Strategies for Low-Cost Test

Rohit Kapur, R. Chandramouli, and T.W. Williams
Synopsys

More collaboration between EDA, semiconductor, and ATE industry segments can improve the manufacturing test environment. These improvements can dramatically reduce the cost of test and lead to a new generation of DFT-aware ATE and ATE-aware DTE.

THE 2001 *International Technology Roadmap for Semiconductors* (ITRS) identifies key manufacturing issues and highlights the cost of test and the associated cost of test equipment. As our analysis of prior roadmap data in Figure 1 (next page) shows, the cost of testing a transistor will equal the cost of fabricating it by around 2012.

Traditionally, various segments of the electronics industry have remained isolated. However, the predicted escalation in the cost of test is creating a new industry coalition of

- the design community,
- electronic design automation (EDA) test tool vendors, and
- test equipment providers.

This informal coalition is addressing various issues that impact the cost of test. Tangible results are possible only with mutual cooperation among these industry segments. This cooperation will make possible design and test technologies that will improve tester performance and decrease the overall cost of test.

Test today

To understand the issues affecting the cost of test, it is helpful to consider the current trends related to test in the semiconductor industry, including various activities in chip design and manufacturing. Not only technical issues affect the cost of test; business issues, such as time to market, can affect test costs as well.

Design methodologies

With the evolution of nanometer technology, the number of transistors on a chip has doubled every 18 months, providing designers with much desired silicon real estate. At the same time, increasing complexity and time-to-market pressures are forcing designers to adopt design methodologies with shorter application-specific IC (ASIC) design cycles. This situation has given rise to system-on-a-chip technology. SoCs use predesigned intellectual property (IP) blocks to build complex functionality and reach markets more quickly.

Reuse is a key component of SoC design.[1] Larger designs are beginning to use predesigned cores, creating myriad new test challenges. Because the user rarely participates in a core's architectural and functional development, the core appears as a black box with known functionality and I/O. Even though core-based design lets designers quickly build products, it requires test development strategies for the core itself and for the entire IC/ASIC with the embedded cores.[2] Reuse methodologies have forced a partitioning of the test data over individual design parts—the IP blocks. This approach directly affects the cost of test in terms of both test application time and test data volume. One of the primary factors affecting the test application time is the level of parallelism—that is, the number of cores the tester can handle simultaneously. The loss of parallelism in the test data dramatically increases test application time.

0740-7475/01/$10.00 © 2001 IEEE

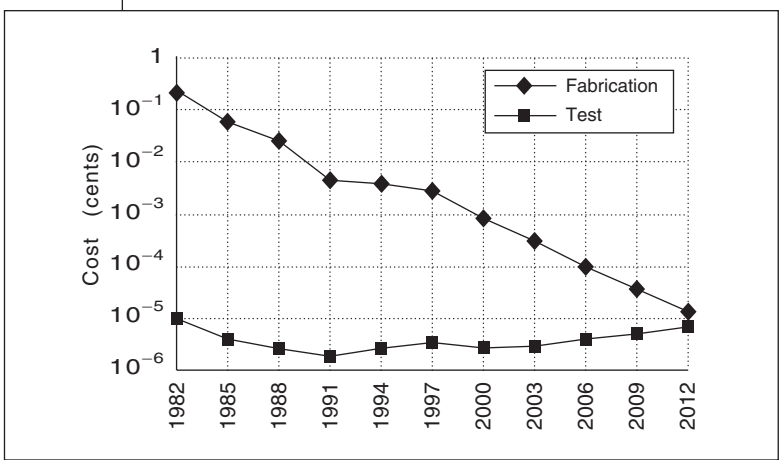


Figure 1. Test and fabrication costs per transistor (derived from the *1997 National Technology Roadmap for Semiconductors*).[3] The cost of testing a transistor is approaching the cost of fabricating it.

Test methodologies

Over the past decade, scan-based design for testability (DFT) has emerged in a wide variety of designs. One key motivator for using scan is the ability to automatically generate test patterns and tests that verify structures such as gates and transistors for scan-based design. Extending these methodologies along with scan-based logic built-in self-test (BIST) for SoC designs engenders other challenges related to the physical implementation of the design, such as layout, power, and area constraints. The resources needed to resolve these issues while implementing test at the SoC level increase the cost of test.

In addition, nanometer technology has a different set of electrical characteristics. These include high quiescent current and, in the physical domain, an increased sensitivity to layout due to test-related signals such as scan chains. Existing methods such as I_{DDQ} and delay fault testing are adapting to this change. In fact, test engineers are adopting I_{DDQ} methods such as ΔI_{DDQ} and statistical analysis to address leakier technologies. Delay fault-testing solutions are dealing with the complexities of crosstalk in nanometer technology. Given the various defects and failure mechanisms in the SoC environment, the ability to debug and diagnose device problems and manufacture the product in volume significantly affects time to market and time to volume. In SoCs, the level and granularity of debugging is important, especially when there are multiple cores, embedded arrays, and user-defined logic. The ability to detect and isolate defects depends on the DFT methodology implemented in the cores, each of which can have different test capabilities. The diagnostics strategy should be similar to the one applied to system-level test. Test engineers should target this strategy toward various testing stages, such as system bring-up, as well as manufacturing, assembly, and field test.

Capital cost

Manufacturing equipment is a key contributor to the overall cost of test. It can be considered a fixed cost—although individual systems vary, depending on the application. For example, applications vary in the number of pins and range of frequencies to be tested. Many existing testers are not tuned to handle devices with built-in DFT. In such cases, test engineers must expend additional effort and become the manual interface between DFT structures and the test equipment. Moreover, testers typically can support a broad range of ICs and test methodologies. Hence, these testers often have

- power supplies that can handle large *di*/*dt*,
- clocking accuracies that match the chip's functional clocking needs,
- high data rate test requirements, and
- low-voltage testing requirements, which are more sensitive to noise.

All these factors increase the cost of a pin designed to provide complete functionality. The number of pins on chips is also increasing. If testers continue to provide everything for everybody on a single machine, tester cost could increase tenfold every 20 years. The cost of probers will also increase. With the increasing number of contact points on a chip and decreasing feature size, the cost of probers could increase twentyfold every 20 years.

Business attitudes also affect the cost of test. Corporations have set their own ad hoc goals, which affect test cost equations. In many cases, the resulting test strategies are suboptimal and can vary with each design.

The lack of any standardized approach to

defining these various factors invariably limits the benefits accrued from lowering test cost. However, various entities in the semiconductor industry, with sometimes diverging visions, are breaking down the barriers and coming together to address the economics of test solutions. Tester manufacturers are working with EDA vendors to develop intelligent interfaces between the design and the automatic test equipment (ATE) hardware to enable better use of testers.

Defining low-cost test

Before attempting to describe solutions for low-cost test, the test industry must agree on what constitutes a low-cost test solution. Low-cost test technology requires cooperation among multiple domains in the semiconductor test life cycle. In this environment, low-cost test requires new functionality to address the following topics.

Predictable test

Test automation is important for designing devices and preparing them for manufacture with predictable quality. The automation process requires implementing a flow that predictably satisfies all design requirements quickly and without excess design iterations or wasted designer effort. These requirements are essential for complex ASICs and SoCs, as are intelligent, up-front planning of design and test architectures and design flows. Key design and test technologies are also necessary. The design and test flow includes

- test-enabled register-transfer-level checking,
- test synthesis,
- DFT links to physical synthesis for placement and layout, and
- automatic test pattern generation (ATPG) for manufacturing test.

Each of these technologies contributes to DFT closure—the completion of all relevant design and test tasks within a single process, and transfer of complete and valid design data to the next process. Designers can then eliminate the risk of long design iteration loops between processes and deliver a design with predictable quality.

Low-tech approach to test complex technology

One approach to lowering the cost of test is to extend the life of existing testers by testing faster, state-of-the-art devices with slower, older testers. This approach will allow testing at a slower speed without compromising test quality. Engineers can test timing-related defects using either

- functional patterns that run at a high speed from the tester, or
- a combination of transition fault and path delay tests that require only critical timing between selected signal edges.

The first approach requires that ATE keep pace with IC technology's increasing speed; the second approach does not. IBM has successfully implemented solutions in the latter area with its reduced pin-count testers and boundary scan technology.[4] Such strategies also extend the life of the testers by letting one tester pin test several IC pins, accommodating the ICs' growing pin counts.

So methodologies that allow for a mismatch in ATE and IC capabilities are possible with the appropriate DFT and ATPG. For DFT-ATPG test patterns to work on the ATE, automation tools will need information about the older testers' limitations. The industry will need standard descriptions to transfer this information from ATE to DFT-ATPG tools, and vice versa.

Efficient test

Low-cost test demands solutions that make the testing process more efficient. Strategies for low-cost test emphasize flows with full-scan designs to address test creation and application time, or flows that address rapid diagnostic turnaround time for volume manufacturing. Test data volume (number of test vectors) has become a critical factor in test cost. It determines the size of the tester's vector memory and the number of tester loads required to complete IC testing.

One technique to reduce the number of test vectors for the desired fault coverage involves adding test points to the design. A test point adds control and observation to the netlist's internal nodes. For example, scan is a special type of test

Low-cost test from an ATE perspective

LeRoy Winemberg
Agilent Technologies

Driven by increasingly complex system-on-a-chip (SoC) devices, automatic test equipment (ATE) vendors face demands from both their customers and the test requirements of their customers' devices. ATE customers are asking vendors for help with mounting economic and technical challenges as growing complexity and increased test development, debugging, and characterization time lead to higher test costs, shorter time to volume (TTV), and shrinking device margins. Moreover, the digital components of SoCs are faster and more complex. These devices require more accurate ATE for characterization and production. Furthermore, the analog portion of SoCs is rapidly increasing as integration of mixed-signal intellectual property (IP) becomes more prevalent. Another complication is design reuse and cores from multiple sources. The design's floorplanning stage must address the ability to control and view all cores, some of which can be deeply embedded in a SoC.

Many semiconductor companies realize that ATE cannot keep pace with the accuracy, frequency, and hierarchical requirements of advanced devices and the cost goals required by shrinking margins. However, most of these same companies still do not have an appropriate level of cooperation between their design and test groups, leading to the all-too-familiar complaint of "untestable" designs being "thrown over the wall" to unsuspecting test groups.

All these challenges compel semiconductor vendors to mature existing DFT, and develop new DFT approaches for the digital portions of SoCs. First and foremost are structural test techniques such as DC scan (for stuck-at fault detection), memory and logic built-in self-test, and I_{DDQ}. The lower accuracy and frequency requirements of these test methodologies enable the use of less restrictive, cheaper ATE hardware. In the case of AC scan (for path delay and transition fault detection), the ATE must deliver accurate high-speed clocks, although an on-chip clock generation circuit can eliminate this ATE requirement. In addition, using more automated test methods can produce significant economic advantages and achieve faster TTV.

For analog IP SoC blocks, mainstream design for testability (DFT) technology that handles more than the most elementary analog blocks (such as phase-locked loops, and

continued on p. 51

point that limits control and observation to flip-flops.[5] Traditionally, test points have been used to increase design coverage. However, if the coverage goals remain constant, test points reduce the number of tests vectors needed to achieve that fault coverage.[6] Because the goal of such technology is to assist automatically generated test vectors, the test point identification technology must be adapted to depend on ATPG. Such technology typically requires tightly coupling ATPG and fault simulation technology for the iterative task of selecting test points.

Steps to lower test costs

Over the last several years, the concept of low-cost test has been attributed mainly to structural test using DFT techniques, which are implemented using EDA test. Recent alliances in the tester, test automation tool, and test methodology industry segments have initiated changes in each of these associated areas. These changes give rise to new interfaces between test automation tools and ATE.

DFT-aware testers

In the early days of chip design, DFT was not integral to design. Most test programs were generated manually, resulting in a sizable investment in test development resources. Today, the test community has moved away from such practices, and DFT has become part of the design methodology, producing designs with high fault coverage.

The most visible trend in test is the creation of DFT-aware testers. These testers rely on chip-embedded DFT and test patterns generated by test automation tools. Creating ATE technology that relies on structures in the tested chip can significantly reduce tester cost.

Conversely, test methodologies are moving away from functional test, permitting increased reliance on testers with lower capabilities during wafer test. This strategy is the most visible aspect of the low-cost test concept. The "Low-cost test from an ATE perspective" sidebar addresses the future of testers and low-cost test.

continued from p. 50

analog-to-digital or digital-to-analog converters) has not yet emerged. Semiconductor manufacturers must continue to fully test their analog cores. Economically testing the "big-A"—analog portions of a SoC—remains an unsolved problem. To reduce the cost of test during wafer probe, some customers are willing to use lower-performance, less-expensive analog instrumentation to automatically test analog block functionality. The specification test is left for the typically more expensive package tester. Other methods to reduce test cost include

- increased use of parallel or multisite testing,
- reduced-pin-count testing,
- concurrent testing of multiple IP cores,
- test scheduling, and
- distributed test.

Agilent and other leading tester companies are researching high-end digital, analog, and radio frequency (RF) DFT. Hence, new DFT tools and IP will become increasingly more available over the next several years. In addition, alliances between electronic design automation (EDA) and ATE companies aim to close the loop between design and test by creating ATE-aware DFT and DFT-aware ATE. Synopsys and Agilent have formed such an alliance.

Several ATE vendors are concentrating on dedicated low-cost test hardware, whereas others offer single, scaleable test platforms that cover the entire cost performance range. Some ATE vendors are working on improving the performance and accuracy of their ATE through higher integration levels, leading to lower test costs. In the area of TTV, companies like TSSI and IMS offer tools that can translate ATE vectors and timing. And new tools, such as Agilent's recently announced SmarTest Program Generator, can automatically generate an entire test program.

Existing DFT, new lower-cost test hardware, and new DFT tools and methodologies from EDA-ATE vendor alliances can all help the semiconductor industry reduce the cost of test and TTV. The challenge for both ATE and design methodology companies is to create a portfolio of test tools, automate the IP insertion and test development tasks, and offer competitively priced test hardware tuned to a particular type of device under test—whether memory, flash, high-speed digital, mixed signal, high-speed serial, or RF.

LeRoy Winemberg is an EDA/DFT scientist at Agilent Technologies in Loveland, Colorado. Contact him at leroy_winemberg@agilent.com.

Linking DFT and ATE

Enabling intelligent communication between DFT tools and testers requires a robust, standardized interface. An IEEE standardization committee consisting of members from industry and academia, is developing a core test language (CTL) to become the basis of such an interface.[7]

Figure 2 shows the transfer of information between the tester and the design environment in low-cost test. This information includes pin limitations related to clock period; waveform generation; and other pin capabilities, such as scan and the memory allotted to the test data for each pin. The information also includes software limitations of the ATE environment that would prevent importing the test patterns into the ATE. As ATE relies more on embedded DFT structures in the design, the standard test pattern information must be augmented to enable the transfer of DFT configuration information.

For example, consider a design that is testable with a configuration that requires applying all scan data through the IEEE 1149.1 tap interface. This design's DFT also lets its test use an efficient parallel configuration of scan chains. Both configurations have their benefits, which ATE can leverage if these configurations are described in an information-rich interface, to provide more efficient solutions. Diagnostics are becoming more important in this new DFT-ATPG link. As ATE and design grow closer, the diagnostic solution requires

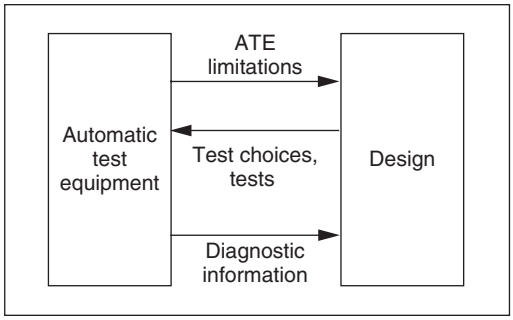


Figure 2. Interface needed between ATE and design/test automation.

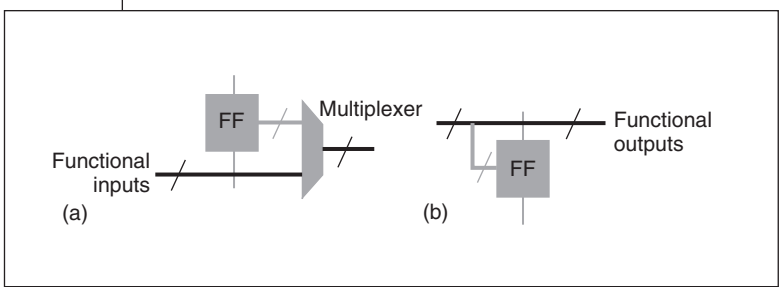


Figure 3. Functional inputs (a) and outputs (b) with reconfiguration structures (shown in gray).

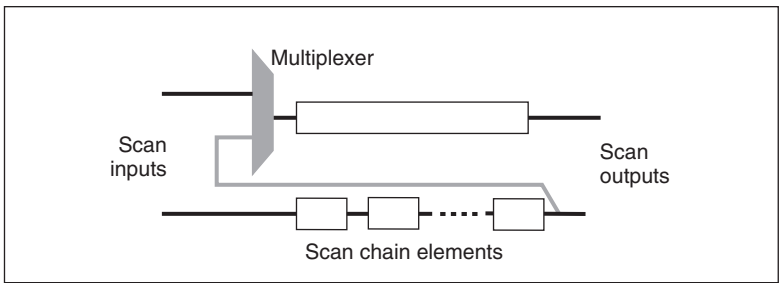


Figure 4. Scan chains with reconfiguration structure (shown in gray) to change the number of scan inputs and outputs.

failure information to pass from the tester to the design environment.

The richness of an interface determines the capabilities that could result from coupling the ATE and the design. Standardizing the format for transferring the data would allow interoperability between tools and associated environments from various vendors.

Retargetable scan patterns

In the coming years, many varieties of testers will be available on the test floor. These might range from small testers with few pins (and comparatively few capabilities) to large testers with many pins and several capabilities. DFT configured to better match the available tester improves tester utilization and optimizes cost.

Retargetable scan patterns—which let test data be applied through different selectable configurations—help maximize the use of tester pins.[8] Multiplexers, controlled by a few selected inputs or signals from a control register, provide reconfiguration.

DFT for retargetable test patterns. There are three types of inputs or outputs for a design:

- control inputs,
- scan inputs and outputs, and
- functional inputs and outputs.

Whereas control inputs typically require control from the design's inputs, the remaining functional and scan inputs/outputs are reconfigurable, thus reducing the number of pins the tester needs.

Functional signals. Figure 3 shows how functional inputs and outputs can be reconfigured to give testers access to the internal signals of the design directly or through scan elements. The multiplexers' select signals (not shown in the figure) determine this configuration.

Scan signals. Figure 4 shows how the device's scan inputs and outputs can be reduced. Two scan chains are multiplexed into a single chain. Once again, the multiplexer's select signals (not shown) determine the configuration.

Chip configurations. Using the techniques shown in Figures 3 and 4, designers can create several design configurations to meet different testers' pin counts. For practical reasons, it makes more sense to limit the number of configurations to work with a low, medium, and high pin count. To allow switching between these configurations at the tester, the control of the configuring multiplexers should be accessible from identifiable chip inputs. If an initialization sequence initiates the control signals, a control register can provide values to the reconfiguring multiplexers.

ATPG for retargetable test patterns. Once the DFT structure is ready, test engineers must create test patterns that can adapt to the different configurations that the on-chip DFT allows. Test patterns comprise

- the test data (the actual 1s and 0s that provide most of the information), and
- the test sequence (the operations required to send the test data into the scan chains and apply the test).

Retargetable test pattern technology changes

the test sequence to accommodate the scan chain's different configurations. Providing the test sequence in a different way permits flexibility at the tester to select the most appropriate DFT and test pattern configuration for maximum tester utilization.

As a basis for this technique, we use the IEEE 1450.1 extension of the Standard Test Interface Language (STIL) because it allows pattern variables and conditional statements. The pseudocode in Figure 5 conveys the test pattern sequences for two configurations, `many_inputs` and `few_inputs`.

If the control signal is set for the `many_input` configuration, the macro's shift operation uses two scan inputs and two scan outputs in its shift operation. The functional inputs and outputs get their values without a shift operation. But when `few_inputs` is selected, the shift operation uses a combined scan chain with scan-in `si1` and scan-out `so2`. Through `funcIn` and `funcOut` scan inputs/outputs, the tester applies the test stimulus for the functional inputs and reads the outputs. This example is just one way to convey the different test pattern sequences for the two configurations.

In the test pattern described by this pseudocode, the previous pattern's scan-out operation overlaps with the current test pattern's scan-in operation. The functional output values (`func_outputs`) are different for the two configurations. Values tagged with `_p` relate to the previous test pattern when the scan-in operation overlaps with the scan-out operation. For the `many_input` configuration, the functional outputs are not scanned, and the values are for the current test pattern. For the `few_input` configuration, the values for the functional outputs are scanned, and come from the previous pattern. Passing a redundant parameter to this sequence makes the selection of the previous or current test pattern's values available for the different ways of applying the test data.

```
MacroDefs {
 Do_one_test {
   W normal_timing;
   C { scan_enable = 1;
     control_inputs = 'control_inputs';}
   If ( reconfig_signal == many_inputs )
   {
     Shift { V { si1 = 'si1'; si2 = 'si2';
       so1 = 'so1_p'; so2 = 'so2_p';
       clk = P;}}
     C { scan_enable = 0; }
     V { func_inputs = 'func_inputs';
   func_outputs = 'func_outputs';}
   }
   If ( reconfig_signal == few_inputs )
   {
     Shift { V { si1 = 'si1 si2';
       so2 = 'so2_p so1_p';
       funcIn = 'func_inputs';
       funcOut = 'func_outputs_p';
       clk = P; }}
     C { scan_enable = 0; }
   }
   V { clk = P; }
 }
}
```

Figure 5. Example of pseudocode conveying the IEEE-1450.1-based sequences for two configurations, `many_inputs` and `few_inputs`.

Flows

Ensuring a viable chip requires reducing the overall time for test activities in the flow. Creating descriptions of the test resources (testers) lets designers tailor DFT implementations to the tester's restrictions. Such an interface can eliminate iterations and allow a single-pass flow for the test patterns. At the other end of the spectrum, flows for diagnostic capabilities can reduce the analysis aspect of test cost.

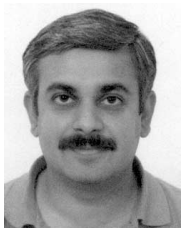
THE INCREASING RELIANCE of DFT and ATE on each other is blurring the associated test boundaries. Rethinking decisions that were based on traditional capabilities will result in the most cost-effective way of achieving high-quality test. For example, designers can place expensive tester-clocking IP on the chip, making IP available to the tester for operation during test. Similarly, DFT can be part of the test equipment to match the throughput for the two entities. In any case, testing must be embedded

in the device itself to increase tester performance and control costs.

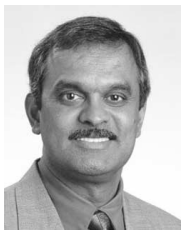
With the cooperation of different industry segments and their common focus on low-cost test, joint activities will likely develop models for quantifying test impact on overall semiconductor cost and time to market. Common units such as dollars/time can help equate the different aspects of chip testing. ■

■ References

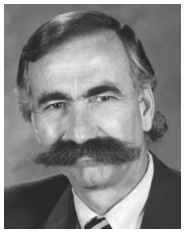
1. *1997 National Technology Roadmap for Semiconductors*, Semiconductor Industry Assoc., San Jose, Calif., 1997.
2. M. Keating and P. Bricaud, *Reuse Methodology Manual*, Kluwer Academic, Norwell, Mass., 1998.
3. E.J. Marinissen, R. Kapur, and Y. Zorian, "On Using IEEE P1500 SECT for Test Plug-n-Play," *Proc. Int'l Test Conf.* (ITC 00), IEEE CS Press, Los Alamitos, Calif., 2000, pp. 770-777.
4. R.W. Bassett et al., "Low Cost Testing of High Density Logic Components," *Proc. Int'l Test Conf.* (ITC 89), IEEE CS Press, Los Alamitos, Calif., pp. 550-557.
5. K.D. Wagner, "Robust Scan-Based Logic Test in VDSM Technologies," *Computer*, vol. 32, no. 11, Nov. 1999, pp. 66-74.
6. M.J. Geuzebroek et al., "Test Point Insertion for Compact Test Sets," *Proc. Int'l Test Conf.* (ITC 00), IEEE CS Press, Los Alamitos, Calif., 2000, pp. 292-301.
7. R. Kapur et al., *IEEE 1450.6 Core Test Language (CTL)*, Synopsys, Mountain View, Calif.; http://grouper.ieee.org/groups/ctl.
8. R. Kapur and T.W. Williams, "Tester Retargetable Patterns," to be published in *Proc. Int'l Test Conf.* (ITC 01), IEEE CS Press, Los Alamitos, Calif., 2001.



Rohit Kapur is a principal engineer at Synopsys. His research interests include VLSI test. Kapur has a BSc in engineering from the Birla Institute of Technology, Mesra, India, and an MS and PhD in computer engineering from the University of Texas at Austin. He chairs the IEEE 1450.6 standard that is creating the core test language.



R. Chandramouli is a product line manager for test automation products at Synopsys. His research interests include SoC test architecture. Chandramouli has a PhD in electrical engineering from Oregon State University. He is a senior member of the IEEE, and chairs the Test Development Working Group for the Virtual Socket Alliance consortium.



T.W. Williams is chief scientist and director of Research and Development of Test Methodology at Synopsys. His research interests include testing, synthesis, and fault-tolerant computing. Williams has a BSEE from Clarkson University, an MA in pure mathematics from the State University of New York at Binghamton, and a PhD in electrical engineering from Colorado State University. He is a fellow of the IEEE and a member of the IEEE Computer Society, Eta Kappa Nu, Tau Beta Pi, the ACM, the Mathematical Association of America, Sigma Xi, and Phi Kappa Phi.

■ Direct questions and comments about this article to Rohit Kapur, Synopsys, Mountain View, CA 94043; rkapur@synopsys.com.