IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15119

Implementation of Urdhva Tiryakbhayam Multiplier using VLSI

Nithya S[1], Varshitha S K[2], Nisarga B R[3], Priyanka M Hiremath[4], Prathibha Y G[5]

Professor, Department of Electronics and Communication, SJBIT, Bangalore, India[1]

Student, Department of Electronics and Communication, SJBIT, Bangalore, India[2]

Student, Department of Electronics and Communication, SJBIT, Bangalore, India[3]

Student, Department of Electronics and Communication, SJBIT, Bangalore, India[4]

Student, Department of Electronics and Communication, SJBIT, Bangalore, India[5]

Abstract: In the era of high-performance computing and miniaturized electronic systems, the demand for high-speed, low- power, and area-efficient arithmetic units has increased significantly. Among all arithmetic operations, multiplication plays a vital role in determining overall system performance, especially in digital signal processing, cryptography, image processing, and embedded systems. Conventional multiplier architectures such as array multipliers, Booth multipliers, and Wallace tree multipliers often suffer from high propagation delay, increased power consumption, and larger silicon area when scaled to higher bit-width operations.

To address these limitations, this paper presents the design and VLSI implementation of a 64-bit multiplier based on the Urdhva Tiryakbhayam (UT) Sutra of Vedic Mathematics. The UT algorithm, meaning "Vertically and Crosswise," enables parallel generation of partial products, resulting in reduced computation delay and improved throughput. The proposed architecture is designed using Verilog HDL and follows a hierarchical approach by decomposing large-bit multiplication into smaller modular blocks. Functional verification is carried out through extensive simulation, followed by synthesis and physical implementation using standard VLSI design flow.
Post-synthesis and post-layout analysis demonstrate that the UT-based multiplier achieves superior speed performance while maintaining competitive power consumption and area utilization when compared with conventional multiplier architectures. The results validate the effectiveness of integrating Vedic mathematical principles with modern VLSI methodologies for high- performance arithmetic circuit design.

Keywords: Vedic Mathematics, Urdhva Tiryakbhayam, VLSI Design, High-Speed Multiplier, Verilog HDL

I. INTRODUCTION

Multiplication is one of the most fundamental arithmetic operations in digital systems and plays a crucial role in determining the performance of processors, digital signal processors (DSPs), cryptographic engines, and embedded systems. As modern applications demand higher computational speed and precision, the efficiency of arithmetic units, particularly multipliers, has become a critical design consideration in Very Large Scale Integration (VLSI) systems.

Traditional multiplication techniques such as array multipliers, Booth multipliers, and Wallace tree multipliers have been widely used due to their simplicity and proven reliability. However, when these architectures are scaled to higher bit- widths such as 64-bit operations, they suffer from increased propagation delay, higher power consumption, and larger silicon area. These limitations make conventional multipliers less suitable for high-performance and low-power applications.

To overcome these challenges, alternative multiplication algorithms that offer higher parallelism and reduced computational complexity are required. Vedic Mathematics, an ancient system of Indian mathematics, provides efficient arithmetic algorithms that are well suited for hardware implementation. Among them, the Urdhva Tiryakbhayam Sutra, meaning "Vertically and Crosswise," offers a systematic and parallel approach to multiplication that significantly reduces computation delay.

This paper focuses on the design and VLSI implementation of a 64-bit multiplier based on the Urdhva Tiryakbhayam algorithm. By exploiting the inherent parallelism of the UT method and integrating it with modern VLSI design methodologies, the proposed architecture aims to achieve high-speed operation while maintaining optimal power and area efficiency.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15119

II. PROPOSED METHODOLOGY

The proposed methodology focuses on the design and VLSI implementation of a 64-bit multiplier based on the Urdhva Tiryakbhayam (UT) Sutra of Vedic Mathematics. The approach integrates the parallel computation advantages of the UT algorithm with a complete VLSI design flow, including RTL modeling, synthesis, simulation, and physical implementation.

The methodology is structured to ensure scalability, modularity, and optimized performance in terms of speed, power, and area. A hierarchical design approach is adopted, where large-bit multiplication is decomposed into smaller modules, enabling efficient reuse and easier verification.

A. Overview of Urdhva Tiryakbhayam Algorithm

The Urdhva Tiryakbhayam Sutra, meaning "Vertically and Crosswise," is a general multiplication algorithm from Vedic Mathematics that allows simultaneous generation of partial products. Unlike conventional multiplication techniques that generate partial products sequentially, the UT algorithm enables parallel computation, significantly reducing propagation delay.

For binary multiplication, the UT method involves dividing the operands into smaller segments and performing crosswise and vertical multiplications. The partial products generated are then accumulated using efficient adder structures. This parallelism makes the UT algorithm well suited for high- speed hardware implementation, especially for large bit-width multipliers.

B. Architectural Design

The proposed 64-bit multiplier architecture is designed using a hierarchical approach. The 64-bit operands are divided into two 32-bit halves, and four 32×32 multipliers are used to generate partial products. These partial products are appropriately shifted and summed to obtain the final 128-bit result.

This modular design enhances scalability and reduces design complexity. Smaller UT-based multiplier blocks such as 2×2, 4×4, 8×8, and 16×16 are recursively used to construct higher-bit multipliers. The architecture ensures efficient utilization of hardware resources while maintaining high computational speed.

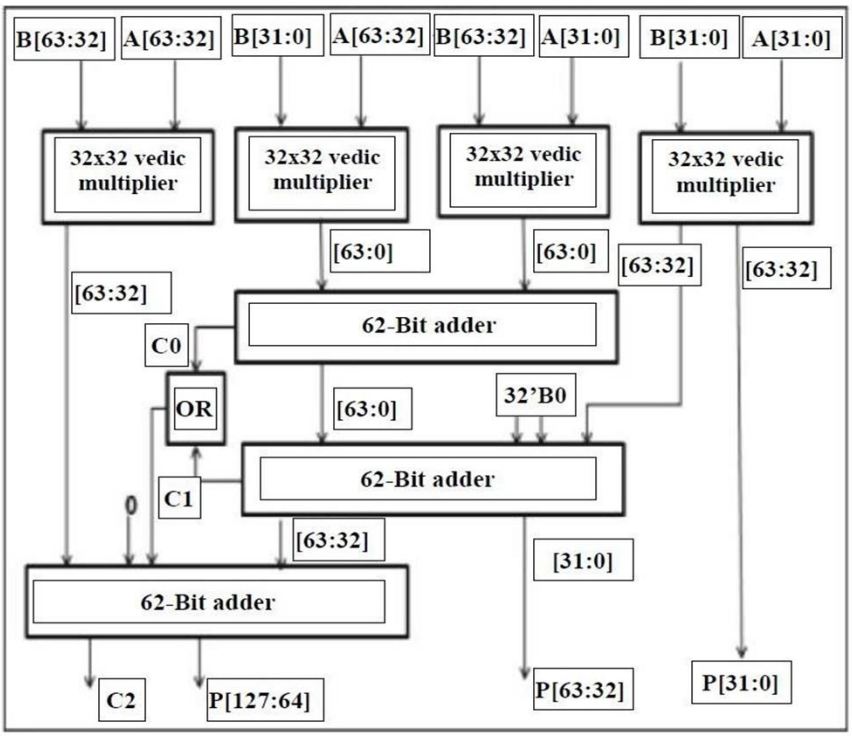

Fig. 1: Block Diagram of the Proposed 64-bit Urdhva Tiryakbhayam Multiplier

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026
DOI: 10.17148/IJARCCE.2026.15119

C. Design Methodology

The proposed design methodology is divided into multiple stages:
Stage 1: Algorithm Analysis and Block-Level Design The UT algorithm is analyzed and adapted for binary multiplication. The 64-bit multiplication is decomposed into smaller modules to simplify implementation and verification.
Stage 2: RTL Design and Modeling The architecture is modeled using Verilog HDL with a modular and hierarchical structure. Parameterized modules are used to improve reusability and scalability.
Stage 3: Functional Simulation Functional verification is performed using simulation tools to ensure correctness across a wide range of input combinations. Waveform analysis is used to validate partial product generation and accumulation.
Stage 4: Synthesis The RTL design is synthesized using industry-standard synthesis tools. Timing, area, and power reports are analyzed, and optimizations are performed to meet design constraints.
Stage 5: Physical Design The synthesized netlist is subjected to physical design stages, including floorplanning, placement, clock tree synthesis, and routing. Design rule

checks (DRC) and layout-versus-schematic (LVS) verification are performed to ensure correctness.
Stage 6: Post-Layout Analysis Post-layout simulations and static timing analysis are carried out to evaluate real-world performance metrics such as delay, power consumption, and area utilization.

D. Tool Chain Used

TABLE II: Tool Chain Used in Design and Implementation

Task	Tool
HDL Design	Verilog HDL
Simulation	ModelSim / XSIM
Synthesis	Synopsys Design Compiler / Cadence Genus
Physical Design	Cadence Innovus / Synopsys ICC
Verification	STA, DRC, LVS
Power and Area Analysis	Innovus Reports / Power Compiler

E. Design Advantages

The proposed Urdhva Tiryakbhayam-based multiplier offers several advantages:
•High-speed operation due to parallel generation of partial products.
•Scalability to higher bit-widths through modular architecture.
•Reduced propagation delay compared to conventional multipliers.
•Efficient utilization of silicon area.
•Suitability for low-power and high-performance applications.

III. FIRMWARE ARCHITECTURE

The firmware architecture for the implementation of the Urdhva Tiryakbhayam multiplier defines the internal organization and control logic required to perform multiplication efficiently on hardware. It acts as a bridge between the mathematical algorithm and the physical VLSI realization, ensuring correct data flow, timing, and control of operations.

The architecture is designed in a modular and hierarchical manner, where complex multiplication is built using smaller functional blocks. At the lowest level, basic arithmetic units such as half adders and full adders are used to generate and sum partial products. These blocks form the core computation units of the multiplier.

The firmware follows the Urdhva Tiryakbhayam principle of vertical and crosswise multiplication. Inputs are processed simultaneously, allowing partial products to be generated in parallel rather than sequentially. This parallelism significantly reduces computation time and improves overall speed. Each stage of multiplication corresponds to a specific vertical or crosswise operation, and the results are accumulated using adders.

Control logic is included to manage signal synchronization and ensure proper timing between stages. This logic coordinates the generation of partial products, their addition, and the propagation of carry signals. Since the architecture is parallel in nature, careful control of data paths is essential to avoid timing mismatches and ensure reliable operation.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15119

The firmware architecture is also designed to be scalable, allowing easy extension to higher bit-width multipliers such as 16-bit, 32-bit, or 64-bit. Larger multipliers are constructed by combining smaller multiplier blocks, which simplifies design verification and reduces complexity. This modular approach makes the architecture flexible and suitable for different performance requirements.

Additionally, the architecture supports low-power operation by minimizing unnecessary switching activity and reducing the critical path delay. Efficient signal routing and optimized adder structures contribute to lower power consumption, making the design suitable for power-sensitive VLSI applications.

Overall, the firmware architecture provides a structured, efficient, and scalable framework for implementing the Urdhva Tiryakbhayam multiplier in VLSI, ensuring high speed, reduced power consumption, and reliable performance in modern digital systems.

IV. EXPERIMENTAL SETUP AND RESULTS

This section presents the experimental setup, testing methodology, and performance analysis of the proposed 64- bit Urdhva Tiryakbhayam multiplier. The design is evaluated through functional simulation, synthesis, and physical implementation to validate its correctness and performance improvements over conventional multiplier architectures.

A. Testing Methodology

The testing methodology follows a structured approach to ensure functional correctness and performance reliability. Initially, the individual UT-based multiplier modules such as 2×2, 4×4, 8×8, and 16×16 are verified independently using test benches. These verified modules are then hierarchically integrated to construct the complete 64-bit multiplier. Functional simulation is performed using industry-standard simulation tools to validate the correctness of partial product generation and accumulation. Multiple test vectors, including boundary conditions and random input combinations, are applied to ensure accurate output generation. Waveform analysis is used to verify timing behavior and logical correctness.

B. Synthesis Results

The verified RTL design is synthesized using standard cell libraries with timing and area constraints. Synthesis reports indicate that the proposed UT-based multiplier achieves reduced propagation delay due to its parallel computation structure. The hierarchical architecture contributes to efficient resource utilization and improved timing performance.

Area reports demonstrate that the modular UT-based design maintains competitive silicon area when compared to conventional multiplier architectures. Power analysis further confirms that the proposed design operates within acceptable power limits, making it suitable for low-power and high-performance applications.

Post-layout timing analysis confirms that the design meets timing constraints without violations. Power analysis reveals balanced dynamic and leakage power consumption due to optimized logic structure and reduced switching activity. These results collectively demonstrate the effectiveness of the proposed UT-based multiplier for large-bit-width arithmetic operations.

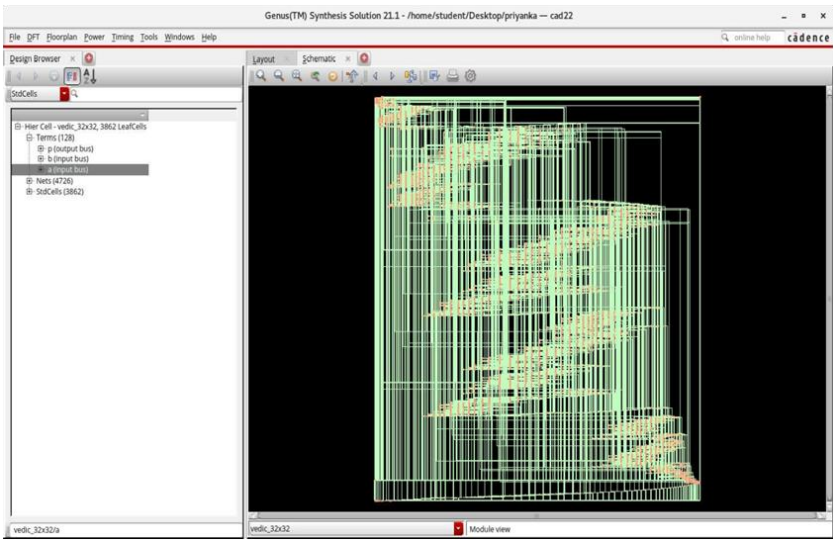

Fig. 2: Functional Simulation Result of the Proposed 64-bit Multiplier

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15119

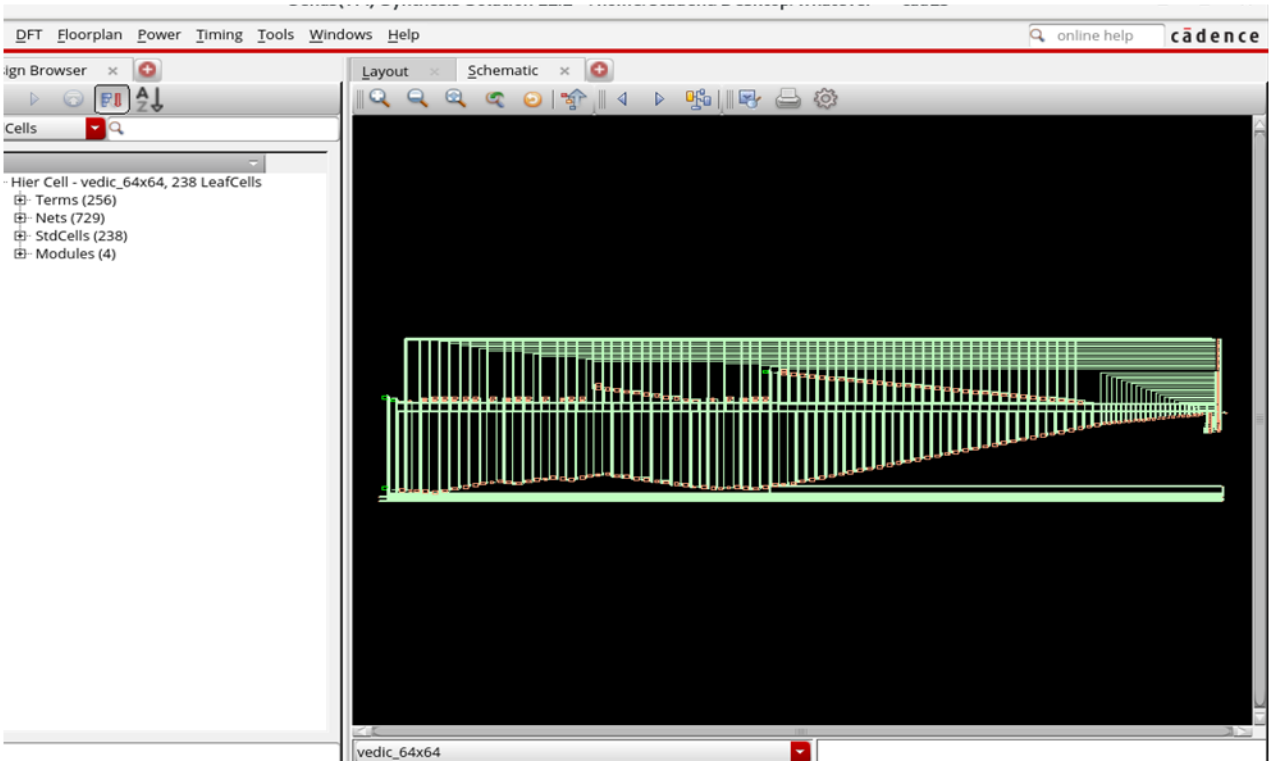

Fig. 3: Synthesis Report Showing Area Utilization

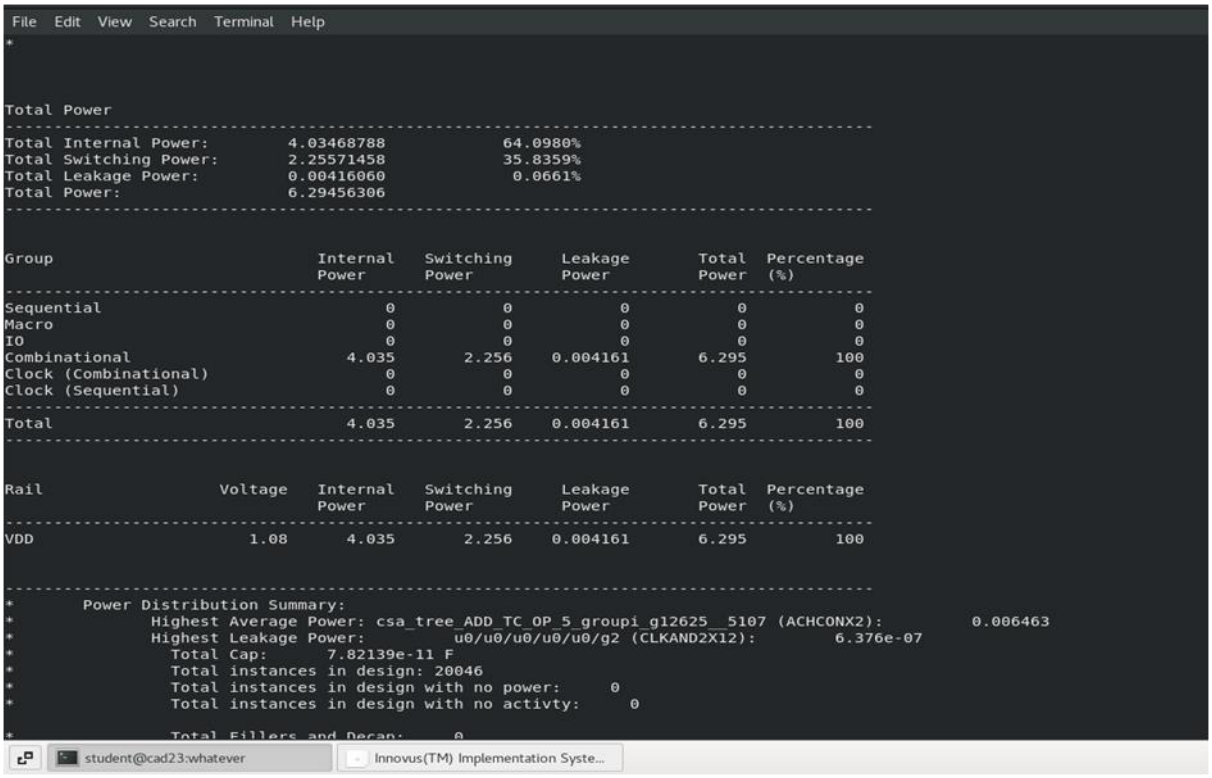

Fig. 4: Power Consumption Analysis of the Proposed Multiplier

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15119

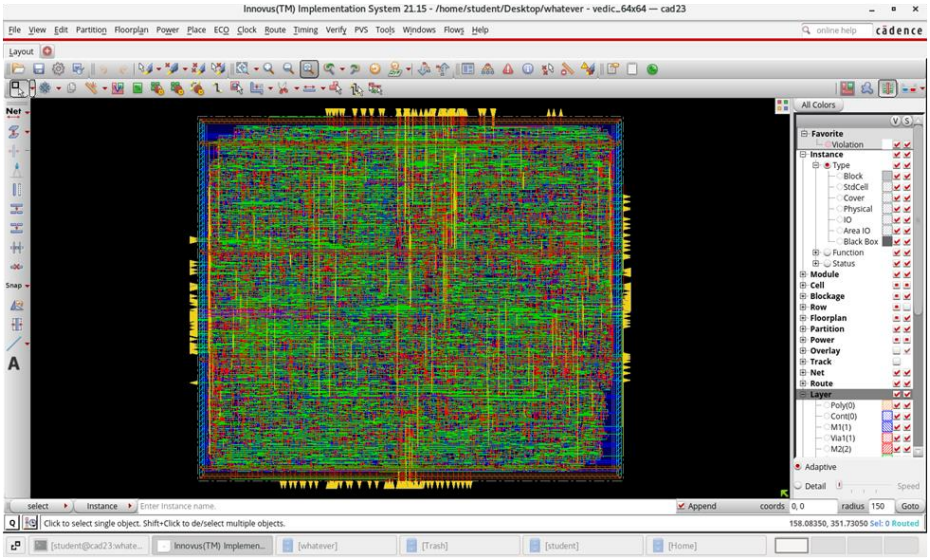

Fig. 5: Implementation of Layout Design of the Proposed Design

C. Performance Analysis

Performance evaluation is carried out based on key metrics such as propagation delay, area utilization, and power consumption. The parallel generation of partial products significantly reduces the critical path delay, resulting in faster multiplication compared to traditional designs.

D. Discussion

The experimental results validate the advantages of integrating the Urdhva Tiryakbhayam algorithm with modern VLSI design methodologies. Compared to conventional multipliers, the proposed architecture achieves improved speed while maintaining efficient area and power characteristics.

The hierarchical and modular design approach enhances scalability and simplifies verification and physical implementation. The results confirm that Vedic Mathematics-based algorithms can be effectively employed in contemporary VLSI systems to meet the growing demand for high-speed and energy-efficient arithmetic units.

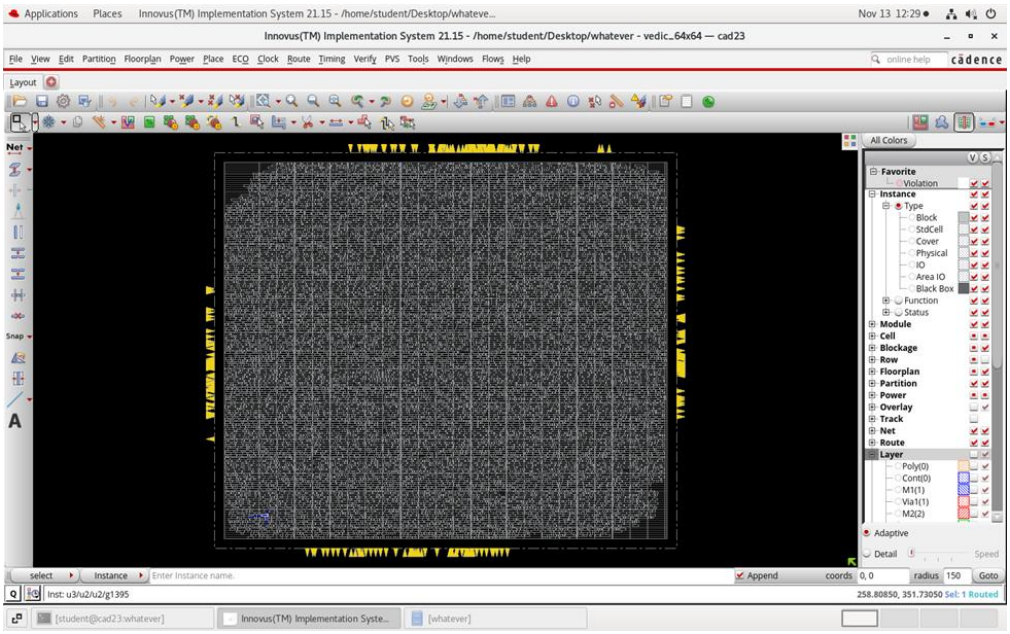

Fig. 6: Physical Layout of the 64-bit Urdhva Tiryakbhayam Multiplier

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15119

V. CONCLUSION AND FUTURE WORK

This paper presented the design and VLSI implementation of a 64-bit multiplier based on the Urdhva Tiryakbhayam Sutra of Vedic Mathematics. The proposed architecture exploits the inherent parallelism of the UT algorithm to achieve high-speed multiplication while maintaining efficient power consumption and area utilization. A hierarchical and modular design approach was adopted to enhance scalability and simplify verification.

The multiplier was modeled using Verilog HDL and verified through extensive functional simulation. Synthesis and physical implementation results demonstrated that the proposed UT-based multiplier outperforms conventional multiplier architectures in terms of propagation delay while maintaining competitive area and power characteristics. The experimental results confirm that the UT algorithm is well suited for large- bit-width arithmetic operations in modern VLSI systems.

The successful integration of ancient mathematical principles with contemporary VLSI design methodologies highlights the practical viability of Vedic Mathematics-based arithmetic circuits. Future work may focus on further optimization techniques such as pipelining, clock gating, and power-aware design strategies, as well as extending the architecture for FPGA and ASIC-based implementations.

REFERENCES

[1]. More, T.V. and Panat, A.R., FPGA Implementation of FFT Processor Using Vedic Algorithm. In 2013 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-5). IEEE, December 2013.

[2]. V. Nandhini, “Implementation of Normal Urdhva Tiryakbhayam Multiplier in VLSI,” 2017.

[3]. P. S. Aswale, “Multiplier Design Using Vedic Mathematics,” 2017.

[4]. V. G. Sarashetti, “Signed 32-bit Vedic Multiplier Using Urdhva Tiryakbhayam Sutra,” 2021.

[5]. T. V. More and A. R. Panat, “FPGA Implementation of FFT Processor Using Vedic Algorithm,” IEEE, 2013.

[6]. Anjana, R., Abishna, B., Harshitha, M.S., Abhishek, E., Ravichandra, V., and Suma, M.S., Implementation of Vedic Multiplier Using Kogge-Stone Adder. In 2014 International Conference on Embedded Systems (ICES), pp. 28-31, IEEE, July 2014.

[7]. Sangani, H., Modi, T.M., and Bhaaskaran, V.K., Low Power Vedic Multiplier Using Energy Recovery Logic. In 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 640-644, IEEE, September 2014.

