A DFT Technique for Delay Fault Testability and Diagnostics in 32-Bit High Performance CMOS ALUs

Bhaskar Chatterjee, Manoj Sachdev, Ali Keshavarzi[*]

Department of Electrical and Computer Engineering
University of Waterloo, Waterloo, ON, Canada
[*]Circuits Research, Intel Labs
Hillsboro, OR, USA
bhaskar@vlsi.uwaterloo.ca

Abstract

Aggressive technology scaling has been the mainstay of digital CMOS circuit design for the past 30 years. This has resulted in the design of multi-gigahertz microprocessors with unprecedented levels of integration. However, this is posing serious challenges to IC testing and long-term reliability. A major source of failures and test escapes in high performance ICs can be attributed to timing-only parametric failures. In this paper, we implement a DFT technique to detect delay faults in a full custom 32-bit high performance ALU. We present the energy-delay tradeoffs and scaling trends associated with our DFT technique for the 180nm-65nm CMOS technologies. In addition, we demonstrate how this technique can be used to detect delay faults with improved resolution (~60ps for 180nm technology) at relatively low, test mode clock frequencies.

1. Introduction

Modern microprocessors operate at clock frequencies more than 3GHz and have close to 100 million transistors on die. Digital IC performance has tracked Moore's Law and improved by 30% annually. However, the performance of the automatic test equipments (ATE) has improved by only 12% per year. Data from the International Technology Roadmap for Semiconductors 2001 (ITRS'01) [1], shown in Figure 1, demonstrates the discrepancy between ATE edge placement accuracy and circuit under test (CUT) performance. In the 1980s, ATEs typically offered headroom of 5x or more over the DUTs. However, this advantage has now almost disappeared, and as the current trends continue, tester-timing errors are approaching the cycle time of the fastest devices [1]. As a result, at-speed testing is becoming more difficult. Thus, tester inaccuracy along with scaled geometry, and higher device speed is expected to compromise IC yield and quality. Moreover, the higher number of DUT pins, demand for higher ATE accuracy, and larger vector memories are expected to increase the cost of the state-of-the-art ATEs.

Further more, in order to maintain improved DUT performance and achieve higher levels of integration,

supply voltage (V_{DD}), transistor threshold (V_{TH}) and oxide thickness (T_{OX}) are being scaled. This is resulting in a 3-5x increase in the transistor $I_{OFF}/\mu m$ and IC background leakage every technology generation as indicated in Figure 1. Consequently, the total and peak current (power) demand of the circuit under test (CUT) is expected to increase. This is eroding the effectiveness of traditional test techniques like I_{DDQ} and stress testing (burn-in) [2, 3]. As a result, parametric defects that cause timing-only failures as opposed to catastrophic logic failures are becoming more common in deep submicron (DSM) technologies [4, 5]. Such defects are difficult to detect and therefore result in increasing number of test escapes. This trend is posing a serious problem to the long-term reliability of future generation digital ICs.

In this paper, we present a design for testability (DFT) technique that is geared towards the detection of such hard-to-detect defects in high performance digital ICs. Further more, we explore the possibility of using relatively low TEST mode clock frequency to detect such defects. This is expected to reduce the overall test cost, while improving the long-term reliability of high end digital ICs.
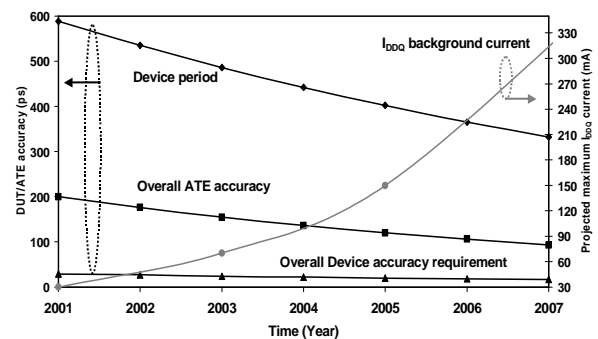

Figure 1: ITRS data for ATE and background leakage

The rest of the paper is organized as follows: in section 2 we discuss the background of high performance circuit testing. Section 3 deals with the basics of our proposed DFT technique. In section 4 we present the design overview of a 32-bit full custom ALU and discuss its

NORMAL mode operation. Section 5 deals with ALU TEST mode operation, while section 6 is for conclusions.

2. High-Performance Circuit Testing Background

VLSI defects are physical deformations caused by missing or extra material and manifest themselves in the form of shorts or opens. Depending on their impact, defects are typically classified as 1) global or 2) local defects. Global defects generally affect large areas on-die or even entire wafers and are normally easier to detect. On the other hand, local defects generally impact a smaller area on die. However, such defects are difficult to detect, and often require rigorous test practices for proper screening. Techniques used to detect IC defects can be broadly categorized as 1) indirect (correlation based) methods and, 2) direct test methods.

Keshavarzi et. al. presented an example of an indirect test technique in [6], where I_{DDQ} test results are correlated with the maximum operating frequency of a 32-bit microprocessor. The fact that shorter channel lengths lead to higher operating frequency and quiescent leakage current forms the basis of this technique. Another methodology proposed by Hao and McClusky [7], is based on the Very Low Voltage Test (VLV) technique where ICs are performance tested at reduced V_{DD}. It was observed that delay faults were more noticeable at a lower V_{DD} and hence easier to detect. However, the VLV technique affects only the transistor delay, while leaving the interconnect delay largely unchanged. In modern microprocessors, interconnects are responsible for an increasingly larger segment of the total delay. Hence, this method's suitability in DSM technologies is being eroded.

There has been an increased focus on direct test techniques which rely on 1) ATEs with improved capabilities/higher frequencies and, 2) DFT and BIST (Built-In Self Test) for improved CUT testability. Some of these methods [8, 9] are based on the incorporation of additional DFT structures and the creation of a low frequency TEST mode. The basic idea is to include an externally controlled, quantifiable delay to enable slow-speed testing. Such techniques are especially suited for combinational circuits bounded by flip-flops. However, these techniques can detect delay faults above a certain minimum value and require the routing of externally available, timing critical clock signals in the TEST mode. In addition, it is difficult to build-in diagnostics to locate a subset of logic gates causing the timing anomalies in large and complex CUTs.

In this paper, we present a DFT technique that can detect delay faults with finer resolution and allows for the lowering of the TEST mode clock frequency. This paper is an extension of the basic circuit level idea originally presented in [10, 16] and demonstrates the applicability of the methodology to a 32-bit full custom ALU design. This is achieved without using any additional external timing critical signals (or pins) while maintaining the NORMAL mode energy-delay penalties within acceptable limits.

3. Circuit Strategy for DSM Digital Testing

Logic circuits implemented using the dynamic CMOS style offer higher performance over their static counterparts. Therefore, the performance critical microprocessor functional unit blocks (FUB) like arithmetic logic units (ALU), and register files (RF) are often implemented using dynamic circuits. Such logic blocks normally have tight timing budgets and are therefore more prone to timing-only failures. In addition, the microprocessor operating frequency is closely tied to the performance of such FUBs and may be adversely affected by the presence of delay faults in such FUBs. Therefore, in this paper we present a DFT strategy geared towards the detection of delay faults in performance critical FUBs that are designed using dynamic logic.

3.1. DFT for Delay Testing in CDL gates

Circuit designers have devised many different logic styles within the domino family in order to maintain high performance while ensuring scalability. In this paper, we focus on the compound domino logic style (CDL) that is used in the design of full-custom digital datapath designs [11, 12, 13]. CDL gates incorporate alternate stages of n-MOS domino and static CMOS logic gates thereby ensuring both improved performance and robustness. In particular, this logic style is used in the design of high performance MPU adders, ALUs, and register files. Figure 2 shows a chain of 7 CDL gates and is representative of the critical path of a 32-bit ALU. In addition, it also shows the DFT structures required to detect delay faults in such a circuit arrangement.
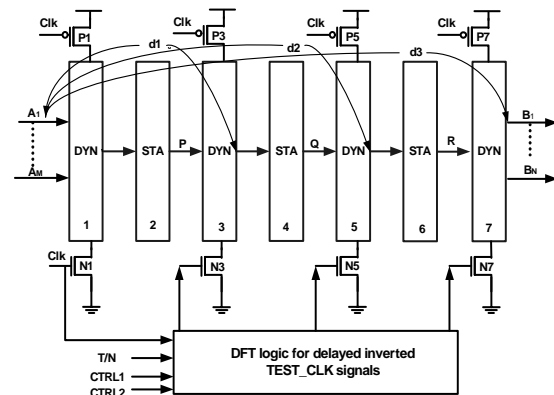

Figure 2: CDL gates with DFT for delay testing

This circuit has 2 modes of operation: 1) NORMAL and 2) TEST. In the NORMAL mode of operation, the mode

control signal T/N is set to logic 0. It is clear from Table 1 that this causes both the signals CTRL1 and CTRL2 to be treated as don't cares. During NORMAL mode operation, the 3 output signals of the DFT logic shown in Figure 2 are set to V_{DD}. As a result, the n-MOS footer transistors (N3, N5, N7) are always ON and allow the circuit to evaluate depending on the vectors applied at the primary logic inputs (A_1-A_M).

In the TEST mode, we create an "evaluation window" for the circuit under test (CUT). This is achieved by applying the system clock signal (CLK) to the first stage of input logic gates and a delayed-inverted clock (TEST_CLK) signal to subsequent logic stages. The "window" duration is equal to the delay between CLK and TEST_CLK signals. We build a safety margin in to this "evaluation window" to account for delay variations caused by process, temperature and voltage fluctuations during testing in order to prevent the rejection of any good parts. The safety margin (~60ps in 0.18μm) is a design parameter, and in this design, it was set to one inverter delay (F.O.=3).

Table1: Truth table for DFT logic and mode selection

T/N	CTRL1	CTRL2	Comment
0	x	x	Normal Mode
1	0	0	Test Section 1 (delay = d1)
1	0	1	Test Section 2 (delay = d2)
1	1	0	Test Section 3 (delay = d3)
1	1	1	Reserved (low power stress testing)

For the case when the CUT is devoid of delay faults, the intermediate nodes (P, Q, R in Figure 2) can evaluate in the available "window". However, when a delay fault is present, circuit evaluation is delayed and signals get pushed out. In case the delay fault is excessive, the CUT fails to evaluate in the available evaluation time. Such a failure can then be detected at the primary outputs (B_1-B_N) as a logic failure. Thus, our DFT technique helps convert delay faults internal to the combinational logic block into readily detectable stuck-at faults observable at the primary outputs. In addition, by setting the CTRL1 and CTRL2 signals appropriately, (Table 1) it is possible to route the TEST_CLK signal to the selected n-MOS footer transistors (N3, N5, N7). This allows us to test a sub-section of the CUT for delay faults using tight evaluation timing while the others are subjected to a more relaxed window. This allows us to trace a logic failure at the CUT primary outputs back to a set of internal gates and helps in creating built-in delay diagnostics.

Another advantage of this DFT technique is the possibility of lowering the TEST mode clock frequency. The evaluation window used to detect delay faults has two edges: 1) opening edge, and 2) closing edge. The system clock provides the opening edge, while the closing edge is obtained locally using the DFT logic. Thus, the detection of delay faults is dependent on the correct phase relationship between CLK and TEST_CLK signals while being independent of their absolute signal frequencies. Hence, this DFT technique can enable delay fault testing at relatively low TEST mode clock frequency using cheaper ATEs. This concept is illustrated with the help of waveforms shown in Figure 3. We show the HSPICE simulations for 0.18μm CDL gates (with DFT) for a variable delay fault. The extent of the delay fault was controlled by introducing a variable resistance in series with the evaluation network of the logic gates. This has the impact of increasing the effective RC time constant and CUT delay [3]. We use a TEST mode clock frequency that is 5x lower than the NORMAL mode of operation. It is clear that when DFT footer transistors are used, the CUT fails when the defect resistance is more than 1.25kOhms. However, in the absence of DFT, the same circuit fails to detect defect resistances of up to 3kOhms.
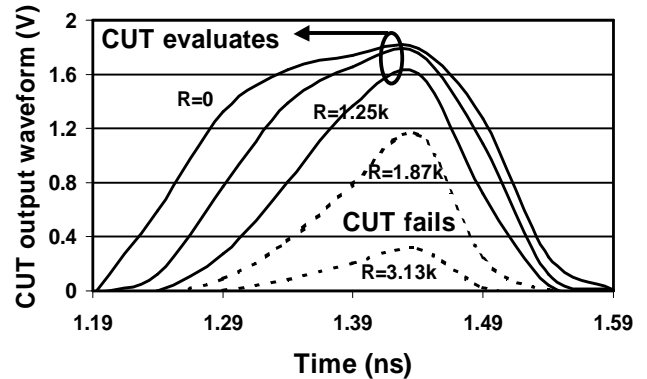

Figure 3: Low frequency delay testing with DFT

3.2. Delay Fault Detection Range

The proposed DFT technique allows us to increase the range of detected defect resistance compared to a non-DFT circuit. This is crucial in high performance DSM circuits and FUBs, and can be better understood with the help of Figures 4(a-b). In Figure 4(a), we show the location of some of the typical resistive defect in the domino logic gates (keeper omitted for clarity) under test. In this study, we considered both the cases when resistive defects were present on the transistor source (R1) and drain terminals (R2). In addition, we considered defects being present in pulldown paths that comprised of single (R1, R2 in series with A) as well as multiple series (R3 in series with B, C) connected n-MOS transistors.
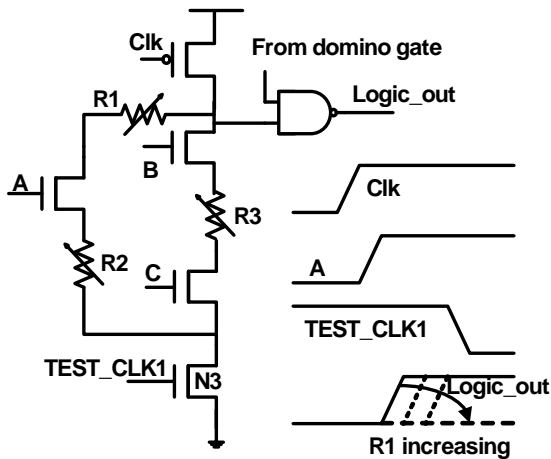
Figure 4(a): Resistive defects: typical location in CUT

For domino circuits with no DFT structures (N3, N5, N7 removed), the circuit has the entire duration when CLK=1 to evaluate. For our specific circuit example shown in Figure 4(b), this duration is about 200ps. As the defect resistance is increased, the CUT evaluation time gets pushed out and fails completely above a value of 3kOhms. However, when DFT is used, the circuit has a smaller evaluation window. Consequently, the CUT fails when the defect resistance is more than 1.25kOhms. However, it should be noted, that even with DFT, a certain range of defects (up to 1.25kOhms) still go undetected. This is because the delay impact of such defects is within the safety margin, and an attempt to detect delay faults with finer resolutions can result in rejection of good parts and yield loss. It should also be noted that the defects in the high resistance range can however (above 3kOhms in this case) always be detected in our example.
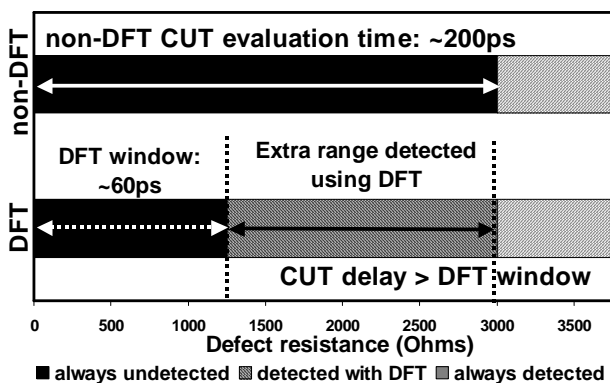

Figure 4(b): Defect resistance detection range DFT vs. non-DFT

Our results demonstrate that the CUT with DFT can consistently detect a larger range of defect resistance. This is clear from the simulation results for the resistance R1, R2, and R3 as shown in Table 2. We considered only one defect being present in the circuit at a given time and observed the defect resistance required for which the CUT begins to fail with and without DFT.

Table2: Defect resistance detection for 0.18μm technology

	With DFT (min. Ω)	No DFT (min. Ω)	Extra resistance range detected with DFT
R1	1.25k	3k	1.25k-3k
R2	3.13k	5k	3.13k-5k
R3	2.5k	3.1k	2.5k-3.1k

4. Design Overview of 32-bit ALU

In this section, we discuss the design of a delay fault testable, 32-bit high performance ALU. This design has two modes of operation, NORMAL and TEST. In the NORMAL mode, the ALU performs arithmetic, logical and shift operations and can also support a low-power mode of operation using a dual-supply scheme. The ALU consists of approximately 11.5k transistors and operates at 1.5GHz for the 0.18μm technology. The ALU performance scales to 4.2GHz under worst-case conditions for the 65nm CMOS technology. The ALU block diagram is shown in Figure 5 and its basic architecture is similar to that presented in [13]. The block diagram indicates that the ALU comprises of several sub-units. The input data stage comprise of master-slave static flip-flops and data drivers for the A[31:0] and B[31:0] busses. The actual instruction executed by the ALU (arithmetic, logical, shift) is determined by the instruction decoder unit. Both the decoder and logic/shift units are non-critical in terms of performance and have relaxed timings. Therefore, the decoder is realized using static CMOS logic, while the logic unit and shifter are implemented using complementary pass transistor logic (CPL) to achieve low power operation. The ALU critical path comprises of the arithmetic unit (adder front-end MUX + 32-bit adder), output MUX-es, and output stage latches. In this design, these units were designed using CDL logic.

In the TEST mode of operation, the DFT logic can be used to perform delay testing on the performance critical units of the ALU. It should be borne in mind that the proposed DFT technique can be integrated with FUBs designed using dynamic logic and is independent of the ALU or its architecture. In this paper, we use the ALU as a vehicle to demonstrate the effectiveness of our proposed test technique in detecting delay faults. The ALU on one hand is performance critical, while on the other, involves a reasonable degree of design complexity and a mix of different circuit design styles. This allows us to quantify the various energy-delay tradeoffs and scaling trends associated with our proposed technique.
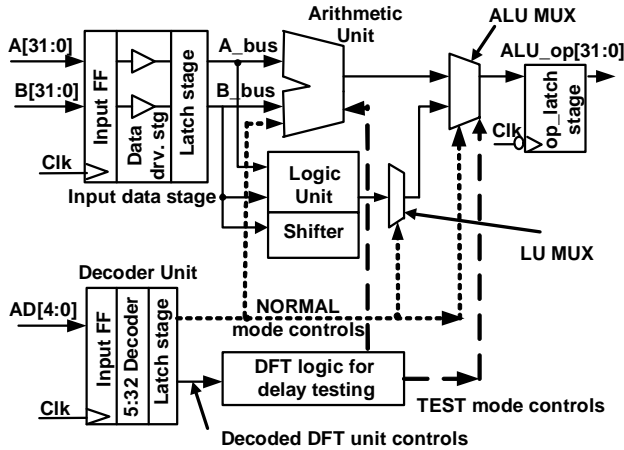
Figure 5: Conceptual block diagram of 32-bit ALU

The DFT logic unit shown in Figure 5 is implemented using static CMOS logic and C^{2}MOS MUX-es. When the input instruction to the ALU indicates that it is in the TEST mode, the T/N signal is set to logic 1. As a result, the NORMAL mode control signals to the ALU are deactivated (logic 0). The decoder is designed such that it allows the arithmetic unit to operate during the TEST mode. In addition, it is possible to select the particular CDL stages within the ALU to be subjected to delay testing. This is indicated by the broken lines in Figure 5, from the output of the DFT logic to the arithmetic unit and the ALU output MUX-es.

4.1. Delay Testing Logic: Implementation

This section discusses the design details of the DFT unit that allows us to generate delayed-inverted TEST_CLK signals for ALU delay fault testing. The primary design objectives for the DFT logic are as follows:

1. TEST_CLK signals should be generated on-chip and locally to the actual CUT logic to be tested,

2. Eliminate the need for additional timing critical input signals to be supplied by the ATE,

3. Minimize any additional clock load due to the DFT logic in NORMAL mode of ALU operation,

4. Minimize the transistor count, additional input pins and design complexity of the DFT logic.

The above considerations allow us to reduce the design overhead and make it easier to integrate the scheme with the overall logic design flow. Further more, it minimizes the additional clock load and reduces the NORMAL mode switching energy penalty and clock skew. We explain the operation of the DFT scheme with the help of Figure 6. The DFT logic comprises of 2 levels of MUX-es and a delay chain implemented using static CMOS inverters.

The input stage MUX-es are connected to system signals, namely CLK, /CLK and the supply V$_{DD}$. The output MUX stage provides the gate control for the n-MOS footer transistors (N3, N5, N7) of the ALU. The transistors of the MUX and inverter chain were sized appropriately in order to obtain the required evaluation window for each ALU section. It should be noted that there are several stages of inversion (act as gain-stages) between the input and output stages of MUX-es. This allows us to use minimum or close to minimum sized transistors for the input MUX stage and reduce the additional load on CLK and /CLK signals.

We used an odd number stages of inverters between the input and output MUX stages in order to obtain TEST_CLK signals that are inverted with respect to the input CLK, /CLK signals. It should be noted that we share a portion of the inverter delay chain between the TEST_CLK1 and TEST_CLK2 signals. This principle can be applied effectively in more complex designs to save transistor count and DFT logic area. The DFT MUX-es were implemented using C^{2}MOS stages as opposed to transmission gate logic. This achieves better drive capability and sharp rise and fall time for the TEST_CLK signals.

It should be noted, that the ALU logic operates on both the clock phases (CLK and /CLK). The input stages of the adder (PG unit) and the Carry Merge Tree evaluate when CLK=1, while the ALU output MUX stage and output drivers evaluate using the negative phase when CLK=0. Therefore, the DFT logic shown in Figure 6 generates 2 of the TEST_CLK signals (TEST_CLK1, TEST_CLK2) that are delayed-inverted with respect to the system clock (CLK) while the TEST_CLK3 signal for the final stage was derived from /CLK. For the DFT unit design, we also ensure that the number of logic inversions on the delay chain equals that of the corresponding CUT section being tested. This helps us to match the delays of the DFT unit and CUT logic stages being tested. For our 0.18μm ALU design example, the TEST_CLK1 and TEST_CLK2 signals were delayed by 230ps, and 390ps with respect to CLK respectively. TEST_CLK3 was delayed by 170ps with respect to /CLK. It should be noted that these delays also include the ~60ps inbuilt safety margins.

In the NORMAL mode, the entire DFT logic is disconnected from the CLK grid via the input MUX that connects both node A and B (Figure 6) to V$_{DD}$. As a result, all the internal nodes of the DFT unit are actively connected to either V$_{DD}$ or ground. This eliminates the possibility of any intermediate node potentials within the DFT logic and excessive leakage currents during NORMAL operation. Further more, the output MUX-es connect the TEST_CLK signals to V$_{DD}$ thereby allowing NORMAL mode ALU operation.
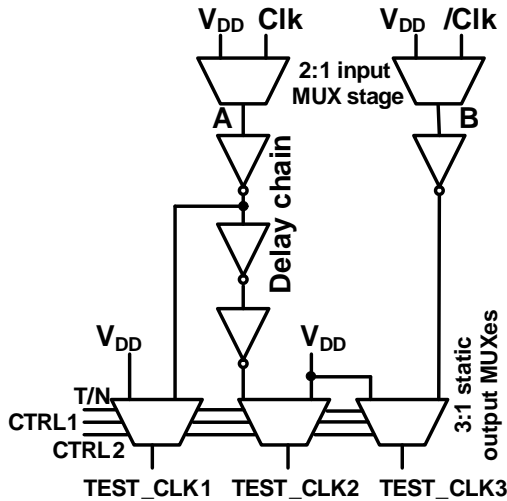
Figure 6: DFT logic for a delay fault testable ALU

In this study, we considered two alternative circuit level implementations for generating delayed-inverted TEST_CLK signals. These schemes are shown in Figure 7(a). Scheme 1 uses a chain of inverters followed by static CMOS NAND gate. In the TEST mode, the control signal from the decoder is set to logic 1, and the delayed clock signal turns the n-MOS footer transistor OFF after a predetermined duration. This scheme is different from that shown in Figure 6, in that it is not a MUX based design. As a result, it does not decouple the delay chain from the input clock signal in the NORMAL mode. This can result in additional clock skew and switching energy consumption.
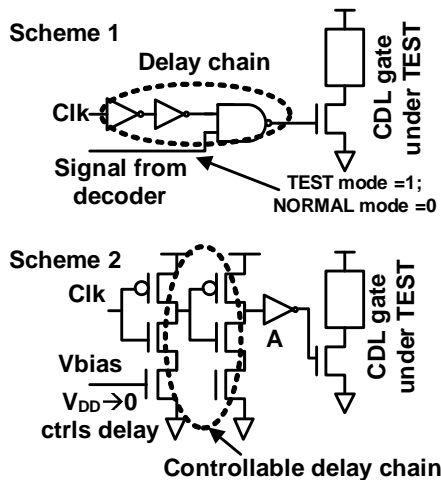

Figure 7(a): Alternate schemes for TEST_CLK generation

Scheme 2 is based on the concept of current-starved inverters. The additional footer transistors on the inverters of the delay-chain are connected to V_{DD} in the NORMAL mode and are fully ON. However, in the TEST mode, the

gate voltage can be connected to an intermediate analog voltage (between V_{DD} and 0V) through an external input pin. The input voltage (Vbias) allows us to control the gate-source overdrive voltage and control the CUT evaluation window. We show the impact of Vbias control voltage on the delay chain in Figure 7(b).
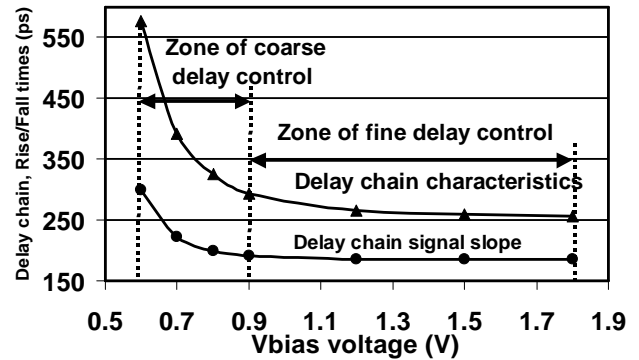

Figure 7(b): Delay control using Vbias voltage (Scheme 2)

Our results indicate that as the Vbias voltage is reduced, the control chain's delay increases and the signal rise/fall times (signal slopes) start to degrade. When the Vbias voltage is in the range between $V_{DD} \rightarrow (V_{DD}-2V_{TH})$, the delay increases in small steps. Thus, this range of Vbias can be used to fine-tune the CUT evaluation window. However, when the Vbias voltage is further lowered (less than $0.5V_{DD}$ for our 0.18μm technology), the delay changes in much larger steps. When the Vbias voltage is in this range, the DFT logic output has degraded rise and fall times.

Typically, the internal signals of high performance CUTs have sharp rise and fall signal slopes. Thus, it is not a good design practice to directly interface the DFT logic output signals (having degraded slopes) with the CUT. This can be mitigated, by allowing the degraded signal(s) to pass through a static CMOS inverter(s) that improves the final signal slope before interfacing with the CUT's footer transistors (inverter A in Scheme 2, Figure 7(b)). This scheme can be used in designs that require more flexibility in the delay margins generated by the DFT logic. However, this design requires access to a controllable external analog voltage, additional input pin and a precise mapping between the input signal voltage and DFT logic delay. Schemes 1 and 2 might be useful in certain applications but for our specific ALU design, we used the scheme enumerated in Figure 6.

4.2. Delay Testable ALU: Energy-Delay Tradeoffs

In this section, we present the simulation results showing the ALU performance and its scaling trends. We also discuss the energy-delay tradeoffs associated with the DFT

technique. Our goal was to devise a DFT strategy for the high performance CUT, while minimizing the NORMAL mode delay and energy penalties. Figure 8 plots the worst-case delay of both the 32-bit adder and ALU, for the 180nm-65nm CMOS technologies. We plot results for both designs with and without DFT. This allows us to quantify the performance impact of the DFT technique on the NORMAL mode operation.
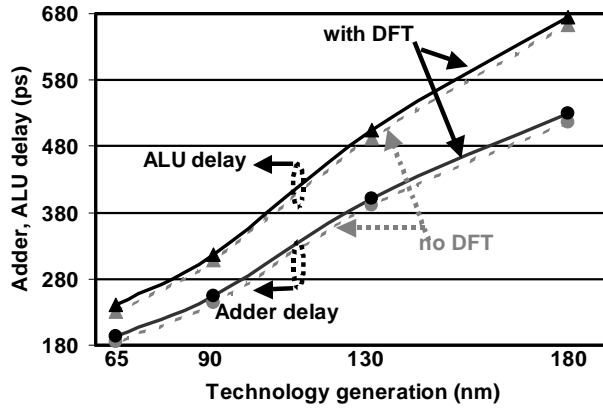

Figure 8: DFT technique: delay impact, scaling trends

The data points in Figure 8 for the 180nm technology correspond to a bulk CMOS TSMC process while the 130nm-65nm results were obtained using the Berkeley Predictive Technology Models [14, 15]. Our results indicate that, for both the adder and ALU, the DFT technique results in delay degradation. This is due to the additional n-MOS footer transistors (N3, N5, N7) inserted in the pulldown paths that increase the stack height and effective ON-state resistance of the evaluation path. However, the delay penalty can be maintained within acceptable limits by observing the following:

1) The footer transistors are added to the dynamic logic gates only, with the alternate static gates left unchanged,

2) In the NORMAL mode, these transistors are connected to V_{DD} and are always ON,

3) Since the DFT transistors do not switch in the NORMAL mode, they can be upsized to minimize delay degradation without significantly increasing switching power.

Our results indicate that the DFT technique results in NORMAL mode delay degradation in the range of 2.7%-4.2% for the adder, and 1.8%-4.4% for the ALU for the 180nm-65nm technologies. In addition, the increase in the NORMAL mode switching energy is limited to less than 1% for the above technologies.

5. ALU TEST Mode Operation

This section deals with the TEST mode operation of the ALU and delay fault detection. In this study, we focused on delay defects existing in the performance critical arithmetic unit and ALU output MUX-es that have tight timing budgets and are hence prone to parametric, timing-only failures. The other units like the logic-shift unit, decoder unit have significantly larger timing margins and hence any timing anomaly in them would be absorbed in the existing slack (unless they are catastrophic failures which is not our focus).
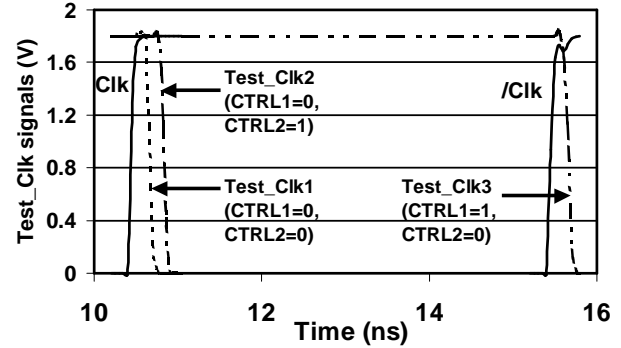

Figure 9: TEST_CLK signals for ALU during delay testing

We adopted a stage-to-stage delay testing strategy, where only a specific ALU stage was under TEST at a given time. This section had a tight evaluation window, while the rest of the ALU had relaxed timing. We carried out the delay testing at a TEST_CLK frequency 5x lower than NORMAL mode of operation. We show the TEST_CLK signals in Figure 9 that were generated by the DFT logic for different CTRL1 and CTRL2 settings (Table 1). When the ALU is in the TEST mode, and both CTRL1 and CTRL 2 signals are equal to logic 0, section 1 is under TEST, and the N3 footer transistor is clocked with TEST_CLK1 signal. This allows us to test the PG unit (propagate-generate) and the first stages of the Carry Merge Tree of the 32-bit adder unit. When TEST_CLK2 is used to control the N5 footer transistor, the rest of the Carry Merge Tree is under test. Finally, when TEST_CLK3 is active, the adder output stage and ALU MUX-es are under test.

We now focus our attention on inserting resistive defects in the ALU and use our DFT test strategy to detect them. We introduced one delay defect at a time in the adder during the course of this study and the possibility of multiple defects being present simultaneously was not explored. The delay defects were introduced in the static gate p-MOS pullup network, and dynamic logic gate pulldown circuitry. The CDL logic precharge operation is non-critical (happens in parallel) and typically has more timing margin than the domino evaluation phase. As a

result, parametric timing anomalies in the precharge network are not of concern in this study.

Table 3: Defect types, distribution, and detection range

Fault No.	Defect Location (Stage No.)	DFT min. detected resistance/ delay fault	Non-DFT min. detected resistance/ delay fault
F1	Stages 1-3 PG unit and Carry-tree (domino and static gate stacks)	1.5k (60ps)	3k
F2			(330ps)
F3			3.5k (330ps)
F4		1k (60ps)	2.5k (330ps)
F5	Stages 4-6 Carry-tree (static/dynamic gate source terminals to power rails)	1k (55ps)	1.5k (160ps)
F6		1k (55ps)	2k (160ps)
F7		0.5k (55ps)	0.5k (60ps)
F8		0.5k (55ps)	0.5k (60ps)
F9	Stages 7-8 Output inverters, MUX pulldowns	1.5k (60ps)	3k (350ps)
F10		2k (60ps)	3.5k (350ps)
F11		3.5k (60ps)	6k (420ps)

In order to conduct a representative study of the effectiveness of our DFT methodology, we introduced 11 unique delay faults in the 32-bit ALU. Table 3 shows the locations and nature of the defects and indicates that they were distributed evenly among the different logic stages. These resistive defects were introduced in the form of parametric resistances in series with the evaluation transistors. Normally for such defects, the delay impact is proportional to their resistance and they can be used to represent resistive metal lines, S-D bridging defects, resistive vias and/or contacts.
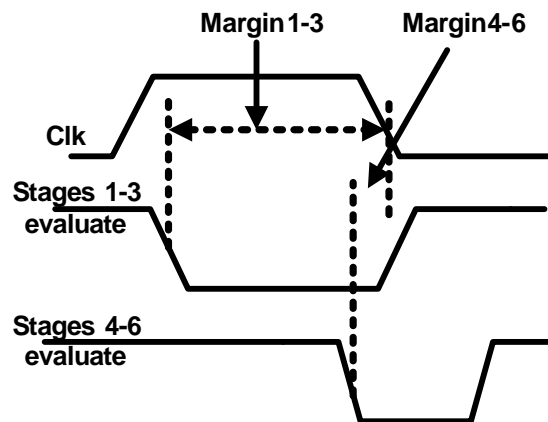

Figure10: Timing diagram showing ALU delay margins

It is clear from the results that the proposed DFT technique can detect a larger range of defect resistance compared to the non-DFT ALU. The resistances shown in Table 3 also map to equivalent circuit delay degradations. Our results indicate that, the DFT ALU can detect faults of magnitude greater than the in-built safety margin (~60ps for 0.18μm technology). Delay faults of smaller resolution however go undetected. It should be noted that for the DFT design, it was possible to lower the TEST mode clock frequency to 200MHz, without compromising the fault detection range.

The results in Table 3 indicate that, for the non-DFT design, a larger range of defect resistance can go undetected. This range is determined by the timing margin between the CUT logic gate with defect, and the CLK edge as shown in Figure 10. In this design, the CDL stages 1-3 evaluate before stages 4-6. As a result, they have more delay margin (Margin 1-3) and larger delay faults (resistance range) can go undetected. However, stages 4-6 evaluate closer to the closing edge of the evaluation phase (CLK 1→0 edge) and have a smaller timing margin (Margin 4-6). Thus, the deeper the logic level, smaller is the timing margin (slack) for the non-DFT design. Consequently, for such gates, the detected resistance range is closer to that of the DFT design. In fact, the faults F7, F8 are in logic stages 5 and 6 that are closest to the CLK 1→0 edge resulting in the same defect detection range as the DFT design. It should be noted that the ALU evaluates on both the clock phases with the logic stages 1-6 evaluating when CLK =1, while stages 7-8 evaluate when CLK=0. This explains the trend in Table 3, where the additional range of detected resistance (delay fault) steadily decreases from stages 1-6 and again picks up for stages 7-8.

5.1 Implementation Issues

Our proposed DFT technique has certain overheads associated with its design and implementation. This scheme requires the designing of a dedicated DFT unit to be activated during the TEST mode. Figure 11 shows the layout for the 32-bit ALU with built-in DFT scheme for delay fault testability. The overall ALU has dimensions of 800μm x 600μm while the DFT unit measures 200μm x 100μm. The DFT unit results in a 1.3% increase in the ALU transistor count with an area penalty of about 4%. Our technique is geared towards performance critical datapath FUBs that are typically full-custom, hand crafted designs. It is therefore expected that the integration of this DFT technique with the logic design flow would not contribute significantly to additional turn around time during layout. In addition, the proposed stage-to-stage testing methodology may result in longer test time or require additional test pattern generation. However, this is an issue that remains the topic of future research and has not been addressed in this current study. Finally, we

adopted a DFT unit design that results in the creation of a fixed, hard-coded evaluation window for the CUT. However, as has been mentioned in this paper, it is possible to design for a delay margin with more flexibility at the expense of extra hardware.
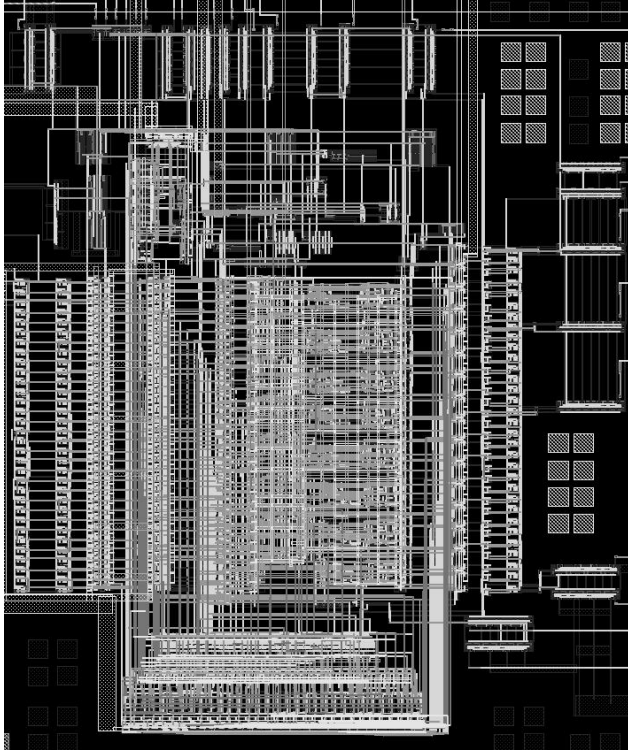

Figure 11: Layout of 32-bit ALU with DFT for delay diagnostics

6. Conclusions

In this paper, we presented a DFT technique that can detect delay faults in a high performance 32-bit ALU design. We integrated this technique with the logic design flow and were able to detect a larger range of delay faults (~60ps for 180nm technology) compared to the non-DT design at a 5x lower test frequency. The delay (energy) penalty associated with this technique was shown to be between 2%-4% (1%) for the 180nm-65nm CMOS technologies. Further more, we demonstrated how this method can be used to convert delay faults into easy to detect stuck-at logic failures and build-in delay diagnostics using the stage-to-stage testing strategy. It is expected that this technique will help in improving delay fault detection and ensuring long term reliability of high end digital ICs.

Acknowledgment

The authors would like to thank R. Krishnamurthy, Intel Corp. for his discussions and suggestions on high performance ALU design, O. Semenov, S. Naraghi and C. Kwong, University of Waterloo for their insights on low frequency testing.

References

[1] Semiconductor Industry Associations, "International Technology Roadmap for Semiconductors, 2001Edition", 2001.

[2] M. Sachdev, "Current-based Testing of Deep-submicron VLSIs", *IEEE Design and Test of Computers*, vol. 18, no. 2, pp. 76-84, Mar.-Apr. 2001.

[3] Z. Cheng, L. Wei, and K. Roy, "On Effective I_{DDQ} Testing of Low Voltage CMOS Circuits Using Leakage Control Techniques*", Proceedings of the IEEE International Symposium on Quality Electronic Design*, pp. 181-188, 2000.

[4] W. Needham, C. Prunty, and E.H. Yeoh, "High Volume Microprocessor Test Escapes, An Analysis of Defects Our Tests are Missing", *Proceedings of International Test Conference,* pp. 25-34, 1998.

[5] P. Nigh, W. Needham, K. Butler, P. Maxwell, R. Aitken, and W. Maly "So What is an Optimal Test Mix? A Discussion of the Sematech Methods Experiment", *Proceedings of International Test Conference*, pp. 1037-1038, 1997.

[6] A. Keshavarzi, K. Roy, and C.F. Hawkins, "Intrinsic Leakage in Low Power Deep Submicron CMOS ICs", *Proceedings of International Test Conference,* pp. 146-155, 1997.

[7] H. Hao, and E.J. Mc Cluskey, "Very-Low Voltage Testing for Weak CMOS Logic ICs", *Proceedings of International Test Conference*, pp. 275-284, 1993.

[8] V. D. Agrawal, T. J. Chakraborty, "High-Performance Circuit Testing with Slow-Speed Tester", *Proceedings of International Test Conference,* pp. 302-310, 1995.

[9] M. Shashaani, and M. Sachdev, "A DFT Technique for High Performance Circuit Testing", *Proceedings of International Test Conference*, pp. 267-285, 1999.

[10] B. Chatterjee, M. Sachdev, and A. Keshavarzi, "A DFT Technique for Low Frequency Delay Fault Testing in High Performance Digital Circuits", *Proceedings of International Test Conference,* pp. 1130-1139, 2002.

[11] K. Bernstein, K. Carrig, C. Durham, P. Hansen, D. Hogenmiller, E. Nowak, and N. Rohrer. *High Speed CMOS Design Styles*. Boston, MA: Kluwer Academic Publishers, 1999.

[12] J. Park, H. Ngo, J. Silberman, and S. Dhong, "470ps 64-bit Parallel Adder*", Proc. of the IEEE Symposium on VLSI Circuits*, pp. 192-193, 2000.

[13] S. Matthew, R. Krishnamurthy, M. Anders, R. Rios, K. Mistry, and K. Soumyanath, "Sub-500ps 64-b ALUs in 0.18μm SOI/Bulk CMOS: Design and Scaling Trends", *IEEE Journal of Solid-State Circuits*, vol. 36, no. 11, pp. 1636-1646, Nov. 2001.

[14] *http://www-device.eecs.berkeley.edu/~ptm*: BSIM3 130nm. 90nm and 65nm predictive technology process files.

[15] Y. Cao, T. Sato, D. Sylvester, M. Orshansky, and C. Hu, "New paradigm of predictive MOSFET and interconnect modeling for early circuit design," *Proceedings of IEEE CICC*, pp. 201-204, Jun. 2000.

[16] B. Chatterjee, M. Sachdev, and A. Keshavarzi, "DFT for Delay Fault Testing of High Performance Digital Circuits", *IEEE Design and Test of Computers,* vol. 21. no. 3, pp. 248-258, May-June 2004.