Pioneering engineers begin to adopt board-level automatic test generation

Dan Strassberg, 
Associate Editor

You can recognize a pioneer by the arrow in his back: Despite that adage's implied warning, some designers of board-level products are beginning to adopt design methodologies compatible with testability techniques like automatic test-pattern generation (ATPG).

These pioneers might well acquire a few surface wounds. But proponents of design for testability, or DFT (see box, "Overcoming the initial problem"), maintain that companies that delay or avoid confronting testability at the design stage risk much more serious—and possibly even fatal—damage.

In its most elemental form, DFT yields designs that are partitioned by function, that can be easily initialized and controlled, and whose behavior can be observed. Some people, however, use the term DFT to imply considerably more.

For example, DFT approaches include such highly formalized design techniques as LSSD (level-sensitive scan design), one of several scan-design methodologies. With the exception of boundary scan, which is a technique for isolating functional blocks and observing them at their I/O terminals, scan design enables you during test to reconnect sequential logic (which most ATPG programs can't handle) to form combinational networks (which ATPG programs can handle).

DFT will change attitudes

Test vectors, also called test patterns, are the combinations of 1s and 0s that a tester applies and looks for when it tests a logic net. Although adherence to DFT approaches in their simplest form doesn't guarantee success in generating vectors automatically, it improves the chances of success and can significantly reduce test-development time, even with manual vector generation. To achieve its goals, DFT imposes new disciplines on designers and requires them to reorient their priorities. DFT advocates insist that new attitudes are mandatory—they insist that tried-and-true approaches (for example, a design engineer's treating test as someone else's problem) simply won't work in today's market.

Many factors influence the pace at which companies are adopting DFT and ATPG, particularly for board-level designs. Although ATPG and DFT have not yet taken device design by storm, these techniques have so far affected the design of ASICs and PLDs much more than they have that of board-level products (Ref 1).

Despite offering solutions to very real and pressing problems, vendors of ATPG software and other test-development aids recognize that if they want to grow they must deal with several legitimate concerns. They must also educate their potential customers and thus dispel a long list of misconceptions. Here are some of the issues they are addressing:

- Greater complexity of boards than of ICs
- The perception that ASIC design is risky enough to justify unusual approaches, but board-level design is not
- Difficulty in measuring costs

This board test system, Hewlett-Packard's 3065ST, performs digital in-circuit and analog-functional tests. An automatic program generator for it runs on the vendor's workstations.
of current test and test-development practices
- Use of traditional methods, because some companies don’t understand the shortcomings of current methods and so they feel comfortable with them
- Cost of learning how to apply new techniques
- Cost of computing resources required to run ATPG software
- Fear of requiring more PCB-board real estate
- Fear of higher component cost
- Fear of lower reliability
- Slow adoption of logic simulation
- Incompatibilities of simulator/ATPG-database formats
- Incompatibility between automatically generated vectors and tester capabilities
- Difficulty in obtaining sellable components.

Big boards challenge ATPG

Although ASICs and PLDs are complex, with gate counts in the neighborhood of 10,000, they tend to be significantly less so than the typical logic board designed with CAE tools. Such boards—especially if they house complex chips—can easily contain more than 100,000 gate equivalents. The diversity and complexity of board-level designs taxes many of today’s ATPG programs beyond their limits. But if you design in partitionable functional blocks, you can frequently use ATPG on networks that at first glance seem to exceed the capabilities of the software.

Board-level test-development procedures evolved at most equipment-manufacturing companies before board complexity approached today’s levels. Many of these companies have little idea of the cost of developing or performing board-level tests, and their managers have not yet recognized the inadequacy of the classical approaches they are using. Naturally, if they are unaware of the shortcomings of existing practices, managers can see no reason to make changes.

But unlike printed circuits, ASICs are a new technology—one which engineers and managers realized warranted a complete set of new tools. These tools now exist, and ironically, are ready to be applied to the older, PCB-design technology, if only engineers and managers realize the techniques’ potential.

Overcoming the initial problem

The electronics industry has its share of confusing acronyms and abbreviations, but few seem quite so unfortunate as DFT—used throughout this article, and throughout the industry, to mean design for testability. Practically everyone who refers to design for testability as DFT uses those same initials to denote digital functional test and sometimes dynamic (as opposed to static) functional test, a subset of digital functional test. To an even larger group of EE’s, DFT means discrete Fourier transform. It’s quite likely that someone reading this article is planning to employ DFT to test a high-speed signal-processing board that performs DFTs, and because of the board’s complexity, he plans to make DFT a guiding principle in his project. Of course, his PCB (printed-circuit) vendor probably uses PCs (personal computers) for both PC (process control) and PC (production control). Now, is that perfectly clear?
the situation is—or appears to be—different, so many of the same companies are unwilling to apply ATPG to their board-level designs.

Much of the automated test generation for ASICs occurs as an offshoot of logic and timing simulation, which are part of the ASIC design phase. The patterns generated are for design verification; they tell a designer primarily whether his design will do what he intended it to—assuming that the manufacturing process hasn't introduced any defects.

At best, such pattern sets provide an incomplete solution for production test or incoming inspection. In fact, some CAE-software vendors insist that you only cause confusion if you use the term ATPG to refer to development of test patterns for design verification. They say that you should use ATPG only to describe the generation of vectors for determining whether manufacturing has introduced defects. (Sometimes such vectors also locate the defects.)

If you use design-verification test patterns in production board test, you can miss faults introduced during board-loading and soldering, and you can waste time checking device properties that you should have checked before you loaded the devices onto the board. But although design-verification patterns are usually inadequate for production test, the converse isn't true. You don't develop production test vectors to find functionality problems in boards without manufacturing-induced faults, but vectors developed for production test frequently reveal problems that design verification should have uncovered.

Another factor inhibiting adoption of ATPG is the slow pace of acceptance of logic simulation itself. If you wanted to use ATPG to generate a board's test vectors, you would almost certainly design the board using simulation; otherwise you would have to create the equivalent of the simulation database from scratch. So far, even for complex boards, use of logic simulation remains in its infancy.

The cost of most ATPG software is not insignificant (Table 1). But some potential customers worry about having to commit great sums for computer resources (CPUs, data-storage hardware, and workstations) to run ATPG programs. For companies using simulation, however, the resources may cost nothing—at least at first. Usually, there's no need to augment a computer system until you are utilizing it fully. On the other hand, if you didn't originally factor ATPG into your plans but are now using it, you may have to acquire additional computer equipment sooner than you had expected to. To the relief of engineers concerned with controlling the cost of individual projects, most companies recognize that such equipment is used on many projects and is part of the cost of being in business; rarely do they assign the charges to a single development program.

Fear of the unknown as well as engineers' innate conservatism affects their willingness to make changes. In the case of DFT, management fear that the time designers spend in learning how to apply the new techniques will add to project costs and slow down product-development schedules. Something to keep in mind is that an organization normally needs to climb the DFT learning curve only once—not once per project. After engineers have learned the techniques, the organization reaps additional benefits with each new project.

Designers fear that testable designs will require more printed-circuit real estate and have higher parts cost and possibly lower reliability than equivalent "untestable" designs. Of these concerns, the one about reliability is easiest to disprove: With appropriate components, the added complexity of a scannable design (measured by gate count) is usually less than 10%. A cost attributable to the added complexity is an estimated 10% decrease in MTBF (mean time between failure); but a benefit is an
TECHNOLOGY UPDATE

Estimated order-of-magnitude reduction in MTTR (mean time to repair). MTTR is usually a small fraction (less than about 1%) of MTBF. Suppose an untestable product has an MTBF of 1000 hours and an MTTR of 10 hours. Its availability is 100(1000−10)/1000, or 99%. If the

| TABLE 1—REPRESENTATIVE BOARD- AND DEVICE-LEVEL TESTABILITY PRODUCTS |
|----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| VENDOR          | PRODUCT         | DESCRIPTION                                              | PREREQUISITES | US BASE PRICE  |
| AIDA CORP       | AIDA ATPG       | GENERATES PATTERNS WITH 100% COVERAGE OF DETECTABLE FAULTS IN SCANNABLE DESIGNS. USES PATH-SENSITIZING AND RANDOM-VECTOR GENERATION. DETECTS "STUCK-AT" FAULTS. SUPPORTS BIDIRECTIONAL IO AND 3-STATE BUS AS WELL AS WIRE-OR AND WIRE-AND GATES. | APOLO OR SUN WORKSTATION AND AIDA FAULT SIMULATOR | $46,000 (AIDA FAULT SIMULATOR: $15,000) |
| ANVIL SOFTWARE  | ANVIL ATG SOFTWARE | GENERATES FUNCTIONAL TESTS FOR PROGRAMMABLE LOGIC DEVICES WITHOUT USING PRELOAD. SUPPORTS >50 DEVICE TYPES. LINKS TO COMPONENT AND IN-CIRCUIT TESTERS. | IBM PC OR COMPATIBLE, 8MB RAM AND HARD DISK | $49,950 |
| GATEWAY DESIGN AUTOMATION CORP | TESTSCAN | ATPG FOR DIGITAL CIRCUITS EMPLOYING SCAN DESIGN. PROVIDES 100% TEST VECTOR COVERAGE, DESIGN-RULE AUDIT, AND FAULT SIMULATION. SUPPORTS SCAN-PATH, LSSD, RANDOM-ACCESS SCAN, AND SCAN-SET TECHNIQUES. | IBM, APOLO, SUN, OR VAX/VMS | $75,000 |
| GATEWAY DESIGN AUTOMATION CORP | BITGRADE | FAULT SIMULATOR AND RANDOM-VECTOR GENERATOR FOR CIRCUITS WITH BUILT-IN SELF-TEST (BIT) ABILITY. INCLUDES HARDWARE DESCRIPTION LANGUAGE. | IBM, APOLO, SUN, OR VAX/VMS | $25,000 |
| GENRAD INC      | GENESIS         | TEST PROGRAMMING ENVIRONMENT FOR GR 275X SERIES OF HIGH-PERFORMANCE BOARD-TEST SYSTEMS. GENERATES MULTISTRATEGY (IN-CIRCUIT/FUNCTIONAL) TEST PLANS RUNS IN TESTER OR OFF-LINE IN WORKSTATION. | SUN WORKSTATIONS WITH SUN UNIX | INCLUDED IN PRICE OF TEST SYSTEM; NOT AVAILABLE SEPARATELY |
| HEWLETT-PACKARD CO | HP 7421A | AUTOMATICALLY GENERATES READY-TO-RUN IN-CIRCUIT ASIC- AND FUNCTIONAL CIRCUIT-TEST PROGRAMS THAT ARE FULLY COMPATIBLE WITH HP-3065 SERIES TESTERS. | HP ELECTRONIC DESIGN SYSTEM; HP 3065-SERIES TESTER | $39,950 |
| HBB SYSTEMS INC | THESEUS         | ANALYSIS TOOL THAT PINPOINTS AREAS NEEDING IMPROVED TESTABILITY ASSURES THAT VECTORS PROVIDE GOOD FAULT COVERAGE. HANDLES DESIGNS USING SEQUENTIAL DEVICES, FEEDBACK LOOPS, AND RECONVERGENT PATHS. | DEC VAX/VMS OR SUN/UNIX WITH 8MB BYTES RAM AND 300MB BYTE HARD DISK | $100,000 |
| LOGICAL SOLUTIONS TECHNOLOGY INC | TESTABILITY CHIP SET | TESTABLE FUNCTIONAL CIRCUITS (SHIFT REGISTERS, LATCHES, DECODERS, ETC.) THAT PROVIDE PERFORMANCE PLUS A TESTABILITY BUS PORT. COMMERCIAL AND MIL-LEVEL AND SURFACE-MOUNT VERSIONS. | NONE | $6 TO $60 (100) |
| NCR CORP        | VITEST          | FACILITATES DEVELOPMENT OF HIGH-SPEED TEST PROGRAMS TO RUN ON TRILLIUM AND SCHLUMBERGER SENTRY DEVICE TESTERS. CHECKS PATTERN TIMING AND BUS CONTENTION. USES BEST AND WORST-CASE SIMULATIONS TO ANALYZE PATTERNS IDENTIFIES POSSIBLE FUNCTION ERRORS. | CADNETIX, DAISY, MENTOR GRAPHICS, OR VALID WORKSTATIONS, OR DEC VAX MAINFRAMES | INCLUDED AS PART OF VENDOR'S VISYS ASIC VERIFICATION PACKAGE. VISYS: $7950 |
| SILICON COMPILER SYSTEMS INC | AUTOMATIC TEST GENERATION (ATG) | SET OF SOFTWARE TOOLS THAT EVALUATES TEST-ABILITY OF BLOCKS, MODULES, AND CIRCUITS. GENERATES COMPRRESSED TEST VECTORS, AND DETERMINES GATE-LEVEL TEST COVERAGE OF THE VECTORS. | GENSIL DESIGN SYSTEM + APOLLO D3000, D4000, D057T, SUN 3 OR DEC VAX 8600 OR VAX2 | $39,500 PER USER |
| TERADYNE INC    | CIRCUIT BREAKER | ACHIEVES >95% COVERAGE OF LOGIC AND FUSE FAULTS. CREATES GRADED DYNAMIC TESTS FOR SLOW-PATH FAULTS ON COMBINATIONAL AND HIGHLY SEQUENTIAL PLDS FOR HIGH-VOLUME COMING INSPECTION AND IN-CIRCUIT BOARD TEST. | LASER VERSION 6 SIMULATION SYSTEM RUNNING ON ANY VAX/VMS SYSTEM | $8250 FOR VAX STATION 2000, $21,000 FOR VAX 8800 |
| TEST SYSTEMS STRATEGIES INC | TEST DEVELOPMENT SERIES (TDS) | TEAMS UP ASIC-DESIGN AND SIMULATION TOOLS TO PRODUCE SETS OF FUNCTIONAL VECTORS FOR SPECIFIC FORMATS. SUPPORTS >30 TESTER TYPES FROM EIGHT VENDORS AND 14 SIMULATION ENVIRONMENTS. | APOLOGAEGIS, SUN/UNIX, OR DEC VAX/VMS | $28,000 (DEPENDS ON SIMULATOR AND TESTER) |
Board test strategies: The bus is waiting

When you use ATE to test a loaded pcb board, the testing can fall into two major categories or into combinations of the two. The categories are in-circuit and functional. Until about five years ago, board-test systems were characterized by which of these two types of testing they performed. Now, so-called combinational systems are commonplace. The word “combinational” really has nothing to do with combinational logic, although a combinational tester clearly can test combinational logic. As used by the ATE manufacturers, the word refers to the systems’ combination of functional and in-circuit test capabilities.

With in-circuit testing, you basically test the components after they have been loaded onto the board to prove that the assembly process hasn’t damaged them. In-circuit testing is popular because, compared with functional testing, test programs are easy to write, and they inherently isolate defective components.

A tester can perform in-circuit tests on a component even though the outputs of other components are connected to some of its pins. The tester accomplishes this feat by “back driving” the pins of the other devices. Suppose you are testing a gate whose input is driven low by the output of a flip-flop. If for a very brief interval, you force enough current into the node, you can make the output of the flip-flop go high, without, you hope, causing permanent damage.

In-circuit test has problems

But in-circuit testing has problems, too. One problem is that military agencies are unwilling to accept the idea that back driving doesn’t introduce latent defects that cause components to fail prematurely, long after a board leaves the factory. A second problem relates to the effect of capacitive loading imposed by the tester on the circuit nodes of the board under test. A third problem results from the physical means used to access the component under test; the customary probing method uses bed-of-nails fixtures.

The trend toward surface-mount technology, with its high circuit densities, multiplies problems with bed-of-nails fixtureing. When component leads are on 0.1-in. centers, the spring-loaded pogo pins used in bed-of-nails fixtures can be relatively durable. But pogo pins for the 0.50-in. centers, which are common with surface mounting, are much less durable, and when you surface-mount components on both sides of a board, you need still more complex and finicky clam-shell fixtures. Furthermore, pressing on the leads of surface-mount components (which don’t pass through holes in a pc board) can cause bad solder joints to test good. Consequently, when laying out surface-mount boards that will be tested on in-circuit testers, printed-circuit designers now usually use probe points separate from the component leads and place them on 0.1-in. centers. Adding such probe points decreases component density to achieve testability.

In-circuit and SMT mix poorly

It isn’t hard to see that if every surface-mounted component requires probe points on 0.1-in. centers, the density advantage of surface mounting quickly evaporates. The fight to retain high densities is one factor that has helped to bring about a resurgence of interest in functional testing. Functional testing actually attempts to make a board perform as it would in its target system. In true functional testing, you access the board only through its I/O connections, although you can add I/O connections to improve observability or controllability.

As mentioned earlier, it is much harder to write functional test programs than in-circuit programs, and functional programs don’t inherently isolate faults. ATPG programs address both of these functional-testing problems. They can check a design for compliance with testability rules (for example, absence of asynchronous circuits, ability to break feedback loops by use of gates included for testability, and ability to substitute external clocks for internal ones that you can disable while testing). ATPG programs also perform “fault grading.”

Like so many other terms in the digital-test
TECHNOLOGY UPDATE

front of the boards—the same connectors used to access the boards during production test (see box, "Board test strategies: The bus is waiting"). You should be able to control the boards and observe their function at these connectors, and so run functional tests of all the boards without unplugging them.

A more advanced (but much more costly) approach builds each board's control and observation points onto a connector that mates with the system backplane. The system itself contains the equivalent of the portable tester. With such capabilities, the system can diagnose its own faults; an operator's only function is

lexicon, fault-grading seems to be a misnomer; a better term would be pattern grading. Fault grading introduces simulated faults into a circuit model. The objective is to determine which faults a test pattern can find. Not only does fault grading let you evaluate the usefulness of particular patterns, it helps you to isolate faults by indicating which patterns detect them. Even though nodes stuck at 1 and stuck at 0 aren't the only possible faults (for example, nodes can be shorted together by a solder bridge), many years of experience have shown that you can get the information you need if you limit the simulated conditions to "stuck-at" faults.

Cluster testing holds answers

A promising approach to board test is called cluster-functional testing. It literally represents a divide-and-conquer strategy. The time required to write a functional test program appears to increase geometrically with the complexity of the circuit. By dividing a board into clusters—small blocks of logic that perform identifiable functions that you can isolate and test functionally—you reduce the test-development time significantly and improve the odds that ATPG will work. Moreover, compared with in-circuit testing, you dramatically reduce the number of probe points you need.

But cluster functional testing still usually presupposes that you probe the board with a bed-of-nails fixture. Testers without such fixtures can be more compact and less expensive than those based on the bed-of-nails approach. Here's where the IEEE testability bus enters the picture. The proposed standard (currently called P1149) is one of several that aim to simplify control and observation of logic—from the device level to the system level. In a board-level design that complied with the testability-bus standard, you would, if possible, place all control and observation nodes on connectors; but if a board lacks room for connectors, the standard allows node access via probe points.

An IEEE committee has been working on P1149 for several years and is ready to submit a formal proposal for acceptance. Despite sweeping goals, the committee feels the proposed standard is workable. It defines a set of testability inputs and outputs that should appear on every complying device, board, and system. As the complexity of the item increases, the number of testability inputs and outputs can increase. Implementation of the standard on an IC usually requires some dedicated pins.

At the board level, the standard recommends using dedicated pins on existing connectors, but if backplane pins are limited, you can substitute special test connectors. If space along the front of a board is at a premium, the test connectors can be accessible only if you place the board on an extender, and if there is no room for test connectors at all, you can use probe points. Using backplane pins provides the greatest potential advantage; test connectors accessible without extending boards are next best, and so on.

Bye bye, bed of nails?

By eliminating bed-of-nails fixtureing in most cases, the testability bus appears to make it possible to use lower-cost ATE systems than those used for cluster functional testing. However, you can implement cluster-functional test's divide-and-conquer strategy and conform to the testability bus standard at the same time.

An intangible advantage of a testability-bus standard is that it can convert an arm-waving exercise into something quantifiable. In the absence of a standard, determining the difficulty of testing a proposed design can be an exercise in circumlocution. With a standard in place, you can replace hours of fruitless, time-consuming discussions with a simple statement of whether a product does or does not comply with the standard, and if it complies, what level of the standard it meets.
to perform the actual board replacement.

At the nuts-and-bolts level, a seemingly trivial problem—compatibility among some logic-simulator database formats and the input requirements of some ATPG programs—has placed roadblocks in the way of automatic test-vector generation. The quick solution is database-translation programs. With a little more effort, vendors can modify the simulators and ATPG programs to cure the incompatibilities. For example, they can make their software support relatively new industry standards such as EDIF (Electronic Design Interchange Format).

Another form of incompatibility can also cause problems: ATE systems can’t always run the test vectors generated by ATPG software. Some vendors are addressing this problem with software that compares generated vectors against a target tester’s resources and flags incompatibilities. You can also use such software to modify the vectors until they are compatible with the tester.

Other vendors are adamant that you shouldn’t start designing a product until you fully understand the resources of the ATE system that you will use to test it. At least in US companies, it is very unusual for designers to let tester capabilities constrain their designs, although the idea of doing so is totally consistent with DFT. However, it is quite clear that you will wind up with a usable vector set a lot sooner if you consider the tester’s capabilities before freezing your hardware design rather than waiting until afterward.

**Scannable components needed**

A real problem with acceptance of ATPG for board-level designs relates to component availability: To effectively implement the scannable designs required for compatibility with ATPG programs, you need scannable devices. Otherwise, you must create their equivalent from unwieldy numbers of SSI devices. Many ASICs now incorporate circuits that implement at least a boundary-scan approach. Scannable equivalents of the most-used MSI devices and μPs still aren’t widely available. (One vendor is represented in Table 1.)

Advocates of DFT categorize several of the above concerns as results of a one-sided view of the product-development process. A good example is the fear of lengthened product-development cycles. Today, the test-development phase of a complex-board project often consumes as much time and money as the traditional design phase. DFT advocates insist that once designers are familiar with the new discipline, the cost and duration of the design portion of a project that uses DFT will probably not exceed what you experience using traditional approaches, but the cost and duration of test development can decrease by 90% or more. Even if adopting DFT does increase the cost and duration of the design phase by 20%, a 90% decrease in test-development time and cost will cut the overall project cost and duration by 35%. In fact, ATPG vendors cite reduced time to market as the selling point that strikes a responsive chord with more potential customers than any other.

Surprisingly, aside from defense-electronics manufacturers required to comply with MIL testability standards, the few US companies that have adopted DFT for board- and system-level designs have done so in response to pressure not from manufacturing but from strong field-service profit centers. These companies have found that to service their products profitably, they must diagnose and repair board faults at depots located close to their customers. Such depots don’t process the volumes of boards that would justify purchases of million-dollar testers. Repair depots need low-cost testers that can quickly reconfigure to troubleshoot and test boards of many types. Unless the boards are designed for test, low-cost testers won’t do the job.

In the Pacific rim, the situation is
For more information...  
For more information on the testability products listed in Table 1, circle the appropriate numbers on the Information Retrieval Service card, or use EDN's Express Request service.

<table>
<thead>
<tr>
<th>Company</th>
<th>Address</th>
<th>Phone Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aida Corp</td>
<td>510 Old Ironside Dr</td>
<td>(408) 980-5200, Circle No 701</td>
</tr>
<tr>
<td>Anvil Software</td>
<td>Box 901</td>
<td>(617) 641-3861, Circle No 702</td>
</tr>
<tr>
<td>Gateway Design Automation Corp</td>
<td>6 Liberty Way Westford, MA 01886</td>
<td>(508) 692-9400, Circle No 763</td>
</tr>
<tr>
<td>General Inc</td>
<td>300 Baker Ave Concord, MA 01742</td>
<td>(617) 369-4400, Circle No 704</td>
</tr>
<tr>
<td>Hewlett-Packard Co</td>
<td>1820 Embarcadero Rd</td>
<td>(415) 898-5200, Circle No 705</td>
</tr>
<tr>
<td>HHB Systems Inc</td>
<td>100 Wyckoff Ave Mahwah, NJ 07431</td>
<td>(201) 848-8000, Circle No 706</td>
</tr>
<tr>
<td>Logical Solutions Technology Inc</td>
<td>310 W Hamilton Ave, Suite 101</td>
<td>(408) 274-3850, TLX 172607, Circle No 707</td>
</tr>
<tr>
<td>NCR Corp</td>
<td>2001 Danfield Ct Ft Collins, CO 80225</td>
<td>(303) 226-9500, Circle No 708</td>
</tr>
<tr>
<td>Silicon Compiler Systems Inc</td>
<td>2845 Hamilton Ave</td>
<td>(408) 371-9200, Circle No 709</td>
</tr>
<tr>
<td>Teradyne Inc</td>
<td>321 Harrison Ave Boston, MA 02118</td>
<td>(617) 492-2700, Circle No 710</td>
</tr>
<tr>
<td>Test Systems Strategies Inc</td>
<td>8265 SW Creekside Pl Beaverton, OR 97005</td>
<td>(503) 643-9221, TLX 992983, Circle No 711</td>
</tr>
</tbody>
</table>

different. Possibly because of their emphasis on serving geographically far-flung export markets, major Japanese equipment manufacturers, unable to purchase what they consider to be suitable ATPG software, have developed it in house. They are also moving aggressively to design their board-level products so that ATPG software can generate the necessary vector sets.

In the US, the prevalent view is that Asian companies lag behind their American counterparts in software-development expertise. ATPG software is complex; the Japanese companies' ability to develop it in house may indicate that Japanese software development is more advanced than Americans suspect. The Asian companies' decision to focus on ATPG appears to indicate a greater commitment to testability in Japan than in the US, and it provides strong evidence that Japanese hardware designers think of test as their problem, not as somebody else's.

If you think about the earlier discussion of the effect of using DFT on a product's availability for use, and if you realize that Asian companies are leading the world in adopting DFT, you come up with yet another reason why American manufacturers should heed the advice of companies urging use of DFT. The push for quality in the US within the past year or two has probably equalized the MTBF of electronic products of similar complexity made in the US and Asia. However, if the use of DFT gives the Asian products a clear edge in availability, US companies will again be uncompetitive.

References

Article Interest Quotient  
(Circle One)  
High 518 Medium 519 Low 520

EDN March 17, 1988