DESIGN OF ON-CHIP SELF-TESTING SIGNATURE REGISTER

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF TECHNOLOGY

IN

VLSI DESIGN AND EMBEDDED SYSTEM

BY

LODHA KALPESH RAJENDRA

212EC2130

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

ODISHA, INDIA. 769008

2014

DESIGN OF ON-CHIP SELF-TESTING SIGNATURE REGISTER

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF TECHNOLOGY

IN

VLSI DESIGN AND EMBEDDED SYSTEM

BY

LODHA KALPESH RAJENDRA

212EC2130

Under the Guidance of

PROF. KAMALA KANTA MAHAPATRA

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

ODISHA, INDIA. 769008

2012 – 14

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA
ODISHA, INDIA- 769008

CERTIFICATE

This is to certify that the thesis report entitled

DESIGN OF ON-CHIP SELF-TESTING SIGNATURE REGISTER

submitted by

Mr. Lodha Kalpesh Rajendra

bearing roll no. **212EC2130**
in partial fulfilment of the requirements for the award of

MASTER OF TECHNOLOGY
in
VLSI DESIGN AND EMBEDDED SYSTEM

during session 2012-14 at National Institute of Technology, Rourkela, is an authentic work carried out by him under my supervision and guidance.
To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/institute for the award of any degree or diploma.

Prof. Kamala Kanta Mahapatra
Department of ECE,
National Institute of Technology,
Rourkela.

Date: Jun 2nd, 2014

Dedicated to

Dr. Rajendra Lodha,

Mrs. Sandhyarani Lodha

and

Mrs. (Dr.) Deepali Vishal Kankariya

whose inspiration

have always motivated me

towards my goal.

CONTENTS

ACKNOWLEDGEMENTS

I have procrastinated (*Procrastinate:* ***verb.*** *postpone the action, The Oxford Dictionary*) writing this passage to the most recent moments prior the printing of this thesis, because I assume, it is the toughest and undeniably the most read part of this dissertation. According to the synaptic information stored in my Central Nervous System (CNS), it has been forty-one days since I composed the first sentence for this manuscript, nearly thirteen months (precisely talking, 392 days) since I started the research which steered to this and altogether almost twenty-two months since I officially got associated with NIT Rourkela for my post-graduation studies. And what this entire period of time have been; packed with profound knowledge and splendid experience, hard work for fun, enjoyment and frustration, teamwork and friendship. At the end of this this wonderful period of time, I wish to give away a vote of thank to all the people who held my hand for pursuing an M. Tech degree.

First and foremost, I am grateful to my research advisor Prof. K. K. Mahapatra, for the opportunities he spawned for me, for promoting to learn the topic absolutely new for me, for his invaluable advice on the big or small problems, for his encouragements, and for believing my abilities throughout the period. I am also greatly indebted to Mr. Sudeendra Kumar, a Ph.D. scholar at NITRKL, for introducing the exciting topics in this domain to me, an infant M. Tech student who started journey on unknown paths in this space. Without his knowledge and priceless guidance for this research, boundless efforts and patience, and long lasting technical discussions, this research would have never been fruitful. Many thanks to both of you for such unmatched support.

I express my whole-hearted gratitude to Prof. D. P. Acharya, Prof. P. K. Tiwari, Prof. A. K. Swain, and Prof. N. Islam for their thoughtful teaching and suggestions during my courses in M. Tech and making available all necessary facilities and infrastructure for

studies. I am also thankful to all research scholars in VLSI lab and all other labs for maintaining the lab, availing access to the same 24 ×7 and creating vibrant atmosphere for research.

Being an M. Tech student here, I had a privilege of getting and working together with many of my peers nationally. Stay with such diversified classmates have made it more delighted and memorable. Thanks to Neel Kamal, one who always give a try, Seshagiri (well known as Seshu), a philanthropist by heart, Yamini, a creative girl, and Anurag, a stylish personality and my roommate, for making joyful learning here.

Finally, I owe my heartiest gratitude to my father Dr. Rajendra Lodha, mother Mrs. Sandhyarani Lodha and sister Mrs. (Dr.) Deepali Vishal Kankariya for their unconditional support, love, inspiration and sacrifices.

Kalpesh Rajendra Lodha

ABSTRACT

Testing is final step of any experiment, project or product manufacturing process. It endorses the correctness of functionality and validate performance of manufactured product. As the result of ever miniaturizing feature size, intensifying density and multi-functionality of today's integrated circuits (ICs), testing with the visual inspection and exhaustive functional check have become impossible. Thus to abet test, Design for Testability (DFT) has turn out to be an integral part of modern ASIC (Application Specific Integrated Circuits) design flow. Improved DFT methodologies along with efficient Automatic Test Pattern Generation (ATPG) algorithms assist high quality test of complex ASIC. Scan design is one of the DFT methodology that engineers predominantly incorporate in the design. Because of simplicity and low area overhead to the design, over the years, it has become the standard of digital IC testing.

However, from last few years, scan test has turn out to be too expensive to implement for industry standard designs due to expanding test data volume and augmented test time. The test cost of a chip is mainly governed by the resource utilization of Automatic Test Equipment (ATE). Also, it directly depends upon test time that includes time required to load test program, to apply test vectors and to analyze generated test response of the chip. An issue of test time and data volume is increasingly appealing designers to use on-chip test data compactors, either on input side or output side or both. Such techniques significantly address the former issues but have little hold over increasing number of input-outputs under test mode. Further, test pins on DUT are increasing over the generations. Thus, scan channels on test floor are falling short in number for placement of such ICs.

To address issues discussed above, we introduce an on-chip self-testing signature register. It comprises a response compactor and a comparator. The compactor compacts large chunk of response data to a small test signature whereas the comparator compares

this test signature with desired one. The overall test result for the design is generated on single output pin. Being no storage of test response is demanded, the considerable reduction in ATE memory can be observed. Also, with only single pin to be monitored for test result, the number of tester channels and compare edges on ATE side significantly reduce at the end of the test. This cuts down maintenance and usage cost of test floor and increases its life time. Furthermore reduction in test pins gives scope for DFT engineers to increase number of scan chains so as to further reduce test time.

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

The following is the list of abbreviations that are encountered in this thesis.

ASIC	Application Specific Integrated Circuits
ATE	Automatic Test Equipment
ATP	Automatic Test Patterns
ATPG	Automatic Test Pattern Generation
BIST	Built-In Self-Test
CAD	Computer Aided Design
CMOS	Complementary Metal Oxide Semiconductor
DC™	Design Compiler
DFS	Design for Security
DFT	Design for Testability
DUT	Design under Test
HDL	Hardware Description Language
IC	Integrated Circuit
IP	Intellectual Property
ISCAS	International Symposium on Circuits and System
ITRS	International Technology Roadmap for Semiconductors
LFSR	Linear Feedback Shift Register
LSSD	Level Sensitive Scan Design
MISR	Multiple Input Signature Register
OPMISR	On-Product Multiple Input Signature Register
ORA	Output Response Analysis
PDF	Path Delay Fault
PI	Primary Input
PO	Primary Output
RTL	Register Transfer Logic
S-A-0/1	Stuck-At-1/0
SAF	Stuck At Fault
SDD	Small Delay Defect
SEM	Scanning Electron Microscope
SIPO	Serial-In Parallel Out

SISR	Single Input Signature Register
SoC	System on Chip
SPF	STIL procedure file
STA	Static Timing Analysis
STIL	Standard Test Interface Language
TDF	Transition Delay Fault
TSMC	Taiwan Semiconductor Manufacturing Company
VLSI	Very Large Scale Integration

PUBLICATION

- **Kalpesh Lodha**, Sudeendra Kumar and K. K. Mahapatra, "A Novel On-Chip Self-Testing Signature Register for Low Cost Manufacturing Test," 18th Int. Conf. on VLSI Design and Test, 2014 (Communicated).

Chapter 1
OVERVIEW

1.1. INTRODUCTION

It's a human tendency to attain perfection in his deeds. But one hardly achieves it in first attempt. It is an iterative process. It does not make any different when it comes to the ICs where millions of devices have to work together for particular application. Over the years, continuously shrinking feature size and advancements in fabrication technology have facilitated high level of integration of logic on single piece of silicon that itself is humanly unperceivable. However, fabricating the reliable electronic devices in dense designs is a challenging task. Increasingly complex and imperfect thermal, chemical and mechanical processes involved in clean rooms introduce defects in fabricated ICs. It may cause incorrect functionality, alteration in performance or an IC may not work at all. Some of the factors those are very vital in limiting the test confidence are the increased operating clock frequency, increased transistor density, integration of mixed signal devices onto single chip and the lack of accurate CAD tools those consider various process parameters in life cycle of an IC. The guaranteed working of these components at every instance necessitates testing the same effectively. Thus every manufactured chip must be tested thoroughly before shipping it to the customer.

This necessity has given birth to the new firm in Silicon Valley called Test Centers. Test Centers have huge setup for high quality and high precision Automatic Test Equipment (ATE) for testing high performance ICs. The cost of ATE is mainly governed by test clock frequency, test precision, test accuracy, memory requirement and the total number of scan channels available on test floor. However, in line with Moore's law [1], prophesied in 1965 and has remained true till the date, the functional complexities, circuit density and performance of ICs are persistently escalating. To keep in pace with this, test centers have to upgrade their setup periodically. It so happens that for every three years down the line, test centers need to install new ATEs sufficing the requirements for the test of the modern

ICs. This installation incur multimillion dollar investment. That in turn augments the cost of testing, charged by test centers on the scale of test time. Besides, there are some chips, such as semiconductor memories, which are mass produced and have small market window. The profit to the design firms, out of such products, is mainly dictated by cost to test them, which in turn is the time to test them. This illustrates that time to test the IC is a crucial factor in deciding the cost of the IC, which falls in its recurring cost.

1.2. MOTIVATION

Functional complexities and performance of VLSI systems are rising up relentlessly. As a result of this, testing of such designs has become critical in various dimensions such as: number of test pins on IC has been increasing demanding same increment on test head of ATE, number of test patterns has increased and so is its response volume that burdens huge chunk of ATE memory to store the same, high performance designs need testing the same critical and precise in time. This requires very accurate launch and capture mechanism on test floor. Again various new defects, for example small delay defects (SDD), have emerged out in designs below 45nm technology. Testing these along with traditional manufacturing defect models, like stuck-at faults, delay faults etc., augments test time per device under test (DUT), dropping the throughput of test floor and increasing test cost.

Test time and data volume issues increasingly appeal designers to use on-chip test data compactors. International Technology Roadmap for Semiconductors (ITRS) has predicted that in future managing test data volume and associated test cost will be a prime challenge in front of designers [2]. They also have forecasted that the flat data volume for MPU will approach petabits by 2030 and same for System on Chip (SoC) will approach near 10 terabits. *Fig. 1.1* shows this guesstimate over progressive years. Thus in future, compression will become ubiquitous across component business segment.

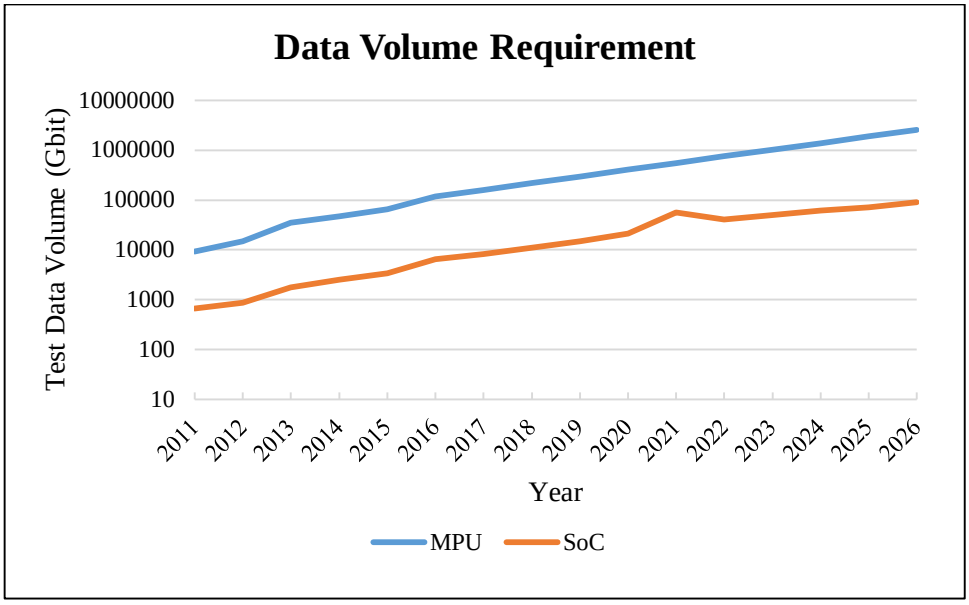

Fig. 1.1 Trend for test data volume over the years (Courtesy: ITRS 2012)

Hurst has given the concept of an ideal test system as shown in *Fig. 1.2* [3]. It shows an additional input pin on DUT to toggle the same between normal and test mode of operation and an output pin to indicate the final test result, this being simple PASS/FAIL indication. Achieving confidence in this test procedure needs no failure of PASS/FAIL test circuit.

An output compression circuit, indicated in *Fig. 1.2*, must have very high compaction ratio, excellent fault detecting capability and design simplicity with little area overhead to the design.

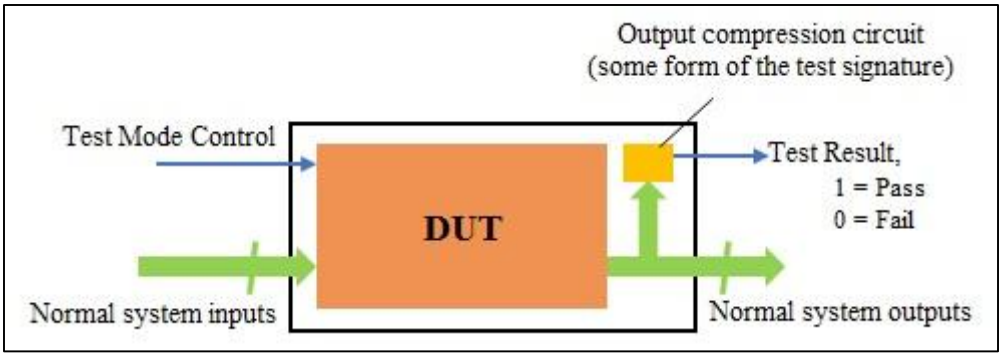

Fig. 1.2 Ideal Test System [3]

1.3. RESEARCH OBJECTIVE

An engineering solution to an ideal test configuration, indicated in *Fig. 1.2*, would be a design of on-chip test architecture comprising an efficient output response compactor and an in-built golden response comparator with ability to detect maximum targeted faults under test with minimum probability of fault masking. Besides this, it must not compromise performance of the design and should have little power and area overhead.

The research was started aiming reduction in test cost for multifunctional and high performance VLSI ICs tested using *de facto* scan test methodology. The parameters focused for test cost reduction are ATE resources such as memory, tester channels, launch and compare edges per tester cycle and throughput of test head.

1.4. CONTRIBUTION OF THIS DISSERTATION

We have proposed and designed a novel on-chip self-testing signature register. This compacts the test response to the small test signature and compares the same with golden one. It generates two bits of PASS/FAIL test result on single pin irrespective of the count of scan chains in design. The response compaction has been achieved by Linear Feedback Shift Register (LFSR) based signature register and a signature comparator has been realized with Ex-OR + OR logic tree.

We have shown that testing using the proposed test architecture significantly reduces the memory and test channels requirement on ATE. Besides, using proposed architecture reduces number of primary inputs (PIs) and outputs (POs) in test mode. This allows DFT engineers to increase scan chains in the design that further reduce test time and hence test cost. Besides this, it has been observed that, the proposed signature register is well suited for testing cryptographic designs where there is always a threat for using scan methodology for testing it.

1.5. ORGANIZATION OF THE THESIS

Following an overview, rest of the dissertation is structured as follows:

- Chapter 2: This chapter introduces the fundamentals of VLSI testing. Among various types of tests, it largely focuses on the test for manufacturing defects, their fault modeling and DFT for the same.
- Chapter 3: This chapter defines the flow of ATPG for stuck-at faults (SAF) and path-delay faults (PDF) using industry standard tools. Same flow has been maintained throughout the experimentation for purpose of ATPG.
- Chapter 4: This chapter details various techniques for output response analysis (ORA). Here, signature analysis, one of the ORA techniques, has been discussed thoroughly.
- Chapter 5: This chapter proposes an on-chip self-testing signature register. Various challenges faced using conventional signature analysis techniques are explored here and their mitigation using the proposed one is exemplified with an experimentation.
- Chapter 6: This chapter lists various domains in which the proposed signature register find its application.
- Chapter 7: This chapter gives the concluding remarks for this dissertation with the discussion on future scope for this work.

Chapter 2
INTRODUCTION TO VLSI TESTING

2.1. VERIFICATION V/S TESTING

2.2. NEED FOR TESTING

2.3. CLASSIFICATION OF VLSI TESTING

2.4. FAULT MODELING

2.5. DESIGN FOR TESTABILITY

2.1. VERIFICATION V/S TESTING

Verification and test related activities are distributed throughout the lifetime of electronic devices. Verification is an analytical and extrapolative analysis to ensure that the synthesized design, after fabrication, will operate in consistent with its desired functionality. On the other side, testing is a post-manufacturing process that ensures that the actual device, fabricated from the synthesized netlist, has no manufacturing defect. The exact distinction between these two processes is tabulated in *Table 2-I.*

TABLE 2-I
VERIFICATION V/S TESTING

Verification	Testing
Verification is a pre-silicon process.	Testing is a post-silicon process.
It verifies correctness of the design and proves that design is mathematically equivalent.	It validates the manufacturing of the fabricated device.
It is usually performed by either simulation, or formal equivalence methods.	Testing is performed in two phases: 1. Test generation: It is performed using a sophisticated computer program once during the design. 2. Test application: In this process, the test patterns are electrically applied at the inputs of hardware.
Verification is performed once prior to the manufacturing.	Every fabricated device undergoes testing.
In design phase, this step is responsible for quality of the design.	Testing is dictates the quality of the device.

2.2. NEED FOR TESTING

Small size and high level of integration makes visual inspection of little use for manufactured electronic devices to check their correctness. To achieve this, one has to rely on statistical or functional test. The failures in electronic circuits may either be due to the wrong test procedure, or imperfection in fabrication process, or incorrect design. Testing detects whether something went wrong in either of these activities.

High density circuits has higher failure probability. This puts question mark on functional correctness and specifications of the design. Both IC manufacturer and the IC designer falls inside the contour of testing. Testing ensures correct implementation of all fabrication steps, functional correctness, reliability and performance of IC. The quality of IC shipped to the customer depends on its test quality measured in terms of test coverage and types of physical defects taken into consideration. An ideal test detects all defects occurred in the fabrication process and segregate bad chips from the produced lot. But such test burdens very large variety and count of physical defects to be tested which often turn out to be impossible for some real defects. The practical test is based on analyzable fault models, which simplifies test without directly mapping actual faults and reason behind them. Also, due to high complexity, test may not give complete coverage of the design.

An important aspect of VLSI testing is the cost for executing the same. An electronic component can be tested at various levels, such as core, chip, board, system or operational system. However, it is crucial to consider that the test cost increases 10 folds when going from one level of abstraction to the next higher one [4]. Thus it is always profitable to detect the faults at early phases of design process. *Fig. 2.1* explains this phenomenon with cost pyramid where area occupied by each level represents its testing cost.

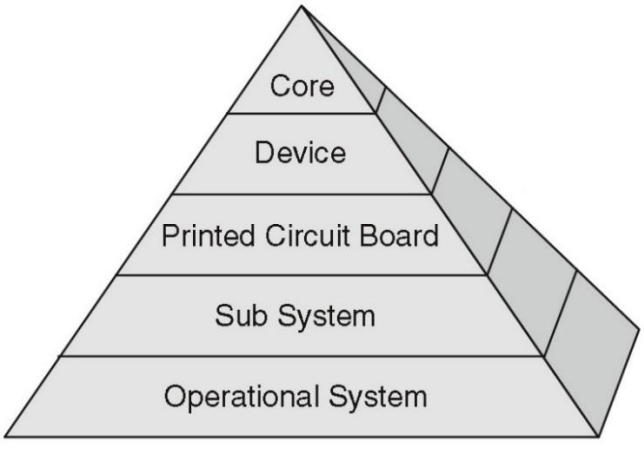

Fig. 2.1 Cost pyramid [4]

This cost pyramid indicates core at the top with minimum base area thus minimum test cost while on field operational systems at the bottom with highest cost of testing which is 10,000 times more than that of the core.

2.3. CLASSIFICATION OF VLSI TESTING

As discussed in previous section, testing is a post-silicon validation of design. Depending on the purpose it undertakes, it can be classified in four types [5].

2.3.1. Characterization Testing

Characterization test is performed to verify correctness and identifies logical and electrical bugs in a design before sending it to mass production. Here functional test patterns are applied to make comprehensive DC and AC analysis. It may require to check internal nodes of chip. For this purpose assistance of some sophisticated tools like scanning electron microscope (SEM), electron beam tester etc. may require. Characterization tests are time taking and involve rigorous analysis. Also, they determine exact limit on device operating values of power supply and clock.

2.3.2. Production Testing

All fabricated chips are passed through production test, also known as manufacturing test. This test is less comprehensive compared with previous one and performed after characterization. Test vectors for this test need not cover all functions however, they must have high coverage for targeted faults for making pass/ fail decision. Since every devices must undergo this test, this test must be completed within short interval of time to reduce test cost.

2.3.3. Burn-in

All electronic devices passing the aforementioned production test cannot be guaranteed to be identical and fault-free. The potential failures in device may enhance at

the raised temperatures. Burn-in test certifies reliability of the device by testing either unceasingly and/or sporadically, for long duration of time, triggering weak devices to fail in reality. A device may be subjected to burn-in for short-term (10-20 hours) or for long-term (100-1000 hours) at elevated temperature and/or over-voltage supply.

2.3.4. Incoming Inspection

It is performed by system manufacturer on purchased components before integrating them to form a system. The purpose of this test is to avoid the placement of unreliable component in system assembly, since cost of their future diagnosis may surpass that for incoming inspection. This test may be similar to or more comprehensive than production testing or even application specific.

2.4. FAULT MODELING

Prior to the discussion of various types of models for realizing imperfections in fabricated chips, some terminologies need to be clearly defined in this regards.

The term *defect* in system refers to the physical imperfection in manufactured device used to assemble the system.

A *fault* on the contrary, is the depiction of the defect meant for understanding it to the simulators for its analysis on device.

An *error* is result of presence of a defect, and it occurs when a defective device causes a signal to have an incorrect value.

A *failure* is said to be occurred when a defect causes a malfunctioning of a system that cannot be overturned or recovered.

With assumptions that certain defects are likely to occur in ICs, faults modeling has to perform in support with available CAD (Computer Aided Design) tools. Further discussion in this section explores different types of fault models.

2.4.1. Functional Faults

A fault which modifies operation of an electronic system is functional fault [5] [6]. A gate-level functional fault modifies truth table of a gate. Similarly at system-level, its presence results system to operate different than the desired one. *Fig. 2.2* shows functional faults for inverter due to indicated defect. Moving to the next level of abstraction, *Fig. 2.3* shows functional fault in 2:1 multiplexer that uses faulty inverter in *Fig. 2.2*. Presence of this fault changes logic equation of 2:1 multiplexer to *A + SB*.

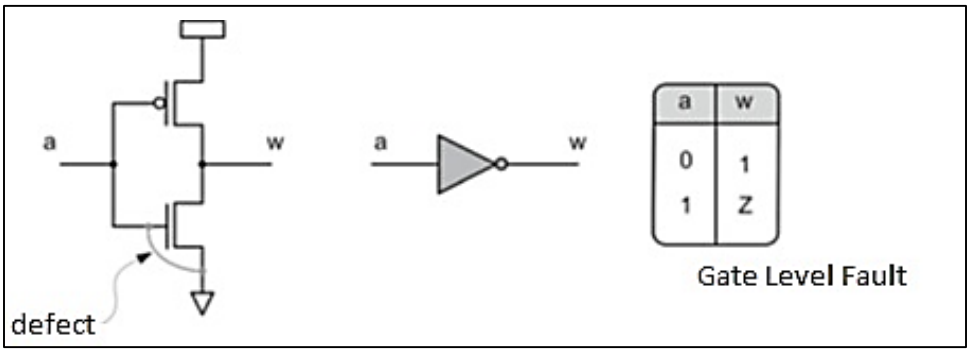

Fig. 2.2 Faulty inverters [4]

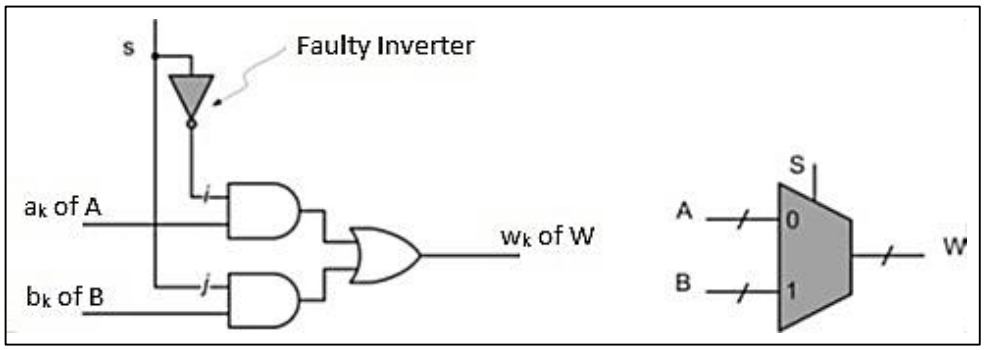

Fig. 2.3 Multiplexer using faulty inverter [4]

The presence of the functional fault in a component at certain level of abstraction ignores the information of the component at its lower level of abstraction. It only considers the terminal behavior of component. Such fault model ignores the exact source of the malfunctioning of the device.

2.4.2. Structural Faults

Functional fault model is competent in representing faults at particular abstraction level but modeling the same for analysis purpose needs elaborate analysis down to the

device level. In contrast to this, structural fault model assumes a fault-free components with only the interconnection of the components may have potential faults. Rest of this section discusses different fault models.

2.4.2.1. Stuck-At Faults

The stuck-at 0/1 (S-A-0/1) fault models the defect on interconnects such that the interconnect line is assumed to be always at logic 0/1value, irrespective of logic it has driven to through primary inputs. This situation is pictorially explained in *Fig. 2.4* and *Fig. 2.5* with gate level schematic for 2:1 multiplexer. Former shows line l_3 S-A-0 which models faults on line l_3 and gate G_3 in shaded area. Similarly, later shows line l_5 S-A-1 which models faults on input I_2 and gate G_2 in shaded area.

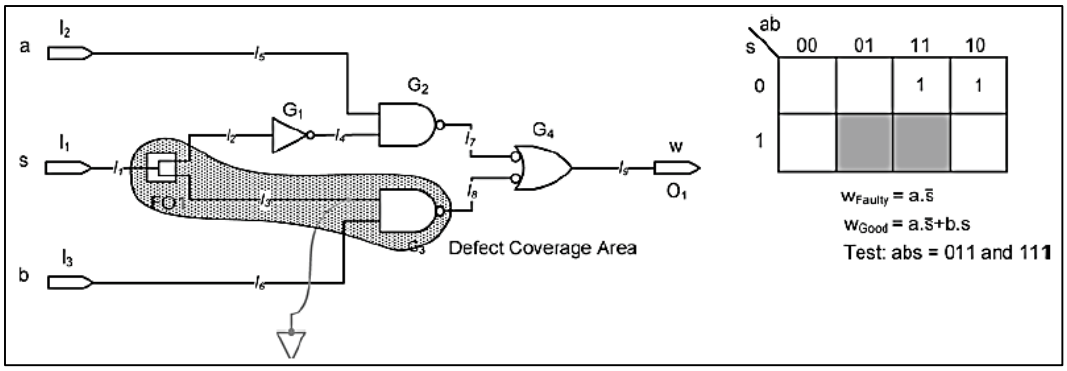

Fig. 2.4 Stuck-at-0 (S-A-0) fault [4]

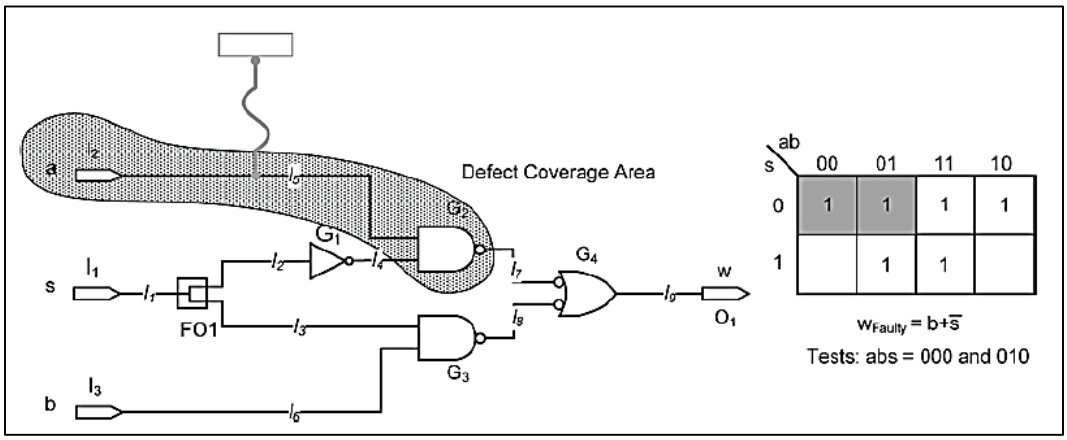

Fig. 2.5 Stuck-at-1 (S-A-1) fault [4]

2.4.2.2. Bridging Faults

Bridging fault models the defects occurred due to presence of neighboring gates and lines in the layout of circuit. Depending upon the technology and process of fabrication these faults may either be logic AND-type (aka AND bridging) or OR-type (aka OR bridging) [7]. An AND (OR)-bridging of two defective lines appear just as they are performing a logic AND (OR) operation, feeding the same value to their destinations. These faults are likely to occur near long parallel lines and those with reduced spacing between them. *Fig. 2.6 (a)* shows microscopic view of the bridging fault in IC [8] and *Fig. 2.6 (b)* shows potential bridging fault locations in layout of design [9].

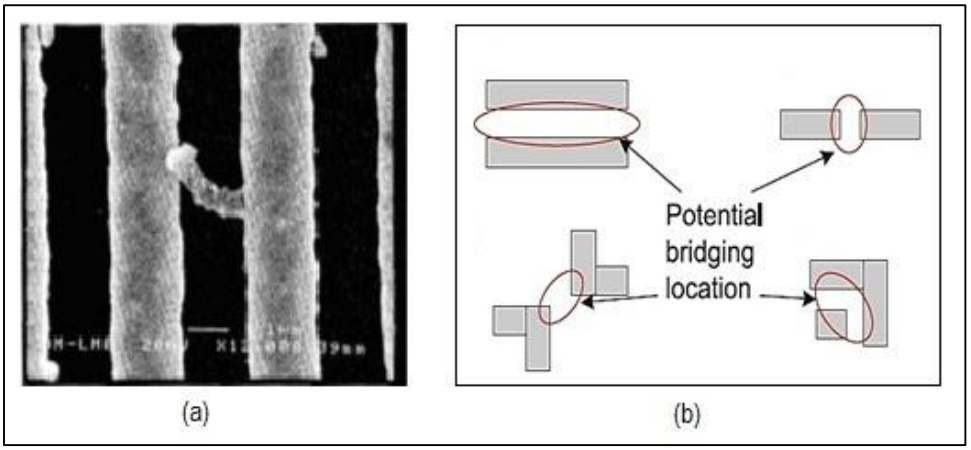

Fig. 2.6 (a) Bridging fault in interconnects in IC [8] (b) potential bridging fault sites [9]

2.4.2.3.Quiescent Current (I_{DDQ}) Fault

I_{DDQ} fault is pertinent to the CMOS technology. CMOS logic gate does not form a conducting path to sink current from power supply, in its steady state. Thus, the Q-current of a gate, in this state, is in the order of a few microamperes only. However, in presence of manufacturing faults, this may upsurge by several orders. This allows fault detection through current monitoring.

2.4.2.4. Delay Faults

Scaling down the technology, some defects in the device likely to amend the performance of design rather their logical functionality. It is not possible to detect such

faults using functional patterns like those used for detecting stuck-at fault in design. This fault model assumes that the defective device performs slower signal propagation on an interconnecting line. This also considers many physical imperfections in actual silicon, like noise, temperature, the effects of process variations, crosstalk, etc. [9]. Two types of delay faults are cited in literatures. Those are:

A. Transition Delay Fault

A transition delay fault (TDF) on an interconnecting line between the gates makes slower signal change on the line, thus create a source of error for signal propagation. This degrades the performance of circuit. TDF is primarily used to model defects in gates and their interconnecting lines which adds up sufficient delay to cause an erroneous signal propagation on the line that runs through such faulty sites to the observation points. For every fault site any of two types of TDF faults possible, *viz.* slow-to-rise (↑) and slow-to-fall (↓). *Fig. 2.7* shows an example of slow-to-rise TDF at line D_2 in a sample circuit.

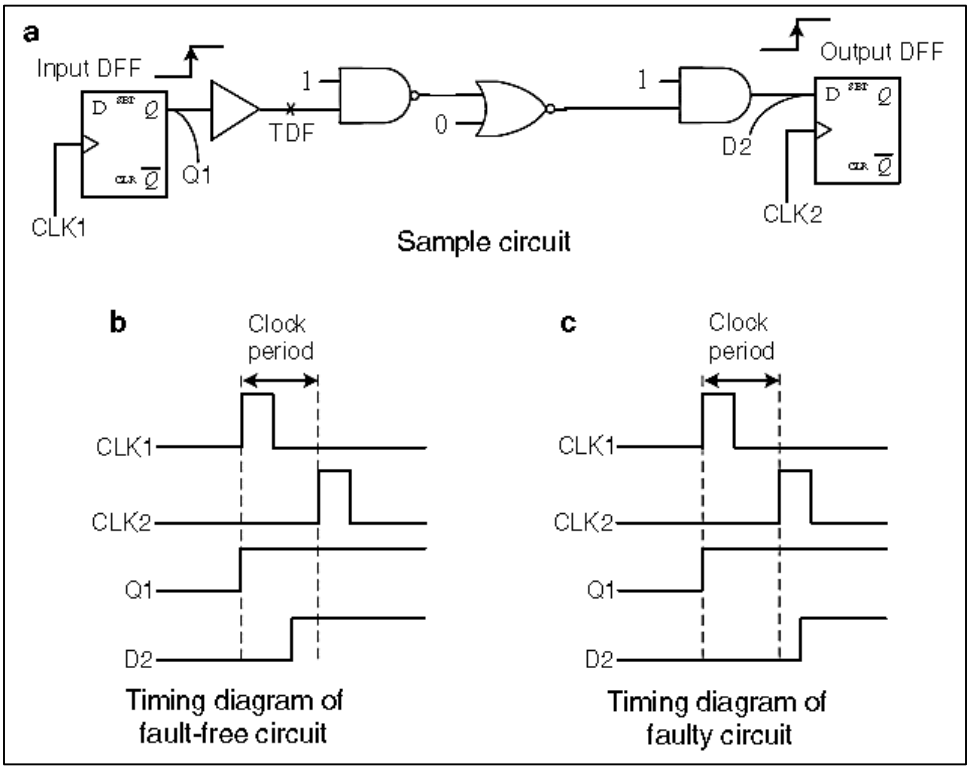

Fig. 2.7 An example of slow-to-rise transition delay fault [9]

B. Path-Delay Fault

This delay fault (PDF) model takes up a cumulative effect of localized delay defect alongside a selected combinational path [10], causing an increment in the propagation delay of the signal to exceed the on-path slack. The path begins either at PI or at an input of flip-flop, passes through a series of interconnected gates, and ends either at an output of flip-flop or PO. The time duration specified for particular path can be the clock period, or the test vector period. Propagation delay for a path is the time taken for the transition of signal through the path. Thus, similar to transition delay fault, path-delay fault also assumes two types of faults, depending on the falling and raising transition, respectively.

The path-delay fault model models the cumulative effect of delay defects throughout the path. This makes it superior to the TDF in its modeling ability [9]. Nevertheless, the total count of paths surge exponentially with the circuit size that makes the tool almost impossible to compute all paths. Thus, a small set of critical paths are selected for generation of test patterns targeting path-delay faults.

2.5. DESIGN FOR TESTABILITY

Test cost and test quality are trade-off, governed by test time of IC. Thus compact test pattern generation and smart test plan in early stage of design is necessary to build enough confidence on manufactured chip. Usually testing is abetted by adding extra logic to the design. An engineering term for this procedure is design for testability (DFT) [5]. It is practiced to address these challenges and adds certain testability features to a design, keeping the circuit functionality and specifications unchanged. The advantage of the additional features is to ease the pattern generation and their application to the DUT.

DFT may also be associated with the design changes that provides an access to the internal nodes so as to control their local internal state (aka controllability) and/or to observe the same (aka observability) with less effort [11]. Testing is usually abetted by

adding extra logic to the design (e.g., inserting a multiplexer at test points) improve its controllability/observability or can be physical in nature (e.g., adding a probe point at test point) or both.

Electronic design mainly comprise three types of components: (a) digital logic, (b) analog or mixed-signal circuits, and (c) memory blocks. Specific DFT methodologies are available for each type of component [5]. Digital logics commonly use scan design and build-in self-test as a DFT methodology.

Over the years, scan design has become a standard methodology for testing digital ICs [12]. *Fig. 2.8* shows architecture for scan testing. Here a multiplexer is inserted at the input port of every flip flop for switching the circuit operation from normal mode to test mode. Switched to the test mode of operation, all flip flops in a design form a chain aka scan chain. This facilitates to initialize state value of each flip flop in a design. Detailed implementation of scan architecture is further discussed in section *5.4.2.1*.

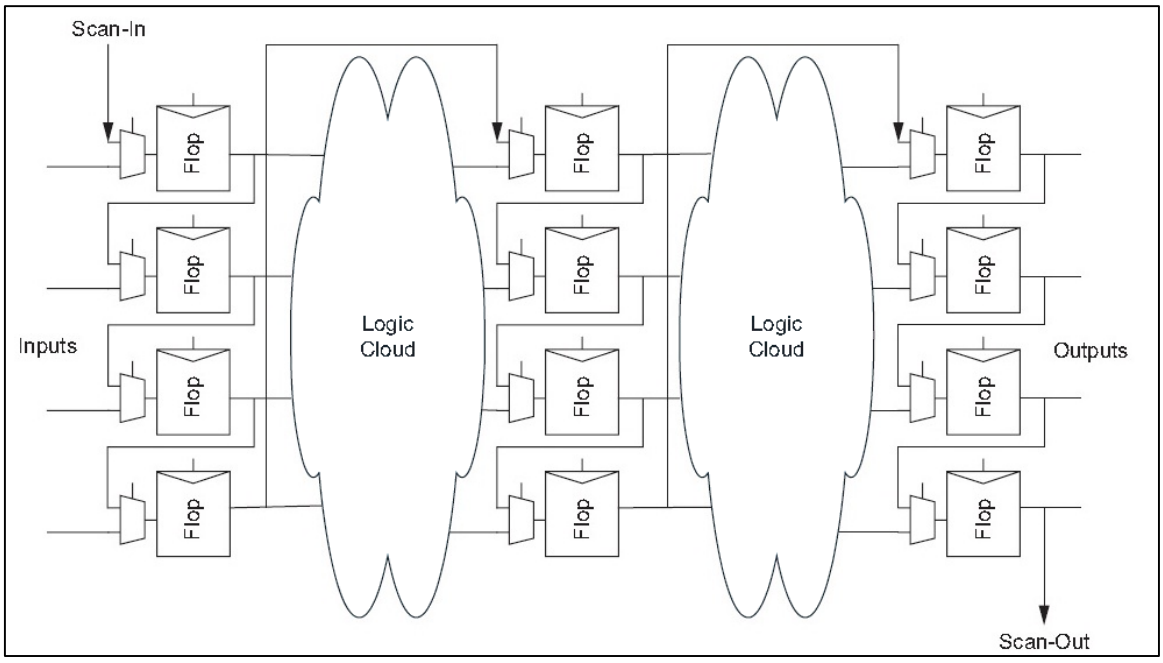

Fig. 2.8 Scan Architecture [12]

Test of scan based design is performed in two phases [5]. In first phase, scan chain is tested by shifting particular sequence of scan bits through primary input pin scan-in for. This phase is also known as scan shift test. An output bit stream is observed at primary output pin scan-out after the number of clock cycles that are equal to the number of flip flops in the scan chain. Observing that both input and output bit streams are same, the test patterns are applied to the DUT in second phase. The response is then downloaded on ATE memory for bit by bit comparison with desired test response.

Putting all the flip flops in a single scan chain increases test time exorbitantly. To minimize this, the flip flops are often grouped in multiple scan chains each having separate scan-in and scan-out as primary input and output respectively. The test mode selection pin remains same for all scan chains in a design. Thus number of test pins on DUT and hence scan channels on test floor demanded for scan methodology is twice the number of scan chains in design.

Increase in the number of scan chains reduces test time linearly but increases the demand of scan channels on ATE which is a limited and expensive ATE resource [5] [13]. Again increased number of scan chains increase amount of data generated thus increasing demand of larger tester memory. All of these reasons make conventional scan test methodology too expensive to implement in today's VLSI designs. It is our endeavor here to present a design which can achieve the complementary objectives simultaneously.

Chapter 3
AUTOMATIC TEST PATTERN GENERATION

3.1 INTRODUCTION

Test vectors targeting different faults in design have to generate for production test of a digital chips. Due to increasing size tests, and demand of higher test quality, algorithmic approaches are deployed for test pattern generation. This process is known as automatic test pattern generation (ATPG). Generated test pattern must have high coverage for detecting the faults in device under test (DUT) and have as short application time as possible.

ATPG is performed by the utilization of efficient algorithm and CAD tool [14]. All of these use some forms of circuit and fault models. Often, circuit model is derived from a circuit netlist. The fault models may either be any of those discussed in *2.4*.

This chapter discusses the basic flow of ATPG for stuck-at-fault and path-delay fault model using industry standard tools form Synopsys® Inc., such as Design Compiler™ (DC™), VCS™, DFTMAX™, PrimeTime™ and TetraMAX™. Again, combinational benchmark circuits (ISCAS'85) and sequential benchmark circuits (ISCAS'89) are used to experiment with the discussed flow.

3.2. ATPG FOR STUCK-AT FAULTS

Stuck-at fault model is the simplest and basic fault model for testing electronic circuits. *Fig. 3.1* shows flow chart for the process of test pattern generation for it.

The digital system is designed and its functionality is verified with appropriate testbench program. This design may be at RTL level in any HDL. Here for experimentation, the benchmark circuits used are Verilog format and CMOS 65 nm library by TSMC Ltd. has been used to synthesize the same.

The efficient and widely accepted scan methodology has been selected for purpose of testing. For this methodology, four DFT structures, viz. multiplexed flip-flop, clocked

scan, level sensitive scan design (LSSD) and auxiliary clock LSSD, are available in DFTMAX™ [14], a tool used for incorporating DFT into the design. An appropriate structure is chosen and incorporated in a design. Throughout experimentation, multiplexed flip-flop has been used as a scan style. The list of signals used and their job in test mode is described to create test protocol. The test signals are of the type master and slave clock, scan clock, input-output control, reset, constant, test mode, scan in, and scan out, test data, scan enable etc.. Further, design rules are checked and violations, if any exist, are fixed before generating .spf (STIL Procedure File) file. Besides this, numbers of scan chains in design are specified.

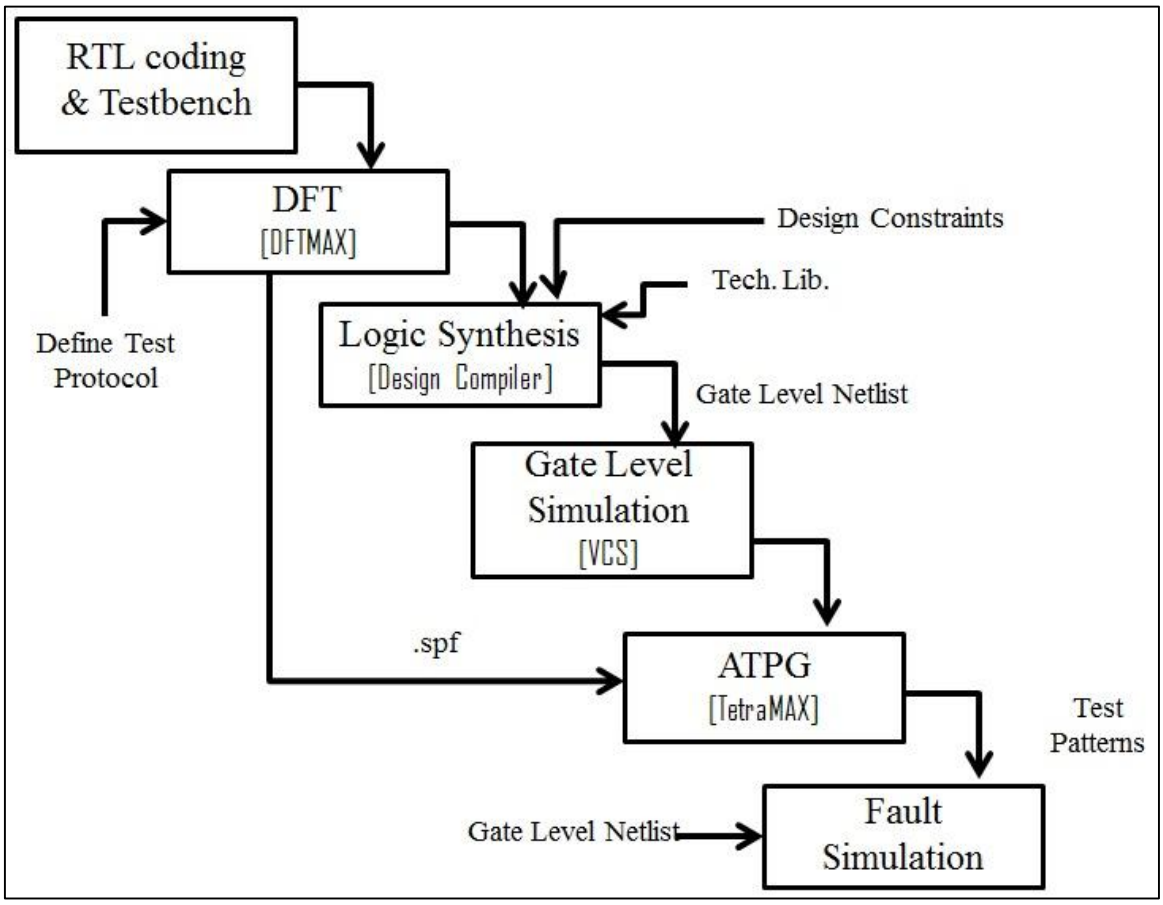

Fig. 3.1 Test generation flow for stuck-at fault with Synopsys tools

Furthermore, design is constrained with appropriate optimization constraints, according to design specification. The optimization constraints include speed, power and

area. The synthesis tool DC™, tries to optimize the design in the same order of precedence for constraints as noted. This order of precedence can be changed according to the requirements. Besides this, DC™ also tries to meet design rule constraints, implicitly defined by technology library, with higher precedence than former optimization constraints. The design rule constraints include maximum transition time, maximum fan-out, maximum and minimum capacitance, and cell degradation. The design is synthesized for given test protocol, design constraints, and technology library using DC™, to generate gate level netlist. Note that no DFT architecture is required for generating test patterns for combinational circuits. This is because the faults in combinational circuits can be sensitized from PIs and their response can be propagated to POs easily.

Functional simulation of gate level netlist generated by synthesis process is performed again with same testbench program used for RTL simulation. This is done to check whether synthesis tool correctly interpreted RTL code or not. Design constraints are checked here to meet specification. If constraints are violated, appropriate changes have to be made in design with re-synthesis, in order to meet the same. Here onwards, whenever gate level netlist is used, it is always associated with the technology library used to synthesize it.

Gate level netlist and STIL procedure file are fed to TetraMAX™, an ATPG tool, for generating test patterns targeting stuck-at faults in the design. The ATPG tool can be constrained for required test coverage, test pattern volume and test generation time. The tool tries to generate concise set of test patterns after optimizing the given constraints. Generated test patterns can be written in Verilog or binary format. Finally, fault simulation is performed for design with generated test patterns.

3.3. ATPG FOR PATH-DELAY FAULTS

Path delay faults are gaining more importance these days. Instead of altering functionality of the digital systems, these faults affects its performance and reliability. It has been seen that about 94.5% of all failing parts can be detected by path delay fault test alone. Out of which, 20 % are failing only due to path delay faults [15]. *Fig. 3.2* shows flow chart for the process of test pattern generation for it.

It can be observed from *Fig. 3.1* and *Fig. 3.2* that ATPG flow for both stuck-at faults and path delay faults is same unlike, test generation for path delay faults needs information about delay paths in the design. Thus unless otherwise specified, the common blocks in the flow chart perform same task discussed in *above* section.

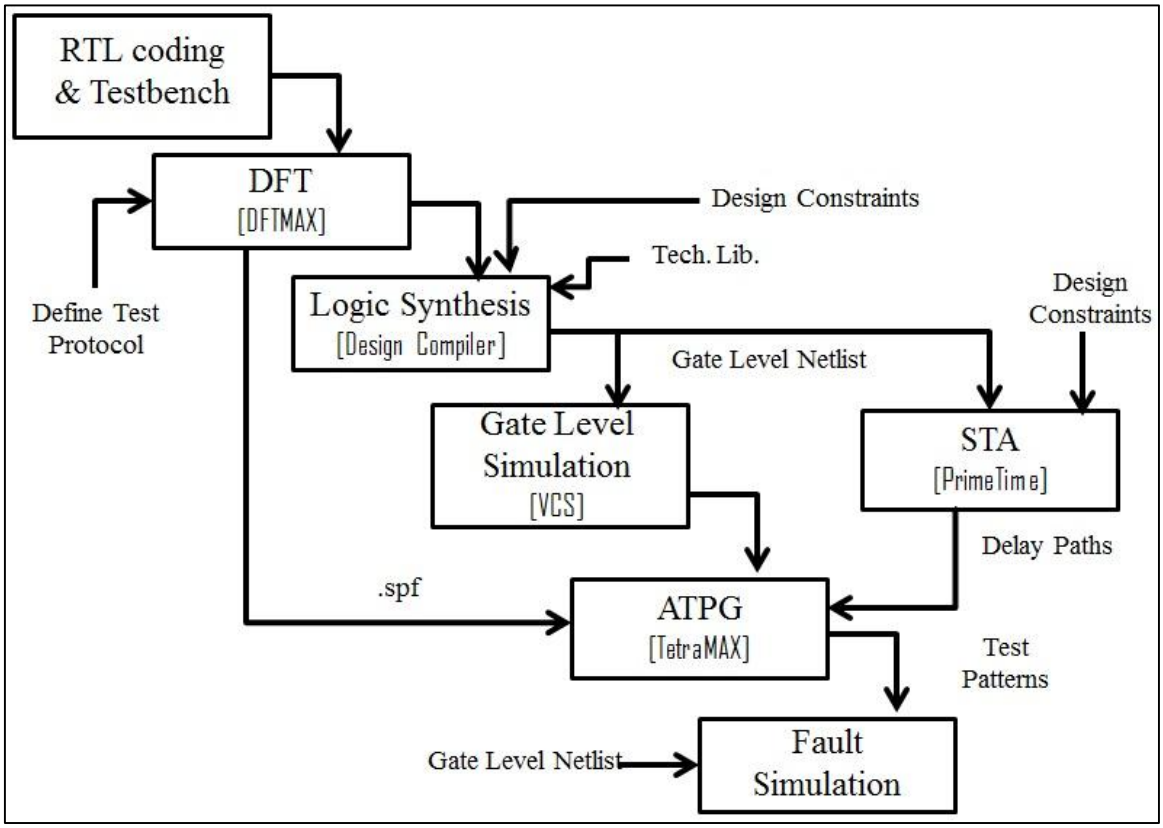

Fig. 3.2 Test generation flow for path-delay faults with Synopsys tools

TetraMAX™ needs information about delay path in design for generating test patterns for path delay faults in the design. This information include system clock, test clock, setup time, hold time, length of path, propagation time of a path, slack, clock latency, clock uncertainty etc. This can be obtained by performing static timing analysis (STA) of design. PrimeTime™-SI is the tool form Synopsys for performing static timing analysis of design. It reads design netlist and design constraints for checking timing related issues in design such as setup, hold, removal and recovery time constraints, clock gating setup and hold constraints, data-to-data timing constraints, minimum period and minimum pulse width of clock, design rules governed by technology library etc [14]. STA gives delay critical paths in design with respective slacks.

Delay paths obtained by STA of design are fed to TetraMAX™ configured for path delay fault test pattern generation. The tool generate concise set of test patterns after optimizing the given constraints and can be written in Verilog or binary format. Finally, fault simulation is performed for design with generated test patterns.

Chapter 4
OUTPUT RESPONSE ANALYSIS

4.1. INTRODUCTION

In conventional scan test methodology, test patterns are applied at PIs and the response is collected from scan-out pins in the design [5]. The response data is downloaded on ATE memory, in harmonic with the test pattern application, when device is under test. This test response data, consisting of large number of 0s and 1s, is checked bit-by-bit against expected (fault-free) response on ATE side. With ever increasing complexity and functionality of the digital systems the volume of the test input data and that of test response data that to be checked when the DUT is under test have been came out to be the principal difficulties in the scan test methodology. They demand huge amount of ATE memory to store the test response data and in turn incur large test time to check it, collectively increasing test cost.

In this chapter, we discuss various techniques that ease the task of checking and analyzing bulky test response data when under test. The techniques discussed here are equally applicable to the combinational and the sequential circuits. Before starting discussion on response analysis techniques, some of the frequently used terms are have to be clarified.

- **Compaction –** This is the method that drastically reduces the number of bits in the original output response during testing with some information loss [5]. It has very high ratio of number of bits in compacted version of bit stream, the signature, to that of original one. However, one cannot anticipate the original bit stream after looking at its compacted form. In short, compaction is a *non-invertible* function.
- **Compression –** This is the method of reducing the number of bits in the output response without loss of information, so that one can fully recover the original response from its compressed version. Thus compression of the given bit stream is

unique and is *invertible* function. But one could hardly expect large compression ratio that compared to the compaction one.

- **Aliasing –** Information loss in the compaction gives rise to the possibility of matching compacted response of faulty device to that of good device. This is called aliasing [3]. Aliasing leads to pass the failing devices through test process.

4.2. RESPONSE COMPACTION TECHNIQUES

Predominantly it has been observed that exhaustive test input patterns are applied at the PIs of DUT and their generated response is compacted in order to avoid its bit-by-bit check against the golden response [3]. Response compaction considers reduction in the number of test pins and volume of test data to be monitored under test. The ultimate data compaction is compacting the response to the one bit. However, more the bit stream is compacted, more it is vulnerable to get corrupted leading to the aliasing. This tread off has to be answered in compaction process. Some of the data compaction techniques are discussed henceforth.

4.2.1. Syndrome (ones-count) Testing

This is the simplest response compaction technique. It counts number of 1s (0s) appeared in the output response bit stream [16]. Clearly, in given *N*-bit long bit stream, the count ranges from zero to a maximum of *N*. The sole ones-count of response bit stream is less trustable for general use. Certainly, if ones-count does not match the expected value, then the generated output response is faulty. But the converse need not be true.

4.2.2. Accumulator-syndrome Testing

This is the modification of above discussed syndrome testing. Here an accumulation of the syndrome value is considered [17]. For example, if the output response bit stream initiates as:

0 1 0 1 1 0 0 1 1 0 …

then the syndrome count *S* and accumulator-syndrome count *AS* rises as:

S:	0	1	1	2	3	3	3	4	5	5…
AS:	0	1	2	4	7	10	13	17	22	27…

Here it is interesting to note that syndrome value *S* of the response bit stream is independent of the order of the appearance of logic 1 (or logic 0) in the bit stream. In contrast, accumulator-syndrome value *AS* does depend on the order. This enhances the effectiveness of the accumulator syndrome testing over former one.

4.2.3. Transition Count Testing

This is an alternative response compaction technique to preceding count based methods. In contrast to the counting logic 1 (or logic 0), so done in previous techniques, here, transitions from 0 to 1 (or from 1 to 0 or both) are counted [3]. Moreover, the transition count testing does depend on the order of appearance of data bits.

Besides these, other response compaction techniques are parity check [18], output data modification [19], signature analysis etc. Due to simple design and high efficiency in catching fault in faulty bit stream signature analysis has become an extremely efficient tool for testing purpose. The signature analysis technique is discussed in next section.

4.3. SIGNATURE ANALYSIS

This response compaction methodology is developed by Hewlett-Pickard as the technique for both testing and diagnosis in complex digital systems [20]. It utilizes an autonomous linear feedback shift register (LFSR) with one extra input at its first stage from a chosen point in DUT [3]. On application of clock, LFSR proceeds according to the logic value on external input and the feedback from its internal states. The remainder in the signature register can be used as the signature for the particular bit stream. The length of signature generated at the end of the test depends on the length of the signature register.

4.3.1. Fundamentals of Signature Register

The linear feedback shift register (LFSR) that utilized for signature analysis (hence also called as signature register) is basically the shift register configuration, which when clocked progress its stored logic values from left to right, but has mod_2 addition feedback from its selected internal states (taps) along with an extra external input to form a serial input to the first stage [3]. Since progression to the next state depend not only on the feedback taps but the external input also, the LFSR counts in pseudorandom manner. Besides, the mod_2 addition for feedback can be realized using the Ex-OR gates.

The count in signature register need not traverse through all possible values. However, chosen the appropriate feedback taps, mathematically defined by the characteristic polynomial, an *n*-stage LFSR will count in pseudorandom manner through all possible $2^n - 1$ states before repeating the sequence. This is known as maximum length sequence. It contains all possible states except one forbidden state. In the forbidden state, the count in the signature register gets stuck and it comes out of this state only on the arrival of the proper logic at the external input. The forbidden state of the LFSR depends on its design configuration like the way in which flip flops in the LFSR connected to each other (that is to say either inverting or non-inverting output connected to the input of next flip flop) and the logic value sampled at the feedback taps (i.e. sampling either at inverting or non-inverting output of the selected feedback stage) [5]. For example, when shift register is designed with non-inverting output connected to the input of next stage and the feedback is also sampled at the non-inverting output of the selected stages then the state of all logic 0s is the forbidden state of that LFSR. The LFSR count will stuck at this state unless first logic 1 is arrived at the external input pin of the signature. Throughout the discussion, without loss of generality, the configuration described in the example is assumed.

LFSR to traverse through maximum length sequence, the polynomial representing the feedback taps must be a primitive polynomial i.e. the polynomial must not factorize in two or more parts. In other words, the binary representation of that polynomial must be a prime number.

For given characteristic polynomial, two configurations are possible based on the way feedback is realized viz. external feedback and internal feedback. Consider a characteristic polynomial $P(x) = 1 + x^1 + x^4$. *Fig. 4.1 (a)* shows schematic diagram of the signature register realized with external feedback. In this, the logic value is sampled at selected taps and fed back at the first stage of LFSR after performing their Ex-OR operation along with external input. Likewise, *Fig. 4.1 (b)* shows realization of the same characteristic equation with internal feedback.

Here, two input Ex-OR gate is inserted between appropriate pair of flip flops. One of the input of Ex-OR gate is connected to the output of previous flip flop while other is the output of the last stage of LFSR. Being the same characteristic polynomial there is no

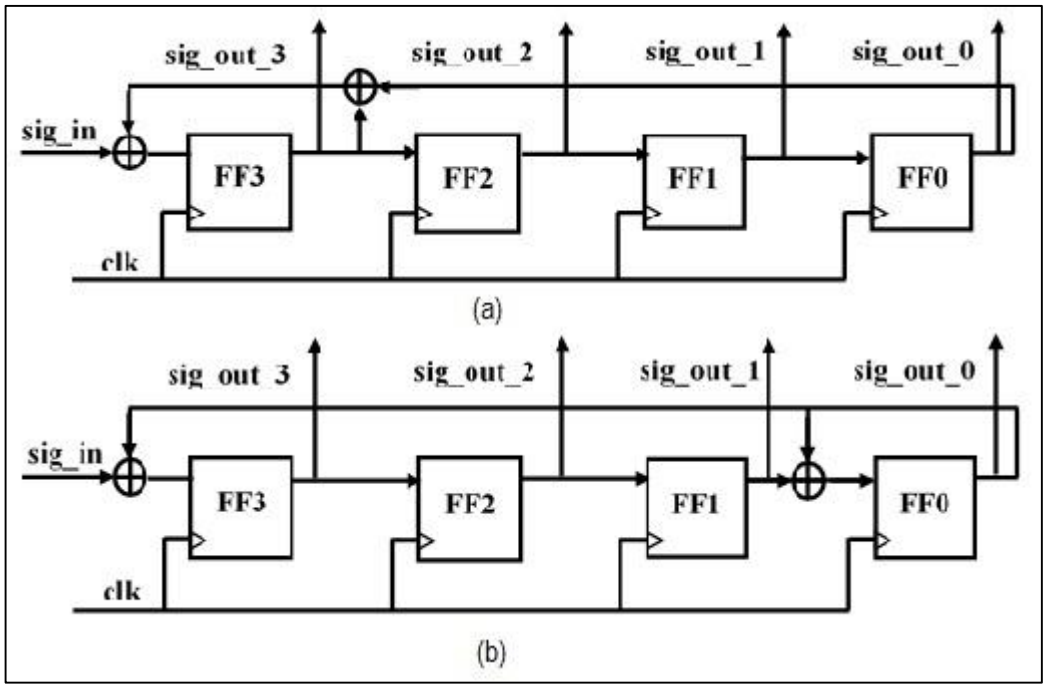

Fig. 4.1 Two alternative configurations of the LFSR with characteristic polynomial $P(x) = 1 + x^1 + x^4$ realized using (a) external feedback (b) internal feedback

difference in hardware utilization and the length of generated sequence for both of the configurations. However, the data held in the stages of LFSR after every clock pulse need not be same. It has also been observed that the maximum operating frequency for the circuit realization using internal feedback is higher than that of external one.

The signature register discussed so far has only one external input hence known as single input signature register (SISR). It may also have multiple inputs. Multiple input signature register (MISR) calculates signature for bit stream generated from number of test points or scan chains unlike SISR, which calculates signature for only one scan chain. In MISR, an Ex-OR is inserted between the every pair of flip flops. One of the inputs of Ex-OR gate is an output of previous flip flop while other is external input. Similar to SISR, feedback in MISR can also be realized either externally or internally. *Fig. 4.2* shows schematic diagram for 4-input MISR with the same characteristic polynomial $P(x) = 1 + x^1 + x^4$ realized using internal feedback. This calculate 4-bit signature for bit streams at available 4 inputs. The MISR can be transformed into SISR by making unnecessary Ex-OR gates between flip flops as transparent.

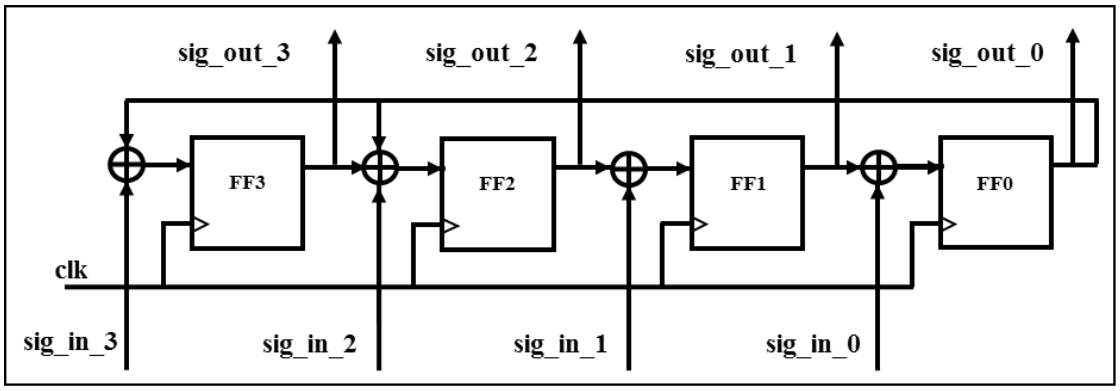

Fig. 4.2 Schematic diagram for 4-input MISR with characteristic polynomial $P(x) = 1 + x^1 + x^4$ realized using internal feedback

4.3.2. Fault Masking Property of Signature Register

Being a response compactor, signature analysis suffers information loss in the input bit stream. Increasing the compaction ratio elevates the possibility of fault masking

(aliasing). The theoretical probability of fault masking is shown here. The mathematical analysis performed here is equally valid for both SISR and MISR. Here for simplicity, SISR is considered.

Assume that q be the number of bits in the input bit stream to the p-bit long signature register. In practice $q > p$. Thus in total there are 2^q possible different input bit streams but only 2^p possible residual signatures. If all possible faults in the input bit stream are equally likely, then we have one good input bit stream and its associated one true signature, with $2^q - 1$ faulty bit streams out of which $2^{q-p} - 1$ gives same signature as that of fault-free input bit stream. Thus probability of fault masking is given as:

$$P_{fm} = \frac{2^{q-p} - 1}{2^q - 1} \times 100\%$$

Here, if 2^q and 2^{q-p} are both >> 1, then

$$P_{fm} = \frac{1}{2^p} \times 100\% \qquad (4.1)$$

It can be observed form above equation that the theoretical value of probability of fault masking only depends on the length of generated signature. Thus increasing the length of signature register minimizes this value. Note that, this theoretical value of probability of fault masking is under the assumption that all bits in the input bit stream are equally likely to be faulty, which is far away from the reality. Again, the derivation does not consider the nature of feedback polynomial. Thus for any feedback, the probability of fault masking remains same, which is certainly not valid. Hence regardless of the serious doubt about cogency of the theoretical performance of the signature register, practically it has proven an extremely competent tool for capturing the mismatches in the input data stream [3].

4.4. SUMMARY

Large volume of test patterns and response data generated at the end of the test have made use of output response analyzer (ORA) as inevitable. ORA compresses or compacts the data either in space or time or both. In this chapter, response compaction using signature analysis is taken into consideration and discussed in detail.

Aliasing is a main issue in compaction scheme and it trades-off with the compaction ratio offered. Thus an intelligent decision is required to solve this riddle.

Chapter 5
DESIGN OF SELF-TESTING SIGNATURE REGISTER

5.1. INTRODUCTION

Signature analysis has proven to be the most efficient response compaction circuit compacting gigabits of data into the small signature. Further this signature is downloaded on ATE and compared with pre-calculated golden signature to make PASS/FAIL decision for the DUT. Certainly, it has significantly reduced the demand of huge memory to store test response data otherwise. However, it has not changed the count of test pins on DUT and so the tester channel requirement on tester head. For example, consider a small design which utilizes 16 scan chains and each scan chain generating say 1000 bits of data under test mode. Thus in total 16 × 1000 = 16000 bits of response is generated at POs of DUT. Besides, 32 tester channels (16 channels for scan-in and other 16 for scan-out operation) are required on tester head. Furthermore, precise comparison of such huge data need more number of compare edges per tester cycle. Using 16-bit MISR, this this response can be compacted to a small 16-bit test signature. The compaction ratio of 1000 can be achieved here. But again, 32 tester channels are demanded (16 channels for scan-in operation and other 16 for downloading test signature).

5.2. NEED FOR ON-CHIP COMPARISON

The increasing functionality, complexity and performance of today's digital ICs are considerably increasing the number of test pins in DUT. But at the same time, the tester channels available on tester head of ATE for scan operation are falling short in number [2]. In addition to this, increasing their number increase the test cost [13] [21]. One way to solve this problem is to use the bit stream sent out of the signature register as the test signature rather the remainder. This significantly reduces the number of test pins of DUT but this solution gives very small compaction ratio.

To address the issue of response compaction along with the reduction in the number of test pins, an on-chip self-testing signature register has been proposed. It consist of a

LFSR based response compactor that utilizes the remainder as a test signature and an embedded comparator that facilitates the comparison of test signature with golden one. The comparison generates 2-bit test result on single test output pin, irrespective of the number of scan chains. Being no storage of test response is demanded on ATE memory, this considerably reduces the usually requirement of large tester memory. Also, with only single pin to be monitored for test result, the number of tester channels and compare edges on ATE side significantly reduce at the end of the test. This cuts down maintenance and usage cost of test floor increasing its life time. Beyond this, it gives scope for DFT engineers to increase number of scan chains so as to further reduce test time.

5.3. Related Work

Most of the industrial designs use on-chip LFSRs to generate small signatures during manufacturing test. In last three decades, researchers identified several issues with LFSR based signature analysis schemes that are: improving signature register design to minimize the cost of testing, handling X-states propagating into signature registers, selection of the 'best' polynomial defining the feedback, minimization of probability of fault masking and besides testing, its support for debugging and diagnosis of a chip [21] [22] [23] [24] [25]. In this work, our focus is mainly on the techniques for reduction in test time and cost during manufacturing test.

To address the problem of test cost and test time, a number of output response compaction schemes have been proposed [26] [27] [28] [29]. Barnhart *et.al* proposed a technique called on-product MISR (OPMISR) [26]. This utilizes IBM's logic BIST structure called STUMPS [30] in which MISR structure is used at the output of product scan chains. In comparison with conventional scan methodology, the OPMISR technique compacts test response and reduces the number of test cycles required for testing a chip on ATE. But at the same time, it demands a large amount of ATE memory buffer. Also, no

reduction has been observed in tester channel requirement. An improved MISR design is proposed in [31]. In this, instead of feeding the scan-out data directly to multiple input signature register, first it is aggregated and then is fed into single input shift register. This makes operation of the MISR tractable and guarantees irreducibility for all input divisor polynomials.

Incorporation of self-testing facility is observed in [32]. They designed a self-testing scan flip flop and demonstrated the test for delta-sigma modulator. The same work has been extended for other digital circuits by Katoh *et.al.* for delay measurement used for detecting SDDs in IC [33]. But replacing every flip flop with self-testing scan flip flop increases area drastically. Again, it demand large decoder circuit

Other improvements in reducing test cost and test time are based on improvements in ATE load boards. An output response analyzing circuit is facilitated on load boards, so that a low cost ATE can be used for production testing [34]. The possible techniques to reduce the cost of production testing are to utilize less ATE resources such as memory, scan channels, drive and compare edges per tester cycle and to minimize test time per chip.

In this paper, the proposed design of self-testing signature register targets reduction in the test time and the number of tester channels on ATE head.

5.4. PROPOSED ARCHITECTURE

5.4.1. Main Idea

It has been discussed and illustrated in section *5.1* that the use of signature analyzer alone does not change the demand of the ATE channels. Thus, a novel self-testing mechanism has been incorporated in signature register. It facilitates the storage of golden signature and comparison of the same with test signature. This significantly reduces the demand of scan channels on test floor. Realizing this self-testing signature register carries signature register as a response compactor, a latch chain to store golden signature and a

cascade of Ex-OR and OR gates as a comparator generating two bits of PASS/FAIL test result to be monitored.

Output response compactor uses *N*-bit signature register to generate 'sufficiently' unique signature for a given bit stream. SISR is a dedicated signature register per scan chain whereas MISR can generate a signature conjointly for number of scan chains depending on its length. The golden signature for particular set of test patterns is pre-calculated with knowledge of scan-out bit stream generated by simulator. This can be loaded into latch chain concurrently with application of test patterns. At the end of the test this golden signature is compared with the test signature bitwise to generate final PASS/FAIL test result on single pin of DUT. In case of multiple self-testing signature registers used in design, the OR operation of test result from all registers gives test status of entire chip.

Latch chain is nothing but a serial-in parallel-out (SIPO) architecture of *N*-bit shift register. With the proposed design, single ATE channel is sufficient to analyze the test result.

5.4.2. Implementation

In this section, an implementation of proposed signature register in a design is explained in detail. It includes scan based DFT insertion, design of self-testing signature register and the design of whole system.

5.4.2.1. Scan Design

Self-testing signature register is proposed to assist design test in scan methodology. On completing the RTL coding and its functional verification, scan architecture as a DFT is incorporated in the design. This is achieved by replacing all flip flops in the design with scan enabled flip flops aka scan flip flop (SFF). This technique has been earlier introduced in section *2.5*.

Fig. 5.1 shows gate level details of conventional scan enabled flip flop used as DFT. The signals *D*, *Q*, and *clk* are the input, output and clock signals respectively. The lines *SI* and *SO* are the scan-input and scan-output for assembling the scan path. The output *SO* is linked to *SI* of a neighboring scan flip-flop or a primary output. The input *SI* is linked to *SO* of a neighboring scan flip-flop or a primary input. The *TM* signal controls multiplexer for defining mode of operation. When *TM* = 0, the flip-flop is in normal mode. While, for *TM* = 1, the flip-flop is in scan mode.

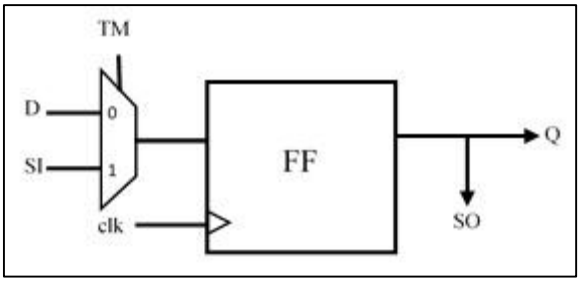

Fig. 5.1 Scan flip flop used as DFT [5]

5.4.2.2. Design of Signature Register

Several factors have to be considered during the design of signature register such as targeted probability of fault masking, selection of feedback tap, constraints around P – P – A matrix etc.

In experimentation with proposed design, 16-bit length for signature register is chosen. Using eq. 4.1, it has theoretical fault masking probability value of

$$\frac{1}{2^{16}} = 0.000015$$

This gives fault detection probability of 99.9985%. Note that this is the pessimistic theoretical value. Chosen a proper characteristic polynomial defining feedback, it has proven to be an outstanding circuit in detecting faults in scan-out bit stream.

Besides, keeping in mind an at-speed testing of today's high performance ICs, internal feedback configuration is chosen for design of signature register with knowledge that it offers higher operating frequency compared to its counterpart.

The selection of the best characteristic polynomial defining the feedback around the LFSR for particular situation has remained an open question over the years. Basically, the feedback polynomial divides the input bit stream and the remainder stays back in the LFSR stages which is used as a test signature. For the 'sufficient' uniqueness of signature, the chosen polynomial must provide a large pool of different reminders. So as to achieve this, the largest possible primitive polynomial is selected for the given length of LFSR. This allows LFSR to cycle through maximum length pseudo-random sequence rather some shorter trivial sequence.

5.4.2.3. Design of Self-Testing Signature Register

The proposed self-testing signature register is basically an on-chip test output response compaction and signature comparison circuit, for scan test methodology. *Fig. 5.2 (a)* shows schematic diagram of the same. Output response compaction block SR, uses either of *N*-bit signature register shown in *Fig. 4.1* or *Fig. 4.2*. The pre-calculated golden signature for particular set of test patterns can be loaded into latch chain LC, concurrently with application of test patterns. The gate level schematic of latch chain is described in *Fig. 5.2 (b)*. At the end of test, an Ex-OR operation is performed between test and golden signatures which compares both signature. Further, bitwise OR operation is conducted on the result of Ex-OR operation. This identifies the mismatch in signatures. The result of OR operation generates final PASS/FAIL test result (TR) for that particular test.

As shown in *Fig. 5.2 (a)*, SO_i are input to SR that is connected to scan-out signals in the design and *LSI* is latch chain input used for loading reference signature in latch chain. *s_clk* and *latch_clk* are clock signals for driving signature register and latch chain. The

clock used for driving signature register is same as scan clock for DUT. However, any clock signal can be used for loading latch chain. In case of variable-clock scan test [35], a slow clock may be applied for latching in golden signature. This gives enough confidence for correctness of stored data by giving sufficient time for logic to settle at the input of each memory element in scan chain. To control the clocking of signature register and latch, separate signature enable *SEn*, and latch enable *LEn*, signals are provided. *latch_clk* is disabled after loading last bit (MSB) of golden signature into latch chain, while *s_clk* is disabled once all test patterns are applied to DUT. This avoids unnecessary transition of count in signature register after completion of scan-out operation and stabilizes the signature for comparison. Both input patterns and golden response can be fed concurrently to DUT (*SEn = LEn = 1*) and once data is stored into latch, *LEn* signal is disabled (*LEn = 0*). This does not intervene the application of test patterns at input side. For *p* scan chains, the proposed architecture demands *(p+5)* ATE channels (*p* channels for scan-in and one each for latch enable, signature enable, latch input, latch chain clock and test result).

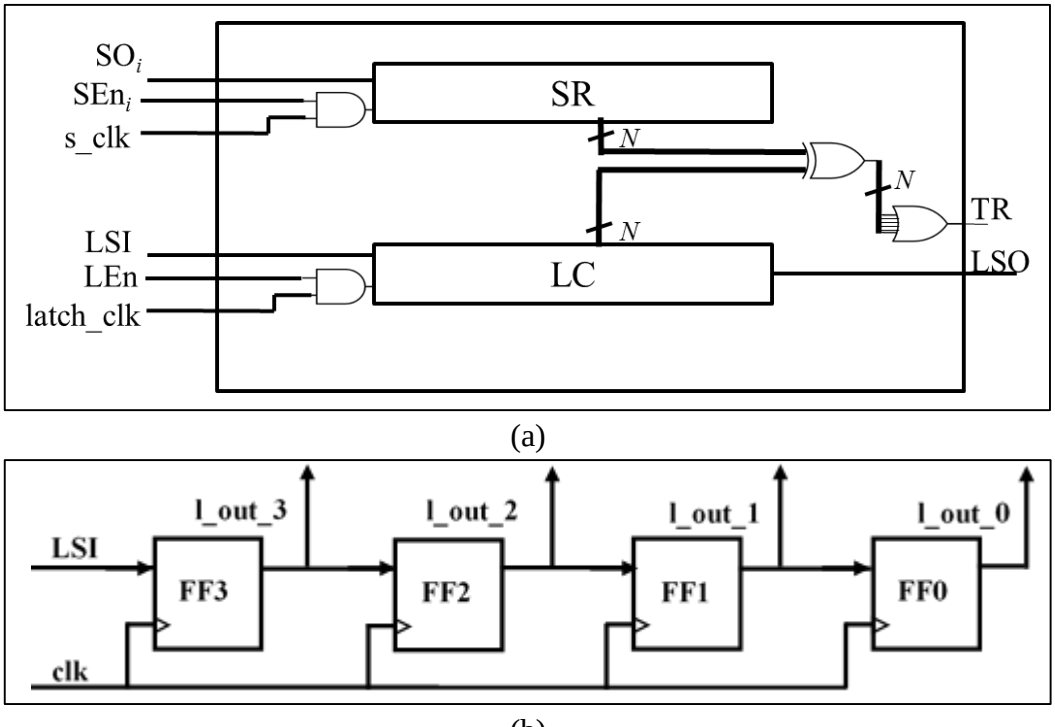

Fig. 5.2 Schematic of (a) proposed self-testing signature-register and (b) Latch chain

In case of multiple self-testing signature-registers in design, latch chains could be further chained by connecting latch output, *LSO*, to *LSI* of next latch chain. By doing this, single *LSI* pin can be used to load all latch chains. Moreover, OR operation of all *TR* pins can be performed to obtain test result of entire chip. For SISR based system, primary input scan-in can also be used to latch data in latch chain, further reducing a test pin. This has been shown in *Fig. 5.3*.

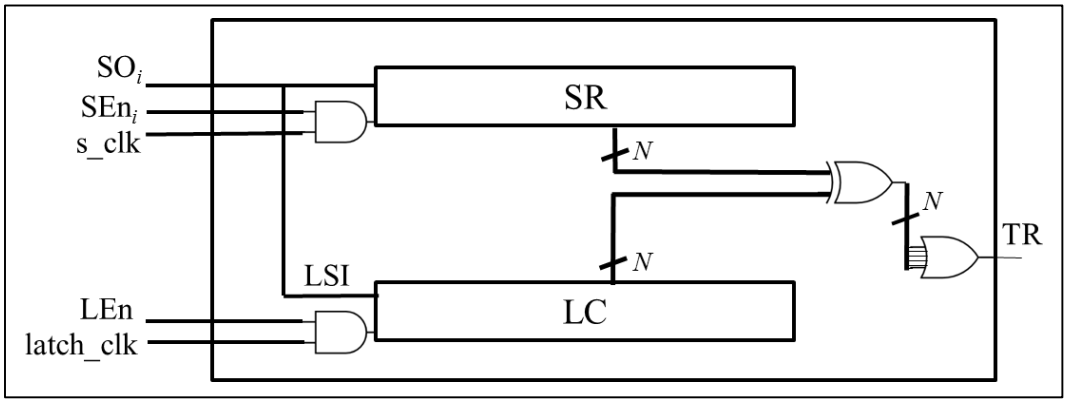

Fig. 5.3 Schematic of self-testing SISR using scan chain to load latch chain

5.4.2.4. Whole System

Fig. 5.4 shows the entire system with proposed architecture. This consists of a 'low cost' tester and a chip designed with embedded self-testing signature register (STSR). An adjective 'low cost' for tester refers to a tester with minimum utilization of its resources such as memory, scan channels and that adds to less tariff. In the diagram shown, it has been assumed that all flip flops on a chip are grouped to form *p*-scan chains, each having same number of flip flops. Every scan chain has its own scan-in line SI_i. The signals such as test mode, reset, and clock for all flip-flops are combined to single signal as *TM*, *rst*, *clk* respectively. Scan chains are further grouped in m groups such that all scan-out signals in one group are fed to one self-testing MISR ($STSR_i$). Thus, as a whole, *m* STSR blocks are required in design. Latch clock and enable signal for signature register and latch are again combined to single *latch_clk*, *SEn* and *LEn* signal respectively. All latch chains are further

chained with *LSI* as single latch input. *TR* is the test result of entire chip given back to ATE for test decision.

In designs with small flip-flop count and no clock domain crossing, a single scan chain can be used for testing. Such designs can utilize self-testing SISR.

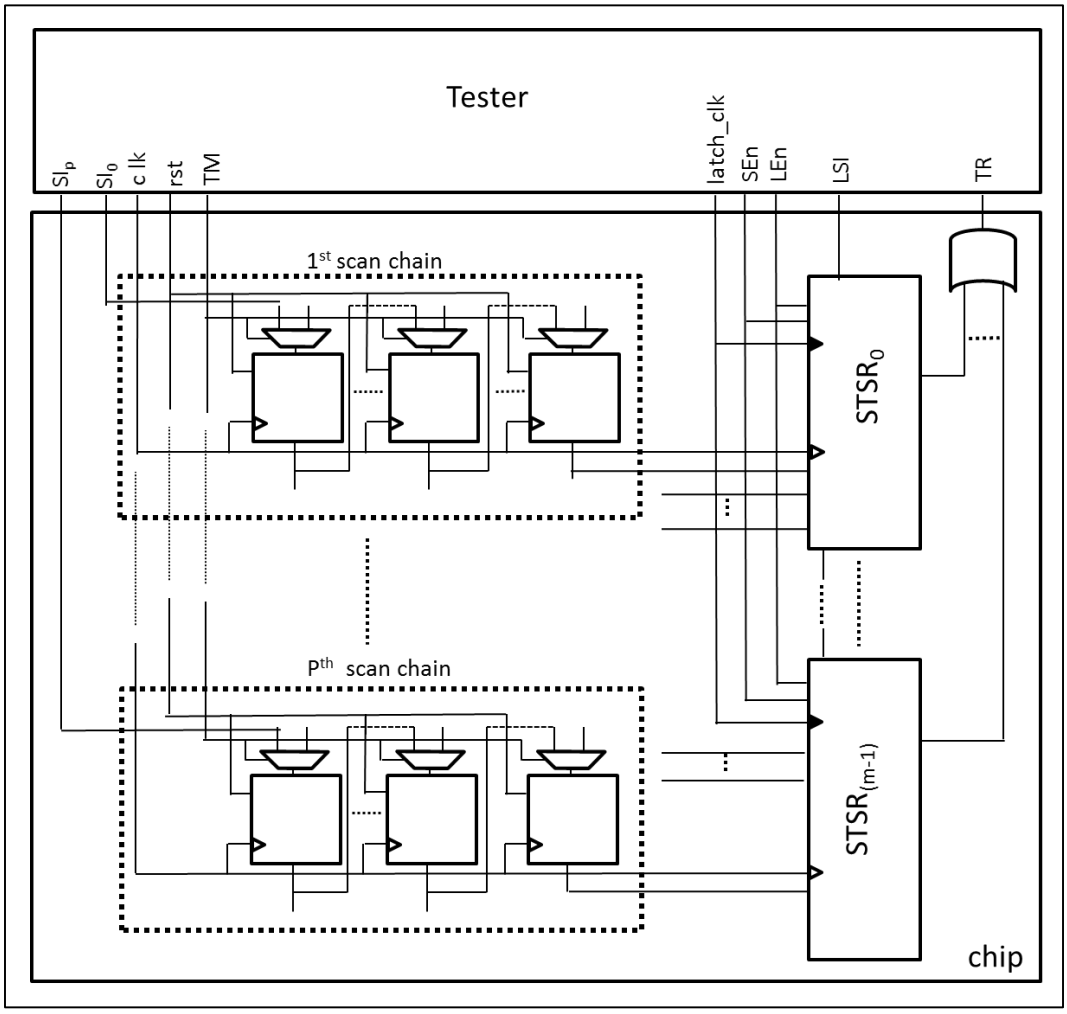

Fig. 5.4 Whole system

5.5. TEST FOR PROPOSED ARCHITECTURE

The signature analyzer has to be tested for manufacturing defects in order to rely on its decisiveness for DUT. Due to randomness in generated signature, functional test is enough to validate operation of the signature register. Thus, this can be easily tested without generating any special patterns or addition of any DFT structures.

Scan shift test used for testing scan register can also be used to test proposed signature register. As discussed in [5], a toggle sequence, for example, 0 0 1 1 0 0 1 1 …, used for testing the scan register can be allowed to enter into STSR to generate signature for the same. This sequence produce all four transitions, 0 – 0, 0 – 1, 1 – 0 and 1 – 1, at the input of each flip-flop and shifts the data to the scan-out end of the chain. Generated signature is compared with desired one to validate the operation of STSR, as stated earlier.

5.6. IMPACT ON DESIGN AND TEST FLOW

The insertion of self-testing signature register for embedded testing environment has little impact on existing design and test flow. The proposed testing environment does not intercede the process of RTL design, scan chain insertion and ATPG. After inserting DFT architecture, we have clear idea about scan chain count. With knowledge of this, proposed testing logic is designed for DUT and this circuit is appended at the output of scan chains.

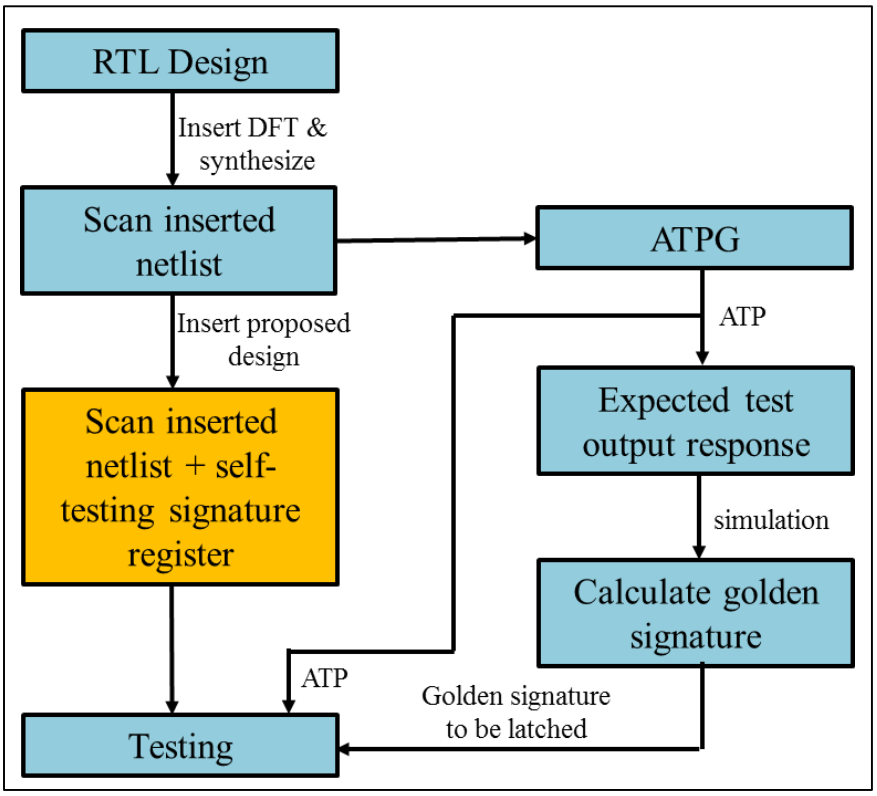

Fig. 5.5 Impact on Design and Test flow

By the time, automatic test patterns (ATP) for scan inserted design are generated with any ATPG tool and scan netlist is simulated to obtain its test output responses. Further, golden signature is calculated for test response which is used for testing fabricated ICs with proposed test architecture incorporated into it. *Fig. 5.5* shows the overall flow. This clearly shows that, ATPG for design does not depend on design of proposed signature or vice versa.

5.7. EXPERIMENTAL RESULTS

In this section, the experimental results and their analysis had been presented. The proposed embedded test architecture is compared with conventional scan methods and conventional response compactor [21] for various parameters. An experiment has been performed in two phases. First phase is simulation based that includes test pattern generation, test response collection, golden signature calculation and simulation based validation of test result generated by proposed architecture in presence of injected faults. While second phase includes FPGA implementation of design with proposed architecture, application of generated test patterns to it and its validation again in presence of injected faults.

Relatively larger ISCAS'89 sequential benchmark circuits are selected for the purpose of experimentation. The scan clock frequency and operating frequency of design are considered to be same in order to validate and to demonstrate the proposed design in at-speed test application. In design of signature register, the length of LFSR is decided to be 16-bit and the biggest possible primitive polynomial is used for defining feedback taps along the LFSR. Each design under consideration is tightly constrained for timing with sufficient on-path slack and minimum area and power. In the first phase of the experiment, the designs are synthesized with TSMC 65 nm technology library using Synopsys Design Compiler™ and TSMC8K_Lowk_Conservative wireload model has been considered for interconnects. During DFT insertion, multiplexed flip flop is used for establishing scan test

architecture in design. Also, scan pins in test mode are multiplexed with pins in normal mode with addition of extra test mode pin to switch mode of operation of DUT. In this experiment, multiple scan chains are used for first five large benchmark circuits, while single scan chain is used for others. To deduce information about delay paths and to check timing violations of design, static timing analysis (STA) has been performed using PrimeTime™. Test patterns for testing stuck-at faults and path delay faults are generated using TetraMAX™, an ATPG tool by Synopsys.

In second phase of the experiment, DFT structure inserted benchmark circuits and the proposed architecture appended to that are synthesized in Xilinx environment and downloaded to Virtex 2 Pro FPGA (Board 1). Also, FSM for applying TetraMAX generated test is downloaded to another FPGA (Board 2). The board 2 can be assumed to be a realization of ATE on FPGA for application of test patterns to DUT on board 1. The final single bit test result is validated in fault-free designs and in designs with injected faults.

For comparison with conventional designs, overhead of proposed design in terms of area and FPGA slice count are defined as follows:

$$O_A = (A_{PROP} / A_{SCAN} - 1) \times 100\% \quad (5.1)$$

$$O'_A = (A_{PROP} / A_{CONV} - 1) \times 100\% \quad (5.2)$$

$$O_S = (S_{PROP} / S_{SCAN} - 1) \times 100\% \quad (5.3)$$

$$O'_S = (S_{PROP} / S_{CONV} - 1) \times 100\% \quad (5.4)$$

Here, *O*, *A*, and *S* denote Overhead, Area and Slice count respectively while suffixes *PROP*, *SCAN, and CONV* denote circuits implemented with proposed test architecture, that with only scan chain inserted and that which uses conventional signature register for response compaction [26], respectively. Also, the primed character signifies overhead of proposed

architecture on standard scan design and the unprimed ones signify overhead on conventional signature register. Besides, the reduction ratio of test pins (R_C), and that of test time is defined as:

$$R_C = (1 - C_{PROP} / C_{CONV}) \times 100\% \quad (5.5)$$

$$R_T = (1 - T_{PROP} / T_{CONV}) \times 100\% \quad (5.6)$$

Here, C is total number of tester channels, T is test time and other suffixes have the same meanings as discussed above. Hereafter, the detailed analysis for results of ATE memory, test time, tester channel and area requirement have been performed.

5.7.1. Memory Requirement

This refers to ATE memory required for storing response collected from the chip during the progress of test. *Table 5-I* shows number of patterns generated and fault-free output data volume collected after simulation of listed benchmark circuits. The columns N_{SAF} and N_{PDF} are the number of test patterns generated targeting stuck-at faults (SAF) and path-delay faults (PDF) in design respectively. The columns V_{SAF} and V_{PDF} are number of output response bits collected, whereas, the column V_{SCAN} is the total response collected in SAF and PDF test.

It can be observed from table that, using conventional scan test methodology, on an average 8100 test responses have to be downloaded from chip and same number of golden responses has to be pre-stored on ATE memory for comparison, in stuck-at fault test alone. This number turns out to be 15850 for designs with multiple scan chains (an average of V_{SCAN} for first five benchmark circuits in table II). Similarly, average response count for path delay test is around 4000. In line of these results, test of today's VLSI designs demand enormously high ATE memory. This puts an extra constraint on ATPG tool to generate less number of test patterns. Besides, in a conventional signature register, the number of

signature bits to be downloaded on ATE will be equal to double the bit length of signature register. In contrast, the proposed architecture requires only two bits for comparison on ATE irrespective of bit-length of MISR and number of scan chains in design. This significantly reduces requirement of ATE memory buffer.

TABLE 5-I
DATA VOLUME

Benchmark Circuits	N_{SAF}	N_{PDF}	V_{SAF}	V_{PDF}	V_{SCAN}	V_{CONV}	V_{PROP}
S35932	49	15	1656	662	2318	32	2
S38584	806	356	25082	17765	42847	24	2
S38417	1054	270	35177	13598	48775	32	2
S15850	458	135	10110	4182	14292	32	2
S13207	409	125	7284	3618	10902	20	2
S9234	178	15	393	44	437	32	2
S5378	274	35	476	92	568	32	2
S1488	145	24	275	52	327	32	2
S1423	101	10	225	26	251	32	2
S1238	212	11	335	20	355	32	2
Average	-	-	8101.3	4005.9	12107.2	30	2

5.7.2. Test Time Reduction

Table 5-II shows test time reduction using proposed architecture. The column N_{FF} represents number of flip flops in design. The time reduction ratio in last column is calculated using eq. 5.6. As earlier stated, less number of tester channels are demanded using proposed architecture. Hence the number of scan chains can further be increased to reduce test time of a chip. In this experiment, 50% more number of scan chains are inserted using proposed architecture, compared to conventional one. With this, on an average, 33% reduction in test time is observed. This is due to generation of less number of patterns generated with increased number of scan chains. This can be better visualized in bar diagram as shown in *Fig. 5.6*. In nomenclature of bars, first character 'T' designates time, next three characters designates the type of fault (SAF or PDF) and last character designates the signature register used (C: Conventional and P: Proposed).

TABLE 5-II
TEST TIME ANALYSIS

Benchmark Circuits	Using Conventional signature register			Using Proposed Architecture			Reduction Ratio	
	N_{SC}	T_{SAF} (ns)	T_{PDF} (ns)	N_{SC}	T_{SAF} (ns)	T_{PDF} (ns)	RT_{SAF} (%)	RT_{PDF} (%)
S35932	16	24800	7100	24	15400	4600	37.90	35.21
S38584	12	388800	143500	28	260200	97300	33.08	32.19
S38417	16	522300	109500	24	357600	75600	31.53	30.95
S15850	10	228900	56300	15	141900	36400	38.00	35.35
S13207	10	188600	52900	15	135800	36300	28.00	31.38
Average	-	270680	73860	-	182180	50040	32.70	32.25

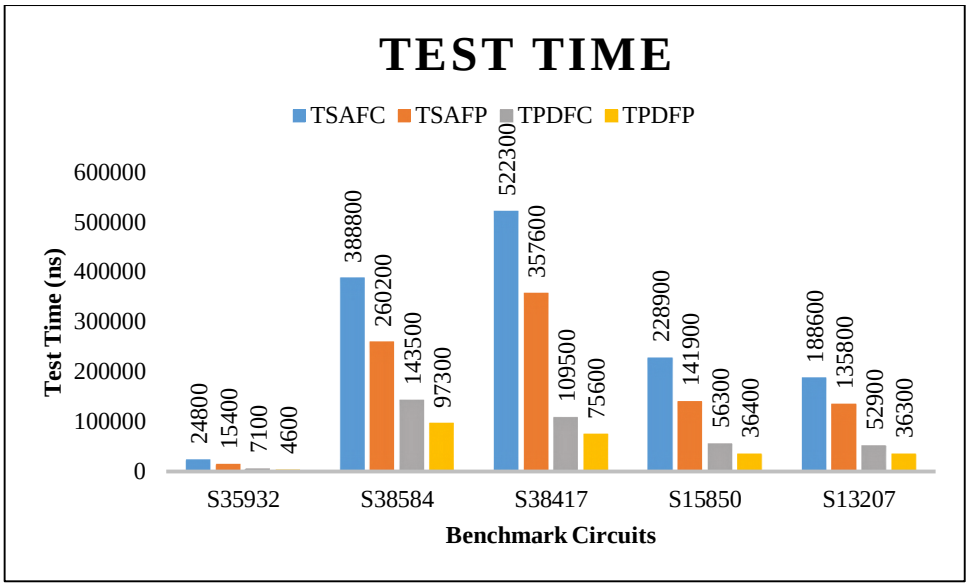

Fig. 5.6 Comparison of test time with conventional and proposed signature register

5.7.3. ATE Scan Channel Reduction

Design with self-testing MISR shows significant reduction in demand of tester channels compared with conventional scan test with multiple scan chains. *Table 5-III* shows tester scan channel requirement for both conventional and proposed test methodology. The value in last column is calculated using eq. 5.5. It can be seen that, for *P* scan chains in a design, test using conventional scan methodology and conventional signature register demands minimum *(2P + 2)* ATE scan channels, whereas, this number is *(P + 4)* for proposed architecture. Thus the reduction ratio turns out to be

$$R_C = \frac{P-2}{2(P+1)} \times 100\ \% \qquad (5.\ 7)$$

As the number of scan channels increases, which is necessary to reduce test application time, R_C approaches a value of 50 %. Same analysis is shown graphically in *Fig. 5.7*. The reduction in tester channels gives a broader scope for increasing scan chains during the process of DFT insertion.

TABLE 5-III
ATE SCAN CHANNEL COUNT

# Scan Chains	# Scan Channels		R_C (%)
	CONV	PROP	
10	22	13	40.90
12	26	15	42.30
16	34	19	44.11

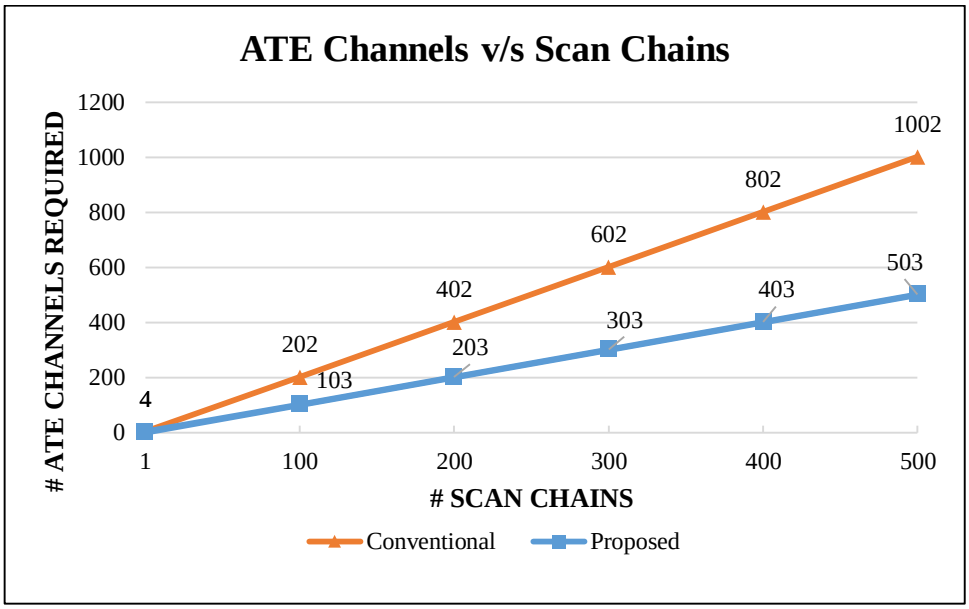

Fig. 5.7 Comparison for ATE channels requirement v/s scan chain count

5.7.4. Area Overhead

Table 5-IV shows the area comparison between conventional and proposed signature register for both ASIC and FPGA implementation. The FPGA used is Xilinx Virtex II Pro. The column N_{FF} is number of flip flops in design. It can be seen that, proposed embedded test architecture shows very small area overhead to both ASIC and FPGA implementation. On an average, proposed architecture occupy 3.25% more area than conventional signature register. Also, FPGA utilization has been increased by 2.5%. *Fig. 5.8* and *Fig. 5.9* shows the tabulated data in *Table 5-IV* in graphical form.

TABLE 5-IV
AREA AND FPGA UTILIZATION ANALYSIS

Benchmark Circuits	N_{FF}	N_{SC}	**Area for 65 nm TSMC library**					**Slice count for Xilinx Virtex II Pro FPGA**				
			A_{SCAN}	A_{CONV}	A_{PROP}	O_A (%)	O'_A (%)	S_{SCAN}	S_{CONV}	S_{PROP}	O_S (%)	O'_S (%)
S35932	1728	16	24526.80	24707.16	24979.21	1.84	1.10	3028	3037	3061	1.09	0.79
S38584	1426	12	21245.40	21425.76	21641.50	1.86	1.00	3222	3231	3257	1.09	0.80
S38417	1636	16	22700.52	22880.88	23152.92	1.99	1.19	3109	3118	3161	1.67	1.38
S15850	534	10	7808.76	7989.12	8176.40	4.70	2.34	1088	1097	1125	3.40	2.55
S13207	638	10	8108.64	8289.00	8476.29	4.53	2.26	1070	1080	1098	2.61	1.67
S9234	211	1	2209.68	2210.68	2584.44	16.95	16.90	328	338	353	7.62	4.44
S5378	179	1	2805.48	2806.48	3180.24	13.35	13.31	404	414	429	6.18	3.62
S1488	6	1	697.32	698.32	1072.08	53.74	53.52	141	150	169	19.85	12.67
S1423	74	1	1249.92	1250.92	1624.68	29.98	29.88	240	250	266	10.83	6.40
S1238	18	1	714.96	715.96	1089.72	52.41	52.20	134	143	163	21.64	13.98
Average	-	-	9206.75	9297.43	9597.75	4.25	3.25	1276	1285	1308	2.50	1.79

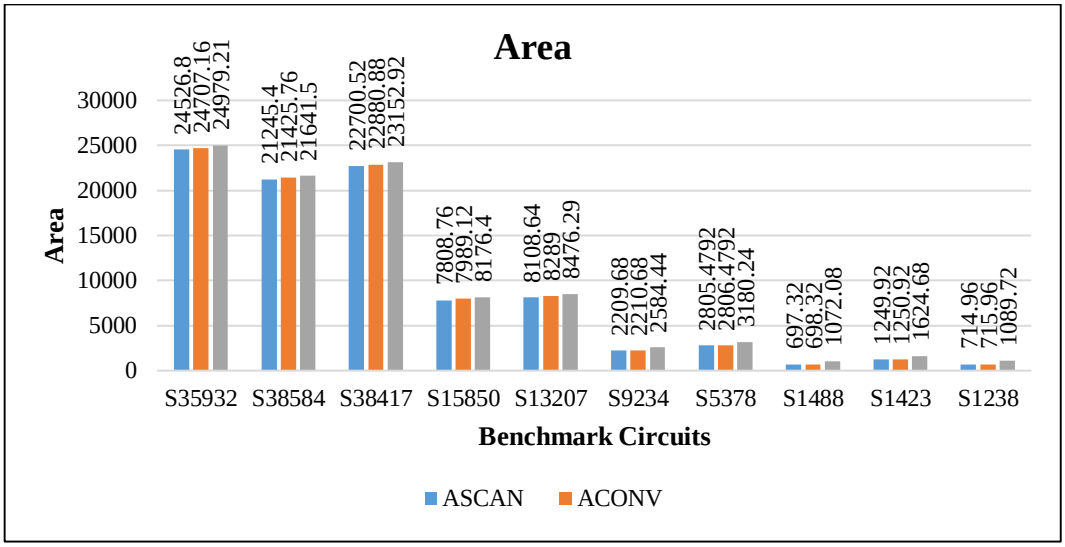

Fig. 5.8 Area overhead comparison for benchmark circuits

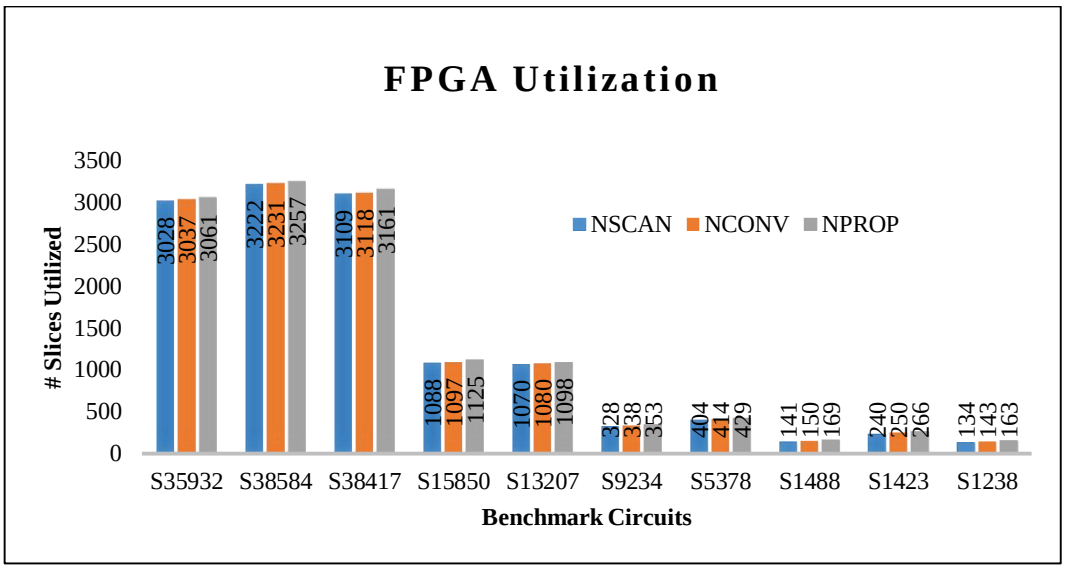

Fig. 5.9 FPGA utilization overhead comparison for benchmark circuits

5.8. SUMMARY

It has been discussed in earlier sections of this chapter that conventional response compactors have succeed in addressing the issue of analyzing large data volume. But the sole ATE resource that has been optimized with this techniques is memory, leaving aside the other resources like tester channel requirement and test time. This chapter has proposed, described and exemplified a novel self-testing signature register that addresses the latter issues alongside the former one. The proposed architecture has close equivalence to the ideal test system requirements.

Chapter 6
APPLICATIONS OF PROPOSED SIGNATURE REGISTER

6.1. MANUFACTURING TEST

6.2. TESTING OF CRYPTOGRAPHIC DESIGNS

6.3. BIST

6.1. MANUFACTURING TEST

Increasing complexity and density of ICs and emergence of new defects in fabrication process need rigorous testing with large number of test patterns to be applied at the inputs of DUT. This accumulates an extra test time and hence test cost to IC [2] [21]. The use of proposed test architecture allows DFT engineer to increase number of scan chains in the design. Knowing that the test time linearly varies with the number of scan chains, it assist to reduce the overall test time and hence test cost of the chip.

Moreover, only two bits of test result is generated with the use of proposed test architecture reduces. This does not need a precise check compared to that with the bit-by-bit checking of the scan-out response data in conventional scan design. This reduce the demand of number of drive and compare edges per test cycle. This in turn reduce the maintenance and improve throughput of test floor.

6.2. TESTING CRYPTOGRAPHIC DESIGNS

In our daily lives, many time we perform electronic transfer of confidential data. To assure the integrity and secrecy of the data variety of cryptographic algorithms are used. These can either be written in software or implemented on a high performance standalone hardware. Testing of such designs using scan methodology creates potential threat for leaking confidential information [36] [37]. Looking at the scan-out sequence of the test patterns, one can retrieve the secret encryption key. Also, one can observe and predict its internal states.

The use of proposed self-testing signature register can be a solution to this problem. Given a test patterns and a golden signature to the test centers, form which one can hardly predict anything, small test result will be generated on single pin of IC. It gives no access of scan-out bit stream to the third party (e.g., test centers) that builds up enough trust on integrity of the design.

6.3. BUILT-IN SELF-TEST

Built-in self-test (BIST) is characterized by on-chip pattern generation, application and comparison of design. Since BIST is an on-chip architecture, it is always constrained by area and memory availability to store test patterns and their responses both under test and desired one [3] [5]. Compaction techniques are highly encouraged in such applications and are being used over the years.

The use of proposed signature register further speeds up the BIST process in two ways. First by reducing the number of comparisons to be performed and other by further reducing memory requirement to store golden response. Besides, the same LFSR, used in the signature analysis, can also be used in pattern generation by disconnecting the first stage from external input. Note that LFSR on choosing proper feedback taps generates exhaustive set of patterns and same is possible to test without requirement of additional on-chip memory, as the proposed signature register facilitates an on-chip comparison.

Chapter 7
CONCLUSION

7.1. OUTCOME OF THE WORK

7.2. SCOPE FOR FUTURE WORK

7.1. OUTCOME OF THE WORK

In this thesis a novel on-chip self-testing signature register is proposed. This not only utilize less ATE resources for implementing widely used scan test methodology in large design but also reduce their test time. As a result of this, cost of performing manufacturing test on every fabricated chip can be significantly reduced.

The use response compactor considerably minimize the requirement of huge ATE memory demanded otherwise. Incorporation of on-chip signature comparator generates test result on single test pin on DUT. This drastically reduces the test pins on DUT and thus uses less number of scan channels on test floor. Irrespective of number of scan chains in design and number of test response bits generated for input patterns, the proposed signature register generates only two bits of binary (PASS/FAIL) test result on single test output pin. This removes need of checking enormous amount of test output data on ATE side that minimizes test time. Besides, knowing that less number of scan channels will be required on tester head, number of scan chains in design could be increased during the phase of DFT insertion to further reduce test time. It has been observed that for 50% increase in the number of scan chains, test time reduces approx. by 33%. With all these competences, the proposed architecture has little area overhead of 4.25% to the scan design and 3.25% to the conventional signature registers.

The design of signature register can also be reused in pattern generation, test of ICs for cryptographic application and in designs for IP authentication and IP anti-counterfeiting.

7.2. SCOPE FOR FUTURE WORK

Future work with this design is possible in the directions listed below:

1. Validation of the proposed signature register for the test of cryptographic designs.
2. X-tolerant design of signature register and incorporating the self-testing mechanism in the same.
3. Using the proposed signature register for in-situ test like wafer probing.
4. Utilization of the proposed signature in design for register anti-counterfeiting.

BIBLIOGRAPHY

[1] G. Moore, "Cramming More Components onto Intergrated Circuits," *Electronics*, vol. 38, April 1965.

[2] R. Barth, "International Technology Roadmap for Semiconductors," December 2012. [Online]. Available: http://www.itrs.net/Links/2012ITRS/Home2012.htm.

[3] S. L. Hurst, VLSI Testing: Digital and Mixed alalogue/digital technique, London: The Institution of Electrical Engineers, London, 1998.

[4] Z. Navabi, Digital System Test and Testable Design: Using HDL Models and Architectures, New York: Springer, 2011.

[5] M. L. Bushnell and V. D. Agarwal, Essentials of Electronics Testing for Digital, Memory and Mixed-Signal VLSI Circuits, New York: Springer, 2000.

[6] N. K. Jha and S. Gupta, Testing of Digital Systems, Cambridge: Cambridge University Press, 2003.

[7] R. Rajsuman and Y. K. Malaiya, Bridging Faults and IDDQ Testing, Los Alamitos: IEEE Computer Society Press Technology Series, 1992.

[8] H.-J. Wunderlich, Models in Hardware Testing, Dordrecht: Springer, 2010.

[9] M. Tehranipoor, K. Peng and K. Chakrabarty, Test and Diagnosis for Small Delay Defects, New York: Springer., 2011.

[10] G. L. Smith, "Model for Path Delay Faults Based on Paths," in *Proc. International Test Conference*, 1985.

[11] Y. L. Sung-Mo Kang, CMOS Digital Integrated Circuits, Boston: Tata McGraw Hill Education, 2003.

[12] N. H. Weste and D. Horris, CMOS VLSI Design: A Circuit and System Perspective, Pearson, 2011.

[13] "Reference Mannual for Verigy 93000," [Online]. Available: www.advantest.com.

[14] "Synopsys On-Line Documentation (SOLD)," [Online]. Available: www.synopsys.com.

[15] R. Mattiuzzo, D. Appello and C. Allsup, "Small Delay Defect Testing," 2009. [Online]. Available: http://www.tmworld.com/ article/CA6660051.html, , 2009..

[16] J. Savir, "Syndrome-Testable Design of Combinational Circuits," *IEEE Transactions*, Vols. C-29, pp. 442-451, 1980.

[17] M. Saxena and J. Robinson, "Accumulator Compaction Testing," *IEEE Transactions,* Vols. 317-321, pp. 317-321, 1986.

[18] W. C. Carter, "The Ubiquitous Parity Bit," in *Proc. FTCS-12*, 1982.

[19] J. P. Hayes, "Check Sum Test Methods," in *Proceedings FTCS-6*, 1976.

[20] R. A. Frohwerk, "Signature analysis: A new digital field service method," *Hewlett-Packard Journal*, pp. 2-8, May 1977.

[21] E. J. McCluskey, "Test Compression Roundtable," *IEEE Design and Test of Computers,* vol. 20, no. 2, pp. 76-87, April 2003.

[22] A. Ivanov and V. K. Agarwal, "An Analysis of Probabilistic Behavior of Linear Feedback Shift Registers," *IEEE Transactions on Computer-aided Design,* vol. 8, no. 10, October 1989.

[23] S. Krishnaswamy, "Signature-Based SER Analysis and Design of Logic Circuits," *IEEE Transaction on CAD of Integrated Circuits and Systems,* vol. 18, no. 1, January 2009.

[24] D. Bhavsar, "Observability Register Architecture for Efficient Production Test and Debug of VLSI Circuits," in *14th International Conference on VLSI Design*, 2001.

[25] S.-P. Feng, "On Maximum Value of Aliasing Probabilities for Single Input Signature Registers," *IEEE Transactions on Computers,* vol. 44, no. 11, November 1995.

[26] C. Barnhart, "Extending OPMISR beyond 10X Scan Test Efficiency," *IEEE Design Test of Computers,* vol. 19, no. 5, p. 65–73, October 2002.

[27] F. Hsu, K. Butler and J. Patel, "A Case Study on the Implementation of the Illinois Scan Architecture," in *Proceedings of International Test Conference*, 2001.

[28] B. Koenemann, "A smart BIST Variant with Guaranteed Encoding," in *Proceedings ATS*, 2001.

[29] J. Rajski, "Embedded Deterministic Test for Low Cost Manufacturing Test," in *Proceedings International Test Conference,* 2002.

[30] R. W. Basset, B. J. Burkus, S. L. Dingle, M. R. Faucher, P. S. Gillis, J. H. Panner, J. G. Patrovick and D. L. Wheater, "Low-Cost Testing of High-Density Logic Components," in *Proceedings International Test Conference*, 1987.

[31] F. Elguibaly and M. W. El-Kharashi, "Multiple-Input Signature Register: An Improved Design," in *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 1997.

[32] K. Noguchi, K. Nose, T. Ono and M. Mizuno, "A Small-Delay Defect Detection Technique for Dependable LSIs," in *Symposium on VLSI Circuits*, 2008.

[33] K. Katoh, K. Namba and H. Ito, "An On-Chip Delay Measurement Technique Using Signature Registers for Small-Delay Defect Detection," *IEEE Transations on VLSI Systems*, vol. 20, no. 5, pp. 804-817, May 2012.

[34] J. Howard, "Fault Diagnosis Aware ATE Assisted Test Response Compaction," in *Asia South Pacific - Design Automation Conference*, 2011.

[35] T. J. Chakraborty, V. D. Agrawal and M. L. Bushnell, "On Variable-Clock Methods for Path Delay Testing of Sequential Circuits," *IEEE Trans.Computer-Aided Design*, vol. 16, pp. 1237-1249, 1997.

[36] B. Yang, K. Wu and R. Karri, "Secure scan: A Design-for-Test Architecture for Crypto Chips," *EEE Transaction on Computer-Aided Design for Integrated Circuits Systems*, vol. 25, no. 10, pp. 2287-2293, October 2006.

[37] J. D. Rolt, G. D. Natale, M.-L. Flottes and B. Rouzeyre, "Thwarting Scan-Based Attacks on Secure-ICs With On-Chip Comparison," *IEEE Transactions on VLSI Systems*, vol. 22, no. 4, pp. 947-951, April 2014.