

Test Time Optimization in Scan Circuits

by

Priyadharshini Shanmugasundaram

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

Auburn, Alabama
December 13, 2010

Keywords: Activity per unit time, Test power, Test application time

Copyright 2010 by Priyadharshini Shanmugasundaram

Approved by

Vishwani D. Agrawal, Chair, James J. Danaher Professor of Electrical and Computer
Engineering

Adit D. Singh, James B. Davis Professor of Electrical and Computer Engineering

Charles E. Stroud, Professor of Electrical and Computer Engineering

Abstract

As circuit sizes increase with scale down in technology, the time required to test the circuits also increases. Expensive automatic test equipment (ATE) is used to test these circuits and the cost of testing becomes a significant fraction of the total cost of the chip. Testing cost of a chip is directly related to the time its testing takes. However, test time cannot be reduced by simply applying the tests at a faster speed because if the test clock frequency is increased, the power consumed during test increases. If this power were to exceed the power consumption the chip can withstand, the circuit might perform slower or might malfunction [52].

This research aims at reducing the time required for test without increasing the power dissipated during test. Full scan design is a popular design for testability (DFT) method [11] in which the flip-flops of the circuit are chained together to function as a shift register during test. Test vectors are scanned in and the responses are scanned out bit by bit. The power consumption during test can exceed the power consumption in the functional mode of operation [12, 53] due to high activity required to achieve high test coverage for the circuit under test. Therefore, the scan-in and scan-out of vectors are normally carried out at frequencies much lower than functional frequencies. However, all vectors do not create the same amount of activity in the circuit. The vectors that cause low activity in the circuit can be scanned in at higher frequencies without exceeding the power limit.

A scheme to reduce test application time by dynamically increasing the scan clock frequency is proposed. The test power is held below the allowed power limit by controlling the activity per unit time. The per cycle scan activity is monitored dynamically to speed up the scan clock for low activity cycles without exceeding the specified peak power budget. The

implementation of the dynamic control of scan frequency in circuits tested by both built-in self-test (BIST) and ATE are discussed.

In the newly proposed techniques, on-chip activity monitors are installed at the front end of every scan chain in the circuit. These activity monitors continuously keep track of the number of transitions in the circuit. The power dissipated in a circuit has a direct relationship with the activity per unit time in the circuit [23]. Thus, if the number of transitions in the scan chain falls below the peak value allowed, the frequency of scan-in of vectors can be increased without exceeding the power limit the circuit can withstand.

A frequency divider circuit was implemented on the circuit. The fastest clock at which scan-in is to be performed is fed to the frequency divider. Based on the number of transitions in the scan chain, the frequency divider block modifies the frequency of clock supplied to the scan chain. Thus, the time required to scan in vectors is reduced by dynamic control of scan clock frequency without exceeding the power limit the circuit can withstand.

In the case of circuits tested with ATE, a handshake protocol may control the rate of test data flow between the ATE and device under test (DUT). The use of ATE allows information about the activity factor of the vectors in the scan chain to be utilized during test. The activity factor of the test vectors can be used to determine the frequency at which scan-in should be carried out. This information is stored on the ATE and used for dynamic control of scan clock frequency.

The dynamic clock control method was implemented on ISCAS89 benchmark circuits. The test-per-scan BIST model [59] and one scan chain per circuit were used for simulation purposes. Test time reductions of up to 19% with a 2-3% increase in area were achieved in large benchmark circuits. All circuits showed test time reduction without increase in test power. It was found that a test time reduction of 19% can be achieved with fully specified test vectors on the s38584 circuit and that a test time reduction of 43% is achievable when don't care bits are present in the test vectors. An accurate mathematical analysis on ITC02 benchmark circuits showed test time reductions of up to 50% for test sets with low activity

and up to 25% for test sets with moderate activity. A reduction of up to 49.9% was achieved in t512505 circuit when the peak activity factor of test vectors was lower than 1. Significant reduction in test time was achieved in t512505 circuit when information about scan chain activity was pre-simulated and the frequency of scan clock was stored on ATE.

It was observed that the proposed method performed better on large circuits. It shows larger reduction in test time when the test vectors have low activity factors. Thus, this method would perform very well on circuits for which vectors are optimized to have minimal activity factors. With the emphasis on power prevalent today, it is common to see vector sets being optimized for power. This indirectly leads to low transition densities in vectors because of the direct relationship between activity factor and power. Hence, the dynamic control of clock frequency method is bound to produce good results on today's industrial circuits.

Acknowledgments

I would like to thank everyone who directly or indirectly helped and inspired me during the course of my graduate study.

This thesis would not have been possible without the consistent support and constant guidance from my advisor, Dr. Vishwani Agrawal. It is difficult to overstate my gratitude for his patience, commitment and immense knowledge. His enthusiasm towards research will continue to inspire and motivate me.

I thank Dr. Adit Singh and Dr. Charles Stroud for being on my committee and Dr. Victor Nelson for his guidance on the project I have been working on. I am grateful for the support from the Wireless Engineering Research and Education Center (WEREC) at Auburn University and for the encouragement I received from its director, Dr. Prathima Agrawal. My work was supported by the National Science Foundation Grant CNS-0708962 for which I am thankful.

It was an honor to be a part of Auburn University and a privilege to learn from its professors. I am indebted to my student colleagues at Auburn University for providing a stimulating environment to learn and grow.

I would like to take this opportunity to thank my colleagues at Texas Instruments, Bangalore and at NVIDIA, Santa Clara for offering me the opportunity and direction to work on exciting projects.

I owe a debt of gratitude to every teacher and professor who taught me Mathematics and Sciences through High School, College and Graduate School. I would especially like to thank Dr. Nagarajan, for mentoring me and urging me to dream bigger. It is a pleasure to thank my parents, Shanmugasundaram V. and Kamalambujam S., sister, Dheepthi S. and the rest of my family and friends for their untiring emotional support.

Table of Contents

Abstract	ii
Acknowledgments	v
List of Figures	ix
List of Tables	xii
List of Abbreviations	xiii
1 Introduction	1
1.1 Problem Statement	2
1.2 Contributions	2
1.3 Organization of Thesis	2
2 Background	4
2.1 Design For Testability (DFT) Techniques	4
2.1.1 Scan Design	4
2.1.2 BIST	6
2.2 External testing	7
2.3 Need for Reduction in Test Time	8
2.4 Power Dissipation during Test	8
3 Previous Work	12
3.1 Reduction in Power Dissipated during Test	12
3.1.1 Low-Power External Testing	12
3.1.2 Low-Power BIST Testing	17
3.2 Reduction in Test Time	22
4 Implementation of Dynamic Control of Scan Clock Frequency in BIST Circuits .	27
4.1 Circuits with Single Scan Chain Tested with BIST	27

4.2	Circuits with Multiple Scan Chains Tested with BIST	32
5	Implementation in BIST circuits with Peak Activity Factors Lower than 1	34
5.1	Circuits with Single Scan Chain Tested with BIST	34
5.2	Circuits with Multiple Scan Chains Tested with BIST	37
6	Background on Communication between Asynchronous Systems	38
6.1	Handshake Protocols	40
6.1.1	Bundled-data protocols	40
6.1.2	Dual-Rail Protocol	41
6.1.3	Other Protocols	44
6.2	The Muller C-element and The Indication Principle	44
6.3	The Muller Pipeline	46
6.4	Circuit Implementation	47
6.4.1	4-Phase Bundled-Data	47
6.4.2	2-Phase Bundled-Data (Micropipelines)	48
6.4.3	4-Phase Dual-Rail	49
6.5	Scan Chain with Mixed Edge-Triggered Flip-Flops	50
6.6	Common Mistakes Made During Synchronization	51
6.6.1	Avoiding the Synchronizer	51
6.6.2	Sneaky Path	52
6.6.3	Wrong Protocol	53
6.6.4	Global Reset	53
6.6.5	DFT Leakage	54
7	Implementation in Externally Tested Circuits with Peak Activity Factors of 1 . .	55
7.1	Circuits with Single Scan Chain Tested with ATE	55
7.2	Circuits with Multiple Scan Chains Tested with ATE	57
8	Implementation in Externally Tested Circuits with Peak Activity Factors Lower than 1	58

8.1	Circuits with Single Scan Chain Tested with ATE	58
8.2	Circuits with Multiple Scan Chains Tested with ATE	60
9	Mathematical Analysis	61
9.1	Circuits with Peak Activity Factors of 1	61
9.2	Circuits with Peak Activity Factors lesser than 1	66
9.3	Externally Tested Circuits	72
10	Experimental Results	82
10.1	Simulations on ISCAS89 Benchmark Circuits	82
10.2	Mathematical Analysis on ITC02 Benchmark Circuits	85
10.3	Mathematical Analysis on t512505 Benchmark Circuit	87
10.3.1	Circuits with Peak Activity Factors lesser than 1	89
10.3.2	Externally Tested Circuits	90
11	Conclusion	95
	Bibliography	97

List of Figures

2.1	Sequential Circuit	5
2.2	BIST Implementation	6
4.1	Implementation of test-per-scan BIST	28
4.2	Circuitry for Implementation of Dynamic Frequency Control Technique	29
4.3	Dynamic Scan Clock Control for Circuit with 8 Flip-Flops - Number of Frequencies Chosen = 4	31
4.4	Modification in Architecture for Circuits with Multiple Scan Chains	32
5.1	Implementation of test-per-scan BIST, activity factor $\neq 1$	35
5.2	Modification in Architecture for Circuits with Multiple Scan Chains, activity factor $\neq 1$	37
6.1	An Abstract Data-Flow View of the Asynchronous Circuit	39
6.2	A Request-Acknowledge Based Hand-Shake	39
6.3	A Bundled-Data Channel	40
6.4	A 4-Phase Bundled-Data Protocol	41
6.5	A 2-Phase Bundled-Data Protocol	41
6.6	A Delay-Insensitive Channel using the 4-Phase Dual-Rail Protocol	42

6.7	Handshaking on a 4-Phase Dual-Rail Channel	43
6.8	Handshaking on a 2-Phase Dual-Rail Channel	44
6.9	The Muller-C Element	45
6.10	The Muller Pipeline	46
6.11	A 4-Phase Bundled-Data Pipeline	47
6.12	Latch in 2-Phase Bundled-Data Pipeline	49
6.13	A 2-Phase Bundled-Data Pipeline	49
6.14	A Simple 3-Stage 1-Bit Wide 4-Phase Dual-Rail Pipeline	50
6.15	Lockup Latch in a Scan Chain	52
7.1	Implementation of Dynamic Control of Scan Clock Frequency in Externally Tested Circuits	56
7.2	Handshake Protocol between ATE and chip	56
7.3	Implementation in Externally Tested Circuits with Peak Activity Factors of 1	57
8.1	Implementation in Externally Tested Circuits with activity factors lower than 1	59
9.1	Variation of Scan-In Time Reduction with the Number of Frequencies for Various Activity Factors	66
9.2	Variation of Scan-In Time Reduction with Activity factor for Varying Number of Clock Frequencies	68
10.1	Activity per Unit Time Analysis for s386 Circuit	84

10.2	Distribution of activity factor for test vectors of s38584 circuit.	86
10.3	Normal Distribution Curve for α_{peak}	88
10.4	Variation of reduction in scan-in time with α_{in} and α_{out} for $\alpha_{peak} \neq 1$	89
10.5	Variation of reduction in scan-in time with α_{in} and α_{out} using Pre-Determined Scan-In-Start Frequencies	91

List of Tables

9.1	Determination of Clock Cycle Range for Different Frequencies	62
9.2	Variation of Scan-In Time Reduction with Chosen Number of Frequencies for an Activity Factor of 0.5	66
9.3	Variation of Scan-In Time Reduction with Activity Factor when the Number of Frequencies Chosen is 8	67
9.4	Non-transition Count Range for $\alpha_{peak} \neq 1$	69
9.5	Determination of Clock Cycle Range for $\alpha_{peak} \neq 1$	70
9.6	Clock Cycle Range for $\alpha_{in} < \alpha_{out}$	74
9.7	Clock Cycle Range for $\alpha_{in} > \alpha_{out}$	78
10.1	Reduction in Test Time in ISCAS89 Circuits - Single Scan Chain and Tested with BIST	92
10.2	Reduction in Test Time in ITC02 circuits	93
10.3	Reduction in Test Time in s38584 Circuit	93
10.4	Reduction in Test Time in t512505 Circuit	94
10.5	Reduction with Pre-Determination of Scan-In-Start Frequency	94

List of Abbreviations

ATE	Automatic Test Equipment
ATPG	Automatic Test Pattern Generator
BIST	Built-In Self-Test
DFT	Design for Testability
DS-LFSR	Dual-Speed Linear Feedback Shift Register
DUT	Device Under Test
EDT	Embedded Deterministic Test
FDR	Frequency-Directed Runlength
FIFO	First-In First-Out
FSM	Finite State Machine
GALS	Globally Asynchronous, Locally Synchronous
HCA	Hybrid Cellular Automaton
ISCAS	International Symposium on Circuits and Systems
LFSR	Linear Feedback Shift Register
LT-RTPG	Low-Transition Random Test Pattern Generator
MISR	Multiple-Input Signature Register
RAM	Random-Access Memory

ROM Read-Only Memory

SAR Signature Analysis Register

SOC System On-Chip

STUMPS Self-Test Using MISR and Parallel Shift register sequence generator

TPG Test Pattern Generator

TRP Test Resource Partitioning

VLSI Very-Large-Scale Integration

Chapter 1

Introduction

The size of circuits has increased phenomenally in the last few decades. The number of transistors on an system on-chip (SOC) rose from a few thousands to billions in a span of 30 years. With the increase in size, the number of test vectors required to test them has also increased. The time taken to test a chip is the product of the number of test vectors applied and the time required to apply each vector. As the number of test vectors increases, the time required to apply them also increases. Since expensive ATE is used to test these chips, the cost per chip increases with increase in test time [11]. There is therefore growing concern about the time required to apply these test vectors.

Full scan design [11] is a popular design for testability (DFT) method in which flip-flops in the circuit are chained together such that input vectors are shifted in and circuit responses shifted out serially through the so-called scan chains. The flip-flops serve as points of controllability and observability, thus increasing the fault coverage.

Power dissipated by a circuit during scan test, is usually higher than during functional operation [12, 53]. If the test power exceeds the specified power limit of the chip the test can cause yield loss [28, 52]. To avoid this, scan testing is carried out at clock frequencies lower than the normal frequency of operation. The scan clock frequency is determined based on the maximum power consumption the circuit under test is designed for based upon its functional requirement.

In general, the clock frequency for scan is computed based on the test vector that causes the most activity in the circuit. All vectors are scanned in and scanned out at this frequency. However, most vectors do not cause maximum activity in the circuit and will dissipate much lower power than the allowed limit. It is possible to scan in these vectors at higher clock

frequencies without exceeding the power budget. This possibility of speed-up of selective vectors has been exploited in the present research in order to reduce the time required for test.

1.1 Problem Statement

This thesis provides solutions to the following problems:

- Computing the activity in the circuit under test caused by every test vector
- Utilizing the information about activity to speed up scan-in and scan-out of test vectors
- Speeding up shift operation in scan chains at run time without exceeding specified power limits

1.2 Contributions

The contributions of the work described in this thesis are:

- Use of on-chip activity monitor to keep track of number of transitions in scan chain(s)
- Introduction of asynchronous communication between ATE and DUT through handshake protocols
- Dynamic control of scan clock frequency to reduce time required for shift operations in scan chains without exceeding peak power limits in self tested and externally tested scan circuits

1.3 Organization of Thesis

Chapter 2 of the thesis introduces readers to various concepts that are important for understanding the significance of the problems solved by the proposed work. Chapter 3 provides information about prior work done to reduce power dissipation during test and to

reduce the time required for test. Chapter 4 explains the implementation of the proposed method for self testing circuits whose test vectors are assumed to have a peak activity factor of 1. Chapter 5 refines the technique of Chapter 4 to include BIST circuits whose test vectors can have any arbitrary peak activity factor, in general, lower than 1. Chapter 6 discusses protocols necessary for application of the proposed techniques to external test by an ATE. Here, asynchronous hand-shaking protocols may be required for communication between the ATE and the DUT. These protocols are similar to those used for communication between digital systems working in different clock domains. Chapter 7 explains the implementation of the proposed method for externally tested circuits whose test vectors are assumed to have a peak activity factor of 1. Chapter 8 generalizes the method of Chapter 7 for externally tested circuits whose test vectors have any activity factor that can be lower than 1. Chapter 9 analyzes the test time reduction achieved with the proposed architecture from a mathematical perspective. Chapter 10 discusses the experimental results obtained during the implementation of the proposed architecture on different benchmark circuits. Chapter 11 is an observation of trends and inferences drawn out of the research work.

Chapter 2

Background

This chapter deals with the theory behind the topics analyzed in this thesis. The first section of this chapter is a study of two DFT techniques that are widely used. The second section discusses why high test application time is a problem worth solving. The third section explains the power considerations during test, which plays an important role in deciding the scan clock frequency.

2.1 Design For Testability (DFT) Techniques

As discussed earlier, as the size of circuits increases, their test complexity also increases. The internal nodes in the circuits become harder to test. Circuits are therefore modified so that they can be tested better. This section describes some of the techniques used to improve the quality of test.

2.1.1 Scan Design

A combinational circuit with n inputs has 2^n possible input combinations. As n increases, the number of possible input vectors increases exponentially. It is therefore impossible to apply all possible input vectors to test the circuit. A subset is therefore chosen such that a sufficient percentage of the faults can be captured by the test.

Sequential circuits are harder to test than combinational circuits. This is due to the presence of memory elements (shown in Figure 2.1) which create internal states during circuit operation. An exhaustive test would involve application of all possible input vectors at all possible states of the memory elements. This number becomes large even for small circuits.

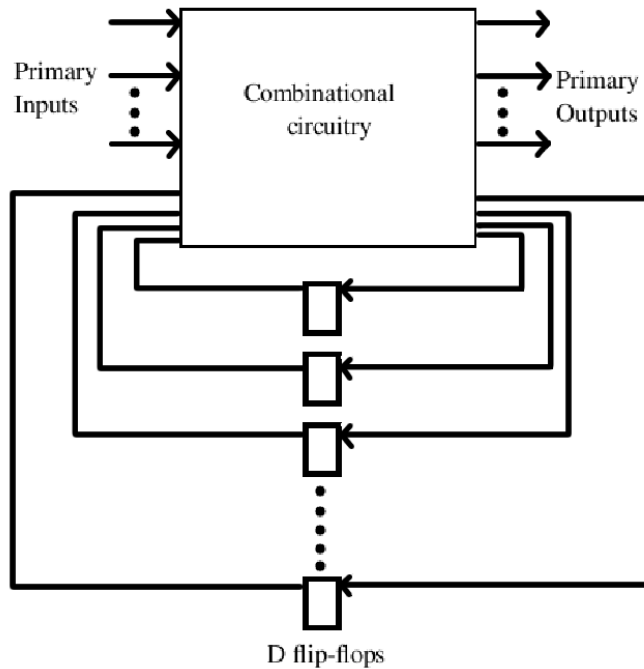


Figure 2.1: Sequential Circuit

In order to improve the testability of sequential circuits, they are enabled with a test mode. When the circuit is in the test mode, the flip-flops in the circuit are chained together to form one or more shift registers. Thus, the flip-flops can be sent to any state without depending on the values at the primary inputs. The flip-flops serve as points of controllability and observability and help in achieving better test coverages.

There are two widely used types of scan designs - full scan and partial scan designs. Full scan designs utilize all flip-flops in the circuit to generate shift registers [11]. Partial scan designs [1] use a selective set of flip-flops to form shift registers. The flip-flops are chosen [18], [13] such that they minimize overhead without loss of coverage.

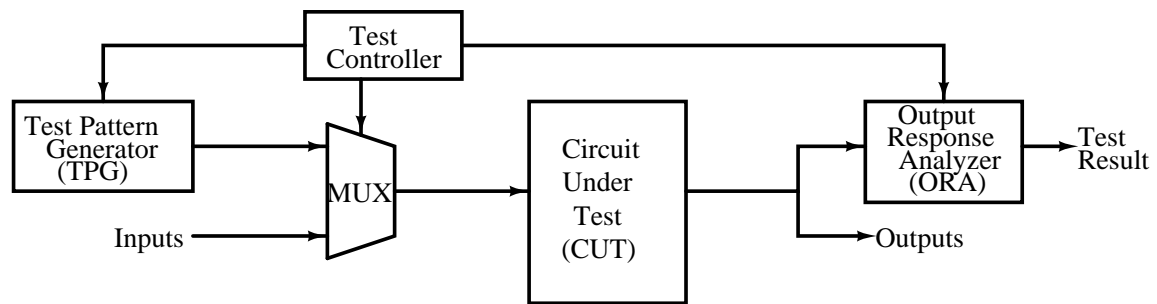


Figure 2.2: BIST Implementation

2.1.2 BIST

BIST is a DFT technique in which additional hardware is added to the circuit to be tested so that it can test itself. BIST is widely used since it makes the chip easier and faster to be tested. The basic circuitry required to implement BIST is shown in Figure 2.2.

The patterns required for test are generated through a number of techniques [11]. One of them is to store the test patterns in a ROM on the chip. This method uses a lot of chip area and is hence not very widely used. Counters can be used to generate exhaustive test sequences. However, the number of exhaustive inputs is very high for any normal-sized circuit and hence the test time required is very high. A more commonly used technique is the use of a Linear Feedback Shift Register (LFSR) that generates pseudo-random pattern sets. A large number of test patterns are used in this method but the area overhead on-chip is very low.

A large number of outputs are received from the circuit under test. Storing the correct values of all these bits would add a lot of extra hardware to the chip. The circuit responses are therefore reduced to a size that can be stored on the chip. This is done through a number of response compaction methods. A widely used method compacts responses with an LFSR [25]. Some other methods count number of transitions, or use parity information and so on.

Test-Per-Clock BIST Systems

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

Test-Per-Scan BIST Systems

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

2.2 External testing

In order to verify if circuits are manufactured without any faults, it is essential to test every chip after it is manufactured. The cost of a chip is normally very low when compared to the final product it is used in. Thus, it is better from an economic perspective to detect a faulty chip right after manufacture.

The patterns required for test are generated with an automatic test pattern generator (ATPG). The patterns have the correct output response for each input vector. If scan design is used in the circuit, the scan clock frequency is decided based on power dissipated during scan and the power the circuit can withstand without malfunctioning.

The patterns, timing information and voltage levels required for testing are stored in the ATE. A test program that puts these information together such that the patterns are applied at the right frequency and voltages is written.

After manufacture, the chips are mounted on the tester, and the patterns are applied to the primary inputs of the circuit. The responses from the primary outputs are analyzed

by the tester to check if they are the same as the correct circuit responses. If the output is different from that stored on the ATE, the chip is rejected.

When ATE is used, the cost of testing the chip adds on to the cost of the chip. However, this method allows the testing of circuits without significant addition of circuitry to the existing hardware.

2.3 Need for Reduction in Test Time

The cost of a chip depends on a number of factors [11] and the cost of test is one of them. Expensive ATE is used to test these chips. The amount of time the ATE is utilized directly affects the cost of the chip.

Due to the increasing size and complexity of circuits, testing these circuits is becoming more complex. This has resulted in an increase in number of test vectors required to test them sufficiently. The time required for application of test is a product of the number of test vectors and the time required to apply each vector. Once a minimal pattern set is obtained, dropping some of them would result in loss of test coverage. It is therefore necessary to reduce the time required to apply each test vector in order to reduce the total test application time.

2.4 Power Dissipation during Test

The power dissipated in CMOS circuits can be classified into static and dynamic power. Power that is drawn continuously from the power supply due to factors such as leakage current cause static power dissipation. Power that is drawn during switching of states in the circuits due to short-circuit current or due to charging and discharging of load capacitances cause dynamic power dissipation.

Activity in circuits is much higher during test than during functional operation [66], [46]. This is because of the dependency of test coverages on high toggle rates in the circuit and this in turn leads to very high switching activity in the circuit.

The power dissipated at any node [23], P is given by

$$P = \frac{1}{2}CV^2\alpha_{node}f \quad (2.1)$$

where C is the capacitance of the node, V is the voltage, f is the clock frequency and α_{node} is the average activity factor of a node in the circuit.

$$\alpha_{node} = \textit{Fraction of gates switching in the circuit} \quad (2.2)$$

$$= \frac{\textit{Number of gates switching in the circuit}}{\textit{Total number of gates in the circuit}} \quad (2.3)$$

Thus, the activity factor for a clock signal is 2 according to this definition since there are two transitions, one rising and one falling in every cycle. In the case of glitch-free combinational circuits, average activity factor of a node in the circuit (α_{node}) ranges from 0 (when there are no transitions in any of the clock cycles) to 1 (when there is a transition in every clock cycle).

In the worst case, the frequency of the scan clock can be based on the power dissipated at all nodes when the circuit undergoes maximum activity, i.e., $\alpha_{node} = 1$, so that the test power can never exceed the power limit. Therefore,

$$P_{budget} = \frac{1}{2}CV^2f_{test} \quad (2.4)$$

where f_{test} is the clock frequency for scan-in of vectors and P_{budget} is the maximum power dissipation the circuit can withstand without malfunctioning. If two power budgets, one for peak power and one for average power are specified, the lower of the two is chosen as P_{budget} .

Thus,

$$f_{test} = \frac{2P_{budget}}{CV^2} \quad (2.5)$$

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

When the number of transitions in the circuit reduces to $\frac{1}{i}$ of the maximum number,

$$P = \frac{1}{2}CV^2\frac{1}{i}f_{test} \quad (2.6)$$

From (2.4) and (2.6),

$$\frac{P}{P_{budget}} = \frac{1}{i} \quad (2.7)$$

At any node, the capacitance and the voltage are constant. Therefore, the power is proportional to the product of activity and frequency. Since the circuit can withstand a power of P_{budget} , the frequency can be multiplied by i , and the power dissipated in every cycle can still be kept below the allowed limit. Girard [28] defines peak power as the highest energy consumed during any clock cycle divided by the clock period and the average power as the total energy divided by the test time. Since the power dissipated never exceeds P_{budget} in any clock cycle, both peak and average power will be below P_{budget} in spite of the increase in shift frequency. [28] defines instantaneous power as the power that is consumed right after the application of a synchronizing clock signal. The instantaneous power dissipated during test depends on the vectors scanned in. It is unaffected by changes in frequency and hence, the change in scan clock frequency does not modify the value of instantaneous power.

It is evident from (2.1) that the power dissipated in a circuit is a function of the signal activity and the frequency at which vectors are scanned in. During scan tests, the gates are either driven by the outputs of the scan flip-flops or by primary inputs. Therefore, the activity in the circuit is proportional to the activity in the scan chain and primary inputs.

The latter do not change during scan in and scan out. Thus, scan chain activity is a direct measure of the test power [58], and hence, by controlling the scan chain activity per unit time, it is possible to control the test power. The measure of scan chain activity per unit time is defined by α which is the activity factor of a scan chain and is defined as

$$\alpha = \frac{\textit{Number of transitions in the scan chain}}{\textit{Total number of flip - flops in the circuit}} \quad (2.8)$$

Chapter 3

Previous Work

This chapter discusses previous work done that relates to the problems solved in this research work. Section I describes the work done to reduce power dissipated during test. Section II deals with prior work done in the area of test time reduction.

3.1 Reduction in Power Dissipated during Test

In order to achieve good test coverage, high toggling of nodes is required during test. However, this leads to high power dissipation during test. Since activity at nodes during test is higher than during normal functioning, test power usually exceeds the power limit set for normal operation of chip. Reduction of test power is therefore a widely recognized problem for which a number of solutions have been presented. Girard [28] summarizes the different techniques proposed for low-power testing of VLSI circuits. They are broadly classified into low-power external testing techniques and low-power BIST techniques.

3.1.1 Low-Power External Testing

A number of techniques have been proposed to reduce test power when circuits are tested with an ATE. They are classified into different categories and presented in this section.

ATPG Algorithms

An ATPG generates test patterns with a primary aim of achieving maximal test coverage. ATPG algorithms that generate test patterns that reduce test power have been proposed.

Wang and Gupta [60] proposed a new (combinational) ATPG algorithm that reduced switching activity between successive test vectors during test application. The objective of the ATPG was to ensure that switching activity during test application was low enough to ensure safe and nondestructive at-speed testing. The new ATPG algorithm reduced average heat dissipation between successive test vectors. The proposed algorithm was implemented on ISCAS85 benchmark circuits and the results demonstrated that the tests generated decreased the average number of weighted transitions between successive test vectors by a factor of 2 to 2.3.

Wang and Gupta extended their work to full-scan designs in [61] to reduce heat dissipation during testing of sequential circuits. The proposed ATPG exploited all don't cares that occur during scan shifting, test application, and response capture to minimize the switching activity in the circuit. Also, an ATPG that maximized the number of state inputs assigned was introduced. The tests generated by the proposed ATPG reduced the average number of transitions during test by 19% to 89% when implemented on ISCAS89 benchmark circuits.

Corno et al. [21] proposed an ATPG technique that reduced power dissipation during the test of sequential circuits by exploiting the redundancy introduced during test pattern generation. The proposed approach selected a subset of sequences that reduced test power dissipation without decreasing fault coverage. The results indicate a reduction of 70% in the power consumption when the ATPG was implemented on ISCAS benchmark circuits.

Test Vector Ordering

Power consumed during test can be reduced by lowering the switching activity generated in the circuit during test. Test vector ordering techniques reduce the switching activity by changing the order in which test vectors are applied to the circuit.

Chakravarty and Dabholkar [14] proposed two techniques for reducing power dissipation during test for circuits employing scan design. A directed graph is constructed in which the vertices represent test vectors and the edges represent the number of transitions generated

by the vector pair represented by the two vertices the edge connects. A greedy algorithm is used to find a Hamiltonian path of minimum transitions in the graph. Experimental results show an improvement of 1.21% to 14.55% in ISCAS89 benchmark circuits.

Girard et al. [34] proposed a technique to reduce test power based on re-ordering of test vectors in the test sequence. The technique used the Hamming distance between test vectors to order them. This was an improvement over the use of circuit switching activity caused by a vector pair since such a method would involve high simulation time in large circuits. The proposed method guaranteed a decrease in power consumption and heat dissipation without any change in fault coverage. The method achieved a reduction of 8.2% to 54.1% in the circuit activity during test application.

Girard et al. [30] presented a method to reduce internal switching activity in circuits by decreasing the transition density at the inputs through test vector ordering. The proposed method minimized the average and peak power dissipation during test by considering the circuit's structural characteristics. Experimental results show reductions between 11% and 66% in switching activity.

Dabholkar et al. [23] introduced techniques for reducing power dissipation during test in scan and combinational circuits. They showed that scan-latch ordering (applied during synthesis) along with test-vector ordering gave a significant improvement in power dissipation. The inputs to the combinational part of the circuit were also disabled during scan-in in order to reduce power dissipation. The scan-latch ordering technique reduced power dissipation during test by 10% to 25

Bonhomme et al. [8] proposed a method that reduced test power by first determining the chaining of the scan cells in order to minimize the occurrence of transitions in the scan chain during shifting operations and then by identifying input and output scan cells of the scan chain to limit the propagation of transitions. The approach works for conventional scan designs and achieved reductions of average and peak power consumption of up to 58% and 24% respectively in ISCAS benchmark circuits

Input Control

This technique involves the control of primary inputs of the circuit such that the switching activity in the circuit is reduced during scan.

Huang and Lee [37] developed an input control technique that minimized the switching activity of full-scan circuits during test application. The technique proposed involves the identification of an input control pattern for a full-scan circuit such that the application of the pattern at the primary inputs of the circuit during scan operation minimized or eliminated the switching activity in the combinational part of the circuit. The experimental results shown indicate that 29.28% average improvement can be achieved if this technique is employed before vector ordering and latch ordering techniques.

Vector Compaction and Data Compression

These techniques involve merging of test cubes and reduction of test vector data without affecting the fault coverage achieved by the test sequence.

Sankaralingam et al. [50] proposed a method to carefully select the order in which pairs of test cubes were merged during static compaction such that both average and peak powers during test are reduced. The proposed approach was found to be more effective than conventional static compaction techniques that randomly merged test cubes. Experimental results showed that the proposed technique always had lower peak power compared to conventional static compaction techniques.

Chandra and Chakrabarty [17] presented a technique to reduce both test data volume and scan power dissipation using test data compression. They have shown that Golomb coding of precomputed test sets reduced peak and average power without slowing the scan clock or blocking logic in the scan cells. They have also shown that a separate cyclical scan register is not necessary for pattern decompression. Experimental results indicate average reductions of 84.32% and 37.72% in peak and average powers respectively.

Modification of Scan Chain

The scan chain architecture can be modified to reduce power consumption during test. Some of the techniques proposed that utilized this technique for power reduction are discussed in this section.

Whetsel [64] described a method of adapting conventional scan architectures so that they operated in a low power mode during test. The conventional scan architecture was modified into a scan path having a desired number of selectable, separate scan paths. An adaptor circuit was added to intercept the scan control output from the tester and transform it into separate scan control outputs to the new scan paths. The adapted architecture maintained the test times of the pre-adapted architecture and was designed such that the test patterns of the pre-adapted architecture was directly reusable. It was found that dividing the scan path into two scan paths achieved a reduction of 50% on power consumption.

Lee et. al [41] proposed a method to lower the peak power of multiple scan chain based circuits during test. An interleaving scan architecture is proposed that uses delay buffers among scan chains in order to reduce the peak power. The improvement was found to be up to 50% when the data output of a scan cell was affected by the scan path during scan. When the data output was disable during scan 76% reduction in peak power was observed.

Modification of Clock Scheme

Some techniques that reduce test power through modification of the clock scheme used are discussed below.

Pouya and Crouch [45] demonstrated that a major contributor towards test power is the clock tree through a case study of a Motorola Version 3 ColdFire microprocessor core.

Sankaralingam et al. [51] proposed a technique that generated the test set and ordered it in such a way that some of the scan chains could have their clock disable for portions of the test set, given a full scan design with multiple scan chains. Flip-flop transitions were prevented due to the disabling of the clock and hence switching activity during test was

reduced. Since the clock tree is a major contributor towards test power, disabling the clock led to reduction in power dissipation. Implementation of the technique on Motorola V3 ColdFire Core led to a reduction of 16% in power dissipation.

Bonhomme et al. [7] presented a method to minimize power consumption during scan test of integrated circuits or embedded cores. The technique was based on a gated clock scheme for the scan path and the clock tree feeding the scan path. The principal idea used was to reduce the clock rate on scan cells during shift operations without increasing the test time. This lowered the transition density in the circuit, scan path and the clock tree feeding the scan path thus leading to minimization of average power, peak power and energy consumption. Peak power reduction of up to 49.5% was reported in experimental results.

3.1.2 Low-Power BIST Testing

This section discusses some of the techniques available to minimize power dissipation during test when the circuit includes BIST. They have been classified into seven major categories and presented in this section.

Test Scheduling Algorithms

Zorian [12] presented a BIST scheduling process that took power, noise, area overhead and other such constraints that limit the possibilities of parallel BIST execution in complex VLSI devices into consideration. The process included a BIST control methodology that implemented the BIST schedule with a highly modular architecture. The control architecture provided an autonomous BIST activation and a diagnostic capability to identify failed blocks. The technique reduces average power and hence avoids temperature related problems.

Chou et al. [20] presented optimum test scheduling algorithms for equal and unequal test length cases under power constraints. The algorithms first found a complete set of time compatible tests with power dissipation information associated with each test, then extracted

the lists of power compatible tests and finally used a minimum cover table approach to find the optimal scheduling of the tests. This algorithm reduces the average power consumption.

Iyengar and Chakrabarty [38] proposed an integrated framework to determine optimal SOC test schedules. They also proposed a new algorithm that used preemption to obtain optimal test schedules in polynomial computation time.

Test Pattern Generators

Test pattern generators that generate test vectors that cause lower power dissipation during BIST are discussed in this section.

Wang and Gupta [62] proposed a test pattern generator for BIST which can reduce heat dissipation during test application. They developed a dual-speed LFSR (DS-LFSR) that consisted of two LFSRs, a low LFSR and a normal speed LFSR. The use of such a technique reduces the transition density at the circuit inputs driven by the slow LFSR which leads to a reduction in the heat dissipation during test. A procedure was introduced to design a DS-LFSR such that high fault coverage was achieved through unique and uniformly distributed patterns. New methods to select inputs driven by the slow LFSR and to increase the number of inputs driven by the slow LFSR were presented. The technique showed 13% to 70% reduction in the number of transitions in ISCAS benchmark circuits without loss of fault coverage.

Corno et al. [22] proposed an algorithm to design a Cellular Automata based Test Pattern Generator (TPG) to test combinational circuits. The TPG was aimed at reducing power consumption in addition to targeting high fault coverage. The algorithm selected an optimal non linear hybrid cellular automaton (HCA) from the point of view of power consumption for given coverage and test length constraints. Experimental results indicated an average test power reduction of 34% without affecting fault coverage, test length and area overhead.

Girard et al. [33] presented a low power test-per-clock BIST test pattern generator (TPG) that generated test vectors capable of reducing the switching activity during test. The technique was based on a modified clock scheme and the clock tree feeding the TPG. Reductions of up to 60% and 61% were noted in power and energy when the proposed technique was implemented on ISCAS benchmark circuits.

Zhang et al. [65] presented an energy conscious weighted random pattern testing technique for BIST applications. A tool, POWERTEST was developed which used a genetic algorithm based search to determine optimal weight sets at primary inputs to minimize energy dissipations. The technique involved the modification of the LFSR by adding weight sets to tune the pseudorandom vector's signal probabilities and thus decreasing energy consumption, along with an increase in fault coverage. Results on ISCAS benchmark circuits showed an energy reduction of up to 97.82% while still achieving high fault coverage.

Wang and Gupta [63] presented a new BIST TPG design, called low-transition random TPG (LT-RTPG) that comprised of an LFSR, a k-input AND gate, and a T flip-flop. The LT-RTPG generated test patterns for test-per-scan BIST that decreased the number of transitions that occurred during scan shifting and thus decreased the heat dissipation during testing. The new TPG reduced the number of transitions in ISCAS89 benchmark circuits by 23% to 59%.

Gizopoulos et al. [35] proposed low power BIST schemes for datapath architectures built around multiplier-accumulator pairs, based on deterministic test patterns. They have also proposed two alternatives based on whether the design is low energy dissipation or low power dissipation during a BIST session. Both methods are based on modified binary counters, operating as Gray counters. The technique offers up to 78.33% energy saving and up to 82.22% power saving compared with pseudorandom BIST.

Toggle Reduction

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

Hertwig and Wunderlich [36] introduced a low power technique for scan-based BIST architectures that modified the scan-path structure's scan cells such that the inputs to the CUT remained unchanged during shift operations. Energy savings of up to 90% were seen in a standard, scan-based BIST architecture.

LFSR Modification

This method involves the tuning of LFSR to minimize energy and power. One such technique is discussed in this section.

Girard et al. [32] proposed a technique to minimize the energy required to test combinational circuits with BIST without altering fault coverage. They have analyzed the impact of the polynomial and seed selection of the LFSR used as TPG on the energy consumed by the circuit and found that appropriate selection of the seed of the LFSR can contribute to energy reduction. They have also proposed a heuristic based on a simulated annealing algorithm to decrease the energy consumption of BIST runs.

Vector Filtering BIST

BIST patterns that do not detect faults can be removed from the test sequence without loss of fault coverage. This section discusses such techniques.

Girard et al. [29] proposed a test vector inhibiting technique to tackle the increased activity during test operation. A mixed solution based on a reseeding scheme and the vector inhibiting technique was also proposed in order to deal with hard-to-test circuits that contain pseudo-random resistant faults. The technique reduced the total energy consumption during test and allowed the test at system speed in order to achieve high delay fault coverage. The test vector inhibiting technique was used to filter out nondetecting subsequences of

pseudorandom test sets generated by LFSRs. A decoding logic to store the first and last vectors of the nondetecting subsequences to be filtered was used. This was implemented through a D-type flip-flop working in toggle mode that switched the enable/disable mode of the LFSR outputs to perform selective filtering. Experimental results showed weighted switching activity reductions ranging from 18.5% to 78.5% without loss of stuck-at fault coverage.

Manich et al. [44] proposed two techniques to reduce the energy and average power consumption of the system that were based on the fact that as the test progresses, the detection efficiency of the pseudo-random vectors decreases. The number of pseudo-random vectors that will not detect faults increases. These vectors consume energy without contributing to fault coverage. The proposed techniques filter all the nondetecting subsequences which is an extension of the technique used in [29]. The techniques when implemented on ISCAS85 benchmark circuits resulted in up to 90% (with an average of 74.2%) reduction in energy consumption using the first technique and up to 97.2% (with an average of 90.9%) reduction in energy consumption using the second technique.

Gerstendörfer and Wunderlich [26] used the technique of filtering nondetecting patterns for scan-based BIST architectures. The paper analyzed scan-based BIST systems like the self-test using multiple-input signature register (MISR) and parallel shift register sequence generator (STUMPS) architecture. The modules and modes with the highest power consumption were identified and design modifications to reduce power consumption were proposed. These modifications included gating logic to mask the scan path activity during shifting, and the synthesis of additional logic for suppressing random patterns that did not contribute to the fault coverage. They combined a pattern-filtering technique with the technique used in [36] to avoid scan-path activity during scan shifting. This method reduced test power by several orders of magnitude with very low penalties in area and performance.

Circuit Partitioning

A technique that lowers the power dissipation during test with BIST through circuit partitioning is discussed in this section.

Girard et al. [31] proposed a low power/energy BIST technique based on circuit partitioning that partitioned the original circuit into structural subcircuits so that each subcircuit could be successively tested in different BIST sessions. This reduced the switching activity in a time interval which in turn reduced the average power and peak power. The technique also reduced the total energy consumed during BIST since the test length required for the subcircuits was not much more than that of the original circuit. The proposed strategy worked for both test-per-scan and test-per-clock BIST schemes. Experiments on ISCAS benchmark circuits showed peak power reduction of up to 57%, average power reduction of up to 62% and energy reduction of up to 82% with low area overhead and almost no penalty on circuit performance.

Low Power RAM Testing

This section is a discussion about a technique used to reduce power dissipation during testing of RAMs.

Cheung and Gupta [19] presented new version of memory tests to reduce heat dissipation during testing. The proposed technique is based on RAM transition reduction through reordering of the read and write accesses and the address counting scheme. The method decreased the energy consumption keeping test time the same, thus reducing average power. The fault coverage and time complexity of the proposed tests are the same as that of the original test but the heat dissipation was reduced by a factor of two or more.

3.2 Reduction in Test Time

A number of techniques have been proposed to minimize test time in scan circuits. Most achieve test time reduction through compression. In a simple compression technique, the

number of scan chains in the circuit is increased by reducing the number of flip-flops per chain. This in turn reduces the time needed to shift the input vector bits through scan flip-flops. This results in an overall reduction in test time. However, compression techniques require alterations in the design and may also suffer from linear dependencies.

Bayraktaroglu and Orailoglu [5] proposed a compression technique that reduces test data volume and test application time. The compression scheme proposed increases the number of scan chains that can be supported by an ATE. However, the functionality of the ATE is left unmodified by moving the decompression task to the circuit being tested. Thus, the number of internal scan chains driven by the decompressed pattern sequence can be increased while still keeping the number of scan chains visible to the ATE small. Compression levels up to 88% are reported using this method.

Bonhomme et al. [9] used a new scan tree architecture to reduce test application time. The architecture was based on a dynamic reconfiguration mode that reduced the dependence between the test set and the final scan tree architecture. Experimental results for benchmark circuits showed up to 95% reduction in test shift time when this technique was used.

Bayraktaroglu and Orailoglu [6] described a method to determine decompression hardware that guaranteed complete fault coverage for a unified compaction/compression scheme. Information about test cubes was used for the determination of an optimal decompression hardware. This method simultaneously increased compression levels and reduced pattern counts through the use of a linear decompression hardware. Test time reduction of 94% is reported in one of the ISCAS89 benchmark circuits. The results shown in this paper indicate that a compression algorithm with concurrent application of compaction and compression is superior to compression schemes that perform compaction and compression serially.

Bayraktaroglu and Orailoglu [4] proposed a test pattern compression scheme for large designs with a high number of scan chains to reduce test data volume and test application time. The number of scan chains visible to the ATE is reduced by using an on-chip decompression network between the ATE outputs and scan chain inputs. The number of

scan chains in the circuit are increased without any increase in the pin count requirements. The reduction in pattern volume reduces the ATE memory requirement and the increase in number of scan chains reduces the number of flip-flops per scan chain thus decrease the test application time. The results shown in the paper indicate a test time reduction of about 84%.

Rajski et al. [48] introduced Embedded Deterministic Test (EDT) technology. The EDT technology proposed consisted of logic embedded on a chip and a new deterministic test pattern generation technique [47]. The EDT logic is inserted along the scan paths outside the design core and consists of an on-chip decompressor between the external scan channel inputs and the internal scan chain inputs and an on-chip selective compactor between the internal scan chain outputs and the external scan channel outputs. The proposed technique achieved compression in excess of one order of magnitude.

Rajski et al. [16] presented a Test Resource Partitioning (TRP) technique that reduced test data volume, test application time and scan power. The technique employed alternating run-length or frequency-directed runlength (FDR) codes [15] for test data compression. The FDR code is a data compression code that maps variable-length runs of 0s to variable-length codewords which makes compressing data more efficient. The approach was shown to reduce test application time from 61.83 ms to 12.757 ms in a production circuit from IBM with 32200 latches and a total of 30 scan chains.

Lai et al. [40] employed a two phase testing strategy where the first phase was a scan-less phase for easy-to-detect faults and the second phase was a scan phase for hard to detect faults. Also, scan was performed only until all effective test bits were shifted to the right position and until all fault-affected response bits were shifted out. The test application reduction problem was studied from three aspects - test generation, selective scan and rearrangement of scan path. An ordering heuristic that maximized the reduction unnecessary scans was also proposed in this paper. A test application time reduction of 20% was reported in ISCAS89 benchmark circuits.

Rudnick and Patel [49] used genetic algorithms to generate compact test sets which limited the scan operations. The test application time can be reduced if flip-flop values are not scanned in and out for every test vector. Deterministic fault-oriented combinational and sequential circuit test generators can be used to determine when flip-flops need to be scanned and this paper took a genetic approach towards this. The results obtained in this paper indicated a significant reduction in test time but cannot be compared to existing techniques since different partial scan flip-flops were used and also, the fault coverages were lower.

Lee and Saluja [42] proposed an algorithm to reduce test application time by generating a test for full scan design using combinational and sequential test generation algorithms adaptively. The paper also presented heuristics combining test measures and scan strategies. The number of test clocks used were reduced to up to 84%. The average reduction in test application time was about 36% ranging from 7% to 84%.

Lee and Saluja [43] presented two algorithms to generate test sequences that reduced the number of test clocks required to apply the test vectors. The paper defined and investigated scan strategies for full and partial scan designs. Approximate measures that were used for selection of a target fault during sequential test generation were proposed. These were integrated into test application time reduction algorithms for full and partial scan designs. In full scan designs, 36% fewer test clocks were required compared to test clock requirements in existing techniques. In partial scan designs, 30% cumulative test clock reduction was reported.

Tsai et al. [59] proposed a strategy for identifying flip-flops to be removed from the scan chains to increase the observability of the circuit so that faults activated during scan cycles could be observed at the primary output. The flip-flops with relatively high pseudo-random observabilities through the primary outputs were not converted into scan flip-flops. This had an advantage of improving the fault coverage in addition to reducing test application time.

Test time reductions ranged from -16.4% to 97.96% when this method was implemented on ISCAS89 circuits.

The technique proposed in this thesis can be applied to any scan circuitry, and can also be applied as an addition to any of the methods mentioned above. Test application times achieved in the above described methods can be further reduced using dynamic scan clock frequency control technique.

Chapter 4

Implementation of Dynamic Control of Scan Clock Frequency in BIST Circuits

This chapter describes the hardware used to implement the dynamic scan clock control technique on circuits tested with BIST circuitry [54]. It is assumed that the peak activity factor used for computing peak power consumption is 1. The scan clock frequency is found from this value of peak power.

4.1 Circuits with Single Scan Chain Tested with BIST

The circuit model chosen for analysis is the test-per-scan BIST model [59]. Flip-flops are added at primary inputs and primary outputs of the sequential circuit as shown in Fig. 4.1. All the flip-flops are converted into scan flip-flops and are connected as a single scan chain. An LFSR, a Signature Analysis Register (SAR) and a BIST controller are added to the circuit to implement the test per scan BIST architecture. In this model, the input vectors are scanned in and the output vectors are captured through the scan flip-flops. A test is applied to the circuit only after all the input bits are scanned into the scan chain.

The circuitry for dynamic frequency control is shown in Figure 4.2. Test vectors are scanned in serially through the scan chain. The activity or inactivity in the scan chain is monitored by examining the activity at the first flip-flop of the chain, because the same activity ripples through other flip-flops in subsequent cycles. The reason why the scan chain is not monitored at the last flip-flop as well is discussed later in the same section. It should be noted that the activity in a scan chain does not change if there are inversions in the chain. When a transition passes through an inverting flip-flop, a rising transition becomes a falling transition and a falling transition becomes a rising transition and thus, the number of transitions remains unchanged.

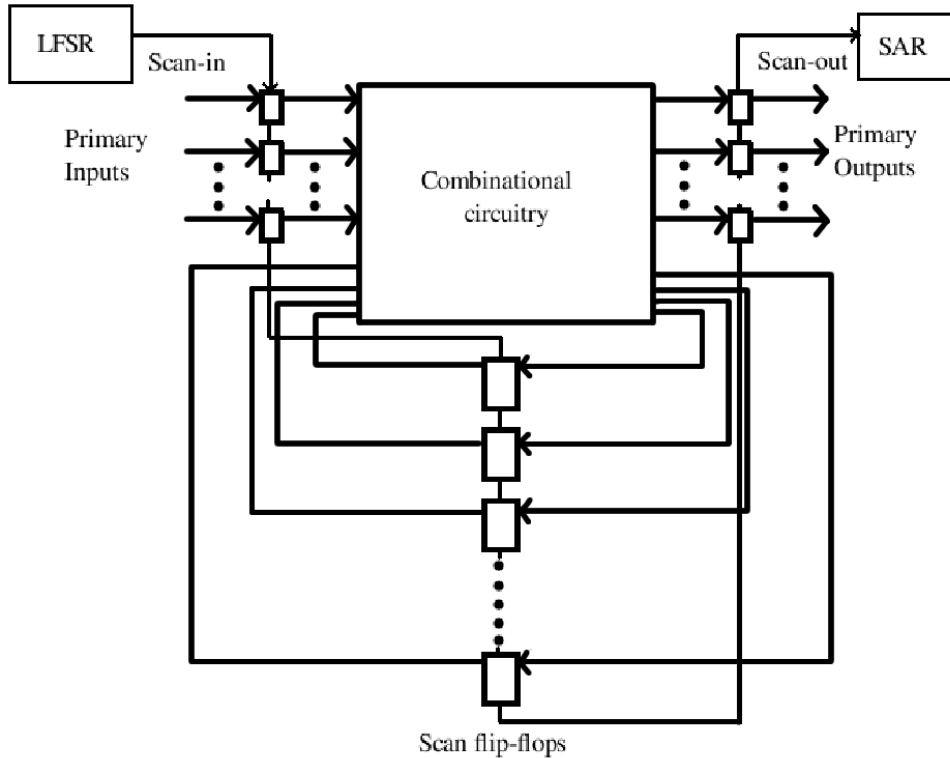


Figure 4.1: Implementation of test-per-scan BIST

We implement an activity monitor by inserting a simple XNOR gate between the input and output of the first flip-flop. The output of the XNOR gate is 0 when a transition enters the scan chain, and is 1 when a non-transition enters the scan chain. The output of the gate is fed to a counter, which counts up every time the gate output is 1, i.e., when a non-transition enters the scan chain. This counter is reset at the start of every scan in sequence.

Since power is proportional to the activity in the scan chain, test power is lower when the number of transitions in the scan chain is lower or, in other words, when the number of non-transitions in the scan chain is higher. As discussed earlier, from (2.7), the scan frequency can be increased when the number of non-transitions in the scan chain increases.

Thus, when the counter counts up to a certain value, the frequency is increased proportionally. This is accomplished through a frequency control block. It is assumed that the vector captured from the combinational circuit for the previous input vector has an activity

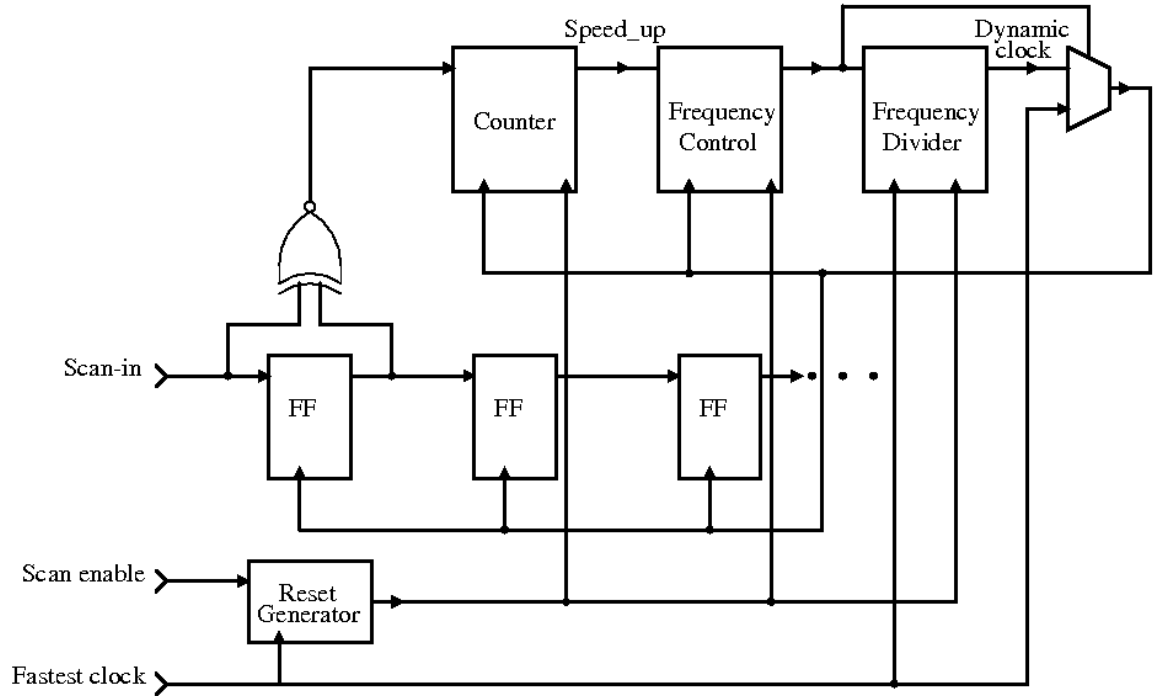


Figure 4.2: Circuitry for Implementation of Dynamic Frequency Control Technique

factor of 1. In other words, it is assumed that the scan chain is filled with alternating 1s and 0s before scan-in begins. There is therefore no need to monitor the last flip-flop in the scan chain. Thus, power dissipated during shift out operation is always assumed to take the highest possible value. This pessimistic assumption is made so that the shift out operation can never cause the power to exceed the limit. Therefore, the scan in of vectors is initially carried out at the slowest frequency, f_{test} permitted by the power budget for $\alpha = 1$. Thus, when the counter has counted up to a certain value, the frequency control block stimulates a frequency divider circuit to generate a clock with appropriate frequency.

The value (number of non-transitions in scan chain) after which speed-up should be initiated can be found through simulation. This number depends on the correlation between circuit activity and scan chain activity. If each transition in the scan chain causes a large number of transitions in the circuit, power consumption reaches large values for low scan chain transition numbers. Thus, a large number of scan chain non-transitions should be encountered by the activity monitor before the scan clock frequency is increased. Similarly,

if a transition in the scan chain causes low activity in the circuit, the number of scan chain transitions required to push power dissipation up to peak power is large. Thus, a few non-transitions in the scan chain are sufficient to increase the frequency of the scan clock.

The reset generator block generates the reset signal for the counter circuit, the frequency control block and the frequency divider circuit, at the positive edge of the scan enable signal, i.e., at the start of scan for every combinational vector. Since the frequency divider cannot generate an $f/1$ signal, a multiplexer is used to choose between the clock generated by the frequency divider and the fastest clock, which is supplied to the circuit.

Let us consider a circuit with 1000 flip-flops. If the scan clock frequency estimated based on the power budget is 80 ns, and if the circuit is to be run at 8 different frequencies, a modulo 125 ($1000/8$) counter is implemented on the circuit. The scan in of vectors is initially carried out at 80 ns. As the input vector is scanned in, the counter counts up every time a non-transition enters the scan chain. When the number of non-transitions reaches 125, the counter is reset and the frequency divider generates a clock of 70 ns, at which the subsequent bits will be scanned in. The counter once again counts up to 125 and the clock period is reduced to 60 ns. This process is repeated until all 1000 bits are scanned in. Thus, if the input were a series of 1000 1s, the first 125 bits are scanned in at a clock period of 80 ns, the second 125 bits at 70 ns, and so on till the last 125 bits are scanned in at a clock period of 10 ns. If the input vector was a series of alternating 0s and 1s, the counter never counts up since there are no non-transitions in the vector, and hence the clock period for the scan in of the entire vector is 80 ns.

Figure 4.3 shows how the dynamic scan clock control technique works for a small circuit with 8 flip-flops. Four frequencies are chosen in this case. The number of non-transitions after which the frequency of the scan clock can be increased is $8/4 = 2$. The clock supplied to the circuit is the fastest clock possible whose frequency = f . The shift process is started with the slowest possible frequency = $f/4$. In cycle 1, the counter is reset to 0. In cycle 2, the counter counts up by 1 due to the incoming non-transition. In cycle 3, the counter retains

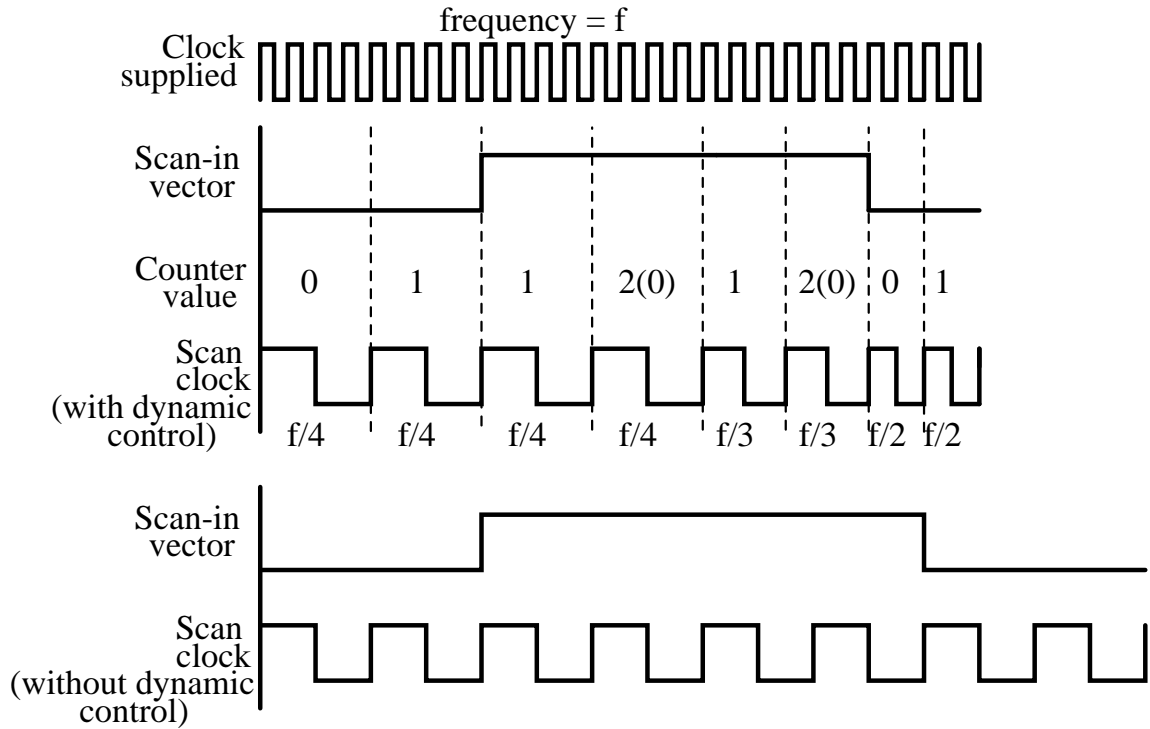


Figure 4.3: Dynamic Scan Clock Control for Circuit with 8 Flip-Flops - Number of Frequencies Chosen = 4

its value since a transition enters the scan chain. In cycle 4, the counter counts up by 1 since a non-transition enters the scan chain. At this point, the value in the counter becomes 2 and hence it is reset. Also, the scan clock frequency is increased to $f/3$ in cycle 5. The value in the counter is increased by 1 in cycles 1 and 2 due to the incoming non-transitions and the counter is reset and the scan clock frequency is increased to $f/2$ in cycle 7. There is no increase in counter value in cycle 7 due to the incoming transition. There is an increase in counter value in cycle 8 because of the incoming non-transition, but the frequency is not increased since the counter value is lesser than 2. If dynamic scan clock technique is not employed, all 8 scan operations take place at a frequency of $f/4$.

Let $T = 1/f$. Total test application time when dynamic scan clock control technique is not used = $4T \cdot 8 = 32T$. Total test application time when dynamic scan clock control technique is used = $4T \cdot 4 + 3T \cdot 2 + 2T \cdot 2 = 26T$. Thus, the reduction in test application time with respect to that without dynamic control = $6/32 = 18.75\%$.

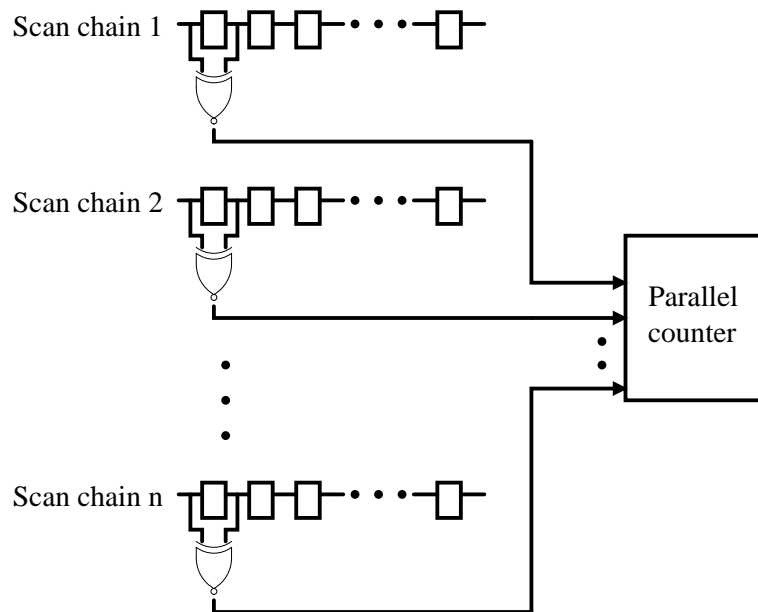


Figure 4.4: Modification in Architecture for Circuits with Multiple Scan Chains

When an input vector with a high number of non-transitions has to be scanned in, more number of frequencies are used for the test than for an input vector with a low number of non-transitions. Thus, an input vector with low activity factor would be scanned in faster than one with high activity factor. Don't cares in deterministic patterns obtained from ATPGs are filled in such a way that the number of transitions in the vector is minimum [50]. Also, techniques to generate BIST patterns with low transition densities [63] are available. This technique would perform well for such patterns.

4.2 Circuits with Multiple Scan Chains Tested with BIST

When the circuit has multiple scan chains, the activity in the circuit depends on the activity caused by the vectors being shifted into every scan chain. Therefore, the activities in all the scan chains have to be monitored. XNOR gates are added across the input and output of the first flip-flop in every scan chain, as shown in Figure 4.4. The outputs of the XNOR gates are fed to a parallel counter [57] that counts up by the number of 1s at its inputs. The rest of the circuitry remains unaltered and still resembles Figure 4.2. Therefore,

when the counter has counted up to a certain threshold value, the frequency is increased and the counter is reset. Thus, the implementation is very similar to that in circuits with single scan chain, except for the counter which is replaced by a parallel counter.

The model proposed in this chapter is based on the assumption that the peak activity factor of the test vectors is 1. While this model does not require any simulations to determine the peak activity factor of the test vectors, it is important to make available a model that can handle circuits whose test vectors have peak activity factors lower than 1. The next chapter introduces such a model.

Chapter 5

Implementation in BIST circuits with Peak Activity Factors Lower than 1

If the peak activity factor (α_{peak}) of the vectors used to test the CUT is not equal to 1, the scan clock frequency chosen can be higher for the same value of peak power limit. This chapter discusses the implementation of dynamic scan clock control technique on such circuits with single and multiple scan chains [55].

5.1 Circuits with Single Scan Chain Tested with BIST

A Test-per-Scan BIST architecture [59] is chosen to illustrate the implementation of dynamic scan clock control in BIST circuits. The scan clock frequency is chosen using Eq. 2.1. The scan clock frequency can be sped up for vectors with low activity factors. In order to do this, the number of transitions in the scan chain is continuously monitored at the input and output of the scan chain.

Figure 5.1 shows the implementation of the technique for BIST circuits with single scan chain and activity factors lesser than 1. The activity monitor comprises of an XNOR gate connected between the input and output of the first flip-flop, and an XNOR gate connected between the input and output of the last flip-flop. The former monitors the number of non-transitions entering the scan chain and the latter monitors the number of non-transitions leaving the scan chain. An up-down counter keeps track of the number of non-transitions in the scan chain. Thus, the former XNOR drives the count_up signal and the latter drives the count_down signal of the up-down counter. The number of non-transitions in the scan chain during any cycle is the difference between that entering the scan chain and that leaving the scan chain.

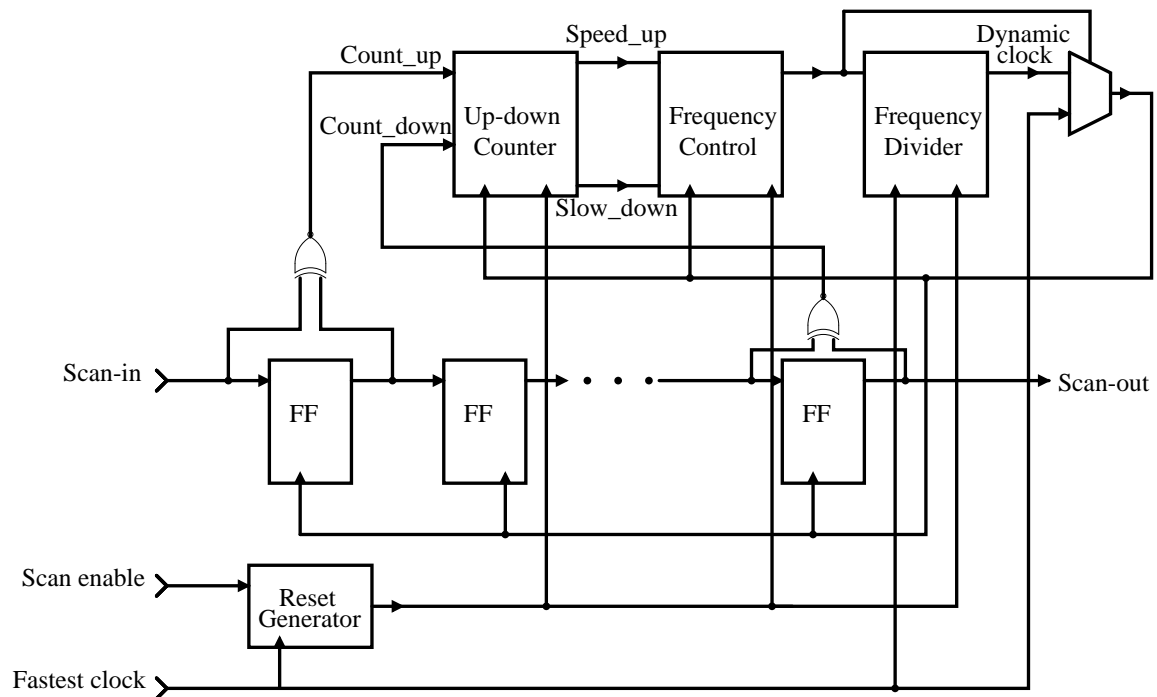


Figure 5.1: Implementation of test-per-scan BIST, activity factor $\neq 1$

Since power is proportional to the activity in the scan chain, test power is lower when the number of transitions in the scan chain is lower or, in other words, when the number of non-transitions in the scan chain is higher. As discussed earlier, from (2.7), the scan frequency can be increased when the number of non-transitions in the scan chain increases.

The up-down counter is reset to 0 at the start of scan-in. The counter value ranges between 0 and a certain threshold value. When there is a non-transition entering the scan chain, the counter counts up and when there is a non-transition leaving the scan chain, the counter counts down. When the counter counts up to the threshold value, the counter signals the frequency control block to increase the frequency of scan clock and the counter is reset to 0. Similarly, when the counter counts down to 0, the frequency control block is signaled to lower the frequency of scan clock and is reset to the threshold value. Thus, whenever the number of non-transitions in the scan chain increases, the frequency is increased and when the number reduces, the frequency is decreased.

The value (number of non-transitions in scan chain) after which speed-up or slow-down should be initiated can be found through simulation. This number depends on the correlation between circuit activity and scan chain activity. If each transition in the scan chain causes a large number of transitions in the circuit, power consumption reaches large values for low scan chain transition numbers. Thus, a large number of scan chain non-transitions should be encountered by the activity monitor before the scan clock frequency is increased. Similarly, if a transition in the scan chain causes low activity in the circuit, the number of scan chain transitions required to push power dissipation up to peak power is large. Thus, a few non-transitions in the scan chain are sufficient to increase frequency of scan clock.

At the start of scan-in of a vector, the frequency control block is reset such that the frequency of scan clock is the slowest possible. This is based on the assumption that the activity factor of the vector captured in the scan chain before the start of scan-in equals α_{peak} . When the frequency of scan clock is the slowest possible frequency, the scan clock frequency is not decreased any further irrespective of the signal from the up-down counter. Similarly, when the frequency of scan clock is at the highest possible frequency, the scan clock frequency is not increased any further regardless of the signal from the counter.

The reset generator block generates the reset signal for the counter circuit, the frequency control block and the frequency divider circuit, at the positive edge of the scan enable signal, i.e., at the start of scan for every combinational vector. Since the frequency divider cannot generate an $f/1$ signal, a multiplexer is used to choose between the clock generated by the frequency divider and the fastest clock, which is supplied to the circuit.

Maximal reduction in scan-in time per vector is achieved when the captured vector has high activity and the scan-in vector has low activity. This ensures that the `count_down` signal to the counter is hardly 1 and the `count_up` signal to the counter is almost always 1.

It can be observed that the implementation proposed in this chapter can be easily modified for circuits with activity factors equal to 1, by removing the flip-flop at the end of the scan chain and tying the `count_down` signal of the up-down counter to 0.

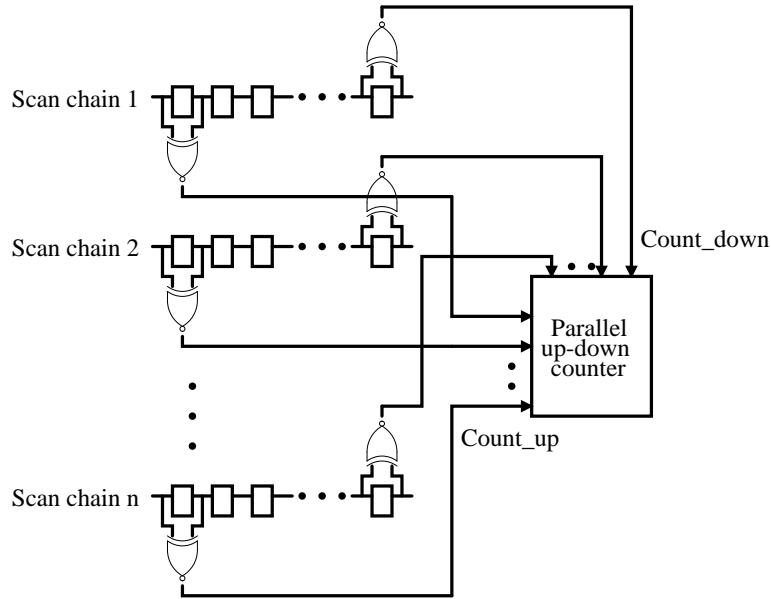


Figure 5.2: Modification in Architecture for Circuits with Multiple Scan Chains, activity factor $\neq 1$

5.2 Circuits with Multiple Scan Chains Tested with BIST

When the circuit has multiple scan chains, the activity in the circuit depends on the activity caused by the vectors being shifted into every scan chain. Therefore, the activities in all the scan chains have to be monitored. XNOR gates are added across the input and output of the first flip-flop and across the input and output of the last flip-flop in every scan chain, as shown in Figure 5.2. The outputs of the XNOR gates at the inputs of the scan chains are fed to the count_up inputs of a parallel counter [57] which counts up by the number of 1s at its count_up inputs. Similarly, the outputs of the XNOR gates at the end of the scan chains are fed to the count_down inputs of the parallel counter [57] which counts down by the number of 1s at its count_down inputs. The rest of the circuitry remains unaltered and still resembles Figure 5.1. Therefore, when the counter has counted up to a certain threshold value, the frequency is increased and the counter is reset to 0 and when the counter has counted down to 0, the frequency is decreased and the counter is reset to the threshold value. Thus, the implementation is very similar to that in circuits with single scan chain, except for the counter which is replaced by a parallel counter.

Chapter 6

Background on Communication between Asynchronous Systems

This chapter is a study of different asynchronous protocols available for communication between two digital systems working at different clock domains. These protocols can be used for communication between ATE and DUT when dynamic scan clock frequency control is implemented.

Two digital systems working at different clock domains require a protocol to communicate with each other in order to ensure validity of the data being shared between the two systems. Such systems are said to be asynchronous, and the protocol used for communication is said to synchronize the two systems. Asynchronous design techniques are not as widely used as synchronous design techniques are. This chapter presents some of the commonly used asynchronous handshake protocols, and how they are implemented. Some common mistakes made in asynchronous design are also discussed.

Synchronization is the process of enforcing an ordering of events on signals. For example, synchronization is required while sampling an asynchronous signal with a clock or when a signal that is synchronous to one clock is sampled by another clock. The difficulty of synchronizing a signal with a clock depends on the predictability of events on the signal relative to the clock. When a signal is to be synchronized to a clock with the same frequency as the sample clock but with an arbitrary phase, the signal and clock are said to be mesochronous [24]. In such systems, a single phase measurement suffices to predict possible transition times arbitrarily far in the future. If the signal is generated by a clock with a slightly different frequency, there is a slow drift in the phase, and the signal and clock are said to be plesiochronous [24].

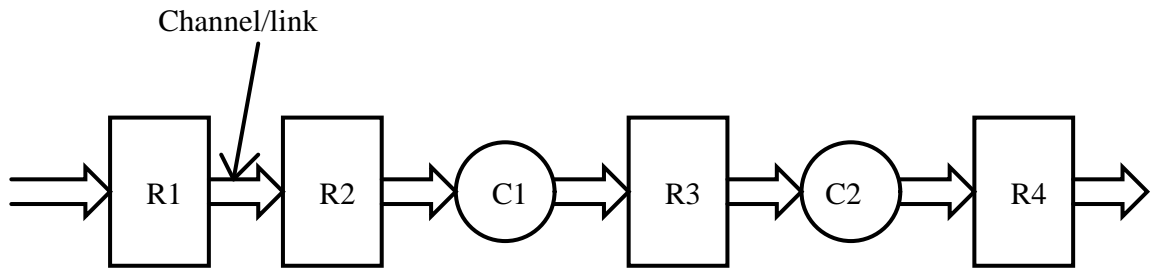


Figure 6.1: An Abstract Data-Flow View of the Asynchronous Circuit

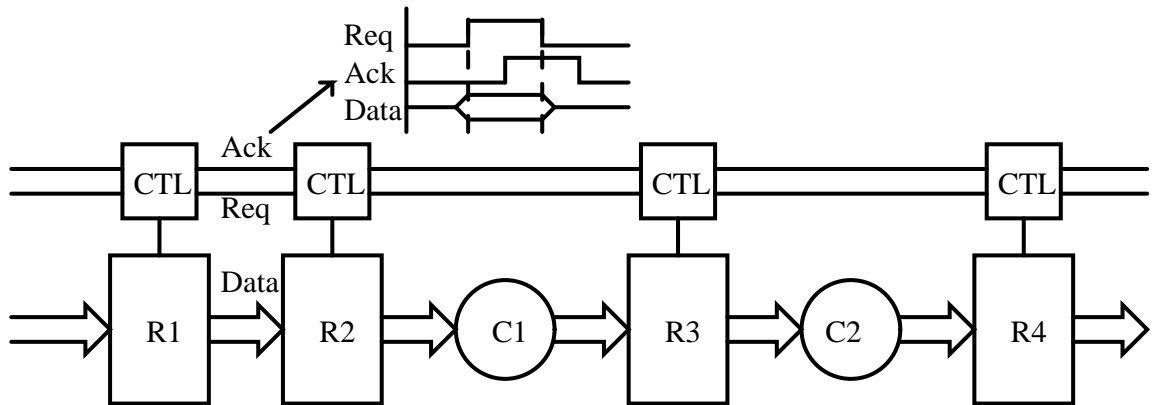


Figure 6.2: A Request-Acknowledge Based Hand-Shake

Figure 6.1 is an abstract representation of an asynchronous system. R1, R2, R3, R4 represent registers and C1, C2 represent combinational circuitry. In an asynchronous circuit, the clock signal is replaced by some form of handshaking between neighboring registers. A simple request-acknowledge based handshake protocol is shown in Figure 6.2. The data and handshake signals connecting one register to the next can be thought of as a handshake channel or link. The data stored in the registers can be thought of as tokens tagged with data values that may be changed along the way as tokens flow through combinational circuits. The combinational circuitry can be thought of as being transparent to the handshaking between registers. A combinational circuitry simply absorbs a token on each of its input links, performs its computation, and then emits a token on each of its output links.

Correct operation requires that data tokens flowing in the circuit do not disappear, that one token does not overtake another, and that new tokens do not appear out of nowhere.

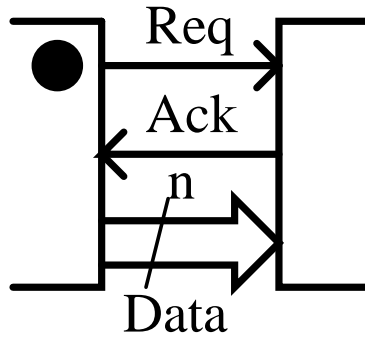


Figure 6.3: A Bundled-Data Channel

6.1 Handshake Protocols

6.1.1 Bundled-data protocols

Bundled-data protocols comprise data signals that use normal Boolean levels to encode information, and have separate request and acknowledge wires bundled with the data signals, as shown in Figure 6.3. The bundled-data protocol is also known as the single-rail protocol.

4-Phase Bundled-Data Protocol

In this type of protocol, the request and acknowledge wires also use normal Boolean levels to encode information. The term 4-phase refers to the communication actions:

- the sender issues data and sets request high,
- the receiver absorbs the data and sets acknowledge high,
- the sender responds by taking request low, at which point the data is no longer guaranteed to be valid, and
- the receiver acknowledges this by taking acknowledge low. At this point the sender may initiate the next communication cycle.

Figure 6.4 illustrates the communication actions explained above.

This protocol is familiar to most digital designers, but it has a disadvantage in the superfluous return-to-zero transitions that cost unnecessary time and energy.

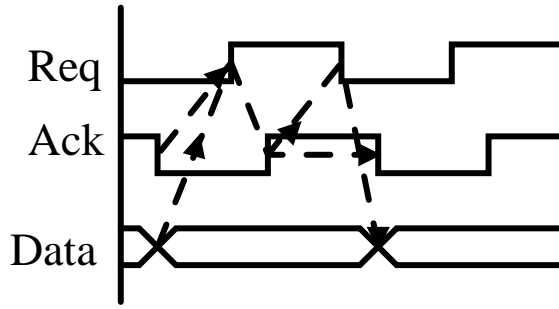


Figure 6.4: A 4-Phase Bundled-Data Protocol

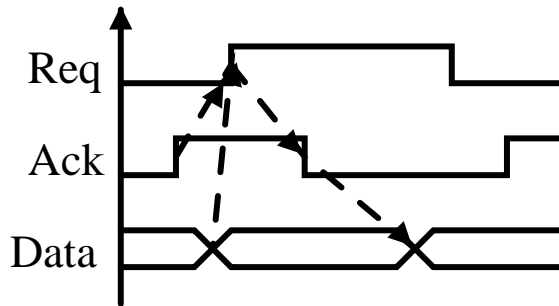


Figure 6.5: A 2-Phase Bundled-Data Protocol

2-Phase Bundled-Data Protocol

In the 2-phase bundled-data protocol, the information on the request and acknowledge wires is encoded as signal transitions on the wires and there is no difference between a 0 to 1 and a 1 to 0 transition, they both represent a single event, as shown in Figure 6.5.

6.1.2 Dual-Rail Protocol

The request signal is encoded into the data signals using two wires per bit of information that has to be communicated. In other words, the dual rail protocol uses two request wires per bit of information d ; one wire $d.t$ is used for signaling a logic 1 (or true), and another wire $d.f$ is used for signaling a logic 0 (or false). This protocol is very robust since two components can communicate reliably regardless of delays in the wires connecting the two components, or in other words, the protocol is delay-insensitive.

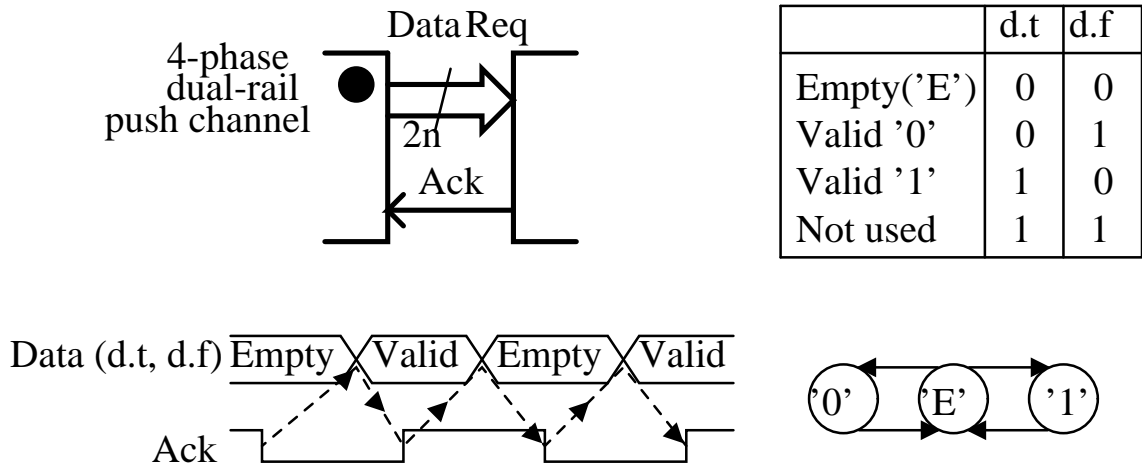


Figure 6.6: A Delay-Insensitive Channel using the 4-Phase Dual-Rail Protocol

4-Phase Dual-Rail Protocol

The information is encoded as follows: $\{x.f,x.t\} = \{1,0\}$ for a logic 0 and $\{x.f,x.t\} = \{0,1\}$ for a logic 1 represent valid data and $\{x.f,x.t\} = \{0,0\}$ represents no data. The codeword $\{x.f,x.t\} = \{1,1\}$ is not used, and a transition from one valid codeword to another valid codeword is not allowed.

An abstract view of 4-phase dual-rail handshaking, as shown in Figure 6.6, can be explained as:

- the sender issues a valid codeword,
- the receiver absorbs the codeword and sets acknowledge high,
- the sender responds by issuing the empty codeword, and
- the receiver acknowledges this by taking acknowledge low. At this point the sender may initiate the next communication cycle.

A more abstract view of what is seen on a channel is a data stream of valid codewords separated by empty codewords.

The above protocol can be extended to an N -bit data channel. The codewords for an N -bit data channel can be divided into

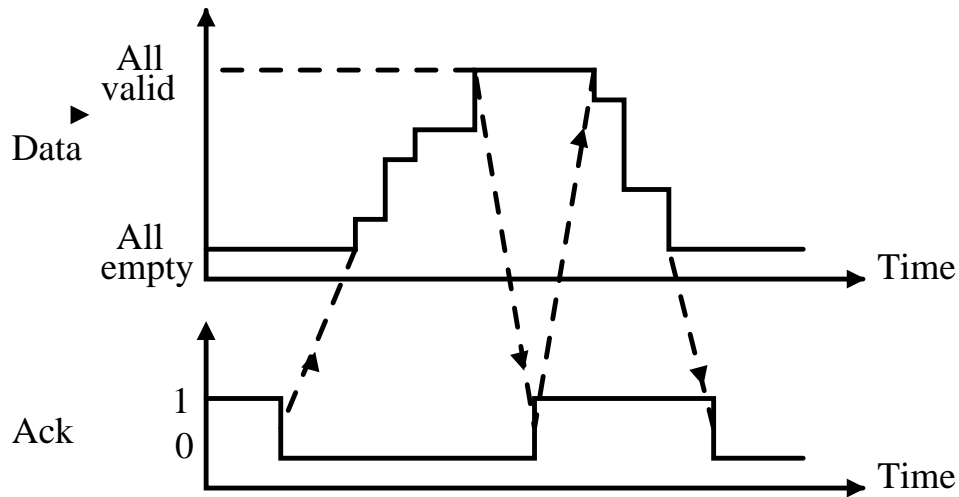


Figure 6.7: Handshaking on a 4-Phase Dual-Rail Channel

- the empty codeword where all N wire pairs are $\{0,0\}$
- the intermediate codewords where some wire-pairs assume the empty state and some wire pairs assume valid data
- the $2N$ different valid codewords

Figure 6.7 illustrates the handshaking on an N -bit channel: a receiver will see the empty codeword, a sequence of intermediate codewords, and eventually a valid codeword. After receiving and acknowledging the codeword, the receiver will see a sequence of intermediate codewords, and eventually the empty codeword to which the receiver responds by driving acknowledge low again.

2-Phase Dual-Rail Protocol

The 2-phase dual-rail protocol also uses two wires per bit, but the information is encoded as transitions. On an N -bit channel a new codeword is received when exactly one wire in each of the N wire pairs has made a transition. There is no empty value; a valid message is acknowledged and followed by another message that is acknowledged. Figure 6.8 shows a 2-phase dual-rail protocol.

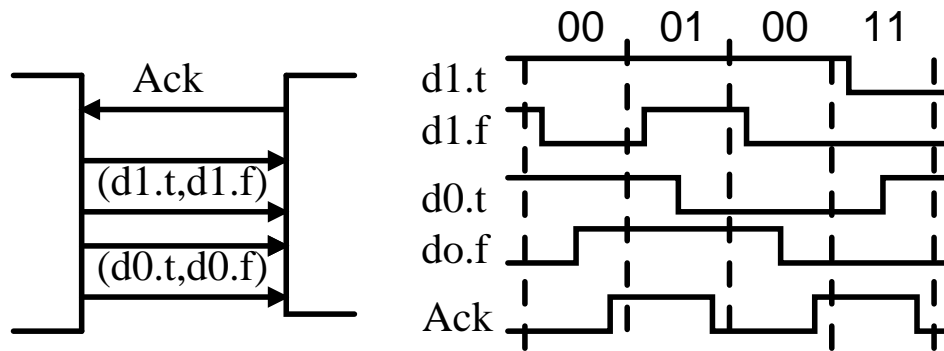


Figure 6.8: Handshaking on a 2-Phase Dual-Rail Channel

6.1.3 Other Protocols

The previous sections introduced the four most common channel protocols, but there are other possibilities as well. The two wires per bit used in the dual-rail protocol can be seen as a one-hot encoding of that bit and often it is useful to extend to 1-to- n encodings in control logic and higher-radix data encodings.

If the focus is on communication rather than computation, m -of- n encodings may be of relevance. The solution space can be expressed as the cross product of a number of options including: {2-phase, 4-phase} X {bundled-data, dual-rail, 1-of- n , ...} X {push, pull}

The choice of protocol affects the circuit implementation characteristics such as area, speed, power, robustness, etc.

6.2 The Muller C-element and The Indication Principle

In asynchronous circuits, signals are required to be valid all the time, every signal transition has a meaning and, consequently, hazards and races must be avoided. The concept of indication or acknowledgement plays an important role in the design of such circuits.

For instance, consider a 2-input OR gate. An output change from 1 to 0 leads to a conclusion that both inputs are now at 0. However, an output change from 0 to 1 indicates that at least one input is 1, but does not indicate which. In other words, the OR gate only

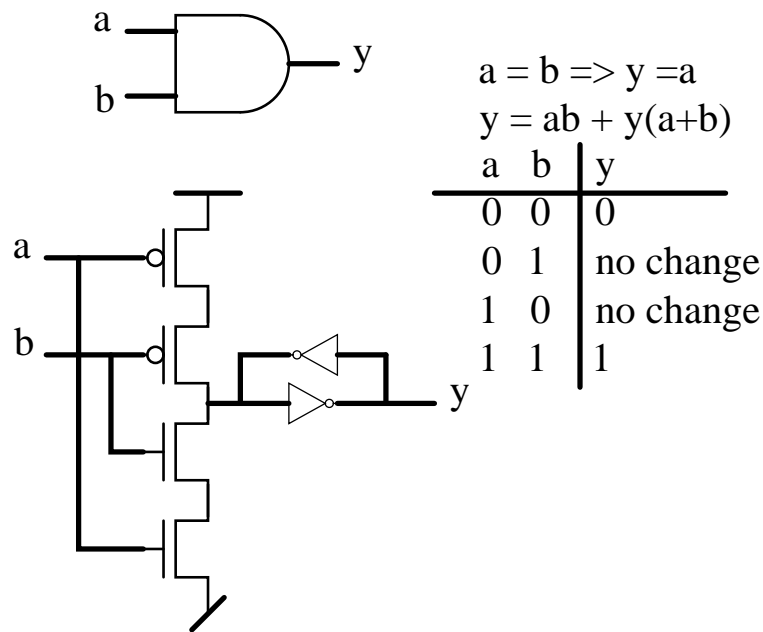


Figure 6.9: The Muller-C Element

indicates or acknowledges when both inputs are 0. Similarly an AND gate indicates only when both inputs are 1.

Signal transitions that are not indicated or acknowledged in other signal transitions are the source of hazards and should be avoided.

A circuit that is better in this respect is the Muller C-element shown in Figure 6.9. It is a state-holding element much like an asynchronous set-reset latch. When both inputs are 0 the output is set to 0, and when both inputs are 1 the output is set to 1. For other input combinations the output does not change. Hence, an output change from 0 to 1 indicates that both inputs are now at 1; and similarly, an output change from 1 to 0 indicates that both inputs are now at 0. The Muller C-element is a fundamental component used extensively in asynchronous circuits because of this property.

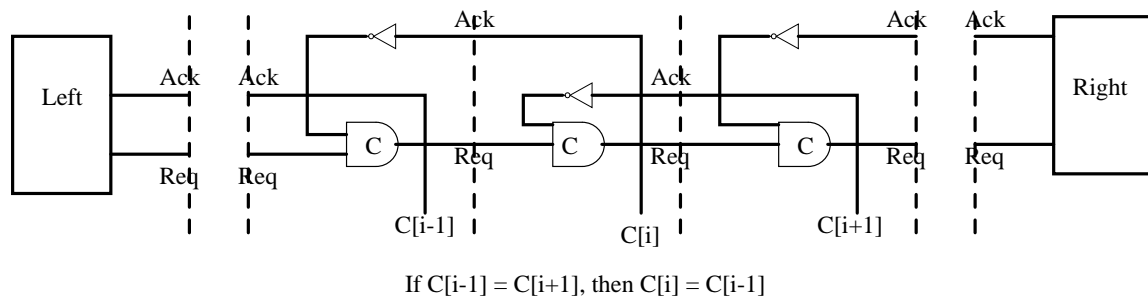


Figure 6.10: The Muller Pipeline

6.3 The Muller Pipeline

Figure 6.10 shows a circuit built from C-elements and inverters. The circuit is known as a Muller pipeline or a Muller distributor. Variations and extensions of this circuit form the backbone of almost all asynchronous circuits.

The Muller pipeline is a mechanism that relays handshakes. After all the C-elements have been initialized to 0 the left environment may start handshaking. The i th C-element $C[i]$ will propagate a 1 from its predecessor, $C[i-1]$, only if its successor, $C[i+1]$, is 0. Similarly, it will propagate a 0 from its predecessor only if its successor is 1.

On any interface between C-element pipeline stages, correct handshaking will be observed, but the timing may differ from the timing of the handshaking on the left hand environment. Eventually the first handshake (request) injected by the left hand environment will reach the right hand environment. If the right hand environment does not respond to the handshake, the pipeline will eventually fill. If this happens the pipeline will stop handshaking with the left hand environment – the Muller pipeline behaves like a ripple through first-in first-out (FIFO).

The implementation of the Muller pipeline is the same for both 2-phase and 4-phase handshaking. The difference lies in the way the signals are interpreted and in the way the circuit is used. Also, the circuit operates equally well from right to left. If the definition of signal polarities is reversed and if the role of the request and acknowledge signals are reversed,

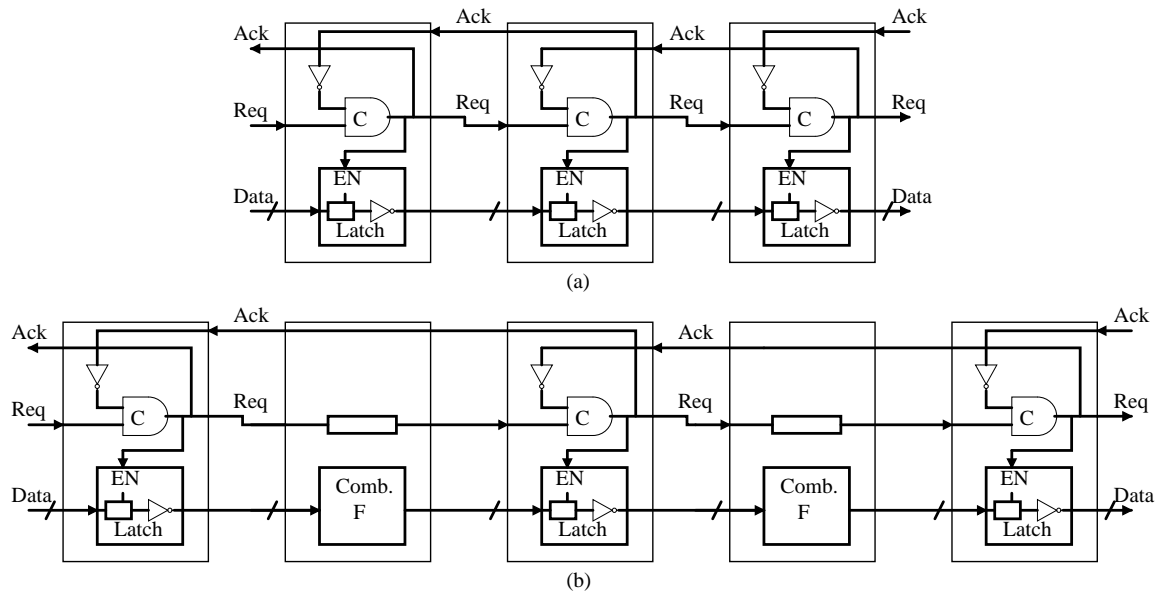


Figure 6.11: A 4-Phase Bundled-Data Pipeline

the circuit can be operated from right to left. The circuit works correctly regardless of delays in gates and wires – the Muller-pipeline is delay-insensitive.

6.4 Circuit Implementation

The choice of handshake protocol affects the circuit implementation in terms of area, speed, power, robustness, etc. Most practical circuits use one of the following protocols: 4-phase bundled-data, 2-phase bundled-data or 4-phase dual-rail. The circuit implementations of these protocols use variations of the Muller pipeline for controlling the storage elements.

6.4.1 4-Phase Bundled-Data

A Muller pipeline is used to generate local clock pulses. The clock pulse generated in one stage overlaps with the pulses generated in the neighboring stages in a carefully controlled interlocked manner. Figure 6.11(a) shows a FIFO, i.e., a pipeline without data processing, and Figure 6.11(b) shows how combinational circuits (also called functional blocks) can be added between the latches. To maintain correct behavior matching delays have to be inserted in the request signal paths.

This circuit may be viewed as a traditional synchronous data-path, consisting of latches and combinational circuits that are clocked by a distributed gated-clock driver or as an asynchronous data-flow structure composed of two types of handshake components: latches and functional blocks.

The pipeline implementation shown in Figure 6.11 is simple but it has some drawbacks: when it fills, the state of the C-elements is (0, 1, 0, 1, etc.), and hence, only every other latch is storing data. This is not any worse than in a synchronous circuit using master-slave flip-flops, but it is possible to design asynchronous pipelines and FIFOs that are better in this respect. Also, the throughput of a pipeline or FIFO depends on the time it takes to complete a handshake cycle and for this implementation, this involves communication with both neighbors. This leads to a lower speed.

6.4.2 2-Phase Bundled-Data (Micropipelines)

A 2-phase bundled-data pipeline also uses a Muller pipeline as the backbone control circuit, but the control signals are interpreted as events or transitions. Therefore, special capture-pass latches are needed: events on the C and P inputs alternate, causing the latch to alternate between capture mode and pass mode. This requires a special latch design as shown in Figure 6.12. The switch symbol in Figure 6.12 is a multiplexer, and the event controlled latch can be viewed as two ordinary level sensitive latches operating in an alternating fashion followed by a multiplexer and a buffer.

Figure 6.13 shows a pipeline without data processing. Combinational circuits with matching delay elements can be inserted between latches in a way similar to the 4-phase bundled-data approach.

The 2-phase bundled-data approach is elegant and efficient compared to the 4-phase bundled-data approach that incurs unnecessary power and performance loss due to the return-to-zero part of the handshaking. However, the implementation of components that

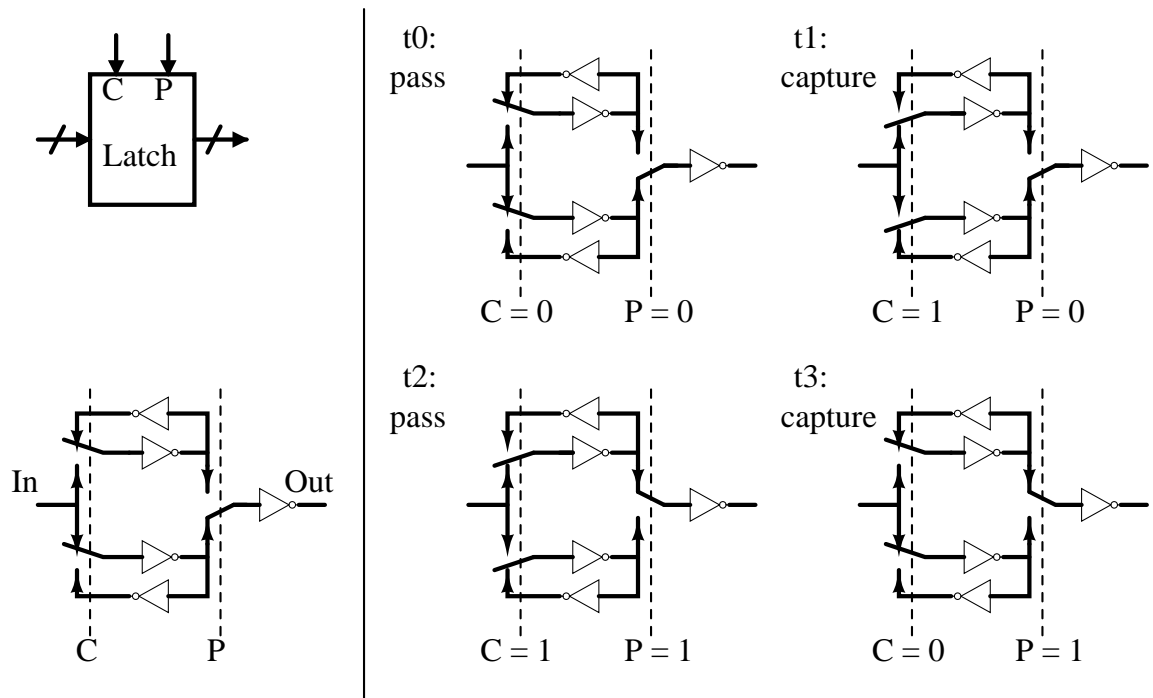


Figure 6.12: Latch in 2-Phase Bundled-Data Pipeline

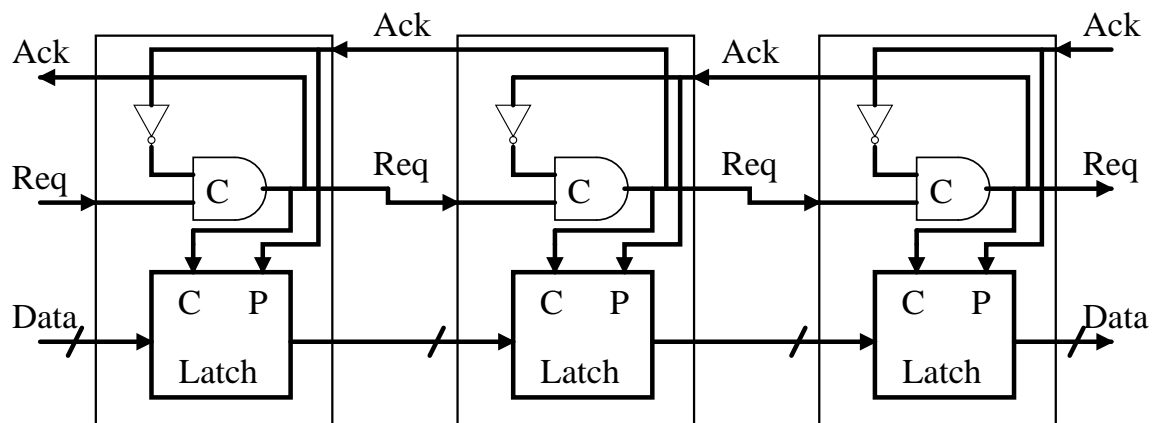


Figure 6.13: A 2-Phase Bundled-Data Pipeline

respond to signal transitions is often more complex than the implementation of components that respond to normal level signals.

6.4.3 4-Phase Dual-Rail

A 4-phase dual-rail pipeline is also based on the Muller pipeline, but in a more elaborate way that has to do with the combined encoding of data and request. Figure 6.14 shows the

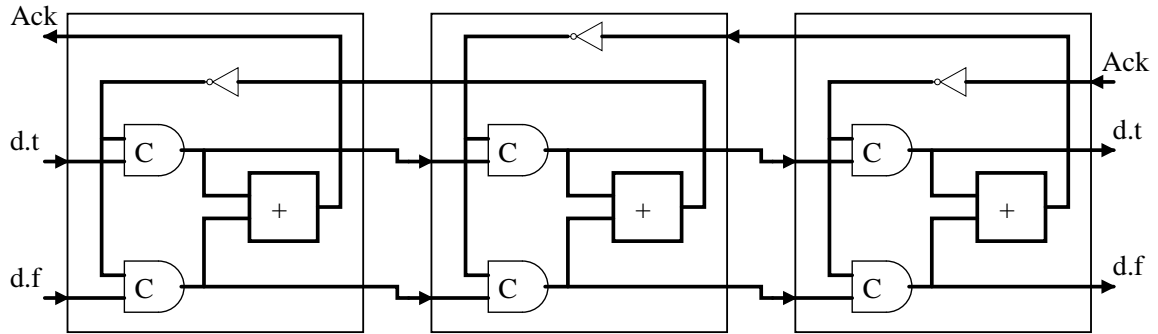


Figure 6.14: A Simple 3-Stage 1-Bit Wide 4-Phase Dual-Rail Pipeline

implementation of a 1-bit wide and three stage deep pipeline without data processing. It can be viewed as two Muller pipelines connected in parallel, using a common acknowledge signal per stage to synchronize operation. The pair of C-elements in a pipeline stage can store the empty codeword $\{d.t, d.f\} = \{0, 0\}$, causing the acknowledge signal out of that stage to be 0, or it can store one of the two valid codewords $\{0,1\}$ and $\{1, 0\}$, causing the acknowledge signal out of that stage to be logic 1. The codeword $\{1, 1\}$ is illegal and does not occur. The acknowledge signal generated by the OR gate safely indicates the state of the pipeline stage as being valid or empty.

An N -bit wide pipeline can be implemented by using a number of 1-bit pipelines in parallel. This does not guarantee to a receiver that all bits in a word arrive at the same time, but often the necessary synchronization is done in the function blocks.

6.5 Scan Chain with Mixed Edge-Triggered Flip-Flops

Scan testing is a widely used technique in the test of sequential circuits. In this technique, the memory elements in the circuit are linked together to act as shift registers during testing and to function as memory elements during normal operation. This is done to increase the controllability and observability of the circuit during testing.

Clock skew between successive scan-storage cells must be less than the propagation delay between the scan output of the first storage cell and the scan input of the next storage cell. Otherwise, data that latches into the first scan cell also latches into the second scan

cell. This results in an error because the second scan cell should latch the first scan cell's old data rather than its new data.

Thus, when a circuit consists of both positive and negative edge-triggered flip-flops, timing problems occur. In this case, all the negative edge-triggered flip-flops are linked together to form one scan chain and likewise all the positive edge triggered flip-flops are linked together to form a second scan chain. These two scan chain are linked together to form a single scan chain through a lockup latch.

Lockup latches [39] are nothing more than transparent latches. They are used to connect two scan-storage elements in a scan chain in which excessive clock skew exists. Figure 6.15 illustrates the use of lockup latches. The circuit contains two flip-flops. Flip-flop 1 represents the end of the scan chain that consists of negative edge-triggered flip-flops. Flip-flop 2 represents the beginning of the scan chain that contains positive edge-triggered flip-flops. The latch has an active-high enable, which becomes transparent only when the clock to the negative-edge triggered flip-flops goes high and effectively adds a half clock of hold time to the output of flip-flop 1. Thus, the two scan chains are synchronized by the addition of a simple latch.

6.6 Common Mistakes Made During Synchronization

This section reviews some common causes of errors in circuits employing more than one time domain [27].

6.6.1 Avoiding the Synchronizer

The most common synchronization error is the transfer of a signal from one clock domain into another without any synchronization since in some cases, the designer might feel that failure probability is too low to worry over. Also, in some cases, if the receiver operates at a much higher clock frequency than the sender, the receiver is expected to always be fast enough to catch the signal and hence synchronization seems unnecessary.

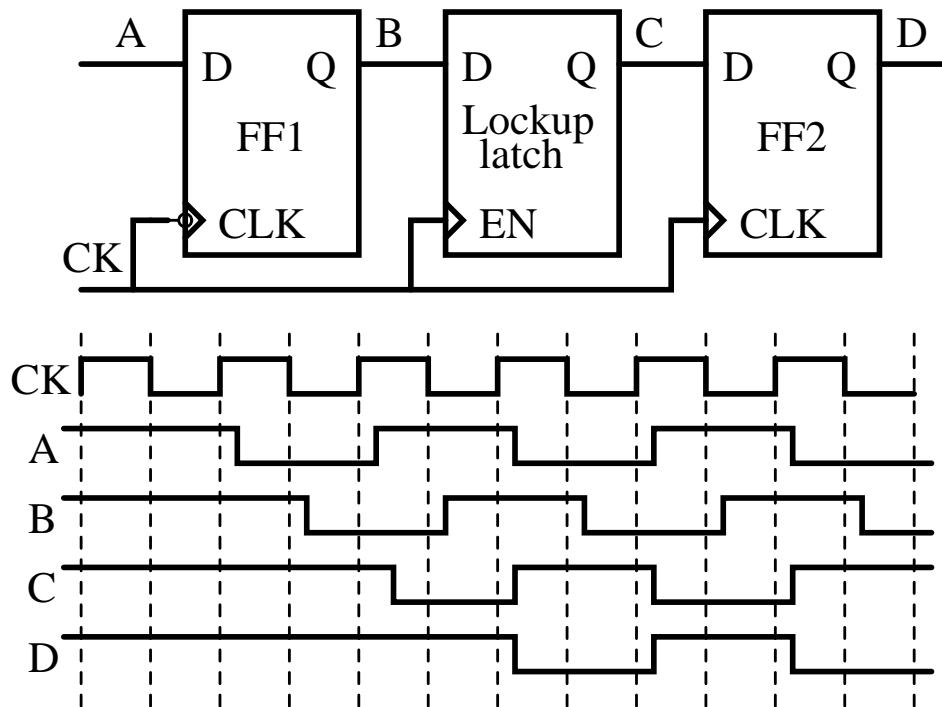


Figure 6.15: Lockup Latch in a Scan Chain

If the incoming data is used as a combinational input to a combinational circuit, which eventually feeds into a flip-flop, there is no way to guarantee the timing of the output of the combinational circuit since the timing of the input is unknown. In particular, it may change simultaneously with the sampling edge of the clock, and the receiving flip-flop may enter metastability or take excessively long to respond, hampering correct operation of the next stage of logic. This error can sometimes evade detection by normal logic validation tools.

6.6.2 Sneaky Path

Occasionally, a signal sneaks through a clock domain boundary unintentionally and unsynchronized. For instance, a signal is sometimes moved from one clock domain to another as part of redesign, and some uses of the signal in its old domain are overlooked.

6.6.3 Wrong Protocol

Consider the following example. The sender is a CPU that can be tuned to operate in the range of 60-100 MHz. The receiver is a communication modem based on a 55 MHz clock. A 2-phase bundled-data protocol is used to transfer data from the CPU to the modem. It is found that it would take four cycles of the receiver's clock to latch the data. Based on the relative speeds, this would mean up to eight cycles of the faster sender's clock. To save time and logic, the designer eliminates the synchronizer and inserts a nine-cycle delay in the sender's finite state machine (FSM).

There would be two problems with this design. First, the safety requirement of the protocol that transitions must be acknowledged is violated. Although the data would be safely latched, at times the receiver might be busy doing something else and would not manage to make use of the data before a new set of data arrives, over-writing the old.

Second, while the modem would remain at 55 MHz, if the CPU were to be sped up in a later chip generation, the sender's nine clock cycles will not be sufficient to cover the four modem cycles anymore.

6.6.4 Global Reset

In a multi-frequency GALS (Globally Asynchronous, Locally Synchronous) SOC, a global reset signal is naturally asynchronous to at least some of the clock domains. The leading edge of the reset signal is harmless, as it forces all circuits to a known starting state. The trailing edge, on the other hand, could cause some damage. During global reset, the various clocks are started and all PLLs settle into their respective different frequencies. When the reset is removed, it can happen simultaneously with the sampling edge of one of the clocks. The global reset is typically connected into the asynchronous clear input of many flip-flops, and its trailing edge must respect a setup constraint, or else the flip-flops may enter metastability. This is true for other asynchronous signals such as the asynchronous clear or preset of flip-flops as well.

6.6.5 DFT Leakage

Simple production testers may have only a single clock. To test a GALS SOC on such testers, all clocks are shorted together. Static faults and some dynamic faults are properly tested that way. The clock shorts of course must be ignored during path analysis, but certain changes of the design may result in an error (sneaky) path masked by the list.

Chapter 7

Implementation in Externally Tested Circuits with Peak Activity Factors of 1

This chapter discusses the features of the circuitry used to implement dynamic scan clock control in circuits externally tested with ATEs [54]. This chapter deals with circuits that have a peak activity factor of 1. This information is used to compute the peak power and scan clock frequency.

7.1 Circuits with Single Scan Chain Tested with ATE

In the case of circuits tested with ATE, a power analysis can be performed for every vector through simulation of the vectors and this information can be used to scan in vectors at appropriate frequencies that would reduce test time without exceeding the power budget of the chip. The frequency of scan clock at the start of scan-in is computed based on the activity factor of the vector captured in the scan chain prior to scan-in. The subsequent scan clock frequencies are changed in steps based on the activity in the scan chain during every clock cycle. The implementation is shown in Figure 7.1. A large amount of information would have to be stored in the ATE to make such an implementation possible. When tests are constrained by ATE memory capacity, a more conservative method can be used.

The implementation of dynamic control of scan frequency in circuits tested by ATEs can be similar to that in circuits using BIST. However, when BIST is used for testing, patterns are generated on-chip and can hence be generated at the same frequency as that of the dynamic clock (since the LFSR can be driven by the dynamic clock). While using ATE, it is important to ensure that the patterns are scanned in and scanned out at the dynamic scan clock frequency. The information about the rate of scan in and scan out can be included in the test program comprising the test vectors, but this amounts to utilizing

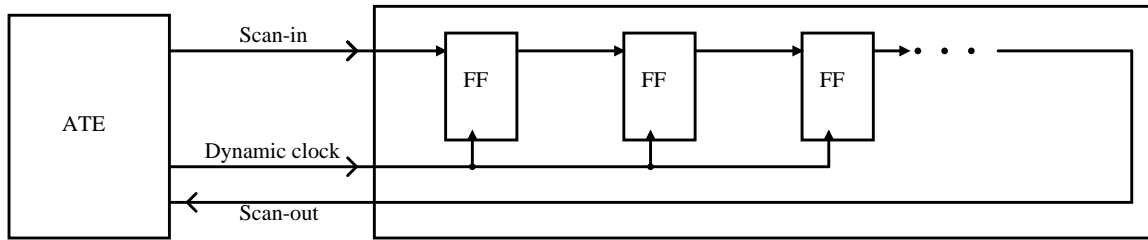


Figure 7.1: Implementation of Dynamic Control of Scan Clock Frequency in Externally Tested Circuits

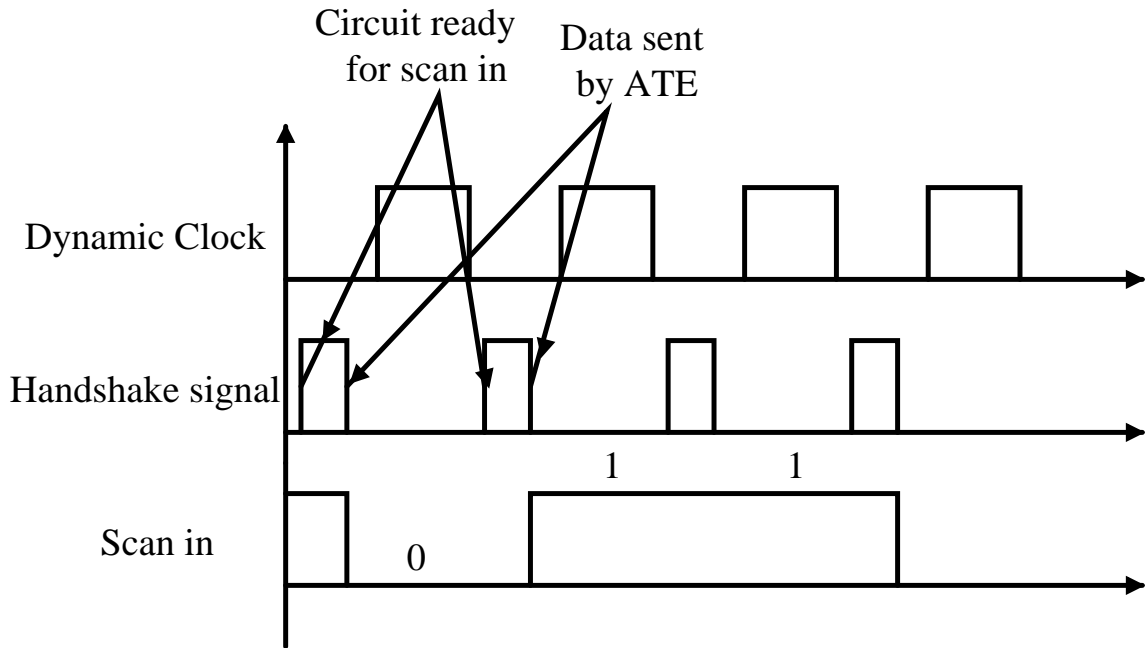


Figure 7.2: Handshake Protocol between ATE and chip

extra ATE memory. This problem is however no different from communication between any two systems operating at different clock frequencies. It can be solved by using any of the handshake protocols [24] used in communication between asynchronous digital systems.

A simple handshake protocol is illustrated in Figure 7.2. When the circuit is ready to scan in data, a synchronizer, which can either reside on the chip or on the tester head, toggles the handshake signal. The ATE acknowledges this by scanning in the next bit into the scan-in pin, scanning out the next bit from the scan-out pin and toggling the handshake signal. The synchronizer recognizes the toggle in the handshake signal and accepts the new scan-in bit. The layout for this implementation is shown in Figure 7.3

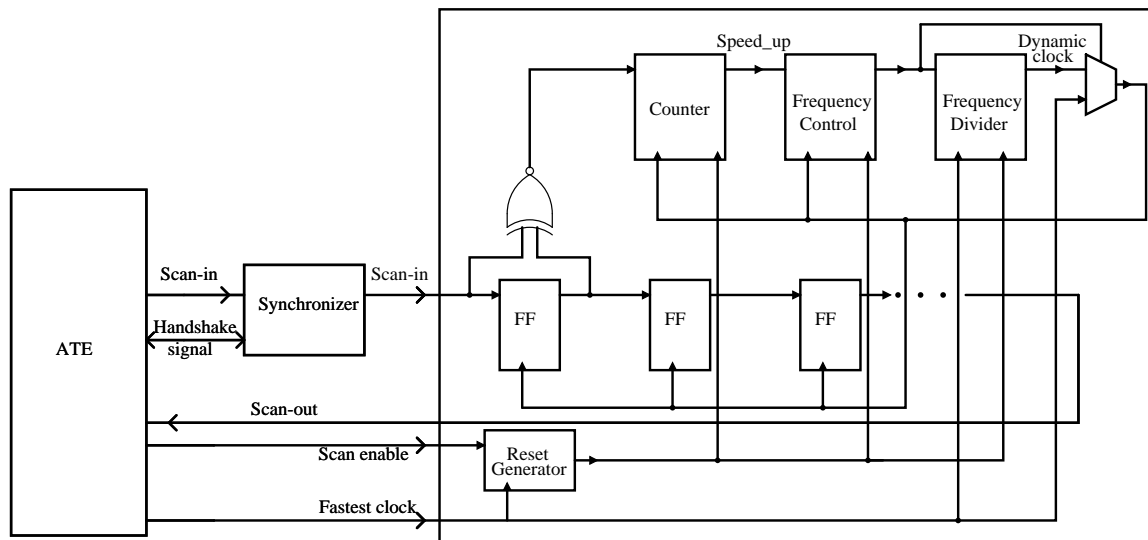


Figure 7.3: Implementation in Externally Tested Circuits with Peak Activity Factors of 1

The use of ATE can reduce the hardware overhead required to implement dynamic control of scan clock frequency. The activity monitor, frequency control block, reset generator and synchronizer can be implemented either on-chip or off-chip on the ATE performance board.

7.2 Circuits with Multiple Scan Chains Tested with ATE

For circuits with multiple scan chains, an XNOR gate is added at the front end of every scan chain as shown in Figure 4.4. A parallel counter is used to monitor the activity and trigger the frequency control block once the activity threshold is reached. The use of compression does not affect the implementation since activity is monitored in every scan chain.

Chapter 8

Implementation in Externally Tested Circuits with Peak Activity Factors Lower than 1

This chapter describes the implementation of dynamic scan clock control in externally tested circuits with peak activity factors (α_{peak}) lower than 1. The scan clock frequency is computed for a known value of power limit using Eq. 2.1.

8.1 Circuits with Single Scan Chain Tested with ATE

As discussed earlier, a power analysis can be performed for every vector and this information can be used to scan in vectors at appropriate frequencies such that the test time can be reduced without exceeding the power budget of the chip (Figure 7.1). If the test program size for such an implementation is too large, the implementation used for BIST circuits can be used with some modifications. It was discussed that the implementation used for BIST circuits cannot be directly used in externally tested circuits since the test vector is supplied by the ATE while the dynamic frequency control is carried out on-chip and hence an asynchronous protocol is required for communication between the two. A simple handshake protocol was illustrated in the previous chapter.

Figure 8.1 shows the circuitry for implementation of dynamic frequency control of scan clock for externally tested circuits with peak activity factors lower than 1. The approach is similar to that shown in Figure 5.1 with the additional use of a synchronizer for the purpose of communication between ATE and DUT.

The XNOR gate at the first flip-flop of the scan chain monitors the number of non-transitions entering the scan chain and drives the count_up signal of the up-down counter. Thus, the counter counts up every time a non-transition enters the scan chain. Similarly, the XNOR gate at the last flip-flop of the scan chain monitors the number of non-transitions

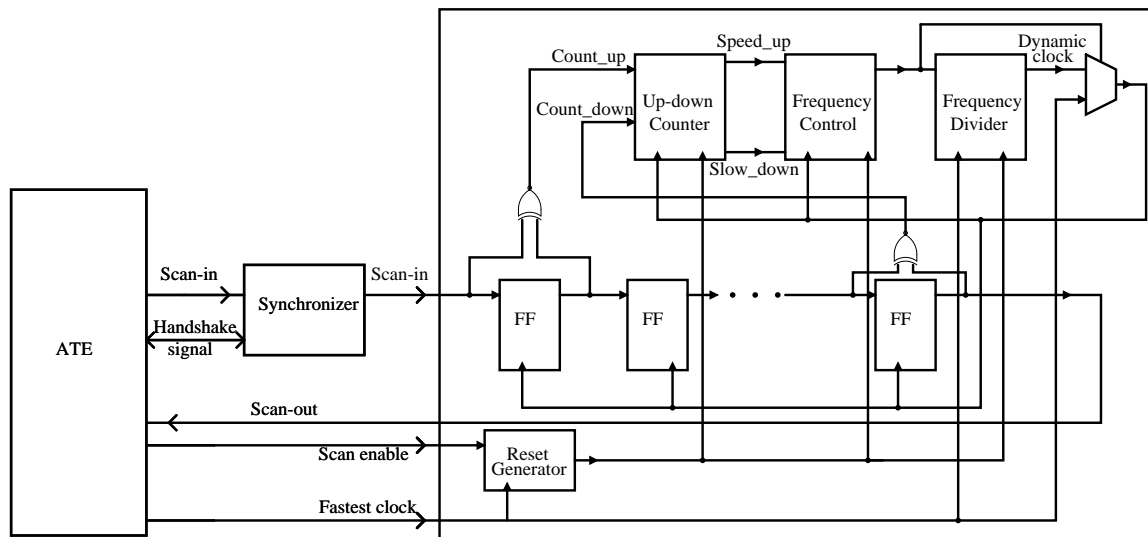


Figure 8.1: Implementation in Externally Tested Circuits with activity factors lower than 1 leaving the scan chain and drives the count_down signal of the up-down counter. Thus, the counter counts down every time a non-transition leaves the scan chain.

As explained in Chapter 5, the frequency of scan clock is increased or decreased based on the value in the counter. The activity factor of the vector captured in the scan chain is assumed to be α_{peak} . The scan-in of vectors is therefore started at the lowest possible frequency. When the counter counts up to a certain threshold value, the frequency is increased since an increase in the counter value indicates an increase in the number of non-transitions in the scan chain. The counter is reset to 0 after this. Similarly, once the counter counts down to 0, the frequency of scan clock is decreased and the counter is reset to the threshold value. Thus, scan clock frequency control is achieved based on the number of non-transitions in the scan chain.

The synchronizer controls the data flow to and from the ATE. The handshake signal is used to determine whether the ATE or DUT is ready for data exchange, as discussed in the previous chapter. When the DUT is ready, the ATE sends scan-in bits and gets ready to receive scan-out bits. The DUT acknowledges this by accepting the scan-in bits and sending scan-out bits.

The use of ATE can reduce the hardware overhead required to implement dynamic control of scan clock frequency. The activity monitor, frequency control block, reset generator and synchronizer can be implemented either on-chip or off-chip on the ATE performance board.

8.2 Circuits with Multiple Scan Chains Tested with ATE

For circuits with multiple scan chains, an XNOR gate is added at the first and last flip-flop of every scan chain as shown in Figure 5.2. A parallel counter is used to monitor the activity and trigger the frequency control block once the activity threshold is reached. The rest of the circuitry remains unchanged (Figure 8.1). The use of compression does not affect the implementation since activity is monitored in every scan chain.

Chapter 9

Mathematical Analysis

This chapter deals with a mathematical model for the scheme proposed in this thesis. The reduction in scan-in time for the different models proposed have been analyzed. The first section deals with the model proposed for circuits with peak activity factors of 1, the second section concerns the model for circuits with peak activity factors lesser than 1 and the third section deals with the model where external test equipment is used to test the circuit and the frequency of scan clock is pre-simulated and stored in the ATE.

9.1 Circuits with Peak Activity Factors of 1

This section deals with the estimation of reduction in scan-in time using dynamic control of scan clock frequency for circuits with peak activity factors (α_{peak}) of 1. It is assumed that the vector captured in the scan chain prior to the start of scan-in has an activity factor of 1. The scan chain is therefore assumed to consist of alternating 1s and 0s.

Let N be the number of flip-flops, A be the non-transition density ($A = 1 - \alpha$), v be the number of frequencies and T be the time period corresponding to the fastest clock.

Time period of the fastest clock is v times faster than the slowest clock. Therefore, the time period of the slowest clock is given by vT . If the vectors were scanned in at the slowest clock, the total scan-in time per vector is given by NvT .

The number of non-transitions in the input vector equals AN . Thus, AN non-transitions occur in N cycles. Therefore a non-transition occurs every $\frac{1}{A}$ cycles. Hence, x non-transitions will occur in $\frac{x}{A}$ cycles.

Table 9.1: Determination of Clock Cycle Range for Different Frequencies

S.No.	Clock period	Number of non-transitions		Clock cycles	
		Lower limit	Upper limit	Lower limit	Upper limit
1	vT	0	$\lceil \frac{N}{v} \rceil$	0	$\lceil \frac{N}{Av} \rceil$
2	$(v-1)T$	$\lceil \frac{N}{v} \rceil$	$\lceil \frac{2N}{v} \rceil$	$\lceil \frac{N}{Av} \rceil$	$\lceil \frac{2N}{Av} \rceil$
.
i	$(v-i+1)T$	$\lceil \frac{(i-1)N}{v} \rceil$	$\lceil \frac{iN}{v} \rceil$	$\lceil \frac{(i-1)N}{Av} \rceil$	$\lceil \frac{iN}{Av} \rceil$
.
v	T	$\lceil \frac{(v-1)N}{v} \rceil$	$\lceil \frac{vN}{v} \rceil$	$\lceil \frac{(v-1)N}{Av} \rceil$	$\lceil \frac{vN}{Av} \rceil$

The scan chain can hold a maximum of N non-transitions and in order to enable speed-up for all ranges of non-transitions, it is important that the frequency is increased only after the counter counts up to $\frac{N}{v}$.

Table 9.1 tabulates the number of non-transitions after which the clock frequency is increased and the clock cycles (incoming scan bit positions in scan vector) during which a particular clock period is employed. The first bit in the scan-in vector is shifted at the lowest possible frequency (first frequency employed) which corresponds to a time period of vT . The frequency is not increased until $\frac{N}{v}$ non-transitions enter the scan chain as discussed earlier. Since a non-transition occurs every $\frac{1}{A}$ cycles, $\frac{N}{v}$ non-transitions occur in $\frac{N}{Av}$ cycles. Thus, the frequency is not increased until $\frac{N}{v}$ non-transitions occur in about $\frac{N}{Av}$ cycles. The counter is then reset and the frequency is increased to the next step which corresponds to a time period of $(v-1)T$. The frequency is not increased any further until the counter counts up to $\frac{N}{v}$, i.e., until the number of non-transitions in the scan chain reaches $\frac{2N}{v}$. This occurs after about $\frac{2N}{Av}$ cycles (since a non-transition occurs every $\frac{1}{A}$ cycles). Thus, the scan clock frequency (second frequency employed) whose clock period is $(v-1)T$ is used between the cycles $\frac{N}{Av}$ and $\frac{2N}{Av}$. The clock period can reach a maximum of T (v^{th} frequency employed)

when the scan-chain can be completely filled with non-transitions. Thus, this frequency is used when the number of non-transitions in the scan chain ranges between $\frac{(v-1)N}{v}$ and N or in other words, this frequency is used between clock cycles $\frac{(v-1)N}{Av}$ and $\frac{N}{A}$. Thus, the i^{th} frequency corresponds to a clock period of $(v - i + 1)T$ when the scan chain has between $\frac{(i-1)N}{v}$ and $\frac{iN}{v}$ non-transitions. The i^{th} frequency is employed between clock cycles $\frac{(i-1)N}{Av}$ and $\frac{iN}{Av}$.

The scan clock initially has a clock period of vT in cycle 1. The scan clock period is decreased in steps until the N^{th} cycle. Thus, the clock cycle corresponding to the last scan clock frequency is N . If the maximum number of speeds the scan clock will reach, for any vector is given by x , then

$$\frac{N}{Av}x = N \quad (9.1)$$

$$x = Av \quad (9.2)$$

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

The total scan-in time per vector is the sum of scan-in times at each frequency. The scan-in time at each frequency is given by the product of the number of cycles run at each frequency and the time period of the clock. These values are given in Table 9.1.

Total scan-in time per vector is given by

$$\sum_{i=1}^{Av} \left\{ \left\lceil \frac{iN}{Av} \right\rceil - \left\lceil \frac{(i-1)N}{Av} \right\rceil \right\} (v - i + 1)T \quad (9.3)$$

where v is usually chosen as a power of 2 since it is possible to design a divide by 2^n frequency divider with n flip-flops. If N was also chosen as a power of 2, the formula can be reduced to

$$\text{Total scan - in time per vector} = \sum_{i=1}^{Av} \left\{ \left(\frac{N}{Av} \right) (v - i + 1) T \right\} \quad (9.4)$$

$$= \left(\frac{N}{Av} \right) \left(v \cdot Av - \frac{Av(Av + 1)}{2} + Av \right) T \quad (9.5)$$

Time per vector if a single speed is used = NvT

$$\text{Reduction in scan - in time} = \frac{\{NTv - NT(v - \frac{Av+1}{2} + 1)\}}{NTv} \quad (9.6)$$

$$= \frac{A}{2} - \frac{1}{2v} \quad (9.7)$$

$$= \frac{(1 - \alpha)}{2} - \frac{1}{2v} \quad (9.8)$$

A C program was written to generate random vectors for a circuit with 1000 flip-flops. Random vectors were generated to have different average activity factors ranging between 0 and 1. Ten vectors with 1000 bits each were generated per value of activity factor. This was achieved by generating a random floating point number which was compared against the value of activity factor. If the random floating point number was lesser than the value of activity factor, a bit that was different from the previous bit was generated. If the random floating point number was greater than the activity factor, a bit that had the same value as the previous bit was generated. Thus, random vectors were generated to have the chosen average activity factors.

The activity factor of a bit sequence that has a 0-bit probability of p_0 and a 1-bit probability of p_1 is given by

$$p_0p_1 + p_1p_0 \quad (9.9)$$

since a transition occurs when a 1 follows a 0 or when a 0 follows a 1. However, $p_1 = 1 - p_0$. Thus, the activity factor can be calculated as

$$p_0(1 - p_0) + (1 - p_0)p_0 = 2p_0(1 - p_0) \quad (9.10)$$

If bits are generated randomly, the bit probability of generating a 1 or a 0 equals 0.5 and hence, the activity factor of such a bit stream would be 0.5. In order to generate bit streams with lower activity factors, the bits should be generated with a positive correlation, i.e., with a higher probability of generating consecutive bits with the same value. Similarly, bit streams with higher activity factors can be generated when a negative correlation is set, i.e., when the probability of generating consecutive bits with the same value is low.

The scan-in time reduction for these vectors was estimated, and compared with the values obtained from the formula. Table 9.2 shows the variation of scan-in time reduction with the number of frequencies, for an activity factor of 0.5. Table 9.3 shows the variation of scan-in time reduction with activity factor, when the number of frequencies is 8. Both tables compare the scan-in time reductions estimated for the random vectors (column II), with that obtained from the accurate formula given by (9.3) (column III) and from the approximate formula given by (9.8) (column IV). Figure 9.1 shows the plot of scan-in time reduction as a function of the number of frequencies chosen, for different values of activity factor. Figure 9.2 shows the variation of scan-in time reduction with activity factor, for different numbers of frequencies chosen.

It can be observed from Figures 9.1 and 9.2 that for a chosen number of frequencies, vectors with lower transition densities achieve higher reduction in scan-in time. It is also seen that the scan-in time reduction increases when the number of frequencies chosen increases. The scan-in time reduces rapidly till a value of 8 frequencies, after which the reduction is more gradual.

Table 9.2: Variation of Scan-In Time Reduction with Chosen Number of Frequencies for an Activity Factor of 0.5

Number of frequencies	Scan-In time reduction (%)		
	Simulation	Eq.(9.3)	Eq.(9.8)
1	0.00	0.00	0.00
2	0.34	0.00	0.00
4	12.64	12.50	12.50
8	18.78	18.75	18.75
16	22.03	21.90	21.88
32	23.56	23.48	23.44
64	25.17	24.26	24.22
128	27.41	24.66	24.61

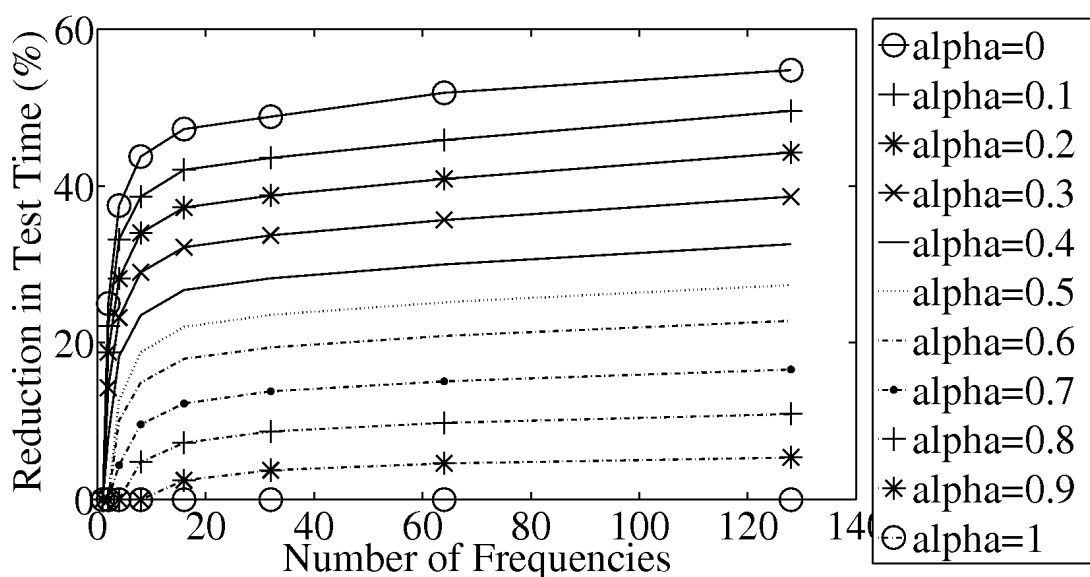


Figure 9.1: Variation of Scan-In Time Reduction with the Number of Frequencies for Various Activity Factors

9.2 Circuits with Peak Activity Factors lesser than 1

This section deals with the estimation of reduction in scan-in time using dynamic control of scan clock frequency for circuits with peak activity factors (α_{peak}) lower than 1. It is

Table 9.3: Variation of Scan-In Time Reduction with Activity Factor when the Number of Frequencies Chosen is 8

Activity factor α	Scan-In time reduction (%)		
	Simulation	Eq.(9.3)	Eq.(9.8)
0	43.75	43.75	43.75
0.1	38.63	38.85	38.75
0.2	34.00	33.95	33.75
0.3	28.97	28.99	28.75
0.4	23.51	23.94	23.75
0.5	18.78	18.75	18.75
0.6	14.92	14.04	13.75
0.7	9.60	9.36	8.75
0.8	4.79	4.68	3.75
0.9	0.00	0.00	0.00
1	0.00	0.00	0.00

assumed that the vector captured in the scan chain prior to the start of scan-in has an activity factor of α_{peak} .

Let N be the number of flip-flops, α_{in} be the activity factor of the scan-in vector, α_{out} be the activity factor of the captured (scan-out) vector, v be the number of frequencies and T be the time period corresponding to the fastest clock.

Time period of the fastest clock is v times faster than the slowest clock. Therefore, the time period of the slowest clock is given by vT . vT is computed from (2.1) using values of peak power limit and α_{peak} . If the vectors were scanned in at the slowest clock, the total scan-in time per vector is given by NvT .

Let αN be the number of transitions in the scan chain in any cycle. The maximum number of transitions the scan chain can hold is $\alpha_{peak}N$. In order to enable frequency

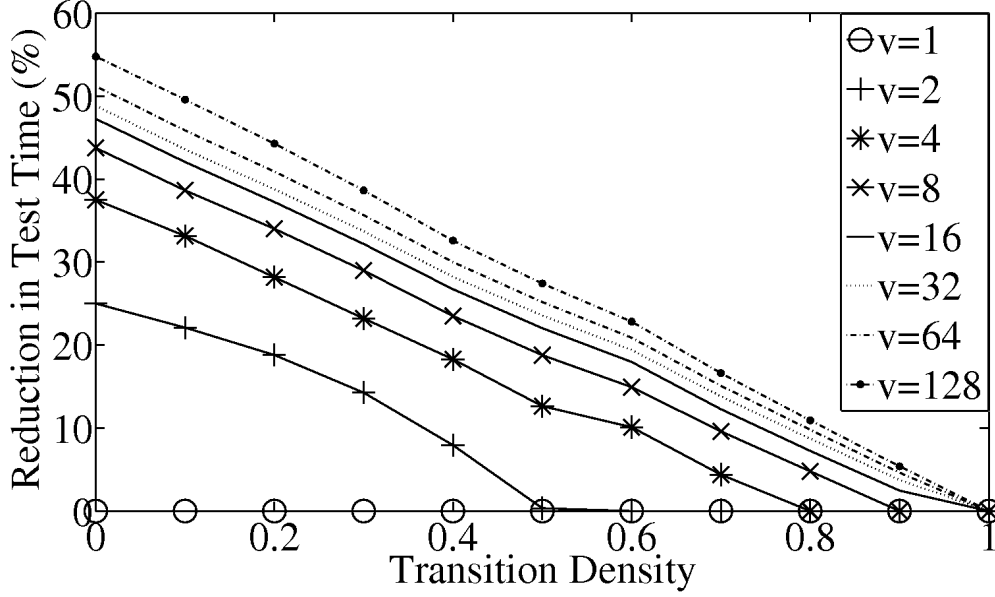


Figure 9.2: Variation of Scan-In Time Reduction with Activity factor for Varying Number of Clock Frequencies

control for all ranges of transitions, a scan clock frequency is specified for every $\frac{\alpha_{peak}N}{v}$ transitions. Thus, the frequency can be modified after encountering $\frac{\alpha_{peak}N}{v}$ transitions. The number of non-transitions in the scan chain equals the difference between N and the number of transitions in the scan chain, i.e., $N - \alpha N$.

It can be seen from Columns V and VI of Table 9.4 that the number of non-transitions can be subtracted by $N(1 - \alpha_{peak})$ to have non-transitions in the range of 0 to $\alpha_{peak}N$. Thus, the modified number of non-transitions can be obtained from the number of transitions as $N - \alpha N - (N(1 - \alpha_{peak}))$ which equals $N(\alpha_{peak} - \alpha)$.

The number of non-transitions in the scan chain in any cycle equals the difference between the number of non-transitions entering and leaving the scan chain. The number of non-transitions entering the scan chain is $(1 - \alpha_{in})N$ and the number of non-transitions leaving the scan chain is $(1 - \alpha_{out})N$. Thus, the number of non-transitions in the scan chain in any cycle is $(\alpha_{out} - \alpha_{in})N$. The non-transition density, A can be defined as $A = \alpha_{out} - \alpha_{in}$. The counter counts up by $\alpha_{out} - \alpha_{in}$ every cycle. Therefore, the counter will count up by 1 in every $\frac{1}{A}$ cycles. Hence, the counter counts up by x in $\frac{x}{A}$ cycles.

Table 9.4: Non-transition Count Range for $\alpha_{peak} \neq 1$

S.No.	Clock period	Number of transitions (αN)		Number of non-transitions ($(1 - \alpha)N$)		Modified number of non-transitions ($(\alpha_{peak} - \alpha)N$)	
		Lower limit	Upper limit	Lower limit	Upper limit	Lower limit	Upper limit
1	vT	$\lceil \frac{(v-1)\alpha_{peak}N}{v} \rceil$	$\lceil \alpha_{peak}N \rceil$	$\lceil (1 - \alpha_{peak})N \rceil$	$\lceil \frac{(v-(v-1)\alpha_{peak})N}{v} \rceil$	0	$\lceil \frac{\alpha_{peak}N}{v} \rceil$
2	$(v-1)T$	$\lceil \frac{(v-2)\alpha_{peak}N}{v} \rceil$	$\lceil \frac{(v-1)\alpha_{peak}N}{v} \rceil$	$\lceil \frac{(v-(v-1)\alpha_{peak})N}{v} \rceil$	$\lceil \frac{(v-(v-2)\alpha_{peak})N}{v} \rceil$	$\lceil \frac{\alpha_{peak}N}{v} \rceil$	$\lceil \frac{2\alpha_{peak}N}{v} \rceil$
.
i	$(v-i+1)T$	$\lceil \frac{(v-i)\alpha_{peak}N}{v} \rceil$	$\lceil \frac{(v-i+1)\alpha_{peak}N}{v} \rceil$	$\lceil \frac{(v-(v-i+1)\alpha_{peak})N}{v} \rceil$	$\lceil \frac{(v-(v-i)\alpha_{peak})N}{v} \rceil$	$\lceil \frac{(i-1)\alpha_{peak}N}{v} \rceil$	$\lceil \frac{i\alpha_{peak}N}{v} \rceil$
.
v	T	0	$\lceil \frac{\alpha_{peak}N}{v} \rceil$	$\lceil \frac{(v-\alpha_{peak})N}{v} \rceil$	N	$\lceil \frac{(v-1)\alpha_{peak}N}{v} \rceil$	$\lceil \alpha_{peak}N \rceil$

Table 9.5: Determination of Clock Cycle Range for $\alpha_{peak} \neq 1$

S.No.	Clock period	Clock cycles $(\frac{(\alpha_{peak}-\alpha)N}{A})$	
		Lower limit	Upper limit
1	vT	0	$\lceil \frac{\alpha_{peak}N}{Av} \rceil$
2	$(v-1)T$	$\lceil \frac{\alpha_{peak}N}{Av} \rceil$	$\lceil \frac{2\alpha_{peak}N}{Av} \rceil$
.	.	.	.
i	$(v-i+1)T$	$\lceil \frac{(i-1)\alpha_{peak}N}{Av} \rceil$	$\lceil \frac{i\alpha_{peak}N}{Av} \rceil$
.	.	.	.
v	T	$\lceil \frac{(v-1)\alpha_{peak}N}{Av} \rceil$	$\lceil \frac{\alpha_{peak}N}{A} \rceil$

If the scan-in vector has a uniform activity factor that is higher than that of the captured vector, the number of non-transitions entering the scan chain will be lower than that leaving it and hence there will be no change in scan clock frequency. However, if the scan-in vector has a uniform activity factor that is lower than that of the captured vector, the number of non-transitions entering the scan chain exceeds that leaving the scan chain. Therefore, the scan clock frequency is continuously increased.

Table 9.4 and Table 9.5 tabulate the clock cycles (incoming scan bit positions in scan vector) during which a particular clock period is employed. The first bit in the scan-in vector is shifted at the lowest possible frequency (first frequency employed) which corresponds to a time period of vT . The frequency is not increased until $\frac{\alpha_{peak}N}{v}$ non-transitions enter the scan chain as discussed earlier. Since the counter counts up every $\frac{1}{A}$ cycles, $\frac{\alpha_{peak}N}{v}$ non-transitions occur in $\frac{\alpha_{peak}N}{Av}$ cycles. Thus, the frequency is not increased until about $\frac{\alpha_{peak}N}{Av}$ cycles. The counter is then reset and the frequency is increased to the next step which corresponds to a time period of $(v-1)T$. The frequency is not increased any further until the counter

counts up to $\frac{\alpha_{peak}N}{v}$, i.e., until the number of non-transitions in the scan chain increases by $\frac{2\alpha_{peak}N}{v}$. This occurs after about $\frac{2\alpha_{peak}N}{Av}$ cycles (since a non-transition occurs every $\frac{1}{A}$ cycles). Thus, the scan clock frequency (second frequency employed) whose clock period is $(v-1)T$ is used between the cycles $\frac{\alpha_{peak}N}{Av}$ and $\frac{2\alpha_{peak}N}{Av}$. The clock period can reach a maximum of T (v^{th} frequency employed). Thus, this frequency is used when the number of non-transitions in the scan chain increases by a value in the range between $\frac{(v-1)\alpha_{peak}N}{v}$ and $\alpha_{peak}N$ or in other words, this frequency is used between clock cycles $\frac{(v-1)\alpha_{peak}N}{Av}$ and $\frac{\alpha_{peak}N}{A}$. Thus, the i^{th} frequency corresponds to a clock period of $(v-i+1)T$ when the scan chain has between $\frac{(i-1)\alpha_{peak}N}{v}$ and $\frac{i\alpha_{peak}N}{v}$ increase in non-transitions. The i^{th} frequency is employed between clock cycles $\frac{(i-1)\alpha_{peak}N}{Av}$ and $\frac{i\alpha_{peak}N}{Av}$.

The scan clock initially has a clock period of vT in cycle 1. The scan clock period is decreased in steps until the N^{th} cycle. Thus, the clock cycle corresponding to the last scan clock frequency is N . If the maximum number of speeds the scan clock will reach, for any vector is given by x , then

$$\frac{\alpha_{peak}N}{Av}x = N \quad (9.11)$$

$$x = \frac{Av}{\alpha_{peak}} \quad (9.12)$$

Thus, the number of scan clock frequencies employed for a scan-in vector with an activity factor of α_{in} when the activity factor of the captured vector is α_{out} is $\frac{(\alpha_{out}-\alpha_{in})v}{\alpha_{peak}}$.

The total scan-in time per vector is the sum of scan-in times at each frequency. The scan-in time at each frequency is given by the product of the number of cycles run at each frequency and the time period of the clock. These values are given in Table 9.5.

Total scan-in time per vector is given by

$$\sum_{i=1}^{\frac{Av}{\alpha_{peak}}} \left\{ \left\lceil \frac{i\alpha_{peak}N}{Av} \right\rceil - \left\lceil \frac{(i-1)\alpha_{peak}N}{Av} \right\rceil \right\} (v-i+1)T \quad (9.13)$$

where v is usually chosen as a power of 2 since it is possible to design a divide by 2^n frequency divider with n flip-flops. If N was also chosen as a power of 2, the formula can be reduced to

$$\text{Total scan - in time per vector} = \sum_{i=1}^{\frac{Av}{\alpha_{peak}}} \left\{ \left(\frac{\alpha_{peak}N}{Av} \right) (v - i + 1) T \right\} \quad (9.14)$$

$$= \left(\frac{\alpha_{peak}N}{Av} \right) \left(v \cdot \frac{Av}{\alpha_{peak}} - \frac{\frac{Av}{\alpha_{peak}} \left(\frac{Av}{\alpha_{peak}} + 1 \right)}{2} + \frac{Av}{\alpha_{peak}} \right) T \quad (9.15)$$

Time per vector if a single speed is used = NvT

$$\text{Reduction in scan - in time} = \frac{\left\{ NTv - NT \left(v - \frac{Av}{2\alpha_{peak}} + \frac{1}{2} \right) \right\}}{NTv} \quad (9.16)$$

$$= \frac{A}{2\alpha_{peak}} - \frac{1}{2v} \quad (9.17)$$

$$= \frac{(\alpha_{out} - \alpha_{in})}{2\alpha_{peak}} - \frac{1}{2v} \quad (9.18)$$

9.3 Externally Tested Circuits

When a circuit is tested externally, it is possible to estimate the scan clock frequency during every clock cycle through power analysis of test vectors. The frequency of scan clock at the start of scan-in is computed based on the activity factor of the vector captured in the scan chain prior to scan-in. The subsequent scan clock frequencies are changed in steps based on the activity in the scan chain during every clock cycle. This section deals with the mathematical analysis of reduction in scan-in time achieved using dynamic control of scan clock frequency in such circuits.

Let N be the number of flip-flops, α_{in} be the activity factor of the scan-in vector, α_{out} be the activity factor of the captured vector, α_{peak} be the peak activity factor observed

in the scan chain during test, v be the number of frequencies and T be the time period corresponding to the fastest clock.

Time period of the fastest clock is v times faster than the slowest clock. Therefore, the time period of the slowest clock is given by vT . vT is computed from Eq. 2.1 using values of peak power limit and α_{peak} . If the vectors were scanned in at the slowest clock, the total scan-in time per vector is given by NvT .

The maximum number of transitions the scan chain can hold is $\alpha_{peak}N$. In order to enable frequency control for all ranges of transitions, a scan clock frequency is specified for every $\frac{\alpha_{peak}N}{v}$ transitions. If the number of transitions in the scan chain equals αN , then the number of non-transitions in the scan chain equals the difference between N and the number of transitions in the scan chain, i.e., $N - \alpha N$.

It can be seen from Columns V and VI of Table 9.4 that the number of non-transitions can be subtracted by $N(1 - \alpha_{peak})$ to have non-transitions in the range of 0 to $\alpha_{peak}N$. Thus, the modified number of non-transitions can be obtained from the number of transitions as $N - \alpha N - (N(1 - \alpha_{peak}))$ which equals $N(\alpha_{peak} - \alpha)$. Thus the modified non-transition density, A can be defined as $A = \alpha_{peak} - \alpha$ and the frequency is modified after encountering $\frac{\alpha_{peak}N}{v}$ non-transitions.

If the scan-in vector has a uniform activity factor (α_{in}) that is higher than that of the captured vector (α_{out}), the number of non-transitions entering the scan chain will be lower than that leaving it and hence the scan clock frequency is continuously decreased. The scan clock frequency will initially correspond to $(1 - \alpha_{out})N$ number of non-transitions and will be decreased until it corresponds to $(1 - \alpha_{in})N$ number of non-transitions.

However, if the scan-in vector has a uniform activity factor (α_{in}) that is lower than that of the captured vector (α_{out}), the number of non-transitions entering the scan chain exceeds that leaving the scan chain. Therefore, the scan clock frequency is continuously increased. The scan clock frequency will initially correspond to $(1 - \alpha_{out})N$ number of non-transitions and will be increased until it corresponds to $(1 - \alpha_{in})N$ number of non-transitions.

Table 9.6: Clock Cycle Range for $\alpha_{in} < \alpha_{out}$

S.No.	Clock period	Clock cycles $(\frac{(\alpha_{peak}-\alpha)N}{\alpha_{out}-\alpha_{in}})$	
		Lower limit	Upper limit
1	vT	0	$\lceil \frac{\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v} \rceil$
2	$(v-1)T$	$\lceil \frac{\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v} \rceil$	$\lceil \frac{2\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v} \rceil$
.	.	.	.
i	$(v-i+1)T$	$\lceil \frac{(i-1)\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v} \rceil$	$\lceil \frac{i\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v} \rceil$
.	.	.	.
v	T	$\lceil \frac{(v-1)\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v} \rceil$	$\lceil \frac{\alpha_{peak}N}{\alpha_{out}-\alpha_{in}} \rceil$

Both the cases mentioned ($\alpha_{in} < \alpha_{out}$ and $\alpha_{in} > \alpha_{out}$) are analyzed in this section.

Case 1: $\alpha_{in} < \alpha_{out}$

The number of non-transitions in the scan chain in any cycle equals the difference between the number of non-transitions entering and leaving the scan chain. The number of non-transitions entering the scan chain is $(1 - \alpha_{in})N$ and the number of non-transitions leaving the scan chain is $(1 - \alpha_{out})N$. Thus, the number of non-transitions in the scan chain in any cycle is $(\alpha_{out} - \alpha_{in})N$. The counter therefore counts up by $\alpha_{out} - \alpha_{in}$ every cycle. Therefore, the counter will count up by 1 in every $\frac{1}{\alpha_{out}-\alpha_{in}}$ cycles. Hence, the counter counts up by x in $\frac{x}{\alpha_{out}-\alpha_{in}}$ cycles.

Table 9.4 and Table 9.6 tabulate the clock cycles (incoming scan bit positions in scan vector) during which a particular clock period is employed. The first bit in the scan-in vector can be shifted at the lowest possible frequency (first frequency employed) which corresponds to a time period of vT . The frequency is not increased until $\frac{\alpha_{peak}N}{v}$ non-transitions enter

the scan chain as discussed earlier. Since the counter counts up every $\frac{1}{\alpha_{out}-\alpha_{in}}$ cycles, $\frac{\alpha_{peak}N}{v}$ non-transitions occur in $\frac{\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$ cycles. Thus, the frequency is not increased until about $\frac{\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$ cycles. The counter is then reset and the frequency is increased to the next step which corresponds to a time period of $(v-1)T$. The frequency is not increased any further until the counter counts up to $\frac{\alpha_{peak}N}{v}$, i.e., until the number of non-transitions in the scan chain increases by $\frac{2\alpha_{peak}N}{v}$. This occurs after about $\frac{2\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$ cycles (since a non-transition occurs every $\frac{1}{\alpha_{out}-\alpha_{in}}$ cycles). Thus, the scan clock frequency (second frequency employed) whose clock period is $(v-1)T$ is used between the cycles $\frac{\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$ and $\frac{2\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$. The clock period can reach a maximum of T (v^{th} frequency employed). Thus, this frequency is used when the increase in number of non-transitions in the scan chain ranges between $\frac{(v-1)\alpha_{peak}N}{v}$ and $\alpha_{peak}N$ or in other words, this frequency is used between clock cycles $\frac{(v-1)\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$ and $\frac{\alpha_{peak}N}{\alpha_{out}-\alpha_{in}}$. Thus, the i^{th} frequency is employed between clock cycles $\frac{(i-1)\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$ and $\frac{i\alpha_{peak}N}{(\alpha_{out}-\alpha_{in})v}$.

The scan clock initially has a clock period that corresponds to $(1-\alpha_{out})N$ number of non-transitions or $\alpha_{out}N$ number of transitions in the scan chain in cycle 1.

The i^{th} scan clock frequency corresponds to $\frac{(v-i+1)\alpha_{peak}N}{v}$ number of transitions in the scan chain. Thus, if

$$\alpha_{out}N = \frac{(v-i+1)\alpha_{peak}N}{v} \quad (9.19)$$

then, i^{th} frequency is used as starting frequency of scan-in.

$$i = v + 1 - \frac{\alpha_{out}v}{\alpha_{peak}} \quad (9.20)$$

The scan clock period is decreased (since $\alpha_{in} < \alpha_{out}$, more non-transitions enter the scan chain every cycle and hence scan clock frequency has to be increased) in steps until the N^{th} cycle. Thus, the clock cycle corresponding to the last scan clock frequency is N . If the maximum number of speeds the scan clock will utilize for any vector is given by x , then

$$\frac{\alpha_{peak}N}{(\alpha_{out} - \alpha_{in})v}x = N \quad (9.21)$$

$$x = \frac{\alpha_{out} - \alpha_{in}v}{\alpha_{peak}} \quad (9.22)$$

Thus, the number of scan clock frequencies employed for a scan-in vector with an activity factor of α_{in} when the activity factor of the captured vector is α_{out} is $\frac{(\alpha_{out}-\alpha_{in})v}{\alpha_{peak}}$.

If the j^{th} frequency is the last frequency utilized during scan-in, then

$$j = \left(v + 1 - \frac{\alpha_{out}v}{\alpha_{peak}}\right) + \left(\frac{\alpha_{out} - \alpha_{in}v}{\alpha_{peak}}\right) - 1 \quad (9.23)$$

$$j = v - \frac{\alpha_{in}v}{\alpha_{peak}} \quad (9.24)$$

The total scan-in time per vector is the sum of scan-in times at each frequency. The scan-in time at each frequency is given by the product of the number of cycles run at each frequency and the time period of the clock. These values are given in Table 9.6.

Total scan-in time per vector is given by

$$\sum_{i=v+1-\frac{\alpha_{out}v}{\alpha_{peak}}}^{v-\frac{\alpha_{in}v}{\alpha_{peak}}} \left\{ \left\lceil \frac{i\alpha_{peak}N}{(\alpha_{out} - \alpha_{in})v} \right\rceil - \left\lceil \frac{(i-1)\alpha_{peak}N}{(\alpha_{out} - \alpha_{in})v} \right\rceil \right\} (v - i + 1)T \quad (9.25)$$

where v is usually chosen as a power of 2 since it is possible to design a divide by 2^n frequency divider with n flip-flops. If N was also chosen as a power of 2, the formula can be reduced to

$$\text{Total scan-in time per vector} = \sum_{i=v+1-\frac{\alpha_{out}v}{\alpha_{peak}}}^{v-\frac{\alpha_{in}v}{\alpha_{peak}}} \left\{ \left(\frac{\alpha_{peak}N}{(\alpha_{out} - \alpha_{in})v} \right) (v - i + 1)T \right\} \quad (9.26)$$

$$= \left(\frac{\alpha_{peak} N}{(\alpha_{out} - \alpha_{in})v} \right) \left(v \cdot \frac{(\alpha_{out} - \alpha_{in})v}{\alpha_{peak}} - \frac{\frac{(\alpha_{out} - \alpha_{in})v}{\alpha_{peak}} (2v + 1 - \frac{(\alpha_{out} + \alpha_{in})v}{\alpha_{peak}})}{2} + \frac{(\alpha_{out} - \alpha_{in})v}{\alpha_{peak}} \right) T \quad (9.27)$$

Time per vector if a single speed is used = NvT

$$Reduction\ in\ scan - in\ time = \frac{\{NTv - NT(v - \frac{(2v+1 - \frac{(\alpha_{out} + \alpha_{in})v}{\alpha_{peak}})}{2} + 1)\}}{NTv} \quad (9.28)$$

$$= \frac{((\alpha_{peak} - \alpha_{in}) + (\alpha_{peak} - \alpha_{out}))}{2\alpha_{peak}} - \frac{1}{2v} \quad (9.29)$$

It can be seen that the formula can be extended to the implementation discussed in Section 1 as well. If scan-in-start frequency is not pre-determined, and the peak activity factor is assumed to be 1, then the activity factor of the captured vector is also assumed to be 1, i.e., $\alpha_{out} = \alpha_{peak} = 1$. This reduces the formula to

$$\frac{(1 - \alpha_{in})}{2} - \frac{1}{2v} \quad (9.30)$$

which is the same as that obtained in Section 1.

Case 2: $\alpha_{in} > \alpha_{out}$

When the activity factor of the scan-in vector is greater than that of the captured vector, the number of non-transitions reduces as the scan-in progresses. Thus, scan clock frequency has to be continuously decreased.

The number of transitions in the scan chain in any cycle equals the difference between the number of transitions entering and leaving the scan chain. The number of transitions entering the scan chain is $\alpha_{in}N$ and the number of transitions leaving the scan chain is $\alpha_{out}N$. Thus, the number of transitions in the scan chain in any cycle is $(\alpha_{in} - \alpha_{out})N$.

Table 9.7: Clock Cycle Range for $\alpha_{in} > \alpha_{out}$

S.No.	Clock period	Clock cycles $(\frac{(\alpha_{peak}-\alpha)N}{\alpha_{in}-\alpha_{out}})$	
		Lower limit	Upper limit
1	vT	0	$\lceil \frac{\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v} \rceil$
2	$(v-1)T$	$\lceil \frac{\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v} \rceil$	$\lceil \frac{2\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v} \rceil$
.	.	.	.
i	$(v-i+1)T$	$\lceil \frac{(i-1)\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v} \rceil$	$\lceil \frac{i\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v} \rceil$
.	.	.	.
v	T	$\lceil \frac{(v-1)\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v} \rceil$	$\lceil \frac{\alpha_{peak}N}{\alpha_{in}-\alpha_{out}} \rceil$

In other words, the counter counts down by $\alpha_{in} - \alpha_{out}$ every cycle. Therefore, the counter (which counts the number of non-transitions in the scan chain) will count down by 1 in every $\frac{1}{\alpha_{in}-\alpha_{out}}$ cycles. Hence, the counter counts down by x in $\frac{x}{\alpha_{in}-\alpha_{out}}$ cycles.

Table 9.4 and Table 9.7 tabulate the clock cycles (incoming scan bit positions in scan vector) during which a particular clock period is employed. The first bit in the scan-in vector can be shifted at the highest possible frequency (first frequency employed) which corresponds to a time period of T . The frequency is not decreased until $\frac{\alpha_{peak}N}{v}$ non-transitions leave the scan chain as discussed earlier. Since the counter counts down every $\frac{1}{\alpha_{in}-\alpha_{out}}$ cycles, $\frac{\alpha_{peak}N}{v}$ non-transitions leave in $\frac{\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$ cycles. Thus, the frequency is not decreased until about $\frac{\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$ cycles. The counter is then reset and the frequency is decreased to the next step which corresponds to a time period of $2T$. The frequency is not decreased any further until the counter counts down to $\frac{\alpha_{peak}N}{v}$, i.e., until the number of transitions that have entered the scan chain reaches $\frac{2\alpha_{peak}N}{v}$. This occurs after about $\frac{2\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$ cycles since a transition enters (at the same rate a non-transition leaves) every $\frac{1}{\alpha_{in}-\alpha_{out}}$ cycles. Thus, the scan clock

frequency whose clock period is $(v-1)T$ is used between the cycles $\frac{\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$ and $\frac{2\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$. The clock period can reach a minimum of vT . Thus, this frequency is used when the number of transitions that entered the scan chain ranges between $\frac{(v-1)\alpha_{peak}N}{v}$ and $\alpha_{peak}N$ or in other words, this frequency is used between clock cycles $\frac{(v-1)\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$ and $\frac{\alpha_{peak}N}{\alpha_{in}-\alpha_{out}}$. Thus, the i^{th} frequency is employed between clock cycles $\frac{(i-1)\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$ and $\frac{i\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}$.

The scan clock initially has a clock period that corresponds to $(1-\alpha_{out})N$ number of non-transitions or $\alpha_{out}N$ number of transitions in the scan chain in cycle 1.

The i^{th} scan clock frequency corresponds to $\frac{(v-i+1)\alpha_{peak}N}{v}$ number of transitions in the scan chain. Thus, if

$$\alpha_{out}N = \frac{(v-i+1)\alpha_{peak}N}{v} \quad (9.31)$$

then, i^{th} frequency is used as starting frequency of scan-in.

$$i = v + 1 - \frac{\alpha_{out}v}{\alpha_{peak}} \quad (9.32)$$

The scan clock period is increased (since $\alpha_{in} > \alpha_{out}$, more transitions enter the scan chain every cycle and hence scan clock frequency has to be decreased) in steps until the N^{th} cycle. Thus, the clock cycle corresponding to the last scan clock frequency is N . If the maximum number of speeds the scan clock will utilize for any vector is given by x , then

$$\frac{\alpha_{peak}N}{(\alpha_{in}-\alpha_{out})v}x = N \quad (9.33)$$

$$x = \frac{(\alpha_{in}-\alpha_{out})v}{\alpha_{peak}} \quad (9.34)$$

Thus, the number of scan clock frequencies employed for a scan-in vector with an activity factor of α_{in} when the activity factor of the captured vector is α_{out} is $\frac{(\alpha_{in}-\alpha_{out})v}{\alpha_{peak}}$.

If the j^{th} frequency is the last frequency utilized during scan-in, then

$$j + \left(\frac{(\alpha_{in} - \alpha_{out})v}{\alpha_{peak}}\right) - 1 = \left(v + 1 - \frac{\alpha_{out}v}{\alpha_{peak}}\right) \quad (9.35)$$

$$j = v - \frac{\alpha_{in}v}{\alpha_{peak}} + 2 \quad (9.36)$$

The total scan-in time per vector is the sum of scan-in times at each frequency. The scan-in time at each frequency is given by the product of the number of cycles run at each frequency and the time period of the clock. These values are given in Table 9.7.

Total scan-in time per vector is given by

$$\sum_{i=v+2-\frac{\alpha_{in}v}{\alpha_{peak}}}^{v+1-\frac{\alpha_{out}v}{\alpha_{peak}}} \left\{ \left\lceil \frac{i\alpha_{peak}N}{(\alpha_{in} - \alpha_{out})v} \right\rceil - \left\lceil \frac{(i-1)\alpha_{peak}N}{(\alpha_{in} - \alpha_{out})v} \right\rceil \right\} (v - i + 1)T \quad (9.37)$$

where v is usually chosen as a power of 2 since it is possible to design a divide by 2^n frequency divider with n flip-flops. If N was also chosen as a power of 2, the formula can be reduced to

$$\text{Total scan - in time per vector} = \sum_{i=v+2-\frac{\alpha_{in}v}{\alpha_{peak}}}^{v+1-\frac{\alpha_{out}v}{\alpha_{peak}}} \left\{ \left(\frac{\alpha_{peak}N}{(\alpha_{in} - \alpha_{out})v} \right) (v - i + 1)T \right\} \quad (9.38)$$

$$= \left(\frac{\alpha_{peak}N}{(\alpha_{out} - \alpha_{in})v} \right) \left(v \cdot \frac{(\alpha_{in} - \alpha_{out})v}{\alpha_{peak}} - \frac{(\alpha_{in} - \alpha_{out})v}{\alpha_{peak}} \frac{(2v + 3 - \frac{(\alpha_{out} + \alpha_{in})v}{\alpha_{peak}})}{2} + \frac{(\alpha_{in} - \alpha_{out})v}{\alpha_{peak}} \right) T \quad (9.39)$$

Time per vector if a single speed is used = NvT

$$\text{Reduction in scan - in time} = \frac{\left\{ NTv - NT \left(\frac{(\alpha_{in} + \alpha_{out})v}{2\alpha_{peak}} - \frac{1}{2} \right) \right\}}{NTv} \quad (9.40)$$

$$= \frac{((\alpha_{peak} - \alpha_{in}) + (\alpha_{peak} - \alpha_{out}))}{2\alpha_{peak}} + \frac{1}{2v} \quad (9.41)$$

Chapter 10

Experimental Results

The dynamic scan clock control technique was implemented on benchmark circuits in order to study the reduction in test time achieved for circuits of varying sizes. The results obtained through simulation on ISCAS89 and ITC02 circuits are discussed in this chapter.

10.1 Simulations on ISCAS89 Benchmark Circuits

The verilog netlists of the ISCAS89 benchmark circuits were used for simulation purposes. Flip-flops were added at all primary inputs and primary outputs. All the flip-flops were then converted to scan flip-flops and chained together. Thus, the number of flip-flops in the circuit would be the sum of the number of primary inputs, the number of primary outputs and the number of D-type flip-flops.

A 23-bit LFSR, a 23-bit SAR, and a BIST controller were chosen [2] and implemented according to the design rules in [56]. A single bit from the LFSR was fed into the scan input of the benchmark circuit, and the scan output was fed into the SAR. The number of random patterns required to achieve sufficient fault coverage was obtained for each circuit from [10], and this information was incorporated into the BIST controller. The sequential circuit along with the BIST circuitry was treated as the core circuit for test time and area analysis.

The counter, frequency control circuitry, and frequency divider circuitry required to achieve dynamic frequency control were implemented. The number of frequencies for each circuit was chosen according to the size of the circuit, which was treated as a function of the number of scan flip-flops in the circuit. The circuit was assembled as shown in Figure 4.2.

The simulation tool from MentorGraphics, ModelSim, was used to simulate the circuits with and without the dynamic frequency control circuitry. The time required for test application was recorded in each case.

The synthesis tool from Synopsys, DesignCompiler, was used to analyze the area of the circuits with and without the dynamic frequency control circuitry.

Since the LFSR generates pseudo random patterns, the activity factor of each bit is about 0.5. From (9.2), $x = 0.5v$, and hence, the number of frequencies the circuit will run at, is half the chosen number of frequencies. This corresponds to a clock period of $\{(0.5v + 1)T\}$ from Table 9.1. However, during power analysis, the next higher frequency is taken into consideration, in order to obtain pessimistic data. Thus, power analysis is done for a clock period of $0.5vT$, i.e., for a clock having twice the lowest frequency. Therefore, the power dissipated by the circuit for a activity factor of 0.5 at every node and operating at twice the lowest frequency, was estimated for every circuit. The dynamic frequency control circuitry was included in this analysis.

Table 10.1 shows the results achieved using dynamic control of scan clock frequency for the ISCAS89 benchmark circuits with single scan chain tested with BIST test-per-scan technique. The number of frequencies chosen for the circuit is shown in column III. The percentage reduction in test time with respect to the test time for the core circuit is shown in column IV and the percentage increase in area with respect to the area of the core circuit is shown in column V.

At any node, the capacitance and the voltage are constant. Therefore, from (2.1), the power dissipated at any node is proportional to the product of activity and frequency. Thus, the activity per unit time is a direct measure of power dissipated in a circuit. Therefore, an analysis to find activity per unit time was performed on s386 benchmark circuit. The Synopsys power analysis tool, PrimeTime PX, was used for this purpose. The activity per unit time in every cycle was found for the circuit for a vector with an activity factor of 1. The peak among these values was set as the limit for activity per unit time. The values of

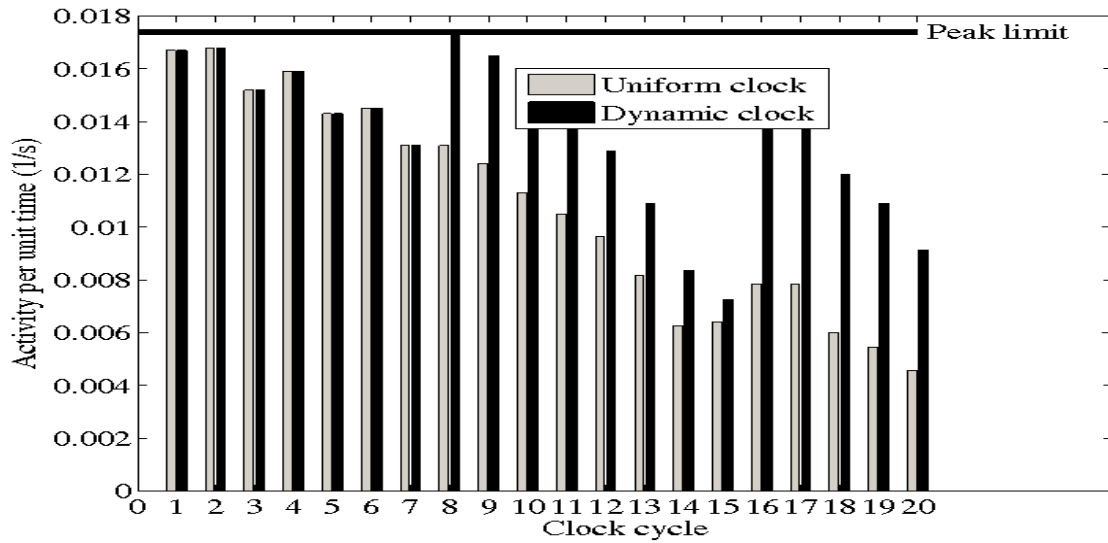


Figure 10.1: Activity per Unit Time Analysis for s386 Circuit

activity per unit time in every cycle were found for a vector with an activity factor of 0.25 using uniform clock and dynamic clock methods. The results are shown in Figure 10.1. It can be seen that the activity per unit time in every cycle is closer to the peak limit when dynamic clock method is used. It can also be seen that the peak limit is never exceeded in both methods. A reduction of 22.5% was observed when the dynamic clock method was used for this vector.

The results for multiple scan chain implementation would be very similar to that obtained for single scan chain implementation. The test time will not vary much since the activity of the circuit will be very similar in both single and multiple chain implementations. However, there would be a marginal increase in area due to the additional xnor gates at the first flip-flop of every scan chain and also due to the use of a parallel counter as opposed to the simple counter used in single scan chain implementation.

It can be seen that the results for reduction in test time conform to the theoretical results given in Figures 9.1 and 9.2. Two trends are clearly observed in Table 10.1. As circuit size increases, the area increase drops and test time reduction improves. These circuits are not very large for today's standards and we can expect better results as predicted by the analysis.

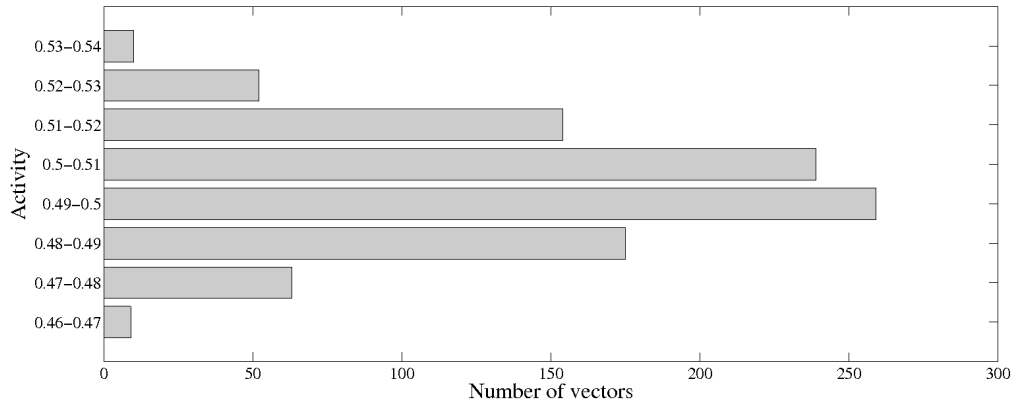
10.2 Mathematical Analysis on ITC02 Benchmark Circuits

In order to estimate test time reduction in larger circuits, accurate mathematical analysis was performed on ITC02 circuits. Test time reduction was computed for best ($\alpha \approx 0$), moderate ($\alpha = 0.5$) and worst ($\alpha \approx 1$) case activity factors. The BIST test-per-scan model was assumed for this purpose. Table 10.2 shows the test time reduction that can be achieved in larger circuits. The number of scan flip-flops indicated in Column II is the sum of number of inputs, number of outputs and number of flip-flops in the circuit. The number of frequencies chosen for the circuit is shown in column III. The test time reduction achieved for the chosen number of frequencies for best, moderate and worst case activity factors are shown in Columns IV, V and VI respectively.

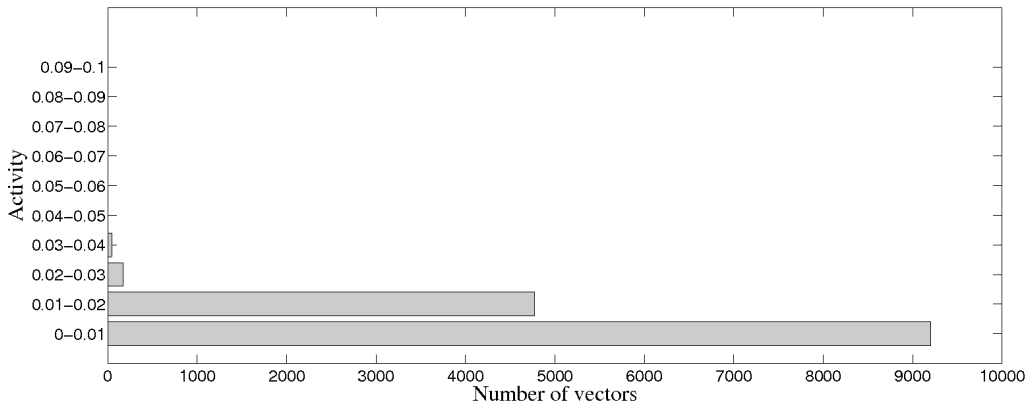
It is evident from Table 10.2 that high values of test time reduction can be achieved in large circuits. The reduction in test time varies from 0% with patterns causing very high activity to 50% with patterns causing no activity.

When external vectors are used to test a chip, an ATPG tool is usually used to generate the vectors. The deterministic patterns generated by the ATPG tool usually have very few care bits. The remaining bits known as don't care bits can be filled using a number of heuristics [3]. If the don't care bits are filled such that they have the minimum number of transitions possible, a large reduction in test time can be seen by employing the dynamic control of scan clock method.

This is illustrated using the ISCAS89 benchmark circuit, s38584 as an example. The ATPG tool from Synopsys, TetraMAX was used to generate vectors for the circuit. The tool was used to generate two sets of vectors, one with no don't care bits comprising of 961 vectors and the other with don't care bits comprising of 14196 vectors. Figure 10.2 shows the activity distribution of the two vector sets. It can be seen that the vector set with no don't care bits has an activity factor of around 0.5 and the vector set with don't care bits has a very low activity factor of around 0.01. The don't care bits in the second set of vectors



(a) Without don't care bits



(b) With don't care bits

Figure 10.2: Distribution of activity factor for test vectors of s38584 circuit.

were filled using the minimum transition heuristic in [3]. The reduction in test time achieved for both test vector sets are shown in Table 10.3.

In a typical test set, the ATPG initially generates random test vectors that would have activity factors around 0.5. However, once the easy to detect faults are detected, a deterministic approach is used to generate the vectors. These vectors have very low activity factors. Thus, the total vector set would be a combination of vectors with and without don't care bits. Therefore, the test time reduction achieved with dynamic control of scan frequency would lie between the values achieved for the individual test vector sets. For instance, a vector set comprising 10% random vectors and 90% deterministic vectors would lead to a 40.7% reduction in test time. Similarly, a vector set with 50% of each type would

result in 31% reduction and that with 90% random vectors and 10% deterministic vectors would result in 21.2% reduction in test time in s38584 circuit.

The results indicated in the first two sections of this chapter are for circuits whose test vectors are assumed to have a peak activity factor of 1. This assumption eliminates the need to simulate the test vectors to find the peak activity factor. It reduces hardware overhead since the activity at the scan-out of the scan chain need not be monitored and also since a regular counter can be used as opposed to the up-down counter used in the model proposed for circuits with test vectors whose peak activity factors are lower than 1. However, the peak activity factor need not always be 1. It has been found through mathematical analysis as discussed in the next section that the peak activity factor during test is usually around 0.65 in large circuits and hence, it is important to analyze the reduction in scan-in time achieved in such circuits.

10.3 Mathematical Analysis on t512505 Benchmark Circuit

In order to estimate the reduction in scan-in time achieved with the model proposed for dynamic scan clock frequency control in circuits with peak activity factors lower than 1, the t512505 ITC02 benchmark circuit was chosen. This specific circuit was chosen because it is large enough to employ 512 different scan clock frequencies since it has 76714 scan flip-flops.

The pattern sets of various large benchmark circuits were studied to analyze trends in peak activity factors. The mean value of peak activity factor (α_{peak}) in these pattern sets was found to be around 0.57 and the standard deviation (σ) was around 0.025. The plot of normal distribution for these values is shown in Figure 10.3.

The value of mean + 3σ was found to be around 0.65. This indicates that the probability that the peak activity factor of the test patterns of a circuit would lie below 0.65 is 99.7%. Therefore, the peak activity factor for the t512505 circuit was set at 0.65. The pattern sets generated by TetraMAX ATPG for large benchmark circuits were analyzed and it was found that the peak activity factor in these test vectors never exceeded 0.65.

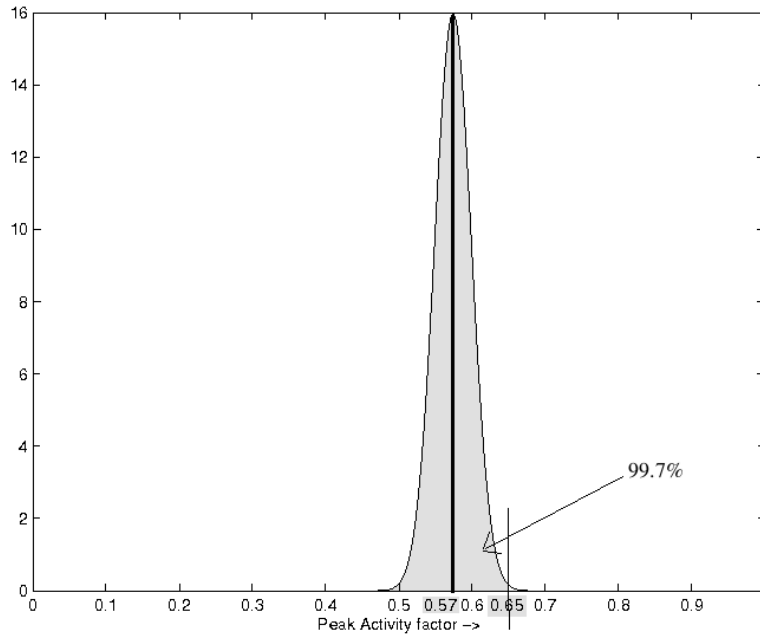


Figure 10.3: Normal Distribution Curve for α_{peak}

A C program was written to confirm the validity of this value. The program was written to generate 100,000 random vectors each consisting of 1000 random bits. The random bits were generated to have an activity factor close to 0.5. This was achieved by generating a random floating point number which was compared against the value 0.5. If the random floating point number was lesser than 0.5, a bit that was different from the previous bit was generated. If the random floating point number was greater than 0.5, a bit that had the same value as the previous bit was generated. Thus, random vectors with an average activity factor of 0.5 (chosen to mimic vectors captured in scan chain from combinational circuitry) were generated. The peak activity factor of these 100,000 vectors was found to be 0.57. Thus, the value 0.65 is a safe assumption for peak activity factor of a large circuit.

It is important to note that the value of 0.65 for peak activity factor can be used only for large circuits having flip-flop numbers in the range of a few hundreds. For smaller circuits with flip-flop numbers in the order of a few tens, the peak activity factor was found to be 1.

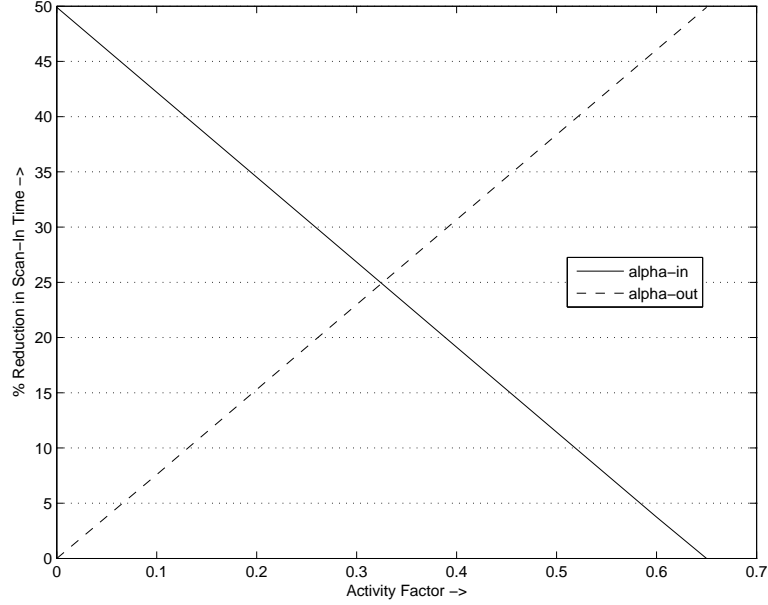


Figure 10.4: Variation of reduction in scan-in time with α_{in} and α_{out} for $\alpha_{peak} \neq 1$

10.3.1 Circuits with Peak Activity Factors lesser than 1

The reduction in scan-in time achieved in the t512505 circuit when the dynamic scan clock frequency control model proposed for circuits with α_{peak} lower than 1 was estimated through mathematical analysis. The results are listed in Table 10.4.

It can be seen from Table 10.4 that when the activity factor of the scan-out vector (α_{in}) is greater than or equal to the activity factor of the captured vector (α_{out}), there is no reduction in scan-in time. The frequency is increased only when the number of non-transitions in the scan chain increases. However, when $\alpha_{in} > \alpha_{out}$ the number of non-transitions (as counted by the counter) never increases and hence the scan-in is carried out at the starting frequency which is the frequency employed when dynamic scan clock frequency control is not implemented. Thus, the reduction in scan-in time is 0% in such cases.

A graph was plotted to study the variation of reduction in scan-in time with variation in α_{in} and α_{out} . Figure 10.4 shows the variation of scan-in time reduction with variation in α_{in} when α_{out} is fixed at 0.65 and the variation of scan-in time reduction with variation in α_{out} when α_{in} is fixed at 0.

Figure 10.4 indicates that scan-in time reduction is higher for lower values of α_{in} and for higher values of α_{out} . This can be explained from the perspective of number of non-transitions in the scan chain. If α_{in} is low, the number of non-transitions entering the scan chain is high and if α_{out} is high, the number of non-transitions leaving the scan chain is low. Thus, the net number of non-transitions in the scan chain is high and hence the higher reduction in scan-in time.

10.3.2 Externally Tested Circuits

The reduction in scan-in time achieved in the t512505 circuit when the dynamic scan clock frequency control model proposed for externally tested circuits, for which scan clock frequency is stored in ATE, was estimated through mathematical analysis. The results are listed in Table 10.5.

A graph was plotted to study the variation of reduction in scan-in time with variation in α_{in} and α_{out} . Figure 10.5 shows the variation of scan-in time reduction with variation in α_{in} when α_{out} is fixed at 0 and the variation of scan-in time reduction with variation in α_{out} when α_{in} is fixed at 0.

Figure 10.5 indicates that scan-in time reduction is higher for lower values of α_{in} and α_{out} . When α_{in} is low, more non-transitions enter the scan chain and hence, the counter counts faster. Thus, the reduction in scan-in time is higher. When α_{out} is low, the frequency at which scan-in begins is low since the frequency is predetermined based on the activity factor of the captured vector. Thus, there is a larger reduction in scan-in time.

It is evident that this method employing pre-determined scan clock frequencies performs extremely well. This is due to the availability of information about the number of transitions or non-transitions present in the scan chain in every cycle. This increases the efficiency of the dynamic scan clock frequency control technique and hence results in very high reduction in scan-in time.

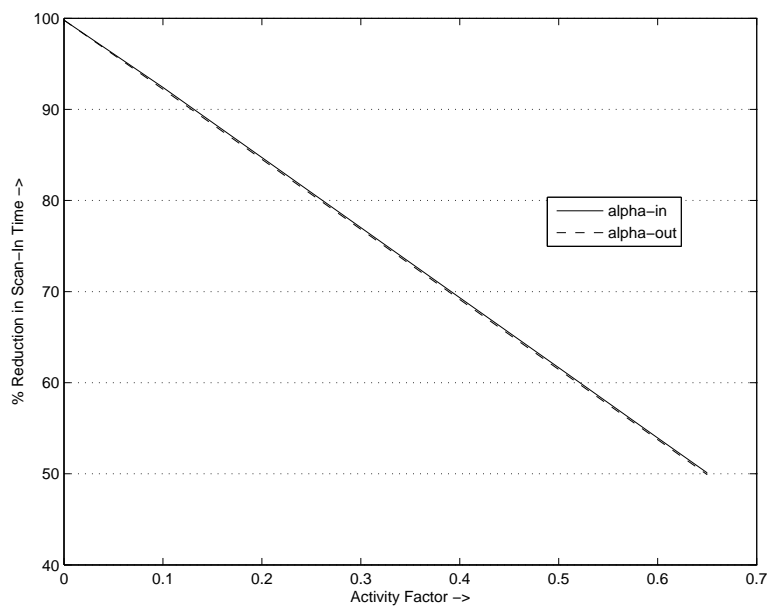


Figure 10.5: Variation of reduction in scan-in time with α_{in} and α_{out} using Pre-Determined Scan-In-Start Frequencies

Table 10.1: Reduction in Test Time in ISCAS89 Circuits - Single Scan Chain and Tested with BIST

Circuit	Number of scan flip-flops	Number of frequencies	Reduction in time (%)	Increase in area (%)
s27	8	2	7.49	14.72
s298	23	4	14.57	16.25
s344	35	4	13.48	15.06
s349	35	4	13.81	13.38
s382	30	4	13.20	12.24
s386	20	4	15.25	15.29
s400	30	4	13.18	11.36
s420	35	4	13.81	13.02
s444	30	4	13.18	11.07
s510	32	4	14.30	7.14
s526	30	4	13.18	11.12
s526n	30	4	13.15	11.34
s641	78	4	13.15	11.81
s713	77	4	12.88	11.86
s820	42	4	13.20	10.69
s832	42	4	13.23	11.10
s838	67	4	13.51	11.73
s953	68	4	13.83	10.60
s1196	46	4	13.24	10.65
s1238	46	4	13.24	10.64
s1423	96	4	13.60	8.77
s1488	33	4	12.61	10.25
s1494	33	4	12.56	10.34
s5378	263	4	13.03	6.65
s9234	286	4	14.01	5.82
s13207	852	8	19.00	3.98
s15850	761	8	18.97	3.23
s35932	2083	8	18.74	2.55
s38417	1770	8	18.83	3.14
s38584	1768	8	18.91	2.13

Table 10.2: Reduction in Test Time in ITC02 circuits

Circuit	Number of scan flip-flops	Number of frequencies	Test time reduction (%)		
			$\alpha \approx 0$	$\alpha = 0.5$	$\alpha \approx 1$
u226	1416	8	46.68	18.75	0
d281	3813	16	46.74	21.81	0
d695	8229	32	48.28	23.36	0
h953	5586	32	48.32	23.38	0
g1023	5253	32	48.19	23.32	0
f2126	15593	64	49.15	24.18	0
q12710	26158	128	49.45	24.53	0
p22810	29006	128	49.52	24.57	0
p34392	23005	128	49.53	24.57	0
p93791	96916	512	49.72	24.81	0
t512505	76714	512	49.85	24.87	0
a586710	41411	256	49.73	24.77	0

Table 10.3: Reduction in Test Time in s38584 Circuit

Without don't care bits		With don't care bits	
Number of patterns	Reduction in time (%)	Number of patterns	Reduction in time (%)
961	18.8	14196	43.14

Table 10.4: Reduction in Test Time in t512505 Circuit

α_{in}	α_{out}							
	0	0.1	0.2	0.3	0.4	0.5	0.6	0.65
0	0	7.59	15.29	22.98	30.67	38.36	46.06	49.9
0.1	0	0	7.59	15.29	22.98	30.67	38.36	42.21
0.2	0	0	0	7.59	15.29	22.98	30.67	34.52
0.3	0	0	0	0	7.59	15.29	22.98	26.83
0.4	0	0	0	0	0	7.59	15.29	19.13
0.5	0	0	0	0	0	0	7.59	11.44
0.6	0	0	0	0	0	0	0	3.75
0.65	0	0	0	0	0	0	0	0

Table 10.5: Reduction with Pre-Determination of Scan-In-Start Frequency

α_{in}	α_{out}							
	0	0.1	0.2	0.3	0.4	0.5	0.6	0.65
0	99.8	92.21	84.52	76.83	69.13	61.44	53.75	49.9
0.1	92.41	84.71	76.83	69.13	61.44	53.75	46.06	42.21
0.2	84.71	77.02	69.33	61.44	53.75	46.06	38.36	34.52
0.3	77.02	69.33	61.64	53.94	46.06	38.36	30.67	26.83
0.4	69.33	61.64	53.94	46.25	38.56	30.67	22.98	19.13
0.5	61.64	53.94	46.25	38.56	30.87	23.17	15.29	11.44
0.6	53.94	46.25	38.56	30.87	23.17	15.48	7.79	3.75
0.65	50.1	42.41	34.71	27.02	19.33	11.64	3.94	0

Chapter 11

Conclusion

A scheme to reduce test application time by dynamically increasing the scan clock frequency was proposed. The test power is held below the allowed power limit by controlling the activity per unit time. The per cycle scan activity is monitored dynamically to speed up the scan clock for low activity cycles without exceeding the specified peak power budget. The activity monitor, frequency control block, reset generator and synchronizer are implemented either as on-chip hardware or off-chip hardware or through pre-simulated and stored test data in externally tested circuits. When the frequency is controlled by hardware, a handshake protocol controls the rate of test data flow between the ATE and DUT.

The implementation of the dynamic control of scan frequency in circuits tested by both BIST and ATE were discussed. The BIST implementation requires an activity monitor on-chip that triggers a frequency divider when the activity in the scan chain falls below a certain value. The frequency divider steps up the frequency every time the scan chain activity becomes low enough. In the case of circuits tested with ATE, asynchronous handshake protocols are used for communication between ATE and the circuitry when hardware is used to control the scan clock frequency. The use of ATE allows information about the activity factor of the vectors in the scan chain to be utilized during test. The activity factor of the vectors can be used to determine the frequency at which scan operations should be carried out. This information is stored in the ATE and used for dynamic control of scan clock frequency.

This method achieved reduction in test times on all ISCAS89 benchmark circuits, with relatively low area overhead, and without exceeding the allowed power limit. A test time reduction of about 19% was achieved using test-per-scan BIST system with an area overhead

of 2-3%. For full scan s38584, the dynamic scan clock control reduced the test time by 19% when fully specified ATPG vectors were used and by 43% for vectors with don't cares.

An analysis on ITC02 benchmark circuits showed a test time reduction of 50% when scan vectors with very low activity ($\alpha \approx 0$) were used. When scan vectors with moderate activity ($\alpha = 0.5$) were used, a test time reduction of 25% was observed.

It was found that reduction in scan-in time varies with the variation of activity factors in both the scan-in and scan-out vectors when the technique is implemented in circuits with peak activity factors lower than 1.

The technique performs better on larger circuits since the number of frequencies that can be chosen for such circuits is high, and the reduction in test time increases with the increase in number of chosen frequencies. The area overhead also reduces with an increase in circuit size. Also, when the input test vectors have low activity factor, the reduction in test application time is higher. Don't cares in deterministic patterns obtained from ATPGs are filled in such a way that the number of transitions in the vector is minimum [50]. Also, techniques to generate BIST patterns with low transition densities [63] are available. This technique would perform well for such patterns.

Bibliography

- [1] V. D. Agrawal, K. T. Cheng, D. D. Johnson, and T. Sheng Lin, "Designing Circuits with Partial Scan," *Design and Test of Computers*, pp. 8–15, Apr 1988.
- [2] V. D. Agrawal, C. R. Kine, and K. K. Saluja, "A Tutorial on Built-In Self-Test, Part 1: Principles," *IEEE Design and Test of Computers*, pp. 73–82, Mar 1993.
- [3] N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, and A. Virazel, "Minimizing Peak Power Consumption during Scan Testing: Test Pattern Modification with X Filling Heuristics," *International Conference on Design and Test of Integrated Systems in Nanoscale Technology*, pp. 359–364, Sep 2006.
- [4] I. Bayraktaroglu and A. Orailoglu, "Concurrent Application of Compaction and Compression for Test Time and Data Volume Reduction in Scan Designs," *IEEE Transactions on Computers*, pp. 1480–1489, Nov 2000.
- [5] I. Bayraktaroglu and A. Orailoglu, "Test Volume and Application Time Reduction through Scan Chain Concealment," *Design Automation Conference*, pp. 151–155, May 2001.
- [6] I. Bayraktaroglu and A. Orailoglu, "Decompression Hardware Determination for Test Volume and Time Reduction through Unified Test Pattern Compaction and Compression," *VLSI Test Symposium*, pp. 113–118, Apr–May 2003.
- [7] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "A Gated Clock Scheme for Low Power Scan Testing of Logic ICs or Embedded Cores," *Asian Test Symposium*, pp. 253–258, Nov 2001.
- [8] Y. Bonhomme, P. Girard, C. Landrault, and S. Pravossoudovitch, "Power Driven Chaining of Flip-flops in Scan Architectures," *International Test Conference*, pp. 796–803, Dec 2002.
- [9] Y. Bonhomme, T. Yoneda, H. Fujiwara, and P. Girard, "An Efficient Scan Tree Design for Test Time Reduction," *European Test Symposium*, pp. 174–179, May 2004.
- [10] F. Brglez, D. Bryan, and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits," *International Symposium on Circuits and Systems*, pp. 1929–1934, May 1989.
- [11] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits*. Springer, 2000.
- [12] K. M. Butler, J. Saxena, A. Jain, T. Fryars, J. Lewis, and G. Hetherington, "Minimizing Power Consumption in Scan Testing: Pattern Generation and DFT Techniques," *International Test Conference*, pp. 355–364, Oct 2004.
- [13] S. T. Chakradhar, A. Balakrishnan, and V. D. Agrawal, "An Exact Algorithm for Selecting Partial Scan Flip-Flops," *Journal of Electronic Testing*, pp. 83–93, Oct 2007.
- [14] S. Chakravarty and V. P. Dabholkar, "Two Techniques for Minimizing Power Dissipation in Scan Circuits during Test Application," *Asian Test Symposium*, pp. 324–329, Nov 1994.

- [15] A. Chandra and K. Chakrabarty, "Frequency-Directed Run-Length (FDR) Codes With Application to System-on-A-Chip Test Data Compression," *VLSI Test Symposium*, pp. 42–47, Apr–May 2001.
- [16] A. Chandra and K. Chakrabarty, "Reduction of SoC Test Data Volume, Scan Power and Testing Time Using Alternating Run-Length Codes," *IEEE International Conference on Computer Aided Design*, pp. 673–678, Aug 2002.
- [17] A. Chandra and K. Chakrabarty, "Combining Low-Power Scan Testing and Test Data Compression for System-on-a-Chip," *Design Automation Conference*, pp. 166–169, May 2005.
- [18] K. T. Cheng and V. D. Agrawal, "A Partial Scan Method for Sequential Circuits with Feedback," *IEEE Transactions on Computers*, pp. 544–548, Apr 1990.
- [19] H. Cheung and S. K. Gupta, "A BIST Methodology for Comprehensive Testing of RAM with Reduced Heat Dissipation," *International Test Conference*, pp. 386–395, Oct 1996.
- [20] R. M. Chou, K. K. Saluja, and V. D. Agrawal, "Power Constraint Scheduling of Tests," *International Conference on VLSI Design*, pp. 271–274, Jan 1994.
- [21] E. Corno, P. Prinetto, M. Rebaudengo, and M. S. Reorda, "A Test Pattern Generation Methodology for Low Power Consumption," *VLSI Test Symposium*, pp. 453–457, Apr 1998.
- [22] F. Corno, M. Rebaudengo, M. S. Reorda, G. Squillero, and M. Violante, "Low Power BIST via Non-Linear Hybrid Cellular Automata," *VLSI Test Symposium*, pp. 29–34, Apr–May 2000.
- [23] V. Dabholkar, S. Chakravarty, I. Pomeranz, and S. M. Reddy, "Techniques for Minimizing Power Dissipation in Scan and Combinational Circuits During Test Application," *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, pp. 1325–1333, Dec 1998.
- [24] W. J. Dally and J. W. Poulton, *Digital Systems Engineering*. Cambridge University Press, 1998.
- [25] R. A. Frohwerk, "Signature Analysis: A New Digital Field Service Method," *Hewlett Packard Journal*, pp. 2–8, May 1977.
- [26] S. Gerstendörfer and H. J. Wunderlich, "Minimized Power Consumption for Scan-based BIST," *International Test Conference*, pp. 77–84, Sep 1999.
- [27] R. Ginosar, "Fourteen Ways to Fool your Synchronizer," *International Symposium on Asynchronous Circuits and Systems*, pp. 89–96, May 2003.
- [28] P. Girard, "Survey of Low-Power Testing of VLSI Circuits," *IEEE Design and Test of Computers*, pp. 80–90, May–Jun 2002.
- [29] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "A Test Vector Inhibiting Technique for Low Energy BIST Design," *VLSI Test Symposium*, pp. 407–412, Apr 1999.
- [30] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "A Test Vector Ordering Technique for Switching Activity Reduction during Test Operation," *Great Lakes Symposium on VLSI*, pp. 24–27, Mar 1999.
- [31] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "Low Power BIST Design by Hypergraph Partitioning: Methodology and Architectures," *International Test Conference*, pp. 652–651, Oct 2000.

- [32] P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, J. Figueras, S. Manich, P. Teixeira, and M. Santos, "Low Energy BIST Design: Impact of the LFSR TPG Parameters on the Weighted Switching Activity," *International Symposium on Circuits and Systems*, pp. 110–113, May–Jun 1999.
- [33] P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, and H. J. Wunderlich, "A Modified Clock Scheme for a Low Power BIST Test Pattern Generator," *VLSI Test Symposium*, pp. 306–311, Apr–May 2001.
- [34] P. Girard, C. Landrault, S. Pravossoudovitch, and D. Severac, "Reducing Power Consumption during Test Application by Test Vector Ordering," *International Symposium on Circuits and Systems*, pp. 296–299, May–Jun 1998.
- [35] D. Gizopoulos, N. Krantitis, A. Paschalis, M. Psarakis, and Y. Zorian, "Low Power/Energy BIST Scheme for Datapaths," *VLSI Test Symposium*, pp. 23–28, Apr–May 2000.
- [36] A. Hertwig and H. J. Wunderlich, "Low Power Serial Built-In Self-Test," *European Test Workshop*, pp. 49–53, May 1998.
- [37] T. C. Huang and K. J. Lee, "An Input Control Technique for Power Reduction in Scan Circuits During Test Application," *Asian Test Symposium*, pp. 315–320, Nov 1999.
- [38] V. Iyengar and K. Chakrabarty, "Precedence-Based, Preemptive, and Power-Constrained Test Scheduling for System-on-a-Chip," *VLSI Test Symposium*, pp. 368–374, Apr–May 2001.
- [39] K. Jaramillo and S. Meiyappan, "10 Tips for Successful Scan Design: Part One," *EDN*, pp. 67–74, Feb 2000.
- [40] W. J. Lai, C. P. Kung, and C. S. Lin, "Test Time Reduction in Scan Designed Circuits," *Electronic Design Automation Consortium*, pp. 489–493, Feb 1993.
- [41] K. J. Lee, T. C. Haung, and J. J. Chen, "Peak-Power Reduction for Multiple-Scan Circuits during Test Application," *Asian Test Symposium*, pp. 453–458, Dec 2000.
- [42] S. Y. Lee and K. K. Saluja, "An Algorithm to Reduce Test Application Time in Full Scan Designs," *International Conference on Computer-Aided Design*, pp. 17–20, Nov 1992.
- [43] S. Y. Lee and K. K. Saluja, "Test Application Time Reduction for Sequential Circuits with Scan," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1128–1140, Sep 1995.
- [44] S. Manich, A. Gabarro, M. Lopez, J. Figueras, P. Girard, L. Guiller, C. Landrault, S. Pravossoudovitch, P. Teixeira, and M. Santos, "Low Power BIST by Filtering Non-Detecting Vectors," *European Test Workshop*, pp. 165–170, May 1999.
- [45] B. Pouya and A. Crouch, "Optimization Trade-offs for Vector Volume and Test Power," *International Test Conference*, pp. 873–881, Oct 2000.
- [46] J. Rajski and J. Tyszer, *Arithmetic Built-In Self-Test for Embedded Systems*. Prentice Hall PTR, 1998.
- [47] J. Rajski, J. Tyszer, M. Kassab, and N. Mukherjee, "Test Pattern Compression for an Integrated Circuit Test Environment," *USA Patent, Serial No.6,327,687*, p. Oct, Dec 2001.
- [48] J. Rajski, J. Tyszer, M. Kassab, N. Mukherjee, R. Thompson, H. Tsai, A. Hertwig, N. Tamarapalli, G. Mrugalski, G. Eide, and J. Qian, "Embedded Deterministic Test for Low Cost Manufacturing Test," *International Test Conference*, pp. 301–310, Dec 2002.

- [49] E. M. Rudnick and J. H. Patel, "A Genetic Approach to Test Application Time Reduction for Full Scan and Partial Scan Circuits," *International Conference on VLSI Design*, pp. 288–293, Jan 1995.
- [50] R. Sankaralingam, R. R. Oruganti, and N. A. Touba, "Static Compaction Techniques to Control Scan Vector Power Dissipation," *IEEE VLSI Test Symposium*, pp. 35–40, Apr–May 2000.
- [51] R. Sankaralingam, B. Pouya, and N. A. Touba, "Reducing Power Dissipation During Test using Scan Chain Disable," *VLSI Test Symposium*, pp. 319–324, Apr–May 2001.
- [52] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreepakash, and M. Hachinger, "A Case study of IR-Drop in Structured At-Speed Testing," *International Test Conference*, pp. 1098–1104, Sep–Oct 2004.
- [53] J. Saxena, K. M. Butler, and L. Whetsel, "An Analysis of Power Reduction Techniques in Scan Testing," *International Test Conference*, pp. 670–677, Oct–Nov 2001.
- [54] P. Shanmugasundaram and V. D. Agrawal, "Dynamic Scan Clock Control for Test Time Reduction Maintaining Peak Power Limit," *VLSI Test Symposium*, May 2011.
- [55] P. Shanmugasundaram and V. D. Agrawal, "Dynamic Scan Clock Control in BIST Circuits," *International Conference on Industrial Electronics*, Mar 2011. submitted.
- [56] C. Stroud, *A Designer's Guide to Built-In Self-Test*. Springer, 2002.
- [57] E. E. Swartzlander, Jr., "A Review of Large Parallel Counter Designs," *IEEE Computer Society Annual Symposium on VLSI*, pp. 89–98, Feb 2004.
- [58] N. A. Touba, "Survey of Test Vector Compression Techniques," *IEEE Design and Test of Computers*, pp. 294–303, Apr 2006.
- [59] H. C. Tsai, S. Bhawmik, and K. T. Cheng, "An Almost Fullscan BIST Solution - Higher Fault Coverage and Shorter Test Application Time," *International Test Conference*, pp. 1065–1073, Oct 1998.
- [60] S. Wang and S. K. Gupta, "ATPG for Heat Dissipation Minimization during Test Application," *International Test Conference*, pp. 250–258, Oct 1994.
- [61] S. Wang and S. K. Gupta, "ATPG for Heat Dissipation Minimization for Scan Testing," *Design Automation Conference*, pp. 614–619, Jun 1997.
- [62] S. Wang and S. K. Gupta, "DS-LFSR: A New BIST TPG for Low Heat Dissipation," *International Test Conference*, pp. 848–857, Nov 1997.
- [63] S. Wang and S. K. Gupta, "LT-RTPG: A New Test-Per-Scan BIST TPG for Low Heat Dissipation," *International Test Conference*, pp. 85–94, Sep 1999.
- [64] L. Whetsel, "Adapting Scan Architectures for Low Power Operation," *International Test Conference*, pp. 863–872, Oct 2000.
- [65] X. Zhang, K. Roy, and S. Bhawmik, "POWERTEST: A Tool for Energy Conscious Weighted Random Pattern Testing," *International Conference on VLSI Design*, pp. 416–422, Jan 1999.
- [66] Y. Zorian, "A Distributed BIST Control Scheme for Complex VLSI Devices," *VLSI Test Symposium*, pp. 4–9, Apr 1993.