

# BUILT-IN SELF-TEST OF PROGRAMMABLE I/O CELLS IN VIRTEX-4 FPGAS

Bradley F. Dutton, Lee W. Lerner, and Charles E. Stroud

Dept. of Electrical and Computer Engineering  
Auburn University, Alabama

**ABSTRACT:** A Built-in Self-test (BIST) approach is presented for testing the programmable I/O cells in Field Programmable Gate Arrays (FPGAs). Using this approach, three BIST architectures and a total of 78 BIST configurations were developed to test the I/O cell logic resources and I/O buffers in all modes of operation in Xilinx Virtex-4 FPGAs. Each BIST configuration is valid for both bonded and unbonded I/O buffers such that the BIST approach is package independent. Furthermore, this general BIST approach is applicable to any FPGA or programmable System-on-a-Chip (SoC) with bidirectional I/O buffers. The experimental results, capabilities, and limitations of the BIST approach are also discussed.<sup>1</sup>

## 1. 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 programmable logic resources associated with the I/O buffers that cannot be tested in this manner. These programmable logic resources typically include multiplexers and flip-flops/latches for improving system timing specifications such as set-up and hold times as well as clock-to-output delay, as illustrated in Figure 1. 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-4 FPGAs, for example, there are at least 32 multiplexers and 10 flip-flops included in the programmable logic associated with each I/O cell to support various modes of operation. The Boundary Scan INTEST feature can be used to test these programmable logic resources in an I/O cell [1]. However, the INTEST feature is supported by very few FPGA manufacturers. While there has been some work in testing I/O cells [2][3][9], previous work in Built-In Self-Test (BIST) for FPGAs has largely overlooked I/O cells and their associated logic resources. BIST approaches for testing the I/O cells of FPGAs have been proposed in [4], but have been implemented only for simple I/O cell architectures and modes of operation.

The work presented in this paper builds primarily on the prior work in [4], in which an I/O cell BIST architecture is proposed and implemented for Atmel AT40K series

FPGAs and Atmel AT94K series SoCs [5]. However, this paper offers several improvements over the previously discussed BIST approaches including the use of multiple test pattern generators (TPGs) for improved fault detection and test patterns stored in block RAMs for increased control. In addition, this paper describes the actual implementation, operation, and verification of BIST configurations developed for Virtex-4 FPGAs [6] whose I/O cells are much more complex than those found in the AT40K and AT94K devices. The configurations presented here test the full functionality of Virtex-4 I/O cells including input logic (ILOGIC), output logic (OLOGIC), Serializer/Deserializer (SERDES) operation, and programmable I/O standards. The paper begins with an overview of the background in I/O cell testing in Section 2, followed in Section 3 by an overview of the Virtex-4 I/O cells and standards. The overall BIST approach is described in Section 4 along with details of the specific BIST architectures used to test various modes of operation. We present our experimental results from actual implementation in Virtex-4 FPGAs in Section 5, followed by a discussion of the limitations of the BIST approach in Section 6, before the summary and conclusions in Section 7.

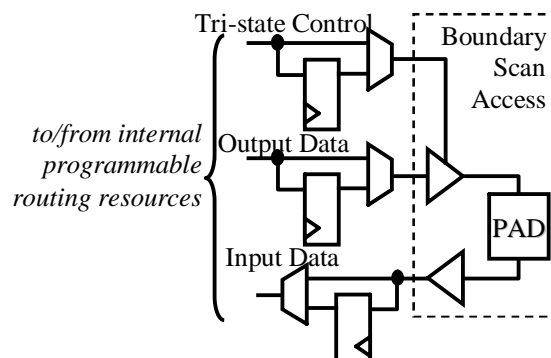


Figure 1. Simplified Programmable I/O Cell

## 2. BACKGROUND

There has been limited prior work in the area of testing I/O cells in, or applicable to, FPGAs [2][3][4][9]. In [4], a system-level BIST architecture is presented for the I/O cells of Atmel FPGAs. The overall BIST approach was similar to that used for programmable logic resources in the FPGA core [7]. The BIST architecture in [4] consists of a single TPG implemented in programmable logic blocks (PLBs) sourcing test vectors to the I/O cells under test. Only a single TPG was implemented under the assumption that internal FPGA resources have already been tested. The I/O cells under test are identically configured with bidirectional I/O buffers such that the output responses are sent back into

<sup>1</sup>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.

the FPGA internal resources. The output responses of the I/O cells are then monitored by PLBs configured as comparison-based output response analyzers (ORAs). While presenting a general architecture applicable to any FPGA or programmable SoC with an FPGA core, [4] only implemented BIST configurations for the Atmel AT94K SoC and AT40K FPGA.

The Atmel I/O cells are much less complex than those of Virtex-4 both in terms of the amount of programmable logic as well as the modes of operation and I/O standards supported, as summarized in Table 1. Reference [4] presented 23 BIST configurations to obtain 100% stuck-at gate-level fault coverage. This indicates that all gate-level faults located in the logic and routing resources of the I/O buffers can be detected. It was pointed out that the BIST approach presented in [4] can also detect major defects affecting the analog programmable features of the I/O buffers, such as pull-up, pull-down, and tristate termination. However, the BIST approach cannot detect all parametric faults such as delay or current sink and source capabilities. As has been shown in [10], the BIST approach does provide the ability to perform tests at different frequencies and produce more reliable results than the conventionally accepted boundary scan testing. It was also observed in [4] that FPGA synthesis tools will sometimes use the logic and routing resources of unbonded I/O cells to implement the system function. These resources cannot be tested by Boundary Scan EXTEST or by an external test fixture.

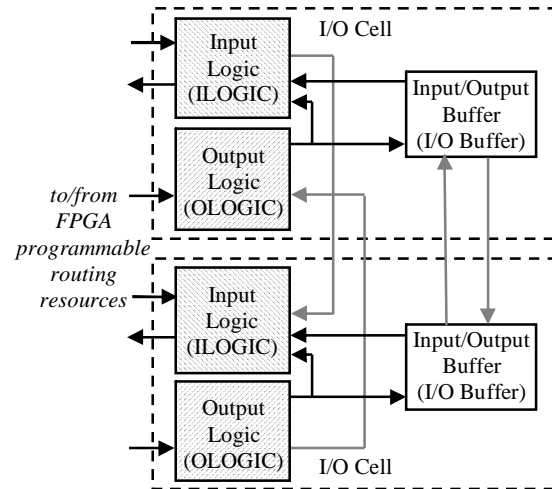
**Table 1. Comparison of I/O Cell Features**

Feature	Atmel AT94K	Xilinx Virtex-4
# Multiplexers	4	32
# Flip-flop/latches	2	10
Flip-flop/latch	flip-flop only	both
Set/reset	async reset	programmable
Clock enable	no	programmable
Pull-up/pull-down	yes	yes plus keeper
Output drive levels	3	7
I/O standards	3	69
Delay options	4	64

### 3. OVERVIEW OF VIRTEX-4 I/O CELLS

The I/O cells of Virtex-4 FPGAs consist of an output path logic block, an input path logic block, and a programmable I/O buffer. Incoming data can be inverted, delayed, or sent via a registered or unregistered path through the use of flip-flops, multiplexers, delay elements, and inverters contained in the I/O cell [6]. Output data logic can also perform operations, such as inverting or registering, on data before it reaches the I/O buffer. Aside from the output data path, output logic typically supports a tri-state control data path. When operating in bidirectional mode, both the input and out-

put logic resources may be configured. Two adjacent I/O cells form an I/O tile as illustrated in Figure 2. The two I/O buffers are connected together to support differential I/O standards that require a pair of inputs or outputs. The input clocks of the two ILOGIC or OLOGIC components in an I/O tile can be shared or not shared. The set and reset (SR and REV, respectively) signals are shared between the ILOGIC components of an I/O tile. Similarly, the SR and REV signals are shared between the OLOGIC components of a single I/O tile [6].



**Figure 2. Virtex-4 I/O Tile**

The ILOGIC and OLOGIC modules can be configured as input serial-to-parallel logic resources (ISERDES) as well as output parallel-to-serial logic resources (OSERDES), respectively [6]. The ISERDES allows for high speed serial-to-parallel conversion of data input to the FPGA from external resources. The OSERDES allows for high speed parallel-to-serial data conversion for data output from the FPGA's internal resources. The ISERDES and OSERDES components can operate in SDR mode with data widths of two, three, four, five, six, seven or eight. They can also operate in DDR mode with data widths of four, six, eight, or ten. Both ISERDES and OSERDES operate in master/slave mode when the two ISERDES/OSERDES blocks of each I/O tile are joined together via shift lines to allow for parallel-to-serial/serial-to-parallel data conversion widths greater than six. The OSERDES can be configured with a SDR tristate width of one or DDR tristate widths of two or four. The ISERDES also include a BITSLLIP input that, when invoked, shifts the data on the ISERDES outputs by one position. The BITSLLIP module allows for the reordering of parallel data for source-synchronous interfaces that include a training pattern. Most input and output signal properties are managed by the programmable I/O buffer.

The I/O buffers in Virtex-4 FPGAs support 69 programmable I/O standards, programmable pull-up, pull-

down, or keeper termination, programmable slew rate, and programmable output drive strength. The I/O standards can be categorized (as illustrated in Figure 3) as single ended termination, single ended with a differential input buffer, and complementary differential. The single ended I/O standards with differential input buffers require a reference voltage be supplied to a multi-purpose reference pin in each I/O bank. In addition, most I/O standards may be configured with Digitally Controlled Impedance (DCI). The DCI actively adjusts the output impedance or input termination impedance to match the characteristic impedance of the transmission line, eliminating the need for external series or parallel termination resistors. The DCI adjusts the impedance of the I/O by matching reference resistors connected to two multi-purpose reference pins in each I/O bank [6]. The I/O buffers associated with the  $V_{REF}$  and DCI reference pins are reserved and cannot be configured when testing DCI standards or an I/O standard that requires a reference voltage. However, these reserved pins can be tested with I/O standard modes that do not require references.

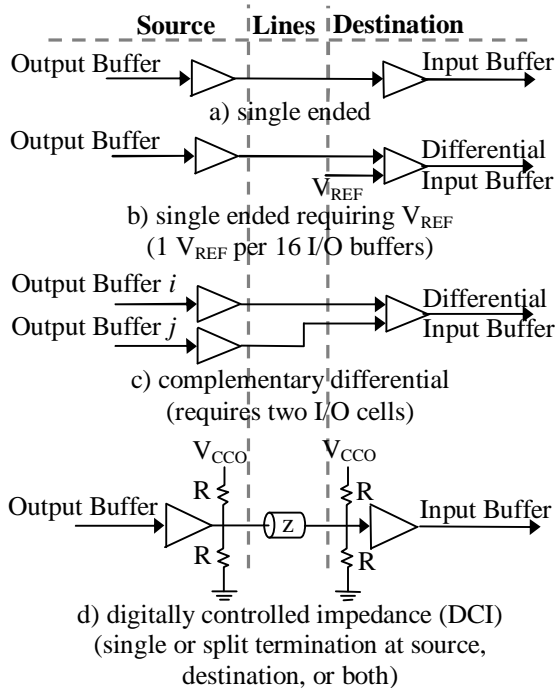


Figure 3. Virtex-4 I/O Standards

Fault modeling and fault simulation analysis (summarized in Figure 4) of the programmable logic resources in the Virtex-4 I/O cells (specifically the ILOGIC and OLOGIC modules) indicated that a minimum of seven BIST configurations are required to obtain near 100% stuck-at gate-level fault coverage. There are a couple of undetectable faults when configured in the bi-directional mode. While these faults are detectable when configured as an input buffer, they cannot be detected by the

BIST approach. These seven configurations do not consider the many I/O standards or other operation modes.

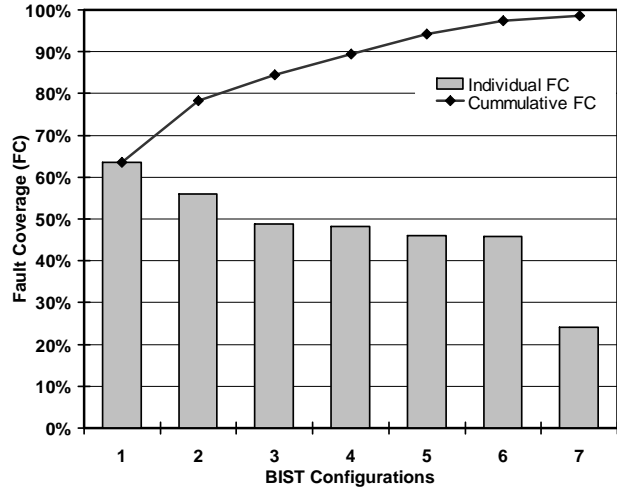


Figure 4. Fault Coverage for Virtex-4 I/O Buffers

#### 4. BIST APPROACH AND ARCHITECTURES

The basic concept of the BIST architecture, illustrated in Figure 5, is to configure I/O buffers as bidirectional buffers to allow test patterns produced by multiple identically configured TPGs to be applied to output buffers while providing a return path through an input buffer leading to the core of an FPGA. The output response of a buffer under test is then compared to responses of other identically configured I/O buffers by circular comparison-based ORAs (illustrated in Figures 5 and 6) to detect faults; ORA contents are retrieved via partial configuration memory readback at the end of the BIST sequence.

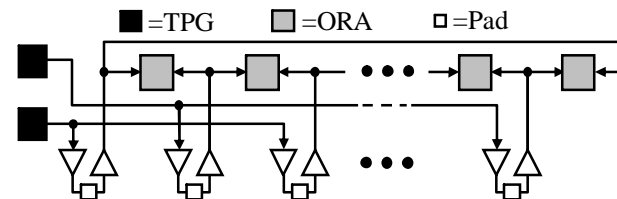


Figure 5. General I/O BIST Architecture

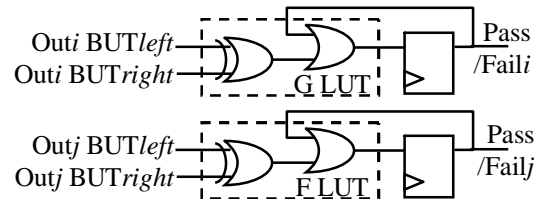


Figure 6. Two ORAs Implemented in One Slice

In Virtex-4, the TPGs are implemented as one column of digital signal processors (DSPs) configured as counters addressing two columns of 18Kbit block random access memories (BRAMs) that store deterministic as well as pseudo-random test vectors. Two TPGs are placed in every four rows of PLBs or I/O tiles in a

Virtex-4 device. The use of two TPGs per four rows of I/O tiles reduces signal loading in FPGAs with large numbers of rows. The two TPGs drive alternating rows of I/O tiles in I/O cell columns of the FPGA. The BRAMs are configured as 1K×18-bit or 512×36-bit RAMs, depending on the I/O logic configuration under test. The use of BRAMs to store test vectors allows for easy modification of the test vector set without having to modify the existing BIST architecture. The output data lines of the TPG BRAMs drive both the ILOGIC and OLOGIC components of an I/O tile.

Three different BIST architectures are used to test the OLOGIC/ILOGIC, OSERDES/ISERDES, and differential I/O buffer configurations. Eight BIST configurations (summarized in Table 2 at the end of the paper) are used to test the ILOGIC and OLOGIC routing and logic resources. This set of eight BIST configurations is also used to test various I/O buffer logic resources such as pull-up, pull-down, etc. Three configurations test the majority of resources in the ILOGIC registered data path. They test the synchronous functionality of all four flip-flops in the registered data path including initialization (INIT) and set/reset value (SRVAL) attributes. They also test the set/reset (SR), reverse (REV), clock enable (CE1), and clock (CLK) programmable active levels and functionality. Finally, they test the input flip-flops, IFF3 and IFF4, selection of the Q1 and Q2 output multiplexers, respectively. The next five configurations test the remainder of the ILOGIC resources using only the top two flip-flops of the registered data path, thus also testing the input flip-flops, IFF1 and IFF2, selection of the Q1 and Q2 output multiplexers, respectively. Configuration six tests the asynchronous latch capability of the flip-flop IFF1. Configurations 7 and eight test the full functionality of the input delay (IDELAY) module. Configuration 7 uses the IDELAY module with a fixed delay while Configuration 8 sets the IDELAY module in a variable delay mode, which also allows the TPG to increment and decrement the delay module values to test every delay value.

Eight BIST configurations are required to test the ISERDES/OSERDES in each of the eight supported data widths. Five configurations test each pair of ISERDES/OSERDES with data widths of 2, 3, 4, 5, and 6. Another three configurations test the SERDES in master/slave mode, where the ISERDES and OSERDES of two adjacent I/O cells are connected via shift lines to create data widths of 7, 8, and 10. These configure the top ISERDES/OSERDES as master and the bottom ISERDES/OSERDES as slave mode. This tests the SHIFTOUT lines of the master SERDES and the SHIFTIN lines of the slave SERDES. All programmable features, including tri-state width, SDR, DDR, clock enables, and inverters, are tested by these eight BIST configurations. Configuration 3 (data width 4) tests the

four tristate inputs in DDR mode with active high inputs. Configuration 5 (data width 6) configures the data inputs, D1-D6, and clock enable inputs as active low inputs. Configuration 8 (data width 10) tests the master and slave SERDES in DDR mode. Eight single ended I/O standards are also tested, one for each of the eight data widths. There are several important architectural differences between the SERDES architecture and the previously discussed ILOGIC/OLOGIC architecture. First, SERDES components require more I/O tile input lines to be controlled by TPGs and more output lines to be monitored by ORAs. The TPG test vector width for SERDES configurations is increased to 36 to allow for the 20 TPG outputs needed in SERDES testing as compared to only 18 TPG outputs needed for testing ILOGIC/OLOGIC modes of operation. This reduces the total test vector count to 512. Another addition to the SERDES BIST architectures is a clock divide circuit. Two clocks are required for data serialization and deserialization. A high speed clock is required for the ISERDES and OSERDES modules, while the TPGs and ORAs are synchronous to a divided clock. The divisor is determined by the SERDES data width [6].

A programmable clock division circuit was implemented for this purpose and is essentially a 4-bit counter with synchronous reset. The four counter bits are routed to a 4-input look-up table (LUT) programmed with a logic equation that activates the LUT output when a predetermined number is reached. The counter is then reset on the active clock edge. When the counter is clocked from the Boundary Scan test clock (TCK), the output of the LUT serves as the divided clock (after routing through a clock buffer). The divisor is adjusted by reprogramming the LUT equation when generating each of the eight SERDES BIST configurations. SDR configurations require that the amount of clock division equal the data width. For DDR configurations, the amount of clock division is half of the data width. Another addition to the SERDES BIST architecture stems from the need for synchronization of the serial bit stream before executing the BIST sequence. In this case, the positioning of deserialized data on the outputs of the ISERDES module can be shifted by asserting the ISERDES BITSLLIP pin. Because of the comparison based ORAs, data on the output of each ISERDES module under test must be synchronized. To ensure identical alignment of deserialized test patterns, the SERDES BIST architecture adds a BITSLLIP synchronizer circuit as illustrated in Figure 7.

Upon download of the SERDES BIST configuration, the ORAs and DSPs are disabled, and the first vector in the RAMs is a BITSLLIP training pattern. The training pattern positions five ones and a single zero on the D1 through D6 OLOGIC inputs. For BIST Configuration 5, which tests all of the OSERDES input inverters, the

BITSLIP training pattern must be inverted. This is accomplished by reprogramming the first test vector in each TPG BRAM with the inverted training sequence. The BITSLIP synchronizer circuit monitors the Q2 ISERDES output and one-shots the BITSLIP control line until the zero is shifted onto Q2 as summarized in the timing example of Table 3. As a result, synchronization will be obtained in no more than  $4(N-1)$  clock cycles where  $N$  is the number of parallel bits in the SERDES configuration (where  $N=10$  for Virtex-4). Each SERDES pair 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.

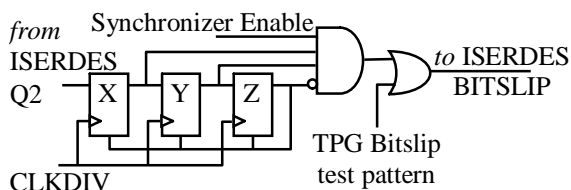


Figure 7. BITSLIP Synchronizer Circuit

A BITSLIP synchronizer enable line is tied to a RAM output that is only asserted in the training pattern. This prevents activation of the BITSLIP synchronizer circuit during the BIST sequence. With data aligned on the ISERDES outputs, the TPGs and ORAs are enabled by asserting TDI and clocking TCK. The remaining 511 test vectors are sourced to the SERDES under test. Because of the clock divide circuitry and BITSLIP synchronization process, the number of BIST clock cycles for each of the eight ISERDES/OSERDES configurations is  $512+4(N-1)$ , where  $N$  is the data width. This assumes the worst case BITSLIP synchronizer timing, in which the data on the D1 through D $N$  outputs must be shifted  $N-1$  times for the zero to reach the Q2 ISERDES output and disable the synchronizer circuit. It should also be noted that the number of BIST clock cycles is independent of the size of the array or the number of I/O cells under test.

Table 3. Synchronizer Circuit Timing Example

Clock Cycle	Init	1	2	3	4	5	6	7	8	9
ISERDES Q2	X	1	1	1	1	0	0	0	0	0
X flip-flop	0	0	1	1	1	0	1	0	0	0
Y flip-flop	0	0	0	1	1	0	0	1	0	0
Z flip-flop	0	0	0	0	1	0	0	0	1	0
BITSLIP	0	0	0	1	0	0	0	0	0	0

Finally, the remainder of the I/O standards must be tested. The eight Virtex-4 supported single ended I/O standards are tested by the eight SERDES configurations. All of the single ended standards that require a reference voltage can be tested by modifying any of the ILOGIC/OLOGIC BIST configurations with data pass-

ing directly through the OLOGIC to the I/O buffer and from the I/O buffer directly through the ILOGIC. In this modification, the  $V_{REF}$  pins are not configured and the I/O buffers under test are configured with the desired I/O standard and analog programmable features. Similarly, single ended standards supporting DCI can be tested by modifying one of the same BIST configuration such that the multi-purpose DCI reference pins in each I/O bank under test are not configured. Finally, some I/O standards support DCI and require a voltage reference simultaneously. These are tested in a similar manner by modifying the BIST configuration such that both the DCI reference pins and  $V_{REF}$  pins are not configured. By using the BIST configuration with data passing directly to and from the pad, only eight clock cycles of alternating 1s and 0s are needed during the BIST sequence for testing.

A third BIST architecture is required to test complementary differential I/O standards. BIST configurations to test complementary differential I/O standards require two data lines to be supplied to the differential input buffer. A TPG data line is supplied to the positive (non-inverting) terminal of the input buffer and its complement is applied to the negative (inverting) input of the buffer in a master I/O cell. The input data inversion (DIINV) of an OLOGIC component is used in each I/O tile to produce the complemented TPG data line that is supplied to the slave I/O cell. The slave I/O cell then sends the complemented data to the differential input buffer of the master I/O cell via its PADOUT line. This BIST architecture is different from ILOGIC/OLOGIC BIST architecture used to test single ended I/O standards. For each I/O tile, one OLOGIC component is instantiated in the slave I/O cell to complement TPG data. The ILOGIC of the slave I/O cell is not used at all as it serves no function in complementary differential operation and is fully tested during the first eight ILOGIC/OLOGIC configurations. Another difference is that only two TPG lines are required for each I/O tile under test. One TPG line is used to source test vectors to the data output pins of both I/O buffers in an I/O tile. A second TPG line is used to source test vectors to the tristate control pins of both I/O buffers in an I/O tile. One final difference is that only one ORA is required for each I/O tile as opposed to six used for the ILOGIC, OLOGIC BIST configurations. Only eight clocks of alternating 1s and 0s are used for the BIST sequence.

## 5. EXPERIMENTAL RESULTS

A total of 78 BIST configurations, summarized in Table 4 (at the end of the paper), were developed to test Virtex-4 I/O tiles in all modes of operation. These configurations also test each I/O standard that supports bidirectional I/O buffers. Each BIST configuration was executed and verified on an SX-35 and/or an LX60 FPGA in an FF668 BGA package. When testing I/O

standards that require a reference voltage, the recommended reference voltage was supplied to one multi-purpose  $V_{REF}$  pin in each bank of I/O cells under test. When testing an I/O standard utilizing Xilinx DCI, two  $51\Omega$  reference resistors were connected to the  $V_{RP}$  and  $V_{RN}$  multi-purpose reference pins and to ground and  $V_{CO}$ , respectively. Probing of internal nets facilitated verification that TPGs and ORAs operate as expected. By connecting light emitting diodes (LEDs) to the pads of I/O cells under test, test patterns can be observed at the output buffer. The RAM-based TPGs can be programmed with periodic test patterns for observation with an oscilloscope. Finally, each configuration was shown to produce passing ORA results by configuration memory read back.

There are a total of nine I/O standards that do not support bi-directional modes of operation in the I/O buffer; these are shaded in gray in Table 4. As a result, these nine I/O standards cannot be tested with our current BIST architecture since it requires bi-directional modes of operation. It is possible, however, to create BIST configurations to test these nine standards if (and only if) external connections are made on the test fixture (for manufacturing testing) or on the printed circuit board (for system-level testing) to provide return connections between specified sets of output buffers and input buffers on the device. Then a total of 18 BIST configurations could be developed to test each pair of connected I/O buffers in the nine I/O standards in both directions of signal flow. At present, however, our BIST approach can test 60 of the 69 different I/O standards supported in Virtex-4 FPGAs.

Table 4 lists 70 BIST configurations to test every bi-directional I/O standard. Another eight BIST configurations are used to test the I/O cells in the SERDES modes of operation. These use the same single ended I/O standards as the first eight configurations in listed in Table 4. The complete set of 78 BIST configurations is summarized in Table 5 in an order of application that minimizes the total memory size of the partial reconfiguration download files. It can be observed in Table 5 that the first eight I/O standards are repeated for the SERDES and the ILOGIC/OLOGIC BIST configurations. Therefore, we first download the SERDES BIST configurations with a compressed full download for the first of the eight configurations followed by seven partial reconfigurations. In Table 5, compressed full configurations are highlighted in the first column in gray and the configuration number is suffixed with an 'F' while partial reconfigurations are suffixed with a 'P'. Next, we download the single ended I/O standard again to test the ILOGIC and OLOGIC in single data rate (SDR) and double data rate (DDR) modes of operation. There are more ORAs and more TPG-to-I/O connections in the SERDES BIST configurations than in the

remaining BIST configurations such that we must reconfigure the FPGA with a compressed full download. This is followed by seven partial downloads for the remaining single ended BIST configurations. At this point, all programmable logic resources in the ILOGIC, OLOGIC, and PAD (with the exception of the differential input buffer and associated dedicated routing) have been tested and only the various I/O standards are left to be tested in the remaining 62 BIST configurations.

We begin by testing seven single ended mode I/O standards that do not require a reference voltage, all of which are partial reconfigurations since the same BIST architecture is used as that for the programmable logic resources in the ILOGIC and OLOGIC. The next set of 15 BIST configurations test single ended mode I/O standards that require a reference voltage. In these configurations, the required reference voltage is supplied via one general purpose reference pin per every 16 I/O buffers (4 pins per bank of 64) from an external voltage source. As a result, this particular pad (the fifth pad up from the bottom in each set of eight I/O buffers in a given bank) cannot be connected to the BIST circuitry. This requires an additional modification to the BIST architecture and, hence, another compressed full download at the beginning of this set of 14 BIST configurations.

The next set of 20 BIST configurations test digitally controlled impedance (DCI) I/O standards which require an external reference resistor be applied to two general purpose reference pads in the tenth row from the bottom of each bank of 32 I/O buffers. As with the general purpose voltage reference pins, these two pads cannot be connected to the BIST circuitry. This requires an additional modification to the BIST architecture in order to test the DCI I/O standards and, similarly, another compressed full download at the beginning of this set of BIST configurations.

The final four sets of BIST configurations test the complementary differential I/O standards with the master and slave orientations grouped such that a compressed full download is required at the beginning of the set of BIST configurations followed by partial reconfigurations for the remaining BIST configurations in the particular set. This is due to the BIST architecture for complementary differential I/O standard testing with only one of two I/O buffer operating as a differential input buffer. In addition, we must reconnect the reference voltage and DCI buffers since they are not used in this mode of operation. These I/O buffers are configured and tested in the complementary differential configurations. The first set of six BIST configurations tests the normal complementary differential I/O standards while the final set of four BIST configurations tests the DCI modes of complementary differential I/O standards.

Hence the DCI pins (but not the reference voltage pins) must not be configured and tested during the second pair of sets of four BIST configurations.

All of the 78 BIST configurations are automatically generated by a set of six programs we have developed. The process, as illustrated in Figure 8, consists of generating the BIST configuration template in XDL format by our programs, converting the XDL to NCD format for routing in FPGA Editor. The routed NCD file is converted back to XDL for modification by our programs to generate the various BIST configurations. The final XDL files are converted to NCD format from which the actual download bit file are generated. There are three BIST template generation programs and three BIST configuration modification programs as follows:

1. *V4iobistios.exe* generates the XDL template file for SERDES BIST configurations.
2. *V4iobist.exe* generates the XDL template file for ILOGIC, OLOGIC, and I/O buffer BIST configurations. This program has the option to remove the reference voltage and/or the reference resistance pads from the BIST configuration to testing those sets of I/O standards that require the reference pads be connected to external sources.
3. *V4iobistd.exe* generates the XDL template file for complementary differential I/O standard BIST configurations. This program also has the option of removing the reference voltage and/or reference resistor pins.
4. *V4iobmodios.exe* modifies the XDL template file for SERDES BIST configurations.
5. *V4iobmod.exe* modifies the template file for ILOGIC and OLOGIC BIST configurations.
6. *V4iobrmmod.exe* modifies the XDL template files for I/O standard BIST configurations.

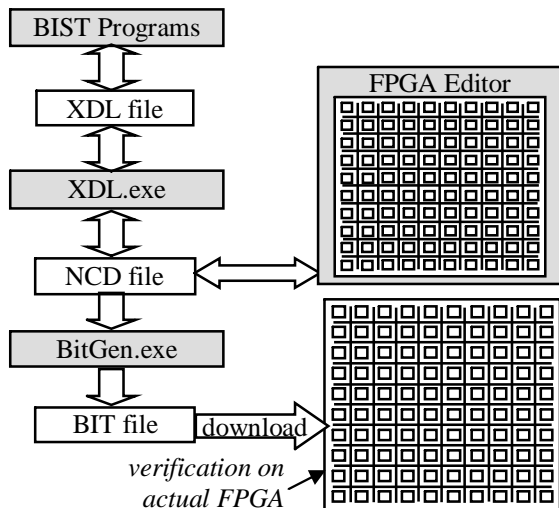


Figure 8. BIST Configuration Generation Process

While the BIST configurations can be generated individually, we use batch files to generate all of the BIST configurations for a given device and usually run the batch file overnight to generate the configurations. We have run batch files for every Virtex-4 device and for every package option for each device. This took an overnight run to produce the NCD files for all BIST configurations, all devices, and all package options; however, we did not generate the download bitstream files during that batch file run. We have, however, generated the download bitstream files for all BIST configurations for SX35 and LX60 devices in FF668 packages. The actual execution time was approximately 3.5 hours per device for generation of all 78 BIST configurations as well as their full and partial reconfiguration download files. In addition, all of the BIST configurations have been downloaded and verified on SX35 and LX60 FPGAs.

The I/O BIST file sizes for an LX15 are summarized in Table 6 (at the end of the paper) in terms of the typical sizes for full, compressed, and partial reconfiguration download files. Note that using partial reconfiguration files for the transition between groups of BIST configurations reduces the size of the file by about 50%, but there are issues related to ORA initialization and BIST results read back that must be considered. The 70 partial reconfiguration files only account for about 4,067 Kbit, which corresponds to only about 10%-17% of the total download time when using compressed full download or partial files, respectively, for transition between groups of BIST configurations. Note that only 27,830 BIST clock cycles are needed for the entire set of 78 BIST configurations, less than 0.2% of total time.

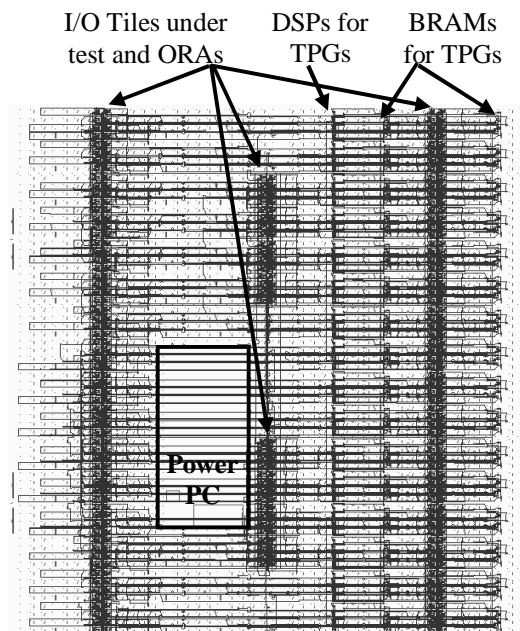


Figure 9. BIST Implementation in Virtex-4 FX20

The actual implementation of the I/O BIST architecture as generated by our programs is illustrated in Figure 9 for a Virtex-4 FX20 FPGA which includes a single PowerPC module. In this device, the I/O cells under test reside eight columns from the left and right edges as well as along portions of the middle column of the array. In LX and SX series Virtex-4 FPGAs, the full columns of I/O cells lie along the left and right edges of the array. This area around the I/O cells also includes the comparison-based ORAs and BITSLLIP synchronizers, which reside in the CLB columns adjacent to the I/O cells. The column of DSPs are used as counters to address two columns of BRAMs which store the pseudo-random and deterministic test patterns used to test the I/O cells in their various modes of operation.

## 6. LIMITATIONS

The bidirectional buffers configured to test the I/O cells during BIST will have different load characteristics depending on the way they are terminated and whether they are normally an input, output, or bi-directional port during system operation. As observed in [4], the BIST is sensitive to external loading conditions. Therefore, we would expect the I/O cells that are connected to large external loads to fail if they are tested at a high frequency. All of the I/O buffers can be tested at a single considerably slow frequency that is guaranteed to be sufficiently slow to allow fault-free I/O cells to pass. However, this may result in faulty I/O cells escaping detection in the case of delay faults. On the other hand, the I/O buffers can be grouped together by loading characteristics to be tested at different frequencies.

## 7. CONCLUSIONS

A BIST approach for testing the programmable I/O cells in FPGAs was presented including the actual development for and implementation in Xilinx Virtex-4 FPGAs. Eight BIST configurations were developed to test the ILOGIC and OLOGIC logic resources. Another eight configurations test the ISERDES and OSERDES parallel-to-serial/serial-to-parallel functionality for all supported data widths. Seventy BIST configurations are required to test each bidirectional Virtex-4 supported I/O standard including single ended, single ended requiring a reference voltage, complementary differential, and standards supporting DCI. Eight of the single ended I/O standards can be tested with the eight ISERDES/OSERDES BIST configurations. Thus, a total of 78 BIST configurations test the full functionality of Virtex-4 I/O tiles. The BIST configurations are package independent because they can test I/O tiles with bonded or unbonded I/O buffers. Testing both bonded and unbonded I/O buffers is important since FPGA synthesis tools sometimes use I/O logic and routing resources to implement the system function.

The BIST configurations can be used in manufacturing and in-system testing. However, FPGAs can be tested in-system only if the I/O of all connecting devices can be tri-stated during testing. The BIST configurations can detect faults in the configuration memory bits associated with I/O tile logic and routing. These configurations can also detect major defects in the analog programmable features of the I/O buffer. However, the BIST configurations cannot detect all minor parametric faults such as  $V_{IH}$ ,  $V_{OH}$ ,  $V_{IL}$ ,  $V_{OL}$ , slew rate, or drive strength. Clocking at system speeds during testing could potentially improve parametric fault coverage. Due to a number of similarities in architectures, features, and operational modes of the I/O cells in Xilinx Virtex-4 and Virtex-5 FPGAs, the automatic BIST configuration generation programs can easily be modified to support generation of BIST configurations to test the I/O cells in Virtex-5 FPGAs.

## REFERENCES

- [1] C. Maunder and R. Tulloss, *Test Access Port and Boundary Scan Architecture*, IEEE Computer Society Press, 1990.
- [2] C. Jia and L. Milor, "A BIST Solution for the Test of I/O Speed," *Proc. IEEE International 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] 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.
- [5] Atmel, "AT94K Series Field Programmable System Level Integrated Circuit," Data Sheet, Atmel Corp., 2001 (available at [www.atmel.com](http://www.atmel.com)).
- [6] Xilinx, "Virtex-4 User Guide," UG070 (v 2.0), Xilinx, Inc., 2007 (available at [www.xilinx.com](http://www.xilinx.com)).
- [7] D. Milton, S. Dhingra, and C. Stroud, "Embedded Processor Based Built-In Self-Test and Diagnosis of Logic and Memory Resources in FPGAs," *Proc. International Conf. on Embedded Systems and Applications*, pp. 87-93, 2006.
- [8] L-T Wang, C. Stroud, and N. Touba, *System-on-Chip Test Architectures*, Morgan Kaufmann, 2007.
- [9] M. Xiaojun and T. Jiarong, "Boundary-scan Test Circuit Designed for FPGA", *Proc. IEEE International Application Specific Integrated Circuits Conf.*, pp. 1190-1193, 2003.
- [10] L. Lerner, S. Vemula, and C. Stroud, "System-Level BIST for Programmable I/O Cells in FPGAs and SoCs," *Proc. IEEE North Atlantic Test Workshop*, pp. 1-9, 2006.

**Table 2. Virtex-4 ILOGIC/OLOGIC BIST Configurations**

Resource	Cfg#1	Cfg#2	Cfg#3	Cfg#4	Cfg#5	Cfg#6	Cfg#7	Cfg#8
PULL	Keep.	Down	Pullup	Keep.	Down	Pullup	Keep.	Keep.
GTSATTR	Off	Disable	Disable	Off	Disable	Disable	Off	Off
SLEW	Slow	Fast	Slow	Off	Slow	Off	Fast	Slow
DIFFTERM	NA	NA	NA	NA	NA	NA	NA	NA
DRIVE	24	4	6	Off	2	Off	8	Off
DRIVE0MA	Off	Drive0	Drive0	Off	Drive0	Drive0	Off	Off
IOATTR	LVTTTL	LVC MOS33	LVC MOS25	PCI66-3	LVC MOS15	PCI33-3	LVC MOS18	PCI-X

**Table 4. Virtex-4 I/O Standards and BIST Configurations**

I/O Standard	Type	V <sub>REF</sub>	Cfg	DCI I/O Standards	Type	V <sub>REF</sub>	Cfg
LVTTTL	SE	N/R	1	LVDCI_33	SE	N/R	36
LVC MOS33	SE	N/R	2	LVDCI_25	SE	N/R	37
LVC MOS25	SE	N/R	3	LVDCI_18	SE	N/R	38
LVC MOS18	SE	N/R	4	LVDCI_15	SE	N/R	39
LVC MOS15	SE	N/R	5	LVDCI_DV2_25	SE	N/R	40
PCI_33_3	SE	N/R	6	LVDCI_DV2_18	SE	N/R	41
PCI_66_3	SE	N/R	7	LVDCI_DV2_15	SE	N/R	42
PCIX	SE	N/R	8	HSTL_II_T_DCI	SER	0.75	43
HSTL_I	SER	0.75	9	HSTL_II_T_DCI_18	SER	0.9	44
HSTL_II	SER	0.75	10	SSTL2_II_T_DCI	SER	1.25	45
HSTL_III	SER	0.9	11	SSTL18_II_T_DCI	SER	0.9	46
HSTL_IV	SER	0.9	12	DIFF_SSTL2_II_DCI(M)	CD	N/R	47
HSTL_I_18	SER	0.9	13	DIFF_SSTL2_II_DCI(S)	CD	N/R	48
HSTL_II_18	SER	0.9	14	DIFF_SSTL18_II_DCI(M)	CD	N/R	49
HSTL_III_18	SER	1.1	15	DIFF_SSTL18_II_DCI(S)	CD	N/R	50
HSTL_IV_18	SER	1.1	16	DIFF_HSTL_II_DCI(M)	CD	N/R	51
HSTL_I_12	SER	0.6	17	DIFF_HSTL_II_DCI(S)	CD	N/R	52
GTL	SER	0.8	18	DIFF_HSTL_II_DCI_18(M)	CD	N/R	53
GTL_P	SER	1	19	DIFF_HSTL_II_DCI_18(S)	CD	N/R	54
SSTL2_I	SER	1.25	20	GTL_DCI	SER	0.8	55
SSTL2_II	SER	1.25	21	GTL_P_DCI	SER	1	56
SSTL18_I	SER	0.9	22	HSTL_I_DCI	SER	0.75	57
SSTL18_II	SER	0.9	23	HSTL_II_DCI	SER	0.75	58
LVPECL_25(M)	CD	N/R	24	HSTL_III_DCI	SER	0.9	59
LVPECL_25(S)	CD	N/R	25	HSTL_IV_DCI	SER	0.9	60
LVDS_25	CD	N/R	NA	HSTL_I_DCI_18	SER	0.9	61
LVDSEXT_25	CD	N/R	NA	HSTL_II_DCI_18	SER	0.9	62
BLVDS_25(M)	CD	N/R	26	HSTL_III_DCI_18	SER	1.1	63
BLVDS_25(S)	CD	N/R	27	HSTL_IV_DCI_18	SER	1.1	64
ULVDS_25	CD	N/R	NA	SSTL2_I_DCI	SER	1.25	NA
LDT_25	CD	N/R	NA	SSTL2_II_DCI	SER	1.25	65
RSDS_25	CD	N/R	NA	SSTL18_I_DCI	SER	0.9	NA
DIFF_SSTL2_II(M)	CD	N/R	28	SSTL18_II_DCI	SER	0.9	66
DIFF_SSTL2_II(S)	CD	N/R	29	LVDS_25_DCI	CD	N/R	NA
DIFF_SSTL18_II(M)	CD	N/R	30	LVDSEXT_25_DCI	CD	N/R	NA
DIFF_SSTL18_II(S)	CD	N/R	31	HSLVDCI_33	SER	V <sub>cc0</sub> /2	67
DIFF_HSTL_II(M)	CD	N/R	32	HSLVDCI_25	SER	V <sub>cc0</sub> /2	68
DIFF_HSTL_II(S)	CD	N/R	33	HSLVDCI_18	SER	V <sub>cc0</sub> /2	69
DIFF_HSTL_II_18(M)	CD	N/R	34	HSLVDCI_15	SER	V <sub>cc0</sub> /2	70
DIFF_HSTL_II_18(S)	CD	N/R	35	<b>Total Required Configurations</b>			<b>70</b>

SE = Single Ended      SER = Single Ended Requiring V<sub>REF</sub>  
 CD = Complementary Differential

**Table 5. Virtex-4 I/O BIST Configuration Order**

Cfg	I/O Standard	Target	Cfg	I/O Standard	Target
1F	LVTTTL	SERDES	40P	HSTL II T DCI 18	SER Standard
2P	LVC MOS33	SERDES	41P	SSTL2 II T DCI	SER Standard
3P	LVC MOS25	SERDES	42P	SSTL18 II T DCI	SER Standard
4P	LVC MOS18	SERDES	43P	GTL DCI	SER Standard
5P	LVC MOS15	SERDES	44P	GTLP DCI	SER Standard
6P	PCI 33_3	SERDES	45P	HSTL I DCI	SER Standard
7P	PCI 66_3	SERDES	46P	HSTL II DCI	SER Standard
8P	PCIX	SERDES	47P	HSTL III DCI	SER Standard
9F	LVTTTL	I/O LOGIC	48P	HSTL IV DCI	SER Standard
10P	LVC MOS33	I/O LOGIC	49P	HSTL I DCI 18	SER Standard
11P	LVC MOS25	I/O LOGIC	50P	HSTL II DCI 18	SER Standard
12P	LVC MOS18	I/O LOGIC	51P	HSTL III DCI 18	SER Standard
13P	LVC MOS15	I/O LOGIC	52P	HSTL IV DCI 18	SER Standard
14P	PCI 33_3	I/O LOGIC	53P	SSTL2 II DCI	SER Standard
15P	PCI 66_3	I/O LOGIC	54P	SSTL18 II DCI	SER Standard
16P	PCIX	I/O LOGIC	55P	HSLVDCI 33	SER Standard
17P	LVDCI 33	SE Standard	56P	HSLVDCI 25	SER Standard
18P	LVDCI 25	SE Standard	57P	HSLVDCI 18	SER Standard
19P	LVDCI 18	SE Standard	58P	HSLVDCI 15	SER Standard
20P	LVDCI 15	SE Standard	59F	LVPECL 25 (M)	CD Standard
21P	LVDCI DV2 25	SE Standard	60P	BLVDS 25 (M)	CD Standard
22P	LVDCI DV2 18	SE Standard	61P	DIFF SSTL2 II (M)	CD Standard
23P	LVDCI DV2 15	SE Standard	62P	DIFF SSTL18 II (M)	CD Standard
24F	GTL	SER Standard	63P	DIFF HSTL II (M)	CD Standard
25P	GTLP	SER Standard	64P	DIFF HSTL II 18 (M)	CD Standard
26P	SSTL2 I	SER Standard	65F	LVPECL 25 (S)	CD Standard
27P	SSTL2 II	SER Standard	66P	BLVDS 25 (S)	CD Standard
28P	SSTL18 I	SER Standard	67P	DIFF SSTL2 II (S)	CD Standard
29P	SSTL18 II	SER Standard	68P	DIFF SSTL18 II (S)	CD Standard
30P	HSTL I	SER Standard	69P	DIFF HSTL II (S)	CD Standard
31P	HSTL II	SER Standard	70P	DIFF HSTL II 18 (S)	CD Standard
32P	HSTL III	SER Standard	71F	DIFF SSTL2 II DCI (M)	CD Standard
33P	HSTL IV	SER Standard	72P	DIFF SSTL18 II DCI (M)	CD Standard
34P	HSTL I 18	SER Standard	73P	DIFF HSTL II DCI (M)	CD Standard
35P	HSTL II 18	SER Standard	74P	DIFF HSTL II DCI 18 (M)	CD Standard
36P	HSTL III 18	SER Standard	75F	DIFF SSTL2 II DCI (S)	CD Standard
37P	HSTL IV 18	SER Standard	76P	DIFF SSTL18 II DCI (S)	CD Standard
38P	HSTL I 12	SER Standard	77P	DIFF HSTL II DCI (S)	CD Standard
39F	HSTL II T DCI	SER Standard	78P	DIFF HSTL II DCI 18 (S)	CD Standard
SE = Single Ended		SER = Single Ended Requiring V <sub>REF</sub>			
		CD = Complementary Differential			

**Table 6. SX35 I/O BIST Configuration File Sizes**

Type	Bits per file	Number	Total bits
Full download	4,765,568	78	371,714,304
Compressed full download	3,376,096	8	27,008,768
Partial transition (opposed to compressed full)	949,230	7	6,644,608
Partial reconfiguration files	93,876	70	6,571,306
Total download for test with compressed full download		78	33,580,092
Total download for test with partial transition download		78	16,592,010
Total BIST clock cycles		78	27,830