RAMAIAH
SKILL ACADEMY

Project Presentation of Designing and Implementing RTL to GDS II for FIFO using Synopsys Tools"

Project Team
1. Akhil Tendulkar
2. Toushifraza Darur
3. Naresh K R

Mentors
Mr. Darshan M R
Mrs. Priyanka C

Project Team

RAMAIAH SKILL ACADEMY

Sl.No	Registration Number	Name of the Trainee
1	RSAGWPVLSI01	Akhil Tendulkar
2	RSAGWPVLSI27	Toushifraza Darur
3	RSAGWPVLSI15	Naresh K R

Outline of the Presentation

RAMAIAH SKILL ACADEMY

- Introduction
- Title and Aim
- Objectives
- Block Diagram
- Solution Methodology
- Results and Discussions
- References

Introduction

A **First-In, First-Out (FIFO)** is a type of memory buffer or queue that operates on the principle that the first data item stored is the first one retrieved. This ensures that data is processed in the same order it arrives, making it essential for efficient data handling in digital systems.

Why FIFO?

Data Flow Management & Synchronization – Ensures smooth transfer of data and safely handles clock domain crossing (CDC).

Order Preservation – Guarantees the correct sequence of incoming data, essential for packet communication, audio/video streaming, and CPU instructions.

Prevents Overflow & Latency – Uses Full/Empty flags to prevent overflow/underflow, while buffering data to reduce processing delays and bottlenecks.

Efficient Resource Utilization – Optimizes memory usage and system performance, reducing power consumption by minimizing unnecessary delays.

Title and Aim

RAMAIAH SKILL ACADEMY

Designing and Implementing RTL to Netlist for FIFO(First-In, First-Out) using Synopsys Tools

Aim:

The goal of this project is to design, verify, and synthesize a FIFO(First In, First-Out) from RTL (Register Transfer Level) to Placement using industry-standard EDA tools.

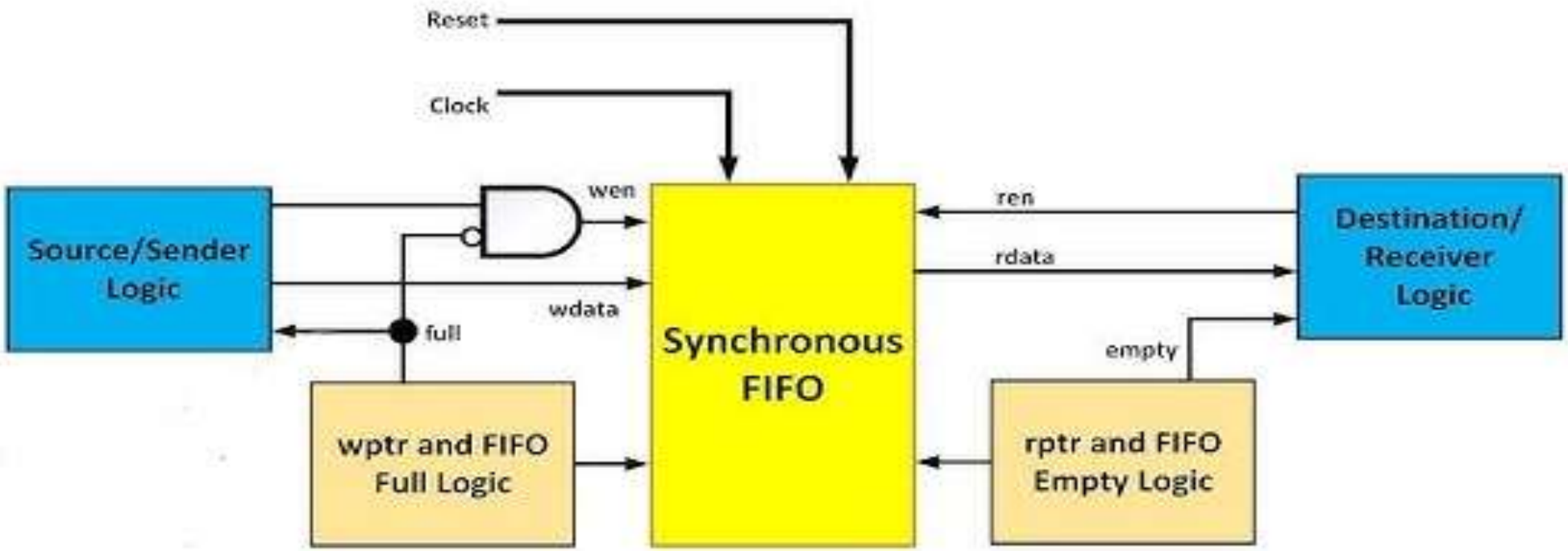
Objectives

RAMAIAH SKILL ACADEMY

- To design an optimized FIFO using Verilog/VHDL.
- To simulate and verify the design using Synopsys VCS.
- To check design constraints and static timing analysis using DesignVision & PrimeTime.
- To perform formal verification using Formality.
- To ensure Design-for-Testability (DFT) compliance using DFT Compiler.
- To generate a netlist using Design Compiler.
- To analyze power, area, and timing for optimization.
- Physical Design Process using IC Complier.

Block Diagram

RAMAIAH SKILL ACADEMY



Solution Methodology

RAMAIAH SKILL ACADEMY

Step 1: RTL Design

- Implement FIFO in **Verilog/VHDL.**
- Use **Synopsys VCS** for functional verification.

Step 2: Linting and Static Checking

- Use **Spyglass** to check for coding errors and design violations.

Step 3: Synthesis

- Convert RTL to netlist using **Design Compiler.**
- Optimize for power, area, and timing.

Step 4: Design for Testability (DFT)

- Insert scan chains using **DFT Compiler.**
- Ensure testability for manufacturing.

Solution Methodology

RAMAIAH SKILL ACADEMY

Step 5: Static Timing Analysis

- Use **PrimeTime** for setup and hold time verification.
- Verify if timing constraints are met.

Step 6: Formal Verification

- Compare RTL and netlist using **Formality** to check equivalence.

Step 7: Final Netlist Generation

- Obtain a gate-level netlist for fabrication-ready implementation.

Physical Design Process

RAMAIAH SKILL ACADEMY

Step 1: Floorplanning

- Define the overall chip layout, organizing blocks, and setting up areas for key components like FIFOs, power, and clock distribution.

Step 2: Placement

- Position the gate-level netlist cells optimally to minimize routing congestion and delay while meeting timing constraints.

Step 3: Clock Tree Synthesis (CTS)

- Generate a balanced clock tree to ensure even distribution of clock signals, minimizing clock skew and ensuring proper timing across the design.

Step 4: Routing

- Connect the placed cells using metal layers, ensuring correct signal integrity and power distribution while avoiding timing issues like crosstalk.

Step 5: Static Timing Analysis (STA)

- Use PrimeTime for setup and hold time verification.
- Verify if timing constraints are met, ensuring no timing violations or delays in the design.

Physical Design Process

Step 6: Design Rule Check (DRC) & Layout Versus Schematic (LVS)

- **DRC**: Ensure the design adheres to foundry-specific design rules regarding spacing and layer width.
- **LVS**: Verify that the final physical layout matches the original schematic and netlist, checking for any discrepancies.

Step 7: Final Netlist Generation

- Obtain the final gate-level netlist after routing, clock tree synthesis, and other optimizations.
- This netlist is used for tape-out and fabrication-ready implementation.

Step 8: Extraction and Optimization

- Extract parasitic information (resistances and capacitances) from the layout for post-layout simulations.
- Optimize for power, performance, and area by adjusting the design parameters.

Step 9: GDSII Generation

- Generate the **GDSII** file, a standard format containing the final physical layout ready for manufacturing.
- This file includes all the data required for the fabrication process.

Results and Discussions

RAMAIAH SKILL ACADEMY

Observations:

1. **Placement and Routing:**
    1. The FIFO design was successfully placed and routed with minimal congestion.
    2. Routing optimization improved signal integrity, ensuring proper data transfer without delay.
2. **Static Timing Analysis (STA):**
    1. The design met all setup and hold time requirements during physical design.
    2. After static timing analysis with PrimeTime, there were no critical timing violations, indicating good timing closure.
3. **Power & Area Optimization:**
    1. Power consumption was optimized through better placement and routing strategies, reducing unnecessary power usage.
    2. The area of the design was optimized, meeting the area constraints for the target technology.
4. **Clock Tree Synthesis (CTS):**
    1. The clock distribution network was successfully synthesized, ensuring that clock skew was minimized and meeting the timing requirements across the design.
5. **GDSII File Generation:**
    1. The final **GDSII** file was generated successfully, ready for tape-out with no critical errors found during the design rule check (DRC) and layout versus schematic (LVS) checks.

Results and Discussions

RAMAIAH SKILL ACADEMY

Challenges Faced:

1.Timing Violations During Placement and Routing:

Some timing violations were observed during the initial placement and routing phase. These were resolved by adjusting the placement of critical cells and re-optimizing the routing.

2.Clock Skew:

The initial clock tree synthesis (CTS) resulted in some clock skew issues. The clock network had to be re-optimized to ensure minimal skew and meet timing requirements.

3.Power Consumption:

Managing power consumption was a challenge during the routing phase. Techniques such as multi-vt cells and power gating were applied to minimize power usage without compromising performance.

4.Design Rule Check (DRC) Errors:

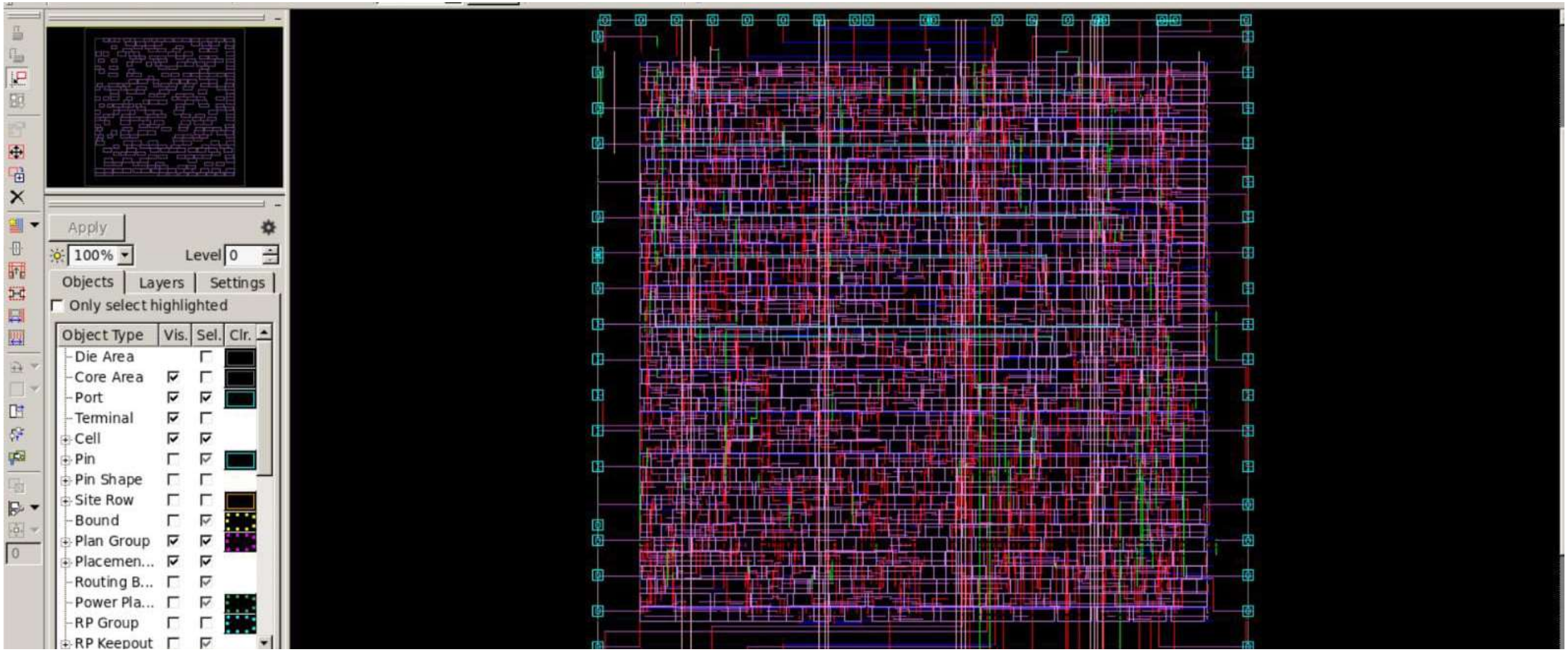
A few design rule check (DRC) violations were found due to tight spacing constraints in the early stages of the routing. These were addressed by adjusting the routing layers and cell placement.

5.LVS Discrepancies:

Minor discrepancies were detected between the netlist and layout during the LVS check. These were resolved by adjusting the layout to ensure it matched the original RTL design.

Worked Layout

RAMAIAH SKILL ACADEMY



RAMAIAH
SKILL ACADEMY

Thank You