An Experimental Chip to Evaluate Test Techniques
Part 1: Description of Experiment

By Piero Franco, Robert L. Stokes*, William D. Farwell*, and Edward J. McCluskey

* With Hughes Aircraft Co.

CRC Technical Report No. 94-5
(CSL TR No. 94-625)
June 1994

CENTER FOR RELIABLE COMPUTING

Computer Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University, Stanford, California 94305-4055

ABSTRACT

A Test Chip has been designed and manufactured to evaluate different testing techniques for combinational or full-scan circuits. The Test Chip is a 25k gate CMOS gate-array using LSI Logic's LFT150K technology, and includes support (design-for-testability) circuitry and five types of circuits-under-test. Over 5,000 die have been manufactured.

The five circuits-under-test include both data-path and synthesized control logic. The tests include design verification (simulation), exhaustive, pseudo-random, weighted random, and deterministic vectors for various fault models (stuck-at, transition, delay faults, and IDDQ testing). The chip will also be tested using the CrossCheck methodology, as well as other new techniques, including Stability Checking and Very-Low-Voltage Testing. The experiment includes an investigation of both serial and parallel signature analysis.

This report describes the Test Evaluation Chip Experiment, including the design of the Test Chip and the tests applied. A future report will cover the experimental results and data analysis.

OVERVIEW

The rest of this report describes the Test Evaluation Chip Experiment in detail. Broadly speaking, the purpose of this experiment is to investigate different test techniques for a variety of design and testing styles. This overview summarizes the main features of this experiment.

- **Realistic Design**
  - CUTs representative of real control and data-path logic
- **Thorough Test Evaluation**
  - Specifically designed chip, many different testing approaches evaluated
  - Exhaustive reference tests
  - Thousands of die manufactured
- **Design Styles**
  - Both logic synthesis and manual design
  - Complex gate, simple gate, and robust delay-fault testable
  - Worst case, statistical, and self-timed clocking
- **Test Pattern Sources**
  - Direct from ATE
  - Simulated serial scan
  - Pseudo-random BIST
- **Test Clocking**
  - At-speed -- both worst-case and aggressive timing
  - Delay tests
- **Response Analysis**
  - Boolean (voltage sampling), including serial and parallel signature analysis
  - IDDQ
  - CrossCheck
  - Post-Sampling Waveform Analysis (Stability Checking)

- **Major Participants**
  - Hughes Aircraft Co.
  - Stanford Center for Reliable Computing
  - LSI Logic
  - Digital Testing Services

TABLE OF CONTENTS

Abstract i
Overview ii
Table of Contents iii
List of Tables iv
List of Illustrations v
1 Introduction 1
  1.1 Purpose of Experiment 1
  1.2 Chip Design 2
  1.3 Testing Strategy 3
2 Circuits Under Test 4
  2.1 MULT12O12: 12x12 multiplier 5
  2.2 MULT6SQ: 6x6 multiplier followed by squarer 6
  2.3 RB Control Logic Blocks 6
3 Test Chip Architecture 8
  3.1 Block Diagram 8
  3.2 Data Source Modes 8
  3.3 Stability Checking 10
  3.4 Clocking Modes 11
    3.4.1 Standard Clocking 12
    3.4.2 External 2-Pattern Clocking 12
    3.4.3 Self-Generated 2-Pattern Clocking 14
  3.5 Observation Circuits 14
  3.6 Failure Counters 16
  3.7 Signature Analysis Register 17
4 Test Plan 18
  4.1 Gross Parametric Tests 18
  4.2 Support Circuitry Tests 18
  4.3 CUT Test Sets 18
  4.4 CUT Test Conditions 24
    4.4.1 Test Ordering 24
    4.4.2 Clocking Modes/Timing 24
    4.4.3 Shifted Vector Pairs 25
    4.4.4 Very-Low-Voltage 25
  4.5 Test Sequence 26

5 Conclusion and Future Work 28
Acknowledgments 29
Appendix A: Test Chip Data Sheet....................30
A.1 Test Chip Input/Output Signals....................30
A.2 Test Chip Operating Modes 32
A.3 Fail Counter Scan Chain Ordering 32
Appendix B: Test Conditions....................33
B.1 ....Stage 1 Tests 3
B.2 ....CrossCheck Tests 3
B.3 ....CUT Tests 3
Appendix C: Fault Coverage Graphs....................36
References 38

LIST OF TABLES

Table **Title**

1. Recent Test Experiments Reported....................2
2. CUT Types 4
3. Detectability of Stuck-at 0 Fault at LSB of 12x12 Multiplier 5
4. RB CUT Implementations 7
5. CUT Test Sets....................19
6. Representative Example of Single Stuck Fault Test Sets for Tool 2 20
7. Number of Vectors Needed to Achieve 100% Coverage of....................21
8. Signature Analysis Tests 24
9. Test Conditions....................26

LIST OF ILLUSTRATIONS

Figure	Title	
1.	Basic Data-Flow Block Diagram of Test Chip	3
2.	Testing Strategy (N0 and NP not disclosed)	3
3.	Partial Product 12x12 Multiplier	5
4.	RB Control Logic Synthesis Steps	6
5.	Block diagram of Test Chip	9
6.	Testing Parallel Data Load LFSR and CUT Inputs	10
7.	Waveforms for Stability Checking	11
8.	Timing Diagram for Standard Clocking	12
9.	Timing Diagram for External 2-Pattern Clocking	13
10.	Timing Diagram for Self-Generated 2-Pattern Clocking	14
11.	Response Observation Circuit	15
12.	Logic Diagram of Stability Checker	16
13.	Failure Counters (Scan chain shown in Fig. A.1 in Appendix A)	16
14.	Reconfigurable (a) Serial and, (b) Parallel Signature Analyzer	17
15.	Graph of Fault Coverage versus Test Length for RB_STD	22
16.	Relationship between Data Source Modes, Clocking Modes, and Timing	25
17.	Flow Diagram of Test Sequence	27
A1.	Scan Chain Connection for Failure Counters	32
C1.	Graph of Fault Coverage versus Test Length for MULT12O12	36
C2.	Graph of Fault Coverage versus Test Length for MULT6SQ	36
C3.	Graph of Fault Coverage versus Test Length for RB_SIMPLE	37
C4.	Graph of Fault Coverage versus Test Length for RB_ROBUST	37

1 INTRODUCTION

1.1 Purpose of Experiment

Many test techniques have been proposed to achieve the quality levels now required for digital integrated circuits. Since it is difficult to determine the effectiveness of tests, a 25k gate *Test Chip* has been designed and manufactured to evaluate the different test techniques for combinational or full-scan circuits, for a variety of design and testing styles. The objective of this experiment is to investigate and compare the escape rates for many different test techniques. This experiment is a collaboration between several industrial and university partners.

The tests include design verification, exhaustive, pseudo-random, weighted random and deterministically-generated vectors. Tests have been generated for stuck-at faults, transition faults, delay faults and IDDQ testing. The Test Chip will be tested using the CrossCheck methodology [Gheewala 88] [Gheewala 89] [Zarrinfar 93], and includes an investigation of the aliasing behavior in both serial and parallel signature analysis. Two new testing techniques will be evaluated in this experiment: delay testing by Stability Checking [Franco 91]; and Very-Low-Voltage Testing [Hao 93]. Packaging and burn-in of selected chips is also planned.

This report describes the Test Evaluation Chip Experiment, including the design of the Test Chip and the tests applied. A future report will cover the experimental results and data analysis.

Testing approaches need to be compared with respect to various criteria. These include the impact on the performance and area of the circuit, as well as the design time, test vector size, test time, automatic test equipment (ATE) requirements, and finally, the thoroughness of the test in detecting faulty circuits. Although all the above factors are considered in this experiment, the escape rate of the test will be the focus of this experiment. The reason is that the other factors can be calculated, but the escape rate of the test can only be estimated experimentally.

Similar experiments reported in the literature are summarized in Table 1. These experiments are either based on production runs or specially designed chips. The advantage of production runs is that a large sample size is available, but the number of test techniques compared is usually small. Fairly small sample sizes have been reported for specially designed chips. The main distinguishing feature of the experiment described here is that a much more thorough comparison of test techniques than has previously been reported is attempted, with a reasonable sample size. Over 5,000 die have been manufactured. It is expected that this experiment will provide some information on the usefulness of the different test techniques and fault models available.

Table 1. Recent Test Experiments Reported

Experiment	CUT	CUT Size	Number	Tests Applied
[Velazco 90] Grenoble	6800 68000 μP		75 IC's	Various Des. Verif. physically cut lines
[Elo 90] Intel	Gate array	12.5k gates	350,000 IC's	9 Modules of Boolean tests
[Das 90] Delco, U Nebraska		7,750 trans.	77,912 IC's	99.7% Stuck-At coverage
[Pancholy 90, 92] Cypress, McGill U	Special IC	9 Blocks	970 faulty blocks	Stuck-At, Open Transition, Robust
[Maxwell 91] HP	Std. cell	8,500 gates	18,500 IC's	6,623 Des. Ver., 77% 285 scan, 92%
[Maxwell 92] HP	Std. cell	8,577 gates	26,415 IC's	[Maxwell 91] + I_{DDQ}
[Sawada 92] Mitsubishi	Sea of Gates	3,000 gates 114k gates	48 IC's 400 IC's	Conventional (1 I_{DDQ}) versus I_{DDQ}
[Perry 92] Storage Tech.	Various	2.5-5.6k gates	3 years of I_{DDQ}	99.6% Stuck-At I_{DDQ}
[Schiessler 93] AT&T	Std. cell	10,000 gates	1,400 IC's	Functional, scan, I_{DDQ}
[Gayle 93] NCR	Various	Various	> 10^6 IC's	Stuck-At, I_{DDQ} At-Speed
[Wiscombe 93] VLSI Tech.	Gate Array	8,000 gates	10,000 die	Stuck-At, I_{DDQ}
This Experiment	Gate Array	25k gates 12k CUTs	>5,000 die	Many different test techniques

1.2 Chip Design

The Test Evaluation Chip is a 25k gate CMOS gate-array manufactured using LSI Logic's LFT150K FasTest array series (LFT150067) [LSI 92]. The chip has 96 I/O pins, 23 supply pins and five CrossCheck pins. The chip includes 5 types of combinational circuits-under-test (CUT) and support circuitry, as shown in Fig. 1. *Support circuitry* refers to both the on-chip data source logic for applying patterns to the CUTs, and the response analysis circuitry. There are four copies of each of the 5 different combinational CUTs on the chip.

A common data source is used for all CUTs, but there is separate response analysis circuitry for each CUT type. Tests can be applied either by the built-in self-test (BIST) circuitry or an external tester (ATE). At-speed, shifted vector pairs, and timed two-pattern tests can be applied to the CUTs. Both external and internally generated clocks can be used.

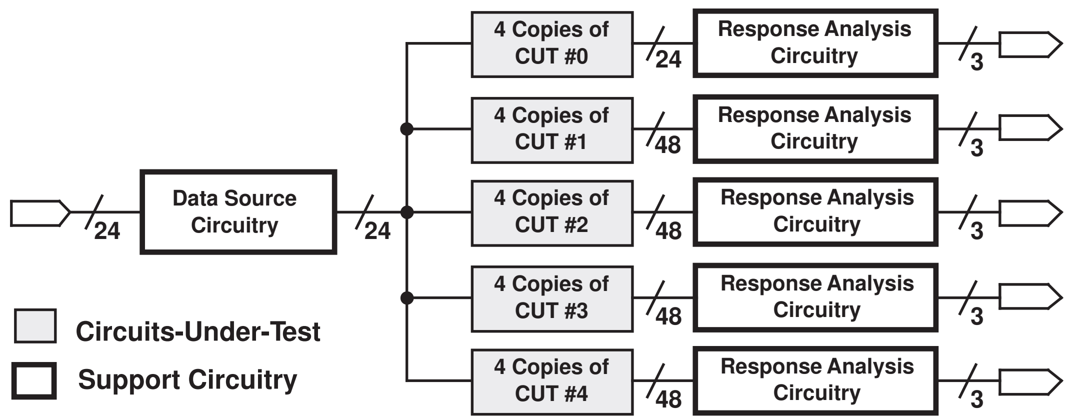


Figure 1. Basic Data-Flow Block Diagram of Test Chip

1.3 Testing Strategy

The two stage testing strategy shown in Fig. 2 resulted from the following considerations. A premise of this experiment is to make as much data publicly available as possible, without disclosing sensitive process information. Furthermore, the main interest is in failures that are fairly difficult to detect, not gross failures. This is because gross failures are generally easy to test, so the ability of tests to detect the difficult failures is more important for achieving high quality.

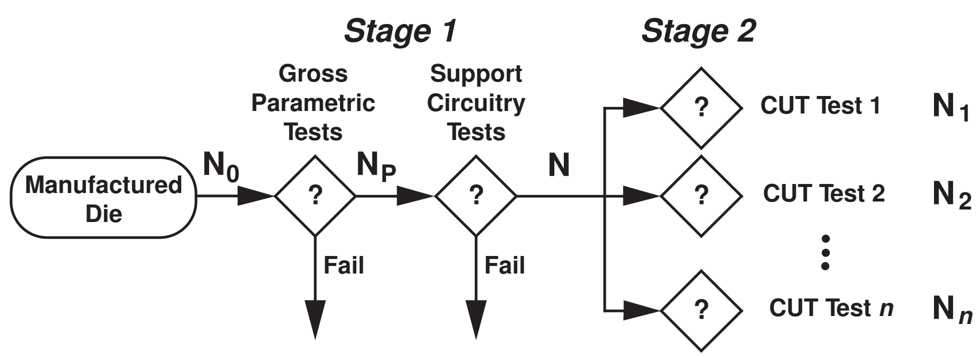


Figure 2. Testing Strategy (N_0 and N_P not disclosed)

Stage 1:

The first stage of testing includes the gross DC parametric tests, support circuitry boolean and IDDQ tests, and a test of the CrossCheck test circuitry. Only the die that pass all these tests will be available for this experiment. There is no constraint on reporting any data after this point.

Stage 2:

The second stage is the testing of the actual CUTs on those die that pass the Stage 1 tests. The wafers have been manufactured in three lots (plus a few in a fourth lot). These test sets are described in Sec. 4.

This report is organized as follows. The circuits-under-test are described in Sec. 2, followed by a more detailed description of the Test Chip Architecture in Sec. 3. The test plan is then discussed in Sec. 4. Section 5 concludes this report.

2 CIRCUITS UNDER TEST

The circuits-under-test (CUT) are described in this section. The CUTs are representative of data-path and control logic, as well as different design styles. The requirements for the CUTs were:

- Combinational logic.
- 24 or fewer inputs. The number of inputs was limited in order to permit exhaustive testing.
- Few outputs to limit the response analysis circuitry, as well as increase the test difficulty by reducing fault detectabilities. The CUTs have either 6 or 12 outputs.

There are five CUT types; two multipliers and three control logic blocks, shown in Table 2. The multipliers were designed by hand, and logic synthesis tools (Synopsys' Design Compiler and SIS from U.C. Berkeley) were used for the control logic.

The three control logic blocks perform the same function, but were designed with different styles. The first implementation was synthesized using all available gates in the LFT150K library, the second was restricted to elementary gates (NOT, AND, OR, NAND, NOR), and the third is robustly path-delay-fault testable. (A circuit is robustly path-delay-fault testable if there exists a robust delay test for every path through the circuit. Robust tests [Lin 87] are tests that cannot be invalidated by delays in other parts of the circuit.)

Table 2. CUT Types

Name	Inps	Outs	LSI Gates	Description
MULT12O12	24	12	1,146	12x12 multiplier
MULT6SQ	12	6	446	6x6 multiplier followed by a squarer
RB_STD	24	12	298	RB - all gates
RB_SIMPLE	24	12	380	RB - simple gates
RB_ROBUST	24	12	898	RB - robust path-delay fault testable

2.1 MULT12O12: 12x12 multiplier

This CUT is a 12x12 partial product multiplier made up of 6x6 multiplier building blocks, as shown in Fig. 3. The 6x6 multipliers use an add-and-shift algorithm and are implemented using AND gates and full adders. The multiplier has a nominal delay of 30 ns. The only significant point is that only the 12 most significant bits of the output are implemented. Using only the most significant bits reduces the response evaluation circuitry, as well as decreasing the observability of some faults in the multiplier. Multipliers have very high detectabilities, as shown by the example in Table 3, which shows the detectability (number of patterns that detect a fault) of the least significant input bit at the different outputs of the multiplier. Even if only the most significant bit of the output is used, faults on the least significant bit of the input are still detectable.

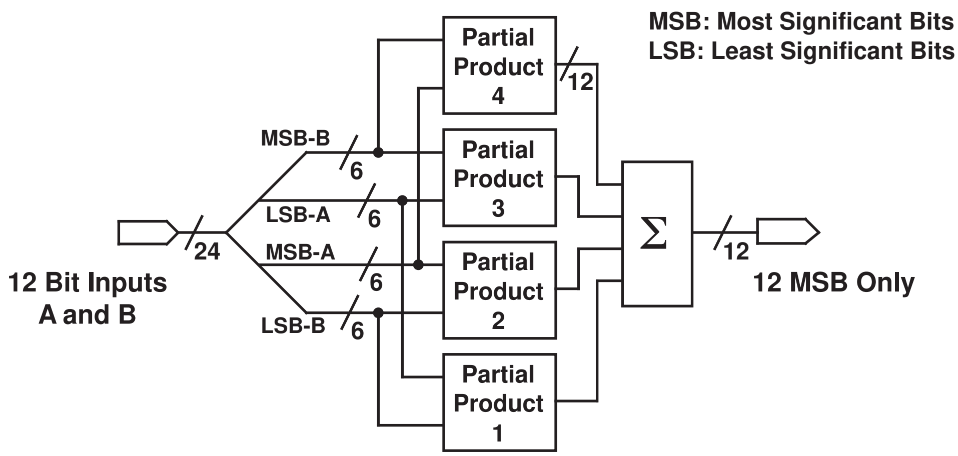


Figure 3. Partial Product 12x12 Multiplier

Table 3. Detectability of Stuck-at 0 Fault at LSB of 12x12 Multiplier

Output bit	Detectability
0-11	4,194,304
12	4,180,992
13	2,095,652
14	1,048,133
15	523,768
16	261,224
17	130,471
18	64,756
19	31,980
20	15,494
21	7,238
22	3,103
23	1,037

2.2 MULT6SQ: 6x6 multiplier followed by squarer

This CUT consists of two 6x6 multipliers similar to those used in the 12x12 multiplier. The multipliers are cascaded, and the second multiplier acts as a squarer since both inputs are fed by the output of the first multiplier. Some redundancies were eliminated by hand in the final circuit, and only the 6 most significant bits of the output are implemented. This CUT also has a nominal delay of 30 ns.

The main reason for including this circuit is to have at least one CUT with few enough inputs to permit N^2 exhaustive testing, where all possible pairs of transitions are applied. The N^2 exhaustive test provides a thorough reference for delay faults.

2.3 RB Control Logic Blocks

Whereas the previous circuits are data-path, this circuit is taken from a control logic block. This circuit is a portion of a larger DMA Read Buffer Controller (RB) used in a Hughes design. The original circuit had 34 inputs and approximately 3,700 literals as measured by the logic synthesis tools MIS/SIS from U.C. Berkeley [Sentovich 92] [Brayton 87]. Ten of the inputs were tied to 0 (to meet the 24 input CUT restriction), before synthesizing the different implementations.

One of the purposes of this part of the experiment was to investigate how the effectiveness of different testing techniques varies for different circuit implementations. The design criterion was to have all three implementations have the same delay. This reduces test time by allowing some of the tests to be performed in parallel. Figure 4 summarizes the steps taken to get the final circuits.

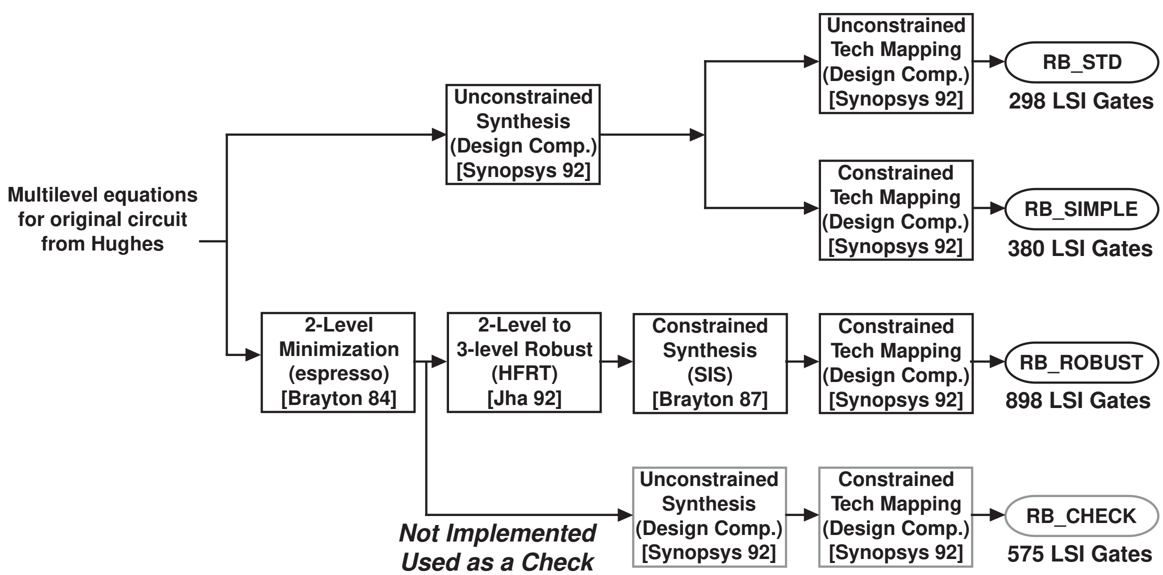


Figure 4. RB Control Logic Synthesis Steps

The first implementation is RB_STD, the "standard" implementation. It uses the complete LFT150K library, and includes complex gates (such as XOR and AND-OR-INVERT type gates). It was synthesized using Synopsys' Design Compiler [Synopsys 92], starting from the original multilevel netlist with 10 inputs tied to 0, with Boolean optimization turned on.

The second implementation is RB_SIMPLE, and uses only elementary gates. It was synthesized in the same way as RB_STD, except that the target library was restricted to elementary gates (AND, OR, NOT, NAND, NOR) for technology mapping.

The third implementation is RB_ROBUST, and was synthesized to be robustly path-delay-fault testable. The original netlist was collapsed to 2 levels and simplified using espresso [Brayton 84], since the synthesis tool used for achieving robust delay fault testability required a flattened netlist. A 3-level robustly path-delay-fault testable circuit was then generated using the procedure described in [Jha 92]. SIS was then used with constrained algebraic optimizations that preserve robustness to get a multi-level circuit. The technology mapping was done in Synopsys, using both simple gates and complex gates with no internal reconvergence.

The final RB_ROBUST circuit was generated first optimizing delay, and then RB_STD and RB_SIMPLE were synthesized with the same target delays. The sizes of the three implementations are given in Table 4 in terms of LSI gates. For example, a 2-input NAND has a gate count of 2, and a 4-input NAND gate has a gate count of 3 [LSI 92]. The number of paths without single-path-propagating hazard-free robust tests (SPP-HFRT) [Pramanick 90] are also shown in the table. Delays are pre-layout, and use estimates of the line capacitances.

Table 4. RB CUT Implementations

Circuit	Delay	LSI Gates	Physical Paths	No SPP-HFRT
RB_STD	8.3 ns	298 gates	841	125
RB_SIMPLE	8.2 ns	380 gates	841	125
RB_ROBUST	8.2 ns	898 gates	1773	6

Each of the circuits represents the best that could be achieved using the tools that were available. RB_ROBUST is larger than the other two implementations, but this is only partly due to the synthesis restrictions necessary to make the circuit robustly path-delay-fault testable. This is shown by RB_CHECK in Fig. 4. RB_CHECK was synthesized by first collapsing the circuit to 2 levels, and then regenerating the multi-level circuit with no robustness constraints. The size of RB_CHECK is 575 gates, which is significantly larger than RB_STD or RB_SIMPLE. Furthermore, SIS was used for RB_ROBUST, as it was not possible to do algebraic optimizations without complementation in Synopsys.

3 TEST CHIP ARCHITECTURE

The Test Chip consists of circuits-under-test and support circuitry. In order to investigate many different testing approaches and simplify the testing process, approximately 50% of the die area is taken up by support circuitry. The support circuitry is described in this section; the choice of CUTs was discussed in Sec. 2. A block diagram is shown first, followed by the data sources, clocking methodologies, and response analysis circuits. Finally, the reconfigurable signature analyzer is discussed.

3.1 Block Diagram

A block diagram of the Test Chip architecture is shown in Fig. 5. The chip has 64 inputs, 32 outputs, 10 power pins, 13 ground pins, and five CrossCheck pins. A brief description of the function of each pin is given in Appendix A.

Undetected faults in the support circuitry will bias the experimental results unpredictably. Therefore DFT techniques have been used to simplify testing the support circuitry. Eleven scan paths are used (five to scan out the counter values, five to scan the registers at the outputs of the CUTs, and one to scan out the signature register), as well as extra inputs and outputs to increase controllability and observability. Details of the scan paths are not shown in the block diagram (see Figs. 11, A1), neither are the enable signals for the CUTs (Fig. 6), and a controllable ring oscillator. The layouts of the four copies of each CUT do not overlap to avoid bridging faults between CUTs.

The different parts of the support circuitry are discussed in Secs. 3.2 to 3.7.

3.2 Data Source Modes

Data is applied to the CUTs on the Test Chip using an on-chip Parallel Data Load Linear Feedback Shift Register (LFSR). There are three ways to apply data to the CUTs:

1. **Direct**: Direct application of external vectors from the automatic test equipment (ATE). Both functional and two-pattern delay tests can be applied in this mode. For two-pattern tests, the CUT outputs are observed on even input vectors, and masked on odd input vectors.
2. **“Shifted Vector Pairs”**: Vectors are applied from the ATE, but shifted by one bit internally to simulate a scan chain. Each input vector becomes two vectors internally. The input vector shifted by one bit position is applied to the CUTs, followed by the vector itself. (This can also be done externally in Direct mode by supplying two-patterns tests from the ATE.)

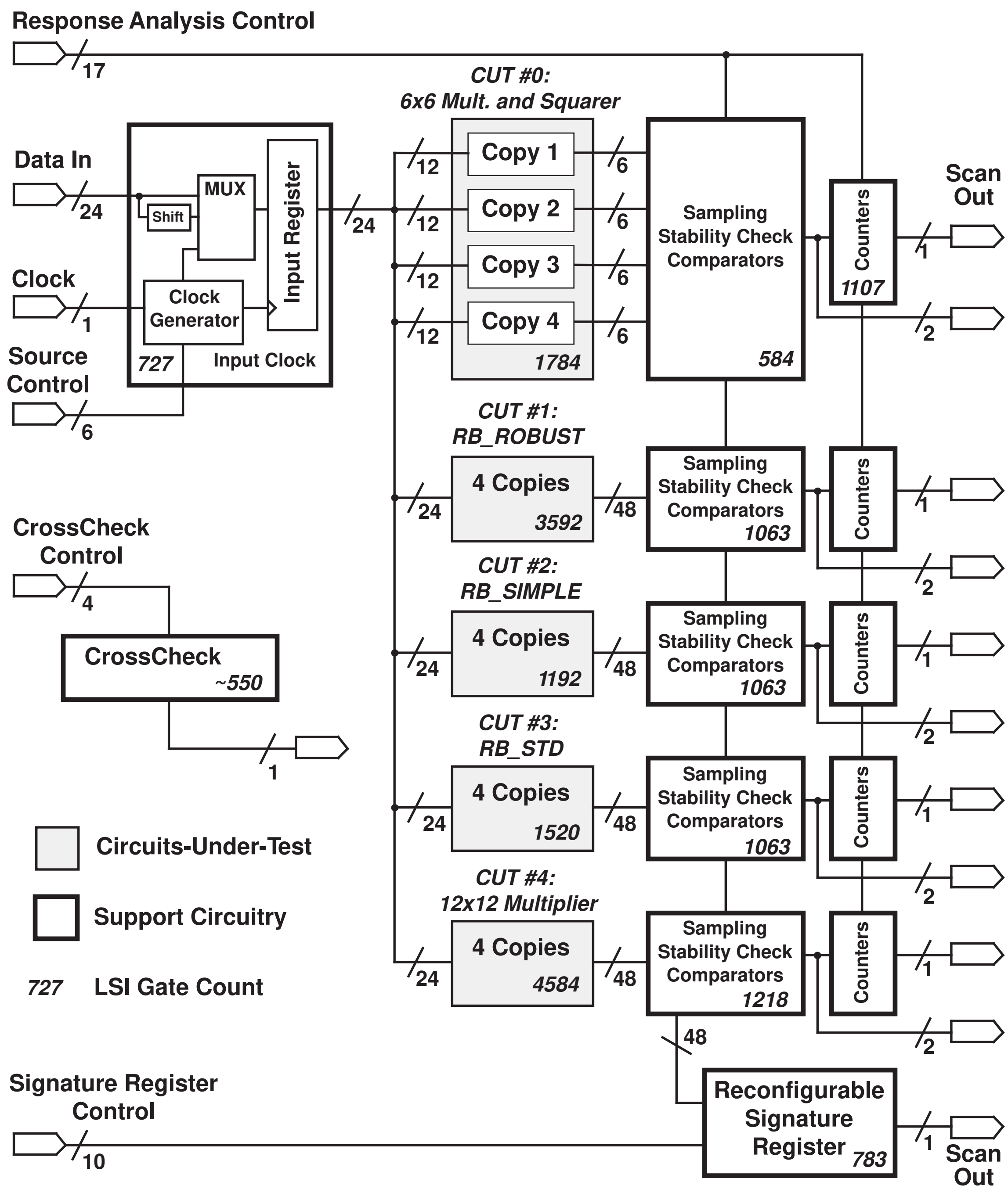


Figure 5. Block diagram of Test Chip

3. **Pseudo-random/Exhaustive**: The Parallel Data Load LFSR has a primitive characteristic polynomial $f(X) = X^{24} + X^7 + X^2 + X + 1$, and is used to generate pseudo-random test vectors. The LFSR cycles through all 2^{24}-1 non-zero states, generating an exhaustive test for the 24 input CUTs (the all-zero vector is applied in direct mode at the end of the test to save area).

The LFSR is also used to generate a super-exhaustive (N^2) test for the 12-input CUT. Super-exhaustive refers to all possible transitions or pairs of patterns. For an n-input circuit, the N^2 exhaustive test length is $2^n(2^n-1)$. This is implemented by taking every second stage of the LFSR.

Figure 6 shows how the Parallel Data Load LFSR and the inputs to the CUTs can be tested. The LFSR can be loaded directly from the primary inputs, and the most significant bit (MSB) is brought out of the chip so that the LFSR state can be observed directly, without having to propagate through the CUTs. The outputs of the Parallel Data Load LFSR are then fed to all the CUTs via CUT Enables. A large (108 input) OR gate has been included to observe the data applied to each CUT type. By loading a walking 1s pattern into the Parallel Data Load LFSR and using the CUT Enables, every input line to each CUT type can be tested. It is necessary to check the inputs to each CUT type, because if incorrect data is fed to all four copies of a CUT type, the four copies will produce the same incorrect result. This would not be signaled as an error, since the observation circuits (Sec. 3.5) compare corresponding outputs to detect errors.

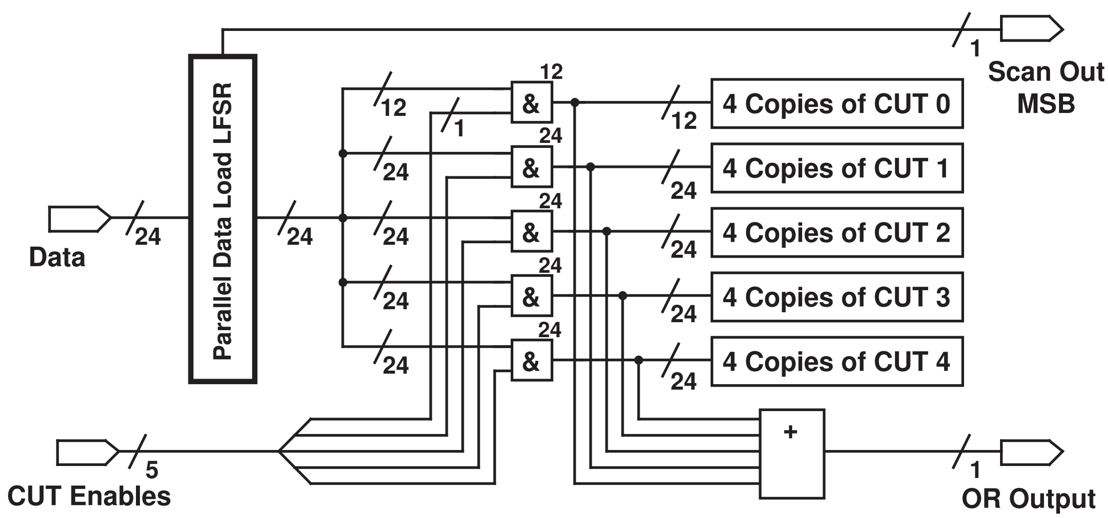


Figure 6. Testing Parallel Data Load LFSR and CUT Inputs

3.3 Stability Checking

Before describing the clocking modes and response analysis circuits on the Test Chip, Stability Checking will be reviewed. Output Waveform Analysis [Franco 91] is based on the principle that information about the delays in a circuit is distributed throughout the waveform at the output of the circuit, so it is useful to look at the waveform between samples when testing for timing failures. Stability Checking is a simple form of Output

Waveform Analysis, and is implemented on the Test Chip. The stability checkers included on the Test Chip are the first thorough experimental evaluation of the technique.

In a fault-free CUT, all the outputs are expected to be stable at their correct values by the sampling time. Therefore delay faults can be detected by designing a stability checker that analyzes the output waveform for any changes after the sampling time. The interval where the output is checked for stability is called the *Checking Period* or post-sample window.

The only complication is that the output of the CUT will start changing after the sampling time due to the application of the next test pattern to the inputs of the CUT. Therefore, implementation of Stability Checking involves both (1) designing circuits to do the checking, and (2) designing a test architecture that provides test patterns with the correct timing. Figure 7 shows the timing waveforms for Stability Checking. T_C is the cycle time for the circuit. Delay testing requires a vector pair $<V_1,V_2>$. The *initialization vector* $<V_1>$ is applied and once the circuit has settled, the test vector $<V_2>$ is applied. After the cycle time T_C, the outputs are sampled. The checking period for the vector pair $<V_1,V_2>$ starts after the sampling time, and ends before the application of the next vector $<V_3>$.

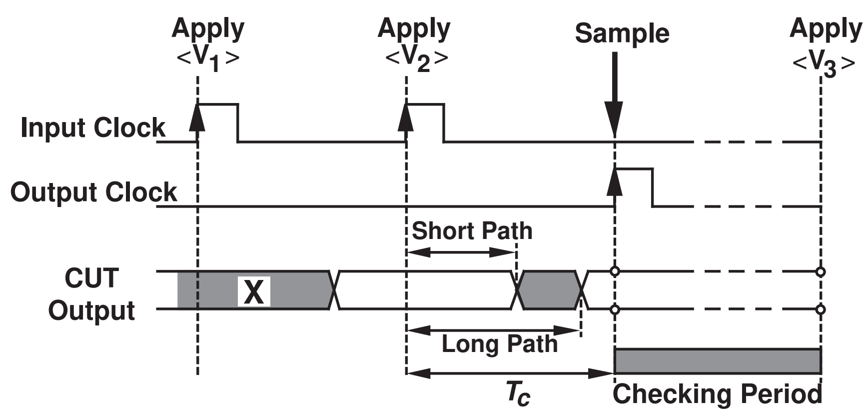


Figure 7. Waveforms for Stability Checking

3.4 Clocking Modes

Three clocking modes are investigated on the Test Chip. The first two are external clocks, and the third is generated internally with a delay line. Two of the clocking modes are designed to enable high speed testing to be done on low speed ATE.

The Input Register that applies patterns to the CUTs is clocked by the **Input Clock**, and the CUT outputs are sampled by the **Output Clock**. All registers are positive edge triggered flip-flops.

3.4.1 Standard Clocking

This is the simplest clocking mode. It corresponds to single clock synchronous designs, and patterns can be applied "at-speed". The master clock from the ATE is used for both the Input Clock and the Output Clock. The timing diagram is shown in Fig. 8.

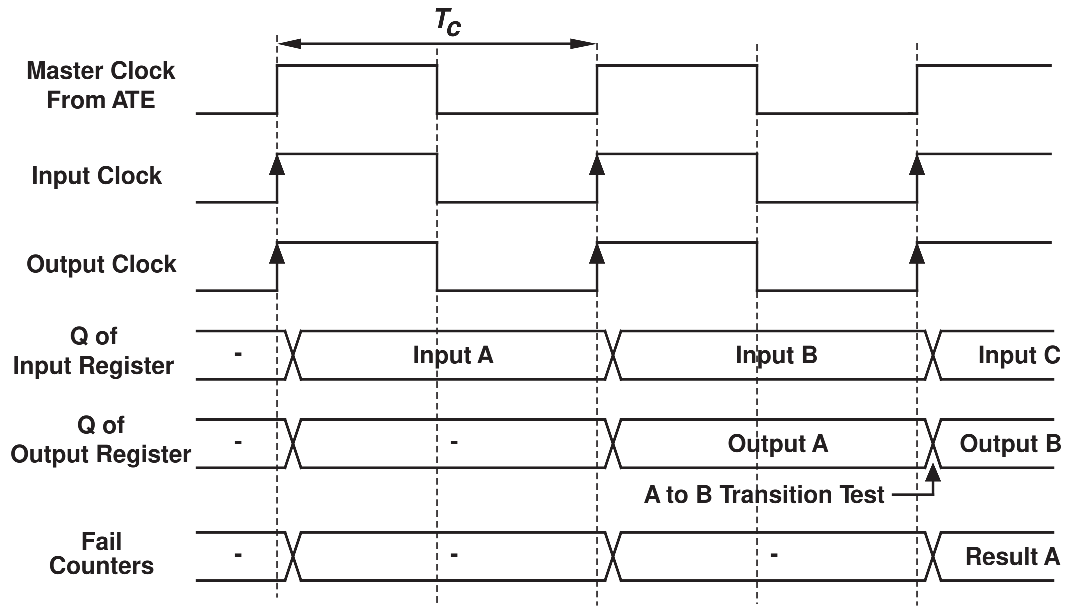


Figure 8. Timing Diagram for Standard Clocking

Three different clock rates will be investigated using this clocking mode. The first clock rate is based on *worst-case* timing design, whereas the second is more aggressive, and statistically based on typical delays. Certain tests will also be repeated at a slower clock rate to determine if errors are due to timing problems or "stuck-at" faults.

The standard clocking mode requires the ATE to be as fast as the CUTs, and was not part of the original design as a 100 MHz ATE was not available. During the design, a high speed ATE became available, so this clocking mode was implemented. The standard clocking mode requires the support circuitry to be faster than the other clocking modes, and since it was added late in the design, some parts of the support circuitry don't work at the most aggressive clock rates for the RB CUTs.

3.4.2 External 2-Pattern Clocking

The second and third clocking modes are *timed 2-pattern clocks*. This means that the time from the application of a pattern to the inputs of the CUT to the sampling of the outputs of the CUT is precisely controlled, but the rate of patterns applied is slower than in the standard clocking mode. The initializing vector is set up in the Input Register, and the

CUT outputs are allowed to settle. Then the test vector is applied, and sampled after the cycle time T_C of the circuit.

Two-pattern tests are used for delay testing and "shifted vector pair" testing in this experiment. All Stability Checking experiments are done using timed two-pattern tests.

The external 2-pattern clocking mode uses the master clock from the ATE. The Input Clock is generated from the rising transitions in the master clock, and the Output Clock is generated from the falling transitions in the master clock. One of the advantages of this clocking mode, is that as long as the duty cycle of the ATE clock can be precisely controlled, high speed testing can be done on a low speed tester. The timing diagram is shown in Fig. 9.

The timing diagram also shows the data at the inputs and outputs of the CUTs. The important point is that after the CUT outputs are sampled, the CUT inputs are not changed during the checking period of the Stability Checker. Therefore, if there are any changes in the CUT outputs during the Checking Period, they must be due to delay faults in the CUT.

Worst-case, typical design, and slow clock rates will be investigated using this clocking mode.

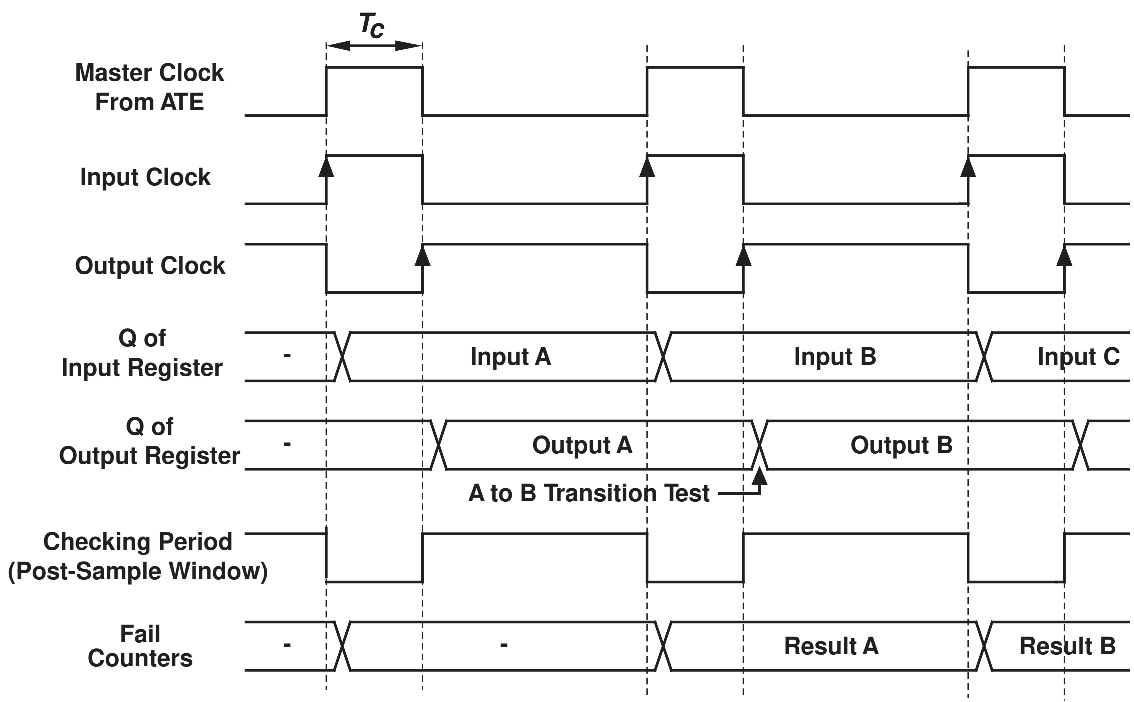


Figure 9. Timing Diagram for External 2-Pattern Clocking

3.4.3 Self-Generated 2-Pattern Clocking

The final clocking mode is "self-generated", where the clock rate is determined by the propagation delays on each individual die. The timing diagram is similar to Fig. 9, with the ATE master clock used as the input clock, but an internal delay line is used to set the offset on the output clock that determines the clock rate as shown in Fig. 10.

One reason for including the self-generated clocking mode is to improve the thoroughness of the experiment. When setting the external fixed clock rates, process drift must be taken into account, so the clock rate will be conservative for many die. With "self-generated" clocking, however, process drift is taken into account automatically, separating local timing defects from global delay variations.

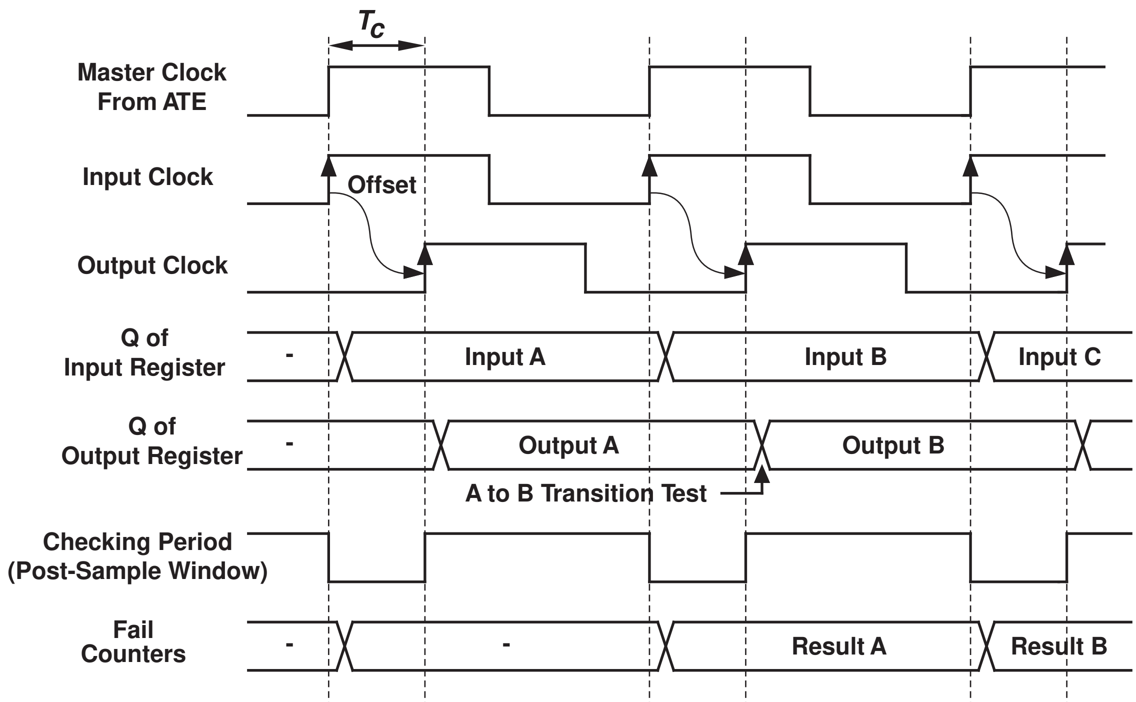


Figure 10. Timing Diagram for Self-Generated 2-Pattern Clocking

3.5 Observation Circuits

The response analysis circuitry on the Test Chip consists of the response observation circuits, failure counters, and a reconfigurable signature register. The design of the response observation circuit is shown in Fig. 11. Each CUT output is latched, and has a Stability Checker. The outputs of 4 copies of each CUT are compared to detect any errors. The latched outputs of Copy 1 are XORed with the corresponding outputs of Copies 2, 3, and 4. The outputs of the comparators are ORed, with a 1 indicating that an error has occurred in one of the sampled values. There is a small probability that all corresponding CUT outputs will be in error for the same vector, and the fault remains undetected. However this is unavoidable, as the tests are too long for the fault-free responses to be stored externally.

Similarly, the outputs of the Stability Checkers are ORed, with a 1 indicating a Stability Checking error has occurred. Note that comparators are not necessary for the Stability Checking, since the fault-free response is not required.

Figure 11 also shows that the flip-flops are scannable. This scan chain is useful for initializing the flip-flops and for testing the support circuitry.

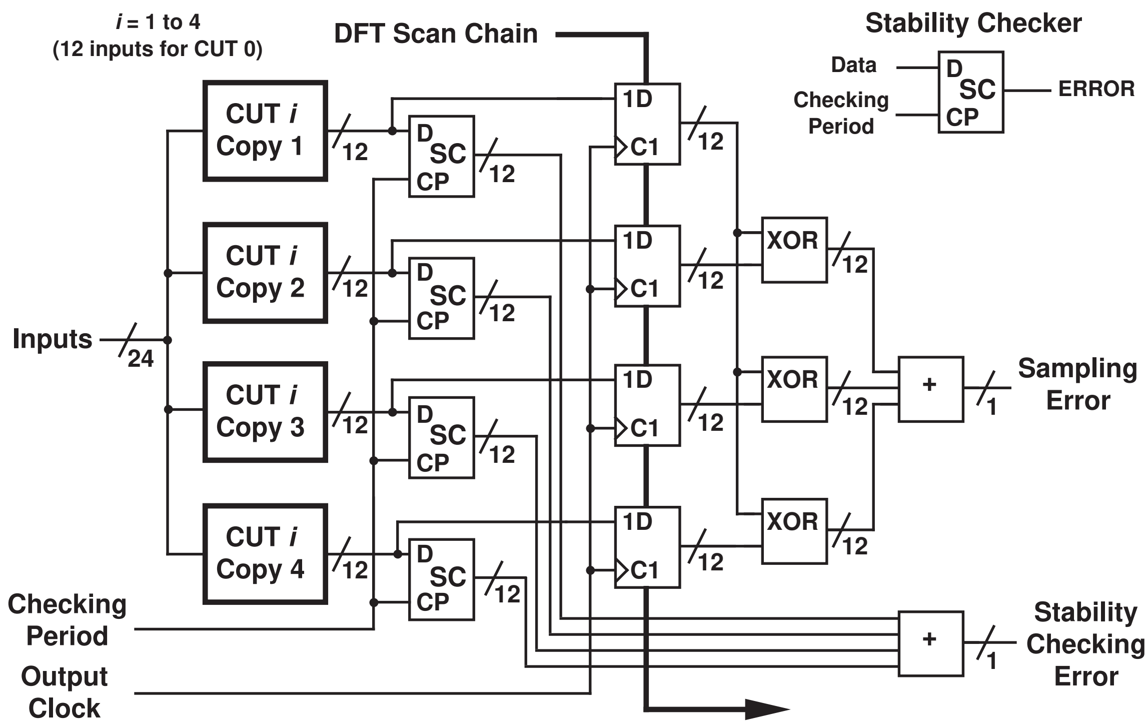


Figure 11. Response Observation Circuit

The logic diagram of the Stability Checker implemented on the Test Chip is shown in Fig. 12. D is the output of the CUT, and CP is the Checking Period signal. The output ERROR signal is driven high if there is any change in D while CP=1. The operation of the circuit is as follows. Assume D=0 when the checking period is inactive, CP=0. We have Y_1=0 and Y_2=1, so ERROR=0. Now the checking period starts, CP→1. If D ever rises, then Y_1→1 and ERROR→1. The circuit is symmetrical for detecting falling edges in D. The Stability Checker is reset every time the Checking Period = 0.

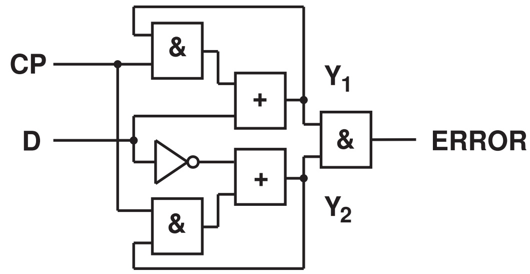


Figure 12. Logic Diagram of Stability Checker

A conservative implementation of the Stability Checkers was chosen for the Test Chip; the area can be reduced by sharing logic with the flip-flop. This is not desirable on the Test Chip, however, as the flip-flop value is needed for comparison.

Extra primary inputs have been provided to test the Stability Checkers. In test mode, controllable signals are multiplexed into the D and CP inputs of the Stability Checkers. This is necessary, because in a fault-free CUT, the Stability Checkers will never be activated.

3.6 Failure Counters

Failure counters are included on the Test Chip to record the first error and the number of errors, for both sampling and Stability Checking. A block diagram of the counters for one CUT type is shown in Fig. 13.

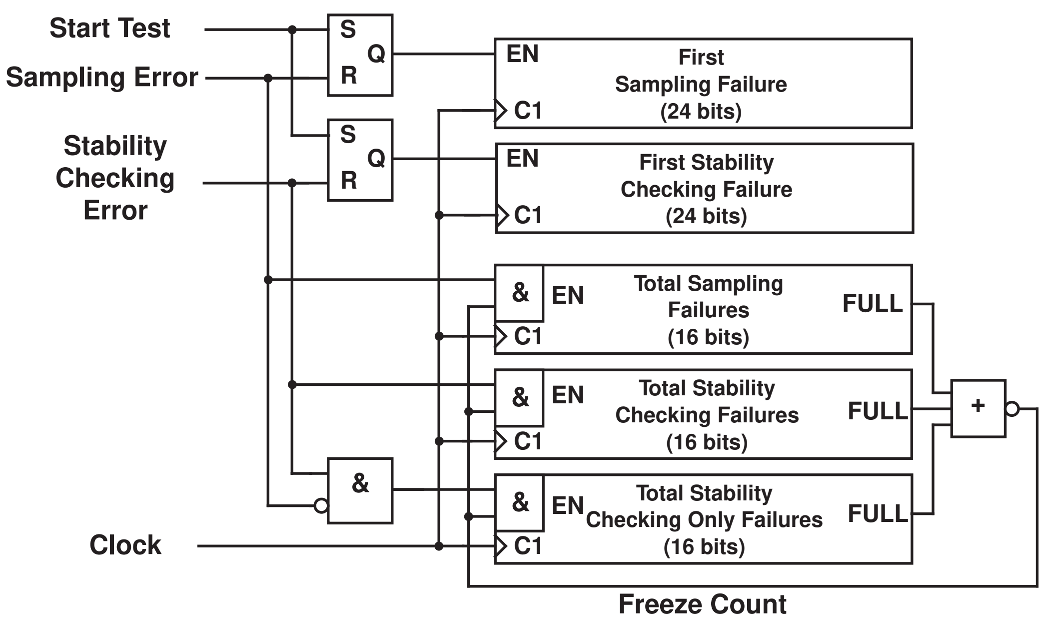
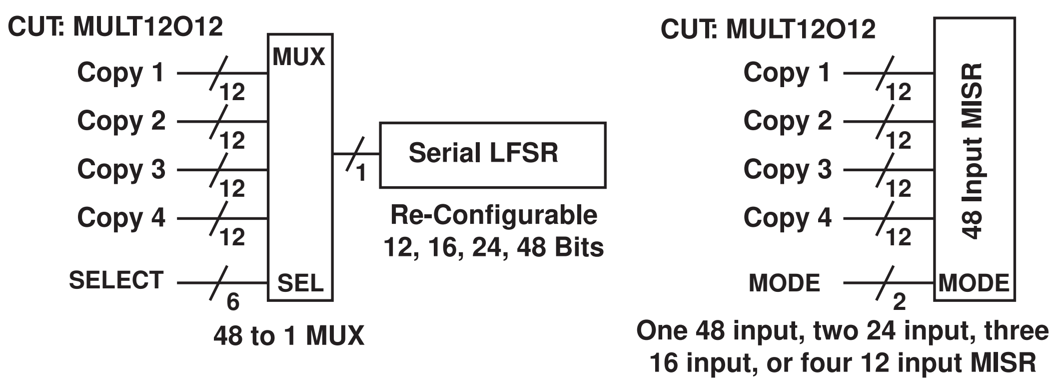


Figure 13. Failure Counters (Scan chain shown in Fig. A.1 in Appendix A)

The counters that record the test pattern that causes the first error are enabled until the first error resets the SR latch. Therefore, if no errors occur, the count will be the length of the test. Three counters have been included to count the number of errors, in order to distinguish between faults detected only by sampling, only by Stability Checking, or both. Due to area limitations, only 16 bit counters are used to count the total number of failures. The count values are 0 if no failures occur. Since the number of failures could exceed 2^{16}, if any of the three counters reaches the maximum count, all three counters are frozen. This way the relative counts are not corrupted. LFSR counters are used since they are significantly smaller than binary counters. The binary counts are recovered by post-processing the logged data from the ATE.

3.7 Signature Analysis Register

Hardware has been included in the Test Chip to investigate the aliasing behavior of both serial and parallel signature analyzers. Figure 14 shows the design of the reconfigurable Serial/Parallel signature analyzer. In serial mode, the size of the LFSR can be selected to be either 12, 16, 24 or 48 bits. A 48 to 1 multiplexer is used to select the desired CUT output bit. In parallel mode, the signature can be configured as a single 48 bit multiple input signature register (MISR), or as four 12 bit MISRs, one for each 12 output CUT Copy. Intermediate sizes of two 24 bit and three 16 bit MISRs are also possible.



(a) (b)

Figure. 14. Reconfigurable (a) Serial and, (b) Parallel Signature Analyzer
(Test mode scan chain not shown)

Due to area limitations, only one of the CUTs, the 12x12 multiplier, has a signature analyzer. The number of tests necessary for a comprehensive analysis is very long, so only a partial test described in Sec. 4.3 will be done on all die. A more thorough investigation is planned by either packaging 50 to 100 "interesting" die, or reprobing some of the die.

4 TEST PLAN

This section describes the test plan for the Test Chip. A brief description of the gross parametric tests and the support circuitry tests is given first (Stage 1 tests in Fig. 2), followed by a description of the CUT tests (Stage 2 tests in Fig. 2). The different conditions under which the CUT tests will be applied are given. Finally, the actual test sequence in the CUT test suite will be described. More detailed information on voltage levels and timing parameters for the tests can be found in Appendix B.

4.1 Gross Parametric Tests

The first tests applied will be conventional gross parametric tests. This includes tests for continuity, shorts, leakage, V_{IH} and V_{IL}, and a single IDDQ current measurement. The IDDQ measurement will be done with 0s applied to all CUT inputs.

4.2 Support Circuitry Tests

The support circuitry tests can be divided into two parts: The CrossCheck test logic, and the support logic. The CrossCheck test logic will be tested by a test set supplied by CrossCheck.

The support circuitry will be tested both with manually-generated functional vectors with 98% single-stuck-fault coverage, and a thorough IDDQ test. The IDDQ test was derived from vectors generated by two commercial ATPG tools, with a combined IDDQ coverage of 95% using 250 strobe points. The IDDQ test is in two parts since the clock is RZ (return-to-zero) in one part, and NRZ (non return-to-zero) in the second part. The number of vectors in the IDDQ test is over 100k as the scan chains on the Test Chip are used to set the state of the Chip, but the number of times the current is measured has been limited to 250, due to the time taken on the ATE for measuring quiescent current. (The IDDQ coverage cannot reach 100% since some parts of the support circuitry such as the ring oscillator are never activated.) As in the gross parametric IDDQ test, all 0s will be applied to the CUT inputs throughout the IDDQ test.

4.3 CUT Test Sets

This section lists the test sets applied to the CUTs. Many test sets have been included in an attempt to make the experiment as thorough as possible. The tests are summarized in Table 5. Both university and commercial ATPG tools have been used to generate the CUT tests. The commercial tools include tools from AT&T, CheckLogic, ExperTest, GenRad, IBM, Sunrise Test Systems, and Syntest; University tools include tools from U. Illinois, U. Iowa, VPI&SU, U. Texas and Yale U. The tools are not identified by name, since some of the vectors were only provided under these conditions.

Table 5. CUT Test Sets

#	Test	Test Length (Number of patterns)				
		6SQ	M12	STD	SIM	ROB
1.1	Design Verification	90	57	-	-	-
2.1	SSF Tool 1 (100%, gate faults)	34	62	68	79	244
2.2	SSF Tool 2 (100%, gate faults)	74	163	129	144	490
2.3	SSF Tool 2 (100%, pin faults)	34	61	129	144	489
2.4	SSF Tool 3 (100%, pin faults)	22	21	69	82	262
2.5	SSF Tool 3 (100%, compressed)	-	-	62	66	234
2.6	SSF Tool 4 (100%, gate faults)	39	68	72	93	275
2.7	SSF Tool 4 (99.0%)	38	78	71	91	254
2.8	SSF Tool 4 (98.0%)	39	69	80	87	245
2.9	SSF Tool 4 (95.0%)	39	62	73	74	219
2.10	SSF Tool 4 (90.0%)	35	63	67	72	190
2.11	SSF Tool 4 (80.0%)	20	49	48	58	159
2.12	SSF Tool 4 -- Min 5 Detects/Fault	168	258	339	397	1235
2.13	SSF Tool 4 -- Min 15 Detects/Fault	473	754	1046	1163	3745
3.1	Switch-level ATPG	56	110	108	109	327
4.1	Pseudo-Random/Exhaustive	4096	2^{24}			
	N^2 Exhaustive (same as 4.1)	2^{24}				
5.1	Weighted Random - (WR-MUR)	417	23,000	3404	1438	34,000
5.2	Weighted Random - (WR-WAI)	372	12341	634	738	7807
6.1	Stuck-Open ATPG (equiv gate)	153	269	203	219	766
7.1	Transition Fault, ATPG Tool 5	68	84	222	256	796
7.2	Transition Fault, ATPG Tool 6	304	434	274	292	586
8.1	Gate Delay Fault -- X→0*	976	-	312	304	612
8.2	Gate Delay Fault -- X→ran	976	-	312	304	612
9.1	Path Delay -- Critical Path - X→0	1692	620	992	408	400
9.2	Path Delay -- Critical Path - X→ran	1692	620	992	408	400
9.3	Path Delay -- Robust Test - X→0	-	-	2864	2864	7068
9.4	Path Delay -- Robust Test - X→ran	-	-	2864	2864	7068
9.5	Path Delay -- Robust Test	-	-	3044	3044	7092
9.6	Path Delay -- Non-Robust-A	-	-	562	542	884
9.7	Path Delay -- Non-Robust-B	-	-	2164	2156	4136
10.1	IDDQ ATPG Tool 6	72	90	35	36	65
10.2	IDDQ ATPG Tool 7	10	19	22	27	68
10.3	IDDQ Pseudo-Random	64	64	64	64	128
11.1	CrossCheck Test	3556				
11.2	Modified CrossCheck Tests	3556	3556	3556	3556	3556
12.1	Signature Analysis Test	12.544M				

* X→0 means that X is replaced by 0; X→ran means that X is replaced randomly with 0 or 1.

The CUT test sets are discussed in more detail below.

Design Verification Tests (Test 1.1)

The first set of tests are design verification vectors. These were manually generated by the designer to verify the functionality of the CUTs. There are no design verification vectors for the RB control circuits, as these are only a part of a control logic design and the function is not known.

Single-Stuck-Fault Tests (Tests 2.1 to 2.13)

These are conventional stuck-at tests. Many different stuck-fault tests have been included, since stuck-fault testing is used almost universally in industry. Separate tests have been included for faults modeled at the I/Os of the LSI cells (pin faults), and internal faults for the complex LSI cells. Test sets that detect every fault at least 5 times and 15 times have been included, as well as tests with lower fault coverages (between 80% and 100%).

Separate test sets with lower fault coverage have been included, even though the first fail counter records at precisely what fault coverage the CUT failed the 100% coverage test. This was done to avoid the assumption that a 90% test set is a subset of a 100% test set. The reason is that most tools do a reverse fault simulation to compress the test set. Therefore to avoid the concern that the results might be biased by assuming the 90% test set is a subset of the 100% test set, both test sets are included.

Table 6 shows an example of the stuck-at ATPG results. There are no aborted faults. There is not much difference between test sets generated using pin faults and internal faults for the RB circuits since no large LSI cells are used (only elementary gates, and AOI and XOR type gates), but there is a significant difference for the multipliers due to the large full-adder macrocell used.

Table 6. Representative Example of Single Stuck Fault Test Sets for Tool 2

Circuit	Elementary Gate		Pin Faults	
	Faults	Patterns	Faults	Patterns
RB_STD	644	129	650	129
RB_SIMPLE	674	144	678	144
RB_ROBUST	1816	490	1830	489
MULT12O12	3392	163	1310	61
MULT6SQ	1318	74	500	34

Switch-Level Tests (Test 3.1)

This is a switch-level single-stuck-fault test set generated using the transistor-level representation of the LSI cells given in the LFT150K data book.

Pseudo-random Tests (Test 4.1)

This is the pseudo-random test generated by the Parallel Data Load LFSR. This test is also the exhaustive test (N^2 exhaustive for MULT6SQ) by clocking the LFSR through all the possible states as described in Sec. 3.3. The pseudo-random vectors are generated with the primitive polynomial $f(X) = X^{23} + X^6 + X + 1$, with alternating 1s and 0s as the initial contents of the LFSR. The single-stuck-fault coverage versus pseudo-random pattern test length is shown in the graphs in Fig. 15 and Appendix C. Note that the vertical axes (Fault Coverage) do not start at 0%.

Weighted Random Tests (Tests 5.1 and 5.2)

Two weighted random pattern generation algorithms have been used. The patterns will be applied externally with the ATE. The first algorithm [Muradali 90] uses equally-weighted pseudo-random vectors, followed by a single set of weights computed using ATPG (WR-MUR in Table 7). The second algorithm [Waicukauski 89] uses initial weights and then multiple weight distributions, also based on ATPG (WR-WAI in Table 7).

Table 7 shows the test length required to detect all non-redundant single-stuck-faults in the CUTs, and is based on elementary gate fault simulation. The simulator described in [Lee 91] was used.

Table 7. Number of Vectors Needed to Achieve 100% Coverage of Detectable Single-Stuck Faults

Circuit	LFSR PR Vectors	WR-MUR Vectors	WR-WAI Vectors	WR-WAI No. Weights
RB_STD	12,316	3,404	634	2
RB_SIMPLE	12,320	1,438	738	3
RB_ROBUST	> 200,000	> 34,330	7,807	7
MULT12O12	63,964	> 23,332	12,341	6
MULT6SQ	1309	417	372	2

The test length versus fault coverage of detectable single-stuck-faults for the equally-weighted pseudo-random and weighted random tests are shown in Fig. 15 for the RB_STD CUT as an example. The graphs for the other CUTs can be found in Appendix C.

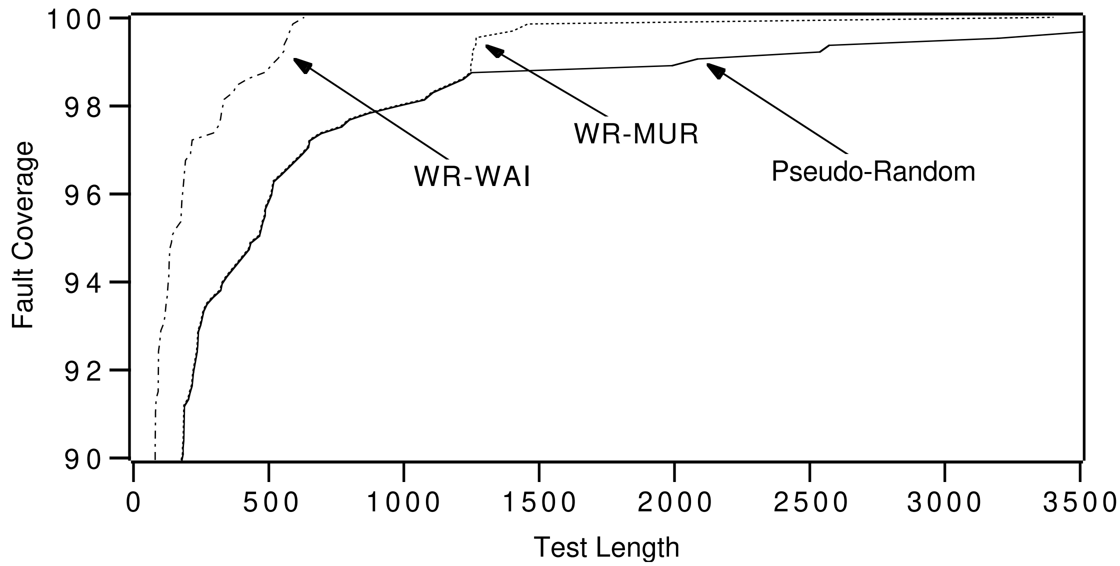


Figure 15. Graph of Fault Coverage versus Test Length for RB_STD

Stuck-Open Tests (Test 6.1)

This test is generated for transistor stuck-open faults. The LSI cells were modeled as elementary gates, which is not accurate, especially for transistor level faults in some of the complex gate macrocells. This test was included since we were not able to generate a more accurate stuck-open test set.

Transition Fault Tests (Tests 7.1 and 7.2)

Two deterministic test sets for transition faults has been generated using commercial APTG tools.

Gate Delay Tests (Tests 8.1 and 8.2)

A gate delay test has been generated in which each gate is tested through the longest path (path with the greatest delay) through the gate. A university tool was used to generate the tests, and unfortunately there was a problem converting the 12x12 multiplier netlist to the required format, so this test will not be performed on the MULT12O12 circuit.

The gate delay test is applied twice to investigate the effect of many signals changing simultaneously in the CUT. This is done by replacing all “X’s” by 0s in one test to minimize the number of transitions propagating through the circuit, and replacing “X’s” with 0s and 1s randomly in the second test. The second test has more transitions than the first test, and could give different results due to signal coupling and ground bounce.

Path Delay Tests (Tests 9.1 to 9.7)

Two types of path delay tests are investigated. Path delay tests have been generated for all paths that are greater than a certain fraction of the longest path in each circuit (normally called *critical* paths). An attempt was also made to generate path delay tests for every path in the CUTs. This was not possible for the multipliers, since there are too many paths. There are 4.2×10^{12} structural paths in MULT12O12, and 7×10^{15} structural paths in MULT6SQ. It was possible to generate complete robust path delay fault tests for the RB circuits (14.8% of the paths in RB_STD and RB_SIMPLE did not have robust tests). Two robust test sets have been included for the RB circuits. As described in the gate delay tests, some of the path delay tests are applied twice, one minimizing the number of extra transitions.

Two non-robust path delay test sets have also been included.

IDDQ Tests (Tests 10.1 to 10.3)

Two ATPG tools have been used to generate IDDQ tests. These test sets generally have fewer vectors than the single stuck-at tests, as expected. Only one CUT type is enabled at a time to localize the cause of high IDDQ. Since the inputs to the other CUTs do not change, any high IDDQ value must be due to the CUT being tested. Pseudo-random vectors of length 64 (128 for RB_ROBUST) are also used as a IDDQ test. The current is measured for every pseudo-random vector.

CrossCheck Tests (Tests 11.1 to 11.2)

The CrossCheck tests have been provided by CrossCheck. These are the CrossCheck tests for the CUTs and support circuitry, not tests of the CrossCheck test logic (Sec. 4.2). Note that the CrossCheck tests do not distinguish between CUT and support circuitry failures.

A modified CrossCheck test has also been generated. This test tries to distinguish between CUT failures by enabling one CUT at a time in the same way as the CUT IDDQ tests.

Signature Analysis Tests (Test 12.1)

As discussed in Sec. 3, there is not enough tester time to do a complete signature analysis evaluation, so only pseudo-random patterns are used for the signature analysis test. This test was chosen since signature analysis is often used in pseudo-random Built-In Self-Test (BIST) environments. The test length required to detect all detectable single stuck-at faults at the elementary gate level for the MULT12O12 CUT is 63,964 (For alternating 0s and 1s initial seed vector). Therefore the signature analysis test will be based on repeated 64k vector test sets, with the reconfigurable signature register in different modes. Table 8 summarizes the modes, which include taking 10 intermediate signatures for two of the MISR tests (this data is useful for investigating different signature analysis schemes, such as fuzzy signatures [Wu 92], where the intermediate signatures are ORed together to reduce the signature storage requirements). Apart from investigating the aliasing behavior of signature analysis, this part of the experiment will also provide information on how closely real defects are modeled by stuck-at faults. This will be done by compiling a fault dictionary of the faulty signature for every stuck-at fault in the CUT. The faulty signatures can then be compared to the signatures generated by all possible single stuck-at faults, to determine if the faulty CUT "behaves" as a stuck-at fault.

Table 8. Signature Analysis Tests

Mode	Test Length	# Signatures
48 bit MISR, Intermediate Sig.	64k	10
Two 24 bit MISRs	64k	1
Three 16 bit MISRs	64k	1
Four 12 bit MISR, Intermediate Sig.	64k	10
48 bit Serial LFSR, all 48 CUT Outputs	3,072k	48
24 bit Serial LFSR, all 48 CUT Outputs	3,072k	48
16 bit Serial LFSR, all 48 CUT Outputs	3,072k	48
12 bit Serial LFSR, all 48 CUT Outputs	3,072k	48

Propagation Delay Measurements

This test is not a CUT test, but is done as part of the Stage 2 tests. The propagation delay of the internal delay lines used to generate the "self-generated" clock mode is measured. This test gives some information of the variance in overall speed for the different die.

4.4 CUT Test Conditions

The CUT test sets will be run under the different test conditions described below.

4.4.1 Test Ordering

The external, 2-pattern clocking exhaustive tests will be applied at the beginning and end of the test suite. This is done to verify the repeatability of the experiment. If the results differ, then either the die failed during the test, or the repeatability of the tester is poor.

4.4.2 Clocking Modes/Timing

Each test is run at different speeds for the three clocking strategies described in Sec. 2.4. Standard clocking will be done both for worst-case (**spW**), typical (**spF**) (not for the RB circuits), and slow (**spS**) timing. Similarly, externally applied 2-pattern clocking will be done at worst-case (**puW**) statistical (**puF**), and slow (**puS**) timing. Typical timing is 25% faster than worst-case timing for the multipliers, and 5% faster than worst-case timing for the RB CUTs, and slow timing is 50% slower than worst-case timing. The timing for the "self-generated" clocking mode (**LF**) is determined by internal delay lines on the die. This is summarized in Fig. 16.

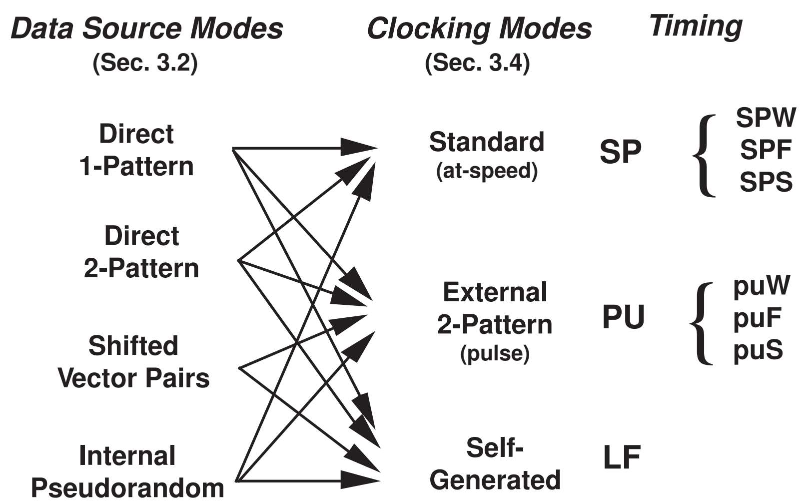


Figure 16. Relationship between Data Source Modes, Clocking Modes, and Timing

4.4.3 Shifted Vector Pairs

Not all possible pairs of patterns can be applied using a scan chain. This is an issue for two-pattern test sets such as stuck-open and delay tests. Therefore, the effect of applying patterns via a scan chain (**shift**) is investigated in this experiment. This is done by applying each vector externally, first shifted by one bit position, and then the vector itself. This is not done with the long internally generated pseudo-random tests.

4.4.4 Very-Low-Voltage

Very-Low-Voltage testing [Hao 93] in which the supply voltage is reduced below the normal operating range is investigated on the Test Chip. The idea is to provoke functional failures in weak circuits by operating the circuit at a reduced supply voltage. The input signal and supply voltage is 1.7 volts, and clocking is done at a reduced speed. The reduced clock rate was determined from the Test Chip prototypes, and is 5.6 times slower than the clock rate at 5 volts. The external 2-pattern clocking mode is used for the Very-Low-Voltage tests (**puV**). The long internally generated pseudo-random tests are not applied at Very-Low-Voltage to reduce tester time.

Table 9 summarizes the test conditions for the CUT tests described in Sec. 4.3.

Table 9. Test Conditions

#	Test	Direct 1-Pattern	Direct 2-Pattern	Shifted	Very-Low Voltage
1.1 2.1-2.13 3.1	Des. Ver. SSF Switch-Level	SP PU LF		PU LF	puV
4.1	Pseudo-Random	SP PU LF			
5.1-2	Weighted Random	SP PU LF			puV
6.1	Stuck-Open	SP PU LF		PU LF	puV
7.1-2 8.1-2 9.1-5	Transition Fault Gate Delay Fault Path Delay Fault		SP PU LF		puV
9.6-7	Non-Robust	SP PU LF			puV
12.1	Signature Analysis	spW			

4.5 TEST SEQUENCE

The flow diagram for the test sequence is shown in Fig. 17. The test is aborted as soon as one of the Stage 1 tests fail.

The test time for the Stage 1 tests is under 10 seconds, and the test time for a die that passes all Stage 2 tests is one minute. The test time for a die that fails the Stage 2 tests depends on the number of tests that fail. Data is collected for each die that passes the Stage 1 tests. For each die, the speeds of the internally generated clocks and ring oscillator are recorded. The actual current values for the IDDQ tests are also recorded whether the tests fail or not. For each test that fails, the data in the failure counters is recorded:

1. Vector number of first sampling failure.
2. Vector number of first stability checking failure.
3. Number of errors detected by:
   - Sampling, S;
   - Stability Checking, P;
   - Stability Checking, but not sampling, $P\overline{S}$.
4. Serial and parallel signatures for MULT12O12 tests.

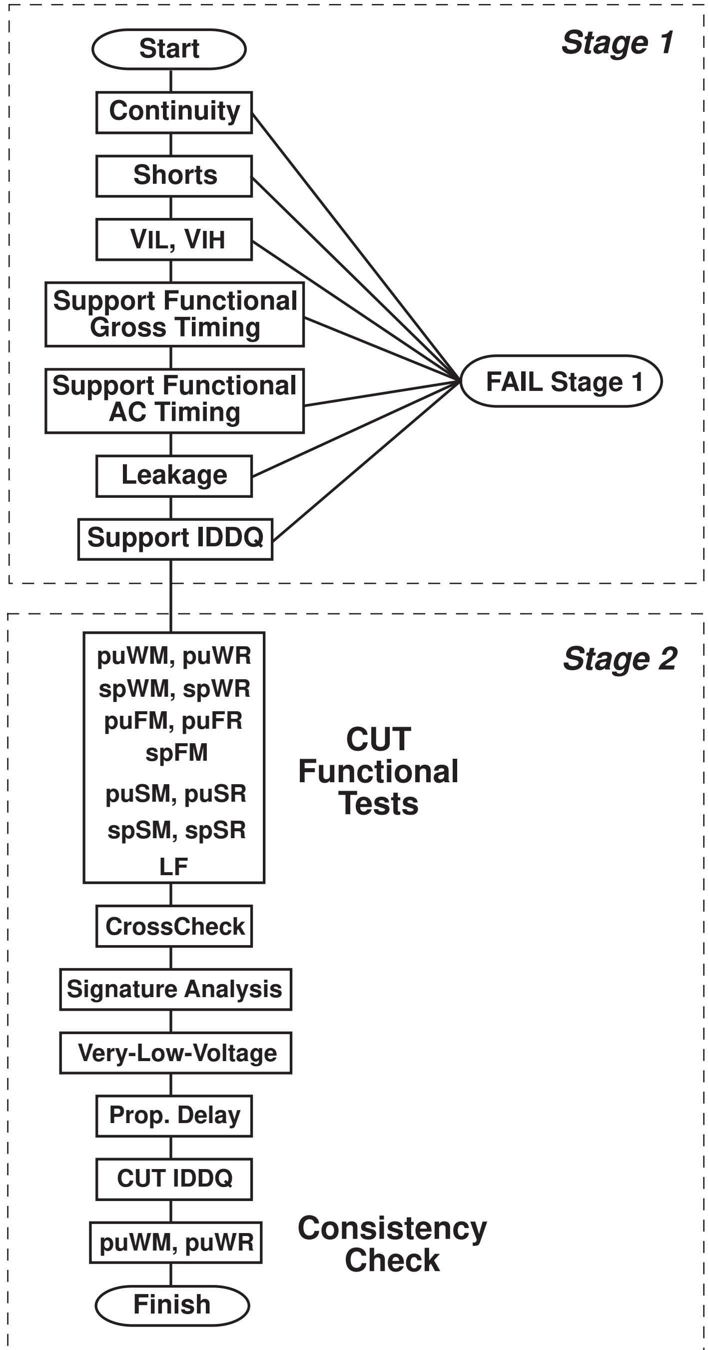


Figure 17. Flow Diagram of Test Sequence

5 CONCLUSION AND FUTURE WORK

This report has described the Test Evaluation Chip Experiment, including the design of the Test Chip and the test sets that will be applied. The experimental results and data analysis will be presented in a future report.

The design of the Test Chip itself was not a small task, and has been a learning experience. Due to the different formats in use, many netlist and vector translations were necessary to generate the test sets.

The choice of CUTs was the subject of much discussion. There is a tradeoff between making the CUTs large to better approximate "real" designs, and making them small to enable thorough testing and a larger sample size. Since the emphasis in this experiment is on thoroughness, the largest CUTs for which exhaustive tests could be reasonable applied were chosen. This limited the number of inputs to 24.

This experiment should shed some light on the usefulness of the different testing approaches and fault models available. It is also hoped that this work can be used as a stepping stone for further experiments.

ACKNOWLEDGMENTS

We would like to acknowledge the many people who contributed to this experiment and helped with the project. In particular, we would like to thank Mike Sarvian from Digital Testing Services for the many hours spent helping to write and debug the test program, and Farzad Zarrinfar for coordinating LSI Logic's involvement in the experiment.

We would like to thank: Alida Mascitelli and Janis Stokes from Hughes Aircraft Co.; John Daane, Connie Jiang, Zarir Sarkari, and John Stahler from LSI Logic; Tushar Gheewala, Pat Hallinan, and Brad Plata from CrossCheck; Sassan Raissi from Digital Testing Services; Charles Hawkins, Tom Linnerooth, and Jerry Soden from Sandia National Laboratories; and LaNae Avra, Hong Hao, Raul Lopez, Siyad Ma, and Siegrid Munda from Stanford Center for Reliable Computing.

We would like to thank Sudhakar M. Reddy, Irith Pomeranz, A. K. Pramanick, and Lakshmi N. Reddy from the University of Iowa for their help in synthesizing the robustly path-delay-fault testable implementation of the RB control logic block and providing a delay fault test pattern generator.

We would like to thank the many people who contributed test sets or the use of ATPG tools for this experiment, including: Jacob Abraham, Debashis Bhattacharya, Rodolfo Betancourt, Susheel Chandra, Bob Duell, Gary Greenstein, Dong Ha, Uwe Huebner, Rathish Jayabharathi, Rohit Kapur, Peter Maxwell, Daniel Saab, L.-T. Wang, and John Waicukauski.

Major funding and support for this experiment has been provided by Hughes Aircraft Co., LSI Logic, and Digital Testing Services. This work was also supported in part by the Innovative Science and Technology Office of the Strategic Defense Initiative Organization and administered through the Office of Naval Research under Contract No. N00014-92-J-1782, and in part by the National Science Foundation under Grant No. MIP-9107760.

APPENDIX A: TEST CHIP DATA SHEET

Part Number	**L1A8063**
Masterslice	**LFT150067**
Usable Gates	**27k**
Used Gates	**25k**
Hard Macrocells	**None**
Custom Macrocells	**None**
I/O Count	**96**
Clock Rate	**100MHz**

A.1 Test Chip Input/Output Signals

Input Signals

Name	**Description**
CLK	Main Clock
RESET	Resets scan and mask states Required only for "scan-applied" data source mode.
EVSE	EValuator Shift Enable Sets evaluator scan chain to shift mode when high.
EVSI(4:0)	EValuator scan chain Shift In
PTEN	Post sample window Test ENable Uses external PTWIN as the post sample window (checking period) instead of the internally generated signal.
PTWIN	Post sample Test WINdow
DCENT	Direct sample CLock ENable True Uses CLK pulse width to determine sampling time.
ATSPEEDT	AT SPEED mode True When high, disables post-sample detection, samples data on every clock pulse.
MASKF	MASK failures False When low, causes failures to be ignored. Used for delay test patterns, to mask failures on the setup vector.
DIN(23:0)	Data IN 24 bit direct data input.
SRCMODE(1:0)	SouRCe MODE Selects direct (00), shifted vector pair (01), or pseudo-random (10) data source mode.
CUTENT(4:0)	CUT ENable True Enables the data output to individual CUT types. When low, all inputs to the corresponding CUT type is 0.

SRSEL(5:0)	Signature Register serial input SELect Selects one of the 48 serial inputs to the signature register. Puts the signature register in scan/test mode when 111XXX. Also controls the ring oscillator, which is enabled when all 6 SRSEL inputs are high.
SRMODE(1:0)	Signature Register MODE Selects the signature register configuration as 12x4(00), 16x3(01), 24x2(10), 48x1(11).
SRSERF	Signature Register SERial False When low, puts the signature register in serial mode.
SRSI	Signature Register Shift In
FCSI(4:0)	Fail Counter Shift In
FCSE	Fail Counter Shift Enable

Output Signals

Name	**Description**
CPOUT	main Clock Pulse OUT Provided as a timing reference.
DOUT23	source Data OUT, bit 23 For testing the data sources; also the output of the ring oscillator, when enabled.
SRSO	Signature Register Shift Out
PSWINOUT(4:0)	Post Sample WINdow OUT
EVSO(4:0)	EValuator Shift Out
CPASSF(4:0)	Clocked PASS False Combinational output of the sample register compare/OR tree, indicating pass(0)/fail(1) status of each CUT type.
PPASSF(4:0)	Post clock PASS False Combinational output of the post-sample stability checker OR tree
FCSO(4:0)	Fail Counter Shift Out
ANYFULLF	ANY fail counter FULL False Low when any of the “total failures” counter reaches maximum count.
SRCTSTO	SouRCe TeST Observation of CUT input gating.
PAROUT	PARametric tree OUTput Output of NAND tree connected to primary inputs.

CrossCheck Signals

Name	Description	I/O
TCK	Test Clock	I
TDI	Test Data In	I
TENA1	Test ENAble 1	I
TENA2	Test ENAble 2	I
TDO	Test Data Out	O (tri-state)

A.2 Test Chip Operating Modes

Data source Modes set with **SRCMODE(1:0).**

Clocking Modes set with **ATSPEEDT** and **DCENT** as follows:

Self-timed clock mode: **ATSPEEDT**=0, **DCENT**=0
Pulse clock mode: **ATSPEEDT**=0, **DCENT**=1
At-speed clock mode: **ATSPEEDT**=1, **DCENT**=X

Signature Register Modes set with **SRSEL(5:0)**, **SRMODE(1:0)**, and **SRSERF**.

A.3 Fail Counter Scan Chain Ordering

Figure A.1 shows the order of the fail counter scan chain. The chain is loaded with 0s before the test to reset the counters. After the test, bits b and d will be 0, a=1 if there has been a sampling error, and c=1 if there has been a stability checking error.

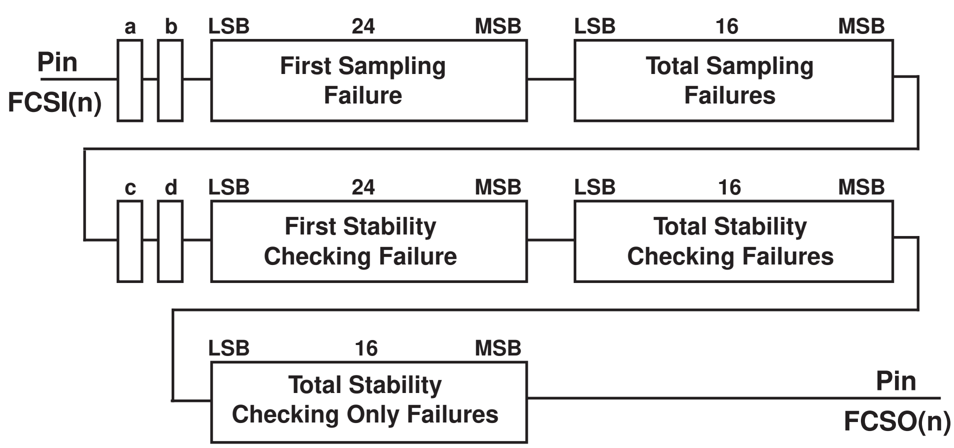


Figure A1. Scan Chain Connection for Failure Counters

APPENDIX B: TEST CONDITIONS

This Appendix lists the conditions for each of the tests applied to the Test Chip. The Support Circuitry and gross DC parametric (Stage 1) tests are covered first, followed by the CrossCheck (Stage 1 and 2) tests, and the CUT (Stage 2) tests.

B.1 Stage 1 Tests

Table B1. Continuity/Shorts

Test	V_{dd}	Conditions	Force	Measure
Continuity1	0	Signal pins floating	+700μA	0 to 1 V Pass
Continuity2	0	Signal pins floating	-700μA	-1 to 0 V Pass
Shorts	0	Signal pins floating	+700μA	< 50mV Fail

Table B2. Leakage (Precondition with Pattern Y)

Test	V_{dd}	Pins	Force	Measure
lkg1_hi	5.25	Signal Inputs	5.25V	-950 to 950 nA Pass
lkg1_lo	5.25	Signal Inputs	0V	-950 to 950 nA Pass
lkg2_hi	5.25	CrossCheck Inputs	5.25V	-2.9 to 2.9 μA Pass
lkg2_lo	5.25	CrossCheck Inputs	0V	-162 to -75 μA Pass

Table B3. Support Circuitry Functional Tests

Test	V_{dd}	Period	Strobe	Clock	V_{IH}	V_{OL}=V_{OH}
A_gross	4.75	2μs	1.9μs	NRZ	4.75	1.4
B_gross				↑400ns, ↓800ns		
C_gross				↑400ns, ↓800ns		
A_high	5.25	2μs	1.9μs	NRZ	5.25	1.5
B_high				↑400ns, ↓800ns		
C_high				↑400ns, ↓800ns		
A_ac	5.00	100ns	65.9ns	NRZ	4	1.5
B_ac		200ns	126.9ns	↑40ns, ↓80ns		
C_ac		200ns	127.5ns	↑40ns, ↓80ns		
AS_ac		25ns	24.5ns	↑9ns, ↓16ns		

Load on Output Pins: I_{OH} = 4.6mA, I_{OL} = 4.7mA

Test A - Stability Checker Test
Test B - Data Source Test
Test C - Counter and MISR Test
Test AS - At-speed test of Support Circuitry

Table B4. V_{IH}, V_{IL} Test

Test	Vdd	Period	Strobe	Clock	V_{IL}	V_{IH}	Load
Low voltage	4.75	2µs	1.9µs	NRZ	1.53V	3.23V	None
High Voltage	5.25	2µs	1.9µs	NRZ	1.68V	3.57V	None

Table B5. IDDQ Tests

Test	V_{dd}	Period	Settling	Threshold	Clock	V_{IH}	Load
Y_gross	5.25	2µs	10ms	100µA	↑100ns, ↓200ns	5.25	None
IDDQ-Sup1	5.25	2µs	10ms	500µA	NRZ	5.25	None
IDDQ-Sup2	5.25	2µs	10ms	500µA	↑400ns, ↓800ns	5.25	None
IDDQ-CUT	5.25	2µs	10ms	Record	↑100ns, ↓200ns	5.25	None

B.2 CrossCheck Tests

Table B6. CrossCheck Tests

Test	V_{dd}	Period/ Strobe	Clock	TCK	TDI TENA	TDO Strobe
Q_gross	4.75	4µs/3.9µs	NRZ	↓200ns,↑3µs	↑400ns	3.943µs
R_gross			↑800ns, ↓1.6µs			
X_gross			NRZ			
Q_high	5.25	4µs/3.9µs	NRZ	↓200ns,↑3µs	↑400ns	3.943µs
R_high			↑800ns, ↓1.6µs			
X_high			NRZ			
Q_ac	5.00	2µs/38.8ns	NRZ	↓100ns,↑1.5µs	↑200ns	1.972µs
R_ac		2µs/822.5ns	↑400ns, ↓800ns			
X_ac		2µs/59.2ns	NRZ			

Test X - CrossCheck Test Logic Test
Tests Q, R - CrossCheck test of Chip

B.3 CUT Tests

Table B7. CUT Functional Tests

Type	Timing	Speed	Abbrev.	Period	Strobe	Clock
MULT,RB	self	-	LF	100ns	95ns	↑20ns, ↓80ns
MULT	at-speed	worst	SPWM	53ns	50ns	↑10ns, ↓37ns
		fast	SPFM	39.7ns	37.5ns	↑5ns, ↓25ns
		slow	SPSM	79.5ns	75ns	↑15ns, ↓55ns
	external	worst	PUWM	100ns	95ns	↑20ns, ↓64ns
		fast	PUFM	100ns	95ns	↑20ns, ↓53ns
		slow	PUSM	120ns	115ns	↑20ns, ↓86ns
RB	at-speed	worst	SPWR	24ns	23ns	↑5ns, ↓20ns
		fast	SPFR			Don't Do
		slow	SPSR	36ns	34.5ns	↑7.5ns, ↓30ns
	external	worst	PUWR	100ns	95ns	↑20ns, ↓33ns
		fast	PUFR	100ns	95ns	↑20ns, ↓32,35n
		slow	PUSR	100ns	95ns	↑20ns, ↓39.5ns

Conditions: V_{dd} = 5V V_{IL}=0V, V_{IH} = 4.5V V_{OL}=V_{OH} = 1.5V
Load: I_{OH}=4.6mA, I_{OL}=4.7mA

Both multipliers have the same timing, and the three RB implementations have the same timing. The signature register test uses the same timing as SPWM.

Before running the Very-Low-Voltage CUT tests, the support circuitry tests A, B, C, and AS, are first run at the reduced voltage to ensure that the support circuitry is functional at low voltage.

Table B8. Very-Low-Voltage Support Circuitry Tests

Test	V_{dd}	Period	Strobe	Clock	V_{IH}	V_{OL}=V_{OH}
A_VLV	1.7	560ns	369ns	NRZ	1.7	0.85
B_VLV	1.7	1120ns	715ns	↑224ns, ↓448ns		
C_VLV	1.7	1120ns	715ns	↑224ns, ↓448ns		
AS_VLV	1.7	140ns	137ns	↑50.4ns, ↓89ns		

Load: NO LOAD

Table B9. Functional at Very Low Voltage

Type	Timing	Speed	Period	Strobe	Clock
MULT	external	PULM	560ns	538ns	↑112ns, ↓358ns
RB	external	PULR	560ns	538ns	↑112ns, ↓185ns

Load: NO LOAD

APPENDIX C: FAULT COVERAGE GRAPHS

The test length versus fault coverage of detectable single-stuck-faults for the pseudo-random and weighted random tests are shown in the figs below. The graph for the RB_STD circuit is in Fig. 15. Note that the vertical axes (Fault Coverage) do not start at 0%.

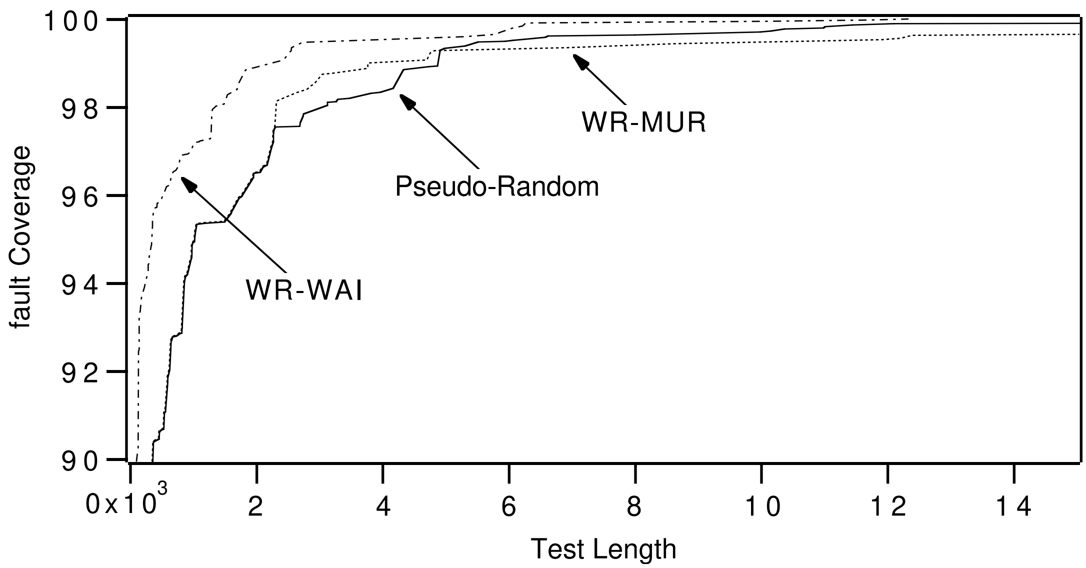


Figure C1. Graph of Fault Coverage versus Test Length for MULT12O12

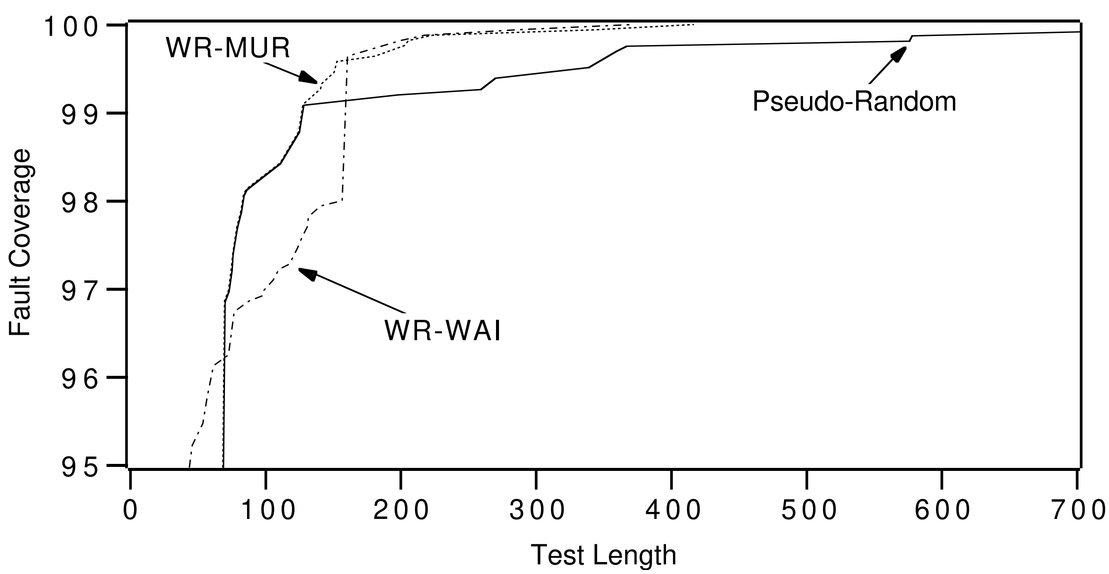


Figure C2. Graph of Fault Coverage versus Test Length for MULT6SQ

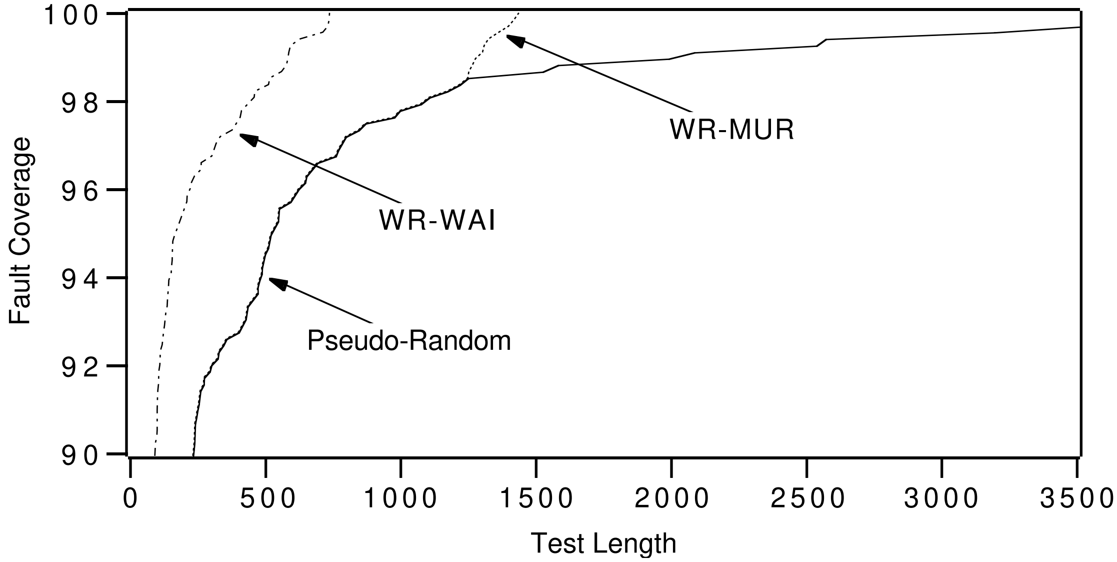


Figure C3. Graph of Fault Coverage versus Test Length for RB_SIMPLE

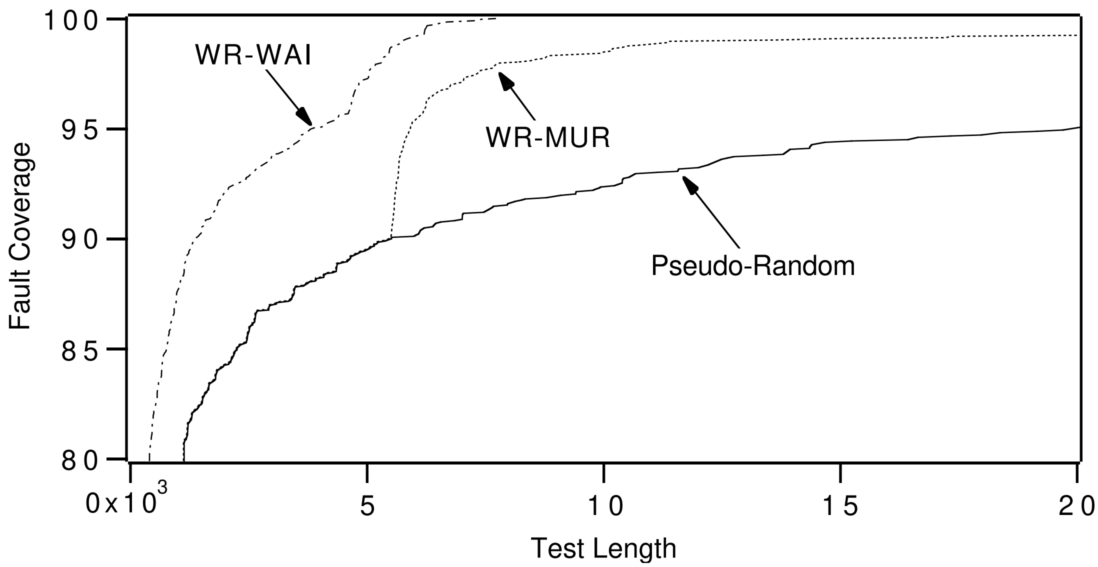


Figure C4. Graph of Fault Coverage versus Test Length for RB_ROBUST

REFERENCES

[Brayton 84] Brayton, R.K., G.D. Hachtel, C. McMullen, and, A.L. Sangiovanni-Vincentelli, *Logic Minimization Algorithms for VLSI Synthesis*, Boston, Kluwer Academic Publishers, 1984.

[Brayton 87] Brayton, R.K., R. Rudell, A.L. Sangiovanni-Vincentelli, and A.R. Wang, “MIS: A Multiple-Level Logic Optimization System,” *IEEE Trans. CAD*, Vol. 6, pp. 1062-1081, Nov. 1987.

[Das 90] Das, D. V., et. al., “An Experimental Study on Reject Ratio Prediction for VLSI Circuits: Kokomo Revisited,” *Proc. 1990 Int. Test Conf.*, Washington, DC, pp. 712-720, Oct. 26-30, 1990.

[Elo 90] Elo, R. B., “An Empirical Relationship Between Test Transparency and Fault Coverage,” *Proc. 1990 Int. Test Conf.*, Washington. DC, pp. 1006-1011, Oct. 26-30, 1990.

[Franco 91] Franco, P., and E.J. McCluskey, “Delay Testing of Digital Circuits by Output Waveform Analysis,” *Proc. 1991 Int. Test Conf.*, Nashville, TN, pp. 798-807, Oct. 26-30, 1991.

[Gayle 93] Gayle, R., “The Cost of Quality: Reducing ASIC Defects with I_{DDQ}, At-Speed Testing, and Increased Fault Coverage,” *Proc. 1993 Int. Test. Conf.*, Baltimore, MD, pp. 285-292, Oct. 17-21, 1993.

[Gheewala 88] Gheewala, T.R., “Grid-Based, ‘Cross-Check’ Test Structure for Testing Integrated Circuits,” *U.S. Patent Number* 4,749,947, 1988.

[Gheewala 89] Gheewala, T.R., “CrossCheck: A Cell Based VLSI Testability Solution,” *Proc. 26th Des. Autom. Conf.*, Las Vegas, NV, pp. 706-709, 1989.

[Hao 93] Hao, H. and E.J. McCluskey, “Very-Low-Voltage Testing for Weak CMOS Logic IC,” *Proc. 1993 Int. Test Conf.*, Baltimore, MD, pp. 275-284, Oct. 17-21, 1993.

[Jha 92] Jha, N.K., I. Pomeranz, S.M. Reddy, and R.J. Miller, “Synthesis of Multi-Level Combinational Circuits for Complete Robust Path Delay Fault Testability,” *Digest, 1992 IEEE Workshop on Fault-Tolerant Parallel and Distributed Systems*, Amherst, MA, pp. 280-287, July 6-7, 1992.

[Lee 91] Lee, H.K., and D. S. Ha, “An Efficient Forward Fault Fault Simulation Algorithm Based on the Parallel Pattern Single Fault Propagation,” *Proc. 1991 Int. Test Conf.*, Nashville, TN, pp. 946-955, Oct. 26-30, 1991.

[Lin 87] Lin C.J., and S.M Reddy, “On Delay Fault Testing in Logic Circuits,” *IEEE Trans. CAD*, Vol. CAD-6, No. 5, pp. 694-703, Sep. 1987.

[LSI 92] *LFT150K FasTest Array Series Databook*, LSI Logic, Milpitas, CA, Apr. 1992.

[Maxwell 91] Maxwell, P.C., R. C. Aitken, V. Johansen, and I. Chiang, “The Effect of Different Test Sets on Quality Level Prediction: When is 80% better than 90%,” *Proc. 1991 Int. Test Conf.*, Nashville, TN, pp. 358-364, Oct. 26-30, 1991.

[Maxwell 92] Maxwell, P.C., R. C. Aitken, V. Johansen, and I. Chiang, “The Effectiveness of I_{DDQ}, Functional and Scan Tests: How Many Fault Coverages Do We Need?,” *Proc. 1992 Int. Test Conf.*, Baltimore, MD, pp. 168-177, Sep. 20-24, 1992.

[Muradali 90] Muradali, F., V. K. Agarwal, and B. Nadeau-Dostie, "A New Procedure for Weighted Random Built-In Self Test," *Proc. 1990 Int. Test Conf.*, Washington, DC, pp. 660-669, Sep. 10-12, 1990.

[Pancholy 90] Pancholy, A., J. Rajski, and L. J. McNaughton, "Empirical Failure Analysis and Validation of Fault Models in CMOS VLSI," *Proc. 1990 Int. Test Conf.*, Washington, DC, pp. 938-946, Sep. 10-12, 1990.

[Pancholy 92] Pancholy, A., J. Rajski, and L. J. McNaughton, "Empirical Failure Analysis and Validation of Fault Models in CMOS VLSI Circuits," *IEEE Design & Test*, pp. 72-83, Mar. 1992.

[Perry 92] Perry, R., "I_{DDQ} Testing in CMOS Digital ASIC's - Putting it All Together," *Proc. 1992 Int. Test Conf.*, Baltimore, MD, pp. 151-157, Sep. 20-24, 1992.

[Pramanick 90] Pramanick, A.K., and S.M. Reddy, "On the Design of Path Delay Fault Testable Combinational Circuits," *Proc. 1990 Int. Fault-Tolerant Computing Symp.*, Newcastle upon Tyne, England, pp. 374-381, June 26-28, 1990.

[Sawada 92] Sawada, K. and S. Kayano, "An Evaluation of I_{DDQ} Versus Conventional Testing for CMOS Sea-of-Gate IC's," *Proc. 1992 Int. Test Conf.*, Baltimore, MD, pp. 158-167, Sep. 20-24, 1992.

[Schiessler 93] Schiessler, G., C. Spivak, and S. Davidson, "IDDQ Test Results on a Digital CMOS ASIC," *Proc. 1993 Custom Integrated Circuits Conf.*, pp. 26.4.1-26.4.4, 1993.

[Sentovich 92] Sentovich, E.M., J. K. Singh, C. Moon, H. Savoj, R. K.Brayton, and A. Sangiovanni-Vincentelli, "Sequential Circuit Design Using Synthesis and Optimization," *Int. Conf. on Comput. Des.,* Los Alamitos, CA, USA, pp. 328-333, 1992.

[Synopsys 91] Synopsys, *Design Compiler Reference Manual*, Version 2.0, May 1991.

[Velazco 90] Velazco, R., C. Bellon, and B. Martinet, "Failure Coverage of Functional Methods: A Comparative Experimental Evaluation," *Proc. 1990 Int. Test Conf.*, Washington, DC, pp. 1012-1017, Sep. 10-12, 1990.

[Waicukauski 89] Waicukauski, J. A., et al, "A method for generating weighted random test patterns," *IBM J. Res. Develop.*, Vol. 33, No. 2, Mar. 1989.

[Wiscombe 93] Wiscombe, P.C., "A Comparison of Stuck-At Fault Coverage and I_{DDQ} Testing on Defect Levels," *Proc. 1993 Int. Test. Conf.*, Baltimore, MD, pp. 293-299, Oct. 17-21, 1993.

[Wu 92] Wu, Y., and A. Ivanov, "Fuzzy Multiple Signature Analysis for BIST," *Technical Report*, University of British Columbia, 1992.

[Zarrinfar 93] Zarrinfar, F., "The four Major Forces That Drive ASIC Testability," *Design & Test Exposium,* Anaheim, CA, Jan. 11-14, 1993.