End-to-End Testing for Boards and Systems Using Boundary Scan

Robert W. Barr and Chen-Huan Chiang
Lucent Technologies
Engineering Research Center
Princeton, NJ 08542-0900

Edward L. Wallace
Lucent Technologies
Wireless Network Systems
Whippany, NJ 07981-0903

Abstract

ICs with IEEE 1149.1 Boundary Scan (BS) Architecture (a.k.a. JTAG) have been widely used in board level design to increase the testability. An end-to-end test methodology that utilizes BS architecture for testing boards and systems throughout the product life cycle is proposed. The proposed test methodology includes a programmable dynamic BS test architecture and a series of test modules that take advantage of the test architecture for complete fault coverage. Proposed design-for-testability (DFT) techniques guarantee the co-existence of BS testing with other system functions, such as in-system programming (ISP) and DSP JTAG emulation. At board level, programmable dynamic scan chains are used in a divide-and-conquer fashion to increase the flexibility in the development phase (or design verification testing (DVT)). Besides, since the DFT techniques are programmable, they can be used as design-for-diagnosis (DFD) to increase diagnosis resolution during DVT. Address Scan Port (ASP) chips are used to enable multi-drop test bus architecture for backplane testing as well as system embedded testing. Other advanced techniques, such as analog subsystem testing and board-level built-in self-test, as well as how to re-use BS architecture in in-circuit testing (ICT) and manufacture testing are also parts of the proposed methodology that takes advantage of BS architecture to provide full scale testing for systems.

We will illustrate the test methodology by its application to the design of a base station for wireless systems.

1 Introduction

ICs with IEEE 1149.1 Boundary Scan (BS) Architecture (a.k.a JTAG) [4] have been widely used in board level design to increase the testability. There have been lots of prior works in various applications using BS architectures [9, 10]. In this paper, an end-to-end test methodology that utilizes BS architecture for board level and system level testing will be presented in this paper. The proposed test methodology includes design-for-testability (DFT) test architectures and a series of test modules that take advantage of the test architectures for complete fault coverage. It is *end-to-end* since the test modules and the DFT test architectures are re-used *throughout the product life cycle* such as the development phase, the manufacture phase and field operation.

The DFT test architectures consist of a programmable dynamic BS test architecture at board level and a multi-drop test bus architecture at system level. They guarantee the co-existence of BS testing with other system functions, such as in-system programming (ISP) and DSP JTAG emulation.

At board level, the programmable dynamic scan chains are used in a divide-and-conquer fashion to increase the flexibility in the development phase, where we refer the testing to as design verification testing (DVT). Since the DFT techniques are programmable, they can also be used as design-for-diagnosis (DFD) to increase diagnosis resolution during DVT. It has been proven that using BS architecture for DVT is one of its real value since no other test access methods have yet developed in the DVT phase. To guarantee the normal and non-invasive operation while in the functional mode, a set of DFT rules are derived for the proposed test architecture. For system level testing, address scan port (ASP) devices [5] are used to construct the multi-drop test bus architecture for back-

0-7803-6546-1/00 $10.00

plane testing and system embedded testing.

The series of test modules in the proposed test methodology provides complete coverage for DVT at board level. It includes BS chain integrity test, device test, interconnect test, and cluster test. A novice backplane testing algorithm, which can be applied in a dynamic system configuration environment, automatically generates tests for backplane interconnect testing. All these test modules can eventually be executed in an embedded test environment. This helps the re-use of BS architecture in field testing.

The propose test methodology also addresses how to re-use BS architecture for in-circuit testing (ICT) and manufacture testing. Tools are created to take advantage of BS architecture to provide full scale testing for systems. There are other advanced techniques, such as analog subsystem testing and Built-In Self-Test (BIST) for interconnect testing.

The end-to-end test methodology can be divided into board level, system level, and other advanced test methodologies depending on the need for fault coverage. In the following, the proposed board level test methodology will be presented first and followed by system level test methodology. Other advanced test techniques will be briefly introduced. All the proposed test methodologies will be illustrated by their application to the design of a base station for wireless systems. At the end, the impact of adopting the proposed test methodology in developing the wireless system will be assessed to conclude this paper.

2 Board Level Test Methodology

The proposed board level test methodology provides complete coverage for DVT by applying a series of tests which includes BS chain integrity test, device test, interconnect test, and cluster test. It guarantees normal and non-invasive operation while in the functional mode. In addition, it supports ISP and flash programming, allows code verification via processor JTAG emulation tools, and has high diagnosis resolution. The guarantee of normal function operation and the support for the abovementioned capabilities are referred to as *essential DFT objectives* in this paper. To achieve the essential DFT objectives, the methodology consists of a test architecture using programmable dynamic BS chain and a set of DFT rules to ensure the operation of the test architecture for complete fault coverage.

2.1 DVT via BS architecture

In this section, the series of tests performed during DVT via BS architecture will be briefly described in the order of the application in the proposed test methodology.

Scan Chain Integrity Test: The integrity of the BS chain, including the four required (TCK, TMS, TDI, and TDO) and one optional ($\overline{\text{TRST}}$) test interface, is verified in this test module. Faults such as shorts and opens on the test interface, and improper placement of the BS devices can be captured by scan chain integrity test. Existing commercial tools are used for scan chain integrity test development.

Device Test: On the board under test (BUT), if a BS device is equipped with BIST or if the test vectors for the device's INTEST are available, BS architecture can be used to either activate the BIST circuitry or transport the INTEST vectors to test the BS device at any phase of the product life cycle. If the test vectors for INTEST is not available, depending on the customer's requirements, a software tool based on automatic pattern test generation (ATPG) techniques [1] is developed to generate compact test vectors to be applied to the device via BS.

Interconnect Test: This test module tests shorts, opens, and stuck-at faults at device I/O pins as well as the traces in the BUT. Test vectors developed in this module not only have complete fault coverage of the interconnect faults, but also provide high diagnosis resolution. For BS interconnects, existing commercial tools are used to generate vectors.

Cluster Test: Cluster devices are the glue logic devices without BS architecture. If they are surrounded by BS devices, two types of cluster tests are generated. One is cluster interconnect test and the other is cluster device test. Cluster interconnect test can detect interconnect faults between cluster devices and BS devices, such as SRAM interconnect test for address and data lines. Cluster device test performs functional test for cluster devices. If the cluster device is a flash memory, such a test module can

be modified to perform flash programming during DVT when BS architecture is the only access method of the BUT.

We have developed a cluster tool that can *automatically generates* test vectors for cluster interconnect test. Unlike most of existing tools rely on the availability of functional tests for cluster testing, our tool generates a set of complete and compact test vectors targeting specifically on interconnect faults.

Note that, once the above test modules are developed for DVT, all of them *can be used repeatedly throughout the product life cycle*. Tools are developed for translating the BS test modules throughout the cycle. Most importantly, the diagnosis information developed for all the test modules are preserved during the translation as much as possible.

2.2 Programmable Dynamic BS Chain

To achieve the abovementioned essential DFT objectives, a test architecture using programmable dynamic BS chain is proposed.

In a divide-and-conquer fashion, BS devices are first partitioned according to their characteristics. For example, in the Controller Unit shown in Figure 1, we divide BS devices into five partitions — PLDs, FPGAs, buffers, microprocessor, and DSPs. Each partition is then placed into the scan chain with *DFT logics* between partitions to establish a programmable dynamic scan chain. The DFT logics between BS partitions can not only disable and enable either the TCK or TMS signal of each of the BS partitions, but also ensure all the essential DFT objectives. The DFT logics can be implemented with very few glue logic devices. These BS enable signals must be accessible by a BS device in the partition at a previous location in the scan chain. Therefore, such a BS test architecture can be programmed into various configuration containing different BS partitions. It is a dynamic BS chain which offers flexibility during DVT and high diagnosis resolution since each partition can be independently isolated from the others (except for the partition which programs BS enable signals), or incrementally added into the scan chain using the programmable DFT logics between BS partitions.

The placement of BS partitions in the scan chain should be carefully arranged. To achieve the programmability, PLD devices are usually placed in the beginning of the scan chain to provide BS enable signals controlling the following partitions since there always exist non-connected pins of PLDs to be used for BS enable signals. Furthermore, we can not use available pins of FPGA devices to implement BS enable signals since most PLD devices have dedicated I/O pins for JTAG signals, while the JTAG I/O pins of most FPGA devices are no longer available once they are programmed.

For certain BS devices such as microprocessor and DSPs, whose architectures are more complicated, are advised to be placed in the end of the scan chain.

As shown in Figure 1, we have implemented the proposed test architecture using an incremental scan chain in the Controller Unit of the wireless base station to reduce the DFT logics between BS partitions. According to the configuration table in Table 1, BS partitions can be incrementally added into the BS test architecture to form three different configurations: (i) Chain 1, consisting of only the PLD; (ii) Chain 2, consisting of PLD, FPGAs, and buffers; and (iii) Chain 3 with all the BS devices.

Additional Flexibility and Portability: A special test fixture is also developed to enhance the programmable dynamic configuration of the BS test architecture as well as to aid the development of LabView test interface. This test fixture can be easily inserted into the beginning of the BS chain using one or two cables, each of which carries five JTAG signals. By programming the dynamic DFT logics on the fixture via BS architecture, we can configure the test architecture for more combinations of the BS partitions than the DFT logics on the BUT can do. Such a design helps increase the flexibility of DFT and the diagnosis resolution without putting excessive DFT logics on the BUT. In addition, using this fixture, all the test modules developed can be easily applied via LabView interface to increase their portabilities and save time of operating the application of the proposed tests.

DFT for DFD: The proposed programmable dynamic BS test architecture helped us tremendously when operated with diagnosis software for system diagnosis. Since each BS partition can be tested and diagnosed independently, any fault and failure could be identified with ease and accuracy. Hence, such DFT techniques can be re-used for diagnosis. For example, we were able to take advantage of this test architecture to isolate the FPGAs on the

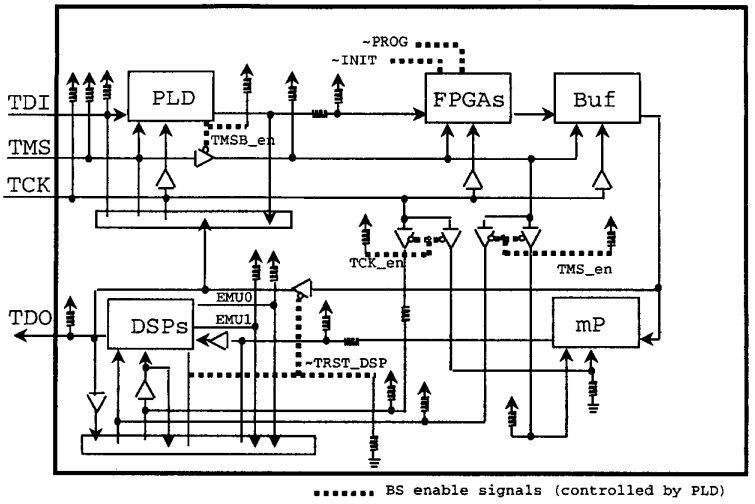

Figure 1: DFT for wireless Controller Unit.

Table 1: Wireless Controller Unit BS Chain Configuration Table.

	TMSB_en	*TCK_en*	*TMS_en*	$\overline{TRST\_DSP}$
	default=1	*default=1*	*default=1*	*default=0*
Normal mode	default	default	default	default
ISP on PLD	default	default	default	default
μ-p emulation	default	default	default	default
DSP emulation	default	default	default	default
DVT Chain 1 (PLD)	default	default	default	default
DVT Chain 2 (PLD, FPGA, Buffer)	0	default	default	default
DVT Chain 3 (all)	0	0	0	1
Embedded system test Chain 2	0	default	default	default
Embedded system test Chain 3	0	0	0	1

Analog/Digital Unit and discovered the incompliance of their JTAG TAP controllers.

Alternative Tradeoff: If the objective of testing a product is to take advantage of the SAMPLE mode of BS architecture during field testing, it is advised to put PLDs and every other BS devices in a static BS scan chain.

2.3 DFT Rules and Recommendations

Within our company, we have compiled testability guidelines [8, 7], such as the requirements for ICT probe point placement and JTAG compatibility, for board designers to follow during the design phase of the products. The following is a subset of DFT rules that must be obeyed in this proposed test methodology.

- Make use of JTAG devices wherever possible.
- TCK be pulled low, be buffered, and be treated as a critical net (typically fanout be no more than 10).
- TMS be pulled high (weak $\approx 5K\Omega$).
- TDI be pulled high (weak $\approx 5K\Omega$) at all potentially non-driven inputs.
- $\overline{TRST}$ be pulled low at power up and during normal device operation with capability to be driven high.

- Provide connectorized access to entire BS chain.
- Tri-state pins of non-BS devices should be accessible via other BS devices.
- Any BS enable pin must be accessible by another BS device which appears at a previous location in the BS chain.

As mention in the Section 2.2, the last DFT rule is essential to implement the abovementioned programmable dynamic BS chain.

Once the proposed BS test architecture is implemented while the DFT rules are enforced, all the proposed test modules can also be re-used in ICT, where only the five (or four) probe points that support JTAG are required.

DFT Recommendations: Besides a set of restricted DFT rules that must be followed in the proposed methodology, we also have a few DFT recommendations to increase the ease of testing and diagnosis. For instance, a zero ohm resistor is recommended to connect the TDI and TDO pins of a new BS device to enable the option of bypassing the device in case its BS circuitry does not work properly. The other common DFT recommendation is to avoid directly connecting all the TMS signals at the output of the TMS buffer. Instead, we recommend to connect each TMS with a 100 ohm resistor to the output of the TMS buffer to increase the diagnosis resolution of short faults on TMS traces.

3 System Level Test Methodology

Since the system backplane is the main communication link of all the circuit boards, its error free operation is crucial to the system's operability. We propose two test architectures to test the interconnect structure of the system bus and the interconnect structure of the backplane edge pin connectors.

The first proposed test architecture at system level is a single daisy chain which serially connects the BS chain of each board in the system as shown in Figure 2. Such a test architecture is easy to be developed and the test vectors can be generated in existing commercial tools by merging interconnects of different boards into a single netlist. It well suits for a prototype system in the DVT phase. However, it does not apply to a large system whose configuration changes constantly since it has very limited flexibility and long test time.

Hence, the second test architecture uses ASP devices to construct a multi-drop test bus architecture which consists of only five JTAG interface signals [12]. As shown in Figure 3, the proposed multi-drop test architecture contains a boundary scan master (BSM) [3] as test bus master which controls several ASP devices as test bus slaves. For example, in the wireless base station system, each of the Controller Unit, Analog/Digital Unit, and ATM module is equipped with an ASP device to communicate with the BSM that resides in a mezzanine controller module.

System Embedded Testing: On the proposed multi-drop system test architecture, a novice walking-window backplane testing algorithm has been implemented for system embedded testing [11, 2]. It is an enhanced version of the previously proposed walking-enable algorithm [6]. In the walking-window backplane testing algorithm, each slave board will be independently connected to the test master according to a test scheduling algorithm programmed in the test master. The test modules developed for each board can then be re-used with only minor modification to ensure safe system testing. This backplane testing algorithm guarantees complete fault coverage on the system backplane and is highly flexible to be applied in a dynamic system configuration with very short test time. Such a system embedded testing is most appropriate for final assembly test (FAT) and during field operation.

4 Other Advanced Test Methodologies

In this section, two advanced test methodologies will be introduced briefly. The primitive prototype test module for mixed-signal testing using BS has been successfully applied to Analog/Digital Unit. Board level interconnect BIST vectors generated by software tool, that simulates the BIST algorithm, for both Analog/Digital Unit and Controller Unit were developed. These BIST vectors have been successfully applied via existing commercial tools. This demonstrates the completeness of the BIST test pattern generation (TPG) algorithm. The BIST hardware for

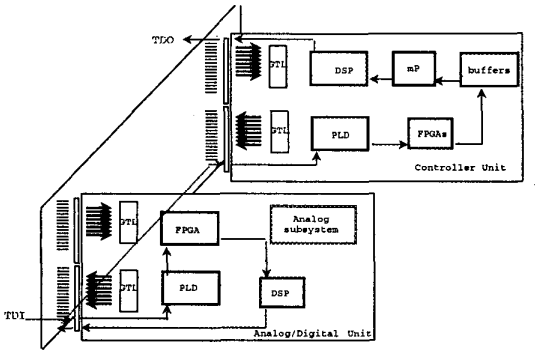

Figure 2: Daisy chain test architecture.

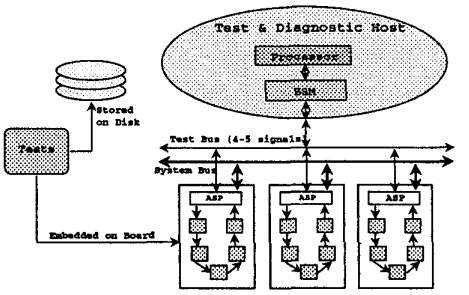

Figure 3: Multi-drop test architecture using ASP devices.

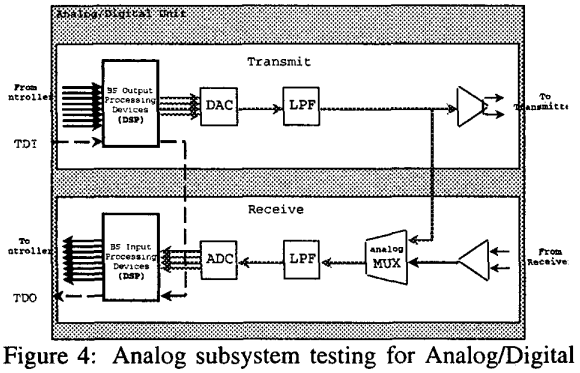

Figure 4: Analog subsystem testing for Analog/Digital Unit.

Analog/Digital Unit has also been implemented using an FPGA.

4.1 DSP-based Mixed-Signal Testing

For the analog subsystem of the Analog/Digital Unit as shown in Figure 4, once the output of transmitter is loopback to the input of receiver, DSP can be programmed via BS architecture to generate test waveforms to test the D/A converter (DAC), low pass filter (LPF), and A/D converter (ADC).

This proposed test architecture is commonly seen in most mixed-signal applications. To take full advantage of this test architecture, the keys are to develop appropriate fault modeling, fast fault simulation and test generation techniques for analog/mixed-signal circuits. We have been developing models and ATPG tools for analog faults. Compact test vectors were generated by waveform generator and applied to Analog/Digital Unit. The test results of more than ten randomly injected faults matched the prediction of the fault simulator, which demonstrates the accuracy of our fault models. We are currently developing techniques to program DSP via BS architecture to generate the test waveforms and use DSP to analyze the test responses.

4.2 Board-Level Built-In Self-Test

As mentioned earlier, IEEE 1149.1 BS architecture is used to simplify testing and diagnosis of faults in board level inter-chip interconnect. If board-level BIST is implemented then BS architecture can not only simplify testing during manufacturing of a board but also simplify the test of the system in which the board is used.

Inter-chip interconnects at board level typically contain a large number of multi-driver 3-state and bidirectional nets. If BS architecture is used to test faults in such interconnects, then the application of a test pattern that enables multiple drivers driving opposite values on a given net can potentially cause circuit damage by causing excessive current flow. While the application of such patterns can be avoided easily when deterministically generated test patterns are applied using external test equipment, a TPG for BIST must be carefully designed to ensure that no such pattern is applied. In addition to ensuring the satisfaction of this condition, the design procedure must also ensure that the resulting TPG guarantees the detection of all stuck-at faults, opens, and shorts in the interconnect circuitry. Furthermore, the above two conditions should be satisfied using a TPG of minimal size (to minimize hardware overheads) that can guarantee the detection of all faults in minimal test time.

We have developed a new BIST test pattern generator architecture and a board-level methodology to program this architecture to generate tests for any given board level inter-chip interconnect circuitry via BS architecture. The proposed BIST TPG design procedure uses the notions of *incompatibility* and newly defined *conditional incompatibility* and generates TPG designs that (i) guarantee that no circuit damage can occur due to multi-driver conflicts, (ii) guarantee the detection of all interconnect faults, (iii) have low area overheads, and (iv) have low test lengths.

5 DFT Impact on Time-To-Market

We have applied the complete board-level test methodology to all the circuit boards of a wireless base station. The test architectures at system level have also been incorporated in the design. We worked closely with designers from the early design phase, developed the test modules, debugged the design, ..., end-to-end till the project was completed. The results are encourage and satisfiable. In Table 2, we have collected designers' inputs about the time spent in DFT design and time saved during DVT by comparing with previous experiences without using the proposed BS DFT architecture. For certain circuit boards, information for time saving is not available. However, for the most complicated Controller Unit design, our fault coverage reaches 75% of board level interconnects and the proposed BS DFT architecture saves about **eight weeks** of lab time for bringing up the circuit board. Only less than a week of design time was spent in learning and implementing the DFT architectures at board and system levels.

6 Conclusion

We have presented the proposed end-to-end test methodology. It includes DFT test architectures at board and system levels and a series of test modules that can be used repeated throughout the product life cycle. Board-level test architecture uses a programmable dynamic BS chain which has exceptional flexibility and guarantees normal system function. BSM and ASP based multi-drop test architecture at system level enables system embedded testing to ensure an error-free backplane.

Implementation examples are given for the application of the proposed test methodologies on a wireless base station. It is shown in Table 2 that the proposed DFT shortened time-to-market by eight weeks for Controller Unit.

Other advanced test technologies, such as DSP-based mixed-signal testing and board-level BIST, will provide us a complete portfolio to achieve end-to-end testing which ensures the quality of our products.

7 Acknowledgment

The authors would like to thank James S. Granitzki, James C. Gualdoni, Edward Wronka, Sotirius (Steve) Manolis, and Tom G. Schneider, who support the design of the proposed DFT techniques and feedback us valuable ideas with their design expertise. Hats off also to Bradford G. Van Treuren and Paul J. Wheatley who support software for embedded testing.

References

[1] M. Abramovici, M. A. Breuer, and A. D. Friedman. *Digital Systems Testing and Testable Design*. Computer Science Press, New York, N.Y., 1990.

[2] K. C. Anderson, R. W. Barr, J. S. Granitzki, J. M. M. B. G. V. Treuren, and P. J. Wheatley. Embedded Test Architecture. In *Proceedings Lucent Test Conference*, 1999.

[3] 497AE and 1215E Boundary-Scan Master 2 User Manual, 1998. Lucent Technology Microelectronics.

[4] IEEE. *IEEE Standard Test Access Port and Boundary-Scan Architecture*. IEEE, 1993.

[5] T. I. Inc. SN74ABT8996, Addressable Scan Ports, Multidrop-addressable IEEE STD 1149.1 TAP Transceivers, Aug 1994. TI Web Site, SCBS489.

[6] W. Ke. Backplane Interconnect Test In A Boundary-Scan Environment. In *Proceedings IEEE International Test Conference*, pages 717–724, 1996.

Table 2: JTAG DFT impact on time-to-market.

Circuit Pack	*Time for DFT Design*	*Time for Test Development*	*Fault Coverage*	*Time Saved*
Controller Unit	$<$ 1 week	1-2 weeks	75%	$\approx$ 8 weeks
Analog/Digital Unit	$<$ 1 week	$<$ 1 week	60%	N/A
Clock Unit	1 day	1 day	15%	$\approx$ 1 week
ATM Unit	$\approx$ 2 weeks	$<$ 1 week	45%	N/A

[7] D. Le and T. Bahlman. Lucent-Wide Design for Testability Standard. In *Proceedings Lucent Test Conference*, 1999.

[8] OneDFT Testability Guidelines for Circuit Packs. Lucent Technologies.

[9] C. M. Maunder and R. E. Tulloss. *The Test Access Port and and Boundary-Scan Architecture.* IEEE Computer Society Press Tutorial, 1992.

[10] K. P. Parker. *The Boundary-Scan Handbook.* Kluwer Academic Publishers, 1992.

[11] B. G. V. Treuren and P. J. Wheatley. Embedded Test Architecture. In *Proceedings Lucent Test Conference*, 1999.

[12] L. Whetsel. A Proposed Method of Accessing 1149.1 in a Backplane Environment. In *Proceedings IEEE International Test Conference*, pages 206–216, 1992.