DFT for Delay Fault Testing of High-Performance Digital Circuits

Bhaskar Chatterjee and Manoj Sachdev
University of Waterloo

Ali Keshavarzi
Intel Laboratories

Timing-only parametric defects are a major source of failures and test escapes in modern ICs. A DFT technique using compound domino logic gates with footer transistors uncovers these hard-to-detect defects with minimal performance and power overheads.

AGGRESSIVE TECHNOLOGY SCALING has been the mainstay of digital CMOS circuit design for the past 30 years. It has resulted in the design of multigigahertz microprocessors and unprecedented levels of integration. Modern microprocessors operate at clock frequencies of more than 3 GHz, and their dies contain close to 100 million transistors. Digital IC performance has followed Moore's law, improving annually by 30%. However, ATE performance has improved by only 12% annually. The discrepancy between ATE edge placement accuracy and circuit under test (CUT) performance[1] will make at-speed logic testing increasingly difficult for future deep-submicron technologies.

As the trend toward improved performance and greater integration continues, supply voltage (V_{DD}), transistor threshold voltage (V_{TH}), and oxide thickness (T_{OX}) scale further. This scaling causes a three- to fourfold increase in transistor off-state leakage current I_{OFF} per micron and IC background leakage for every technology generation.[1] Consequently, the deep-submicron regime is eroding the effectiveness of current-based (I_{DDQ}) test techniques and stress testing (burn-in).[2,3]

In addition, the number of parametric defects that cause timing-only failures as opposed to catastrophic failures is increasing.[4,5] These difficult-to-detect defects are causing an increasing number of test escapes. This poses a serious obstacle to the long-term reliability of future digital ICs.

These problems have motivated us to investigate DFT techniques that augment existing test strategies and detect delay faults. Here, we present a DFT technique for high-performance digital circuits that provides delay fault detection and lowers test mode clock frequency.

High-performance-circuit testing

VLSI circuit defects are physical deformations caused by missing or extra material, which manifest themselves as shorts or opens. Depending on their impact, defects are typically classified as global or local. Global defects affect large die areas or even entire wafers and are normally easier to detect. Local defects affect a smaller die area but are difficult to detect and often make rigorous test practices necessary for proper screening.

We broadly categorize techniques for detecting IC defects as indirect (correlation based) and direct. One indirect test technique correlates I_{DDQ} test results with the maximum operating frequency of a 32-bit microprocessor.[6] The basis of this technique is that shorter channel lengths lead to higher operating frequency and higher quiescent leakage current. Another approach is the very low voltage (VLV) test technique, in which ICs are performance tested at a reduced V_{DD}.[5] The basis of this technique is that delay faults are more noticeable at a lower V_{DD} and hence are easier to detect. However, aggressive scaling of V_{DD}, V_{TH}, and T_{OX} are increasing IC background leakage.[2,3,5] As a result, the VLV and I_{DDQ} techniques are both becoming less effective with technology scaling. The increased delay caused by interconnects in deep-submicron technologies also reduces VLV effectiveness.

Two types of direct test techniques are receiving increasing attention: those that rely on ATE with

0740-7475/04/$20.00 © 2004 IEEE Copublished by the IEEE CS and the IEEE CASS

improved capabilities and higher frequencies, and those that rely on DFT and BIST for improved circuit testability. Some of these methods require incorporating additional DFT structures and creating a low-frequency test mode.[7] 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 detect delay faults only above a certain minimum value, and they require the routing of an externally available, timing-critical clock signal in test mode.

The technique we present here detects delay faults with improved resolution in high-performance combinational logic circuits.[8] In addition, our technique lowers the test mode clock frequency to allow the use of cheaper testers. It achieves these goals without using an additional external timing signal and with minimal performance and power penalties.

Dynamic logic and high-performance functional unit blocks

Logic circuits implemented in the dynamic CMOS style offer higher performance than their static counterparts. Therefore, performance-critical functional unit blocks (FUBs) such as ALUs and register files often use dynamic circuits. The microprocessor's operating frequency is closely tied to FUB performance and can be adversely affected by the presence of delay faults in the FUBs. However, most performance-noncritical FUBs have more timing slack and thus normally use static logic to reduce switching activity and power consumption. Noncritical FUBs have greater timing budgets, so microprocessor performance is less prone to degradation when delay faults are present in these FUBs. Therefore, we geared our DFT strategy toward delay fault detection in dynamic circuits and performance-critical FUBs.

The dynamic-logic family includes several circuit styles.[9] One style prevalent in the design of high-performance data path elements is compound domino logic. As Figure 1 shows, CDL implements complex logic functions using alternate stages of dynamic and static CMOS gates. This circuit style uses dynamic gates with NMOS pull-down stacks followed by static CMOS NAND/NOR gates.

CDL gates offer the improved performance of dynamic gates, and the intermediate static CMOS gates improve overall noise immunity and DC robustness.[9,10] Furthermore, the CDL circuit style eliminates explicit inverters on the critical path as required by NMOS dynamic logic, thereby improving overall performance.

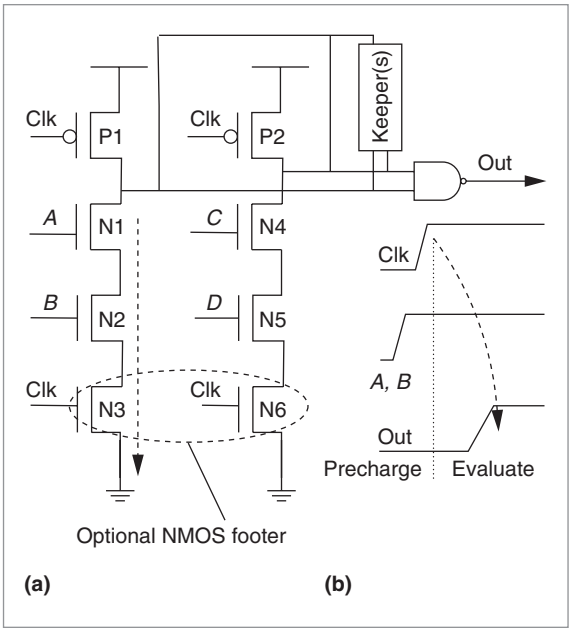

Figure 1. Generic compound domino logic gates: circuit diagram (a); timing diagram (b).

On the other hand, n-p domino logic improves performance but has a greater clock load, poorer noise immunity, and higher area overhead. The output of the CDL gate labeled Out in Figure 1 is logic low during precharge (Clock = 0). As a result, we can directly interface the gate output with the next stage of NMOS pull-down domino gates. This lets us completely remove the footer NMOS clocked transistor from all subsequent dynamic logic gates, improving circuit performance and reducing overall clock load and area.

In this article, we compare two designs:

- Design 1 uses CDL gates amenable to high performance with footer transistors removed.
- Design 2 uses CDL gates with footer transistors for DFT. This design allows delay fault testing but involves power and delay overheads.

CDL gates and delay testing

For Design 2, we create two operation modes, normal and test, for a chain of cascaded CDL gates. Figure 2 illustrates the CDL gates and the operation modes.

Normal mode. In this mode, DC control signals T/N (test/normal), Ctrl1, and Ctrl2, in conjunction with the DFT logic, connect the gate terminals of NMOS transistors N3, N5, and N7 to V_{DD}. As a result, these transistors are always on, allowing circuit operation and evalua-

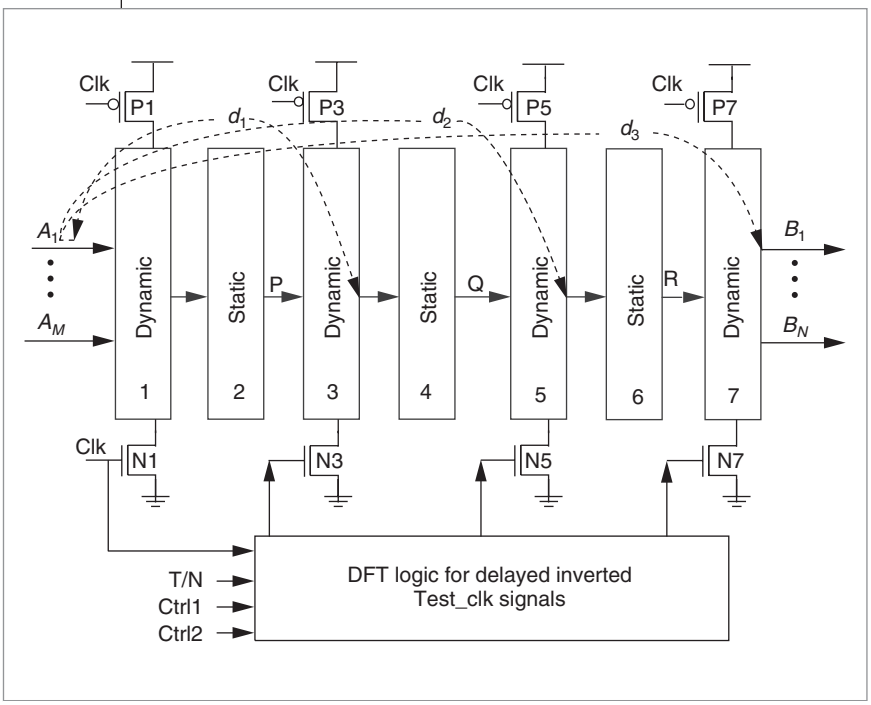

Figure 2. CDL gates and modes of operation.

tion depending on the vectors applied at primary logic inputs A_1 through A_M.

Test mode. In this mode, we create an evaluation window for the CUT. We do this by applying the system clock signal (Clk) to the first stage of input logic gates and a delayed inverted clock signal (Test_clk) to subsequent logic stages under test. The window duration equals the delay between the Clk and Test_clk signals. We allow a 20% safety margin when applying the delayed clock signals. This helps us account for delay variations caused by process, temperature, and voltage fluctuations during testing and prevents the rejection of any good parts.

If the CUT is devoid of delay faults, intermediate nodes P, Q, and R can evaluate in the available window. However, when a delay fault is present, the CUT evaluation is pushed out. If the delay fault is excessive, the CUT fails to evaluate in the available window. Such a failure is then detectable at primary outputs B_1 through 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.

Figure 3 helps illustrate this concept. It shows HSpice simulations for 0.13-micron CDL gates with and without DFT for a variable-delay fault. We varied the delay fault's extent by introducing a variable resistance in series with the logic gates' evaluation network. This increased the effective circuit RC time constant and CUT delay.[3]

Clearly, without DFT, the circuit has 50% of the cycle time for evaluation. As a result, the CUT fails only when the series resistance is greater than 4 kΩ, and a large range of delay faults can go undetected. With DFT, however, the CUT fails when the resistance is more than 0.5 kΩ; thus, DFT lets us detect a greater range of delay faults.

Fault resolution and lower test frequency

We use our DFT scheme on the chain of cascaded CDL gates shown in Figure 2. We divide these gates into three separate sections and test them independently for the presence of delay faults. For example, when section 1 (gates 1, 2, and 3) is under test, sections 2 and 3 are not. The nominal logic delay for section 1 is d_1, and the corresponding delays for sections 2 and 3 are d_2 and d_3. Thus, we apply Test_clk with an evaluation window equal to $1.2d_1$ to the gate of transistor N3. This lets us incorporate a 20% safety margin for section 1. When section 1 is under test, the system clock, using a relaxed evaluation window, clocks NMOS transistors N5 and N7. Any logic failure at the primary outputs under these test conditions can be attributed to section 1's logic gates. This strategy lets us detect delay faults with a finer resolution and trace them to a subset of logic gates within the FUB, thereby helping delay fault diagnostics.

Another advantage of our DFT technique is the possibility of lowering the test mode clock frequency. The evaluation window used to detect delay faults has two edges: The system clock provides the opening edge, and DFT logic from the Test_clk generates the closing edge locally.

Thus, detecting delay faults with this technique depends on the correct phase relationship and delay between the Clk and Test_clk signals but is independent of their absolute signal frequencies. Figure 4 indicates that this technique can detect delay faults even when the test mode clock frequency is lowered to 170 MHz. Hence, the technique might enable delay fault testing at the wafer sort stage with relatively cheaper ATE.

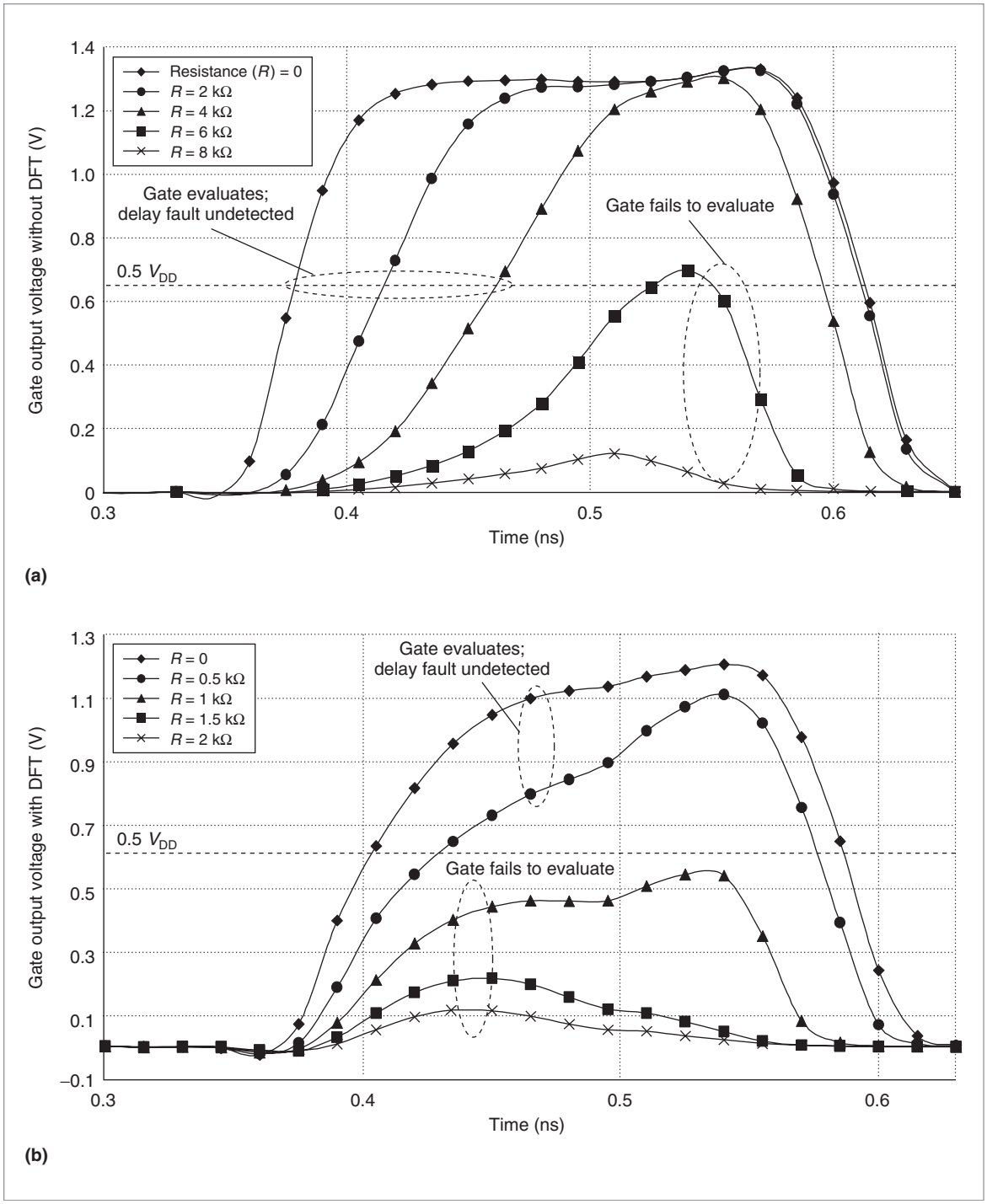

Figure 3. Delay fault simulation: Design 1, without DFT (a); and Design 2, with DFT (b).

High-performance delay-fault-testable adder

We've used the circuit design techniques and DFT ideas discussed here to develop a 0.13-micron, 16-bit high-performance adder that is delay-fault testable. The adder is an ideal testbed for our DFT scheme because its design involves different circuit styles, including dynamic, static, and transmission gates, and it has a reasonably high transistor count and complexity level. In addition, VLSI adders are performance-critical FUBs. Therefore, designers usually use a full-custom approach for these adders to ensure high-performance, low-power, and robust operation.

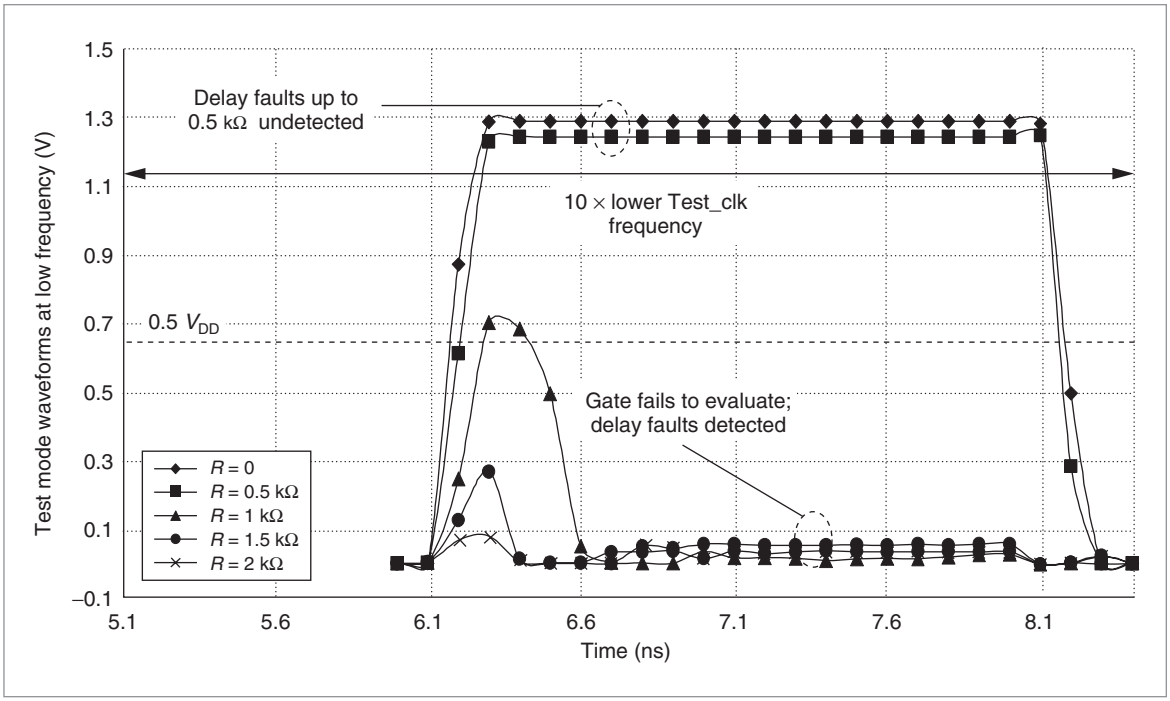

Figure 4. Delay fault detection with a test clock frequency of 170 MHz.

Figure 5 shows the basic adder architecture, which is similar to that discussed by Matthew et al.[10] The propagate-generate block and the carry-merge tree form the adder's carry-generate section; the carry-select adders (CSAs) and the output multiplexers form the sum-generate section.

We implemented the propagate-generate block, the carry-merge tree, and the output muxes, which are performance-critical units, with CDL gates. The carry-select adders, which are implemented with static CMOS gates, operate in parallel with the carry-generate section. The propagate-generate block forms the propagate (P) and generate (G) terms based on primary logic inputs $A[15:0]$ and $B[15:0]$ (Equations 1 and 2). The carry-merge tree employs a binary merge algorithm, which implements the recursive logic equation (Equation 3), to produce the 1-in-4 carry signals C_3, C_7, C_{11}, and C_{15} shown in Figure 5. The basic logic equations for the adder are

$$P_i = A_i + B_i \text{ (propagate function)} \quad (1)$$

$$G_i = A_i B_i \text{ (generate function)} \quad (2)$$

$$C_i = G_i + P_i C_{i-1} \text{ (carry-merge function)} \quad (3)$$

where A_i and B_i represent the ith bit position of vector inputs $A[15:0]$ and $B[15:0]$, P_i and G_i are the corresponding propagate and generate signals, and C_{i-1} and C_i are the input and output carry signals for the ith bit position.

For each 4-bit adder block (blocks A, B, C, and D) in Figure 5, there are two parallel 4-bit carry-select adders. One of these adders generates sum outputs, assuming the input carry to be logic 0; the other assumes the input carry to be logic 1. For example, the carry-select adders pertaining to block C generate two sets of sum signals $S11{:}8^{(1)}$ and $S11{:}8^{(0)}$, using the corresponding block input carry C_{in}^{c}, to be logic 1 and logic 0 respectively. This occurs in parallel with the two sets of sum generation in the carry-merge tree, and the carry-select adder output signals become available at the inputs of the 2:1 muxes. When the C_7 signal becomes valid, the appropriate sum signals are selected and become available at the adder's $S11{:}8$ primary outputs.

DFT logic for delay testing

In designing the 16-bit adder's DFT logic, our objectives were to

- generate the delayed inverted Test_clk signals locally for delay fault detection,
- eliminate the need for additional timing-critical signals supplied by the ATE,
- minimize any clock load or power penalty (switch-

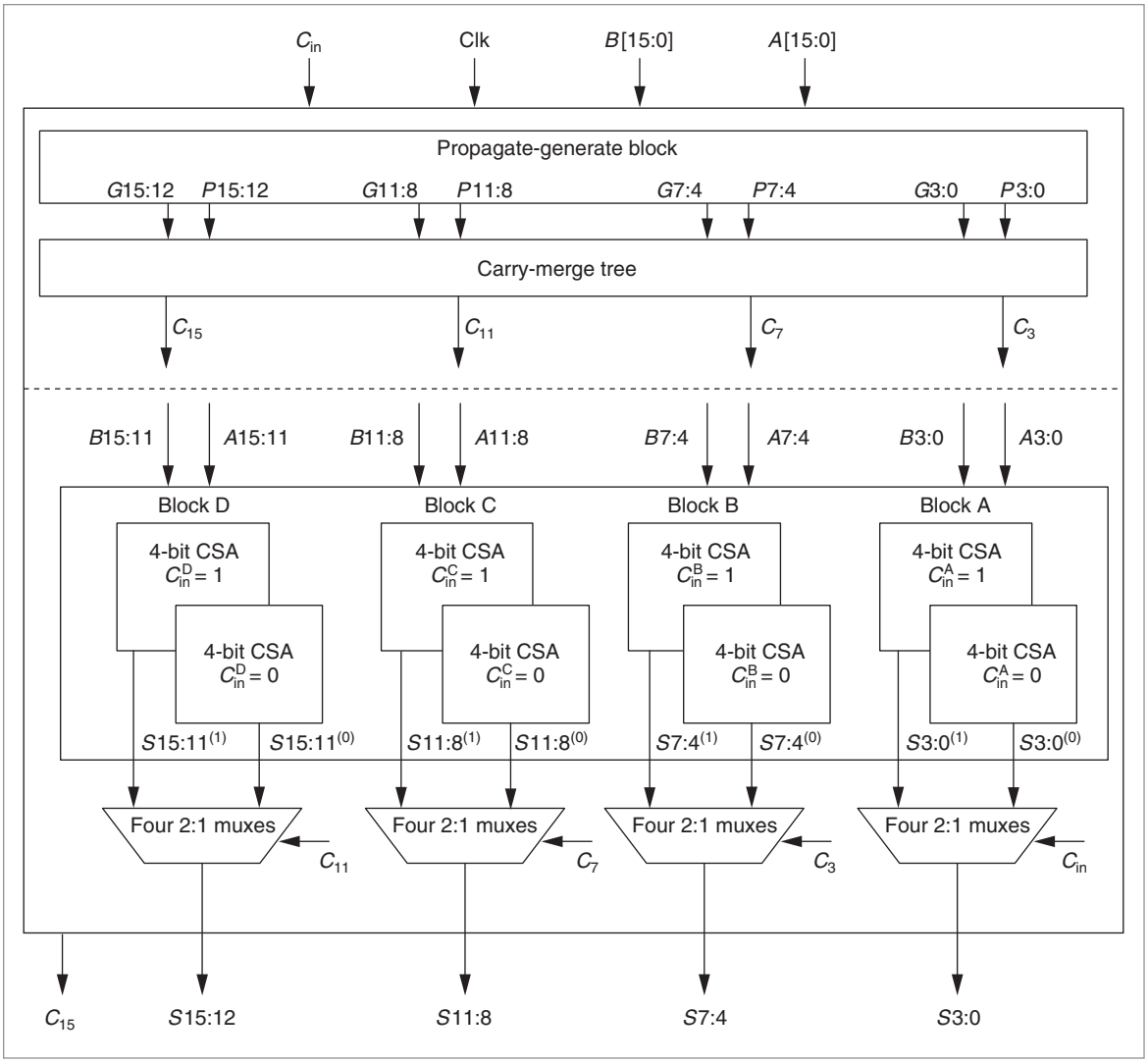

Figure 5. Delay-fault-testable 0.13-micron, 16-bit adder.

ing and leakage) caused by DFT logic in the adder's normal operation mode, and

- minimize the DFT logic's transistor count and design complexity.

The adder's critical path comprises seven stages of CDL gates similar to the chain of gates shown in Figure 2. The critical path contains four NMOS domino gates and three intermediate static CMOS logic gates. To make the adder delay-fault testable, we divided the adder into three sections: 1) the input propagate-generate units, 2) the intermediate carry-merge units, and 3) the output muxes. We introduced NMOS footer transistors at three dynamic gates and designed the DFT logic to generate three delayed inverted signals with respect to the system clock (Clk).

Figure 6 helps explain the DFT scheme's operation. The DFT logic uses high-V_{TH} transistors to minimize leakage during normal operation. It has two levels of muxes and a chain of static CMOS inverters that acts as a delay chain. We appropriately sized the mux and inverter chain transistors to obtain the required evaluation window for delay testing of each adder section. A more generic DFT scheme with a programmable evaluation window is possible, but such a scheme would require greater hardware overhead.

The DFT logic requires DC control signals T/N, Ctrl1, and Ctrl2 to switch between normal and test modes and select the adder section to be tested. Table 1 shows the truth table for DFT logic and mode selection. To achieve better drive capability and sharp rise and fall times for the DFT logic's output signals, we implemented the

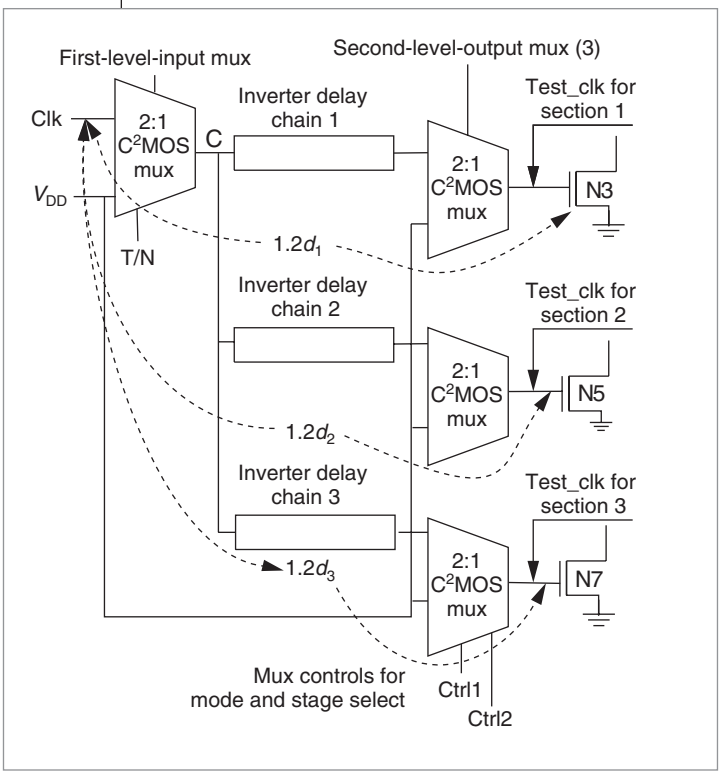

Figure 6. DFT logic for delay-fault-testable adder.

Table 1. Truth table for DFT logic and mode selection.

Control signal			
T/N	**Ctrl1**	**Ctrl2**	**Selection**
0	X	X	Normal mode
1	0	0	Test section 1 (delay = d_1)
1	0	1	Test section 2 (delay = d_2)
1	1	0	Test section 3 (delay = d_3)
1	1	1	Reserved for 32-bit adder

muxes with C²MOS stages rather than transmission gate logic. Furthermore, to minimize switching energy overhead during normal operation, the input first-level mux uses minimum-size transistors.

Figure 7 shows the DFT logic output for different combinations of the Ctrl1 and Ctrl2 signals during the DFT adder's test mode. The control signals for the second-level muxes derive from DC signals Ctrl1 and Ctrl2. Because these signals are noncritical, we generate the mux controls using two- and three-input static CMOS NAND/NOR logic gates with minimum-size transistors. This minimizes the DFT logic's leakage power and area overheads. In addition, when the CUT is in normal mode, the input mux selects supply voltage V_{DD}. As a result, node C in Figure 6 connects to V_{DD}, thereby ensuring that all internal nodes are actively held to either the supply voltage or ground. This eliminates the possibility of intermediate node potentials within the DFT logic or excessive leakage currents during normal operation.

Adder performance and tradeoffs

In quantifying the proposed DFT technique's effect on the CDL adder's performance, we consider the following parameters: adder delay, worst-case switching power, unity gain noise margin (UGNM), and worst-case DC droop.[10,11] The UGNM and steady-state DC droop for the

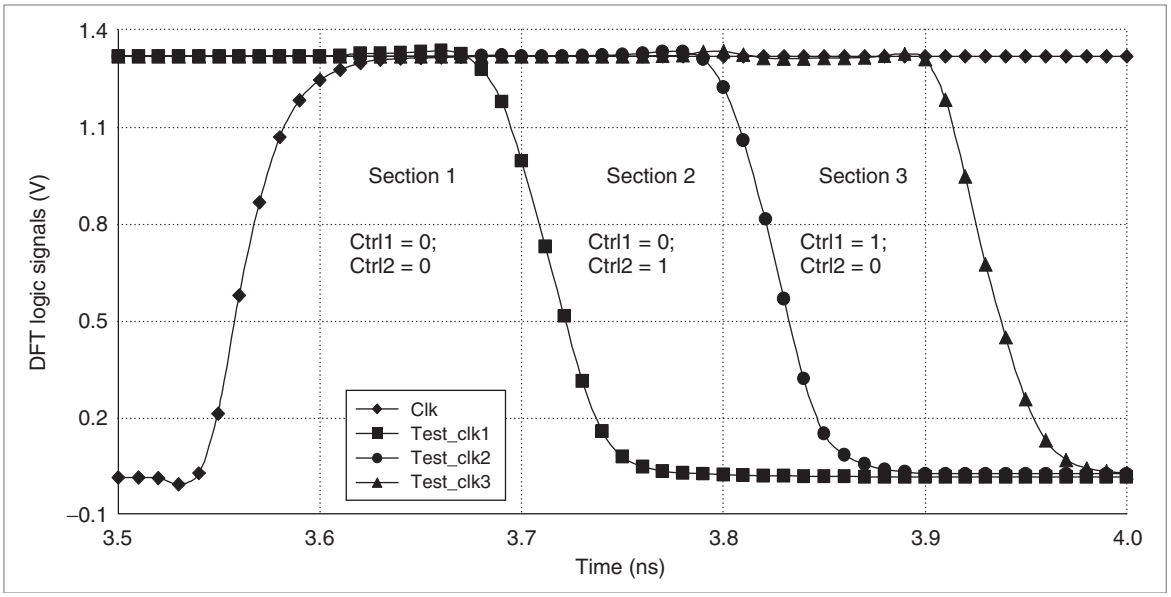

Figure 7. DFT logic outputs in test mode.

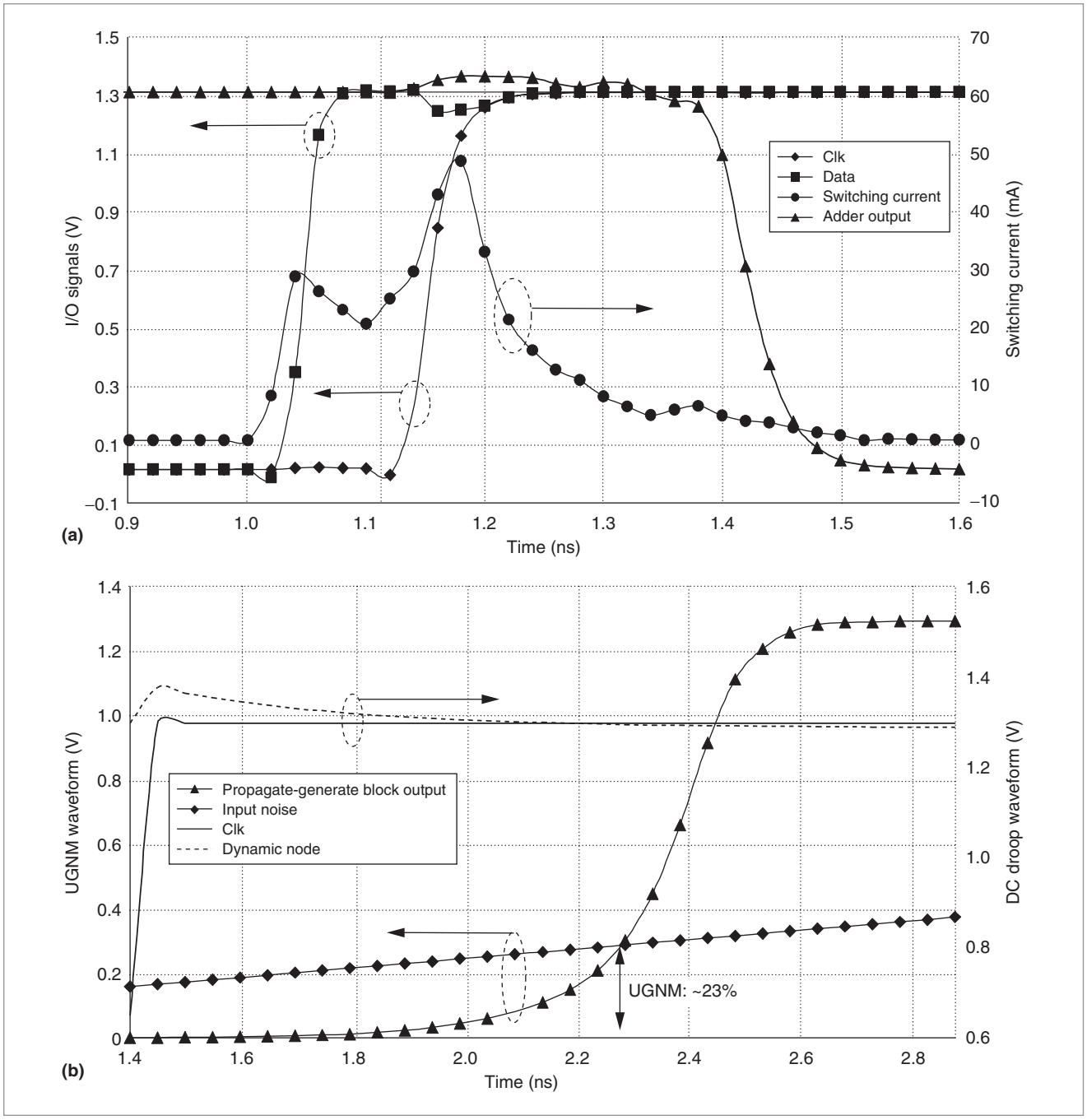

Figure 8. DFT adder performance parameters: current and voltage waveforms in normal mode (a); unity gain noise margin (UGNM) and DC droop waveforms in test mode (b).

dynamic gates are of particular interest during delay testing at a reduced clock frequency in test mode. Figure 8a shows the DFT adder's worst-case switching current and delay waveforms in normal mode. Figure 8b shows the UGNM and worst-case DC droop waveforms in test mode.

Our most important design consideration was to minimize our DFT scheme's overall delay impact. Adding footer NMOS transistors to the dynamic-gate pull-down stacks causes delay degradation. However, the delay penalty remains within acceptable limits for the following reasons:

- We add the footer transistors to only three dynamic gates out of the total seven cascaded CDL gates on all adder paths (Figure 2).

Table 2. Performance parameters: DFT versus non-DFT adders.

Adder design	Delay (ps)	Switching power (mW)	DC droop (mV)	Unity gain noise margin (%)
Non-DFT	263	5.5	5	23.5
DFT	272	5.6	9	23

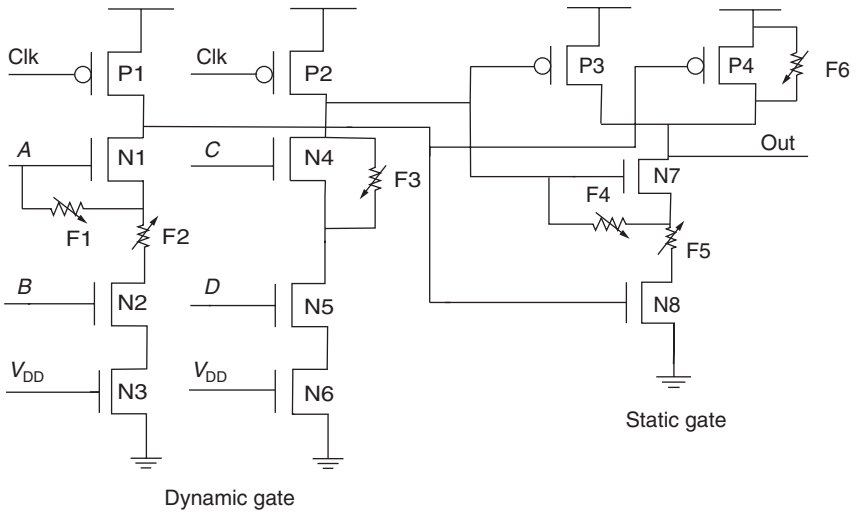

Figure 9. Locations of defects introduced in CDL gates.

Table 3. Defect types and distribution in the 16-bit adder.

Defect no.	Fault no.	Defect location (stage no.)	Defect type
1, 2, 3	F1	1	Parallel
4, 5, 6, 7	F2	2	Parallel
8	F3	3	Parallel
9, 10	F4	4	Parallel
11	F5	4	Series
12, 13	F6	5	Parallel
14	F7	6	Parallel
15, 16, 17, 18	F8	2	Parallel
19, 20	F9	7	Parallel
21, 22	F10	2	Series

- In normal mode, these transistors connect to V_{DD} and are always on.
- Since the DFT transistors don't switch in normal mode, we can upsize them to improve adder performance without significantly increasing switching power.

Our DFT scheme allows delay fault detection in the CUT's critical as well as noncritical paths. The additional footer transistors on the noncritical paths cause extra delay, which the existing timing slack absorbs. However, the DFT structures introduced on the critical path cause increased delay and performance degradation. Table 2 compares the parameters of interest for the adder with and without DFT. Our results indicate that adder delay increases by 3.4%, and worst-case switching energy increases by 1.8%. The UGNM and DC droop degradations are minimal.

In addition, the DFT logic itself causes a 4% increase in total transistor count. Because the DFT control signals are DC-level signals and the additional logic does not switch in normal mode, we can downsize a significant number of DFT logic transistors. This limits the total area penalty to within 5% to 7%. In fact, as the CUT grows in complexity, the overall area overhead decreases because part of the DFT logic delay chain and input mux stages can be reused and need not be repeated. These results show that it is possible to design a delay-fault-testable adder that operates without excessive performance overhead in normal mode.

Low-frequency test results

We used our DFT technique to detect delay faults in the 16-bit adder. We used the T/N signal to select test mode. Next, using the Ctrl1 and Ctrl2 signals, we tested different sections of the adder for delay faults. We lowered the test mode clock frequency to 170 MHz. During the course of this study, we introduced one delay defect at a time in the adder and didn't explore the possibility of multiple defects being present simultaneously. We introduced delay defects in the adder's static and dynamic logic gates, as well as its critical and noncritical paths. Figure 9 shows the locations of some representative resistive defects.

To conduct a representative study of our DFT technique's effectiveness, we introduced 22 defects in the 16-bit adder. However, some of these defects mapped to the same type of fault, resulting in a total of 10 unique delay faults. Table 3 lists the locations of the defects,

which were distributed among the various logic stages and were both series and parallel types. We refer to resistive defects in series with the transistor evaluation path as series defects. Normally, for series defects, the delay impact is proportional to their resistance, and they can represent resistive vias and contacts. Parallel defects, on the other hand, usually manifest as delay faults at lower resistance values.

Figure 10 shows the delay fault simulation results for the 10 different faults. Clearly, we can detect a larger range of delay faults in the DFT adder than in the non-DFT adder. The DFT adder detects delay faults of magnitudes greater than 19 ps, 35 ps, and 46 ps for adder sections 1, 2, and 3 respectively. Delay faults of smaller resolution go undetected. However, for the non-DFT adder, delay faults of up to 247 ps, 156 ps, and 93 ps for the corresponding sections go undetected. In addition, we can lower the test mode clock frequency without compromising the fault detection range.

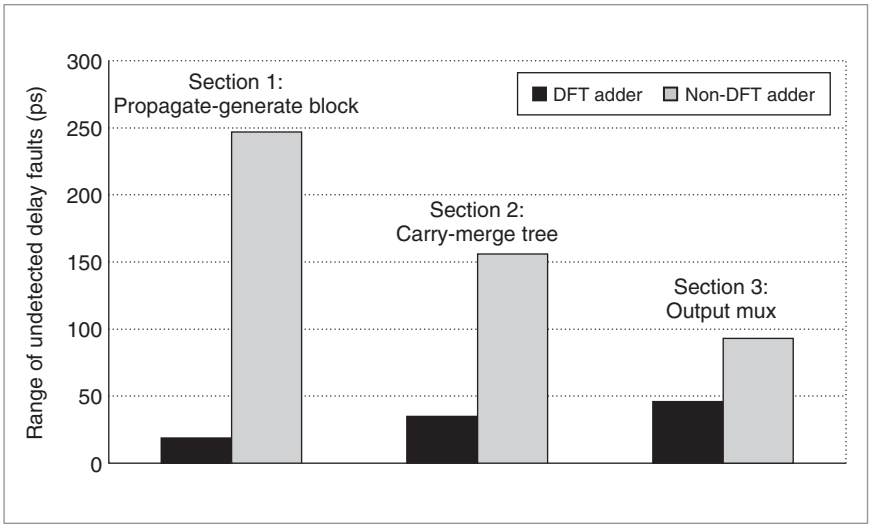

Figure 10. Fault detection range in low-frequency test mode with and without DFT.

OUR DFT TECHNIQUE allows detection of a wider range of delay faults with improved resolution and is suited for dynamic-logic testing. The technique will improve the long-term reliability of high-performance digital circuits and decrease test costs while maintaining design overhead within acceptable limits. In future work, we plan to extend our DFT technique to a 32-bit ALU. We also plan to determine the technique's impact on overall test time and automatic test pattern generation. ■

Acknowledgments

We thank Mohammad Nummer and Shahrzad Naraghi (University of Waterloo) and Ram Krishnamurthy (Intel) for their discussion and insights.

■ References

1. *International Technology Roadmap for Semiconductors*, Semiconductor Industry Assoc., 2001.
2. M. Sachdev, "Current-Based Testing for Deep-Submicron VLSIs," *IEEE Design & Test*, vol. 18, no. 2, Mar.-Apr. 2001, pp. 76-84.
3. Z. Cheng, L. Wei, and K. Roy, "On Effective I_{DDQ} Testing of Low-Voltage CMOS Circuits Using Leakage Control Techniques," *Proc. 1st IEEE Int'l Symp. Quality Electronic Design* (ISQED 00), IEEE CS Press, 2000, pp. 181-188.
4. W. Needham, C. Prunty, and E.H. Yeoh, "High-Volume Microprocessor Test Escapes, An Analysis of Defects Our Tests Are Missing," *Proc. Int'l Test Conf.* (ITC 98), IEEE Press, 1998, pp. 25-34.
5. H. Hao and E.J. McCluskey, "Very-Low Voltage Testing for Weak CMOS Logic ICs," *Proc. Int'l Test Conf.* (ITC 93), IEEE Press, 1993, pp. 275-284.
6. A. Keshavarzi, K. Roy, and C.F. Hawkins, "Intrinsic Leakage in Low-Power Deep-Submicron CMOS ICs," *Proc. Int'l Test Conf.* (ITC 97), IEEE Press, 1997, pp. 146-155.
7. V.D. Agrawal and T.J. Chakraborty, "High-Performance Circuit Testing with Slow-Speed Tester," *Proc. Int'l Test Conf.* (ITC 95), IEEE Press, 1995, pp. 302-310.
8. B. Chatterjee, M. Sachdev, and A. Keshavarzi, "A DFT Technique for Low-Frequency Delay Fault Testing in High-Performance Digital Circuits," *Proc. Int'l Test Conf.* (ITC 02), IEEE Press, 2002, pp. 1130-1139.
9. K. Bernstein et al., *High-Speed CMOS Design Styles*, Kluwer Academic, 1999.
10. S. Matthew et al., "Sub-500ps 64-b ALUs in 0.18µm SOI/Bulk CMOS: Design and Scaling Trends," *IEEE J. Solid-State Circuits*, vol. 36, no. 11, Nov. 2001, pp. 1636-1646.
11. S. Thompson et al., "Dual Threshold Voltages and Substrate Bias: Keys to High-Performance, Low-Power, 0.1µm Logic Designs," *Proc. Symp. VLSI Technology, Systems and Applications*, IEEE Press, 1997, pp. 69-70.

Bhaskar Chatterjee is pursuing a PhD in the Department of Electrical and Computer Engineering at the University of Waterloo, Ontario, Canada. His research interests include high-performance digital circuit design and testability techniques. Chatterjee has an MS in electrical engineering from the Indian Institute of Science. He is a student member of the IEEE and the ACM.

Manoj Sachdev is a professor in the Department of Electrical and Computer Engineering at the University of Waterloo. His research interests include low-power and high-performance digital circuit design, mixed-signal circuit design, and IC test and manufacturing. Sachdev has a BE in electronics and communication engineering from the University of Roorkee, India, and a PhD in electrical engineering from Brunel University, UK. He is a senior member of the IEEE.

Ali Keshavarzi is a senior staff research scientist in Intel's Microprocessor Research Laboratories, Portland, Oregon. His research interests include low-power and high-performance circuit techniques and transistor device structures for future generations of microprocessors. Keshavarzi has a PhD in electrical engineering from Purdue University.

■ Direct questions and comments about this article to Bhaskar Chatterjee, Dept. of Electrical and Computer Engineering, University of Waterloo, 200 University Ave. West, Waterloo, ON N21 3G1 Canada; bhaskar@vlsi.uwaterloo.ca.