BEST: International Journal of Management, Information Technology and Engineering (BEST: IJMITE)
Vol. 1, Issue 3, Dec 2013, 79-84

DESIGN OF LOGIC BIST USING BIPARTITE LFSR

SABIR HUSSAIN[1] & KAMAL UZ ZAMAN[2]

[1]Assistant Professor, Department of Electronics & Communication, MJCET, Osmania University, Hyderabad, India

[2]ME Student, Department of Electronics & Communication, MJCET, Osmania University, Hyderabad, India

ABSTRACT

This paper discusses the design of ASIC (Application Specific Integrated Circuit) for LFSR (Linear feedback shift register) for testing of digital VLSI circuits using BIST technique. In this design an ASIC based programmable LFSR is used as Test Pattern Generators (TPG). Implementation of any design on ASIC is only possible with EDA (Electronic Design automation) tools. In this paper the cadence tool is used to accomplish the task. The simulation and synthesis results are presented. Further analysis of power, logic area usage and timing of controller is done on the synthesis results.

KEYWORDS: LFSR, Logic BIST, ASIC, Verilog HDL, Cadence Tool

INTRODUCTION

The system on chip (SoC) revolution challenges both design and test engineers, especially in the area of power dissipation. Generally, a circuit or system consumes more power in test mode than in normal mode. This extra power consumption can give rise to severe hazards in circuit reliability or, in some cases, can provoke instant circuit damage. Moreover, it can create problems such as increased product cost, difficulty in performance verification, reduced autonomy of portable systems, and decrease of overall yield. Low power dissipation during test application is becoming increasingly important in today's VLSI systems design and is a major goal in the future development of VLSI design [1][2]. Electronic design automation tools and proper selection of pattern generator can be achieved above problems. Linear feedback shift registers (LFSRs) is in the generation of test vectors for digital circuits. This is because, with little overhead in the hardware area, a normal register can be configured to work as a test generator and, with an appropriate choice of the generator (G1, G2 and G3 Sequences), the LFSR can generate all possible output test patterns. Furthermore, the pseudorandom behavior of the LFSR reduces the correlation between successive test patterns, which means that it can achieve high fault coverage in a relatively short run of test vectors. However, this lack of correlation substantially increases the weighted switching activity (WSA) within the circuit-under-test (CUT). This often causes the power consumed during test mode operation to be much higher than during normal mode operation, which can lead to problems with battery lifetime and system reliability [3, 4].

In this research article, we focused on properly selecting the characteristics of LFSR for ISCAS'85 benchmark circuit and Section 3 describes experiments run on the ISCAS'85 benchmark circuits and describes the power usage, area overhead and timing analysis. We also focused on design of ASIC using CAD tools.

The remainder of the paper is organized as follows: Sections 2 describes BIST Architecture and function of each block. Bipartite LFSR as test pattern generators (TPG) explains in Section 3. Section 4 shows that the experimental results -simulation, synthesis and post layout simulation are extracted. Finally, section 5 draws some conclusions and future scope.

BASIC BIST ARCHITECTURE

Built- in self-test (BIST) [2] has been proven to be one of the most cost-effective and widely used solutions for VLSI circuit testing. BIST is basically same as off-line testing using ATE where the test pattern generator and the test response analyzer are on-chip circuitry (instead of equipments). As equipments are replaced by circuitry, so it is obvious that compressed implementations of test pattern generator and response analyzer are to be designed [6]. The basic architecture of BIST is shown in Figure 1.

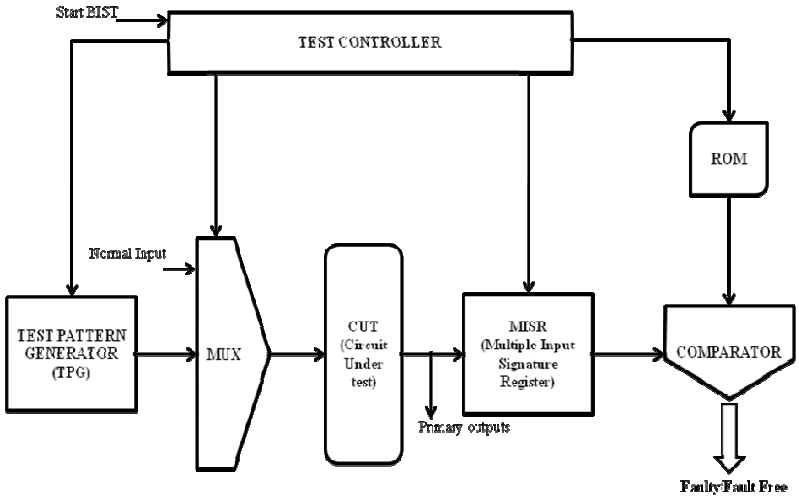

Figure 1: BIST Architecture

Hardware Test Pattern Generator

This module use to generates the random pattern generation as test patterns required to sensitize the faults here in our design functional fault testing performed of ISCAS‘85 benchmark circuits. As the test pattern generator is a circuit (not equipment) its area is limited. So storing and then generating test patterns obtained by ATPG [1-3] algorithms on the CUT using the hardware test pattern generator is not feasible. In other words, the test pattern generator cannot be a memory where all test patters obtained by running ATPG algorithms (or random pattern generation algorithms) on the CUT are stored and applied during execution of the BIST [7]. Instead, the test pattern generator is basically a type of register which generates random patterns which act as test patterns. The main emphasis of the register design is to have low area yet generate as many different patterns (from 0 to 2n, if there are n flip-flops in the register) as possible.

Input Multiplexer

This multiplexer is to allow normal inputs to the circuit when it is operational and test inputs from the pattern generator when BIST is executed. The control input of the multiplexer is fed by a central test controller.

Output Response Compactor

Output response compacter performs lossy compression of the outputs of the CUT (here we have used Benchmark circuit to test). As in the case of off-line testing, in BIST the output of the CUT is to be compared with the expected response called golden signature we have used 16 bit “AAD2”in Hexadecimal format shown in simulation results as golden signature. if CUT output does not match the expected response, fault is detected otherwise fault free[8]. Similar to the situation for test pattern generator, expected output responses cannot be stored explicitly in a memory and compared with the responses of the CUT.

Read Only Memory (ROM)

Stores golden signature that needs to be compared with the compacted CUT response.

Comparator

We have used a Hardware comparator to compare compacted CUT response and golden signature (from ROM) the output of comparator is O1 It indicate logic 1 for fault free and logic 0 for faulty.

Test Controller

We designed a mealy FSM machine to control the BIST modules. Whenever an IC is powered up (signal start BIST is made active) the test controller starts the BIST procedure. Once the test is over, the status line is made high if fault is found. Following that, the controller connects normal inputs to the CUT via the multiplexer, thus making it ready for operation. Among the modules discussed above, the most important one is hardware test pattern generator (LFSR) as applied algorithm to test VLSI circuits. In the next sections we will discuses the algorithm steps and implemented algorithm on ASIC.

ALGORITHM FOR BIPARTITE LFSR

We designed Bipartite LFSR by dividing LFSR into two halves by applying two complement (non-overlapping) enable signals. In other words, when one half is working, the other half is in idle mode. This LFSR includes FFs with enable signals. Let consider en1 and en2 are two non overlapping enable signals. When en1en2=10, the first half of LFSR is active and second half is ideal, while with en1en=01, the second half active and first half is ideal. The Bipartite LFSR patterns are generated by combining T^1 and T^{1+i} gives T^k. Note carefully that the new pattern does not change the characteristic function of LFSR. The LFSR's operation is effectively split into two halves and the resultant pattern T^k is an interface between these two patterns.

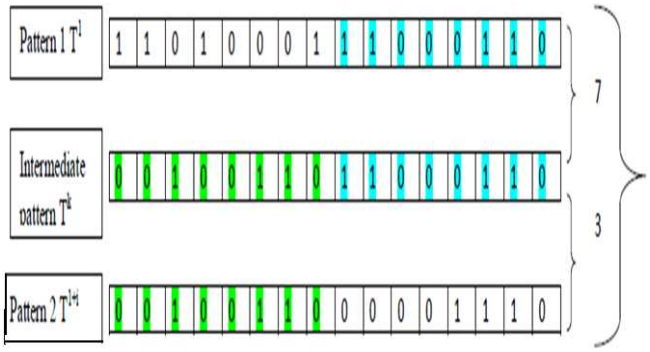

Figure 2: Bipartite LFSR

The randomness property of the this LFSR is due to dividing it into two smaller LFSR and also it requires generating and distributing two non-overlapping clocks (with half frequency) which in turn increases the area overhead [9]. The Bipartite LFSR keeps the randomness property of the n-bit LFSR intact and it also reduces the overall power consumption. The above Figure 2 explains the inserting intermediate pattern T^k, between two consecutive patterns T^i and T^{i+1}.

EXPERIMENTAL RESULTS

The below figure 3 is simulation view of Bipartite LFSR is executed in Cadence RTL Compiler. There are number of buffers (as d-ff) whose outputs are connected to each other internally. Finally all the outputs of LFSR's are stored and

stream of bits are generated as Test pattern Generators (TPG). To validate the effectiveness of the proposed technique simulation and synthesis were carried out with Cadence SimVision and Cadence RTL Compiler GPDK 180nm CMOS library is used.

The below simulation of Bipartite LFSR the enable signals as en1 and en2 performs main role in operation. Hence the pattern saved in the signature is “AAD2” as golden signature, as the comparator signal o1 verifies the d_out and signature outputs and logic 1 for fault free, logic 0 for faulty CUT.

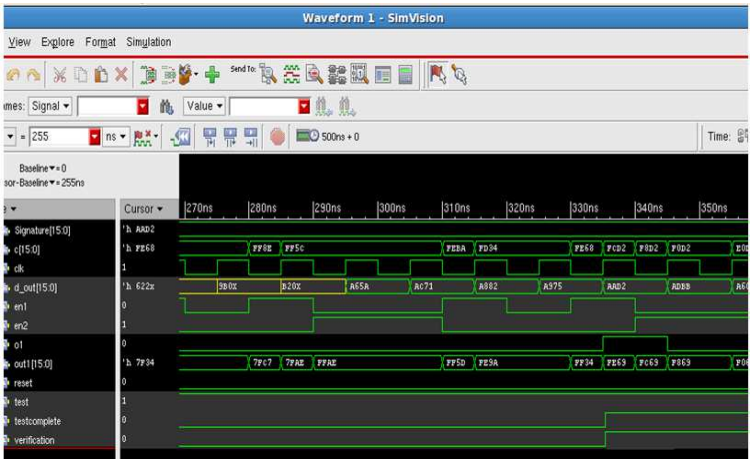

Figure 3: Simulation Results of Bipartite LFSR

Synthesis Report

The table1 is the analysis of Bipartite LFSR compared in terms of Power, Area, timing and number of gates. In designing of architecture we have considered four Evaluation metrics are Total Power, Timing, Area overhead and Numbers of Gates.

Table 1: Parameters of Bipartite LFSR

Parameters	Power (nw)	Timing (ns)	Area (Cells)	Gates
Bipartite LFSR	19355.70	781	91	788

Post Layout Simulation of Bipartite LFSR

This experimental part deals with the post layout simulations of LFSRs with top modules, which are executed in Cadence Encounter RTL Tools. First invoking the tool then proceeding for the design, Layout steps have been proposed for execution, each step will take you near to fabrication of design [10].

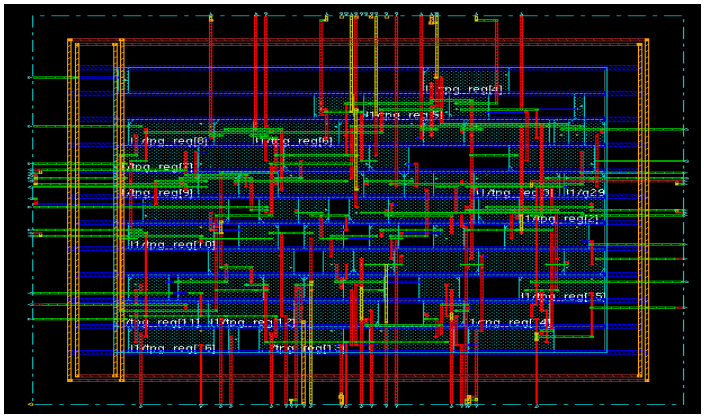

Figure 4: Post Layout Simulation of Bipartite LFSR

CONCLUSIONS

This paper presented Bipartite LFSR architecture for ISCAS'85 benchmark circuits. This design can be applied to almost all Test Pattern Generators. Our method is based on dividing LFSR into two halves by applying two complement enable signals. We achieved ASIC implementation with the help of this technique and analyzed power and area overhead and timing for BIST controller. It is concluded that Bipartite LFSR is very useful for BIST Implementation in which the CUT may be Combinational, sequential and memory circuits. Using this technique we can further improve the performance in BILBO implementation.

ACKNOWLEDGEMENTS

The authors are thank full to Dr Basheer Ahmed Principal for his encouragement and support in publishing of this paper and are also highly thankful to management of Muffakham Jah College of engineering and technology for their financial support.

REFERENCES

1. Ahmed N. Awad & Abdallatif S. Abu-Issa "Low Power Address Generator for Memory Built-In Self Test" The Research Bulletin of Jordan ACM, Vol II(III).
2. Wang, Gupta, "DS-LFSR: A BIST TPG for Low Switching Activity, IEEE Transactions On Computer-Aided Design of Integrated Circuits And Systems, 7,July, 2002.
3. Sabir Hussain, K. PadmaPriya, "Test Pattern Generator (TPG) for Low Power Logic Built In Self Test (BIST)", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,Vol. 2, Issue 4, April 2013,pp.1634-1640.
4. P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "A Test Vector Inhibiting Technique for Low Energy BIST Design," in Proc. IEEE 17th VLSI Test Symp., Apr. 1999, pp.407–412.
5. P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, J. Figueras, S. Manich, P. Teixeira,and M. Santos, "Low Energy BIST Design: Impact of the LFSR TPG Parameters on the weighted Switching Activity," in Proc. International Symp. Circuits and Systems, June, pp.110–113.
6. P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and H. J. Wunderlich, "A ModifiedClock Scheme for a Low Power BIST Test Pattern Generator," in Proc. IEEE 19th VLSI TestSymp., May 2001, pp. 306–311.
7. P. Girard, N. Nicolici, and X. Wen, editors, Power-Aware Testin and Test Stargegies for Low Power Devices. Springer, 2009.
8. M. Tehranipoor, M. Nourani, and N. Ahmed, "Low Transition LFSR for BIST-Based Application," in Proc. IEEE 14th Asian Test Symposium, 2005.
9. A. S. Abu-Issa and S. F. Quigley, "Bit-Swapping LFSR and Scan-Chain Ordering: ANovel Technique for Peak and Average-Power Reduction in Scan-Based BIST," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol. 28, no. 5, May 2009.
10. Valarmathi Marudhai," Implementation of LFSR on ASIC" in Pro IEEE indicom conference, pp May 2012 pp 978-1-4673.