# Improving Transition Delay Test Using a Hybrid Method

**Nisar Ahmed and Mohammad Tehranipoor**
University of Connecticut

*Editor's note:*
Structured delay test using scan transition tests is becoming commonplace. But high coverage and compact tests can still be elusive in some situations. The authors propose a novel technique combining the cost-effectiveness of launch-from-capture test with the coverage/pattern volume advantages of launch-from-shift.

*—Ken Butler, Texas Instruments*

**This transition-fault-testing technique** combines the launch-off-shift method and an enhanced launch-off-capture method for scan-based designs. The technique improves fault coverage and reduces pattern count and scan-enable design effort. It is practice oriented, suitable for low-cost testers, and implementable with commercial ATPG tools.

Scan-based structural tests increasingly serve as a cost-effective alternative to the at-speed functional-pattern approach to transition delay testing.[1,2] Transition fault testing involves applying a pattern pair ($V_1$, $V_2$) to the circuit under test. $V_1$ is the *initialization pattern*, and $V_2$ is the *launch pattern*. $V_2$ launches the desired signal transition ($0 \rightarrow 1$ or $1 \rightarrow 0$) at the target node, and the response of the circuit under test is captured at functional speed (the rated clock period). The entire operation consists of three cycles:

- *initialization*—a scan-in operation applies $V_1$;
- *launch*—a transition is launched at the target gate terminal ($V_2$ is applied); and
- *capture*—the transition is captured at an observable point.

Transition fault test patterns can be generated and applied in three ways: the launch-off-shift (LOS) or skewed-load method, the launch-off-capture (LOC) or broadside method, or the enhanced-scan method. In this article, we focus only on the first two methods. In LOS, the transition at a target gate output is launched in the last shift cycle during the shift operation. Figure 1a shows the waveforms during a LOS operation's cycles. The launch cycle is part of the shift operation and is immediately followed by a fast capture pulse. The time period for the scan-enable signal (*SEN*) to make this $1 \rightarrow 0$ transition corresponds to the functional frequency. Hence, LOS requires that *SEN* be timing critical. In LOC, the transition is launched and captured through the functional pin (D) of any flip-flop in the scan chain.

Figure 1b shows the waveforms of the LOC method, which separates the launch cycle from the shift operation. Because launch pattern $V_2$ depends on the functional response of initialization vector $V_1$, the launch path is less controllable, so test coverage is low. LOC relaxes the at-speed constraint on *SEN* and adds dead cycles after the last shift to provide enough time for *SEN* to settle low.

As device frequencies become higher, production test equipment capabilities limit the ability to test a device at speed. Rather than purchasing a more expensive tester, test engineers use one of several on-chip DFT alternatives, such as an on-chip clock generator for at-speed clock, pipeline *SEN* generation, or on-chip at-speed *SEN* generation[3] for LOS transition fault testing. The LOS method is preferable to the LOC method in terms of ATPG complexity and pattern count. However, because of increasing design sizes, the *SEN* fan-out exceeds any other net in the design. LOS constrains *SEN* to be timing critical, requiring a design effort that makes it difficult to implement products in reasonable turnaround times. That's the main reason for the widespread use of the LOC method, especially on very low-cost testers.[2] In this article, we propose a hybrid technique that uses both LOS and LOC in scan-based designs, pro-


0740-7475/06/$20.00 © 2006 IEEE Copublished by the IEEE CS and the IEEE CASS

viding higher fault coverage and lower pattern count with a small scan-enable design effort. (The "Related work" sidebar discusses other approaches to improving transition delay test quality.)

## Overview

Our proposed scan architecture controls a small subset of selected scan cells by the LOS method, and controls the remaining scan cells by the enhanced launch-off-capture, or ELOC, method (see the "Related work" sidebar). We use an efficient ATPG-based controllability-and-observability measurement approach to select the scan cells controlled by LOS or ELOC. The selection criteria improve fault coverage and reduce the overall pattern count. Because a few scan cells are LOS controlled, only a small subset of the scan chains' *SEN* signals must be timing closed; this reduces the scan-enable design effort. The method is robust and practice oriented, and it uses existing commercial ATPG tools.[4]

To control the scan chain operation mode (LOS or ELOC), two new cells called local scan-enable generators (LSEGs) generate on-chip *SEN* signals. The scan-enable control information for the launch and capture cycles is embedded in the test data itself. The LSEGs can be inserted anywhere in the scan chain with negligible hardware area overhead. The proposed technique is suitable for low-cost testers because it doesn't require external at-speed *SEN*.

## Motivation

ELOC improves the controllability of launching a transition through either the scan path or the functional path.[5] However, it provides less observability than LOS does because a scan chain working in shift mode to launch a transition is not observable at the time of capture (*SEN* is held high during the launch and capture cycles). Therefore, ELOC's fault coverage is less than that of LOS but greater than that of LOC. Figure 2a (on p. 405) shows fault coverage analysis for the three transition fault methods. A common set of transition faults is detected by both LOS and LOC, and some faults in the LOC transition fault set are not detected by LOS, such as shift-dependency untestable faults.[6,7] However, ELOC covers LOC's entire transition fault set and also detects some extra faults in the LOS-detected fault set. This is because LOC is a special case in which all local *SEN* signals are held at 0 during the launch and capture cycles. ELOC provides an intermediate fault coverage point between LOS and the conventional LOC method.[5]

To improve fault coverage and identify the union of fault sets detected in both the LOS and ELOC modes, the scan cells must be controllable in both modes. Also, to reduce the design effort for at-speed, scan-enable signal (required for LOS), we must determine the minimum number of scan cells that require very high controllability and observability during pattern generation. We must control the resulting smaller subset of scan cells in LOS mode, and the remaining scan cells in ELOC mode. This reduces the design effort to timing-close the *SEN* signal at speed as required for LOS-controlled scan flip-flops.

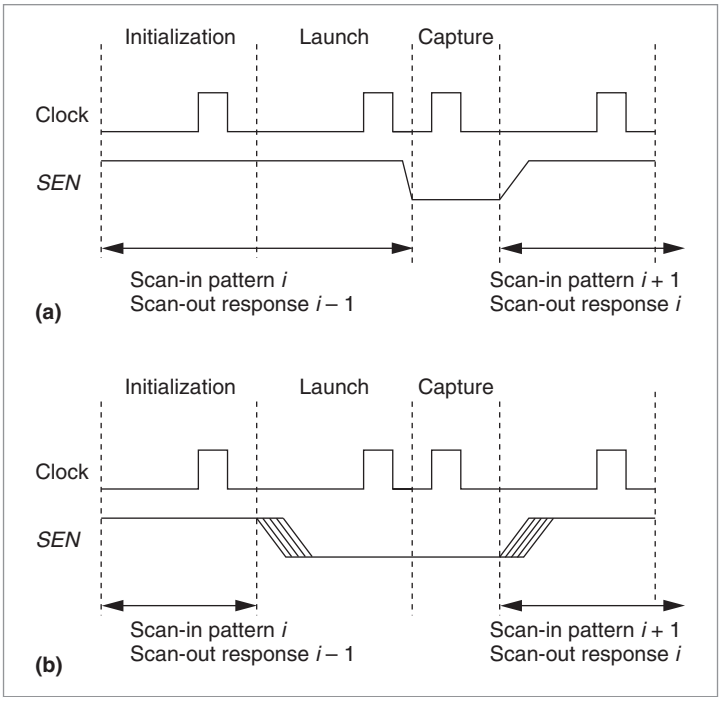


**Figure 1. Transition delay fault pattern generation methods: launch-off-shift (LOS) (a) and launch-off-capture (LOC) (b).**

Figure 2b shows an example of a hybrid scan architecture with eight scan chains. The LOS-controlled scan flip-flops are stitched in separate scan chains. A fast *SEN* signal controls the first three scan chains containing LOS-controlled flip-flops, and a slow *SEN* signal controls the remaining scan chains in ELOC mode. Moreover, this architecture also requires configuring the LOS-controlled scan chains in functional mode because some faults are detected only by LOC and not by LOS.

## Local *SEN* generation

The new method for testing transition faults provides more controllability in launching a transition but requires an independent *SEN* for each scan chain. We can use multiple scan-enable ports, but this increases

## Related work

Wang, Liu, and Chakradhar propose a hybrid scan architecture that controls a small subset of selected scan cells by launch-off shift (LOS), and the rest by launch-off capture (LOC).[1] The authors have designed a fast scan-enable signal (*SEN*) generator that drives the LOS-controlled scan flip-flops. The selection criteria of the LOS-controlled scan flip-flops determine the method's effectiveness. In some cases, the number of patterns generated by the hybrid method exceeds the LOC pattern count. Moreover, the LOS-controlled flip-flops cannot be used in LOC mode. Figure A1 shows the *SEN* waveforms of this hybrid technique.

In a new scan-based, at-speed test called enhanced launch-off-capture (ELOC), the ATPG tool deterministically targets the transition launch path either through a functional path or the scan path.[2] The technique improves transition fault testing controllability and fault coverage, and it does not require *SEN* to change at speed. Figure A2 shows *SEN* waveforms in the ELOC technique. The *SEN* signal of a subset of scan chains stays at 1 ($SEN_1$) during the launch and capture cycles to launch the transition only. The second *SEN* signal ($SEN_2$) controls the remaining scan chains to launch a transition through the functional path during the launch cycle and capture the response during the capture cycle. Figure A3 shows a circuit with two scan chains, chain 1 acting as a shift register, and chain 2 acting in functional mode. The conventional LOC method is a special condition of the ELOC method in which the *SEN* signals of all chains are 0 during the launch and capture cycles.

Two other proposed techniques improve LOS fault coverage by reducing shift dependency.[3,4] A technique by Li et al. reorders the scan flip-flops to minimize the number of undetectable faults, and restricts the distance by which a scan flip-flop can be moved to create the new scan chain order.

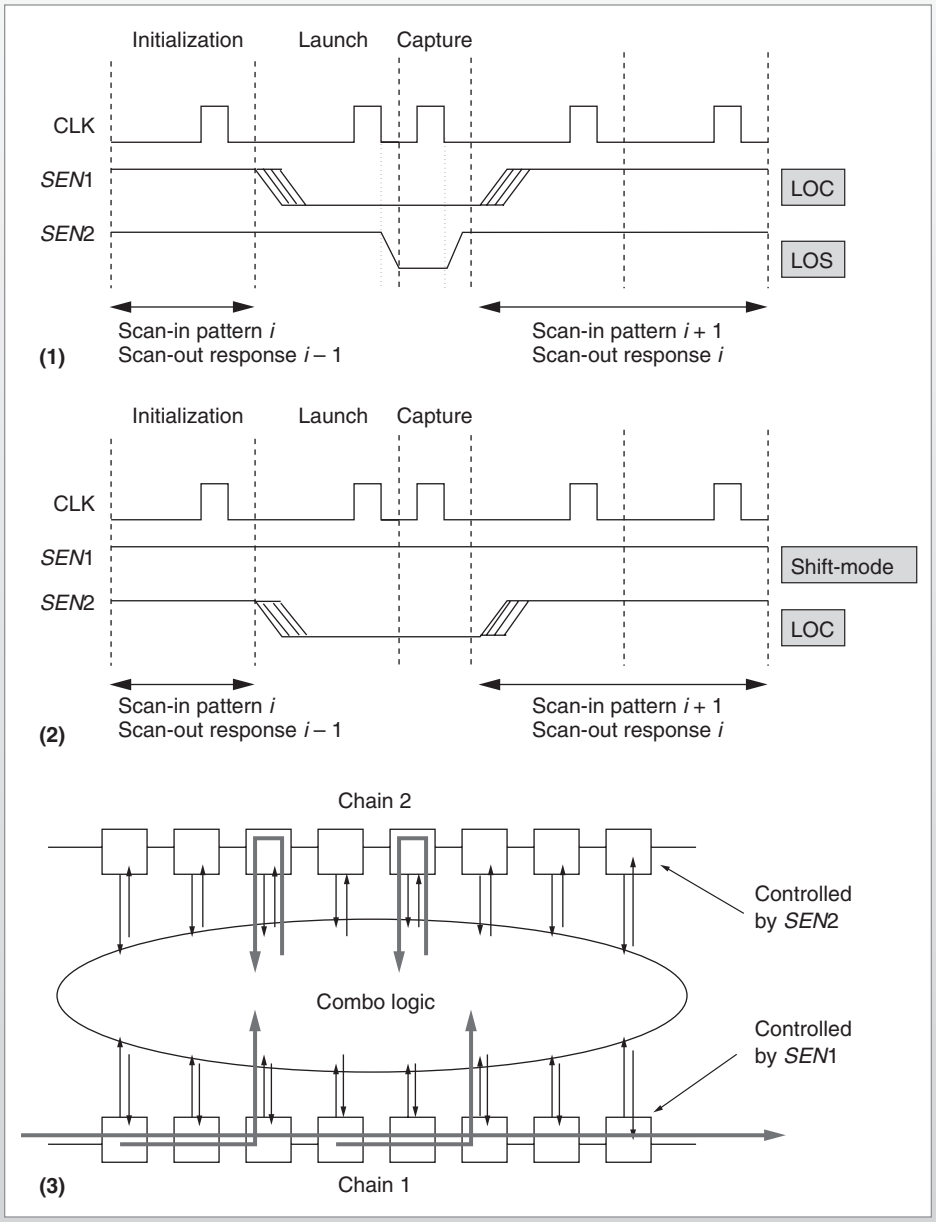


**Figure A. Previously proposed techniques: *SEN* waveforms in hybrid scan technique (1), *SEN* waveforms in enhanced LOC (ELOC) technique (2); ELOC controllability—chain 1 used in shift mode, and chain 2 in functional mode (3).**

Gupta et al. propose a technique that inserts dummy flip-flops and reorders scan flip-flops, considering wire length costs to improve path delay fault coverage. Wang and Chakradhar propose using a special ATPG to identify pairs of adjacent flip-flops and inserting test points (dummy gates or flip-flops) between them.[5]

## References


1. S. Wang, X. Liu, and S.T. Chakradhar, "Hybrid Delay Scan: A Low Hardware Overhead Scan-Based Delay Test Technique for High Fault Coverage and Compact Test Sets," *Proc. Design, Automation and Test in Europe* (DATE 03), IEEE Press, 2004, pp. 1296-1301.
2. N. Ahmed, M. Tehranipoor, and C.P. Ravikumar, "Enhanced Launch-off-Capture Transition Fault Testing," *Proc. Int'l Test Conf.* (ITC 05), IEEE Press, 2005, pp. 246-255.
3. W. Li et al., "Distance Restricted Scan Chain Reordering to Enhance Delay Fault Coverage," *Proc. 18th Int'l Conf. VLSI Design*, IEEE Press, 2005, pp. 471-478.
4. P. Gupta et al., "Layout-Aware Scan Chain Synthesis for Improved Path Delay Fault Coverage," *Proc. Int'l Conf. Computer-Aided Design* (ICCAD 03), IEEE Press, 2003, pp. 754-759.
5. S. Wang and S.T. Chakradhar, "Scalable Scan-Path Test Point Insertion Technique to Enhance Delay Fault Coverage for Standard Scan Designs," *Proc. Int'l Test Conf.* (ITC 03), IEEE Press, 2003, pp. 574-583.


the number of pins. Two types of *SEN* signals must be generated on chip. The scan-enable control information for the scan flip-flops differs only during the pattern's launch and capture cycles. Hence, we can use the low-speed *SEN* signal from the external tester for the scan shift operation and internally generate the scan-enable control information for only the launch and capture cycles.

### LSEG cells

Because our hybrid technique uses both LOS and enhanced LOC techniques, we must generate both fast and slow local *SEN* signals. We propose two LSEG cells to generate on-chip local *SEN*s using a low-speed external *SEN* generated by a low-cost tester.

**Slow scan-enable generator (SSEG).** We designed an LSEG to control a scan flip-flop's transition launch path.[5] In this article, we refer to this cell as the slow scan-enable generator (SSEG) because the local *SEN* signal does not make an at-speed transition. Figure 3a shows the SSEG cell architecture. It consists of a single flip-flop that loads the control information required for the launch and capture cycles. The input scan-enable ($SEN_{in}$) pin connected to the external *SEN* signal from the tester is called global scan-enable (*GSEN*). An additional output scan-enable pin ($SEN_{out}$ *LSEN*) represents the local scan-enable (*LSEN*) signal. Therefore, after going to control state *Q* at the end of the shift operation (that is, after *GSEN* is deasserted), *LSEN* remains in this state as long as *GSEN* asynchronously sets it to 1. The SSEG cell essentially holds the value 0 or 1 loaded at the end of the shift operation (*GSEN* = 1) for the launch and capture cycles:

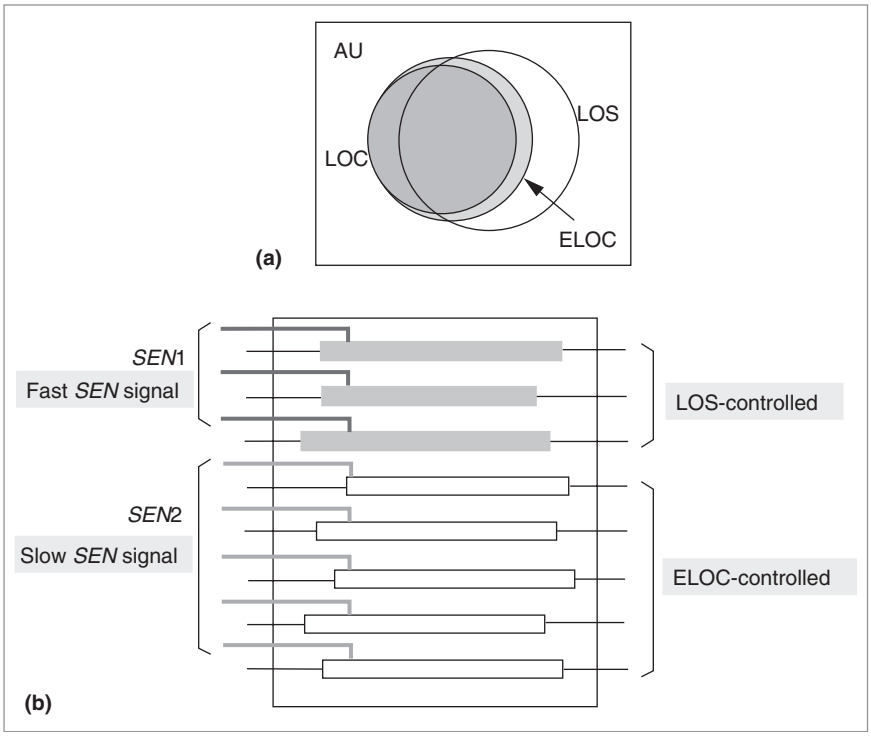


**Figure 2. Hybrid method analysis and architecture: Fault analysis of LOS, LOC, and ELOC techniques (a), and hybrid scan architecture: with LOS-controlled scan chains using fast *SEN* signal and ELOC-controlled scan chains using slow *SEN* signal (b).**

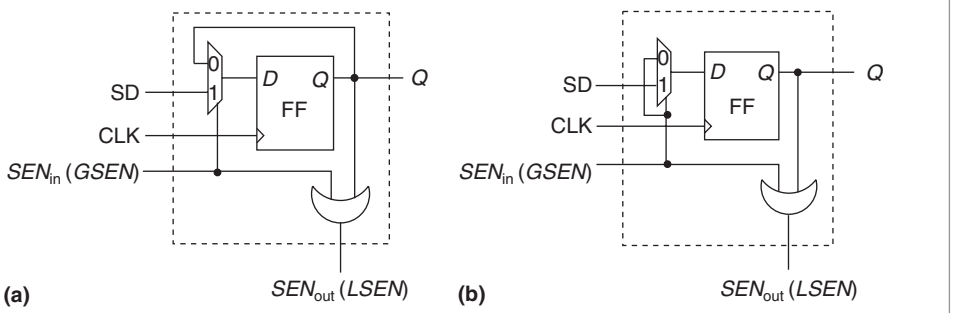


**Figure 3. LSEG cells: slow scan-enable generator (SSEG) cell (a) and fast scan-enable generator (FSEG) cell (b).**

**Table 1. *SSEG* operation, where *GSEN* is the global scan-enable signal, Q is the flip-flop's state, and *LSEN* is the local scan-enable signal.**

| *GSEN* | *Q* | *LSEN* | Operation |
|---|---|---|---|
| 1 | X | 1 | Shift |
| 0 | 1 | 1 → 1 | Shift-launch (no capture) |
| 0 | 0 | 0 → 0 | Functional launch and capture |

**Table 2. FSEG operation.**

| *GSEN* | *Q* | *LSEN* | Operation |
|---|---|---|---|
| 1 | X | 1 | Shift |
| 0 | 1 | 1 → 0 | Shift-launch capture |
| 0 | 0 | 0 → 0 | Functional launch and capture |

$$LSEN = (GSEN + Q) = \begin{cases} 1 \text{ if } GSEN = 1 \\ Q \text{ if } GSEN = 0 \end{cases}$$

Table 1 shows the SSEG cell's operation modes. *GSEN* = 1 represents the pattern's normal shift operation. When *GSEN* = 0 and *Q* = 1, *LSEN* = 1 and the controlled scan flip-flops act in the shift mode to launch the transitions-only, shift-launch (no-capture) mode. Moreover, there is no capture, because the *LSEN* signal is 1 (*LSEN* = 1 → 1 at the launch edge). The other observable scan flip-flops perform the capture. The *LSEN*-controlled scan flip-flops act in the conventional LOC mode when *GSEN* = 0 and *Q* = 0 (functional-launch-capture mode).

**Fast scan-enable generator (FSEG).** Figure 3b shows our new local, at-speed, scan-enable generator architecture, which we call the fast scan-enable generator (FSEG). Table 2 shows the FSEG cell's operation modes. As in SSEG cell operation, *GSEN* = 1 represents the pattern's normal shift operation. When *GSEN* = 0 and *Q* = 1, *LSEN* = 1 and the scan flip-flops act in the shift-launch-capture mode to launch the transition from the scan path and capture the response at the next capture cycle (conventional LOS method). The *LSEN* from the FSEG cell makes a 1 → 0 at-speed transition at the launch cycle. The *LSEN*-controlled scan flip-flops act in the conventional LOC mode when *GSEN* = 0 and *Q* = 0 (functional-launch-capture mode).

## LSEG cell operation

LSEG cells inserted in the scan chains pass control information as part of the test data. The scan-enable control information is part of each test pattern and is stored in the tester's memory. Figure 4a shows the normal scan architecture with a single *SEN* signal from the external tester. The scan chain contains eight scan flip-flops, and the shifted test pattern is 10100110. Figure 4b shows the same circuit, which generates an *LSEN* signal from the test pattern data for the hybrid transition fault test method. The main objective is to deassert the external *GSEN* signal after the entire shift operation and then generate the *LSEN* signal from the test data during the launch and capture cycles. In this case, the shifted pattern is modified to [*C*]10100110, where *C* is the scan-enable control bit stored in the LSEG cell at the end of the scan operation.

The *GSEN* signal asynchronously controls the shift operation. *GSEN* is deasserted after the *n*th shift (initialization) cycle, where *n* = 9; *n* is the length of the scan chain after insertion of the LSEG cell. After the *GSEN* signal is deasserted at the end of the shift operation, the scan-enable control during the launch and capture cycles is control bit *C* stored in the LSEG. At the end of the capture cycle, *GSEN* asynchronously sets the *LSEN* signal to 1 for scanning out the response.

Figure 4c shows the process of test pattern application and the timing waveforms for the two LSEG cells, SSEG and FSEG.

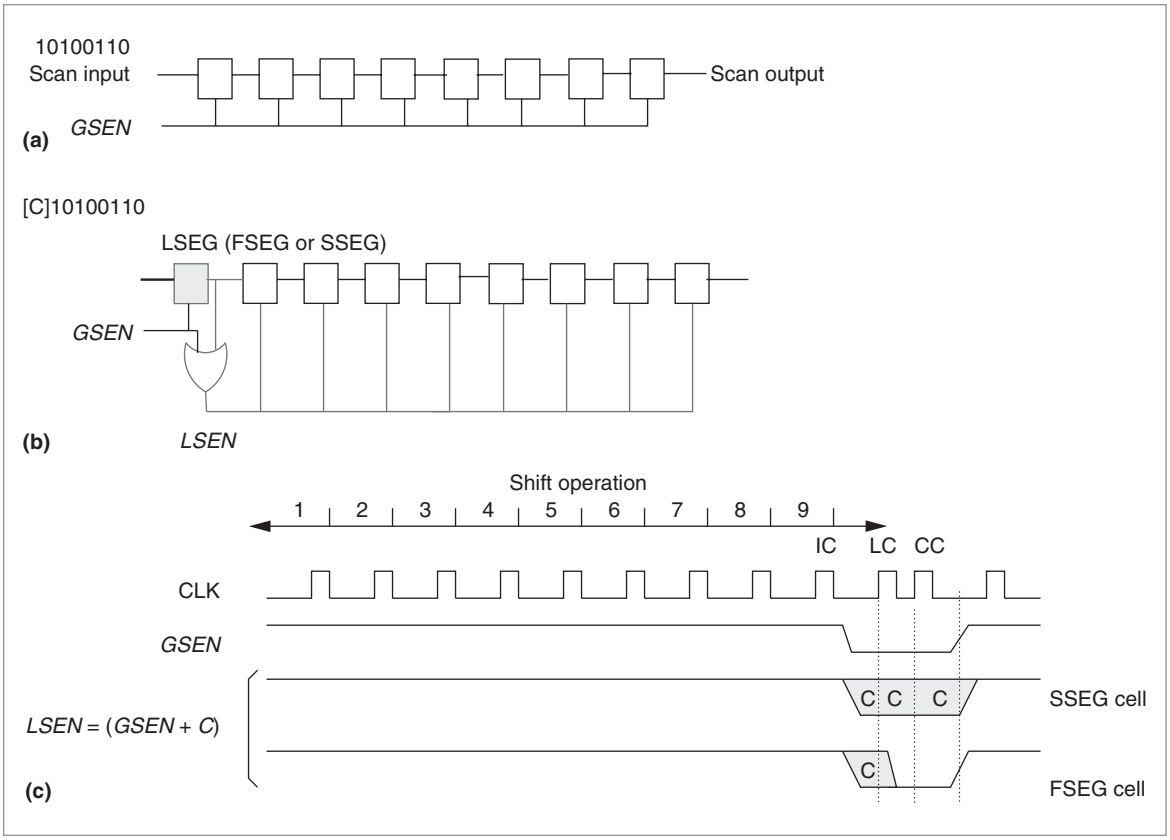


**Figure 4. LSEG cell operation: Scan chain architecture (a), *LSEN* generation using LSEG (b), and *LSEN* generation process and waveforms (c).**

## Flip-flop selection: Measuring controllability and observability

In the LOS technique, the fault activation path (scan path), unlike the functional path used in the LOC method, is fully controllable from the scan chain input. Hence, in most cases, for the same detected fault, a LOS pattern requires fewer care bits than a LOC pattern. The controllability measure of a scan flip-flop is the percentage of patterns in the entire pattern set ($P$) for which a care bit is required in the scan flip-flop to enable activation or propagation of a fault effect. Figure 5 shows a scan flip-flop with an input (observability) and output (controllability) logic cone. A large output logic cone implies that the scan flip-flop will control a greater number of faults; that is, a care bit will be required in their activation or propagation. Similarly, the input logic cone determines a scan flip-flop's observability. We define this observability as the percentage of patterns in the entire pattern set ($P$) for which a valid care bit is observed in the scan flip-flop.

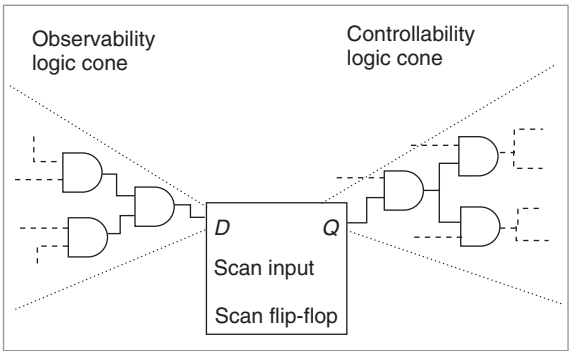


**Figure 5. Scan flip-flop controllability-and-observability measure.**

In a transition fault test pattern pair ($V_1$, $V_2$), initialization pattern $V_1$ is essentially an $I_{DDQ}$ pattern to initialize the target gate to a known state. In the next time frame, pattern $V_2$ is a stuck-at-fault test pattern to activate and propagate the required transition at the target node to an observable point. Therefore, to find the controllability-observability measure of a scan flip-flop, we use an ATPG tool to generate stuck-at patterns and force it to fill in don't-care (X) values for scan flip-flops that don't affect any fault's activation or propagation. The

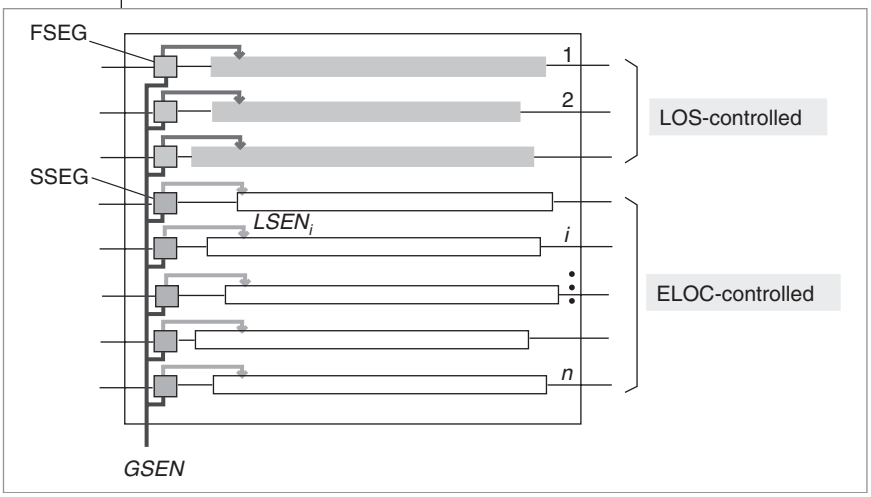


**Figure 6. Hybrid scan test architecture: FSEG cells driving LOS-controlled scan chains, and SSEG cells driving ELOC-controlled scan chains.**

**Table 3. Case study design characteristics.**

| Characteristics | No. |
|---|---|
| Clock domains | 6 |
| Scan chains | 16 |
| Scan flip-flops | 10,477 |
| Nonscan flip-flops | 13 |
| Transition delay faults | 320,884 |

*i*th scan flip-flop's controllability is $C_i = p_c/P$, where $p_c$ is the number of patterns with a care bit in the scan flip-flop during scan-in, and $P$ is the total number of stuck-at patterns. Similarly, observability is $O_i = p_o/P$, where $p_o$ is the number of patterns with a care bit in the scan flip-flop during scan-out.

We then use each scan flip-flop's measured controllability and observability factors to determine cost function $CF_i = C_iO_i$. The scan flip-flops are arranged in decreasing order of cost function, and a subset with very high cost functions is selected as LOS-controlled flip-flops. The ATPG-based controllability-observability measurement technique overcomes the limitation of the Scoap-based method[8] used by Wang, Liu, and Chakradhar,[6] which makes it possible to select a scan flip-flop that has high 0(1) controllability but is not controlled to 0(1) during pattern generation by the ATPG tool.

## Case study

The following case study illustrates DFT insertion and ATPG flow of our hybrid scan transition fault-testing technique. It includes an analysis of extra detected faults.

### Test architecture

The LSEG-based solution presented here provides a method of generating internal *LSEN* signals from pattern data, and *GSEN* signals from the tester. The overhead of generating the *LSEN* signal is the additional LSEG (SSEG or FSEG) cell in the scan chain. An LSEG cell's area overhead is a few extra gates, which is negligible in modern designs. We assume that the area overhead of the buffer tree required to drive all the LOS-controlled scan flip-flops through the LSEG cells is equal to the overhead of applying an at-speed *GSEN* signal from external ATE.

Figure 6 shows a multiple-scan-chain architecture with $n$ scan chains. The LOS-controlled scan flip-flops determined by the controllability-observability measurement are stitched in separate scan chains. Each scan chain $i$, where $1 \le i \le n$, consists of an LSEG (FSEG or SSEG) cell that generates signal $LSEN_i$ for the respective scan chain. The *GSEN* signal connects only to the $SEN_{in}$ port of the LSEG cells.

### Study description

In this case study, we experimented with a subchip of an industrial-strength design with the characteristics listed in Table 3. One LSEG cell is inserted per scan chain. The test strategy was to get the highest possible transition fault test coverage. When generating test patterns for transition faults, we targeted only faults in the same clock domain. During pattern generation, only one clock is active during the launch and capture cycles. Hence, only faults in that particular clock domain are tested. All primary inputs remain unchanged, and all primary outputs are unobservable during test-pattern generation for transition faults. This is because the very low cost testers are not fast enough to provide PI values and strobe POs at speed.

### DFT insertion

We measure a scan flip-flop's cost function (*controllability* × *observability*) using the ATPG-based technique explained earlier. Figure 7 shows the cost function of each scan flip-flop in our design. Approximately only 20% to 30% of the flip-flops require very high controllability and observability. Hence, *SEN* need not be at speed for all scan flip-flops. We arrange the scan flip-flops in decreasing order of cost function, and we use this order during scan insertion.

In the new order of scan chains, the few initial chains consist of very high controllability-observability flip-flops, and we select them for LOS according to their average cost function. We measure a scan chain's average cost function as $\sum CF_i/N$, where $CF_i = C_i \times O_i$ is the cost function of the $i$th scan flip-flop in the chain, and $N$ is the number of flip-flops in the scan chain. Figure 8 shows each chain's average cost function for normal scan insertion and after scan insertion based on controllability and observability. As Figure 8b shows, after this scan insertion, the average cost function of the first five scan chains is very high (due to scan flip-flops with very high cost functions) and very low for the rest of the chains. Therefore, we can design the first five chains' *SEN* signal to be at speed (controlled by the FSEG cell), and the rest of the scan chains can use a slow-speed *SEN* (controlled by the SSEG cell).

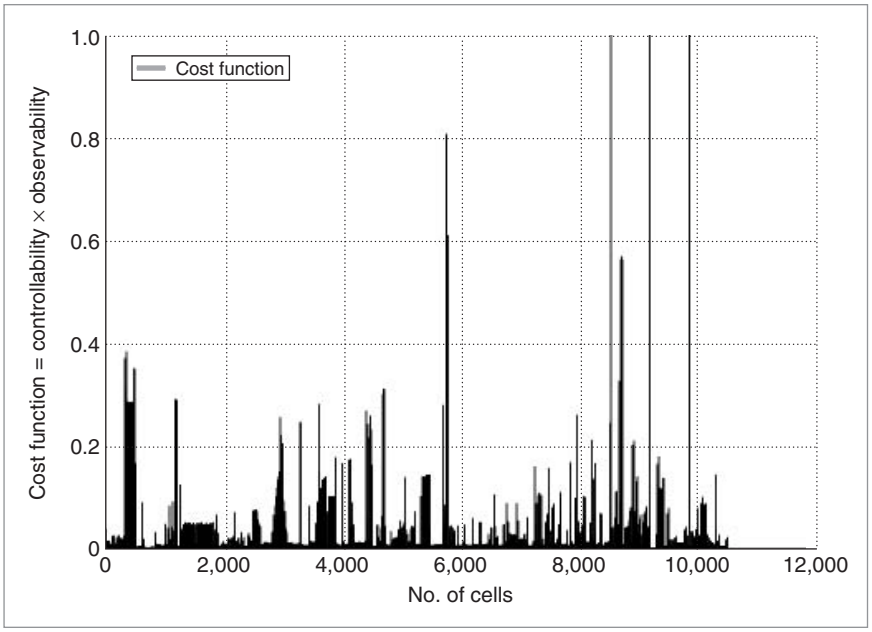


**Figure 7. Cost functions of scan flip-flops in case study design.**

We used the Synopsys DFT Compiler for scan chain insertion.[4] To insert the LSEG cells, the synthesis tool must recognize the LSEG cell as a scan cell and stitch it into the chain. This means that the LSEG cell must be defined as a new library cell with scan cell attributes. A workaround is to design the LSEG cell as a module, instantiate it, and declare it as a scan segment of length 1. The *GSEN* signal is connected to all LSEG cell $SEN_{in}$ pins. During scan insertion, we specify only the LSEG cell in the scan path because the tool will stitch the rest of the cells, including the LSEG cell, and balance the scan chain, depending on the longest scan chain length parameter. Additionally, the tool provides the flexibility to hook up each LSEG cell's $SEN_{out}$ port in a particular chain to all the $SEN_{in}$ ports of the scan flip-flops in the respective chain.

## ATPG

The ATPG tool must understand the LSEG methodology and deterministically choose the transition fault activation path. We used a commercial ATPG tool, Synopsys TetraMax,[4] which supports two ATPG modes: basic scan and sequential. Basic-scan ATPG is a combinational-only mode with only one capture clock between pattern scan-in and response scan-out; the sequential mode uses a sequential time-frame ATPG algorithm. By default, when generating test patterns for the transition fault model in functional launch mode, the ATPG tool uses a two-clock ATPG algorithm that has some features of both the basic-scan and sequential engines. The tool understands the LSEG technique and can choose the launch path for the target transition fault deterministically. Hence, there is no fundamental difference in ATPG methodology when we use the LSEG-based solution.

The *SEN* signal for the flip-flops in the launch and capture cycles comes from an internally generated signal. The OR gate in the LSEG cell generates the *LSEN* signal through a logical OR of the flip-flop's *GSEN* and *Q* output (see Figure 3). The *GSEN* signal is high during scan shift operation. The tool determines each chain's *LSEN* and shifts the control value into the LSEG cell during pattern shift for launch and capture. It also deterministically decides the combination of scan chains to work in shift or functional launch mode, to activate a transition fault.

Table 4 shows results for conventional LOS and LOC (normal scan insertion), ELOC, and hybrid transition delay ATPG on the case study design. We see that LOS gave approximately 3% higher fault coverage than LOC. ELOC gave approximately 1.9% higher fault coverage than the LOC method. The hybrid technique gave better fault coverage than the other methods and provided a better pattern count than the LOC and ELOC methods. The pattern count was greater than that of LOS but at the advantage of less scan-enable design effort—only five scan chains being timing closed for at-speed *SEN*. (The hybrid scan technique proposed by Wang, Liu, and Chakradhar[6] sometimes gives a greater pattern count than the LOC technique.) Our hybrid method used more CPU time than

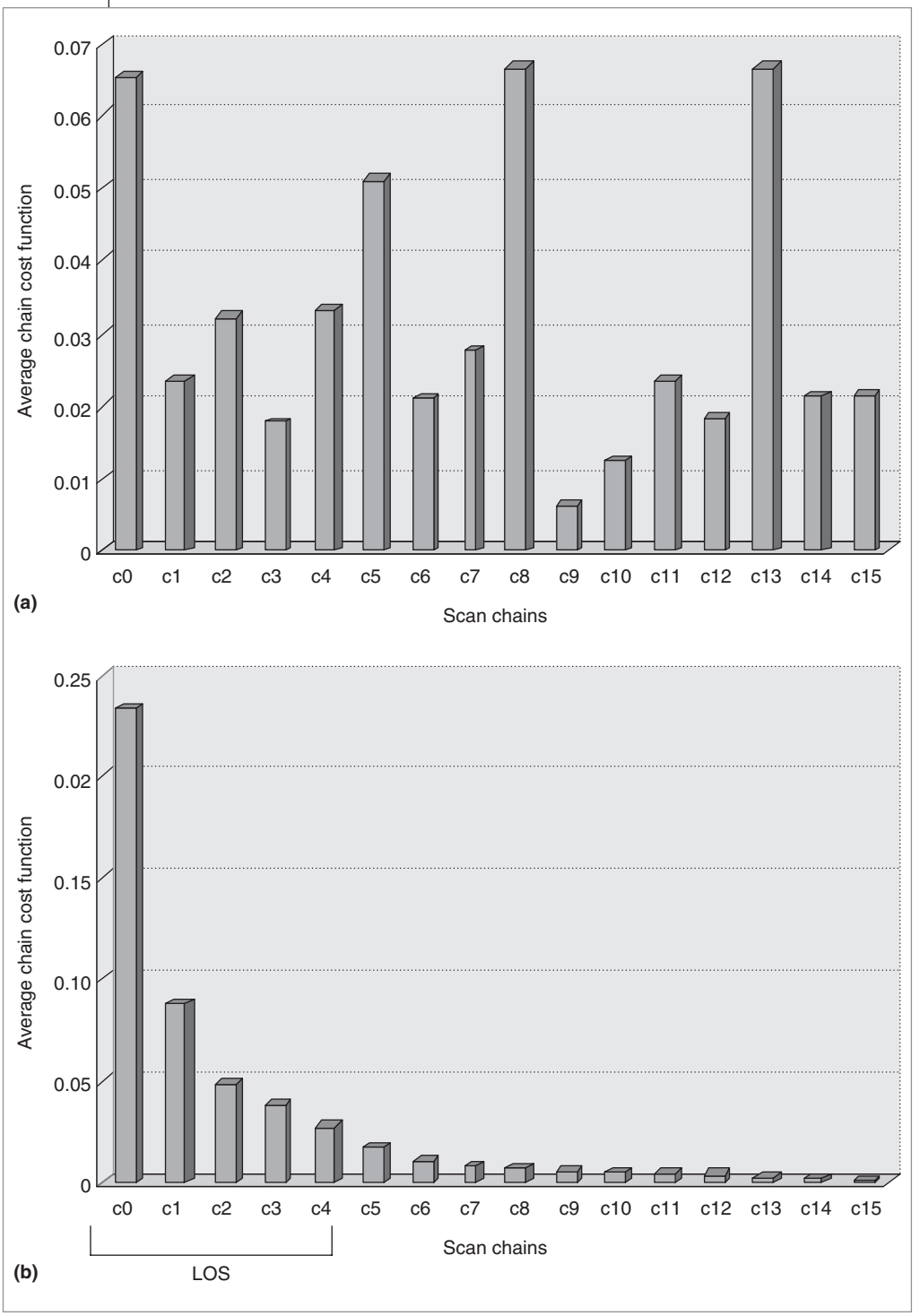


**Figure 8. Average cost function before (a) and after (b) scan insertion based on controllability and observability.**

the other techniques because for hard-to-detect faults, the ATPG tool must do more processing to determine the possible combinations of the SSEG-controlled scan chains in shift register mode or functional mode.

## Analysis of extra detected faults

As Rearick discusses, the detection of functionally untestable faults poses a potential yield loss problem.[9] We analyzed the additional faults detected by the hybrid scan architecture over the conventional LOC technique. To determine the nature of these extra faults, we performed conventional LOC ATPG on them. For example, for ITC99 benchmark circuit b17, the hybrid scan method detected 17,926 extra faults. LOC ATPG on these faults showed all of them as nonobservable faults—faults that can be controlled but cannot be propagated to an observable point.

It can be argued that some of these nonobservable detected faults can result in yield loss because some of them might be functionally untestable. However, some of these faults are actually functionally testable but become nonobservable because of low-cost tester ATPG constraints such as no primary input changes or no primary output measures. For example, of the 17,926 extra faults detected by hybrid scan in the nonobservable class, 1,155 were detectable without the low-cost tester constraints. Also, Lai, Krstic, and Cheng show that functionally untestable nonobservable faults might not need testing if the defect doesn't cause a delay exceeding twice the clock period.[10] With technology scaling and increasing operating fre-

quencies, detecting multicycle delay faults might become important, and more than two vectors are required to detect such faults.[10] The hybrid scan technique can be advantageous because it eases ATPG and detects multicycle faults using a two-vector pair.

## Experimental results

We experimented with our hybrid scan technique on the three largest 1999 International Test Conference (ITC) benchmark circuits and on four more industrial designs ranging in size from 10,000 to 100,000 flip-flops. We inserted 16 scan chains in each design. For the LOS and LOC techniques, we used the Synopsys DFT Compiler to perform normal scan insertion. For the ELOC and hybrid techniques, we performed scan insertion based on controllability and observability, and we inserted one LSEG cell in each scan chain. In the case of ELOC, we inserted only SSEG cells in each scan chain. In the hybrid technique, we selected only the first four scan chains to be LOS controlled (FSEG) after controllability-observability measurement; the remaining scan chains were ELOC controlled (SSEG). This reduced the at-speed scan-enable design effort significantly because the *SEN* signal to only one fourth of the scan flip-flops needed to be timing closed.

During ATPG, the faults related to clocks, scan-enable, and set or reset pins, referred to as untestable faults, are not added to the fault list. Table 5 shows the ATPG results, comparing the LOS, LOC, ELOC, and hybrid methods. The ELOC method provides higher fault coverage than the LOC method (up to 15.6% for design b19), and in most cases an intermediate fault coverage and pattern count between LOS and LOC. The hybrid method provides better coverage than all other methods because it has the flexibility to use combinations of functional and scan paths for launching a transition. This method provides higher fault coverage, by up to 2.68% (design D) and 19.12% (design b19) than LOS and LOC, respectively.

**Table 4. Case study ATPG results.**

| Parameter | LOS | LOC | ELOC | Hybrid |
|---|---|---|---|---|
| Detected faults | 292342 | 282658 | 288681 | 295288 |
| Test coverage (%) | 91.30 | 88.27 | 90.15 | 91.92 |
| Fault coverage (%) | 91.11 | 88.09 | 89.96 | 91.74 |
| Pattern count | 1,112 | 2,145 | 2,014 | 1,799 |
| CPU time (s) | 329.30 | 896.96 | 924.74 | 1,014.60 |

In a worst-case analysis, the lower bound for ELOC is LOC with no extra faults detected over LOC, and the upper bound is LOS. Similarly, for the hybrid technique, the lower bound is ELOC, and the upper bound can be greater than or equal to LOS. However, in the worst-case scenario, for a given fault coverage, the hybrid method will still benefit in test-pattern count reduction compared to LOC, thereby reducing test time, with minimum scan-enable design effort. In some cases, the CPU time for the hybrid or ELOC method is greater than that of the LOC method because the ATPG tool needs a larger search space to find the transition launch activation path for hard-to-detect faults.

Typically, in an ASIC design flow, scan insertion takes place in a bottom-up manner, independent of a physical synthesis step. The DFT insertion tool stitches the scan chains based on the alphanumeric order of scan flip-flop names in each module. The resulting scan chains are then reordered during physical synthesis to reduce the scan chain routing area. At the top level, the module-level

**Table 5. ATPG results for 1999 International Test Conference (ITC) benchmark circuits and industrial designs.**

| | | LOS | | | LOC | | | ELOC | | | Hybrid | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Design | No. of FFs (1,000s) | Fault coverage (%) | No. of patterns | CPU time (s) | Fault coverage (%) | No. of patterns | CPU time (s) | Fault coverage (%) | No. of patterns | CPU time (s) | Fault coverage (%) | No. of patterns | CPU time (s) |
| b17 | 1.4 | 95.09 | 1,088 | 95.4 | 81.02 | 1,190 | 1,000.8 | 94.29 | 1,328 | 325 | 96.50 | 1,179 | 187.9 |
| b18 | 3.3 | 92.67 | 1,451 | 279.7 | 77.50 | 1,309 | 1,020.9 | 93.01 | 1,876 | 726 | 95.18 | 1,334 | 336.6 |
| b19 | 6.6 | 85.98 | 2,280 | 645.3 | 69.21 | 1,153 | 1,050.4 | 84.81 | 1,422 | 1,000 | 88.33 | 1,590 | 1,000.9 |
| A | 10 | 91.11 | 1,112 | 329 | 88.09 | 2,145 | 896 | 89.96 | 2,014 | 924 | 91.74 | 1,799 | 1,014 |
| B | 30 | 87.94 | 4,305 | 3,569 | 85.14 | 8,664 | 7,800 | 86.57 | 8,539 | 8,702 | 88.03 | 8,062 | 6,611 |
| C | 50 | 81.10 | 6,869 | 8,415 | 79.42 | 12,073 | 22,930 | 80.48 | 11,583 | 25,642 | 83.29 | 8,134 | 14,451 |
| D | 104 | 92.15 | 5,933 | 6,559 | 91.56 | 10,219 | 12,088 | 92.28 | 12,505 | 47,788 | 94.83 | 9,674 | 18,410 |

scan chains are stitched together. Similarly, in our bottom-up scan insertion flow, the scan chains in each module are stitched based on the decreasing order of scan flip-flops' cost functions, and the resulting scan chains are reordered during physical synthesis to reduce the scan chain routing area. Therefore, the new scan insertion method will not be affected significantly because scan insertion and physical synthesis are not performed for the entire chip. Although, it can be argued that our scan chain stitch for controllability and observability might slightly increase the scan chain routing area in some cases, the decreases in scan-enable design effort and area overhead compared with LOS are significant. Moreover, the technique has the flexibility to shuffle and reorder the different groups of scan chains (LOS controlled and ELOC controlled) if any scan-chain-routing problem arises.

**THE PROPOSED HYBRID TECHNIQUE** significantly reduces the design effort and eases the timing closure by selecting a small subset of scan chains to be controlled using LOS. The experimental results also show that the pattern count is reduced and fault coverage is considerably increased. A statistical analysis is required to find the optimum number of LOS-controlled scan chains. Minimizing the number of LOS-controlled scan chains will even further reduce the design effort, and future work must focus on this issue. ■

## Acknowledgments

Mohammad Tehranipoor's work was supported in part by SRC grant no. 2005-TJ-1322. Nisar Ahmed performed the implementation work at Texas Instruments, India.

## ■ References

1. X. Lin et al., "High-Frequency, At-Speed Scan Testing," *IEEE Design & Test*, vol. 20, no. 5, Sept.-Oct. 2003, pp. 17-25.
2. J. Saxena et al., "Scan-Based Transition Fault Testing—Implementation and Low Cost Test Challenges," *Proc. Int'l Test Conf.* (ITC 02), IEEE Press, 2002, pp. 1120-1129.
3. N. Ahmed et al., "At-Speed Transition Fault Testing with Low Speed Scan Enable," *Proc. 24th VLSI Test Symp.* (VTS 05), IEEE Press, 2005, pp. 42-47.
4. *User Manual for Synopsys Toolset Version 2005.09*, Synopsys, 2005.
5. N. Ahmed, M. Tehranipoor, and C.P. Ravikumar, "Enhanced Launch-off-Capture Transition Fault Testing," *Proc. Int'l Test Conf.* (ITC 05), IEEE Press, 2005, pp. 246-255.
6. S. Wang, X. Liu, and S.T. Chakradhar, "Hybrid Delay Scan: A Low Hardware Overhead Scan-Based Delay Test Technique for High Fault Coverage and Compact Test Sets," *Proc. Design, Automation and Test in Europe* (DATE 03), IEEE Press, 2004, pp. 1296-1301.
7. S. Wang and S.T. Chakradhar, "Scalable Scan-Path Test Point Insertion Technique to Enhance Delay Fault Coverage for Standard Scan Designs," *Proc. Int'l Test Conf.* (ITC 03), IEEE Press, 2003, pp. 574-583.
8. L.H. Goldstein and E.L. Thigpen, "SCOAP: Sandia Controllability/Observability Analysis Program," *Proc. 17th Design Automation Conf.* (DAC 80), IEEE Press, 1980, pp. 190-196.
9. K.J. Rearick, "Too Much Delay Fault Coverage Is a Bad Thing," *Proc. Int'l Test Conf.* (ITC 01), IEEE Press, 2001, pp. 624-633.
10. W.C. Lai, A. Krstic, and K.T. Cheng, "On Testing the Path Delay Faults of a Microprocessor Using Its Instruction Set," *Proc. 19th VLSI Test Symp.* (VTS 00), IEEE Press, 2000, pp. 15-20.

**Nisar Ahmed** is a PhD student in the Electrical and Computer Engineering Department of the University of Connecticut. His research interests include design for testability, at-speed testing, and CAD. Ahmed has an MS in electrical engineering from the University of Texas at Dallas. He is a member of the IEEE.

**Mohammad Tehranipoor** is an assistant professor in the Electrical and Computer Engineering Department at the University of Connecticut. He has a PhD in electrical engineering from the University of Texas at Dallas. His research interests include computer-aided design and test, DFT, delay fault testing, test resource partitioning, and test and defect tolerance for nanoscale devices. He is a member of the IEEE, the ACM, and ACM SIGDA.

■ Direct questions and comments about this article to Mohammad Tehranipoor, ECE Dept. of Univ. of Connecticut, Storrs, CT 06268; tehrani@engr.uconn.edu.