Controlled Transition Density Based Power Constrained Scan-BIST with Reduced Test Time

by

Farhana Rashid

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
May 7, 2012

Keywords: Scan-BIST, Transition density, Weighted Random Patterns, Test time

Copyright 2012 by Farhana Rashid

Approved by

Vishwani D. Agrawal, Chair, James J. Danaher Professor of Electrical and Computer Engineering
Adit D. Singh, James B. Davis Professor of Electrical and Computer Engineering
Victor P. Nelson, Professor of Electrical and Computer Engineering

Abstract

Controlling power dissipation in large circuits during test sessions is one of the major concerns in VLSI testing. The reason behind the high power dissipation during test is because unlike normal mode operation of the system correlation between consecutive test patterns does not exist in test mode.

To increase the correlation between consecutive vectors during testing, several techniques have been proposed for creating low transition density in the pattern sets and thus control the power dissipation. However, this in turn increases the test application time as the test has to run for longer test sessions to reach sufficient fault coverage. Increase in test time is also undesirable.

This research aims to provide a common way to deal with both the problems by optimizing test lengths for power constraint scan BIST circuits and reduce required test application time. It has been shown that a specific weight or transition density results in producing effective test with shortest test length for a given fault coverage. Thus the test length is optimized to reduce test application time. .Test time is further reduced by adapting the scan clock dynamically based on the transition density of the pattern set staying within power budget. A new pattern generator has been proposed to produce the test patterns of desired properties. Finally we propose a greedy algorithm for mixing various transition densities to reduce the test application time further without sacrificing the fault coverage. Time saving up to 43% has been seen in this proposed method in ISCAS89 circuits.

Acknowledgments

I would like to convey my heartiest gratitude to my supervisor, Dr. Vishwani D. Agrawal, the James J. Danaher Professor at Electrical Engineering Department. Without his patience and guidance this thesis would not have been possible. I honestly appreciate his not giving up on me when I myself lost confidence while working through the research.

I thank my advisory committee members, Dr. Victor P. Nelson and Dr. Adit D. Singh for their valuable suggestions and concern towards my work. I highly appreciate Dr. Charles Stroud for helping me running his AUSIM simulator on Auburn University's High Performance Cluster Computer and letting me audit BIST course.

I sincerely appreciate Auburn University Network Services for partly supporting my graduate study and my colleagues Shannon Price, Zebediah Whitehead, Jeffery Walker for always being so supportive. It was a great learning experience working with these amazing people.

Graduate study at Auburn University has been a pleasure. I am thankful to my friends and colleagues for always being ready to discuss and give opinions whenever I was in doubt. Without them my graduate experience would not have been so enjoyable. I am indebted to my parents, brothers and sisters for all their love and support. Also I am very thankful to my husband, Abdullah Al Owahid, for being with me in every step of life.

Finally, I thank everyone who directly or indirectly helped me during the course of graduate study here in Auburn.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ASIC	Application Specific Integrated Circuit
ATPG	Automatic Test Program Generator
BIST	Built-in-Self-Test
BS-LFSR	Bit-Swapping LFSR
CA	Cellular Automata
CMOS	Complementary Metal Oxide Semiconductor
CUT	Circuit Under Test
DFT	Design for Testability
DS-LFSR	Dual-speed LFSR
IC	Integrated Circuit
ISCAS	International Symposium on Circuits and Systems
LFSR	Linear Feedback Shift Register
LT-LFSR	Low Transition LFSR
PRESTO	Pre-Selected Toggling
RI-LFSR	Random Bit Injection LFSR
SAR	Signature Analysis Register
SoC	System-on-a-Chip

TPG	Test Pattern Generator
VLSI	Very Large Scale Integration

Chapter 1

Introduction

Test power and test application time are the two major challenges in today's VLSI design and test area. Time spent in the expensive tester machines directly contributes to the cost of a chip. Also for self testing circuits shorter test application time is desirable. Due to advancement of technology circuit size has increased which naturally claims longer test time. On the other hand, the test process causes higher power dissipation in the circuits compared to the power dissipated in the normal mode of the circuit. The excess power dissipation gives rise to many problems like hot spots, chip failure, performance degradation; testing may even dramatically shorten the battery life when on-line testing is involved [15, 20].

Many techniques have proposed to tackle these two issues separately by DFT engineers. For reducing test power one of the widely used technique is to decrease the clock frequency used for testing. This is very often used for scan testing, a more popular method in DFT. This method, however, is responsible for making the test application time longer. One other method for reducing power consumption in the test session is to use test vectors created that aim for lower switching activity in the circuit during testing. This method is widely used if circuits are being self-tested and the vectors are produced on chip. The down side of this technique is that it requires longer test sequences to achieve targeted fault coverage. This, once again, results in longer test time.

More DFT techniques, like test compression are needed to make the test process faster by reducing test vector application time [41].

A method that can contribute to both the causes mentioned above is the motivation behind this work.

1.1 Problem Statement

The aim of this work is to:

- *Analyze the effect of transition density on fault coverage*
- *Deploy an effective test generation process using the information from the analysis*
- *Adapt the scan frequency to the transition density for power constrained testing*

1.2 Thesis Contribution

The unique contribution of the work presented in this thesis is:

- Determination of best transition density in a vector set to achieve a target fault coverage with the shortest test length
- Adjust the scan frequency according to the transition density for a power constrained scan-BIST circuit to speed up the test in multiple scan chains
- Deployment of a variable transition density test pattern generator in a BIST circuit that is capable of producing pre-selected transition density vectors
- Reduction of test application time further by adapting the scan clock to the pre-selected transition density
- A greedy algorithm to construct a vector set with mixed transition density in a controlled way, aimed to effectively apply test in a power constrained setup for scan -BIST with reduced test application time

1.3 Thesis Organization

Chapter 2 of the thesis introduces readers to various concepts that are relevant for understanding the significance of the problems solved by the proposed work. Chapter 3 describes briefly the prior work done to reduce both test application time and power dissipation during test. Chapter 4 analyzes the effect on fault coverage when transition density in a vector set is modified from the conventional transition density that is present in random or pseudo-random patterns. Chapter 5 presents a technique to dynamically adjust the scan clock based on the transition density present in the vector sets with test compression technique. Chapter 6 utilizes the information from chapter 4 and chapter 5 to combine the benefits in a BIST implementation. Here we present a modified test pattern generator (TPG) that has a capability to produce a desired transition density in a vector set. Thus a refined scheme to adjust the scan clock adaptively to reduce the test application time is described in this chapter. Chapter 7 proposes an algorithm to select among transition densities to construct a controlled transition density based vector set for further reduction of test time. Chapter 8 discusses the experimental results obtained from the implementation of the algorithm on different benchmark circuits. Finally, chapter 9 concludes the work with suggestion for future research.

Chapter 2

Background

This chapter discusses the background information for a better understanding of the work presented in this thesis. The first section provides some definitions of the terms used in this work, the second section explains the power dissipation during test and its consequences, the third section briefly discusses DFT techniques and finally the last section gives an overview of different test pattern generation methods.

2.1 Definitions

2.1.1 Transition Density

The transition density T, of a logic signal $N(t)$ is defined as number of transitions per unit time, i.e., $T = \frac{N(t)}{t}$. For a continuous signal, transition density $T = Lim_{t \to \infty} \frac{N(t)}{t}$ [30].

Thus, the transition density of a clock signal is two, according to this definition, as there are two transitions, one rising and the other falling in a unit time (which is one clock period).

2.1.2 Static Signal Probability

Viewing a signal as a random process [31] and observing it for a time interval $t0 + t1$, where signal remains 1 for duration $t1$ and the signal remains 0 for duration $t0$, then the probability of the signal being 1, is given by

$$p1 = \frac{t1}{t1 + t0} \tag{2.1}$$

And the probability of the signal being 0 is given by

$$p0 = \frac{t0}{t1 + t0} = 1 - p1 \tag{2.2}$$

2.1.3 Transition Power

The power consumption in CMOS circuits can be classified into static and dynamic power [46]. Leakage current or other current that is drawn continuously from power supply causes static power dissipation. Dynamic dissipation occurs when switching occur either due to short circuit current or charging and discharging load capacitance.

The average energy consumed at node i due to switching is given by E= $\frac{1}{2}C_iV_{dd}^2$, where C_i is the equivalent output capacitance and V_{dd} is power supply voltage. Therefore, transition power dissipation is given by,

$$P = E\alpha f = \frac{1}{2}C_iV_{dd}^2\alpha f \tag{2.3}$$

where α is transition density and f is clock frequency.

2.2 Power Dissipation During Test

This section describes different power consumptions in a CMOS circuit that is relevant to testing and then discusses the reason why test power is needed to be controlled during testing.

Let us assume that the energy consumption of a circuit after application of successive input vectors (V_{k-1},V_k) is $E_{vk} = \frac{1}{2}\mathrm{C}V_{dd}^2\alpha_k$, where, α_k is the number of switching in the circuit that occurred due to application of the vector V_k. Therefore, for a pseudorandom test sequence of length L, where the test length is determined from the number of vectors to reach a targeted fault coverage, the total energy consumed in the circuit during application of the complete test sequence is given by, $E_{total} = \frac{1}{2}\mathrm{C}V_{dd}^2\sum\alpha_k$.

If f_{ck} denotes the clock frequency then instantaneous power consumed in the circuit after application of vectors (V_{k-1},V_k) is given by, $P_{inst}(V_k) = E_{vk}f_{ck}$. This is because, by definition, the instantaneous power is the consumed power during one clock period [16, 20].

The peak power consumption corresponds to the maximum instantaneous power consumed during the test session. It therefore, corresponds to the highest energy consumed during one clock period, multiplied by clock frequency.

Also, the average power consumed during the test session is the total energy multiplied by the test time.

$$P_{avg}= E_{total} * \frac{f_{ck}}{Length,L}$$

According to the expressions of power and energy consumption mentioned above and assuming a given CMOS technology and supply voltage for the circuit design, number of the switching in the circuit caused by applying a test vector is the only parameter that affects the energy, peak power and average power consumption. The clock frequency used during testing affects both the peak power and average power and the test length which is the number of the patterns applied to the circuit under the test (CUT) affects only the total energy consumption.

It is important when dealing with high density systems such as modern ASICs and SOCs, to design tests for these circuits that are non-destructive . But excessive switching activity during tests leads to increased current flow in the circuit, resulting in circuit failures due to altered electromigration, increase in cost for packaging, decreased circuit reliability, and autonomy of battery powered remote and portable system.

For circuits that have BIST circuitry incorporated within them, switching activity during test session is a major concern. Therefore, many low power or low transition based BIST techniques, especially for scan BIST have been proposed by researchers.

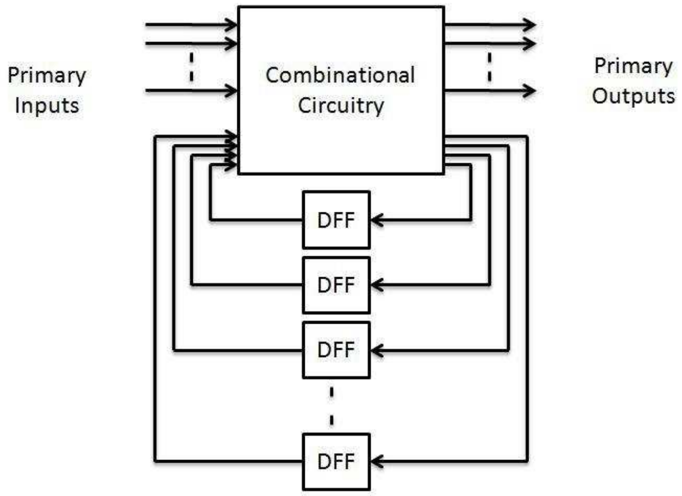

Figure 2.1: Architecture of a sequential circuit.

Low transitions in test vectors, on the other hand, tends to increase the test length resulting in increased test time that is required to apply the longer test sequence.

Therefore we propose in our work a test scheme for controlled transition based test application that will optimize test length and speed up test without exceeding the power budget of the circuit.

2.3 Design for Testability (DFT) Techniques

This section briefly describes the two DFT techniques that are widely used. As the size of the circuit increases, test complexity also increases. Their internal nodes become harder to test. Circuits are therefore modified so that they can be tested effectively [8].

2.3.1 Scan Design

Sequential circuits are harder to test than combinational circuits. This is because the presence of memory elements, as shown in Figure 2.1, which creates internal states during circuit operation. An exhaustive test would involve application of all possible input vectors at all possible states of the memory elements.

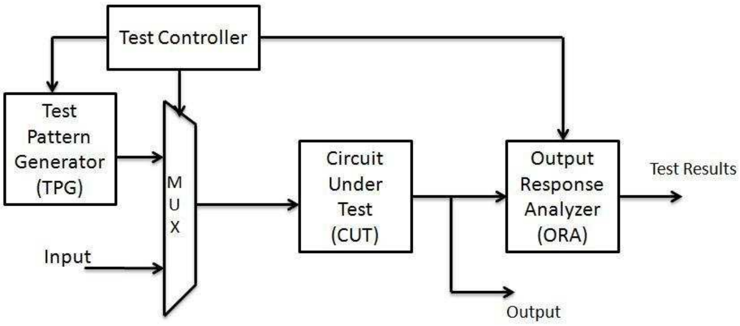

Figure 2.2: Basic BIST circuitry.

For a circuit with n inputs there are 2^n possible input combinations. As n increases the number of possible input vectors increases exponentially. This phenomenon is even more severe for sequential circuits.

The DFT technique that seeks to improve testability of sequential circuits is scan design [8] or its partial scan variations [5, 8]. Here the sequential circuit is modified such that it can operate in test mode. When the circuit is in test mode, the flip-flops in the circuit are chained together to form one or more shift registers. The flip-flops serve as a point of controllability and observability and help achieve better test coverage.

2.3.2 Built-In Self-Test (BIST)

BIST is a DFT technique in which additional hardware is added to the circuit to be tested so that it can test itself [6, 8, 38]. The basic BIST circuitry is shown in Figure 2.2. The patterns required for test are generated using various techniques. Among them, the use of a Linear Feedback Shift Register (LFSR) that generates pseudorandom pattern sets is most common.

A large number of outputs are received from the circuit under test. It is necessary to store the correct values of all those bits without adding a lot of extra hardware. This in turn calls for some more design techniques. A LFSR, most commonly known as Signature Analysis Register (SAR) or Multiple Input Signature Register (MISR) is used for this purpose.

Test-Per-Clock BIST Systems: In this type of system, a test is applied every clock cycle, i.e., a new set of faults is tested in every clock cycle. This type of system has short pattern lengths. A major concern for BIST is the simulation time required to compute good circuit behavior. It is therefore advantageous to have short pattern lengths.

Test-Per-Scan BIST Systems: In test-per-scan BIST, each test comprises scan-in of one input vector, one clock to conduct the test and scan-out of output responses. This type of system therefore requires a larger test time. Also, it involves larger simulation time than in test-per-clock BIST systems due to the longer pattern lengths.

2.4 Types of Test Patterns

This sections briefly describes types of test patterns as based on the test pattern generators are classified [38].

1. Deterministic test patterns are developed to detect specific faults and/or structural defects for a given CUT. An example of hardware for applying deterministic vectors would include a ROM with a counter for addressing thr ROM. This type of approach has limited applicability for BIST. This approach is often referred to as stored test patterns in the context of BIST applications.

2. Algorithmic test patterns are similar to deterministic test patterns in that they are specific to a given CUT and are developed to detect specific fault models in the CUT. However, because of repetition and/or sequence typically associated with algorithmic test patterns, the hardware for generating algorithmic vectors is usually a finite state machine. There are considerable applicability of this test pattern generation approach to BIST for regular structure such as RAMs.

3. Exhaustive test patterns produce every possible combination of input test patterns. In case of an N-input combinational logic circuit where an N-bit counter produces all possible 2^N test patterns and will detect all detectable gate level stuck-at faults. Exhaustive test patterns are not practical for large N.

4. Pseudo-exhaustive test patterns are an alternative to exhaustive test patterns. In this case, each partitioned combinational logic sub-circuit will be exhaustively tested. Each K-input sub circuit receives all 2^K possible patterns, where $K < N$.

5. Pseudo-random patterns are most commonly produced patterns by TPG hardware found in BIST applications. These patterns have properties similar to those of random pattern sequences but the sequences are repeatable.

6. Weighted-random test patterns are good for circuits that contain random pattern resistant faults. This type of pattern generation uses an LFSR or cellular automata (CA) to generate pseudo-random test patterns and then filters the patterns with combinations of AND/NAND gates or OR/NOR gates to produce more logic 0s or logic 1s in the test patterns applied to the CUT. The number of 1s (or 0s) are referred to as the weight of the vectors.

7. Lastly, random patterns have frequently been used for external functional testing of microprocessors as well as in ATPG software.

Chapter 3

Previous Work

This chapter presents previous work that has been done in the area of low power testing, especially techniques involved in reducing transitions or toggles to reduce switching activity during the test. The first section summarizes the work done for controlling power dissipation during testing circuits with BIST circuitry. The second section gives brief descriptions of the work done to reduce the test application time in scan testing.

3.1 Reducing Test Power in BIST Circuits

Reduction of test power is a widely recognized problem, as described earlier, and therefore a number of solutions have been presented. Girard summarizes the different techniques proposed for low-power testing of VLSI circuits [16, 20]. They are broadly classified into low-power external testing techniques and low-power BIST techniques. This section focuses on the low power techniques proposed for BIST.

3.1.1 New Test Pattern Generators

Test pattern generators have been modified to reduce the power that is generated during test because of low correlated test vectors. This section discusses some of those techniques briefly.

Tehranipur et al. proposed a low transition BIST pattern generator called LT-LFSR, to reduce average and peak power of a circuit during test by reducing transitions within random test patterns and between consecutive patterns [39]. The proposed LT-LFSR reduced transitions by inserting intermediate vectors between two consecutive vectors generated by the LFSR. This was done by combining properties of two different LFSRs, the Bipartite

LFSR and the Random Bit Injection LFSR (RI-LFSR). The experimental results showed four ISCAS benchmark circuits up to 77% and 49% reduction in average and peak power respectively. However, the test length increased to achieve targeted fault coverage while using this method.

Abu-Issa and Quigley proposed a novel low transition LFSR, called Bit-Swapping LFSR (BS-LFSR) which consisted of an LFSR and 2-to-1 multiplexer [1]. They have showed that when BS-LFSR was used to generate test patterns for scan-based BIST, it reduced the number of transitions that occur at the scan chain input during scan shift operations by 50% when compared to those patterns produced by a conventional LFSR. Thus they reduced the overall switching activity in the circuit under test during test application. They also combined the BS-LFSR with a scan chain ordering algorithm that ordered the cells in a way that reduced the average and peak (scan and capture) in the test cycle or while scanning out a response to a signature analyzer. Result showed up to 65% and 55% reduction in average and peak power, respectively, with negligible effect on fault coverage or test application time.

Wang proposed a low hardware overhead test pattern generator (TPG) for scan-based BIST that reduced switching activity along with achieving very high fault coverage with a reasonable test length [43]. The proposed TPG comprised two TPGs LT-RTPG(low transition random TPG) and 3-weight WRBIST (weighted random BIST) TPG, where the LT-RTPG generated patterns for easy to detect faults and test patterns generated by the 3-weight WRBIST detects faults that remained undetected after LT-RTPG patterns was applied. Close to 100% fault coverages for ISCAS benchmark circuits were seen with significantly reduced activity during test sessions.

Rajski et al. proposed a pseudorandom test pattern generator with pre-selected toggling (PRESTO) activity that comprised a finite state machine, a pattern generator, appropriate phase shifter. The experimental results for eight industry standard circuits showed reduced switching activity with a cost of increased test length [25].

Wang and Gupta proposed a test pattern generator for BIST called dual-speed LFSR (DS-LFSR), aiming to reduce heat dissipation during test application [44]. As the name implies, the dual-speed LFSR (DS-LFSR) consisted two LFSRs, a slow LFSR and a normal speed LFSR. The inputs of the circuit under test were provided through the slow LFSR in order to reduce the transition density at the inputs, which resulted in reduced heat dissipation during test. A procedure was introduced to design a DS-LFSR such that high fault coverage was achieved through unique and uniformly distributed patterns. New methods of selecting inputs driven by the slow LFSR and increasing the number of inputs driven by the slow LFSR were presented. Reductions of 13% to 70% in the number of transitions were observed for ISCAS benchmark circuits without loss of fault coverage using this method.

A Cellular Automata based Test Pattern Generator (TPG) was proposed by Corno et al. to test combinational circuits. The TPG was designed to reduce power consumption while achieving high fault coverage [12]. An algorithm was presented here that selected an optimal non-linear hybrid cellular automaton (HCA) based on power consumption for given coverage and test length constraints. Experimental results showed an average test power reduction of 34% without affecting fault coverage, test length and area overhead.

Girard et al. presented a low power test-per-clock BIST test pattern generator (TPG) that generated test vectors capable of reducing the switching activity during test [19]. The technique was based on a modified clock scheme and the clock tree feeding the TPG. Therefore, this method reduced test power in the TPG and clock tree in addition to power reduction in the circuit under test (CUT). Reductions of up to 60% and 61% were noted in power and energy when the proposed technique was implemented on ISCAS benchmark circuits.

Zhang, Roy and Bhawmik proposed a modified LFSR by adding weight sets to tune the pseudorandom vector's signal probability in order to achieve increased fault coverage but with reduced energy consumption [48]. A tool, POWERTEST was developed which used a genetic algorithm based search to determine optimal weight sets at primary inputs to minimize energy dissipations. Results on ISCAS benchmark circuits showed an energy reduction of up to 97.82% while still achieving high fault coverage.

Wang and Gupta presented a new BIST TPG design, called low-transition random TPG (LT-RTPG) that comprised an LFSR, a k-input AND gate, and a T flip-flop [45]. The LT-RTPG generated test patterns for test-per-scan BIST that decreased the number of transitions that occurred during scan shifting and thus decreased the heat dissipation during testing. The new TPG reduced the number of transitions in ISCAS89 benchmark circuits by 23% to 59%.

Gizopoulos et al. proposed low power BIST schemes for datapath architectures built around multiplier-accumulator pairs, based on deterministic test patterns [21]. They have also proposed two alternatives based on whether the design is low energy dissipation or low power dissipation during a BIST session. Both methods are based on modified binary counters, operating as Gray counters. The technique offers up to 78.33% energy saving and up to 82.22% power saving compared with pseudorandom BIST.

3.1.2 Test Scheduling Algorithms

Test scheduling techniques have been proposed by different researchers to control the test power for complex ICs.

Zorian presented a technique which consists of a distributed BIST control scheme [21]. The process included a BIST control methodology that implemented the BIST schedule with a highly modular architecture. The control architecture provided an autonomous BIST activation and a diagnostic capability to identify failed blocks. The technique reduces average power and hence avoids temperature related problems but with the cost of increased test time.

Chou, Saluja and Agrawal presented an optimum test scheduling algorithms for equal and unequal test length cases under power constraints [11, 10, 26]. Their algorithms find the optimum solution by first constructing a test compatibility graph from a resource graph, and then using the compatibility graph to identify time compatible tests with power information associated with each test, followed by identifying power compatible tests among the time compatible tests. Finally the optimal scheduling of the tests was found using a minimum cover table approach. This algorithm reduces the average power consumption.

Iyengar and Chakrabarty proposed an integrated framework to determine optimal SOC test schedules [24]. They also proposed a new algorithm that used preemption to obtain optimal test schedules in polynomial computation time.

3.1.3 Toggle Suppression

The toggle reduction technique involves the suppression of toggles in the circuit during test. This reduces the net activity and hence the power dissipation during test.

Hertwig and Wunderlich introduced a low power technique for scan-based BIST architectures that modified the scan-path structure's scan cells such that the inputs to the CUT remained unchanged during shift operations [23]. Energy savings of up to 90% were seen in a standard, scan-based BIST architecture.

3.1.4 LFSR Tuning

Girard et al. proposed a technique to minimize the energy required to test combinational circuits with BIST without altering fault coverage [18]. They have analyzed the impact of the polynomial and seed selection of the LFSR used as TPG on the energy consumed by the circuit and found that appropriate selection of the seed of the LFSR can contribute to energy reduction whereas the polynomial selection does not affect the power consumption. A heuristic based on a simulated annealing algorithm was proposed to decrease the energy consumption of BIST runs.

3.1.5 Vector Filtering BIST

Not all the patterns generated by the TPG contribute to the fault detection. Therefore, a number of works aimed to reduce the energy consumed during test by filtering out the non-detecting vectors. Girard et al. proposed a test vector inhibiting technique to tackle the increased activity during test operation [17]. A mixed solution based on a reseeding scheme and the vector inhibiting technique was also proposed in order to deal with hard-to-test circuits that contain pseudo-random resistant faults. The technique reduced the total energy consumption during test and allowed the test at system speed in order to achieve high delay fault coverage. Experimental results showed weighted switching activity reductions ranging from 18.5% to 78.5% without loss of stuck-at fault coverage.

As the test progresses the detection efficiency of the pseudo-random vectors decreases. The number of pseudo-random vectors that will not detect previously undetected faults increases. These vectors consume energy without contributing to fault coverage. This fact was used by Manich et al. to propose two techniques to reduce the energy and average power consumption of the system [28]. The first technique filters all the non-detecting subsequences and the second technique uses reseeding which is an extension of the technique used in [18]. Energy and average power consumption savings up to 90% have been observed while applying these two techniques in ISCAS benchmark circuits.

Gerstendörfer and Wunderlich used the technique of filtering non-detecting patterns for scan-based BIST architectures, combined with Hertwig and Wunderlich to avoid scan path activity during scan shifting [14, 23]. The modules and modes with the highest power consumption were identified and design modifications to reduce power consumption were proposed. The proposed modifications reduced the test power by several orders of magnitude with nominal cost in terms of area and performance penalties.

3.2 Reduction in Test Time

Shanmugasundaram and Agrawal proposed a dynamic scan clock control scheme in scan testing to reduce test time while maintaining peak power limit [33, 35, 34, 36]. Per cycle scan activity is monitored in the scan chain to speed up the scan clock for low activity cycles without exceeding the specified peak power budget.

The scheme was based on the fact that not every vector has the highest activity and hence can be scanned-in using a faster test clock without exceeding the power budget. The power P dissipated at a node is given by

$$P = \frac{1}{2}CV^2\alpha f \tag{3.1}$$

where C is the capacitance of the node, V is supply voltage, f is clock frequency and α is node activity factor.

In the worst case, scan clock frequency f_{test} can be can be determined based on the maximum activity $\alpha = 1$, so that the test power can never exceed the power limit. Therefore,

$$P_{budget} = \frac{1}{2}CV^2 f_{test} \tag{3.2}$$

and,

$$f_{test} = \frac{2P_{budget}}{CV^2} \tag{3.3}$$

In general, the worst case assumption can be modified for any value α_{node}. All vectors are scanned in and scanned out at this frequency. However, most vectors do not cause the maximum activity in the circuit and will dissipate much lower power than the allowed limit. It is possible to scan in these vectors at higher clock frequencies without exceeding power budget.

Excepting the worst case, if the number of transitions in the circuit reduces to fraction $\frac{1}{i}$ of the maximum number of transitions, then power consumption is reduced. Given that the power should not exceed P_{budget}, we can increase the test clock frequency to $f_{test} \times i$. That is,

$$\text{Actual Power} = \frac{1}{2}CV^2 f_{test} \times \frac{1}{i} \leq P_{budget} \tag{3.4}$$

Since the capacitance and the voltage are constant for a node, the power is proportional to the product of activity and frequency.

The authors showed that if the activity reduces to $\frac{i}{i}$th fraction of the maximum α = 1, then the scan frequency can be increased to i times of the test frequency without causing the power to exceed power budget. The analytical results showed that for very low activity ($\alpha \approx 0.0$) test application time can potentially reduce by 50%. Experimental results showed up to 19% reduction in time in the largest ISCAS 89 circuit with 2-3% area overhead [33]. These experiments add scan chain activity monitoring and clock frequency adjustment hardware to test-per-scan BIST circuits with a single scan chain.

The work presented in this thesis uses the above method, further extending it to a test compression technique by breaking a single scan chain into multiple scan chains to reduce the test application time to limit the power consumption during scan activity.

Chapter 4

Transition Density and Its Effect on Fault Coverage

To keep the power consumption low while testing, low transition density test vectors are applied [7, 47]. This, in general, increases the test application time for achieving a target fault coverage. To study the effect of transition density of vectors on fault coverage a detailed analysis has been done. This chapter describes the variation in fault coverage due to different transition density selection and compared to fault coverage attained by weighted random patterns. A best case transition density is also determined from that analysis.

4.1 Weighted Random Pattern

Weighted random patterns have been used before to reduce test length for combinational circuits [2, 3, 4, 22, 32, 47]. Proper selection of the input probability can increase the efficiency of test vectors in detecting faults, resulting in reduced test time [27]. Therefore, to achieve higher fault coverage with shorter test lengths weighted pseudo random patterns are used [13]. For demonstrating the effectiveness of weighted pseudorandom test patterns, fault simulation was done on ISCAS89 benchmark circuits.

A Matlab [29] program was written to construct different test vector sets. Each of the sets contained 10,000 vectors but with different weights. Here, the weights are defined as the probability of a bit being 1 in a vector. The weights are varied from 0.1 to 0.95 at 0.05 intervals. Thus a total 18 sets of vectors are constructed for the weights 0.1, 0.15, 0.2, etc, up to 0.95.

Targeted fault coverage was set to 95% of the total faults and then fault simulation was done using the 18 different vector sets as mentioned earlier. In each case the number of vectors needed to reach the target fault coverage by each vector set was recorded. For

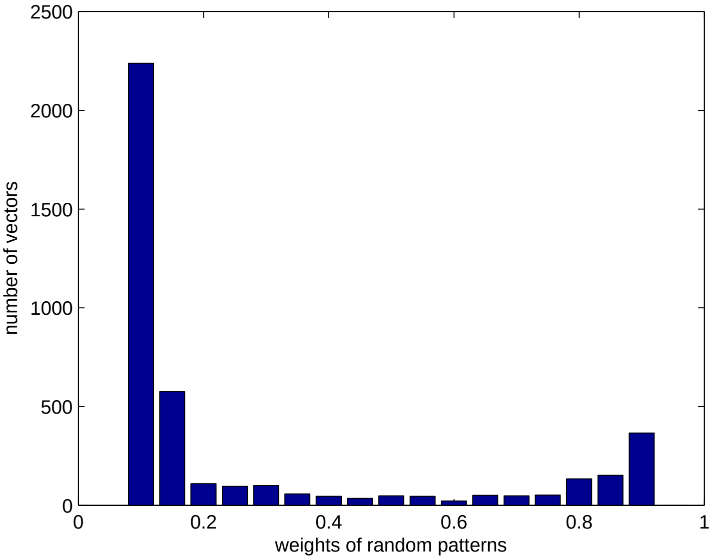

Figure 4.1: Number of test-per-scan vectors for 95% coverage in s1269 when 1-probability of scan-in bits was weighted.

every circuit that was simulated there exists one specific weight that resulted in shortest test length. The number of vectors obtained in this experiment for s1269 circuit as a function of the weight (probability of 1 in the scan-in bits) is shown in Figure 4.1. For this circuit the minimum is 22 vectors for a weight of 0.6.

4.2 Computing Best Case Transition Density from Best Case Weight

This section deals with the assumption of a best case transition density from the best case weighted random patterns. The transition density in an uncorrelated-bit sequence that has a 0-bit probability of $p0$ and 1-bit probability of $p1$ is given by $p0p1 + p1p0$ since a transition occurs when a 1 follows a 0 or a 0 follows a 1. However, $p0 = 1 - p1$, thus, the transition density can be calculated as:

$$TD = (1 - p1)p1 + p1(1 - p1) = 2p1(1 - p1) \tag{4.1}$$

Hence, from Figure 4.1, for circuit s1269, if best case weighted random pattern has a 1-bit probability of 0.6 then the corresponding transition density will be $2 \times 0.6 \times 0.4 = 0.48$.

This implies that if a test vector set is constructed to have a transition density of 0.48, then that vector set will generate an effective test for the circuit with shortest test length. In other words it can be assumed that a vector set of average transition density of 0.48 will result in detecting more faults with fewer vectors when compared to the numbers of vectors applied with transition densities higher or lower than 0.48.

4.3 Effect of Controlled Transition Density on Fault Coverage

If bits are generated randomly, the probabilities of generating a 1 or a 0 are equal, i.e., $p0 = p1 = 0.5$. Hence the transition density of the bit stream is also 0.5. To generate a transition density higher or lower than 0.5, bits must be generated with negative or positive correlation, respectively. Therefore, the bit stream will contain shorter runs of consecutive 1s or 0s for a transition density higher than 0.5 and longer runs of consecutive 1s or 0s for a transition density lower than 0.5.

A Matlab [29] program was written to generate test vector sets, each set containing 10000 vectors but with different transition densities. Here also the transition density was varied from 0.1 to 0.95, with 0.05 intervals. The vector set generated for 0.1 transition density has longer runs of 1s and 0s in consecutive bit positions. Likewise the vector set having transition density of 0.95 has very short runs of 1s and 0s in consecutive bit positions.

Target fault coverage was set to 95% of the total faults and then fault simulation was done using the 18 different vector sets as mentioned above. In each case number of vectors needed to reach the target fault coverage by each vector set was recorded. For every circuit that was simulated, there existed a best transition density (TD) that resulted in the shortest test length.

The same set of ISCAS89 benchmark circuits was again used for fault simulation. Table 4.1 shows the best case results obtained from fault simulation using AUSIM [37]. The

Table 4.1: Best case weighted random and transition density vectors for 95% fault coverage in ISCAS89 circuits obtained from fault simulation experiments.

Circuit name	Target FC (%)	Weighted random vectors		Transition density vectors		
		$p1$	No. of vectors	$2p1(1-p1)$	TD	No. of vectors
S298	77.1	0.6	18	0.48	0.55	423
S382	95	0.3	56	0.42	0.45	124
S510	95	0.4	136	0.48	0.5	152
S635	95	0.9	97	0.18	0.1	1883
S820	95	0.45	2872	0.495	0.45	5972
S1196	95	0.55	1706	0.495	0.45	2821
S1269	95	0.6	22	0.48	0.5	24
S1494	98.8	0.5	4974	0.5	0.45	3158
S1512	95	0.75	538	0.375	0.2	338

table shows the numbers of vectors that achieved 95% fault coverage. The third column gives the weighted random bit probability ($p1$) that required minimum number of vectors shown in column 4. In column 5, the probability $p1$ of column 3 is used to compute the transition density from equation 4.1.

The last two columns of Table 4.1 give the best case transition density (TD) and the corresponding number of vectors obtained from simulation. The differences in the transition densities of columns 5 and 6 can be because the two were obtained from two different statistical test samples. Also, equation 4.1, used for computing TD in column 5, assumes uncorrelated neighboring bits, an assumption that is yet to be validated.

Figure 4.2 shows a bar chart of the number of transition density vectors obtained from fault simulation experiments to reach 95% fault coverage in circuit s1269. A vector set generated with 0.5 transition density has the best fault detecting capability with smallest number (only 24) as compared with the other transition density vector sets.

However, unlike highly efficient weighted random patterns the patterns constructed based on transition density were not able to detect 100% of faults for some circuits. As shown in Figure 4.3, the weighted random patterns and the transition density based vectors do not always have the same effectiveness. Which is better, often depends upon the circuit.

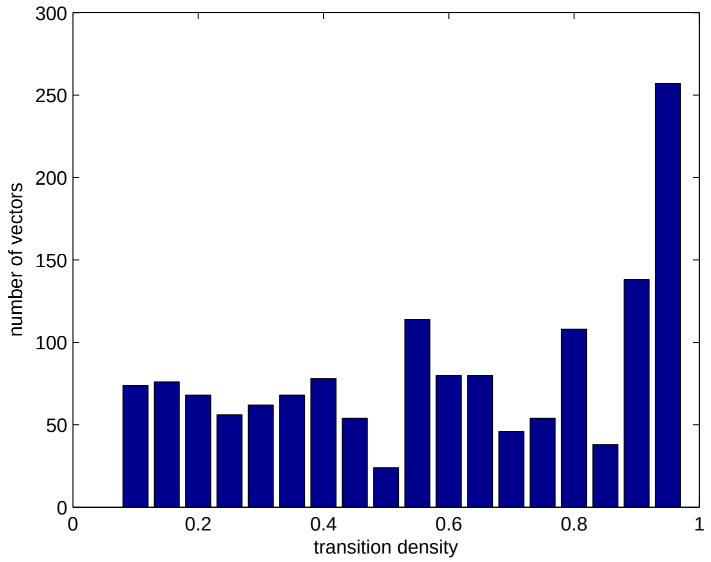

Figure 4.2: Number of test-per-scan vectors for 95% coverage in s1269 for various transition densities of scan-in bits.

While the generation of weighted random patterns is well understood, transition density patterns need further study.

Note weighted random bits have a transition density of their own. But our transition density patterns generated by the toggle flip-flop always have equal number of 0s and 1s. Though the transition density of weighted random bits for any $p1$ can never be higher than 0.5, the toggle flip-flop can produce transition densities greater than 0.5. Such patterns will produce high power consumption, which can be lowered by the adaptive test clock procedures [33, 34, 35, 36] as discussed in a later chapter, if the vectors gave accelerated fault coverage. This aspect needs additional study.

A more detailed analysis has been done on the fault coverage of ISCAS89 circuits and the results has been tabulated in a later chapter on experimental results. It may be noted that many techniques has been used to detect hard to detect faults in scan-BIST, such as vector reseeding, test point insertion, etc. We show here that to reach certain fault coverage, a specific weight (probability of a bit being 1) or a specific transition density (per vector transition probability that determines the number of transitions in a bit stream) can produce

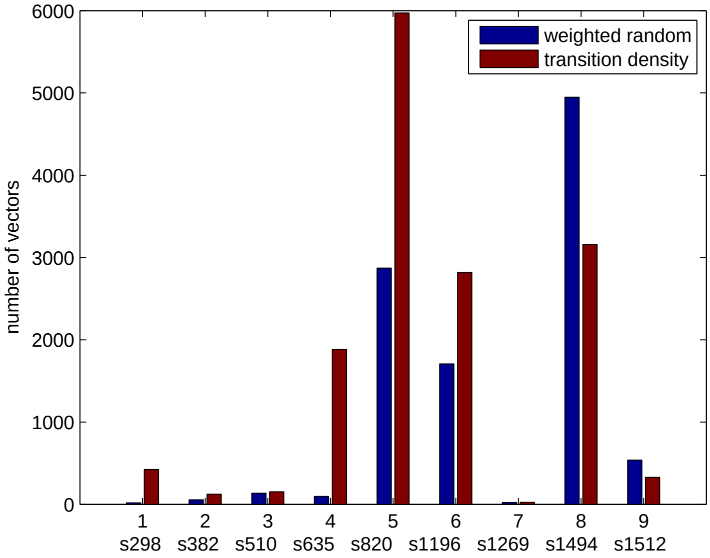

Figure 4.3: Numbers of weighted random and transition density vectors for 95% fault coverage in several ISCAS89 circuits.

an effective test with the minimum number of vectors as compared to arbitrary weights or transition densities.

If the vectors are scanned in with a fixed test clock frequency, then the number of vectors in a test sequence determines the test time. The shorter the test length the lesser time it will take to apply the test. Hence, finding the best case weight or the best case transition density is useful for constructing an effective test sequence for scan-BIST circuits in order to optimize the test application time.

The test time can be further reduced by adapting the test frequency dynamically depending on the transition density in the vectors, which will be described in the next chapter.

Chapter 5

Adapting Scan Clock Based On Transition Density

This chapter describes the scheme of reducing test application time for scan testing by adapting the test clock according to the transition density of the test vectors without exceeding the power limit. It is assumed that the test power limit is set by the maximum activity in the scan chain during capture cycles. Therefore, by monitoring the transition density of the scan in vectors, the scan clock can be dynamically adjusted [33, 34, 35, 36].

5.1 Dynamic Control of Scan Clock in a BIST Circuit

The circuit model chosen for the analysis is the test-per-scan multiple scan chain based BIST model [38]. To implement this model, flip-flops are added to primary inputs and primary outputs of the sequential circuit under test (CUT). All flip-flops are converted into scan flip-flops and partitioned into multiple scan chains. A test pattern generator (TPG), a multiple input signature analysis register (MISR) and a BIST controller are also added. A frequency divider module is added, which provides either the scan clock or the system clock, based on the mode of operation of the circuit. If the circuit is in the system mode, the BIST circuitry that consists of TPG, MISR and BIST controller are kept idle and the circuit runs with the system clock provided by the control clock select block. Likewise if the circuit is in test mode then the BIST circuitry is active and the control clock selects the scan clock, as shown in Figure 5.1. Here the frequency of the scan clock is determined by the peak power consumption of the circuit, which is assumed to be the power consumption in the scan chains having activity $\alpha = 1$.

Figure 5.2 shows the hardware implementation of an adaptive scheme for the test clock in a test-per-scan BIST model with multiple scan chains. A larger frequency divider divides

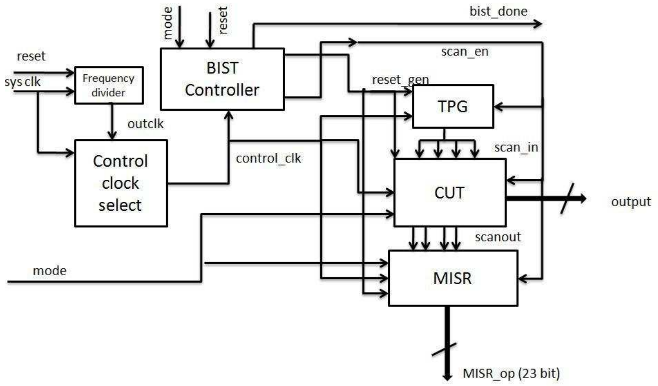

Figure 5.1: BIST circuitry for non-adaptive scan test clock.

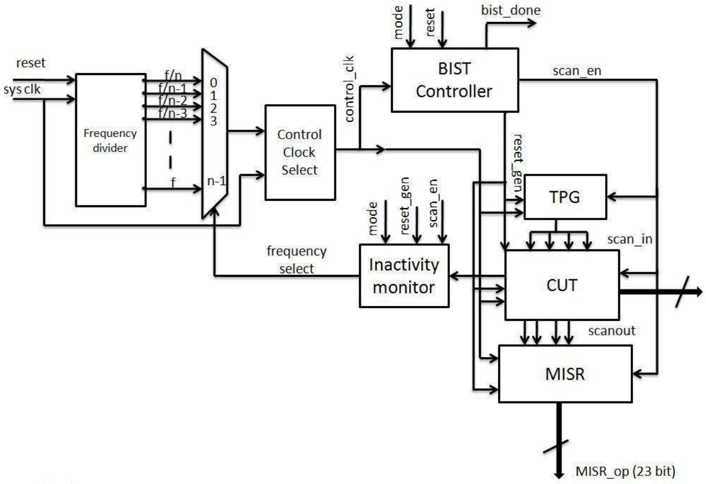

Figure 5.2: BIST circuitry for adaptive scan test clock.

the system clock into n different frequencies, where the fastest clock is the system clock and the slowest frequency is the test clock, based on the peak power consumption as described earlier. An n:1 multiplexer MUX is also added to select from the range of frequencies generated by the frequency divider block.

Figure 5.3 shows an inactivity monitor block implementation with monitors attached to the first scan flip-flop of each chain. The inactivity monitors are simple XNOR gates that produce a 1 whenever inactivity enters the attached scan chain and produces a 0 when an activity enters the scan chain. We feed the output of all monitors to a counter. Depending

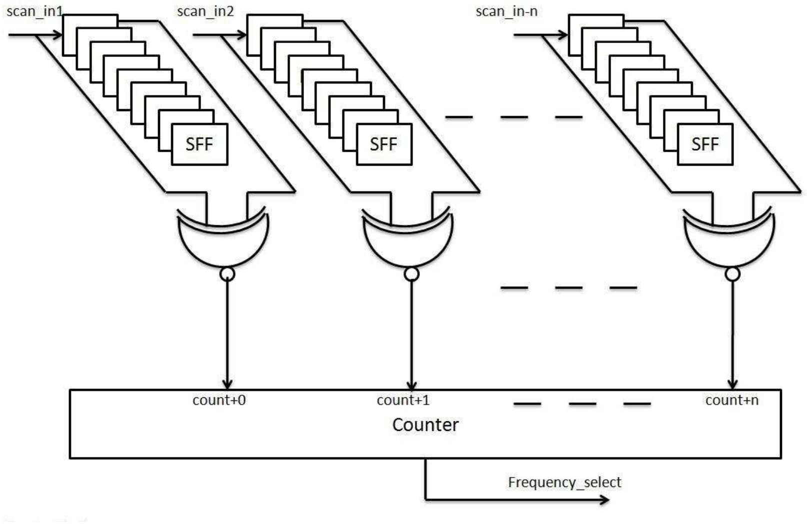

Figure 5.3: Inactivity monitor.

on the number of lines that are logic 1 at the output of the XNOR gates counter adds from 0 to n (number of scan chains) per clock. Hence all the inactivity that has entered the entire scan chains per clock has been accounted for. If no inactivity enters any of the scan chains, then the counter stays in its previous state by adding 0. If 1 inactivity enters one of the scan chains, the counter counts up by 1, and so on. Therefore, if inactivity enters every one of the n scan chains, the counter adds n to the previous count.

While counting up, if the counter reaches a certain threshold it signals the frequency selector MUX to deploy a higher frequency and hence dynamically adapts the scan frequency according to the inactivity in the chain.

The counter shown in Figure 5.3 is a simplified block diagram. The actual hardware consists of an adder, a combinational block with a register and a MUX. At every clock, if a non-activity enters a scan chain, the inactivity monitor attached to the first flip-flop of the scan chain becomes high. The inactivity monitor from every scan chain feeds to a combinational block. The output of the combinational block is connected to a separate select line of a MUX. The inputs of the MUX are 0, 1, 2... n, where n is the number of scan chains in the design. The inputs to the adder are the previous state of the register and the output from the MUX. The hardware is shown in Figure 5.4.

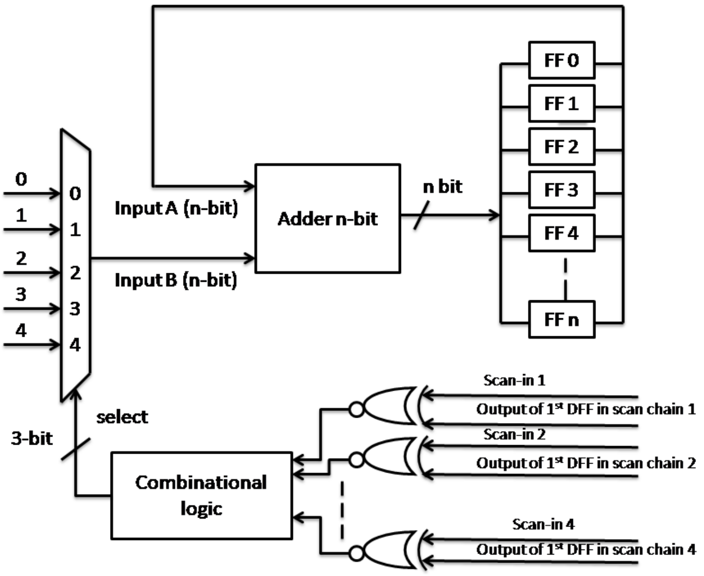

Figure 5.4: The inactivity counter

If 1 output of the inactivity monitor is high, the output of the combinational block will be 01 (assuming the number of scan chains in the design to be 4). the first input to the adder is the present state of the register and the second input to adder is the output to the mux. For our example, 1 will be added to the current contents of the of the register. Hence, at every clock the contents of the register wil be updated according to the number of inactivities that entered the scan chains. Similarly, If 1 inactivity enters in each of the scan chains, i.e., if the total number of inactivity is 4, then the combinational circuit output will be 11. This in turn will choose the second input of the adder to be 4. Hence, 4 will be added to the current state of the register. However, If no inactivity enters in any of the scan chains, the combinational circuit will produce 00 output. Hence, 0 will be added to the current state of the register.

5.2 Estimation of Scan-in Time Reduction

The reduction in scan-in time for this multiple scan chain based circuit can be estimated almost similarly as for a single chain implementation [34]. It is assumed here that the captured vector in the scan chain prior to the scan in vector has an activity factor of 1. That is, the scan chain is filled up with alternating 0s and 1s.

Let N be the total number of flip-flops in the circuit, A be the non-transition density ($A = 1 - \alpha$), v be the number of frequencies, T be the time period corresponding to the fastest clock and m be the number of scan chains. Hence, the largest scan chain will have $\frac{N}{m}$ flip-flops. The time reduction for scan-in of vectors will be dominated by the length of the longest scan chain.

The time period of the fastest clock is v times faster than the slowest clock. Therefore, the time period for the slowest clock is given by vT. If bits are scanned in with the slowest clock, the total scan-in time is given by $\frac{N}{m}vT$. The number of non-transitions in the input vector equals AN. These AN non transitions occur in $\frac{N}{m}$ cycles. Therefore, a non-transition occurs every $\frac{1}{mA}$ cycles. Hence, z transitions will occur in $\frac{z}{mA}$ cycles.

The cumulative number of non-transitions that can be held by all the scan chains is equal to N and to be able to deploy the speed up mechanism for all ranges of non-transitions, the frequency is increased only after counting $\frac{N}{v}$ non-transitions. Since a non-transition occurs in every $\frac{1}{mA}$ cycles, $\frac{N}{v}$ non-transitions occur in $\frac{N}{mAv}$ cycles. Thus, the frequency is not increased until $\frac{N}{v}$ non-transitions occur in about $\frac{N}{mAv}$ cycles. The counter keeps on counting until it reaches $\frac{2N}{v}$ non-transitions in the next $\frac{N}{mAv}$ cycles, before the next step up in frequency is signaled by the counter. Therefore, the first $\frac{N}{v}$ non-transitions are scanned in using a scan clock, which has a period of vT. Then the next $\frac{N}{v}$ non-transitions are scanned in using a scan clock which has a period of $(v-1)T$, and so forth. The clock period can reach the maximum of T, i.e., v^{th} frequency, when the scan chain is filled with only non-transitions.

As shown in Table 5.1, the ith frequency corresponds to a clock period of $(v - i + 1)T$ when the largest scan chain has between $\frac{(i-1)N}{v}$ and $\frac{iN}{v}$ non-transitions. The ith frequency is employed between clock cycles $\frac{(i-1)N}{mAv}$ and $\frac{iN}{mAv}$. The scan clock initially has a period of vT in cycle 1. The scan clock period is decreased in steps until the $\frac{N}{m}$th clock. Therefore, the

Table 5.1: Determination of clock cycle range for different frequencies.

	Clock period	Number of non-transitions		Clock cycles	
		Lower limit	Upper limit	Lower limit	Upper limit
1	vT	0	$\frac{N}{v}$	0	$\frac{N}{mAv}$
2	$(v-1)T$	$\frac{N}{v}$	$\frac{2N}{v}$	$\frac{N}{mAv}$	$\frac{2N}{mAv}$
.	.	.	.	.	.
.	.	.	.	.	.
i	$(v-i)T$	$\frac{(i-1)N}{v}$	$\frac{iN}{v}$	$\frac{(i-1)N}{mAv}$	$\frac{iN}{mAv}$
.	.	.	.	.	.
.	.	.	.	.	.
v	T	$\frac{(v-1)N}{v}$	$\frac{vN}{v}$	$\frac{(v-1)N}{mAv}$	$\frac{vN}{mAv}$

clock cycle corresponding to the last scan clock frequency is $\frac{N}{m}$. If the maximum number of frequencies the scan clock will speed up to is given by x, then

$$\frac{N}{mAv} \times x = \frac{N}{m} \tag{5.1}$$

$$x = Av \tag{5.2}$$

Thus, the number of scan clock frequencies employed for a scan-in vector with non-transition density of A is Av.

The total scan-in time per vector is the sum of scan-in times at each frequency. The scan-in time at each frequency is given by the product of the number of cycles run at that frequency and the time period of the clock, as shown in Table 5.1 The total scan-in time per vector is given by

$$\sum_{i=1}^{Av}(\lceil\frac{iN}{mAv}\rceil - \lceil\frac{(i-1)N}{mAv}\rceil) \times (v-i+1)T \tag{5.3}$$

Substituting $\frac{N}{m} = \bar{N}$ we get

$$\sum_{i=1}^{Av}(\lceil\frac{i\bar{N}}{Av}\rceil - \lceil\frac{(i-1)\bar{N}}{Av}\rceil) \times (v-i+1)T \tag{5.4}$$

The above equation simplifies to

$$\sum_{i=1}^{Av}\frac{\bar{N}}{Av}(v-i+1)T \tag{5.5}$$

If v and $\bar{N}$ are chosen to be powers of 2, then the total scan-in time is

$$\frac{\bar{N}}{v}(v \times Av - Av(Av + \frac{1}{2}) + Av)T \tag{5.6}$$

So, the reduction in scan-in time in largest scan chain is

$$\begin{aligned} &= \frac{\bar{N}vT - \frac{\bar{N}}{v}(v.Av - Av(Av + \frac{1}{2}) + Av) \times T}{\bar{N}vt} \\ &= \frac{A}{2} - \frac{1}{2v} \\ &= \frac{1-\alpha}{2} - \frac{1}{2v} \end{aligned} \tag{5.7}$$

Hence maximum reduction in scan-in time can be reached only if non-transitions are scanned in and maximum reduction is close to 50% if the number of frequencies chosen is very high. For random patterns, where non-activity is 0.5, the highest reduction that can be reached is around 25%.

5.3 Time Reduction and Power Consumption

This section gives experimental results on reduction in scan-in time for random patterns with activity 0.5 for ISCAS89 benchmark circuits. Table 5.2 summarizes the results. The transition density in random patterns or pseudo random patterns is approximately 0.5. Theoretically, using the dynamic scan clock control scheme we get 50% reduction in test

Table 5.2: Scan-in time reduction in ISCAS89 benchmark circuits.

Circuit	No. of FF	No. of gates	No. of vectors	Time savings (%)
s298	23	282	19	29.23
s382	30	361	90	32.52
s510	32	447	138	29.55
s820	42	655	3455	27.55
s1196	46	885	2528	27.7
s38417	443	31834	1000	27.1

application time if transition density is 0, that is if the vector is filled with all 1s or all 0s. When patterns are applied from a pseudo random TPG the transition density will be close to 0.5 if not exactly 0.5. From the table, it is to be noted that it takes a while for the TPG to start becoming random and the activity is less than 0.5 in the beginning. Therefore, for a small number of vectors and for small circuits the saving in scan-in time dose not conform to the theoretical analysis. But as larger numbers of vectors are used for we see the time reduction conforming to 25% saving as transition density approaches 0.5.

The power while applying these tests is kept under a peak power constraint. Here the peak power is assumed to be the power consumed when the scan chain has activity 1, that is, the scan chain is filled with alternating 1s and 0s. The power graphs in Figures 5.5 and 5.6 show the power consumption for circuit s1196 with fixed clock and adaptive clock scan, respectively.

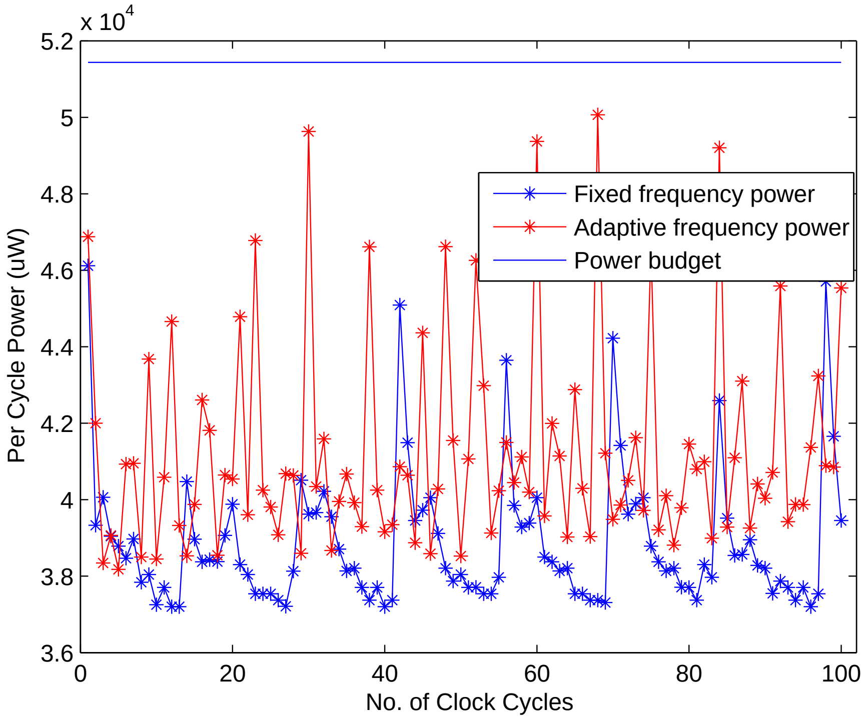

Figure 5.5: Power per test clock in s1196.

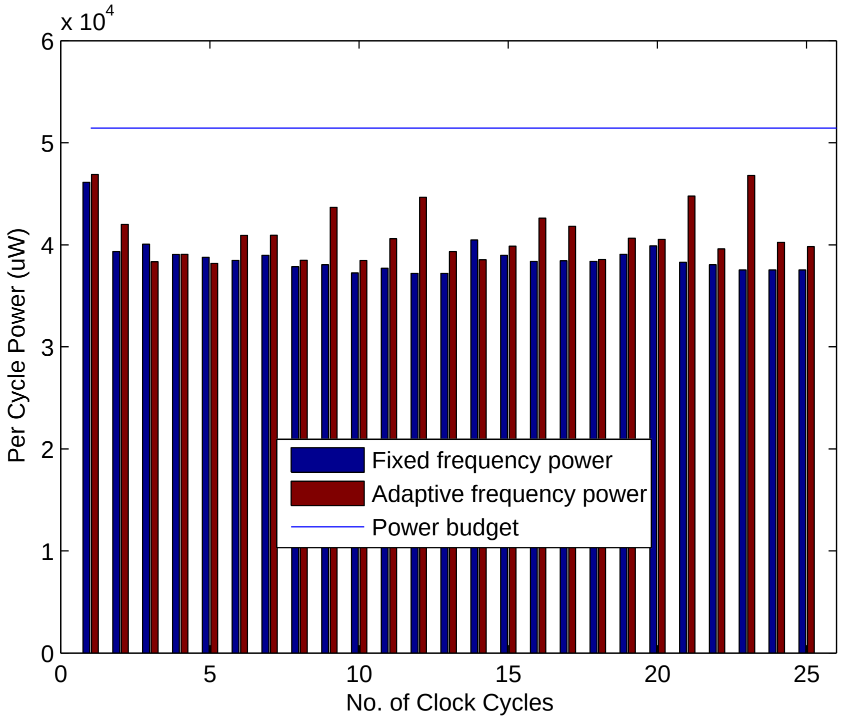

Figure 5.6: Power per test clock for first 25 cycles.

Chapter 6

Controlled Transition Density Patterns for BIST

This chapter presents a hardware implementation of the controlled transition density based vector generation by a BIST-TPG. The first section describes the hardware used to implement the pattern generator, the second section estimates the randomness of the generated vectors and the last section describes the implementation of the TPG in the adaptive scan clock scheme described earlier.

6.1 BIST-TPG Circuit for Controlled Transition Density

The test pattern generator (TPG) chosen for the analysis is a 28 bit external LFSR using the polynomial $p(x) = x^{28} + x^3 + 1$. Its combinational part consists of only AND gates and inverters, an eight input MUX to select from eight different probability of a bit being 1, a simple finite state machine (FSM) to control the MUX, and a toggle flip-flop. Figure 6.1 shows the circuitry for the test pattern generator. The combinational network generates eight different weighted random bit sequences. The weights are constructed by ANDing two or more outputs from non-adjacent cells of the LFSR. Here it is to be noted that two cells in an n-bit LFSR are adjacent if the output of one cell feeds the input of the second directly, without an intervening XOR gate.

As shown in Figure 6.1, eight weights for the probability of a bit being 1 are 0.125, 0.25, 0.375, 0.4375, 0.5, 0.625, 0.75 and 0.875, respectively. The probability of a bit being 1 or 0 at the output of any cell of the TPG is 0.5. This weight is directly fed to one of the inputs of the MUX. Two outputs from two non-adjacent cells were ANDed to produce a weight of 0.25, three outputs from three non-adjacent cells were ANDed to produce a weight of 0.125, and inverting these two weights we get weights of 0.75 and 0.875, respectively.

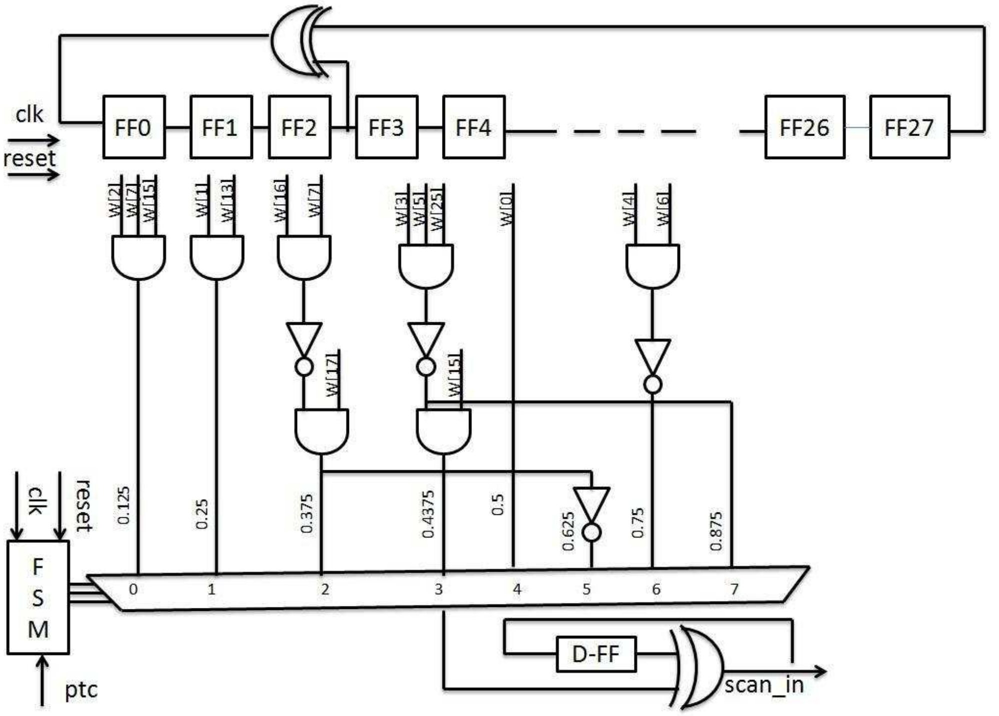

Figure 6.1: Hardware implementation of TPG.

For generating weight of 0.375, weight 0.75 is again ANDed with another cell output that is not adjacent to any of those two cells that are used in creating the 0.75 weight. Similarly for generating weight of 0.4375, weight 0.875 is ANDed with another non-adjacent cell output. Finally, to construct a weight of 0.625, weight 0.375 is inverted.

The different weight lines are the inputs to the 8:1 MUX. An FSM controls the select lines of the MUX to choose the intended probability.

A toggle generating flip-flop constructed with a D-flip-flop and an XOR gate is added to produce the required transition density in the vectors that are to be fed to scan chain as shown in Figure 6.1. Through the select lines of the MUX a weight is selected and the bit sequence fed to one of the inputs of the XOR gate; the other input line of the XOR gate is the output of the D flip-flop.

Once a weight is selected, the corresponding bit sequence will then control the transition at the output of the XOR gate. A 1 in the bit sequence will produce a transition at the

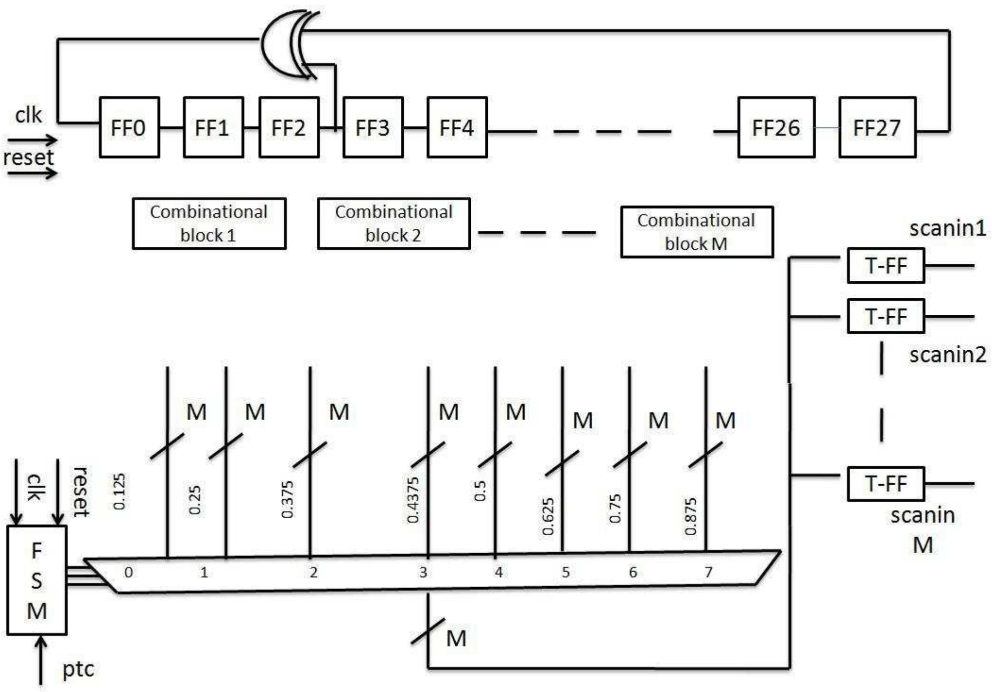

Figure 6.2: Hardware implementation of TPG for M scan chains.

output of the XOR gate and a 0 will produce no transition. Thus the resulting transition density in the bit stream at the output of the XOR gate will have the same weight (i.e., the probability of a transition to occur) as the weight selected from the MUX.

The output of the D type flip-flop is fed back to the input with an XOR gate to generate the toggling of the bits. Whenever a 1 is encountered at the input of the XOR gate from the output of the MUX there is a toggle. Whenever a 0 is encountered the bit remains unchanged. The generated bit stream is then fed to the scan chains.

This bit stream then feeds the scan chain input. For multiple chain implementations, only the combinational part to generate weights is copied multiple times and all the same weighted lines are fed to the corresponding input of the MUX. Also, toggle flip-flops are added for each output line of the MUX and fed to individual XOR gates to produce separate bit streams. Thus separate bit streams are used to feed each separate scan chain, as shown in Figure 6.2.

Table 6.1: Estimation of randomness in generated 1000 random patterns.

Avg. no. of 1s in each bit position	$p1$	$\sum_{j=1}^{n}(p_{0j} \times \log_2 p_{0j} + p_{1j} \times \log_2 p_{1j})$
1255	0.1255	15.25
2508	0.2508	22.74
3746	0.3746	26.74
4378	0.4378	27.68
5007	0.5007	27.99
6254	0.6254	26.71
7491	0.7491	18.21
8750	0.875	15.21

6.2 Randomness of Weighted Random Patterns

Entropy has been used by many researchers as a measure of randomness metric

$$\sum_{i=1}^{r} p_i \times \log_2 p_i \tag{6.1}$$

where p_i is the probability that the signal is in state 1 and r denotes the total number of states [9, 40]. This metric can quantify how the quality of randomness deviates if there is a bias in the random number generator. For an n-bit perfect random number generator, $r = 2^n$ and $p_i = 0.5^n$ and hence entropy will be $H_{max} = n$, reflecting the maximum randomness. For any biased random bit generator, entropy will be $0 \leq H \leq n$. For an easier computation in an n-bit LFSR, if $p_{1j}(p_{0j})$ denotes probability of having 1(0) in bit b_j then entropy can be approximated by adding the entropies of individual bits:

$$\sum_{j=1}^{n}(p_{0j} \times \log_2 p_{0j} + p_{1j} times \log_2 p_{1j}) \tag{6.2}$$

To verify the weighted randomness in case for the above mentioned weights for a vector set of 10000 vectors of $n = 28$ bits, the number of 1s was counted for each weight. Table 6.1 summarizes the randomness of the intended weight.

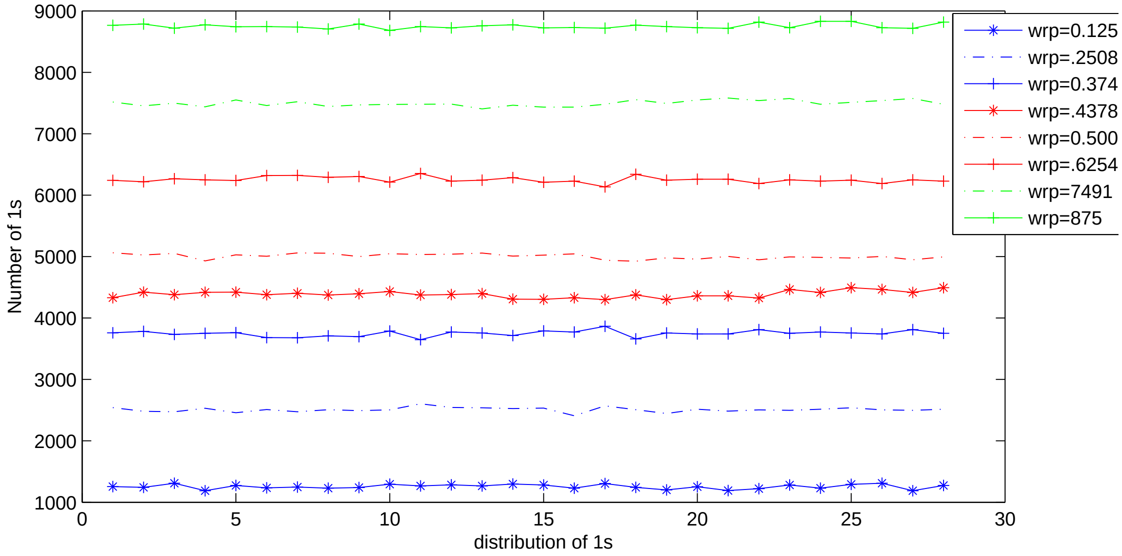

Figure 6.3: Distribution of 1s in weighted random patterns.

As the toggle flip-flop controls the transitions depending on probability of the input signal being 1, the transition density will follow the randomness value closely. Also, Figure 6.3 shows the distribution of 1s generated from each of the weights.

6.3 Dynamic Control of Scan Clock in BIST Circuit with Modified TPG

We replaced the conventional test pattern generator with the modified test pattern generator, along with the finite state machine that selects transition density from the test pattern generator as shown in Figure 6.4. The finite state machine (FSM) takes the number of the patterns applied as inputs from the BIST controller and controls the transition density of the test vectors applied.

From the analysis of circuits in chapter 4, we can pre-determine the best case transition density with the modified TPG and run the whole test session with a pre-selected transition density. Also the circuitry to dynamically adapt the scan clock will help speed up the test clock by monitoring the inactivity and, therefore, keep the whole test session power-constrained.

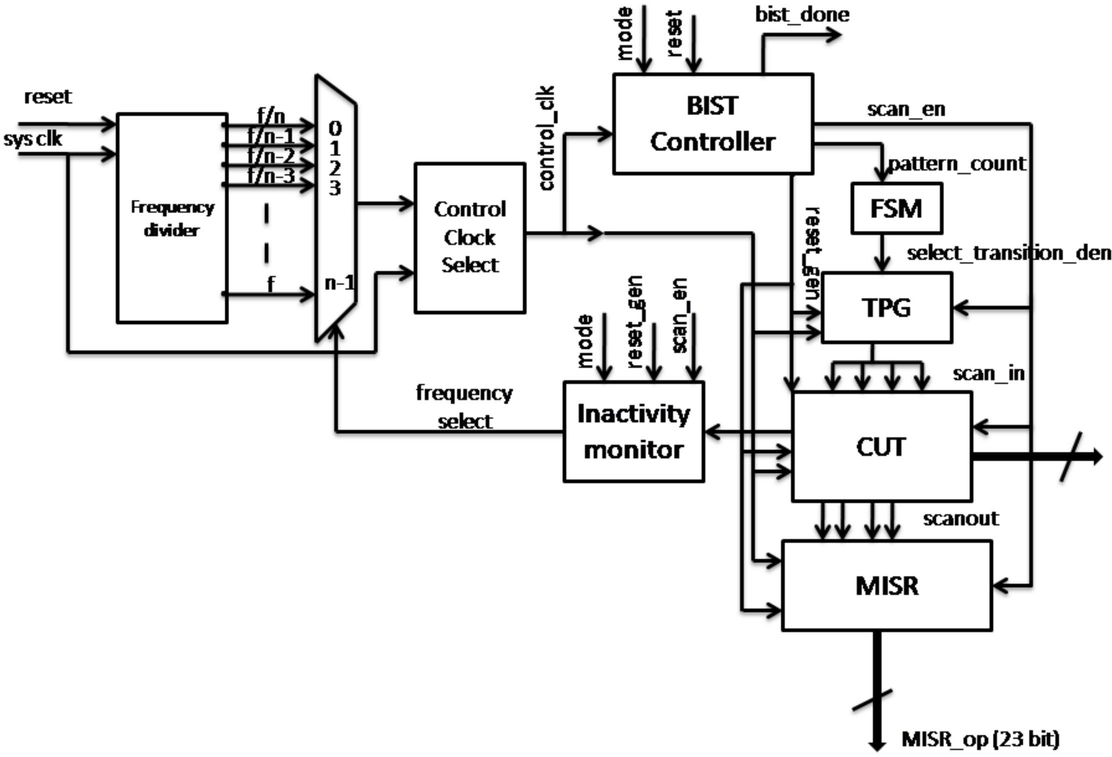

Figure 6.4: Adaptive scan clock scheme with modified TPG.

6.4 Fault Coverage by the Modified TPG

The newly proposed TPG gives similar fault profiles as the simulation results. Figures 6.5 and 6.6 show the number of vectors needed to reach fault coverage of 95% in circuits s510 and s1512, respectively, when test vectors were applied from the new modified TPGs. It is to be noted that, in circuit s510 the best case transition density is 0.4375 whereas the best case transition density for s1512 is 0.25. This actually conforms to the best case weights for random pattern, 0.4375 and 0.875, respectively, as described earlier.

From both the graphs it can be seen, that the proposed LFSR is capable of producing vectors with the desired weight (probability of a bit being 1) and the desired transition density (number of transitions in a vector bit stream). Fault coverage obtained by the two

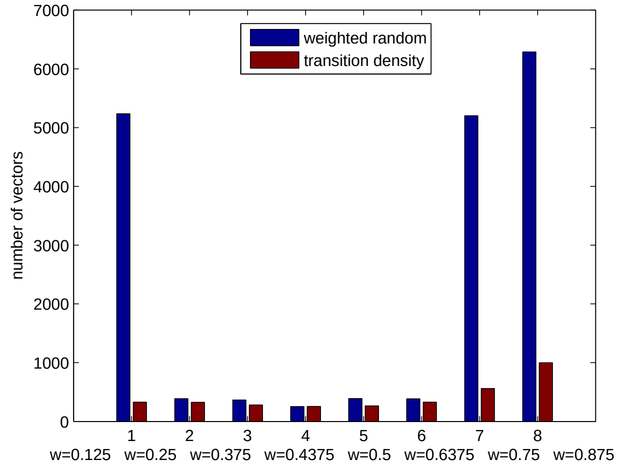

Figure 6.5: Performance of transition density and weighted random patterns of s510.

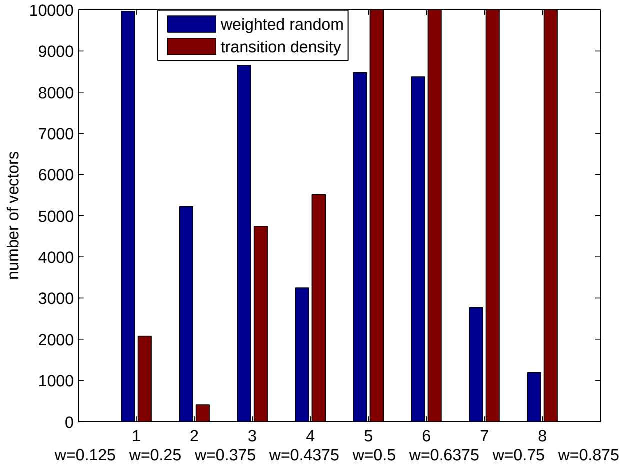

Figure 6.6: Performance of transition density and weighted random pattern of s1512.

different kind of vectors followed the trends of the vectors produced by the Matlab [29] program that was described earlier.

Chapter 7

A Greedy Algorithm to Apply Tests with Different Transition Densities

This chapter describes a scheme to reduce test application time by mixing vectors of different transition densities in a controlled way. Here we propose the scheme through a greedy algorithm to find an optimal vector set that consists of various transition densities but in a controlled manner.

7.1 Analysis of Fault Profiles

From Chapter 4, it can be seen that to achieve certain lower fault coverage the low transition densities are almost as good as the best case transition densities in terms of number of vectors to reach that fault coverage. Figure 7.1 shows the fault coverage in circuit s510 by vectors with transition densities from 0.1 to 0.5. According to our analysis in Chapter 4 we see that the best case transition density is 0.5.

The steep rise in the fault coverage indicates that to achieve a lower fault coverage the lower transition density based vectors are almost as good as the best case transition density vectors. So even if the lower transition density vectors need a few vectors more to reach the partial fault coverage using that transition density, with the adaptive scan clock scheme we can reduce the test application time. Figure 7.2 shows a logarithmic scale plot of the fault coverage in s510 from the data of Figure 7.1.

The idea is to break down the target fault coverage into a number of partial fault coverages and apply a greedy algorithm to select a best case transition density in each partial target fault coverage stage.

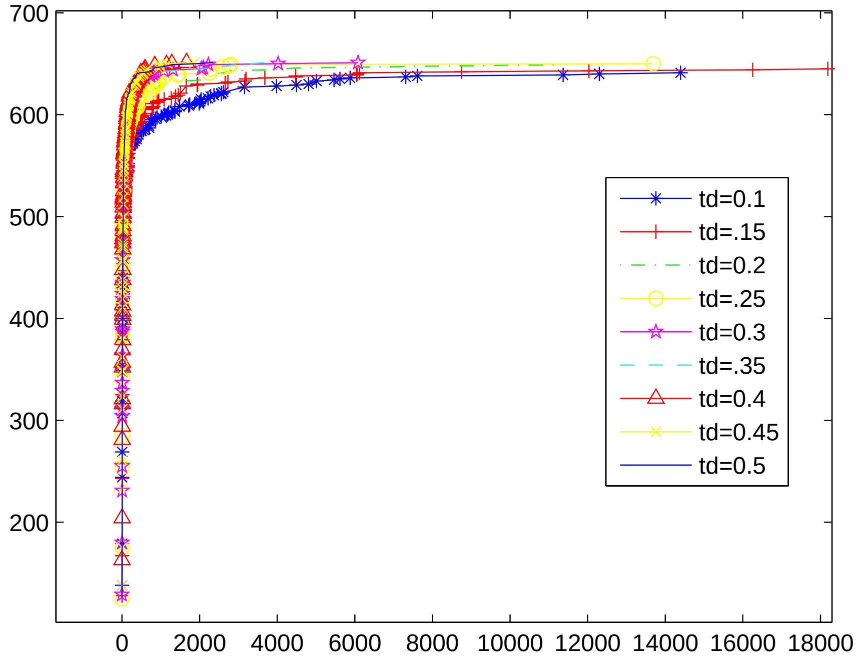

Figure 7.1: Detected faults vs. number of transition density vectors obtained from fault simulation of s510

7.2 Algorithm to Apply an Optimal Set of Vectors

The greedy algorithm described here finds a locally optimal solution for reaching partial target fault coverage in each stage based on the number of vectors to reach that partial test coverage multiplied by the time taken to apply those vectors in adaptive scan clock scheme, to reach a globally optimal vector set to achieve the final target fault coverage.

The time needed to apply each vector in adaptive scheme for a given transition density can be computed from an equation described here. Using adaptive scan clock scheme for a transition density α and the number of frequencies available to adapt from v, the reduction in scan-in time is given by,

$$\frac{1}{2}(1-\alpha) - \frac{1}{2v} \qquad (7.1)$$

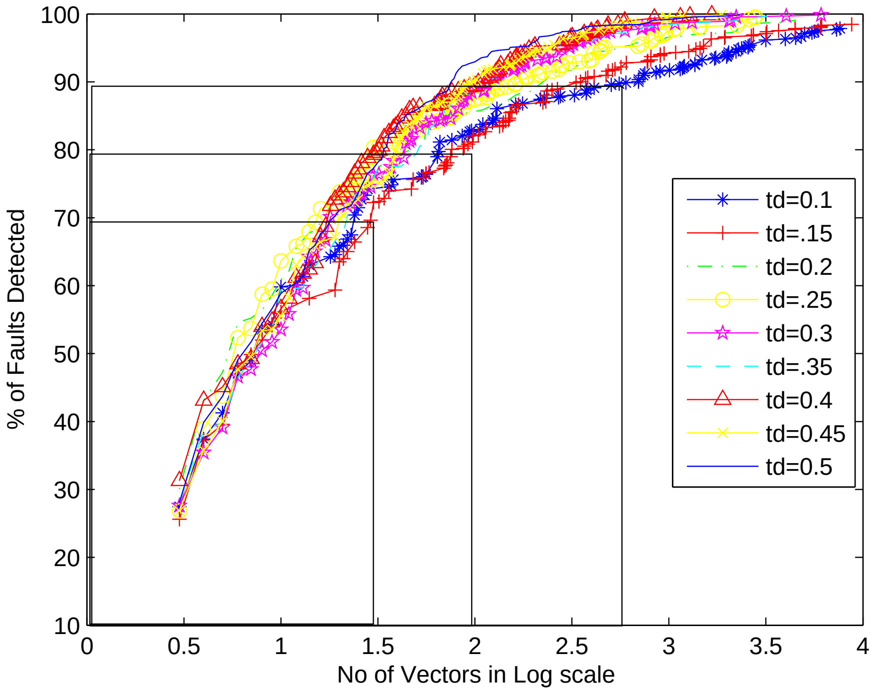

Figure 7.2: Fault coverage by transition density vectors obtained by simulation of s510

Now in a setup where the number of frequencies to adapt from is fixed as v, the reduction in time varies with α. Therefore, for simplicity we can take reduction in time to be,

$$\frac{1}{2}(1-\alpha) \tag{7.2}$$

Hence, the time taken to scan in vectors with a given transition density can be found by deducting the scan-in time reduction from the total time taken to scan-in N vectors at a fixed frequency.

The proposed algorithm considers all transition densities and the number of vectors needed in each transition density to reach the fault coverage at each stage and chooses the transition density that will result in the minimum time to scan-in vectors for that stage.

Algorithm :

Given for a circuit a target fault coverage FC, fault profiles data table for each transition density to reach target fault coverage FC, a fixed number of vectors and partial fault coverage pfc1, pfc2, pfc3 . . . pfcn find an optimal set of vectors of various transition densities.

- Step 1: Get the best case transition density - x
- Step 2: Get the fault coverage achieved by the best case transition density - y
- Step 3: Set FC $= y$
- Step 4: Set the number of vectors - z
- Step 5: Divide the FC into n partial fault coverages Pfc1 Pfc2....Pfcn
- Step 6: For each partial fault coverage starting from the lowest partial fault coverage up to FC
    1. Compute the time taken to scan-in the number of vectors needed to reach that partial fault coverage
    2. Choose the minimum value
    3. Select that transition density as optimum for that partial fault coverage
    4. Set the number of vectors

For illustrating the algorithm, the circuit s510 is chosen. 10000 thousand vectors were generated for each the eight different transition densities from the LFSR described earlier. Running fault simulation using those vectors it was observed that transition density 0.5 is the best. We then divide the target fault coverage into 6 partial fault coverages (Pfc) and run the algorithm to find a vector mix that will reduce the transition density in the vectors in a controlled way so that the final target fault coverage can be achieved without lengthening the test session and without loss of fault coverage. In addition, if a transition density lower than the best case transition density is chosen in order to reach some of the partial fault coverages then that will reduce the test application time using the scan clock adaptive scheme.

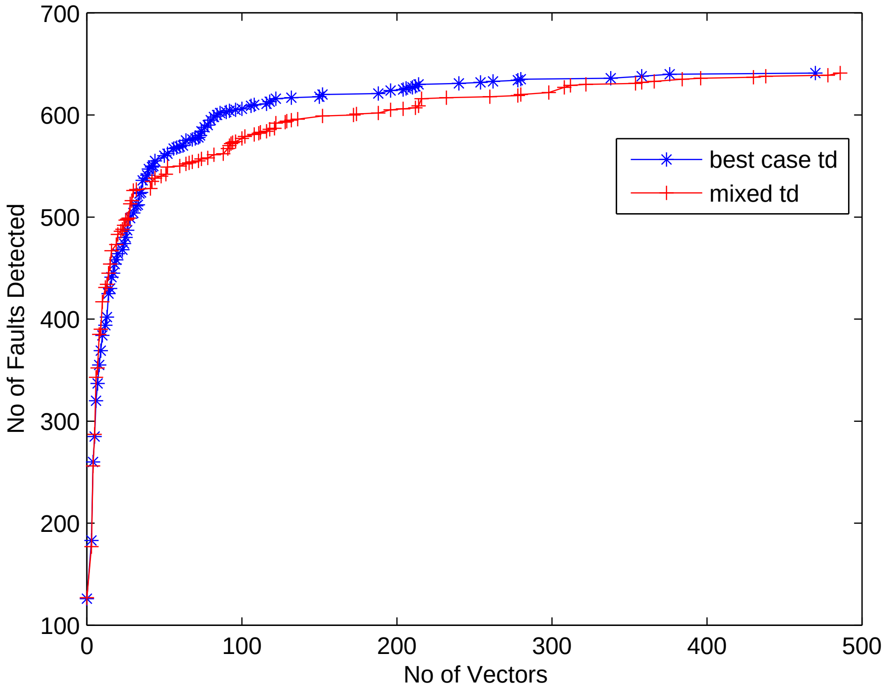

Figure 7.3: Detected faults vs. number of vectors in s510 for best case transition density vectors and mixed transition density vectors.

It can be seen from Figure 7.3 and Table 7.3 that even if few more vectors compared with the best case transition density are required, we can speed up the test by using optimal transition density in each partial case without increasing total number of vectors and without sacrificing fault coverage. Figure 7.4 shows the flow chart of the algorithm described in this section.

7.3 Implementation of Controlled Mixed Transition Density Based TPG in BIST Circuit

The same adaptive scheme described in Chapter 6 (Section 3) is used to implement the controlled mix of transition densities with a small modification in the FSM as shown in Figure 7.5. In the best case transition density setup the FSM selected the best case transition for the whole test session. It keeps track of number of patterns applied from the BIST controller. In place of running the whole test session in one transition density now

Table 7.1: Performance of mixed transition density vectors.

	Best case TD	Mix TD
FC	99%	99%
Number of vectors	592	592
Transition density	0.5	Pfc1 70% – TD 0.25 vectors 16 Pfc2 80% – TD 0.25 vectors 14 Pfc3 85% –TD 0.4 vectors 14 Pfc4 90% –TD 0.5 vectors 32 Pfc5 95%–TD 0.5 vectors 76 Pfc6 99%–TD 0.4 vectors 404

the FSM is programmed to select different transition density as the pattern count reaches certain limit. The FSM can be programmed to change the selection according to the Greedy algorithm described earlier. For example, from the Table 7.1, the FSM can be programmed to run first 30 vectors in 0.25 transition density and hence speed up the test application time by adapting the scan clock. Also running 32 vectors and later 404 vectors in transition density 0.4 we can still speed up as compared running all the vectors in 0.5 transition density.

Thus the hardware provides a controlled mixing of transition densities in the vector sets. Using vectors that have lower transitions at the beginning of the test sequence helps to detect the easy to detect faults also facilitates the speeding up the scan clock. The transition density of the vector sets can be tuned to higher transition density by the use of a simple FSM later towards the end of the test sequence. The power is held under the power budget by dynamically adapting to a slower scan clock.

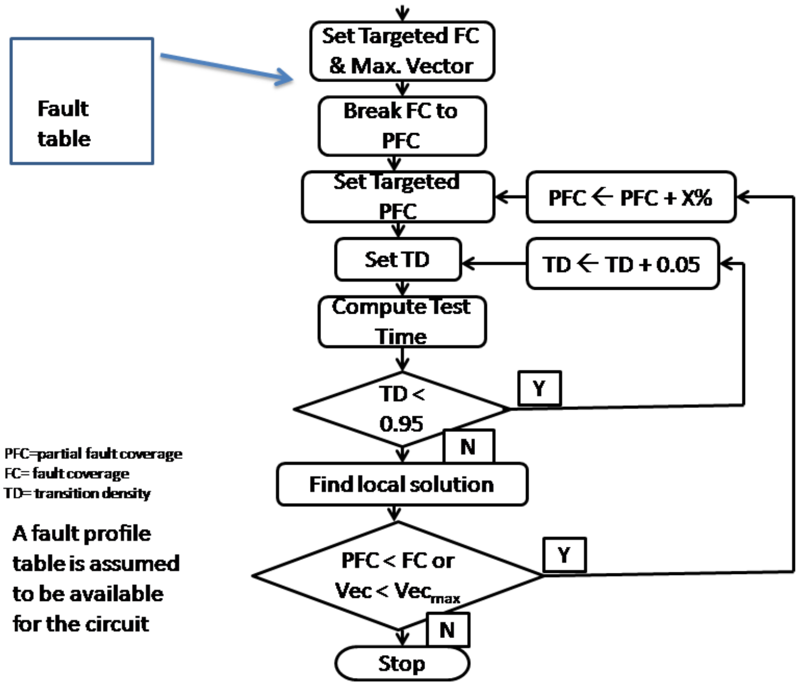

Figure 7.4: Flow chart of proposed algorithm.

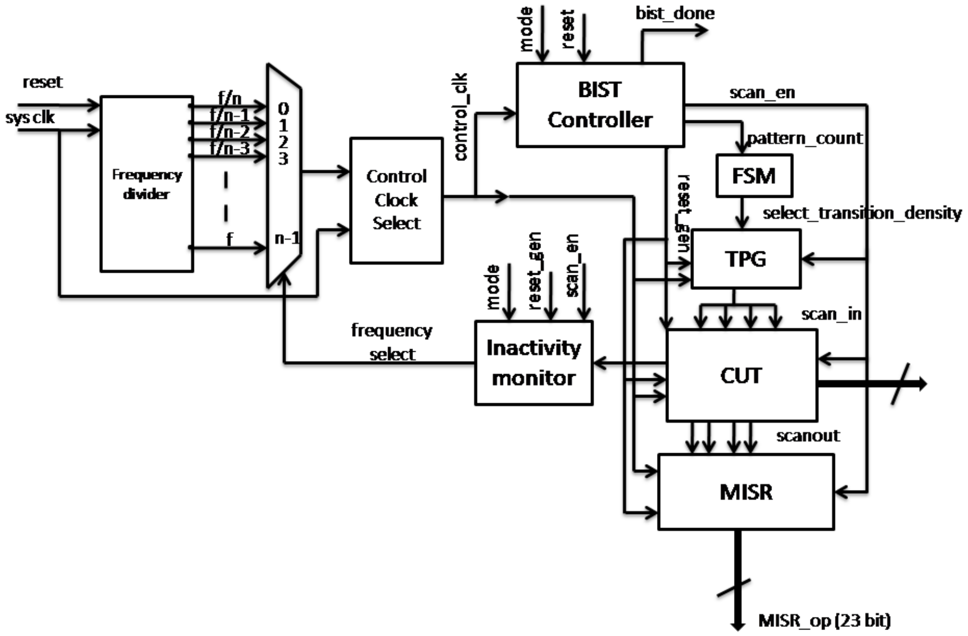

Figure 7.5: Hardware Implementation for controlling a mix of various transition densities.

Chapter 8

Experimental Results

This chapter describes the entire procedure followed in experiments, followed by results obtained. ISCAS 89 circuits were chosen to run the experiments.

8.1 Fault Coverage Analysis

The benchmark circuits from ISCAS89 suite were used to do the analysis of fault coverage based on three different types of test vectors. A Matlab [29] program was written to generate all the different vector sets. Scripts were written to convert the generated vector into scan patterns for AUSIM, an Auburn University fault simulation program [37].

Matlab [29] and AUSIM [37] programs were run on Auburn University's High Performance Compute Cluster (HPCC) [42].

Number of vectors needed to reach target fault coverage has been recorded for each set of vectors with a different weight or a different transition density. The number of vectors was thus chosen for the experiments and the BIST controller was set accordingly.

The newly proposed TPG was implemented in the previous netlists in place of conventional 28 bit LFSR. It was again simulated in Modelsim and the randomness of the generated vectors based on weights were measured after capturing the vectors Modelsim transcript. This transcript files were fed to another Matlab [29] program to compute the randomness. Those vectors were simulated by AUSIM [37] to obtain their fault coverage.

Table 8.1 shows the comparison between the number of vectors needed to achieve 95% fault coverage for ISCAS89 circuits. Also Table 8.2 shows the comparison between the number of vectors needed for the same circuits to reach 90% of fault coverage. These fault

Table 8.1: Test lengths for random and best-case weighted random (WRP) and transition density (TDP) patterns for 95% fault coverage in ISCAS89 circuits.

Circuit	DFFs	Gates	PI+ PPI	No. of vectors (0.5)	No. of vectors (WRP)	No. of vectors (TDP)	Weight	TD
S27	4	10	7	16	7	35	0.15	0.8
S298	14	119	17	113	73	10000**	0.45	0.0
S382	21	158	24	122	56	124	0.3	0.45
S386	6	159	13	1042	540	869	0.4	0.6
S344	15	160	24	82	68	10000	0.4	0.0
S510	6	211	32	138	136	152	0.4	0.5
S420	16	218	34	10000	644	735	0.9	0.2
S635	32	286	34	10000	97	1883	0.9	0.1
S832	5	287	23	7840	3771	10000	0.45	0.0
S820	5	289	23	2872	2827	5972	0.5	0.45
S641	19	379	54	175	175	270	0.5	0.7
S713	19	393	54	10000	10000	10000	0.75	0.55
S967	29	394	45	2390	934	4346	0.4	0.35
S953	29	395	45	3088	1214	4502	0.35	0.35
S838	32	446	66	10000	1902	2848	0.95	0.1
S1238	18	508	32	10000	10000	10000	0.55	0.45
S991	19	519	84	100	76	62	0.55	0.1
S1196	18	529	32	2528	1706	2821	0.55	0.45
S1269	37	569	55	48	22	24	0.6	0.5
S1494	6	647	14	1090	1046	562	0.55	0.4
S1488	6	653	14	1002	937	512	0.55	0.4
S1423	74	657	91	86	86	104	0.6	0.2
S1512	57	780	86	5556	538	338	0.75	0.2
S13207	669	7951	700	15000*	15000	15000	0.35	0.3
S15850	597	9772	611	15000	15000	15000	0.5	0.3

simulations were done with 10000 vectors for each case of weight or transition density. Therefore, for the circuits (marked as ** in table) that could not reach 90% or 95% fault coverage within those 10000 vectors, the best case was taken as 10000 vectors. The bigger circuit s13207 and s15850 were simulated for faults with 15000(*) vectors.

The first four columns show the circuit name, number of flip-flops, gates , primary and pseudo-primary inputs (PPI) in the respective circuits. The next column shows number of vectors needed to reach the target fault coverage with conventional random test patterns. The sixth and seventh column indicate the number of vectors needed to reach the same target

Table 8.2: Test lengths for random and best-case weighted random (WRP) and transition density (TDP) patterns for 90% fault coverage in ISCAS89 circuits.

Circuit	DFFs	Gates	PI+ PPI	No. of vectors (0.5)	No. of vectors (WRP)	No. of vectors (TDP)	Weight	TD
S27	4	10	7	15	5	12	0.15	0.6
S298	14	119	17	52	52	10000	0.5	0.0
S382	21	158	24	42	30	16	0.3	0.4
S386	6	159	13	621	317	609	0.35	0.4
S344	15	160	24	50	34	10000	0.45	0.0
S510	6	211	32	70	84	76	0.4	0.5
S420	16	218	34	10000	300	263	0.9	0.1
S635	32	286	34	10000	3	13	0.85	0.15
S832	5	287	23	1380	1126	2128	0.45	0.55
S820	5	289	23	962	806	1524	0.45	0.45
S641	19	379	54	38	31	41	0.7	0.55
S713	19	393	54	109	131	139	0.75	0.55
S967	29	394	45	746	268	656	0.4	0.45
S953	29	395	45	746	312	458	0.4	0.35
S838	32	446	66	10000	488	491	0.95	0.15
S1238	18	508	32	2506	1759	2598	0.55	0.45
S991	19	519	84	34	34	23	0.5	0.75
S1196	18	529	32	675	537	712	0.65	0.3
S1269	37	569	55	14	10	12	0.6	0.85
S1494	6	647	14	460	416	266	0.6	0.4
S1488	6	653	14	438	410	247	0.6	0.55
S1423	74	657	91	12	6	11	0.85	0.2
S1512	57	780	86	46	46	78	0.5	0.6
S13207	669	7951	700	4262	2127	1490	0.35	0.3
S15850	597	9772	611	2463	2463	3293	0.5	0.3

fault coverage but with specific weight (probability of it being 1) and specific transition density respectively. These are denoted as best case weight and best case transition density.

8.2 Dynamic Scan Clock Implementation

Verilog netlists of ISCAS 89 circuits were used for simulation purpose. The dynamic scan clock control scheme was implemented in circuits with multiple scan chains as described in Chapter 5. The simulation tool from MentorGraphics Modelsim, was used to simulate the circuit with and without the dynamic scan clock circuitry. Time needed to apply the test sequences in the both cases were recorded.

Table 8.3: Reduction in scan-in time for conventional random patterns of weight 0.5.

Circuit	Patterns	Test time Fixed Freq. (ns)	Test time Adap. Freq. (ns)	Test time reduc -tion (%)
s298	52	14605	10050	31.18
s382	42	15165	10320	31.94
s510	70	25245	18200	27.9
s820	962	461805	348392	24.55
s953	746	537165	418073	22.17
s1196	675	351045	264652	24.61
s1488	438	175245	124572	28.91
s13207	4262	36482765	31565011	13.47
s15850	2463	18915885	16341260	13.61

Table 8.4: Reduction in scan-in time for best-case weighted random patterns (WRP).

Circuit	Best Case Weight	Patterns	Test time Fixed Freq (ns)	Test time Adap. Freq (ns)	Time reduc -tion (%)
s298	0.5	52	14605	10050	31.18
s382	0.3	30	10845	6661	38.57
s510	0.4	84	30285	19570	35.38
s820	0.4	806	386925	268971	30.48
s953	0.4	312	224685	162371	27.73
s1196	0.6	537	279285	221416	20.72
s1488	0.6	410	1 64045	117901	28.12
s13207	0.35	2127	18207165	16180025	11.13
s15850	0.5	2463	18915885	16341260	13.61

The synthesis tool from MentorGraphics Leonardo Spectrum was used to analyze the area of the circuits with and without the dynamic scan clock circuitry. To insert the scan flip-flops in the basic verilog netlists MentorGraphics DFT Advisor tool was used.

In this section scan in time reduction has been shown for various circuits from ISCAS89 suite. Tables 8.3, 8.4 and 8.5 show the scan-in time reduction to reach 90% fault coverage by using conventional vectors, best case weighted random patterns and best case transition density patterns respectively. In Table 8.3, the test application time required to complete the test sequence in fixed scan frequency is tabulated in third column and the test application

Table 8.5: Reduction in scan-in time for best-case transition density patterns (TDP).

Circuit	Best Case TD	Patterns	Test time Fixed Freq (ns)	Test time Adap. Freq (ns)	Time reduc -tion (%)
s298	0.5	10000	2800045	1974026	29.5
s382	0.4	32	11565	8287	24.72
s510	0.4	76	27405	19852	28.34
s820	0.4	1524	731565	504453	31.04
s953	0.3	458	329805	231833	29.7
s1196	0.3	712	370285	262350	29.14
s1488	0.5	247	98845	72831	26.31
s13207	0.3	1490	12754445	10149712	20.42
s15850	0.3	3293	25290285	20109065	20.48

time required to complete the test sequence in dynamically adaptive scan frequency is shown in fourth column. The last column in the table shows the reduction in scan-in time by deploying the latter method over the first one. Similarly, Table 8.4 shows the test application time and the test time reduction when weighted random patterns are used by deploying the modified TPG as described in Chapter 6. The second column shows the best case weight (probability of a bit being 1) for the respective circuit. Also, Table 8.5 shows the test application time and reduction in test time when transition density patterns are applied using the proposed TPG. The second column in this table shows the best case transition density for the respective circuit.

Table 8.6 provides a comparison between required test time to reach 90% fault coverage with adaptive scan clock scheme deployed in conventional random, weighted random and transition density patterns. We note that the pattern choice for best (minimum) test time is circuit dependent.

Table 8.6: Comparing test times for 90% coverage by conventional random (R), weighted random (WRP) and transition density (TDP) patterns when adaptive scan clock is used.

Circuit	Test time (R) Adaptive Frequency (ns)	Weight (WRP)	Test time (WRP) Adaptive Frequency (ns)	Transition density (TDP)	Test time (TDP) Adaptive Frequency (ns)
s298	10050	0.5	10050	0.5	1974026
s382	10320	0.3	6661	0.4	8287
s510	18200	0.4	19570	0.4	19852
s820	348392	0.4	268971	0.4	504453
s953	418073	0.4	162371	0.3	231833
s1196	264652	0.6	221416	0.3	262350
s1488	124572	0.6	117901	0.5	72831
s13207	31565011	0.35	16180025	0.3	10149712
s15850	16341260	0.5	16341260	0.3	20109065

8.3 Power Consumption Analysis

Both the synthesized netlists with adaptive scheme and without adaptive scheme were used to generate SPICE netlists using MentorGraphics tool Design Architect using TSMC018 um.

Sysnopsys Tool NanoSim was used to do the SPICE simulation for capturing power dissipation for the per clock peak power consumption shown in Figures 5.4 and 5.5. Figure 8.1 shows the power consumption for the circuit s1196 for some clock cycles. The peak power consumption is decided by the peak power consumed by the circuit while running in the fixed clock, which is also the slowest clock of the adaptive scheme. The figure plots per clock power consumption against number of test clock cycles. The power budget is set by the peak power consumption by the fixed frequency. It is shown in the figure that though the per clock peak power increases when the scan clock is dynamically adapted but it remains below the power budget line. However, the average power will increase as the time required to apply the test decreases.

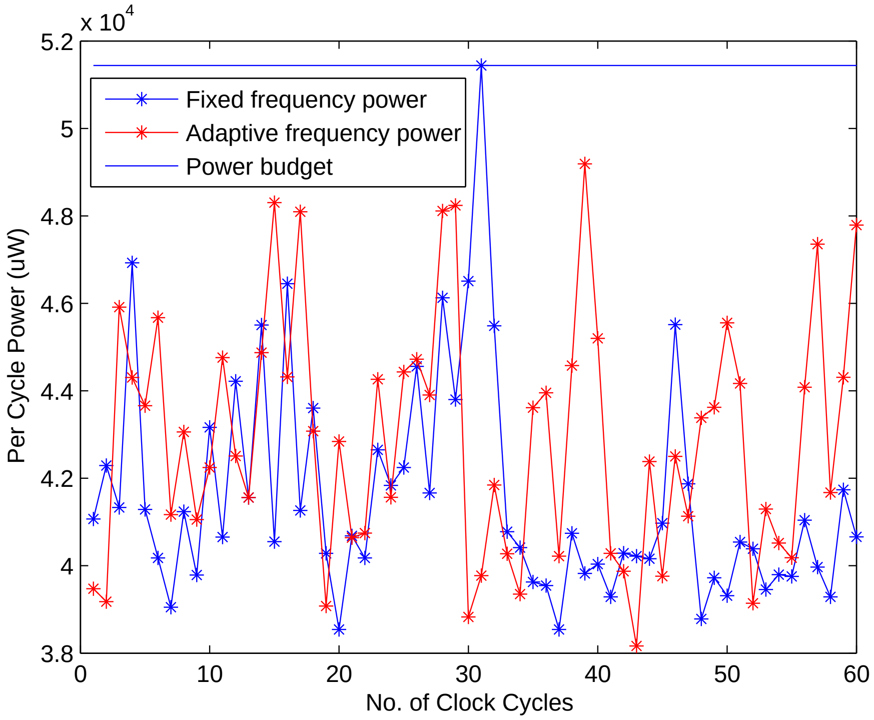

Figure 8.1: Per clock power consumption with and without adaptive schemes for s1196.

8.4 Greedy Algorithm Implementation

For the circuits that have best case transition density around 0.5 or 0.4 the greedy algorithm was implemented on them to reduce scan in time further.

The fault profiles generated by AUSIM [37] were converted into tables for each circuit. A Matlab [29] program was written to take the table as input and apply the greedy algorithm to give the transition density needed to be applied for each partial fault coverage stage. Figures 8.2 and 8.3 shows that the algorithm helps to detect the same number of faults with the same number of vectors as the best case transition density if the vectors were applied with the transition density determined by the algorithm for each partial fault coverage stage.

As seen from the graphs, it is possible to run the test even faster using the controlled transition density mixing in the vector set during the application of lower transition density patterns.

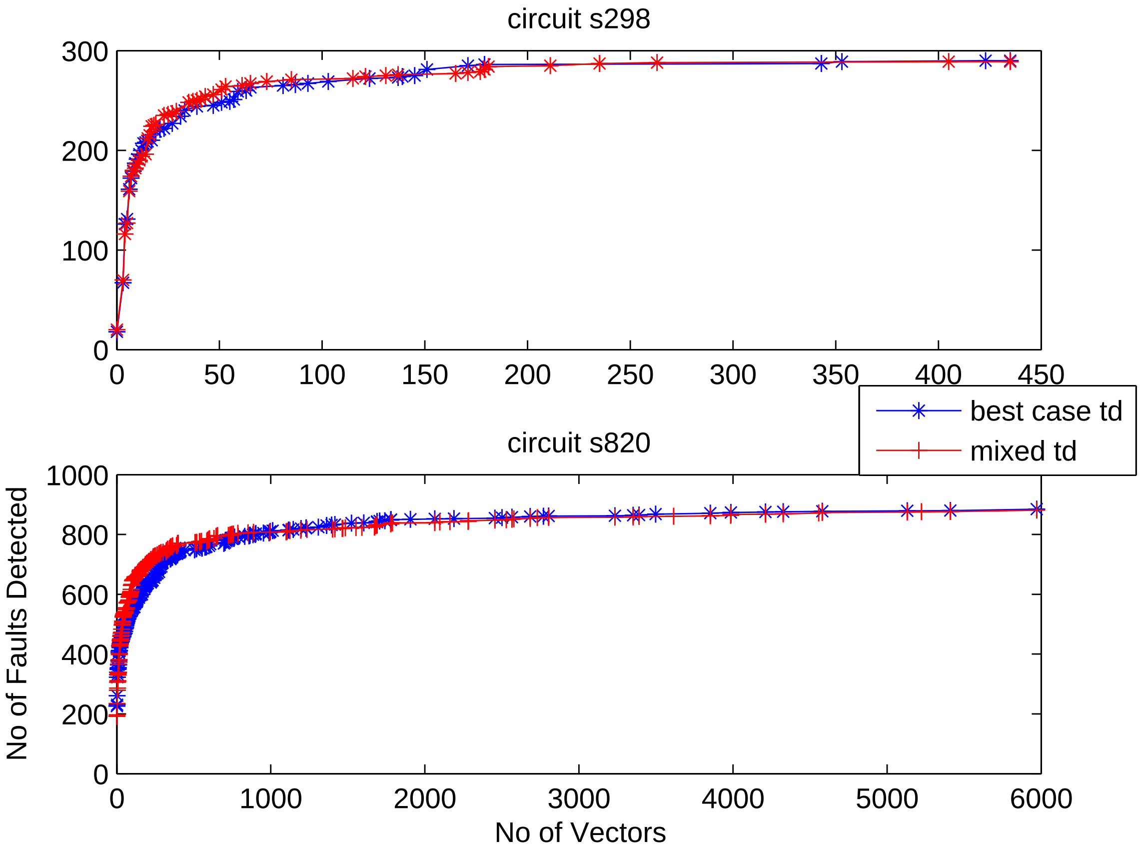

Figure 8.2: Performance of greedy algorithm for s298 and s820.

The lower transition density vectors are as good as the best case transition density until certain percentage faults are detected. This can be seen from the graphs 8.2 and 8.3. Keeping the total number of vectors fixed and mixing the lower transition density vectors in a controlled manner the same fault coverage can be reached.

Table 8.6 shows the increase in time reduction if transition density is chosen by the greedy method in cases where best case transition density lies around 0.4-0.5. The second column in the table shows the best case transition density for the respective circuit. The fifth column shows the number of vectors needed to reach a certain fault coverage for that circuit. The third column shows the reduction in test application time when the scan clock is dynamically adapted as compared with keeping the scan clock at fixed frequency. The fourth column shows test time reduction when transition density is varied according to the

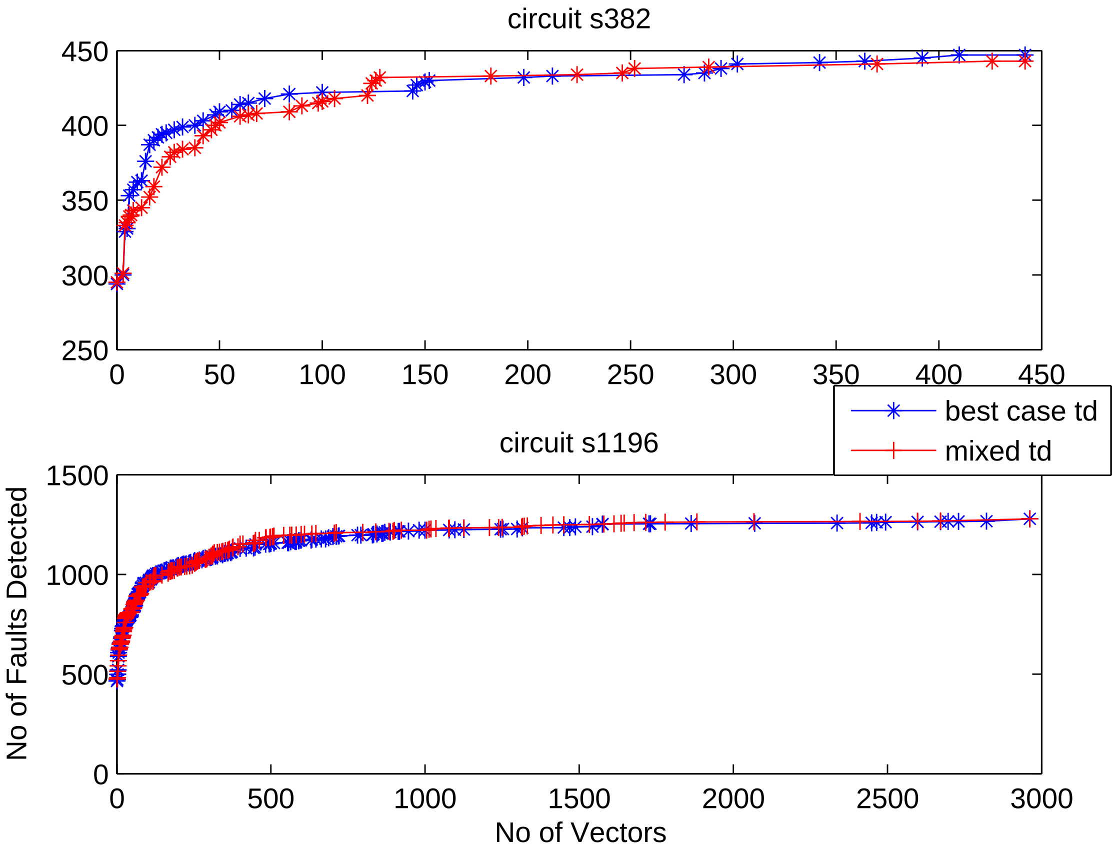

Figure 8.3: Performance of greedy algorithm for s382 and s1196.

algorithm. The sixth column indicates the partial fault coverages (pfc). It is to be noted that, for circuit s298 the test has been applied to reach 77.1% . But for the rest of the circuits, test has been applied to reach above 90% fault coverage.

The seventh and eighth columns show the transition density and the number of vectors needed to run in that transition density to reach the partial fault coverage as indicated in the sixth column.

Running initially in low transition density and accordingly adapting the scan frequency to the reduced transition density allows to reduce the over all test application time.

Table 8.7: Mixing transition densities selected by Greedy Algorithm based on partial fault coverage.

Circuit	TDP	time red.(%)	Alg. time red (%)	Pattern	Pfc %	TD mix	Vector
s298	0.5	12.93	37.5	435	70	0.65	53
					77	0.35	382
s382	0.45	18.8	43.53	442	70	0.1	4
					80	0.2	6
					85	0.4	6
					90	0.4	16
					95	0.35	94
					98	0.4	316
s510	0.5	18.41	42.22	492	70	0.25	16
					80	0.25	14
					85	0.4	14
					90	0.5	32
					95	0.5	76
					99	0.4	310
s820	0.45	24.32	26.51	5972	70	0.4	110
					80	0.4	211
					85	0.4	323
					90	0.45	880
					95	0.45	4448
s1196	0.45	28.78	35.56	2962	70	0.25	83
					80	0.3	169
					85	0.3	148
					90	0.3	312
					95	0.25	2250

Chapter 9

Conclusion

For scan testing it is important to note that both power and test time contribute to the test cost as well as quality of the test. This work proposes to strike a balance between these two factors and aims to reduce test application time as much as possible without sacrificing the fault coverage while keeping the test power controlled the entire time.

The main ideas forwarded in this thesis are, transition density can be effectively selected for any circuit analogous to weighted random patterns to generate test session with shorter test length. Once the transition density is known the test application time can be further reduced by dynamically controlling the test clock keeping the test power controlled.

At the beginning of the test session, any transition density is capable of reaching a fault coverage that is lower than 80% faults with almost same number of vectors. The lower transition density based vectors though need more number of vectors but the difference between numbers of the vectors needed to detect those faults is small. Thus a lower transition density can be chosen deterministically to reach that partial coverage while speeding up the scan clock without crossing the power budget.

Thus using a control mix of transition density, the test session is kept optimal. Power dissipation can be controlled by dynamically adjusting the scan clock with the transition density chosen. The experimental results show a further speed up of 10-15% when transition densities are mixed.

In the future, more sophisticated methods for obtaining the controlled transition density mixing in the vector set by using linear programming should be examined to balance the test time and test power more efficiently.

Bibliography

[1] A. S. Abu-Issa and S. F. Quigley, "Bit-Swapping LFSR and Scan-Chain Ordering: A Novel 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.

[2] P. Agrawal and V. D. Agrawal, "On Improving the Efficiency of Monte Carlo Test Generation," in *Digest of 5th International Fault Tolerant Computing Symp.*, (Paris, France), June 1975, pp. 205–209.

[3] P. Agrawal and V. D. Agrawal, "Probabilistic Analysis of Random Test Generation Method for Irredundant Combinational Networks," *IEEE Trans. Computers*, vol. C-24, pp. 691–695, July 1975.

[4] P. Agrawal and V. D. Agrawal, "On Monte Carlo Testing of Logic Tree Networks," *IEEE Trans. Computers*, vol. C-25, pp. 664–667, June 1976.

[5] V. D. Agrawal, K.-T. Cheng, D. D. Johnson, and T. Lin, "Designing Circuits with Partial Scan," *IEEE Design & Test of Computers*, vol. 5, no. 2, pp. 8–15, Apr. 1988.

[6] V. D. Agrawal, C. R. Kime, and K. K. Saluja, "A Tutorial on Built-In Self-Test, Part 1: Principles," *IEEE Design & Test of Computers*, vol. 10, pp. 73–82, Mar. 1993.

[7] F. Brglez, D. Bryan, and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits," in *Proc. Int. Symp. Circuits and Systems*, May 1989, pp. 1929–1934.

[8] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits.* Springer, 2000.

[9] S. Chiu and C. Papachristou, "A Design for Testability Scheme with Applications to Datapath Synthesis," in *Proc. Design Automation Conf.*, June 1991, pp. 271–177.

[10] R. M. Chou, K. K. Saluja, and V. D. Agrawal, "Power Constraint Scheduling of Tests," in *Proc. 7th International Conference VLSI Design*, Jan. 1994, pp. 271–274.

[11] R. M. Chou, K. K. Saluja, and V. D. Agrawal, "Scheduling Tests for VLSI Systems Under Power Constraints," *IEEE Trans. VLSI Systems*, vol. 5, no. 2, pp. 175–185, June 1997.

[12] F. Corno, M. Rebaudengo, M. S. Reorda, G. Squillero, and M. Violante, "Low Power BIST via Non-Linear Hybrid Cellular Automata," in *Proc. IEEE 18th VLSI Test Symp.*, May 2000, pp. 29–34.

[13] E. B. Eichelberger and E. Lindbloom, "Random-Pattern Coverage Enhancement and Diagnosis for LSSD Logic Self-Test," *IBM Jour. Research and Development*, vol. 27, no. 3, pp. 265–272, May 1983.

[14] S. Gerstendörfer and H.-J. Wunderlich, "Minimized Power Consumption for Scan-Based BIST," *Jour. Electronic Testing: Theory and Applications*, vol. 16, no. 3, pp. 203–212, 2000.

[15] P. Girard, "Low Power Testing of VLSI Circuits: Problems and Solutions," in *Proc. First IEEE symp. Quality Electronic Design (ISQED)*, Mar. 2000, pp. 173–179.

[16] P. Girard, "Survey of Low-Power Testing of VLSI Circuits," *IEEE Design & Test of Computers*, vol. 19, no. 3, pp. 80–90, May-June 2002.

[17] 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.

[18] 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.

[19] P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and H. J. Wunderlich, "A Modified Clock Scheme for a Low Power BIST Test Pattern Generator," in *Proc. IEEE 19th VLSI Test Symp.*, May 2001, pp. 306–311.

[20] P. Girard, N. Nicolici, and X. Wen, editors, *Power-Aware Testin and Test Stargegies for Low Power Devices.* Springer, 2009.

[21] D. Gizopoulos, N. Krantitis, A. Paschalis, M. Psarakis, and Y. Zorian, "Low Power/Energy BIST Scheme for Datapaths," in *Proc. IEEE 18th VLSI Test Symp.*, May 2000, pp. 23–28.

[22] J. Hartmann and G. Kemnitz, "How to Do Weighted Random Testing for BIST," in *Proc. IEEE/ACM International Conf. Computer-Aided Design*, Nov. 1993, pp. 568–571.

[23] A. Hertwig and H.-J. Wunderlich, "Low Power Serial Built-In Self-Test," in *Proc. European Test Workshop*, May 1998, pp. 49–53.

[24] V. Iyengar and K. Chakrabarty, "Precedence-Based, Preemptive, and Power Constrained Test Scheduling for System on a Chip," in *Proc. IEEE 19th VLSI Test Symp.*, May 2001, pp. 368–374.

[25] J. Rajski et al., "Test Generator with Preselected Toggling for Low Power Built-In Self-Test," in *Proc. IEEE 29th VLSI Test Symp.*, 2011.

[26] E. Larsson, *Introduction to Advanced System-on-Chip Test Design and Optimization.* Springer, 2005.

[27] A. Majumder, "On Evaluating and Optimizing Weights for Weighted Random Pattern Testing," *IEEE Trans. Computers*, vol. 45, no. 8, pp. 904–916, Aug. 1996.

[28] S. Manich, A. Gabarro, M. Lopez, J. Figueras, P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, P. Teixeira, and M. Santos, "Low Power BIST by Filtering Non-Detecting Vectors," in *Proc. European Test Workshop*, May 1999, pp. 165–170.

[29] MathWorks, "MATLAB - The Language of Technical Computing." http://www.mathworks.com/products/matlab/, accessed on Mar. 6, 2012.

[30] F. Najm, "Transition Density: A New Measure of Activity in Digital Circuits," *IEEE Trans. CAD*, vol. 12, pp. 310–323, Feb. 1993.

[31] K. P. Parker and E. J. McCluskey, "Probabilistic Treatment of General Combinational Networks," *IEEE Trans. Computers*, vol. C-24, no. 6, pp. 668–670, June 1975.

[32] H. D. Schnurmann, E. Lindbloom, and R. G. Carpenter, "The Weighted Random Pattern Genrerator," *IEEE Trans. Computers*, vol. C-24, no. 7, pp. 695–700, July 1975.

[33] P. Shanmugasundaram, "Test Time Optimization in Scan Circuits," Master's thesis, Auburn University, Dec. 2010.

[34] P. Shanmugasundaram and V. D. Agrawal, "Dynamic Scan Clock Control for Test Time Reduction Maintaining Peak Power Limit," *Proc. 29th IEEE VLSI Test Symp.*, pp. 248–253, May 2011.

[35] P. Shanmugasundaram and V. D. Agrawal, "Dynamic Scan Clock Control in BIST Circuits," in *Proc. Joint IEEE Int. Conf. on Industrial Electronics and 43rd Southeastern Symp. on System Theory*, Mar. 2011, pp. 237–242.

[36] P. Shanmugasundaram and V. D. Agrawal, "Externally Tested Scan Circuit with Built-In Activity Monitor and Adaptive Test Clock," in *Proc. 25th International Conf. VLSI Design*, Jan. 2012, pp. 448–453.

[37] C. E. Stroud, "AUSIM - Auburn University SIMulator." http://www.eng.auburn.edu/users/strouce/ausim.html, accessed on Mar. 6, 2012.

[38] C. E. Stroud, *A Designer's Guide to Built-In Self-Test.* Springer, 2002.

[39] M. Tehranipoor, M. Nourani, and N. Ahmed, "Low Transition LFSR for BIST-Based Application," in *Proc. IEEE 14th Asian Test Symposium*, 2005.

[40] K. Thearling and J. A. Abraham, "An Easily Computed Functional Level Testability Measure," in *Proc. International Test Conf.*, Aug. 1989, pp. 381–390.

[41] N. A. Touba, "Survey of Test Vector Compression Techniques," *IEEE Design & Test of Computers*, vol. 23, no. 4, pp. 294–303, Apr. 2006.

[42] vSMP HPCC, "Virtual Symmetric Multiprocessing High Performance Compute Cluster." http://www.eng.auburn.edu/ens/hpcc/index.html, accessed on Mar. 6, 2012.

[43] S. Wang, "Generation of Low Power Dissipation and High Fault Coverage Patterns for Scan-Based BIST," in *Proc. International Test Conf.*, Dec. 2002, pp. 834–843.

[44] S. Wang and S. K. Gupta, "DS-LFSR: A New BIST TPG for Low Heat Dissipation," in *Proc. International Test Conf.*, Nov. 1997, pp. 848–857.

[45] S. Wang and S. K. Gupta, "LT-RTPG: A New Test-Per-Scan BIST TPG for Low Heat Dissipation," in *Proc. International Test Conf.*, Sept. 1999, pp. 85–94.

[46] F. M. Wanlass and C. T. Sah, "Nanowatt Logic using Field-Effect Metal-Oxide-Semiconductor Triodes," in *IEEE International Solid-State Circuits Conf. Digest*, volume IV, Feb. 1963, pp. 32–33.

[47] H.-J. Wunderlich, "On Computing Input Probabilities for Random Tests," in *Proc. Design Automation Conf.*, June 1987, pp. 392–398.

[48] X. Zhang, K. Roy, and S. Bhawmik, "POWERTEST: A Tool for Energy Conscious Weighted Random Pattern Testing," in *Proc. 12th International Conf. VLSI Design*, Jan. 1999, pp. 416–422.