DELAY TEST *

MEFIRE Mama (s001067)   HE Jie (s002182)

Abstract

Test of digital circuits implies verification of both functional and temporal behaviour of the circuits manufactured. Test used for stuck-at fault may avail insufficient to guarantee an acceptable quality level, as some defects or perhaps random process variations will affect the dynamic, but not the steady state behaviour of the system. Therefore two delay fault models are given in this paper. With emphasis on path delay fault model we discuss the implementation of parallel test generation, random sampling times for delay fault testing and delay verification test sets in delay fault models.

1. Introduction

Correct operation of a logic circuit requires testing for both DC or static behavior as well as the temporal behavior of the circuit. Under normal operating conditions at the specified clock speed, signal transitions initiated at the input of the circuit are expected to arrive at the output in appropriate time to be latched properly at the output latches. But due to process variations and stray capacitances, an incorrect value may be latched at the output, caused by the path delay falling outside the permitted tolerance. Faults can also occur due to propagation delays falling below a specified limit. Faults due to variation of propagation delays beyond the specified tolerance are called delay faults. The common type of delay fault model are the gate delay fault model and the path delay fault model. In this paper with use mainly the path delay fault. We shall assume that delay faults are caused by propagation delays of paths exceeding the expected value (called long path delay faults). However, the problem of generating tests for delay faults caused by delays of paths falling below their expected value (called short path delay faults) is just the dual of the corresponding problem for long path delay faults. The first section of the paper deals with the description and comparison of the gate and path delay fault models. In the second section we give the principles of test pattern generation for path delay faults and an explanation of how a parallel test can be implemented. The next section treats the use of random sampling times for delay fault testing and its impacts on test hardware. Before concluding we give a definition and the concept of ”delay verification test sets” along with a way of implementing verifiable circuits.

2. Description and comparison of the gate delay and path delay faults models

The goal of delay fault testing is to be able to detect every fault that may bring about late or early transition at some output of the circuit during normal operation. This section describes the two most common delay faults model (gate model and path model) used in delay fault testing. At the end of the section, the two models are compared. In the description we consider a case of a combinational circuit, although this can be extended to a structured sequential circuits.

2.1. Gate Delay fault Model Description

Gate delay faults model those delay defects occuring at inputs or outputs of a gate, which cause the gate delay to be outside its specified range [2]. They assum that delays through logic gates are known with some precision. The characteristics (size and location) of lilkely delay faults are also known. The delays through a gate are represented by intervals in this model. A fault is an added delay of certain size, say δ, in the propagation of a rising or falling transition from the gate input to its output. The set of faults considered includes numerical delay information. Before describing the model, we would like to give an example of a delay fault that may occur in a gate.

Let consider a case of an OR gate shown in figure 2.1. a and b are the inputs to the gate and y is the output and assum that for this cell the resulting delay is the one in which

*This paper is handed in as an assignment answer in lecture course *02208 Test of Integrated Circuits (2001)* at the Department of Informatics and Mathematical Modelling (IMM), Technical University of Denmark.

transition is applied to a single input (input a in this example 2.1-a). Let d be the delay needed to have the output y for this transition. The delay may be halve of d if both the inputs are to transion simultanuously 2.1-b. If the manufaturing tolerance for this cell is less than d/2 then we may experience a delay fault or error at the output of the gate, since this output will be present a little bit earlier than predictated.

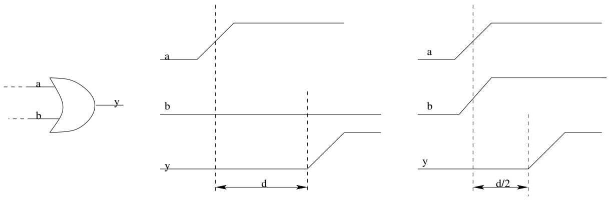

Figure 1. path-specific

The model for gate delay is shown in figure 2. It presents a gate with delay and the model of gate.

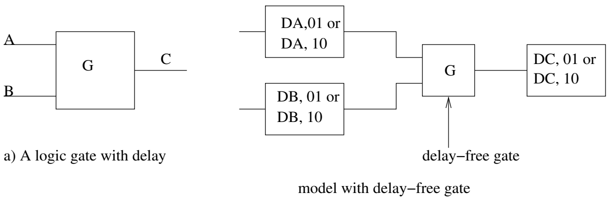

Figure 2. Gate delay model with gate delay-free, each input or output delay is model as a definite value

The appraoch used in [4] is used in describing the model, We assum that each rising (falling) output delay of the gate has a definite interval and any rising or falling delay out of this interval is said to be faulty.

Let $Dfall$ and $Drise$ be the nominal values of the falling and rising delay respectively and σ the manufactring tolerance of the gate. We define the upper bound (lower bound) of the rising delay Ubr (Lbr) as

$$Ubr = Drise + \sigma \tag{1}$$

$$Lbr = Drise - \sigma \tag{2}$$

Likewise the upper bound (Ubf) and lower bound (Lbf) of the fall delay are given by:

$$Ubf = Dfall + \sigma \tag{3}$$

$$Lbr = Dfall - \sigma \tag{4}$$

We define the set of all allowable rising delay (R) and the set of all allowable falling delay (F) by:
R = (Ubr, Lbr) and
F = (Ubf, Lbf).
For a given signal s, its falling delay and rising delay are computed or observed. A rising delay (falling delay) is said to be faulty if it does not belong to R (F).

2.2. Description of the Path Delay fault Model

Recently the path delay fault model has received greater attention than the gate delay fault model. A path delay fault occurs along a path P of a circuit if the delay along P falls outside its specified limits(either upper or lower bounds)[1]. This distributed defects affect an entire path. For each physical path P_p, connecting a prime input to a primary output of the circuit, there are two corresponding delay paths. The rising path (falling path) is the path traversed by a transition that is initiated as a rising (falling) transition at the input of path P and changes the direction of transition whenever it passes through an inverting gate.

The approach used in [2] is also used here, with some modification to describe the path delay fault model. Figure 3 depicts the modified delay of [2]. In this model we assum that input lines or wires are represented by their respective delay. $DA(r, f)$ represents the rising or the falling delay or an interval of time as described in the gate delay model. Figure 3-c represents the graph model where the delay represent the edges and the delay-free gate as a node. Therefore a combinational circuit can be represent as an acyclic graph $G(D,N)$, where D is the set of all the edges in the graph and N the set of all the nodes, having primary inputs as initial nodes and primary output as final nodes. Figure 4 depicts the general representation of the graph model. The primary inputs node are labeled i_n, the primary output o_m and the internal nodes g_j. A physical path P_p consists of all the edges $d_k(r, f)$ (r (f) represents rising (falling) transition in the path) linking one primary input to one primary output.
The propagation delay dP_p of P_p (time it takes a transition to propagate through the path P_p) given by:

$$dP_p = \sum d_k(r, f) \tag{5}$$

In [3] the analysis method given by (Smith - 1985) is described. The combinational circuit under test is embedded between a block of input and output latches, with all the latches assumed to be without gliches. At the times t_0 and t_1, two inputs pattern (V_1 and V_2) are applied to the inputs of the circuit under test, and the output is sampled at time t_2. If we assum that $dn = t_2 - t_1$ is the nominal delay of the path dP_p, therefore the path dP_p is said to be faulty if dP_p

is greater than dn.

A fanout stem is represented by a delay-free gate having one input and multiple outputs (figure 5). The general form of the path delay is depicted in figure 4. The primary inputs and primary outputs are defined as a pecial type of node (delay-free gate), the former is used to initiate the signal and the latter is used for observation or sampling.

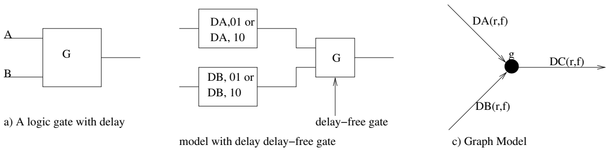

Figure 3. Transformation of the wire and gate delay to the graph delay model

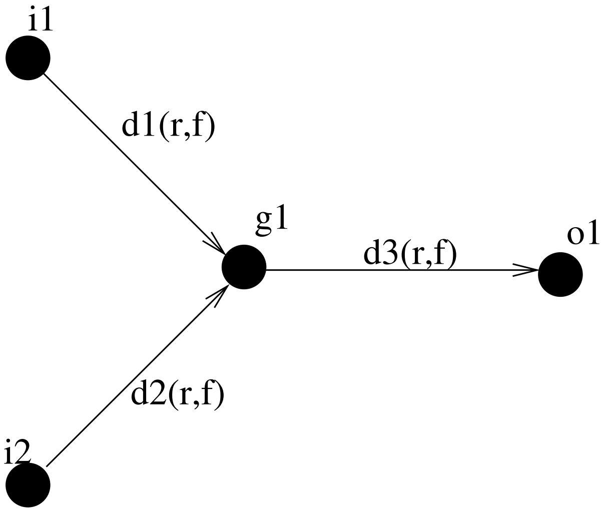

Figure 4. General form of the graph model

———-

2.3. Comparison of the two faults models

To compare the two models, their advantages as well as their limitations[6].

2.3.1 Path delay fault model

Advantages:
Distributed failure are considered. Detection can be guaranteed by robust test.
Limitations:

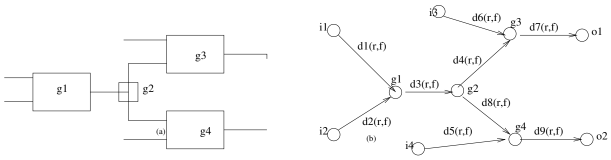

Figure 5. General form of the graph model. (a)- Combinational model, (b) Graph momdel: In this model, the primary inputs are nodes labeled i_n, the primary ouputs o_m and the internal delay-free gates are labeled g_k. The fanout stem is also a type of gate

Impossible to enumerate all possible paths. Many faults may not have robust test.

2.3.2 Gate delay fault model

Advantages:
All gates can be modeled. Detection with a threshold σ guaranteed detection of all fault size greater than σ.
Limitations:
Distributed failure not considered. Exact defect size not possible.

It should be noted that path delay model is widely used because it can cover some gate delay faults.

3. Test Pattern Generation for Path Delay Fault

3.1. Principles

To detect a path delay fault, a two-pattern test $< V_1, V_2 >$ is applied on the circuit under test at time t_0 and t_1, and the output is sampled at time t_2, with the assumption that there is no glitches at the primary inputs. The two-pattern test vector creates and propagates appropriate signal transitions (rising or falling) along the path to the be tested. The method is shown in figure 6.
Generally two types of test are possible for path delay fault testing. These are robust test and nonrobust test. In the model presented, the transition are initiated by the primary input. A path delay fault is said to be robust detectable if and only if its detection is independent of all other delay fault in the circuit and the fault is nonrobust detectable if delays of arbitrary gate may invalidate its detection.

For a structure path P_p defined in the previous section, we have the following definition.

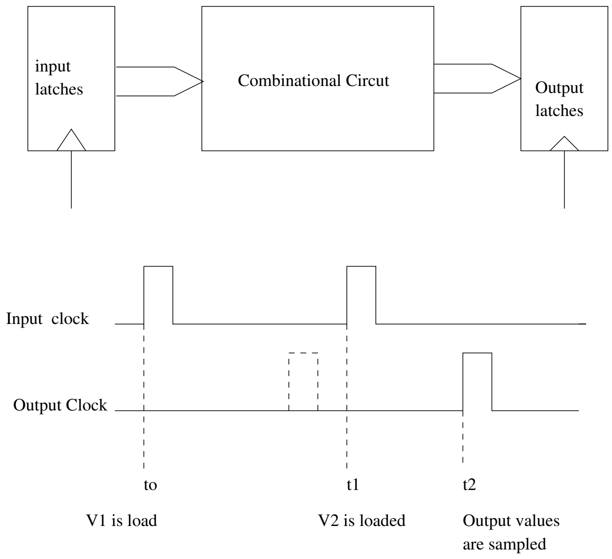

Figure 6. Method used in path delay fault pattern generation

Definition 1: A two-pattern test $< V_1,V_2 >$ is said to be a robust delay test for transition path P_p if and only if, when P_p is faulty and test $< V_1,V_2 >$ is applied, the circuit output is different from the expected state at sampling time t_2, independent of the delays along gate input leads not on path P_p [2]. (The expected state of the output at time t_2 is the state of the path output corresponding to input V_2.)

Definition 2: An input α of a gate G is called off-path sensitizing input of path P_p or simply off-path sensitizing input if and only if G is on path P_p and α is not on path P_p.

We use the five-valued logic proposed in [2]. For the sake of clarity, these logics are again reproduced here. The logic values are $\{S0,S1,U0,U1,XX\}$ and they are shown in figure 7and the implication table according to their respective logic gate is given in figure 8. S0 and S1 represent the signals with a final value 0 and 1 respectively, whereas U0 and U1 represent also the signals of final values 0 and 1 respectively with some probablity of hazard or transient. U0 and U1 may also represent the signal transition from 0 to 1 and 1 to 0 signal transition. XX represents a signal having unspecified initial and final value.

For robust tests, the propagated signals on the wires or leads of the tested path are U0 or U1, changing from U0 to U1 and U1 to U0 when going through an inverting gate.

Theorem: A two-pattern test $< V_1,V_2 >$ is a robust test delay test of path P_p if and only :

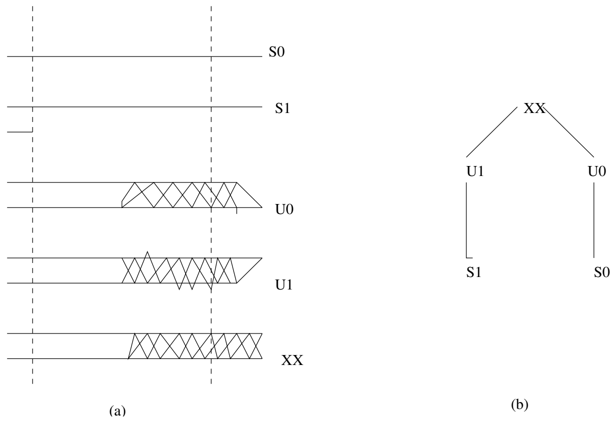

Figure 7. (a)Five-valued logic proposed in and (b)the relationship between them

- $< V_1$, $V_2 >$ will launch the desired transition at the input of path P_p, and
- at each gate on path P_p, the off-path sensitizing inputs have logic values that are cover by the values indicated on sensitizing table in figure 8-d .
The theorem is proved in the paper [2].

3.2. Parallel test pattern generation implementation

To implement a parallel test pattern generation, We propose a multiprocessing approach and make use of the above definitions and theorem from [2]. . For a given two-pattern vector $< V_1,V_2 >$, multiple paths (equal to the number of the processing elements) are checked simultaneous to determine the paths for which the pattern may be robust or nonrobust. It involves the following steps:

1- The two vectors are initiated at the primary inputs (one vector at the time) and are propagated through the circuit upto the primary outputs

2- paths are assigned to the processing elements (one path per processing element)

3- Then by backtracing multiple paths simultaneously, we determine all paths for which the test is robust or nonrobust according to the definition given above.

Since the signal through individual path can be initiated for rising or falling transition, for a given transition through P_p, if fault detected by the pattern vector is robust, then the path can be cancelled from the list for that the type of transition.

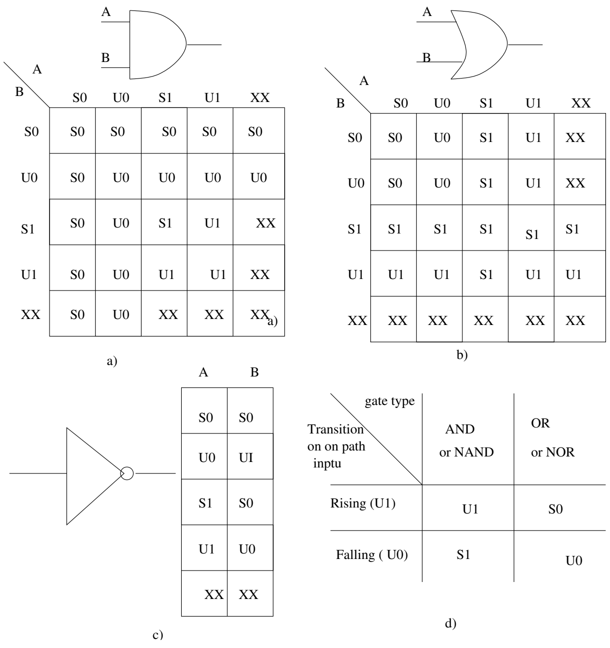

Figure 8. a), b), c) are Implication table for different logic gates AND, OR and NOT. d) Sensitizing values

With the five-valued approach proposed in [2] (i.e. the logic value throgh the path should either U0 or U1 for the path to be robust), since a primary input and output values should be part of the path,if the logic value observed is neither U0 nor U1, we can easily conclud that the pattern is nonrobust for all the paths. This can reduce in great the number of computation.

If some or all the input signals of a gate belong to the paths being tested, these inputs are treated mutual exclusively as off-path sensitizing input. This allow the use of theorem 1 in individual processor. (note this can also permit a concurrent processing of the path for test generation).

The parallel pattern test generation reduces greatly the time needed to generate the test and this is well appropriate for the path delay fault test generation since the number of path grows exponentially with the number of the gates in the circuit.

Example:
We use the example found in [2] figure 5 which is reproduced here as figure 9

For this example, we would like to determine the path for which the test pattern $< 0X000, 00101 >$ is robust or nonrobust. After signal propagation from the primary

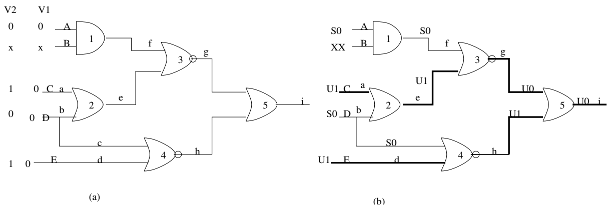

Figure 9. An example for parallel test pattern generation

imnputs to primary output, the five logic are assigned to lines according to figure 8 as depicted in figure 9.
Let suppose there are four processing elements M1, M2, M3 and M4 assigned respectively to the paths a-e-g-i, b-e-g-i, b-h-i and d-h-i during backtracing. Since for the a-e-g-i and d-h-i there are only the logic U0's and U1's the applied test pattern is robust for both simultaneously, whereas it is nonrobust for b-e-g-i and b-h-i.

An example of mutual exclusive treatment of off-path sensitizing input can be observed at the gate 5 (last OR gate). It can be noticed that U0 is an off-path sensitizing input for paths b-e-g-i and U1 is the off-path sensitizing input for the d-h-i and both the path are dealt with simultaneously

4. Random Sampling for Times for Delay fault Test

The increasing emphasis on delay testing of integrated circuit is driven by the combination of higher quality requirements and sensitivity of high performance circuit to delay defects. Both gate oriented and path oriented fault models have been proposed as the basis for test generation and fault simulation algorithms. Tools based on these models can be used to generate delay tests. The areas of fault modeling, fault simulation and test generations applied to delay testing have received most of the attention so far. The relatively unexplored side of delay test is the determination of the test application timing, which has a great influence on test results [8].

4.1. Testing delay faults using a single observation time

It is well known that a pair of test patterns $< v1, v2 >$ are required for detecting a delay fault. A typical test environment for delay fault testing is shown in figure 10. A combinational circuit C is the circuit under test. Two latches, an input latch and an output latch, are connected to the input and output of C. These two latches are clocked by

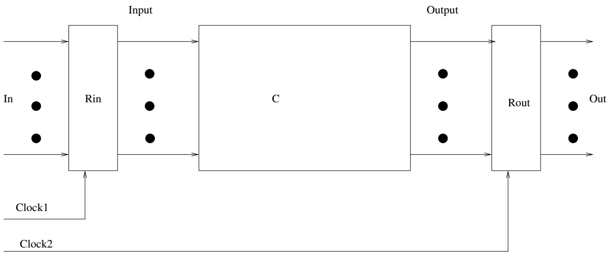

Figure 10. Testing envivronment for single observation time

Clock1 and Clock2. Initially, pattern v1 is load into the input latch. After the circuit has become stable, v2 is applied to the circuit by Clock1 at time t0. Then, at time t1, the circuit's output logic value are loaded into the output latch by Clock2. At this moment, if any output value is different from the expected one, a delay fault can be identified. The time difference between t1 and t0 is called the observation time, denoted as tobs.This is called fixed observation time in delay fault testing. Almost all fault simulation and test generation methods use this.

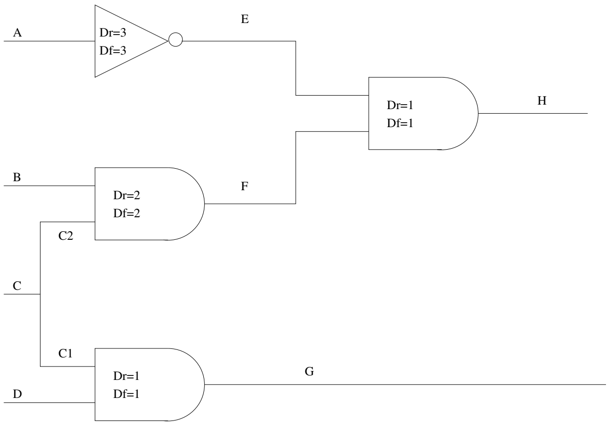

Figure 11. Example

Below some drawbacks of using a fixed observation time in delay fault testing are shown:

Drawback 1: Use of a fixed observation time may lead to pessimistic detection thresholds. That is, a different observation time might allow detection of a smaller fault.

Example:
For the circuit shown in figure 11, assume tobs=5. It is easy to see that any delay fault with fault size $\sigma < 4$ at line C1 will be claimed as undetectable at output G for the chosen observation time. However , if we sample output G using tobs=2,a delay fault with size $\sigma > 1$ at line C1 can be definitely detected by an observation at G when the fault is sensitized by a proper transition input at line C and D. A test procedure that uses tobs=5 will lead to pessimistic results.

Drawback 2 Use of a fixed observation time might lead to a poor choice of the stimulus pattern.

Example:
Assum that tobs=5 and that there is a slow-to-rise fault at line C of figure 11. There are two candidate test pairs: TP1=(A,B,C,D)=(xx,00,01,11) and TP2=(A,B,C,D)=(00,11,01,00). Each pair assigns test patterns to the inputs. Assum that input transitions occur at time t0=0. Using tobs=5, TP1 can definitely detect the delay fault at C having size $\sigma > 4$ by observing the output at line G. Using tobs=5, TP2 can definitely detect the delay fault at C having size $\sigma > 2$ by observing the fault at the output line H. So, on the basis of minimizing the detection threshold, test pair TP2 with tobs=5 would appear to be the best choice. However, if we sample the output of line G using tobs=2, TP1 can definitely detect the fault for fault size $\sigma > 1$. So, TP1 should be chosen to detect the fault.

4.2. Testing delay faults using variable observation time

In normal operation, a circuit should reach a stable state before the output are sampled. From the delay fault testing point of view, a given output should be sample at the time when a delay fault can be definitely detected for fault size $\sigma > \varepsilon_1$ with ε_1 as small as possible, regardless of whether other output have yet stabilized.

Let's see a example[7]. If a given test pair is capable of revealing a delay fault, how can the minimum detection threshold and the corresponding observation time be determined.

Example :
For the circuit in figure 11, it was shown in above example that the observation time for the minimum detection threshold deponds on the test pair and the output to be sampled. It is also easy to see that a delay fault at line E requires $tobs \geq 4$, and a a delay fault at line G requires tobs=2. So, the observation time for the minimum detection threshold depends on the test pair , the detected fault and the output to be sampled.

After the two methods are compared we can see: the method based on random sampling time of the circuit outputs during testing produces higher delay fault coverages than metnod based on fixed sampling time.

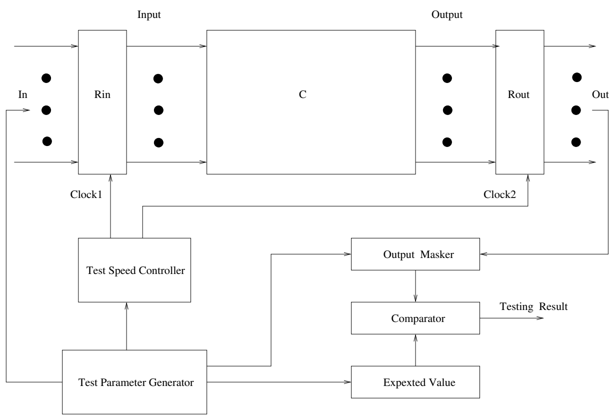

Figure 12. Test Environment for Random Testing

4.3. Impact on the hardware

A new testing environment is required for delay fault testing using a variable observation time. This is shown in figure 12, has the same circuit under test and input and output latch as in figure 10. The test generator first applies an input pattern v1 to the input latch and adjusts the testing speed to a value specified by tobs. After the circuit reaches a stable state and the clock adjustment has been made, the input pattern v2 is applied to the input latch. The outptu values are sampled from the output latchat at the prescribed observation time. The sampled output values are masked and compared with expected values.

This capability of using random sampling time during testing greatly adds to the hardware complexity of the tester. And also it may be actually impossible to using random sampling time in certain testing situation, such as in some BIST enviroments, where the tester clock is under complete internal control.But it overcomes the drawback of the fixed observation time method and give a tradeoff between gains in detection thresholds and the hardware cost.

5. Delay verification Test sets

5.1. Concept

One approach for verifying the correct temporal behavior of a circuit is to apply delay tests for each path. If every path of a circuit is tested and passes the test, the circuit is considered to be fault-free and will operate correctly at the intended speed. To ensure test quality, tests are required to be robust. A robust test for a particular path delay fault will detect that fault independent of the delays in the rest of the circuit. But robust tests may not exist for all paths in an arbitrary circuit.

Some nonrobust tests can be shown to be valid if certain other paths have been tested robustly[5] such test, called validatable nonrobust (VNR) test, can be used together with robust tests to improve the coverage of path delay faults.

The term RV-tests is used to denote tests that are robust or validatable nonrobust. A circuit is RV-testable if there is a robust or VNR test for every path delay fault.

Since many combination circuits may not be RV-testable, a new approach is proposed to delay testing using what we shall call delay-verification test sets. The new delay test set can detect the presence of any path delay faults that can affect the timing of the circuit.

We shall illustrate the concept of delay-verification test sets by means of a few simple examples. Path delay faults will be represented by direction of transition, $\uparrow$ (rising) or $\downarrow$ (falling) followed by the path.

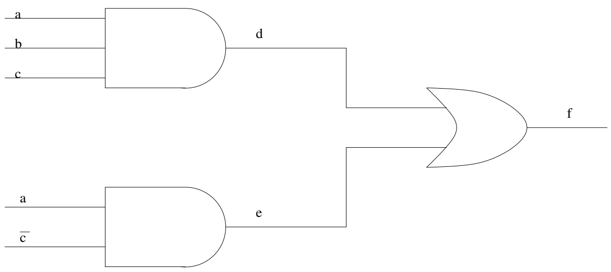

Figure 13. Circuit for example1

Example 1:

In the circuit of figure 13, all path delay faults except those on adf are detectable by robust test. Fault $\uparrow$ adf is testable by a VNR test abc=$< 011, 111 >$; this nonrobust test can be validated by a robust test for fault $\downarrow$ aef. Fault $\downarrow$ $\overline{a}$df is untestable.

The signal values for the vector pair $< 111, 011 >$, which is the only one that propagates a falling transition from a to d. the fault adf can produce an incorrect output only if the rising transition at e is also late, but this condition will be detected by a robust test for aef. If the circuit passes the robust test for fault aef, fault adf is guaranteed not to affect the timing of the circuit. The circuit is thus delay-verifiable, although not RV-testable, and the set of test for the detectable faults forms a strong delay-verification test set.

** strong delay-verification test set-if the correct temporal response to this set at a time t, implies that circuit will perform correctly at the tested clock period, t, and also at any slower speed.

** a complete RV-test set is a strong verification test set.

**delay-verifiable-delay of each path of the circuit is guaranteed to be within the specified range or is guaranteed not to affect the timing of the circuit.

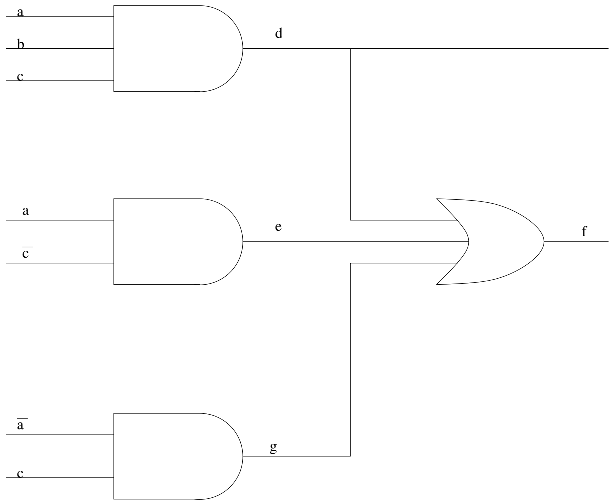

Figure 14. Circuit for example 2

Example 2:

In the circuit of figure 14 , all path delay faults except those on adf and cdf are detectable by robust tests. Faults $\uparrow$adf and $\uparrow$cdf are testable by VNR tests abc=$<110,111>$ and adc=$<011,111>$, respectively. But $\downarrow$adf and $\downarrow$cdf are untestable. The circuit is not RV-testable. Fortunately we can add the test abc=$<111,010>$ to the set of tests for the detectable faults to form a strong verification test set; thus the circuit is delay-verifiable.

As the example shows, a strong verification test set may require the inclusion of a special kind of test which can detect the simultaneous presence of multiple path faults. For a circuit which is no RV-testable and passes the RV-test set, a delay fault on a non-RV-testable path may not affect the timing of the circuit except when some other non-RV-testable path delay fault is also present. The multiple-path-delay-fault(MPDF) tests are motivated by this observation and are only used to detect the coexistence of multiple faulty non-RV-testable paths that can affect the timing of the circuit.

5.2. Implementation of delay verifiable circuit

First let's see some definitions. And then see some theorems and lemmas to know how delay verifiable circuits can be implemented (the two-level AND-OR circuits are focused below).

Definitions

A literal is a Boolean variable or its complement. A cube is set of literal and corresponds to an AND gate in circuit. A cover is a set of cubes which connect to an OR gate in circuit. A 0-vertex (1-vertex) is a vertex in the OFF(ON) set. A vertex is called a relatively essential vertex of a cube q if this vertex is contained only in q. a set of literals L is said to be prime in a cube q, if q-L contains a 0-vertex. Otherwise it is called a non-prime literal set.

Theorem 1:

Consider a two-level AND-OR single-output circuit. Let P be a set of paths that start with a set of literals L in a cube q. for $1\rightarrow 0$ output transition, the multiple path delay fault F for falling transition on P is robustly testable iff L is prime in q.

Theorem 2:

Let C be an irredundant two level AND-OR single-output circuit with a cover E. let q be a cube in E and D=E- q . For $0\rightarrow 1$ output transition, a path that starts with a literal is RV-testable iff there exists a vertex v2 that is relatively essential vertex of q, and a vertex v1=v2-l+l that satisfies one of the following condition:

1, C(v1)=0 or

2, for each cube $d \in D$ that contains v1, $\bar{l}$ is prime in d.

Lemma 1:

In an irredundant two-level AND-OR single output circuit C, if there is a path which is not RV-testable for $0\rightarrow 1$ output transition, then

1, there exists some path which is not RV-testable for $1\rightarrow 0$ output transition; and

2, C is not delay-verifiable.

Lemma 2:

Let C be a two-level AND-OR single-output circuit. The delay of a set of path that starts from a non-prime literal set in cube q will not affect the delay of circuit C for $1\rightarrow 0$ output transition if 1, every path has been tested for $0\rightarrow 1$output transition and passes the test; and

2, every multiple path delay fault that starts with a prime literal set in q has been tested for $1\rightarrow 0$ output transition and passes the test.

Theorem 3:

An irredundant two-level AND-OR circuit is delay-verifiable iff every path is RV-testable for $0\rightarrow 1$ output transition.

An irredundant two-level AND-OR single-output circuit is delay-verifiable if and only if each cube and for each literal in this cube, the condition listed in theorem 2 are satisfied. For a multiple-output circuit, these conditions have to be satisfied for each output function.

In summary, To obtain a delay-verifiable circuit, the conditions described in Theorem 2 have to be satisfied. The conditions for delay verifiability are less stringent than those for RV-testablility which requires the implementation of each output be prime and irredundant. But these conditions are sufficient to guarantee that a circuit that passes all delay-verification tests will exhibit the correct timing behavior even if one or more untested delay faults occur. The MPDF test plays an important role in forming delay-verification test sets. And also these delay verifiable circuits usually have lower area overhead than RV-testable circuits[9].

6. Conclusion

VLSI circuits must be verified for both logic and timing correctness. Departure from the correct temporal behavior is usually modeled by delay fault. We have surveyed the two fault models(gate delay fault model and path delay fault model) and their advantages and limitations. It is generally accepted that path delay model is most comprehensive. By means of using parallel test generation we can save simulation time. Using random sampling times we can get high fault coverage. Using delay verifiable circuit we can get high fault coverage and area efficient. The drawbacks of these approachs may be related to the design complexity and high cost.

References

[1] G. S. Author. Model for delay faults based upon paths. *International Conf*, 14(1):342–341, November 1985.

[2] C. J. L. Authors and S. M. REDDY. On delay fault testing in logic circuits. *IEEE Transaction on Computer-Aided Design*, CAD-6(5):694–703, September 1987.

[3] W. K. L. A. S. R. K. B. Authors and A. L. Sangiovanni-Vincentilli. Delay fault coverage, test set size, and performance trade-offs. *IEEE Transaction on Computer-Aided Design of IC and System*, 14(1):32–44, January 1995.

[4] V. B.K.Rosen and I.Spillinger. Delay test generation 1–concepts and coverage metrics. *IEEE International Test Conference*, pages 857–866, 1988.

[5] S. Devadas and K. Keutzer. Validaatable nonrobust delay-fault testable circuits via logic synthesis. *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN*, 11(12):1559–1573, December 1992.

[6] A. K. Majhi and V. D. Agrawal. Tutorial: Delay fault models and coverage. *IEEE*, pages 364–369, 1997.

[7] W.-W. Mao and M. D. Ciletti. A variable observation time method for testing delay faults. *IEEE Design Automation Conference*, pages 728–731, 1990.

[8] D. D. P.Girard and C.Landrault. Effectiveness of a variable sampling time strategy for delay fault diagnosis. *IEEE*, pages 518–523, 1994.

[9] W.Ke and P.R.Menon. Synthesis of delay-verifiable two-level circuits. *European Design and Test Conference*, pages 297–301, 1994.