On-Chip Delay Measurement for Silicon Debug [*]

Ramyanshu Datta
Computer Engineering Research Center
The University of Texas at Austin
Austin, TX 78712
rdatta@cerc.utexas.edu

Antony Sebastine
Computer Engineering Research Center
The University of Texas at Austin
Austin, TX 78712
antonys@cerc.utexas.edu

Ashwin Raghunathan
Computer Engineering Research Center
The University of Texas at Austin
Austin, TX 78712
ashwin@cerc.utexas.edu

Jacob A. Abraham
Computer Engineering Research Center
The University of Texas at Austin
Austin, TX 78712
jaa@cerc.utexas.edu

ABSTRACT

Efficient test and debug techniques are indispensable for performance characterization of large complex integrated circuits in deep-submicron and nanometer technologies. Performance characterization of such chips requires on-chip hardware and efficient debug schemes in order to reduce time to market and ensure shipping of chips with lower defect levels. In this paper we present an on-chip scheme for delay fault detection and performance characterization. The proposed technique allows for accurate measurement of delays of speed paths for speed binning and facilitates a systematic and efficient test and debug scheme for delay faults. The area overhead associated with the proposed technique is very low.

Categories and Subject Descriptors

B.8.1 [**Hardware**]: Testing Reliability Fault-Tolerance—*Design for Testability and Reliability*

General Terms

Reliability, Performance, Measurement

Keywords

Delay Fault Testing, Silicon Debug, Design for Testability

1. INTRODUCTION

Modern VLSI chips have stringent timing requirements driven by shrinking feature sizes. This has resulted in the emergence of delay faults as a significant problem and introduced the necessity to detect delay faults of the order of few picoseconds in logic paths of a digital integrated circuit. Delay faults are a category of faults which cause an otherwise functional chip to fail at specified clock speed and the objective of delay fault testing and debug is to detect timing defects and ensure that the design meets the desired performance specifications. Although timing verification and functional simulation during the design cycle ensure that a chip meets its performance specifications, it should be noted that these techniques are applied to a model of an *Integrated Circuit* (IC) and not to actual silicon. Hence these techniques are applied to a model of an *Integrated Circuit* (IC) and not to actual silicon and cannot detect delay faults which are caused by factors like distributed delay variations, crosstalk induced delay, logic errors, excessive voltage drop and swing on the supply nets, etc. Additionally, process variations can have a significant influence on a chip's failure to meet specified performance. Process parameter variations can result in distributed delay faults in the chip, which cause minor delay faults on multiple gates in a given path to accumulate and result in the path failing to meet performance specifications. Adding more details to models of an IC to incorporate these factors will cause the computational costs of timing verification methods to become prohibitive.

Diminishing feature sizes limit the observability of chips, making test and debug more difficult especially for timing violations. Also testing for delay defects using *Automated Testing Equipment* (ATE) for GHz range processors is very expensive and most testers in test facilities still run at a few 100 MHz. This makes on-chip circuitry an ideal choice for performance characterization and delay fault testing and debug. Also, the use of on-chip testing circuitry allows for at-speed testing essential for accurate detection of timing violations.

A commonly used technique for speed binning and detecting delay defects in chips is increasing clock frequency till the chip fails. There are sevaral disadvantages associated with this method. Primarily in GHz range frequencies, con-

[*] This work was supported in part by an IBM Faculty Partnership Award, and in part by Subcontract No. SA3271JB from UC Berkeley under prime Contract 2003-DT-660 from Microelectronic Advanced Research Corporation (MARCO).

GLSVLSI'04, April 26–28, 2004, Boston, Massachusetts, USA.
Copyright 2004 ACM 1-58113-853-9/04/0004 ...$5.00.

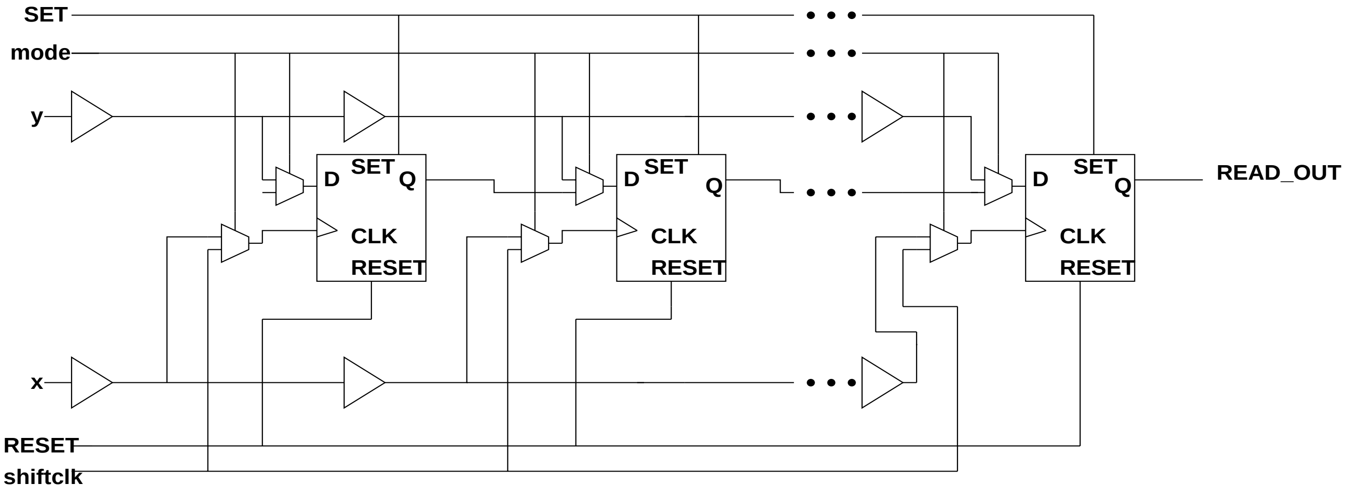

Figure 1: Modified Vernier Delay Line

Table 1: Possible Path Transitions and Corresponding Schemes to Handle Them

Input	Output	Proposed Solution
rising	rising	Feed input and output and feed into x and y of MVDL respectively
falling	falling	Invert input and output and feed into x and y of MVDL respectively
rising	falling	Feed input as it is into x of MVDL and invert output and feed into y
falling	rising	Invert input and output and feed into x and y respectively

trolling the clock edges in an accurate manner externally is not very feasible. Even if this was possible, this technique can only assure whether a chip meets its performance specifications or not and cannot point to the failing path. A solution to this problem is to use existing *Design for Test* (DFT) structure on chip by scanning out and observing the outputs of speedpaths or paths that are most likely to fail. However this is an iterative process and takes a lot of time considering that scan-chains have a typical length of 1800-2000 scan cells. A systematic debug approach essential for reduction of *time to market* requires addition of on-chip circuitry for delay measurement and characterization of speed paths. Several methods have been proposed for on-chip delay measurement. In [2] a sampling circuit is proposed based on two clocks which are 180° phase shifted with each other. However this method requires ratioed capacitors, and fails if the transitions of the two clocks are skewed. Also this sampling circuit can detect delay faults only if the delayed transition on a path occurs after the sampling instant. Franco and McCluskey [3] propose a DFT technique to detect delay faults using transient switching currents in CMOS inverters. However this scheme has a low noise margin which hinders its fault detection capability, and has a high switching power overhead since it is a dynamic circuit. In [4], a DFT technique based on capacitor voltage levels is proposed. This scheme requires determination of a threshold voltage which can be difficult to implement. Delay measurement schemes based on digitizing short intervals of time include a shift register/fast counter based *Time-to-Digital Converter* (TDC) [7], oscillator based TDC [5] and CMOS tapped delay line configurations [1, 6]. Analog methods based on voltage ramp generation have also been proposed [8], where the the voltage on a capacitor is proportional to the time difference between two rising edges. The major advantage of using delay line configurations is that they can be implemented in a standard digital CMOS process, which results in low power dissipation, high integration levels and good noise margins. However, the minimum achievable resolution of a TDC implemented using a single delay line is limited by the minimum gate delay of the technology in which it is implemented. This can be overcome by using a balanced delay line.

The organization of the rest of the paper is as follows. In Section 2, we introduce a balanced delay line scheme which can test a path for delay defects for all possible transitions. Section 3 explains a systematic debug scheme for delay fault testing and debug. Results are presented in Section 4 and conclusions are presented in Section 5.

2. MODIFIED VERNIER DELAY LINE

In this section we propose a *Modified Vernier Delay Line* (MVDL) that can be used to characterize critical path delays. The block diagram of the MVDL is shown in Figure 1. It consists of two delay buffer chains with the delay of individual buffers in the lower chain (t_{buf}) greater than the delay of individual buffers in the upper buffer chain (t_{buf_low}). The first arriving signal is fed to the input x of the lower buffer chain and the late arriving signal is fed to the input y of the upper buffer chain. As x and y propagate through their respective delay chains, the time difference between the two signals is reduced in every stage by an amount which equals the difference in delay of individual buffers in the respective chains. This is basically the resolution of the MVDL, i.e.,

$$t_{res} = t_{buf} - t_{buf\_low}$$

Edge triggered latching elements at every stage are clocked by propagating x and latching in the value of propagating y when the latch is transparent. The stage/latch number n in the delay line where x catches up with y indicates the range of time difference between x and y signals. The event of x catching up with y is indicated by the presence of the first '1' among the flops in the delay line. All subsequent stages will latch in a '1'. The range of the delay difference t_{diff} between the two signals is then given as

$$n.t_{res}<t_{diff}<n.t_{res}+t_{res}$$

In theory, any difference between the two signals can be measured by making the resolution as low as possible. However in practice, minimum resolution is limited by factors like mismatch of transistors, delay mismatch due to loading, length of the delay line.

In delay fault testing, the worst case delays in paths could be due to other different types of transitions. The MVDL can handle all possible transitions on a path as follows. Flops with set-reset capability are used to handle such transitions. When the input and output of the *Path Under Test*(PUT) are rising, the MVDL is RESET (i.e., all the flops are made to store a '0') and then the input and output of the PUT are fed into the x and y inputs of the MVDL, respectively. When both are falling, then the flops are RESET and the input and output of the PUT are inverted and fed into the x and y respectively. The first '1' stored in the flops in both the above cases is proportional to the delay of the path.

The above cases hold good only for non-complementing paths, i.e., paths in which an input transition in one direction causes the output to transition in the same direction. However there are two additional cases possible for complementing paths, where a transition on the input causes the output to transition in the opposite direction. For a complementing path, if input is rising and output is falling then we first SET the delay line, i.e., store 1s in all the flops, and subsequently feed in the input and output as it is into x and y of MVDL, respectively. Now all the flops are initially set to '1' and the delay of the path can be indicated by the first flop which latches in a '0'. The second possible case for complementing paths is when the input is falling and the output is rising. Now since the input is being used to clock the flops in the MVDL, we have to feed in the input as a rising edge. So we SET the flops and then complement both input and output before feeding them into x and y of MVDL respectively. Now all the flops initially store a '1' so the path delay is indicated by the presence of the first flop which latches in a '0'.

Implementation of these schemes requires a multiplexer each for the selection of complemented or non-complemented input and output. This does not create any problems with the accuracy of the delay measurement since the MVDL is symmetric and depends only upon the difference between the two delays and inaccuracies can be eliminated by matching the upper and lower delay lines. Table 1 shows a list of possible transitions and proposed schemes to handle them.

Reading out of the values stored in the delay line is a crucial task and most proposed solutions involve tremendous hardware overhead like a separate asynchronous read-out architecture using registers [1]. We have resolved this issue in the MVDL by using a readout scheme which has minimal hardware and pin overhead. As shown in Figure 1, the output of every flop in the MVDL except for the last one is multiplexed with y line input of that stage and fed to the D input of the corresponding flop. The output of the last flop is an output pin. Similarly the x line input of a stage is multiplexed with a clock signal to be fed into the the clock input of the flop of that particular stage. Once we have completed the measurement of delay of a path under test, we switch to the readout mode by asserting the *mode* signal which activates both these multiplexers. Now we can shift out the values in the flops serially using *shiftclk* to obtain the delay of the path under test.

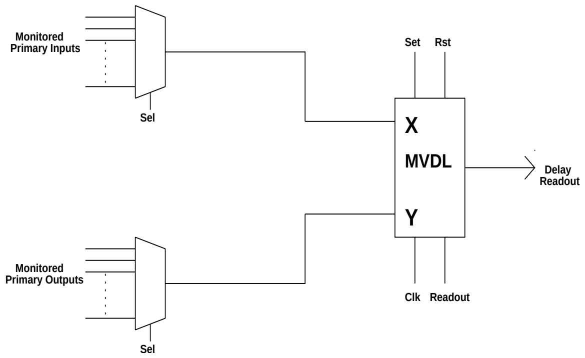

Figure 2: Proposed Delay Measurement and Debug Scheme

3. MVDL FOR DELAY MEASUREMENT AND DEBUG

The proposed debug scheme using the *Modified Vernier Delay Line* (MVDL) is shown in Figure 2. A single MVDL is used per chip to test all modules. Critical paths above a certain designer specified threshold are susceptible to failure and these need to be tested for delay faults. The path delay fault model has been used, but if there are an exponential number of paths then the segment delay fault model can be used. Start and end points of critical paths from different modules on the same chip that are above the specified threshold value of delay are multiplexed into the the MVDL delay measurement unit. Measurement of delay of multiple paths/modules requires a structured debug technique. In [10] a debug interface using IEEE 1149.1 is proposed. We extend this to test and debug multiple modules within the same chip using a single MVDL module. The proposed methodology is outlined below.

- Reset the *Circuit Under Test* (CUT) to a known state.
- Select the particular critical path to be tested from the multiplexed set and feed it into the MVDL for testing.
- SET/RESET the flops in the MVDL as per the path functionality. If the path is a complementing one then we SET the flops, else RESET the flops.
- Activate the MVDL in the delay measurement mode by de-asserting the *mode* and SET/RESET signals.
- Apply the appropriate worst case patterns to the path and allow values to be latched into the flops.
- Assert the *mode* signal to scan out the values stored in the MVDL.
- Count the number of number of indicator values i.e. 1s for non-complementing paths/modules and 0s for complementing paths as be the case.

Table 2: Delay Measurement Results

Test Circuit	Delay obtained with commercial STA tool ps	Delay range with MVDL ps
4 bit Wallace multiplier	2420	2328-2425
4 bit Dadda multiplier	1980	1940-2037
8 bit Wallace multiplier	4320	4268-4365
8 bit Dadda multiplier	3740	3686-3783
16 bit Wallace multiplier	7630	7566-7663
16 bit Dadda multiplier	6870	6790-6887
32 bit Wallace multiplier	14430	14356-14453
32 bit Dadda multiplier	12160	12028-12125

- Calculate path delay from this information.
- Select the next critical path and repeat the above steps for the next critical path.

Once we have obtained the number of indicator values that have been latched onto the MVDL we can calculate delay as follows. Let m be the number of indicator values and N be the total number of stages in the MVDL. Let n be the stage where the first indicator value is latched. Then the path delay is calculated as follows.

$$n = N - m$$

$$n.t_{res} < PathDelay < (n+1).t_{res}$$

The measured values of path delay are then compared with the expected values in order to determine if a delay fault exists on the path. The approximate range of the size of the delay fault can also be obtained as follows.

$$n_e = \frac{ExpectedDelayValue}{t_{res}}$$

and

$$|n_e - n|.t_{res} < FaultSize < |n_e - (n+1)|.t_{res}$$

4. RESULTS

The *Modified Vernier Delay Line* (MVDL) was designed in 0.18 μm CMOS technology. Wallace and Dadda multipliers of different sizes were used as test circuits. We used a commercial static timing analysis tool [9] to extract the top critical path for each circuit as well as the delay of the critical path. Each such critical path was then sensitized for the worst case propagation delay and the range of the delay values was measured using the MVDL. The input and output of each critical path were connected to the x and y inputs of the MVDL respectively. The resolution of the MVDL was set to 97ps for all measurements. Table 2 compares critical path delay measurements obtained using the proposed technique against critical path delays obtained from a commercial timing analysis tool [9] for different test circuits.

The area overhead of the MVDL module is minimal. A 12-20 stage MVDL is sufficient to for detecting a delay faults as well as indicating its size with a resolution of 10% of the cycle time and worst case size of delay faults ranging from 20% to 100% of cycle time. The area overhead of a such a MVDL module for a processor having approximately 200 mm^2 die area is 0.0029% - 0.0047%.

5. CONCLUSION

In this paper we have presented a novel scheme to detect and debug delay faults using on-chip capture of delays in the range of few hundred picoseconds. The scheme also has an efficient read out technique involving minimal pin and area overhead. Path delays and delay faults smaller than 100 picoseconds can be measured using this scheme, thus reducing the time-to-market for high-performance chips. Future work would focus on optimal placement of MVDL modules on-chip for obtaining maximum delay fault coverage.

6. ACKNOWLEDGMENTS

The authors would like to thank Wendy Belluomini and Robert Montoye of IBM Research - Austin Research Laboratory for their helpful discussions and useful insights. We would also like to thank Whitney J. Townsend for helping us design the multiplier test circuits.

7. REFERENCES

[1] P. Dudek, S. Szczepanski, and J. V. Hatfield. A High-Resolution CMOS Time-to-Digital Converter utilizing a Vernier Delay Line. *IEEE Transactions on Solid State Circuits*, 35(2):240–247, February 2000.

[2] M. Favalli, P. Olivo, and M. Damiani. Novel Design for Testability Schemes for CMOS IC's. *IEEE Journal of Solid State Circuits*, 25(5):1239–1246, October 1990.

[3] P. Franco and E. J. McCluskey. Delay Testing of Digital Circuits by Output Waveform Analysis. In *International Test Conference*, pages 798–807. IEEE, October 1991.

[4] K. Raahemifar and M. Ahmadi. Design for Testability Techniques for Detecting Delay Faults in CMOS/BiCMOS Logic Families. *IEEE Transactions on Circuits and Systems -II*, 47(11):1279–1290, November 2000.

[5] T. Rahkonen, J. Kostamovaara, and S. Saynajakangas. CMOS ASIC Devices for the Measurement of Short Time Intervals. In *International Symposium on Circuits and Systems*, pages 1593–1596. IEEE, June 1988.

[6] T. Rahkonen and J. T. Kostamovaara. The Use of CMOS Delay lines for Digitization of Short Time Intervals. *IEEE Journal of Solid State Circuits*, 28(8):887–894, August 1993.

[7] O. Sasaki et al. 1.2ghz GaAs Shift Register IC for Dead-Time-Less TDC Application. *IEEE Transactions on Nuclear Science*, 36(1):512–516, February 1989.

[8] A. Stevens, R. P. Vanberg, J. V. D. Spiegel, and H. H. Williams. A Time-to-Voltage Converter and Analog Memory for Colliding Beam Detectors. *IEEE Journal of Solid State Circuits*, 24(6):1748–1752, December 1989.

[9] Synopsis Inc. *Primetime Reference - Version 2000.11*, November 2000.

[10] B. Vermeulen and S. K. Goel. Design for Debug: Catching Design Errors in Digital Chips. *IEEE Design and Test of Computers*, 19(3):37–45, May-June 2002.