Turbo1500: Core-Based Design for Test and Diagnosis

Laung-Terng Wang, Ravi Apte, Shianling Wu, and Boryau Sheu
SynTest Technologies

Kuen-Jong Lee
National Cheng Kung University

Xiaoqing Wen
Kyushu Institute of Technology

Wen-Ben Jone and Jianghao Guo
University of Cincinnati

Wei-Shin Wang
Silicon Integrated Systems

Hao-Jan Chao, Jinsong Liu, Yanlong Niu, Yi-Chih Sung, Chi-Chun Wang, and Fangfang Li
SynTest Technologies

Editor's note:
Tool support is crucial in widespread adoption of a standard. This article describes a set of tools and associated flow for DFT insertion and test generation based on IEEE Std 1500.

—Erik Jan Marinissen, IMEC

IC TESTING BASED on the full-scan design methodology and its associated test methods, such as ATPG and BIST, is the most widely used test strategy.[1] However, the applicability of scan-based testing is being severely challenged by rapidly increasing development and test costs, largely influenced by design size and complexity as circuit sizes grow and feature sizes shrink. Currently, the most common approach to this problem is to adopt the SoC methodology that uses IP cores for chip design and then use the divide-and-conquer method by testing each core individually. Of course, such a test method is possible only when individual cores can be isolated.[2] Debug and failure analysis also become easier when using a core test method.

Because cores used in a SoC can be legacy, designed in house, or supplied by external vendors, integration and subsequent debug and diagnosis of diverse cores requires that these cores follow certain industry standards. One such standard, IEEE 1500 Standard for Embedded Core Testing, enables test and diagnosis of embedded cores and interconnects.[3]

This article describes Turbo1500, a core-based test-and-diagnosis integration and automation system for incorporating the IEEE 1500 standard.[4] We developed the Turbo1500 system to automatically wrap the cores with 1500-compliant wrapper cells and generate associated testbenches under the control of the chip-level IEEE Std 1149.1 (IEEE Standard Test Access Port and Boundary-Scan Architecture), and to facilitate core test and diagnosis on printed circuit boards (PCBs) whose chips are embedded with boundary scan and where cores are surrounded with 1500-compliant wrappers.

During wrapper synthesis, in addition to supporting 1500-compliant wrappers for providing serial access to boundary I/O signals through the wrapper serial port (WSP), Turbo1500 inserts an on-chip test access mechanism (TAM) controller at the chip level to give designers the flexibility to also access cores in parallel via the wrapper parallel port (WPP). During test, debug, and diagnosis, each core is embedded with a unique core-identifier (CID) bit, so multiple core tests can be performed in series or in parallel to meet various power consumption, test time, and other requirements.

To demonstrate how the Turbo1500 test automation suite of tools works in accordance with IEEE Std 1500, we built two hardware boards. The first board includes

0740-7475/09/$25.00 © 2009 IEEE Copublished by the IEEE CS and the IEEE CASS

two memory-BIST cores, a logic-BIST core, and two FPGAs for implementing scan compression in a small design. The other board is a peripheral component interconnect (PCI) interface card that plugs into a PC backplane. A 1500-compliant wrapper cell surrounds each core in the FPGA. To validate the implementation, we manually injected single and multiple stuck-at faults and memory-cell faults into the scan design and the logic-BIST or memory-BIST cores wherever applicable. Experimental results show that these faults were detected during core test and located during debug and diagnosis.

We also provide a case study on an industrial design that contains three scan cores with more than 5 million gates. We show how the three scan cores are synthesized with scan compression logic and wrapped with 1500-compliant wrapper cells. We also illustrate how staggered and one-hot test patterns are generated for cores with shared clocks so that these core test patterns can be used at each stage—manufacturing test, silicon debug, and fault diagnosis.

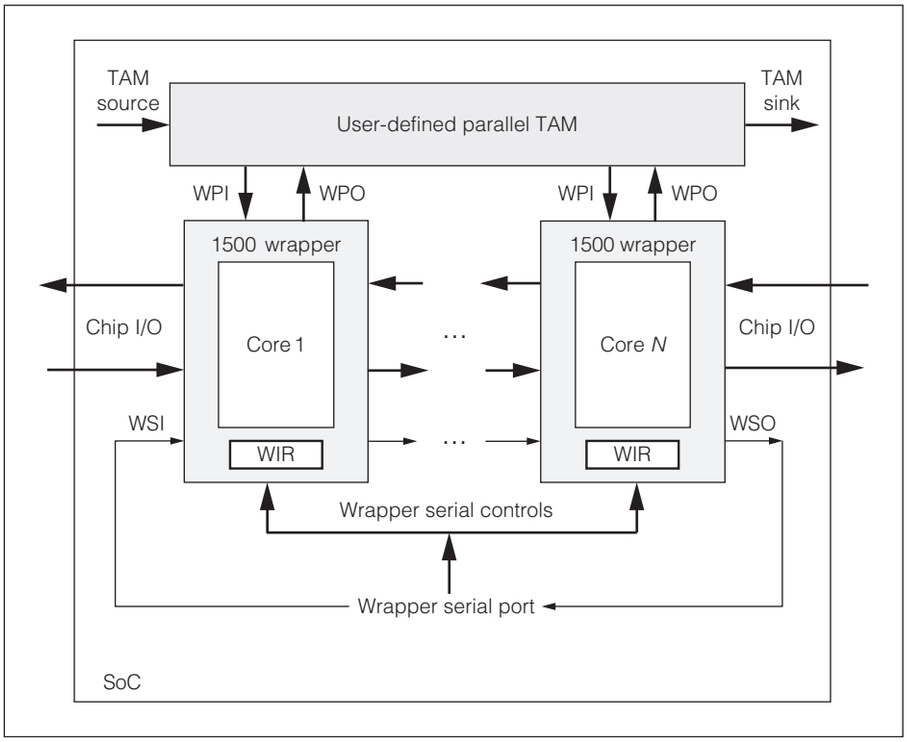


Figure 1. Overall SoC architecture in accordance with IEEE Std 1500. (TAM: test access mechanism; WIR: wrapper instruction register; WPI: wrapper parallel input; WPO: wrapper parallel output; WSI: wrapper serial input; WSO: wrapper serial output.)

IEEE Std 1500

IEEE Std 1500 effectively supports various TAMs for testing core-based designs within a SoC. These mechanisms constitute hardware architecture, and they leverage the IEEE 1450.6 Core Test Language (CTL), which is a subset of the IEEE 1450.6 standard, to facilitate communication between core designers and core integrators.[5] The primary structure is a wrapper surrounding each core, so that its SoC environment can invoke the wrapper to access or isolate the core. The purpose of the wrapper surrounding the core's I/O signals is to standardize the core's test interface. IEEE Std 1500 provides mechanisms only for core-based test. Any method can be used to generate the actual test patterns.

Figure 1 shows an overall SoC architecture with N cores, each wrapped with a 1500-compliant wrapper. The WSP is a set of wrapper I/O signals for serial operations. These signals consist of a wrapper serial input (WSI), a wrapper serial output (WSO), and several wrapper serial control (WSC) signals. Each wrapper has a wrapper instruction register (WIR) for storing the instruction to be executed in the corresponding core. The WIR controls operations in the wrapper, including accessing the wrapper boundary register (WBR), the wrapper bypass (WBY) register, and other user-defined function registers. The WBR consists of wrapper boundary cells (WBCs), which can be as simple as a single storage device (for observation only) or as complex as a cell with multiple storage devices on its shift path.

WSP support for the serial test mode is similar to the support provided in the boundary scan architecture, but without using a TAP controller. This means that the serial control signals defined in IEEE Std 1500 can be applied directly to the cores, thereby providing greater test flexibility. In addition to the serial test mode, IEEE Std 1500 also provides an optional parallel test mode with a user-defined parallel TAM. Each core can have its own wrapper parallel control (WPC), wrapper parallel input (WPI), and

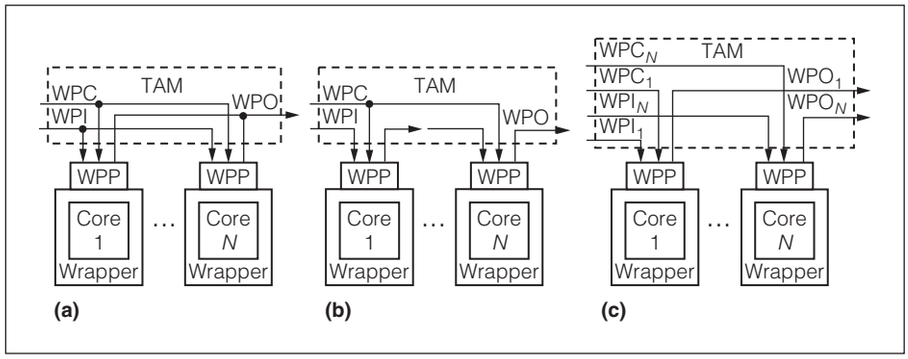


Figure 2. User-defined parallel TAM architectures: multiplexed (a), daisy-chained (b), direct access (c). (WPC: wrapper parallel control; WPP: wrapper parallel port).

wrapper parallel output (WPO) signals. A user-defined parallel TAM can transport test signals from the TAM source (either inside or outside the chip) to the cores through the WPC and the WPI and from the cores to the TAM sink through the WPO in a parallel manner, greatly reducing total test time.[6]

Turbo1500 core test automation

Turbo1500 provides two major core-test automation functions: 1500-compliant wrapper synthesis and testbench generation. For wrapper synthesis, Turbo1500 creates an IEEE Std 1500 test interface by surrounding each core with a 1500-compliant wrapper. The Turbo1500 wrapper synthesis program can

- create synthesizable Verilog RTL codes for the 1500-compliant wrappers,
- support mandatory and optional instructions defined in IEEE Std 1500, and
- add user-defined (or private) instructions that go beyond ad hoc core test to support hierarchical core test and diagnosis.

These standard instructions include `WS_INTEST_RING`, `WS_EXTEST`, `WS_PRELOAD`, `WS_BYPASS`, and `WS_CLAMP`. Optional instructions supported include `WP_PRELOAD`, `WP_EXTEST`, `WS_INTEST_SCAN`, `WP_INTEST_SCAN`, and `WP_INTEST_RING`. For testbench generation, Turbo1500 generates required testbenches and test programs to verify the correctness of the synthesized core-based test logic. Included are capabilities to

- generate Verilog testbenches that can verify the IEEE Std 1500 implementation,
- create STIL (Standard Test Interface Language) files based on the IEEE 1450-6 CTL for manufacturing test, and
- convert and output tester-specific test programs to test the chips.

The Turbo1500 system supports 1500-compliant wrappers by providing serial access to boundary I/O signals through the WSP. The external test pin count decreases because these WSC signals are generated directly at the chip level using a TAP controller, as defined in the IEEE 1149.1 standard for core test. Turbo1500 also automatically inserts a user-defined parallel TAM at the chip level, so that multiple architectures of the parallel TAM supported in IEEE Std 1500 become available for providing parallel access to both input and output signals through the WPP. Some of these architectures appear in Figure 2, including multiplexed access whereby the cores time-share the test control and data ports, daisy-chained access whereby the output of one core is connected to the input of the next core, and direct access to each core.

In addition to the capability of user-defined parallel TAM architectures, the Turbo1500 system also provides a capability called *test sessions*. A test session, involving multiple cores, enables testing the cores sequentially or in parallel. One challenge is how to deal with multiple clock domains in a core and the chip I/O pins (including control, data, and clock signals) among cores. If cores are to be tested sequentially (see Figure 4 in Wang et al.[4]), no constraints are imposed on the order of the scan clocks and the use of shared chip I/O pins to control the cores. For parallel testing (see Figure 5 in Wang et al.[4]), however, constraints on scan clock order and the use of shared chip I/O pins will affect the applicability of core test patterns to these interacting cores. For Turbo1500 use, we highly recommend that these shared data or control pins and core-level clocks reside inside the wrapper. If this isn't feasible, one option would be to merge the individually generated core test patterns that have the same capture clock order. Another would be to apply these shared clocks to all cores in a predetermined staggered order for capture before applying one-hot test patterns to each core. This latter option needs an n-bit (or $2n$-bit) shift register for n clocks in the design to indicate

whether the chosen clock is to be enabled or disabled for stuck-at (or delay) fault testing during each capture window.

In Figure 3, for example, we assume that two clocks (CK_1 and CK_2) control both scan cores and that a global-scan-enable signal, *GSE*, is used during the shift and capture operations for stuck-at testing. During ATPG, we first generate staggered (stuck-at) test patterns on the basis of a predetermined order imposed on the two clocks given in Figure 3a. The delays (d_1, d_2, and d_3) shown in the figure must be long enough to ensure the correct shift and capture operations. Then, to increase the core's overall fault coverage, we generate one-hot (stuck-at) test patterns for each core on the basis of the clock sequence given in Figure 3b or 3c. As long as the staggered clock order is maintained across all shared cores, parallel testing will be possible. If only one scan core is to be tested or debugged in series, the clock order becomes irrelevant.

This test session capability thus gives designers the flexibility to test cores with constraints imposed by test time, power consumption limits, debug and diagnosis requirements, and so on. In other words, these cores can be tested selectively in serial-parallel or parallel-serial fashion, depending on the TAM controller's chosen architecture. This strategy will reduce, if not solve, the SoC test time problem, as discussed elsewhere.[7]

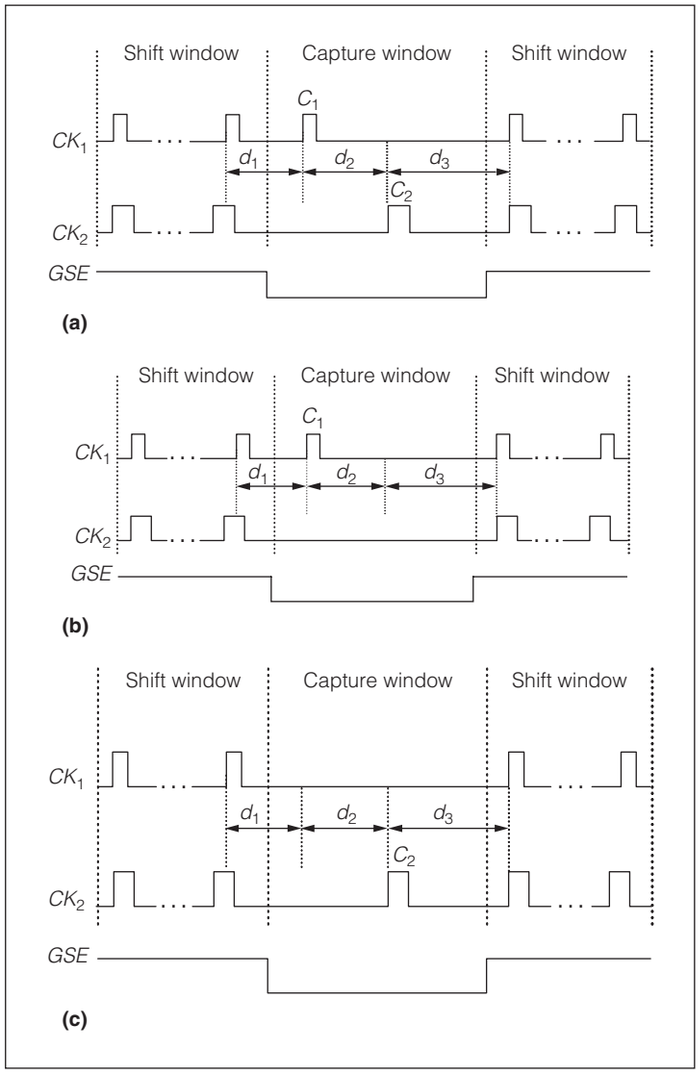


Figure 3. Staggered clock order used in the capture operation: staggered clocks with CK_1 followed by CK_2 when the shift register is set to {11} (a), one-hot CK_1 clock when the shift register is set to {10} (b), one-hot CK_2 clock when the shift register is set to {01} (c). (*GSE:* global scan enable.)

Hierarchical core test and diagnosis

Testability implementation can be inherently applied to an entire SoC design at one time, or it can be designed block by block, followed by integration at the chip-top level. However, the one-shot approach has proved ineffective. Engineers can also use Turbo1500 to test and diagnose scan-compression, logic-BIST, and memory-BIST cores when these cores are designed in a hierarchical (block by block) manner. Wang et al. provide additional information.[4]

Hierarchical scan compression

The scan compression technology is built upon VirtualScan.[8] The VirtualScan technology, intended mainly for scan test cost reduction, is based on the idea of splitting n scan chains into $n \times s$ shorter scan chains, for a split ratio of s, where s is typically 10 or greater. The VirtualScan architecture uses a simple combinational-logic-based broadcaster inserted at the inputs of the short scan chains for broadcasting input stimuli from n external scan input ports to $n \times s$ internal scan chain inputs. The architecture uses a simple combinational-logic-based compactor inserted at the outputs of the short scan chains for compacting test responses from $n \times s$ internal scan chain outputs to n external scan output ports. Typically, scan compression reduces test data volume and test application time by 10× to 100×, thereby drastically reducing scan test cost.[1,9] A proposed solution for further reducing test data volume and test application time exploits hierarchy in core-based designs by first applying Turbo1500 to synthesize a 1500-compliant

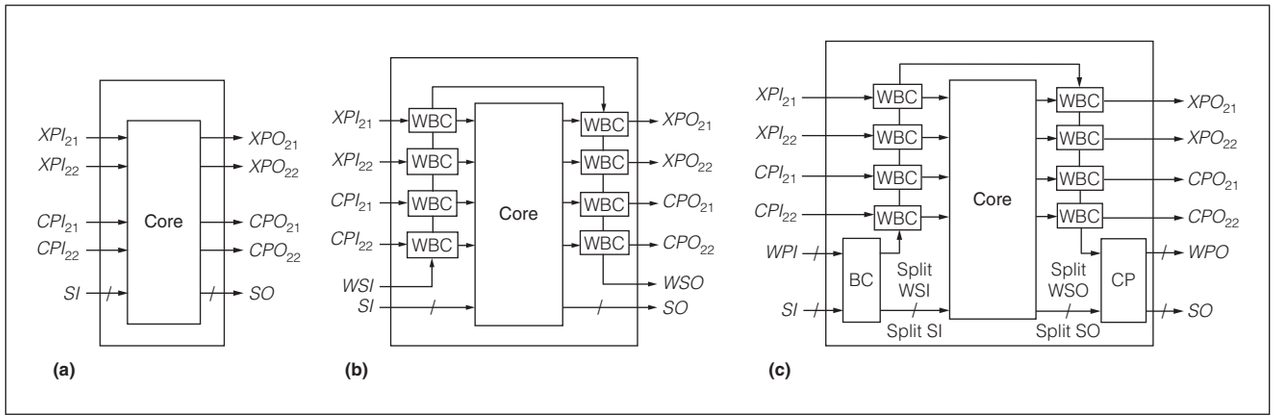


Figure 4. Core wrapping: original core (a), original core with wrapper (b), wrapped scan compression core (c). (BC: broadcaster; CP: compactor; CPI: core primary input; CPO: core primary output; SI: scan input; SO: scan output; WBC: wrapper boundary cell; XPI: external primary input; XPO: external primary output.)

wrapper around each scan core that is accessible through an integrated TAP and TAM controller, and then applying the scan compression methodology to each such core. Each 1500-compliant scan compression core can then be tested from the chip top. As expected, the core becomes a testable core that can be reused in any SoC that supports IEEE Std 1500. Because of the varieties of hierarchy inherent in a design, Turbo1500 currently handles at most two levels of hierarchy, meaning that only one level of scan compression cores can be embedded within a chip-top module. There is no nesting beyond the first level.

For example, Figure 4a is the original core in which external primary I/O (XPI/XPO) pins are signals from and to other cores within the SoC, and core primary I/O (CPI/CPO) pins are directly accessible from the chip-top primary I/O (PI/PO) pins. Scan input (SI) and scan output (SO) pins are connected to the scan chains in the core.

Figure 4b shows a wrapped core with wrapper boundary cells (WBCs) surrounding the boundary XPI/XPO and CPI/CPO pins that are stitched together to form a wrapper serial chain between the wrapper serial input (WSI) and the wrapper serial output (WSO) pins. Alternatively, all CPI/CPO pins can be directly accessible externally. Wrapper serial control (WSC) signals (not shown in the figure) control the core's operation.

In Figure 4c, scan compression has been implemented on the wrapped core. On the basis of a chosen split ratio, the existing scan input and scan output chains are split into shorter chains. A broadcaster (BC) and a compactor (CP) inserted at the shorter chain inputs and outputs decompress input stimuli and compact output responses, respectively, thereby reducing the number of boundary I/O signals surrounding the core. The existing WSI/WSO chain then splits into several WPI/WPO chains, each not to exceed the longest scan input or scan output chain length. Although this example shows wrapper insertion before compression logic insertion, the reverse also works.

One major objective of Turbo1500 applications is to facilitate on-board, in-system, and in-field debug and diagnosis, so Turbo1500 further allows generating these core-level shift and capture clocks from the test clock (*TCK*) through boundary scan control. Given a core, Turbo1500 inserts an on-chip clock controller (OCC) into each clock domain of the core. The OCC controls and generates the needed shift clock pulses and (at-speed) capture clock pulses during the shift and capture operations.

Hierarchical logic and memory BIST

Given a core composed of logic gates, SRAM, or DRAM, Turbo1500 automates the insertion of 1500-compliant wrappers surrounding the BIST cores that have been synthesized with their respective logic-BIST or memory-BIST controllers through boundary scan control. Each internal core has an isolation wrapper. A core identifier (CID) bit is added to each core, and these bits form a shift register, called a CID register, such that during test, debug, and diagnosis, these bits can be programmed on the fly to test the respective cores in series or in parallel.

To debug or diagnose memory-BIST and logic-BIST cores, Turbo1500 first sets the CID bits of the BIST cores to be diagnosed to 1. These cores can then be processed in parallel. Engineers can further choose to have the CID register include additional bits for each logic-BIST or memory-BIST core so as to increase diagnostic resolution for multiple errors that may arise from different faults, including memory faults, data retention faults, stuck-type faults, or timing faults.

Experimental results

We validated Turbo1500's effectiveness in two case studies, one involving demonstration boards and the other industrial applications.

Case study 1

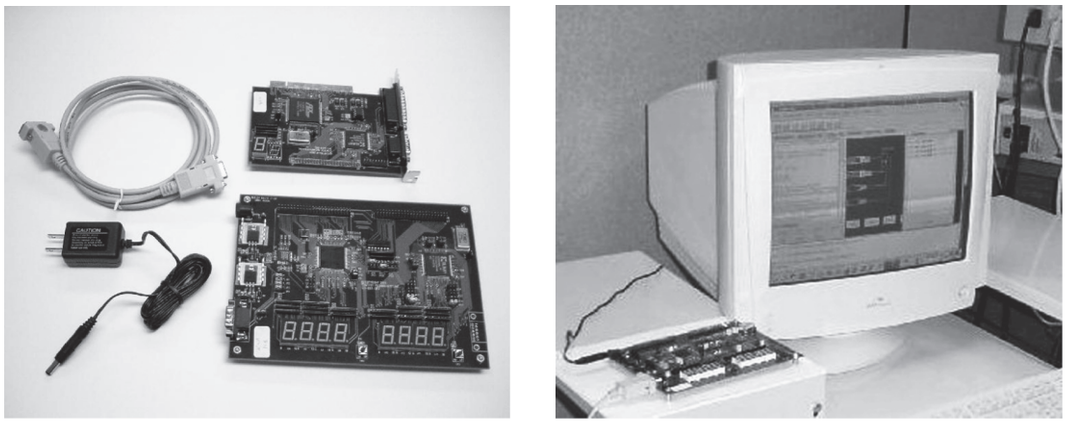
To study the effectiveness of Turbo1500's implementation of the 1500-compliant wrappers in conjunction with different user-defined TAM architectures and demonstrate that IEEE Std 1500 can actually facilitate SoC debug or diagnosis, we implemented a 1500-compliant SoC design with two FPGAs that included a core with scan compression, two memory-BIST cores, and a logic-BIST core. Two PCBs, shown in Figure 5a, were also built for this purpose. One demo board contains the two core-based FPGAs and an LED display to indicate whether the test passes or fails. The other board is a PCI interface card that plugs into the backplane of a PC running the Linux operating system. We developed TurboDiagnosis, a set of core diagnosis tools with a graphical user interface, for debug and failure analysis of each individual core on the two FPGAs (see Figure 5b). Various failure mode diagnosis methods can localize the root cause of a failure or a test escape.[10]

Because both FPGAs are implemented with boundary scan, we performed integrity testing for compliance with the IEEE 1149.1 standard. Then, interconnect testing determined whether the wires interconnecting the two FPGAs were broken. Once the boundary scan and interconnect tests were passed, core diagnosis could begin.

First, we programmed the unique CID bit embedded in each core of the two FPGAs, which allowed us to select an individual core or a group of cores taking part in debug or diagnosis in parallel. Then, by coordinating the TAP and TAM controllers, we were able to pinpoint the cores' manually injected failures down to the gate level (in the scan compression or logic-BIST core case) or to the bit level (in the memory-BIST core case). The TurboDiagnosis software and Turbo1500 have been applied to several industrial designs in which ATE serves in core-based debug and diagnosis.

Case study 2

We embedded IEEE Std 1500 circuitry in an industrial design for test, debug, and diagnosis. The chip-top design contained more than 5 million logic



(a) (b)

Figure 5. Turbo1500 demonstration system: hardware boards to demonstrate the test and diagnosis of 1500-wrapped cores (a); graphical user interface to locate faults injected into the four cores implemented on the two FPGAs (b).

gates in three scan cores (A, B, and C) and more than 100 instances of embedded memories. Because of space limitations, we illustrate only how we performed core test automation on the three scan cores for testing stuck-at faults within these three cores.

The first step is wrapper synthesis (see Figure 4c). The coresyn program synthesizes a 1500-compliant wrapper around each scan core. In each wrapper, WBCs are inserted into all XPI/XPO and CPI/CPO signals. To reduce the number of test I/O pins, we inserted an integrated TAP and TAM controller (not shown in the figure) at the chip-top level for generating all necessary wrapper parallel control (WPC) signals, and thereby to control the operation of the three cores simultaneously. Then, on the basis of the scan compression technology proposed by Wang et al.,[8] the vscansyn program inserts a broadcaster and a compactor—both purely combinational logic based—at the scan input and scan output ports of each scan core. To fully utilize core test patterns, from test to debug to diagnosis, we ensure that all WPI/WPO and scan input and scan output signals are directly accessible through primary I/O ports. The TAM supports the direct-access architecture, as shown in Figure 2c, to provide parallel access to these primary I/O signals.

The next step is testbench generation. First, the clockgroup program analyzes and groups those clock domains at the chip top that do not interact with one another, primarily to reduce ATPG runtime and total pattern count. As Table 1 shows, although the three cores have a total of 13 clock domains, the number of clock groups at the chip top after clock grouping is only six. Because of overhead and timing concerns, the *GSE* signal and all clocks must be shared in the three scan cores, so we opted to generate staggered test patterns in a predetermined clock order on all cores, followed by one-hot test patterns to improve each core's fault coverage. Then, to obtain the highest fault coverage possible (when $GSE = 0$), the asicgen program performs staggered ATPG and one-hot ATPG on each individual scan compression core. The switch from staggered ATPG to one-hot ATPG is based on a decision criterion upon reaching a time limit (say, 2 hours). Finally, the tpout and testerout programs convert the generated (staggered and one-hot) test patterns into Verilog testbenches for verification, and into tester-specific test programs for test and diagnosis. Flush tests are also generated (when $GSE = 1$) and added to the Verilog testbenches to verify the correctness of the shift operation of all scan and wrapper chains in the three cores.

To signal which clock groups are active during each capture operation, we embed a 13-bit CID register for controlling the 13 clock domains in the three cores for stuck-at fault testing. For example, if the 13-bit CID register contains 13 1s, or {11, 1111111, 1111}, then a staggered test pattern will apply to all three cores according to the

Table 1. Design statistics and results for a chip top containing three cores.

Characteristic	Core A	Core B	Core C	Chip top
No. of primitives	700,363	893,130	3,618,518	5,212,011
No. of flip-flops	36,015	51,471	162,868	250,354
No. of clock domains	2	7	4	13
No. of clock groups, *G*	2	4	4	6
No. of primitives after scan compression and before wrapper synthesis	700,867	893,634	3,620,002	5,214,505
No. of primitives after wrapper synthesis	707,275	909,990	3,627,898	5,245,163
Split ratio	10	10	10	10
No. of internal scan chains	100	100	300	500
No. of CPI/CPO ports	8/0	16/0	11/0	35/0
No. of XPI/XPO ports	772/1,023	2,860/1,944	1,080/1,164	4,712/4,131
No. of WPI/WPO ports	5/5	10/10	5/5	20/20
No. of scan input and scan output ports	10/10	10/10	30/30	50/50
Maximum scan-chain length, *L*	361	515	543	543
Total no. of test patterns, *P*	1,489	2,484	3,380	3,380
No. of test cycles	548,313	1,301,616	1,866,303	1,873,063

clock sequence supplied at the six grouped clock inputs, where any two clock domains that do not interact with each other are controlled by the same grouped clock. If the CID register is set to {01, 0000000, 1111}, then a one-hot test pattern and a staggered test pattern will apply to cores A and C, respectively, and core B is not targeted for test, debug, or diagnosis.

The results appear in Table 1, which indicates that the total area overhead (without considering bus routing) associated with wrapper synthesis on the three cores (number of primitives after wrapper synthesis minus number of primitives before wrapper synthesis) is less than 0.6%, where a primitive represents a flip-flop, a multiplexer, or an *n*-input combinational gate, not a two-input equivalent gate. During scan compression, we picked a split ratio of 10 for each core, to limit the total number of scan input and scan output ports to 100. There are five WPI and five WPO ports for core A because all 1,803 (8 + 0 + 772 + 1,023) CPI/CPO and XPI/XPO ports are stitched together to form WPPs with a maximum scan chain length not to exceed 361 (36,015/100) scan flip-flops. Because we constrained ATPG to generate staggered patterns on the basis of a predetermined clock order, we can apply parallel testing using core test patterns during manufacturing test, and we can apply serial testing on selected cores during silicon debug and fault diagnosis whenever needed.

The design will require approximately 1.88 million test cycles during manufacturing test and approximately 3.72 million test cycles (the sum of the test cycles of the three cores) for silicon debug and fault diagnosis. Let's assume there are P test patterns, L is the maximum scan chain length, and there are G clock groups in a core. Testing the core will require $P(L + 5 + G) + L$ test cycles (excluding the initialization cycles). The five *TCK* cycles are required to perform the *UpdateDR* and *CaptureDR* tasks for each pattern, as described by Cheng et al.[11]

The demonstration boards and industrial applications proved that using IEEE Std 1500—with certain enhancements—is a viable solution for incorporating testability in core-based SoC designs in a hierarchical (block by block) manner to facilitate on-board, in-system, or in-field core diagnosis. IEEE Std 1500 does not suggest using the same four mandatory pins as the IEEE 1149.1 standard for core access through boundary scan. Integrating Turbo1500 with the IEEE 1149.1 standard avoided the use of additional pins. An on-chip TAM controller enables serial and parallel core access. Also, the use of BIST cores and other embedded instruments is becoming prevalent for current designs. These instruments can increase the device's fault coverage, reduce test and debug time, and correlate the system and ATE environments. However, the lack of a standard interface (through the IEEE 1149.1 TAP) to these instruments makes automation virtually impossible. Consequently, investigation and planning are under way for enhancing Turbo1500 to follow the IEEE P1687 (IJTAG) proposed standard, which allows access to the embedded test and debug instruments through the IEEE 1149.1 TAP.[12] ■

Acknowledgments

We are grateful to the anonymous referees for pointing out unclear descriptions in the article and giving constructive suggestions. This work was supported in part by the National Science Foundation under grant CCF-0541103 and in part by the National Science Council of Taiwan under contract NSC096-2917-I-006-120.

■ References

1. L.-T. Wang, C.-W. Wu, and X. Wen, eds., *VLSI Test Principles and Architectures: Design for Testability,* Morgan Kaufmann, 2006.
2. Y. Zorian and A. Yessayan, "IEEE 1500 Utilization in SoC Test and Design," *Proc. IEEE Int'l Test Conf.* (ITC 05), IEEE CS Press, 2005, pp. 1203-1212.
3. *IEEE Std 1500-2005, IEEE Standard for Embedded Core Test,* IEEE, 2005.
4. L.-T. Wang et al., "Turbo1500: Toward Core-Based Design for Test and Diagnosis Using the IEEE 1500 Standard," *Proc. Int'l Test Conf.* (ITC 08), IEEE CS Press, 2008, paper 29.3.
5. *IEEE Std 1450.6-2001, Core Test Language,* IEEE, 2001.
6. E. Marinissen et al., "On IEEE P1500's Standard for Embedded Core Test," *J. Electronic Testing: Theory and Applications,* vol. 18, nos. 4-5, Aug. 2002, Springer Netherlands, pp. 365-383.
7. C. Liu, K. Chakrabarty, and W.-B. Jone, "System/Network-on-Chip Test Architectures," *System-on-Chip Test Architectures: Nanometer Design for Testability,*

L.-T. Wang, C.E. Stroud, and N.A. Touba, eds., Morgan Kaufmann, 2007, pp. 171-224.

8. L.-T. Wang et al., "VirtualScan: Test Compression Technology Using Combinational Logic and One-Pass ATPG," *IEEE Design & Test,* vol. 25, no. 2, Mar./Apr. 2008, pp. 122-130.
9. R. Kapur, S. Mitra, and T.W. Williams, "Historical Perspective on Scan Compression," *IEEE Design & Test,* vol. 25, no. 2, Mar./Apr. 2008, pp. 114-120.
10. T. M. Mak and S. Venkataraman, "Design for Debug and Diagnosis," *System-on-Chip Test Architectures: Nanometer Design for Testability,* L.-T. Wang, C.E. Stroud, and N.A. Touba, eds., Morgan Kaufmann, 2007, pp. 463-504.
11. K.-L. Cheng et al., "An SoC Test Integration Platform and Its Industrial Realization," *Proc. Int'l Test Conf.* (ITC 04), IEEE CS Press, 2004, pp. 1213-1222.
12. *IEEE P1687-2007 Proposal, IEEE Internal Boundary-Scan Proposal for Embedded Test and Debug,* IEEE, 2007.

Laung-Terng Wang is founder and CEO of SynTest Technologies. His research interests include DFT, ATPG, logic and memory BIST, test compression, and fault diagnosis. He has a PhD in electrical engineering from Stanford University. He is an IEEE Fellow.

Ravi Apte is senior vice president of marketing at SynTest Technologies. His research interests include DFT and simulation. He has a PhD in electrical engineering from Southern Methodist University, Dallas.

Shianling Wu is vice president of engineering at SynTest Technologies. Her research interests include all aspects of DFT, from design and manufacturing to field applications. She has an MS in computer science from Columbia University.

Boryau Sheu is director of R&D at SynTest Technologies. His research interests include at-speed ATPG, logic BIST, and test strategy for SoC design. He has an MS in electrical engineering from Washington University, St. Louis.

Kuen-Jong Lee is a professor in the Department of Electrical Engineering and director of the SoC Research Center at National Cheng Kung University, Taiwan. His research interests include VLSI testing and testable design, electronic system-level design, and CAD. He has a PhD in electrical engineering from the University of Southern California.

Xiaoqing Wen is a professor and department chair in computer science and electronics at Kyushu Institute of Technology, Japan. His research interests include VLSI test and fault diagnosis. He has a PhD in applied physics from Osaka University, Japan.

Wen-Ben Jone is an associate professor in the Department of Electrical and Computer Engineering at the University of Cincinnati. His research interests include VLSI testing and low-power circuit design. He has a PhD in computer engineering from Case Western Reserve University.

Jianghao Guo is a graduate student in the Department of Electrical and Computer Engineering at the University of Cincinnati. His research interests include VLSI design and DFT. He has an MS in electrical engineering from Shanghai Jiaotong University, China.

Wei-Shin Wang is director of engineering at Silicon Integrated Systems. His research interests include design methodology, physical design, package and substrate design, and DFT for SoC design. He has an MS in electrical engineering from Santa Clara University.

Hao-Jan Chao is an R&D manager at SynTest Technologies, Taiwan. His research interests include logic BIST and novel DFT technology development. He has an MS in electrical engineering from National Central University, Taiwan.

Jinsong Liu is an applications engineer at SynTest Technologies, Shanghai. His technical interests focus on DFT and BIST applications, including the latest SoC technology. He has a BS in electrical engineering from Shanghai Jiaotong University, China.

Yanlong Niu is an R&D engineer at SynTest Technologies, Shanghai. His research interests include boundary scan, and logic and memory BIST. He has an MS in electrical engineering from Shanghai Jiaotong University, China.

Yi-Chih Sung is an applications engineering manager at SynTest Technologies, Taiwan. His research

interests include boundary scan, test compression, and logic BIST. He has a BS in computer science from Chung-Hua University, Taiwan.

Chi-Chun Wang is an applications engineer at SynTest Technologies, Taiwan. His research interests include scan ATPG, logic and memory BIST, and boundary scan. He has an MS in computer science from Leader University, Taiwan.

Fangfang Li is the regional manager at SynTest Technologies, Shanghai. Her research interests include DFT, BIST, and SoC design. She has an MS in automation from Shanghai Jiaotong University, China.

■ Direct questions and comments about this article to Laung-Terng Wang, SynTest Technologies, 505 S. Pastoria Ave., Suite 101, Sunnyvale, CA 94086; wang@syntest.com.