A Test Procedure for Boundary Scan Circuitry in PLDs and FPGAs

Bashar Al-Khalifa
Department of Medical Instrumentation Engineering, Technical College, Iraq

Abstract: *A test procedure for testing mainly the boundary scan cells, and testing partially the test access port controller in programmable logic devices, and field programmable gate array devices, is suggested. The test procedure involves; the configuration of programmable logic devices or field programmable gate array device, the application of test vectors, and finally the verification of the response. These steps are repeated with two different configurations of the device under test, to ensure high faults coverage. Both the configuration, and the application of test vectors, is performed through the joint test access group port of the device under test. The parts of the boundary scan circuit and the type of faults which are covered are mentioned.*

Keywords: *Boundary scan circuit test, programmable logic devices, and test procedure*.

Received July 1, 2008; accepted September 25, 2008

1. Introduction

Since 1990, when Boundary Scan became a standard (IEEE standard 1149.1 known as JTAG), manufacturers of Application Specific Integrated Circuits (ASICs), PLDs, and FPGAs have introduced the Boundary Scan Circuitry (BSC) in their manufactured devices. The introduction of BSC has replaced the in-circuit test and its physical contacts (bed-of-nails), since it became impossible to use the bed-of-nails in testing the multilayer Printed Circuit Board (PCB) mounted with fine-pitch ICs. The BSC enables test engineers to test logic systems in device, board, and system levels. Thus a new test technology emerged known as Boundary Scan Test (BST). The BSC is used for testing both on-chip, and off-chip logic circuits in PLDs and FPGAs. Thus testing BSC in PLDs and FPGAs is an important task. A typical boundary-scan-based PLD or FPGA architecture is shown in as shown in Figure 1. The JTAG architecture comprises; the TAP controller, the instruction register, and a set of data registers (boundary scan register, bypass register, identification register, and other manufacturer private registers such as configuration, and test registers). Many authors have proposed different methods for testing the core logic of PLD, and FPGA devices [4, 5, 6], while other authors have introduced a tool which is capable of verifying an 1149.1 test logic implementation, and its compliance to the IEEE 1149.1 standard in ASIC devices [3]. In this paper a test procedure for testing the BSC of PLD and FPGA devices is proposed. In section 2 some background knowledge on the JTAG architecture, basic instructions, and BST operations are reviewed. Section 3 explains the test procedure, while section 4 indicates the tested parts, and the type of faults covered in the proposed procedure. Finally, section 5 includes conclusions, and possible future work.

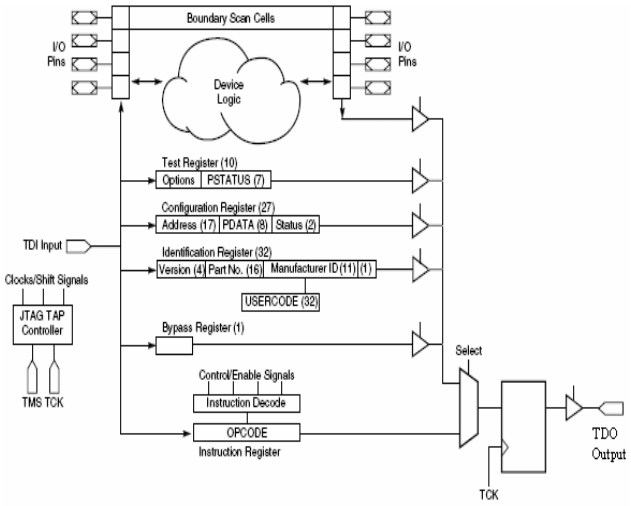

Figure 1. JTAG Architecture.

2. Background

2.1. JTAG Architecture

The JTAG architecture comprises as shown in Figure 1, the TAP controller, the instruction register, and a set of at least two data registers the boundary scan register, and the bypass register. In addition, other optional registers may be included, the identification register, and other private (manufacturer) registers such as configuration, and test registers [8]. Each of these data registers (except the boundary scan register) has a fixed size, and can be selected by one or more instructions. The size of the boundary scan register depends on the number of pins in the device, and the

way of implementing this register. The TAP controller is a 16-state state machine as shown in Figure 2, which is clocked on the rising edge of its CLocK input (CLK). The change from one state to another depends on the Test Mode Select (TMS) signal. The other two JTAG inputs are Test Data Input (TDI), and Test Data Output (TDO). The TAP controller can select any of these serial registers to be connected between TDI input pin, and TDO output pin according to the executed instructions. The test logic reset state can be forced either after power-up the device, or by applying at least five clock pulses on CLK pin, while keeping the TMS pin high. Entering this state will reset all test logic, and normal device operation can be performed. In order to be able to shift data in any data register, an instruction code has to be shifted in the instruction register. This can be achieved by a transition from test logic reset state to the Shift-IR state (by clocking TMS with the pattern 01100 as shown in Figure 2. At the Shift-IR state, the Instruction register is selected to be between TDI, and TDO pins. Therefore an instruction code can be shifted in the instruction register (through TDI pin), while the TMS is kept low. Once the instruction is loaded into the instruction register, a transition is required to the Update-IR state, so that the instruction becomes effective, and the corresponding data register is selected to be between TDI and TDO pins. Shifting data into the selected data register requires the transition to the Shift-DR state.

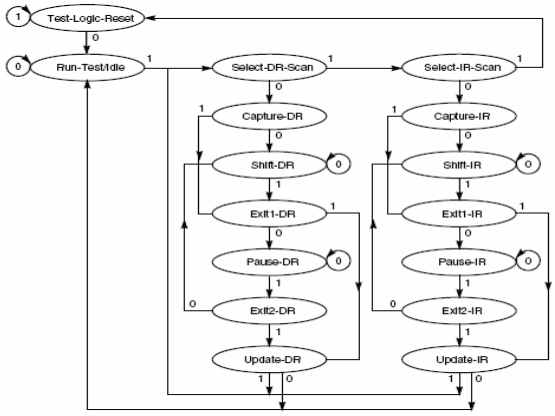

Figure 2. Boundary scan cell in a typical input output block.

2.2. The Boundary Scan Register

The boundary scan register consists of a large number of serially connected scan cells. Each scan cell consists of a 3-bit elements, and it is connected to the internal device logic from one side, and a device pin from the other side. Figure 3 shows a boundary scan cell in a typical Input Output Block (IOB) of an FPGA [8]. Since each IOB can be configured as independently controlled bidirectional pin, three data registers are provided per IOB; the first register is for input data, the second is for output data, and the third is for 3-state control.

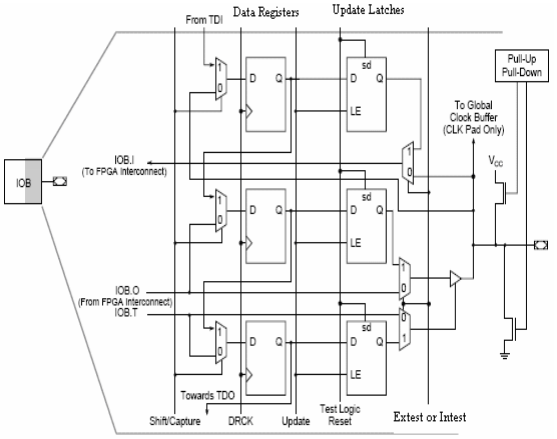

Figure 3. Boundary scan cell in a typical input-output block.

Each data register is connected to an update latch as shown in Figure 4. Test data coming from TDI, may be loaded to the first data register through a multiplexer, while the output of the first data register is also connected to the next data register through a multiplexer, and so on towards the TDO pin. This connection permits loading the data register with test data serially from TDI pin, while at the same time data can be read serially through TDO pin (shift operation). The boundary scan cell has other parallel connections with the device pin from one side, and with the device internal logic from the other side. The first data register can be loaded (through a multiplexer) from the device pin. The second data register can be loaded (through a multiplexer) from the device internal logic (IOB.O). The third data register can be loaded (through a multiplexer) from the 3-state control line of the device internal logic (IOB.T). These connections enable a capture operation, which can load the data register with the signals on the device pin, the internal logic outputs, and the 3-state control line of the IOB. The output of the first update latch is connected to the input of the device internal logic (IOB.I). The output of the second update latch is connected to the data input of the output buffer, while the third update latch is connected to the 3-state control of the output buffer. These connections enables an update operation, in which test data at the data register can be loaded to the inputs of the device internal logic and to the device output pin.

2.3. JTAG Instructions

The JTAG public instructions are SAMPLE, PRELOAD, BYPASS, EXTEST, INTEST, RUNBIST, and HIGHZ. In this section only the EXTEST, and INTEST instructions will be explained in detail, since the main objective of this proposal is to test the boundary scan circuit. The HIGHZ instruction places the bypass register between TDI and TDO pins, while tri-stating all of IO pins. Both EXTEST and INTEST instructions selects the boundary scan register when they are executed. As was mandated in IEEE Std. 1149.1 [1, 2, 7] the EXTEST is used for testing the

connections between devices on PCB, and can not test the device internal logic (the on-chip logic is forced in a hold or reset state). While the INTEST instruction is used to test the device internal logic, and can not test the device external connections (the input pins are isolated from the inputs of the capture registers). However some manufacturers [8], which their devices do not support INTEST instruction, they have their EXTEST instruction incorporates INTEST-like functionality. When any instruction (including INTEST, and EXTEST) is selected, three phases of operation are feasible, the capture, the shift, and the update. The capture operation is active when the TAP controller is at the "Capture-DR" state as shown in Figure 2. The shift operation is active at the "Shift-DR" state. The update operation is active at the "Update-DR" state. When the INTEST instruction is selected (Update-IR state is entered), a transition to the "Capture-DR" state can be forced. At this state the signals on the device pin, IOB.O, and IOB.T are loaded in the boundary scan register cells. Then a transition to "Shift-DR" state can be forced. At this state, test data can be serially shifted into the boundary scan register by keeping TMS low, and clocking test data through TDI pin. The next operation could be the update, which can be performed by a transition to the "Update-DR" state. At this state the signals loaded in the boundary scan register cells (capture registers) are clocked to the update latches, which in turn apply these signals to the internal logic input IOB.I, the data input of the output buffer, and the 3-state control line of the output buffer as shown in Figure 3.

3. The Test Procedure

The proposed test procedure is intended for one PLD or FPGA device. If there are other devices connected to the device under test, all other devices pins should be tri-stated (by selecting HIGHZ instruction). The test procedure involves the following steps:

- The PLD or FPGA under test is configured so that half of its IO pins are inputs and the other IO pins are outputs, and a direct (wire) connection is designed to be between each pair of input and output pins.
- The INTEST instruction is then selected (as explained in section 2.1).
- Two test vectors {101, 110} are serially applied to the data registers for both input cells and output cells during Shift-DR state as shown in Table 1. The first test vector is applied to the input cells, while the second one is applied to the output cells. Notice that the first digit from the left (in the test vector) is designated for the data register which is connected to the IOB.T line. The middle digit is designated for the register which is connected to the IOB.O line. The last digit is designated for the register connected to the IOB.I line. In order to apply the test vectors to all the cells (IOBs), a bit stream, consists of a repetition of these two test vectors, has to be constructed. Then this bit stream is applied serially through TDI pin.
- After the application of each test vector, The response data from the pin in both input, and output cells, and the outputs of the internal logic (IOB.T, and IOB.O lines) in an output cell, are captured, then shifted out and verified. The response data (for functional circuitry) should be as shown in Table 1 at the Capture-DR state.
- The device under test is reconfigured, so that each input pin becomes an output, and each output pin becomes an input, and a direct (wire) connection between each pair of input and output pins.. The procedure in steps (2, 3, and 4) is repeated, using the same test vectors.

Table 1. Logic states and operations performed during the application of two test vectors, and after INTEST has been selected.

TAP State	Logic States in Input Cells						Logic State in Output Cells						Operations Performed
	Data Registers			Update Latches			Data Registers			Update Latches			
	T	O	I	T	O	I	T	O	I	T	O	I	
Capture-DR	X	X	X	X	X	X	X	X	X	X	X	X	Loading data registers with undetermined states
Shift-DR	1	0	1	X	X	X	1	0	1	X	X	X	Shifting in 1st test vector
Update-DR	1	0	1	1	0	1	1	0	1	1	0	1	Applying test data to internal logic input and output buffer
Capture-DR	X	X	0	1	0	1	1	1	0	1	0	1	Loading data registers with the specified states
Shift-DR	1	1	0	1	0	1	1	1	0	1	0	1	Shifting in 2nd test vector, and shifting out the response to be verified
Update-DR	1	1	0	1	1	0	1	1	0	1	1	0	Applying test data to internal logic input and output buffer
Capture-DR	X	X	1	1	1	0	1	0	1	1	1	0	Loading data registers with the specified states
Shift-DR													Shifting out the response to be verified
T = the output of data register or update latch which drives the 3-state line of the output buffer.													
O = the output of data register or update latch which drives the input line of the output buffer.													
I = the output of data register or update latch which drives the input of the internal logic.													
X = undetermined logic state.													

4. Test Vectors and Detected Faults

The two test vectors are chosen so that they can detect most of the stuck at faults in the BSC. The first bit of the test vector, which is dedicated for the 3-state control line of the output buffer, is chosen to be logic high in both test vectors, so that it always enables the output buffer. The second bit, which is dedicated for the input line of the output buffer, is chosen to be logic low in the first test vector, and logic high in the second test vector. The third bit, which is dedicated for the input of the internal logic, is also toggled in both test vectors. Notice that in a cell which is configured as input, the IOB.I line is connected to the internal logic input, while the IOB.T and IOB.O lines are not connected to the output of the internal logic. The opposite is true for a cell which is configured as output. Therefore, the third bit of the test vector (first bit from the right) in an input cell, will be applied to the input of the internal logic, after clocking the update latch in the Update-DR state. The logic state of this third bit should appear on the IOB.O line of the output cell which is connected to the mentioned input cell. In the next Capture-DR state, the logic state of this third bit (IOB.O), the logic state of the IOB.T, and the logic state of the buffer output are loaded in the data registers of the output cell. In a Shift-DR state, the logic states of the data register, is shifted out, and therefore can be verified. The following components of the BSC are tested in the proposed test procedure;

- The boundary scan cells, which includes the data registers, the update latches, the multiplexers, and the output buffer.
- The TAP controller.
- The instruction registers.

Figure 4 shows the data lines (thick lines) which are tested in a boundary scan cell, which has been configured as input cell, while Figure 5 shows the data lines (thick lines) which are tested in a configured output cell. Table 2 shows the detected stuck at faults in the corresponding data lines of the boundary scan cells.

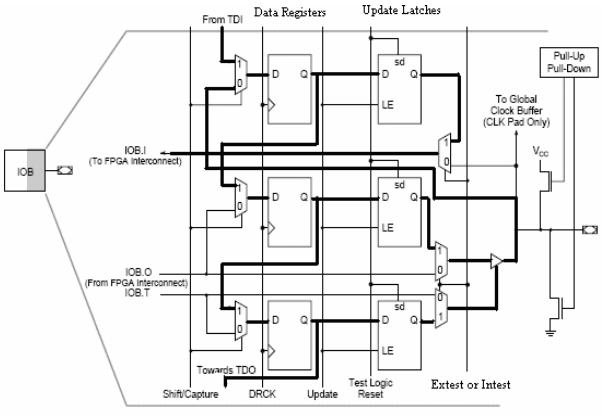

Figure 4. Tested data lines in an input cell.

The TAP controller is partially tested, since only twelve states of its sixteen states are performed in this test procedure. The instruction register is also partially tested, since only the INTEST instruction is used.

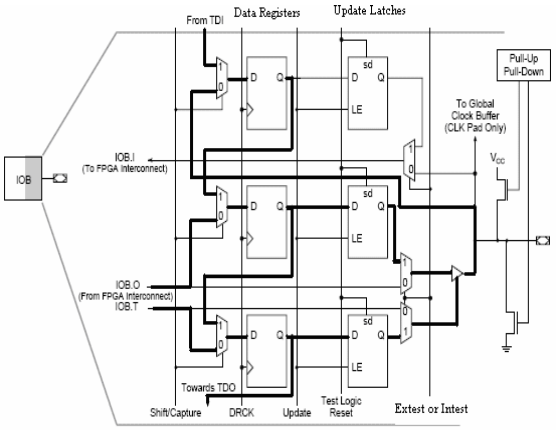

Figure 5. Tested data lines in an output cell.

Table 2. Tested data lines and their detected stuck at faults.

Data Line	Stuck at "0"	Stuck at "1"
Both input and output of all the data registers	Yes	Yes
Both input and output of all the update latches, except those driving the 3-state control line of the output buffer.	Yes	Yes
Both input and output of the update latches, which are driving the 3-state control line of the output buffer.	Yes	No
Input 0, input 1, output, and control line of all multiplexers, which are driving the data registers, except those whose their input 0 is connected to IOB.T	Yes	Yes
Input 1, output, and control line of multiplexers, which are driving the data registers, and whose their input 0 is connected to IOB.T	Yes	Yes
Input 0 of multiplexers which are driving the data registers, and whose input 0 is connected to IOB.T	Yes	No
Input 1 and output of all multiplexers which have their input 1 is connected to the update registers, except those driving the 3-state line of the output buffer.	Yes	Yes
Input 1 and output of all multiplexers which have their input 1 is connected to the update registers, and driving the 3-state line of the output buffer.	Yes	No
Both input, and output of output buffer	Yes	Yes
3-state control line of the output buffer	Yes	No
IOB.I , and IOB.O lines of all cells	Yes	Yes
IOB.T line of all cells	Yes	No
Data registers clock, and update latches clock lines	Yes	Yes

5. Conclusions

A test procedure has been proposed to test mainly the boundary cells of PLD, and FPGA devices. Table 2 shows that most of the stuck at faults in the boundary scan cells (at gate level), are detected. A future work could be the implementation of the proposed test procedure by a high level language such as C++ on a host computer. The host computer can drive the JTAG port of the device under test through the printer port. The proposed test procedure can be further developed to include the testing of other data registers such as the Bypass register, and ID register, by executing the BYPASS, and the IDCODE instructions.

References

[1] Application Note 39, www.altera.com, 2005.

[2] Brian F., Vivek C., Bing L., Harry L., and Gary K., "IEEE Standard Test Access Port and Boundary Scan Architecture IEEE Std 1149.1-1990," *Computer Journal of IEEE*, vol. 39, no. 2, pp. 195-205, 1990.

[3] Melocco K., Arora H., and Setlak P., "A Comprehensive Approach to Assessing and Analyzing 1149.1 Test Logic," *in Proceedings of the International Test Conference*, USA, pp. 358-367, 2003.

[4] Rebaudengo M., Sonza M., and Violante M., "A New Functional Fault Model for FPGA Application Oriented Testing," *in Proceedings of 17th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, Germany, pp. 372-374, 2002.

[5] Renovell M., Portal M., Figueras J., and Zo-Rian Y., "RAM-based FPGAs: A Test Approach for the Configurable Logic," *in Proceedings of the Design Automation and Test in Europe*, Los Alamitos, pp. 82-88, 1998.

[6] Renovell M., Portal M., Figueras J., and Zo-Rian Y., "Testing the Interconnect of RAM Based FPGAs," *Computer Journal of IEEE Design and Test of Computers*, vol. 15, no. 1, pp. 45-50, 1998.

[7] Shaikh A., *IEEE Standard Test Access Port and Boundary Scan Architecture IEEE Std 1149.1-2001*, *Computer Journal of IEEE*, vol. 37, no. 6, pp. 104-109, 2001.

[8] Xilinx Application Note XAPP017, www.xilinx.com, 1999.

Bashar Al-Khalifa received the BSc degree in electrical and electronic engineering in 1982 and the MSc degree in electronic and communication from University of Mosul in 1993. He is a lecturer in the Technical College/ Mosul, Iraq since 1994.