

Paper Presentation

CSE 460 - VLSI Design

Paper -A method to speed up VLSI hierarchical physical design in floorplanning.

PRESENTED BY -

Name - Ashakuzzaman Odree ; **ID** - 20301268

Name - Md Rakibul Islam ; **ID** - 20301373

Name - Priti Saha ; **ID** - 20101475

Abstract

Background :

- VLSI (Very-Large-Scale Integration) has seen rapid increases in size and complexity.
- Meeting speed and quality in IC physical design is challenging.

Proposed Solution:

- Efficient model for swift VLSI floorplanning.
- Utilizes Active-Logic Reduction Technology.
- Replaces some modules in netlist file with filler units (no logical connections).

Abstract

Benefits:

- Reduces internal logical units.
- Quickly predicts timing closure after top and block implementation.
- Aids in quick quality assessment of the floorplan.
- Maintains design quality while accelerating design flow.

Results from Experiments:

- Drastic runtime reduction: **6.2 times faster** on average.
- Memory usage reduction: **2.8 times less** on average.

Introduction

IC Physical Design Process:

- Gate-level netlist synthesis
- Floorplanning
- Power planning
- Placement
- Preroute (mainly trail route)
- CTS (Clock Tree Synthesis)
- Routing
- Creation of GDSII layout file

Importance of Floorplanning:

- Performance Optimization.
- Power Efficiency
- Thermal Management
- Area Utilization
- Cost Reduction
- Routing Complexity
- Area Utilization



Introduction

Challenges in Modern VLSI Design:

- Advanced manufacturing processes leading to increased chip integration and reduced size.
- Emergence of Nano scale VLSI design makes EDA tools more crucial.
- Complex netlists with tens of millions of gate-level units.

Historical Perspective:

- R. Otten introduced automatic floorplan design method in 1982.
- EDA tools require multiple constraints for floorplanning due to its complexity.
- Multiple iterations and adjustments needed to achieve desired physical design outcomes.

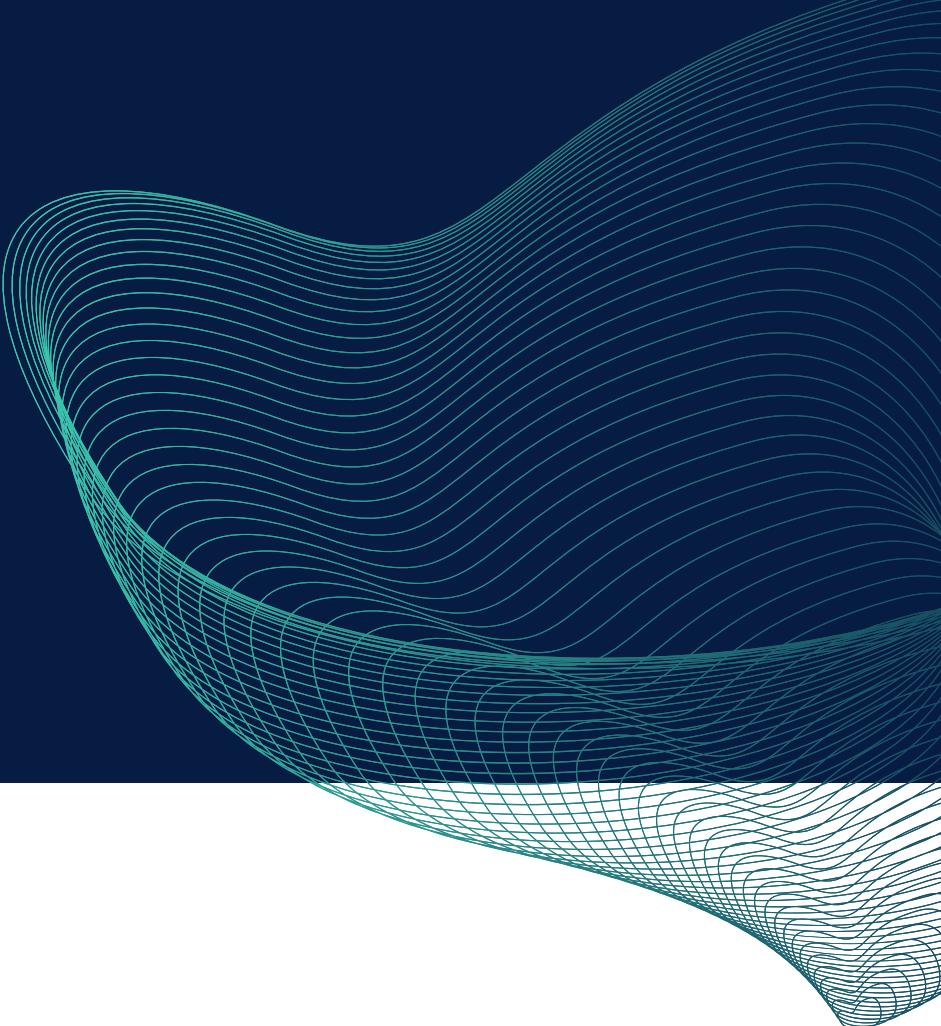
Introduction

Recent Innovations in Floorplanning:

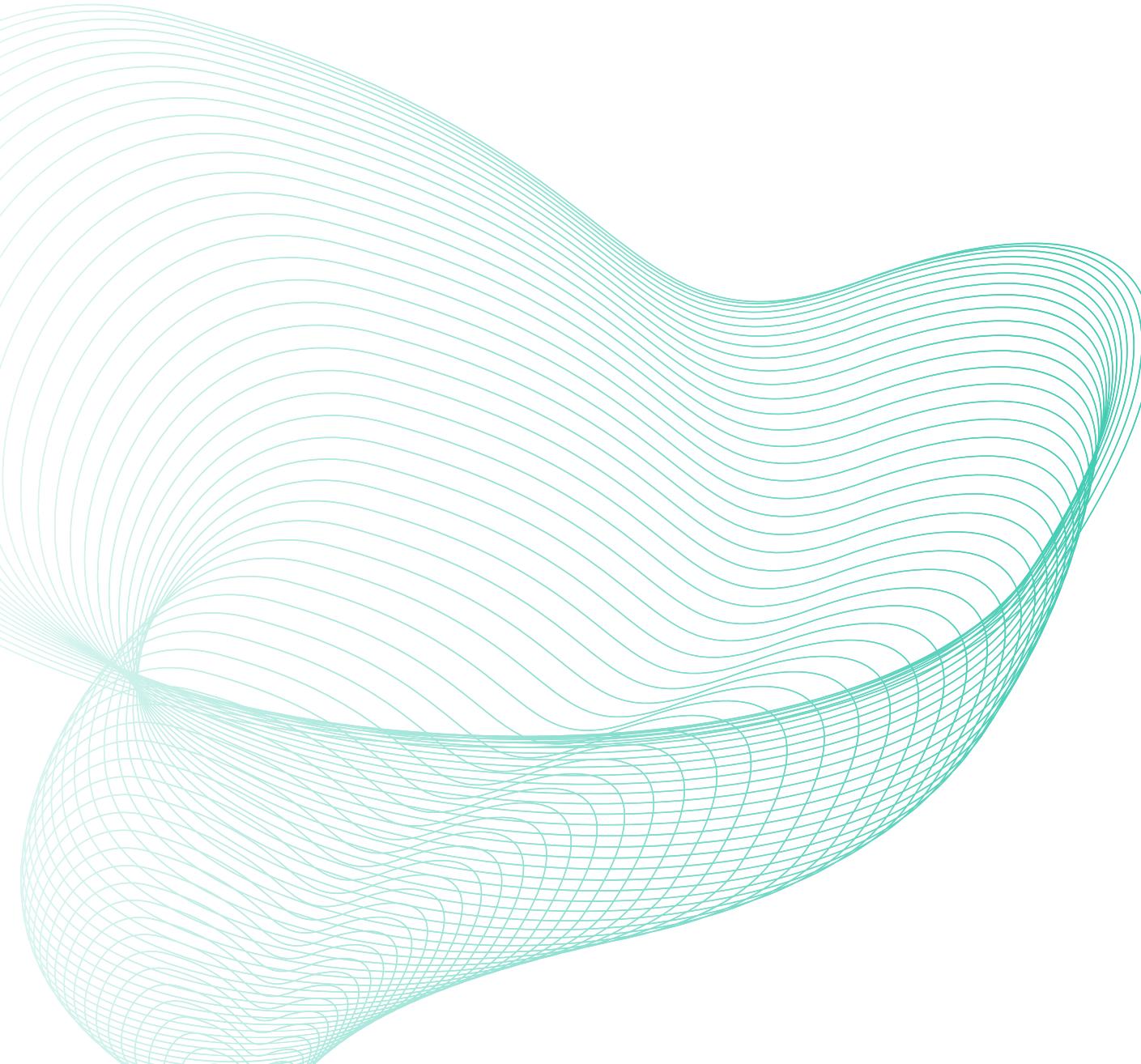
- Simulated annealing algorithm.
- Guided incremental floorplan algorithm to reduce IR-drop violations using B*-tree representation.
- P/G network and floorplan methods for quick design convergence.

Primary Concern:

- Inefficiency in current methods due to repeated iterations and adjustments.
- Need for advanced algorithms and model optimization to accelerate ASIC physical design without compromising on floorplan quality.



Quick floorplanning method



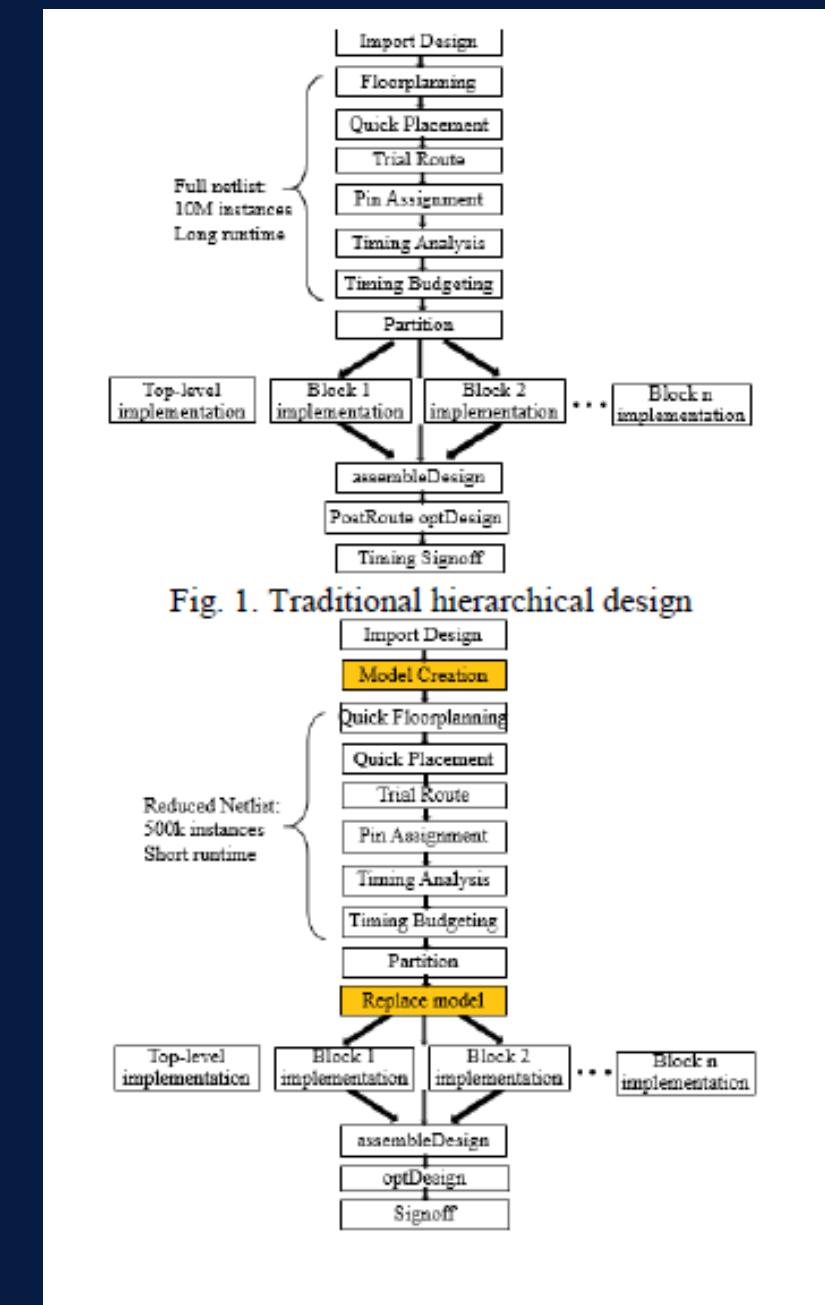
Hierarchical Design in VLSI

Active-Logic Reduction Technology (ART)

Simplified Model and Flex Filler: Quick Floorplanning

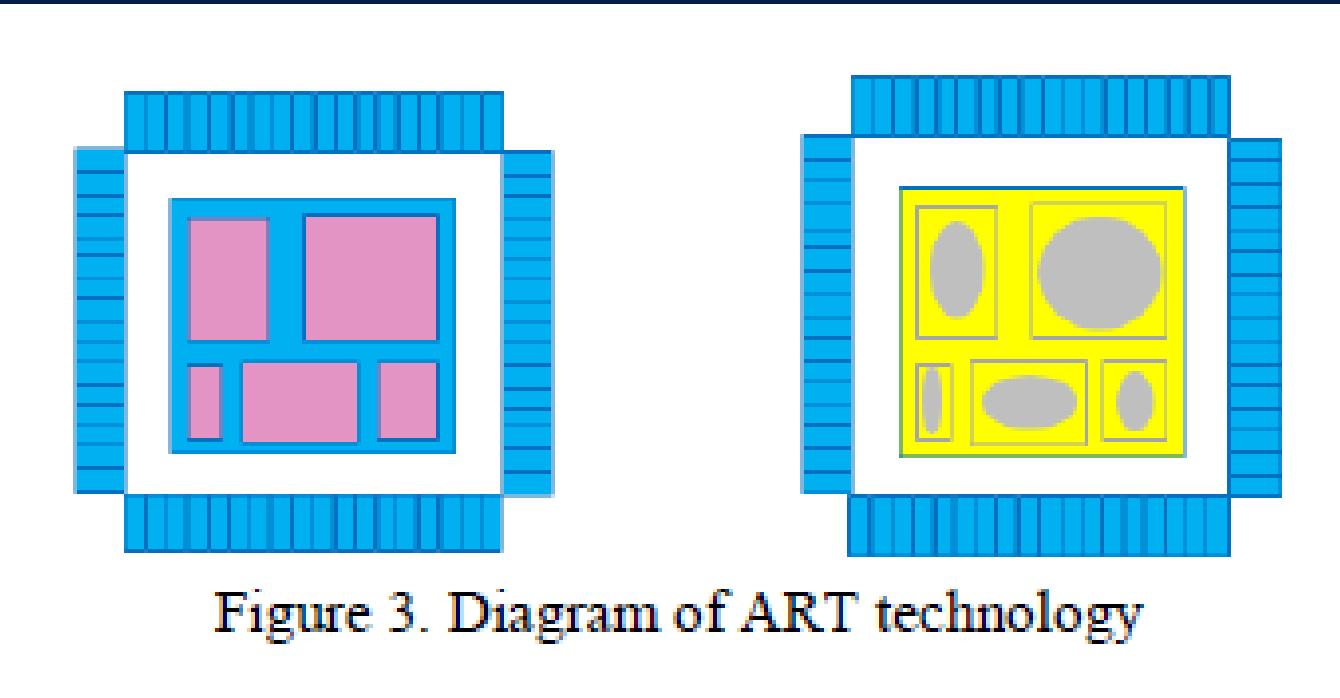
Hierarchical Design in VLSI

- Introduction to the escalating complexity of VLSI design.
- Highlight the proposed approach combining simplified modeling and hierarchical design.
- Traditional process versus the proposed method's runtime comparison.
- Emphasize the challenge of runtime in the traditional approach.
- Introduce the concept of applying a simplified model to accelerate floor planning and the overall flow.



Active-Logic Reduction Technology (ART)

- Explore Active-Logic Reduction Technology (ART).
- Explain its foundation in Logical Reduction from computer science.
- Showcase the significance of generating an active logical view.
- Contrast the shortcomings of the traditional approach's timing prediction.
- Describe how ART bridges the gap between partitioning and assembling stages.
- Highlight ART's role in masking inactive logic units and boosting computation efficiency.



Simplified Model and Flex Filler

- Delve into the concept of a simplified model for swift floorplanning.
- Describe the use of logic minimization algorithms for design size reduction.
- Detail the core of the method: a simplified model utilizing ART technology.
- Introduce the flex filler as a solution for replacing irrelevant internal logic units.
- Highlight flex filler advantages: larger area, absence of logical connections, reduced P&R runtime.
- Emphasize that the simplified model preserves interface connections, enhances floorplanning, and maintains design quality.

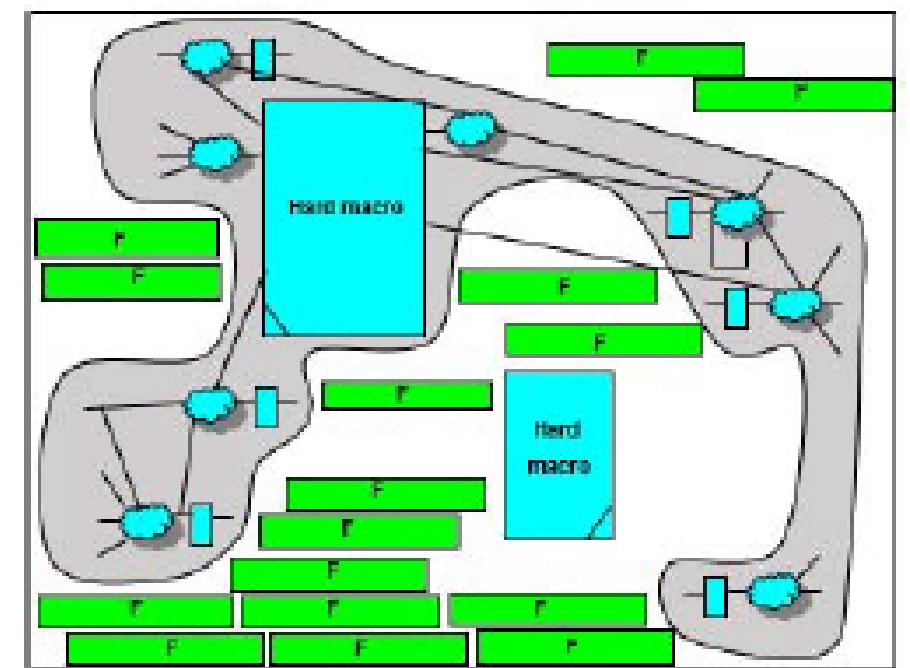
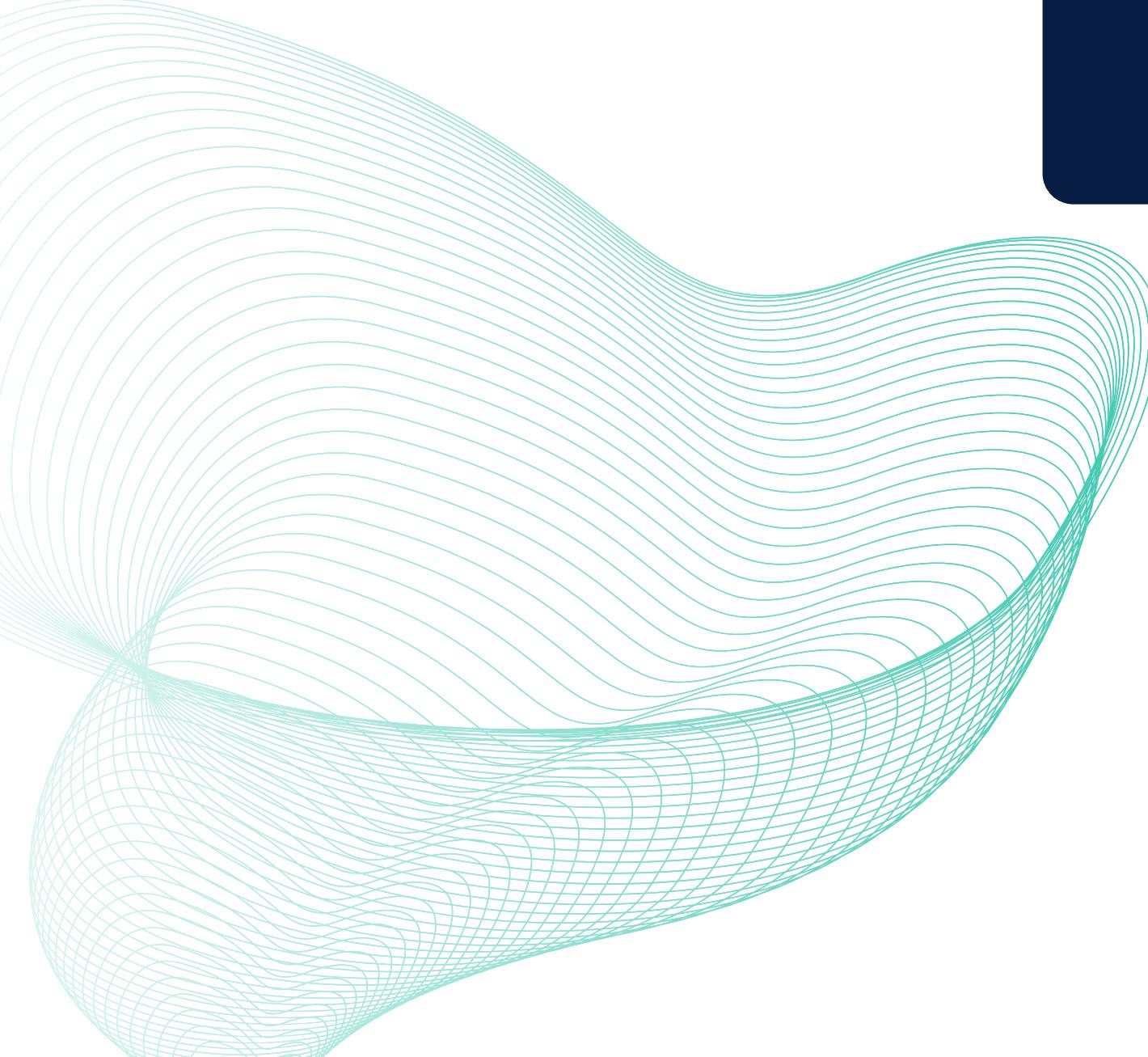


Figure 4. Schematic diagram of simplified model

Experimental Result and Analysis



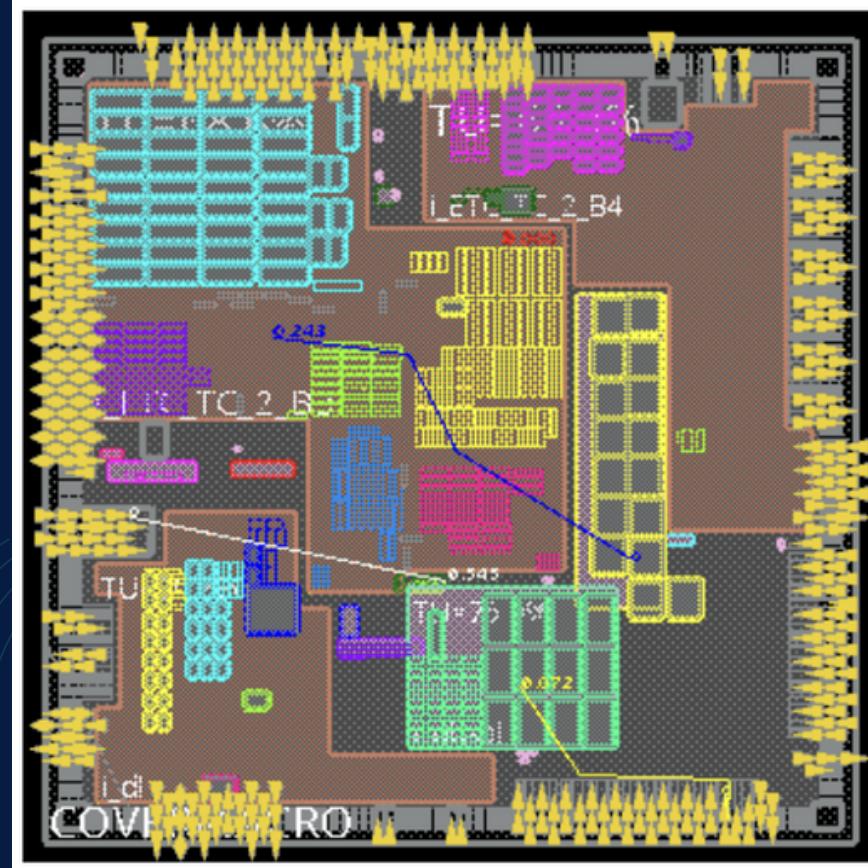
Chip Size Reduction

Runtime Improvement

Congestion Analysis

Combined Benefits

Chip Size Reduction



- Original design size: 2.1M (2,086,769 instances)
- After simplified model creation: 0.2M (128,695 instances + 71,829 inactive logics reduced)
- Netlist size reduction: 9.6%
- The simplified model effectively reduced the netlist size and improved subsequent design steps.

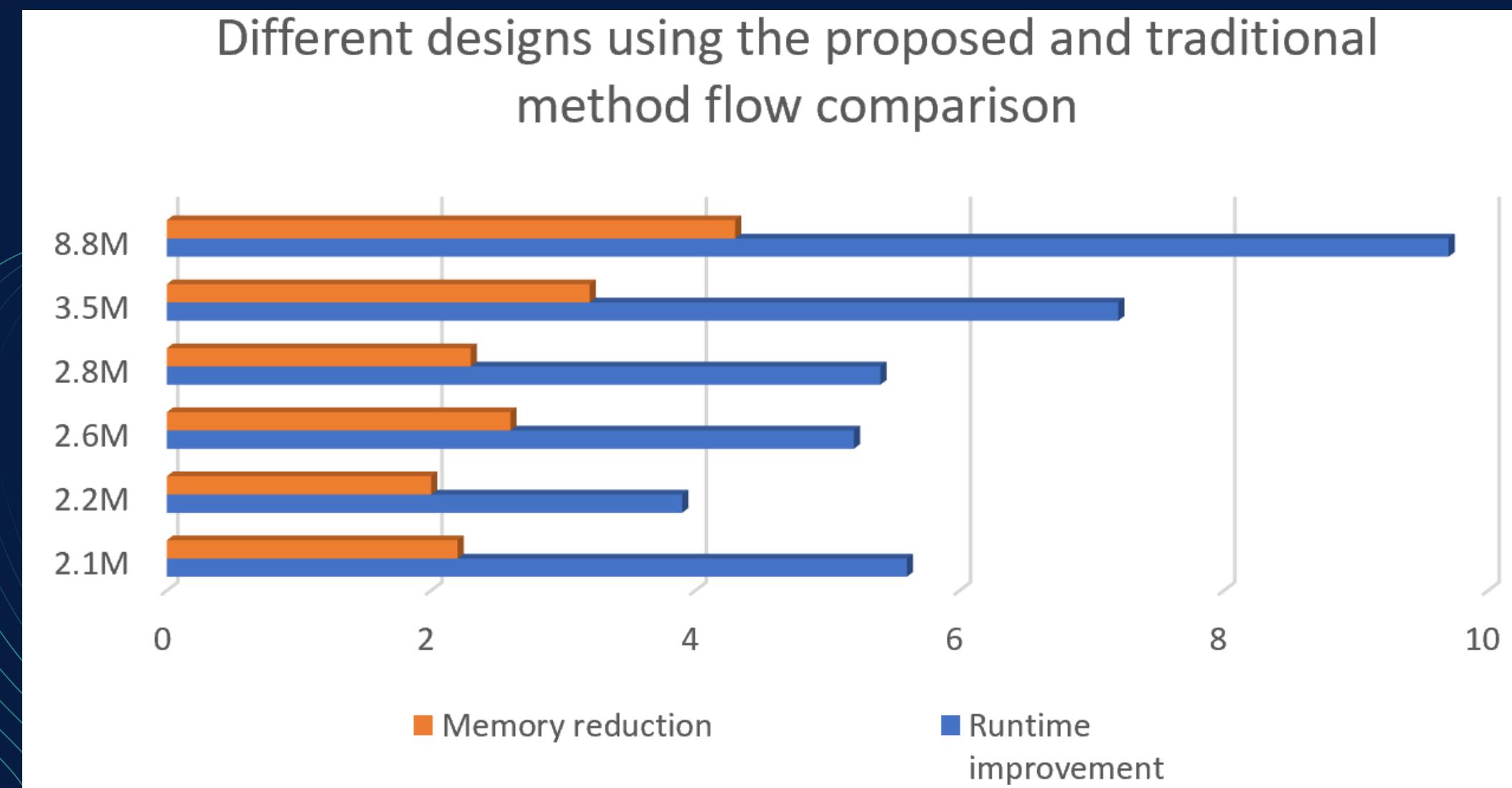
Fig: Chip design result of floorplanning with simplified model

Runtime Improvement



- Total runtime without simplified model: 807.95 minutes
- Total runtime with simplified model partition: 145.08 minutes
- Runtime reduction by 5.57 times (from 807.95 to 145.08 minutes)
- Peak memory usage reduction by 2.22 times
- The introduction of the simplified model significantly accelerated the design flow and reduced memory usage.

Compare



Congestion Analysis

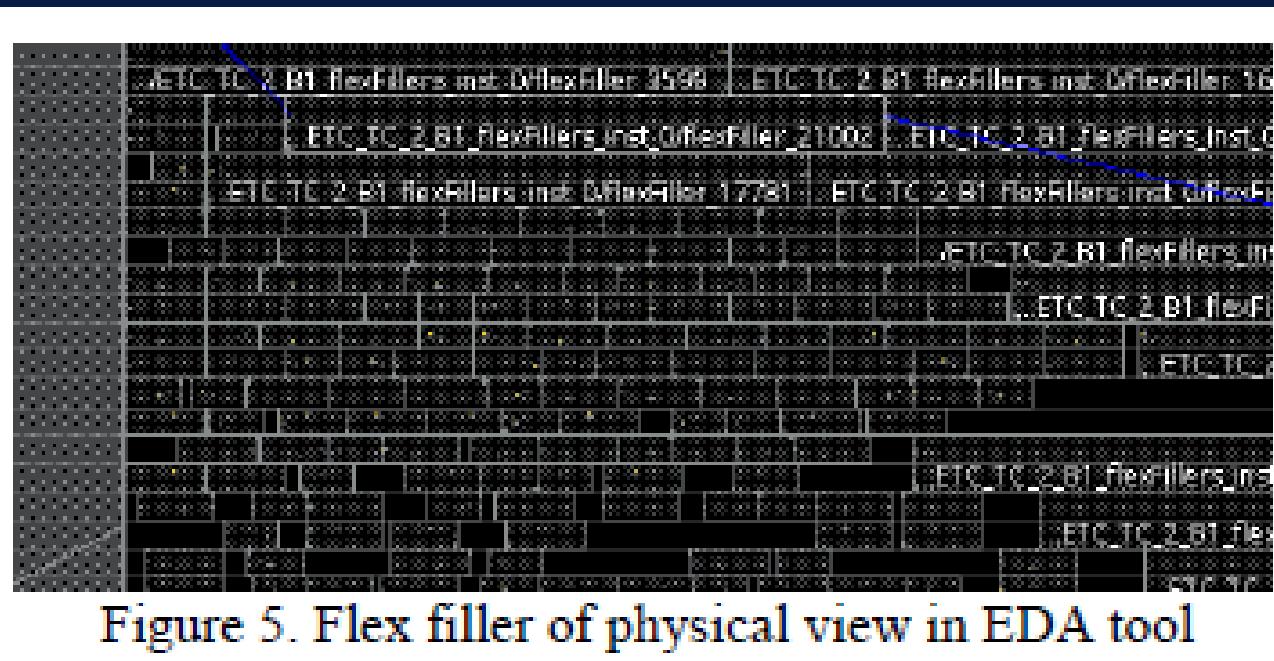


Figure 5. Flex filler of physical view in EDA tool

- Overall horizontal congestion: 0.10%
- Vertical congestion: 0.35%
- Congestion levels were within acceptable limits for a 2.1M design.
- The method maintained reasonable congestion results while significantly accelerating the design process.

Benefits

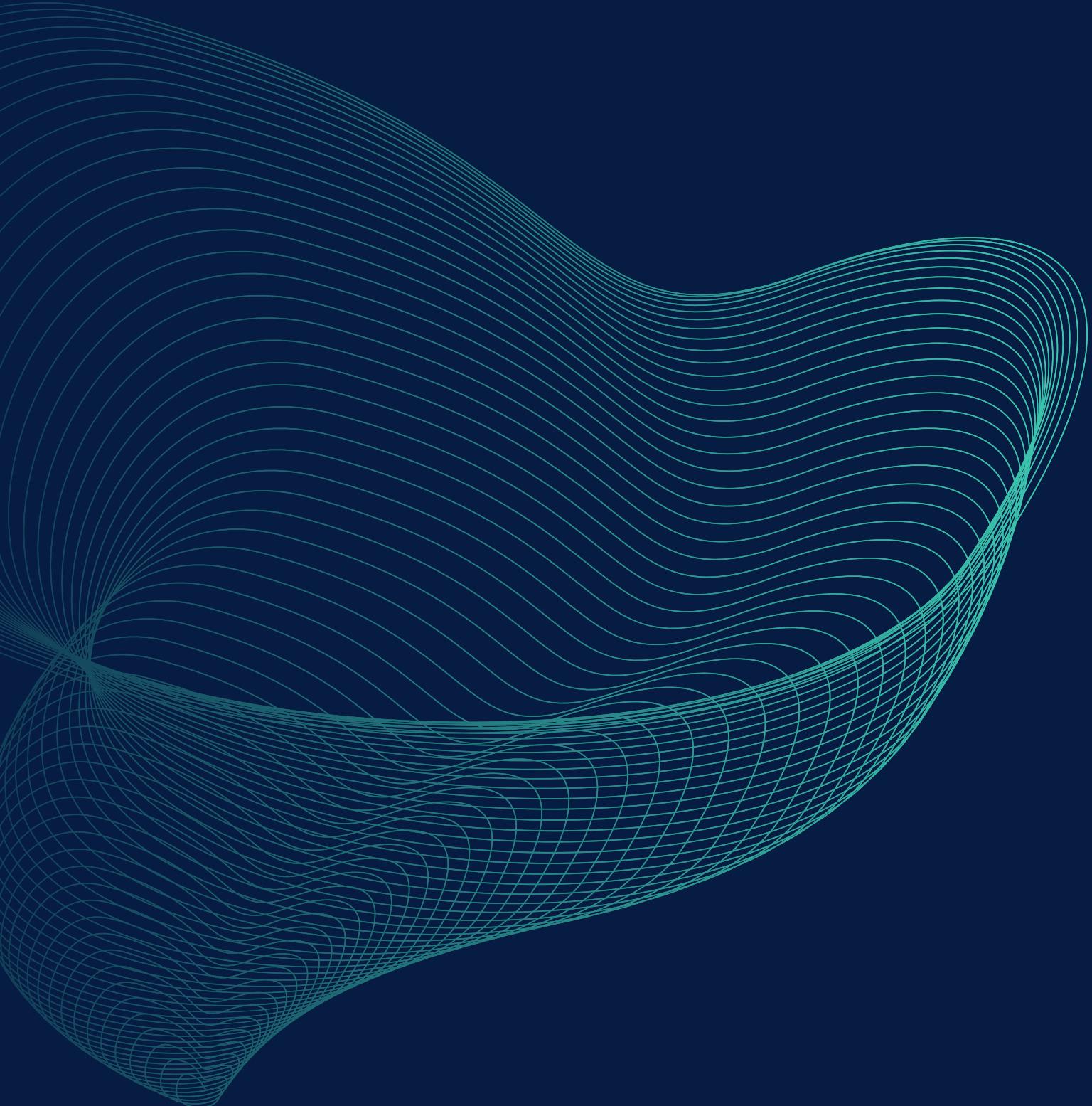


- Substantial reduction in design size, enhancing subsequent optimization.
- The design flow runtime was improved, accompanied by reduced memory usage.
- The method successfully achieved timing closure and maintained congestion within acceptable bounds

Conclusion

- Floorplanning: determines the shapes and arrangement of sub-circuits or modules, as well as the locations of external ports and IP or macro-blocks power and ground routing
- Shortening the floorplanning and evaluation runtime is an effective way to speed up chip's physical design.
- This method can drastically reduce the runtime by 6.2 times and memory usage by 2.8 times





THANK YOU

