NAIST-IS-DD0361041

Doctoral Dissertation

Studies on Power Constrained Test Techniques for VLSI Circuits

Zhiqiang You

October 31, 2005

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Zhiqiang You

Thesis committee: Professor Hideo Fujiwara, (Supervisor)
Professor Minoru Ito, (Co-supervisor)
Associate professor Michiko Inoue, (Co-supervisor)

Studies on Power Constrained Test Techniques for VLSI Circuits*

Zhiqiang You

Abstract

Due to the chip density increasing drastically, power consumption becomes one of the most important factors of very-large-scale integration (VLSI) design. Furthermore, power and energy consumption of digital systems are considerably higher in test mode than in normal mode. This extra power consumption leads to many drawbacks, such as, decreased reliability, loss of yield, increased product cost, difficulty in performance verification, reduced autonomy of portable systems and so on. Hence, many techniques have investigated power minimization or power constraints test.

This thesis proposes several schemes and approaches to reduce peak power as well as average power at register transfer level (RT-level) and at gate level respectively. Test application time and hardware overhead are also important factors. In our approaches, we try to minimize one or both under the given power constraints to make the test more efficiently.

Focused on test power reduction of non-scan built-in self-test (BIST) register transfer level (RTL) data path, this thesis proposes three non-scan BIST schemes, formulates three problems concerning the schemes, and introduces three power-constrained design for testability (DFT) algorithms to resolve these problems.

In adjacent non-scan BIST scheme, some registers are enhanced to test registers so that each functional module can be tested by the test registers connected with the module directly or only through multiplexers. Though this scheme

* Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0361041, October 31, 2005.

achieves short test application time, hardware overhead is very high. To overcome this problem, in the techniques of our laboratory, test pattern generators (TPGs) and response analyzers (RAs) are placed only at primary inputs (PIs) and primary outputs (POs) respectively, and test patterns and test responses are transferred along paths in the data paths. We call this BIST scheme boundary non-scan BIST scheme. We also propose a more general BIST scheme that covers the above two schemes, adjacent non-scan BIST scheme and boundary non-scan BIST scheme. Generally, TPGs and RAs can be placed not only at the boundary of the data path but also inside of the data path. Any register inside the data path can be a candidate to be augmented to a TPG or an RA. We call this BIST scheme non-scan BIST scheme.

We formulate three problems employing the above testabilities satisfying peak power limit to minimize test application time, hardware overhead and both respectively. Three power-constrained DFT approaches are given to solve them. The first algorithm is about adjacent non-scan BIST scheme that is intended for short test application time. The second algorithm uses a boundary non-scan BIST scheme that focuses on achieving a low hardware overhead. This scheme, therefore, is more efficient in reducing the hardware overhead than previous methods. The third algorithm is based upon a general non-scan BIST scheme that explores possible trade-offs between hardware overhead and test application time under power constraints, rather than consider only one such factor, as previous published power-constrained methods do.

Focused on test power reduction of the circuits at gate level, this thesis also proposes a low power scan test scheme and formulates a problem based on this scheme. In this scheme the flip–flops are grouped into N scan chains. At any time, only one scan chain is active during scan test. Therefore, both average power and peak power are reduced compared with conventional full scan test methodology. To resolve this formulated problem, a tabu search–based approach is introduced to minimize test application time. In this approach we handle the information during deterministic test efficiently. For various benchmark circuits, this approach drastically reduces both average power and peak power dissipation at a little longer test application time.

Keywords:

design for testability, RTL data path, built-in self-test, low power testing, test scheduling , full scan testing, scan chain disable

List of Publications

Journal Papers

1. Zhiqiang You, Ken-ichi Yamaguchi, Michiko Inoue, Jacob Savir and Hideo Fujiwara, "Power-constrained test synthesis and scheduling algorithms for non-scan BIST-able RTL data paths," *IEICE Trans. on Information and Systems*, Vol. E88-D, No. 8, pp.1940-1947, Aug. 2005.

2. Zhiqiang You, Tsuyoshi Iwagaki, Michiko Inoue and Hideo Fujiwara, "A low power deterministic test using scan chain disable technique," *IEICE Trans. on Information and Systems* (To appear).

International Conferences (Reviewed)

1. Zhiqiang You, Michiko Inoue and Hideo Fujiwara, "On the non-scan BIST schemes under power constraints for RTL data paths," *IEEE 4th Workshop on RTL and High Level Testing (WRTLT'03)*, pp. 14–21, Nov. 2003.

2. Hao Wu, Zhiqiang You, Michiko Inoue and Hideo Fujiwara, "Test length minimization under power constraints for combinational circuits," *IEEE 4th Workshop on RTL and High Level Testing (WRTLT'03)*, pp. 125–127, Nov. 2003.

3. Zhiqiang You, Ken-ichi Yamaguchi, Michiko Inoue, Jacob Savir and Hideo Fujiwara, "Power-constrained test scheduling for RTL datapaths of non-scan BIST schemes," *Proceedings of the IEEE 13th Asian Test Symposium (ATS'04)*, pp. 32–39, Nov. 2004.

4. Zhiqiang You, Tsuyoshi Iwagaki, Michiko Inoue and Hideo Fujiwara, "A low power deterministic test using scan chain disable technique," *IEEE 6th Workshop on RTL and High Level Testing (WRTLT'05)*, pp. 184–191, July 2005.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Motivation

Electronic equipment systems are almost everywhere in our life. The key components of the systems are integrated circuits (ICs). However, with the complexity of VLSI technology increasing drastically the challenge of design and testing electronic systems has grown rapidly over the several decades. In these days, power consumption becomes one of the most important factors of VLSI design due to the continuous growth in power requirements. Since high heat causes high temperature, if the temperature increase is excessive, it may causes many problems, such as decreasing the reliability of ICs, loss the yield, increasing product cost, difficulty in performance verification, reducing autonomy of portable systems, and so on. Many mothodologies about low power design have proposed to avoid the extra power consumption.

To overcome the challenge of test, many DFT methodologies, such as scan design and BIST, which can loosely be defined as changes to a given circuit design that help decrease the overall difficulty of testing have been proposed. However, these DFT methodologies give the opposite solutions to save power. Power and energy consumption of digital systems are considerably higher in test mode than in normal mode. The main reason of high power consumption is that the unrelation values are applied during test which is avoided by the design. Hence, many techniques have investigated power minimization or power constraints test.

This thesis focuses on test power reduction. It proposes power constraints

methodologies of non-scan BIST RTL data paths and full scan circuits respectively shown as following order.

1.2. Thesis Organization

The thesis is organized as follows. Chapter 2 gives some basic concepts and techniques concerning this thesis. It firstly describes two DFT methods, full scan design and non-scan BIST. Thereafter, it gives some concepts about RTL, such as data path, controller and RTL circuit. Previous works in our laboratory that achieve low hardware overhead for non-scan BIST RTL data paths are described. Finally, it shows some low power testing techniques for non-scan BIST RTL data paths and circuits at gate level.

Chapter 3 describes three power constraints test methodologies for non-scan BIST RTL data path. First, it presents the concept of the data path digraph. Then, three testabilities and problems to minimize test application time, hardware overhead or both under power constraints are described. After that, it shows an example to illustrate the formulated problems. Next, it introduces three power constained approaches to perform the test satifying the given constaints. Finally, in order to demonstrate the effectiveness of the approaches, experimental results for various circuits are reported.

Chapter 4 is devoted to a low power deterministic test using scan chain disable technique. At first, it presents a low power scan test scheme. Thereafter, the test flow is given. Then, it shows a problem based on this scheme. After that, a flip-flops and test cubes grouping algorithm is described. Finally, experimental results for various benchmark circuits are displayed.

Finally, Chapter 5 concludes this thesis with the outline of the main accomplishments and future works.

1.3. Contributions of This Thesis

This thesis turns out four basic contributions to power-constrained test techniques for VLSI circuits.

First, this thesis proposes three schemes for RTL data paths, called adjacent non-scan BIST scheme, boundary non-scan BIST scheme, and (general) non-scan BIST scheme. Each of them has advantages itself. Adjacent non-scan BIST scheme is intended for short test application time; boundary non-scan BIST scheme is aimed to low hardware overhead; non-scan BIST scheme is proposed to explore trade-offs between hardware overhead and test application time.

Second, this thesis formulates three problems, and introduces three power constraints approaches to resolve them. The first algorithm that is for adjacent non-scan BIST scheme achieves short test application time. The second algorithm focuses on achieving a low hardware overhead based on a boundary non-scan BIST scheme. The third algorithm is based upon a general non-scan BIST scheme that explores possible trade-offs between hardware overhead and test application time under power constraints, rather than consider only one such factor.

Third, this thesis presents a low power scheme at gate levle. In this schame, at any time only one scan chain is active among N scan chains during scan test. Hence, both average power and peak power are reduced significantly compared with those of conventional full scan test methodology.

Fouth, this thesis also formulates a problem based on the low power scheme, proposes a flip-flops and test cubes grouping algorithm that targets a short test application time.

Chapter 2

Low Power Testing

This chapter describes some basic concepts and previous works concerning this thesis.

2.1. Introduction

A digital circuit is faulty, if after exercising a sequence of test patterns the responses are different with the correct (expected) one. With the complexity of VLSI increasing, the difficulty of testing is increasing. There are many DFT methodologies that have been proposed to decrease the overall difficulty of testing. This chapter will introduce two DFT methodologies. To evaluate the quality of a methodology, there are five main test parameters. They are *fault coverage* (*fault efficiency*), *test application time*, *hardware overhead*, *test power*, *performance degradation*. We utilize these test parameters to assess the methods throughout this thesis.

2.2. DFT Techniques

There are many kinds of DFT techniques. In this thesis, we only describe the techniques used or concerned with this thesis.

2.2.1 Full Scan Test

The main idea in full scan design is to achieve full controllability and observability for flip-flops by adding some logic, such as wires and multiplexers (MUXes), between them, and a test mode to the circuit, such that when the circuit is in this mode flip-flops functionally form one or more scan chains. Using the test mode, all flip-flops can be set to any desired values by shifting those logic values into the scan chain. Similarly, the values of flip-flops are observed by shifting the contents of the scan chain out. Thus, the inputs and outputs of these scan chains are made into POs and PIs respectively. To distinguish with the PIs and POs in the circuits, they are called as pseudo-primary inputs (PPIs) and pseudo-primary outputs (PPOs) respectively. All flip-flops can be set or observed in a time that equals the number of flip-flops in the longest scan chain. In Chapter 4, we will compare the proposed method to the case that only forms one single scan chain. In this case, the test application time is a function of the number of flip-flops; the more flip-flips in the circuit, the longer the testing time will be.

2.2.2 Non-Scan BIST

Since automatic test equipment (ATE) is extremely expensive and its cost is expected to grow in the future as the number of chip pins and chip's frequency increasing, BIST has been one of the most important DFT methods, where parts of the circuit are used to test the circuit itself by employing on chip TPG and RA. TPG can be one of the follows, linear feedback shift register (LFSR), Built-In Logic-Block Observations (BILBOs) [1], or concurrent BILBOs (CBILBOs) [2]. And, an RA usually uses multiple-input signature registers (MISRs), BILBOs or CBILBOs. When the circuit is in the test mode, TPG generates test patterns to the circuit under test (CUT) to differentiate the faulty and fault-free circuits, and an RA evaluates the responses of the CUT.

According to the manner of applying test patterns, BIST methods can be classified into scan BIST and non-scan BIST. This thesis focuses on non-scan BIST methodologies where test patterns are applied to the CUT every clock cycles.

In non-scan BIST methodologies, the functional registers are enhanced to test

regisiters, such as LFSRs, BILBOs, CBILBOs and MISRs, which generate test patterns and/or analyze test responses in test mode. The test patterns and test responses are propagated along the functional paths of the circuit.

2.3. Test Methodologies on Non-Scan BIST RTL Circuits

2.3.1 RTL Circuits

Digital systems are designed using a modular approach to overcome the difficulties, such as that of specification the state table. The modules are constructed as registers, multiplexers, functional modules, and control logic. A data path [3] consists of hardware elements and lines. Hardware elements, in this context, include PIs, POs, registers (Rs), MUXes, and functional modules (Ms) that have any number of input ports and one output port. Since the multiplexing function can be embedded within an M, we will use the term M in this wider sense of its capability (including multiplexing). Input patterns enter the circuit through the PIs, and exit through the POs. Input values enter into a hardware element through its input ports, and exit through its output port. For any given data path, we assume that every non-constant input port of any M has at least one path from some PI, and every output port of any M has at least one path to some PO. Data flow transfers inside of data path under the control of control logic. A data path and a controller (control logic) compose a RTL circuit shown in Fig. 2.1.

2.3.2 Test Methodologies on Non-Scan BIST RTL Circuits

Non-scan BIST is a promising approach that can realize at-speed testing with a short application time. The techniques in [4]–[8] enhance some registers to test registers such as BILBOs or CBILBOs so that each module can be tested by test registers connected with the module directly or only through multiplexers.

However, in these techniques, in order that all the modules can be tested

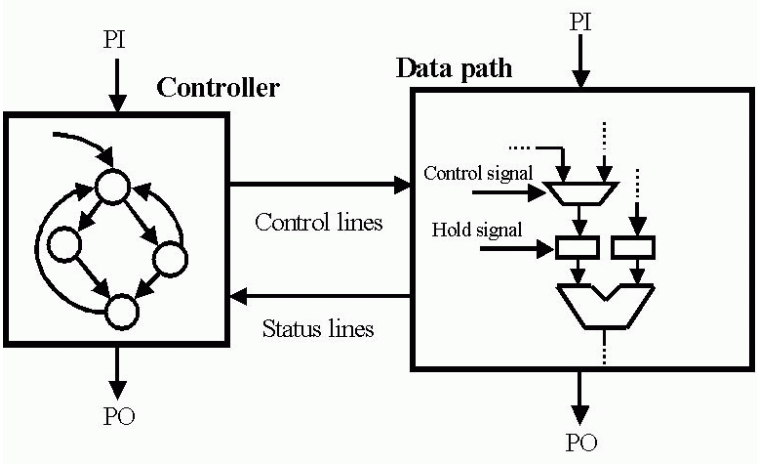


Figure 2.1. An RTL circuit.

directly or only through MUX(es) connected test registers, too many registers are modified to test registers, and hence, hardware overhead is very high. Moreover, the excessive power dissipation during these BIST schemes constitutes a considerable problem in some applications.

2.3.3 The Methodologies in Our Laboratory

To reduce hardware overhead, Masuzawa et al. [3] propose a BIST scheme for RTL data paths. In this scheme, TPGs and RAs are placed only at PIs and POs respectively, and test patterns and test responses are transferred along paths in the data paths. [3] also proposes single-control testability for RTL data paths and presents a DFT method based on this testability. To realize the testability, the DFT elements are test MUXes (T_MUXes) and *thru* functions [9], shown in Fig. 2.2, which propagate a logic value from an input to the output of a functional module without any change. The single-control testability guarantees that, for each combinational module, test patterns generated by TPGs can be fed into the module at consecutive system clocks and its test responses can be consecutively propagated to an RA. Thus, the single-control testability guarantees high fault coverage. However, this BIST method suffers from a long test application time

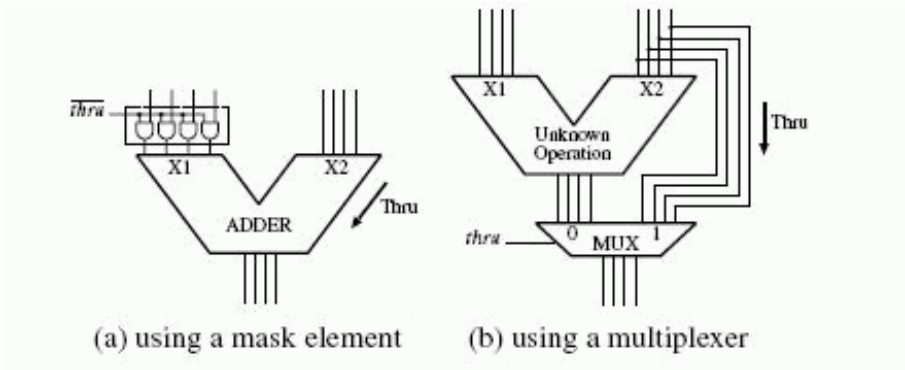


Figure 2.2. Realizations of a thru function [9].

because in this method only a single module is tested at a time.

The approaches in [10, 11] improve the method in [3] by introducing concurrent testing, exploiting time division between existing TPGs, so that two different input ports of a module can share the same TPG. However, these previous works did not consider the problem of power dissipation during test. More specifically, when these methods try to excite a single module under test (MUT), and observe its test response, multiple modules and registers, that are not adjacent to the data path of the MUT, also dissipate power. As a result, the accumulated power dissipation is quite high. For some applications, this high power dissipation is unacceptable. Furthermore, hardware overheads in these methods are way too high. As we show in Chapter 3, lower hardware overheads are achievable, while still limiting the power dissipation during test.

2.4. Low Power Techniques

Power and energy consumption of digital systems are considerably higher in test mode than in normal mode [12]. Hence, numerous techniques have proposeed to reduce power during test. The following subsections show the formula of the power dissipation in CMOS circuits and some lowe power techniques concerning our methods.

2.4.1 Power Dissipation in CMOS Digital Circuits

The power dissipation of a CMOS circuit is given by the following equation [13]. $P_{circuit}$=$0.5\times(V_{DD}^2/T_{CYC})\sum_{forallgatesG} N_G \times C_{load}$, where, V_{DD} is the supply voltage, T_{CYC} is the global clock period, N_G is the total number of gate output transitions ($0 \rightarrow 1$ and $1 \rightarrow 0$), C_{load} is the load capacitance.

2.4.2 Techniques at Gate Level

Test power dissipation depends directly on the global clock frequency and switching transitions of the CUT. Therefore, decreasing both the clock frequency and the switching activity can reduce test power. The method [14] reduces average power in sequential circuits by decreasing the test clock frequency. The main disadvantages of this method are that the test application time increases as the clock frequency decreases and the peak power cannot be reduced.

The main direction to reduce power is to reduce the switching activity in the circuit. Various techniques have been proposed to reduce switching activity during test. The methodologies in [15]–[18] employ test vector or scan cell reordering technique where test vectors in a test set or scan cells for a test set are reordered for minimal power consumption. The basic idea of these techniques is to find a new order of the test set such that the correlation between consecutive test patterns is increased. The methodologies in [19],[20] also explore the correlation between consecutive test patterns by filling each don't care bit in the test cubes with appropriate value 0 or 1.

There are some methods [21]–[27] that reduce power consumption by using scan chain disabling techniques. Whetsel [21] and Saxena et al. [22] proposed two schemes that divide scan chain into multiple sub-scan chains, and at any time only one sub-scan chain is activated during scan shifting to reduce power consumption. The power during scan shifting is reduced to $1/N$, where N is the number of sub-scan chains. However, these methods did not consider peak power dissipation. During capture cycle, all the sub-scan chains are active. Therefore, the peak power dissipation may be very high. The scheme in [23] employs two different clocks that work at half of the initial frequency such that these two scan chains are operated at different clock cycle during scan shifting. Though this methodology

reduces average power efficiently, it suffers the same disadvantage with those in [21],[22]. Bhattacharya et al. [24] proposed a double-tree scan architecture where the scan flip-flops are organized as two complete k-level binary trees whose leaf nodes are merged pair-wise. In this scheme, during scan shifting only the scan flip-flops in a scan path are active. The average power is reduced significantly. But the power dissipation during capture cycle cannot be reduced. In [25], scan chains are grouped into two sets while the given test set T is divided into two subsets. For one test subset except for the test group boundaries, only one-group scan chains are active. Hence, this method can reduce power consumption. However, this method did not consider peak power dissipation. [26] extended the scheme in [25] by exploring a more general architecture. In this method, the test vectors are partitioned into some groups. For each group, while test vectors in the group are applied, a subset of the scan chains are disabled using a programmable scan chain disabling mechanism. This method is effective to reduce average power dissipation. However, to preserve the fault coverage some test vectors may be applied while all the scan chains are active. Therefore, the peak power reduction may not be guaranteed. To reduce peak power dissipation, Basturkmen et al. [27] proposed a low-power pseudo-random BIST methodology for scan circuits. In this method, the scan chains are partitioned into N groups. At any time, only the scan chains in a group are active throughout scan cycles and capture cycle. Therefore, both average and peak power dissipation are reduced. However, this method, which suffers from a very long test application time, is not efficient for deterministic test.

2.4.3 Techniques about Non-Scan BIST RTL Data Paths

Nicolici et al. [7, 8] propose a test synthesis and scheduling algorithm under power constraints for BISTed RTL data paths. Here, test synthesis is the process of allocating test hardware to each module. They define necessary power dissipation as the power dissipated in tested modules and test registers for them, and useless power dissipation as the power dissipated in untested modules and registers which are not used as test registers. Their approach saves power dissipation during applying tests and shifting out of test responses by considering both necessary and useless power dissipation. However, the main objective of this approach is to

eliminate useless power dissipation. It may be inefficient to reduce test application time and hardware overhead. In practice, modules also can share TPGs, i.e., it is possible to test different type modules concurrently. Therefore, we can explore more efficient test synthesis and scheduling algorithm to resolve these problems.

Chapter 3

On Non-scan BIST Schemes under Power Constraints for RTL Data paths

3.1. Introduction

We call the scheme in [4]–[8] where some registers are enhanced to test registers so that each module can be tested by test registers connected with the module directly or only through multiplexers as *adjacent non-scan BIST* scheme.

[3] also gives a scheme, where TPGs and RAs are placed only at PIs and POs respectively, and test patterns and test responses are transferred along paths in the data paths. We call this BIST scheme *boundary non-scan BIST* scheme.

In this chapter [28]–[30], we propose a more general BIST scheme that covers the above two schemes, adjacent non-scan BIST scheme and boundary non-scan BIST scheme. Generally, TPGs and RAs can be placed not only at the boundary of the data path but also inside of the data path. Any register inside the data path can be a candidate to be augmented to a TPG or an RA. We call this BIST scheme *non-scan BIST* scheme. A new design for testability and test scheduling algorithm under power constraints need to be investigated.

In this chapter, we also introduce three power-constrained DFT algorithms. The first is for adjacent non-scan BIST scheme intend for short test application time (referred to in the chapter as "problem 1"). The second algorithm uses

a boundary non-scan BIST scheme that focuses on achieving a low hardware overhead (referred to in the chapter as "problem 2"). This scheme, therefore, is more efficient in reducing the hardware overhead than previously described methods. The third algorithm is based upon a general non-scan BIST scheme that explores possible trade-offs between hardware overhead and test application time under power constraints (referred to in this chapter as "problem 3"), rather than consider only one such factor, as previous published power-constrained methods do.

This chapter is organized as follows. Section 2 introduces the concept of the data path digraph. Section 3 defines the testabilities concerning the schemes, and outlines the problems to be solved. Section 4 shows an example to illustrate the formulated problems. Section 5 addresses the power constraints for problem 1 and shows algorithms for performing the test while meeting the given constraints. Sections 6 and 7 addresses the same issues for problems 2 and 3 respectively. Section 8 reports on some experimental results using our proposed schemes. Section 9 concludes with a brief summary.

3.2. The Data Path Digraph

Similar to the definition in [3], we define a data path digraph $G = (V, A)$ as follows.

- $V = V_H \cup V_{IN} \cup V_{OUT}$, where
  - V_H is the set of nodes that correspond to all hardware elements in the data path. Let $V_H = V_M \cup V_R \cup V_{OTH}$, where, V_M, V_R and V_{OTH} are the set of nodes which represent modules, registers and other hardware elements respectively.
  - V_{IN} is the set of nodes which correspond to all input ports in the data path, and
  - V_{OUT} is the set of nodes which correspond to all output ports in the data path.
- $A = A_1 \cup A_2 \cup A_3$, where

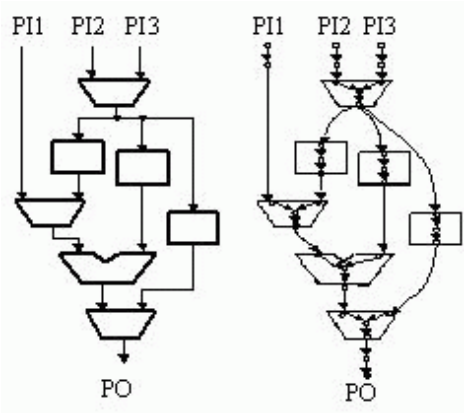


Figure 3.1. A data path and its associated digraph.

- A_1={$(x, y) \in V_{OUT} \times V_{IN}$ | output port x is connected to input port y by a line } ,
- A_2={$(y, u) \in V_{IN} \times V_H$ | y is an input port of u } , and
- A_3={$(u, x) \in V_H \times V_{OUT}$ | x is an output port of u } .

Note that in a digraph, each PI or PO corresponds to a pair of nodes, and not to a single node. For example, Fig. 3.1 shows a data path fragment with its associated digraph.

An input port $i_j \in V_{IN}$ is an input port of a node $u_M \in V_M$, such that they are connected together by an arc in A_2. We denote the arc outgoing from node u_M by e_M; and the head node of an arc e by h_e. The sequential depth of a path is the number of register elements along the path.

3.3. Problem Formulation

3.3.1 Definitions

We define the following three concepts of testability concerning the BIST schemes.

Definition 1 A data path is *adjacent non-scan BIST-able* if each module M in the data path can be tested as follows.

There exists a TPG for each input port of M, and an RA for the output port of M such that

(I-i). Each port of M is connected with the corresponding TPG or RA directly or only through multiplexers.

(I-ii). Test patterns generated by the TPGs and test responses of M can be propagated to the corresponding input ports and RA concurrently without conflict of control signals.. ■

In this definition, the control signals include select signals for multiplexers, mode signals for BILBOs and CBILBOs, and hold inputs for registers, .

Definition 2 A data path is *boundary non-scan BIST-able* if each module M in the data path can be tested as follows.

There exists a TPG for each input port of M, and an RA for the output port of M such that

(II-i). TPGs and RAs are placed only at PIs and POs respectively.

(II-ii). There are paths that propagate test patterns generated by the TPGs to the input ports of M, and test responses of M to the corresponding input ports of the RA, concurrently, without any conflict of control signals.

(II-iii). For any two input ports of any M, test patterns can either be propagated to these from two different TPGs, or from the same TPG, provided it has different sequential depths leading to these two ports.

■

Notice that we allow test patterns to be propagated through a module M using its *thru input function*, if such a function exists. Thus, a module with a thru input can be operated in a transparent mode to pass test patterns generated upstream to other components downstream.

In Definition 2, the control signals include select signals for MUXes; hold inputs for registers, and thru inputs for functional modules.

Definition 3 A data path is *non-scan BIST-able* if each module M in the data path can be tested as follows.

There exists a TPG for each input port of M, and an RA for the output port of M, such that properties (III-i), (II-ii), and (II-iii) in Definition 2&3 hold.

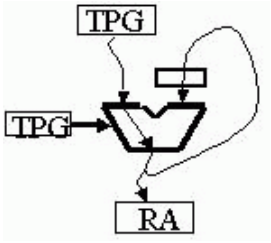


Figure 3.2. A module with a type 3 path.

(III-i). TPGs and RAs can be placed at PIs and POs respectively, and any register inside the data path can be a candidate for augmentation into a TPG or an RA. ■

In boundary non-scan BIST, and non-scan BIST schemes, we categorize the different types of *control paths* that propagate test patterns from TPGs to the inputs of a module under test. We distinguish, therefore, between the following cases:

Type 1: A control pattern can be chosen such that no two input ports of M share a TPG.

Type 2: Some input ports share a TPG with paths of different sequential depths.

Type 3: Some input ports share a TPG, and the control path for one of its input ports passes through another input and output ports of this same module (See Fig. 3.2).

An *observation path* propagates test responses from the output port of a module to an RA. In the sequel, we will refer to both control paths and observation paths simply as test paths.

In the above three BIST schemes, two or more modules can be tested concurrently if test patterns generated by the TPGs and test responses of the modules are propagated into the corresponding input ports and RAs concurrently without conflict of control signals and any two modules do not share the same register or primary output as their RAs.

3.3.2 Problem Description

We formulate the following three problems. Let f_H(HOH,TAT) be a hardware-intensive cost function, such that $f_H(h_1, t_1) < f_H(h_2, t_2)$ if $h_1 < h_2$ or ($h_1 = h_2$ and $t_1 < t_2$). Similarly, let f_T(HOH,TAT) be a time-intensive cost function, such that $f_T(h_1, t_1) < f_T(h_2, t_2)$ if $t_1 < t_2$ or ($t_1 = t_2$ and $h_1 < h_2$). The "hardware" argument reflects hardware overhead (HOH), and the "time" argument of the function reflects test application time (TAT).

Problem 1 Minimize the test application time of a given data path an adjacent non-scan BIST, and atest scheduling algorithm, under a given power constraint. Stating it more formally,

Given:

- **Input:** a data path and peak power dissipation limit P_{max}.

Task:

- **Output:** an adjacent non-scan BIST-able data path, a test schedule that satisfys P_{max}, and that achieves the
- **Objective:** minimization of f_T(HOH,TAT), i.e. minimize test application time. ■

In adjacent non-scan BIST scheme, for each module, only registers adjacent to the module can be augmented to TPGs and RAs. BILBO and CBILBO are candidate register architectures to TPGs and RAs. If necessary, hold functions are added for registers.

Problem 2 Minimize the hardware overhead of a given data path under a boundary non-scan BIST, and a test scheduling algorithm, subject to a given power constraint. Stating it more formally,

Given:

- **Input:** a data path and peak power dissipation limit P_{max}.

Task:

- **Output:** a boundary non-scan BIST-able data path, a test schedule that satisfies P_{max}, and that achieves the
- **Objective:** minimization of f_H(HOH,TAT), i.e. minimize hardware overhead. ■

In order to achieve this task we are allowed to add DFT elements, such as LFSRs, MISRs, T_MUXes, hold functions for registers, and thru-functions for functional modules.

Problem 3 Given a design parameter α, design a non-scan BIST-able data path, and a test scheduling algorithm, under a given power constraint. More formally,

Given:

- **Input:** a data path, co-optimization ratio α ($0 \leq \alpha \leq 1$), and a peak power dissipation limit P_{max}.

Task:

- **Output:** a non-scan BIST-able data path, a test-schedule satisfies P_{max}, and that achieves the
- **Objective:** minimization of $\alpha \cdot \text{HOH} + k(1-\alpha) \cdot \text{TAT}$[1] . ■

In order to achieve this task, we are allowed to add DFT elements, such as BILBOs, CBILBOs, LFSRs, MISRs, T_MUXes, hold functions for registers, and thru-functions for functional modules.

We can see that Problems 1 and 2 are special cases of Problem 3. The hardware overhead, power dissipation for a module and test application time for this case is between the above two cases.

[1] k is a unit conversion constant with value $|k|$=1.

3.4. An Examples of the Formulated Problems

In this section we will present an example to illustrate the formulated problems. The following example shows trade-offs between test application time and hardware overhead among these three BIST schemes.

For example, we consider three BIST schemes for a benchmark circuit, Paulin. Here, we consider testing four functional modules Add.1, Mult.1, Mult.2 and Sub.1. The peak power constraint is assumed to be P_{max}=$100P_u$.

There are some assumptions about test application time and test power. We first treat modules of type 1 test paths. Let T_M be the test application time for a MUX, T_M= T_u, where T_u is an integer unit. We assume that the test application time of an adder (T_+), subtractor (T_-), multiplier (T_*), constant-input multiplier ($T_{*'}$), AND gate ($T_\&$), and OR gate ($T_|$) are T_+=T_-=$5T_u$, T_*=$20T_u$, $T_{*'}$=$3T_u$ and $T_\&$=$T_|$=$4T_u$, respectively. The test application time of a module with test path of either type 2, or type 3, are assumed to be T_{type2}=$1.5T_{type1}$, and T_{type3}=$2T_{type1}$, respectively. Let P_u be a standard unit of power. Using the technique in [37], we further assume that the power dissipations for MUX (P_M), AND gate ($P_\&$), OR gate ($P_|$), register (P_{Reg}), adder (P_+), subtractor (P_-), multiplier (P_*), constant-input multiplier ($P_{*'}$), BILBO (P_{BIL}), and CBILBO (P_{CBIL}), are P_M=$P_\&$=$P_|$=P_u, P_{Reg}=P_+=P_-=$5P_u$, P_*=$20P_u$, $P_{*'}$=P_{BIL}=P_{CBIL}=$10P_u$, respectively.

If we resolve this example using adjacent non-scan BIST scheme the registers R4, R5, R6 and R7 can be enhanced to BILBOs and R1, R2 can be enhanced to CBILBOs (Fig. 3.3). CBILBO1 and BILBO5 can generate test patterns for Add.1 and CBILBO1 analyses its test response. The test registers of modules Mult.1, Mult.2 and Sub.1 are BILBO5, BILBO4, BILBO6, BILBO5, CBILBO2, BILBO7 and BILBO7, CBILBO2 respectively. The power dissipation is equal to $66P_u$ and $86P_u$ for Add.1, Sub.1 and Mult.1, Mult.2 separately. Therefore, we can schedule the test under given peak power constraint by two test sessions Add.1, Sub.1, Mult.1, Mult.2. The test length is equal to $25T_u$.

If we use boundary non-scan BIST scheme, it only needs to add three thru-functions (Fig. 3.4). Module Add.1 can be test by test registers TPG1 and RA1 and choosing type 3 path. Module Sub.1 can be test by test registers TPG2 and RA2 and choosing type 3 path through the right port of Mult.1. Modules

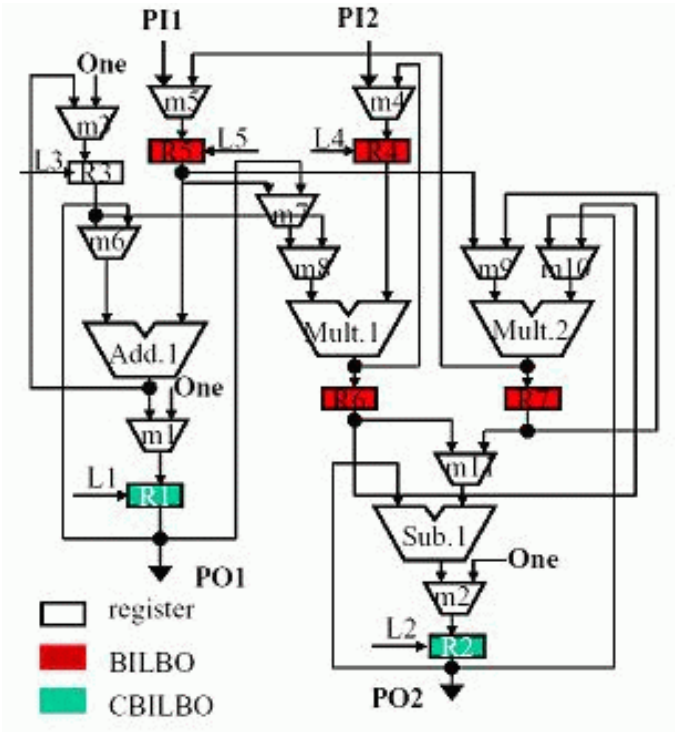


Figure 3.3. Data path example for adjacent non-scan BIST scheme.

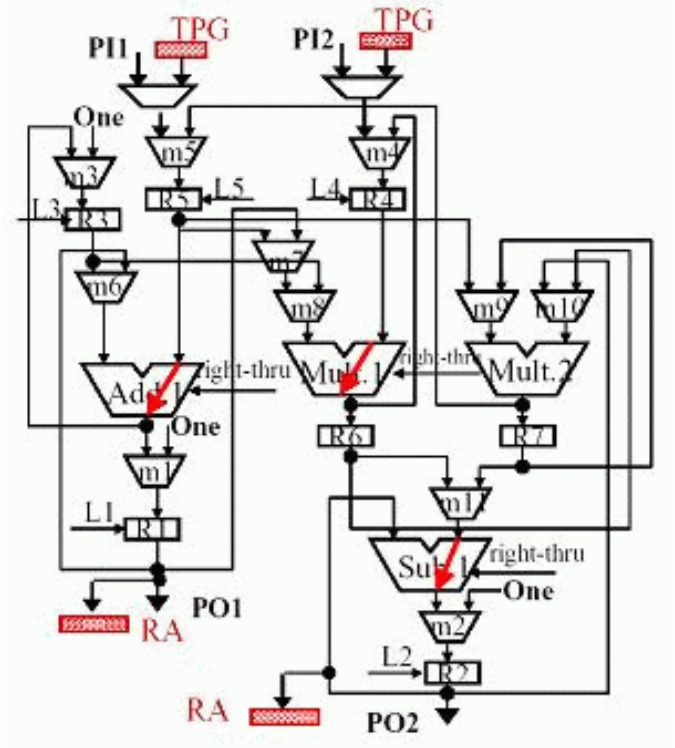


Figure 3.4. Data path example for boundary non-scan BIST scheme.

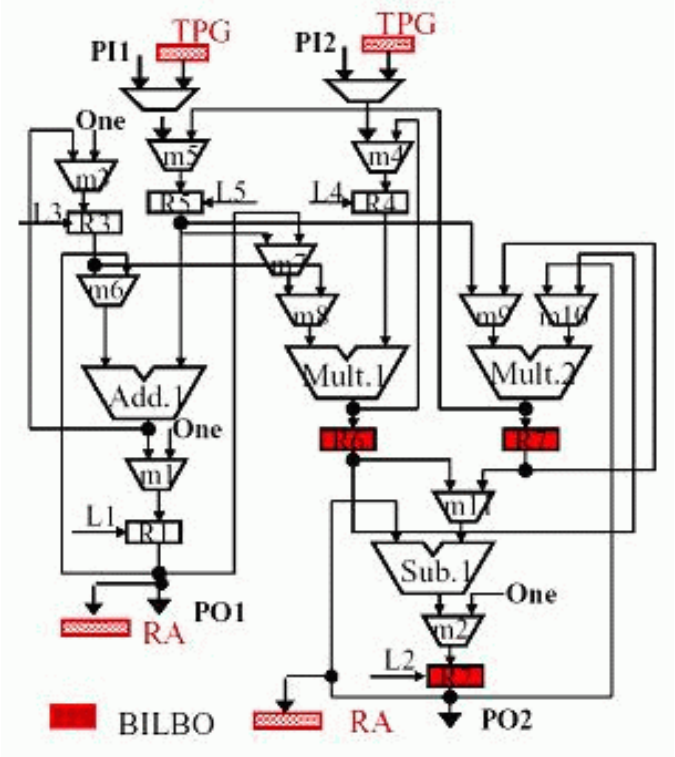


Figure 3.5. Data path example for non-scan BIST scheme.

Mult.1 can be test by test registers TPG1 and RA1 and choosing type 3 path. And module Mult.2 use TPG1, TPG2 and RA2 to test. The power dissipation is equal to $30P_u$, $80P_u$, $99P_u$ and $98P_u$ for Add.1, Sub.1, Mult.1 and Mult.2 respectively. Therefore, the test schedule under given peak power constraint is Add.1, Sub.1, Mult.1, Mult.2. The test length equals to $80T_u$.

However, if we use non-scan BIST scheme, we enhance registers R2, R6 and R7 to BILBOs (Fig. 3.5). The module Add.1 can be test by test registers TPG1 and RA1 and choosing type 3 path while the module Sub.1 can be test by test registers BILBO6 and RA2 and choosing type 3 path. The test registers of module Mult.1 and Mult.2 are TPG1, TPG2, BILBO6 and TPG1, BILBO2, BILBO7 respectively. The power dissipation is equal to $100P_u$ and $99P_u$ for Add.1, Sub.1 and Mult.1, Mult.2 separately. By considering power dissipation the test schedule

Table 3.1. Comparison among the three schemes for Paulin under the given power constraint.

	A-NS-BIST	NS-BIST	B-NS-BIST
TAT(T_u)	25	30	80
HOH	4992	1920	384

is Add.1, Sub.1, Mult.1, Mult.2. The test length is equal to $30T_u$.

Table 3.1 summarizes the results of the three schemes for Paulin with 32-bit widths. Row 3 shows the transistor equivalent as synthesized and reported by the Synopsys Design Compiler for the DFTed hardware. The table shows the trade-offs between test application time and hardware overhead among these three BIST schemes. Here, the result for non-scan BIST scheme is the one possible solution among the co-optimization ratio α. Designer can select better solution using non-scan BIST scheme according to his preference.

It notes that the last two BIST schemes also can test multiplexers which cannot be completely tested using the first BIST scheme. It makes them have more potential to get higher fault coverage. To simplify and achieve comparability we only give an example by test the function modules of the data path.

3.5. Power Constrained DFT Algorithm for Problem 1

This section proposes a power constrained test synthesis and scheduling algorithm for RTL data paths of adjacent non-scan BIST scheme.

3.5.1 Overview of the Algorithm

This subsection gives an overview of the algorithm. More details of this algorithm are showed in the following subsections.

This algorithm consists of the two stages, test synthesis and test scheduling. In the test synthesis, we allocate test registers for all modules as TPGs or RAs.

Similarly to that given in [31], we define test incompatibility for adjacent non-scan BIST scheme as follows.

Definition 4 Two modules are *incompatible* for adjacent non-scan BIST scheme, i.e. they cannot be tested simultaneously, if they have one of the following conditions.

i. One module uses a register as an LFSR while the other module uses the same register as an MISR.

ii. Both modules use the same register as an MISR. ■

The algorithm repeats the following stages until all modules are scheduled.

Phase 1. Test synthesis. In this stage, we try to allocate test registers for all modules to lower area overhead and increase test compatibility to achieve shorter test application time.

1. Allocate tentative TPGs for all unscheduled modules.
2. Allocate tentative RA for each unscheduled module if possible.

Phase 2. Test scheduling. In this stage, we schedule one test session by following steps.

1. Get the test incompatibility graph
2. Schedule a test session by using test incompatible graph
3. Refine the result
4. Update the data path.

3.5.2 Allocate Tentative TPGs for All Unscheduled Modules

To reduce hardware overhead and increase test compatibility between modules, we try to minimize the number of test registers assigned as TPGs. Let $RSET_j$ be a set of input ports of modules with which a register R_j is connected directly or only through multiplexer(s). This minimization problem is equivalent to the minimum set cover problem to find the minimum set R of registers such that

$\bigcup_{R_j \in R} RSET_j$ covers all the unscheduled modules. However, this problem is NP-complete, therefore, we propose a heuristic algorithm.

First, we select some mandatory registers as follows. If an input port of a unscheduled module in connected with only one register, we select the register. Then, we select registers one by one from the reigeter R_j whose $RSET_j$ includes the most uncovered input ports. To lower hardware overhead, it is necessary to reuse the registers which have been test registers in previous test sessions. We select the register which is modified to test register rather than other registers when all of them connect the most undecided input ports directly or only through multiplexer(s).

3.5.3 Allocate Tentative RA for Each Unscheduled Module

In this step, we try to allocate as many RAs as possible for modules under the condition that the modules can be tested in one test session.?One module needs only one register as an RA, and one register can be an RA for one module. We reduce this problem into the following the maximum weighted matching problem.

The Hungarian algorithm [32] is used to resolve this problem. The input to the algorithm is a weighted bipartite graph described below. Let G=(V, E) be the complete bipartite graph with vertex set V=$M \cup R$, where M=$\{M_1, M_2, ..., M_n\}$ is the set of modules and R=$\{R_1, R_2, ..., R_k\}$ is the set of registers which can be modified as RAs for modules. A weight function is defined as follows.

$w(i,j)$=l-$R(j)$, if there is a direct connection or a connection only through multiplexer(s) from M_i to R_j.

=-1, otherwise.

Where $R(j)$ is the number of unscheduled modules that R_j is TPG for; l is a large enough natural number so that l is great than all $R(j)$.

If there is a direct connection or a connection only through multiplexer(s) from M_i to R_j, $R(j)$ also implies the number of incompatible modules with M_i. Thus the maximal weighted matching means the minimum incompatibility. In another word, the aim of this step is to enhance test compatibility to achieve shorter test application time.

3.5.4 Get the Test Incompatibility Graph

We get the incompatibility graph from the results of the last step, where node set consists of the unscheduled modules and edges exists only between incompatible modules.

3.5.5 Schedule a Test Session Using Test Incompatible Graph

Before presenting this step a concept is necessary to definite. For modules can share TPGs, the power dissipation in LFSRs need not be counted for all modules under test.

Definition 5 *Essential power dissipation* for adjacent non-scan BIST scheme: only the power dissipated in the tested module and its MISR are essential power dissipation for adjacent non-scan BIST scheme. ■

In this step, we shall select a set of modules that are tested in one test session under power constraint. We extend the scheduling algorithm [33] that obtains a test scheduling under power constraint from a given incompatibility graph. The approach in [33] schedules modules based on necessary power dissipation. We extend it to consider useless power dissipation as well as essential power dissipation.

3.5.6 Refine the Result

Now, we have obtained a set of modules scheduled in one test session, where TPGs and an RA are assigned to each module in the session. We refine the assignment keeping compatibility of the modules in the test session. An example is given to illustrate how to refine the assignment. We consider Fig. 3.6 as an example, where R2 is assigned to an input of M1 as a TPG. If R2 has not been modified as a TPG but R4 has been modified, we can reduce hardware overhead by reassign R4 to the input port in of M. We repeat such refinement until we fail to reduce hardware overhead and useless dissipation. Finally, if the peak power dissipation is decreased, we try to add module(s) in the test session using left edge algorithm under peak power constraint.

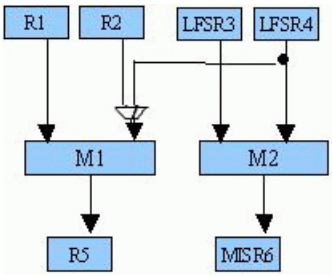


Figure 3.6. Reassign test registers.

We do the following sub steps keeping compatibility of the modules in the test session. First, if possible, we change another register as an RA for each module in the test session one by one. If the change makes to decrease hardware overhead or useless power dissipation then accept, else reject. Then, being similar as last sub step, if possible, we change another register as TPG for the input ports which can share a register as TPG until all input ports of modules in the test session are considered. If the change makes to decrease hardware overhead or useless power dissipation then accept, else reject. Finally, if the peak power dissipation is decreased, we try to add module(s) in the test session using left edge algorithm under peak power constraint.

3.5.7 Update the Data Path

In this step, we denote the selected modules by "scheduled" and enhance each register which is used in the test session as test register to a BILBO or a CBILBO.

3.6. Power Constrained DFT Algorithm for Problem 2

3.6.1 Algorithm Description

This algorithm consists of the following three phases.

Phase 1. Convert the given data path to a boundary non-scan BIST-able one utilizing the following steps:

1. Eliminate *critical arcs* for modules.
2. Add thru-functions for functional modules whenever necessary.

Phase 2. Determine the test paths for each module. If the power constraint is violated, consider adding minimum number of T_MUXes to bypass some paths to reduce power. Determine the test paths again until the modules can be tested one by one, while satisfying the power constraint.

Phase 3. Schedule the test.

3.6.2 Critical Arc Elimination

Definition 6 For a data path digraph G and an arc e, let G_e be a digraph obtained from G by deleting e. An arc e is critical for a node $u_M \in V_M$ if one of the following three cases holds (for the sake of simplicity we state the conditions for modules with two ports only):

Case 1: None of the input ports of u_M is reachable from any PI in G_e, and the sequential depth of any path from h_e to the two ports is identical.

Case 2: None of the input ports of u_M is reachable from any PI in G_e; the sequential depths of any path from h_e to the two ports are different, and no PO is reachable from u_M in G_e.

Case 3: Let u_{M_1}, u_{M_2} and u_{M_3}, be members of V_M, and let e_{M_1} and e_{M_2} be the outgoing arcs from nodes u_{M_1} and u_{M_2} respectively. Arcs e_{M_1} and e_{M_2} are critical for u_{M_3} if no PO is reachable from u_{M_3} in $G_{e_{M_1}}$ or $G_{e_{M_2}}$, and input port i_j of u_{M_3} is unreachable from any PI in $G_{e_{M_j}}$, for j=1,2, respectively.

If e is a critical arc of u_M, we say u_M is dominated by e. ■

The hardware area of a T_MUX is usually higher than that of a module-embedded thru-function. There are, however, instances where only T_MUXes can be used to establish the desired testability. These instances occur when there

is a need to eliminate critical arcs. We, therefore, consider adding a minimum number of T_MUXes into the data path only when it is necessary.

Theorem 1 If all modules have thru-functions for their input ports, a data path is boundary non-scan BIST-able if and only if (iff) there does not exist a critical arc in its associated digraph. ■

If more than one module are dominated by a critical arc, the order by which we handle these modules plays a key role in reducing the overall hardware overhead. To determine this order, we introduce notions that reflect the relationship between two dominated modules, called a *down-stream module* (DSM), and an *up-stream module* (USM).

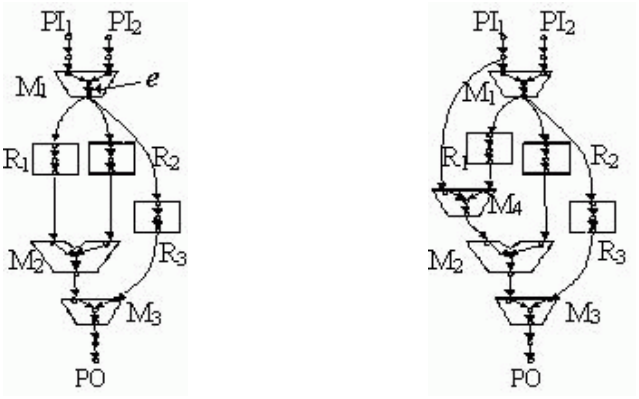
For a dominated node $u_M \in V_M$ of a data path digraph G, let $E(u_M)$ be the set of critical arcs of u_M.

Definition 7 For two dominated nodes u_M and u'_M, we say that u_M is the *up-stream module*, iff u_M is a predecessor of u'_M in the digraph G', where $G'.V = G.V$, $G'A = G.A - E(u_M) - E(u'_M)$, or conversely, we say that u'_M is the *down-stream module* (DSM) iff u'_M is a successor of u_M in the digraph G', provided the dominating critical arc does not meet the condition stated in case 1 of definition 3. ■

From the above definition, the following theorem follows.

Theorem 2 If M is the USM of M', the critical arcs of both M and M'can be eliminated by introducing a T_MUX to add a path from one PI to some other input port of M. Similarly, if M' is a DSM of M, the critical arcs of both M and M' can be eliminated by introducing a T_MUX to add a path from the output port of M' to some PO. ■

Fig. 3.7 illustrates how to eliminate a critical arc. From Definition 3, and the original data path digraph (Fig. 3.7(a)), we find that both modules, M_2 and M_3, have one critical arc e in Fig. 3.7(a). M_2 is the predecessor of M_3, in other words, M_2 is the USM of M_3. Therefore, according to Theorem 2, addition of a T_MUX (M_4, in Fig. 3.7(b)) to establish a path from PI_1 to one input port of M_2, eliminates the critical arc e for both modules. The data path digraph after adding the T_MUX for e is shown in Fig. 3.7(b).



(a)Before adding a T_MUX (b)After adding a T_MUX for e

Figure 3.7. Example of adding a T_MUX to eliminate a critical arc

The problem of adding a minimum number of T_MUXes to eliminate critical arcs is equivalent to the minimum prime-implicant covering problem, which is known to be NP-hard. We, therefore, use a greedy algorithm, where we select a dominated module that can eliminate critical arc(s) of the maximum number of dominated modules, by adding an extra path to that module. We repeat this algorithm until we eliminate all the critical edges in the system.

3.6.3 Thru-Function Addition

After adding the necessary T_MUXes, we consider adding a minimum number of thru-functions, whose hardware overhead is usually lower than that of a T_MUX, in order to achieve boundary non-scan BIST-ability. First, we add some necessary thru-functions as described in the following theorem.

Theorem 3 If there exists a module M, that is an immediate successor of another functional module M', then an addition of a thru-function to M' is needed to test M. ■

After adding the necessary thru-functions, it may still be possible that the data path in question is not boundary non-scan BIST-able. We, therefore, may need to add some more thru-functions. In Fig. 3.8 there is no critical arc. However,

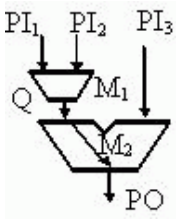


Figure 3.8. Need for adding a thru-function.

a thru function from Q to PO needs to be added in order to facilitate vector propagation through module M_2.

3.6.4 Control Paths and Observation Paths Determination

After the thru-function addition, the data path is boundary non-scan BIST-able. We now determine the control paths and observation path for each module using the shortest, power-weighted, path.

3.6.5 Bypassing Overly Power Consuming Paths

In a boundary non-scan BIST scheme, TPGs and RAs are placed only at PI and PO sites respectively. Therefore, some modules may end up having long test paths, thus dissipating an extended amount of power. If some modules have long test paths, which dissipate more power than P_{max}, we try to bypass some of them by inserting T_MUXes. In this case, if two or more modules share a portion of their test paths (sub-paths), these modules might be able to share the added bypass as well. In this stage, we search for a minimum number of common sub-paths, so that when being bypassed, the underlying modules satisfy the given power constraints. This problem is also equivalent to the minimum prime-implicant covering problem. We, therefore, use a greedy algorithm, where we always select the common sub-path such that, if bypassed, it reduces the maximum sum-of-powers for the modules involved. Finally, we add the needed T_MUXes to bypass these sub-paths so identified.

3.6.6 Test Scheduling

We proceed to obtain the test incompatibility graph defined similarly to that given in [31].

Definition 8 Two modules M_1 and M_2 are test incompatible for boundary non-scan BIST scheme, if one of the following conditions holds.

i. The observation path of M_1 is joined with the test path of M_2.

ii. The control section associated with M_1 is of type 3, and joins the test path of M_2. ■

Since modules can share TPGs and parts of control paths, the power dissipated in these LFSRs and parts of these control paths, need not be accounted for repeatedly, when considering all modules under test. We, therefore, introduce the following concept.

Definition 9 *Essential power dissipation* for boundary non-scan BIST scheme is:

i. the power consumed by the module itself and its associated observation path, if the test path of the module is either of type 1 or of type 2.

ii. the power dissipated in the tested module, its associated observation path, and its feed-around portion of the control path, if the test path of the module is of type 3. ■

For example, the hardware elements on the bold lines of Fig. 3.9 (line feeding the RA and the feedback line) dissipate essential power for the module and its type 3 path.

After bypassing the overly power-consuming sub-paths, we create the incompatibility graph. In this graph, the nodes are the tested modules, and edges only exist between incompatible modules. We extend the scheduling algorithm from [33] for concurrent testing of multiple modules. In [33] the power is evaluated as the sum of the powers consumed by the individual logic blocks. In our extended algorithm, presented here, two important features come to light:

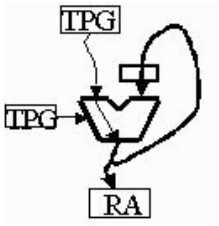


Figure 3.9. Essential power for a module with type 3 path.

a. By sharing control paths of different tested modules, we decrease the total consumed power.

b. If it so happens that two modules activate secondary paths off their main test paths, and the paths reach different ports of the same MUX, and since we cannot stop the activity at the MUX, the total power consumed is larger than the sum of the powers of their individual stand-alone paths.

The approach in [33] schedules blocks based on the "necessary" power dissipation. Here we consider "unnecessary" power dissipation, as well as essential power dissipation.

All the time complexities of the sub-algorithms in subsections 3.6.2, 3.6.3, 3.6.4, 3.6.5 and 3.6.6 are O(n^2), where n is the number of RTL elements. Therefore, the time conplexity of the approach is O(n^2).

3.7. Power Constrained DFT Algorithm (Tabu Search-Based) for Problem 3

Fig. 3.10 summarizes the tabu search-based algorithm [34]. Line 1 starts with an initial solution, taken as the solution for Problem 2. Lines 3-19 are the heart of the optimization process. For every register and functional module, we try every possible move[2] , which is not in the tabu list (lines 4-5). After a move,

[2] A move is a general term for adding/removing thru functions in a module; reconfiguring a register into a BILBO, or a CBILBO, adding a hold function to a register, or removing of some previously added hardware.

if the data path D_i is non-scan BIST-able, proceed to schedule the test (S_i). If it meets the power constraints, compute the test application time (T_i), and hardware overhead (H_i), (lines 6-9). Here, we treat the internal test registers as either PIs or POs, depending on whether they are used to generate values, or capture responses. We, then, search for a solution[3] S_k that minimizes the value of the cost function $\alpha \cdot H_i + (1-\alpha) \cdot T_i$, and set $S_{current} = S_k$,. This move is then recorded in the tabu list (line 15). If this solution turns out to be the best one so far, we set $S_{best} = S_k$. The algorithm ends when either the maximum number of iterations is reached (N_{itr1}), or the maximum number of iterations since the last obtained best solution exceeds some predetermined value (N_{itr2}).

The most complex part of this algorithm is lines 3-19 shown in Fig. 3.10. The time compexity of the sub-algorithm corresponding to lines 7-10 is O(n^2). Since the number of possible moves for a register or a module is a constant, the number of iterations corresponding to lines 4-5 is limited by $c \cdot n$, where c is a constant. Therefore, the order of the time compexity of the sub-algorithm (lines 4-12) is 3. The time complexity of the remaining part (lines 13-17) is O(n^2). The number of iterations (line 19) is limited by two constants values, N_{itr1} and N_{itr2}, that is L. Therefore, the time complexity of this approach is O($L \cdot n^3$).

3.8. Experimental Results

We have conducted experiments on the data paths of LWF[3], Paulin[35] and Tseng[36]. Table 3.2 shows the characteristics of these data paths. Columns #Pi, #Po, #R, #Mux, #M, denote the number of PIs, POs, registers, MUXes and functional modules, respectively. Columns "Bit" and "Area" denote bit-width, and the equivalent area as synthesized and reported by the Synopsys Design Compiler.

We first treat modules of type 1 test paths. Let T_M be the test application time for a MUX, $T_M = T_u$, where T_u is an integer unit. We assume that the test application time of an adder (T_+), subtractor (T_-), multiplier (T_*), constant-input multiplier ($T_{*'}$), AND gate ($T_\&$), and OR gate ($T_|$) are $T_+ = T_- = 5T_u$, $T_* = 20T_u$,

[3] A solution is a complete test scheduling with established values for TAT, HOH, and the resulting power.

Algorithm: **Power constrained test synthesis and scheduling algorithm for Problem 3 (PCTSP3)**

1. Generate an initial solution;
2. $S_{current} \leftarrow S_{init}$;
3. repeat {
4. for every register and functional module {
5. for every possible move that is not in tabu list{
6. Obtain data path D_i;
7. if D_i is non-scan BIST-able{
8. Schedule the test, get S_i;
9. If it meets power constraints, compute TAT (T_i) and HOH (H_i);
10. }
11. }
12. }
13. Find S_k for which $\alpha \cdot H_i + (1-\alpha) \cdot T_i$ is minimum;
14. $S_{current} \leftarrow S_k$;
15. Record the move into tabu list;
16. If this solution is the best so far, then
17. set $S_{best} \leftarrow S_k$;
18.}
19.until #iterations>Min$\{N_{itr1}, N_{itr2}\}$

Figure 3.10. PCTSP3 algorithm.

Table 3.2. Circuit characteristics.

Ckt	#Pi	#Po	Bit	#R	#Mux	#M	Area
LWF	2	2	32	5	5	3	6714
Paulin	2	2	32	7	11	4	36114
Tseng	3	2	32	6	7	7	23234
JWF	5	5	32	14	25	3	20373

$T_{*'}$=$3T_u$ and $T_{\&}$=$T_{|}$=$4T_u$, respectively. The test application time of a module with test path of either type 2, or type 3, are assumed to be T_{type2}=$1.5T_{type1}$, and T_{type3}=$2T_{type1}$, respectively. Let P_u be a standard unit of power. Using the technique in [37], we further assume that the power dissipations for MUX (P_M), AND gate ($P_{\&}$), OR gate ($P_{|}$), register (P_{Reg}), adder (P_+), subtractor (P_-), multiplier (P_*), constant-input multiplier ($P_{*'}$), BILBO (P_{BIL}), and CBILBO (P_{CBIL}), are P_M=$P_{\&}$=$P_{|}$=P_u, P_{Reg}=P_+=P_-=$5P_u$, P_*=$20P_u$, $P_{*'}$=P_{BIL}=P_{CBIL}=$10P_u$, respectively. The hardware overhead, in our experiments, has been determined from the Synopsys Design Compiler for DFT elements.

Tables 3.3–3.6 and Figures 3.11–3.17 display the experimental results of the Power-Constrained Test Synthesis and Scheduling algorithm for Problem 2 (PCTSP2), Problem 3 (PCTSP3), and the power-driven optimization TCSC (PTCSC) methods. TCSC is our previous methodology [11]. We have extended it here mainly in order to save power by assigning fixed values to unused control signals. Columns α, P_{max}, Pow, HOH and TAT are the co-optimization ratio, peak power dissipation limit, actual peak power dissipation, hardware overhead, and test application time, respectively. For each benchmark circuit, we employed three peak power constraints. To show the results simply, in Figures 3.11–3.17 we denote the tightest peak power constraint as power constrained degree(PCD)=1, the loosest one as PCD=3, and the in-between one as PCD=2. Notice that for a fixed P_{max}, the hardware overhead decreases with the increase of α. By the same token, the test application time increases with the increase of α. There is, therefore, a tradeoff between HOH and TAT. Notice that when P_{max} is increasing, the hardware overhead and test application time are both decreasing due to a potentially higher

Table 3.3. Experimental results for data path of LWF.

Method	α	$P_{max}(P_u)$	Pow(P_u)	HOH(%)	TAT(T_u)
PCTSP3	0	60	59	32.4	15.5
		65	65	33.4	12.0
		70	65	33.4	12.0
	0.5	60	58	14.3	23.5
		65	58	12.4	23.5
		70	68	9.1	23.5
	1	60	58	14.3	23.5
		65	58	12.4	23.5
		70	68	9.1	23.5
PCTSP2		60	60	21.0	22.5
		65	64	15.7	24.0
		70	68	9.1	23.5
PTCSC		—	69	14.3	15.0

test activity. If we relax the peak power dissipation limit, we can use this relaxation in power to schedule more modules in a given test session, or, equivalently may need less hardware to test the modules in a given test session.

In Table 3.5, for the case of α=1 and P_{max}=60, notice that PCTSP3 enjoys lesser hardware overhead than PCTSP2. This is because in the non-scan BIST scheme we can add more kinds of DFT elements that will make the approach more hardware-efficient. For cases other than α=1, the results are pretty much the same.

In Tables 3.3–3.6, when P_{max} is large enough, the hardware overheads of PCTSP2 and PCTSP3 (for α=1) are lower than that of PTCSC. This shows that our methodology is more efficient, even when there are no power constraints.

Table 3.4. Experimental results for data path of Tseng.

Method	α	$P_{max}(P_u)$	Pow(P_u)	HOH(%)	TAT(T_u)
PCTSP3	0	72	70	27.1	65.5
		82	82	29.5	44.0
		92	92	25.1	41.0
	0.5	72	70	15.4	78.0
		82	81	9.6	65.0
		92	92	8.7	59.0
	1	72	70	12.1	78.0
		82	81	9.6	65.0
		92	86	7.3	93.5
PCTSP2		72	72	12.1	76.5
		82	81	10.2	65.0
		92	92	9.3	59.0
PTCSC		—	77	11.8	103.0

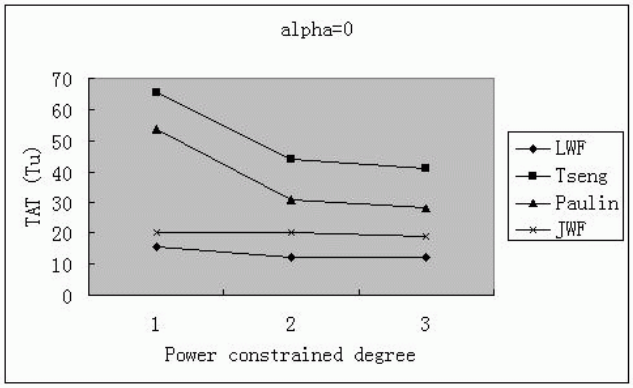


Figure 3.11. Test application time when α=0 for PCTSP3.

Table 3.5. Experimental results for data path of Paulin.

Method	α	$P_{max}(P_u)$	Pow(P_u)	HOH(%)	TAT(T_u)
PCTSP3	0	60	60	25.3	53.5
		100	99	25.1	31.0
		140	137	19.8	28.0
	0.5	60	58	7.0	72.5
		100	87	5.8	61.5
		140	114	3.1	71.5
	1	60	60	6.4	91.5
		100	100	4.9	91.5
		140	114	3.1	71.5
PCTSP2		60	58	7.9	89.0
		100	99	4.9	91.5
		140	114	3.1	71.5
PTCSC		—	112	3.4	82.0

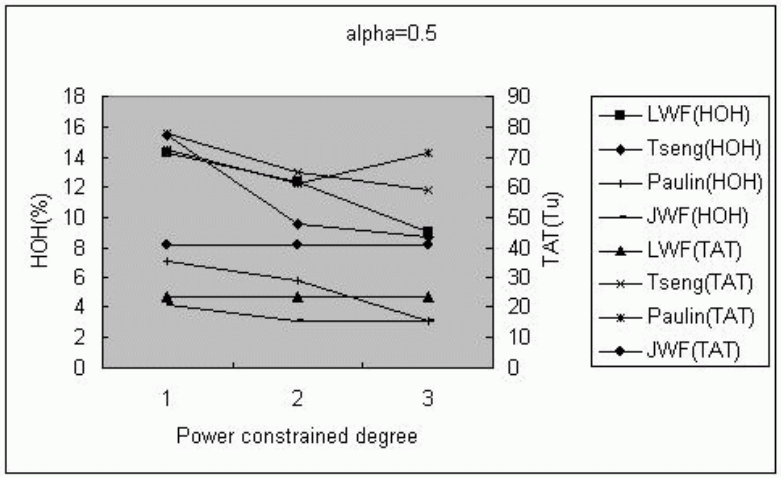


Figure 3.12. Hardware overhead and test application time when α=0.5 for PCTSP3.

Table 3.6. Experimental results for data path of JWF.

Method	α	$P_{max}(P_u)$	Pow(P_u)	HOH(%)	TAT(T_u)
PCTSP3	0	70	70	25.9	20.0
		100	95	30.2	20.0
		130	122	33.0	19.0
	0.5	70	70	4.2	41.0
		100	80	3.1	41.0
		130	80	3.1	41.0
	1	70	70	4.2	41.0
		100	80	3.1	41.0
		130	80	3.1	41.0
PCTSP2		70	70	4.2	50.0
		100	80	3.1	41.0
		130	80	3.1	41.0

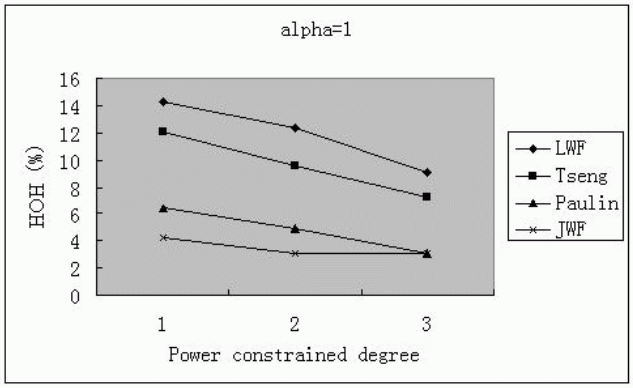


Figure 3.13. Hardware overhead when α=1 for PCTSP3.

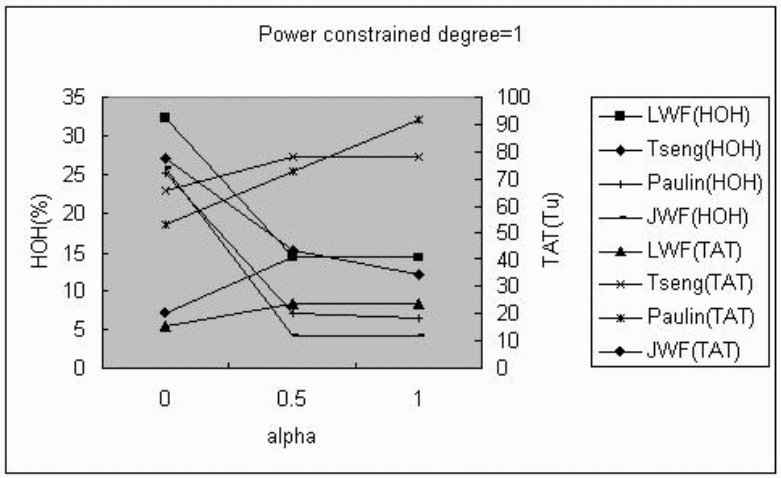


Figure 3.14. Hardware overhead and test application time when PCD=1 for PCTSP3.

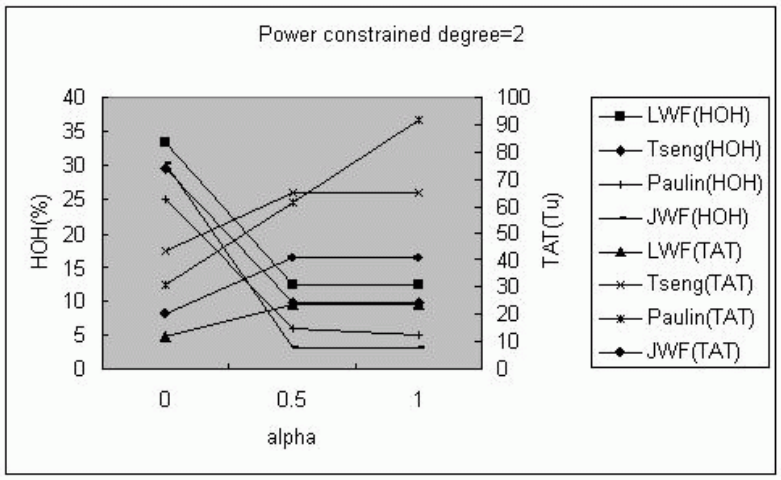


Figure 3.15. Hardware overhead and test application time when PCD=2 for PCTSP3.

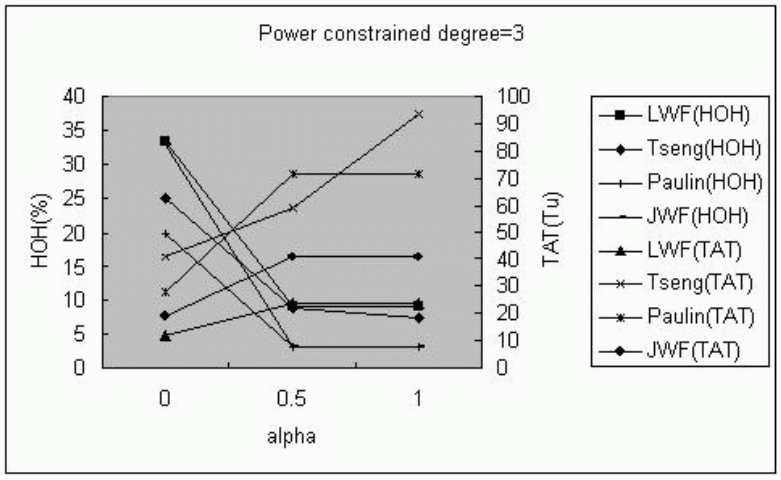


Figure 3.16. Hardware overhead and test application time when PCD=3 for PCTSP3.

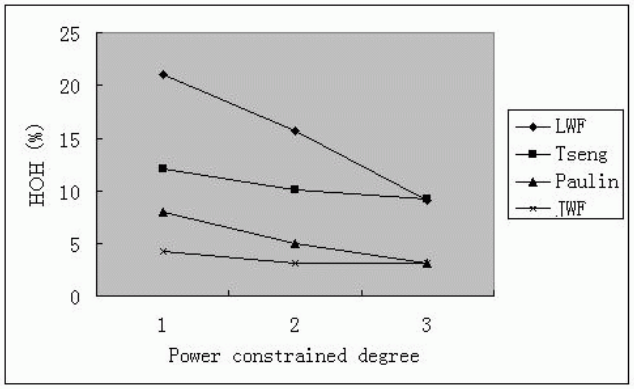


Figure 3.17. Hardware overhead and test application time for PCTSP2.

Table 3.7. Added DFT elements and their overhead figures for the data path of Paulin.

Method	α	$P_{max}(P_u)$	HOH(%)	#T_MUX	#THRU	#HOLD	#BILBO	#CBILBO
PCTSP3	0	60	25.3	0	8	0	1	6
		100	25.1	0	8	0	1	6
		140	19.8	0	7	0	2	4
	0.5	60	7.0	3	7	1	1	0
		100	5.8	2	7	0	1	0
		140	3.1	1	7	0	0	0
	1	60	6.4	3	7	0	1	0
		100	4.9	4	6	0	0	0
		140	3.1	1	6	0	0	0
PCTSP2		60	7.9	9	6	0	0	0
		100	4.9	4	6	0	0	0
		140	3.1	1	6	0	0	0

Table 3.8. Performance degradation for LWF.

Method	α	$P_{max}(P_u)$	Degrade(%)
PCTSP3	0	60	4.72
		65	4.72
		70	4.72
	0.5	60	1.90
		65	1.90
		70	1.90
	1	60	1.90
		65	1.90
		70	1.90
PCTSP2		60	1.90
		65	1.90
		70	1.90

In Table 3.7, TM stands for T_MUX; T stands for thru functions; H stands for hold-functions; B stands for BILBO, and C stands for CBILBO. For this table, in the case of a=1 (biased towards saving on hardware), more T_MUXes and thru-functions were added to share TPGs, and hence to reduce the hardware overhead. On the other hand, for the case of a=0 (biased in favor of saving TAT), more BILBOs and CBILBOs were added to achieve a short test application time. To summarize, notice that CBILBOs are efficient in reducing TAT, while T_MUXes and thru-functions are efficient in achieving a low HOH.

We evaluate the performance degradation of the DFTed circuits. Table 3.8–3.10 shows the percentages of performance degradation for LWF, Tseng and Paulin respectively. From these tables, we can see that the performance degradation is very low. They are 1.90%, 1.12% and 1.94% for LWF, Tseng and Paulin respectively except the cases of α=0 (prior time). In the cases of α=0, the performance degradation is a little high, up to 4.72% for LWF, up to 5.97% for Tseng, and up to 2.76% for Paulin. It is due to the utilization of BILBOs and (or) CBILBOs. Therefore, our approaches that uses T_MUXes, hold functions

Table 3.9. Performance degradation for Tseng.

Method	α	$P_{max}(P_u)$	Degrade(%)
PCTSP3	0	72	5.97
		82	2.37
		92	2.37
	0.5	72	1.12
		82	1.12
		92	1.12
	1	72	1.12
		82	1.12
		92	1.12
PCTSP2		72	1.12
		82	1.12
		92	1.12

for registers, and thru-functions for functional modules can achieve lower performance degradation than the DFT method employed BILBOs and/or CBILBOs.

Notice that, in this chapter we do not show the experimental results of the approach for Problem 1. This approach that was presented in a workshop is an efficient method to achieve short test application time. However, there is a problem that it is hard to compare with other methods due to the absence of circuits they used. Therefore, we put it aside for the approaches of Problems 2 and 3.

3.9. Conclusions

This chapter proposed three non-scan BIST schemes for RTL data paths and formulated DFT problems for the schemes under power constraints. This chapter also proposed three power constrained DFT algorithms based on three non-scan BIST schemes for RTL data-paths. The first algorithm is for adjacent non-scan BIST scheme. The second proposed algorithm is for a boundary non-scan BIST

Table 3.10. Performance degradation for Paulin.

Method	α	$P_{max}(P_u)$	Degrade(%)
PCTSP3	0	60	1.96
		100	2.76
		140	2.76
	0.5	60	1.94
		100	1.94
		140	1.94
	1	60	1.94
		100	1.94
		140	1.94
PCTSP2		60	1.94
		100	1.94
		140	1.94

scheme. Experimental results have shown that this method is efficient in achieving a low hardware overhead. The third algorithm is for a generic non-scan BIST scheme. We use a Tabu search algorithm to explore the solution space. Experimental results presented here show that it can co-optimize the hardware overhead, test application time, and the power dissipation. A chip designer may utilize these tradeoffs to prioritize one such parameter over the rest.

Chapter 4

A Low Power Deterministic Test Using Scan Chain Disable Technique

4.1. Introduction

In this chapter [38], [39], we present a low power deterministic test methodology for sequential circuits using scan chain disable technique. The flip-flops are grouped into N scan chains. At any time, only one scan chain is active during both scan shifting and capture cycles. Since the switch activity of logic will be confined to the fanout cone of the activated scan chain, this technique reduces peak power as well as average power dissipation.

This chapter also proposes an approach to group scan flip-flops and test cubes for minimizing test application time. In this method, we group scan flip-flops considering not only the compatibility of some bits of the test cubes but also the test information which flip-flops should capture the test response. If the test for a test vector with scan chain disabling loses fault coverage, we apply this vector again, and activate another scan chain to capture the test response. Therefore, this approach achieves a short test application time while average power and peak power reduction are guaranteed.

This chapter is organized as follows. Section 2 presents a low power scan test scheme. The test flow is given in section 3. Section 4 shows a problem based on

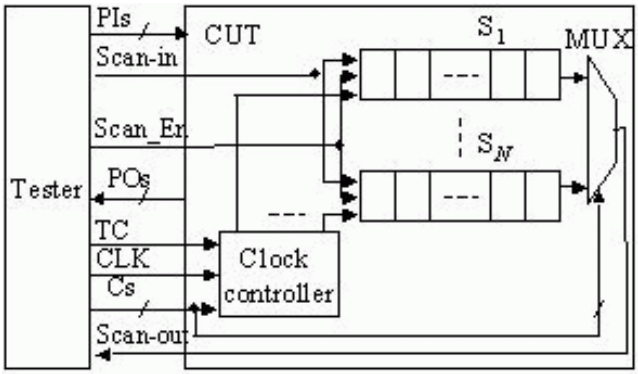


Figure 4.1. Proposed scheme.

this scheme. Section 5 describes a flip-flops and test cubes grouping algorithm to minimize the test application time. Some experimental results using our proposed approach are reported in section 6. Section 7 concludes with a brief summary and some future works.

4.2. Proposed Scheme

In this section, we present a low power scan test scheme shown in Fig. 4.1. The proposed scheme divides the flip-flops into N groups. In this chapter, we do not care of the order of flip-flops in a group. The flip-flops are lined into a scan chain according to the order of appearance in the circuit description. In this scheme, during a test mode (TC=1), a clock controller propagates a system clock CLK into only one scan chain selected by Cs. The selected scan chain shifts a test vector or captures a test response according to Scan_En. In a normal mode (TC=0), CLK is propagated into all the scan chains to enable the normal operation. When TC=1 and Scan_En=1, a tester applies Cs and test vectors to CUT. Only the CLK for one scan chain selected by Cs is not gated by the clock controller. Test vectors are shifted into the scan chain while the MUX controlled by Cs selects the activated scan chain to shift out the test responses. When Scan_En=0, we perform normal operations where all the scan chains are active in conventional full scan methodology. However, in this scheme, normal

operations are separated into the operations of capturing test response and normal capture by the extra pin TC. When TC=0, which means normal captures, CLK is propagated into all the scan chains. Whereas, CLK is propagated into only one scan chain selected by Cs when TC=1. One scan chain captures the test response of the circuit. Therefore, this scheme can reduce both average and peak power dissipation during scan test. The clock controller can be implemented by a decoder and a small number of gates. The additional hardware elements are a clock controller and a multiplexer. Thus, this scheme has a little higher hardware overhead compared with conventional full scan design.

Though this scheme treats circuits with a single clock, it also can be applicable to multiple clock domains with a little modification of the extra logic for multiple test clocks. If only a part of or all of the scan flip-flops that belong to the same clock domain are grouped into a scan chain, the skew during shift is minimized and there is no skew during capture which is very difficult to deal with. One-hot clocking scheme, where the scan chains in a clock domain are active during capture operation, efficiently deal with skew problem in multiple clock circuits. Nevertheless, it suffers from a long test application time and high power dissipation.

A test application procedure is summarized as follows. First, some vector is shifted into the scan chains according to the following way. By setting the appropriate value of Cs, the scan chains are active in turn. The bits corresponding to the activated scan chain are shifted into the scan chain. Secondly, by setting appropriate value Cs, only one scan chain captures the test response. If the next vector has the same specified values for the disabled scan chain, Cs keeps the same value. We shift in the bits of vector and shift out test response corresponding to the activated scan chain until the bits in disable scan chain of the vector to be applied are different from that of the previous vector. We continue to apply next vector as shown in the above steps until all the test vectors are applied. Finally, the test response of the last activated scan chain is shifted out.

Notice that other power reduction techniques, such as test vector reordering, scan cell reordering and minimum transition fill (MT-fill), adapt to this scheme. If we apply one or more these techniques to this scheme, both average power and peak power dissipation can be reduced significantly. This scheme can be

enhanced by employing multiple scan trees [40] so that the test application time is reduced drastically.

Though the proposed scheme has these advantages, if the fault effect of a fault cannot propagate to the activated scan chain, the fault response cannot be captured. In this case, to achieve the same fault coverage with conventional full scan testing, we should repeat to apply this test vector while one of the scan chains that can capture its fault effect turns active.

If we apply each test vector N times, and the N scan chains are active in turn among the N same vectors, the test achieves the same fault coverage with conventional full scan design. Although it reduces average power and peak power dissipation, test application time is so long. Fortunately, there are mainly two pieces of information.

i. Usually, several flip-flops capture the fault effect of a fault when a test vector is applied. It is enough to test the fault if only one of the flip-flops is active.

ii. Usually, a fault can be detected by several test vectors in a test set. We can use any one of those test vectors.

If we handle the information efficiently, the test application time can be reduced drastically. Section 5 will describe the approach in detail.

4.3. Test Flow

This section shows a procedure of test. It consists of the following five steps.

The first step is to perform automatic test pattern generation (ATPG) to generate test cubes, and do static compaction to merge the compatible test cubes together.

In the second step, we obtain the detect-capture information about which POs and/or PPOs capture the fault effect of a fault when applied one test cube. Fig. 4.2 shows a simple example of a test set with test cubes. Its detect-capture information is given in Table 4.1. The first row shows the test cubes. The second row describes which faults can be detected by the corresponding test cube. The flip-flops that can capture the corresponding fault are given in row 3.

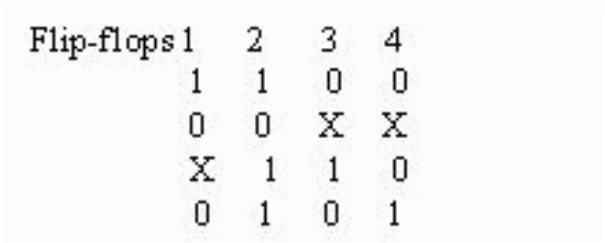


Figure 4.2. A test set with test cubes.

Table 4.1. Detect-capture information for the test cubes in Fig. 4.2.

	Cube 1					Cube 2					Cube 3			Cube 4		
Fault	1	4	6	8	9	2	4	6	7	10	3	5	7	1	3	11
Flip-flop	1,3,4	2,3	4	1	2	2,3	3	4	4	1	1,2	3	4	1,3,4	2	3

In the third step, the flip-flops are grouped into N scan chains. For each test cube, only one scan chain is set to be active. According to the detect-capture information, if a fault cannot be detected, we duplicate some test cube while a scan chain that can capture its fault effect turns active. The test cubes with the information of scan chain activity are divided into some *D-compatible* subsets where the scan chains have the same activity and the bits have non-conflicting values for the disabled scan flip-flops. The detailed information to group flip-flops and test cubes will be shown in section 5. Here, we give a simple example, as shown in Fig. 4.3 flip-flops 1 and 2 are grouped into Group 1 while flip-flops 3 and 4 are grouped into Group 2. And, the flip-flops corresponding to the bits in the rectangle field are active. For the first cube, flip-flops 1 and 2 are active. Therefore, as shown in Table 4.1, faults 1, 4, 8 and 9 can be detected. However, it loses fault coverage by only applying the first four cubes, since fault 3 is detected by cube 3 and 4 and its fault effect should be captured by flip-flops in group 1 whereas flip-flops in group 2 are active. We duplicate cube 3, and activate the flip-flops in Group 1 shown as the last cube in Fig. 4.3. According to the concept

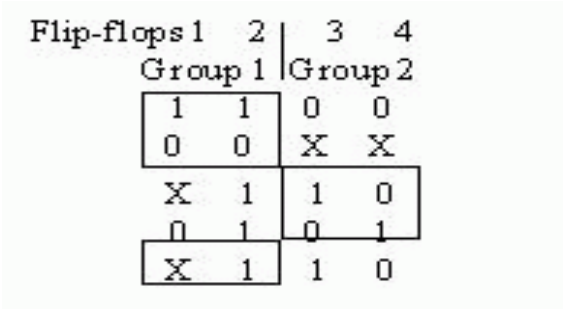


Figure 4.3. Flip-flop and test cube groups for Fig. 4.2.

of D-compatibility, the test cubes are grouped into three D-compatible subsets shown as Fig. 4.3.

After that, all X's in the test cube are filled with specified value 0's or 1's. For the X's in the disabled scan chains, we fill them in the way of non-conflicting D-compatibility. For example, the X's in Cube 2 are filled by 0's because the corresponding bits of Cube 1 that is in the same D-compatible subset with Cube 2 are 0's. For the remaining X's, we randomly fill them with 0's or 1's. Therefore, we can randomly fill the X in cube 5 by 0 or 1.

Finally, fault simulation is done to drop any test vector that does not detect any new faults.

We mentioned the problem where we group flip-flops, duplicate and divide test cubes in the third step. This problem is very complex. To clarify this problem, we introduce the following proposition.

Proposition 1 The peak power of this scheme is near $1/N$ of full scan design method, and hardware overhead is a little higher than that of multiple scan design while keeping the same fault coverage. ■

Therefore, the only factor that needs to be optimized is test application time, which is direct to the total test power.

Theorem 4 The test application time (clock cycles) of a scan circuit based on the proposed scheme is: $TAT = M \cdot \lceil F/N \rceil (N-1) + (n+r+1) \cdot (\lceil F/N \rceil + 1) - 1$,

where, n is the number of original test cubes, F is the number of flip-flops, N is the number of scan chains, M is the number of D-compatible subsets, and r is the number of increased test cubes. ■

The test application time can be calculated according to the test application procedure shown in section 2 as follows. To shift in the first test vector of each D-compatible subset, it takes $N \cdot \lceil F/N \rceil$ clock cycles. There are M D-compatible subsets. Therefore, the total time is $M \cdot N \cdot \lceil F/N \rceil$. For the remaining $n + r - M$ test vectors, to shift in them it takes $(n + r - M) \cdot \lceil F/N \rceil$ clock cycles. $n + r$ clock cycles are needed to capture test response. And, finally, to shift out the test response of the last activated scan chain, it takes $\lceil F/N \rceil$ clock cycles. The test application time is the sum of the above items. That is, $TAT = M \cdot \lceil F/N \rceil (N - 1) + (n + r + 1) \cdot (\lceil F/N \rceil + 1) - 1$.

The problem that minimizes test application time is reduced into the problem that minimizes $M \cdot \lceil F/N \rceil (N - 1) + r \cdot (\lceil F/N \rceil + 1)$ since F, N and n are given values. Thus, we formulate the following problem.

4.4. Problem Formulation

Problem 4 Minimize the test application time of a scan circuit under the proposed scheme. Stating it more formally, given:

- **Input:** a sequential circuit, its detect-capture information, and the number of scan chains N.

Task:

- **Output:** Multiple(N)-scan chains design with M compatible test sets, that achieves:

- **Objective:** minimizing test application time, i.e. $M \cdot \lceil F/N \rceil (N - 1) + r \cdot (\lceil F/N \rceil + 1)$. ■

4.5. Flip-Flops and Test Cubes Grouping

In this section, we introduce the approach to group flip-flops into N scan chains and test cubes into some D-compatible subsets. In this approach, to solve the formulated problem, we use tabu search algorithm to explore the solution space.

4.5.1 Overview

A tabu search algorithm [34] is a heuristic to find the optimal solution. It starts with some initial solution, and repeats to select the best solution among the candidates that can be obtained by small changes (move) from the current solution. The algorithm maintains a tabu that is a list of solutions that are not allowed to be seleted as the next solution. Our algorithm uses the tabu to avoid selecting the same solution twice, and hence avoid a local optimal solution.

Fig. 4.4 summarizes the flip-flops and test cubes grouping algorithm. Lines 1–5 generate an initial solution. First, we extract a set MCC of mandatory cube-capture pairs (MCPs) from the detect-capture information (line 1). Then, a set TCC of temporary cube-capture pairs (TCPs) to test more faults by several flip-flops with several test cubes is obtained (line 2). The flip-flops are grouped into N scan chains utilizing the MCC and TCC sets, and initial flip-flop groups GFF_{init} are obtained (line 3). After that, we use MCC, TCC and GFF_{init} to group test cubes TCG_{inti} (line 4). The detailed information of above steps is given in the following subsections. Lines 6–23 are the heart of the optimization process. $\text{TAT}_{current\_best}$ is set to ∞ (line 7). For every flip-flop pair of flip-flops from different groups, we try every possible move, which is not in the tabu list (lines 8, 9). Here, a move is a term for swapping these flip-flops. After a move, we obtain a new flip-flop group GFF_{tmp}, and a new test cube group TCG_{tmp} (lines 10, 11). If the test application time of the solution[1] $\text{TAT}(\text{GFF}_{tmp},\text{TCG}_{tmp})$ is less than $\text{TAT}_{current\_best}$, we potentially set it to $(\text{GFF}_{next},\text{TCG}_{next})$ that will be the current solution in the next generation (lines 12, 13). After the for loop, $(\text{GFF}_{next},\text{TCG}_{next})$ is assigned to $(\text{GFF}_{current},\text{TCG}_{current})$, and the corresponding move is then recorded in the tabu list (lines 18, 19). If this solution turns out

[1] A solution is flip-flop groups GFF and test cube groups TCG with established values for test application time that is given in the problem.

to be the best one so far, we set (GFF_{best},TCG_{best}) = ($\mathrm{GFF}_{current}$,$\mathrm{TCG}_{current}$). The algorithm ends when either the maximum number of iterations is reached (N_{itr1}), or the maximum number of iterations since the last obtained best solution exceeds some predetermined value (N_{itr2}).

4.5.2 Cube-Capture Pair Extraction

A cube-capture pair (i,q) consists of a test cube i and a flip-flop q and represents a relation that q captures the response of i. We use cube-capture pairs to get not only an initial solution but also flip-flops and test cubes grouping.

In a test, the fault effect of some fault can be captured by a PO, and some flip-flop should be active to detect some fault when a test cube is applied. This information can reduce the solution space efficiently. Starting from the fault list consists of all the detected faults by the test cubes, we first delete the following faults from the fault list.

1. For each fault f in the fault list, if its fault effect can be propagated to a PO when a test cube exercising, then the fault f is removed from the fault list. This is because even if all scan chains are disabled fault f can be detected by capturing the response at the PO.

2. For each fault f in the fault list, if there exists only one cube i that detects the fault, and only one flip-flop q that captures its fault effect, then record the cube and flip-flop as an MCP (i,q), and remove the fault from the fault list.

Here, the MCP describes the information that the flip-flop should be active once when we apply the test cube.

For example, as shown in Table 4.1, fault 8 only appears in the columns of Cube 1, and its fault effect can be captured only by flip-flop 1. Hence, there is a MCP (1,1). Other MCPs about Table 4.1, are (1,2), (2,1), (3,3) and (4,3).

3. For each fault f in the fault list, if it can be detected by an MCP (i,q), then we remove the fault from the fault list.

The tabu search algorithm will explore the solution space that is reduced by this step.

The following steps try to detect the remaining faults. The MCPs are not enough to obtain a good initial solution. We extract more cube-capture pairs called TCPs to detect more faults.

4. If only one flip-flop q captures the fault effect of a fault f, we put f into a fault set F_1, and for each test cube i that can detect f we record the cube flip-flop pairs (CFPs) (i,q) into a set CF_1 of CFPs. After that, we obtain the minimum number of CFPs in CF_1 that detect all the faults in F_1, and record them as TCPs.

5. If only one test cube i detects a fault f, we put f into a fault set F_2, and for each flip-flop q that captures its fault effect we record the CFPs (i,q) into a set CF_2 of CFPs. Then, we obtain the minimum number of CFPs in CF_2 that detect all the faults in F_2. After that, we record them as TCPs.

As shown in Table 4.1, both fault effects of faults 6 and 7 are captured only by flip-flop 4. There is no other fault in F_2. And CFP (2,4) can detect these faults. Therefore, we record a TCP (2,4). Fault 2 can be only detected by cube 2. And its fault effect can be captured by flip-flops 2 and 3. Therefore, the CFPs are (2,2) and (2,3). Since fault 2 is the only fault in F_2, each of the above CFPs can be the TCP. We randomly select one, such as CFP (2,2), as a TCP.

The problems in steps 4 and 5 are equivalent to the minimum prime-implicant covering problem, which is known to be NP-hard. We, therefore, use a greedy algorithm, where we select a CFP from CF_1 (CF_2) that can detect the maximum number of faults from F_1 (F_2). We repeat this step until all the faults in F_1 (F_2) are detected.

6. For each remaining fault f, if it can be detected by some TCP, we record it in fault set F_3.

Notice that, TCPs are only used to get an initial solution so that the flip-flops and test cubes grouping algorithm can obtain better solution. When we use the following flip-flops grouping algoritm and test cubes grouping algorithm to get an initial solution, we delete F_1, F_2 and F_3 from the fault list. In the tabu search part, we restore them to the fault list.

4.5.3 Flip-Flops Grouping

In this subsection, we employ a greedy algorithm, where we group the flip-flops into N scan chains, so that for all the test cubes we try to group the maximum number of flip-flops that have cube-capture pairs for the same test cube to the same groups. First, we give some concepts.

Definition 10 We denote the number of test cubes where both flip-flops j and k have the cube-capture pairs as a *flip-flop relative degree* (*FRD*) $w_{j,k}$.

For instance, there are two cube-capture pairs about Cube 1, (1,1) and (1,2), and three cube-capture pairs about Cube 2, (2,1) (2,2) and (2,4). Both flip-flops 1 and 2 have the cube-capture pair for cubes 1 and 2. As a result, the FRD $w_{1,2}$=2. The FRDs of pairs of flip-flops (1,4) and (2,4) are 1. Other FRDs are 0.

Definition 11 *Flip-flop test relative graph* (*FTRG*). Let $G=(V, E)$ be a weighted undirected graph, where each node $v_i \in V$ corresponds to a flip-flop and the weight of the edge between two nodes j and k is an *FRD* $w_{j,k}$.

The greedy algorithm, which is similar to that of [27], groups the flip-flops into N scan chains shown as Fig. 4.5. An example is shown for N=2 in Fig. 4.6 to construct an *FTRG* and group the flip-flops.

4.5.4 Test Cubes Grouping

After getting the N scan chains, the final phase is to obtain the D-compatible subsets. In this phase, we use the greedy algorithm shown as Fig. 4.7. First, we extend each test cube i into the cube scan chain pair(s) (CSP(s)) (i,k) where k is an activated scan chain when i is applied as the following way. Let $N_{MCP(i)}$ be the number of scan chains, each of them containing at least one flip-flop q that is a component of an MCP (i,q). If $N_{MCP(i)} >0$, the test cube i is extended into CSPs (i,k) for each k of the $N_{MCP(i)}$ scan chains. Otherwise, cube i is extended into CSP (i,k) where k is the scan chain that the inside flip-flops have the maximum number of TCPs (i, q) (in the tabu search part, in this case we randomly select one scan chain k). From the fault list, we remove all the faults that can be detected by the CSPs. If there are some remaining faults, we append

such a CSP that detect the maximum number of faults among the remaining faults, and delete the detected faults from the fault list. We do the above step until the fault list is empty. Finally, we obtain the D-compatible subsets using the following steps. First, we regard each CSP as a D-compatible subset. For every D-compatible subset, we try every other D-compatible subset. If two of them are D-compatible, we merge them into a D-compatible subset. Until there are no D-compatible subsets can be merged.

For example, we can retrieve the set of test cubes by the following way for the test cubes given in Fig. 4.2 and the flip-flop groups shown in Fig. 4.6. There are two MCPs, (1,1) and (1,2), about cube 1. All the flip-flops in the MCPs, flip-flops 1 and 2 are in group 1. Hence, there is a CSP (1,1). As the same way, there are CSPs (2,1), (3,2) and (4,2) for cubes 2, 3 and 4 respectively. Fault 3 remains in the fault list after removing the detected faults. After we apply (3,1), fault 3 is detected. That is, all the faults are detected by the five CSPs. At the end, we group the CSPs into some D-compatible subsets. (1,1) and (2,1) have the same scan chain activity. And the bits of disabled scan chain in cubes 1 and 2 are compatible. Thus, (1,2) and (2,1) are grouped into a D-compatible subset. Though (3,1) also has the same scan chain activity with (1,1) and (2,1), they are not D-compatible due to the conflict of bits of their disabled scan chain. The results are shown in Fig. 4.3.

Notice that, this flip-flops and test cubes grouping algorithm can also deal with the circuits that already have some sub-scan chains by regarding a sub-scan chain as a scan flip-flop.

4.6. Experimental Results

We have conducted experiments on full scan version of ISCAS89 benchmark circuits. In the experiments, we use the ATPG tool "TestGen" of Synopsys to generate test cubes and do fault simulation. Table 4.2 shows the results of our proposed method compared with previous methods. We do not report the fault coverage in the results because it is the same as that of full scan test.

Flip-flops and test cubes grouping algorithm:

/* initial solution generation */

1. extract a set MCC of MCPs
2. extract a set TCC of TCPs
3. $GFF_{init} \leftarrow$ FF grouping from MCC and TCC
4. $TCG_{init} \leftarrow$ Test cube grouping using MCC, TCC and GFF_{init}
5. $(GFF_{current}, TCG_{current}) \leftarrow (GFF_{init}, TCG_{init})$

/* tabu-search */

6. repeat {
7. $TAT_{current\_best} \leftarrow \infty$
8. for every pair (q_1, q_2) of flip-flops from different groups in $GFF_{current}$ {
9. if this pair is not in tabu list {
10. $GFF_{tmp} \leftarrow$ swap q_1 and q_2 for $GFF_{current}$
11. $TCG_{tmp} \leftarrow$ test cube grouping from MCC and GFF_{tmp}
12. if $TAT(GFF_{tmp}, TCG_{tmp}) < TAT_{current\_best}$ {
13. $(GFF_{next}, TCG_{next}) \leftarrow (GFF_{tmp}, TCG_{tmp})$
14. }
15. }
16. }
17. $(GFF_{current}, TCG_{current}) \leftarrow (GFF_{next}, TCG_{next})$
18. add $GFF_{current}$ into tabu list
19. if $TAT(GFF_{current}, TCG_{current}) < TAT(GFF_{best}, TCG_{best})$ {
20. $(GFF_{best}, TCG_{best}) \leftarrow (GFF_{current}, TCG_{current})$
21. }
22. }
23. until #iterations>Min$\{N_{itr1}, N_{itr2}\}$

Figure 4.4. Flip-flops and test cubes grouping algorithm.

```
1. for (j=1 to N){
2.   from the unselected nodes in FTRG, select
     the pair with the highest edges weight;
3.   add the pair to group j;
4.   group_j=2;
5.   while(group_j < ⌈F/N⌉){
6.     select a node such that the sum of the weights of the
       edges between the node and the already selected
       nodes in group j is maximum;
7.     Add the node to group j;
8.     group_j++;
9.   }
10.}
11.return the flip-flop groups;
```

Figure 4.5. Flip-flop grouping algorithm.

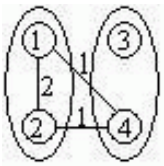


Figure 4.6. Flip-flop grouping.

```
1. for (i=1 to n){
2.   N_MCP(i) ← # scan chains, each of them containing at
     least one flip-flop q that is a component of an MCP (i,q);
3.   If N_MCP(i) > 0, extend cube i to CSP (i,k)
       for each k of the N_MCP(i) scan chains;
4.   Else, extend cube i to CSP (i,k), where k is the scan
       chain that the inside flip-flops have the maximum
       number of TCPs (i,q) (in the tabu search part,
       we randomly select one scan chain k);
5. }
6. remove all the detected faults;
7. repeat {
8.   find a CSP (i,k) that detect the maximum # undetected
     faults;
9.   Remove the detected faults from the fault list;
10.}
11.until the fault list is empty;
12.for (m=1 to n+r)
   for (j=m+1 to n+r) {
13. if there exist D-compatible subsets (DSs) DS_m and DS_j
     if DS_m and DS_j are D-compatible {
14.     merge them into DS_m, remove DS_j;
15. }
16.}
17.return the D-compatible subsets;
```

Figure 4.7. Test cubes grouping algorithm.

Table 4.2. Results of power saving using scan disable technique.

Ckt.	#ff	Time Red.(%)			Power Red.(%)(N=2)				Power Red.(%)(N=3)				Power Red.(%)(N=4)			
		N=2	N=3	N=4	vs. conv.		vs. [21], [22]		vs. conv.		vs. [21], [22]		vs. conv.		vs. [21], [22]	
					Av.	Pk.	Av.	Pk.	Av.	Pk.	Av.	Pk.	Av.	Pk.	Av.	Pk.
s838	32	4.3	27.4	36.1	49.9	50.0	0.1	50.0	65.3	65.6	1.8	65.6	74.0	75.0	1.1	75.0
s953	29	22.4	35.5	37.6	51.8	31.8	6.1	16.7	64.8	54.5	2.1	44.4	72.9	63.6	7.4	55.6
s1196	18	39.0	54.8	60.9	61.8	46.7	38.0	46.7	72.7	60.0	28.2	60.0	75.6	66.7	19.6	66.7
s1238	18	35.8	54.4	60.7	53.0	42.9	11.0	42.9	70.2	57.1	23.9	57.1	75.6	64.3	16.4	64.3
s1423	74	-17.7	-18.2	-26.2	53.4	34.9	7.9	28.2	67.8	53.5	6.4	48.7	74.4	65.1	3.0	61.5
s9234	211	-15.0	-35.5	-40.6	48.6	15.2	-0.3	13.1	64.7	43.2	-0.8	41.8	73.2	57.6	-0.6	56.6
s13207	669	47.6	45.4	42.5	45.7	48.6	-5.4	48.6	63.7	65.2	-5.7	65.2	73.6	73.6	-3.1	73.6

Columns Ckt. and #ff give the circuit name and the number of scan elements respectively. After them, the columns Time Red.(%) report the test application time reduction of our proposed approach when N=2, N=3 and N=4 compared to the conventional full scan test with one scan chain, where the N scan chains are grouped using the procedure of the last section. Notice that, the test application time of the methods in [21], [22] is the same as that of the conventional full scan test. Hence, these columns also show the test application time reduction compared with the methods in [21], [22]. The following columns show the reduction in average power and peak power for N=2, N=3 and N=4 compared to conventional full scan test with one scan chain, and the approach in [21], [22] separately. In this experiment, we use the technique of weight transition count described in [19] to estimate power. Therefore, the columns Power Red(%) in Table 4.2 describe the percentages reduction of the weighted transitions with previous methods.

In this table, compared with the conventional full scan test, when N=2, N=3 and N=4, the average power is reduced up to 61.8%, 72.7% and 75.6% respectively. The peak power dissipation is also reduced drastically up to 50.0% when N=2, up to 65.6% when N=3, and up to 75.0% when N=4. In comparison with the method in [21], [22], the peak power is reduced. The maximum reduction ratios in peak power dissipation are the same with that of the conventional full scan test though they are sometimes a little smaller. And the average power is reduced or comparable. For test application time, except S1423 and S9234, our method can achieve better results than that of the other two methods. For example, applying test to S1196 the test application time is reduced by 60.9% when N=4. The test application time of S1423 and S9234 are a little longer. This is because in these circuits many test vectors need to be applied several times to preserve the fault coverage of the test cubes.

Table 4.3 displays the test application time reduction compared with random grouping flip-flops approach. In this random grouping algorithm, the information of cube-capture pairs and test cubes grouping algorithm in our approach are utilized. We run the algorithm ten times, and select the best solution as the comparison. Given in Table 4.3, the test application time has up to 41.2% reduction. Therefore, our approach is more efficient to reduce test application

Table 4.3. Comparison of test application time with random flip-flop grouping algorithm.

Ckt.	Time Red.(%)		
	N=2	N=3	N=4
s838	10.1	23.0	25.2
s953	13.6	26.9	32.0
s1196	6.6	11.3	15.6
s1238	9.7	15.3	20.7
s1423	26.5	37.7	41.2
s9234	17.3	19.3	24.7
s13207	2.9	4.1	6.6

Table 4.4. Comparison of average power reduction with [25], [26].

Ckt.	Proposed Approach			[25]	[26]
	N=2	N=3	N=4		
s9234	48.6	64.7	73.2	19.1	28
s13207	45.7	63.7	73.6	43.4	33

time compared with the random flip-flop grouping algorithm.

The comparison of average power reduction with [25], [26] is given in Table 4.4. Columns 2, 3 and 4 show the percentages of average power reduction for N=2, N=3 and N=4 respectively, which are also given in Table 4.2. The percentages of average power reduction of the methods in [25], [26] are displayed in columns 5 and 6. The purpose of both methods in [25], [26] that employ two flip-flops and test cubes grouping algorithms is to reduce average power dissipation. As described before, our methodology is mainly for peak power reduction. However, Table 4.4 demonstrates our method is more efficient to reduce power compared with these methods.

From Tables 4.2–4.4, we can conclude that the proposed method is efficient

in reducing both average power and peak power dissipation during test without loss the fault coverage. This method is also efficient to reduce test application time that is one of the most important factors in the test.

4.7. Conclusions

This chapter proposed a low power scan test scheme. In this scheme, both average power and peak power reduction are achieved by activating only one scan chain during scan test. To minimize test application time as well as the total test power, this chapter also formulated a problem based on this scheme. A tabu search-based algorithm is presented to solve the problem. Experimental results show that the proposed approach is more efficient in reducing average power and peak power dissipation at a little longer test application time compared to the full scan test methodology.

Chapter 5

Conclusions and Future Works

This chapter first summarizes what works are completed in this thesis. Then future works to improve the techniques are given.

5.1. Summary of This Thesis

This thesis proposed several methodologies to reduce peak power as well as average power at RT-level and at gate level respectively.

In Chapter 3, we proposed three non-scan BIST schemes for RTL data paths and formulated three DFT problems for the schemes under power constraints. The proposed schemes include one generic non-scan BIST scheme where we can explore trade-offs among hardware overhead, test application time and power dissipation. We also proposed other two schemes, adjacent non-scan BIST scheme and boundary non-scan BIST scheme, as special cases that intend short test application time and low hardware overhead respectively. This chapter also presented three power constrained DFT algorithms , under three non-scan BIST schemes, for RTL data paths. The first algorithm is for adjacent non-scan BIST scheme. The second proposed algorithm is for a boundary non-scan BIST scheme, and achieves low hardware overheads. The third algorithm is for a generic non-scan BIST scheme. We use a Tabu search algorithm to explore the solution space. This approach can offer some tradeoffs between hardware overhead, test application time and power dissipation. A designer can easily select an appropriate design parameter based on the desired tradeoff. Experimental results confirm the

good performance and practicality of our new approaches.

Chapter 4 proposed a low power scan test scheme. In this scheme, the flip-flops are grouped into N scan chains. At any time, only one scan chain is active during scan test. Therefore, both average power and peak power are reduced compared with conventional full scan test methodology. This chapter also formulated a problem based on this scheme. A tabu search-based algorithm is presented to solve this problem. In this approach we handled the information during deterministic test efficiently to minimize test application time as well as the total test power. Experimental results demonstrate that the proposed approach drastically reduces both average power and peak power dissipation at a little longer test application time on various benchmark circuits.

5.2. Future Works

Although we proposed some efficient methodologies that achieve short test application time or low hardware overhead under peak power constraint, there are still some rooms to improve.

The methods in Chapter 3 will be great improvement if one or more of the following problems are resoved or suggestions are employed.

1. **The realization of Thru-function.** In this thesis, we employ two kinds of thru-functions shown in Chapter 2. As we know, the aim of thru function is to pass a pseudo-random sequence through a function module. In this sense, the hold function for register can realize the function of thru by hold a constant value of an input of a functional module. The main advantage of using a hold function is to stop the propagation of useless transitions to save power. Consequently, test application is reduced by scheduling more modules in a test session. The disadvantage is that it needs hareware to implement. This is the trade-off between test application time and hardware overhead. Notice that, in the original RTL data paths, there are many registers have hold functions. If we utilize them, both test application time and hardware overhead may be reduced.

2. **The test of MUX.** In our methods for Problems 2 and 3, to achieve

complete test we regards a MUX as a module, and schedule it in a test session. In fact, some MUXes has already tested after testing some funtional modules. If we can recognize them in advance, test application time can be reduced. Moreover, there is no need thru-functions to propagate test patterns or test responses of the MUXes. Hence, hardware overhead is also reduced without increasing power.

3. **The extension of the testability.** The strong testability proposed by Ohtake et al in [9] is more general than the single-control testability in [3]. If we explore an approach based on the general testability, we may obtain better solution.

4. **The improvement of the approaches.** The power estimation formula of the test for multiple cores that equals the sum of the power of each is not fit for that of the modules at RTL. The new formula or inequation is need to be investigated. In our approach, as shown in Chapter 3, first we try to determine test paths that have the minimum test power for each module. If there exist some modules where test power exceeds the power constraints we bypass some power consuming paths untill the test power of all the modules satisfies the power constraints. Finally, after doing simulation, we get the test power of some compatible modules. From this view, our approaches may not be efficient to use the information of sharing control path and minimize useless power dissipation.

For the method in Chapter 4, the futher works might be the following three directions.

1. **Approach improvement.** Though this approach reduces test application time for most benchmark circuits we believe that there are still rooms to reduce it. Future work will investigate new approach based on the proposed scheme to reduce test cost.

2. **Layout impact.** Scan routing is one of the main issues when designing a scan chain. In our approach, we divide the scan flip-flops into N groups, and line the flip-flops in a group into a scan chain without considering the

wire length of scan connections. It may cause routing congestion problem. Thus, it will be very practical to consider layout impact to our methodology.

3. **Application for high level circuits.** This method also can be applied for the circuits at high level, especially, for SOC. Employing this method, the test application time will be reduced drastically under power constraints though there are still some problems need to be resolved and some ideas need to be turned out.

Acknowledgements

I would like to express my appreciation to my supervisor, Professor Hideo Fujiwara, who taught me the basic knowledge about circuit test, guided me the research methodology and supported me in many things with great care and patient during my study at Nara Institute of Science and Technology (NAIST). His down-to-earth attitude to science and research impressed me deeply and will greatly benefit me in my future career.

I would also like to thank Professor Minoru Ito for his valuable comments and important suggestions concerning this thesis.

I am very much grateful to Associate Professor Michiko Inoue for her continuous guidance, insightful advices, valuable comments and suggestions at various stages of my research work, which made me target higher and higher in doing research.

I am also thankful to Assistant Professor Satoshi Ohtake and Assistant Professor Tomokazu Yoneda for their valuable comments, suggestions and friendly discussion.

I would like to express my heartiest gratitude to Professor Jacob Savir, New Jersey Institute of Technology, USA, Assitant Professor Kenichi Yamaguchi, Nara National College of Technology, and Assitant Professor Tsuyoshi Iwagaki, Japan Advanced Institute of Science and Technology, for their constant guidance and encouragement throughout the course of this research. I feel honored and privileged to have worked with them for this thesis.

I am grateful to Professor Samiha Mourad, Santa Clara University, and Professor Kewal K. Saluja, University of Wisconsin-Madison, for their friendly discussions.

I would also like to extend my gratitude to Ms. Yoshiko Hirayama and Ms.

Saeko Ono for helping me in different official correspondence.

I extend my gratitude to Dr. Erik Larson, Dr. Md. Altaf-Ul-Amin, Dr. Shintaro Nagai, Dr. Kazuko Kambe, Dr. Thomas Clouqueur, Dr. Ilia Polian, Dr. Virendra Singh, Dr. Yoshiyuki Nakamura, Mr. Yusuke saga, Ms. Chia Yee Ooi, Mr. Hiroyuki Iwata, Mr. Masato Nakazato, Mr. Yuki Yoshikawa, and Mr. Fawnizu Azmadi Hussin for helping me to use CAD tools. And their support and cooperation made my stay very pleasant and comfortable. Thanks are also due to all other members of the Fujiwara Laboratory for making my stay pleasurable.

I am grateful to Ministry of Education, Culture, Sports, Science and Technology (MEXT), Government of Japan for supporting me with Japanese government scholarship through out my course of study.

This work was supported in part by JSPS (Japan Society for Promotion of Science) under Grant-in-Aid for Scientific Research and Foundation of NAIST under the grant for activity of education and research. I express my gratitude to all these sponsoring organizations.

I would like to express my gratitude to Professor Yinghua Min, Professor Dafang Zhang and Professor Jishun Kuang for his continuous guidance during Master course study and motivating me to pursue doctoral course in Japan. I am also thankful to Professor Chuzhong Liu and Mrs. Kejian Wang for their encouragement.

I wish to acknowledge my parents and sisters for their endless love and support. I am also grateful to my wife for her understanding and encouragement and my sons for their child's taste.

References

[1] B. Koenemann, J. Mucha and G. Zwiehoff, "Built-in logic block observation techniques," in Proc. Int'l Test Conf. (ITC79), pp. 37–41, October, 1979.

[2] L.T. Wang and E.J. McCluskey, "Concurrent Built-in logic block observer (CBILBO)," in Proc. The Int. Symp. on Circuits and Systems, pp. 1054–1057, 1986.

[3] T. Masuzawa, M. Izutsu, H. Wada and H. Fujiwara, "Single-Control Testability of RTL Data Paths for BIST," in Proc. 9th Asian Test Symposium. (ATS2000), pp. 210–215, 2000.

[4] M.S. Abadir and M.A. Breuer, "A knowledge based system for designing testable VLSI chips," IEEE Design and Test of Computers, vol. 2, no. 4, pp. 56–68, Aug. 1985.

[5] S. Lin, C. Njinda and M. Breuer, "Generating a family of testable designs using the BILBO methodology," Journal of Electronic Testing: Theory and Applications (JETTA), vol. 4, no. 1, pp. 71–89, 1993.

[6] N. Nicolici and B. Al-Hashimi, "Efficient BIST hardware insertion with low test application time for synthesized data paths," in Proc. IEEE/ACM Design Automation and Test in Europe (DATE 99), pp. 289-295, March 1999.

[7] N. Nicolici and B. M. Al-Hashimi, Power-Constrained Testing of VLSI Circuits. Kluwer Academic Publishers, Boston, MA, 2003.

[8] N. Nicolici and B. Al-Hashimi, "Power Conscious Test Synthesis and Scheduling for BIST RTL Data Paths," in Proc. Int'l Test Conf. (ITC2000), pp. 662–671, 2000.

[9] S. Ohtake, H. Wada, T. Masuzawa and H. Fujiwara, "A non-scan DFT method at register-transfer level to achieve complete fault efficiency," Asia and South Pacific Design Automation Conference 2000 (ASP-DAC2000), pp. 599–604, Jan. 2000.

[10] K. Yamaguchi, H. Wada, T. Masuzawa and H. Fujiwara, "A BIST Method Based on Concurrent Single-Control Testability of RTL Data Paths," in Proc. 10th Asian Test Symposium. (ATS2001), pp. 313–318, 2001.

[11] K. Yamaguchi, M. Inoue and H. Fujiwara, "Hierarchical BIST: Test-Per-Clock BIST with Low Overhead," Digest of papers 3rd Int'l Workshop on RTL and High Level Testing. (WRTLT2002), pp. 42–47, 2002.

[12] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," In Proc. IEEE VLSI Test Symposium, pp. 4–9, 1993.

[13] K. Roy and S. Prasad, "Low-power CMOS VLSI circuit design," John Wiley&Sons, 2000.

[14] H. Vranken, T. Waayers, H. Fleury and D. Lelouvier, "Enhanced reduced-pin-count test for full-scan design," In Proc. IEEE International Test Conference, pp. 738–747, 2001.

[15] S. Chakravarty and V. Dabholkar, "Two techniques for minimizing power dissipation in scan circuits during test application," In Proc. IEEE Asian Test Symposium, pp. 324–329, 1994.

[16] V. Dabholkar, S. Chakravarty, I. Pomeranz and S.M. Reddy, "Techniques for minimizing power dissipation in scan and combinational circuits during test application," IEEE trans. on Computer-Aided Design of Integrated Circuits and Systems, Vol. 17, No. 12, pp. 1325–1333, 1998.

[17] P. Flores. J. Costa, H. Neto, J. Monterio and J. Marq1ues-Silva, "Assignment and reordering of incompletely specified pattern sequences targeting minimum power dissipation," Proc. of International Conference on VLSI Design, pp. 37–41, 1999.

[18] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault and S. Pravossoudovitch, "Efficient scan chain design for power minimization during scan testing under routing constraint," In Proc. IEEE International Test Conference, pp. 488–493, 2003.

[19] R. Sankaralingam, R. R. Oruganti and N. A. Touba, "Static compaction techniques to control scan vector power dissipation," In Proc. IEEE VLSI Test Symposium, pp. 35–40, 2000.

[20] K. M. Butler, J. Saxena, T. Fryars, G. Hetherington, A. Jain and J. Lewis, "Minimizing power consumption in scan testing: pattern generation and DFT techniques," In Proc. IEEE International Test Conference, pp. 355–364, 2004.

[21] L.Whetsel, "Adapting scan architectures for low power operation," In Proc. IEEE International Test Conference, pp. 863–872, 2000.

[22] J. Saxena, K. M. Butler and L. Whetsel, "An analysis of power reduction techniques in scan testing," In Proc. IEEE International Test Conference, pp. 670–677, 2001.

[23] 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," In Proc. IEEE Asian Test Symposium, pp. 253–258, 2001.

[24] B. B. Bhattacharya, S. C. Seth and S. Zhang, "Double-tree scan: a novel low-power scan-path architecture," In Proc. IEEE International Test Conference, pp. 471–479, 2003.

[25] R. Sankaralingam, B. Pouya and N.A. Touba, "Reducing power disspation during test using scan chain disable," In Proc. IEEE VLSI Test Symposium, pp. 319–324, 2001.

[26] R. Sankaralingam and N.A. Touba, "Reducing test power during test using programmable scan chain disable," The First IEEE International Workshop on Electronic Design, Test and Applications, pp. 159–163, 2002.

[27] N. Z. Basturkmen, S. M. Reddy and I. Pomeranz, "A low power pseudo-random BIST technique," In Proc. IEEE International Conference on Computer Design, pp. 468–473, 2002.

[28] Z. You, M. Inoue and H. Fujiwara, "On the non-scan BIST schemes under power constraints for RTL data paths," Digest of papers 4th Int'l Workshop on RTL and High Level Testing. (WRTLT2003), pp. 14–21, 2003.

[29] Z. You, K. Yamaguchi, M. Inoue, J. Savir and H. Fujiwara, "Power-constrained test scheduling for RTL datapaths of non-scan BIST schemes," Proceedings of the IEEE 13th Asian Test Symposium (ATS'04), pp. 32–39, Nov. 2004.

[30] Z. You, K. Yamaguchi, M. Inoue, J. Savir and H. Fujiwara, "Power-constrained test synthesis and scheduling algorithms for non-scan BIST-able RTL data paths," IEICE Trans. on Information and Systems, Vol. E88-D, No. 8, pp.1940-1947, Aug. 2005.

[31] R. Chou K. Saluja, and V. Agrawal, "Scheduling Tests for VLSI Systems Under Power Constraints," IEEE Trans on VLSI systems, Vol. 5, No. 2, pp. 175–185, June 1997.

[32] D. Jungnickel, Graphs, Networks and algorithms, Springer-Verlag, Berlin, 1999.

[33] V. Muresan, X. Wang, V. Muresan and M. Vladutiu, "The left Edge Algorithm and the Tree Growing Technique in Block-Test Scheduling under Power Constraints," in Proc. of the 18th IEEE VLSI Test Symposium (VTS2000), pp. 417–422, 2000.

[34] F. Glover and M. Laguna. Tabu search, In C.R. Reeves, editor, Modern Heuristic Techniques for Combinatorial Problems, pp. 70-150. McGraw-Hill Book Company, 1995.

[35] P.G. Paulin and J.P. Knight, "Force-Directed Scheduling for the Behavioral Synthesis of ASICs," IEEE Trans. on Computer-Aided Design, pp. 661–679, June 1989.

[36] C. Tseng, and D.P. Siewiorek, "Automated Synthesis of Data Paths in Digital Systems," IEEE Trans. on Computer-Aided Design, pp. 379–395, July 1986.

[37] E. Macii, M. Pedram and F. Somenzi, "High Level Power Modeling, Estimation, and Optimization," IEEE Trans. on Computers, vol. 44, pp. 234–247, Feb. 1995.

[38] Z. You, T. Iwagaki, M. Inoue and H. Fujiwara, "A low power deterministic test using scan chain disable technique," Digest of papers 6th Int'l Workshop on RTL and High Level Testing (WRTLT'05), pp. 184–191, July 2005.

[39] Z. You, T. Iwagaki, M. Inoue and H. Fujiwara, "A low power deterministic test using scan chain disable technique," IEICE Trans. on Information and Systems (To appear).

[40] K. Miyase, S. Kajihara, S. M. Reddy, "Multiple scan tree design with test vector modification," In Proc. IEEE Asian Test Symposium, pp. 76–81, 2004.