

RAFT: A NOVEL PROGRAM FOR RAPID-FIRE TEST AND DIAGNOSIS OF DIGITAL LOGIC FOR MARGINAL DELAYS AND DELAY FAULTS

Abhijit Chatterjee, Georgia Institute of Technology, Atlanta, GA
Jacob A. Abraham, University of Texas at Austin, Austin, TX.

ABSTRACT

The problem of delay fault-testing and detection of chips with marginal performance has become even more critical than before due to advancing clock speeds. In this paper, a methodology for detection of marginal digital circuits and diagnosis of gate delay failures is developed. A new test application methodology is proposed in which test vectors may be applied to digital combinational circuits at intervals smaller than the critical path delay of the circuit and signal waveform analysis is used to interpret the test results. The resulting tests are called **RA**pid **F**ire **T**ests (for RAFT) and allow classification of circuits from "good" to "bad" along a continuous scale.

1. Introduction

The problem of testing manufactured digital parts has become very difficult due to high levels of integration and the increase in complexity of circuit designs. In many instances, entirely novel and unorthodox methods have been used to simplify the testing problem and to expose failure modes that are not easily detectable by conventional testing techniques. Such an example is current testing¹. With evolving technology, the problem of high-speed testing of digital circuits has become very important. With clock speeds in excess of 100Mhz, it is important to detect marginal chips, i.e. chips that work at a specified clock speed but fail at speeds marginally higher than the specified clock speed. In mission critical and long-life applications, such as in space exploration, it is extremely important that marginal chips not be used.

The problem of testing for delay faults was studied by Hsieh, et. al. in², by Malaiya and Narayanswamy in³, by Smith in⁴, and by Lin and Reddy in⁵. Automatic test generation algorithms for delay faults were developed by Reddy, et. al. in⁶, by Schulz et. al. in⁷, and by Lesser and Schedletsky in⁸. The problem of logic synthesis for delay fault testability has been recently addressed by Roy and Abraham in⁹, by Pramanik and Reddy in¹⁰, by Kundu and Reddy in¹¹, and by Devadas and Keutzer in¹². In¹³, Iyengar and Vijayan have formulated the problem of tight test application timing during ac test as a graph theoretical problem. They assume that the tester is capable of sampling each output and of applying test stimulus at each input at different times. A nonenumerative method for estimating path delay fault coverage in combinational circuits has been investigated by Pomeranz and Reddy in¹⁴. Their method is polynomial to the number of circuit input lines. In¹⁵, Hao and McCluskey have proposed the use

of low voltage testing for the detection of weak CMOS logic ICs. Most relevant to the research presented in this paper is the work of Franco and McCluskey¹⁶. They have suggested output signal waveform analysis using time-domain integration as a means of analyzing circuit response to delay tests. We use a similar signal waveform analysis approach in this paper. Our research complements their earlier work by investigation of the test generation problem, the use of rapid-fire tests, the ability to identify marginal chips and the ability to perform fault diagnosis.

2. Problem Specification

The problem addressed in this paper is based on the following objectives:

- (1) To reduce test cost by allowing application of test sequences at intervals ranging from the critical path delay of a combinational circuit to any subinterval thereof.
- (2) To apply test sequences that allow classification of marginal chips and identification of delay failures
- (3) To locate the source of the delay fault (or potential source in case of marginal chips) down to as few logic elements as possible

3. Premise and Motivation

The purpose of this paper is to explore the potential of a new test application and response analysis methodology that conveys more information about the timing characteristics of a digital circuit than is possible with simple go/no-go tests at a specified test application speed. The *lumped gate-delay* model is used in RAFT. It is assumed that only one gate is faulty (slow to rise or slow to fall). We propose that test vectors may be applied at intervals (called the *test insertion interval* or TIV) smaller than the critical path delay of the circuit being tested. As an example, Figure 1 shows a NAND implementation of a full adder consisting of 9 gates. Figure 2 shows the fault-free response of the full adder to a stimulus consisting of the vectors $(abc) = 111, 101, 000, 001, 000$ applied at intervals of one NAND gate delay to the full adder. The vector 111 is applied initially to the full adder for a length of time equal to 6 NAND gate delays (the depth of the circuit of Figure 1) and the vector 000 held for the same period at the end of the test sequence, to 'flush' out residual transitions inside the circuit. In the proposed test methodology, several chosen test vector sequences, called *rapid fire test sequences* (RFTS), are applied to the circuit under test (CUT).

To analyze the circuit response to the rapid fire test sequences, we assume that accurate integrators¹⁶ are used to integrate the waveforms obtained at each of the outputs of the combinational circuit being tested. If t_i and t_f are the times

This research was supported in part by the Georgia Tech Foundation and by NSF Grant No. MIP-9222481 to the University of Texas at Austin.

Permission to copy without fee all or part of this material is granted, provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

at which the initial and final test vectors of a RFTS are applied to the CUT, then the integration interval is chosen to be $t_i + T_{crit}$ to $t_f + T_{crit}$, where T_{crit} is the critical path delay of the CUT. Having described the test application and response analysis methodology, we now ask whether it is possible to design each RFTS such that the following linearity property is satisfied. We assume that δ_i is the nominal delay value associated with the i 'th logic gate of the CUT and MARGIN is an externally specified numerical quantity.

Linearity Property: The value of the integral computed at one or more outputs of the CUT changes *linearly* with the delay value of each gate tested by the RFTS over a range of delay values of the gate given by δ_i through $\delta_i + \text{MARGIN}$.

The motivation for the above is explained by Figure 3 which shows the specified linear relationship. When the integral value is larger than a calibrated threshold, the circuit under test is deemed to be faulty. Under fault, the integral value can lie between its nominal value and the threshold. If the circuit *passes* the test, the value of the integral can be used to determine whether the circuit is marginal or not. The closer the integral is to the threshold, the more 'marginal' the circuit is with regard to high-speed performance. Whereas, if the integral is close to its nominal value, then the circuit is not 'marginal'. The tests should be so designed that the linear relationship of Figure 3 extends beyond the threshold on the vertical axis. Note that if there exists a gate delay failure whose effects are observable at the primary outputs under a RFTS and for which there does not exist any output satisfying the linearity property, then that stimulus is rejected as a test.

4. Rapid-Fire Test Issues and Fault Diagnosis

Owing to the manner in which the test stimulus response is observed, multiple path sensitization is possible in circuits with reconvergent fanout, since the presence of glitches [11] does not invalidate the tests applied. When multiple paths are sensitized, signal transitions traveling through the faulty gate are delayed, resulting in translation of pulses in time as observed at the circuits outputs or changes in the pulse widths. RAFT uses an accurate event-driven timing simulator that includes the effects of glitches to compute the value of the integral of the waveforms obtained at the circuit outputs. As an example, Figure 4 shows the response of the full adder of Figure 1 to a rapid-fire test under a delay fault in gate X4. Note the change in the width of the largest pulse generated by the test as opposed to the fault-free output waveform of Figure 2.

DEFINITION 1: We define the *sensitivity* of the integral of the waveform obtained at the i 'th output to the magnitude of a delay fault in gate G_j as the quantity S_{ij} . This represents the ratio of the change in the value of the computed integral to the change in the delay of G_j under fault in the region that their relationship is forced to be linear by the test generation process.

The problem of determining the output waveform integral threshold at which to indicate failure is closely related to the fault diagnosis problem which is briefly described below. Delay faults in operators G_u and G_v are distinguishable if $S_{iu} = 0$ and $S_{iv} \neq 0$. As an example, these are also

distinguishable if the signs of S_{iu} and S_{iv} are different. RAFT contains algorithms for detectability and diagnosability of gate delay faults. These in turn are used to determine the fault coverage that can be achieved for detection of marginal delay faults.

5. High-Level Description of RAFT

RAFT is implemented in about 3500 lines of C code and contains a random test generator, event-driven timing simulator, fault simulator with output waveform integral computation and a diagnostics routine. The flow diagram of RAFT is shown in Figure 8.

RAFT first selects a set-up vector. This is chosen to maximise the chance of propagating input changes through the circuit. In step 2, a test vector is chosen to be applied at the end of the current test sequence for a duration of time equal to the test insertion interval. This same test vector is chosen as the flush vector in step 3. Step 4 shows the key approach of RAFT. A timing simulator is used to simulate the circuit with the current test sequence as input. The delay of each operator is increased incrementally in an iterative manner. If the integral behaves linearly with regard to changing delay values of each gate, then the sensitivity S_{ij} , defined earlier, is computed. If the above behavior is nonlinear, then the respective circuit output is 'invalidated' (in case all circuit outputs become 'invalid' the test sequence is abandoned and a new test sequence initiated). In step 5, the fault coverage is determined and fault diagnosis is performed. If the fault coverage is increased, then the test vector is added to the test sequence in step 6. In step 7, RAFT determines whether to terminate the current test sequence and initiate a new one. A new test sequence is initiated if no new faults are detected over a predetermined number of intervals. In step 2, a random test generator is used to generate RAFT tests. This test generator uses internal node activity statistics to guide the transition probabilities at the circuit inputs.

6. Results

RAFT was used to find tests for several circuits. Since RAFT uses random test generation, the results should be compared to what one would expect with a random test generator. However, a one-to-one comparison with an existing test generator would be improper as the problems that RAFT solves are considerably different from conventional stuck-at fault testing and many times more complex. Not only does RAFT have to sensitize a fault to an output, it has to do so under the constraint of linearity for the detection of marginal chips. The computation times for RAFT are expectedly high due to repeated and expensive timing simulation. We are currently developing RAFT_2 which will solve some of these problems.

Table 1 shows the results. The ckt fa is the full adder of Figure 1, 2_add is a two-bit serial adder, w_tree is a 7-input generalized counter, 2_3_mult is a 2 bit by 3 bit multiplier and the remaining circuits are taken from the ISCAS benchmark set. NGATES is the number of gates in the circuit, DEPTH is the number of gates along the critical circuit path(s), TIV is the test insertion interval (in units of one gate delay), NSEQ is the number of rapid-fire test sequences in the rapid-fire test set, MEM is the total number of times new data is applied to

Table 1. RAFT Results.

CKT	NGATES	DEPTH	TIV	NSEQ	MEM	TTIME	FC	DIAG	CPU
fa	9	6	1	2	7	27	100%	1.14/9	15s
2_add	18	12	1	4	15	60	95%	1.08/17	30s
w_tree	20	7	3	4	17	65	100%	1.5/20	45s
s27f	11	6	1	3	9	41	100%	1.1/11	25s
s208f	121	12	1	8	33	208	72%	4/87	>5h
s344f	166	18	1	8	35	312	88.6%	11/141	>10h
s349f	165	18	1	8	34	310	89.7%	7/141	>10h

the circuit inputs (this is analogous to the number of test vectors in stuck-at testing), TTIME is the total test time in terms of unit gate delays, FC is the fault coverage (total number of gates detected; delay faults in these gates must cause the output waveform integral to change linearly), DIAG is the ratio of the average size of each diagnosable set (in terms of unit gates) to the total number of gate delay faults detected by the rapid-fire test and CPU is the CPU time taken to run the program. It was observed that higher fault coverage was obtained with rapid fire as opposed to conventional tests for specific cases. Also test application times were reduced by about a factor of 6.

7. Conclusions

In this paper, we have introduced the concept of rapid-fire tests and shown the viability and usefulness of such a testing methodology. The technique complements existing delay fault testing approaches and is very powerful. Not only can we detect marginal chips but also diagnose gate delay failures. The granularity of diagnosis is remarkably small. We are currently developing RAFT_2 with the objective of significantly speeding up test generation time and improving fault coverage.

REFERENCES

1. W. Maly and P. Nigh, "Built-in Current Testing - A Feasibility Study," *International Conference on Computer-Aided Design*, pp. 340-343 (1988).
2. E. P. Hsieh et. al., "Delay Test Generation," *Design Automation Conference*, pp. 486-491 (1977).
3. Y. K. Malaiya and R. Narayanswamy, "Testing for Timing Faults in Synchronous Sequential Integrated Circuits," *International Test Conference*, pp. 560-571 (1983).
4. G. L. Smith, "Model for Delay Faults Based Upon Paths," *International Test Conference*, pp. 342-349 (1985).
5. C. J. Lin and S. M. Reddy, "On Delay Fault Testing in Logic Circuits," *International Conference on Computer-Aided Design*, pp. 694-703 (1985).
6. S. M. Reddy et. al., "An Automatic Test Pattern Generator for the Detection of Delay Faults," *International Conference on Computer-Aided Design*, pp. 284-287 (1987).
7. M. H. Schulz, "Advanced Automatic Test Pattern Generation Techniques for Path Delay Faults," *International Symposium on Fault-Tolerant Computing*, pp. 44-51 (1989).
8. J. P. Lesser and J. J. Schedletsy, "An Experimental Delay Test Generator for LSI Logic," *IEEE Transactions on Computers*, pp. 235-248 (March 1980).
9. K. Roy and J. A. Abraham, "Synthesis of Delay Fault Testable Combinational Logic," *International Conference on Computer-Aided Design*, pp. 418-421 (1989).
10. A. K. Pramanik and S. M. Reddy, "Synthesis of Combinational Logic for Path Delay Fault Testability," *International Symposium on Circuits and Systems*, pp. 3105-3108 (1990).
11. S. Kundu and S. M. Reddy, "On the Design of Robust Testable CMOS Combinational Logic Circuits," *International Symposium on Fault-Tolerant Computing*, pp. 220-225 (1988).
12. S. Devadas and K. Keutzer, "Validatable Nonrobust Delay-Fault Testable Circuits Via Logic Synthesis," *International Symposium on Circuits and Systems*, pp. 3109-3113 (1988).
13. V. S. Iyengar and G. Vijayan, "Test Application Timing: The Unexplored Issue in AC Test," *International Test Conference*, pp. 840-847 ().
14. I. Pomeranz and S. M. Reddy, "An Efficient Nonenumerative Method to estimate the Path Delay Fault Coverage in Combinational Circuits," *IEEE Transactions on Computer-Aided Design* **13**, No 2(February 1994).
15. H. Hao and E. J. McCluskey, "Very-low-voltage Testing for Weak CMOS ICs," *International Test Conference*, pp. 275-284 (1993).
16. P. Franco and E. J. McCluskey, "Delay Testing of Digital Circuits By Output Waveform Analysis," *International Test Conference*, pp. 798-807 ().

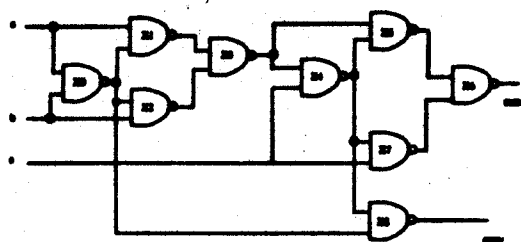


Figure 1. NAND implementation of Full Adder.

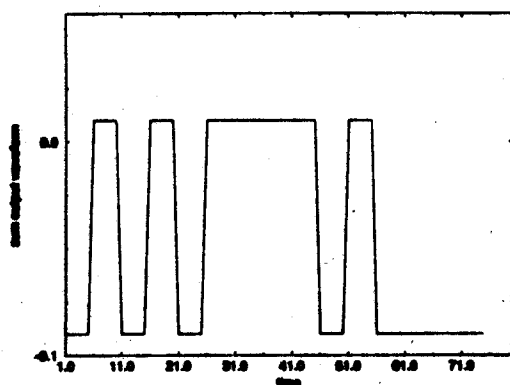


Figure 2. Fault-free Response of Full Adder.

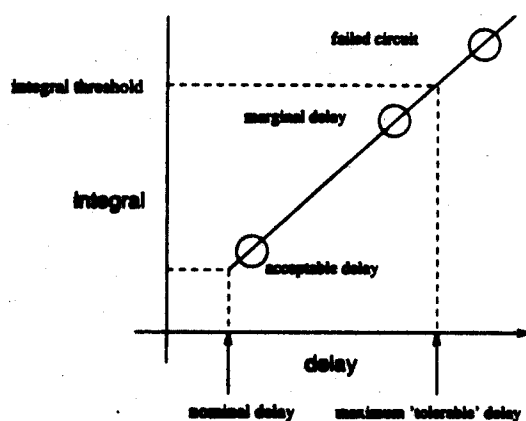


Figure 3. Output Integral vs Delay Value Under Fault.

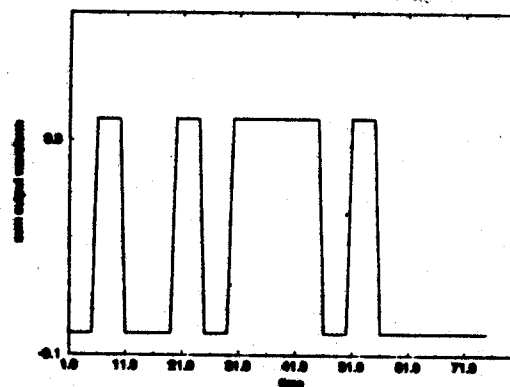


Figure 4. Pulse Width Variation Due to Gate Delay Fault

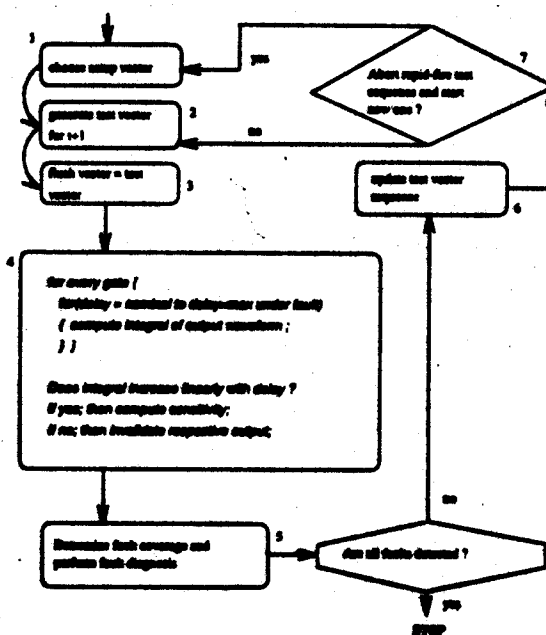


Figure 5. Flow Diagram of RAFT