

Trends in Digital Device Test Methodologies

F. Edward Crane, III
Scott Davidson
Jack Kane
Charles E. Stroud
Shianling Wu

The rapid movement toward ultra-large-scale integration (ULSI) is significantly increasing the demands on testing technology for digital devices. Evolving test methodologies must be well integrated into the overall product realization process — from initial concept, through design, to field support. To maintain ever-increasing quality standards, high-quality tests must identify defective units at each level of system hierarchy. In addition, they must isolate faults to allow for efficient replacement and repair. Future demands on testing technology can be divided into six major areas: product quality, device technology, testability scope, process integration, efficiency, and automation.

Introduction

Within the past decade, test technology for digital devices has made great strides in response to the rapid growth in device complexity. Early in the decade, a passive approach was taken, one in which devices, treated as “black-boxes,” were tested by toggling device input pins and matching output responses to expected values. Now, a more aggressive approach is needed, one in which technology and the complexity of devices require that test facilities be built into the device hardware. Testing is no longer a post-process consideration, but rather an in-process necessity. Despite this advance, however, many test problems must still be mastered, especially while the evolution of device technology continues to outpace that of device test technology. In the upcoming decade, the evolution of test will be driven by:

- Product quality,
- Device technology,
- Testability scope,
- Process integration,
- Efficiency, and
- Automation.

This paper describes each of these factors.

Product Quality

Product quality has long been a motivating force in AT&T products, but the focus has expanded as the company has evolved. As

a regulated monopoly, AT&T was charged with maintaining the integrity of the telephone system, and focused heavily on quality to support that goal. Today, AT&T is engaged in a highly competitive marketplace. High quality continues to be essential to meeting or exceeding customer expectations, encompassing not only low parts-per-million defect rates, but also low cost, fast time to market, and complex functionality. Now, test-related work must address all these sometimes conflicting factors.

AT&T's goal is to make best-in-class products with high quality and reliability. Working toward this goal has fostered numerous quality improvement teams, and has led to a new culture of continuous improvement: It is neither practical nor wise to “test” quality into marginal designs. Quality must be recognized as part of the new product realization process. Device designers must create robust devices with an adequate performance margin, one that can be fully tested to assure nearly 100-percent fault coverage. *Fault coverage* reflects how well a given test can detect a particular type of fault or defect in a device. A test has 100-percent fault coverage for a device if the test has ensured that all of certain types of defects or faults have been detected. One type of fault or defect is signal “stuck at” high or low. If a single line is shorted to ground, it is stuck at 0; if it is

shorted to the power supply, it is stuck at 1. Some devices delivered today are so robust that they have very few defective parts. Clearly, these devices must have had nearly 100-percent fault coverage when tested by the device manufacturer.

Printed-circuit-board designers must create robust designs that can be manufactured and thoroughly tested using today's processes. System designers must produce "smart systems" that can monitor their own operations and rapidly, accurately isolate faults. And, because technology does not sit still, all this must be accomplished in a framework of constant change.

It is not practical to "buy" quality. AT&T's design, test, and manufacturing solutions must be cost-effectively built in to enable our products to withstand the test of competition. Tools and techniques must deliver the highest quality at the lowest possible cost, and present the maximum opportunity for reuse throughout their entire product life cycle.

Device Technology

The rapid evolution of integrated circuit (IC) technology, from the 5-micron (μm) design rules of the 1980s to the submicron design rules of the 1990s, has evoked industry visions of chips with small logic cores and vast amounts of unused space. Device designers were motivated by fears of their inability to either define or manage the complexity of designs needed to fill a large submicron IC. Today, as a result of advances in computer-aided design tools (primarily high-level synthesis), logic designers routinely define and manage such large complexities, and continue to stress each new technology to its limits — not only in size, but also in speed of operation. The systems of the 1980s — today's IC chips — continue to create new demands for IC testability.

Next-generation systems will continue to emphasize increased functionality, higher densities, lower costs, shorter design intervals, and increased quality levels. The need to reduce development costs and time-to-market will promote the use of standard products, mask-programmable gate arrays, and field-programmable gate arrays to augment the more customized application-specific integrated circuits (ASICs) within emerging systems. As large digital functions meet the reality of the analog world, many evolving ICs will combine analog and digital circuits. This mix of IC product types will create testability challenges at both the individual component

Panel 1. Abbreviations, Acronyms, and Terms

ASIC	— application-specific integrated circuit
BIST	— built-in self-test
DFT	— design for test
IC	— integrated circuit
MCM	— multi-chip module
ODE	— output data evaluator
OOP	— objected-oriented programming
RAM	— random access memory
TAP	— test access port
ULSI	— ultra-large-scale integration

level and the board and system levels. Solutions will demand standardization of testability strategies and test access, along with thorough testing at each level.

Because test costs increase dramatically at each level of the system hierarchy, individual IC products will be tested rigorously for faults. High levels of integration, made possible by the evolution of IC technology, have impaired observability and controllability of many structures embedded within ICs. Ad-hoc testability techniques of the past, which often yielded low and unpredictable fault coverage, and required extensive design time, will be inadequate. They will continue to be replaced by high-fault-coverage structured testability techniques, such as scan and built-in self-test (BIST), featured in this issue of the *Journal*. (See Figures 1 and 2 for a before and after view of incorporating testability into a device.) The increasing operating speeds of devices and the increasing integration of analog circuitry into digital devices will demand new models for timing and parametric faults. Mixed analog/digital devices require verification of functional, rather than structural, faults, and are often more sensitive to temperature and voltage variations than are strictly digital circuits. Timing and delay faults are being extensively researched today.

High operating frequencies, coupled with ever-shrinking feature size, will cause even digital circuits to exhibit problems that are more characteristic of analog circuits, such as cross-talk between signals. New models need to be created to account for these types of defects. For digital devices, fault coverage of gate-level, single stuck-at faults traditionally has been used to measure device quality during manufacturing, and possibly for board assembly, and beyond. This gauging factor is no

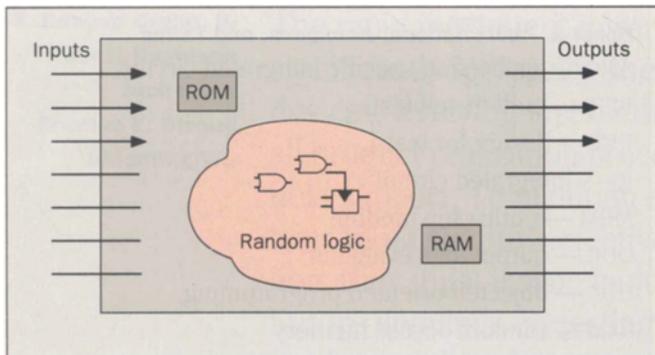


Figure 1. A typical device can be divided into three major building blocks: primary inputs and outputs, random logic, and regular structures. Primary inputs and outputs are used to communicate with the external world. Random logic normally consists of gates and flip-flops that perform control functions. A regular structure can be a memory or a large data path that is regular and dense in nature.

longer sufficient. Its scope has to be expanded as the device technology evolves to include transistor-level faults in complementary metal-oxide semiconductor technology, delay faults in high-performance technologies, and other realistic fault models that accurately represent failure and defect mechanisms.

Testability Scope

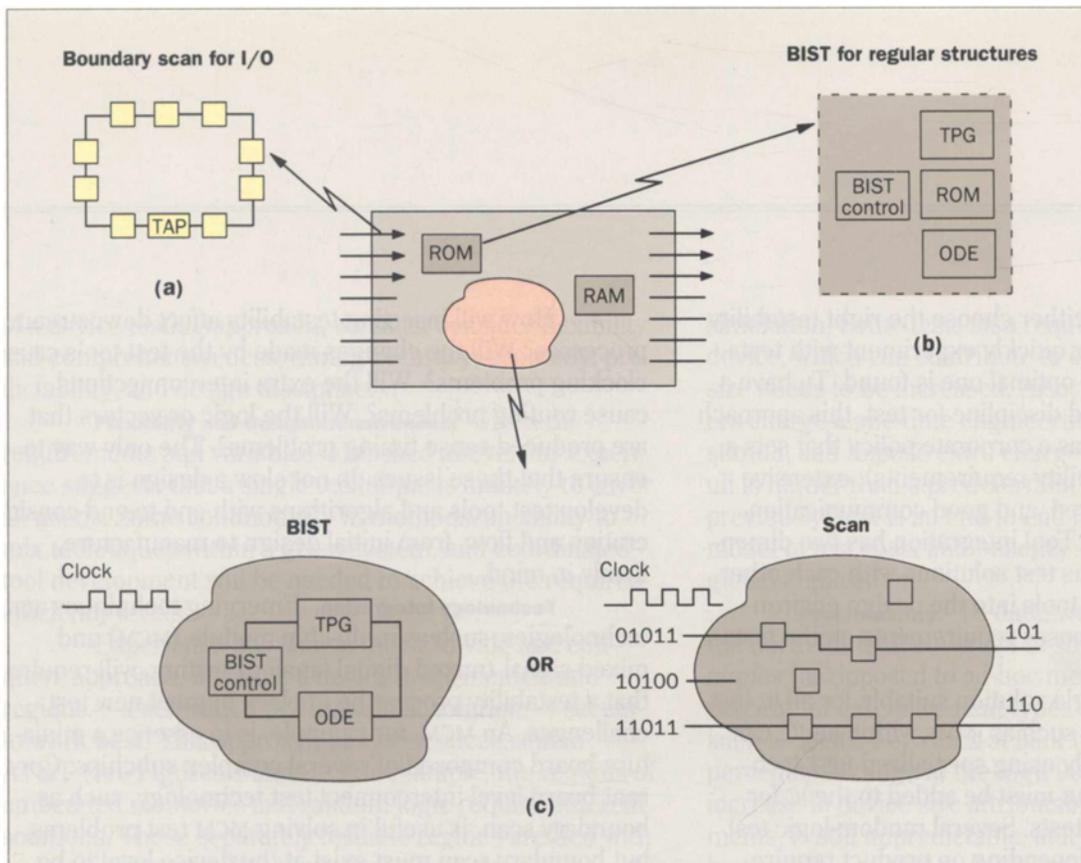
The traditional target of testability techniques, to provide a high fault-coverage test for each IC, is not enough. We must broaden the scope of testability techniques to reduce testing time and cost, and to improve product quality for the entire system during a product's life cycle.

Fault Isolation/Diagnostics. System diagnostic software and tests have become part of testing complex systems, not only in the field, but also during the manufacturing process. Traditionally, they have provided fault isolation to the replaceable unit, such as a printed circuit board, as opposed to the replaceable component, such as an IC. As a result, when defective printed circuit boards are encountered, they can be repaired without additional board-level testing steps, thereby reducing the time and cost of the overall system test/repair program. Similarly, during field testing, the ability of system diagnostics to isolate the faulty IC can reduce the time and cost of factory repair of the faulty printed circuit board. This, in

turn, reduces the "bone pile" of circuit boards that tested well on board-level automatic test equipment in the repair depot or factory, but which, nevertheless, were found to be faulty in system test or in the field. Therefore, testability techniques for diagnosing circuits must not only detect faults in a circuit, but must also isolate them and identify their locations. With these self-testing and self-diagnosis capabilities, systems can some day include self-repair. This will reduce the time and cost of maintenance, while increasing system availability.

Burn-in and Environmental Stress Testing. At both device- and board-level testing, the scope of testability techniques must be broadened to include burn-in, environmental stress, and electrical parametric testing. However, we must develop a better understanding of the types of defects these tests detect, their effectiveness, and their cost. BIST techniques, which have been used successfully in ASICs for device-level burn-in and environmental stress testing,¹ generate random, or almost random, test patterns for units being tested. These patterns are more effective for burn-ins and environmental stress testing than typical patterns written for system test or diagnostics purposes, because random and almost random patterns tend to exercise a unit to its fullest extent. The higher data activity created during BIST increases power dissipation and transistor junction temperatures, which stress the ASIC further. In addition, BIST capabilities have sometimes improved stand-alone, simultaneous testing of many devices in an environmental chamber without the need for expensive test equipment. Yet, BIST traditionally has been designed for single stuck-at fault testing in digital circuits, not to test new elements in mixed-signal and analog circuits.

Fault Tolerance. Increased density, made possible by the continually shrinking feature sizes in ULSI circuits, heightens the probability of defects in increasingly expensive ICs. Simply put, yields can be expected to decrease just when high yield is most vital. Fault- and defect-tolerant circuits will become common, if not absolutely necessary. We see this already in the designs of high-density random access memories (RAMs). In fault- and defect-tolerant environments, testability techniques must be able to detect and distinguish between circuits that contain faults/defects and circuits that are properly masking faults/defects, while providing the correct operations. The ability to detect the existence of faults/defects in a fault- and defect-tolerant circuit



designed to mask faults/defects presents a challenge to most existing testability approaches. Still, it will be necessary, by means of manufacturing tests, to ensure the high reliability for which the circuit is designed.²

Off-line and On-line Testing. To increase system reliability further, in/out-of service (on/off line) testing must be carefully balanced. This balance can lead to reductions in the resources needed for testing. Currently, a portion of the circuitry in a system is dedicated to concurrent fault detection and diagnostics for on-line testing, and for other techniques. These techniques, detailed in other papers in this issue of the *Journal*, include BIST, full or partial scan design, and boundary scan for off-line testing. While these testing approaches complement each other's testing capabilities, used alone they rarely support both on-line and off-line testing. Improving existing techniques, or developing new techniques that can be used for both on-line and off-line testing, can reduce the area overhead of the hardware dedicated to testing, as well as the cost of the system.

Process Integration

The diversity of designs within AT&T requires optimal test strategies for a wide range of design styles, cost, area and performance targets, and customer requirements.

General Considerations. The test process must be designed to support applications of new test solutions for

Figure 2. Based on the divide and conquer concept, each building block needs its effective testability solution(s). For example, the (a) IEEE 1149.1 boundary-scan standard is suitable for primary I/O, because it provides a gateway to internal device testing, as well as testing of interconnections between the device and its neighbors in support of board-level testing, and beyond. (b) BIST is strongly recommended to test regular structures. With its three main blocks — the test pattern generator, output data evaluator, and controller — BIST provides the most effective, least expensive test quality for regular structures. BIST is also reusable for board-level testing, and beyond. (c) Either scan or BIST can be used for random logic testing. Descriptions of each testability solution are detailed in other papers in this issue of the *Journal*.

continuous or discontinuous improvements in test, with minimal effect on the user. BIST/scan/boundary scan must, for example, be inserted into existing design environment libraries and existing processes, and the resulting designs must be tested on existing automatic test equipment.

One approach to this problem has been to build a set of test tools that have diverse capabilities, but a similar look and feel. These test tools provide a range of solutions, from support for manual testability improvements with no additional logic, to automatic insertion of high-fault-coverage BIST. Ideally, they are integrated so

that a design team can either choose the right testability solution immediately, or quickly experiment with testability solutions until the optimal one is found. To have a well-defined process and discipline for test, this approach requires a policy (such as a corporate policy that sets a high standard for testability requirements), extensive training for those involved, and good communication.

Tool Integration. Tool integration has two dimensions: integrating various test solutions with each other, and integrating the test tools into the design environment. Each of these imposes requirements on the testability process.

There is no single solution suitable for all IC testing. Regular structures, such as RAMs within an IC, can be tested most effectively using specialized BIST techniques.^{3,4} Boundary scan must be added to the IC for board and system level tests. Several random-logic test solutions can be used, depending on product requirements,⁵⁻⁷ all of which must work together. Regular structure BIST sessions are initiated and controlled using boundary scan logic. Random logic BIST and scan can also be controlled from the boundary scan, which saves package pins. There must be system-level access, however, to the IC test circuitry.

Although most of these schemes differ in test technology, all involve analysis of the circuit, followed by incorporation of the test solution. Software reuse to support these common capabilities provides a natural integration and a platform for the rapid introduction of new testability solutions.

The second dimension is integration of the test tools with the design environment. Test tools must smoothly accept the device description and vendor libraries used by the design. The vectors produced must be compatible with simulators and the test equipment of choice. These factors are the bare-minimum requirements for integration. Test tools should also integrate into the framework provided by a design tool vendor to provide a common user interface.

How does inserting testability affect other tools? Adding test logic can cause a carefully optimized design to miss performance targets. The test tools must also interact with synthesis and timing analysis tools to ensure that circuit elements whose timing is critical are not modified. Tools should be flexible enough to get good fault coverage even when parts of the circuit cannot be modified.

How will inserting testability affect downstream processes? Will the changes made by the test tools cause clocking problems? Will the extra interconnections cause routing problems? Will the logic or vectors that are produced cause timing problems? The only way to ensure that these issues do not slow a design is to develop test tools and algorithms with end-to-end consideration and flow, from initial design to manufacture, firmly in mind.

Technology Integration. Emerging techniques and technologies, such as multi-chip-module (MCM) and mixed-signal (mixed digital/analog) testing, will require that a testability process be in place to meet new test challenges. An MCM, for example, is in essence a miniature board composed of several complex subchips. Current board-level interconnect test technology, such as boundary scan, is useful in solving MCM test problems, but boundary scan must exist at the device level to be available at the MCM level. This is equivalent to applying board-level test technology to a chip.

Last, but not least, the test process cannot be open-ended. To reap the benefits of testing — detecting process and product weaknesses that need improvements, increasing product yield, and decreasing product cost — test effectiveness must be provided to vendors, designers, and manufacturing engineers.

Efficiency

In the next decade, test solutions will call for efficiency, as well as solutions that are “smart,” flexible, easy to use, and inexpensive. A few examples of the variables to consider when deriving an effective solution for testing are:

- Design specification, such as operating range, which will render some well-known solutions unworkable;
- Design styles, which vary greatly from one organization to another;
- Test requirements from subsequent levels (e.g., MCM, board, system, and field). They do vary and may conflict, making them sometimes difficult to meet fully;
- Automatic test equipment requirements and limitations;
- Design for manufacturability rules; and
- Failure mechanisms.

To be effective, the solution must ensure reliability, improve yield, and reduce testing cost, while simultaneously addressing these variables. In evaluating efficiency

of a device-testing approach, we must consider flexibility and comprehensiveness, timing accuracy, economy, predictability, and design discipline.

Flexibility and Comprehensiveness. Given the requirements and variables described above, our experience suggests that a single technique is unlikely to cover all needs. Some commonality in methods, an ability to mix techniques within a given system, and coordinated tool development will be needed to achieve the required efficiency levels.

Experience has shown that a “divide and conquer” approach, in which a device is segmented into regions — each with a dedicated test solution — seems to work best. This approach has been successful in AT&T (see Figures 1 and 2). For example, the regions of embedded memories and random logic require different solutions. These separately testable regions are tied with an intelligent controller that executes the testability. Because memories are denser than random logic, and are subject to defects that are not significant in random logic, the optimal solution must address that particular set of failures. For random logic, there are various schemes to choose from, depending on the circuit. Deterministic, random, exhaustive, weighted random, and other test solutions have been proposed, and are effective for certain types of designs. Each solution has certain area and performance implications for the device. The key to an effective, efficient test solution is a suite of methodologies and tools. Designers can use these to evaluate the tradeoffs between solutions in a timely fashion, and, early in the design process, make quick, correct decisions on testability.

Accuracy. Timing accuracy will be critical as operating speed increases. As usual, clock skews induced by test schemes must be avoided. At-speed testing will weed out performance-sensitive devices. As we noted previously, fault models must be extended from traditional, stuck-at faults to delay faults. These can model abnormal operations when unacceptable delays exist within elements, or within connections between elements.

Economy. Elements contributing to cost exist not only in the most obvious items, such as chip area, but in other items as well. These include assessing non-recurring engineering effort in test-stimuli preparation (either manually, by software, or by dedicated test hardware embedded within the device), and determining the effectiveness of the test stimuli using a lengthy fault

simulation. Some tests also require extra test pins on a device, which can contribute to extra cost if the package size needs to be increased. Also, some chip manufacturers charge a one-time engineering fee for handling test stimuli, and impose extra charges if the size of test stimuli is higher than a predetermined threshold. As stated previously, test is an end-to-end issue, and an economic model of test costs and benefits should consider these global implications.

Predictability. To date, we have determined that the norm for fault coverage of structural testability techniques (as opposed to ad-hoc methods) is more than 99 percent for extensive fault types with regular structures, such as RAMs. For stuck-at fault types with random logic, percentages range in the high 90s. *Area overhead*, the increase in device size attributable to testability enhancements, is still unpredictable, at least for random logic, because it is related to the “gate to flip-flop ratio” of the circuit under test. There are solutions with fixed costs, such as those in which fixed-size test-pattern generators and output evaluators are inserted. Because the percentage of area used for these solutions decreases as the circuit gets larger, they seem attractive.⁸ The resultant test quality is, however, highly circuit-dependent. Other techniques, such as inserting test points to enhance controllability and observability, are used to increase fault coverage. The locations and numbers of test points needed are circuit-dependent, and can make prediction of device area difficult.

Design Discipline. We are far from having solutions and tools that will fix all testability problems for any circuit. All solutions and tools impose restrictions on a design. This is not necessarily a limiting factor, because it ensures good design discipline. For example, it is difficult to use testability tools in asynchronous designs, such as those with mixed clock and data. If such regions exist in practical circuits, the solution needs to provide “isolation” capabilities that simplify the test circuit, at the cost of reducing fault coverage. The designer then must ensure that these additional faults are detected by manually written test vectors. In general, the more regular the design, the faster the tools can achieve high fault coverage. Unfortunately, certain products may have features that are inherently “difficult to test,” such as long counter chains, internal generation of clocks, etc. In these cases, the design margin must be large enough to accommodate additional test features.

A set of comprehensive, flexible test solutions should result in widespread use of the technology. This, however, must be coupled with extensive awareness training and test education, and an understanding of how they affect design.

Automation

Finally, high degrees of automation with "intelligent" tools — those that offer structure, yet allow flexibility — will be essential to achieving system testability goals. The tool suite must include, at minimum:

- Testability analysis and audits;
- Design for test (DFT) synthesis and incorporation (of internal scan, BIST, boundary scan, control logic, and other future techniques);
- Test generation;
- Fault simulation; and
- A framework that can easily integrate other tools.

There are three software building blocks: (1) test generation and fault simulation that are computationally intensive; (2) object-oriented programming (OOP) for testability incorporation; and (3) a set of software class libraries that support a framework for integration.

Some test generation and fault simulation runs can take days or weeks to complete. These are considered computationally intensive. Hardware accelerators are sometimes used for fault simulation, but they tend to be expensive and inflexible. Event-driven and timing-driven simulation techniques are used for fault simulation. A recent breakthrough in the fault-simulation algorithm used in AT&T has shortened the run time by a factor of ten over the traditional fault-simulation method.^{9,10}

Test generation, also discussed in this issue, is time-consuming because a large amount of solution space must be searched. Various pruning and learning techniques have been explored in the search for run-time efficiency.¹⁰⁻¹² More importantly, good results are not guaranteed in a reasonable time unless some type of testability is used, such as scan or BIST.

In DFT incorporation, OOP is intuitive in that circuit elements and their connections are naturally represented as software data elements. Further, all element types can be modified with operations such as cut and connect. OOP has proven effective in fostering software reuse and quick implementation of novel DFT techniques. Based on the experience that no single technique is

suitable for all designs, quick prototyping is essential for identifying a set of best techniques.

A set of software class libraries support a framework for inter-tool communication and integration. The framework enables testability techniques to be driven by synthesis and/or layout, taking into account design and device manufacturing constraints. It also allows designers to compare the test intent to the original design intent and annotate their findings.

Future trends in automation will focus more on exploiting advanced software algorithms and programming techniques. Testability must respond as design information migrates to ever-higher levels of representation. This high-level circuit, more abstract than the commonly used gate-level circuit, may be behavioral or semi-structural, and is used as an input for logic synthesis. The idea that a high-level circuit representation makes test generation easier than a gate-level representation is appealing. Unfortunately, it has not been achieved after more than ten years of effort, because search space at a high level is just as large at the gate level, but more difficult to prune. Work is beginning on a more effective approach, DFT insertion (either BIST or scan) at the high level.

Software techniques that automatically suggest an optimal solution for a given device, such as simulated annealing and artificial intelligence techniques, should be explored. This will help novice users master testability.

Both functionality and testability will be fighting for the timing and area margin in speed-critical designs. Little space will be devoted to testability solutions in high-speed circuits. For such cases, device-level testability may take place in neighboring chips, implying that tools must consider board-level topology.

Using DFT in mixed-signal devices will pose a new challenge. Normally, it can be effective to use the digital portion to test analog sub-blocks. If so, the tools must integrate analog tool suites and be able to select schemes that work across digital/analog boundaries.

One would hope that a device test standard comparable to the successful IEEE-1149 standards will emerge.¹³ The standard will enable device test intent to be readily accessible at various stages of product manufacture. Other potential uses, from diagnostics and repair to component level, will become a reality, even with chips from multiple vendors. It is crucial for designers of future tools to track the trends of the standards.

Conclusion

Our vision of digital IC test extends beyond device manufacturing to encompass board, system, and field considerations, and to ensure product quality, reliability, and the ability to diagnose and repair circuits. Software will continue to play a significant role in automation, with a greater emphasis on exploiting state-of-the-art techniques, such as OOP and artificial intelligence, to explore the maze of complex test solutions in a speedy fashion. As more designers become familiar with testability, we hope that they will provide the test community with new ideas to improve the test methods and processes needed to face the challenge of the next decade.

References

1. C. E. Stroud and G. Liang, "Design Verification Techniques for System Level Testing Using ASIC Level BIST Implementations," *Proceedings of the IEEE International ASIC Conference*, Rochester, New York, September 27 - October 1, 1993, pp. 140-144.
2. C. E. Stroud and A. E. Barbour, "Testability and Test Generation for Majority Voting Based Fault Tolerant Circuits," *Journal of Electronic Testing: Theory and Applications*, Vol. 4, No. 3, August 1993, pp. 201-214.
3. S. K. Jain and C. E. Stroud, "Self-Testing of Embedded Memories," *IEEE Design and Test of Computers*, Vol. 3, No. 5, October 1986, pp. 27-37.
4. D. R. Aadsen, H. N. Scholz, and Y. Zorian, "Automated Built-In Self-Test for Regular Structures Embedded in ASIC Devices," *AT&T Technical Journal*, Vol. 69, No. 3, May/June 1990, pp. 97-109.
5. M. Pradhan, et al., "Circular BIST with Partial Scan," *Proceedings of the IEEE International Test Conference*, Washington, D.C., September 12-14, 1988, pp. 719-729.
6. E. Wu and P. W. Rutkowski, "PEST - A Tool for Implementing Pseudo-Exhaustive Self-Test," *Proceedings of the European Design Automation Conference*, Glasgow, Scotland, March 12-15, 1990, pp. 639-643.
7. C. J. Lin, S. Bhawmik, and Y. Zorian, "PSBIST: A Partial Scan Based BIST Technique," *Proceedings of the IEEE International Test Conference*, Baltimore, Maryland, October 17-21, 1993, pp. 507-516.
8. C. E. Stroud, "Built-In Self-Test for High Speed Data Path Circuitry," *Proceedings of the IEEE International Test Conference*, Nashville, Tennessee, October 26-30, 1991, pp. 47-56.
9. F. Maamari and J. Rajski, "The Dynamic Reduction of Fault Simulation," *IEEE Transactions on Computer-Aided Design*, Vol. 12, No. 1, January 1993, pp. 137-148.
10. S. Davidson, et al., "Automatic Test Generation for Digital Electronic Circuits," *AT&T Technical Journal*, Vol. 73, No. 2, March/April 1994, pp. 19-29.
11. R. Bencivenga, T. Chakraborty, and S. Davidson, "Architecture of the GENTEST Sequential Test Generator," *1991 Custom Integrated Circuits Conference*, San Diego, California, May 13-15, 1991, pp. 17.1-17.4.
12. S. Mallela and S. L. Wu, "A Sequential Circuit Test Generation System," *International Test Conference*, 1985, Philadelphia, Pennsylvania, November 19-21, 1985, pp. 57-61.
13. *IEEE Std 1149.1-1990 Test Access Port and Boundary Scan Architecture*, IEEE Computer Society, May 1990.

(Manuscript approved January 1994)

F. Edward Crane, III is a technical manager in the Test Strategy and Process Department of AT&T Network Systems Transmission Systems Business Unit in Merrimack Valley (North Andover), Massachusetts. He manages test strategy at the Merrimack Valley Works and chairs two Network Systems Technology Board committees, one on built-in self-test (BIST) and boundary scan, and the other on product reliability. Mr. Crane joined AT&T in 1970, after receiving a B.S.E.E. from the University of Florida, Gainesville.



Scott Davidson is a technical manager of the Design for Testability and Test Data Management Group at the Engineering Research Center of AT&T Bell Laboratories in Princeton, New Jersey. His group develops digital test algorithms and tools. Mr. Davidson received a B.S. from the Massachusetts Institute of Technology in Cambridge, an M.S. from the University of Illinois, Urbana-Champaign, and a Ph.D. from the University of Southwestern Louisiana in Lafayette, all in computer science. He joined AT&T in 1980.



Jack Kane is a technical manager in the Very Large-Scale Integration (VLSI) Design Department at AT&T Bell Laboratories in Allentown, Pennsylvania. He is responsible for the design of complementary metal-oxide semiconductor (CMOS) application-specific integrated circuits, and the management of testability issues related to these products. Mr. Kane received a B.S. in engineering from UCLA, Los Angeles, California, and an M.S. in electrical engineering from Stanford University, Palo Alto, California. He joined AT&T in 1968.



Charles E. Stroud is an assistant professor in the Department of Electrical Engineering at the University of Kentucky at Lexington. Before leaving AT&T in July 1993, he was a distinguished member of technical staff at AT&T Bell Laboratories in Naperville, Illinois. For his 15 years at AT&T, Mr. Stroud was an application-specific integrated circuit (ASIC) designer for 5ESS[®] switching systems. He worked on VLSI design and testing, including self-testing, diagnosis, and repair, as well as fault-tolerant VLSI implementations. Mr. Stroud received a B.S. and M.S. in electrical engineering from the University of Kentucky, and a Ph.D. in electrical engineering and computer science from the University of Illinois, Chicago.



Shianling Wu is a technical manager in the Engineering Research Center of AT&T Bell Laboratories at Princeton, New Jersey. She is responsible for research and development of computer-assisted technology and tools, as well as their deployment and applications. Ms. Wu received a B.S. from City College of New York, New York, and an M.S. from Columbia University, New York, New York, both in computer science. She joined AT&T in 1979.

