Planning for Built-In Self Test to Handle your Testing Needs
Published in The BestTest Newsletter, Dec 1, 2004

Introduction

Built-In Self-Test (BIST) is no longer a pie-in-the-sky dream. BIST can be incorporated into the integrated circuit (IC) chip, into a field programmable gate array (FPGA) or other programmable logic device (PLD), onto the circuit board or into a system. BIST can be realized in hardware, in software, or in firmware. It can apply to digital circuits, to analog circuits and to mixed-signal circuits. BIST can be made to merely report the existence of faults, to provide diagnostic information, or to provide prognostics. In some cases, such as with memory BIST, even built-in repair of failing memory cells is possible. BIST can be an extension of testability techniques, such as the JTAG/IEEE-1149.1 boundary scan, or be totally independent of it. In attempting to provide guidelines on how to go about selecting BIST, four leading experts in the field have combined their expertise to try to address as many of these considerations as possible.

In selecting from various BIST methods, one should consider the fundamental advantages BIST offers over ATE. First, BIST is always available to run the test and does not require external physical test equipment. As a result products can be tested after fielding. Chips with BIST can be tested on boards and in systems, and boards with BIST can be tested within the actual operating system at operating speeds. Second (an often overlooked feature) is that test development with BIST should be trivial, inexpensive and completed in parallel with the design. This enables designs with BIST to get to market faster, which can greatly increase the overall profitability of the product.

We will start our discussion of BIST as it applies to digital integrated circuits (ICs). Next, we look at analog and mixed-signal applications that pave the way to Prognostics/BIST. We then expand our focus to encompass board-level and system-level application of BIST. In each of these cases we weigh using BIST against using traditional automatic test equipment (ATE) tests. Our discussion briefly discusses software built-in test at the system level, but the remainder of the discussion is centered on BIST hardware.

Chip-level Built-In Self-Test Strategies and Solutions

Before deciding on a test strategy, it is important to understand why a device is being tested. Chip manufacturers use BIST for one of two primary reasons: in-system test or manufacturing test. Certain devices must be testable without the use of ATE, and BIST is the ideal solution for such devices. BIST enables these devices to be tested on the board or in system as well. Since it does not make economic sense to install a defective part on a board during manufacturing, using chip-level BIST to verify a part before it is assembled onto the board is an important part of an effective manufacturing test strategy. For logic testing, better manufacturing test solutions that provide high quality and lower test cost should be used when in-system test is not a requirement. For manufacturing and in-system test of memories, BIST continues to be very effective, although new techniques such as Macro Testing provide designers with additional options.
**Logic BIST**

Companies that manufacture devices for satellite and remote applications have used logic BIST for many years. Typically, a BIST controller includes a Pseudorandom Pattern Generator (PRPG), a Multiple Input Shift Register (MISR), control logic, and test points for increasing testability. The pseudorandom patterns generated by the PRPG are applied to the device through scan chains and results are captured in the MISR which calculates a signature based on circuit response. The test fails if the final signature is different than a pre-calculated value. Test points are added to the design in order to increase controllability and observability of areas in the circuit that are hard to test with pseudorandom patterns. Additionally, test points are used to prevent unknown states from propagating to the MISR and corrupting the signature. Insertion of this additional logic is acceptable because otherwise it would be impossible to test the device when it is deployed in the field and inaccessible by a tester.

Another important motivation for BIST has been to lower manufacturing test cost. BIST does not require any test patterns to be stored on the tester, yet logic BIST has not proven to be an effective manufacturing test strategy for several reasons. Logic BIST solutions use random patterns unable to achieve the high test quality levels possible with deterministic pattern generation. After running the BIST patterns on the tester, manufacturers must apply additional deterministic patterns in order to achieve high test coverage for all necessary fault types (stuck-at, at-speed, etc.). Additionally, since BIST requires many more test cycles than deterministic automatic test pattern generation (ATPG), test application time is typically much longer resulting in higher test costs. An important aspect of manufacturing test is the need for diagnostics. Diagnosis with a MISR signature is cumbersome and time consuming because visibility into the design is limited. [LYU1]

So a deterministic solution is ideal for manufacturing test and effective diagnostics. Deterministic ATPG addresses the test quality and diagnostics issues by providing targeted test patterns without the drawbacks of BIST. Moreover, Embedded Deterministic Test (EDT™), an advanced compression technology, solves the test application time while maintaining the high test coverage achieved through the use of deterministic patterns. By adding a very small amount of logic (less than 1%) in the scan path, EDT reduces test application time and pattern volume by up to 100x while maintaining high test quality. EDT[LYU2] patterns can be diagnosed on the tester in the exact same manner and accuracy as normal ATPG patterns.

**Memory BIST**

Memories of various sizes and configurations make up more than 50% of today’s nanometer designs thus increasing the importance of quality memory testing. Because of the structured construction of memories, algorithmic test solutions are ideal for providing quality and comprehensive test. Most algorithms, such as the Marching algorithm, apply various background patterns to the memory using a specific sequence of addresses. A variety of algorithms are needed to test each memory location, its neighboring cells and the address decoder logic. Another key requirement for quality memory test is the ability to test the memory using its functional frequency. At-speed testing ensures that the
memory can be written to and read from in the same manner as its intended system functionality.

Designers typically have two options for applying memory test algorithms to the design’s numerous memories. Traditional BIST controllers provide on-board logic that apply the pre-specified algorithm, monitor the memory’s outputs, and provide a pass/fail flag at the end of test. They also provide diagnostics information to the tester if failures are detected on chip. To effectively test for defects, test application and diagnostics must be done at-speed. Additionally, new fault models and algorithms are needed in order to keep up with ever-changing defect mechanisms. This implies that the practice of using a legacy BIST controller over several projects is outdated and dangerous. The test approach must be reevaluated for each design in order to determine the best solution for the types of memory used, their location, as well as the target manufacturing process and tester.

BIST has been and continues to be the preferred test solution for large memories on a device. Multiple memories can be tested with a single BIST controller especially if they are geographically close to each other. However, today’s designs contain many smaller memories that are scattered across the chip making it hard to justify the area and routing overhead of one or several BIST controllers. A technique called Macro Testing[LYU3] is capable of testing the small and timing critical memories without the need for a BIST controller or additional logic. Macro Testing uses the existing scan chains in a design to deliver the required algorithm to the memory and observe its response. Additionally, these tests can be applied at-speed in order to meet nanometer test requirements.

**Boundary Scan**

Although boundary scan was primarily defined in the IEEE 1149.1 standard as a mechanism for board testing, it has become the standard interface for controlling on-chip test structures like BIST. The ability to define private instructions in the control registers enables BIST operations and to monitor the results after the test has completed. Additionally, the existence of a simple and standard test access port (TAP) provides boundary scan access to the chip even after it is assembled on the board and in the system.

**Digital BIST Selection Criteria**

As designs move to 130 nanometer and smaller geometries, the importance of quality chip-level test has become even more apparent. BIST solutions are effective for designs that require board or in-system testing. For logic test, effective manufacturing test can be achieved by using deterministic pattern sets generated by ATPG and EDT products. BIST continues to be an effective in-system and manufacturing test solution for large memories. For small and timing-critical memories, Macro Testing provides a non-intrusive solution capable of testing multiple memories in parallel. All test solutions used on modern designs must be applied at-speed in order to detect new and ever changing defect types.
**Mixed-Signal BIST and Prognostics**

BIST offers significant advantages to IC, Board and System Level test applications. Prognostics/BIST is a close relative of Analog/Mixed Signal BIST and can be applied at all thereof these levels of abstraction. Through the observation of precursor signatures, Prognostics predicts latent defects and failures before they occur, providing a useful augmentation to BIST-based test backbones.

Prognostics/BIST, as applied in wider-scale Prognostics/Health Management (PHM) systems in Electronic Systems, offers many benefits. These include improved operational readiness in critical systems, and optimization of the logistics support systems through provisioning of spare boards and modules.

The focus below will be on selecting a BIST architecture that is extensible for adding Prognostics/BIST.

**Analog/Mixed Signal (AMS) BIST**

Analog/Mixed-Signal (AMS) BIST is more complex than digital BIST implementations, primarily due to the various levels that analog test conditions can assume. While a Digital BIST implementation permits examination of register values, Analog is more involved. For example, to test a Mixed Signal stage, such as an Analog-to-digital converter (ADC), the designer must establish the performance criteria. Integral Non-Linearity (INL) provides a measure and extracts the value in steps. While more complex, designers often find that the advantages outweigh the initial complexity of designing the AMS BIST into the circuit. Today, there are BIST test structures that can test ADCs, DACs Amplifiers, and PLLs.

**Prognostics/BIST**

Because Prognostics/BIST requires similar kinds of measurement and extraction, Prognostics/BIST is a close relative to AMS BIST. With Prognostics, the designer is implementing a method to detect an impending failure in the PCB or Module before it occurs. This is a newer technique that can be quite effective for critical systems such as Avionics, Automatic Teller Machines, and Telecom Switches. Because refinements in BIST infrastructure have been developed for Digital and Analog BIST over the years, Prognostics/BIST can readily apply the same collection and extraction methods employed by the standard BIST implementations. Prognostics have been successfully applied on systems ranging from critical ICs for space applications to high efficiency DC to DC power converters.

**Noise issues**

Noise is an issue with all new designs. The type of PCB or module must consider the noise climate inside the system. If there is a great deal of EMI present it could corrupt your measurements. Some kinds of analog measurements are simply too difficult to make in a noisy environment. Alternatively, the signals themselves can be corrupted by noise before they are measured. With good layout and grounding methods, these kinds of issues can be minimized.
Efficient extraction of data via the scan bus
Some designs take advantage of IEEE1149.1 to distill analog measurements to digital representation, before extraction by the scan data bus. For example, using an Oscillation BIST approach, an amplifier could be made to deliberately oscillate during a BIST test sequence. The frequency of this forced oscillation is correlated with the amplifier’s performance, and can be extracted, compared to a reference and a simple “1” or “0” entered to the scan bus register. In other words, analog measurement, but digital extraction. Other techniques, such as using the analog extensions to the data bus, are also possible but noise considerations are important to analyze before using this approach.

Advantages and positioning with respect to Analog/Mixed-Signal BIST
Prognostics/BIST solutions have been developed for IC, PCB and Module level. With ever-shrinking process geometries, some designers have expressed concern about the ultrathin gate oxide layers and other fragile parameters. IC life is no longer measured in decades, but rather, in years. These lifetimes are also reduced through exposure to radiation, temperature and other severe conditions. Prognostic/BIST cells have been developed to be co-located with the host IC to indicate an impending failure so that averting actions can be taken. At the PCB or module level, IC’s are not the only failure-prone components, so designers have adopted techniques to measure key pre-cursor signatures associated with failures in other electronic components such as capacitors, power MOSFETs, and even solder joints.

Many of the same integration techniques used for AMS BIST are also applicable to Prognostics/BIST, including careful layout, low-noise analog design techniques, and the use of the scan bus to extract the information. A key issue in the design is making the system “Fail-Safe’ so that the BIST does not damage or impact the host circuit.

Prognostics Application: High Efficiency Power Converter
A good “acid-test” example is applying Prognostics/BIST to standard Commercial-Off-theShelf (COTS) power modules. Power modules are usually the weak link in systems. Plagued by repetitive stress conditions, high temperatures, and confined space, the power modules are at the top of the reliability problem list. However, using Prognostics design analysis tools and IP, it is possible to “instrument” a power supply with Prognostics/BIST, and provide a “gas gauge” type of indication on the remaining useful life available in the power supply. This can be very valuable to critical systems in aerospace and commercial applications where it can be tied to a comprehensive Prognostics/Health Management (PHM) system.

Mixed-Signal and Prognostics/BIST Selection Criteria
BIST provides unique capabilities to designers who want to develop robust and easily tested systems. It is also possible to add Prognostics to the implementation, to provide an added dimension of providing estimated remaining life to the users. For critical systems, this is extremely valuable information. There is a need for further development of the physics-of-failure library and additional tools to assist in the integration process.
**Board and System Level BIST Approaches**

In-circuit ATE (ICT) cannot test printed circuit boards (PCBs) throughout the product’s life. Failures are introduced by temperature, humidity, vibration and other environmental factors during the product life.

Failures can occur during system integration, shipping or worse at the customer site. ICT may not be able to test the entire PCB as many of the daughter cards, memory DIMMs, Processors may not be plugged into the card until system integration. When BIST is executed on-board, all of these plug-ins are available to test.

Another important advantage that all board-level BIST mechanisms offer is the ability to test the circuit board in its normal operating surroundings. With board BIST, however, the circuit card is not in an artificial test environment, but in its normal system environment, with system power supplies and other PCBs that make the system operate. BIST can find faults that escape ATE. For instance, when a PCB is in-system, the PCB temperature is elevated, sometimes resulting in intermittent opens for ball grid arrays (BGAs) as the FR4 expands. Similar faults can be found by BIST during Environmental Stress Screening (ESS).

Board-level ATE, such as ICT, cannot test the interaction of components at-speed or between PCBs. Board BIST can exercise IC level BIST, IC-to-Memory at-speed tests, and IC-to-IC at-speed interconnects, which typically cannot be run in a pogo-pin environment. BIST measurements are not compromised by wiring, switching and interfacing between the unit under test (UUT) and the ATE. With IEEE 1149.4 and 1149.6 mechanisms, mixed-signal observation and control points as well as AC coupled nets can be tested in addition to digital signals through the use of Board and System level BIST.

**Software based Built-in-Self-Test**

Use the term BIST in some circles and it means writing CPU specific software code to exercise the board or system functionally. The diagnostic firmware is developed by test engineers, diagnostics engineers and stored on-board the product in the host CPU’s FLASH memory or FLASH card. These embedded tests are then used as a means to test the integrated systems, both in manufacturing and in the field. There are several disadvantages to this approach:

- Diagnostic code development is not automated requiring significant resources
- Engineers must be familiar with the functional designs, the effects of various faults and fault classes, in order just to begin to write software.
- Test quality and fault coverage can not be easily or automatically measured
- Software based embedded test requires a (mostly) working CPU to execute.
- Diagnostic isolation for board repair is often poor.
- High risk in that if the fault is missed in the field, development work is wasted effort.
Isolating the fault is important so the proper corrective action can be taken either during manufacturing or during field service. The sheer complexity of a multi-PCB system makes writing software to cover all the faults and correctly diagnose the fault a formidable task. Due to this, many companies have looked to have lower level hardware tests, and IC level self-test functions executed automatically.

**Board and System BIST in a Chip**

Single IC devices exist today to either assist or completely manage PCB level or System Level BIST. In FPGA based systems, typically the IC design could be loaded into a FPGA temporarily, removing some of the component cost. Two approaches are discussed in the next sections.

**Functional Test Analyzer**

From the above discussions on digital BIST applied to ICs, we find that approaches such as Logic BIST offer little in the way of diagnostics. It is possible, however, through board-level Design for Testability to strategically partition the circuit board and utilize a number of signature analyzers so that faults can be attributed to one of the partitions. This process, however, is complicated by the fact that in uninitialized circuits, X (don’t care) states will compromise the integrity of the results. One patented design, called Built-In Test Exerciser and Sensor (BITES), combines pseudorandom patterns with deterministic test generators and response collectors so that X states can be eliminated from the test data processed. Additionally, BITES is a uniform structure with a known test sequence, so test program development remains trivial. BITES can be added as an extra chip to the circuit board or programmed into an existing FPGA. Communication with the BITES structure is accomplished via the IEEE-1149.1 port, which most FPGAs are equipped with. Once the FPGA is uploaded with BITES it can be used to test the FPGA internally. Then the same BITES structure can be used to test the remaining part of the board, providing fault isolation to one of 64 internal points. Additionally, BITES can process non-X state boundary-scan data, thereby improving its observability beyond its 64 observation points to include all points connected to a boundary-scan cell.

**Embedded IEEE 1149.1 Tester**

IEEE 1149.1 / JTAG can test large portions of a digital PCB today. New advances have been made that enable a single IC to take the place of the PC based boundary scan tester. The SystemBIST processor from Intellitech was developed to address this application.

It functions as an embedded centralized manager for configuring and testing PCBs and Systems. By including the BIST Processor in products, board and system designers can simplify in-system device configuration (FPGAs) while enabling comprehensive structural test throughout the system, including the system’s CPU. The BIST Processor can be provided as an IC, download-able intellectual property (IP) binary or as infrastructure IP that can be embedded in an ASIC. The processor can be used at the board and system levels and allow designers to take advantage of cost efficiencies over the entire product life cycle. It can also provides for a scalable and reusable methodology, which augments existing test and configuration standards.
If we embed SystemBIST, we can eliminate the need to run boundary-scan based digital tests on ICT equipment. This lowers manufacturing costs by greatly reducing the time a board spends sitting on higher cost capital equipment. SystemBIST also will run 1149.1 assisted at-speed tests not possible during ICT. It allows the same set of high quality tests to be used in many different environments and throughout all phases of the product’s life cycle. This includes lab prototyping, volume PCB manufacturing, system integration, vibration test, HALT/HASS test, power-up self-test, field service and depot repair. At power-up, or under CPU start, a single BIST Processor can automatically run the entire manufacturing test stream, including ASIC ATPG tests, logic BIST, memory BIST, and board/system interconnect tests – as well as configure all the programmable logic devices in the system.

**Better, Flexible FPGA configuration - Get PCB BIST for Free**

Many products utilize programmable logic devices, such as FPGAs, CPLDs, and programmable non-volatile memories, such as EEPROM, Serial EEPROM and FLASH. These devices support the adoption of programmable architectures, which enable system designers to achieve a quick time to market through field upgrade-able fixes and enhancements. In-the-field reconfiguration provides a compelling value proposition as the enhancements can extend the life of the product and also provide downstream revenue. These benefits have great economic appeal versus the cost of putting a part on-board. You might be saying, “Even so, I can’t afford to put a part on my PCB”, however, SystemBIST removes many other parts that would be required without SystemBIST such as configuration PROMS. The total parts cost and count on a FPGA based PCB with SystemBIST is less than the parts count without it.

In order to remotely upgrade a system in the field, access to each volatile and non-volatile device over in-system mission-mode busses is increasingly becoming more difficult. This is especially true for boards with mezzanine cards or multi-board backplane based systems. For the system designer, these capabilities can add to the costs and design effort required to develop, and later to manufacture such configurable products. Often, the designers create their own ad-hoc methods, which are costly and time-consuming and do not provide for a solution that is readily re-usable on future product designs.

The SystemBIST processor is a vendor independent FPGA programming solution, eliminating the need for proprietary configuration PROMs or ad-hoc FLASH based solutions. As a result, designers no longer need to develop customized methods and “one-off” designs for embedded in-system solutions. Further comparisons with FPGA based programming methods can be found in the following white paper: [http://www.intellitech.com/pdf/itc_systembist.pdf](http://www.intellitech.com/pdf/itc_systembist.pdf). You may want to look at the article published in IEEE D & T Magazine on Infrastructure IP compared with home-brews approaches, [http://www.intellitech.com/pdf/ieeedandt.pdf](http://www.intellitech.com/pdf/ieeedandt.pdf).

**Hierarchical BIST**

The interaction of BIST at all these levels is known as Hierarchical BIST. As one selection criterion, board-level BIST should be able to invoke IC-level BIST and system-
level BIST should be able to invoke board-level BIST. This not only capitalizes on previous BIST investments, but it also provides better diagnostic measures and reduces life-cycle costs.

**Conclusion**

A desired utility of BIST is to eliminate the dependence on ATE. Boundary-scan and the IEEE-1149.1, 1149.4 and 1149.6 have gone part of the way by reducing both the test equipment hardware and the test program development costs that ATE test requires. BIST is tasked with completing the job, by making tests less dependent (if not totally non-dependent) on sophisticated test equipment. Similarly, BIST will make test program development a simpler task.

The use of structured techniques for test, such as 1149.1 and BIST, will increase as boards and systems become more complex, with higher IC-to-IC speeds and with less physical access. At-speed 1149.1 based tests are becoming more common place; at-speed 1149.1 based memory test is more the rule than the exception today. Embedded structural test will replace software-based functional testing at all but the highest-levels of abstraction.

BIST is not a futuristic dream, but an available technology. It has transformed from an application-specific test developed after intimate knowledge of the unit under test, into universal structures that can be implemented into ICs, boards and systems. These structures lend themselves to faster test development, reductions in time-to-market and post-deployment support. They offer a remarkable economic alternative to other forms of test.