Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 13, Issue. 2 : 2022
ISSN : **1906-9685**

ALU IMPLEMENTATION IN BIST-BASED FPGA

[#1]DOOSA KAVITHA, ***Asst. Professor,***
[#2]MAVURAPU SWAPNA, ***Asst. Professor,***
Department of Electronics Communication Engineering,
Sree Chaitanya Institute of Technological Sciences, Karimnagar, Ts.

ABSTRACT: As sub-micron technology improves, it becomes increasingly difficult to do VLSI testing. The importance of testing and developing with testability in mind has been widely acknowledged in recent years. Many forms of digital production make use of Field Programmable Gate Arrays. Because FPGAs may be rewritten, it's simple to overlook issues once their locations have been pinpointed. This study presents a novel, improved Built-In Self-Test (BIST) technique, often known as "test point insertion," that detects all possible errors with minimal setup time and without altering the CUT. The final circuit design is implemented on an FPGA Spartan-3e board.

Keywords:*FPGA,BIST,ASIC*

1.INTRODUCTION

Field programmable gate arrays (FPGAs), which are ready-made silicon chips, can be utilized to build practically any type of digital computer or system. For low to medium volume manufactures, Application Specific Integrated Circuits (ASICs) are more expensive and take longer to market. FPGAs, on the other hand, are a less expensive option that can reach the market faster. An FPGA, on the other hand, costs between a few hundred and a few thousand dollars and takes less than a minute to set up. For a variety of reasons, one section of an FPGA can be altered while the rest of the FPGA remains operational.A conventional FPGA is composed of two major components: programmable logic blocks, which perform logic functions, and programmable routing, which connects these logic functions.

I/O blocks that connect to logic blocks using non-chip routing interconnects.

The customizable logic blocks (CLBs) are arranged in a two-dimensional grid and linked by programmable routing resources in Fig. 1, which depicts a relatively basic version of an FPGA. I/O blocks are put and connected to the customisable routing connection around the grid's edge. When it comes to FPGAs, the word "programmable/reconfigurable" refers to the ability to add new features to the chip after it has been manufactured. Because of the programming technique that allows FPGAs to be reconfigurable and programmable, the behavior of a previously manufactured device can alter after it has been manufactured. FPGAs have been tested using a variety of ways. In certain works, the circuits under consideration are logic circuits based on custom-designed FPGAs. This method cannot be used for manufacturing time testing since we do not know how an FPGA will be configured in the end; it can be configured in a variety of ways. Many experts have advised testing for common problems in FPGAs. These methods do not allow you to connect the test FPGA to a specific processing function. As a result, it is normal to require multiple test sessions, each working on a separate setup.

BIST is becoming more interesting as an option in IC testing for random reasoning. However, as external testing becomes more difficult and expensive, new breakthroughs in IC process technology will almost probably lead to increased use of logic BIST. According to the ITRS (International Technology Roadmap for Semiconductors), if logic BIST is not employed, it may cost more to test a transistor than to produce one in 2014. Digital modules utilize logic BIST, analog modules use analog BIST, and memory modules use memory BIST.

This means that only one low-cost external tester is required. Figure 1 depicts the fundamental FPGA layout for VLSI architecture.

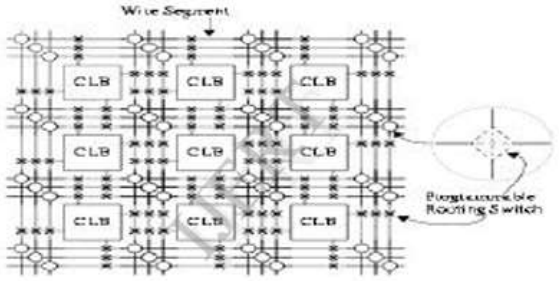

Figure 1: Generic FPGA

Arithmetic and logic units (ALUs) are digital circuits that can perform both arithmetic and logic functions. The ALU is a critical component of the computer's core unit. Modern CPUs and GPUs can feature processors that can manage a large number of sophisticated and complex ALUs in a single chip. An external control unit instructs the ALU on what action to take with the data. After that, the ALU reads the data from the input registers and puts the result in an output register. The Control Unit is responsible for moving processed data between these registers, the ALU, and memory.

The main topic of this article is how to develop an ALU with BIST capability in VHDL using LFSR techniques using Field Programmable Gate Array (FPGA) technology.

2.BIST ARCHITECTURE

A pattern generator, a response analyzer, and a test controller are the three pieces of hardware that must be added to a digital circuit in order to use the BIST architecture. Pattern generators include counters, linear feedback shift registers, and ROMs with pre-programmed patterns. LFSRs are constructed using flip-flops and XOR gates that work together to form a shift register. The feedback channels are connected in a straight line. An LFSR can be used to generate bogus random patterns, split polynomials, shorten responses, and perform other functions. Because it is compact and simple to understand, LFSR is commonly utilized for TPG and ORA implementation. A popular response analyzer is either a comparator that uses previously saved replies or an LFSR used as a signature analyzer. Instead of typical chip testing, which simply searches for flaws, system-level fault analysis is frequently employed. This is due to the fact that the chip's components cannot be changed. FPGA defects can easily overlooked by leaving troublesome components out of the final circuit. So, as long as we can identify the problem areas, we can continue to employ FPGA chips that have issues. Because our method is likewise based on the BIST method, we don't require much assistance from an outside ATE because the chip performs the majority of the tests alone. Figure 2 depicts the BIST components.

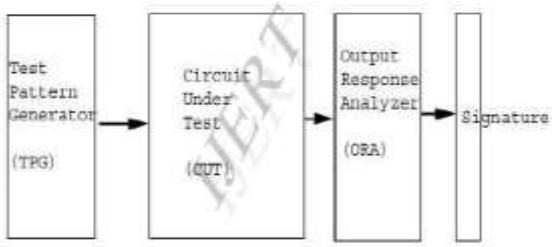

Figure 2 BOIST Building Blocks

3.RELATED WORK

Many scholars are interested in BIST implementation and FPGA-related research. Here are a few examples:

Noorbasha et al. discussed approaches for detecting failures and determining what's wrong with FPGA CLBs. To keep things simple, we will simply discuss the Configurable Logic Block (CLB) and not the entire FPGA. Nur A. Touba spoke on Synthesis Techniques for Pseudo-Random BUILT-IN SELF-TEST. Crouch et al. focused on the Built-In-Self Test (BIST) design for sequential circuits based on cellular automata (CA).Before finalizing the UART design, Yamani et al. added the BIST approach. They accomplished this by reorganizing the current design to fulfill the testability requirements. Hegde Suma T. et al., on the other hand, discussed the design and implementation of an ALU employing redundant binary signed. FAGOT et al. demonstrated the effectiveness of the proposed technique by assessing area overhead, fault coverage, and test sequence time.

4.ALU IMPLEMENTATION WITH BIST FEATURE

When the beginning seed and primitive polynomial values in the TPG block are set to the correct values, a software application automatically converts built-in self-test blocks into VHDL models of ital circuits. When the

ALU code is entered into CUT, the code for BIST is generated. This code is then developed in the Xilinx web pack 12.4 for the Spartan 3e devices. In the Xilinx 12.4 project browser, you may view the hardware summary for each method execution log file.Figure 3 depicts the RTL perspective of the ALU with BIST capabilities.

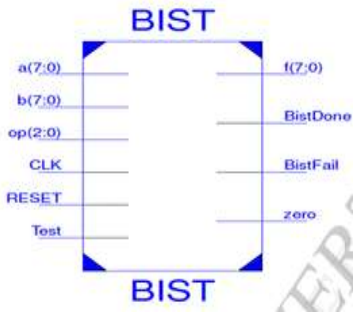

Figure 3: An examination of the BIST in RTL

The Test Pattern Generator, Signal Register, Comparator, Controller, MISR, MUX, ALU, and MISR are some of its components. The BIST has all of these elements. In this scenario, the MUX is utilized to select the appropriate action, and the ALU subsequently executes it. The MUX then selects the Test Pattern. The output of the Test Pattern and the ALU after passing through the MISR is compared. Everything is controlled by the player. Based on the TEST signal, the CPU will determine whether or not to test the ALU. The controller also assigns a BIST Pass or BIST Fail based on the success of the test.In the RTL view, the BIST is in the correct location and is properly linked. On its pins, it contains signal inputs, operation inputs for selecting the operation (AND, OR, addition, subtraction, and so on), and more. Send the RESET signal to restart the bist processes. When the TEST light turns on, the tests will begin. If TEST = 1, ALU will be examined, but BIST will not be checked if BIST = 0. If not, the test is completed (bidone=1); otherwise, it fails (bifail=1).

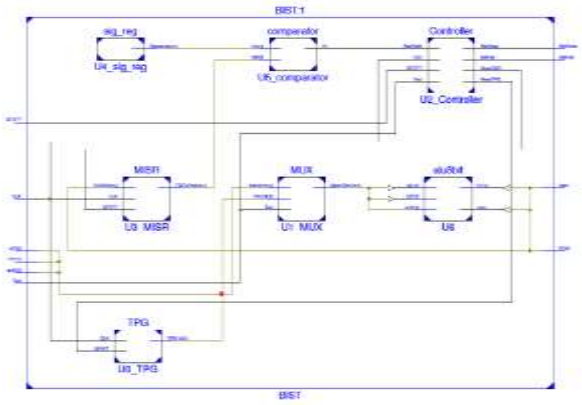

Figure 4: ALU with BIST from the right side

5.SIMULATION OF COMPLETE BIST FOR ALU

Figure 5 depicts the modeling of a complete BIST for an ALU, which comprises the four components we discussed earlier (mux, ALU, TPG, and controller section).The inputs are "op", "bist/test", "bist/a", "bist/b", clk, and reset. The outputs are as follows: "signal" f, "bistdone", "bistfail", and "zero".

Figure 5: Complete the BIST exercise for ALU.

REFERENCES

1. Noorbasha. F, Harikishore K., Hemanth Ch., Sivasairam A., Vijaya Raju," LFSR Test Pattern for Fault Detection and Diagnosis for FPGA CLB
2. CELLS" International Journal of Advances in Engineering & Technology,Vol. 3, Issue 1, pp. 240-246 March 2012.
3. Fagot C., Gascuel O., Girard P. and Landrault C.," A Ring Architecture Strategy for BIST Test Pattern Generation" c 2003 Kluwer Academic Publishers .Manufactured in The Netherlands. Received December 6, 2000; Revised December 3, 2002.
4. Mohd Yamani Idna Idris "A VHDL Implementation of UART design withBIST

capability" Malaysian Journal of Computer Science, Vol. 19 (1), 2006.
5. Nur A. Touba," Synthesis Techniques for Pseduo-Random BUILT-IN SELF-TEST" a Dissertation in partial fulfillment of the requirements forthe Ph.d,1996.
6. Bushnell M. L. and Agrawal V. D. (2000) Essentials of Electronic Testing. Kluwer Academic Publishers.
7. Suma T.Hegde ,Dr.Siva Yellampalli And Nandeesh R.," Design andImplementation of ALU using Redundant Binary Signed Digit"
8. International Conference on VLSI, Communication & Instrumentation (ICVCI) 2011 Proceedings published by International Journal of Computer Applications® (IJCA) 30.
9. C. H. Roth "Digital System Design Using VHDL" PWS PublishingCompany, 1998
10. Crouch, A. (1999), "Design-for-test for Digital IC's and Embedded CoreSystems", Prentice Hall, 347 pp.