Design And Dvelopment Of 64-Bit Alu Desing Using Vedic Mathematics

Dr. M. Shafi[1], Telukuttu Lakshmi Koteswari[2], Shaik Amreen[3], Putta Srinadh[4], Rachabanti Naveen[5],

[2,3,4,5]UG Student,ECE,Chalapathi Institute Of Engineering&Technology Guntur-Andhra Pradesh,India

[1] Professor ECE,Chalapathi Institute Of Engineering&Technology Guntur-Andhra Pradesh,India

Abstract—**The Arithmetic Logic Unit (ALU) is a fundamental component of modern processors, responsible for performing arithmetic and logical operations essential to computational systems. With the increasing demand for high-speed and low-power digital systems, efficient ALU design has become a critical area of research in VLSI technology. This paper presents the design and implementation of a 64-bit ALU using Vedic Mathematics, aimed at enhancing computational speed and reducing hardware complexity.The proposed design incorporates Vedic multiplication techniques, particularly the *UrdhvaTiryagbhyam* (vertical and crosswise) algorithm, to optimize arithmetic operations. This approach enables parallel generation of partial products, thereby significantly reducing propagation delay compared to conventional multiplication methods. The ALU supports multiple operations, including addition, subtraction, multiplication, division, and logical functions, with improved performance characteristics.The design is implemented using Hardware Description Language (HDL) and verified through simulation tools. Synthesis results demonstrate that the proposed ALU achieves better speed, reduced latency, and efficient resource utilization when compared to traditional ALU architectures.**

Keywords— Arithmetic Logic Unit (ALU), Vedic Mathematics, UrdhvaTiryagbhyam, 64-bit ALU, VLSI Design, High-Speed Computing, Low Power Consumption, Hardware Description Language (HDL), FPGA Implementation, Parallel Processing, Digital Signal Processing (DSP), Pipelining.

I. INTRODUCTION

In modern digital systems, the demand for high-speed and energy-efficient computation has increased significantly due to rapid advancements in technologies such as embedded systems, signal processing, and communication networks. At the core of every processor lies the **Arithmetic Logic Unit (ALU)**, which performs essential arithmetic and logical operations. The performance of the ALU directly influences the overall efficiency, speed, and power consumption of the computing system.

With the evolution of Very Large Scale Integration (VLSI) technology, designers are constantly seeking methods to optimize computational units in terms of delay, area, and power. Among all arithmetic operations, multiplication is one of the most critical and time-consuming processes, as it involves multiple stages such as partial product generation, reduction, and final addition. Therefore, improving multiplier efficiency plays a vital role in enhancing ALU performance.

Traditional multiplier architectures, such as array multipliers and Booth multipliers, often suffer from higher propagation delay and increased hardware complexity. To overcome these limitations, alternative computational techniques are being explored. One such promising approach is **Vedic Mathematics**, an ancient system of mathematics that provides efficient algorithms for arithmetic operations. The *UrdhvaTiryagbhyam* (vertical and crosswise) method, in particular, enables fast multiplication by generating partial products in parallel, thereby reducing computation time.

In this work, a **64-bit ALU design using Vedic Mathematics** is proposed to achieve high-speed and efficient computation. The design integrates Vedic multiplication techniques with optimized adder structures to support various arithmetic and logical operations. The implementation is carried out using Hardware Description Language (HDL) and verified through simulation and synthesis tools, ensuring accuracy and performance.

The objective of this paper is to demonstrate that incorporating Vedic mathematical principles into digital design can significantly improve the speed and efficiency of ALU operations while maintaining low power consumption and reduced hardware complexity. This makes the proposed design suitable for applications in digital signal processing, high-performance computing, and modern microprocessor architectures.

Doi:10.48047/ijiee.2026.16.2.33

II. REVIEW LITERATURE SURVEY

The design of high-performance Arithmetic Logic Units (ALUs) and multipliers has been an active area of research in VLSI systems, with a strong focus on improving speed, reducing power consumption, and optimizing hardware utilization. Since multiplication is one of the most time-consuming operations in digital systems, many researchers have concentrated on developing efficient multiplier architectures.

Several studies have explored advanced multiplier designs based on compression techniques and parallel processing. For instance, Wallace tree and Dadda multipliers have been widely used to reduce partial products and improve computational speed. Research works have demonstrated that the use of compressors (such as 4:2 and 15:4 compressors) significantly reduces delay and power consumption while maintaining acceptable accuracy levels.

In addition, approximate computing techniques have gained attention for applications where slight inaccuracies are tolerable. These methods reduce hardware complexity and power consumption, making them suitable for multimedia and image processing applications. Studies have shown that approximate multipliers can achieve notable reductions in delay and energy usage while maintaining high structural similarity in output quality.

Parallel prefix adders, such as Kogge-Stone and Brent-Kung adders, have also been extensively studied to improve addition speed in multipliers. These adders reduce carry propagation delay and enhance overall system performance. Research indicates that replacing conventional adders with parallel prefix structures can lead to significant improvements in speed and efficiency.

Another important area of research is the application of Vedic Mathematics in digital design. The *UrdhvaTiryagbhyam* algorithm has been widely used to design fast multipliers due to its ability to generate partial products in parallel. Studies have demonstrated that Vedic multipliers, when combined with carry save adders, offer faster computation compared to traditional ripple carry-based designs.

Furthermore, pipelining techniques have been employed to enhance throughput in digital circuits. By dividing computations into multiple stages, pipelined multipliers reduce critical path delay and improve performance. Research has also explored the trade-offs between pipeline depth, power consumption, and area utilization.

Recent advancements include FPGA-based implementations of multipliers and ALUs, where hardware description languages such as Verilog and VHDL are used for design and simulation. These implementations provide flexibility, reconfigurability, and faster prototyping for complex digital systems.

Despite these developments, challenges such as high delay, increased power consumption, and hardware complexity still persist in conventional ALU designs. Therefore, integrating efficient algorithms like Vedic Mathematics with optimized architectures presents a promising solution. The proposed 64-bit ALU design aims to address these limitations by combining high-speed Vedic multiplication with efficient hardware implementation techniques.

III. RESEARCH METHODOLOGY

The research methodology adopted in this work focuses on the systematic design, development, and evaluation of a **64-bit Arithmetic Logic Unit (ALU) using Vedic Mathematics** to achieve high-speed and efficient computation. The study begins with identifying the limitations of conventional ALU architectures, particularly the high propagation delay and increased hardware complexity associated with arithmetic operations such as multiplication. Since multiplication is one of the most critical and time-consuming operations in digital systems, improving its performance directly enhances the efficiency of the entire ALU. To address these challenges, Vedic Mathematics is chosen as an alternative computational approach due to its inherent capability for parallel processing and reduced computational steps.

The *UrdhvaTiryagbhyam* (vertical and crosswise) algorithm is selected as the core technique for implementing multiplication within the ALU. This algorithm enables simultaneous generation of partial products, significantly reducing the delay compared to conventional methods such as shift-and-add or Booth multiplication. The algorithm is extended and optimized for 64-bit operations by decomposing the design into smaller modules, such as 4-bit, 8-bit, or 16-bit Vedic multipliers, which are then combined hierarchically to form a high-bit architecture. This modular approach improves scalability, simplifies design complexity, and enhances reusability.

Doi:10.48047/ijiee.2026.16.2.33

The overall ALU architecture is designed using a structured and hierarchical methodology, where the system is divided into functional blocks including arithmetic units (adder, subtractor, multiplier, and divider) and logical units (AND, OR, XOR, NOT operations). Efficient adder designs, such as carry look-ahead or carry save adders, are incorporated to reduce carry propagation delay and improve overall performance. Control logic is also developed to select the desired operation based on input control signals, ensuring flexibility in performing multiple operations within a single ALU framework.

The design is implemented using **Hardware Description Language (HDL)**, typically Verilog, to model the behavior and structure of the ALU at the Register Transfer Level (RTL). The RTL design is simulated using tools such as ModelSim or ISim to verify functional correctness. Test benches are created to validate different arithmetic and logical operations under various input conditions, ensuring accuracy and reliability of the design. Following simulation, synthesis is carried out using Xilinx tools (such as Xilinx ISE 14.5), where the design is mapped onto FPGA hardware to evaluate parameters such as area utilization, timing delay, and power consumption.

Further, performance analysis is conducted by comparing the proposed Vedic-based ALU with conventional ALU designs. Metrics such as propagation delay, resource utilization (LUTs, slices), and computational speed are considered. The results demonstrate that the proposed design achieves reduced latency and improved throughput due to parallel processing and optimized architecture. Additionally, pipelining techniques may be incorporated to further enhance performance by dividing operations into multiple stages, thereby increasing the throughput of the system.

Finally, the methodology emphasizes achieving a balance between speed, area, and power consumption. The proposed approach ensures that the ALU is not only faster but also efficient in terms of hardware resources, making it suitable for applications in digital signal processing (DSP), embedded systems, and high-performance computing. This structured methodology validates that integrating Vedic Mathematics into ALU design provides a practical and effective solution for modern VLSI systems.

IV. PROPOSED METHODOLOGY

The proposed methodology focuses on the design and implementation of a 64-bit Arithmetic Logic Unit (ALU) using Vedic Mathematics to achieve high-speed, low-delay, and efficient hardware utilization. The approach is centered on integrating the *UrdhvaTiryagbhyam* (vertical and crosswise) algorithm for multiplication, which enables parallel generation of partial products and significantly reduces computation time compared to conventional multiplication techniques. Unlike traditional methods that rely on sequential processing, the Vedic approach improves speed by allowing simultaneous operations, making it highly suitable for high-bit architectures.

The design follows a modular and hierarchical structure, where the 64-bit ALU is constructed using smaller building blocks such as 4-bit, 8-bit, and 16-bit modules. These smaller units are combined to form a complete 64-bit system, ensuring scalability, reduced design complexity, and ease of implementation. The multiplier, being the most critical component, is designed using Vedic principles, while efficient adder structures such as carry look-ahead or carry save adders are used to minimize carry propagation delay. This combination enhances the overall speed and performance of the ALU.

The proposed ALU supports multiple operations including addition, subtraction, multiplication, division, and logical functions such as AND, OR, XOR, and NOT. A control unit is incorporated to select the desired operation based on input signals, enabling flexible functionality within a single architecture. Special attention is given to optimizing data paths and reducing critical path delay to ensure faster execution of operations.

The entire system is modeled using **Hardware Description Language (HDL)** at the Register Transfer Level (RTL). The design is simulated using tools such as ModelSim or ISim to verify functional correctness. Comprehensive test benches are developed to validate different operational scenarios and ensure accurate results. After simulation, synthesis is carried out using Xilinx tools to analyze hardware parameters such as area utilization, timing performance, and power consumption.

To further enhance performance, **pipelining techniques** may be incorporated, allowing multiple operations to be executed concurrently in different

Doi:10.48047/ijiee.2026.16.2.33

stages, thereby increasing throughput. The performance of the proposed design is evaluated and compared with conventional ALU architectures, demonstrating improvements in speed, reduced latency, and efficient resource usage.

Overall, the proposed methodology provides an optimized solution for designing high-performance ALUs by combining Vedic mathematical principles with modern VLSI design techniques, making it suitable for advanced applications in digital signal processing, embedded systems, and high-speed computing environments.

V. BLOCK DIAGRAM

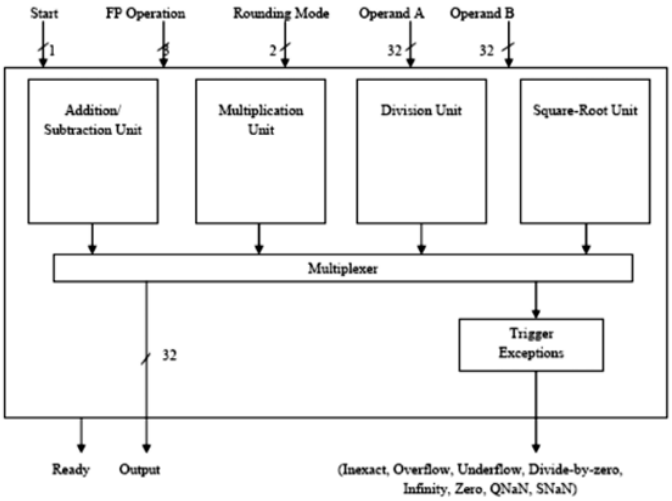


Fig. 6.2. Block Diagram

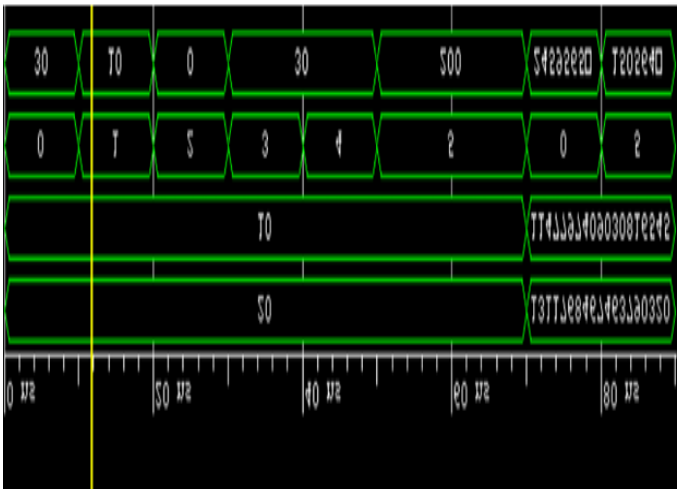
VI. RESULTS AND OUTCOMES

The proposed **64-bit ALU using Vedic Mathematics** was successfully designed, simulated, and synthesized using HDL and Xilinx tools. The functional verification results confirm that the ALU correctly performs all arithmetic operations such as addition, subtraction, multiplication, division, and logical operations including AND, OR, XOR, and NOT under different input conditions. The simulation waveforms validate the correctness and reliability of the design for 64-bit data processing.

Fig: 7.1: Output 1

The implementation results show that the use of the *UrdhvaTiryagbhyam* algorithm significantly improves the speed of multiplication by enabling parallel generation of partial products. This leads to a considerable reduction in propagation delay compared to conventional multiplier-based ALU designs. The synthesis report indicates optimized hardware utilization in terms of Look-Up Tables (LUTs), slices, and logic elements, demonstrating efficient resource usage.

Timing analysis reveals that the proposed ALU achieves lower critical path delay, resulting in faster computation and higher throughput. Additionally, the modular design approach contributes to scalability and ease of implementation for higher bit-width systems. Power consumption is also reduced due to minimized switching activity and optimized logic structures, making the design suitable for low-power applications.



Doi:10.48047/ijiee.2026.16.2.33

Fig : 7.2: Output2

Furthermore, when compared with traditional ALU architectures, the proposed design shows improved performance in terms of speed, reduced latency, and better area efficiency. The integration of pipelining techniques further enhances system throughput by allowing multiple operations to be processed simultaneously at different stages.

Overall, the outcomes demonstrate that incorporating Vedic Mathematics into ALU design provides a high-performance, reliable, and efficient solution for modern digital systems. The proposed ALU is well-suited for applications in Digital Signal Processing (DSP), embedded systems, and high-speed computing environments where fast and efficient arithmetic operations are essential.

VII. CONCLUSION

This paper presented the design and implementation of a **64-bit Arithmetic Logic Unit (ALU) using Vedic Mathematics**, with the primary objective of improving computational speed, reducing propagation delay, and optimizing hardware resource utilization. The study focused on overcoming the limitations of conventional ALU architectures, particularly in multiplication operations, which significantly affect overall system performance. By adopting the *UrdhvaTiryagbhyam* (vertical and crosswise) algorithm, the proposed design achieves parallel generation of partial products, thereby enhancing execution speed and reducing computational complexity.

The implementation of the ALU using a modular and hierarchical approach proved to be highly effective, as it allowed the construction of a complex 64-bit system from smaller, reusable building blocks. This not only simplified the design process but also improved scalability and flexibility for future enhancements. The integration of efficient adder structures further minimized carry propagation delay, contributing to faster arithmetic operations.

Simulation and synthesis results confirmed that the proposed ALU performs all arithmetic and logical operations accurately while achieving improved performance metrics. Compared to traditional ALU designs, the proposed system demonstrated reduced delay, efficient utilization of hardware resources, and lower power consumption. The use of HDL-based design and FPGA implementation ensured practical feasibility and ease of testing, making the design suitable for real-world applications.

Moreover, the incorporation of Vedic Mathematics highlights the potential of ancient computational techniques in modern digital system design. The ability to perform operations in parallel makes Vedic-based architectures highly suitable for high-speed and high-performance applications such as Digital Signal Processing (DSP), embedded systems, and advanced microprocessor design.

In conclusion, the proposed 64-bit ALU design provides an efficient, scalable, and high-performance solution for modern computing requirements. Future work can focus on further optimization techniques such as advanced pipelining, low-power design strategies, and implementation using emerging technologies to enhance performance even further. Additionally, extending this approach to higher bit-width architectures and integrating it into complete processor designs can open new possibilities in high-speed computing systems.

REFERENCES

1. Vellela, S. S., & Balamanigandan, R. (2024). Optimized clustering routing framework to maintain the optimal energy status in the wsn mobile cloud environment. Multimedia Tools and Applications, 83(3), 7919-7938.
2. Vellela, S. S., & Balamanigandan, R. (2023). An intelligent sleep-awake energy management system for wireless sensor network. Peer-to-Peer Networking and Applications, 16(6), 2714-2731.
3. Vellela, S. S., & Balamanigandan, R. (2024). An efficient attack detection and prevention approach for secure WSN mobile cloud environment. Soft Computing, 28(19), 11279-11293.
4. Vellela, S. S. (2023). Enhanced speckle noise reduction in breast cancer ultrasound imagery using a hybrid deep learning model. Ingénierie des Systèmes d'Information.
5. Polasi, P. K., Vellela, S. S., Narayana, J. L., Simon, J., Kapileswar, N., Prabu, R. T., & Rashed, A. N. Z. (2026). Data rates transmission, operation performance speed and figure of merit signature for various quadurature light sources under spectral and thermal effects. Journal of Optics, 55(1), 633-643.

Doi:10.48047/ijiee.2026.16.2.33

6. Praveen, S. P., Nakka, R., Chokka, A., Thatha, V. N., Vellela, S. S., & Sirisha, U. (2023). A novel classification approach for grape leaf disease detection based on different attention deep learning techniques. International Journal of Advanced Computer Science and Applications (IJACSA), 14(6), 2023.
7. Vellela, S. S., Rao, M. V., Mantena, S. V., Reddy, M. J., Vatambeti, R., & Rahman, S. Z. (2024). Evaluation of Tennis Teaching Effect Using Optimized DL Model with Cloud Computing System. International Journal of Modern Education and Computer Science (IJMECS), 16(2), 16-28.
8. Vellela, S. S., & Krishna, A. M. (2020). On Board Artificial Intelligence With Service Aggregation for Edge Computing in Industrial Applications. Journal of Critical Reviews, 7(07).
9. Madhuri, A., Jyothi, V. E., Praveen, S. P., Sindhura, S., Srinivas, V. S., & Kumar, D. L. S. (2024). A new multi-level semi-supervised learning approach for network intrusion detection system based on the 'goa'. Journal of Interconnection Networks, 24(supp01), 2143047.
10. Raju, V. V. K., Bhavani, Y. V. K. D., Nandikonda, P., Kareemunnisa, F. N. U., Brahmeswara, K. B., & Sindhura, S. (2026). Iterative and Statistical Analytical Review of Predictive Modeling Approaches in Educational Systems: A Comprehensive Benchmark of AI-Driven Methods. International Journal of Innovative Technology and Interdisciplinary Sciences, 9(1), 490-522.
11. Biyyapu, N., Veerapaneni, E. J., Surapaneni, P. P., Vellela, S. S., & Vatambeti, R. (2024). Designing a modified feature aggregation model with hybrid sampling techniques for network intrusion detection. Cluster Computing, 27(5), 5913-5931.
12. Praveen, S. P., Vellela, S. S., & Balamanigandan, R. (2024). SmartIris ML: harnessing machine learning for enhanced multi-biometric authentication. Journal of Next Generation Technology (ISSN: 2583-021X), 4(1).
13. Vuyyuru, L. R., Purimetla, N. R., Reddy, K. Y., Vellela, S. S., Basha, S. K., & Vatambeti, R. (2025). Advancing automated street crime detection: a drone-based system integrating CNN models and enhanced feature selection techniques. International Journal of Machine Learning and Cybernetics, 16(2), 959-981.
14. Vellela, S. S., Roja, D., Purimetla, N. R., Thalakola, S., Vuyyuru, L. R., & Vatambeti, R. (2025). Cyber threat detection in industry 4.0: Leveraging GloVe and self-attention mechanisms in BiLSTM for enhanced intrusion detection. Computers and Electrical Engineering, 124, 110368.
15. Vellela, S. S., Pushpalatha, D., Sarathkumar, G., Kavitha, C. H., & Harshithkumar, D. (2023). Advanced intelligence health insurance cost prediction using random forest. ZKG International, 8.
16. Vellela, S. S., Babu, B. V., & Mahendra, Y. B. (2024). IoT-based tank water monitoring systems: enhancing efficiency and sustainability. International Journal for Modern Trends in Science and Technology, 10(02), 291-298.
17. Vellela, S. S., Varshini, K., Jeevana, M., Kadheer, S. K., & Kumar, T. P. (2024). Iot based smart irrigation and controlling system. IoT Based Smart Irrigation and Controlling System, International Journal for Modern Trends in Science and Technology, 10(02), 77-85.
18. Vellela, S. S., Chaganti, A., Gadde, S., Bachina, P., & Karre, R. (2022). A Novel Approach for Detecting Automated Spammers in Twitter. Mukt Shabd, 11, 49-53.
19. Vellela, S. S., Narapasetty, S., Somepalli, M., Merikapudi, V., & Pathuri, S. (2022). Fake News Articles Classifying Using Natural Language Processing to Identify in-article Attribution as a Supervised Learning Estimator. Mukt Shabd Journal, 11.
20. Vellela, S. S., Vineeth, S., & Suresh, V. (2024). IoT Based ICU Patient Monitoring System. IoT Based ICU Patient Monitoring System, International Journal for Modern Trends in Science and Technology, 10(02), 265-273.

Doi:10.48047/ijiee.2026.16.2.33

21. Vellela, S. S., & Balamanigandan, R. (2025). Designing a Dynamic News App Using Python. Available at SSRN 5250912.
22. Vellela, S. S., Rao, M. V., Krishna, C. V. M., Rao, T. S., & Dasthavejula, R. (2026). Piezoelectric and Shape-Memory Materials for Actuators and Energy Harvesting in Mechanical, Electronics, and Biomedical Engineering Using AI-Based Design. In Advanced Materials for Biomedical Devices (pp. 195-206). CRC Press.
23. Vellela, S. S., Singu, K., Kakarla, L. S., Tadikonda, P., & Sattenapalli, S. N. R. (2025). NLP-Driven Summarization: Efficient Extraction of Key Information from Legal and Financial Documents. Available at SSRN 5250908.
24. Vellela, S. S., Anusha, P., Vullam, N. R., Jala, J., Bellapu, V. S., & Vindhya, A. S. (2025, October). Quantum Cryptography and Key Distribution for Secure Communication in the Post Quantum World. In 2025 International Conference on Sustainable Communication Networks and Application (ICSCN) (pp. 619-624). IEEE.
25. Roja, D., Jidugu, S. K., Rao, T. S., Vuyyuru, L. R., Vellela, S. S., & Ranjani, B. S. (2025, December). High-Fidelity Image Synthesis using Enhanced Generative Adversarial Networks with Attention Mechanisms. In 2025 International Conference on NexGen Networks and Cybernetics (IC2NC) (pp. 885-890). IEEE.
26. Vellela, S. S., Vuyyuru, L. R., Jidugu, S. K., Rao, M. P., & Srinivas, B. R. (2025, November). The Impact Of Quantum Computing On Blockchain Security And Quantum Resistant Protocols. In 2025 2nd International Conference on Intelligent Systems for Cybersecurity (ISCS) (pp. 1-6). IEEE.
27. Yanamadala, N., & Vellela, S. S. (2025, June). Ensuring Authenticity and Confidentiality in Images using SHA-ECC Fusion. In 2025 Second International Conference on Networks and Soft Computing (ICNSoC) (pp. 684-689). IEEE.
28. Vellela, S. S. (2024). A Comprehensive Review of AI Techniques in Serious Games: Decision Making and Machine Learning.
29. Burra, R. S., APCV, G. R., & Vellela, S. S. (2024). Strategic Insights: Unleashing the Power of Big Data Analytics for Credit Investigation and Risk Mitigation in Commercial Banking. International Journal of Progressive Research in Engineering Management and Science, 4(01), 458-464.
30. Vellela, S. S., Purimetla, N. R., Vindhya, A. S., Vullam, N. R., Srinivas, B. R., & Vuyyuru, L. R. (2025, October). Design and Simulation of Quantum Error Correction Codes for Scalable Quantum Architectures. In 2025 7th International Conference on Innovative Data Communication Technologies and Application (ICIDCA) (pp. 1570-1575). IEEE.
31. Vellela, S. S., Purimetla, N. R., Rao, P. V., Daniel, V. A. A., Koppolu, H. K. R., & Janani, B. (2025). AI-Enabled Wearable Hemodynamic Monitoring System for Early Identification of Thrombotic Events. Vascular and Endovascular Review, 8(16s), 321-336.
32. Venkatesh, N., Maheswari, S., & Triveni, P. (2024). Harnessing IoT for Real-Time Plant Health Monitoring: Challenges and Opportunities.
33. Reddy, B. V., Kumar, A. H., Gopi, C., Prasad, Y. V. D., Vellela, S. S., & Roja, D. (2025, April). Machine learning based automated liver fibrosis stage diagnosis with prediction. In 2025 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE) (pp. 1-6). IEEE.
34. Rao, M. V., Sreeraman, Y., Mantena, S. V., Gundu, V., Roja, D., & Vatambeti, R. (2024). Brinjal Crop yield prediction using Shuffled shepherd optimization algorithm based ACNN-OBDLSTM model in Smart Agriculture. Journal of Integrated Science and Technology, 12(1), 710-710.
35. Haritha, K., Geethika, N. S., Venkateswarlu, K., Kumar, R. H., & Ramakrishna, Y. Enhancing Public Safety with AI & ML-Based CCTV Surveillance.

Doi:10.48047/ijiee.2026.16.2.33

36. Haritha, K., Prakash, P. B., Pravallika, D., Venkatesh, K., & Venkatesh, G. Enhancing Object Detection in Autonomous Vehicles Under Low-Light Conditions Using Federated Learning and YOLOv5.
37. Ram, C. S., Vellela, S. S., Sravanthi Javvadi, D. V., Rashid, S. Z., & Madhumathi, S. M. (2025). Integrated Robotic–Imaging Platforms in Endovascular Surgery: Current Capabilities and Future Directions. Vascular and Endovascular Review, 8(16s), 285-298.
38. Roja, D., Navya, G., Srujana, B. S., Mamatha, P., & Sai, C. Y. K. Deep Learning for Hotel Reviews: A Framework for Sentiment Classification and Fake Review Detection.
39. Pakalapati, S., Rani, C. J., Vellela, S. S., Thanuja, N., & Bindu, M. N. H. (2025, November). Progressive GAN-based Framework for Realistic Image Generation and Style Transfer. In 2025 5th International Conference on Evolutionary Computing and Mobile Sustainable Networks (ICECMSN) (pp. 474-479). IEEE.
40. Balamanigandan, R., Vellela, S. S., Gorintla, S., Vuyyuru, L. R., Thanuja, N., & Rao, T. S. (2025, September). Quantum-Enhanced Data Security for Electronic Health Records: A Framework for Post-Quantum Cryptography in Healthcare Systems. In 2025 6th International Conference on Smart Electronics and Communication (ICOSEC) (pp. 1924-1929). IEEE.
41. Roja, D., Amulya, P., Nagasai, M., Prasad, D. D., & Babu, A. V. Machine Learning-Based Early Diagnosis of Fish Diseases via Water Quality Data.
42. Sai, M. B., & Vellela, S. S. (2025, December). Hybrid ML Driven Multi-Cloud Service Work Load Prediction For Financial Systems. In 2025 1st International Conference on Advancement in Futuristic Technologies (ICAFT) (pp. 1-6). IEEE.
43. Kareemunnisa, D., Haritha, K., Ranjani, B. S., Venkateswarlu, K., & Bindu, M. N. H. DUAL-STAGE PRIVACY PROTECTION FOR GRAPH NEURAL NETWORKS AGAINST INFERENCE ATTACKS.
44. Mandava, R., Haritha, K., Vellela, S. S., Purimetla, N. R., Mohan, B. K., & Harinadh, T. (2025, June). Analysing User Perceptions of Trust in Financial Systems Using Explainable AI. In 2025 Second International Conference on Networks and Soft Computing (ICNSoC) (pp. 26-30). IEEE.

Doi:10.48047/ijiee.2026.16.2.33