

Path Delay Testing

Vippa Prakash Reddy

Dept. of Electrical and Computer Engineering

Auburn University

Auburn, AL 36849

Reviewer: Garimella Srinivas

ABSTRACT - *A literature survey on the Path Delay Testing has been carried out. The necessary and sufficient conditions for the existence of general robust tests in the multi-input and multi level circuits have been studied [3]. A background survey on the testability problem of the path delay faults has been carried out. Robust Path delay faults are explained and the delay test implementations of the non robust paths are studied. Path enumeration techniques and block oriented techniques used for timing verification of a circuit are discussed [4]. Methods of generating critical paths have been explained using an example circuit.*

I. INTRODUCTION

The ever-growing demand for faster machines in the market place is self evident. Cycle times of being set more aggressively, resulting in a burden for better delay testing. Delay testing requires at least a sequence of two input vectors to create and propagate the signal transitions, called two pattern tests [1]. These delay faults cause the logic circuits to malfunction at the desired clock signals. These can be classified into: Path Delay Faults and Gate Delay Faults. In Path Delay Fault models the defects cause the cumulative path delay faults along the paths to exceed the specified values, whereas in the Gate Delay Faults the faults are assumed to be 'lumped' at the inputs/outputs of the gates.

Test invalidation, a signal propagating in a specific designed path may be preempted by any other fast switching signal path that shares a common segment, problems can be compensated by the Robust Delay Path Tests.

Robust Test: A two pattern test $T = \langle V1, V2 \rangle$ is said to be a robust delay test for a path P (for a raising or falling transition at the output of the path), if and only if, when P is faulty and test T is applied, the circuit output is different from the expected state at the sampling time, independent of the delay along gate input leads not on P [2].

The Circuit under Test (CUT) often doesn't have robust tests, thus giving some erroneous results. The main idea of the path delay testability is not to impact the cycle time and to give a Circuit under Test where all paths have robust tests. Although, the CUT in which all paths are robust testable may not be a complete solution. As we need to add some additional combinational circuitry and also need to find a way of applying input vectors. This may not be a feasible solution both in terms of extra hardware and cost.

In general, a single robust test may be a test for a several paths to some output in the circuit [3]. So this robust test may propagate the signal transitions to the output through more than one path. These are called as Multiple-Path Propagating (MPP) robust tests [3]. And the robust test that propagates through only one path is called Single-Path Propagating (SPP) robust test. Similarly if the number of the inputs that change when the input vector is changed from V1 to V2, is one, then this circuit is called Single Input Change (SIC) robust test and if more than one, then Multiple Input Change (MIC) robust test [3].

II. PATH DELAY TEST

When testing for path delay faults, only a subset of faults is considered, as it is

expensive to consider all the paths in the circuit. The paths that are considered are the longest paths in the circuits. These paths are called critical paths. And the critical path of the circuit can be found by using the techniques given in [4]. Path enumeration techniques and block oriented techniques used for timing verification of a circuit are discussed in [4]. The path enumeration techniques are characterized by back tracing, but, this tends to have long running times since the number of paths through a graph grows exponentially with the size of the graph. Another method discussed is the Block oriented technique. These programs start at a primary input PI or at any storage element output. The blocks (the signal feeds) are processed to find the latest/earliest time at which the signal could pass through them. And the outputs of these blocks are calculated once. This algorithm will identify the worst path leading up to each block (the critical path up to that point), and then feeds the information forward.

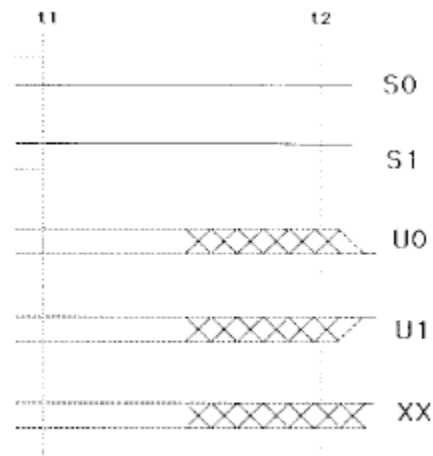
The method often used to detect delay faults is to apply an input vector $V1$ to the circuit under test at time $t0$. After the signals in the circuit have stabilized, a second input vector $v2$ is applied at time $t1$, such that a rising or falling transition is propagated along the critical path. By sampling the path output at time $t2$, where $(t2 - t1)$ corresponds to the desired operational time interval, one detects the existence or non-existence of a delay fault. Thus two vectors are applied in pair to propagate required transition along a critical path to detect a delay fault along that path.

In [2] concept of robust path delay test is introduced. It is defined as follows : A two-pattern test is said to be a robust delay test for transition path Lx , if and only if, when Lx is faulty and test is applied, the circuit output is different from the expected at sampling time $t2$, independent of the delays along gate input leads not on path Lx [2]. Robust delay test insures that tests to detect a path delay fault

are not masked by delay characteristics of devices not on the tested path.

In [2] five valued logic $\{S0, S1, U0, U1, XX\}$ is developed for the robust delay test. Each of these values represents a signal on the lines in the circuit at the time interval $(t1, t2)$. The proposed 5 value logic and the relation between the values is depicted in fig.1.

(a)



(b)

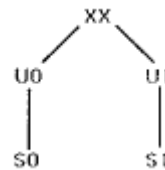


Fig.1. (a) Proposed 5 value logic and (b) relationship between these logics [2].

III. DESCRIPTION

The longest delay combinational path of the circuit is known as the critical path. In finding critical path, a unit delay may be assumed for all the gates. Initially all gates are levelized. The level is the maximum distance, counting gates from primary inputs. For determining the critical path, we start at the primary output from a gate with maximum level. While back-tracing, next gate is selected whose level is maximum compared to any of

its fan-in gates. If the level of a gate is 'L', there should be at least one gate at its input whose level is 'L-1'. Thus level of the gate chosen while back-tracing will always be one level below the previous gate in the path. This procedure is repeated till primary input is reached, and the path connecting all these gates gives a critical path of the circuit. This traversal gives all possible critical paths in the circuit. If n is number of gates in the circuit, levelization and critical path counting is accomplished in $O(n)$ time.

The critical path of a sample circuit is shown in fig.2. The gates are leveled from 1 to 5. Primary Output F2 is connected to gate G9, with a maximum level of 5. And of the fan-in gates, G8 comes from previous level. For G8, both G6 and G5 come from previous level. This gives rise to two critical paths. The procedure (backtracing) is repeated until we reach a Primary Input (PI). This procedure gives rise to 6 critical paths; path1: F2-G9-G8-G6-G2-5, path2: F2-G9-G8-G6-G4-G1-P3, path3: F2-G9-G8-G6-G4-G1-P4, path4: F2-G9-G8-G6-G3-P2, path5: F2-G9-G8-G5-G3-G1-P3, path6: F2-G9-G8-G5-G3-G1-P4.

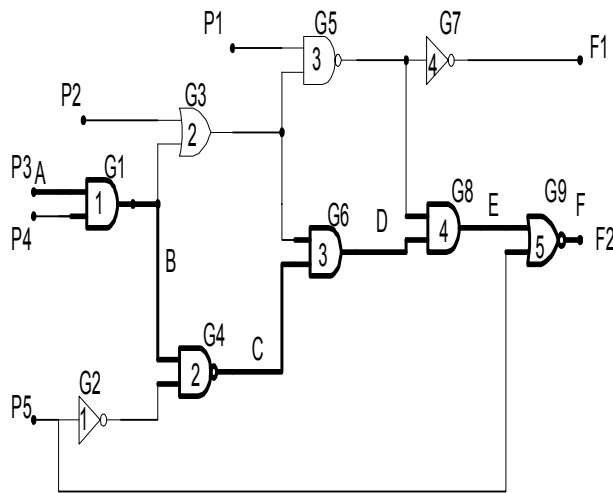


Fig.2. Combinational Circuit for finding the critical path.

Now the delays of the critical paths are converted into equivalent fault delays. The critical paths are controlled by choosing proper

AND/OR gates. Generally the controlling values are assigned to the on-path signals and the non-controlling signals are assigned to the off-path signals

IV. TIMING ANALYSIS

The timing analysis examines the combinational paths in the circuit topology. Delays of gates and interconnections are obtained from the simulator database. The results of timing analysis are used to improve the design and testability of the circuit. The following ways are used.

Timing Simulation: Critical paths are simulated by circuit level or timing simulators, provided by the designer based on the functional knowledge of the circuit.

Critical path tests: The critical path of a circuit is the longest combinational path between two flip-flops. Critical path delay determines the clock period. The design test vectors are sent through the critical paths.

Layout optimization: Critical path data is used for placement of standard cells and custom blocks, to prioritize interconnects for routing, transistor sizing. Chip size can be drastically reduced by choosing the non-critical paths, by letting their delays increase.

The manufacturing yield can be increased by careful timing analysis, but this may not guarantee one with the fault free chip.

V. CONCLUSION

The design for the testability is always associated with costs for the design procedure. And the path delay faults are one of the most important faults which have to be recovered for the effective implementation of the VLSI chip. The extra hardware required to implement these path delay faults may effectively increase the chip area, which is not desired. The necessary conditions for the robust delay path testing are described and the critical path finding is described with an example. The importance of timing analysis is also elicited in this paper.

ACKNOWLEDGEMENTS

First I would like to thank Dr. Vishwani Agrawal for his support and encouragement throughout this course. I would also like to thank my friends and colleagues who helped me directly or indirectly.

REFERENCES:

- [1] Pramanick, A. K.; Kundu, S.; Design of scan-based path delay testable sequential circuits, *International Test Conference, 1993. Proceedings*, 17-21 Oct. 1993 Pages: 962 – 971
- [2] Chin Jen Lin; Reddy, S.M. “On Delay Fault Testing in Logic Circuits”; *Computer-Aided Design of Integrated Circuits and Systems*, IEEE, Vol. 6, No: 5, Sep. 1987, pp. 694 – 703
- [3] Pramanick, A.K.; Reddy, S.M.; on the design of path delay fault testable combinational circuits Fault-Tolerant Computing, 1990. *FTCS-20 Digest of Paper, 20th International Symposium*, 26-28 June 1990 Pages: 374 - 381
- [4] Hitchcock, R. “Timing Verification and the timing Analysis Program”, *Proceedings of DAC*, 1982.