iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** X **Month of publication:** October 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 3 Issue X, October 2015
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Fault Detection by Pseudo-Exhaustive Two-Pattern Generator

Akanksha Pandey[1], Prof. Pravin Tiwari[2]
Electronics & Communication Department, Takshshila Institute of Technology, Jabalpur (RGPV)

Abstract— The main objective of this research is to design a Built-in self-test (BIST) technique based on pseudo-exhaustive testing. Two pattern test generator is used to provide high fault coverage. To provides fault coverage of detectable combinational faults with minimum number of test patterns than the conventional exhaustive test pattern generation, increases the speed of BIST and may posses minimum Hardware Utilization. A Built-in self-test (BIST) is a commonly used design technique that allows a circuit to test itself.Built-in self-test (BIST) technique based on pseudo-exhaustive testing, two pattern test generator is used to provide high fault coverage. It is widely known that a large class of physical defects cannot be modelled as stuck-at faults. For example, a transistor stuck- open fault in a CMOS circuit can convert a combinational circuit under test (CUT) into a sequential one, while a delay fault may cause circuit malfunction at clock speed, although it does not affect the steady-state operation. Detection of such faults requires two-pattern tests. It is proposed to design a BIST circuit for fault detection using pseudo exhaustive two pattern generator using minimum hardware utilization and increasing the speed of BIST.
Keywords— BIST, VHDL, FPGA, RPET, CUT

I. INTRODUCTION

In modern VLSI circuits, containing millions of transistors, the utilization of the same BIST pattern generator to test more than one module can drive down the cost of BIST hardware. Modules whose inputs are driven (during BIST) from the same pattern generator may have different cone sizes. Two solutions have been proposed to this direction. One solution is to utilize recursive pseudo exhaustive testing with recursive pseudo exhaustive testing, (n, k) PETS are generated for all *k*=1, 2, 3…... n. An alternative solution is generic pseudo exhaustive testing, introduced in generic pseudo exhaustive generator can generate a (n, k) PETS for any value of by enabling a respective signal PE[k]. BIST pattern generators are commonly discerned into one pattern and two-pattern. One-pattern generators target the detection of combinational (mainly stuck-at) faults. For the correct behaviour of the circuit delay fault is detected by the two pattern generator [1][2].

In this paper, we start by presenting a recursive pseudo exhaustive two-pattern generation scheme. A generic pseudo exhaustive two-pattern generation scheme generates an (n, k) pseudo exhaustive two-pattern test for any value of k, by enabling a proper input signal PE[k], $1 \leq k \leq n$. The generic pseudo-exhaustive two-pattern generator consists of a generic counter, 1's complement adder, a controller and a carry generator (C gen). The n-bit pattern PE [n:1] is given as an input to the controller, generic counter and C_gen and the output A [n:1] is taken from the Accumulator.

II. BASIC BIST PRINCIPLES

The general Built-in Self-Test structure consists of three main parts see Fig.1 the TPG (Test Pattern Generator) produces test patterns that are fed to the inputs of a Circuit under Test (CUT) and the responses of a circuit are then evaluated in a Response Evaluator (RE).

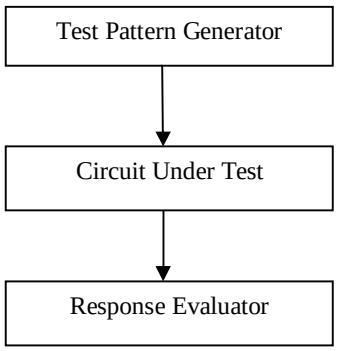


Fig 1 BIST structure

©IJRASET 2015: All Rights are Reserved

Online BIST is performed when the functional circuitry is in normal operational mode. It can be done either concurrently or non concurrently. In concurrent online BIST, testing is conducted simultaneously during normal functional operation. The functional circuitry is usually implemented with coding techniques or with duplication and comparison. When an intermittent or transient error is detected, the system will correct the error on the spot, rollback to its previously stored system states, and repeat the operation, or generate an interrupt signal for repeated failures. In nonconcurrent online BIST, testing is performed when the functional circuitry is in idle mode. This is often accomplished by executing diagnosis software routines (macrocode) or diagnosis firmware routines (microcode). The test process can be interrupted at any time so that normal operation can resume. Offline BIST is performed when the functional circuitry is not in normal mode. This technique does not detect any real-time errors but is widely used in the industry for testing the functional circuitry at the system, board, or chip level to ensure product quality.[2][3]

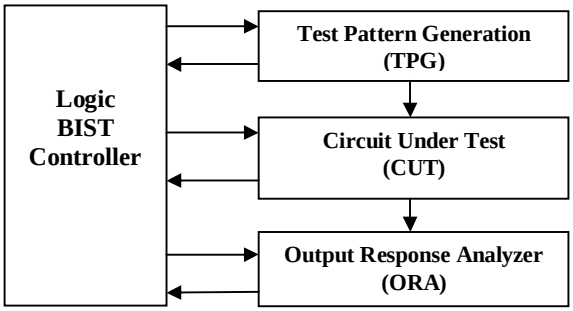


Fig 2 Typical logic BIST system

Fig 2 shows a typical logic BIST system using the structural offline BIST technique. The test pattern generator (TPG) automatically generates test patterns for application to the inputs of the circuit under test (CUT). [4] The output response analyzer (ORA) automatically compacts the output responses of the CUT into a signature. Specific BIST timing control signals, including scan enable signals and clocks, are generated by the logic BIST controller for coordinating the BIST operation among the TPG, CUT, and ORA. The logic BIST controller provides a pass/fail indication once the BIST operation is complete. It includes comparison logic to compare the final signature with an embedded golden signature, and often comprises diagnostic logic for fault diagnosis.

III. BIST TEST PATTERN

Stored Pattern is an automatic test pattern generation (ATPG) and fault simulation technique is used to generate the test patterns. A good test pattern set is stored in a ROM on the chip. When BIST is activated, test patterns are applied to the CUT and the responses are compared with the corresponding stored patterns. Although stored-pattern BIST can provide excellent fault coverage, it has limited applicability due to its high area overhead.[5]

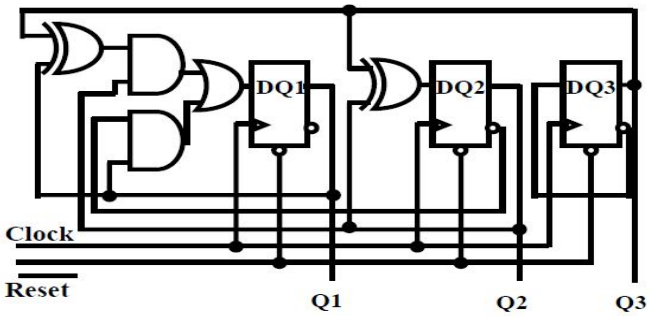


Fig 3 Exhaustive pattern generator

Exhaustive pattern BIST shown in Fig. 3 eliminates the test generation process and has very high fault coverage. To test an n-input block of combinational logic, it applies all possible 2^n-input patterns to the block. Even with high clock speeds, the time required to apply the patterns may make exhaustive pattern BIST impractical for a circuit with n>20.

In pseudo-exhaustive pattern generation, the circuit is partitioned into several smaller sub-circuits based on the output cones of influence, possibly overlapping blocks with fewer than *n* inputs. Then all possible test patterns are exhaustively applied to each

©IJRASET 2015: All Rights are Reserved

sub-circuit. In pseudo-random pattern generation a string of 0's and 1's is called a pseudo-random binary sequence when the bits appear to be random in the local sense, but they are in someway repeatable.[3][6] The linear feedback shift register (LFSR) pattern generator is most commonly used for pseudo-random pattern generation**.** [7]

IV. COMPARISON OF TEST GENERATION STRATEGIES

Implementing a BIST strategy, the main issues are fault coverage, hardware overhead, test time overhead, and design effort. These four issues have very complicated relationship. Table I summarizes the characteristics of the test strategies mentioned earlier based on the four issues.

TABLE I: COMPARISON OF DIFFERENT TEST STRETEGIES

Test Generation Methodology	Fault Coverage	Hardware Overhead	Test Time Overhead	Design Effort
Stored Pattern	High	High	Short	Large
Exhaustive	High	Low	Long	Small
Pseudo-Exhaustive	High	High	Medium	Large
Pseudo-Random	Low	Low	Long	Small
Weighted Pseudo-Random	Medium	Medium	Long	Medium

V. DESIGN AND IMPLEMENTATION

Pseudo exhaustive test pattern generators provide for complete fault coverage without the need for fault simulation. In pseudo exhaustive less number of test patterns is used.

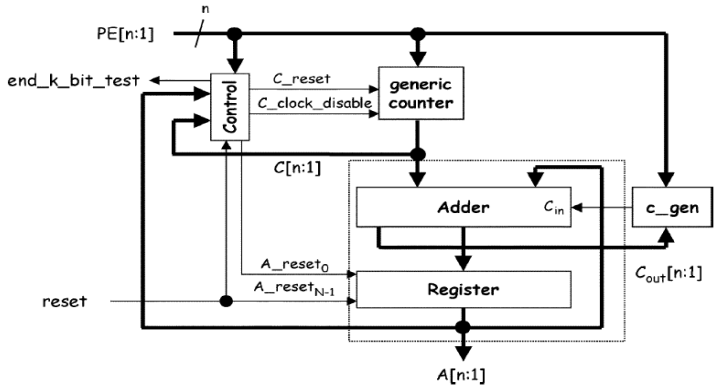


Fig 4 Generic exhaustive pattern generator

In Fig 4, a generic pseudo-exhaustive two-pattern generator is a module with n inputs (PE [n:1]) and n outputs (A [n: 1]) that can generator a two pattern (n, k) pseudo exhaustive test set for any value of k, (k < n).

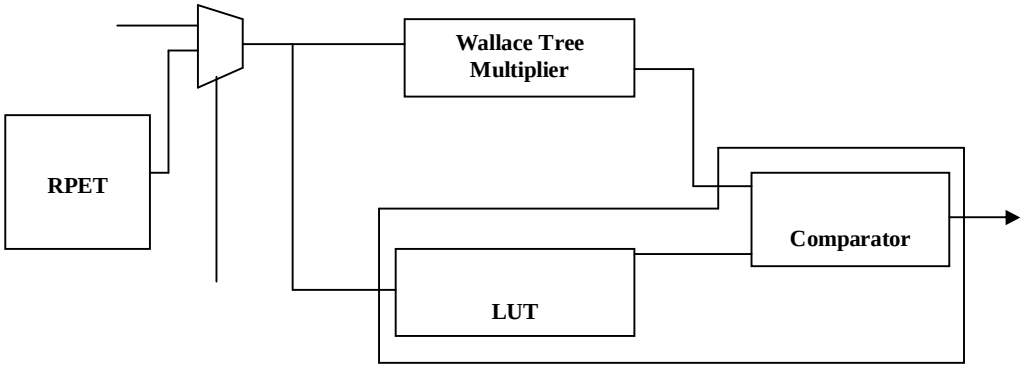


Fig 5 Proposed Block Diagram of BIST

©IJRASET 2015: All Rights are Reserved

The proposed BIST block diagram shown in the Fig 5 consist of generic pseudo exhaustive two pattern generator (GPET), Wallace tree, comparator, look up table (LUT), multiplexer. In the proposed block diagram of BIST as we see there is 2x1 mux whose inputs are 7bit-GPET and the 7bit-user input i.e. the input given directly by the user there is a select line in the multiplexer which decides which function is going to be perform, if test i.e. 1 is selected then the testing operation is perform and the input goes from the GPET and if the Run i.e. 0 is selected then the input is given by the user. The 7bit input goes to 7bit-Wallace tree which perform the 4x3 multiplication operation and that 7bit input goes to the look up table which already contain the binary value of $2^7 = 128$ after multiplication the values of the Wallace tree and the look up table is compared by the comparator if the output were the same the fault bit is 0 otherwise it will be 1. [6]

VI. TOOL USED FOR DESIGN

We used Xilinx ISE 9.2i for our programming and considered VHDL as our primary language. A brief description of VHDL and tools has been employed on this work to evaluate a perfect design structure which is the key for achieving the aim of this work. Xilinx XST is used to synthesis the various modules.

Using a Hardware Description Language (HDL) allows you to test different design implementations early in the design flow. Use the synthesis tool to perform the logic synthesis and optimization into gates. Xilinx® FPGA devices allow you to implement your design at your computer. Since the synthesis time is short, you have more time to explore different architectural possibilities at the Register Transfer Level (RTL) .You can reprogram Xilinx FPGA devices to test several design implementations.[5][7]

VII. SIMULATION RESULTS

TABLE 2

RECURSIVE PSEUDOEXHAUSTIVE TWO-PATTERN GENERATORS COMPARISON

Scheme	Exiting modules	Transformation	Gate equivalent
[6]	Two n-stage register	Transform two registers into counters + XOR gates + n 2-input multiplexer + m-stage group counter + Detect	7 x n +XOR gates + 3 x n + 8 x m
[7]	Two n-stage register	Transform two registers into + n 2-input multiplexer + m-stage group counter + Detect	7 x n +XOR gates + 3 x n + 8 x m
[5]	Accumulator + register	m-stage counter + m-to-n decoder + Control + Detect + c_gen + Transform registers into generic counters	18 x n + 8 x m
[Proposed]	Accumulator + Counter	m-stage counter + m-to-n decoder + Control +c_gen + generic counters	15 x n + 8 x m

VIII. CONCLUSIONS

BIST plays a vital role in modern VLSI technology.BIST two pattern generators play an increasingly important role in testing VLSI circuits and systems, due to a number of reasons, including higher coverage of sequential faults and assurance of correct temporal behaviour. By the simulation results on Table II, conclude that exhaustive and pseudo-exhaustive test generation schemes are commonly utilized to provide for extremely high fault coverage in combinational circuits. However, with pseudo-exhaustive generators, test generation time is greatly reduced. The BIST should occupy less area for compact design of digital circuit.

IX. ACKNOWLEDGMENT

I am deeply grateful to my guide Prof. Praveen Tiwari and Prof. Sobhit Verma for their valuable discussions and theoretical insights.

REFERENCES

[1] K. Nivitha1, Anita Titus “A BIST Circuit for Fault Detection Using Recursive Pseudo-Exhaustive Two Pattern Generator” International Journal of

©IJRASET 2015: All Rights are Reserved

Modern Engineering Research (IJMER) www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-676-681

[2] IoannisVoyiatzis, Dimitris Gizopoulos and Antonis Paschalis, "Recursive Pseudo-Exhaustive Two Pattern Generation," IEEE trans. Very Large Scale Integration (VLSI) sys, vol. 18, no. 1, Jan 2010.

[3] IoannisVoyiatzis, Accumulator-based pseudo-exhaustive two-pattern generation. Department of Informatics, Technological Educational Institute of Athens, Greece, Science Direct 20 February 2007.

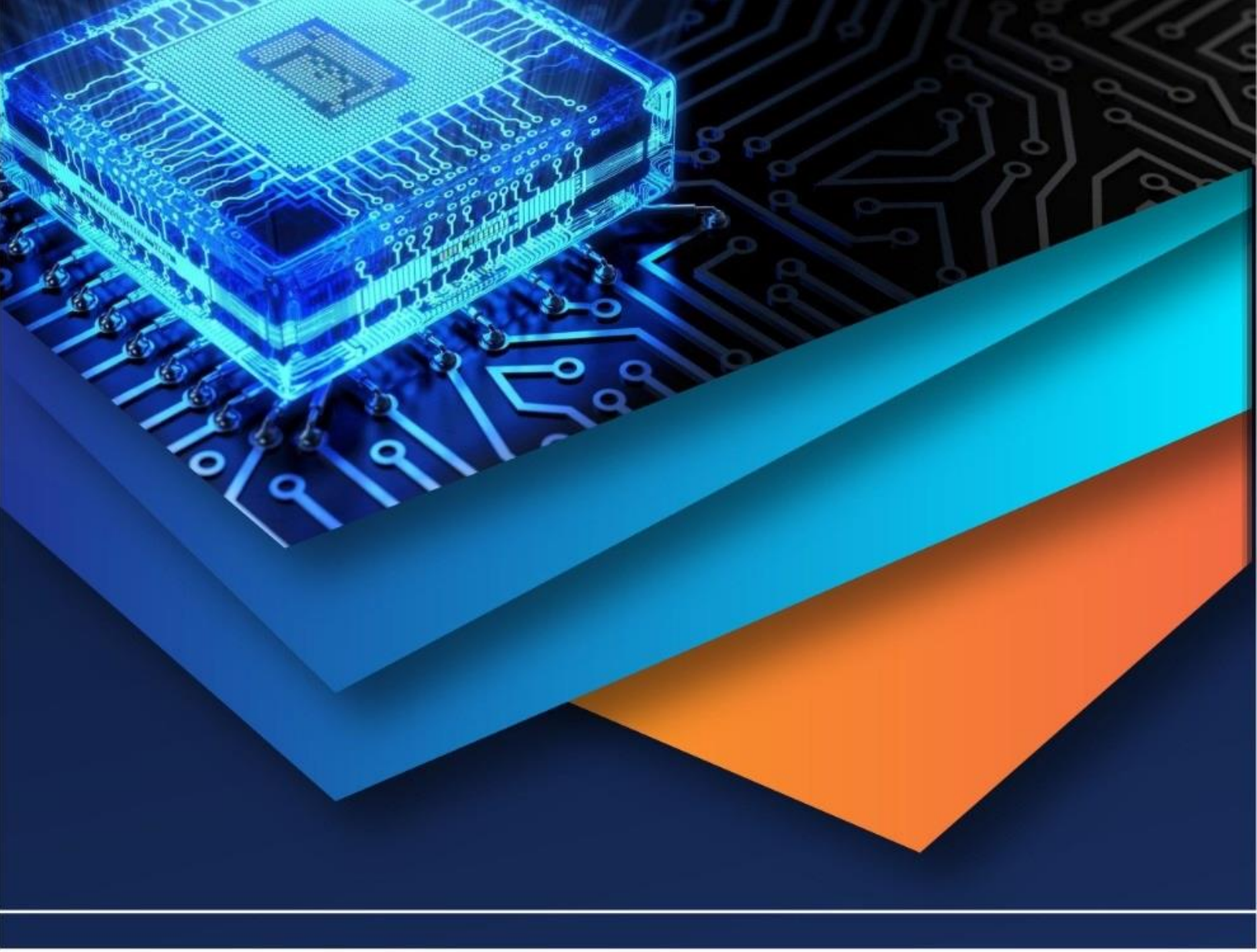
[4] P. Dasgupta, S. Chattopadhyay, P. P. Chaudhuri, and I. Sengupta, "Cellular automata-based recursive pseudo-exhaustive test pattern generation IEEE Trans. Comput., vol. 50, no. 2, pp. 177–185, Feb. 2001.

[5] Dr. V.K. Agrawal, Group Director, ISRO, Bangalore, Asst Professor, Department of ECE, DSCE, Bangalore, "Implementation of BIST Structure using VHDL for VLSI Circuits"International Journal of Engineering Science and Technology (IJEST) 2009.

[6] R. Srinivasan, S. K. Gupta, and M. Breuer, "Bounds on pseudoexhaustive test lengths," IEEE Trans. Very Large Scale Integr. (VLSI) Syst.,vol. 6, no. 3, Sep. 1998.

[7] VOYIATZIS, A. PASCHALIS, D. NIKOLOS, JOURNAL OF ELECTRONIC TESTING: "An Accumulator-Based BIST Approach for Two-Pattern Testing" Theory and Applications 15, 267–278 (1999) 1999 Kluwer Academic Publishers. Manufactured in The Netherlands.

©IJRASET 2015: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)