NAIST-IS-DD0461025

Doctoral Dissertation

Studies on Defect Level and Diagnosis for Built-in Self Test Architecture

Yoshiyuki Nakamura

March 17, 2006

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

A 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

Yoshiyuki Nakamura

Thesis Committee:

Professor Hideo Fujiwara	(Supervisor)
Professor Minoru Ito	(Co-supervisor)
Associate Professor Michiko Inoue	(Co-supervisor)

Studies on Defect Level and Diagnosis for Built-in Self Test Architecture*

Yoshiyuki Nakamura

Abstract

Design for testability (DFT) has been a major area of research and practice for the last 40 years. This has emerged after many digital systems manufacturers have realized that test can no longer be an "after-thought".

Computer and semiconductor manufacturers have changed their design practices to incorporate test as an integral part of their design cycle. Strict coverage requirements have been put in place before tape-out stage. Coverage requirements in the high percent against single stuck-at faults and speed related faults are quite common today. In order to reach this goal, two major DFT methodologies have been adopted: scan design and built-in self-test (BIST). Even though scan design has greatly alleviated the test process, it did not bring the test generation time and the test data file down to acceptable levels. The breakthrough in this domain has been achieved through BIST.

BIST hardware is included today in many chips. However, there are two problems still remaining. One problem is the reliability of the BIST hardware. Since the BIST hardware is manufactured using the same technology as the functional circuits, it is possible for it to be faulty. It is important, therefore, to assess the impact of this unreliable BIST on the product defect level after test.

Another problem is the fault diagnosis in the BIST environment. Since BIST compacts test responses, it requires only small tester memory and it can perform at-speed test even if the test frequency is much higher than the tester frequency

*Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0461025, March 17, 2006.

limitation. On the other hand, BIST causes problems in diagnosis due to its compacted responses. Indeed, pass/fail information obtained from BIST response analyzer is insufficient for diagnosis.

In this thesis, first, the formula, relating the product defect level as a function of the manufacturing yield and fault coverage, is re-examined. In particular, special attention is given to the influence of an unreliable BIST on this relationship. This thesis also studies the product quality improvements as induced by the BIST pretest, and provides some insight as to when this pretest maybe worthwhile performing.

Next, this thesis investigates how a relatively slow tester can observe the at-speed behavior of fast circuits. In a first place, this thesis presents a method to identify all errors without any aliasing and negligible extra hardware. Then, the scheme is modified to reduce test application time by using signature analyzers. This approach can be used to identify all error occurrences in BIST environment.

Keywords:

built-in self test, fault coverage, defect level, fault diagnosis, error identification, at-speed test

List of Publications

Journal Papers

1. Yoshiyuki Nakamura, Jacob Savir and Hideo Fujiwara, "Defect level vs. yield and fault coverage in the presence of an unreliable BIST," IEICE Transactions on Information and Systems, Vol.E88-D, No.6, pp.1210-1216, Jun. 2005.

2. Yoshiyuki Nakamura, Thomas Clouqueur, Kewal K. Saluja and Hideo Fujiwara, "Error identification in at-speed scan BIST environment in the presence of circuit and tester speed mismatch," IEICE Transactions on Information and Systems, Vol.E89-D, No.3, pp.1165-1172, Mar. 2006.

International Conferences (Reviewed)

1. Yoshiyuki Nakamura, Jacob Savir and Hideo Fujiwara, "Defect level vs. yield and fault coverage in the presence of an imperfect BIST," IEEE 5th Workshop on RTL and High Level Testing,

pp.79-84, Nov. 2004.

2. Yoshiyuki Nakamura, Thomas Clouqueur, Kewal K. Saluja and Hideo Fujiwara, "Perfect error identification in at-speed BIST environment," IEEE 6th Workshop on RTL and High Level Testing, pp.1-11, Jul. 2005.

3. Yoshiyuki Nakamura, Jacob Savir and Hideo Fujiwara, "BIST Pretest of ICs: Risks and Benefits," IEEE 24th VLSI Test Symposium, May 2006, (To appear).

Contents

1 Introduction **1**

1.1. Built-in Self Test (BIST) 1

1.2. Defect Level and Fault Coverage 3

1.3. Fault Diagnosis 4

1.4. Thesis Organization 6

2 Defect Level, Yield and Fault Coverage **9**

2.1. Introduction 9

2.2. Definitions of Terms 10

2.3. Previous Works 11

2.4. Williams and Brown's Equation 12

2.5. Conclusion 14

3 Defect Level in the Presence of an Unreliable BIST **15**

3.1. Introduction 15

3.2. Unreliable BIST 16

3.3. Enhanced Equation in the Presence of an Unreliable BIST 17

3.3.1 Fault Detection Scenarios 17

3.3.2 The Enhanced Equations 19
3.3.3 The Impact of the BIST Impurity 24
3.4. Some Typical Behaviors 27
3.4.1 Early Life Impact 28
3.4.2 Impact at Maturity 31
3.4.3 The Impact Trend 34
3.5. Conclusion 36

4 Effect of BIST Pretest on Defect Level 39
4.1. Introduction 39
4.2. BIST Pretest 40
4.3. Effects of BIST Pretest 41
4.3.1 Analysis 41
4.3.2 Sizing the Effect of the BIST Pretest 48
4.4. Some Typical Behaviors 52
4.4.1 Early Life Impact 52
4.4.2 Impact at Maturity 56
4.4.3 Comparison of Defect Levels with and without BIST Pretest 60
4.4.4 The Case $\rho\prime \neq \rho$ 71
4.4.5 The Impact Trend 72
4.5. Conclusion 73

5 The Problem of Fault Diagnosis in At-speed BIST Environment 75
5.1. Introduction 75

5.2. Error Information for Diagnosis 76
5.3. Previous Works 77
5.4. Problem Formulation 78
5.5. Conclusion 81

6 Error Identification Method for Maximum Diagnostic Resolution 83
6.1. Introduction 83
6.2. Observing Responses 84
6.3. Problem of Observing Every Response 87
6.4. Conditions to Achieve Maximum Resolution 88
6.5. Adjusting N or P to Achieve Maximum Resolution . . 91
6.6. Procedure for Identifying Every Error Occurrence . . . 93
6.6.1 Procedure for the First Problem Formulation . . 93
6.6.2 Extending the Procedure for the Second Problem Formulation 95
6.7. Conclusion 97

7 Diagnosable At-speed BIST 99
7.1. Introduction 99
7.2. BIST Architecture for Maximum Diagnostic Resolution 100
7.3. Using Signature Analyzer to Reduce the Test Application Time 102
7.4. Procedure for Identifying Errors Using the Diagnosable BIST 104

7.5. Effect of the Error Detector on Reducing Test Application Time 108
7.5.1 Analytical Expression 108
7.5.2 Effect of the Error Detector 110
7.6. Experiments 115
7.6.1 Verifying the Analytical Results 115
7.6.2 Experiments for an Industry's Circuit 118
7.7. Conclusion 121

8 Conclusions and Future Works **123**
8.1. Summary of the Thesis 123
8.2. Future Works 126

Acknowledgments **129**

References **131**

List of Figures

3.1 F'/F and $\Delta D'/D$ as a function of Y (at early life) . . 29

3.2 F'/F and $\Delta D'/D$ as a function of ρ (at early life) . . . 29

3.3 F'/F and $\Delta D'/D$ as a function of α (at early life) . . . 30

3.4 F'/F and $\Delta D'/D$ as a function of F (at early life) . . 30

3.5 F'/F and $\Delta D'/D$ as a function of Y (at maturity stage) 31

3.6 F'/F and $\Delta D'/D$ as a function of ρ (at maturity stage) 32

3.7 F'/F and $\Delta D'/D$ as a function of α (at maturity stage) 32

3.8 F'/F and $\Delta D'/D$ as a function of F (at maturity stage) 33

3.9 D and $\Delta D'$ as a function of F 35

4.1 F''/F and $\delta D/D$ as a function of Y (at early life) . . . 53

4.2 F''/F and $\delta D/D$ as a function of ρ (at early life) . . . 54

4.3 F''/F and $\delta D/D$ as a function of α (at early life) . . . 54

4.4 F''/F and $\delta D/D$ as a function of F (at early life) . . . 55

4.5 F''/F and $\delta D/D$ as a function of μ (at early life) . . . 55

4.6 F''/F and $\delta D/D$ as a function of Y (at maturity stage) 57

4.7 F''/F and $\delta D/D$ as a function of ρ (at maturity stage) 57

4.8 F''/F and $\delta D/D$ as a function of α (at maturity stage) 58

4.9 F''/F and $\delta D/D$ as a function of F (at maturity stage) 58
4.10 F''/F and $\delta D/D$ as a function of μ (at maturity stage) 59
4.11 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of Y (at early life) 61
4.12 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ (at early life) 62
4.13 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of α (at early life) 63
4.14 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of F (at early life) 64
4.15 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of μ (at early life) 65
4.16 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of Y(at maturity stage) 66
4.17 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ (at maturity stage) 67
4.18 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of α (at maturity stage) 68
4.19 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function F (at maturity stage) 69
4.20 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of μ (at maturity stage) 70
4.21 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ'(at early life) 71
4.22 $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ' (at maturity stage) 72

6.1 Response observation by low speed tester 85
6.2 CUT and observe intervals 85
6.3 Responses and observed results 85
6.4 Successful observation 86

7.1 Diagnosable BIST 101

7.2 Diagnosis with error detector SAs 103
7.3 Iteration reduction rate as a function of P 112
7.4 Iteration reduction rate as a function of number of SAs 112
7.5 Iteration reduction rate as a function of N 113
7.6 Iteration reduction rate as a function of 1 bit error probability 114
7.7 TAT as a function of CUT frequency 116
7.8 TAT as a function of number of SAs 117
7.9 TAT as a function of BIST sequence length 117

List of Tables

3.1 Test outcomes in the presence of an unreliable BIST . . 18

7.1 An industry's circuit 119

7.2 Experimental results of the industry's circuit 120

Chapter 1

Introduction

1.1. Built-in Self Test (BIST)

Design for testability (DFT) has been a major area of research and practice for the last 40 years. This has emerged after many digital systems manufacturers have realized that test can no longer be an "after-thought".

Computer and semiconductor manufacturers have changed their design practices to incorporate test as an integral part of their design cycle. Strict coverage requirements have been put in place before tape-out stage. Coverage requirements in the high percent against single stuck-at faults and speed related faults are quite common today. In order to reach this goal, two major DFT methodologies have been adopted: scan design and built-in self-test (BIST).

Even though scan design [1][2] has greatly alleviated the test pro-

cess, it did not bring the test generation time and the test data file down to acceptable levels. The breakthrough in this domain has been achieved through BIST. BIST may come in many different flavors. There are BIST designs that rest on exhaustive or pseudo-exhaustive patterns, that use functional patterns (mostly for off the shelf products where there is no knowledge of the design details), and there are BIST designs that use pseudo-random patterns [3]. Pseudo-random BIST designs are the most widely used due to their relative simplicity and low cost. They also enjoy the added benefit of potentially detecting many un-modeled defects, and therefore achieving a higher shipped-quality level.

In pseudo-random-based BIST designs, the patterns are generated by a linear feedback shift register (LFSR) and the test responses are compressed in a multiple-input signature register (MISR). At the end of the test, the MISR contains a short signature (typically 16-64 bits) of the entire test history. The good machine signature is computed beforehand for reference during test. A product is declared fault-free if the measured signature coincides with the reference signature. A circuit is declared faulty if these two signatures differ from one another.

All BIST-based methodologies, without exception, are subject to what is called masking, or aliasing, a phenomenon of having a faulty product end up with a measured signature that equals the fault-free signature. This phenomenon is inevitable because all BIST-based methodologies lose test information during the data compression process. The question is how much loss in test quality is encountered

using BIST. In pseudo-random-based BIST designs the probability of aliasing is approximately 2^{-n}, where n is the number of stages in the MISR. Thus, the probability of aliasing is negligible for, for which it is already below 0.1 %. The attainable fault coverage in BIST-based designs is a function of the test length. The higher the test length is the higher the fault coverage.

1.2. Defect Level and Fault Coverage

The object of the test including the BIST is not to ship out the defective products. The **defect level** is the fraction of defective products that passed the test and shipped out. The **product yield** which is used to measure the quality of manufacturing process is the fraction of the manufactured products that is fault-free. The **fault coverage** which is a measure to grade the quality of the test is the fraction of the detected faults by the test. The goal of the test is to keep the defect level to acceptable level; however, the defect level cannot be exactly measured before shipping out the products. Therefore, the defect level should be estimated using the known parameters: the product yield, i.e., the quality of the manufacturing process and the fault coverage, i.e., the quality of the test.

Williams and Brown [4] had first theoretically shown the relationship between the product defect level, the manufacturing yield, and the fault coverage of the test process used to screen it into either a good lot or a bad lot. This well-known relationship is derived assum-

ing that the test equipment is fault-free. Other works [8]-[34] discuss multitude of subjects relating to yield, fault coverage and defect level after test. However, [8]-[34] still assume that the test equipment is fault-free.

Many chips today have BIST circuitry in them. These BIST circuits are used to test the chips and perform the screening described above. Since the BIST hardware is manufactured using the same technology and process as the functional circuits, it is unrealistic to assume that it is fault-free. Moreover, chip manufacturers do not insert any redundancy into their BIST hardware in order to keep the cost down. As a result, the BIST hardware is not made to be fault-tolerant. It is, therefore, imperative to allow the BIST hardware be subjected (during the analysis) to the same defect density as the functional circuits themselves. It is one of the subjects of this thesis to investigate the effects of an unreliable (possibly faulty) BIST environment on the Williams and Brown's equation.

1.3. Fault Diagnosis

The fault diagnosis is the process of locating faults in a digital system usually for recovering from failure by repairing identified faults. On the other hand, the most of LSI products are not repairable except such as memory devices with redundancy or field programmable devices. However, the fault diagnosis is frequently performed in industry for un-repairable LSI products also since semiconductor manufactur-

ers have to improve their design rules or their manufacturing process if the product yield or the quality level is unacceptable. The fault information which is obtained by the fault diagnosis is quite valuable to improve them.

The most of fault diagnosis algorithms use the error information. Two kinds of information are required to identify a fault in the CUT. These are

1) the time information (i.e., the input pattern(s) which causes errors),

and

2) the space information (i.e., scan cells where errors occur for scan based architecture).

Using the time information, fault diagnosis can be performed for a given fault model by methods using dictionary or fault simulation [35]. Using the space information, diagnosis can be performed by cone of logic methods [36]. High resolution diagnostic for a given fault model can be achieved by diagnosis techniques combining the space information with the time information [37][38].

BIST has become the major test technique for today's large scale and high speed system-on-chip (SoC) designs. Since BIST compacts test responses, BIST requires only small tester memory and it can perform at-speed test even if the tester frequency is substantially lower than the frequency of the circuit during test. On the other hand,

BIST causes problems in diagnosis due to its compacted responses. Indeed, pass/fail information obtained from a BIST response analyzer is insufficient time or space information for diagnosis.

In this thesis, we investigate methods to identify every error occurrence in at-speed scan based BIST environment. Every error can be identified even if the circuit test frequency is higher than the tester frequency.

1.4. Thesis Organization

The thesis is organized as follows. Chapter 2 introduces the model of the relationship between defect level, the product yield and the fault coverage reviewing the Williams and Brown's formula and other previous works.

Chapter 3 presents the defect level equation in BISTed products with unreliable BIST circuitry. The Williams and Brown's equation is shown to be a special case of this more generalized formula, i.e. this new formula reduces to Williams and Brown's in the absence of BIST circuitry. The effectiveness of the new proposed formula is presented by the case studies which involve both an early life phase, and a product maturity phase.

Chapter 4 presents the defect level equations in BISTed products that have undergone both a pretest and a functional test. The equations proposed in Chapter 3 are shown to be special cases of these more generalized formulas.

In Chapter 5, the problem of the fault diagnosis in at-speed BIST environment is discussed and the problem of identifying every error occurrence is formulated.

Chapter 6 presents a procedure to identify every error by slow tester in minimum test application time without any hardware for diagnosis to solve the problem presented in Chapter5.

Chapter 7 presents an at-speed BIST architecture which enables the diagnosis procedure proposed in Chapter 6. The diagnosis procedure is also extended to identify erroneous scan chains and to reduce test application time by existing signature analyzers. Some experimental results are shown to demonstrate the effectiveness of the new proposed method.

Finally, Chapter 8 summarizes the conclusions of this thesis and the direction of future works is discussed.

Chapter 2

Defect Level, Yield and Fault Coverage

2.1. Introduction

Williams and Brown [4] had shown the relationship between the product defect level, the manufacturing yield, and the fault coverage of the test process used to screen it into either a good lot or a bad lot. This chapter introduces the model of the relationship between the defect level, the yield and the fault coverage referring the Williams and Browns' works and other previous works are also introduced.

2.2. Definitions of Terms

For the context of this chapter we use the following terms:

Definition. The **fault coverage** (F) is the fraction of the detected faults by the test, i.e.,

$$F = \frac{number\, of\, detected\, faults\, by\, test}{number\, of\, defined\, faults}. \tag{2.1}$$

The fault coverage is a measure to grade the quality of the test is the fraction of the detected faults by the test. The fault coverage can be calculated using fault simulator.

Definition. The **product yield** (Y) is the fraction of the manufactured products that is fault-free, i.e.,

$$Y = \frac{number\, of\, fault\, free\, products}{number\, of\, all\, products\, that\, manufactured}. \tag{2.2}$$

The fault coverage should be 100% to measure the exact product yield. However, it is unrealistic to develop test pattern which detects all the possible faults. Therefore, we have to use the estimated product yield or theoretical product yield instead of the real product yield. The number of test passed products is often used instead of the number of fault-free products as the estimate the product yield. The theoretical product yield can be expressed by the theory of probability. In section 2.4 the product yield will be expressed using the probability of a fault occurrence.

Definition. The **defect level** (D) is the fraction of defective products that pass the test and shipped out i.e.,

$$D = \frac{number\,of\,defective\,products\,that\,shipped\,out}{number\,of\,all\,products\,that\,shipped\,out}. \tag{2.3}$$

Defective products will be shipped out when product yeild is less than 100% and fault coverage of the test pattern is less than 100% which is quite common case in industory. And since 100% fault coverage testing is not avilable, the exact defect level cannot be known. Therefore, only the estimated defect level or theoretical defect level is available. To estimate the defect level, number of returned products is used instead of number of defective products that shipped out. However, this estimated defect level is also unknown before shipping out the products. The theoretical defect level can be expressed using the fault coverage and the product yield as shown in later.

2.3. Previous Works

Williams and Brown's well-known relationship is derived assuming that the test equipment is fault-free.

In [5][6] the effects of an unreliable tester on the resulting yield during a delay (AC) test is discussed. Modeling of yield loss is discussed in [7].

In [8][9] a more generalized fault probability model is introduced to re-examine the Williams and Brown's defect vs. yield equation. Poisson's probability model is used in [8] along with a weighting scheme

biased towards faults that are more likely to occur. The authors of [8] show that even if the distribution of faults is taken to account, the Williams and Brown's equation still holds. A non-uniform fault probability model is introduced in [9].

In [10] a defect level model for other fault types (delay faults and stuck-open faults) as a function of yield and fault coverage is proposed. The authors in [10] show that the relationship between defect level, fault coverage and yield, depicted in the Williams and Brown's equation, still holds for deley fault model and stuck-open fault model.

Other previous works [11]-[34] also discuss multitude of subjects relating to yield, fault coverage and defect level after test. However, the concept of these previous models of the relationship between the defect level and the yield is equal to Williams and Brown's model.

2.4. Williams and Brown's Equation

Let the circuit under test (CUT) have n_c possible faults, each having the same probability of occurrence, p. The yield, Y, is the probability that the circuit is fault-free, i.e.,

$$Y = (1-p)^{n_c}. \tag{2.4}$$

The raw defect level of the product coming out of the manufacturing line (without any test) is

$$D_0 = 1 - Y$$
$$= 1 - (1-p)^{n_c}. \quad (2.5)$$

Assuming that the test process can detect m out of the n_c possible faults, the fault coverage is given by[1]

$$F = \frac{m}{n_c}. \quad (2.6)$$

A circuit that passes the test is guaranteed to be free of any covered faults (m in total), but can still possess an uncovered fault that escaped the test. Since there are $n_c - m$ uncovered faults, the defect level after test is given by

$$D = 1 - (1-p)^{n_c - m}, \quad (2.7)$$

which can be further reduced to

$$D = 1 - [(1-p)^{n_c}]^{(1-\frac{m}{n_c})}$$
$$= 1 - Y^{1-F}. \quad (2.8)$$

The defect level can be estimated by Eq.(2.8). Notice that this equation assumes that the test process is fault-free.

[1]Note: fault detection is independent of fault occurrence.

Example 2.1. Consider a chip manufacturing line with 90% yield. The test procedure has 95% coverage of the functional faults. The test procedure is assumed to be fault-free. Compute the chip defect level after testing.

Solution. We have the following parameters:

$$Y = 0.9, \qquad F = 0.95.$$

The defect level is:

$$\begin{aligned} D &= 1 - 0.9^{1-0.95} \\ &\approx 5.254 \times 10^{-3} \\ &\approx 5254ppm. \end{aligned}$$

□

2.5. Conclusion

This chapter introduces the model of the relationship between the defect level, the yield and the fault coverage referring the Williams and Browns' prior work. The defect level can be estimated by using this equation. However, this equation and other prior works also assume that the test process is fault-free, i.e., a circuit being declared by the test process to be faulty is truly faulty. This is the underlying assumption of the prior models.

Chapter 3

Defect Level in the Presence of an Unreliable BIST

3.1. Introduction

Chapter 2 introduces the model of the relationship between the defect level, the yield and the fault coverage. However, this equation assumes that the test process is fault-free.

Many chips today have BIST circuitry in them. These BIST circuits are used to test the chips and perform the screening described above. Since the BIST hardware is manufactured using the same technology and process as the functional circuits, it is unrealistic to assume that it is fault-free. Moreover, chip manufacturers do not insert any redundancy into their BIST hardware in order to keep the cost down. As a result, the BIST hardware is not made to be fault-tolerant. It is,

therefore, imperative to allow the BIST hardware be subjected (during the analysis) to the same defect density as the functional circuits themselves. It is the subject of this chapter to investigate the effects of an unreliable (possibly faulty) BIST environment.

3.2. Unreliable BIST

For the context of this chapter we use the following definition:

Definition. BIST circuitry is said to be *unreliable* if

$$\Pr\{BIST\, is\, faulty\} > 0.$$

Consequently, BIST circuitry is considered reliable if

$$\Pr\{BIST\, is\, faulty\} = 0.$$

Generally speaking, there are two side effects resulting from an unreliable BIST. One side effect is to cause a good product (i.e., no functional defects present) be declared faulty, resulting in a yield loss [5]-[7]. A second side effect which is discussed in this chapter is to have a bad product be passed as good, increasing the shipped-product defect level.

3.3. Enhanced Equation in the Presence of an Unreliable BIST

3.3.1 Fault Detection Scenarios

The BIST hardware tests the CUT in order to determine whether it is faulty or fault-free. The BIST hardware is an entity residing on chip and is separate from the CUT. If the BIST hardware happens to be fault-free, the outcome of the test will depend on its fault coverage against functional faults, which the Williams and Brown's equations already accounts for.

If the BIST circuitry is faulty, the outcome of the test will also depend upon the ability of the impaired BIST to detect CUT faults on one hand, and its ability not to "accuse" a fault-free CUT as being faulty, on the other. As a result, the test may encounter both fault escapes [3] and yield loss[1] [5]-[7]. Notice that in the Williams and Brown's case a yield loss is not possible. Table 3.1 displays all the possible test outcomes in the presence of an unreliable BIST hardware. The assumption in Table 3.1 is that BIST faults do not affect the functional operation of the CUT.

In this section we derive a new set of formulas that cover the case where there is no knowledge prior to the launch of the CUT test as to the state of the BIST hardware. This uncertainty as to whether or not the BIST hardware is faulty or fault-free is likely to cause an

[1]Note: Yield loss is the probability that a fault-free circuit fails the test.

Table 3.1. Test outcomes in the presence of an unreliable BIST

CUT	BIST	Test Result	CUT Fault Condition
Fault-free	Fault-free	Pass	No fault. Case covered by Williams and Brown.
Faulty	Fault-free	Pass	Fault escapes. Case covered by Williams and Brown.
		Fail	Fault detected. Case covered by Williams and Brown.
Fault-free	Faulty	Pass	No fault.
		Fail	No fault. Case of yield loss.
Faulty	Faulty	Pass	Fault escapes.
		Fail	Fault detected.

increase in the shipped-product defect level. This increment in defect level is later analyzed based upon this newly derived formulas.

3.3.2 The Enhanced Equations

In the sequel we will refer to the product functional circuits as the CUT. We use the following parameters in our analysis.

D - Product defect level after test under fault-free BIST hardware

D' - Product defect level after test under unreliable BIST hardware

F - Fault coverage of the CUT under fault-free BIST hardware

F' - Effective fault coverage of the CUT in the presence of an unreliable BIST hardware

Y - Product yield

p - Fault occurrence probability in both CUT and BIST hardware

A_c - CUT area

A_b - BIST area

n_c - Total number of possible faults in the CUT, $n_c = GA_c$, where G is a constant.

n_b - Total number of possible faults in the BIST hardware, $n_b = GA_b$, where G is a constant.

m - Number of CUT faults covered by fault-free BIST hardware

m' - Expected number of CUT faults covered by an unreliable BIST hardware

k - Average number of CUT faults covered by a faulty BIST hardware

α - Ratio between BIST area and CUT area

ρ - Fault coverage alteration factor

The meaning of α and ρ will become evident from the following analysis.

Notice that we are allowing the test procedure, as conducted by the faulty BIST hardware, to detect CUT faults. The number of CUT faults detected by a faulty BIST depends upon the type of fault actually existing in the BIST hardware. Let k_{f_i} $(0 \leq k_{f_i} \leq n_c)$ be the number of detected CUT faults in the presence of BIST fault f_i $(1 \leq i \leq n_b)$. Further denote by k the average number of all such k_{f_i}s.

We proceed to calculate m', the expected number of CUT faults covered by the unreliable BIST.

$$m' = m \times \Pr\{Fault\, free\, BIST\} + k \times \Pr\{Faulty\, BIST\}$$

Therefore,

$$m' = m(1-p)^{n_b} + k[1 - (1-p)^{n_b}]. \tag{3.1}$$

The expected CUT fault coverage, as conducted by the BIST circuitry, is:

$$\begin{aligned} F' &= \frac{m'}{n_c} \\ &= \frac{m}{n_c}(1-p)^{n_b} + \frac{k}{n_c}[1-(1-p)^{n_b}] \\ &= \frac{m}{n_c}\{(1-p)^{n_b} + \frac{k}{m}[1-(1-p)^{n_b}]\} \end{aligned}$$

Define:

$$\rho = \frac{k}{m} \tag{3.2}$$

to be the *fault coverage alteration factor*. Notice that ρ can be larger than 1. The reason for this is that it is possible for a BIST fault to create a situation where every CUT, good or bad, is rejected by the test. We refer to this case as a *catastrophic* case. Thus, the largest ρ may become is $n_c/m = 1/F$. The possible range for ρ is, therefore, $0 \leq \rho \leq 1/F$. Notice also that the case of $\rho > 1$ is actually a case of fault coverage "amplification" rather than a case of fault coverage reduction.

We can rewrite the expected CUT fault coverage, as exercised by the unreliable BIST, as:

$$F' = F\left\{(1-p)^{n_b} + \rho[1-(1-p)^{n_b}]\right\}. \tag{3.3}$$

We call this expected CUT fault coverage the effective CUT fault coverage in the presence of the unreliable BIST.

Eq.(3.3) can also be written as:

$$F' = F\left[Y^{\frac{n_b}{n_c}} + \rho(1 - Y^{\frac{n_b}{n_c}})\right]. \tag{3.4}$$

Denote by

$$\alpha = \frac{n_b}{n_c} = \frac{GA_b}{GA_c} = \frac{A_b}{A_c}$$

(see relationship in list of parameters earlier in this section). The effective fault coverage, F', can now be written as

$$F' = F[Y^{\alpha} + \rho(1 - Y^{\alpha})]. \tag{3.5}$$

The new formula relating the product defect level to the yield and the effective fault coverage becomes:

$$\begin{aligned} D' &= 1 - (1-p)^{n_c - m'} \\ &= 1 - (1-p)^{n_c\left(1 - \frac{m'}{n_c}\right)}. \end{aligned}$$

Notice that D' behaves similar to D (see Eq.(2.7)), with the exception of the replacement of m by m'. This leads to:

$$D' = 1 - Y^{1-F'}. \tag{3.6}$$

Example 3.1. Consider a chip manufacturing line with 90% yield. The chips are screened using their BIST circuitry. The BIST circuitry constitutes 5% of the entire chip area. The BIST procedure has 95% coverage of the functional faults when assumed to be fault-free, and only 40% coverage when assumed faulty. Compute the chip defect level after its BIST screening.

Solution. We have the following parameters:

$$\alpha = \frac{5}{95} = \frac{1}{19} = 5.263 \times 10^{-2}$$

$$\rho = \frac{40}{95} \approx 0.4211$$

$$F' = 0.95 \times [0.9^{5.263\times10^{-2}} + 0.4211 \times (1 - 0.9^{5.263\times10^{-2}})]$$

$$\approx 0.9470$$

$$D' \approx 1 - 0.9^{1-0.9470} \approx 1 - 0.9^{0.053} \approx 5.569 \times 10^{-3}$$

$$\approx 5569 ppm$$

Notice that if we ignore the effects of the BIST, the defect level is:

$$D = 1 - 0.9^{1-0.95} = 1 - 0.9^{0.05}$$

$$\approx 5.254 \times 10^{-3}$$

$$\approx 5254 ppm$$

□

Special Cases

It is interesting to take note of the following special cases:

1. If there is no BIST circuitry ($\alpha = 0$), we have $F' = F$, and $D' = D$. This is the Williams and Brown's case.

2. If there is no CUT fault coverage alteration by the BIST circuitry ($\rho = 1$), we have $F' = F$, and $D' = D$. This is, again, the Williams and Brown's case.

3. If the BIST procedure has zero coverage against functional faults while being itself faulty, then $\rho = 0$. The effective fault coverage, in this case, reduces to:

$$F' = FY^{\alpha}. \tag{3.7}$$

4. For the extreme case of $\rho = 1/F$ we get:

$$F' = 1 - (1 - F)Y^{\alpha}. \tag{3.8}$$

3.3.3 The Impact of the BIST Impurity

The impact of the BIST impurity on the product defect level can be best measured by the differential

$$\Delta D' = D' - D,$$

or, even better, by its normalized form,

$$\Delta D'/D.$$

When a product manufacturing process reaches maturity, its yield is close to 1, and its defect level is close to 0 ($Y \approx 1$, $D \approx 0$). By using calculus approximation techniques, and under the restrictions just described, $\Delta D'$ and $\Delta D'/D$ can be approximated as follows.

$$\begin{aligned} D' &= 1 - Y^{1-F'} \\ &= 1 - \left(Y^{1-F}\right)^{\frac{1-F'}{1-F}} \\ &= 1 - (1-D)^{\frac{1-F'}{1-F}} \end{aligned} \tag{3.9}$$

But since for this case $D \approx 0$, Eq.(3.9) can be approximated as:

$$\begin{aligned} D' &\approx 1 - \left(1 - \frac{1-F'}{1-F} D\right) \\ &= \frac{1-F'}{1-F} D. \end{aligned} \tag{3.10}$$

From Eqs.(3.5) and (3.10) we get:

$$\begin{aligned} \frac{\Delta D'}{D} &\approx \frac{F-F'}{1-F} \\ &= \frac{F(1-\rho)(1-Y^{\alpha})}{1-F}, \end{aligned} \tag{3.11}$$

and since for $Y \approx 1$ we can approximate

$$\begin{aligned} 1 - Y^{\alpha} &= 1 - [1 - (1 - Y)]^{\alpha} \\ &\approx \alpha(1 - Y), \end{aligned} \tag{3.12}$$

then, Eq.(3.11), for the case where $Y \approx 1$, can be further approximated as:

$$\frac{\Delta D'}{D} \approx \frac{F\alpha(1-\rho)(1-Y)}{1-F}. \tag{3.13}$$

From Eq.(2.8), and for $Y \approx 1$, we get:

$$\begin{aligned} D &= 1 - Y^{1-F} \\ &= 1 - [1 - (1 - Y)]^{1-F} \\ &\approx (1 - Y)(1 - F). \end{aligned} \tag{3.14}$$

From Eqs.(3.13) and (3.14) we get:

$$\Delta D' \approx F\alpha(1-\rho)(1-Y)^2. \tag{3.15}$$

Example 3.2. Consider again the case described in Example 3.1. By using Eqs. (3.13) and (3.14) we get:

$$\Delta D' \approx 0.95 \times 5.263 \times 10^{-2} \times (1 - 0.4211\,) \times (1 - 0.9)^2$$

$$\approx 289ppm$$

$$\frac{\Delta D'}{D} \approx \frac{0.95 \times 5.263 \times 10^{-2} \times (1 - 0.4211\,) \times (1 - 0.9)}{1 - 0.95}$$

$$\approx 5.789 \times 10^{-2}$$

Compare these to the exact results of 315ppm and 5.995×10^{-2} respectively, derived from Example 3.1. □

3.4. Some Typical Behaviors

During the product's early life its yield is relatively low. This is mostly due to not quite knowing how to best fine-tune the manufacturing parameters of an emerging new technology. Typical early life yields may vary between 40% to 60%, even though lower figures are also possible. As the manufacturing process matures, the yield figures may rise to as much as 90%, or even higher. In this section we try to shed some light on the impact of the BIST unreliability during these two distinct periods of the product's life. The parameters chosen in this study reflect likely operating conditions of an IC manufacturing fab.

3.4.1 Early Life Impact

In order to study the impact of the BIST circuitry on the product's early life defect level after test, we let $0.4 \leq Y \leq 0.6$. The other parameter ranges are $0.9 \leq F \leq 0.99$, $0.4 \leq \rho \leq 1.1$ and $0.05 \leq \alpha \leq 0.1$. These parameter ranges are used again in the next subsection, and they reflect practical values for BIST-based IC products.

In Fig.3.1 we show the behavior of F'/F and $\Delta D'/D$ as a function of Y, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$ and $\alpha = 0.05$. In Fig.3.2 we show the behavior of F'/F and $\Delta D'/D$ as a function of ρ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.4$ and $\alpha = 0.05$. In Fig.3.3 we show the behavior of F'/F and $\Delta D'/D$ as a function of α, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$ and $Y = 0.4$. In Fig.3.4 we show the behavior of F'/F and $\Delta D'/D$ as a function of F, while keeping the other parameters fixed at $Y = 0.4$, $\rho = 0.4$ and $\alpha = 0.05$.

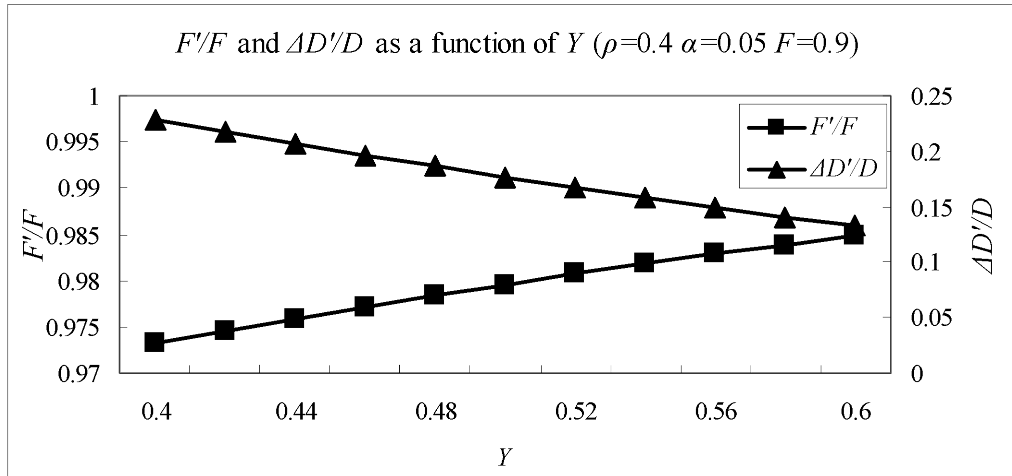

Figure 3.1. F'/F and $\Delta D'/D$ as a function of Y (at early life)

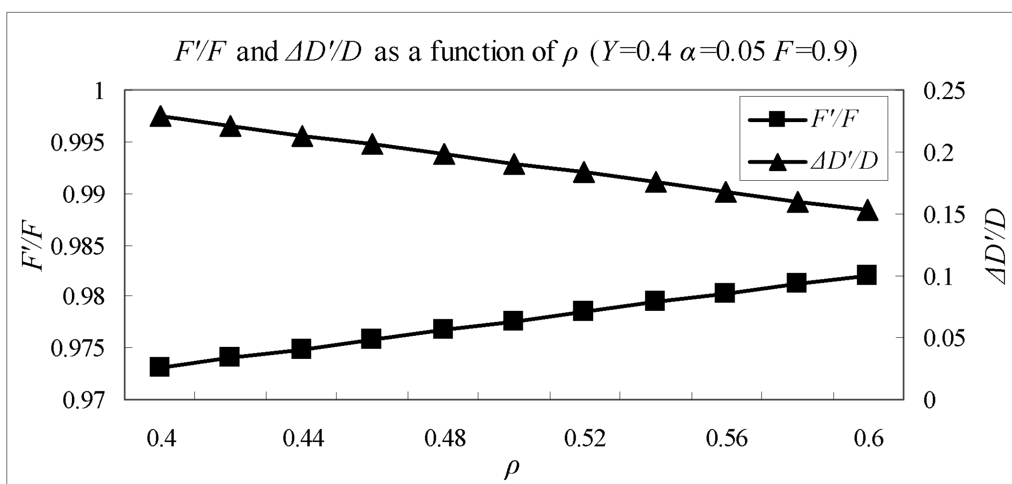

Figure 3.2. F'/F and $\Delta D'/D$ as a function of ρ (at early life)

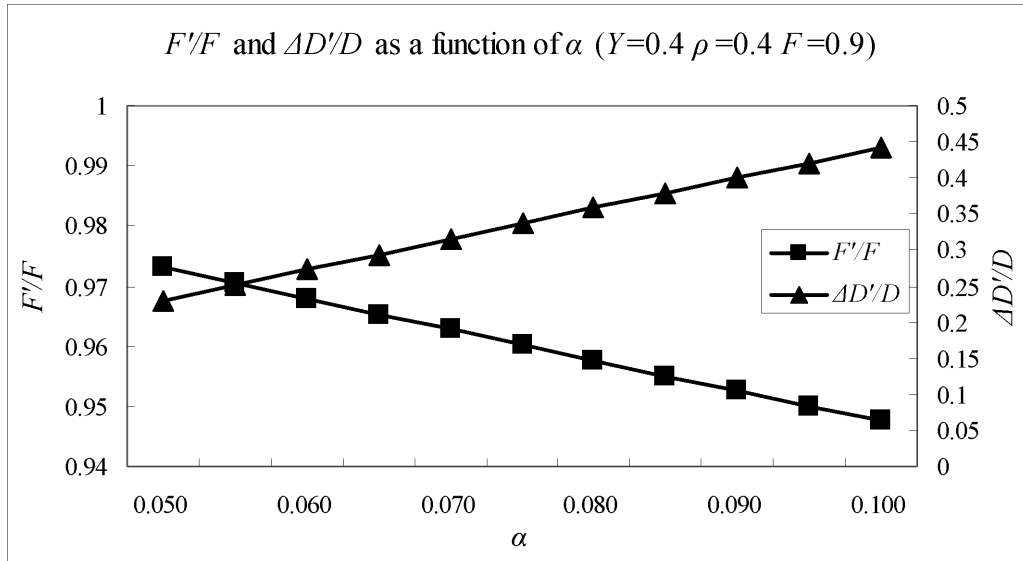

Figure 3.3. F'/F and $\Delta D'/D$ as a function of α (at early life)

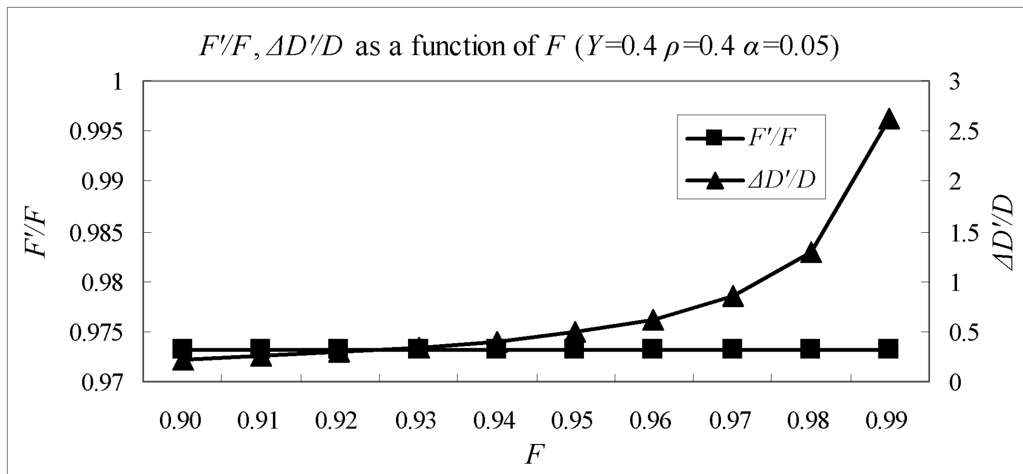

Figure 3.4. F'/F and $\Delta D'/D$ as a function of F (at early life)

3.4.2 Impact at Maturity

Since at maturity $Y \approx 1$, we plot F'/F and $\Delta D'/D$ for the parameter ranges $0.9 \leq Y \leq 0.95$, $0.9 \leq F \leq 0.99$, $0.4 \leq \rho \leq 1.1$ and $0.05 \leq \alpha \leq 0.1$.

In Fig.3.5 we show the behavior of F'/F and $\Delta D'/D$ as a function of Y, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$ and $\alpha = 0.05$. In Fig.3.6 we show the behavior of F'/F and $\Delta D'/D$ as a function of ρ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.9$ and $\alpha = 0.05$. In Fig.3.7 we show the behavior of F'/F and $\Delta D'/D$ as a function of α, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$ and $Y = 0.9$. In Fig.3.8 we show the behavior of F'/F and $\Delta D'/D$ as a function of F, while keeping the other parameters fixed at $Y = 0.9$, $\rho = 0.4$ and $\alpha = 0.05$.

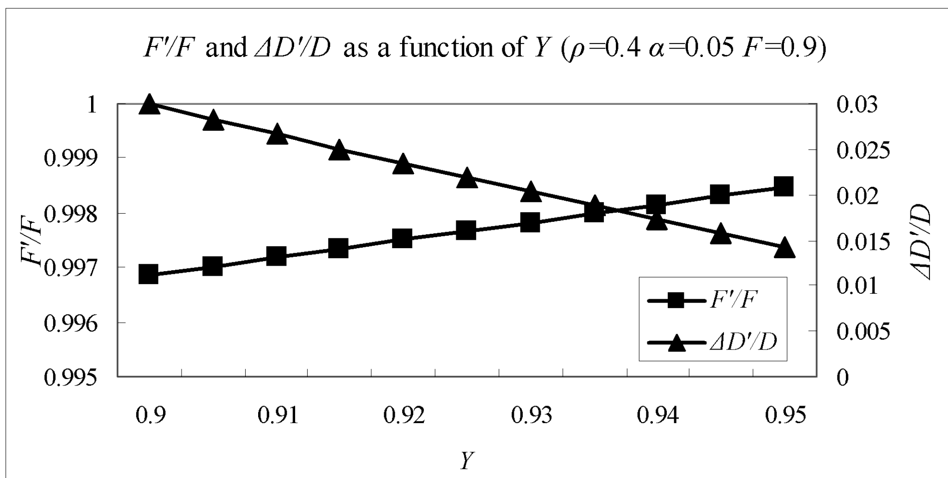

Figure 3.5. F'/F and $\Delta D'/D$ as a function of Y (at maturity stage)

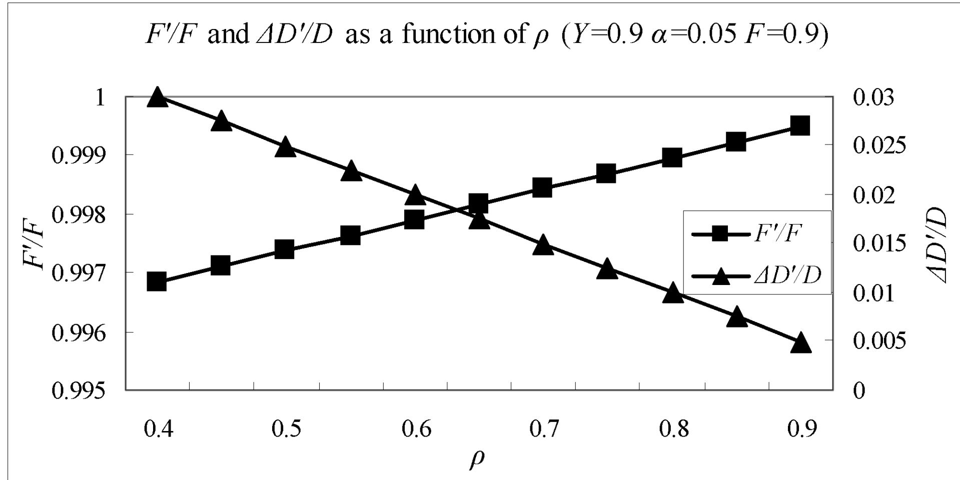

Figure 3.6. F'/F and $\Delta D'/D$ as a function of ρ (at maturity stage)

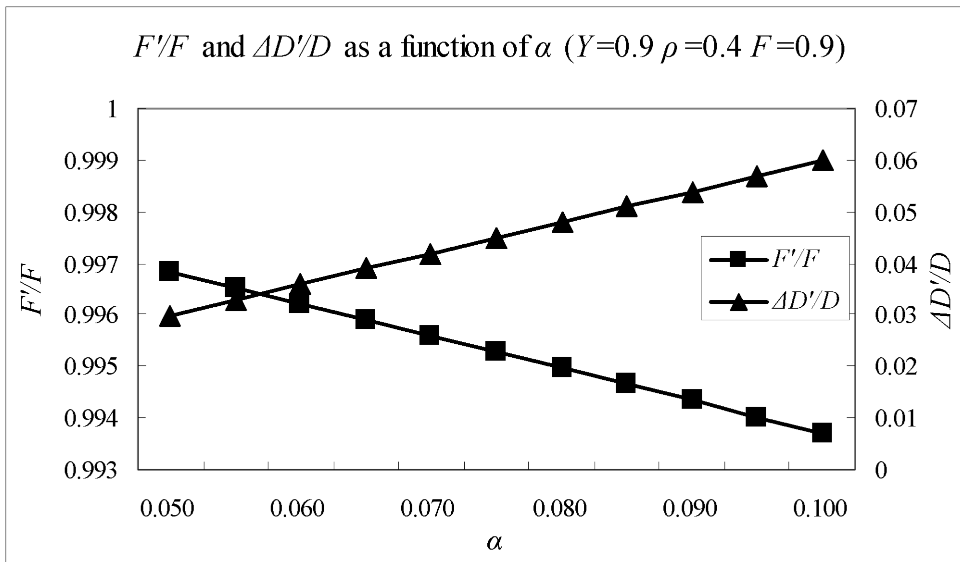

Figure 3.7. F'/F and $\Delta D'/D$ as a function of α (at maturity stage)

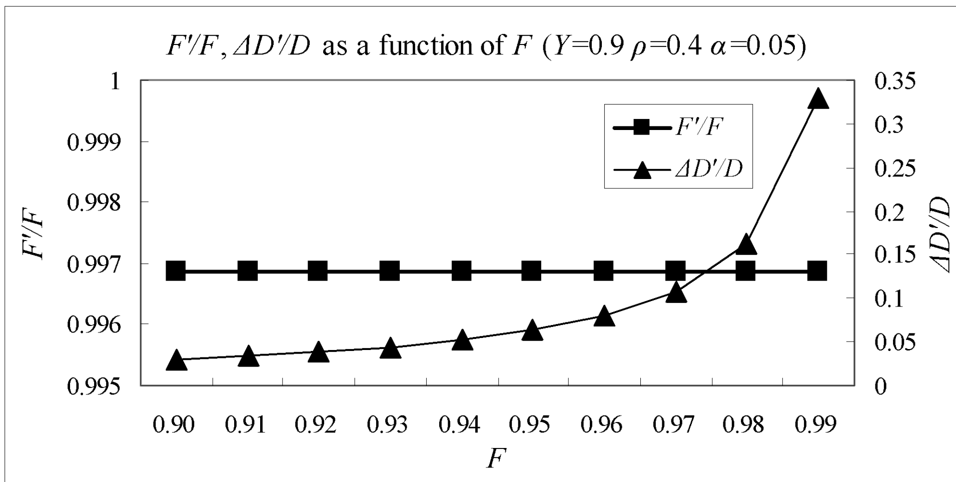

Figure 3.8. F'/F and $\Delta D'/D$ as a function of F (at maturity stage)

3.4.3 The Impact Trend

At maturity, and for fault coverages under 98%, the impact of the BIST circuitry unreliability is mostly minor. In this case, the drop in fault coverage, and the defect level increment, rising from the presence of an unreliable BIST circuitry, is relatively small (few percent). During early life, on the other hand, the impact of the BIST circuitry unreliability is far more significant. Even for fault coverages around, or under, 98%, the defect level increment can easily exceed 100%.

It is important to note that regardless of which stage in life the product is in, when F is very close to 1 (say $F = 0.99$), the $\Delta D'$ differential starts to grow substantially faster (see trend in Figs.3.4 & 3.8). The reason for this phenomenon is that in this range of fault coverages D is already very small, and the impact of the unreliable BIST makes D' so much worse compared to D. Since $\Delta D' = D' - D$, this differential worsens when F approaches 0.99 (see Fig.3.9). The behavior of $\Delta D'/D$ is even more pronounced since $\Delta D'$ is increasing while D is decreasing. In these range of fault coverages, therefore, there is a very significant departure from the Williams and Browns' results.

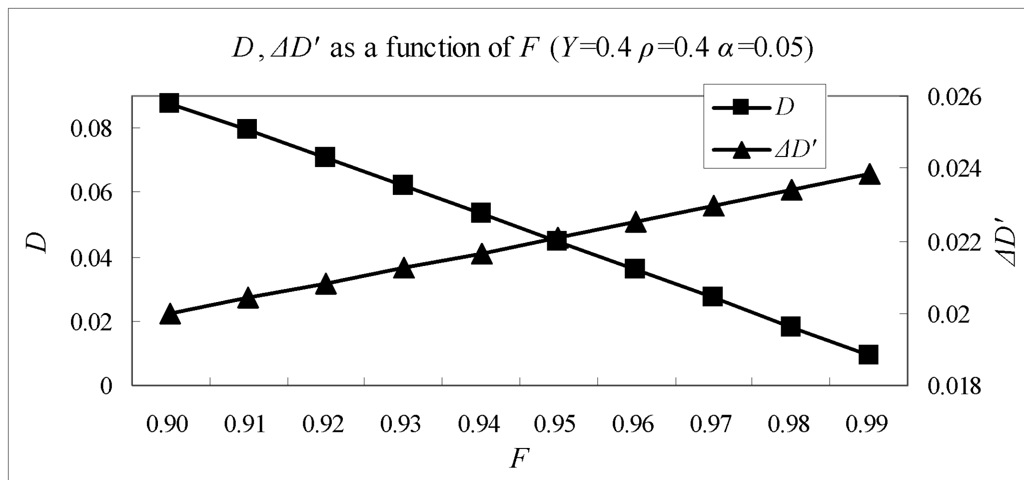

Figure 3.9. D and $\Delta D'$ as a function of F

3.5. Conclusion

This chapter extends Williams and Brown's formula for products with BIST hardware, where the screening into pass/fail lots is done by the BIST hardware itself. The BIST hardware is assumed to suffer from the same defect density as the functional circuits themselves. The impact of this unreliable BIST is studied in detail. We have shown that the general form of Williams and Brown's formula still holds in this case, provided the CUT's fault coverage is replaced by the CUT's effective fault coverage. If the BIST circuitry does not possess a catastrophic fault, its impact is to increase the defect level of the products passing the test procedure. If the BIST does possess a catastrophic fault, it may decrease the defect level of the products passing the test procedure. This "artificial" improvement in defect level comes at the expense of having to reject almost every circuit, good or bad, subjected to the test. Formulas to assess these impacts have been derived.

During maturity, and for fault coverages under 98%, the impact of the BIST circuitry unreliability is minor (about a 5% departure). In this case the Williams and Brown's formula constitutes a reasonable approximation even in the presence of an unreliable BIST. For fault coverages above 98%, however, the defect level increment can easily grow by 30-50%. Therefore, the Williams and Brown's formula no longer represents the true situation.

During early life, and even for fault coverages below 98%, we see a considerable departure from the Williams and Brown's results. The

departures in defect levels, for example, may be as small as 20% and as high as 150%. These departures worsen for fault coverages above 98%. Thus, the Williams and Brown's formula cannot be used for this stage in the product's life. It is paramount to use this enhanced equations instead.

Chapter 4

Effect of BIST Pretest on Defect Level

4.1. Introduction

The object of this chapter is to analyze the potential benefits of conducting a BIST pretest before launching a functional test of ICs during post manufacturing screening. In Chapter 3, formulas are derived to assess the impact of the BIST circuitry on the final integrated circuit (IC) defect level after test. It was assumed that no measures are taken to assure that the BIST circuitry is, in fact, working properly before the initiation of the functional test. The formulas derived in Chapter 3 show a considerable departure from those in [4]. The object of this Chapter is to analyze the potential benefits of conducting a BIST pretest before launching a functional test of ICs during post manu-

facturing screening. As the study shows, in many cases the potential benefits outweigh any potential risks.

4.2. BIST Pretest

In this chapter we assume that a BIST pretest is first conducted in order to rid of all chips that fail it. Only chips whose BIST circuitry has passed the pretest are kept, while the rest are discarded. The BIST pretest, however, is assumed to have only a limited coverage against its own faults. The reason for this is that only primitive operations, such as scan and capture, are possible during pretest. A more comprehensive BIST pretest will require the use of external test equipment, which defeats the incentive for BIST altogether. Thus, only a subset of chips with faulty BIST can be identified and eliminated. Chips with faulty BIST that escape the pretest are used later on to conduct the functional test. Generally speaking, therefore, there are two side effects resulting from this BIST pretest. One side effect is to cause a good product (i.e., no functional defects present) to be dropped, resulting in a yield loss. A second side effect is to have a bad product be passed as good by a faulty BIST during the functional test, increasing the shipped-product defect level.

4.3. Effects of BIST Pretest

4.3.1 Analysis

In this case the BIST circuitry undergoes an operation pretest in order to discard of any chips with faulty BIST in them. This pretest, however, is conducted by the BIST circuitry itself, and is far from being comprehensive. In this primitive test, the LFSRs/MISRs are cycled, starting with a known seed, to see if they can end up with a correct signature after a predetermined number of clocks. BIST circuitry that passes this pretest is by no means guaranteed to be fault-free. BIST circuitry that passes this test can still possess, for example, interconnect faults between the LFSRs/MISRs and the CUT. This pretest, therefore, has relatively low fault coverage against its own faults. The reason why a primitive, rather than a comprehensive, pretest is conducted is that the latter requires the use of external test equipment that totally defeats the purpose of BIST to begin with.

We use the following notations in the following analysis.

D - Product defect level after test under fault-free BIST hardware

D' - Product defect level after test under unreliable BIST hardware and without BIST pretest

D'' - Product defect level after test under unreliable BIST hardware and with BIST pretest

F - Fault coverage of the CUT under fault-free BIST hardware

F' - Effective fault coverage of the CUT in the presence of an unreliable BIST hardware and without BIST pretest

F'' - Effective fault coverage of the CUT in the presence of an unreliable BIST hardware and with BIST pretest

Y - Product yield

p - Fault occurrence probability in both CUT and BIST hardware

n_c - Total number of possible faults in the CUT

n_b - Total number of possible faults in the BIST hardware

m - Number of CUT faults covered by fault-free BIST hardware

m_b - The number of BIST circuitry faults covered by the BIST operation pretest

m' - Expected number of CUT faults covered by an unreliable BIST hardware and without BIST pretest

m'' - Expected number of CUT faults covered by an unreliable BIST hardware and with BIST pretest

k - Average number of CUT faults covered by a faulty BIST hardware and without BIST pretest

k^* - Average number of CUT faults covered by a faulty BIST hardware and with BIST pretest

α - Ratio between BIST area and the CUT area

ρ - CUT Fault coverage alteration factor without BIST pretest

ρ' - CUT Fault coverage alteration factor with BIST pretest

μ - BIST circuitry fault coverage during pretest

λ - Yield coefficient

Let k be the average number of CUT faults detected by a faulty BIST that did not undergo an operation pretest. Let k^* be the average number of CUT faults detected by a faulty BIST that passed the operation pretest.

The parameter ρ is the CUT fault coverage alteration factor without BIST pretest, and is given by

$$\rho = \frac{k}{m} \quad (0 \leq \rho \leq 1/F).$$

The parameter ρ' is the CUT fault coverage alteration factor with BIST pretest, and is given by,

$$\rho' = \frac{k^*}{m} \quad (0 \leq \rho' \leq 1/F).$$

We proceed to calculate m'', the expected number of CUT faults covered by BIST.

Since the BIST circuitry that conducts the CUT test has passed the operation pretest, it is guaranteed to be free of the m_b faults covered by it. Therefore,

$$m'' = m \times \Pr\{Fault\,free\,BIST\} + k^* \times \Pr\{Faulty\,BIST\},$$

$$m'' = m(1-p)^{n_b-m_b} + k^*[1-(1-p)^{n_b-m_b}]. \tag{4.1}$$

The expected CUT fault coverage, as conducted by the BIST circuitry, is:

$$\begin{aligned} F'' &= \frac{m''}{n_c} \\ &= \frac{m}{n_c}(1-p)^{n_b-m_b} + \frac{k^*}{n_c}[1-(1-p)^{n_b-m_b}] \\ &= \frac{m}{n_c}\{(1-p)^{n_b-m_b} + \frac{k^*}{m}[1-(1-p)^{n_b-m_b}]\}, \\ F'' &= F\left[Y^{\frac{n_b-m_b}{n_c}} + \rho'(1-Y^{\frac{n_b-m_b}{n_c}})\right]. \end{aligned} \tag{4.2}$$

The exponent in Eq.(4.2) can be written as

$$\begin{aligned} \frac{n_b-m_b}{n_c} &= \frac{n_b}{n_c}(1-\frac{m_b}{n_b}) \\ &= \alpha(1-\mu), \end{aligned}$$

where

$$\alpha = \frac{n_b}{n_c},$$

and

$$\mu = \frac{m_b}{n_b}.$$

We define

$$\lambda = \alpha(1 - \mu).$$

We call λ the yield coefficient, $0 \leq \lambda \leq 1$. The parameter μ is the BIST circuitry fault coverage during the pretest.

The effective fault coverage, F'', can now be written as

$$F'' = F[Y^{\lambda} + \rho'(1 - Y^{\lambda})], \tag{4.3}$$

and the defect level after the CUT functional test becomes

$$D'' = 1 - Y^{1-F''}. \tag{4.4}$$

Example 4.1. Consider a chip manufacturing line with 90% yield. The chips are screened using their BIST circuitry. The BIST circuitry constitutes 5% of the entire chip area. The BIST procedure has 95% coverage of the functional faults when assumed to be fault-free, and only 40% coverage when assumed faulty. Let the BIST circuitry undergo a pretest with self-fault coverage of $\mu = 0.3$. All chips failing the pretest are discarded. The chips passing the pretest are kept and used to perform the BIST CUT test. Chips that fail the CUT test are discarded. Compute the defect level of the chips passing both tests.

Solution. We have the following parameters:

$$\alpha = \frac{5}{95} = \frac{1}{19}, \qquad \mu = 0.3,$$

$$\lambda = \frac{0.7}{19} \approx 3.68 \times 10^{-2}, \qquad \rho' = \frac{40}{95} \approx 0.421,$$

$$F'' = 0.95 \times [0.9^{3.68\times10^{-2}} + 0.421 \times (1 - 0.9^{3.68\times10^{-2}})]$$

$$\approx 0.9479,$$

$$D'' \approx 1 - 0.9^{1-0.9479} \approx 1 - 0.9^{0.0521} \approx 5.474 \times 10^{-3}$$

$$\approx 5474ppm,$$

which is 95ppm smaller than the detect level obtained without a pretest in Chapter 3. □

Special Cases

It is interesting to take note of the following special cases:

If there is no BIST circuitry ($\alpha = 0$), we have $F'' = F$, and $D'' = D$. This is the Williams and Brown's case. Also, in the case of an ideal BIST pretest, we have $\mu = 1$. In this case also, the formulas reduce to the Williams and Brown's case. The reason for this is that when $\mu = 1$ the BIST pretest is able to rid of all the chips with faulty BIST hardware. The CUT, therefore, is tested by a reliable "tester",

which was the underlying assumption used by Williams and Brown in the first place.

If the BIST procedure has zero coverage against functional faults while being itself faulty, then $\rho' = 0$. The effective fault coverage, in this case, reduces to:

$$F'' = FY^{\lambda}. \tag{4.5}$$

Note that the case of $\mu = 0$ is the case of a "pretest with no coverage against its own faults". This is, therefore, identical to the case of CUT screening without a BIST pretest. The formulas in this case reduce to those derived in Chapter 3.

The Impact of the BIST Impurity

We measure the impact of the BIST impurity on the product defect level by the differential

$$\Delta D'' = D'' - D,$$

or, equivalently, by its normalized form,

$$\Delta D''/D.$$

When a product manufacturing process reaches maturity, its yield is close to 1. Furthermore, in most real-life cases $F'' \approx F$,$\lambda << 1$.

By using calculus approximation techniques we get two sets of approximation formulas. The first set:

$$\Delta D'' \approx F\lambda(1-\rho')\ln^2 Y, \tag{4.6}$$

and

$$\frac{\Delta D''}{D} \approx \frac{F\lambda(1-\rho')\ln^2 Y}{(1-F)(1-Y)}. \tag{4.7}$$

The second set of formulas can be obtained from the first set by letting $\ln Y \approx -(1-Y)$:

$$\Delta D'' \approx F\lambda(1-\rho')(1-Y)^2, \tag{4.8}$$

$$\frac{\Delta D''}{D} \approx \frac{F\lambda(1-\rho')(1-Y)}{1-F}. \tag{4.9}$$

For the catastrophic case ($\rho' = 1/F$), we get from Eqs.(4.8) and (4.9):

$$\Delta D''|_{cat} \approx -\lambda(1-F)(1-Y)^2, \tag{4.10}$$

$$\left.\frac{\Delta D''}{D}\right|_{cat} \approx -\lambda(1-Y). \tag{4.11}$$

4.3.2 Sizing the Effect of the BIST Pretest

It is interesting to assess the influence of the BIST pretest on the shipped-product defect level. To assess this impact we compute the

difference in $\Delta D/D$ with and without the BIST pretest. This will help determine if the alteration in product defect level, achieved as a result of the BIST pretest, is worth the added risk of having to compromise the loss in product yield.

Let δD be the difference between the two defect level differentials with and without a BIST pretest. Let $\delta D/D$ denote the difference between the two normalized differentials (normalized against the Williams and Brown's case). We, therefore, have:

$$\delta D = \Delta D' - \Delta D''.$$

At maturity, and under relatively high fault coverages, we get:

$$\delta D \approx F\alpha[\mu(1-\rho') + (\rho' - \rho)] \ln^2 Y, \tag{4.12}$$

and

$$\frac{\delta D}{D} \approx \frac{F\alpha[\mu(1-\rho') + (\rho' - \rho)] \ln^2 Y}{(1-F)(1-Y)}. \tag{4.13}$$

We define the differential impact factor as:

$$\zeta = \frac{\Delta D'}{\Delta D''}.$$

At maturity, and under relatively high fault coverages, we get:

$$\zeta \approx \frac{1-\rho}{(1-\mu)\,(1-\rho')}. \tag{4.14}$$

There is no good reason why k^* should (statistically) be any different from k. The reason for this is that the BIST operation pretest will only guarantee that those who pass it are free of some, but not all, of the totality of possible faults. The eliminated BIST faults will remove some faults with detectability larger than k, and some faults with detectability smaller than k, not affecting (in principle) the average k.

By letting $k^* \approx k$ we get $\rho' \approx \rho$. In this case we, therefore, get:

$$\delta D \approx F\alpha\mu(1-\rho)\ln^2 Y, \tag{4.15}$$

and

$$\frac{\delta D}{D} \approx \frac{F\alpha\mu(1-\rho)\ln^2 Y}{(1-F)(1-Y)}. \tag{4.16}$$

By letting $\ln Y \approx -(1-Y)$ in Eq.(4.16) we get:

$$\frac{\delta D}{D} \approx \frac{F\alpha\mu(1-\rho)(1-Y)}{1-F}, \tag{4.17}$$

$$\zeta \approx \frac{1}{1-\mu}. \tag{4.18}$$

For the catastrophic case ($\rho = 1/F$), we get from Eq.(4.17):

$$\left.\frac{\delta D}{D}\right|_{cat} \approx -\alpha\mu(1-Y). \tag{4.19}$$

Example 4.2. As a continuation of Example 4.1, we use Eqs.(4.15) and (4.16) to assess the BIST pretest impact on the final product defect level.

Solution. We have

$$\begin{aligned}\delta D &\approx 0.95 \times 0.0526 \times 0.3 \times (1 - 0.421) \times \ln^2 0.9 \\ &\approx 96ppm.\end{aligned}$$

Compare this to the 95ppm computed in Example 4.1.

Similarly,

$$\begin{aligned}\frac{\delta D}{D} &\approx \frac{\delta D}{(1-0.95)(1-0.9)} \\ &\approx 0.019,\end{aligned}$$

which is less than 2%.

The differential impact factor in this case is:

$$\zeta \approx \frac{1}{0.7} = 1.4285,$$

which indicates that the BIST pretest strategy has reduced the defect level by a factor larger than 1.4. □

4.4. Some Typical Behaviors

During the product's early life its yield is relatively low. This is mostly due to not quite knowing how to best fine-tune the manufacturing parameters of an emerging new technology. Typical early life yields may vary between 40% to 60%, even though lower figures are also possible. As the manufacturing process matures, the yield figures may rise to as much as 90%, or even higher. In this section we try to shed some light on the impact of the BIST pretest during these two distinct periods of the product's life. The parameters chosen in this study reflect likely operating conditions of an IC manufacturing fab. In the following study we assume $\rho' \approx \rho$.

4.4.1 Early Life Impact

In order to study the impact of the BIST pretest on the product's early life defect level after the CUT test, we let $0.4 \leq Y \leq 0.6$. The other parameter ranges are $0.9 \leq F \leq 0.99$, $0.4 \leq \rho \leq 0.6$, $0.05 \leq \alpha \leq 0.1$ and $0.4 \leq \mu \leq 0.6$. These parameter ranges are used again in the next subsection, and they reflect practical values for BIST-based IC products.

In Fig.4.1 we show the behavior of F''/F and $\delta D/D$ as a function of Y, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.2 we show the behavior of F''/F and $\delta D/D$ as a function of ρ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.3 we show the

behavior of F''/F and $\delta D/D$ as a function of α, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$, $\mu = 0.5$ and $Y = 0.4$. In Fig.4.4 we show the behavior of F''/F and $\delta D/D$ as a function of F, while keeping the other parameters fixed at $Y = 0.4$, $\rho = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.5 we show the behavior of F''/F and $\delta D/D$ as a function of μ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.4$, $\rho = 0.4$ and $\alpha = 0.05$.

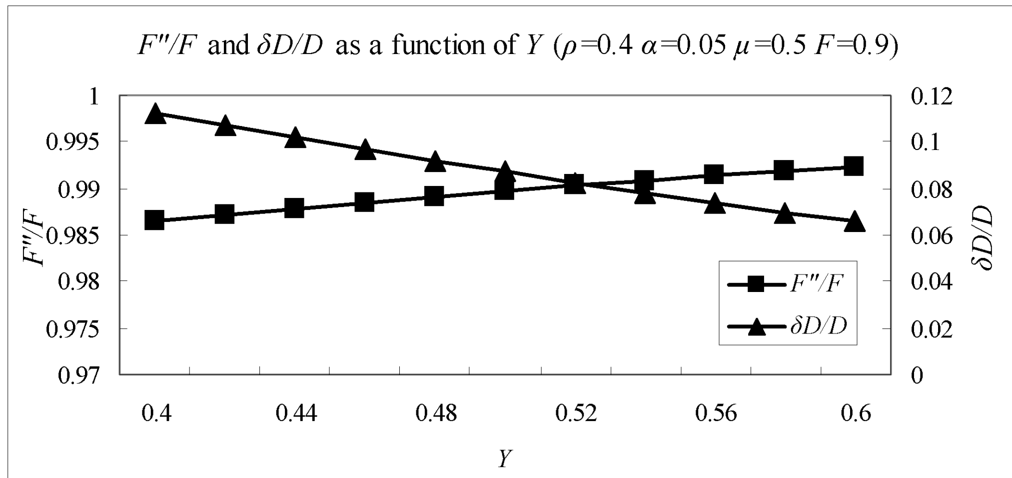

Figure 4.1. F''/F and $\delta D/D$ as a function of Y (at early life)

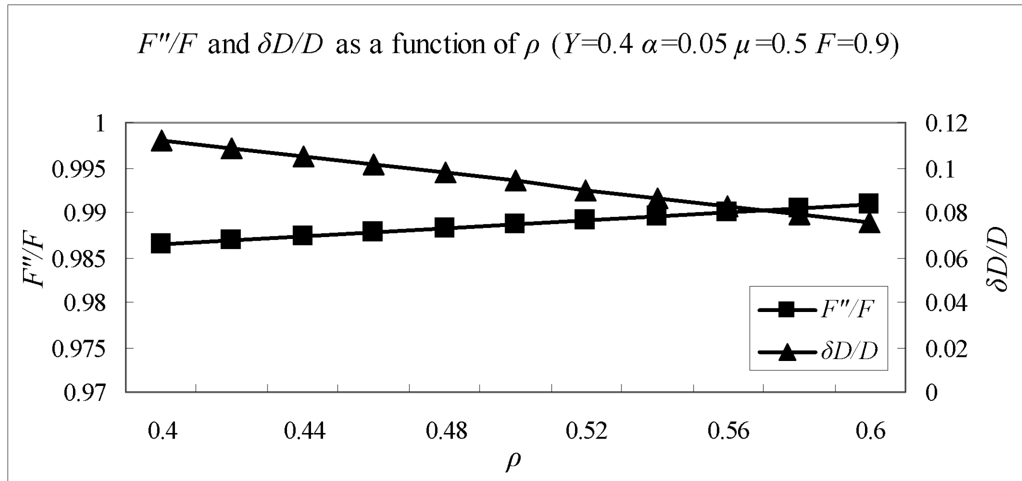

Figure 4.2. F''/F and $\delta D/D$ as a function of ρ (at early life)

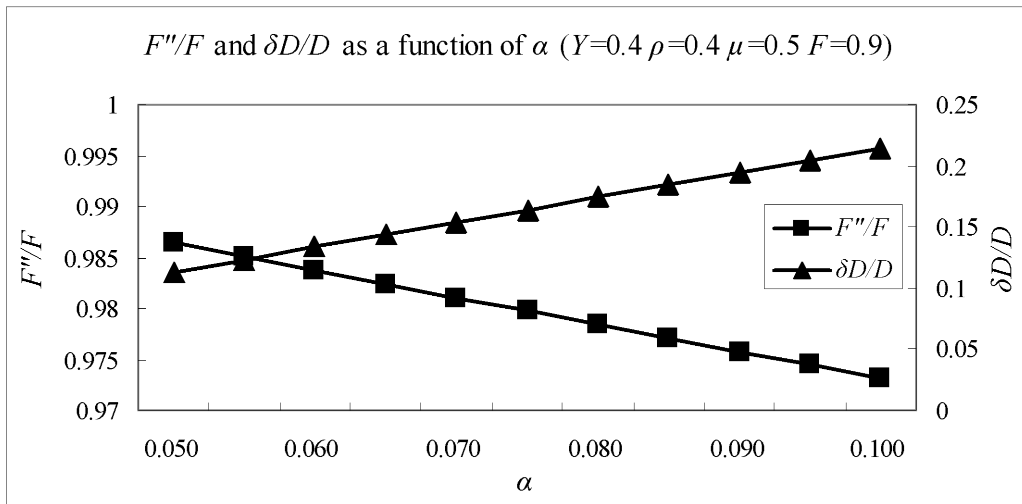

Figure 4.3. F''/F and $\delta D/D$ as a function of α (at early life)

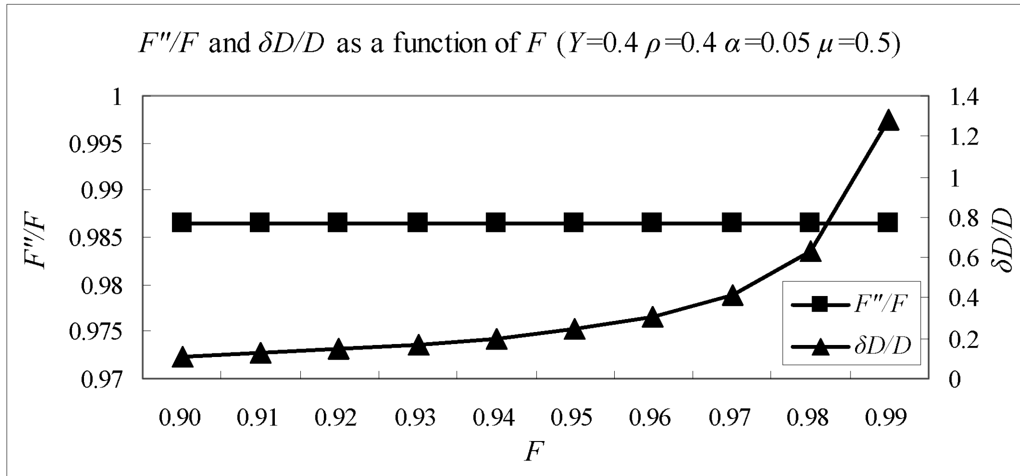

Figure 4.4. F''/F and $\delta D/D$ as a function of F (at early life)

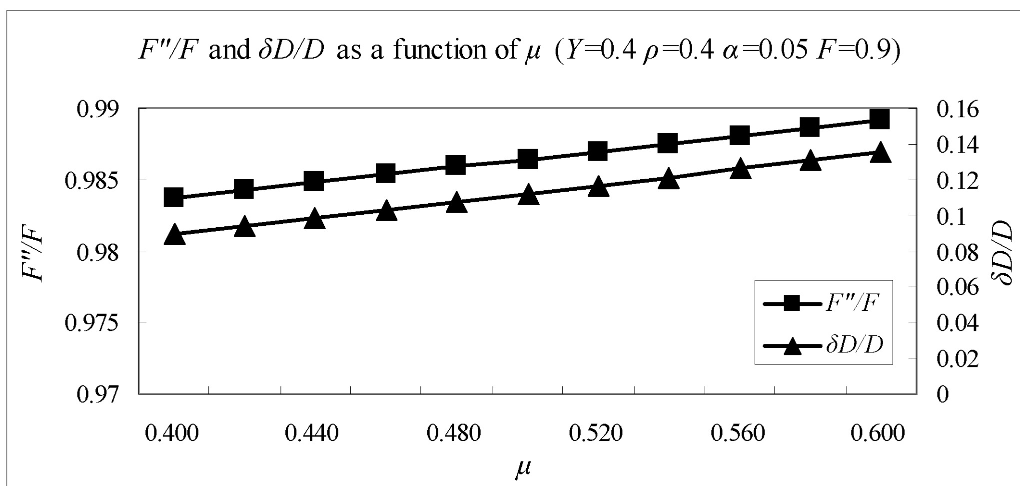

Figure 4.5. F''/F and $\delta D/D$ as a function of μ (at early life)

4.4.2 Impact at Maturity

Since at maturity $Y \approx 1$, we plot F''/F and $\delta D/D$ for the parameter ranges $0.9 \leq Y \leq 0.95$, $0.9 \leq F \leq 0.99$, $0.4 \leq \rho \leq 0.6$, $0.05 \leq \alpha \leq 0.1$ and $0.4 \leq \mu \leq 0.6$. In Fig.4.6 we show the behavior of F''/F and $\delta D/D$ as a function of Y, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.7 we show the behavior of F''/F and $\delta D/D$ as a function of ρ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.9$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.8 we show the behavior of F''/F and $\delta D/D$ as a function of α, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$, $Y = 0.9$ and $\mu = 0.5$. In Fig.4.9 we show the behavior of F''/F and $\delta D/D$ as a function of F, while keeping the other parameters fixed at $Y = 0.9$, $\rho = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.10 we show the behavior of F''/F and $\delta D/D$ as a function of μ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.9$, $\rho = 0.4$ and $\alpha = 0.05$.

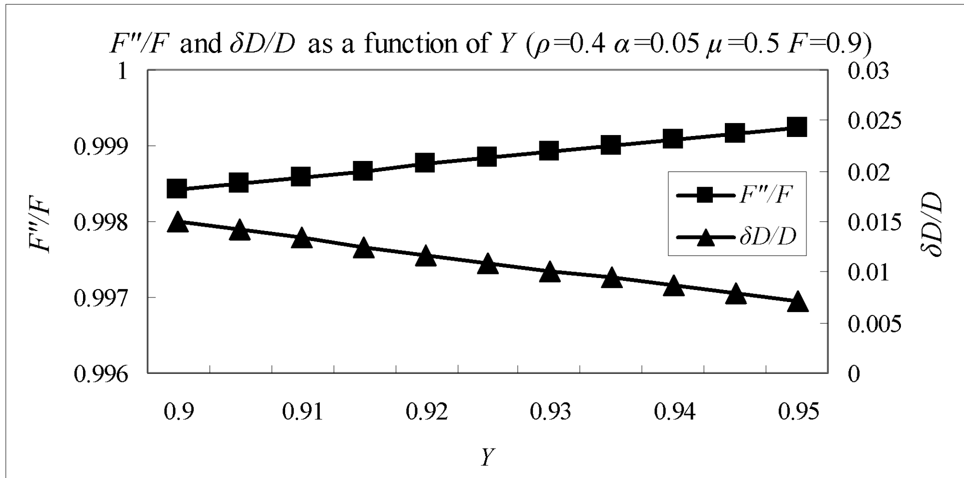

Figure 4.6. F''/F and $\delta D/D$ as a function of Y (at maturity stage)

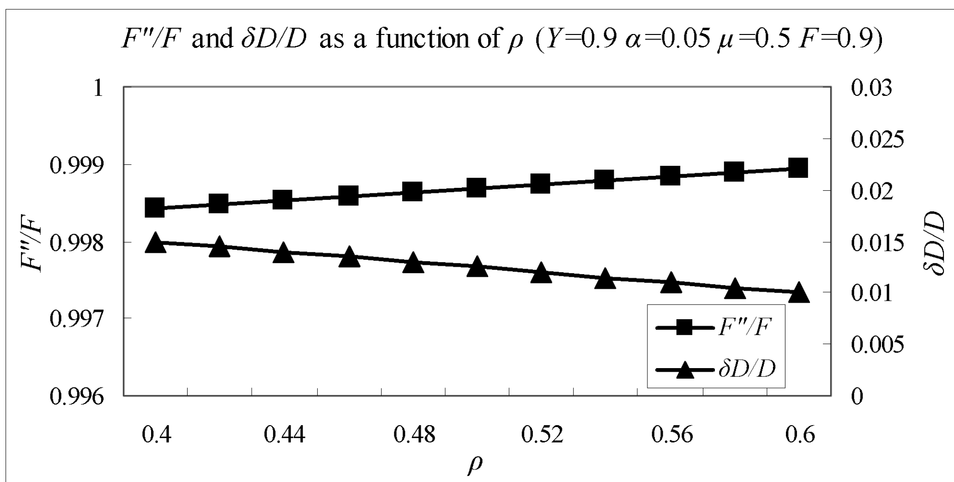

Figure 4.7. F''/F and $\delta D/D$ as a function of ρ (at maturity stage)

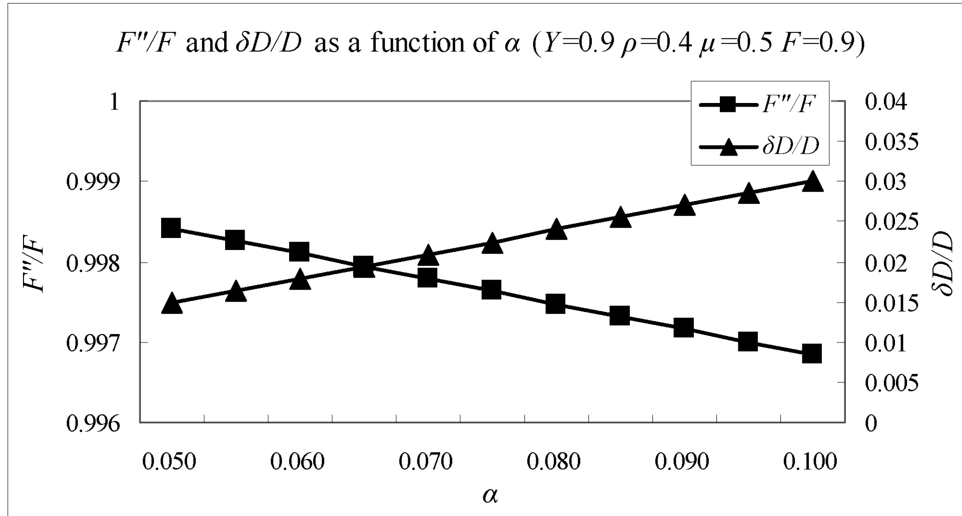

Figure 4.8. F''/F and $\delta D/D$ as a function of α (at maturity stage)

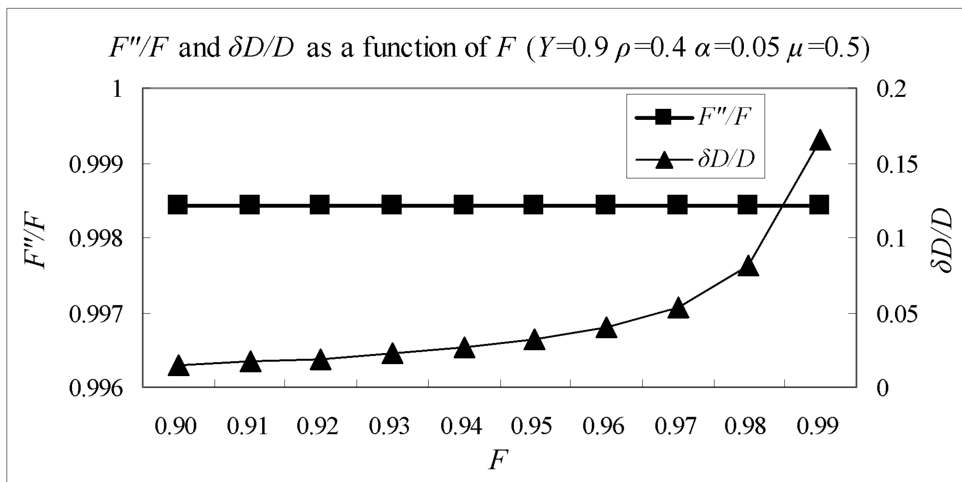

Figure 4.9. F''/F and $\delta D/D$ as a function of F (at maturity stage)

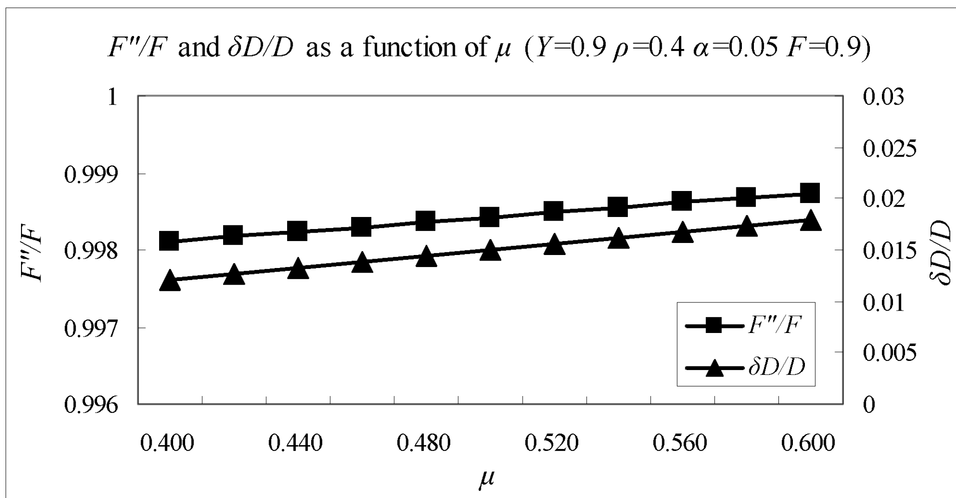

Figure 4.10. F''/F and $\delta D/D$ as a function of μ (at maturity stage)

4.4.3 Comparison of Defect Levels with and without BIST Pretest

In order to see the BIST pretest effect on the product defect level we let $0.4 \leq Y \leq 0.6$ for early life analysis and $0.9 \leq Y \leq 0.95$ for maturity life period. The other parameter ranges are $0.9 \leq F \leq 0.99$,$0.4 \leq \rho \leq 0.6$, $0.05 \leq \alpha \leq 0.1$ and $0.4 \leq \mu \leq 0.95$.

In Fig.4.11 we show the behavior of $\Delta D''/D$, $\Delta D'/D$ and the differential impact factor ,ζ, as a function of Y, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.12 we show the behavior of $\Delta D''/D$, $\Delta D'/D$ and ζ as a function of ρ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig.4.13 we show the behavior of $\Delta D''/D$, $\Delta D'/D$ and ζ as a function of α, while keeping the other parameters fixed at $F = 0.9$, $\rho = 0.4$, $\mu = 0.5$ and $Y = 0.4$. In Fig.4.14 we show the behavior of $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of F, while keeping the other parameters fixed at $Y = 0.4$, $\rho = 0.4$, $\alpha = 0.05$ and $\mu = 0.5$. In Fig. 4.15 we show the behavior of $\Delta D''/D$, $\Delta D'/D$ and ζ as a function of μ, while keeping the other parameters fixed at $F = 0.9$, $Y = 0.4$, $\rho = 0.4$ and $\alpha = 0.05$.

In Figs.4.16-4.20 we show the corresponding behavior at maturity. The difference between these figures and Figs.11-15, is that the range of Y is $0.9 \leq Y \leq 0.95$ in Fig.16, and Y is fixed at 0.9 in Figs.17-20.

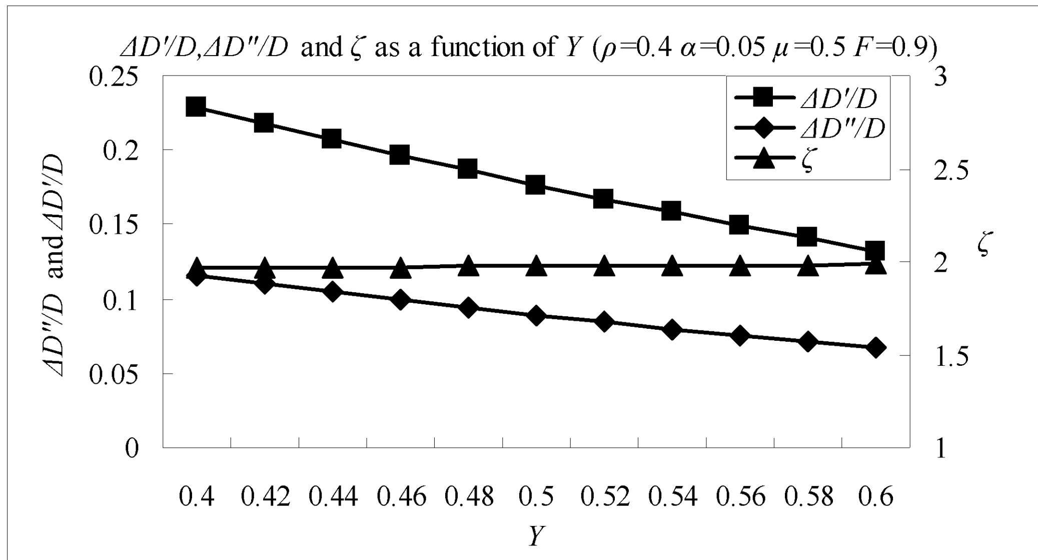

Figure 4.11. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of Y (at early life)

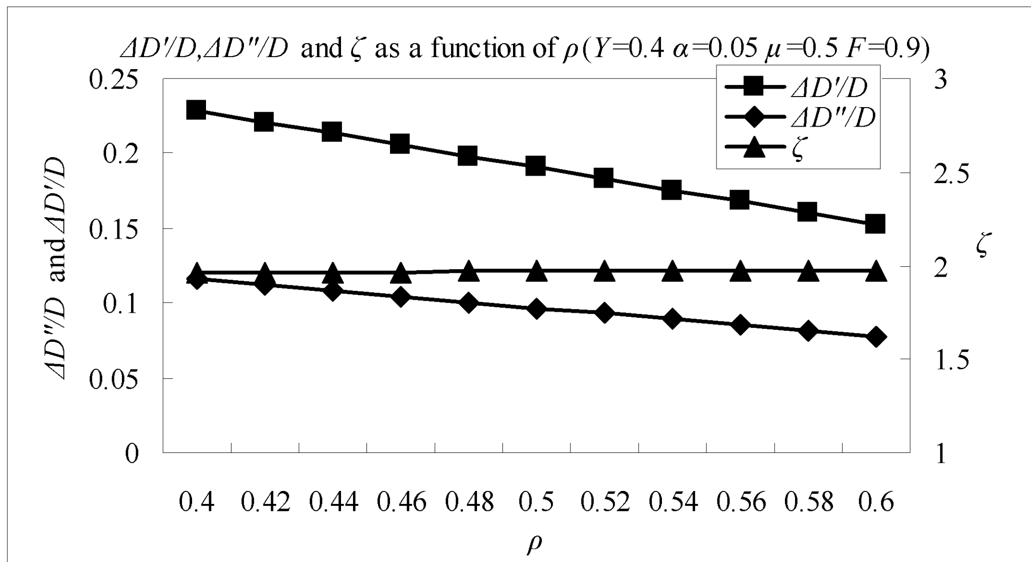

Figure 4.12. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ (at early life)

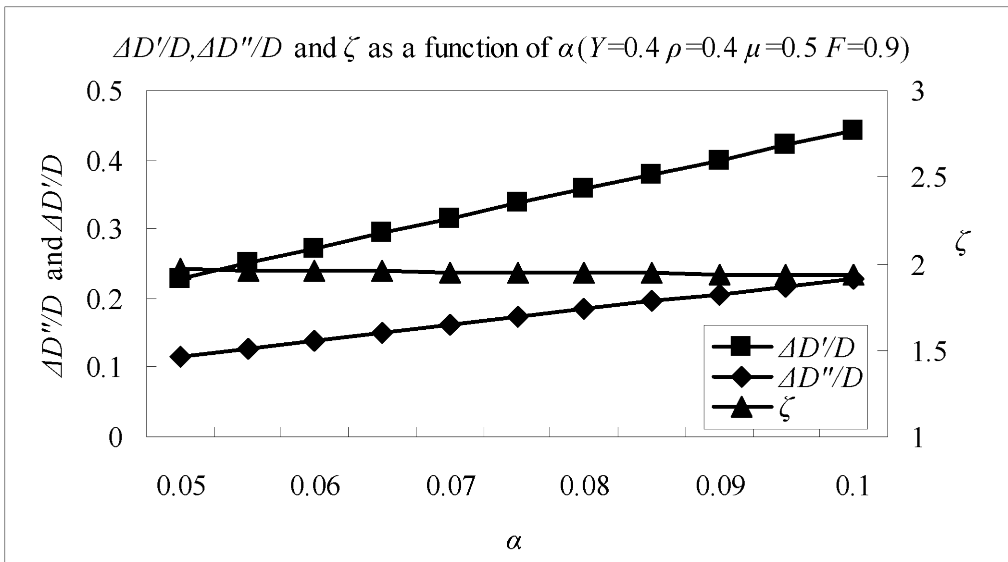

Figure 4.13. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of α (at early life)

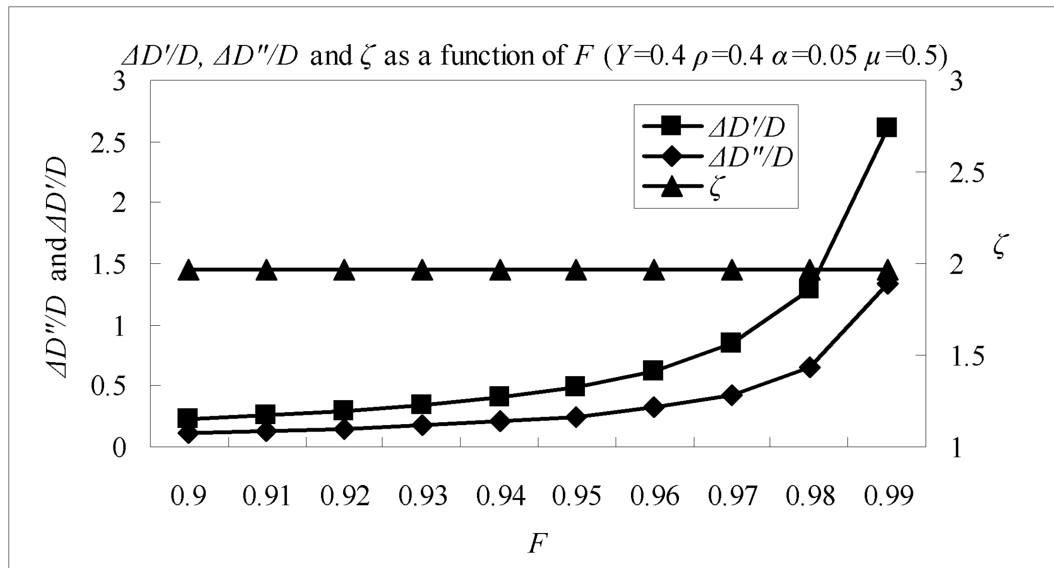

Figure 4.14. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of F (at early life)

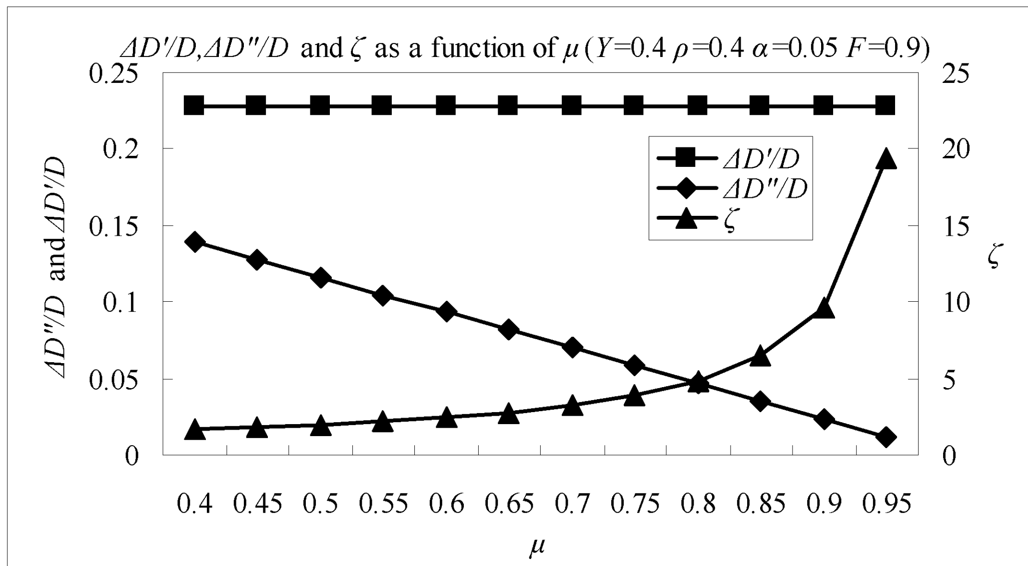

Figure 4.15. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of μ (at early life)

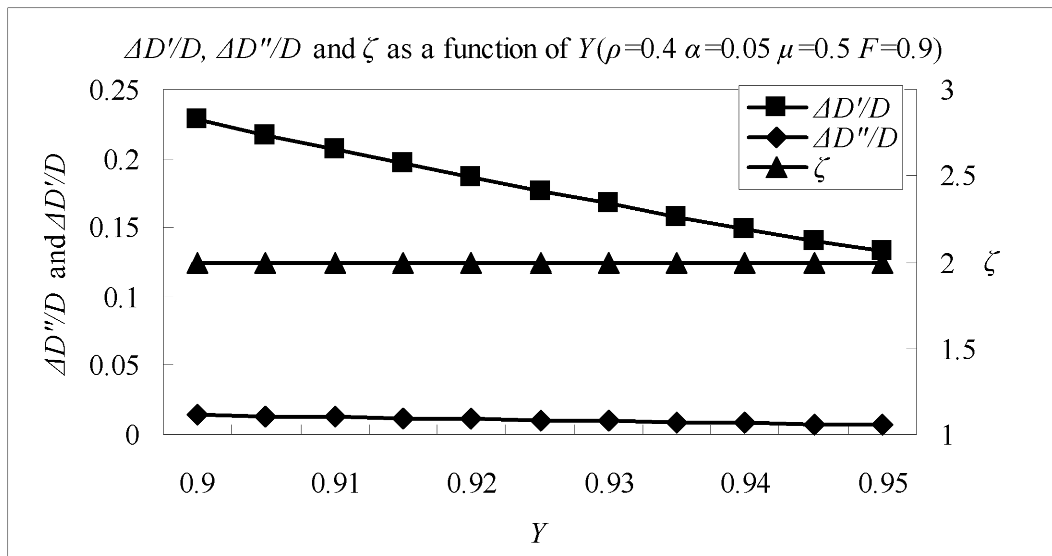

Figure 4.16. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of Y(at maturity stage)

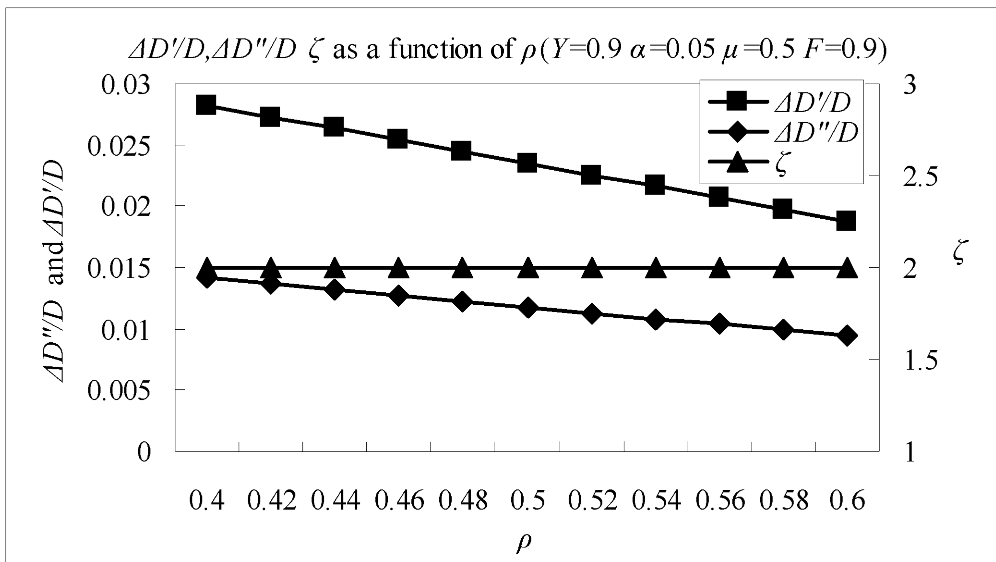

Figure 4.17. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ (at maturity stage)

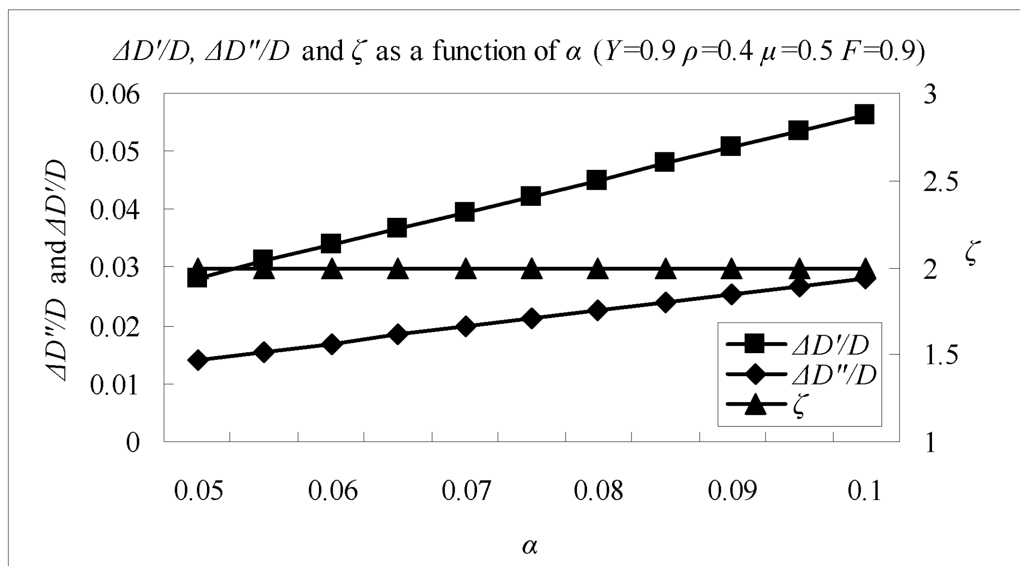

Figure 4.18. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of α (at maturity stage)

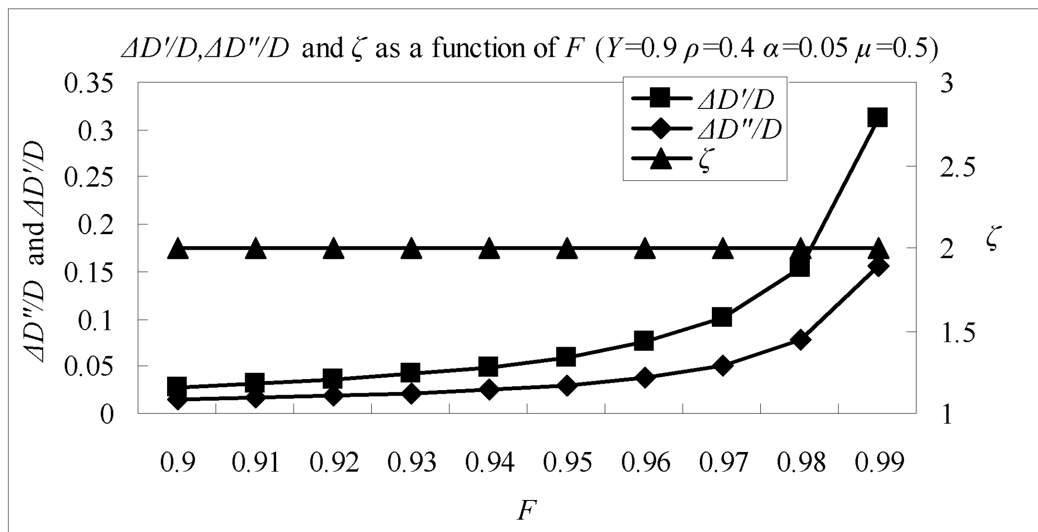

Figure 4.19. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function F (at maturity stage)

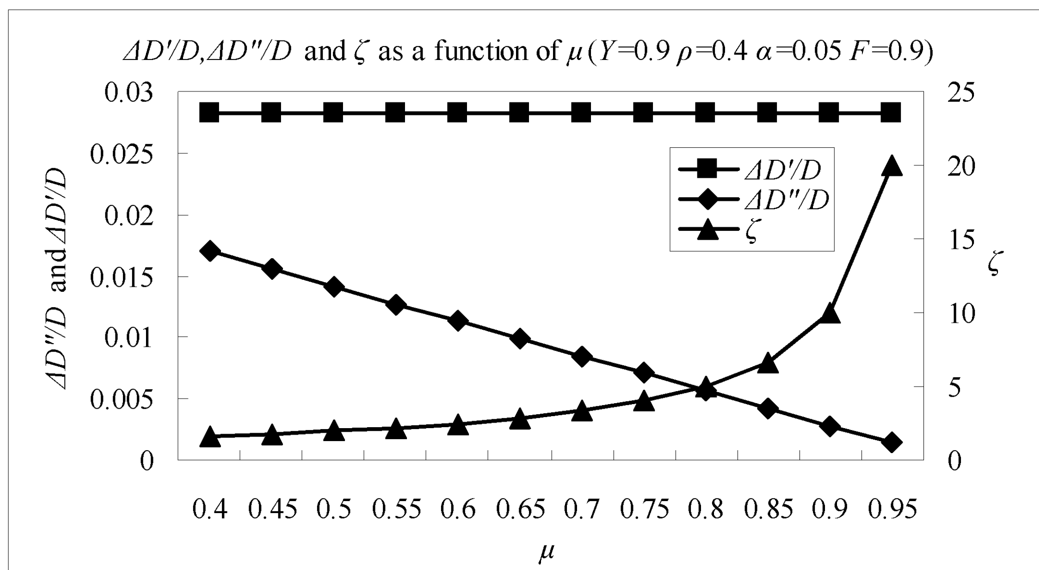

Figure 4.20. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of μ (at maturity stage)

4.4.4 The Case $\rho\prime \neq \rho$

In sections 4.4.1-4.4.3, we assumed that $\rho' \approx \rho$. In this section we investigate the behavior when $\rho' \neq \rho$.

In the following graphs we let $Y = 0.4$ during early life and $Y = 0.9$ during maturity stage. The other parameters are fixed at $F = 0.9$, $\rho = 0.5$, $\alpha = 0.05$ and $\mu = 0.5$. The parameter ρ' is chosen to cover the range $0.4 \leq \rho' \leq 0.6$. In Figs.4.21-4.22 we show the behavior of $\Delta D''/D$, $\Delta D'/D$ and ζ as a function of ρ' during early life and at maturity, respectively.

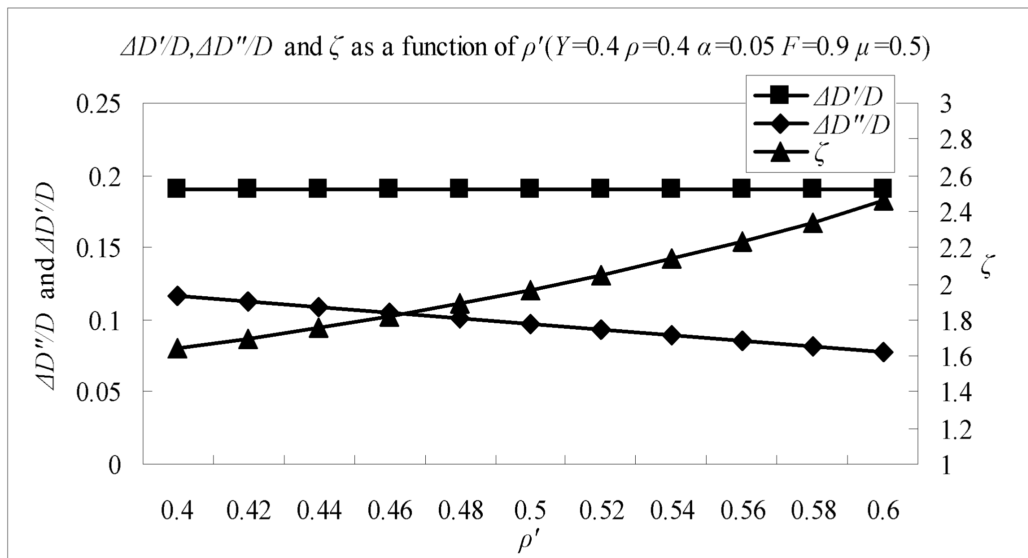

Figure 4.21. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ'(at early life)

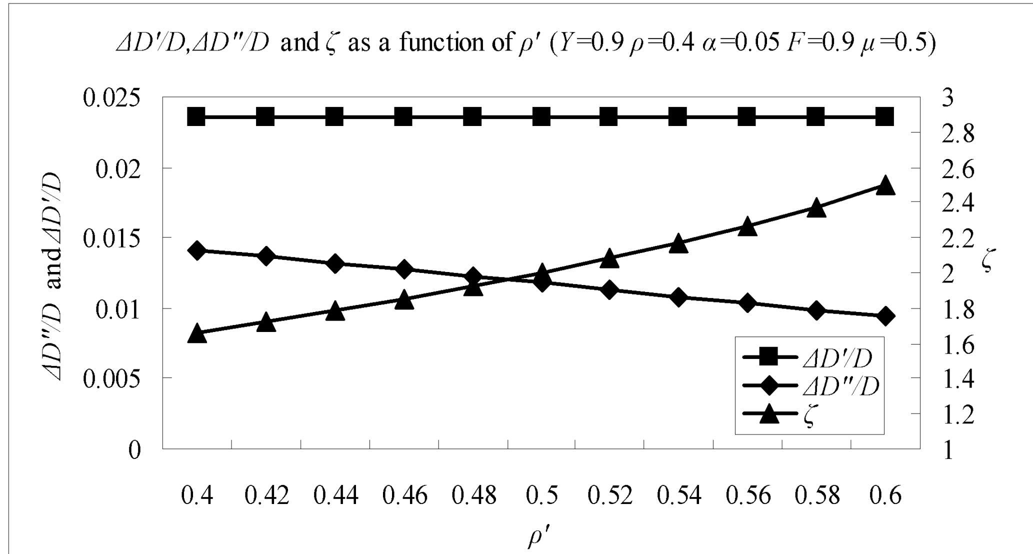

Figure 4.22. $\Delta D''/D$,$\Delta D'/D$ and ζ as a function of ρ' (at maturity stage)

4.4.5 The Impact Trend

As was mentioned earlier, by discarding of chips that fail the pretest we are risking loosing products that would otherwise be functional. This will, undoubtedly, increase the yield loss. Given the fact that the pretest will not rid of all chips with faulty BIST circuitry, some people may argue that this pretest is not worth the risk of loosing yield.

As seen in the previous subsection, unless the CUT fault coverage is in the high 90 percent, the pretest won't buy you much quality improvement during maturity. For CUT fault coverages below 98% the impact of the pretest on the product defect level is quite minor

(around 2%). This quality improvement grows substantially when the CUT fault coverage exceeds 98%, and can be as high as 20-30%.

During early life the BIST pretest has a greater effect on the product defect level. Even for CUT fault coverages around 90%, the BIST pretest can decrease the product defect level by as much as 10%. This quality improvement grows to 80% for fault coverages around 98%.

The differential impact factor highly depends upon the BIST circuitry self-fault-coverage during pretest. We have shown that even when this BIST self-fault-coverage is relatively low, the BIST pretest strategy may reduce the defect level by a factor of 2. For cases where the BIST circuitry fault coverage during pretest are closer to 90%, the differential impact factor is around 10, i.e., the BIST pretest strategy has managed to reduce the overall defect level by a factor of about 10. The differential impact factor also depends upon the CUT fault coverage alteration factors, ρ' and ρ. We assumed that $\rho' \approx \rho$ in the analysis of Sections 4.2 and 4.3. However, Figs.4.21 and 4.22 show that even when $\rho'$and ρ differ by about 10%, the BIST pretest is still efficient.

4.5. Conclusion

In this chapter it is assumed that the BIST circuitry is pretested before launching the CUT functional test. The intent of the BIST pretest is to rid of all chips that fail it, and, therefore, avoid a situation where a faulty BIST has to determine whether or not the functional circuits

operate correctly. By discarding of chips that fail the pretest we are risking loosing chips that would otherwise be functional. This will, undoubtedly, increase the yield loss. Given the fact that the pretest will not rid of all chips with faulty BIST circuitry, some people may argue that this pretest is not worth the risk of loosing yield. This chapter provides some insight as to when this BIST pretest maybe worthwhile.

We show that the BIST pretest has an effect of reducing the product defect level of chips passing the CUT BIST. The question is whether or not the improvement in the shipped-product defect level is worth loosing functional chips as well.

This analysis indicates that for products with CUT fault coverages exceeding 98%, it makes sense to do the BIST pretest. The BIST pretest has the effect of reducing the product defect level by at least 80% during early life, and by as much as 10% during maturity.

During early life, and even for fault coverages below 98%, the BIST pretest offers a non-negligible improvent in product quality. Since this improvent can be as small as 20-30%, and as high as 100%, BIST pretest is worthwhile performing.

Chapter 5

The Problem of Fault Diagnosis in At-speed BIST Environment

5.1. Introduction

Built-in self-test (BIST) has become the major test technique for today's large scale and high speed system-on-chip (SoC) designs. Pseudo-random BIST designs are the most widely used due to their relative simplicity and low cost [3]. Since BIST compacts test responses, BIST requires only small tester memory and it can perform at-speed test even if the test frequency is much higher than the tester frequency limitation. On the other hand, BIST causes problems in diagnosis due to its compacted responses. Indeed, pass/fail information ob-

tained form BIST response analyzer is insufficient for diagnosis. This chapter provides practical formulations of the problem of identifying all error occurrences and all failed scan cells in at-speed scan based BIST environment.

5.2. Error Information for Diagnosis

The most of fault diagnosis algorithms use the error information. Two kinds of information are required to identify a fault in the CUT. These are :

1) the time information (i.e., the input pattern(s) which causes errors) ,

2) the space information (i.e., scan cells where errors occur for scan based BIST architecture [3]).

Using time information, fault diagnosis can be performed for a given fault model by methods using dictionary or fault simulation [35]. Using space information, diagnosis can be performed by cone of logic methods [36]. High resolution diagnostic for a given fault model can be achieved by diagnosis techniques combining space information with time information [37][38].

For scan-based BIST architecture, finding the time when errors occur as the scan chains are unloaded gives both time (failing input pattern) and space (position of erroneous scan cells within the scan chain) information.

5.3. Previous Works

A number of methods to identify space information have been proposed, especially for scan-based BIST architecture [39]-[43], however, only a few practical techniques have been developed to identify time information.

Some of the existing techniques are based on signature analysis using cycling register [44][45] and error correcting codes [46]. These methods compact the complete test response into one signature and can identify certain errors from the signature. Since they observe signature only once, they are suitable even if the circuit frequency is much higher than the tester frequency. However, for large number of error bits, say r errors, they need as many as r-LFSRs or signature registers and may have over 40% diagnostic aliasing if the actual number of errors is higher than r [46]. Thus, they either have poor diagnostic resolution or require impractically high hardware overhead to achieve maximum diagnosis resolution. An alternative approach trades off overhead for time by repeating the test sequence and compacting it at each iteration into a different signature [47]. Thus, instead of using r-LFSRs, the test sequence is repeated r times using programmable LFSR to identify r errors. Since it is mathematically equivalent to [46], diagnostic aliasing is same as using r-LFSRs. Thus, identifying all the errors requires repeating the test sequence an impractically large number of times.

Techniques that use two phases for diagnosis have also been proposed [48]-[50]. During the first phase, intermediate signature is checked

a few times during test in order to narrow down the failing candidates within some windows of fixed or variable size. The failing patterns are then identified inside the windows by applying the corresponding patterns one at a time [48]. These methods use small hardware overhead or test application time. However, they don't propose a mechanism to observe the at-speed behavior inside of failing windows. Enhancement of these methods has also been studied using multiple signature analyzers [50], but they do not achieve maximum diagnosis resolution.

A commonly used diagnosis technique requires and collects the failing space and time information, without compacting responses, during the diagnosis phase [51]. However, this method requires the circuit to operate at the tester frequency during test. Therefore, the faults that affect only at-speed operation may not be diagnosable.

5.4. Problem Formulation

In this section, we formulate the problem of identifying failing response time.

We first identify some characteristics of the diagnosis process and production testing process. Diagnosis can be performed for devices that didn't pass the production test or devices that passed the production test and were found to be faulty in the field. In each case, testing during diagnosis should be performed at the speed that resulted in the failure of the device.

Another characteristic is that the test application time is not the

first priority for diagnosis. Rather, the quality of diagnosis is far more important than the test application time.

The first formulation concerns the complete diagnosis of a scan based BIST circuit. In this formulation, the BIST is constrained to operate at-speed during diagnosis.

In the at-speed BIST environment,

1. the CUT operates at frequency f_c , and
2. the tester has a frequency limitation and can not operate at a frequency higher than $f_{t\max}$, such that $f_{t\max} < f_c$.

are assumed. The problem is to locate the errors that occur when applying the test set to the circuit at frequency f_c.

There are two priorities of objectives.

1. Maximizing resolution in error location (i.e., identify every error occurrence at frequency f_c) is the first priority, while
2. minimizing test application time is the second priority.

There are two constraints:

1. The CUT should be tested exactly at frequency f_c, and
2. the tester frequency for observation can be no more than $f_{t\max}$.

In the second formulation, it is assumed that the actual faults may not depend so much on the frequency at which the fault was originally detected. Thus it may be possible to test the circuit at a frequency lower than $f_{c\max}$, however,

1. the CUT should operate higher than at frequency $f_{c\min}$ to ensure detecting faults, and,
2. the tester can not operate at a frequency higher than $f_{t\max}$, such that $f_{t\max} < f_{c\min}$.

are assumed.

As before, there are two priorities of objectives.

1. Maximizing resolution in error location is the first priority, while
2. minimizing the test application time is the second priority.

Again, there are two constraints:

1. the CUT test frequency (f_c) should range between $f_{c\min}$ and $f_{c\max}$ ($f_{c\min} \leq f_c \leq f_{c\max}$), and
2. the tester frequency for observation must not exceed $f_{t\max}$.

Note that when $f_{c\min} = f_{c\max}$, this problem reduces to the first problem formulation.

Note also that the goal of the above methods is to achieve the maximum resolution under the given constraints without increasing tester memory and with little or no hardware overhead.

Previous works identified in Section 5.3 do not solve these problems. Some of the techniques do not satisfy the conditions imposed on the tester speed, $f_{t\max} < f_c$ [51], while others do not achieve maximum resolution [45][48]-[50]. Also, most of the known techniques require substantial hardware overhead [46] or test application time [47].

5.5. Conclusion

This chapter provides practical formulations of the problem of identifying all error occurrences and all failed scan cells in at-speed scan based BIST environment. In the first formulation, the BIST is constrained to operate at-speed during diagnosis. In the second formulation, it is assumed that the actual faults may not depend so much on the frequency at which the fault was originally detected. Previous works are discussed in this chapter and it concluded that they do not solve the problems.

Chapter 6

Error Identification Method for Maximum Diagnostic Resolution

6.1. Introduction

During debug phase of VLSI devices, the identification of errors in the test response is challenged by the limited bandwidth available for observation by the tester. This chapter investigates the problem how a relatively slow tester can observe the at-speed behavior of fast circuits which is formulated in Chapter 5. A method to solve the problem with no extra hardware is proposed. Every error can be identified even if the circuit test frequency is faster than the tester frequency.

6.2. Observing Responses

As shown in [51], we can identify all failing responses by observing responses of CUT if CUT test frequency is slower than the tester frequency limitation. However, if CUT test frequency is higher than tester frequency limitation, a tester will not be able to observe every response. Figs 6.1-6.3 show an example. When the CUT clock period is 2ns and the tester observing period is 6ns, a tester can observe only 1/3 of the responses. Thus, it will fail to identify failing responses.

If the test sequence is 128 cycles long, as shown in Fig 6.4, every cycle can be observed by repeating the test sequence three times (i.e., applying 384 clocks). During the first sequence, the tester observes response bit 0,3,..,126. Then bits 1,4,7,...,127 are observed during second sequence and bits 2,5,...,128 during the third sequence. However, it may not be able to observe all bits in all cases. For example if the length of the test sequence is 129, or if the tester observes with a period of 8ns, the tester can not observe every response by simply repeating the sequence. In the next section, we derive conditions for observing every response bit and describe a method to identify every error occurrence for the above architecture.

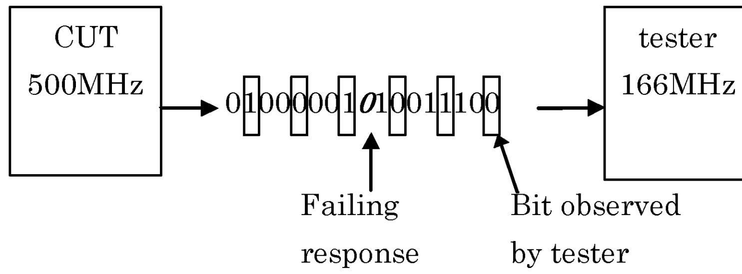

Figure 6.1. Response observation by low speed tester

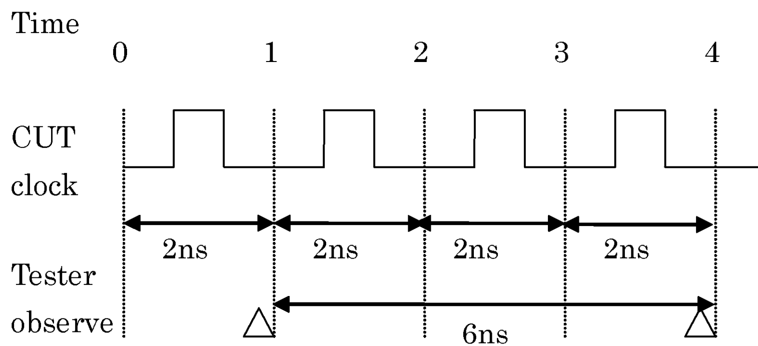

Figure 6.2. CUT and observe intervals

Time	0	1	2	3	4	5	6	7	...	124	125	126	127
Output	OK	OK	ERR	OK	OK	OK	OK	ERR	...	OK	OK	OK	OK
observe No.	0	-	-	1	-	-	2	-	...	-	-	42	-

Figure 6.3. Responses and observed results

Absolute time (relative time)	0	1	2	3	4	5	6	7	...	124	125	126	127
Output	OK	OK	ERR	OK	OK	OK	OK	ERR	...	OK	OK	OK	OK
Observe No.	0	-	-	1	-	-	2	-	...	-	-	42	-

128 (0)	129 (1)	130 (2)	131 (3)	132 (4)	133 (5)	134 (6)	135 (7)	...	252 (124)	253 (125)	254 (126)	255 (127)
OK	OK	ERR	OK	OK	OK	OK	ERR	...	OK	OK	OK	OK
-	43	-	-	44	-	-	45	...	84	-	-	85

256 (0)	257 (1)	258 (2)	259 (3)	260 (4)	261 (5)	262 (6)	263 (7)	...	380 (124)	381 (125)	382 (126)	383 (127)
OK	OK	ERR	OK	OK	OK	OK	ERR	...	OK	OK	OK	OK
-	-	86	-	-	87	-	-	...	-	127	-	-

Figure 6.4. Successful observation

6.3. Problem of Observing Every Response

In this chapter, we use the following terms.

Absolute time - The number of a test clock cycle starting from the beginning of the first test iteration.

Relative time - The number of a test clock cycle starting from the beginning of the current test iteration.

We use the following notation.

N - Length of the test sequence.

P - Period of the tester relative to the CUT test clock period. We assume that P is an integer and that $1 < P < N$.

$R_{\min}$ - Minimum number of test iterations to observe every response.

$M(i)$ - Relative time at $(i+1)$th observation. The range of $M(i)$ is $0 \leq M(i) < N$.

Using the above notation, the method for observing every response can be described as follows.

Observation method : Apply the test sequence of length N $R_{\min}$ times, while observing its response at every time period P.

Our goal is to use the above method to observe all the responses to identify every error occurrence. This maximum resolution of observation is defined as follows.

Definition. The response at relative time t $(0 \leq t \leq N-1)$ can be observed provided that the equation

$$t = M(x) \tag{6.1}$$

has a solution x. The maximum resolution can be achieved when all the responses are observed.

6.4. Conditions to Achieve Maximum Resolution

In this section, we derive the relationship between P and N to achieve maximum resolution of observation.

Lemma 6.1. *The response at the relative time 1 is observable if and only if P and N are co-prime (i.e., $\gcd(N, P) = 1$).*

Proof. $M(i)$ can be expressed as:

$$M(i) = iP \bmod N \tag{6.2}$$

or,

$$M(i) = iP - kN \tag{6.3}$$

if the relative time $M(i)$ is observed during the $(k+1)$th test iteration.

When the response at relative time 1 is observable, the following equation has a solution.

$$M(i) = 1 \tag{6.4}$$

or,

$$iP - kN = 1. \tag{6.5}$$

Eq.(6.5) has a solution (i,k) if and only if P and N are co-prime.

□

Let us assume that the maximum resolution is achieved. Then the response at the relative time 1 is observable. Therefore, Lemma 1 shows that

$$\gcd(N, P) = 1$$

is necessary to achieve maximum resolution. Next, we show that it is also sufficient.

Lemma 6.2. *If* $\gcd(N, P) = 1$ *and the number of test iterations is no more than* P*, then the equation*

$$M(i) = t \tag{6.6}$$

cannot have more than one solution.

Proof. If Eq.(6.6) has two solutions i_1, i_2, then $M(i)$ can be expressed in two ways by Eq.(6.5):

$$M(i) = i_1 P - k_1 N = i_2 P - k_2 N,$$

$$(i_1 - i_2)P = (k_1 - k_2)N. \tag{6.7}$$

Since the number of iterations is smaller than P, we have $0 \leq |k_1 - k_2| < P$. Furthermore P divides $k_1 - k_2$ since P divides $(k_1 - k_2)N$ (from Eq.(6.7)) and $\gcd(P, N) = 1$. Therefore, $k_1 = k_2$ and $i_1 = i_2$ is deduced. Thus, Eq.(6.6) cannot have more than one solution. □

Theorem 6.1. *The maximum resolution is achieved if and only if* $\gcd(N, P) = 1$ *and the number of test iterations is* P*.*

Proof. We assume $\gcd(N, P) = 1$. The number of observations in P test iterations is PN/P,i.e., N. Since $M(i)$ for every $0 \leq i < N$ are different by lemma 2, the set

$$\{M(i) : 0 \leq i < N\}$$

has to be

$$\{0, 1, 2, \ldots, N - 1\},$$

i.e., the observing resolution is maximum.

Therefore, $\gcd(N, P) = 1$ is a necessary and sufficient condition to achieve the maximum resolution in P iterations. □

Example 6.1. Let the length of a test sequence be 2^{32} clocks, the CUT test frequency be 500MHz, and the tester frequency be 100MHz. In this case the tester observing period is

$$P = \frac{500}{100} = 5,$$

which is co-prime with 2^{32}, therefore the maximum resolution of observation is achieved.

Example 6.2. Let the length of a test sequence be 2^{10} clocks, the CUT test frequency be 600MHz, and the tester frequency be 100MHz. In this case the tester observing period is

$$P = \frac{600}{100} = 6,$$

which is not co-prime with 2^{10}, therefore the maximum resolution of observation is not achieved.

6.5. Adjusting N or P to Achieve Maximum Resolution

In Section 6.4, we showed that the maximum resolution of observation is always achieved if N and P are co-prime. However, in general, N and P may not be co-prime. In such cases, the maximum resolution

of observation can be achieved by adjusting N or P. For the problem formulation described in Chapter 5, the following two possibilities exist:

- Increasing the length of test sequence N by inserting additional tests or dummy clock cycles.
- Slowing down the tester by increasing tester observation period P.

The adjustment of N and/or P is chosen to minimize the test application time. Let $N' = N + i$ be the adjusted length of the test sequence and $P' = P + j$ be the adjusted tester observing period. The test application time is:

$$TAT = \frac{N'P'}{f_c} = \frac{1}{f_c}(NP + iP + jN + ij). \tag{6.8}$$

The problem is to find a pair (i,j) that minimizes $iP + jN + ij$, with $N + i$ and $P + j$ co-prime.

Theorem 6.2. *If $N \geq P(P-1)$, the solution (i,j) that minimizes Eq.(6.8) with $N+i$ and $P+j$ co-prime is such that $j = 0$.*

Proof. Since $\gcd(\alpha P + 1, P) = 1$ for any integer $\alpha \geq 0$, there exists a co-prime of P in any consecutive P integers. Therefore, the range of i in Eq.(6.8) is $0 \leq i < P$. Similarly, the range of j is $0 \leq j < N$.

First we consider the case where $j = 0$. The worst case of minimum $iP + jN + ij$ is the case where $i = P - 1$, therefore:

$$iP + jN + ij = (P - 1)P. \tag{6.9}$$

Next, we consider the case when $j \neq 0$. The best case of minimum $iP + jN + ij$ is the case where $i = 0$, therefore:

$$iP + jN + ij = jN \geq N. \tag{6.10}$$

If $N > P(P - 1)$, Eq.(6.10) is always larger than Eq.(6.9). Therefore, $j = 0$ is the solution that minimizes Eq.(6.8). □

A typical tester can operate at about 50MHz and the CUT test frequency in modern DSM circuits in increasing to as high as 5GHz. Thus, we can assume $P < 100$. On the other hand, typical N can be of the order of several million, making $N > P(P - 1)$. Therefore, in most practical cases it is sufficient to adjust only N by inserting dummy clocks since in such cases $j = 0$ provides the optimal solution.

6.6. Procedure for Identifying Every Error Occurrence

6.6.1 Procedure for the First Problem Formulation

Summarizing Section 6.2-6.5, the procedure for identifying every error occurrence at test frequency f_c is as follows. This procedure solves

the first formulated problem in Chapter 5.

Given condition

Test frequency of CUT : f_c

Tester frequency limitation: $f_{t\max}$ ($f_{t\max} < f_c$)

Initial test length: N

Step 1. Set observing time period P as

$$P = \left\lfloor \frac{f_c}{f_{t\max}} \right\rfloor .$$

Adjust test length by adding minimum α dummy clocks (i.e., $N' = N + \alpha$) such that N' is co-prime with P.

Step 2. Apply $P \cdot N'$ clocks to CUT, observing responses every P test cycle.

Step 3. If an error is detected at the $(i + 1)$th observation, then relative time of error occurrence e is:

$$e = iP \bmod N.$$

This procedure solves the first formulated problem in Chapter 5 since it works even when $f_{t\max} < f_c$ and always achieve maximum resolution (i.e., identify every error occurrence at frequency f_c) by minimum test iterations (i.e., test application time) with no extra hardware under the constraints that the CUT should be tested at frequency f_c and the tester frequency for observation is no more than $f_{t\max}$.

6.6.2 Extending the Procedure for the Second Problem Formulation

In Chapter 5, we defined a second problem formulation. It relaxes the constraints of CUT clock frequency f_c to be allowed in the range $f_{c\,\min} \le f_c \le f_{c\,\max}$, whereas the first formulation only allows $f_{c\,\min} = f_c = f_{c\,\max}$. Note that the test application time is constant, regardless of the CUT clock frequency f_c because it is dictated by the tester frequency as shown below.

To minimize test application time, CUT clock frequency f_c is selected from the range

$$f_{c\,\min} \le f_c \le f_{c\,\max}$$

with

$$P = \left\lfloor \frac{f_c}{f_{t\,\max}} \right\rfloor$$

and N co-prime. If there is no such P co-prime with N, we select f_c such that P is co-prime with $N+1$, and so on. The procedure for identifying every error occurrence for second formulation is as follows.

<u>Given condition</u>

Maximum test frequency of CUT : $f_{c\,\max}$

Minimum test frequency of CUT: $f_{c\,\min}$ $(f_{c\,\min} \le f_{c\,\max})$

Tester frequency limitation: $f_{t\,\max}$ $(f_{t\,\max} < f_{c\,\min})$

Initial test length: N

Step 1. Set the maximum observing time period $P_{\max}$ and the minimum observing time $P_{\min}$ as

$$P_{\max} = \left\lfloor \frac{f_{c\max}}{f_{t\max}} \right\rfloor,$$

$$P_{\min} = \left\lfloor \frac{f_{c\min}}{f_{t\max}} \right\rfloor.$$

Select P from the range $P_{\min} \leq P \leq P_{\max}$, and $N' = N + \alpha$ where α is minimum with P and N' co-prime .

Step 2. Apply $P \cdot N'$ clocks to CUT, observing responses every P test cycles.

Step 3. If an error is detected at the $(i+1)$th observation, then relative time of error occurrence e is:

$$e = iP \bmod N.$$

Notice that the second problem formulation defined in Chapter 5 is an extension of the first problem formulation, and the above procedure is also an extension of the procedure proposed in Section 6.6.1. When

$$f_{c\min} = f_{c\max},$$

then

$$P = P_{\min} = P_{\max},$$

thus this procedure reduces to the procedure proposed in Section 6.6.1. Therefore, the procedure proposed in this section solves the problem both for the fist and the second formulated problem in Chapter 5.

6.7. Conclusion

In this chapter, a method of identifying all erroneous responses of CUT is shown. This approach is efficient even if the CUT test clock frequency is much higher than the tester frequency. Tester can observe every response in the limited number of test iterations determined by the ratio of CUT clock frequency and the tester frequency. Therefore, the proposed approach solves the problem formulated in Chapter 5 without extra hardware.

Chapter 7

Diagnosable At-speed BIST

7.1. Introduction

This chapter presents the approach which can be used to identify all error occurrences and all failed scan cells in at-speed scan based BIST environment. In a first place presents a practical implementation of the procedure proposed in Chapter 6 for at-speed scan based BIST environment. Then, we modify the scheme to reduce test application time by using signature analysis.

7.2. BIST Architecture for Maximum Diagnostic Resolution

Fig 7.1 shows a BIST architecture with single output and the logic required for the procedure proposed in Chapter 6. The BIST architecture of Fig 7.1 is based on the scan-based BIST which is one of the most commonly used architectures. The BIST pattern generator (PG) provides scan inputs, and the signature analyzers (SAs) compact its responses. MISRs are used as signature analyzers during testing, and during diagnosis, a masking circuit allows only one scan chain to feed a SA which is selected by input "mask select". As shown in Fig 7.1, scan outs are connected to an output port via a multiplexer during diagnosis as in the non-compaction based approach [51]. A Register, FF, is inserted at the multiplexer output to synchronize the scan chain with the tester since the CUT test frequency may be higher than the tester frequency. The FF samples the signal produced by the scan chain and holds the value during one tester period.

Erroneous scan chains can be identified using masking circuit. In Fig 7.1 for example, by masking scan chains 2 and 4, the two SAs compact only scan chains 1 and 3. Similarly, by masking scan chains 1 and 3, the two SAs compact only scan chains 2 and 4 respectively. Erroneous scan chains can be identified by applying

$$k = \left\lceil \frac{No.\,of\,scan\,chains}{No.\,of\,SAs} \right\rceil$$

iterations of BIST patterns. After erroneous scan chains are identified,

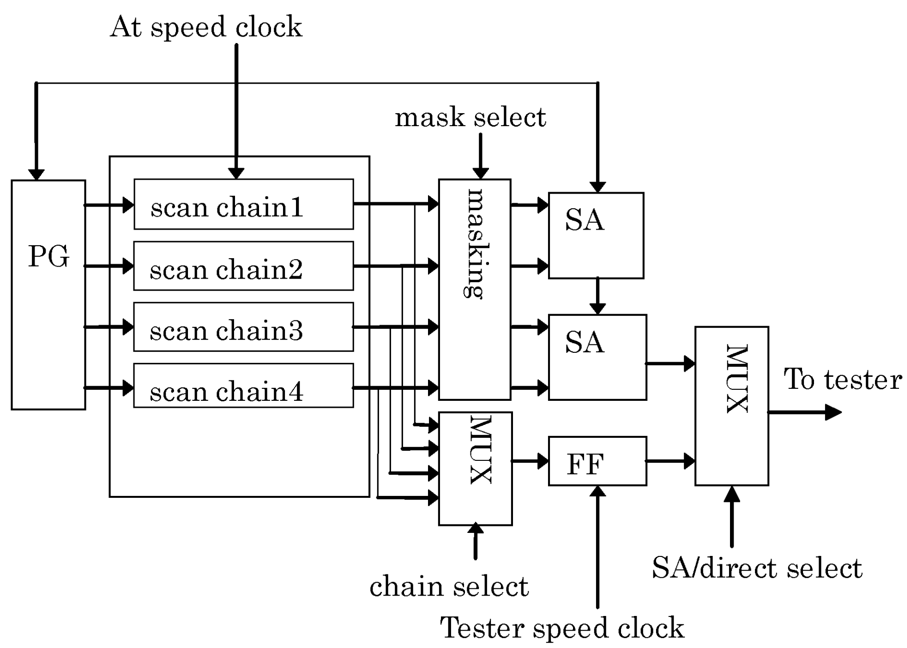

Figure 7.1. Diagnosable BIST

failing responses are identified by observing each erroneous scan chain one at a time using the procedure proposed in Chapter 6.

Any error can be completely identified without aliasing and with negligible hardware for diagnosis by using the method proposed in this section. However, this method always requires a number of iterations of BIST sequences that equals the ratio of circuit frequency over tester frequency as shown in Chapter 6. In the next section, this scheme is modified to reduce the test application time.

7.3. Using Signature Analyzer to Reduce the Test Application Time

The approach introduced in Chapter 6 solves the problems defined in Chapter 5. It has been shown that we have to repeat the BIST sequence at least P times to identify every error occurrence where P is the ratio between the CUT test frequency and the tester frequency.

The approach introduced in Chapter 6 does not use any existing signature analyzers in BIST environment to identify failing responses. There is a way to reduce the number of BIST iterations if we re-use signature analyzers as error detectors albeit at the expense of possibility of aliasing during diagnosis.

Fig 7.2 shows a diagnosable BIST structure with the error detectors. While the tester observes the response of the first iteration of the BIST sequence, signature analyzers compact the responses which are to be observed by the tester in the second and the third BIST

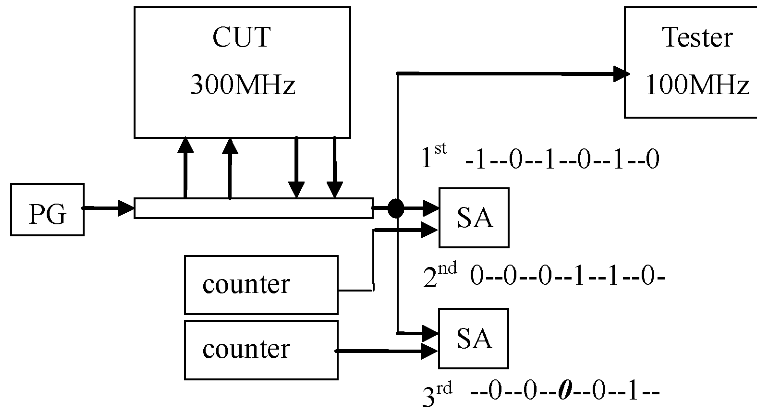

Figure 7.2. Diagnosis with error detector SAs

iterations. A counter is used to select responses for the signature analyzers.

If a signature is not erroneous, we can skip the corresponding iteration. For example, if the first signature analyzer detects no error and the second signature analyzer detects an error, tester skips the second iteration and observes the third iteration. Note that skipping iterations can be done without extra hardware by adjusting N as explained in Chapter 6. Also, during the third iteration the signature analyzers compact the responses which are to be observed by the tester in the 4th and 5th iterations, and so on. The hardware overhead of the scheme consists of signature analyzers and counters. It is obvious that if we use more signature analyzers as error detectors, fewer iter-

ations may be required, albeit at the expense of increasing the hardware overhead. Note also that extra signature analyzers slow down each iteration since each signature needs to be scanned out serially for comparison on tester. Nevertheless, such slow down is marginal since signatures are typically less than hundred bit long while the BIST signature consists of million of cycles. The optimal number of signature analyzers depends on the probability of error occurrence which is shown later.

7.4. Procedure for Identifying Errors Using the Diagnosable BIST

The procedure for identifying every error occurrence using the method proposed in Chapter 6 with the enhanced diagnosis scheme proposed in Section 7.2-7.3 is as follows. This procedure solves the second formulated problem defined in Chapter 5 as well as first formulated problem also when $f_{c\,\min} = f_{c\,\max}$.

Given condition

Maximum test frequency of CUT : $f_{c\,\max}$

Minimum test frequency of CUT: $f_{c\,\min}$ $(f_{c\,\min} \leq f_{c\,\max})$

Tester frequency limitation: $f_{t\,\max}$ $(f_{t\,\max} < f_{c\,\min})$

Initial test length: N

Scan cells in a chain: L

Number of the scan chains: n_{chain}

Number of the signature analyzers: n_{SA}

Step 1. Identify erroneous scan chain

Configure the BIST as shown in Fig 7.1 and do following steps.

Step 1.1. Select untested scan chain for each SAs.

Step 1.2. Apply BIST sequence and identify erroneous SA, i.e, erroneous scan chain.

Step 1.3. Repeat Step 1.1-1.2 until all scan chains are tested. The number of iterations is

$$\left\lceil \frac{n_{chain}}{n_{SA}} \right\rceil .$$

Step 2. Applying BIST to identify erroneous scan cell and pattern

Connect the signature analyzer as shown in Fig 7.2 and do following steps.

Step 2.1. Set the maximum observing time period $P_{\max}$ and the minimum observing time $P_{\min}$ as

$$P_{\max} = \left\lfloor \frac{f_{c\min}}{f_{t\max}} \right\rfloor ,$$

$$P_{\min} = \left\lfloor \frac{f_{c\max}}{f_{t\max}} \right\rfloor .$$

Select P from the range

$$P_{\min} \leq P \leq P_{\max}$$

and,

$$N' = N + \alpha$$

where α is minimum with P and N' co-prime . Reset the BIST iteration counter as $r = 0$ and tester observation counter $i = 0$.

Step 2.2. Apply N' clocks to CUT, observing a scan output every P test cycles by the tester, while the n_{SA} signature analyzers compacts the scan output which will be observed by tester in:

$$r + 1, r + 2, \ldots, r + n_{SA}$$

BIST iteration, respectively. The counter i is incremented by each tester observation.

Step 2.3. Observe the each result of the signature analyzers by the tester. Set the next BIST iteration counter r_{next} as:

1. if no signature analyzers detect errors then, set

$$r_{next} = r + n_{SA} + 1,$$

2. else, the each signature analyzer which results erroneous corresponds to a BIST iteration number. Let

$j_{\min}$ is the minimum BIST iteration number which corresponding signature analyzer results erroneous, set

$$r_{next} = j_{\min}.$$

Step 2.4. Select

$$N' = N + \alpha + \beta$$

where β is

$$\beta = (r_{next} - r - 1)(N + \alpha) \mod P.$$

Set $r = r_{next}$ and tester observation counter i to :

$$i = i + \left\lfloor \frac{(r_{next} - r - 1)(N + \alpha)}{P} \right\rfloor.$$

Repeat Step2.2-Step2.4 while $r \leq P$.

Step 3. Identify erroneous scan cell and pattern

If an error is detected at the $(i+1)$th observation by tester, then:

Relative time of error occurrence e is: $e = iP \bmod N'$

Failing scan pattern $= \left\lfloor \frac{e}{L+1} \right\rfloor$

Erroneous scan cell $= e \bmod (L+1)$

Note that the scan chain length is incremented by one to identify the failing scan pattern and erroneous scan cell in order to account for the capture cycle between successive scans.

7.5. Effect of the Error Detector on Reducing Test Application Time

In the procedure proposed in Section 7.4, the most time consuming process is Step2 since the BIST sequence should be repeated at most P times for each erroneous scan chains which is identified in Step1. The test application time of Step2 ,i.e., the number of BIST iterations, can be reduced using signature analyzers; however, it may depend on the number of erroneous responses and the number of signature analyzers. In this section, the effect of reducing test application time by signature analyzers is discussed using the analytical expression.

7.5.1 Analytical Expression

At the procedure Step2.2, each signature analyzer compacts a sequence of length $\lfloor N'/P \rfloor$. Now, making conventional assumptions about the occurrence of errors [52], the probability that a signature analyzer detects no errors is the probability that all $\lfloor N'/P \rfloor$ bit responses are not erroneous.

Therefore,

$$\Pr\{no\,error\} = (1 - \Pr\{1\,bit\,error\})^{\left\lfloor \frac{N'}{P} \right\rfloor}, \tag{7.1}$$

where $\Pr\{1\,bit\,error\}$ is the probability that 1 bit response is erroneous.

A BIST iteration can be skipped only when it has already been checked by a signature analyzer and resulted into no error.

Therefore, if we use n_{SA} signature analyzers as error detectors, the probability that a BIST iteration is skipped is:

$$\Pr\{1\,skip\} = \Pr\{checked\,by\,SA\} \cdot \Pr\{no\,error\}, \tag{7.2}$$

where

$$\Pr\{checked\,by\,SA\} \ = 1 - \Pr\{1\,skip\}^{n_{SA}},$$

since the BIST iteration cannot be checked by a signature analyzer if all n_{SA} iterations preceding it have been skipped.

Therefore,$\Pr\{1\,skip\} \ = \ x$, is obtained by finding a root of the following equation:

$$\Pr\{no\,error\}x^{n_{SA}} + x - \Pr\{no\,error\} = 0. \tag{7.3}$$

The probability of skipping m ($m \leq n_{SA}$) BIST iterations can be expressed by binominal distribution.

$$\Pr\{m\,skip\} = \binom{P}{m}(\Pr\{1\,skip\})^m(1 - \Pr\{1\,skip\})^{P-m}. \tag{7.4}$$

Therefore, the expected number of BIST iterations to be skipped is:

$$\begin{aligned} E(skip) &= \sum_{m=1}^{P} m \cdot \Pr\{m\,skip\} \\ &= P \cdot \Pr\{1\,skip\}. \end{aligned} \tag{7.5}$$

We may have to add some dummy clocks to adjust N for some BIST iterations. However, this impacts the test application time only marginally.

Therefore, $N' \approx N$ and the expected test application time is:

$$TAT \approx \frac{N(P - E(skip))}{f_c} \approx \frac{1}{f_t} N(1 - \Pr\{1skip\}). \quad (7.6)$$

7.5.2 Effect of the Error Detector

In order to see the effect of the error detectors, we plot the reduction ratio $E(skip)/P$ for the following different parameters, $1 \le n_{SA} \le 10$, $10^3 \le N \le 5 \times 10^6$, $10^{-4} \le \Pr\{1bit\,error\} \le 10^{-3}$ and $1 \le P \le 100$. In Fig 7.3 we show the iteration reduction rate as a function of P, while keeping the other parameters fixed at $\Pr\{1bit\,error\} = 10^{-4}$, $n_{SA} = 5$ and $N = 2^{20} - 1$. In Fig 7.4 we show the iteration reduction rate as a function of n_{SA}, while keeping the other parameters fixed at $\Pr\{1bit\,error\} = 10^{-4}$, $P = 64$ and $N = 2^{20} - 1$. In Fig 7.5 we show the iteration reduction rate as a function of N, while keeping the other parameters fixed at $\Pr\{1bit\,error\} = 10^{-4}$, $n_{SA} = 5$ and $P = 64$. In Fig 7.6 we show the iteration reduction rate as a function of $\Pr\{1bit\,error\}$, while keeping the other parameters fixed at $n_{SA} = 5$, $N = 2^{20} - 1$ and $P = 64$.

The number of iterations P increases as the CUT clock frequency becomes higher relative to the tester frequency. Fig 7.3 shows that

the error detector reduces the number of iterations as linearly proportional to P, and the reduction is over 10% for 50 iterations. Fig 7.4 shows that two to three signature analyzers are sufficient as error detectors. The reduction rate is almost constant for more than 3 signature analyzers. Fig 7.5 and Fig 7.6 show that the error detectors are no more effective when the length of the BIST sequence is very long or the error probability of 1 bit error is high. However, these conditions merely occur in practice under scan based BIST architecture. In general, length of the BIST sequence is not very long since typical scan based BIST architectures use shorter length multiple scan chains [3]. Also, the probability of 1 bit error is known to be very low for the scan based BIST design since scan registers split CUT into smaller independent combinational components and an error is propagated to a very limited number of scan registers.

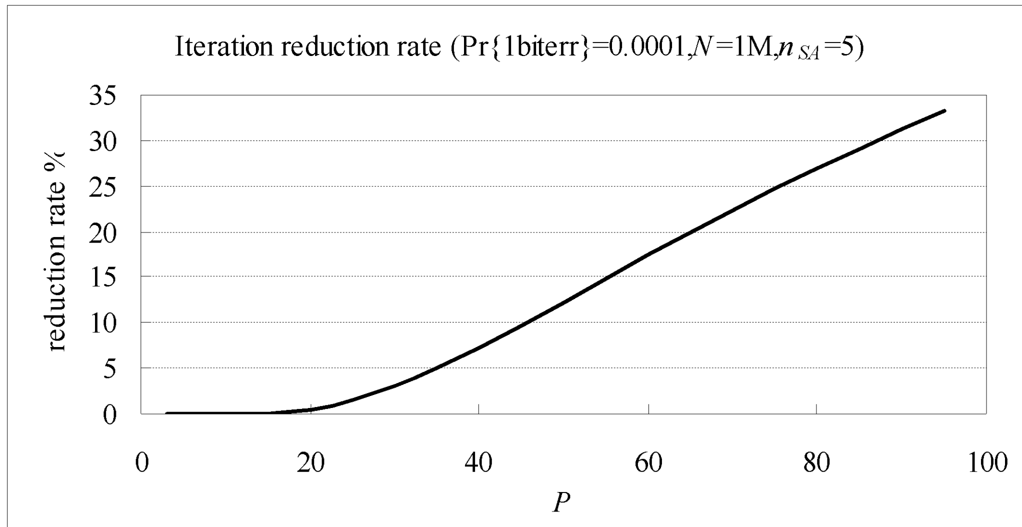

Figure 7.3. Iteration reduction rate as a function of P

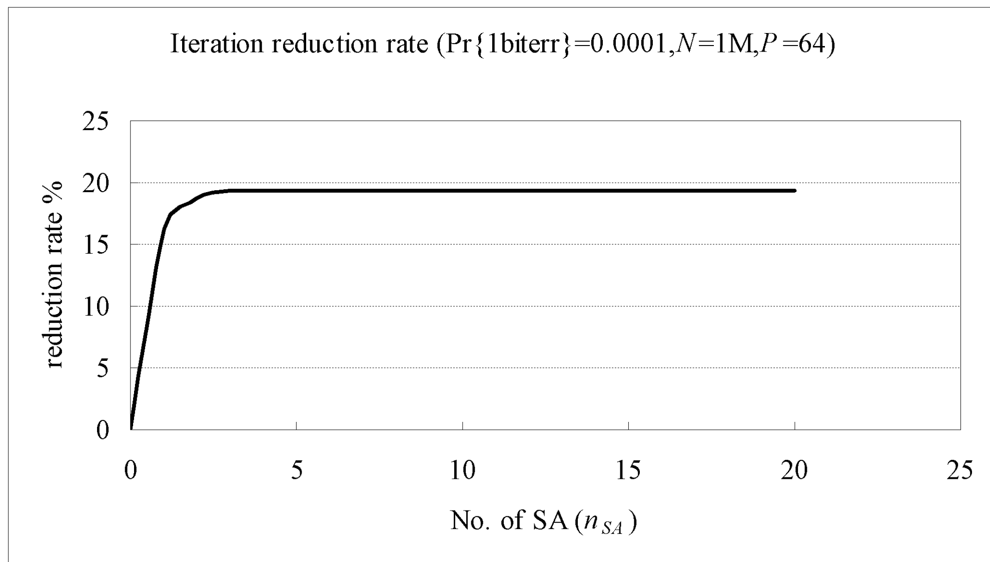

Figure 7.4. Iteration reduction rate as a function of number of SAs

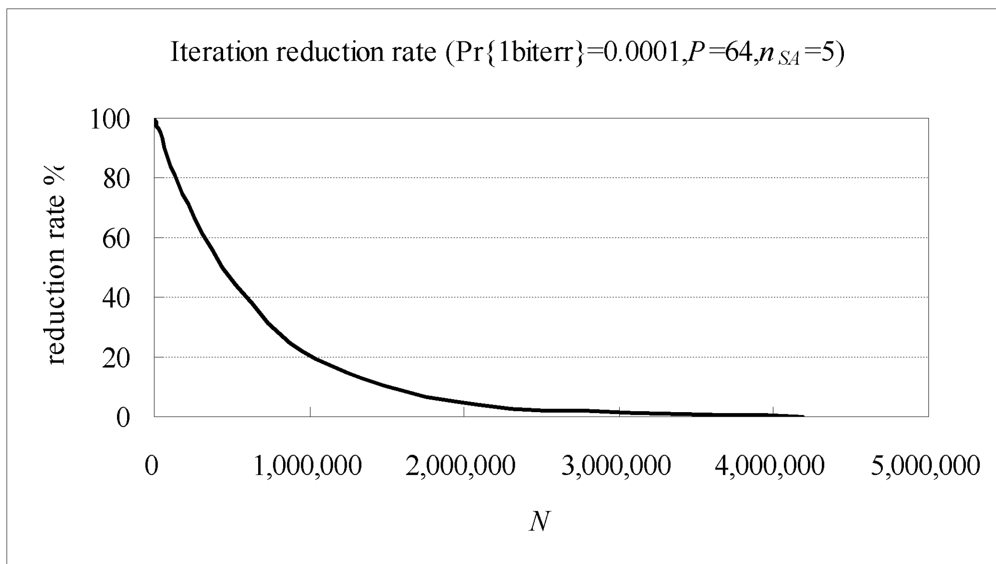

Figure 7.5. Iteration reduction rate as a function of N

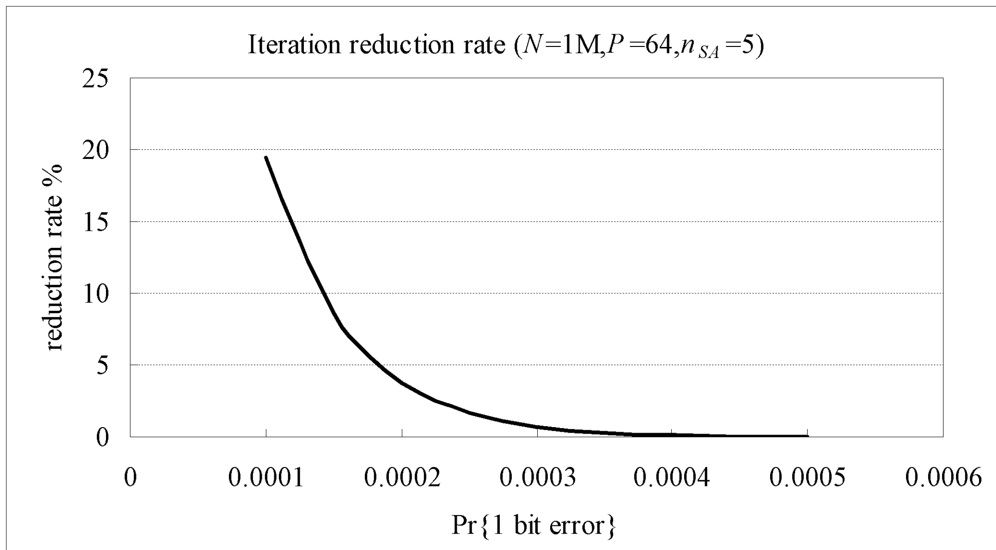

Figure 7.6. Iteration reduction rate as a function of 1 bit error probability

7.6. Experiments

In order to verify the analytical results discussed in Section 7.5 and to show the effectiveness for large industry's circuit, two experiments are conducted.

7.6.1 Verifying the Analytical Results

The CUT for the first experiment is 74181 ALU [53] which is instantiated 23 times and for which all inputs and outputs are connected to scan cells. The total CUT size is 1495 gates, 322 FFs. 16 bit LFSR is used for a PG and SAs. Simulation results are averaged over 10 randomly selected faults. We assume the tester frequency is 40MHz. We plot both theoretical test application time (Eq.(7.6)) and simulated test application time for the following different parameters: $120 \leq f_c \leq 3880$MHz, the number of SAs was varied between 1 and 30, and the length of the BIST sequence that the PG generates was varied between 5K and 2.6M. Fig 7.7 shows the test application time as a function of f_c while keeping $n_{SA} = 3$ and $N = 82$K. Fig 7.8 shows the test application time as a function of n_{SA}, while keeping $f_c = 3880$MHz and $N = 82$K. Finally, Fig 7.9 shows the test application time as a function of N while keeping $f_c = 3880$MHz and $n_{SA} = 3$.

Clearly more iterations are required as the CUT clock frequency becomes higher relative to the tester frequency. Fig 7.7 shows that the use of error detectors reduces the test application time for high clock

frequency. From Fig 7.8 we can conclude that two to three signature analyzers are sufficient as error detectors as no additional reduction in test application time takes place if more SAs are used. Indeed, the test application time is almost constant for more than 3 signature analyzers. Fig 7.9 shows that the test application time is proportional to the length of BIST sequence. Fig 7.7-Fig 7.9 collectively also show that the test application time computed by using Eq.(7.6) is quite close to the simulation results.

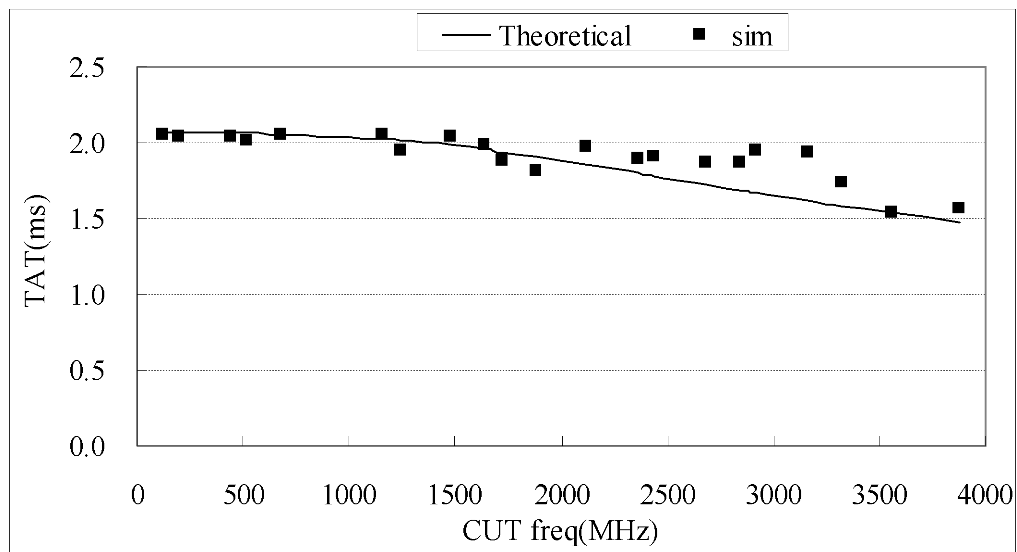

Figure 7.7. TAT as a function of CUT frequency

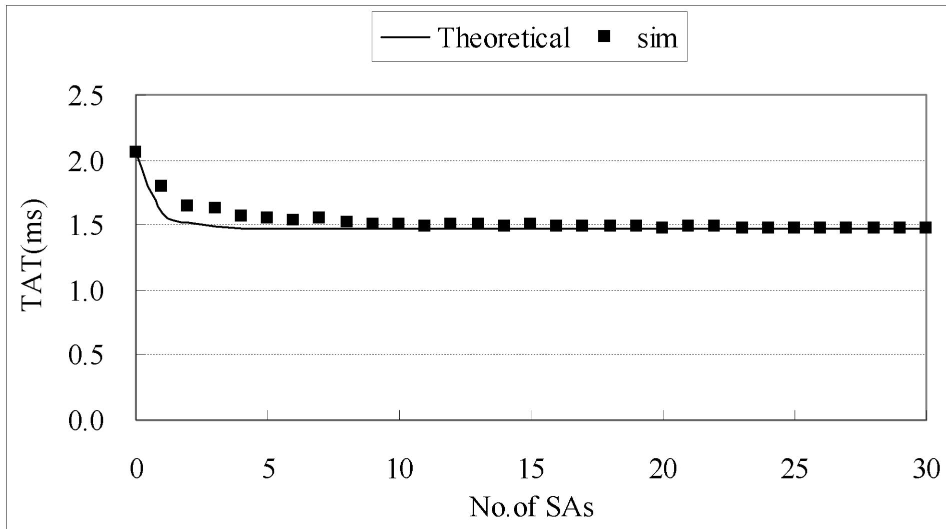

Figure 7.8. TAT as a function of number of SAs

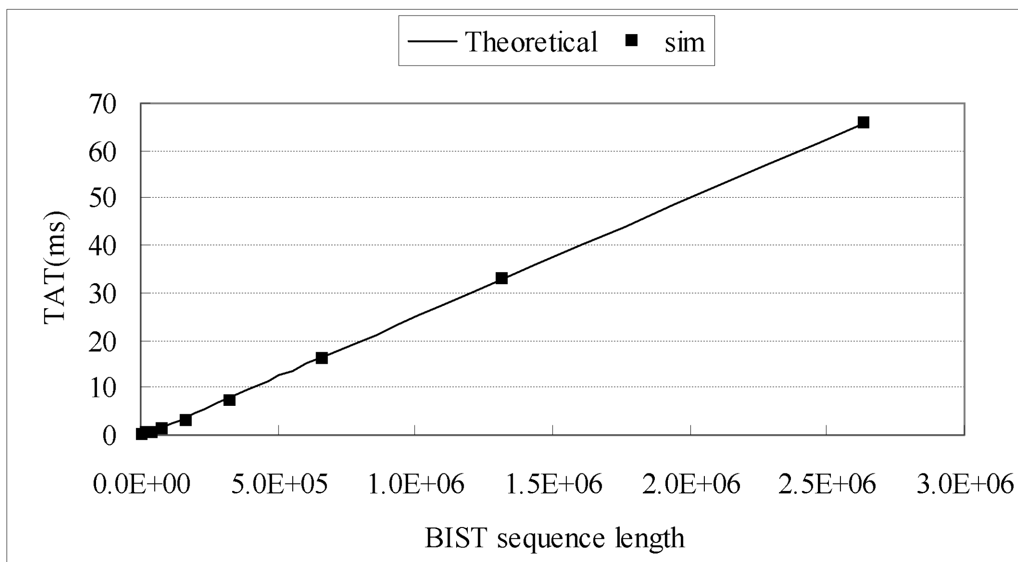

Figure 7.9. TAT as a function of BIST sequence length

7.6.2 Experiments for an Industry's Circuit

Next, we present simulation results for a large industrial circuit. The CUT is a part of a SoC developed at an industry and we added diagnosable BIST to it for the experiments. Details of the circuit are provided in Table 7.1.

We show the results in Table 7.2 for 20 randomly selected faults. Note that the faults 8 and 11 are not detected by the BIST sequence used in the experiment. This table shows that errors are observed by only a small number of chains in most cases and error probability is quite different for each fault case. The simulated and theoretical skip ratio ($E(skip)/P$) and test application time are also shown in Table 7.2. Once again we observe a close match between the real simulation data and the results of this theoretical expression (Eq.(7.5) and Eq.(7.6)). We also notice from Table 7.2 that the test application time for identifying all erroneous scan chains in case 10 is substantially larger than in the other cases. This is because the number of erroneous scan chains for this case is very large. None the less total test application time for case 10 is still within 1 second, which is quite practical for diagnosis. Longer BIST sequence may be needed to detect faults 8 and 11; we expect the test application time to be proportional to the length of the BIST sequence as described in the first experiment.

Table 7.1. An industry's circuit

No. of gates	6M gates
No. of FFs	54505
No. of external ports	317
No. of Scan chains	64
No. of SAs	8 (16bit LFSR)
clock frequency of CUT	1.64GHz
clock frequency of tester	40MHz
Length of Test pattern	1000

Table 7.2. Experimental results of the industry's circuit

fault	Error chain	Pr{1bit error}	sim skip %	Eq.7.5 skip %	sim TAT (ms)	Eq.7.6 TAT (ms)
0	1	5.3E-05	31.7	33.4	18.8	18.4
1	1	1.4E-04	4.9	5.8	24.5	24.3
2	1	6.8E-05	26.8	24.3	19.8	20.4
3	7	1.8E-06	85.7	96.2	25.6	9.8
4	1	7.3E-05	19.5	22.0	21.4	20.8
5	1	7.7E-05	24.4	20.0	20.3	21.3
6	3	1.2E-06	87.0	97.6	12.5	5.7
7	1	1.4E-04	2.4	5.0	25.0	24.5
8	0	0	-	-	-	-
9	6	1.4E-06	86.6	97.2	21.4	7.8
10	51	1.1E-05	76.5	80.1	260.7	220.8
11	0	0	-	-	-	-
12	1	5.8E-04	0.0	0.0	25.6	25.5
13	1	6.8E-05	24.4	24.3	20.3	20.4
14	3	5.8E-06	82.9	88.5	15.1	11.5
15	1	1.2E-04	4.9	8.1	24.5	23.8
16	1	1.9E-05	68.3	67.7	11.0	11.1
17	1	5.8E-04	0.0	0.0	25.6	25.5
18	1	9.8E-05	12.2	12.9	22.9	22.8
19	1	1.4E-04	0.0	5.0	25.6	24.5

7.7. Conclusion

In this chapter presents an implementation of a diagnosable BIST for identifying every failing pattern and all erroneous scan cells. This approach is efficient even if the CUT test clock frequency is much higher than the tester frequency. Tester can observe every response in the limited number of BIST iterations determined by the ratio of CUT clock frequency and the tester frequency.

The proposed architecture uses signature analyzers as error detectors to reduce the number of BIST iterations. The effectiveness of the error detector is discussed both analytically and experimentally. Analytical and experimental results show that the error detectors can reduce the number of BIST iterations by more than 10% when large number of iterations is required and two or three signature analyzers are sufficient as error detectors. Experimental results also show that this approach is quite efficient for the large scale industry's circuit. Therefore, the approach proposed in this chapter achieves the maximum resolution with very low hardware overhead and in practical test application time.

Chapter 8

Conclusions and Future Works

8.1. Summary of the Thesis

Computer and semiconductor manufacturers have changed their design practices to incorporate test as an integral part of their design cycle. Coverage requirements in the high percent against single stuck-at faults and speed related faults are quite common today. In order to reach this goal, built-in self test (BIST) hardware is included today in many chips. However, there have been still existing two problems: one is the reliability of the BIST hardware, and another is the fault diagnosis in the BIST environment.

The thesis presented studies on defect level and diagnosis for BIST architecture. In Chapter 2, the formula, relating the product defect

level as a function of the manufacturing yield and fault coverage, is introduced. However, previous approaches assume that the test process is fault-free, i.e. a circuit being declared by the test process to be faulty is truly faulty. Since the BIST hardware is manufactured using the same technology as the functional circuits, it is possible for it to be faulty. Therefore previous proposed approaches are not suitable for the BIST environment.

In Chapter 3, we extended the Williams and Brown's formula for products with BIST hardware which is assumed to suffer from the same defect density as the functional circuits themselves. The impact of this unreliable BIST is studied in detail. During maturity, and for fault coverage under 98%, the impact of the BIST circuitry unreliability is minor (about a 5% departure). In this case the Williams and Brown's formula constitutes a reasonable approximation even in the presence of an unreliable BIST. For fault coverage above 98%, however, the defect level increment can easily grow by 30-50%. Therefore, the Williams and Brown's formula no longer represents the true situation. During early life, and even for fault coverage below 98%, we see a considerable departure from the Williams and Brown's results. The departures in defect levels, for example, may be as small as 20% and as high as 150%. These departures worsen for fault coverage above 98%. Thus, the Williams and Brown's formula cannot be used for this stage in the product's life. It is paramount to use this enhanced equations instead.

In Chapter 4, we assume that the BIST circuitry is pre-tested

before launching the CUT functional test. The intent of the BIST pretest is to rid of all chips that fail it, and, therefore, avoid a situation where a faulty BIST has to determine whether or not the functional circuits operate correctly. We show that the BIST pretest has an effect of reducing the product defect level of chips passing the CUT BIST. This analysis indicates that for products with CUT fault coverage exceeding 98%; it makes sense to do the BIST pretest. The BIST pretest has the effect of reducing the product defect level by at least 80% during early life, and by as much as 10% during maturity. During early life, and even for fault coverage below 98%, the BIST pretest offers a non-negligible improvement in product quality. Since this improvement can be as small as 20-30%, and as high as 100%, BIST pretest is worthwhile performing.

Chapter 5 presents a problem of the fault diagnosis in the BIST environment. Since BIST compacts test responses, it requires only small tester memory and it can perform at-speed test even if the test frequency is much higher than the tester frequency limitation. On the other hand, BIST causes problems in diagnosis due to its compacted responses. Indeed, pass/fail information obtained form BIST response analyzer is insufficient for diagnosis.

Chapter 6 presents a method for identifying all erroneous responses for maximum diagnostic resolution. This approach is efficient even if the CUT test clock frequency is much higher than the tester frequency. Tester can observe every response in the limited number of test iterations determined by the ratio of CUT clock frequency and the tester

frequency.

Chapter 7 presents the diagnosable at-speed BIST using the method proposed in Chapter 6, and presents also an enhanced architecture using signature analyzers as error detectors to reduce the number of BIST iterations. Experimental results show that the error detectors can reduce the number of BIST iterations by more than 10% when large number of iterations is required. Experimental results also show that two or three signature analyzers are sufficient as error detectors. Therefore, this approach achieves the maximum resolution with very low hardware overhead and in practical test application time.

8.2. Future Works

This thesis presented studies on defect level and diagnosis for BIST architecture. In studies on defect level, we used the Williams and Brown's equation to discuss the model of relationship between the defect level, the product yield and the fault coverage. However, there are some more equations regarding it has been proposed. Our proposed enhancement of the equation should be also discussed for such other proposed equations comparing to real product's defect data in future.

This thesis also presented a perfect error identification methodology for scan based BIST environment. The proposed method can identify all the error information; however, the fault diagnosis may not need all the error information, i.e., the required number of errors to archive the targeted diagnosis resolution can be a limited number.

In this case, BIST sequence can be stopped when the target number of errors is detected. Therefore, test application time will be more reduced. This approach also leaves room for improvement of the BIST pattern generator to divide whole test pattern into some groups since divide-and-conquer approach minimizes more test application time.

We assume the scan based BIST architecture in this thesis. However some non-scan based BIST architectures and high-level BIST architectures have been proposed. They also should be investigated in future.

Acknowledgments

I would like thank to my supervisor, Professor Hideo Fujiwara, for his kindly guidance, a lot of advices and directions during my study at Nara Institute of Science and Technology (NAIST). Things he taught me are not only for graduate studies, but also it is quite valuable for my carrier at the industry.

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

I am thankful to Associate Professor Michiko Inoue for her important suggestions on this research work.

I am also thankful to Assistant Professors Satoshi Ohtake and Tomokazu Yoneda for their continuous cooperation and suggestions.

I deeply appreciate Dr. Thomas Clouqueur, COE Assistant Professor, to join my research and gave me valuable suggestions.

I would like to thank to Professor Jacob Savir, New Jersey Institute of Technology, and Professor Kewal K. Saluja, University of Wisconsin-Madison, for their constant guidance and instructions throughout the course of this research.

I would like to express my special thanks to Dr. Kazuko Kambe,

Dr. Virendra Singh and other former and current members of Fujiwara laboratory for friendly discussion.

I am grateful to Professor Tomoo Inoue of Hiroshima City University for his friendly advice and useful comments.

NEC Electronics Corporation arranged and supported for me to study at NAIST. I am thankful to Mr. Masaaki Yoshida, Mr. Toshiharu Asaka, Mr. Hitoshi Sakuma and Mr. Kazuyoshi Yamada for their cooperation and support.

I would like to express my gratitude to the late Professor Teruhiko Yamada who taught me the basic theory of testing and diagnosis during undergraduate and master course study. I would also like to express my gratitude to former members of Yamada laboratory and Associate Professor Yukihiro Iguchi of Meiji University for their encouragement.

Finally, on a personal level, I wish to thank my parents, Makio and Tokiko Nakamura, my wife Akiko, and my daughters Hana and Suzu for their continuous support and encouragement.

References

[1] E. B. Eichelberger and T. W. Williams, "A Logic Design Structure for LSI Testability," Journal of Design Automation and Fault Tolerant Computing, vol. 2, pp. 165-178, 1978.

[2] M. Abramovici, M. A. Breuer, and A. D. Friedman, Digital Systems Testing and Testable Design, IEEE Press, Piscataway, 1994.

[3] P. H. Bardell, W. H. McAnney and J. Savir, Built-in Test for VLSI: pseudorandom techniques, Wiley Interscience, 1987.

[4] T. W. Williams and N. C. Brown, "Defect Level as a Function of Fault Coverage," IEEE Transactions on Computers, vol. C-30, No. 12, pp.987-988, 1981.

[5] J. Savir, "AC Product Defect Level and Yield Loss," IEEE Transactions on Semiconductor Manufacturing, vol. 3, No. 4, pp. 195-205, Nov. 1990.

[6] J. Savir, "AC Product Defect Level and Yield Loss," Proc. 1990 International Test Conference, pp. 726-738, Sept. 1990.

[7] Z. Stramenkovic, N. Stojadinovic and S. Dimitrijev, "Modeling of Integrated Circuit Yield Loss Mechanisms," IEEE Transactions on Semiconductor Manufacturing, vol. 9, No. 2, pp. 270-271, May 1996.

[8] F. Corsi, S. Martino and T. W. Williams, "Defect Level as a Function of Fault Coverage and Yield," Proc. European Test Conference, pp. 507-508, Apr. 1993.

[9] P. C. Maxwell, R. C. Aitken and L. M. Huisman, "The Effect on Quality of Non-Uniform Fault Coverage and Fault Probability," Proc. International Test Conference, pp. 739-746, 1994.

[10] E. S. Park, M. R. Mercer and T. W. Williams, "Statistical Delay Fault Coverage and Defect Level for Delay Faults," Proc. International Test Conference, pp. 492-499, Sept. 1988.

[11] E. J. Aas, "A Closer Look at Multiple Faults and Defect Levels in Digital Circuits," Proc. International Symposium on Circuits and Systems, pp. 441-444, June 1988.

[12] E. J. Aas and V. T. Minh, "Defect Level Calculation: the Importance of Accurate Models for Defect Distribution and Multiple Fault Coverage in Low Yield Situations," Proc. International Symposium on Circuits and Systems, pp. 939-944, 1989.

[13] C. N. Berglund, "A Unified Yield Model Incorporating Both Defect and Parametric Effects," IEEE Transactions on Semiconductor Manufacturing, vol. 9, No. 3, pp. 447-454, Aug. 1996.

[14] J. Dworak et. al. "Defect-Oriented Testing and Defective-Part-Level Prediction," IEEE Design & Test of Computurs, vol. 18, No. 1, pp. 31-41, Jan.-Feb. 2001.

[15] E. B. Eichelberger and T. W. Williams, "A Logic Design Structure for LSI Testability," Journal of Design Automation and Fault Tolerant Computing, vol. 2, pp. 165-178, 1978.

[16] V. F. Flack, "Estimating Variation in IC Yield Estimates," IEEE Jr. of Solid-State Circuits, vol. 21, No. 2, pp. 362-365, April, 1986.

[17] P. Franco, W. D. Farwell, R. L. Stokes and E. J. McCluskey, "An Experimental Chip to Evaluate Test Techniques Chip and Experiment Design," Proc. International Test Conference, pp. 653-662, 1995.

[18] J. Hirase, "Improvement of the Defect Level of Microcomputer LSI Testing," Proc. International Test Conference, pp. 377-383, Oct. 1995.

[19] W-B Jone, "Defect Level Estimation of Circuit Testing using Sequential Statistical Analysis," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 12, No.2, pp. 336-348, Feb. 1993.

[20] W-B Jone and S. R. Das, "A Stochastic Method for Defect Level Analysis of Pseudorandom Testing," Proc. International Conference on VLSI Design, pp. 382-385, Jan. 1998.

[21] W-B Jone, P. Gondalia and A. Gutjahr, "Realizing a High Measure of Confidence for Defect Level Analysis of Random Testing," IEEE Transactions on VLSI Systems, vol. 3., No. 3, pp. 446-450, Sept. 1995.

[22] C. Longeaud et. al. "New Techniques for the Characterization of Defect Levels in Semi-Insulating Materials," Proc. Semicondutcur And Insulating Materials Conference, pp. 72-75, June 1998.

[23] S-K Lu, T-Y Lee, and C-W Wu, "Defect Level Prediction using Multi-Model Fault Coverage," Proc. 1999 Asian Test Symposium, pp. 301-306, Nov. 1999.

[24] T. J. Powell et.al. "Correlating Defect Level to Final Test Fault Coverage for Modular Structured Designs," Proc. VLSI Test Symposium, pp. 192-196, Apr. 1994.

[25] J. Shier, "A Statistical Model for Integrated-Circuit Yield with Clustered Flaws," IEEE Transactions on Electron Devices, vol. 35, No. 4, pp. 524-525, April, 1988.

[26] A. D. Singh and C. M. Krishna, "On the Effect of Defect Clustering on Test Transparency and IC Test Optimization," IEEE Transactions on Computers, vol. 45, No. 6, pp. 753-757, June 1996.

[27] A. D. Singh and C. M. Krishna, "On Optimizing VLSI Testing for Product Quality using Die-Yield Prediction," IEEE Transactions

on Computer-Aided Design of Integrated Circuits and Systems, vol. 12, No. 5, pp. 695-709, May 1993.

[28] D. Singh, D. R. Lakin, and P. Nigh, "Binning for IC Quality: Experimental Studies on the SEMATECH Data," Proc. International Symposium on Defect and Fault Tolerance in VLSI Systems, pp. 4-10, Nov. 1998.

[29] J. T. De Sousa et. al. "Defect Level Evaluation in an IC Design Environment," IEEE Transactions . on Computer-Aided Design of Integrated Circuits and Systems, vol. 15, No. 10, pp. 1286-1293, Oct. 1996.

[30] J. J. T. Sousa and J. P. Teixeira, "Defect Level Estimation for Digital ICs," 1992 International Workshop on Defect and Fault Tolerance in VLSI Systems, pp. 32-41, Nov. 1992.

[31] J. T. Sousa et. al. "Fault Modeling and Defect Level Projections in Digital ICs," Proc. Europian Conference on Design Automation, pp. 436-442, Mar. 1994.

[32] C. H. Stapper, "On a Composite Model to the IC Yield Problem," IEEE Jr. of Solid-State Circuits, vol. 10, No. 6, pp. 537-539, Dec. 1975.

[33] Z. Stramenkovic and S. Mitrovic, "Integrated Circuit Yield Prediction," Proc. International Conference on Microelectronics, vol. 2, pp. 479-483, Sept. 1995.

[34] B. Wang, Y. B. Cho, S. Tabatabaei and A. Ivanov, “Yield, Overall Test Environment Timing Accuracy, and Defect Level Trade-offs for High-Speed Interconnect Device Testing,” Proc. Asian Test Symposium, pp. 348-353, Nov. 2003.

[35] H.Y.Chen, E.Manning and G.Mets, Fault Diagnosis of Digital Systems, John Willy & Sons Inc., 1970.

[36] J.A.Waicukauski and E.Lindbloom, “Failure Diagnosis of Structured VLSI,” IEEE Design & Test of Computeres, pp.49-60, vol. 6, No. 4, Aug, 1989.

[37] M.Abramovici and M.A.Breuer, “Multiple Fault Diagnosis in Combinational Circuits Based on Effect-Cause Analysis,” IEEE Transactions on Computers, vol.C-29,No.6, pp.451-460, 1980.

[38] K.Shigeta and T.Ishiyama, “An Improved Fault Diagnosis Algorithm Based on Path Tracing With Dynamic Circuit Extraction,” Proc. International Test Conference, pp.235-244, 2000.

[39] J.Rajski and J.Tyszer, “Fault Diagnosis in Scan-Based BIST,” Proc. International Test Conference, pp. 894-902, 1997.

[40] I.Bayraktaroglu, A.Orailoglu, “Improved Fault Diagnosis in Scan-Based BIST via Superposition,” Proc. Design Automation Conference, pp.55-58, 2000.

[41] I.Bayraktaroglu, A.Orailoglu, "Deterministic Partitioning Techniques for Fault Diagnosis in Scan-Based BIST," Proc. International Test Conference, pp.273-282, 2000.

[42] J. Ghosh-Dastidar and N.A.Touba, "A Rapid and Scalable Diagnosis Scheme for BIST Environments With a Large Number of Scan Chains," Proc. VLSI Test Symposium, pp.73-78, 2000.

[43] C.Liu and K.Chakrabarty, "A Partition-Based Approach for Identifying Failing Scan Cells in Scan-BIST with Application To System-on-Chip Fault Diagnosis," Proc. Design, Automation and Test in Europe, pp.230-235,2003.

[44] J.Ghosh-Dastidar, D.Das, A.Touba, "Fault Diagnosis in Scan-Based BIST Using both Time and Space Information," Proc. International Test Conference, pp. 95-102, 1999.

[45] J.Savir and W.H.McAnney, "Identification of Failing Tests with Cycling Registers," Proc. International Test Conference, pp.322-328, 1988, .

[46] T.R.Damarla, C.E.Stroud and A.Sathaye, "Multiple Error Detection and Identification via Signature Analysis," Journal of Electronic Testing: Theory and Applications, vol.7, pp.193-207,1995.

[47] Y.Wu and S.Adham ,"BIST Fault Diagnosis in Scan-Based VLSI Environments," Proc. International Test Conference, pp.48-57, 1996.

[48] J.Savir, "Salvaging Test Windows in BIST Diagnostics," Proc. VLSI Test Symposium, pp.416-425, 1997.

[49] T.Clouqueur, O.Ercevik, K.K.Saluja and H.Takahashi, "Efficient Signature-Based Fault Diagnosis Using Variable Size Windows," Proc. International Conference on VLSI Design , pp.391-396, 2001.

[50] C.Liu and K.Chakrabarty, "Failing Vector Identification Based on Overlapping Intervals of Test Vectors in a Scan-BIST Environment," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 22, No.5, pp.593-604, 2003.

[51] P.Wohl, J.A.Waicukauski, S.Patel and G.Maston, "Effective Diagnostics through Interval Unloads in a BIST Environment," Proc. Design Automation Conference, pp.10-14, 2002.

[52] T.W.Williams, W.Daehn, M.Gruetzner and C.W.Starke, "Bounds and Analysis of Aliasing Errors in Linear Feedback Shift Registers," IEEE Transactions on Computer-Aided Design of Intergrated Circuits and Systems, vol.7, No.1, pp.75-83, 1988.

[53] Texas Instruments, The TTL Logic Data Book, Dallas, 1988