ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/37-44
B. Sai Rohith *et. al.,* / International Journal of Engineering & Science Research

Power Efficient VLSI Architecture of Fault Tolerant BIST using LCG

Dr. C. Gangaiah Yadav[1], B. Sai Rohith[2], K.Jaya Krushna[3], K. Sai Jhansi[4], B. Nithin Kumar Reddy[5], I. Jahnavi[6]

[1]Associate Professor, Department of ECE ,Chaitanya Bharathi Institute of Technology,Vidhya Nagar, Pallavolu (V), Proddatur -516360, Y.S.R (Dt.), AP.

[2,3,4,5,6]UG scholar, Department of ECE ,Chaitanya Bharathi Institute of Technology, Vidhya Nagar, Pallavolu (V), Proddatur -516360, Y.S.R (Dt.), AP.

ABSTRACT:

The detection of faults in electronic systems is critical for ensuring reliability and optimal performance, especially in safety-critical applications. One innovative technique for fault detection is the use of Built-In Self-Test (BIST) technology, which integrates self-diagnostic capabilities directly into the system's architecture. This paper explores the concept of "Stuck-at Fault" detection using BIST technology, focusing on identifying and isolating faults where a signal is stuck at a constant logic level, either '0' or '1'. The study reviews the methodologies employed in designing BIST systems tailored for stuck-at faults, including fault models, test pattern generation, and fault coverage analysis. A key aspect of this research involves evaluating the effectiveness of various BIST approaches in detecting and diagnosing stuck-at faults in combinational and sequential circuits. Through simulations and case studies, the paper demonstrates the potential of BIST to improve fault detection accuracy, reduce testing time, and minimize hardware overhead. The results show that BIST technology offers a cost-effective, efficient solution for enhancing fault detection capabilities in modern integrated circuits and systems.

Keywords: *Fault detection, Built-In Self-Test (BIST), Stuck-at Faults, Test pattern generation, Fault coverage, Circuit reliability.*

1. INTRODUCTION

In modern digital systems, ensuring the reliability and functionality of integrated circuits (ICs) is crucial for the successful operation of various applications, especially in safety-critical environments like aerospace, automotive, and medical devices. One of the most common types of faults that can occur in digital circuits is the "stuck-at fault," where a signal or node is permanently fixed at a logic level, either '0' or '1'. These faults can arise due to manufacturing defects, aging, or environmental conditions and can severely affect the performance and reliability of electronic systems.

Traditional fault detection methods often rely on external testing equipment or complex diagnostic procedures, which can be costly and time-consuming. To address these limitations, Built-In Self-Test (BIST) technology has emerged as a promising solution. BIST is a self-diagnostic technique where the system is equipped with the necessary hardware and software to perform its own testing without the need for external instruments. This technique has gained significant attention in fault detection applications because it enables real-time, efficient testing during both manufacturing and operational phases.

The use of BIST for detecting stuck-at faults has become an area of considerable interest due to its potential to identify these common fault types early in the design and testing phases, as well as its ability to continuously monitor circuits in the field. The effectiveness of BIST for stuck-at fault detection depends on several factors, including the test pattern generation process, fault coverage, and the overhead introduced by the BIST circuitry itself. Optimizing these elements is essential for achieving high detection accuracy while minimizing resource usage.

This paper aims to explore the application of BIST technology specifically for stuck-at fault detection. We examine various methodologies and techniques for generating test patterns, analyze the fault coverage achieved, and assess the trade-offs between detection performance and hardware overhead. Through detailed case studies and simulations, we demonstrate how BIST can be effectively

employed to detect and diagnose stuck-at faults in both combinational and sequential circuits, ultimately contributing to enhanced system reliability and reduced testing costs.

2. LITERATURE SURVEY

W. B. Jone and D. C. Huang and S. C. Wu and K. J. Lee, An efficient BIST method for small buffers, Astonishing progress in Silicon devices and circuits and highly reliable mass manufacturing techniques have prompted unprecedented revolutions in electronics for the last 4 decades to the extent that electronics is growingly permeating numerous aspects of our life. The application world, consumer and infrastructure, is now used to exponential performance improvements and high yield. Complexity and competitiveness of modern electronic systems demand for optimization across multiple disciplines. Joint optimization of device, circuit, packaging, and test are increasingly important for high performance systems. Design for Manufacturing and Testing is more critical but increasingly challenging in complex systems. So what does optimization mean for the future of ever-growing and complex solid state electronics? A daunting challenge worth spending a talk on.D. Bronzi, Y. Zou, F. Villa, S. Tisa, A. Tosi, and F. Zappa, "Automotive three-dimensional vision through a single-photon counting SPAD camera," Linear-feedback shift register (LFSR) counters have been shown to be well suited to applications requiring large arrays of counters and can improve the area and performance compared with conventional binary counters. However, significant logic is required to decode the count order into binary, causing system-on-chip designs to be unfeasible. This paper presents a counter design based on multiple LFSR stages that retains the advantages of a single-stage LFSR but only requires decoding logic that scales logarithmically with the number of stages rather than exponentially with the number of bits as required by other methods

3. EXISTING DESIGN (BIST DESIGN USING LFSR)

In computing, a linear-feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state. The most commonly used linear function of single bits is exclusive-or (XOR). Thus, an LFSR is most often a shift register whose input bit is driven by the XOR of some bits of the overall shift register value. The initial value of the LFSR is called the seed, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its current (or previous) state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle.

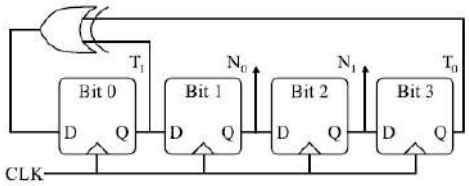

Fig 1: Structure of LFSR

In a BIST system, the DUT is tested using internal test mechanisms that are built into the device itself, reducing the need for external test equipment. BIST techniques are typically used to detect faults in the DUT and can perform tests like self-diagnosis, error detection, and fault isolation, all without external testers. The DUT interacts with these built-in self-test circuits to assess its operation during manufacturing or in-field operation.The DUT in BIST is the target hardware being tested, and BIST systems are designed to test it automatically using its internal test capabilities.

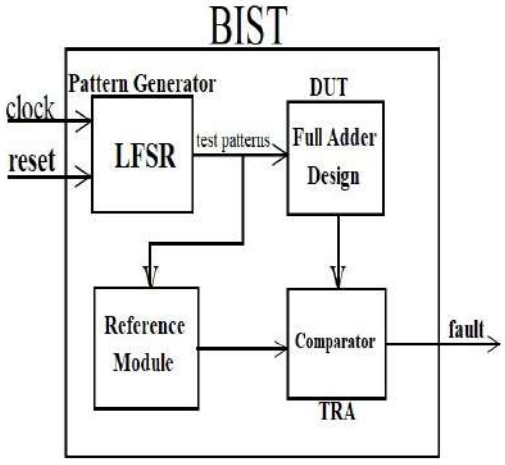

Fig 2: BIST Architecture of existing design

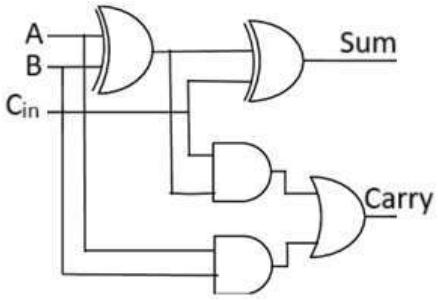

Fig 3: DUT(Full Adder)

DISADVANTAGES:

1.Existing design consumed more power.
2.LFSR having limitation of it can't generate the binary value "0" as it important. So, we can't include value "0" for testing the design module.

4. PROPOSED DESIGN (BIST DESIGN USING LCG)

During BIST, the DUT is stimulated with test patterns or input signals. The DUT then produces an output, which is captured by the TRA. The TRA examines these outputs to determine whether they match the expected values. If there is a discrepancy, the TRA flags an error or reports a failure, indicating that a fault may exist in the DUT. The TRA is responsible for detecting faults by performing a comparison between the DUT's actual response and the reference response. It ensures that the DUT operates correctly under the given test conditions. If any deviations are detected, the TRA often generates error flags or status signals that indicate the type or location of the fault. In some systems, it may also provide diagnostic information to help identify the specific issue.

LINEAR CONGRUENTIAL GENERATOR

The generation of random numbers, however, is not an easy task for a computer, since the computer is a deterministic machine with no built-in randomness. Thus, it is impossible to create true random numbers without any additional hardware. What can be done, is to create pseudo random numbers which behave almost like random numbers but which are repeated after a fixed (mostly quite long) period. These pseudo random numbers are generated by linear congruential generators (LCG). The principle of an LCG is quite simple: a new pseudo random number is generated on the basis of the previous random number by adding a certain offset and wrapping the result if it exceeds a certain limit. The process can be denoted by the following equation: The linear congruential method produces a sequence of integers X_0, X_1, X_2… between zero and m-1 according to the following recursive relationship:

$$x_{i+1} = a_1 \times x_i + b_1 \bmod 2^n \qquad (1)$$

Here, a1, b1 are the constant parameters x_i is the initial seed, Following are the necessary conditions to get the maximum period. b_1 is relatively prime with 2^n(m). a_1-1 must be divisible by 4.

LCG method is developed to generate pseudorandom bit at an equal interval of time for encrypting continuous data stream in the stream cipher. The architecture is designed with two comparators, four LCG blocks, one controller unit and memory (flip-flops) as shown in Fig. 1. The LCG is the basic functional block in the dual-CLCG architecture that involves multiplication and addition processes to compute n-bit binary random number on every clock cycle. The multiplication in the LCG equation can be implemented with shift operation, when a is considered as (2^r+1). Here, r is a positive integer, $1 < r < 2^n$.

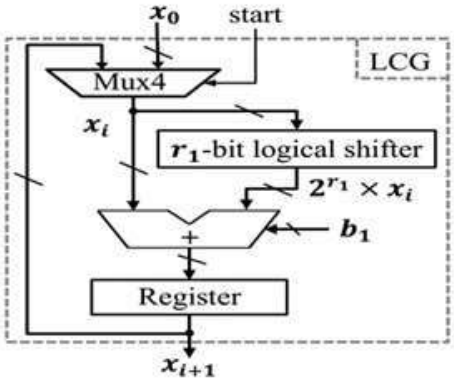

Fig 4: Architecture of the linear

congruential generator
In this design Linear congruential generator

generated test patterns, these patterns are given to the tested circuit (DUT) and reference module parallelly and comparator (TRA) compare the results from DUT and reference module, if both results are similar there is no fault other wise there is a fault in DUT.

ADVANTAGES:

1. Proposed design consumed less Power compared to existing design.
2. There is no limitation to generate patterns.

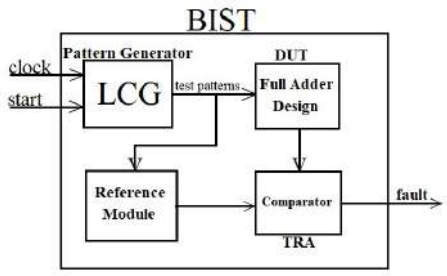

Fig5: BIST Architecture of proposed design

5. RESULTS

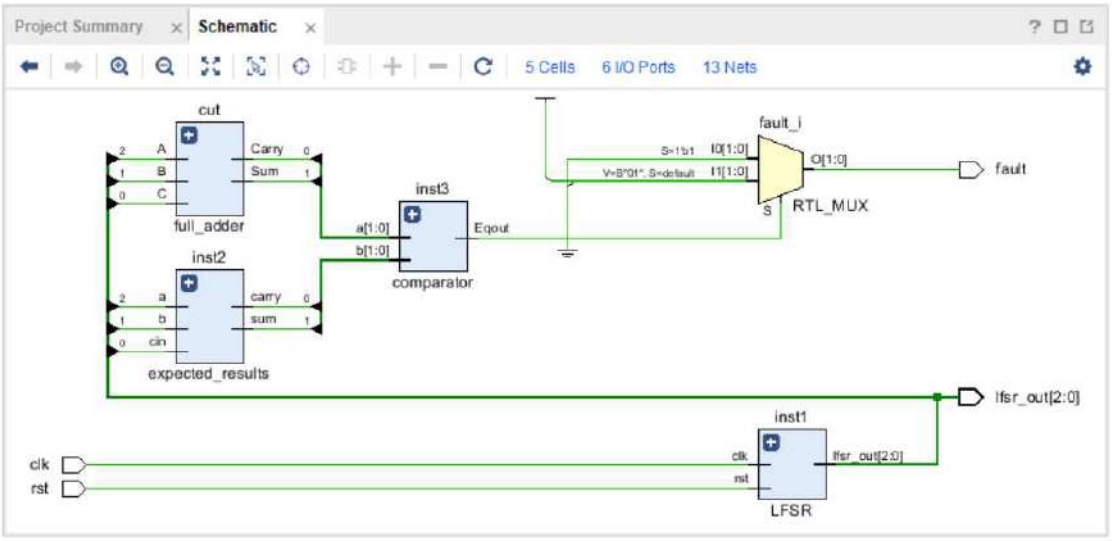

Fig 6: RTL Schematic of Existing Fault detecting BIST using LFSR

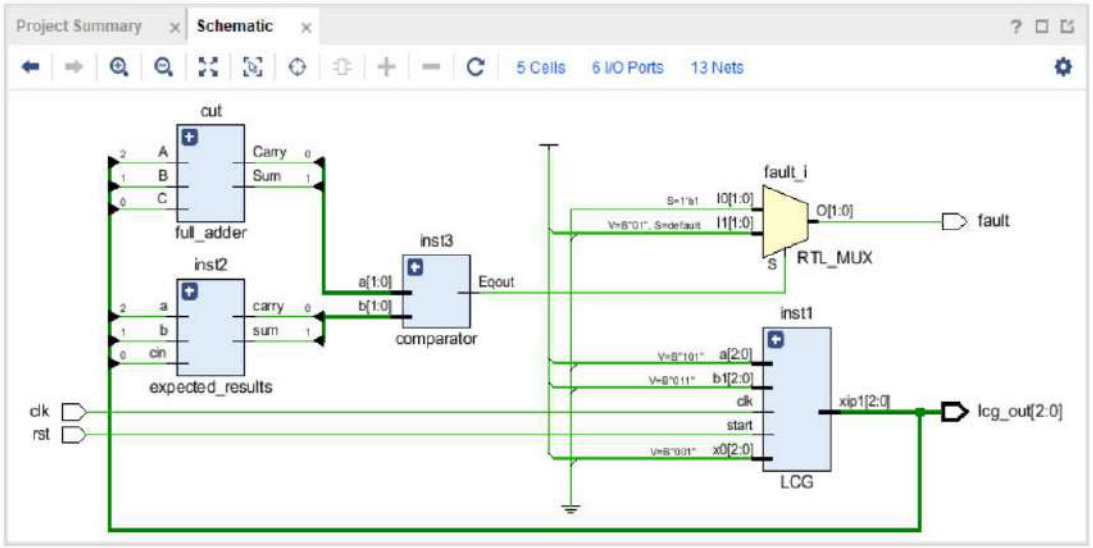

Fig 7: RTL Schematic of Proposed Fault detecting BIST using LCG.

RTL SCHEMATIC: The RTL schematic is abbreviated as the register transfer level it denotes the blue print of the architecture and is used to verify the designed architecture to the ideal architecture that we are in need of development .The hdl language is used to convert the description or summery of the architecture to the working summery by use of the coding language i.e verilog, vhdl. The RTL schematic even specifies the internal connection blocks for better analyzing .The figure represented below shows the RTL schematic diagram of the designed architecture.

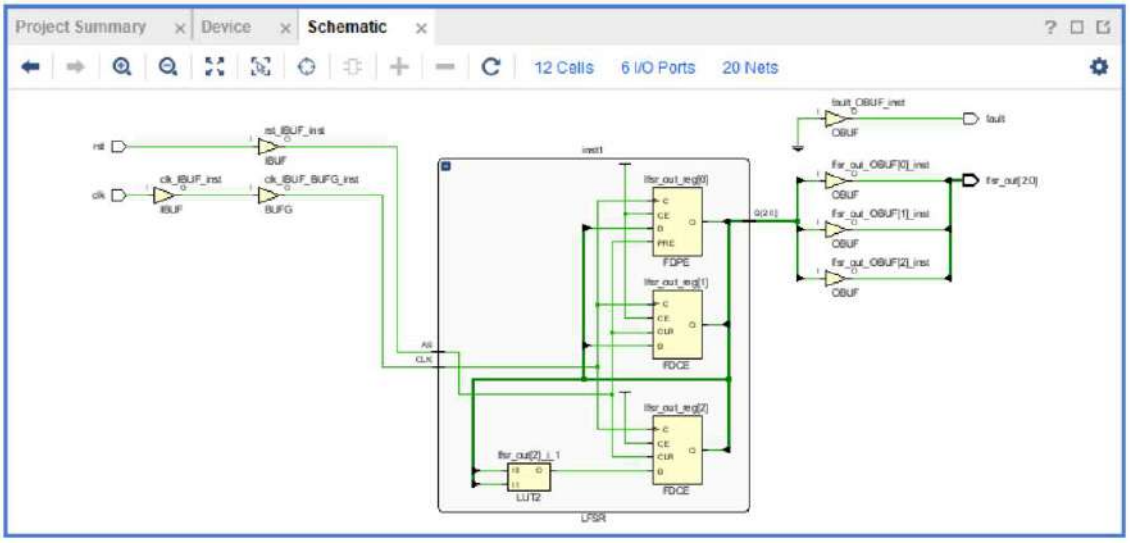

Fig 8: View Technology Schematic of existing Fault detecting BIST using LFSR

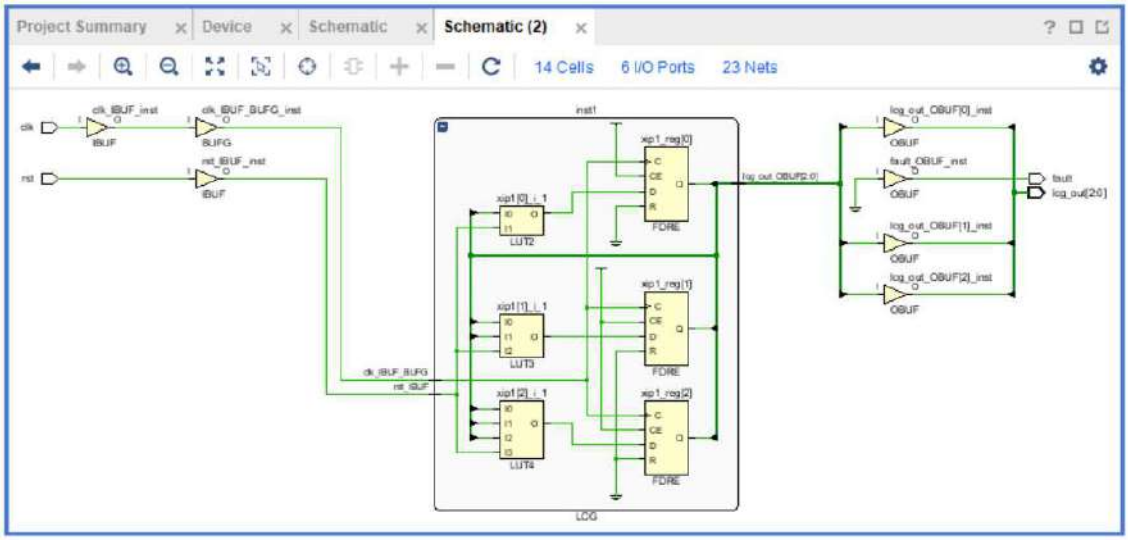

Fig 9: View Technology Schematic of proposed Fault detecting Bist using LCG.

TECHNOLOGY SCHEMATIC: The technology schematic makes the representation of the architecture in the LUT format ,where the LUT is consider as the parameter of area that is used in VLSI to estimate the architecture design .the LUT is consider as an square unit

the memory allocation of the code is represented in there LUT s in FPGA.

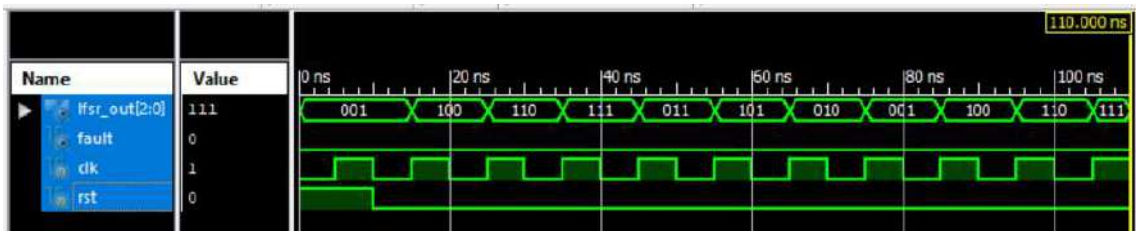

Fig 10: simulated of existing Fault free bist.

Fig 11: simulated of proposed design Fault free bist.

Fig 12: simulated of proposed design Faulty bist.

Simulation: The simulation is the process which is termed as the final verification in respect to its working where as the schematic is the verification of the connections and blocks. The simulation window is launched as shifting from implementation to the simulation on the home screen of the tool ,and the simulation window confines the output in the form of wave forms out put. Here it has the flexibility of providing the different radix number systems.

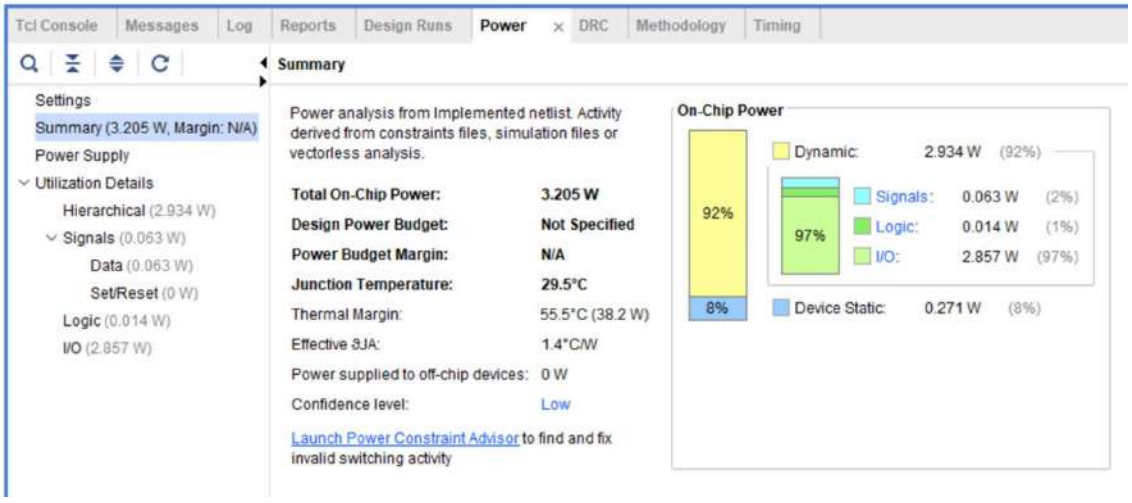

Table 1: Power Report for Existing Design

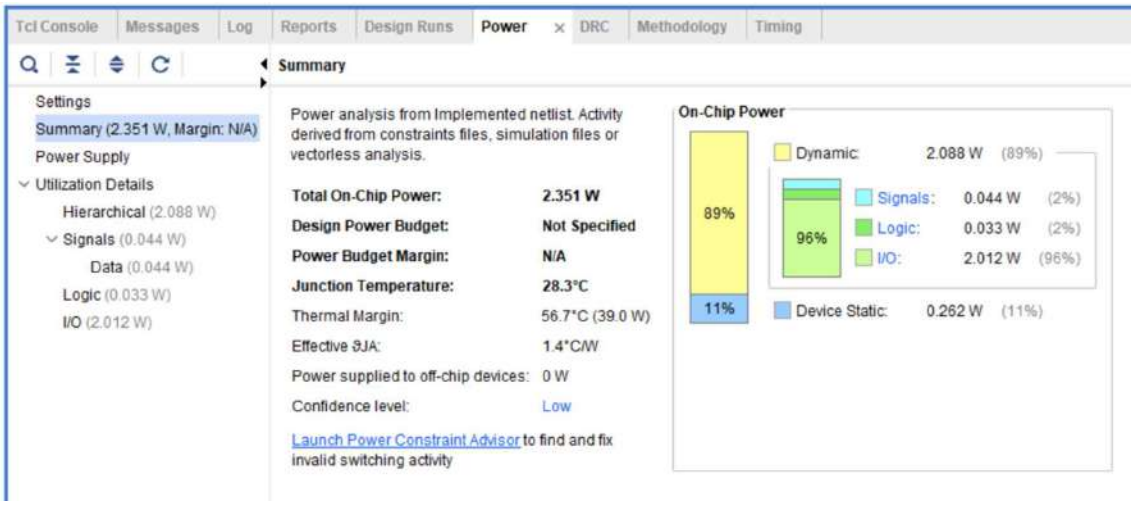

Table 2: Power Report for Proposed Design

CONCLUSION

Detecting stuck-at faults using Built-In Self-Test (BIST) has become a crucial technique for ensuring the reliability and performance of digital circuits, especially in integrated circuits (ICs) and System-on-Chip (SoC) designs. A stuck-at fault occurs when a signal in a circuit is fixed at either logic "1" or logic "0" and cannot switch to the opposite value. This type of fault can significantly impact the functionality of a system. BIST provides a self-sufficient mechanism for detecting these faults during the manufacturing process, during operation, or in-field maintenance. The BIST system typically includes, Test pattern generation (LCG) to stimulate the DUT (Device Under Test), Reference modules to provide expected outputs, Test response analyzers (comparator) to compare the actual outputs with expected values.

By embedding the self-test capability within the device, BIST enables effective and efficient detection of stuck-at faults without the need for external test equipment, offering advantages such as reduced test costs, faster turnaround times, and continuous monitoring in operational settings. Detecting stuck-at faults using BIST has significantly improved the testing and reliability of digital systems. As technology evolves, future BIST systems will continue to advance by incorporating smarter test strategies, AI-driven diagnostics, and fault correction mechanisms. This will help address the increasing complexity of modern devices and improve fault detection coverage, leading to more reliable and robust systems.

REFERENCES

[1] L.Bushnell, and Vishwani D.Agrawal, "Essentials of Electronic Testing for Digital, Memory & Mixed–Signal VLSI Circuits",:Springer, 2002

[2] MironAbramovici, Melvin A.Breuer, and Arthur D.Friedman. " Digital Testing and Testable Design", Piscataway-New Jersey: IEEE Press,1994.

[3] N. Weste and K. Eshragian, “Principles of CMOS VLSI Design”, A Systems Perspective, second ed. Addison-Wesley, 1994.

[4] Vinod Kumar Khera , R.K. Sharma, and A.K. Gupta " A heuristic fault based optimization approach to reduce test vectors count in VLSI testing", Journal of King Saud University – Computer and Information Sciences (2019)

[5] K.L.V.Ramana Kumari, M.Asha Rani and N. Balaji "Design Verification And Test Vector Minimization Using Heuristic Method Of A Ripple Carry Adder", International Journal on Cybernetics & Informatics (IJCI) vol. 5, No. 4, pp. 307-313, August 2016.

[6] Dhanabalan G and Tamil Selvi S, "Design of parallel conversion multichannel analog to digital converter for scan time reduction of programmable logic controller using FPGA", Computer standards and interfaces, vol. 39, pp. 12 – 21, 2015.
[7] Mary D. Pulukuri and Charles E. Stroud " On Built-In Self-Test for Adders", J Electron Test (2009) 25:343–346 DOI 10.1007/s10836-009-5114-6.
[8] Ashok Kumar, Rahul Raj Choudhary, and Pooja Bhardwaj, "Universal Pattern Set for Arithmetic Circuits", International Journal of Computer Applications (0975 – 8887) vol. 40– No.15, February 2012.
[9] S Jayanthy, and MC Bhuvaneswari, "Delay Fault Testing of VLSI Circuits", – Springer 2019, Test Generation of Crosstalk Delay Faults, chapter-2, pp 15-35.
[10] Matthias Sauer, Jie Jiang, Sven Reimer, Kohei Miyase, Xiaoqing Wen, Bernd Becker and Ilia Polian, "On Optimal Power-aware Path Sensitization", IEEE 25th Asian Test Symposium, 2016, pp. 179-184.
[11] Nelson, Nagle, Carroll, and Irwin",Digital Logic Circuit Analysis & Design" Prentice-Hall,1995, Chapter 12, pp. 739 to 757.
[12] Mojtaba Valinataj, Abbas Mohammadnezhad, and Jari Nurmi, "A Low-Cost High-Speed Self-Checking Carry Select Adder with Multiple Fault Detection", ELSEVIER Microelectronics Journal, Volume 81,November 2018, pp. 16-27.