Design and Implementation of an Area and Power Efficient 8-bit Low Power Carry Select Adder based LFSR

Sanjeevini Poonaykar B K, student, Department of ECE, BIT, Bengaluru, India (email: sanjeevini27061998@gmail.com.com)

Dr Vidyasaraswathi H N, Assistant professor. Department of ECE, BIT, Bengaluru, India (vidyasaraswathi@bit-bangalore.edu.in)

***Abstract*— The rapid growth of Very Large-Scale Integration (VLSI) technology has increased circuit complexity, creating significant challenges in testing and validation. Traditional Automatic Test Equipment (ATE)-based testing is costly, time-consuming, and less effective for nanometer-scale designs. Built-In Self-Test (BIST) has emerged as a promising alternative, enabling on-chip test pattern generation and response analysis. This work proposes the design and realization of a compact and energy-efficient BIST framework for testing Arithmetic Logic Units (ALUs), employing Linear Feedback Shift Registers (LFSRs) along with a low-power Carry Select Adder (LP-CSLA). Multiple LFSR variants— Fibonacci, Galois, and Complete—are configured to generate pseudo-random operands and opcodes, while a Parallel Input Signature Register compacts ALU outputs for fault detection. The architecture was modeled in Verilog HDL, simulated using Xilinx Vivado, and synthesized with Cadence tools on a 45 nm technology library. Results demonstrate significant improvements in power and area efficiency, with standalone LFSRs consuming ~0.015 mW, the complete ALU-based BIST consuming ~0.595 W, and FPGA utilization remaining below 2%. The proposed architecture achieves minimal hardware cost and extensive fault detection capability and reduced dynamic power, making it a scalable and energy-efficient solution for modern VLSI systems and System-on-Chip (SoC) applications.**

***Index Terms*— Built-In Self-Test (BIST), Linear Feedback Shift Register (LFSR), Multiple Input Signature Register (MISR), Carry Select Adder (CSLA), Arithmetic Logic Unit (ALU), Low Power VLSI, FPGA Implementation.**

I. INTRODUCTION

The continuous scaling of semiconductor technology has enabled the integration of billions of transistors on a single chip, giving rise to highly complex System-on-Chip (SoC) and Very Large-Scale Integration (VLSI) designs [1]. With these developments, it has become possible to advanced functionalities and high-speed operations, it has also increased susceptibility to faults such as stuck-at faults, bridging faults, and delay faults [2]. Reliable and efficient testing has therefore become a critical requirement to ensure system correctness, yield, and long-term dependability.

Traditional testing methodologies rely heavily on Automatic Test Equipment (ATE), which, although effective for smaller circuits, becomes impractical for nanometer-scale VLSI systems due to excessive cost, long test times, and limitations in at-speed testing [3]. As circuit complexity increases, the volume of test data required grows exponentially, further burdening external test infrastructures [4].

To address these challenges, Design-for-Testability (DFT) strategies have been introduced, with Built-In Self-Test (BIST) emerging as one of the most promising solutions [5]. BIST integrates test pattern generation and output response analysis directly within the circuit, thereby enabling autonomous testing without significant reliance on external testers. A key advantage of BIST is its capacity to carry out in-field testing at system clock speed, offering extensive fault detection with reduced additional hardware [6].

Among the different test pattern generators used in BIST, the Linear Feedback Shift Register (LFSR) is widely preferred due to its hardware efficiency, scalability, and capability to generate high-quality pseudo-random sequences [7]. Different LFSR architectures, such as Fibonacci, Galois, and hybrid models, have been proposed to improve randomness and fault coverage while minimizing power overhead [8]. In parallel, the Carry Select Adder (CSLA) has been recognized as an effective arithmetic unit design, offering a balance between speed and area efficiency. Recent low-power CSLA (LP-CSLA) variants further optimize switching activity, making them suitable for energy-constrained BIST-enabled designs [9].

This work focuses on designing an area- and power-efficient BIST-enabled Arithmetic Logic Unit (ALU) using configurable LFSR architectures in combination with an LP-CSLA. The proposed approach achieves high test coverage, reduced dynamic power consumption, and minimal area overhead, validated through FPGA-based implementation and synthesis in advanced VLSI tools.

II.BACKGROUND

The rapid evolution of semiconductor technology has made it possible to integrate billions of transistors onto a single chip. This breakthrough has driven the development of high-performance microprocessors, digital signal processors, and application-specific integrated circuits (ASICs).

Conventional testing methods rely heavily on Automatic Test Equipment (ATE)**.** Although ATE offers high fault coverage, it comes with drawbacks such as high cost, large data handling requirements, and limited scalability for ultra-high-speed circuits. With integrated circuits operating in the multi-gigahertz range, meeting the required speed for at-speed testing using external testers is extremely difficult, making such approaches less practical for modern System-on-Chip (SoC) devices.

To overcome these issues, the industry has adopted the concept of Design for Testability (DFT)**,** which introduces test structures directly within the circuit to improve controllability and observability. Among various DFT strategies, Built-In Self-Test (BIST) has gained prominence as it allows circuits to evaluate themselves without relying on expensive external equipment. BIST not only reduces dependency on ATE but also enables efficient fault detection and in-system diagnosis, providing a scalable and cost-effective testing solution for complex VLSI systems.

An essential component of BIST is efficient test pattern generation, for which Linear Feedback Shift Registers (LFSRs) are commonly employed. These registers generate pseudo-random sequences that ensure broad fault coverage with very little hardware overhead. Their simplicity, ease of implementation, and ability to operate at the maximum system frequency make them a highly effective choice for BIST designs.

Conventional testing techniques rely on storing large volumes of test vectors externally and applying them fed to the Circuit Under Test using Automatic Test Equipment (ATE). However, with the exponential growth in circuit complexity, the number of required test vectors also increases, resulting in excessive memory demands and prolonged test durations. Furthermore, advanced SoCs, which often incorporate multiple clock domains, embedded memories, and mixed-signal components, present additional challenges that external test methods alone cannot efficiently address.

Built-In Self-Test (BIST**)** effectively addresses these limitations by embedding the test infrastructure directly into the circuit. A typical BIST architecture consists of the following components:

- **Test Pattern Generator (TPG):** Responsible for generating the input test sequences.
- **Output Response Analyzer (ORA):** Captures and compresses the output responses for fault detection.
- **BIST Controller:** Manages and coordinates the overall test operation.

BIST enables at-speed self-testing with minimal external support, ensuring continuous fault detection, reduced downtime, and improved system reliability. It also provides cost-effective and scalable testing for modern VLSI designs.

The **conventional test methodologies** generally rely on storing large numbers of test vectors externally and applying them to the **Circuit Under Test (CUT)** through Automated Test Equipment (ATE). However, as circuit complexity increases, the number of required test vectors grows exponentially. This leads to excessive memory usage, longer test application times, and higher overall testing costs. Moreover, modern **System-on-Chip (SoC)** designs integrate multiple clock domains, embedded memories, and mixed-signal modules, making external-only testing methods impractical and inefficient.

To overcome these issues, **Built-In Self-Test (BIST)** has emerged as a practical solution. Unlike traditional methods, BIST incorporates the testing infrastructure directly into the circuit itself. This allows the IC to perform self-testing without relying heavily on external test equipment. A typical BIST architecture includes three main components:

Faults in Scaled Technologies & Need for BIST

1. As technology scales into the **nanometer regime**, ICs become more prone to faults such as:
    - Stuck-at faults
    - Bridging faults
    - Transition faults
    - Delay faults
    - Soft errors (due to radiation/voltage fluctuations)
2. **Traditional testing using ATE (Automated Test Equipment):**
    - Requires storing and applying large sets of test vectors externally.
    - Test vector count grows exponentially with circuit complexity.
    - Leads to high memory requirements and long test times.
    - Not suitable for **SoCs** containing multiple clock domains, embedded memories, and mixed-signal modules.
3. **Limitations of traditional methods:**
    - High cost and time overhead.
    - Scalability issues with modern circuits.
    - Heavy dependence on external test hardware.
4. **Built-In Self-Test (BIST) advantages:**
    - Embeds testing capability inside the IC.
    - Reduces external equipment dependency.
    - Enables testing at **actual operating frequency**.
    - Provides **in-field and periodic self-testing**.
    - Enhances **fault coverage, reliability, and system dependability**.
5. **Typical BIST components:**
    - **Test Pattern Generator (TPG):** Produces input patterns.
    - **Output Response Analyzer (ORA):** Captures and compresses responses.
    - **BIST Controller:** Manages and coordinates the test process.

III. Work Done

Overview

The proposed work focuses on the architecture and deployment of a Built-In Self-Test (BIST) enabled Arithmetic and Logic Module (ALU) using adjustable Linear Feedback Shift Registers (LFSRs) along with a low-power Carry Select Adder (LP-CSLA) The primary goal is to achieve power-efficient and area-optimized testing while ensuring high fault coverage in modern VLSI systems.

Architecture Description: The architecture consists of three Key Components

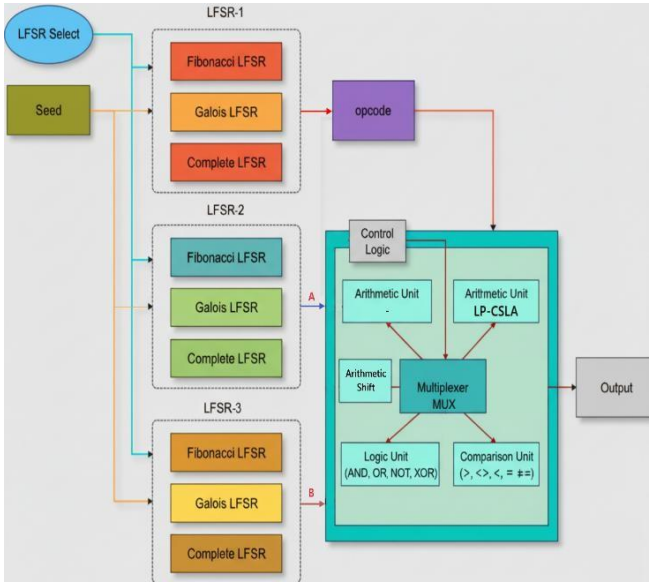

Fig.1. Block diagram of ALU testing architecture

Test Pattern Generator (TPG): Implemented using three LFSRs (Fibonacci, Galois, and Complete variants). Configurable via lfsr_select [1:0] input and initialized with user-defined seed values.

Circuit Under Test (CUT): An 8-bit ALU implementing arithmetic and logical functions.

Output Response Analyzer (ORA): Executed with the help of an 8-bit Multi-Input Signature Register (MISR)

Verilog Implementation: RTL design modeled in Verilog HDL. Functional verification performed using Xilinx Vivado 2025.1. Synthesis and timing optimization carried out in Cadence Genus (45 nm library).

Modular design: LFSR, ALU, MISR coded separately and instantiated in a top-level module.
Final signature compared with a pre-computed golden signature to determine fault presence.

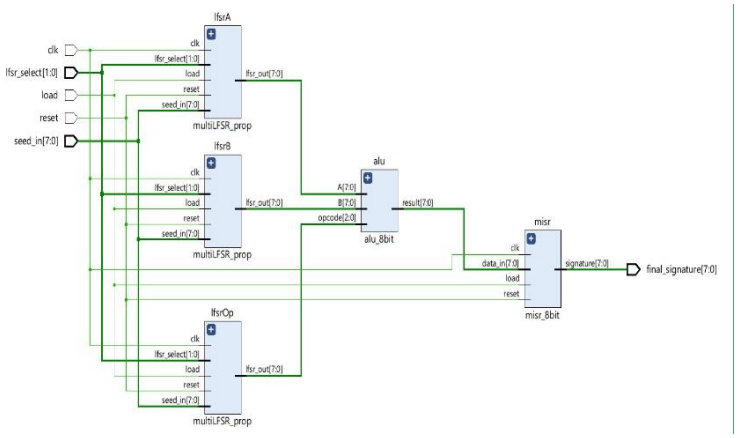

Fig.2. Schematic for the Multi_LFSR

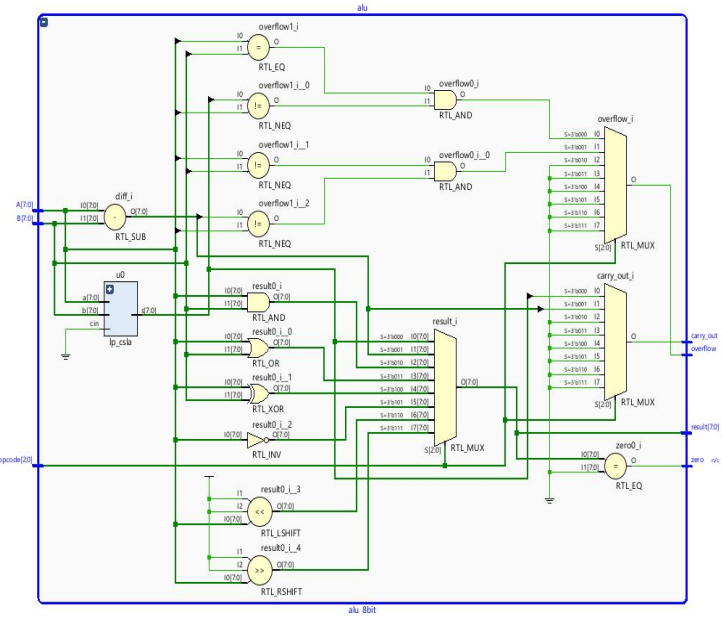

Fig.3. Schematic Of 8- Bit ALU

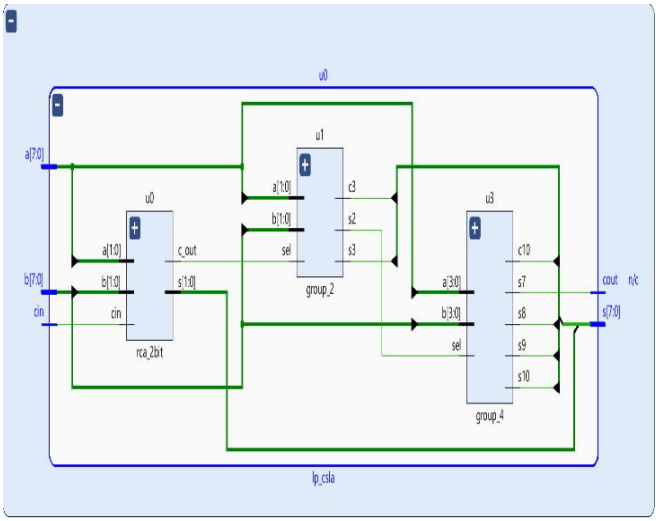

Fig.4. Low- Power Carry Select Adder (LP-CSLA)

SIMULATION

Functional Simulation:

- LFSRs successfully generated pseudo-random test patterns.
- ALU operations executed correctly under different opcode inputs.
- MISR produced compact test signatures.
- Waveforms confirmed proper synchronization among modules.

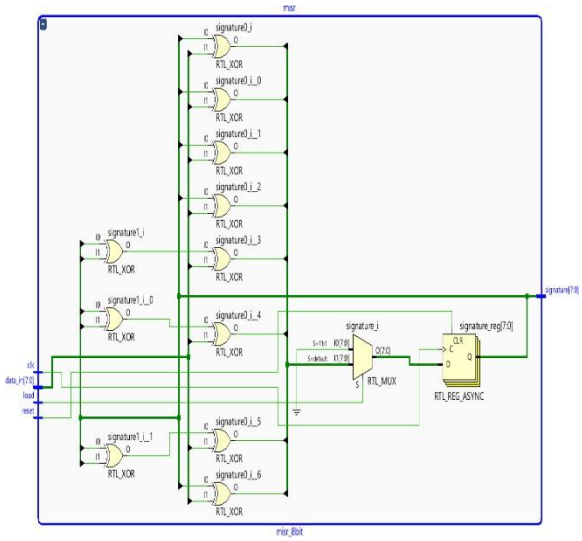

Fig.5. 8-bit multiple input signature Power

Analysis:

- Each LFSR consumed **~0.015 mW**.
- Complete BIST-enabled ALU consumed **0.595 W**.
- Operating temperature stabilized at **25.6 °C** with a thermal margin of **74.4 °C**.

AREA UTILIZATION:

Post -synthesis results show extremely low FPGA resource utilization.
Table 1 shows FPGA resource utilization Table 2 shows power consumption

Resource	Used	Available	Utilization %
LUTs	34	230,400	0.01 %
FFs	38	460,800	0.01 %
I/O	21	360	5.83 %
BUFG	1	544	0.18 %

Table I: FPGA Resource Utilization

Module	Static Power (mW)	Dynamic Power (mW)	Total Power (mW)
Fibonacci LFSR	0.00084	0.0140	0.0150
Galois LFSR	0.00084	0.0143	0.0152
Complete LFSR	0.00084	0.0149	0.0157
Full ALU BIST	–	–	595.0

Table II: Power

Consumption Key Observations

The Complete LFSR variant was found to be the most power- efficient. The integration of LP-CSLA significantly reduced switching activity, lowering overall dynamic power. FPGA utilization was <2%, indicating compact hardware implementation with scalability potential. The architecture ensures robust fault detection while consuming minimal power and area, making it suitable for low-power SoCs and embedded processors.

II. RESULTS

The proposed BIST-enabled ALU architecture was implemented in Verilog HDL and tested using Xilinx Vivado 2025.1 for functional verification. Synthesis was performed using Cadence Genus with a 45 nm technology library.

Functional Verification:

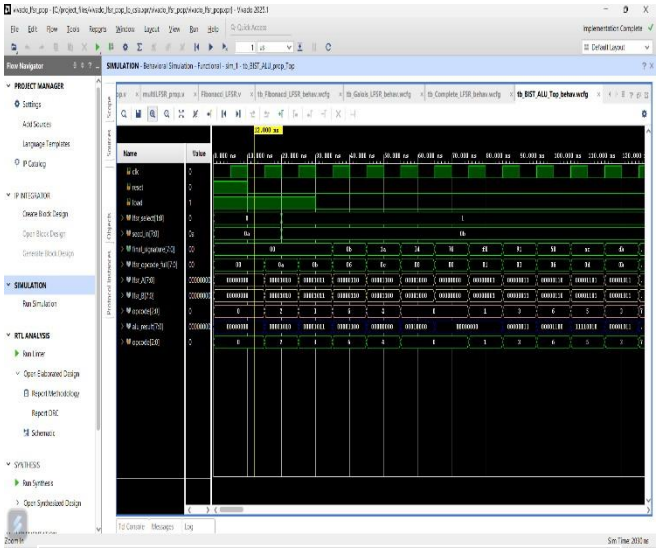

Fig:6. Timing Diagram of BIST-Enabled ALU

LFSRs generated diverse pseudo-random operands and opcodes, validating their configurability across Fibonacci, Galois, and Complete modes. The ALU correctly executed arithmetic and logic operations under BIST conditions. The MISR successfully compressed ALU outputs into final

signatures for comparison with golden references. Simulation waveforms confirmed proper synchronization between pattern generation, ALU operations, and signature analysis.

Power Analysis:

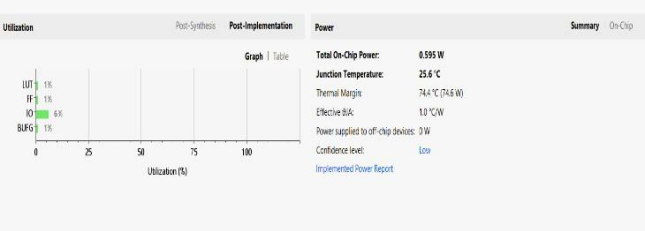

Fig.7. Power Analysis Report

Individual LFSRs consumed only ~0.015 mW, demonstrating negligible power overhead. The complete BIST-enabled ALU consumed 0.595W. Substantially lower in comparison with traditional architectures of similar complexity. Junction temperature stabilized at 25.6
°C, ensuring thermally safe operation.

Area Utilization:
FPGA implementation results show minimal resource consumption: 1% LUTs, 1% Flip-Flops, 6% I/O blocks, and 1% BUFGs.This highlights the compactness of the design, leaving significant scope for scalability.

III. CONCLUSION

This paper presented the Development and realization of a power- and area-optimized BIST framework for Arithmetic Logic Units, employing configurable LFSRs along with a low-power Carry Select Adder.The experimental results demonstrate that the proposed system achieves:

Low Power Consumption: Each LFSR consumes ~0.015 mW; complete ALU BIST ~0.595 W.
Compact Hardware Utilization: <2% FPGA resource usage, enabling efficient SoC integration.
Robust Fault Coverage: Pseudo-random test patterns and MISR-based response compaction ensure reliable fault detection.
The architecture successfully combines scalability, low energy overhead, and fault tolerance, making it highly suitable for modern embedded systems, IoT processors, and low-power SoCs. Further improvements can be explored by Extending the proposed 8-bit architecture to 16-bit or 32-bit ALUs for processor-grade applications.

REFERENCES

[1] A. Verma and R. Singh, "Built-in Self-Test (BIST) Methodologies in VLSI: A Comprehensive Survey," Journal of Electronic Testing: Theory and Applications (JETTA), vol. 39, pp. 55–72, 2023.

[2] K. S. Khushalani and R. A. Potdar, "Efficient BIST for VLSI Circuits Using LFSR Variants," IEEE Transactions on VLSI Systems, vol. 31, no. 4, pp. 512–520, 2023.

[3] N. Das and P. Reddy, "Survey on BIST and Scan-Based Testability Techniques in VLSI," AEÜ – International Journal of Electronics and Communications, vol. 165, 154414, 2023.

[4] R. Thomas et al., "Optimized ALU with Integrated BIST Functionality," Journal of Systems Architecture, Elsevier, vol. 142, 102872, 2023.

[5] S. Roy and P. Dutta, "LFSR-Based Test Pattern Generation for ALU Reliability Improvement," Journal of Circuits, Systems, and Computers, vol. 32, no. 8, 2350132, 2023.

[6] Y. Zhang and C. Li, "Reconfigurable Test Pattern Generators for ALU Self-Testing," IEEE Design & Test, vol. 40, no. 2, pp. 78–87, 2023.

[7] H. Zhao and Y. Lin, "Reconfigurable LFSR Architectures for Enhanced Pseudo-Random Test Pattern Generation," Integration, The VLSI Journal, vol. 96, pp. 44–54, 2023.

[8] S. Kumar and R. Malhotra, "Energy-Efficient LFSR Architectures for Built-In Self-Test of Digital Systems," Journal of Systems Architecture, Elsevier, vol. 142, 102872, 2023.

[9] M. Patel and N. Gupta, "Design of Low Power Carry Select Adder for High-Speed Applications," Microelectronics Journal, Elsevier, vol. 132, pp. 105–113, 2023.