VALIDATION OF STANDARD CELLS AND MEMORY DESIGNS

A Thesis submitted in partial fulfillment of the requirement for the Award of the Degree of

MASTER OF TECHNOLOGY

in VLSI DESIGN

Submitted By

Rishi Tiwari

602362028

Under Supervision of

Dr. Sujit Kumar Patel

(Assistant Professor)

&

Dr. Pravindra Kumar

(Assistant Professor)

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT

THAPAR INSTITIUTE OF ENGINEERING AND TECHNOLOGY

(A DEEMED TO BEUNIVERSITY), PATIALA, PUNJAB

JULY 2025

STMicroelectronics INDIA Pvt. Ltd., Greater Noida, Uttar Pradesh 201308, India

CERTIFICATE

Date: 27th November,2024

This is to certify that Rishi Tiwari (ID: 362015), a student of M.Tech (VLSI Design), Thapar Institute of Engineering & Technology, Patiala, is undergoing internship program in TDP SQH group at STMicroelectronics Pvt. Ltd, Greater Noida.

Mr. Deepak Harod

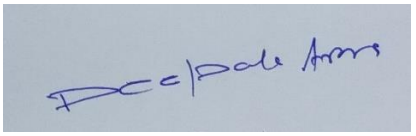
Technical Lead,

TDP SQH, ST Microelectronics India Pvt. Ltd.

CERTIFICATE

Date : 18th July 2025

This is to Certify that project entitled "VALIDATION OF STANDARD CELLS AND MEMORY DESIGNS" which is being submitted by Rishi Tiwari (University Registration No. 602362028) to the Department of Electronics and Communication Engineering, TIET, Patiala, Punjab, is a record of project work carried out by him under guidance and supervision of Mr. Deepak Kumar Arora (Senior Group Manager, TDP-SQH, STMicroelectronics India Pvt. LTD) , during a period from 7th August 2024 to 29th August 2025 . This is to certify that the candidate's above statement is correct to the best of my knowledge.



Mr. Deepak Kumar Arora
Senior Group Manager, SQH TDP
STMicroelectronics India Pvt. LTD

DECLARATION

I, **Rishi Tiwari** hereby declare that the work presented in this thesis entitled **"VALIDATION OF STANDARD CELLS AND MEMORY DESIGNS"** in partial fulfillment of the requirement for the award of degree of **Master of Technology (VLSI Design)** submitted at **Electronics and Communication Engineering Department**, Thapar Institute of Engineering & Technology (Deemed to be University), Patiala is an authentic record of work carried out in ST MICROELECTRONICS INDIA PVT. LTD., KNOWLEDGE PARK II, GREATER NOIDA as part of one year internship program from **8th August, 2024 to 25th July, 2025.**

Date : 10-June-2025

Rishi Tiwari
602362028

(Signature of Industry Mentor)
MR. DEEPAK KUMAR HAROD
Technical Lead, TDP

(Signature of College Supervisor)
DR. SUJIT KUMAR PATEL
Assistant Professor

(Signature of College Supervisor)
DR. PRAVINDRA KUMAR
Assistant Professor

ACKNOWLEDGEMENT

This thesis would not have been possible without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First and foremost, my sincere gratitude to **Dr. Sujit Kumar Patel** and **Dr. Pravindra Kumar** for his continuous support, motivation, immense knowledge. Their guidance has helped me in all the time of research and writing of the thesis.

Mr. Deepak Arora, Senior Manager of my team SQH, TDP, ST Microelectronics India Pvt Ltd, for accepting me as an intern in his team. I am grateful to him for his expertise, kind concern and consideration regarding my academic requirements.

Mr. Deepak Harod, Technical Lead, SQH, TDP, ST Microelectronics India Pvt Ltd, for his patience and steadfast encouragement, his constructive criticisms at different stages of my work were thought- provoking and he helped me focus on my ideas. I am deeply grateful to him for all discussions that helped me in understanding the technical details of my work.

Last but not the least, my family and dear friends. None of this would have been possible without their love. And the one above all of us, the omnipresent God for giving me the strength to plod on, thank you so much everyone.

ABSTRACT

In the realm of semiconductor design, standard cells are the fundamental building blocks used to create complex digital circuits. These cells, which include basic logic gates, flip-flops, and other essential components, are meticulously designed and characterized to ensure they meet specific performance, power, and area requirements. However, their role becomes even more crucial when integrated into memory designs, such as SRAM, DRAM, and Flash, where precision and reliability are paramount.

This report explores the methodologies and tools used for the validation of standard cells and memory designs. It addresses the critical need for comprehensive methodologies that efficiently validate these cells to ensure they meet necessary performance, power, and area requirements. The research highlights the challenges faced in achieving the desired efficiency and reliability of memory units, which are essential for the overall performance of electronic devices.

The proposed research approach encompasses multiple methodologies and tools, including RTL generation, simulation, synthesis, post-synthesis simulation, Design for Testability (DFT) insertion, post-DFT simulation, and Automatic Test Pattern Generation (ATPG). By integrating these techniques, the research aims to enhance the reliability and performance of memory units in electronic devices.

Despite significant advancements, several research gaps remain, such as the limited scope of existing methods, lack of standardization, and inadequate tools for integration. Addressing these gaps is crucial for guiding future research efforts and developing more robust methodologies and tools.

The objectives of this research are to develop advanced methodologies for the validation of standard cells, implement rigorous validation processes, and integrate with other designs and validation techniques to create a comprehensive framework that enhances the reliability and efficiency standard cells and memory designs.

LIST OF FIGURES

ABBREVIATIONS USED

ATPG	Automatic Test Pattern Generation
ATE	Automatic Test Equipment
ASIC	Application Specific Integrated Circuits
DRAM	Dynamic Random Access Memory
DFT	Design for Testability
EUV	Extreme Ultraviolet
OLT	On-line Testing
STA	Static Timing Analysis
VLSI	Very Large Scale Integration
BIST	Built-in Self Test

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION TO THE AREA OF WORK

In the ever-evolving landscape of modern technology, standard cells stand as the bedrock upon which the edifice of complex digital circuits is constructed. The design and development of memory units such as SRAM (Static Random Access Memory), DRAM (Dynamic Random Access Memory), and Flash memory are at the forefront of innovation in the semiconductor industry. At the heart of these sophisticated memory designs lie standard cells, which play a pivotal role in ensuring the efficiency, reliability, and scalability of modern electronic devices. Standard cells are the quintessential building blocks in the design of complex digital circuits, playing a pivotal role in the architecture of memory designs such as SRAM, DRAM, and Flash. These cells, which include logic gates, flip-flops, and multiplexers, are meticulously crafted to meet rigorous performance, power, and area specifications. In the context of memory designs, the characterization and validation of standard cells become even more critical, ensuring that these memory units operate with the highest levels of efficiency and reliability.

The significance of in memory designs is further highlighted by the increasing complexity and density of modern memory units. As technology nodes shrink and memory designs become more intricate, the need for precise and rigorous validation becomes even more critical. Advanced fabrication technologies, such as FinFET and EUV lithography, introduce new challenges that require sophisticated simulation and testing techniques to ensure the reliability and performance of memory units.

1.2 VALIDATION OF DESIGNS

Validation involves a rigorous suite of tests designed to ensure that the characterized memory cells function correctly within the actual design environment. This process encompasses functional verification to confirm that the cells perform the correct logical operations, Design Rule Checking (DRC) to ensure compliance with fabrication process rules, and Layout Versus Schematic

(LVS) verification to confirm that the physical layout matches the schematic design. Static Timing Analysis (STA) is conducted to ensure that the cells meet timing constraints, while power and thermal analyses are performed to ensure stable power distribution and effective heat dissipation. Signal integrity analysis addresses issues such as crosstalk and electromagnetic interference (EMI), ensuring reliable signal transmission within the memory design.

The need for validation in memory designs is underscored by several critical factors. Firstly, these processes ensure the functional correctness and reliability of memory units, which are integral to the performance of electronic devices. By rigorously testing memory cells under various conditions, validation and characterization mitigate the risk of functional errors and failures, thereby enhancing the robustness and longevity of electronic devices. This is particularly crucial in applications where reliability is paramount, such as in medical devices, automotive systems, and aerospace technology.

Moreover, validation contribute to the optimization of the design and manufacturing process. By identifying and addressing potential issues early in the design cycle, these processes reduce the likelihood of costly design iterations and manufacturing defects. This not only accelerates the time-to-market but also enhances the overall quality and yield of the final product. The integration of Design for Testability (DFT) structures and Automatic Test Pattern Generation (ATPG) further facilitates the detection and diagnosis of manufacturing defects, ensuring high fault coverage and efficient testing.

The significance of validation in memory designs is further highlighted by the increasing complexity and density of modern memory units. As technology nodes shrink and memory designs become more intricate, the need for precise and rigorous validation becomes even more critical. Advanced fabrication technologies, such as FinFET and EUV lithography, introduce new challenges that require sophisticated simulation and testing techniques to ensure the reliability and performance of memory units.

1.3 MOTIVATION OF WORK

The meticulous validation of standard cells and memory designs are pivotal processes that underpin the reliability, performance, and efficiency of memory components within integrated circuits. These processes are indispensable for several compelling reasons. Firstly,

ensuring reliability and robustness is paramount, as memory designs are integral to a myriad of electronic devices, ranging from consumer electronics to critical infrastructure systems. Through characterization, potential failure modes are identified and mitigated, guaranteeing that memory cells can endure variations in voltage, temperature, and manufacturing processes. Secondly, performance optimization is a crucial objective in memory designs.

Additionally, these processes help identify and resolve potential issues early in the design cycle, reducing the risk of costly design iterations and rework, ultimately shortening the time-to-market for new memory products. In a highly competitive industry, reducing time-to-market can be a significant advantage. Lastly, rigorously characterized and validated standard cells can be reused across multiple designs and projects, saving time and resources while ensuring consistency and reliability across different memory products. This reusability enables designers to leverage proven components, reducing the risk of introducing new errors and improving overall design quality. In conclusion, the characterization and validation of standard cells in memory designs are essential for ensuring the reliability, performance, and efficiency of memory components. These processes provide the necessary data to optimize designs, facilitate integration, support advanced technology nodes, reduce time-to- market, and enhance design reusability, ultimately enabling designers to create robust and high- performing memory solutions that meet the demands of modern electronic devices.

CHAPTER 2
LITERATUE REVIEW

The survey of work done by different individuals in the field of VLSI design has been highlighted in this chapter. The several approaches and difficulties in validating and characterizing standard cells will be covered in this chapter. Furthermore, we will talk about standard cells, memory types, testability design, and associated literature. We will determine the appropriate methodology for the execution of validation and characterization of standard cells in memory architectures based on the literature analysis.

2.1 STANDARD CELL

Standard cells are a fundamental concept in VLSI, revolutionizing integrated circuits (ICs) design and fabrication. These modular building blocks form the foundation of efficient and scalable chip designs, offering an ideal balance between flexibility and performance. By using pre-designed and tested components such as logic gates, designers can focus on higher-level design, speeding up the IC development process while maintaining consistency in performance. Standard cells represent a transformative paradigm in VLSI design, offering a versatile and efficient approach to building complex digital circuits. Understanding their structure, advantages, and integration into the design flow is pivotal for VLSI engineers aiming to create high-performance and scalable semiconductor devices.

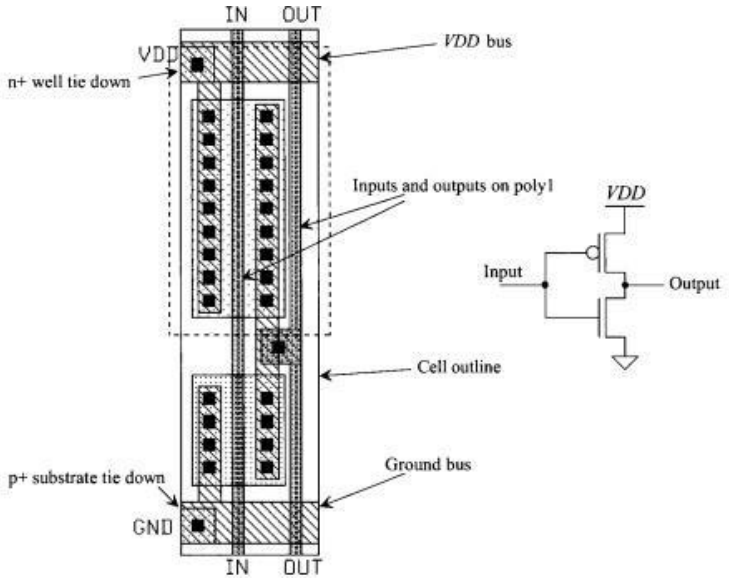


Figure. 2.1 Standard Cell layout of an CMOS inverter

2.2 MEMORY

Memory in semiconductor technology represents a cornerstone of modern electronic systems, embodying the intricate and sophisticated mechanisms that store and retrieve data. These memory units, encompassing various types such as SRAM (Static Random Access Memory), DRAM (Dynamic Random Access Memory), and Flash memory, are integral to the functionality and performance of a vast array of devices, from ubiquitous smartphones and laptops to advanced supercomputers and embedded systems.

An array of memory cells makes up a memory model's architecture. It is a two-dimensional array of memory cells. An address bus feeds a whole address to a column and row decoder in memory. Based on the address, the row decoder decodes the address and activates the rows. In a similar manner, the Column activates related column cell arrays by decoding the address from the bus. The additional memory cell is linked to a sense amplifier, which detects, amplifies, and transmits the data held in the memory cell array.

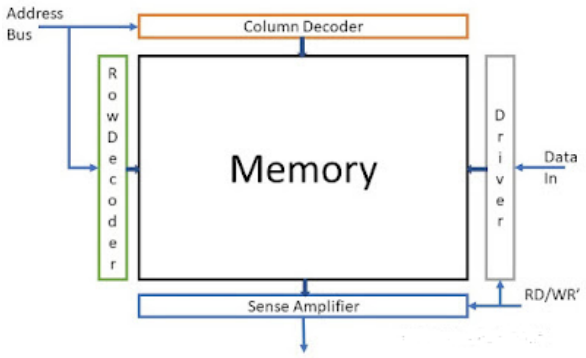


Figure. 2.2 Basic Memory Architecture

The intended data can either be written into the memory or retrieved out of it, as the memory model above illustrates. To carry out the memory operations, the special circuitry receives the Read or Write enable signal.

An essential task is to check the memory to see if the right data is being read out or written there. A sense amplifier is essential for boosting data while taking logic 1 and logic 0 thresholds into account. Calculating the number of clock cycles needed to write data into the cell and read data out of the cell is one way to assess memory performance.

2.3 DESIGN FOR TESTABILITY

Design for testing, or design for testability (DFT), is a term used to describe IC design methodologies that integrate testability features into a hardware product design. The extra features facilitate the development and application of manufacturing tests to the designed hardware. The purpose of manufacturing tests is to ensure that the product hardware is free of manufacturing defects that can jeopardize its correct functioning.

For certain goods, tests can be used for hardware maintenance in the customer's environment in addition to being used at different phases of the hardware manufacturing process. The tests are frequently conducted by test programs that operate on automatic test equipment (ATE) or, in the case of system maintenance, within the built system itself. In addition to detecting and indicating the presence of defects (i.e., the test fails), tests may also be able to capture diagnostic data on the kind of test failures experienced. The diagnostic information can be used to identify the failure's cause.

In other words, using the identical patterns, the response of vectors (patterns) from a good circuit is compared to the response of vectors from a DUT (device under test). If the response is the same or matches, the circuit is working correctly. Otherwise, the circuit is not manufactured as intended.

In addition to serving as an interface for test application and diagnostics, DFT is crucial to the creation of test programs. The implementation of suitable DFT principles and recommendations makes automatic test pattern generation, or ATPG, much simpler. The semiconductor industry's ASIC Design flow cycle is seen in Fig. 2.3.

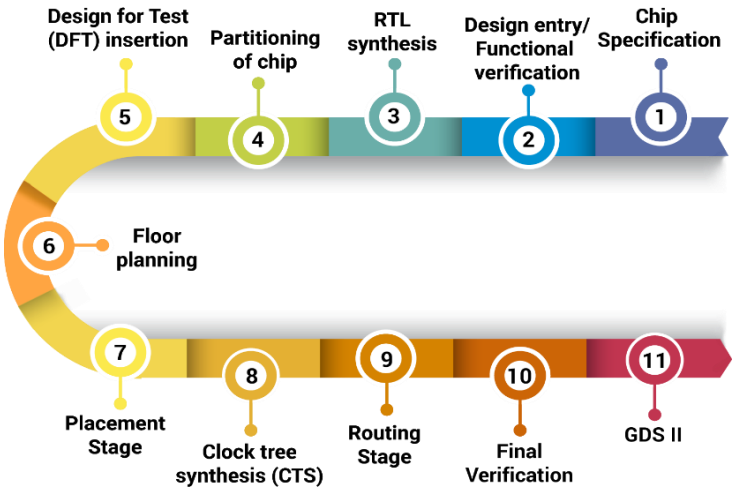


Figure 2.3 ASIC Design Flow

2.4 ATPG

ATPG, which stands for automatic test pattern generation and automatic test pattern generator, is an electronic design automation method or technology that aims to identify an input (or test) sequence that, when applied to a digital circuit, enables automatic test equipment to distinguish between the correct circuit behavior and the faulty circuit behavior caused by defects. The resulting patterns can be used to test semiconductor devices after they are created or to assist in determining the reason of failure. The effectiveness of ATPG is measured by the number of identified modeled faults, or fault models, and the number of patterns that are produced. These metrics frequently display test quality (higher with more fault detections) and test application time (higher with more patterns).

ATPG efficiency is another important factor that depends on the fault model being studied, the type of circuit being tested (full scan, synchronous sequential, or asynchronous sequential), the level of abstraction used to represent the circuit being tested (gate, register-transfer, switch), and the required test quality. Figure 2.4 below illustrates the ATPF basic tool flow.

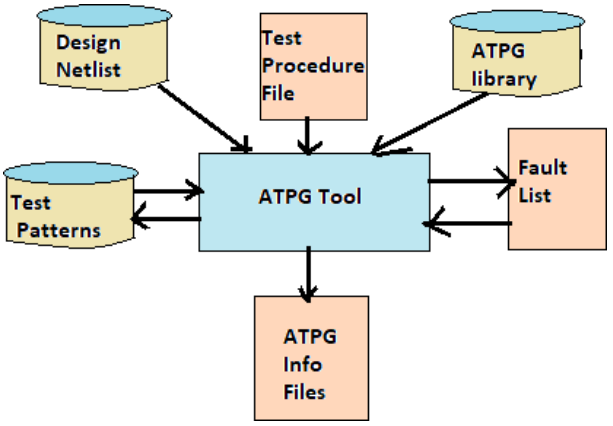


Figure 2.4 ATPG Tool Flow

2.5 RELATED LITERATURES

In [1], Gordon Moore talked on how electronics equipment needs to get smaller and more complex in order to fit in a smaller space. This is sometimes referred to as the well-known Moore's law, which states that an integrated circuit's (IC) transistor count will double roughly every two years. As a result, designs become increasingly complex [2] and there are numerous design-related issues. In order to achieve very high densities in VLSI (Very Large-Scale

Integration) circuits, the semiconductor industry has achieved a number of interesting advancements [3]. To advance with the more complicated designs, design engineers must create new methods and approaches.

A crucial step in the chip design and electrical design automation (EDA) process is memory built-in self-testing (BIST). To test for the appropriate fault models, a BIST tool must comprehend memory at the topological and layout levels. For the BIST[4] to have the least effect on the chip area and to be the most user-friendly for the chip designer, it must also be completely integrated into the overall EDA flow. Millions of transistors make up today's intricate and dense SoC architectures. For testing, these designs use a sophisticated DFT architecture. When silicon is brought up, thousands of DFT patterns are used.

A crucial step in the chip design and electrical design automation (EDA) process is memory built-in self-testing (BIST). To test for the appropriate fault models, a BIST tool must comprehend memory at the topological and layout levels. For the BIST[4] to have the least effect on the chip area and to be the most user-friendly for the chip designer, it must also be completely integrated into the overall EDA flow. Millions of transistors make up today's intricate and dense SoC architectures. For testing, these designs use sophisticated DFT architecture. When silicon is brought up, thousands of DFT patterns are used.

A crucial step in the chip design and electrical design automation (EDA) process is memory built-in self-testing (BIST). To test for the appropriate fault models, a BIST tool must comprehend memory at the topological and layout levels. For the BIST[4] to have the least effect on the chip area and to be the most user-friendly for the chip designer, it must also be completely integrated into the overall EDA flow. Millions of transistors make up today's intricate and dense SoC architecture. For testing, these designs use sophisticated DFT architecture. When silicon is brought up, thousands of DFT patterns are used.

The achievement of consistent, high-quality outputs across the whole EDA tool chain is one of the primary challenges in the design of contemporary microelectronic systems. This research demonstrates the significance of continuous tool collaboration for the quality of the outcome using a case study including sequential logic synthesis tools. The paper compares various alternative logic synthesis flows, such as the conventional flow utilizing JEDI and SIS, with a novel uniform and consistent information-driven logic synthesis approach [7].

The experimental study shows that the new uniform method and the new information-driven logic synthesis tools produce circuits that are significantly superior than those from the previous flows. Compared to the circuits from the traditional flow, the information-based synthesis flow generated circuits that are, on average, 25% smaller and 30% faster.

The benefits brought about by contemporary microelectronic technology cannot be fully utilized because of flaws in the circuit synthesis techniques employed in today's CAD tools. The problems and specifications of circuit synthesis for nano CMOS technologies are examined in this work, along with our novel circuit synthesis method [8] that meets these specifications. The novel technology eliminates the primary drawbacks of all existing synthesis methods and differs significantly from them. The experimental findings show that it creates circuits that are incredibly quick, small, and power-efficient. However, there are numerous other significant benefits of the new technology that are covered in the study.

Digital design tools are becoming more and more common in contemporary design as a result of advances in science, technology, and computer and digital technology. Through the dynamic simulation of things and the computation of influencing factors, computer simulation may efficiently solve prospective design challenges and has emerged as a significant design research approach. This paper addresses the use of simulation techniques from the standpoint of how computer simulation and contemporary design interact, incorporates simulation techniques [9] into contemporary design production, and methodically explains the usefulness of structural, morphological, and functional simulation in design.

It is commonly known that Cell-Aw's Test Methodology improves the quality of ATPG testing in today's SOC. By focusing on particular shorts and open flaws inside standard cells, it gets around the drawbacks of conventional fault models and significantly lowers the DPPM level. However, the drawbacks of Cell-Aware ATPG include a pattern volume that is 4–6 times larger than that of traditional ATPG, which results in an increase in both test volume and overall test duration [10]. By taking into account only pertinent complicated library cells for cell-aware internal defects out of all library cells, a novel approach to optimizing Cell-Aware ATPG pattern volume is presented in this study without significantly sacrificing test quality.

DFT methods are suggested for implementing ECC circuitry on memory macros without the need for extra testing. To evaluate the entire ECC system using standard memory BIST and a traditional test sequence, a new way for designing a Hamming code[11] matrix is employed. The suggested ECC approaches are shown through hardware characterisation using 90nm technology and implemented in a 512Kb SRAM macro.

These days, the primary obstacle to the development of effective hardware accelerators is

memory synthesis. A design methodology for effectively and automatically implementing memory access in High-Level Synthesis is presented in this work. Specifically, the method begins with a set of design constraints, including the memory addresses where some of the data are kept, and a behavioral specification (written in pure C). In order to create the appropriate architecture, the technique categorizes which variables can be assigned to the various modules either internally or externally [12], completely supporting a variety of C features including array accesses, function calls, and pointer arithmetic. Additionally, it makes it possible to parallelize the accesses when the memory address is known at compile time, which makes the specification run more efficiently.

For the embedded SRAM array, an in-array Build-In Self-Test (BIST) scheme is suggested. The pattern generator is implemented using the linear feedback shift register (LFSR), and the response compactor is implemented using the single/multiple-input signature register (SISR/MISR) [13]. In comparison to the traditional LFSR/MISR BIST, the suggested BIST architecture uses half as many transistors and takes up half as much space. It can also be tested. If the suggested BIST system is used, the traditional BIST circuit outside the SRAM array is not required. To test the other portion of the memory array, the suggested BIST scheme can apply other March sequences.

Today's ASICs have more than 10 million devices, therefore meeting deadlines for ASIC design projects requires meticulous planning. In this situation, there are two issues. The first involves estimating the number of man-months needed for a project while keeping in mind the ASIC design flow that will be used to carry it out. Making small adjustments to the design flow to shorten the project's completion time is the second issue. In a theoretical framework, we examine these two issues. A model called the hierarchical concurrent flow graph (HCFG) model is built from a textual description of the design flow in order to capture the inherent hierarchy and concurrency in the execution of an ASIC design flow.

a flow. We may (a) rapidly estimate the project execution time using the HCFG model[14] and (b) examine the impact of adding AND and OR concurrency to the flow to shorten the execution time. We provide two examples to demonstrate the application of the potent estimation technique. In the first case, AND concurrency is used in a back-end flow, and in the second example, OR concurrency is used in a software design flow.

The creation of design data pertaining to macrocells, such as graphic symbols, logic simulation models, input-to-output path delays, and timing parameters for the ASIC CAD system, is crucial to a successful ASIC business's ability to launch a new product. This allows ASIC customers to start designing with the new ASIC product. Considerations include creating logic models, switching from logic to circuit simulation, designing circuits with estimated parasitics,

verifying the physical architecture of the circuit, and encapsulating the macrocell design CAD flow.

Almost everyone in the industry is talking about full chip mixed-signal simulation these days [16], both in pre-layout and post-layout conditions, as many applications are moving from fully digital to mixed-signal and full chip simulation with parasitic components and IR drop analysis is becoming strictly required before going to silicon. A single mask set for either 90nm or 65nm technology is actually becoming more and more expensive at an exponential rate, surpassing one million dollars. Setting up a very thorough mixed-signal design pipeline that allows designers to reach the silicon securely and with the least amount of failure risk is strategically significant because of these characteristics. In this study, possible solutions have been discussed.

For ASIC/SOC designers, power minimization is increasingly a crucial design requirement. In order to fulfill power budgets for portable devices and to guarantee that the systems that these ASICs are used in meet their cooling and packaging expenses, it is essential to reduce both dynamic and leakage power [17]. Furthermore, an ASIC's manufacturing yield and dependability are greatly influenced by its power. The majority of automated power optimization solutions have historically concentrated on optimizations at the gate and physical levels. However, only by addressing power at the RTL and system levels can significant power reductions be achieved. At these levels, sequential clock gating, power gating, voltage/frequency scaling, and other micro-architectural approaches can be used to produce the sequential changes required to lower power and energy consumption.

Almost everyone in the industry is talking about full chip mixed-signal simulation these days [16], both in pre-layout and post-layout conditions, as many applications are moving from fully digital to mixed-signal and full chip simulation with parasitic components and IR drop analysis is becoming strictly required before going to silicon. A single mask set for either 90nm or 65nm technology is actually becoming more and more expensive at an exponential rate, surpassing one million dollars. Setting up a very thorough mixed-signal design pipeline that allows designers to reach the silicon securely and with the least amount of failure risk is strategically significant because of these characteristics. In this study, possible solutions have been discussed. We view the Electronic Design Automation (EDA) industry as a key enabler to bridge the gaps between pre-silicon and post-silicon validation and transfer the substantial quantity of intellectual property in pre-silicon tools to the post-silicon validation space.

Potential solutions to these issues are covered in the paper "Post-silicon validation challenges: How EDA and academia can help."

Miniaturization of transistor characteristics and the introduction of multi-core architecture are the results of rapid advancements in semiconductor process technology. The thermo-mechanical design problems [19] of thermal margining tools are presented in this study. We start by outlining the specifics of the CPU, chipset, and ASIC thermal margining head designs, along with the retention design for socketed or soldered down silicon as needed. Second, we show how the design optimization of the thermal head has benefited from the incorporation of CFD modeling and retention design technique. This thorough process made it possible to create and provide thermal tools with a predictable temperature margining range, enhance the quality of the products that were given to validation clients, and significantly reduce the tools' cost. Third, we outline these thermal tools' field problems, thermal performance deterioration, and failure modes. In order to address the decreasing KOV of silicon component placements across all market segments and enable increasing bus speeds, features, functionalities, and TDP/power density, we conclude by outlining the challenges we face and the technological advancements we require from the rest of the industry, particularly in TEC technology, in order to design small form factor thermal margining tools.

In order to show that the implemented circuit behaves as anticipated, post-silicon validation is a crucial stage in the design process. To help with post-silicon validation, design-for-debug[20] hardware is used because in-system controllability and observability are lacking. Numerous approaches have been put forth to both implement the design-for-debug hardware and evaluate the obtained debug data. The current methods to support post-silicon validation mainly rely on the data taken from the gate level circuit descriptions, even if the design entry is carried out at the register-transfer level. We predict that obtaining and processing circuit information at this level will get more challenging as design complexity increases. In this work, we provide a brief overview of the state of the art and talk about several potential research avenues that could use high-level circuit models to enhance the current solutions.

A versatile Memory Built-in Self-Test (MBIST) that may be readily adjusted to different memory configurations and user needs is presented in this study. Programming scripts are used to construct its RTL code, which eliminates the need for complicated compiler directives and produces easily readable code. The fundamental architecture[21] can be modified to accommodate various test schemes, such as sequential, which tests the memories one at a time,

or parallel, which tests all the memories simultaneously.

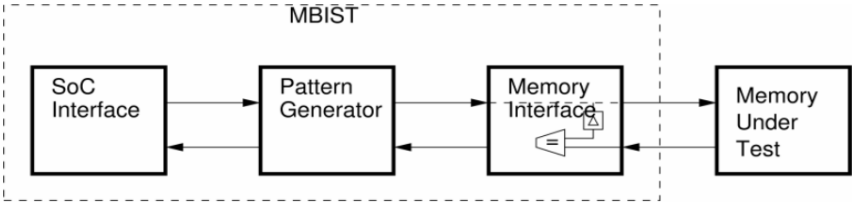


Figure 2.5 MBIST Architecture

Complex dependent memory architectures' structural complexity and testing difficulties are explained [22]. An isolation technique is offered to reduce the time cost and test logic overhead. To test the memory cells, bridging, and multi-port errors in intricate dependent memory architectures, a collection of unique memory test algorithms is created. To implement the created custom testing algorithms, a novel programmable memory BIST architecture has been described. The dependent memory structures can be tested at various phases of their construction and assembly using the suggested memory BIST architecture. The experimental findings show the area overhead of the various parts of the suggested BIST architecture for programmable memory.

2.6 RESEARCH GAPS

Despite significant advancements in the validation of complex standard cells and memory designs, several research gaps remain that need to be addressed validation and to enhance the reliability and performance of memory designs. Identifying these gaps is crucial for guiding future research efforts and developing more robust methodologies and tools.

2.6.1 LIMITED SCOPE OF EXISTING METHODS :

Current methodologies often focus on specific performance metrics (e.g., speed or power consumption) without providing a holistic approach that encompasses all critical parameters such as area efficiency, thermal behavior, and signal integrity.

2.6.2 LACK OF STANDARDIZATION :

There is a lack of standardized procedures for the validation of standard cells, leading to inconsistencies in performance metrics across different designs and fabrication technologies.

2.6.3 INADEQUATE TOOLS FOR INTEGRATION :

Existing tools do not adequately support the seamless integration of complex validation processes, resulting in increased complexity.

2.7 RESEARCH OBJECTIVES

2.7.1 OBJECTIVE 1 :

Develop an understanding of advanced methodologies for the validation of standard cells and memory to ensure they meet specific performance metrics.

2.7.2 OBJECTIVE 2 :

Implement rigorous validation processes to test the functionality of standard cells within memory designs under various conditions.

2.7.3 OBJECTIVE 3 :

Integrate automated validation techniques to create a comprehensive framework that enhances the reliability and efficiency of designs.

CHAPTER 3

RESEARCH METHODOLOGY

To address the problem of efficiently validating standard cells, we propose a detailed research approach that encompasses multiple methodologies and tools. This approach aims to ensure that standard cells meet the necessary performance, power, and area requirements, thereby enhancing the reliability and performance of memory units in electronic devices. Below Flowchart 3.1 illustrates a methodology used in this project.

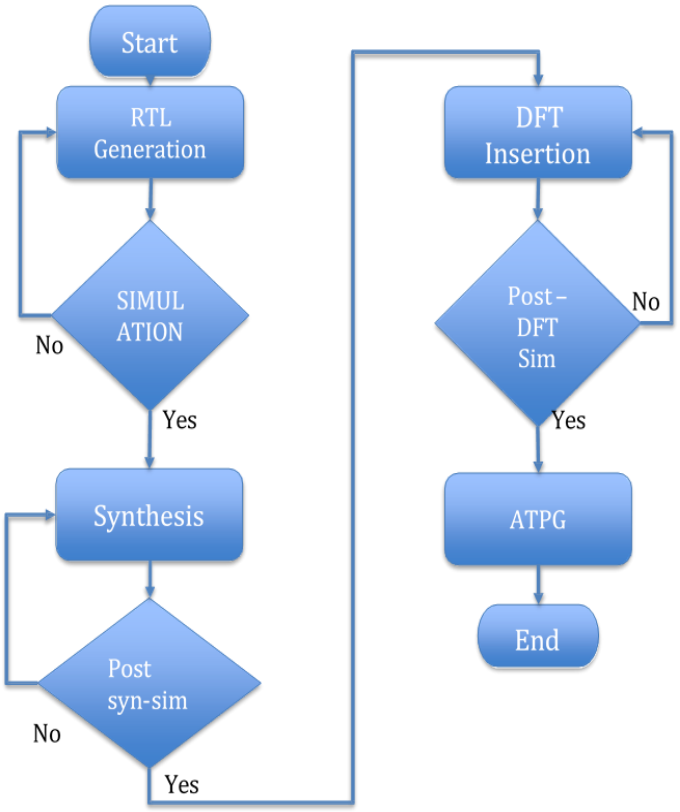


Figure 3.1 Proposed Methodology Flowchart

3.1 RTL GENERATION

Generate a RTL netlist and its testbench according to the memory design specifications with respect to standard cells through an automation script.

3.2 SIMULATION

Use Electronic Design Automation (EDA) tools to simulate the behavior of standard cells within memory designs before layout.

Tools: Utilize tools such as Cadence Xcelium.v.08.2024 , for detailed circuit simulations and analysis.

3.3 SYNTHESIS

Convert the high-level design description of the memory and standard cells into a gate-level netlist. Ensure the synthesized design meets the desired performance, area, and power specifications.

Tools: Use synthesis tools like Synopsys Design Compiler or Cadence Genus.

3.4 POST-SYNTHESIS SIMULATION

Perform simulations on the synthesized netlist to verify the functional correctness of the design. Identify and address any discrepancies between the high-level design and the synthesized netlist.

Tools: Use simulation tools like Cadence Xcelium.v.08.2024 .

3.5 DESIGN FOR TESTABILITY (DFT) INSERTION

Insert DFT structures into the design to facilitate testing and ensure test coverage. Implement scan chains, built-in self-test (BIST), and other DFT techniques.

Tools: Use DFT tools like Synopsys DFTMAX or Cadence Modus.

3.6 POST-DFT SIMULATION

Perform simulations on the design with DFT structures inserted to verify the integrity and functionality of the test structures. Ensure that the DFT insertion has not introduced any functional errors or timing violations.

Tools: Use simulation tools like Cadence Xcelium.v.08.2024 .

3.7 AUTOMATIC TEST PATTERN GENERATION (ATPG)

Generate test patterns to detect manufacturing defects in the standard cells and memory design. Ensure high fault coverage and optimize test patterns for efficiency.
Tools: Use ATPG tools like Synopsys TetraMAX.

To tackle the challenge of efficiently validating memory cells, we have developed a comprehensive research approach that incorporates various methodologies and tools. Our goal is to ensure that memory cells fulfill the required performance, power, and area specifications, ultimately improving the reliability and functionality of memory units in electronic devices. Some of the methodologies used in this project are summarized below :

3.8 DYNAMIC MEMORY TESTING (DMT)

Dynamic Memory Testing involves applying dynamic test patterns to validate the memory cells under various operational conditions. This ensures that the memory cells can handle different data patterns and access sequences without errors. The key steps include:

- Generating dynamic test patterns.
- Applying these patterns to the memory cells.
- Monitoring the output for any discrepancies.
- Analyzing the results to identify potential faults.

3.9 NON-ADDRESSABLE LOGIC (NAL)

Non-Addressable Logic focuses on testing memory cells that are not directly addressable. The methodology includes:

- Identifying non-addressable memory cells.
- Applying specific test patterns to these cells.
- Monitoring the output to detect any faults.
- Analyzing the results to ensure the reliability of non-addressable memory cells.

3.10 ACCESS TIME CHARACTERIZER (ATC)

Access Time Characterizer is used to measure and validate the access time of memory cells. The methodology includes:

- Generating access time test patterns.
- Applying these patterns to the memory cells.
- Measuring the time taken for read and write operations.
- Analyzing the results to ensure that the access times meet the required specifications.

3.11 BUILT-IN SELF-TEST (BIST)

Built-In Self-Test is a widely used method for memory validation. The key steps are:

- Designing the memory with built-in test circuits.
- Activating the BIST mode to initiate the test sequences.
- Generating and applying test patterns internally.
- Analyzing the memory's response to these patterns.
- Reporting any detected faults for further analysis.

By combining these methodologies—DMT, NAL, ATC, and BIST—we can achieve comprehensive memory validation, ensuring that memory cells meet the necessary performance, power, and area requirements. This approach enhances the reliability and performance of memory units in electronic devices.

CHAPTER 4
RESULTS & DISCUSSION

In this chapter, we discuss the validation of standard cells and memory design using advanced methodology.

To comprehend the design validation of standard cells, we examine a small design named FF_ARRAY. This chapter further elaborates on the steps and partial results involved in the complete validation flow, using a reference technology, referred to as RT18 technology. Outcome of this design validation cycle is discussed in next few sections.

4.1 RTL GENERATION

Initially, we generated an RTL netlist of the FF_ARRAY, incorporating RT24 library cells. Additionally, a testbench for block-level simulation was created using a Python script. The resulting output is illustrated in the Figure 6.1 below.

```
Output CSV=> /work/tcauto/FE_AREA/FF_ARRAY/FF_ARRAY_GENERIC_v1.5/RTL/FF_ARRAY_SPEC.csv
The time is: 14:59:49
The date is: 03/31/23
Project name :demo
Design name {module_name} :ff_demo
if you Want to generate RTL press Y else any other key :y
.D(D), .CP(CP), .R(RN_int), .TI(1'b0), .TE(1'b0), .Q(Q)
.D(D), .CP(CP), .R(RN_int), .TI(1'b0), .TE(1'b0), .Q(Q)
.D(D), .CP(CP), .R(RN_int), .TI(1'b0), .TE(1'b0), .Q(Q)
.D(D), .CP(CP), .TI(1'b0), .TE(1'b0), .SN(SN), .Q(Q)
Copying constant file in OUTPUT directory...

------------------------------------------------
RTL generated successfully in OUTPUT directory
------------------------------------------------

if you Want to generate block level testbench press Y else any other key :y
Testbench generation start...
 ...running...
if you Want to generate block level testbench for croll press Y else any other key :n
------------------------------------------------------
Block level Testbench generated successfully in VERILOG_TB/block directory
------------------------------------------------------
if you Want to generate top level testbench press Y else any other key :n
You skiped top level testbench
[dlhsx04727] RTL 52:-)
```

Figure 4.1 Console Output RTL Generation

4.2 PRE-SYNTHESIS SIMULATION

After successfully generating netlist with its block level testbench, we simulate the netlist for ensuring the functional correctness of our design. For this task, we used a simulator tool XCELIUMAGILE (developed by CADENCE Inc.). Checking the design functionalities after

mapping the RTL netlist with technology library using test patterns specific to functionality. In this pattern, after the number of DUTs repetition, the data is shifted as you can observe in following waveform also including console output snippet.

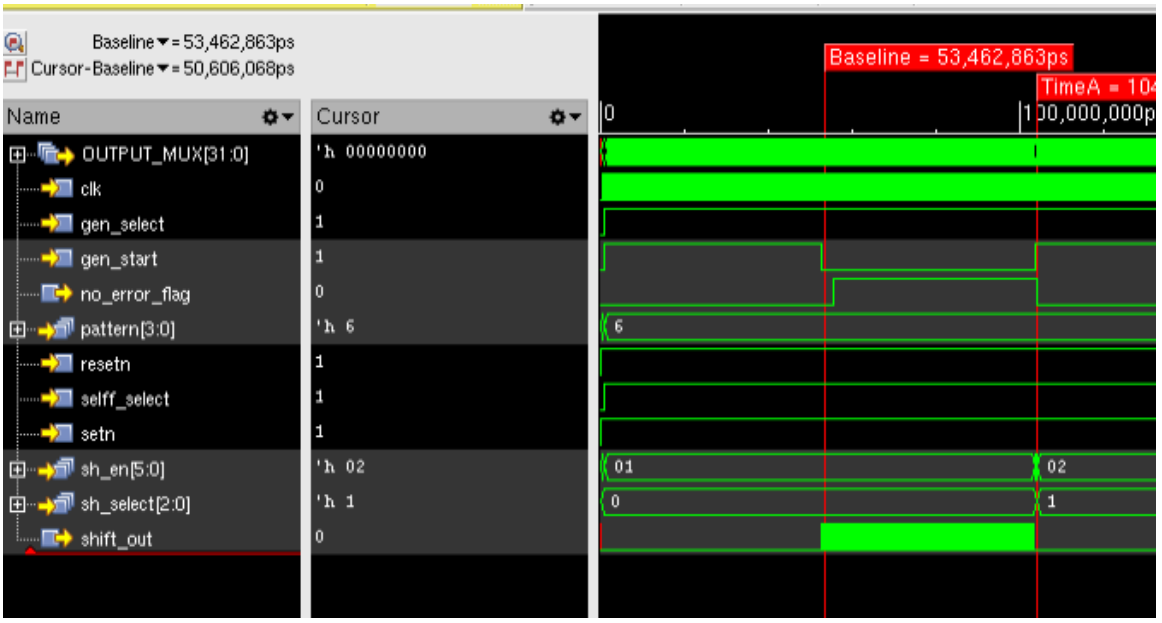


Figure 4.2 Simulation Waveform of Pat-gen pattern

```
Created default SHM database waves
xcelium> probe -create  -all -depth  all -shm -database waves -memories -packed 0 -unpacked 0 ;
xmsim: *W,PRHOPT: The design contains optimized signals. To improve simulation performance these signals are excluded.
Created probe 1
xcelium>
xcelium>
xcelium> run

*******************************************************************************
 TEST COMPLETED FOR functional_pat_gen with 49152 no_of_comparison & 0 ERROR
*******************************************************************************
Simulation complete via $finish(1) at time 618700 NS + 1
../RTL/VERILOG_TB/stimuli/func_pat_gen.v:117 $finish;
xcelium> date
Fri Nov 29 21:37:15 IST 2024
```

Figure 4.3 Console Output of Pre-synthesis simulation

Similarly, we can verify the other functionality including top pad functionality, shifter bypass and more.

4.3 SYNTHESIS

Synthesis is a crucial step in the design flow of digital circuits, particularly after the simulation of a netlist that has been created by mapping with technology. Now, we synthesize the RTL netlist into gate level netlist. Below snippet shows the synthesis of the Verilog has

been done and output files has been successfully created.

```
Information: Uniquified 1 instances of design 'CT_buffer_9'. (OPT-1056)
Information: Uniquified 1 instances of design 'CT_buffer_10'. (OPT-1056)
1
change_names -rules verilog_unicad -verbose -hier > ./REPORTS/change_name.rpt
rename_design -postfix _${DESIGN}  [remove_from_collection [get_designs *] [get_designs ${DESIGN}]] > ./REPORTS/rename_.rpt
#################################
###### write out NETLIST ########
#################################
write_script -format dctcl -nosplit -output ./NETLIST_OUT/${DESIGN}_synth_constraint.sdc
1
write -format verilog -hier -output ./NETLIST_OUT/${DESIGN}_post_synth.v
Writing verilog file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/SYNTH/NETLIST_OUT/FF_ARRAY_8T_L_NRPL_po
st_synth.v'.
1
sh rm -rf command.log default.svf filenames.log
exit

Memory usage for this session 802 Mbytes.
Memory usage for this session including child processes 802 Mbytes.
CPU usage for this session 878 seconds ( 0.24 hours ).
Elapsed time for this session 1046 seconds ( 0.29 hours ).

Thank you...
[1] 2756883
[dlhsx04725] SYNTH 37:-)
```

Figure 4.4 Terminal showing Synthesis Completion

4.4 POST-SYNTHESIS SIMULATION

Now after the post-synthesis file that was created requires a testing to verify functional correctness of our design. Again, perform the simulation and this step is known as **post-synthesis simulation.** For this step, we take different test pattern to ensure that all aspects of design work in good manner. In Figure 4.5, shows the OLT pattern assures functionality with no errors.

```
xcelium>
xcelium> run

*******************************************************************************
 TEST COMPLETED FOR olt pattern with 0 ERROR ***************
*******************************************************************************
Simulation complete via $finish(1) at time 1000700 NS + 0
../RTL/VERILOG_TB/stimuli/func_olt.v:19 $finish;
xcelium> date
Sat Nov 30 12:43:30 IST 2024
xcelium>
```

Figure 4.5 Console Output for OLT pattern

In the waveforms figures below, we can observe that after enabling OLT bit with pattern insertion no error bits is high, displaying our designs performs perfectly after the synthesis.

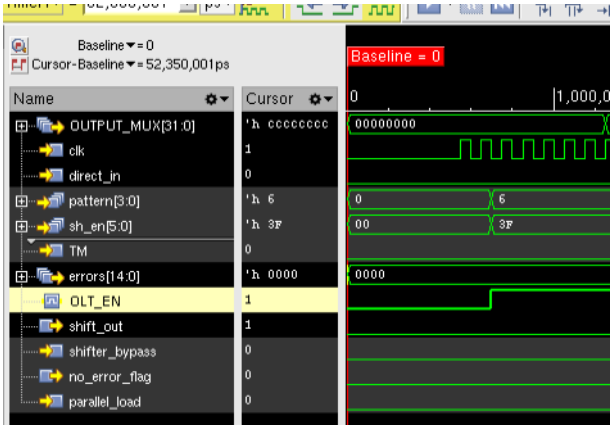


Figure 4.6 Simulation Waveform of OLT pattern-1

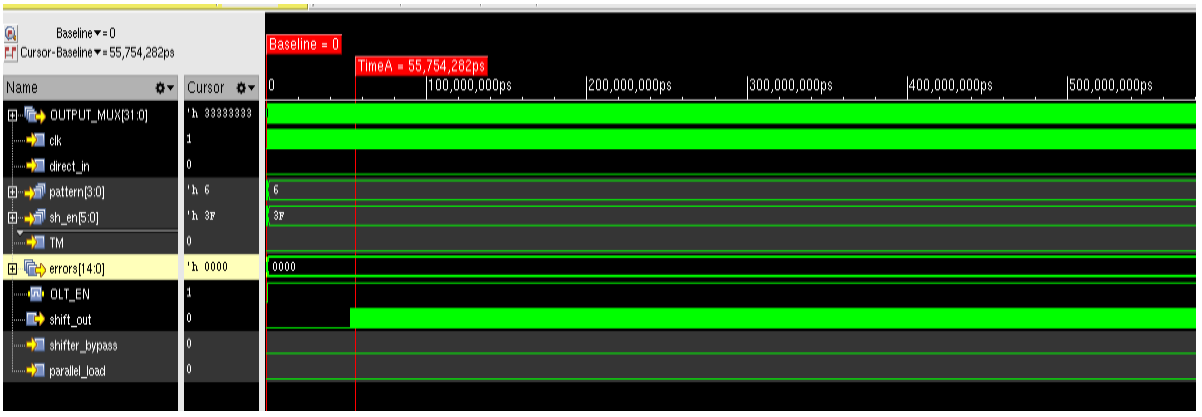


Figure 4.7 Simulation Waveform of OLT pattern-2

4.5 DESIGN-FOR-TESTABILITY

Now, we need to modify the design to make it more testable by inserting DFT structures. In this step scan chains were inserted in our design. Scan chains provide a mechanism to access and control the internal flip-flops and registers of a digital circuit. This is crucial for testing internal states that are not directly observable from the primary inputs and outputs. It also increases the controllability and observability of designs.

Figure 4.8 shows an terminals output showing DFT insertion in our design has been successfully completed.

```
   -test_mode_port TM \
   -cycles_per_clock 4 -chain_count 1
Accepted clock controller specification.
Current test mode is set to 'Internal_scan'.
Writing test protocol file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/DFT/NETLIST_OUT/Internal_scan.spf
' for mode 'Internal_scan'...
Current test mode is set to 'Mission_mode'.
Writing test protocol file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/DFT/NETLIST_OUT/Mission_mode.spf'
 for mode 'Mission_mode'...
Accepted insert_dft configuration specification.
Writing test model file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/DFT/NETLIST_OUT/FF_ARRAY_8T_L_NRPL_S
CAN.ctl'...
Writing test model file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/DFT/NETLIST_OUT/FF_ARRAY_8T_L_NRPL_S
CAN.ctlddc'...
Writing ddc file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/DFT/NETLIST_OUT/FF_ARRAY_8T_L_NRPL_SCAN.ctl
ddc'.
Writing verilog file '/work/P18_1/PROLIBP18_V6.0/TEAM/RISHI01/FE/RTL/FF_ARRAY/FF_ARRAY_8T_L_NRPL/DFT/NETLIST_OUT/FF_ARRAY_8T_L_NRPL_SCAN
.v'.
Warning: Verilog 'assign' or 'tran' statements are written out. (VO-4)

Memory usage for this session 945 Mbytes.
Memory usage for this session including child processes 945 Mbytes.
CPU usage for this session 671 seconds ( 0.19 hours ).
Elapsed time for this session 936 seconds ( 0.26 hours ).

Thank you...
[1] 4155436
[dlhsx04725] DFT 51:-)
```

Figure 4.8 Terminal showing DFT completion

4.6 POST-DFT SIMULATION

To verify the functionality of designs, simulation is very crucial after DFT insertion. Sometimes an unconnected node, register, caused malfunctions that needs to be addressed swiftly. To simulate the design, we execute SHIFTER BYPASS pattern, on DFT inserted netlist.

Figures below showing console output, waveforms for the shifter bypass pattern.

```
Created probe 1
xcelium>
xcelium>
xcelium> run

*******************************************************************************
 TEST COMPLETED FOR shifter_bypass with 49920 no_of_comparison & 0 ERROR
*******************************************************************************
Simulation complete via $finish(1) at time 318700 NS + 1
../RTL/VERILOG_TB/stimuli/func_SHIFTER_bypass.v:90 $finish;
xcelium> date
Sat Nov 30 14:01:00 IST 2024
xcelium>
```

Figure 4.9 Console Output of shifter bypass pattern

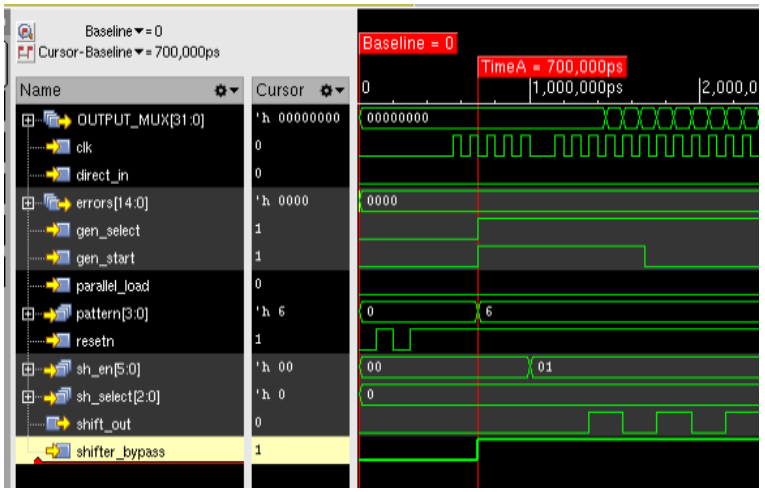


Figure 4.10 Simulation Waveform of shifter bypass pattern-1

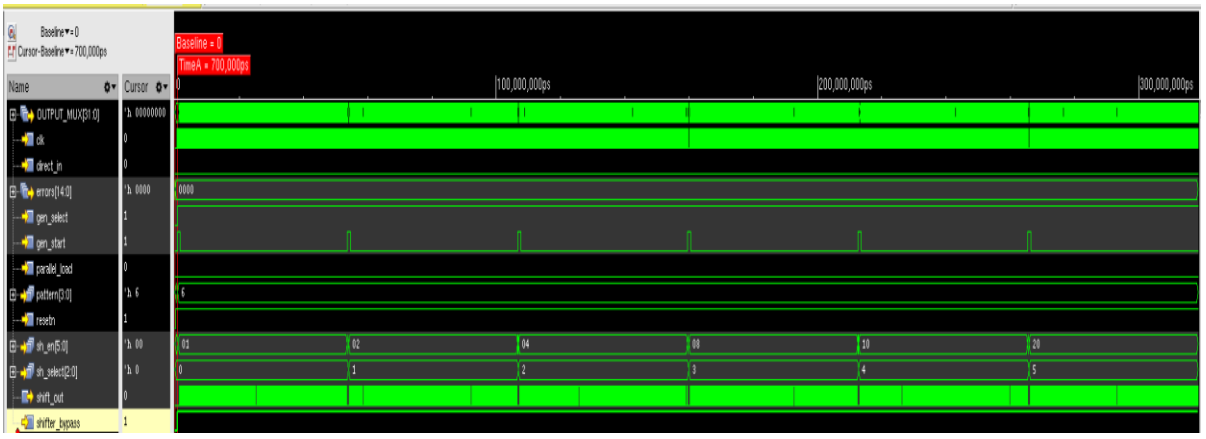


Figure 4.11 Simulation Waveform of shifter bypass pattern-2

4.7 ATPG

ATPG(Automatic test pattern generation) is used to generate test patterns to detect manufacturing defects in the standard cells and memory designs. Ensure high fault coverage and optimize test patterns for efficiency. After verifying the functionality of the DFT inserted file, we perform ATPG on the same file to analysis the test coverage. Figure 4.10, shows the log file output for the ATPG test coverage.

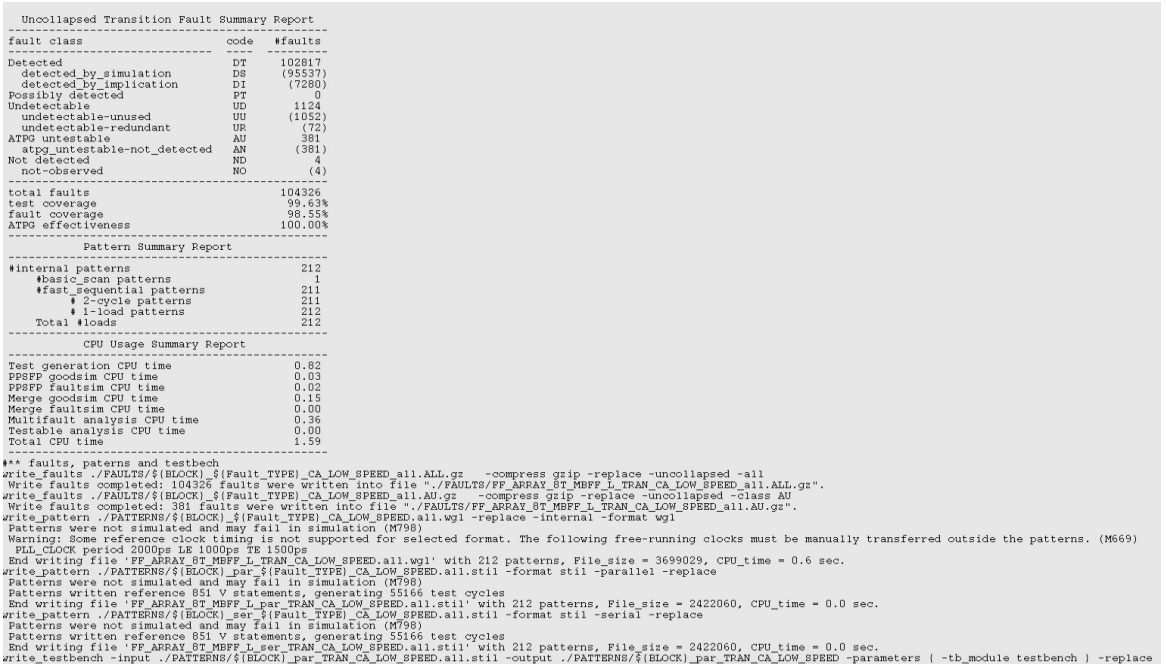


Figure 4.12 ATPG Fault Coverage

By systematically following steps such as RTL generation, pre-synthesis and post-synthesis simulations, synthesis, DFT insertion, and ATPG, we ensured the functional correctness, performance, and reliability of the memory cells. This rigorous approach not only validated the design but also optimized it for power and area requirements, thereby enhancing the overall reliability and performance of memory units in electronic devices. The detailed results and observations provide valuable insights for future design validation projects.

Now, to comprehend the validation of memory cells, we examine a small memory design. This chapter further elaborates on the steps and partial results involved in the complete validation flow, using a reference technology, referred to as RT18 technology. The outcomes of this design validation cycle are discussed in the following sections.

4.8 DIRECT-MEMORY TESTING

In this testing method, we perform read and write operations on both the nearest and the farthest memory locations within the memory array. This approach allows us to thoroughly observe and verify the functional correctness of the memory cells across the entire memory range. By testing these extreme locations, we can ensure that the memory operates reliably and accurately, regardless of the specific address being accessed. This comprehensive testing helps identify any potential issues related to data integrity, access times, or other functional aspects of the memory cells.

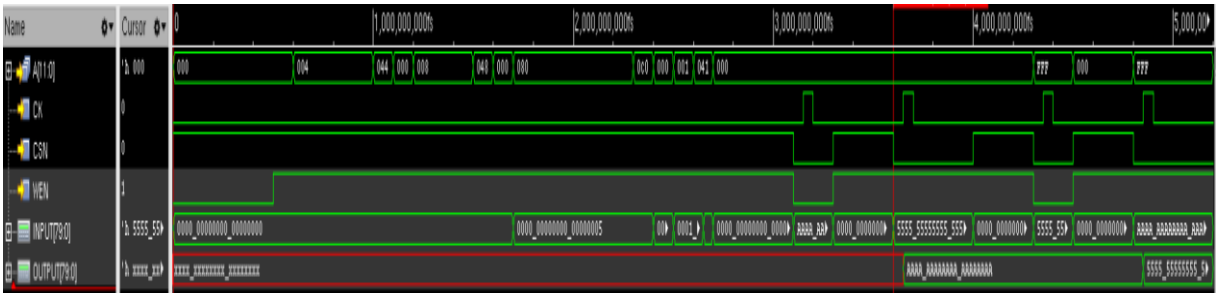


Figure 4.13 Waveform displaying Writing and Reading Operation

4.9 ACCESS TIME CHARACTERIZER(ATC)

In this process, we generate a pulse that represents the time difference between the moment the execution signal is issued and the moment the response is observed.

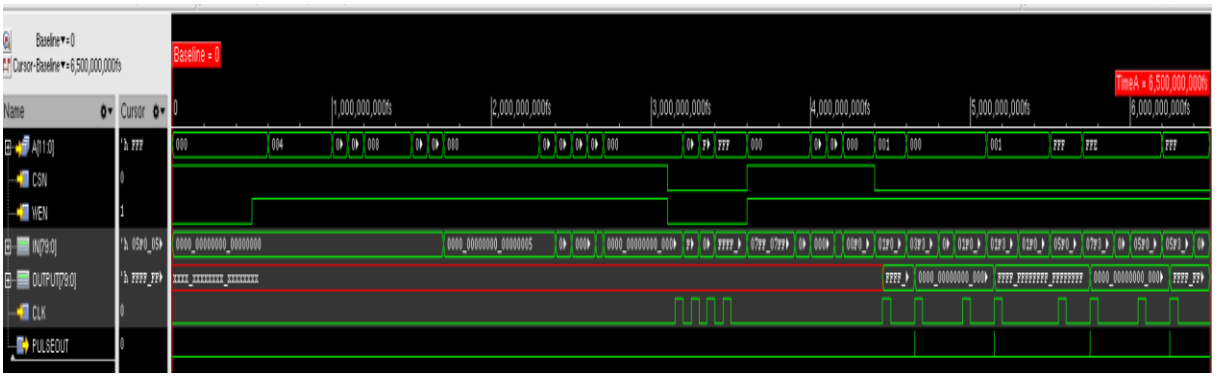


Figure 4.13 Waveform displaying ATC functionality

CHAPTER 5
CONCLUSION & FUTURE SCOPE

The comprehensive validation of the FF_ARRAY and memory block using RT18 technology underscores the meticulous nature of modern digital design workflows. Each phase, from RTL generation to ATPG, plays a critical role in ensuring the design's robustness, reliability, and manufacturability. The process begins with the generation of the RTL netlist, incorporating RT18 library cells, and the creation of a testbench for block-level simulation. Pre-synthesis simulation using XCELIUMAGILE verifies the functional correctness of the design before synthesis transforms the RTL netlist into a gate-level netlist. This step not only translates the design but also optimizes it for performance, area, and power consumption.

Post-synthesis simulation further verifies the design's functionality, ensuring that the synthesis process did not introduce any errors. The insertion of DFT structures, such as scan chains, enhances the design's testability by increasing the controllability and observability of internal states.
Post-DFT simulation is crucial for identifying and addressing any malfunctions caused by unconnected nodes or registers.

Finally, ATPG generates test patterns to detect manufacturing defects, ensuring high fault coverage and optimizing test patterns for efficiency. The iterative nature of the validation process, with multiple simulation and synthesis phases, ensures that potential issues are identified and addressed early, minimizing the risk of costly errors and rework. This proactive approach, coupled with the use of advanced tools and techniques, ensures that the final design meets the highest standards of performance, reliability, and manufacturability.

The iterative nature of the validation process, with multiple simulation and synthesis phases, ensures that potential issues are identified and addressed early. This proactive approach minimizes the risk of costly errors and rework in later stages. The integration of DFT structures further enhances the design's testability, making it easier to diagnose and fix issues during manufacturing.

Moreover, the increasing complexity of memory designs will necessitate more robust and scalable validation techniques. As we move towards higher-density memory technologies and more intricate architectures, the traditional methods of RTL generation, synthesis, and

simulation will need to be augmented with new strategies. For instance, the incorporation of formal verification methods alongside conventional simulation techniques can provide a more comprehensive validation framework. Formal verification can mathematically prove the correctness of certain design aspects, complementing the empirical evidence obtained from simulations. Additionally, the use of hardware emulation and prototyping can provide early insights into the design's performance in real-world scenarios, allowing for more informed design decisions and refinements.

The future scope also extends to the realm of Design-for-Testability (DFT) and Automatic Test Pattern Generation (ATPG). As manufacturing processes continue to shrink and the risk of defects increases, the importance of DFT will grow. Future developments could include more advanced DFT techniques that offer even greater controllability and observability, such as adaptive scan chains that can dynamically reconfigure based on the specific test requirements. Similarly, ATPG tools will need to evolve to handle the increasing complexity and size of modern memory designs. This could involve the use of parallel processing and cloud-based solutions to generate and analyze test patterns more efficiently. Ultimately, the goal will be to ensure that despite the growing complexity of memory designs, the validation and testing processes remain robust, efficient, and capable of delivering high- quality, reliable products to the market.

In conclusion, the design validation of the FF_ARRAY design and memory block exemplify the rigorous processes required to achieve high-quality digital designs. The careful planning and execution of each phase, coupled with the use of advanced tools and techniques, ensure that the final design meets the desired performance, reliability, and manufacturability standards. This comprehensive approach is essential in today's competitive and fast-paced technology landscape, where the demand for reliable and efficient memory designs continues to gro

REFERENCES

[1] Moore, Gordon E. "Cramming more components onto integrated circuits." 19 Apr. 1965,Haissam El-Aawar, “Structural and Hardware Complexities of Microprocessor Design According to Moore’s Law”, International conference on Advanced Computer Science and Information Technology, August 2014

[2] El-Aawar, Haissam. "Structural and Hardware Complexities of Microprocessor Design According to Moore's Law." International Journal of Computer Science & Information Technology 6.4 (2014): 175.

[3] “International Technology Roadmap for Semiconductor”. [www.itrs2.org].

[4] Adams, R. Dean, et al. "An integrated memory self-test and EDA solution." Records of the 2004 International Workshop on Memory Technology, Design and Testing, 2004. IEEE, 2004.

[5] K. J. Hass and J. C. Trinkle. A Survey of Machine Learning Applications in Robotics. IEEE Access, 2021

[6] Kifli, Augusli, et al. "A practical DFT approach for complex low power designs." 2009 Asian Test Symposium. IEEE, 2009.

[7] Rosen, Kenneth H. Elementary number theory. London: Pearson Education, 2011.

[8] Borkar, Shekhar. "Thousand core chips: a technology perspective." Proceedings of the 44th annual design automation conference. 2007.

[9] C. Zhang, Y. Liu, and J. Wang. A Survey on Edge Computing Systems and Tools. IEEE Transactions on Industrial Informatics, 2023.

[10] Singh, Ashok Kumar, Ashok/Ali Siddiqui Kumar Singh (Zeesha), and Siddharth Singh. Recent Advances in Computational Intelligence and Cyber Security. 2024.

[11] Kleinrock, Leonard. "An Internet vision: the invisible global infrastructure." Ad Hoc Networks 1.1 (2003): 3-11.

[12] Remain all -L. Zhou, L. Zhang, and M. Anshel, Key-Evolution Schemes Resilient to Differential Power Analysis, 2011

[13] S. S. Manvi and P. Venkataram, A Comprehensive Survey on Edge Computing: State-of-the-Art, Challenges, and Future Directions, 2022

[14] Sahula, Vineet, C. P. Ravikumar, and D. Nagchoudhuri. "Improvement of ASIC design processes." Proceedings of ASP-DAC/VLSI Design 2002. 7th Asia and South Pacific Design Automation Conference and 15h International Conference on VLSI Design. IEEE, 2002.

[15] Dahl, L-M., et al. "Development of an ASIC macrocell design CAD system." Third Annual IEEE Proceedings on ASIC Seminar and Exhibit. IEEE, 1990.

[16] Daglio, Pierluigi. "A complete and fully qualified design flow for verification of mixed-signal SoC with embedded flash memories." Proceedings of the Design Automation & Test in Europe Conference. Vol. 2. IEEE, 2006.

[17] Mathur, Anmol, and Qi Wang. "Power reduction techniques and flows at RTL and system level." 2009 22nd international conference on VLSI design. IEEE, 2009.

[18] Keshava, Jagannath, Nagib Hakim, and Chinna Prudvi. "Post-silicon validation challenges: How EDA and academia can help." Proceedings of the 47th Design Automation Conference. 2010.

[19] Mohammed, Rahima K., Ridvan A. Sahan, and Mohanraj Prabhugoud. "Design Challenges of thermal margining tools for Silicon Validation." 2010 12th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems. IEEE, 2010.

[20] Nicolici, Nicola, and Ho Fai Ko. "Design-for-debug for post-silicon validation: Can high-level descriptions help?." 2009 IEEE International High Level Design Validation and Test Workshop. IEEE, 2009.

[21] Reinaldo, Qadeer Qureshi, and Rodrigo Zeli. "Flexible architecture of memory BISTs." 2018 IEEE 19th Latin-American Test Symposium (LATS). IEEE, 2018.

[22] Zarrineh, Kamran, et al. "Self-test architecture for testing complex memory structures." Proceedings International Test Conference 2000 (IEEE Cat. No. 00CH37159). IEEE, 2000.

new_rishi

ORIGINALITY REPORT

18%	10%	15%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	wikimili.com Internet Source	2%
2	scholar.archive.org Internet Source	2%
3	www.researchgate.net Internet Source	2%
4	D. Nagchoudhuri. "Improvement of ASIC design processes", Proceedings of ASP-DAC/VLSI Design 2002 7th Asia and South Pacific Design Automation Conference and 15h International Conference on VLSI Design ASPDAC-02, 2002 Publication	1%
5	L. Jozwiak. "Interoperability and quality of new EDA tools for sequential logic synthesis", Proceedings International Symposium on Quality Electronic Design ISQED-02, 2002 Publication	1%