Fault Detection and Diagnosis –An Analysis

Akilandeswari R, Department of Electronics, Sindhi College
Rashmi B, Department of Electronics, Sindhi College

Abstract:

With increased level of integration in VLSI chips, the complexity of testing is also increasing. Testing costs are also high. Testing cost of the chip depends mainly on the time required to test it. So a test set that is small in size has to be used. Another important aspect is Failure Analysis, the process of determining the cause of failure of a chip. It is important to determine the cause of failure as this can lead to improvement in the design of the chip and increase the yield of the chips. Fault diagnosis is the first step in failure analysis, which gives a list of defects. Fault diagnosis reduces the area of the chip on which physical examination needs to be done to locate defects.

Keywords: *CUT, CUD, test pattern, fault analysis, fault diagnosis*

Introduction

Types of Circuits

This paper address an analysis of minimal test set generation for testing and diagnosis in combinational logic circuits. In Sequential circuits it is accomplished by implementing scan-based design, in which all flip-flops in the circuit are modified and are stitched together to from one or more scan chains. During scan based testing, input data is scanned into the flip-flops via the scan chains and other input data is applied to the input pins of the circuit. Once these inputs are applied and the circuit has stabilized its response, the circuit is clocked to capture the results back into the flip-flops, and the data values at the output pins of the circuit are noted. The combination of values at the primary outputs and the values scanned out of the scan chains make up the response of the circuit to the test. Scan based testing is used for fault diagnosis.

Testing/Diagnosis Process

The circuit is referred to as Circuit under Test (CUT) during testing, and during diagnosis it is referred as Circuit under Diagnosis (CUD). Binary test vectors are applied and the response of the CUT/CUD to a test vector is compared with the stored response. A difference between the stored and observed responses indicates a bad circuit.

The input data to the CUT/CUD is referred to as the test pattern or test vector. A collection of test vectors of the circuit is called a test set. The two tests are fault detection tests and fault diagnosis tests. Fault detection tests are used to test the circuit under test is faulty or fault free. This test does not give any information on the type or location of the fault. A diagnostic test is applied after a circuit has failed the testing process and it identifies the cause of failure of the circuit.

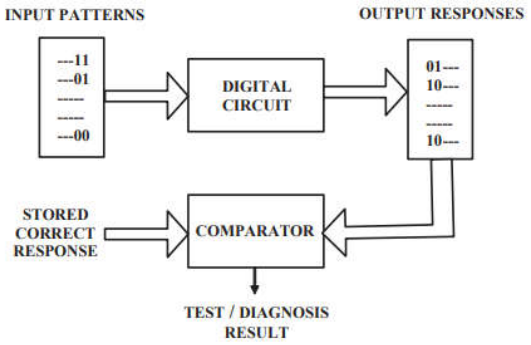


Figure 1: Testing/Diagnosis process

Testing of Logic Circuits

- Fault Models
- Test Generation and Coverage
- Fault Detection
- Design for Test

Fault Models

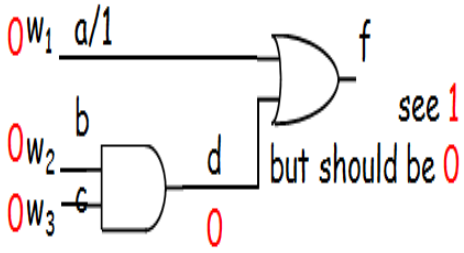
Stuck-At Model

The selected wires either gate input or output are “stuck at” logic value 0 or 1.There may be

some kind of fabrication flaws that short circuit wires to ground or power, or broken wires that are floating

- Wire w stuck-at-0: w/0
- Wire w stuck-at-1: w/1

Only one fault at a time is assumed even though in real circuits multiple simultaneous faults are possible and can mask each other



A test case is generated to determine if a is stuck at 1.Try with 000,if a is stuck at 1, expect to see f = 0, but 1 will be the output instead of 0.

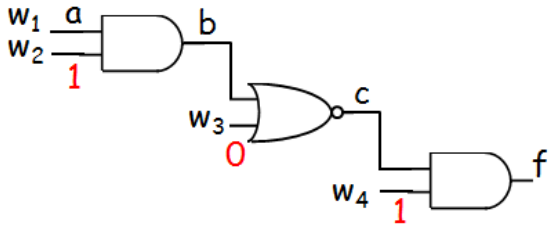
Test	Fault Detected									
w1 w2 w3	a/0	a/1	b/0	b/1	c/0	c/1	d/0	d/1	f/0	f/1
000		X						X		X
001		X		X				X		X
010		X				X		X		X
011			X		X		X		X	
100									X	
101	X								X	
110	X								X	
111	X								X	

Test Set

In general, n-input circuits require much less than 2^n test inputs to cover all possible stuck-at-faults in the circuit.

Path Sensitization

Testing is complex for wire-at-time. Focusing on wiring paths, enables multi-wire testing at the same time. A path is activated so that changes in signal propagating along the path affect the output. To observe the fault path idea can be used to "propagate" a fault to the output. To drive the internal wire to a known value inputs and intermediate values are set so as to pass an internal wire to the output while setting inputs to drive that internal wire to a known value. If propagated value is not as expected, then a fault on the isolated wire has to be found.



To activate the path, inputs are set so that w_1 can influence f

E.g., $w_2 = 1$, $w_3 = 0$, $w_4 = 1$

AND gates: if one input is at 1 passes the other input

NOR gates: if one input is at 0 inverts the other input

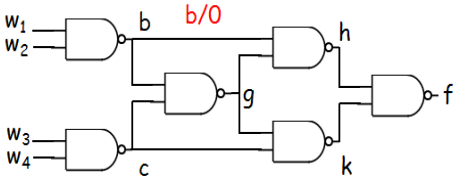
To test:

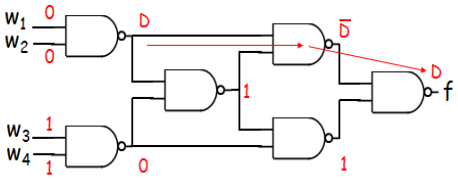
w_1 is set to 1 should generate f = 0

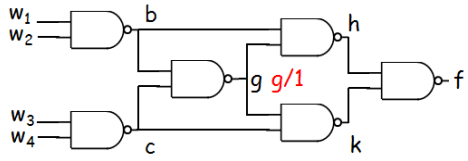
if path is ok Faults a/0, b/0, c/1 can cause f = 1

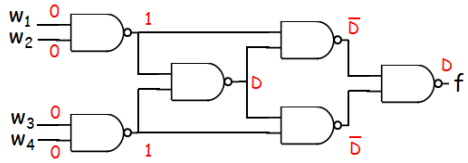
w_1 is set to 0 should generate f = 1

if path ok Faults a/1, b/1, c/0 will cause f = 0

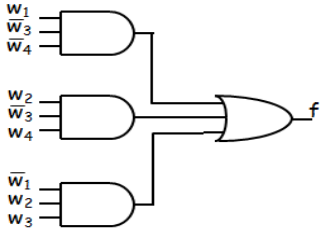








Tree Structured Circuits

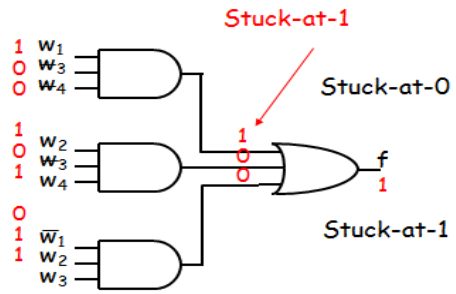


To test inputs stuck-at-0 at particular AND gate

- To generate AND output of zero inputs are set at other gates
- Inputs are forced at selected gate to generate a one
- If the output is 1 then circuit is ok, else fault

To test inputs stuck-at-1 at particular AND gate

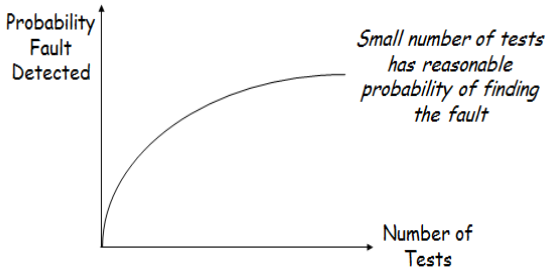
- To test to 0 the inputs are driven, rest of inputs driven to 1
- Other gates are driven with inputs that force gates to 0
- If the output is 0 then fault, else the circuit is OK



	Product Term									Test			
	w_1	$\bar{w}_3$	$\bar{w}_4$	w_2	$\bar{w}_3$	w_4	$\bar{w}_1$	w_2	w_3	w_1	w_2	w_3	w_4
1	1	1	1	0	1	0	0	0	0	0	0	0	0
2	0	1	0	1	1	1	1	1	0	1	1	1	0
3	0	0	0	1	0	1	1	1	1	1	1	1	1
4	0	1	1	1	1	0	1	1	0	0	1	1	0
5	1	0	1	1	0	0	0	1	1	0	0	1	1
6	1	1	0	0	1	1	0	0	0	1	0	0	0
7	1	0	0	1	0	1	0	1	1	1	0	1	1
8	0	0	0	0	0	1	1	0	1	1	1	0	1

Random Testing

Generate random input patterns to distinguish between the functions i.e, faulty or correct.

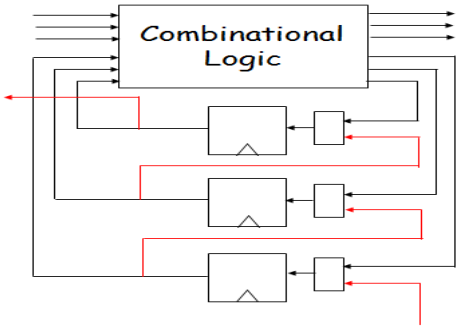


Sequential Testing

In sequential testing due to the presence of memory state inside flip-flops, it is difficult to employ the same methods as with combinational logic. An alternative approach is design for test. In scan path technique flip flop (FF) inputs passes through the multiplexer stages to allow them to be used in normal mode as well as a special test shift register mode.

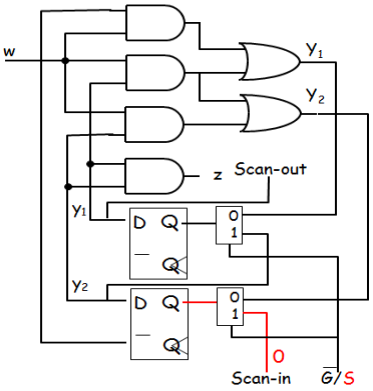
Scan Path Technique

Configure FFs into shift register mode. Test pattern in 0s and 1s are scanned. In scan path non-state inputs can also be used. Run system for one clock cycle in "normal" mode and next state is captured in scan path. Then return to shift register mode and shift out the captured state and outputs.

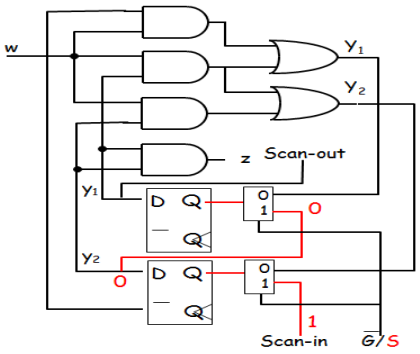


Scan Path Example

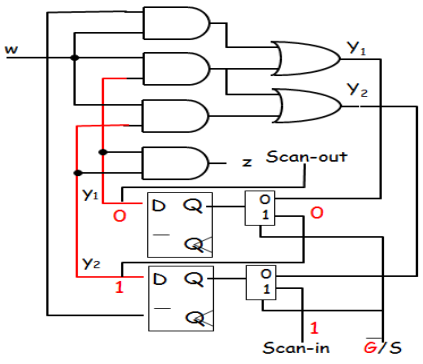
w is input.y1,y2 are outputs of combinational logic. Test vector is 001 and 00 is scanned into y1, y2 FFs



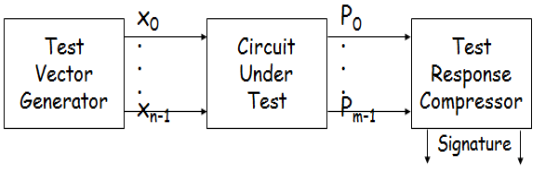
w is input.y1,y2 are outputs of combinational logic. Test vector is 001. 01 is scanned into y1, y2 FFs



w,y1,y2 test vector is 001.Scan 01 into y1, y2 FFs. Normally w=0.Output z=0, Y1=0, Y2=0. Observe z directly. Scan out Y1, Y2

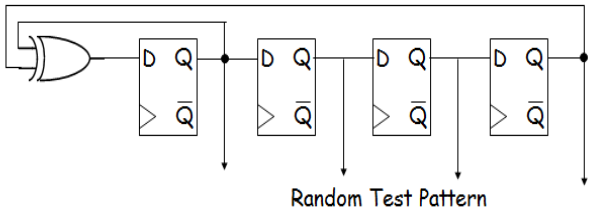


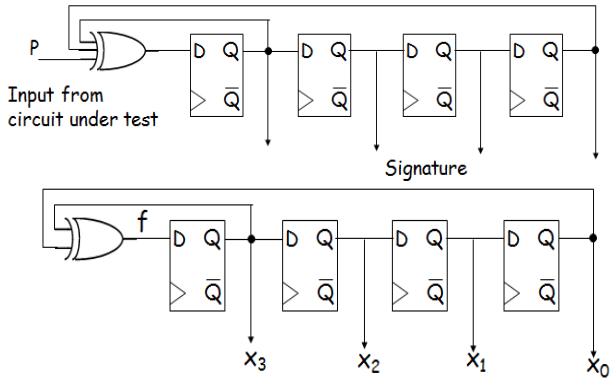
Build In Self-Test

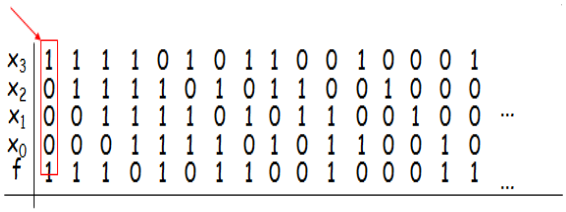


To generate test vector, pseudorandom tests are done with a feedback shift register. Seed generates a sequence of test patterns. Outputs are combined using the same technique. It generates a unique signature that can be checked to determine if the circuit is correct

Linear Feedback Shift Register

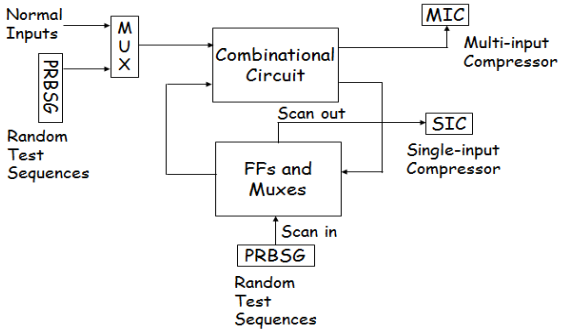






Starting with the pattern 1000 generate 15 different patterns in sequence and then repeats.

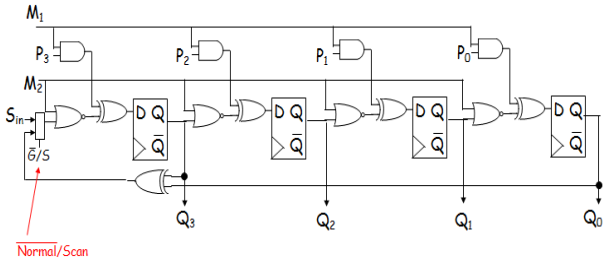
Complete Self-Test system



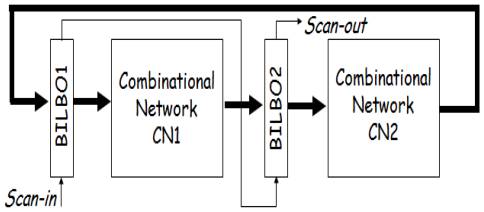
Built-in Logic Block Observer

In Build-in Logic Block server test generation and compression is done in a single circuit.

- $M_1, M_2 = 11$: Regular mode
- $M_1, M_2 = 00$: Shift register mode
- $M_1, M_2 = 10$: Signature generation mode
- $M_1, M_2 = 01$: Reset mode



Bilbo Architecture



Initial pattern is scanned in Bilbo 1, reset FFs in Bilbo2. Use Bilbo1 as PRBS generator for given number of clock cycles and use Bilbo2 to produce signature. Scan out Bilbo2 and compare signature; Scan in initial test pattern for CN2; Reset the FFs in Bilbo1. Use Bilbo2 as PRBS generator for a given number of clock cycles and use Bilbo1 to produce signature. Scan out Bilbo1 and compare signature

Fault Simulation

The process of computing the response of a faulty circuit to given input stimuli is called Fault Simulation. Fault simulation is performed by fault simulator. The operation of a fault simulator is as follows: A circuit, a test input and a fault is given, the fault simulator inserts the fault into the circuit and computes its output response for the test input. It computes the good circuit response to the same test input. Finally it detects the fault as if there is mismatch between the good and faulty responses. For a test set and a fault list the fault simulator gives the fault coverage of the test set. Fault coverage is defined as ratio of Number of faults detected to number of faults in initial fault list. For test generation fault simulator is used with Automatic Test Pattern Generator (ATPG).

Several algorithms are used for fault simulation, the simplest being the serial fault simulation. One fault at a time is performed in Fault simulation. When one fault is detected, the simulation is stopped and a new simulation is started for another fault. Fault simulation method is simple but time consuming.

A fault simulation method, which simulates more than one fault in one pass is called parallel fault simulation. Bit-parallelism of logical operations is used in this technique. The parallel fault simulator can simulate a maximum of w − 1 faults in one pass, where w is the machine word size. So, a parallel fault simulator runs w − 1 times faster than a serial fault simulator.

Test Generation

Automatic test pattern generation is the process of generating input patterns to test a circuit, which is described strictly with a logic-level net list. These programs usually operate with a fault generator program, which creates the collapsed fault list. Test generation approaches can be classified into three categories: exhaustive, random and deterministic.

In the exhaustive approach, for an n-input circuit, 2^n input patterns are generated. The exhaustive test set guarantees 100% fault coverage. It is evident that this approach is feasible only for circuits with very small number of primary inputs.

Random test generation is a simple approach where the input patterns are generated randomly. For every randomly generated pattern fault simulation is performed. If it detects new faults a pattern is selected. In a typical test generation process, random patterns are used for fault coverage, after that deterministic test pattern generation is employed to achieve the remaining coverage.

In Deterministic Automatic Test Pattern Generator (D-ATPG) algorithms a fault is injected into a circuit, and the values for the primary inputs of the circuit that would activate that fault and propagate its effect to the circuit output is determined. In deterministic test generation, the search for a solution involves a decision process for selecting an input vector from the set of partial solutions using an algorithmic procedure known as backtracking. All previously assigned signal values are recorded in backtracking, so that the search process is able to avoid those signal assignments that are inconsistent with the test requirement. The exhaustive nature of the search causes the worst-case complexity to be exponential in number of signals in the circuit. To minimize the total time required, a typical test generation program is allowed to do only a limited search in the number of trials or backtracks, or the CPU time.

The most widely used deterministic automatic test pattern generation algorithms are: D-algorithm, PODEM (Path-Oriented Decision Making) and FAN (Fan-out-Oriented Test Generator). The faults left undetected after the D-ATPG phase are either redundant faults for which no test exists, or aborted faults that could not be detected due to CPU time limit.

Failure Analysis

Testing of fabricated chips prevents the shipment of defective parts, but improving the production quality of the chips depends on effective failure analysis. A good quality production process means higher yield, i.e., more usable die. An IC product goes through two manufacturing stages: (1) prototype stage, and (2) high-volume manufacturing stage.

During prototype stage, a small number of sample chips are produced to verify the functionality and timing of the design. The chips may fail badly due to design bugs or manufacturing imperfections. All the issues that makes the chip to fail must be addressed so that these do not recur when the design goes into mass production. Failure analysis is done for yield improvement. After the prototype stage, the product is ready for high-volume production. In this stage there could be fluctuations in the yield mainly due to manufacturing errors. Continuous yield monitoring is necessary from time to time to respond to unexpected low-yield situations.

Some of the methods of failure analysis consists of etching away certain layers, imaging the silicon surface by scanning electron microscopy (SEM) or focused ion beam (FIB) systems, particle beams, etc. The analysis performed directly on the defective chip is called physical analysis. Physical analysis process is often laborious and time consuming with millions of transistors and several layers of metals. It is the job of fault diagnosis to guide the physical analysis process by providing the suspected defect locations.

Fault Diagnosis

Fault diagnosis is the initial step in failure analysis which by logical analysis gives a list of likely defect sites or regions. Fault diagnosis reduces the area of the chip on which physical examination needs to be done to locate defects. It involves applying tests to failed chips and analyzing the test results to determine the nature of the fault. A test set used for the purpose of diagnosis is referred to as a diagnostic test set. The test sets are constructed to identify a single fault. This characteristic of a diagnostic test set is called its diagnostic resolution. The highest number of fault candidates reported for a test in the diagnostic test set is referred to as its diagnostic resolution. Fault diagnosis leads to physical analysis. If a wrong location is predicted then the actual defect site could be damaged during the physical inspection process of the predicted site.

Conclusion

Fault models are an approach for determining how to develop a test pattern sequence. It can assume single fault. Scan path is a technique for applying test inputs within the system, usually for

asserting state. It is a technique for getting internal state to edges of circuit for observation. Built-in Test is founded on the approach of random testing. To generate pseudo random sequences; compute signature; determine if signature generated is same as signature of a correctly working circuitry. For future scope primal dual method is used for minimized test and lower complexity.

References

[1] M. Abramovici and M. A. Breuer, “Multiple Fault Diagnosis in Combinational Circuits Based on an Effect-Cause Analysis,” IEEE Transactions on Computing, vol. C-29, no. 6, pp.451–460, June 1980.
[2] V. D. Agrawal and P. Agrawal, “An Automatic Test Generation System for Illiac IV Logic Boards,” IEEE Trans. on Computers, vol. C-21, no. 9, pp. 1015–1017, Sept. 1972.
[3] V. D. Agrawal, D. H. Baik, Y. C. Kim, and K. K. Saluja, “Exclusive Test and its Applications to Fault Diagnosis,” in Proc. International Conf. on VLSI Design, 2003, pp. 143–148.
[4] V. D. Agrawal, A. V. S. S. Prasad, and M. V. Atre, “Fault Collapsing via Functional Dominance,” in Proc. International Test Conf., 2003, pp. 274–280.
[5] S. Mukherajee, "Architectural Design for Soft Errors," in Morgan Kaufmann Publishers, 2008.