International Journal for Modern Trends in Science and Technology
Volume 11, Issue 07, pages 176-182.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol11issue07.html
DOI: https://doi.org/10.5281/zenodo.16130991

Optimal Approach for Memory Testing and Repair using Redundancy in System on SOC

Dr. T Lakshmi Narayana, Dharmesh Reddy Undela, Jayadeep Ilipilla, Manoj Mannepalli

Department of Electronics and Communication Engineering, Andhra Loyola Institute of Engineering and Technology, Vijayawada, India.

To Cite this Article
Dr. T Lakshmi Narayana, Dharmesh Reddy Undela, Jayadeep Ilipilla & Manoj Mannepalli (2025). Optimal Approach for Memory Testing and Repair using Redundancy in System on SOC. International Journal for Modern Trends in Science and Technology, 11(07), 176-182. https://doi.org/10.5281/zenodo.16130991

Article Info
Received: 14 May 2025; Accepted: 06 July 2025.; Published: 17 July 2025.

KEYWORDS

SoC; BISR; BIRA; memory test and repair; repair-rate

ABSTRACT

The current system-on-chip (SoC)-based devices uses embedded memories of enormous size. Most of these systems' area is dense with memories and promotes different types of faults appearance in memory. The memory faults become a severe issue when they affect the yield of the product. A memory-test and - repair scheme is an atractive solution to tackle this kind of problem. The built-in self-repair (BISR) scheme is a prominent method to handle this issue. The BISR scheme is widely used to repair the defective memories for an SoC-based system. It uses a built-in redundancy analysis (BIRA) circuit to allocate the redundancy when defects appear in the memory. The data are accessed from the redundancy allocation when the faulty memory is operative. Thus, this BIRA scheme affects the area overhead for the BISR circuit when it integrates to the SoC. The spare row and spare column–based BISR method is proposed to receive the optimal repair rate with a low area overhead. It tests the memories for almost all the fault types and repairs the memory by using spare rows and columns. The proposed BISR block's performance was measured for the optimal repair rate and the area overhead. The area overhead, timing, and repair rate were compared with the other approaches. Furthermore, the study noticed that the repair rate and area overhead would increase by increasing the spare-row/column allocation.

1. INTRODUCTION

The recent SoC-based devices play a more important role as technology enhances day by day. These modern SoC designs are dense with memory, and the users need more promising features from their devices. A smooth-functioning memory-test algorithm and architecture are required to maintain the product's reputation. Present system-on-chip (SoC) designs consist of embedded memory in a large portion. The embedded memory area in recent SoC-based devices is higher and is approximately equal to 95% of the total chip area [1–3]. Due to the high density of memory, there is a high

probability of defects in SoC. Furthermore, the memories are more prone to faults than the actual logic, as memories do not consist of the logic elements such as flip-flops [3,4], and the defects in the embedded memory of the devices or the systems can cause a critical error. Therefore, regress testing embedded memories in today's complex SoC-based systems becomes necessary to retain the products' reputation in the market. Thus, the SoC-based product yield is drastically affected by the memory in the chip. The effective yield-improvement method becomes essential for SoC design. Testing memories for faults and repairing defected memory methods play a vital role in improving the SoC design yield [5]. Memory- fault test and repair are popular techniques for yield improvement [6], and the built-in self-repair (BISR) is a widespread scheme to enhance the yield of the memorybased product.

1.1 Background of ATPG Technology:

Automatic Test Pattern Generation (ATPG) is a critical process in the design and testing of integrated circuits (ICs), including System-on-Chip (SoC) designs. ATPG is used to generate test patterns that help detect manufacturing defects in semiconductor devices. These test patterns are applied to the chip using test equipment, and the responses are analyzed to identify potential defects. System-on-Chip (SoC) designs are highly complex, integrating multiple functionalities such as processors, memory, interfaces, and custom logic into a single chip. Given this complexity, ensuring manufacturing reliability is a challenge.

- ATPG tools use fault models (e.g., stuck-at faults, transition faults) to identify potential defects.
- The ATPG tool generates a set of test vectors that can detect the modeled faults.
- The generated test patterns are simulated to verify their effectiveness in detecting faults.
- The patterns are applied to manufactured chips using Automatic Test Equipment (ATE).

1.2 Existing Solutions and Their Limitations:

- Optimal Method for Test and Repair Using Redundancy Mechanism

A study by Alnatheer and Ahmed proposes a BISR scheme that utilizes spare rows and columns to achieve an optimal repair rate with minimal area overhead. The method employs a March-sift algorithm capable of detecting a wide range of faults, including stuck-at faults (SAF), transition faults (TF), address decoder faults (ADF), coupling faults (CFs), neighborhood pattern-sensitive faults (NPSFs), write disturb faults (WDFs), read disturb faults (RDFs), and data retention faults (DRDFs). The study indicates that increasing the allocation of spare rows and columns enhances the repair rate but also increases area overhead.Limitations:

- Area Overhead: While effective in fault coverage, the approach incurs increased area overhead as redundancy increases, which may not be suitable for all SoC designs..

- Built-In Redundancy-Analysis Scheme for RAMs with Two-Level Redundancy

Research presented at the IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems introduces an efficient Built-In Redundancy-Analysis (BIRA) scheme for RAMs incorporating two levels of redundancy: spare rows, spare columns, and spare words. The proposed BIRA scheme achieves a repair rate comparable to exhaustive search methods but with significantly lower hardware overhead. For instance, the hardware overhead for an 8K 64-bit RAM with specified redundancy is approximately 2%. Limitations:

Complexity in Redundancy Allocation: Implementing two-level redundancy requires careful management and may introduce design complexity.

- Built-In Self-Repair Architecture for SRAM Using Redundancy

BISR architecture for SRAM that includes a Built-In Self-Test (BIST) module and a BIRA module. The BIST employs March tests to detect faults, while the BIRA uses redundancy analysis algorithms to allocate spare rows for repair. The approach aims to minimize power consumption and area penalty, making it suitable for high-performance applications. Limitations:

Limited to Specific Fault Models: The effectiveness of the proposed BISR architecture may be constrained by the types of faults it can detect and repair, potentially limiting its applicability in diverse fault scenarios

1.3 Overview of the Paper:

Memory testing and repair are critical in modern System-on- Chip (SoC) designs due to the increasing density of embedded

memories. The presence of defects in memory cells can significantly impact chip performance and yield. This

project aims to develop an optimal approach for testing and repairing memories using a redundancy mechanism, ensuring improved reliability, fault coverage, and cost-effectiveness.

The paper is structured as follows:

- Section 2: Project Objectives
- Section 3: Key Components in the Project
- Section 4: Tools and Technologies
- Section 5: Challenges and Solutions
- Section 6: Expected Outcomes
- Section 7: Proposal and Main Thing of the project
- Section 8: Conclusion and Future Scope

2. OBJECTIVES

The primary objective of this project is to develop an efficient and optimized memory testing and repair system for System-on-Chip (SoC) designs by leveraging a Built-In Self- Test (BIST) and Built-In Self-Repair (BISR) framework. The project aims to detect, diagnose, and repair memory faults using an adaptive redundancy allocation mechanism that optimally utilizes spare rows, columns, and blocks to enhance chip reliability while minimizing area and power overhead. A March- based fault detection algorithm will be implemented to identify common memory defects such as Stuck-at Faults (SAF), Transition Faults (TF), Address Decoder Faults (ADF), Coupling Faults (CF), Retention Faults, and Write Disturb Faults (WDF). The system will incorporate a Built-In Redundancy Analysis (BIRA) unit, which will employ hybrid search techniques (a combination of heuristic and exhaustive search methods) to accelerate redundancy allocation decisions, reducing overall test and repair time. The repair strategy will ensure maximum yield improvement by efficiently mapping defective memory cells to available redundant resources without incurring excessive hardware costs. Additionally, the project will integrate Design-for-Testability (DFT) features, such as scan chains and compression-based testing, to facilitate seamless fault detection and improve testing efficiency. A low-overhead repair controller will be developed to automate the decision-making process for redundancy allocation, reducing computational complexity.

Moreover, AI-based prediction models may be explored to optimize fault analysis and repair strategy selection, further reducing test costs and increasing accuracy. The solution will be designed with a focus on scalability, ensuring it remains applicable to next-generation SoC architectures with higher memory densities and more complex fault scenarios. To validate the effectiveness of the proposed approach, extensive simulations will be conducted using industry-standard tools such as ModelSim, Cadence, or Synopsys, followed by prototyping on FPGA or ASIC platforms. Ultimately, this project seeks to enhance SoC reliability, improve manufacturing yield, reduce test time, and optimize hardware efficiency while maintaining cost-effectiveness.

3. KEY COMPONENTS

A. Memory Testing Mechanism

1. Fault Models: The system will target common memory faults such as:

o Stuck-at Faults (SAF)
o Transition Faults (TF)
o Address Decoder Faults (ADF)
o Coupling Faults (CF)
o Retention Faults

2. Test Algorithm: A March-based test algorithm will be implemented to detect faults efficiently.

B. Memory Repair Mechanism

1. Redundancy Mechanism:

o Incorporation of spare rows, columns, and blocks for defect repair.
o Optimization of redundancy allocation to minimize hardware overhead.

2. Built-In Redundancy Analysis (BIRA):

o A redundancy analysis algorithm will be developed to allocate spare resources efficiently.
o A hybrid search approach (heuristic + exhaustive search) will be used to speed up repair decisions.

C. Design-for-Testability (DFT) Integration

- Integration of Scan Chains and BIST to facilitate testing.
- Implementation of a low-overhead repair controller for automated defect handling.

4. Tools and Technologies

This project will utilize Hardware Description Languages (HDL) such as Verilog or VHDL for designing and implementing the memory testing and repair mechanism. Simulation and verification tools like ModelSim, Cadence, and Synopsys will be used to analyze functionality, fault coverage, and performance.

FPGA prototyping with platforms like Xilinx or Intel FPGA will be conducted to validate the design before ASIC fabrication. Additionally, Design-for- Testability (DFT) tools will be integrated to implement scan chains, BIST, and compression-based testing for efficient fault detection. AI-based machine learning models may also be explored to optimize redundancy allocation strategies. The combination of these tools will ensure accurate, scalable, and cost-effective memory testing and repair in SoC designs.

Hardware Description Language (HDL):

Verilog/VHDL for implementation. Simulation Tools: ModelSim , Cadence, or Synopsys for verification.

FPGA/ASIC Prototyping: Testing on FPGA platforms before ASIC fabrication.

5. CHALLENGES AND SOLUTIONS

High Area and Power Overhead

• Challenge: Implementing Built-In Self-Test (BIST) and Built-In Self- Repair (BISR) mechanisms in SoC designs can introduce significant area and power overhead, which may impact overall chip efficiency.

• Solution: To minimize overhead, an optimized redundancy allocation strategy will be used, ensuring only necessary spare rows/columns are allocated for repair. Additionally, lightweight test algorithms and low-power design techniques will be incorporated to reduce the impact on power consumption.

2. Complexity in Repair Decision-Making

• Challenge: Analyzing defective memory cells and selecting optimal redundancy configurations in real-time can be computationally intensive.

• Solution: A hybrid search approach combining heuristic methods and exhaustive search will be implemented to strike a balance between efficiency and accuracy. AI-based predictive models may also be explored to further optimize repair decision- making.

3. Increased Testing Time

• Challenge: Running comprehensive fault detection algorithms for large SoC designs can significantly

increase test time and production costs.

• Solution: To overcome this, compressed testing techniques such as scan compression and adaptive test pattern generation will be implemented to reduce the number of test vectors while maintaining high fault coverage.

4. Redundancy Resource Wastage

• Challenge: Fixed redundancy allocation can lead to underutilization of spare rows/columns, resulting in unnecessary hardware overhead.

• Solution: An adaptive redundancy allocation mechanism will be developed, which dynamically assigns redundant memory blocks based on fault distribution and severity, maximizing repair efficiency.

5. Difficulty in Testing Emerging Fault Models

• Challenge: Traditional March-based algorithms are effective for common faults like stuck-at and transition faults but may not fully cover advanced fault models such as retention faults, write disturb faults, and multi-bit upsets (MBUs) in newer memory technologies.

• Solution: Hybrid fault detection techniques, combining March tests with pattern-sensitive and machine-learning-based detection methods, will be integrated to improve fault detection accuracy for emerging memory defects.

6. Scalability for Future SoC Designs

• Challenge: As SoC designs evolve with higher memory densities and smaller nodes, conventional redundancy mechanisms may not scale efficiently.

• Solution: The project will focus on scalable architectures by designing a modular and hierarchical testing and repair system that can be easily adapted for future SoC generations.

6. EXPECTED OUTCOMES

6.1 Simulation Outputs in Xilinx Software

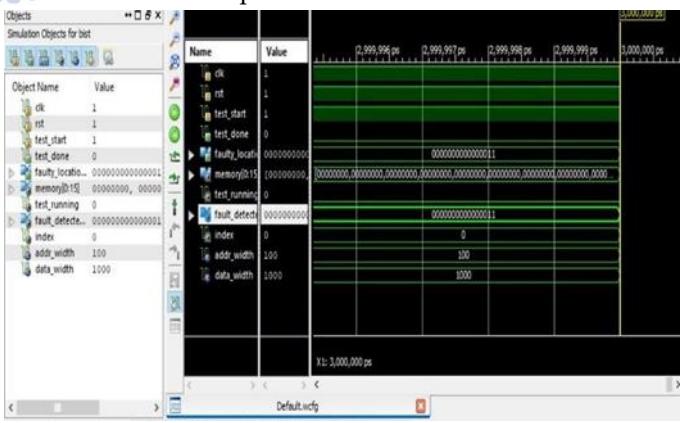

Fig 1: Expected output for BIST in Xilinx

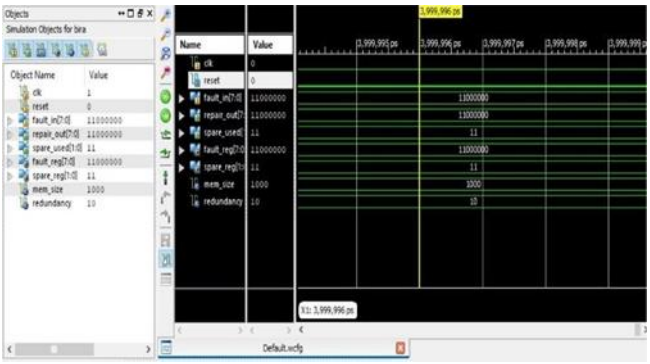

Fig 2: Expected Output for BIRA in Xilinx

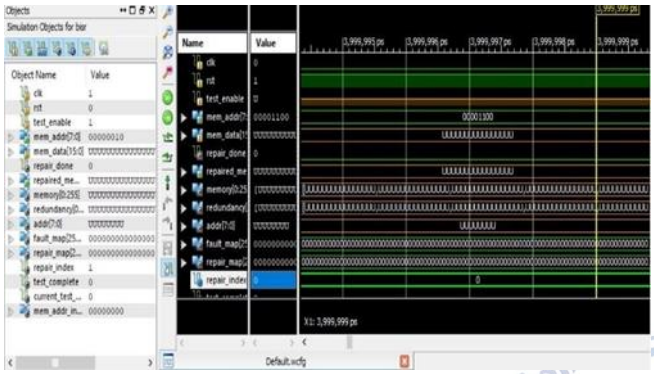

Fig 3 : Expected Output for BISR in Xilinx

6.2 RTL Schematic for Required Outcomes

The BISR block interfaces with the memory system through multiple input and output signals.

On the input side, the module receives mem_data[15:0], which represents the 16-bit memory data input, and mem_addr[7:0], which specifies the 8-bit memory address being tested or repaired. The clk (clock) signal synchronizes the module's operations, while the rst (reset) signal initializes or resets the system. The test_enable signal activates the BISR module, enabling memory testing and fault detection.

On the output side, the module provides repaired_mem_data[15:0], which delivers the corrected or repaired memory data after processing, ensuring reliable memory operation. Additionally, the repair_done signal indicates the completion of the repair process, signaling that faulty memory locations have been successfully mapped to redundant resources.

The BISR module enhances memory reliability and manufacturing yield by identifying defective memory cells and replacing them with redundant ones, reducing waste and improving system performance. This automated approach eliminates the need for external repair procedures, making it ideal for self-healing embedded memory systems in SoCs.

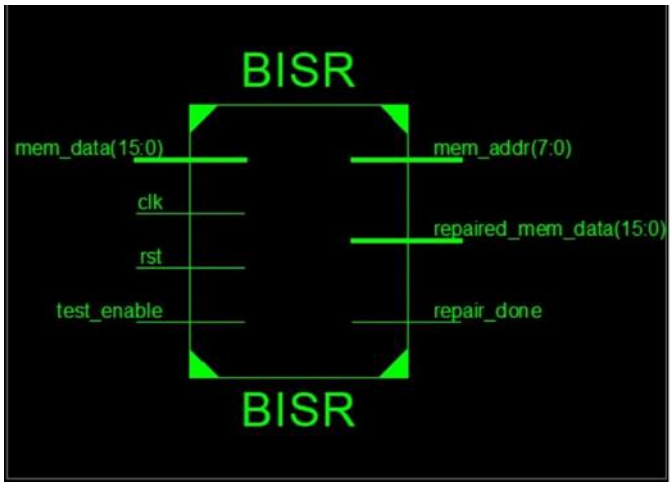

Fig 4: RTL Schematic on BISR Output

6.2.2 Overall Schematic Of The RTL Design

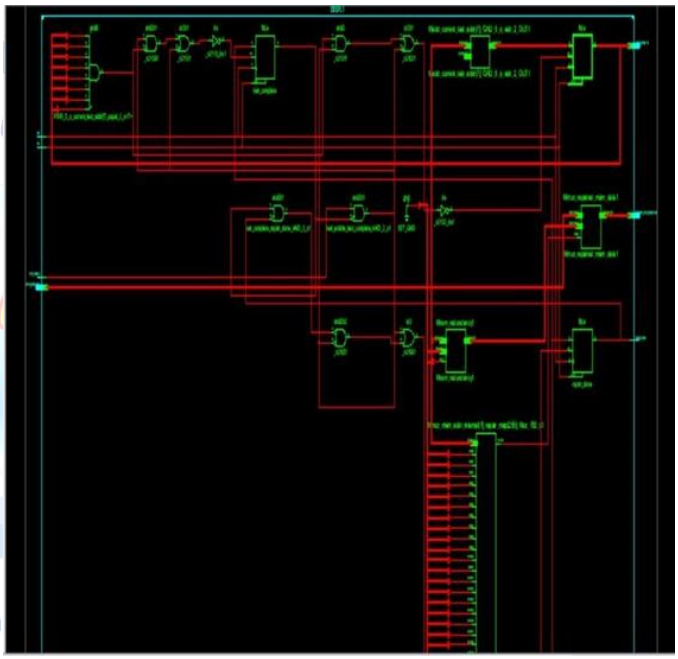

Fig 5: Proof of the BISR Schematic in the RTL

7. PROPOSAL AND MAIN THING OF THIS PROJECT

Stuck-at faults (SAFs) are among the most common faults in digital circuits, where a node is permanently stuck at logic '0' or '1', regardless of input conditions. This proposal

aims to design an Automatic Test Pattern

Generation (ATPG) method to detect whether Node X in the given circuit is stuck at '0' or not. The proposed approach will generate test patterns to ensure full controllability and observability of Node X.

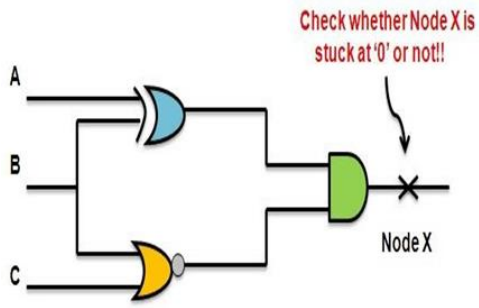

Fig 6: It Indicates That Proposal For SAF Detection

• The proposed ATPG-based approach will successfully detect stuck-at-0 faults at Node X.
• The BIST system will enable on-chip fault detection with minimal area and power overhead.
• The generated test vectors will ensure high fault coverage, reducing the need for external testing.

This method enhances reliability, yield, and manufacturability of digital circuits in modern SoC designs.

The below given flowchart represents a Built-In Self-Repair (BISR) process for detecting and repairing memory faults. This is highly relevant to the March C test algorithm, which is widely used for memory fault detection and analysis in embedded systems and SoCs. In the flowchart, Memory BIST initiates the testing process.

The March C algorithm is executed as part of BIST to systematically detect stuck-at faults, transition faults, and coupling faults.

This flowchart perfectly aligns with the March C testing methodology, making it a suitable reference for implementing fault detection and self- repair mechanisms in SoC memory systems.

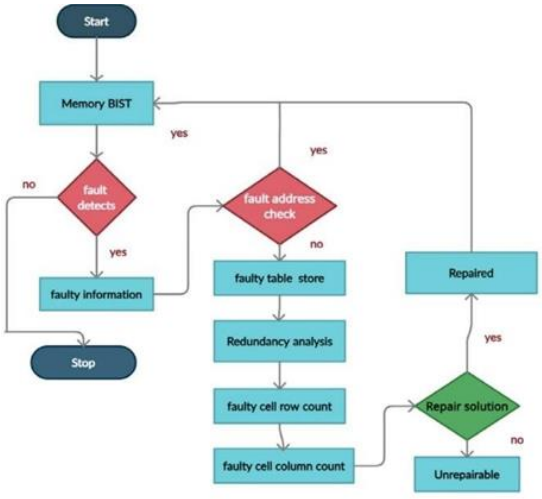

Fig 7: BISR Operational flow relate with the March C Test Algorithm

8. CONCLUSION AND FUTURE SCOPE

Future advancements in memory testing and repair mechanisms will focus on improving AI-driven fault diagnosis, adaptive self-healing memory architectures, and power-efficient error correction techniques. With the rise of 3D ICs and FinFET- based SoCs, new memory structures will require more advanced BIST and BISR strategies. Machine learning algorithms can be incorporated to predict failure patterns and dynamically allocate redundancy, improving memory resilience. Furthermore, integrating low-power BIST architectures will address power constraints in portable and IoT-based applications. Future research can also explore self-repairing memories that dynamically adapt to aging effects, improving longevity in mission-critical applications like automotive, aerospace, and healthcare. With the growing demand for high-speed and high-density memory architectures, innovative redundancy techniques will play a key role in next-generation SoC designs, ensuring high performance, reliability, and cost-effectiveness.

Memory testing and repair using redundancy in System-on-Chip (SoC) designs is a crucial step to ensure reliability, performance, and yield improvement. The Built-In Self-Test (BIST) and Built- In Self-Repair (BISR) methodologies, integrated with March algorithms, provide an efficient approach to detect and repair faults in embedded memory systems. The proposed approach leverages redundancy analysis to replace faulty memory cells with spare rows and columns, thereby enhancing the overall lifespan and fault tolerance of the memory system. Techniques like March C and

March algorithms effectively detect stuck-at, transition, and coupling faults, making the testing process comprehensive. The use of Automatic Test Pattern Generation (ATPG) further ensures high fault coverage while minimizing test time and power consumption. Implementing adaptive redundancy allocation enables better utilization of spare memory blocks, reducing yield loss and enhancing SoC manufacturing efficiency. This approach significantly reduces the need for external memory testing, thus lowering costs while maintaining high system .

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] Du, X.; Reddy, S.M.; Cheng, W.-T.; Rayhawk, J.; Mukherjee, N. At-speed built-in self-repair analyzer for embedded word-oriented memories. In Proceedings of the 17th International Conference on VLSI Design, Mumbai, India, 9 January 2004.

[2] Cheng, W.-T.; Hill, C.; Kebichi, O. Full-Speed Bist Controller for Testing Embedded Synchronous Memories. U.S. Patent Application No. US 2005/0066247 A1, 24 March 2005.

[3] Semico Research Corp., Semico: System(s)-on-a- Chip–A Braver New World. Semico Research. 2015. Available online: https: //semico.com/content/semico-systems-chip-%E2%80%93-braver-new-world (accessed on 9 July 2021).

[4] Zhang, G.; Yuan, Y.; Liang, F.; Wei, S.; Yang, C.-F. Low Cost Test Pattern Generation in Scan-Based BIST Schemes. Electronics 2019, 8, 314. [CrossRef]

[5] Peng, X.; Liu, R.; Yu, S. Optimizing Weight Mapping and Data Flow for Convolutional Neural Networks on Processing-in-Memory Architectures. IEEE Trans. Circuits Syst. I Regul. Pap. 2020, 67, 1333–1343. [CrossRef]