Comparative Study of 16-bit Ripple Carry Adder and Weinberger Adder Architectures in Digital Circuit Design

Srinithi R.M[1], Bhavatarini M[2], Bojja Divya[3]

Geetha Sri.P,[4]

[1]srinithi.rm2023@vitstudent.ac.in

[2]bhavatarini.m2023@vitstudent.ac.in

[3]bojja.divya2023@vitstudent.ac.in

[4]peddintigeetha.sri2023@vitstudent.ac.in

[*]*SENSE VIT Chennai, SENSE VIT Chennai, SENSE VIT Chennai, SENSE VIT Chennai*

Abstract— **This paper designs and tests high-performance, energy-efficient VLSI adders of varying bit-widths, including 64- bit using advanced CMOS technology. In digital arithmetic, adders play a critical role, particularly in VLSI and high-speed computing. A simple design is the 16-bit Ripple Carry Adder, where carry propagates sequentially across all the bit positions with the delay being a function of bit-width. On the other hand, 16-bit Weinberger Adder has better performance by parallel processing with generate (G) and propagate (P) signals. The delay caused by carry reduces to a large extent , making it more suitable for applications requiring high-speed operations such as DSPs and processors. RCA is simpler and area-efficient; however, its delay limitations give Weinberger Adder a better scope in modern computing, which demands high speed**

Keywords— **Ripple Carry Adder (RCA),Weinberger Adder ,VLSI Design ,High-Speed Arithmetic, CMOS Technology**

I. INTRODUCTION

In modern digital systems, arithmetic operations are fundamental to the performance and efficiency of processors, digital signal processors (DSPs), and application-specific integrated circuits (ASICs). Among these operations, addition is the most frequently used, making the design of fast and efficient adders a critical area in digital circuit design. Digital adders are integral components in arithmetic logic units (ALUs), where speed, area, and power consumption directly influence overall system performance.

The Ripple Carry Adder (RCA) is one of the simplest and most intuitive adder architectures. It operates by allowing the carry generated at each bit position to propagate sequentially through all higher-order bits. Although this approach minimizes hardware complexity, it suffers from significant propagation delay, especially as the bit-width increases.

To overcome these limitations, more advanced adder architectures such as the Weinberger Adder have been proposed. The Weinberger Adder utilizes generate (G) and propagate (P) signals to calculate carry bits in parallel, dramatically reducing delay. This makes it especially suitable for high-speed applications where performance is critical, such as in real-time DSP and high-end computing systems.

The purpose of this study is to conduct a comparative analysis between the 16-bit Ripple Carry Adder and the 16-bit Weinberger Adder in terms of key performance metrics—namely speed, area, and power consumption. By implementing and simulating both designs using modern VLSI tools and CMOS technology, this paper aims to highlight the trade-offs involved and guide design decisions based on application requirements.

II. LITERATUR REVIEW

A. Limitations of ripple carry adder

Ripple Carry Adders (RCAs) are widely known for their simplicity and area efficiency, making them a popular choice in basic digital systems and low-power applications [1]. However, their key limitation lies in performance: the carry bit must propagate sequentially through all stages, leading to significant delay as the bit-width increases [2].

While RCAs are low-power and low-cost, they become inefficient as the demand for faster arithmetic operations grows, particularly in real-time and high-performance computing systems [3]. This sequential nature of RCAs limits their scalability in advanced VLSI designs, where fast computation is critical.

B. Advantages of Weinberger Adders and Parallel Techniques

To overcome the delays in RCAs, the **Weinberger Adder** employs **generate (G)** and **propagate (P)** signals, allowing carry computation to be done in parallel [4]. This significantly reduces carry propagation delay, resulting in faster addition, particularly in wider bit-width designs adders.

The Weinberger Adder's parallel approach makes it more suitable for applications where high speed is essential, such as **Digital Signal Processors (DSPs)**, **Arithmetic Logic Units (ALUs)**, and other high-performance computing devices [5].

III. 16 Bit Adder Architecture

The architecture of 16-bit adders plays a vital role in determining the performance and efficiency of arithmetic units in digital systems. The Ripple Carry Adder (RCA) remains one of the most straightforward and widely used adder architectures due to its simplicity and ease of implementation. In the RCA design, each full adder generates a sum and a carry-out which propagates sequentially to the next stage. Although this linear propagation introduces a delay proportional to the number of bits, the RCA remains a popular choice in low-power and area-constrained applications due to its minimal complexity. In a 16-bit RCA, the carry-out from each stage must wait for the computation of the previous stage, leading to a cumulative propagation delay across all 16 stages. Despite this, its regular layout and predictable structure allow for straightforward physical implementation and integration into larger digital systems.

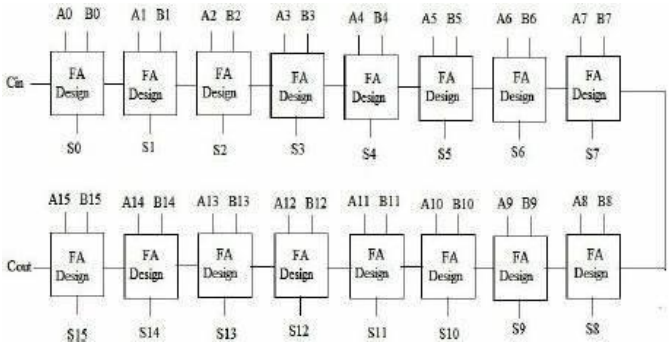

Fig.1 16 bit RCA

In contrast, the Weinberger adder architecture enhances performance by reducing the propagation delay associated with carry generation. The Weinberger adder is based on a carry-save adder approach where intermediate results are stored without immediate carry propagation. This architecture is particularly effective in pipelined structures and multi-operand addition systems, where delay reduction is critical. In a 16-bit configuration, the Weinberger adder uses parallel carry-save blocks to generate intermediate sum and carry vectors, which are then processed through a final stage to produce the correct output. This parallelism enables faster computation compared to traditional RCA designs. The primary advantage of the Weinberger architecture lies in its suitability for high-speed arithmetic operations, making it an effective solution in DSP and high-performance computing applications.

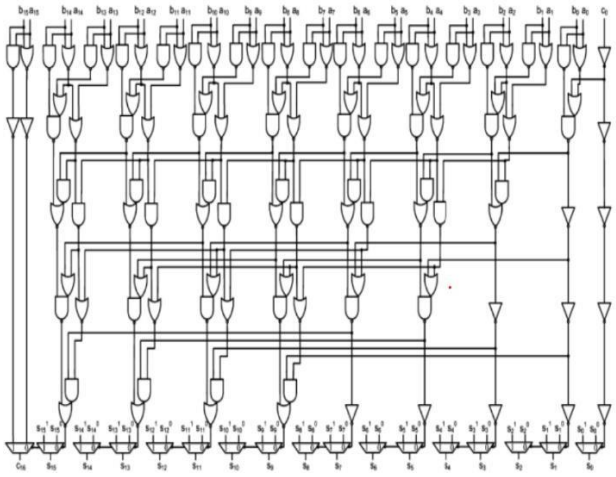

Fig.2 16 bit Weinberger Adder

Comparative analysis of the 16-bit RCA and Weinberger architectures highlights the trade-off between simplicity and performance. While RCA is optimal for low-power and area-sensitive designs, the Weinberger adder offers a compelling alternative when speed and throughput are the primary concerns. Implementing these architectures with modern CMOS technologies can further enhance their power-delay characteristics, thus providing optimized solutions tailored to specific application requirements.

IV. METHODOLOGY

A.16 Bit Ripple Carry Adder

The implementation of the 16-bit Ripple Carry Adder (RCA) follows a structured methodology aimed at optimizing the design for low power and area efficiency, while maintaining functional accuracy. The RCA is constructed using a series of 1-bit full adder (FA) blocks connected in cascade, where the carry output of each FA is propagated to the carry input of the subsequent stage. This sequential carry propagation defines the core operational characteristic of the RCA architecture.

Initially, a behavioral model of the 1-bit full adder is developed and validated using basic logic gates—AND, OR, and XOR. These gates are further synthesized into a transistor-level netlist suitable for simulation under a 180 nm CMOS technology node. Each full adder is then replicated 16 times to form the complete 16-bit RCA structure, with inter-stage connections established to enable carry forwarding from the least significant bit (LSB) to the most significant bit (MSB).

The design is simulated using industry-standard EDA tools such as Cadence Virtuoso. Functional verification is carried out to ensure the correctness of addition across all input combinations. Following verification, the design undergoes synthesis and optimization for timing, area, and power using

physical constraints like power supply voltage (1.8V) and operating frequency .

Simulations are performed to extract performance metrics including total delay, power consumption. These parameters are compared against alternative adder architectures to evaluate trade-offs in speed versus resource utilization. The ripple carry architecture, despite having a higher critical path delay due to linear carry propagation, demonstrates notable advantages in layout regularity, gate count, and static power consumption, making it suitable for low-power and resource-limited embedded systems.

B.16 Bit Weinberger Adder

The 16-bit Weinberger Adder is realized through a carry-select, modular architecture to reduce propagation delay by parallel computation. The adder is segmented into four identical 4-bit blocks, each comprising two 4-bit Ripple Carry Adders (RCAs). One RCA in each block is considered to have a carry-in of logic '0', and the other to have a carry-in of logic '1'. Both adders run in parallel to generate the sum and carry-out for their respective hypotheses. A 2:1 multiplexer is then employed to choose the proper output depending on the actual carry propagated from the preceding block.

In the schematic depicted in Fig.2, each block's dual RCA architecture and corresponding multiplexer logic are well represented. The vertical pattern represents the multiplexing cascading of carry signals, while horizontal connections address sum bit routing for each sub-block. A single fixed carry-in RCA is employed in the initial block to proceed with the calculation immediately. Remaining blocks use the carry-out from the previous stage as the selection input to each multiplexer. The final 16-bit output is constructed by concatenating the results of all chosen sum lines.

The design is modelled at the gate level with proven 1-bit full adders built from elementary logic gates (AND, OR, XOR). Each block is mapped to a 180-nm CMOS technology node and implemented in Cadence Virtuoso. Functional verification is achieved through exhaustive simulation over all input combinations for correctness. The circuit is then synthesized with typical constraints such as a power supply of 1.8V and a target operating frequency. Simulations are performed to extract important performance parameters like delay and power consumption.

V. SIMULATION AND IMPLEMENTATION

The operation of the 16-bit Weinberger Adder and the 16-bit Ripple Carry Adder (RCA) was tested by simulating them extensively using Cadence Virtuoso simulations. Both the circuits were simulated in the same environment to compare their operation, speed, and power dissipation.

A. Functional Verification

Both the 16-bit Weinberger Adder and the 16-bit RCA were verified through exhaustive simulations of all possible input combinations. Both adders were found to produce the correct sum and carry-out outputs in the results. Both adders were verified at the gate level, and 1-bit full adders made up of basic logic gates (AND, OR, XOR) were used for both designs. The implementations of the adders functioned correctly under all input values, as needed.

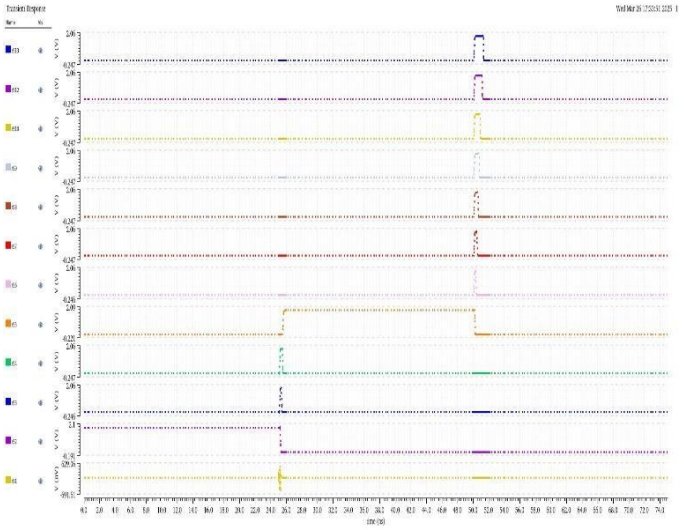

Fig.3 16 Bit Weinberger adder Output waveforms

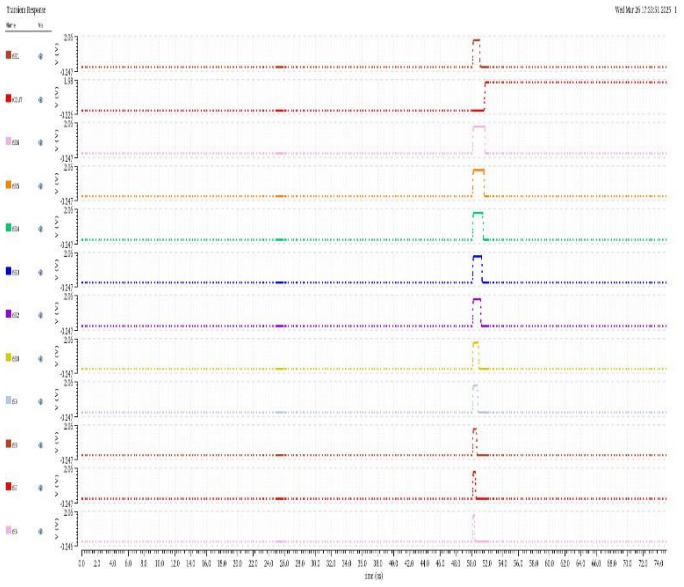

Fig.4 16 Bit Weinberger adder Output waveforms

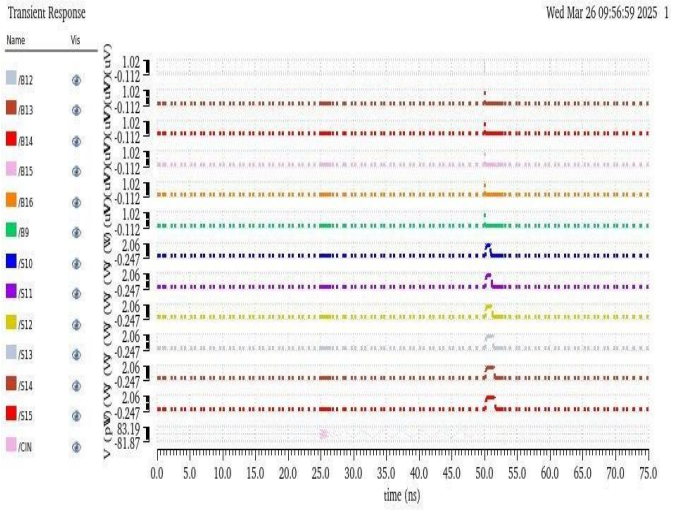

Fig 5. 16 Bit RCA simulation waveforms

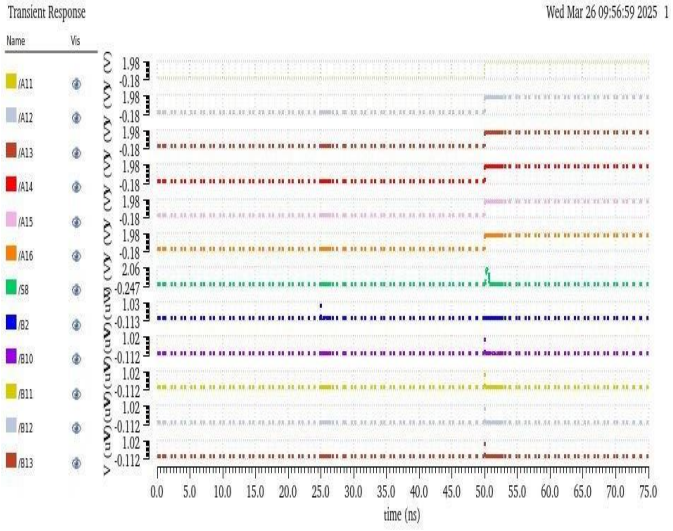

Fig 6. 16 Bit RCA simulation waveforms

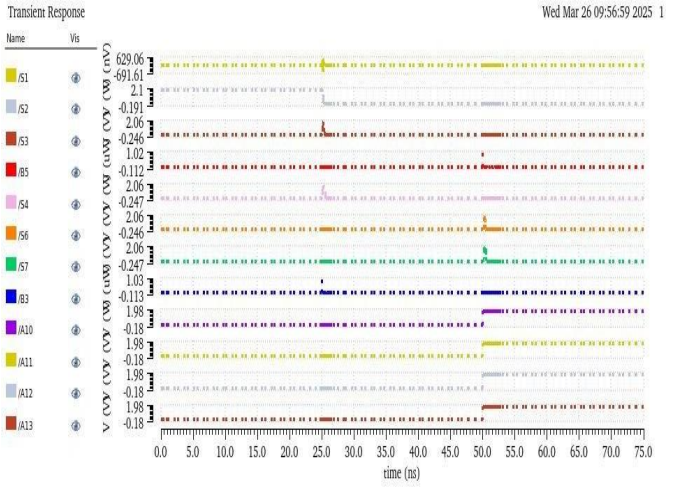

Fig 7. 16 Bit RCA simulation waveforms

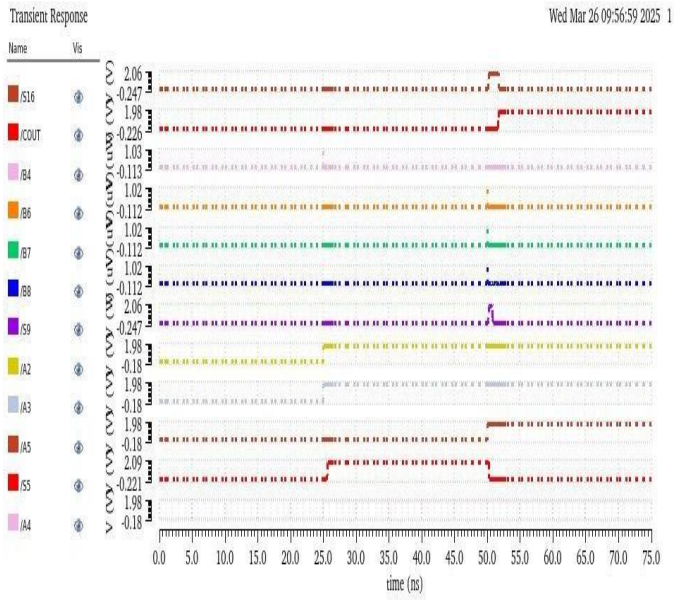

Fig 8. 16 Bit RCA simulation waveforms

VI. DESIGN COMPARISON

A. Speed

The 16-bit RCA has large propagation delay since each carry bit propagates sequentially, and the delay is linearly dependent on bit-width. This is not ideal for high-speed applications, particularly with large word-lengths. On the other hand, the 16-bit Weinberger Adder employs parallel carry-select computation, which has the effect of decreasing delay by a large amount by computing carries in parallel. This yields faster operation and more uniform performance across bit-widths and is thus better suited for high-speed applications.

B. Power Consumption:

The 16-bit RCA exhibits greater power consumption because of carry propagation sequentially, resulting in recurring switching per stage. The power consumption increases in proportion with an increase in the bit-width. The Weinberger Adder, being parallel, saves power consumption by avoiding redundant transitions and by optimizing carry calculation, and is thus more power-efficient, especially for higher bit-widths.

C. Scalability

The RCA does not work well for larger bit-widths since the carry propagation delay is directly proportional to the number of bits. For large word-lengths, this makes the performance poor. The Weinberger Adder, on the other hand, scales better because it has a parallel architecture. With increasing bit-width, the delay is relatively constant, and the design can still maintain performance even for wider adders.

VII. RESULTS AND ANALYSIS

Comparison of the 16-bit Ripple Carry Adder (RCA) and 16-bit Weinberger adder's performance was done by using Cadence Virtuoso in a shared simulation environment. Propagation delay and power dissipation were the most important performance parameters compared.

A. Delay Analysis

The delay measured for the RCA varied from 1.672 ns to 25.25 ns, depending on the input transitions and timing paths. The Weinberger adder, on the other hand, had much lower delay values ranging from 121.6 ps to 251.7 ps. This is due to the parallel carry-generation mechanism used in the Weinberger architecture, which minimizes critical path dependency and logic depth over the linear propagation of carry in the RCA.

B. Power Analysis

The overall average power dissipation for the RCA was found to be 13.35µW, but the Weinberger adder had only a power dissipation of 66.25µW. The reduced power dissipation of the Weinberger adder is attributed to its optimized carry calculation and minimized switching activity resulting from fewer transitions on the critical path.

Fig.9 Weinberger s4 Graph

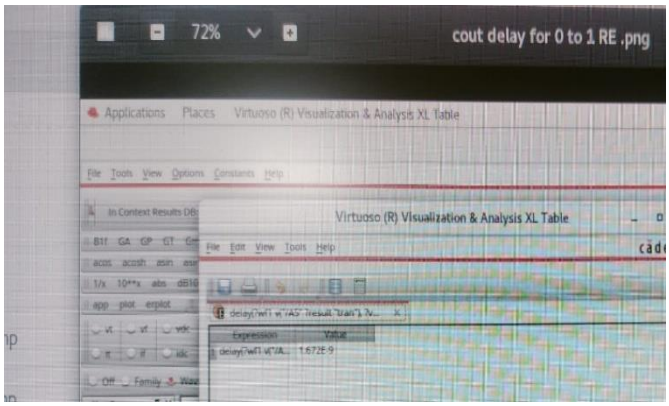

Fig.10 RCA Adder Delay Cout

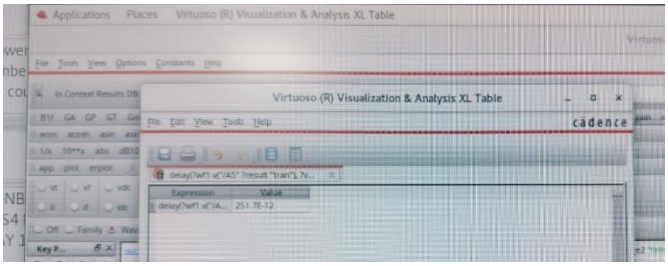

Fig.11 Weinberger adder Delay S5

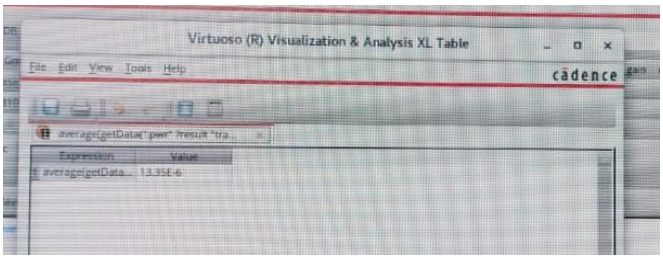

Fig.12 RCA Adder power

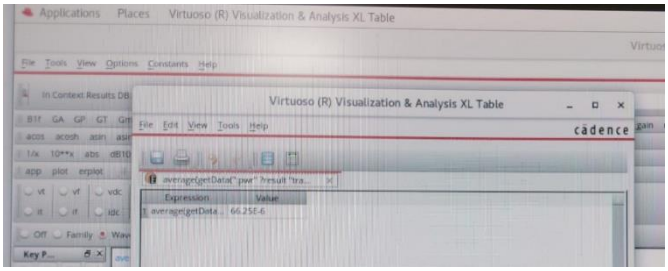

Fig.13 Weinberger Adder power

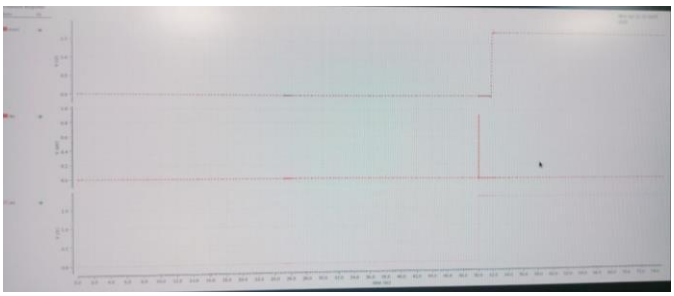

Fig.14 RCA Delay cout Graph

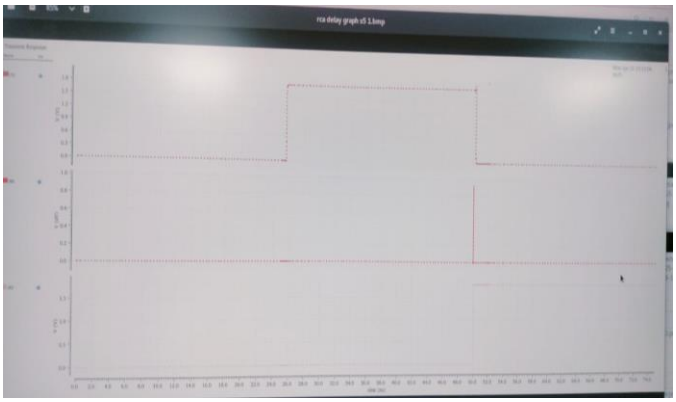

Fig.15 RCA Delay s5 Graph

Fig.16 Weinberger Delay Cout Graph

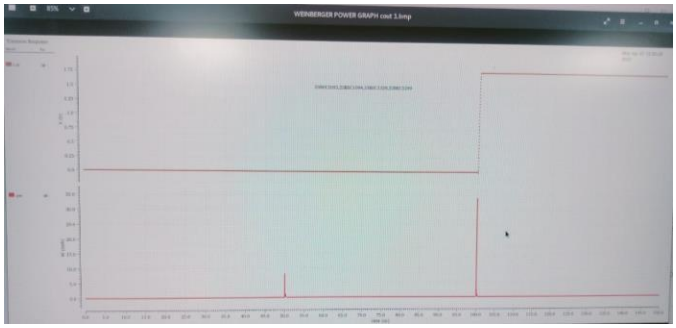

Fig.17 Weinberger Power Graph

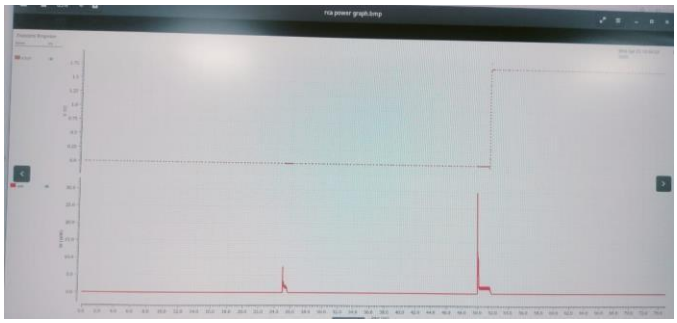

Fig.18 RCA Power Graph

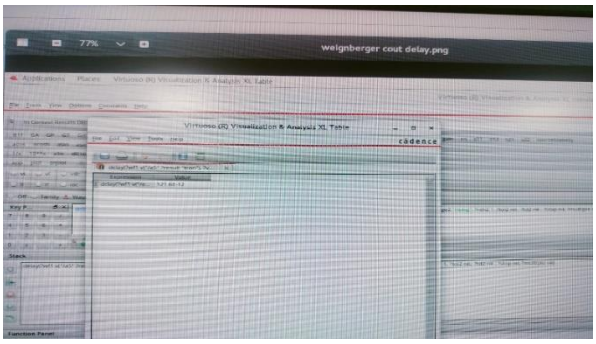

Fig.19 Weignberger delay Cout

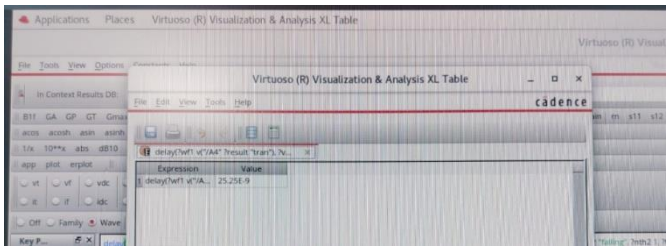

Fig.20 RCA Adder delay S5

The results show that the Weinberger adder is better than the conventional RCA both in delay and power terms. This renders the Weinberger adder a better option for low-power and high-speed VLSI applications. Its parallelism and fewer logic stages make its computational efficiency improved, which is vital in current digital arithmetic units.

VIII. CONCLUSION

In this work, a comparative analysis of a 16-bit Ripple Carry Adder (RCA) and a 16-bit Weinberger adder was performed by implementing their circuits in Cadence Virtuoso and comparing important performance metrics like delay and power dissipation. The simulation results conclusively show that the Weinberger adder considerably outperforms the conventional RCA in terms of speed and energy efficiency.The RCA, being less complex in structure, displays a great deal of propagation delay as a result of sequential carry propagation. Contrastingly, the Weinberger adder with parallel calculation of carries and highly optimized architecture provides quicker addition with less delay and power consumption. In particular, the Weinberger adder registered a delay of as low as 121.6 ps and power consumption of 13.35 µW, as compared to 1.672 ns and 66.25 µW for the RCA. These results demonstrate the applicability of the Weinberger adder to high- performance and low-power VLSI scenarios, where both speed and energy efficiency are critical. The project can be extended further by scaling the project to higher bit-widths and further exploring its performance over other process technologies or real-time workloads.

IX. ACKNOWLEDGMENT

The authors extend their heartfelt appreciation to the members and staff of the Department of Electronics and Communication Engineering, Vellore Institute Of Technology, for their active support and inspiration during the execution of this work. We gratefully thank our project supervisor, Dr.S.Umadevi, for her able guidance, advice, encouragement, and technical observations that made our work an overall success.

We also acknowledge the VLSI Laboratory team for offering the necessary tools and simulation platform in Cadence Virtuoso that proved to be very important during the design and analysis of the adder architectures. We are also thankful to our family members and peers for their constant support and motivation.

X. REFERENCES

N. T. Quach and M. J. Flynn, "High-Speed Addition in CMOS," *IEEE Transactions on Computers*, vol. 41, no. 12, pp. 1385–1396, Dec. 1992.

A. Weinberger and J. L. Smith, "A Logic for High-Speed Addition," *National Bureau of Standards*, Circulation 591, pp. 3–12, 1958.

K. Nehru and A. Shanmugam, "Design of High-Performance Low-Power Full Adder," *International Journal of Computer Applications in Technology*, vol. 49, no. 2, pp. 134–140, 2014.

P. G. Akila, "Design and Implementation of Ripple Carry Adder," *International Journal of Advanced and Innovative Research*, vol. 5, no. 6, pp. 64–68, June 2016.

M. S. Bhat and S. M. R. Islam, "Design and Implementation of Hybrid Cascaded Kogge-Stone Prefix-2 Adder Using Weinberger and Ling Recurrence Algorithm," *ARPN Journal of Engineering and Applied Sciences*, vol. 12, no. 11, pp. 6252–6257, Nov. 2017.

Y. Ma et al., "Cross-Layer Optimization for High-Speed Adders: A Pareto Driven Machine Learning Approach," *arXiv preprint arXiv:1807.07023*, 2018.

P. Balasubramanian et al., "Approximate Ripple Carry and Carry Lookahead Adders: A Comparative Analysis," *arXiv preprint arXiv:1710.05474*, 2017.

S. Naffziger, "A Sub-Nanosecond 0.5μm 64b Adder Design," in *Proceedings of the IEEE International Solid-State Circuits Conference*, San Francisco, CA, USA, Feb. 1996.

G. Dimitrakopoulos and D. Nikolos, "High-Speed Parallel-Prefix VLSI Ling Adders," *IEEE Transactions on Computers*, vol. 54, no. 2, pp. 225–231, Feb. 2005.

H. Ling, "High-Speed Binary Adder," *IBM Journal of Research and Development*, vol. 25, no. 2, pp. 156–166, Mar. 1981.

S. K. Mohapatra and A. K. Panda, "Design of Ripple Carry Adder Using GDI Logic," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 3, pp. 1234–1238, Feb. 2020.

S. K. Mohapatra and A. K. Panda, "Design of Novel High-Speed Parallel Prefix Adder," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 3, pp. 1239–1243, Feb. 2020.

XI. APPENDICES

A. APPENDIX I

Simulation Environment Details

- *Design Tool*: Cadence Virtuoso
- *Technology Node*: GPDK 180 nm
- *Simulation Library*: gpdk180
- *Supply Voltage (VDD)*: 1.8 V
- *Simulation Type*: Transient Analysis
- *Temperature*: 27°C (Room Temperature)
- *Input Vectors*: Applied manually through stimulus generation
- *Output Analysis*: Delay measured at 50% VDD threshold, power calculated using average consumption over the transient window.

B.APPENDIX II

Measured Results Summary:

Metric	**RCA Adder**	**Weinberger Adder**
Power consumption	13.35 μW	66.25μW
Delay(Min)	1.672 ns	121.6 ps
Delay(Max)	25.25 ns	251.7 ps