (LFSR) generates test patterns and a multiple input shift register (MISR) compacts test responses. Test vectors applied to a circuit under test at nominal operating frequency may have more average and/or peak powerdissipation than those in normal mode. The reason is that the random nature of patterns reduces the correlation between the pseudorandom patterns generated by LFSR compared to normal functional vectors. It results in more switchings and powerdissipation in test mode

Several techniques have been reported to address the low power BIST problem. The technique proposed in [1] consists of a distributed BIST control scheme that simplifies the BIST execution of complex ICs, especially during higher test activity levels. This approach can schedule the execution of every BIST element to keep the power dissipation under the specified limit. A BIST strategy called dual-speed LFSR [2] is proposed to reducethe circuit's overall switching activities. Having two different speed LFSRs, the proposed strategy applies some test patterns using low-speed LFSR by connecting to some inputs that have elevated transition densities. The low-power test pattern generator presented in [3] is based on cellular automata, reducing the test power in combinational circuits. Another low-power test pattern generator based on a modified LFSR is proposed in [4]. This scheme reduces the power in circuit under test (CUT) in general and clock tree in particular. A low power BIST for data path architecture built around multiplier-accumulator pairs is proposed in [5].

In this paper, read faults are addressed along with classic faults with an improvement in the efficiency of the architecture and test time in detecting the faults. Traditional March C- algorithm is used to test the single cell faults and linked faults. However, the effects of read faults such as Deceptive Read Destructive and Data Retention Faults are ignored. In [1], a March test BLC of a 46n complexity is presented to detect all static faults caused by CC_{BL}. Despite the effectiveness of March BLC, its time complexity is still high. Moreover, traditional March algorithms ignore the effect of Data Retention Faults, which are likely to occur when the standby mode is applied during low supply voltage. Hence, the optimized March C- (complexity 12n) algorithm proposed in [3], which covers all the single cell and read faults including Data Retention Faults with less time complexity is chosen to analyze and test faults in Look Up Tables. To address the above mentioned faults the current research uses a Built-in Self Test (BIST) technique for testing the SRAM memories in a Xilinx Virtex-4 series FPGA. In the field of FPGAs, BIST has been a topic of research and development for the past decade [1-8]. The BIST technique employs a Test Pattern Generator (TPG) which generates the input test pattern to test the Circuit Under Test (CUT). Responses from the CUT are fed into an Output Response Analyzer (ORA) for comparison of the test outputs. Deviations of any kind from the expected values are considered as a fault. Our scheme offers an improved architecture with increase in number of CLBs covered in one test session. The TPG-CUT interconnect problems are also examined.

2. SRAM-READ OPERATION

Static Random Access Memory (Static RAM or SRAM) is a type of RAM that holds data in a static form, that is, as long as the memory has power. Unlike dynamic RAM, it does not need to be refreshed. SRAM stores a bit of data on four transistors using two cross-coupled inverters. The two stable states characterize 0 and 1.

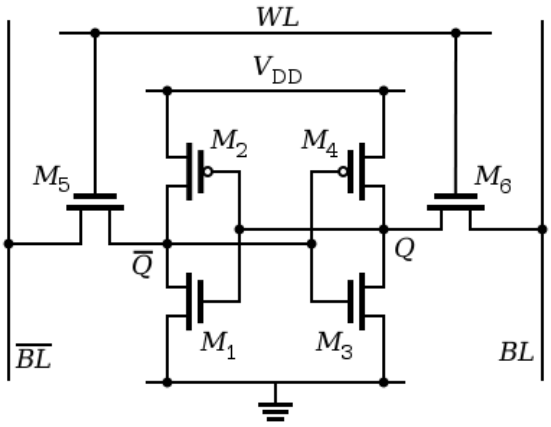

Figure 1 SRAM bit Cell

During read and write operations another two access transistors are used to manage the availability to a memory cell. To store one memory bit it requires six metal-oxide-semiconductor field-effect transistors (MOFSET). MOFSET is one of the two types of SRAM chips; the other is the bipolar junction transistor. The bipolar junction transistor is very fast but consumes a lot of energy. MOFSET is a popular SRAM type. S-RAM Architecture For Read and Write. Assume that the content of the memory is a 1, stored at Q. The read cycle is started by precharging both the bit lines to a logical 1, then asserting the word line WL, enabling both the access transistors. The second step occurs when the values stored in Q and Q- are transferred to the bit lines by leaving BL at its precharged value and discharging BL- through M1 and M5 to a logical 0 (i.e. eventually discharging through the transistor M1 as it is turned on because the Q is logically set to 1). On the BL side, the transistors M4 and M6 pull the bit line toward VDD, a logical 1 (i. e. eventually being charged by the transistor M4 as it is turned on because Q is logically set to 0). If the content of the memory was a 0, the opposite would happen and BL would be pulled toward 1 and BL toward 0. Then the BL and BL- lines will have a small voltage difference between them while reaching a sense amplifier, which will sense which line has the higher voltage thus determining whether there was 1 stored or 0. The higher the sensitivity of the sense amplifier, the faster is the speed of the read operation.

3. MARCH C- ALGORITHM

March algorithms are known for memory testing because March-based tests are all simple and possess good fault coverage hence they are the dominant test algorithms implemented in most modern memory BIST. The proposed march algorithm is modified march c- algorithm which uses concurrent technique. Using this modified march c- algorithm the complexity is reduced to 8n as well as the test time is reduced greatly. Because of concurrency in testing the sequences the test results were observed in less time than the traditional March tests. This technique is applied for a memory of size 256x8 and can be extended to any memory size.

{ ⇳(wo) ; ⇑(r0,w1) ; ⇑(r1,w0); ⇓(r0,w1) HOLD
M0 M1 M2 M3

⇓(r1,r1,w0) HOLD; ⇳(r0,r0) }
M4 M5

March –C algorithm

The fault coverage of the algorithm is detailed below.

- All SAFs, RDFs, and IRFs are detected because a '1' and '0' are read from each cell. All TFs are detected since a cell is read after an up and a down transition operation. In this case, Up-transient faults are detected by March element M1 followed by M2 and downtransients are detected by M2 followed by M3.
- All ADFs are detected after writing a value to all thecells and reading them with the expected values. Thisfault is detected by M2 element.

4. BIST Architecture

A conventional scan-BIST architecture uses an LFSR to generate pseudo-random patterns that are serially loaded into each scan chain of the circuit under test.

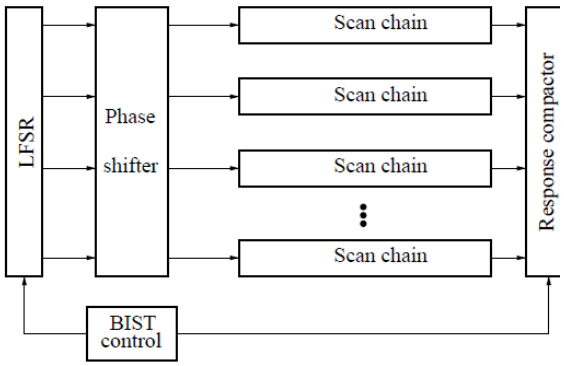

Figure 2: A typical BIST architecture based on pseudorandom patterns.

The proposed BIST scheme for testing is designed based on three factors: 1) FPGA test time, 2) reducing dependencies on TPG- CUT inter connects, and 3) the ease of tracing the fault location in the device. This section analyzes the BIST methodologies for testing FPGAs that have been proposed in the past and then compares it to the proposed scheme. BIST architecture can be mapped on FPGAs in different ways. The simplest way is where each TPG-CUT-ORA occupies one configurable logic block (CLB). This method, although simple, is very time consuming.

The method outlined in [9] uses Circular BIST comparison-based ORAs to compare the outputs of multiple identical CUTs. Although it increases the accuracy of fault detection, it increases the load on TPGs as this technique uses two TPGs for the entire row. Hence, the accuracy of cell selection is dependent on interconnects. Because of this, tracing an error to a genuine fault or a faulty TPG is quite difficult in this method. The re-programmability of FPGAs is well exploited in [10]. However, external memory is required to store the BIST configurations and the time required to download and execute the BIST is considerable. The BIST scheme proposed in [11], has TPG in a different CLB and the scheme proposed in [12] has TPG/ORA in a different CLB, hence to test all the slices in

a CLB using [11] [12] would take four sessions. Also, these] methods are dependent on TPG-CUT-ORA interconnects for sending the correct address and detection of faulty output. Furthermore, each CUT needs TPG and an ORA, and in these past studies each TPG/ORA occupies an entire CLB. So in one session only half of the CLBs act as CUTs. Hence, they were able cover half CLBs in one test session.

The current work proposes a different BIST architecture where the TPG, CUT, and ORA are incorporated into one CLB (Figure 4). A separate TPG and ORA is used for each CUT. Multiple identically configured TPGs supply test patterns to identically configured CUTs. The outputs of CUTs are monitored by two ORAs and compared with the outputs of two other identically configured CUTs in an adjacent row, as shown in Figure 4. For example, if the third memory is faulty, comparing it with the second memory in the chain results in a faulty signal at ORA#3. ORA#3 results a 1 indicating fault at CLB#3. CLB #1 has no previous ORA output, hence to complete the circular comparison the output of the first CUT is compared with last CUT. After this, the entire CLB is tested twice such that TPG and ORA can serve as CUTs.

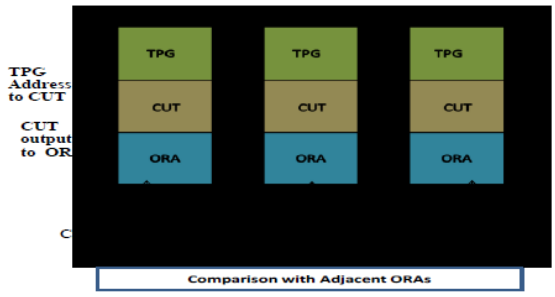

Figure 3 .Proposed Bist Architucture

A single CLB in a Virtex- 4 FPGA has 4 slices (Slice L and Slice M). Each slice has two LUTs. The TPG is constructed using four LUTs. It produces addresses 0 to 15. For testing using March algorithms, the addresses need to be generated sequentially. A simple up/down counter is used for this purpose. A single Slice L is used to build an ORA and the CUT is constructed using a Slice M, since Slice M contain LUTs which can operate as RAMs. Hence, using the proposed architecture the LUT RAMs in a CLB can be tested in two test sessions. The detailed interconnection scheme is shown in Figure5.

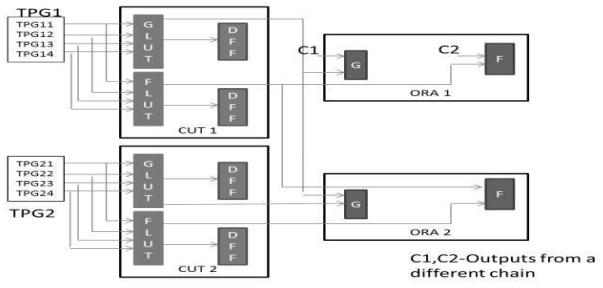

Figure 4. Interconnections of the proposed BIST scheme

Detection of the faulty LUT/ RAM (F or G) is possible through the ORA outputs which have two faulty signals, one for each LUT. If all the ORA outputs (F1-F4 or G1-G4) show "0000" then it can be concluded that no fault exists in the row. When a fault exists, the corresponding signal goes high. For example, when the ORA output shows F2 "0010", then it can be determined that the fault exists at CLB#2 of F LUT.

Similarly, "0100" (CLB#3) and "1000" (CLB#4) identify the fault. The exact address at which the fault is present can be found from the TPG address.

Advantages of the proposed scheme are

1. The usage of multiple TPGs prevents fault aliasing that may occur when using a single TPG that has been synthesized containing a fault.

2. It reduces the load on TPGs, as there is one TPG for every CUT.

3. It uses only one CLB per CUT while ensuring that all LUTs of a CLB are tested in three sessions.

4. The dependency on TPG-CUT interconnects is nullified as the TPG and CUT are incorporated into the same CLB. This in turn reduces the Cross talk effects and delay faults.

5. Using the architectures proposed in [9] only 46% of the CLBs are covered in one test session and using and [12] 50% of the CLBs are covered. This is because of the dedicated CLBs for TPGs and ORAs. Using the architecture proposed the number of CLBs covered in a single session has been increased to 54% and 50% compared to [9] and [11] [12] respectively.

All of these advantages make this method better suited for testing LUTs using BIST

	BIST Summary
#logic Slices	808
#lines of verilog	1069
Test time taken	22.45us

Table 1.Summary of BIST Results

Table I summarizes the actual implementation of BIST circuitry in Virtex-4 FPGAs. This includes the number of slices occupied by the BIST circuitry, the number of lines of Verilog code for the complete BIST circuit, and the total test time taken to detect the faults.

5. Experimental Results

The functional model of the Virtex-4 FPGA is implemented using Verilog and simulations are done in ModelSim. The optimized March C- algorithm is used for testing purposes. Initial simulations are done without introducing faults and then the simulations are done for each of the individual faults: stuck-at, transient, IRF, RDF, DRDF, DRF, and address decoder. The detailed explanation for the read faults is given below

Case I: Incorrect Read Fault at address 1001.

IRF is introduced at the address 1001 of GLUT in CLB#1. Due to this, the read operation that is performed on cell 9 always returns an incorrect logic value even though the correct value is

still stored in the cell. The fault is detected by the March element M1.

Case II: Read Destructive Fault at address 1010

RDF is introduced at address 1010 of GLUT in CLB#1. Due to this, whenever a read operation is performed on cell 10, the data stored in the cell changes to its complement and returns an incorrect value at the output. The fault is identified first by March element M1.The difference between RDF and IRF is that in case of RDF the content of the cell changes. However, in case of IRF, the content of the cell remains same.

Case III: Deceptive Read Destructive Fault at address 1100

DRDF is introduced at the address 1100 of FLUT in CLB#1. Whenever cell 12 is read for an expected value, it initially returns the actual value and then the next read it returns its complement because the first read changes the content of the cell. The fault is detected by M4.

Case IV: Data Retention Fault at address 1011

DRF is introduced at the address 1001 of FLUT in CLB#1.

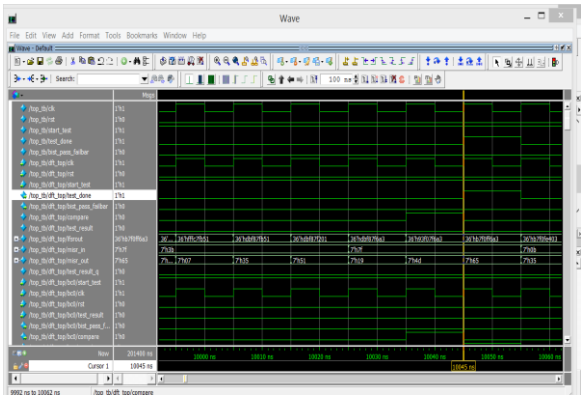

Figure 5.Simulation Results of LPLFSR.

Whenever cell 9 is not accessed for a certain period of time and read for an expected value, the state of the cell flips and returns an incorrect value at the output. The cell is placed in an idle state using HOLD command in the March algorithm. The Fault is identified by M4

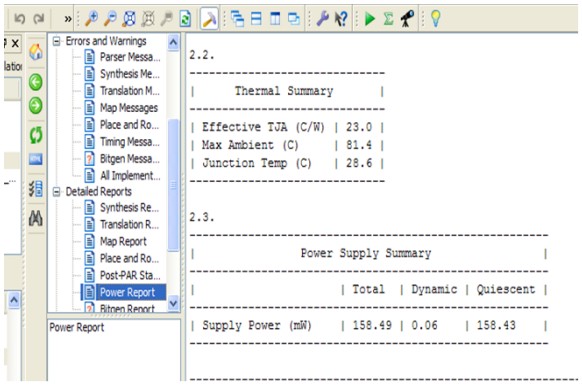

Figure 6.Power Report of LPLFSR

#	Fault Models with fault at address 0001	No. of clock Cycles taken by the proposed method	No.of clock Cycles [10]
1	SAFI	18	17
2	SAF0	34	77
3	ADF	34	77
4	Up-TF	34	77
5	Down-TF	76	83
6	IRF	18	NA
7	RDF	18	NA
8	DRF	108	NA
9	DRDF	109	NA

Table 2: Fault coverage and test time taken to cover faults

6. Conclusions

This paper presents a novel BIST architecture to test SRAM based FPGAs is presented. BIST is developed and implemented to test the memory resources of Virtex-4 FPGAs using an optimized March C- algorithm. The number of CLBs in a single session using the proposed BIST scheme is increased by 54% and 50% as compared to [9] and [11] [12] respectively. The number of slices occupied by the BIST scheme, the number of lines of VHDL code for the implementation of BIST scheme, and the total test time taken is summarized in table 1.

Optimized March C- algorithm requires 12n operations to completely test the memory for the presence of stuck-at, transient, IRF, RDF, DRDF, DRF, and address decoder faults. Assuming a clock period of 5 MHZ, the exact number of clock cycles to detect the fault and the time taken to locate the fault is calculated, compared with [11], and listed in Table 2. Simulations are done using ModelSim to verify the successful testing and fault location capabilities of the approach.

References

[1] S. Irobi, Z. Al-Ars, and S. Hamdioui, "Memory Test Optimization for Parasitic Bit Line Coupling in SRAMs," European Test Symposium
(ETS), 2011.
[2] A.J. van de Goor, S. Hamdioui, H. Kukner , "Generic, orthogonal and low-cost March Element based memory BIST," IEEE International Test Conference (ITC), 2011, pp.1-10.
[3] Qikai Chen, H. Mahmoodi, S. Bhunia, and K. Roy, "Efficient testing of SRAM with optimized March sequences and a novel DFT technique for emerging failures due to process variations," IEEE Transactions on VLSI Systems, 2005, vol.13, pp. 1286-1295.
[4] M.J. Geuzebroek, J.T. van der Linden, A.J. van de Goor, "Test point insertion that facilitates ATPG in reducing test time and data volume," Proceedings of International Test Conference, 2002, pp. 138- 147.

[5] L. Dilillo, B. M. Al-Hashimi, "March CRF: an efficient test for complex read faults in SRAM memories", Design and Diagnostics of Electronic Circuits and Systems, 2007, pp. 1-6.
[6] Z. Lixing, V.D Agrawal, "Net diagnosis using stuck-at and transition fault models," IEEE VLSI Test Symposium (VTS), 2012,vol.23-25, pp.221-226.
[7] A.J. van de Goor, S. Hamdioui, "MBIST architecture framework based on orthogonal constructs," Design and Test Workshop (IDT),2010,vol. 14-15, pp.128-133.
[8] E. Atoofian and Z. Navabi "A BIST architecture for FPGA look-up table testing reduces recomfigurations", Proceedings of IEEE Asian Test Symposium, 2003, pp.84 -89
[9] B. Dutton and C. Stroud, "Built-in self-test of embedded

SEU detection cores in virtex-4 and virtex-5 FPGAs," International Conference on Embedded Systems and Applications, 2009, pp. 149-155.
[10] C. Carmichael and C. Wei Tseng. "Correcting Single-Event Upsets in Virtex-4 FPGA Configuration Memory." Xilinx Corporation, 2009.

Author Profile

P.Ananda Babu received his B.Tech Degree in 2010 from Chirala Engineeing College,AP,India. He is currently working toward Post Graduation degree in the Department of Electronics and Communication Engineering, Aditya Engineering College,AP,India. His Research Interests are VLSI Implementation of Digital Systems, Image Processing, and Communication Systems.

Jagadeesh Samudrala received her post Graduation Degree from JNTUK University,AP,India. Currently he is working as Assistant Professor in the Department of Electronics and Communication Engineering in Aditya Engineering College, AP, India with as Experience of 10 Years in teaching. He is Associate life member in IETE. His Research interests are Image and Signal Processing.