# Built-In Self-Test of Programmable Input/Output Tiles in Virtex-5 FPGAs

Bradley F. Dutton and Charles E. Stroud
Dept. of Electrical and Computer Engineering
Auburn University
Auburn, Alabama 36849
duttobf@auburn.edu strouce@auburn.edu

***Abstract*— A Built-In Self-Test (BIST) approach is presented for the logic resources in the programmable input/output (I/O) tiles in Virtex-5 field programmable gate arrays (FPGAs). A total of 15 BIST configurations were developed to test the I/O cell programmable logic resources in all modes of operation. The approach utilizes dedicated I/O buffer bypass routing in the I/O tile such that the BIST is package independent and applicable to all levels of testing from wafer-level to system-level. The approach offers control of BIST execution and maximal diagnostic resolution of faulty I/O tiles for device and package independent testing. Either the Boundary Scan interface or a simple system-level interface may be used for BIST execution, control, and diagnosis independent of the configuration interface. Experimental results are presented including fault detection capabilities.**

***Keywords - Built-In Self-Test; Field Programmable Gate Array; programmable input/output tiles; Virtex-5***

## I. INTRODUCTION

The input/output (I/O) buffers of JTAG compliant devices are typically tested using the Boundary Scan EXTEST feature [1]. However, field programmable gate arrays (FPGAs) have a significant amount of configurable logic resources associated with the I/O buffers that cannot be tested in this manner. These configurable logic resources typically include multiplexers and flip-flops/latches, as illustrated in Fig. 1, for improving system timing specifications such as set-up and hold times as well as clock-to-output delay. Additional logic resources are included to support single data rate (SDR) and double data rate (DDR) transmission and reception as well as for serialization/de-serialization (SerDes) modes of operation. In Xilinx Virtex-5 FPGAs, for example, there are at least 32 multiplexers and 47 flip-flops included in the configurable logic associated with each I/O cell to support various modes of operation. The Boundary Scan INTEST feature can be used to test the configurable logic resources in an I/O cell [1]. However, the INTEST feature is supported by few FPGA manufacturers. While there has been some prior work in testing I/O cells [2][3][4][5], previous work in Built-In Self-Test (BIST) for FPGAs has largely overlooked I/O cells and their associated logic resources. However, it has been observed that the programmable logic in unused or un-bonded I/O cells is sometimes used by FPGA synthesis tools for implementing system logic functions [5].

The work presented in this paper builds primarily on the prior work in [5], in which an I/O cell BIST architecture was proposed and implemented for Atmel AT40K series FPGAs and Atmel AT94K series programmable system-on-a-chip (SoC) [6]. However, this paper offers several improvements over that previous BIST approach. In addition, this paper describes the actual implementation, operation, and verification of BIST configurations developed for Virtex-5 FPGAs [7] whose I/O cells are much more complex than those found in the AT40K and AT94K devices [6]. The BIST configurations presented here test the full functionality of logic resources included in the Virtex-5 I/O cells including input logic (ILOGIC), output logic (OLOGIC), as well as input and output Serializer/Deserializer (SerDes) operation. The paper begins with an overview of the prior work in I/O cell BIST in Section II, followed in Section III by an overview of Virtex-5 I/O tiles. The overall BIST approach is described in Section IV, and details of the specific BIST configurations are discussed for Logic and SerDes modes in Sections V and VI, respectively. We present experimental results from actual implementation in Virtex-5 FPGAs in Section VII. Section VIII discusses a BIST approach for the configurable I/O buffers before the summary and conclusion in Section IX.

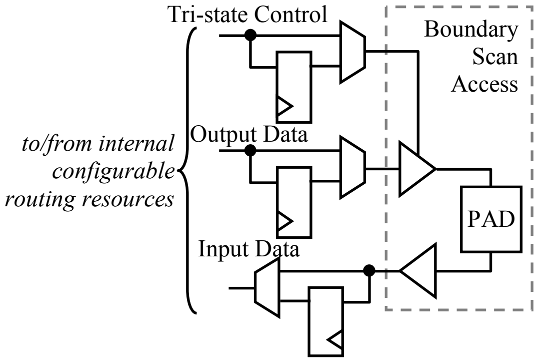

Figure 1. Simplified programmable I/O cell

## II. PRIOR WORK

There has been limited prior work in the area of testing I/O cells in, or applicable to, FPGAs [2][3][4][5]. In [5], a system-level BIST architecture is presented for the I/O cells of Atmel

---

This work was sponsored by the National Security Agency under contract H98230-04-C-1177 and supported in part by the National Science Foundation Grant CNS-0708962.

FPGAs. The overall BIST approach was similar to that used for configurable logic resources in the FPGA core [8]. The BIST architecture in [5] consists of a single TPG implemented in configurable logic blocks (CLBs) sourcing test vectors to the I/O cells under test. A single TPG was implemented under the assumption that internal FPGA resources had already been tested and found to be fault-free. The I/O cells under test are identically configured with bidirectional I/O buffers such that the output responses are sent back into the FPGA internal resources. However, for in-system testing, this requires that all connecting devices be tri-stated during testing. The output responses of the I/O cells are monitored by CLBs configured as comparison-based output response analyzers (ORAs). While presenting a general architecture applicable to any FPGA or configurable SoC with an FPGA core and bidirectional I/O buffers, [5] implemented 27 BIST configurations applicable to the Atmel AT94K SoC and AT40K FPGA only.

## III. OVERVIEW OF VIRTEX-5 I/O TILES

The I/O cells in Virtex-5 FPGAs include an output logic block (OLOGIC), input logic block (ILOGIC), I/O delay block, and a bidirectional I/O buffer, as illustrated in Fig. 2 [7] The number of I/O cells in Virtex-5 ranges from 360 to 1,200 depending on the size of the particular FPGA.

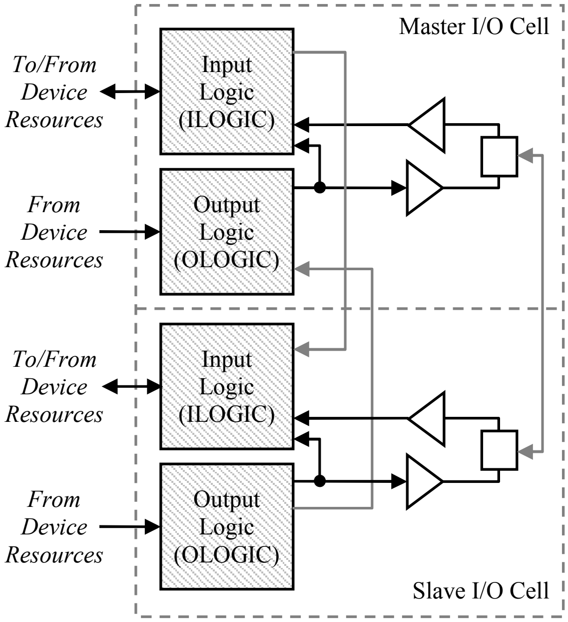

Figure 2. Virtex-5 programmable I/O tile

Each OLOGIC includes registers for improving system clock-to-output timing and supporting SDR and DDR transmission of data. The OLOGIC can also perform parallel-to-serial conversion of output data for widths between 2 and 6-bits when operating in SerDes mode. The ILOGIC includes registers for improving system set-up and hold times and supporting SDR and DDR reception of data. It can also perform serial-to-parallel conversion of input data for widths between 2 and 6-bits when operating in SerDes mode. The ILOGIC also incorporates a Bitslip sub-module for synchronizing serial interfaces that include a training pattern. Invoking the Bitslip input re-orders the data on the parallel outputs of the input logic block in a barrel-shifter operation [7]. In Virtex-5 FPGAs, two I/O cells are grouped to form an I/O tile, as illustrated in Fig. 2. Each I/O tile includes dedicated shift routing to support expanded SerDes data widths. In master/slave mode, two I/O cells in the same I/O tile are connected via the dedicated shift routing to support data widths of 7, 8 and 10-bits [7]. Each I/O cell also includes dedicated routing (also shown in Fig. 2) directly from the OLOGIC to the ILOGIC that bypasses the I/O buffer.

## IV. OVERVIEW OF BIST ARCHITECTURE

Our BIST approach for I/O tiles is similar to other BIST approaches that we have developed for testing CLBs in Virtex-4 and Virtex-5 FPGAs [9]. A set of deterministic test patterns is stored in 36-kbit block random access memories (RAMs) in the FPGA fabric. The outputs of the block RAMs are connected directly to the inputs of alternating rows of I/O tiles under test. One block RAM is configured for every 5 rows of I/O tiles under test. One digital signal processor (DSP) per block RAM is configured as a counter to sequentially address the block RAM. Collectively, one 36-kbit block RAM and one DSP form the TPG for every I/O tile BIST configuration. However, the block RAM contents are modified for some configurations to target specific resources/functions under test. The advantage of configuring multiple TPGs is twofold: first, multiple TPGs reduce loading, thereby maximizing the BIST execution frequency in large devices, and, secondly, configuring multiple identical TPGs eliminates the assumption that the TPG logic resources are fault-free. Any fault affecting the behavior of a TPG will be detected by the comparison-based ORAs monitoring the I/O cells at the boundaries of any faulty and fault-free TPG.

BIST of I/O cells is well suited for circular comparison-based ORAs since many identical I/O cells are tested simultaneously. The outputs of each I/O cell under test are monitored by two ORAs and compared with the outputs of two other identically configured I/O cells in an adjacent row, as shown in Fig. 3. To complete the circular comparison, I/O cells in the top row of the test area are compared with I/O cells under test in the bottom row of the test area.

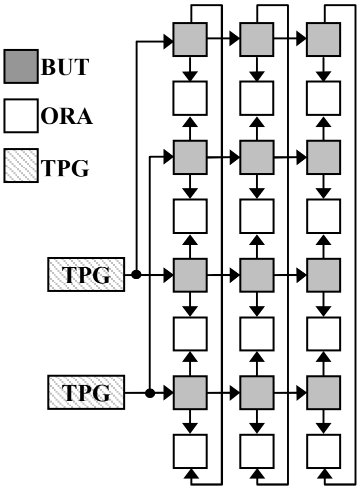

Figure 3. Column oriented circular comparison

The circular comparison approach does not suffer from aliasing effects as long as all of the BUTs being compared do not fail identically and at the same time. Furthermore, circular comparison improves diagnostic resolution [4]. An output response mismatch between two identically configured I/O cell outputs is latched as a logic 0 in the ORA flip-flop for the duration of the test session. Otherwise, logic 1 is retained in the ORA and is interpreted as a passing result at the conclusion of the BIST sequence. In previous implementations of the comparison-based ORA, the dedicated carry logic and routing resources in the ORA CLBs were un-used [4]. However, in all BIST configurations that we have developed for Virtex-5 FPGAs, these resources are utilized to form an iterative-OR chain of every ORA in the test area. In each ORA, a passing result of logic 1 selects the Carry-in input to the CLB, which is the Pass/Fail result of an adjacent ORA. The carry-in input of the first MUX in the iterative-OR chain is connected to a system input, with the carry-out of the last ORA connected to a system output. If any ORA in the chain records a failure (e.g. mismatch), a logic 0 on the output of that ORA will select a logic 1 as the input to the carry MUX, as illustrated in Fig. 4.

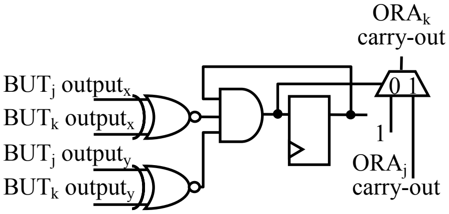

Figure 4. Virtex-5 equivalent ORA architecture

If no failure is observed in the ORA, the carry-in input is propagated through the CLB. If no ORAs in the iterative-OR chain observe failures, the carry-in input to the first ORA in the chain will propagate through every ORA slice to the carry-out output of the final ORA such that an overall pass/fail result is obtained without reading back the configuration memory to obtain the contents of the ORA flip-flops. By toggling the OR-chain input and observing the OR-chain output at the end of each BIST sequence, the integrity of the iterative OR-chain is verified. If the output of the iterative OR-chain indicates failures were detected, the contents of the ORAs can be retrieved via partial configuration memory readback for precise fault diagnosis.

Another important difference between our I/O tile BIST architecture and the prior work is in the configuration of the I/O tiles under test. Previous approaches have relied on bidirectional I/O buffers to provide the return path for test patterns exiting the output logic and returning to the ORAs via input logic [5][10]. However, the reliance on bi-directionally configured I/O buffers severely limits the applicability of this type of BIST for in-system testing. With every I/O buffer configured in the path of the logic under test, we required that all connecting devices be tri-stated during in-system testing. Connecting passive devices, such as termination resistors or light emitting diodes (LEDs), introduce another problem since these devices cannot be disconnected or tristated during in-system tests. In [9], the authors observed that, at certain frequencies, LEDs connected to I/O buffers under test caused the comparison ORAs to erroneously report failures for otherwise fault-free I/O tiles. These failures were observed at frequencies as low as 325 kHz [9], which is unacceptable for an at-speed test of the logic resources. As a result, the generality of the BIST is compromised. Fortunately, the I/O tiles in Virtex-4 and Virtex-5 FPGAs include dedicated routing from the OLOGIC to the ILOGIC that bypasses the I/O buffer [7]. Using this feedback routing instead of the I/O buffer means that no signals from the FPGA under test can reach, and therefore be influenced by, external devices. Furthermore, bypassing the I/O buffer does not sacrifice fault coverage in the I/O tile logic resources. With the I/O buffers removed from all tests for logic resources, these tests may be applied without concern for the external test environment, thus making our approach applicable to all levels of FPGA testing.

The obvious disadvantage of this approach is that it does not concurrently test the I/O buffer. However, we have developed a stand-alone BIST architecture for the I/O buffers that is applicable to device and wafer-level testing. This architecture tests the programmable analog features of the I/O buffers in every bidirectional mode of operation. Additionally, the Boundary Scan EXTEST feature may be used for in-system tests of the I/O buffers in their system mode of operation.

## V. CONFIGURATIONS FOR I/O LOGIC MODES

Six test configurations are required to fully test the I/O tile logic resource in all ILOGIC/OLOGIC modes of operation. The I/O delay module is concurrently tested in these I/O Logic mode tests in two of three modes of operation. Feedback routing from the OLOGIC to the ILOGIC has two possible routes: one through the I/O delay module and one dedicated route which bypasses the I/O delay module. The route through the I/O delay module allows for testing of the output delay functionality in all supported delay modes (fixed delay, variable delay, and default). However, testing delay of input and output signals simultaneously is not possible without configuring the I/O buffers in bidirectional mode. Three of the six I/O logic BIST configurations test the DDR transmit and receive modes of operation, including, in the OLOGIC, opposite-edge, same-edge, and same-edge pipelined output modes. The fourth and fifth configurations test the flip-flop and latch functionality of the primary registers. In the sixth and final configuration, the combinatorial (un-registered) path through the I/O tile logic resources is tested. Programmable initialization values, set/reset values, and synchronous/asynchronous reset/toggle inputs are concurrently tested. The number of clock cycles for BIST execution is 2048 for all I/O Logic BIST configurations.

## VI. CONFIGURATIONS FOR I/O SERDES MODES

A total of nine configurations are required to fully test the I/O tile logic resource in the SerDes modes of operation. Six of these configurations test the I/O SerDes logic configured for data widths of 2, 3, 4, 5, and 6-bits. Two configurations are included for the 4-bit data width to test the programmable active level on the tri-state inputs of the OLOGIC. Another three configurations test the master/slave SerDes modes for data widths of 7, 8, and 10-bits. Two of the nine configurations test the SerDes in DDR mode, with the other seven configurations testing SDR modes of operation. SerDes operations require two clocks: a high speed clock for serial data and a divided clock for the FPGA fabric. The amount of clock division is an integer equal to the data width when testing SDR modes, and is half of the data width when testing DDR modes. We use regional clock buffers with integrated clock division, called BUFRs [7], to provide the divided clock for the ORAs and TPGs in SerDes configurations. The BUFR has programmable clock division, from 1 to 8, and BYPASS modes. There are also clear (CLR) and clock enable (CE) inputs to the BUFR. We connect the CLR and CE inputs of every BUFR to the TPGs to achieve a simultaneous test of the BUFRs and the I/O SerDes logic. Concurrent testing of the BUFRs is beneficial since they would likely be used in conjunction with SerDes. Since each BUFR clocks only one adjacent clock region, a faulty BUFR will cause failures in the ORAs along at least one boundary of an adjacent clock region. As with the I/O tiles under test, a faulty BUFR can only escape detection if every BUFR in the test area fails identically and on the same clock cycle(s).

One addition to the BIST architecture for SerDes mode testing stems from the need for synchronization of the serial bit streams before executing the BIST sequence. In SerDes mode, the positioning of deserialized data on the parallel side of the OLOGIC is initially indeterminate. Due to the nature of comparison-based ORAs, data on the parallel outputs of every I/O cell under test must be synchronized. To ensure identical alignment of deserialized test patterns, the SerDes BIST architecture adds a Bitslip synchronizer circuit, illustrated in Fig. 5. Upon download of any SerDes mode configuration, the ORAs are held disabled and the TPGs are held in reset. A training pattern, stored in the programmable set/reset values of the block RAM output registers, is presented to the inputs of the I/O cells under test. The training pattern positions a single zero in a field of ones on the parallel side of the output logic block. The Bitslip synchronizer circuit monitors the Q2 parallel I/O tile output and one-shots the Bitslip control line until the zero is shifted into the Q2 position. As a result of the clock division and Bitslip latency, synchronization will be obtained in no more than $4N^2–4N$ clock cycles, where $N$ is the SerDes data width for the configuration. Each I/O cell has a dedicated Bitslip synchronizer circuit that will continue to one-shot the Bitslip control line until the training pattern is positioned with the single zero at the Q2 output, thereby identically aligning the test patterns for the comparison-based ORAs. The synchronizer is then disabled by the TPG during the BIST execution.

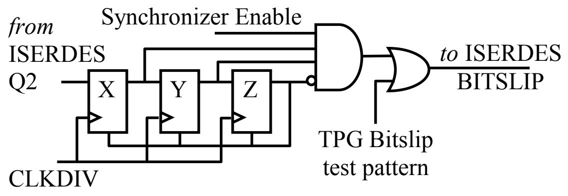

Figure 5. Bitslip synchronizer circuit

For SerDes configurations, the number of BIST clock cycles is equal to 1024 times the amount of clock division used during that configuration plus the worst case synchronization time for the data width being tested. It should also be noted that the number of BIST clock cycles is independent of the size of the array, and independent of the number of I/O cells under test.

## VII. EXPERIMENTAL RESULTS

All of the BIST configurations are automatically generated for any size and family of Virtex-5 FPGAs by a set of ANSI C programs that we have developed. Two programs are used to generate the six configurations for the I/O logic modes of operation described in Section V. Another set of two programs generates all nine of the configurations to test the I/O SerDes modes of operation described in Section VI. Our first program in each set generates a template BIST configuration in Xilinx Description Language (XDL) and then converts the template to Native Circuit Description (NCD) format using Xilinx's conversion tool, *XDL.exe*. The BIST template is routed by Xilinx's place and route software, *PAR.exe*, before conversion back to XDL format. Our second program modifies the routed XDL file to produce the various BIST configurations, and converts those files back to NCD format. The final download configuration files are created using Xilinx's bitstream generation software, *BitGen.exe*.

Table I summarizes the total size of the 15 I/O BIST configuration files, the maximum BIST clock frequency, and the total number of BIST clock cycles for all Virtex-5 LXT and SXT devices. Note that the total number of BIST clock cycles is device-independent due to concurrent testing of I/O cells by the BIST architecture. The totals shown in Table I were used to calculate the best- and worst-case total test times, which are dependent on the configuration interface. The total test time for Boundary Scan and SelectMap 32-bit parallel configuration interfaces are shown in Fig. 6 and Fig. 7, respectively. A 50 MHz BIST clock is assumed for all configurations and all devices. Readback time is for partial configuration memory readback of the ORA contents after every configuration for diagnosis of failing BIST configurations. However, when diagnosis is not required, or there are no failures, the single bit pass/fail result can be determined via the ORA iterative-OR chain. To minimize the test time and achieve maximum fault resolution, a combination of the two methods is used. First, the pass/fail status of the BIST is determined by observing the output of the ORA iterative-OR chain. If the OR chain indicates failures, partial configuration memory readback can be used to obtain the locations of the failing ORA(s) and, thereby, determine the location(s) of the failing I/O Tile(s).

TABLE I. I/O TILE BIST TOTALS (15 CONFIGURATIONS)

| Device | Total Config. Size (kB) | Max. BIST Clock Freq. | BIST CCs |
|---|---|---|---|
| LX20T | 862 | 102.8 MHz | 47112 |
| LX30T | 1482 | 89.38 MHz | 47112 |
| LX50T | 2186 | 102.4 MHz | 47112 |
| LX85T | 2726 | 73.96 MHz | 47112 |
| LX110T | 3641 | 74.40 MHz | 47112 |
| LX155T | 4181 | 66.10 MHz | 47112 |
| LX220T | 4706 | 58.75 MHz | 47112 |
| LX330T | 6985 | 56.17 MHz | 47112 |
| SX35T | 1740 | 91.19 MHz | 47112 |
| SX50T | 2511 | 75.17 MHz | 47112 |
| SX85T | 3923 | 69.59 MHz | 47112 |

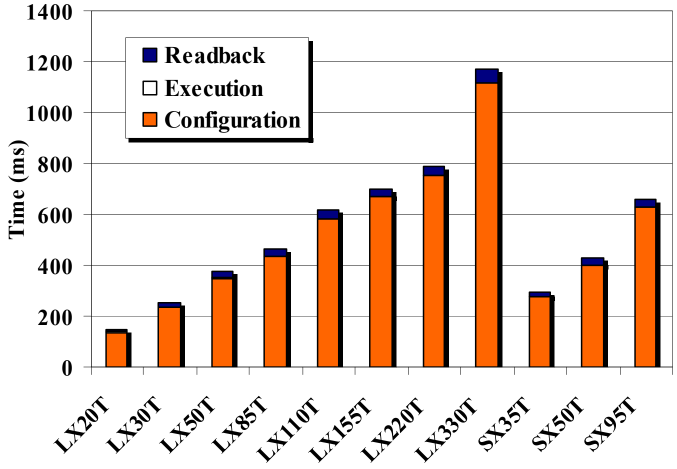

Figure 6. 50 MHz Boundary scan configuration interface test time

Figure 7. 100 MHz 32-bit parallel configuration interface test time

## VIII. BIST FOR PROGRAMMABLE I/O BUFFERS

In addition to the BIST approach presented for I/O Logic and SerDes modes of operation, we have developed a stand-alone BIST approach for the I/O buffers in FPGAs. The approach tests the I/O buffers in all bidirectional modes of operation and associated I/O standards, requiring 77 configurations for Virtex-5 FPGAs. The approach is directly applicable to device and wafer-level testing, and is applicable to in-system testing with some customization of configurations. The bidirectional buffers configured during in-system tests can be expected to have different load characteristics in the system, depending on the way they are terminated and whether they are normally an input, output, or bidirectional port during system operation. For example, we would expect the I/O buffers that are connected to large external loads to fail if they are tested at a high frequency. For in-system testing, all of the I/O buffers can be tested at a single low frequency that is guaranteed to be sufficiently slow to allow fault-free I/O buffers to pass. However, this may result in faulty I/O buffers escaping detection in the case of delay faults. Alternatively, the I/O buffers can be grouped together by loading characteristics to be tested independently and at different frequencies.

## IX. CONCLUSIONS

A BIST approach for testing the programmable logic resources of I/O cells in FPGAs was presented including the actual development for and implementation in Xilinx Virtex-5 FPGAs. Six BIST configurations were developed to test the input and output logic resources in ILOGIC and OLOGIC modes. Another nine configurations test the SerDes functionality of the I/O logic resources for all supported data widths. By testing the I/O buffers separately, the logic resources in the I/O tiles may be tested in-system in all modes of operation. The BIST configurations are package independent because they can test I/O tiles with both bonded and unbonded I/O buffers. This is important since FPGA synthesis tools sometimes use I/O logic and routing resources to implement the system function. All of these BIST configurations have been generated, downloaded, and verified on LX30T, LX50T, SX35T, and SX50T FPGAs. Due to similarities in architectures, features, and operational modes of the I/O cells in Xilinx Virtex-4 and Virtex-5 FPGAs, we have also applied the BIST approach described in this paper to Virtex-4 FPGAs where a total of five I/O Logic, nine I/O SerDes, and 76 I/O buffer BIST configurations were developed, downloaded, and verified on LX60, SX35, and FX12 FPGAs. The iterative-OR ORA provides a simple interface for BIST results retrieval that is very fast relative to partial configuration memory readback and is independent of the configuration interface. However, for fault-tolerant applications, maximal diagnostic resolution of faulty I/O tiles can still be obtained via partial configuration memory readback. The BIST configurations can detect faults in the configuration memory bits associated with I/O tile logic and routing excluding the I/O buffer. Clocking at system speeds during testing could potentially improve parametric fault coverage in the I/O delay element.

## REFERENCES

[1] *IEEE Standard Test Access Port and Boundary-Scan Architecture*, IEEE Std 1149.1-2001, 2001.

[2] C. Jia and L. Milor, "A BIST Solution for the Test of I/O Speed," *Proc. IEEE Int. Test Conf.*, pp. 1023-1030, 2003.

[3] L. Zhao, D. Walker and F. Lombardi, "IDDQ Testing of Input/Output Resources of SRAM-Based FPGAs," *Proc. Asian Test Symp.*, pp. 375-380, 1999.

[4] L-T Wang, C. Stroud, and N. Touba, *System-on-Chip Test Architectures*, Morgan Kaufmann, 2007.

[5] S. Vemula and C. Stroud, "Built-In Self-Test for Programmable I/O Buffers in FPGAs and SoCs", *Proc. IEEE Southeastern Symp. on System Theory*, pp. 534-538, 2006.

[6] *AT94K Series Field Programmable System Level Integrated Circuit*, Data Sheet, Atmel Corp., 2001. Available: www.atmel.com

[7] *Virtex-5 FPGA User Guide*, UG190 (v 4.2), Xilinx Inc., San Jose, CA, May 2008. Avaliable: www.xilinx.com

[8] D. Milton, S. Dhingra, and C. Stroud, "Embedded Processor Based Built-In Self-Test and Diagnosis of Logic and Memory Resources in FPGAs," *Proc. Int. Conf. on Embedded Systems and Applications*, pp. 87-93, 2006.

[9] L. Lerner, S. Vemula, and C. Stroud, "System-Level BIST for Programmable I/O Buffers in FPGAs and SoCs," *Proc. IEEE North Atlantic Test Workshop*, pp. 1-9, 2006.

[10] L. Lerner, "Built-In Self-Test for Input/Output Tiles in Field Programmable Gate Arrays," M.S. thesis, Dept. of Elect. and Comput. Eng., Auburn Univ., Auburn, AL, Dec. 2007.