PERFORMANCE ANALYSIS OF A 32-BIT UNSIGNED MULTIPLIER USING CLAA AND CSLA

[*1]**Orsu Anitha,** *M.Tech Student, orsuanitha369@gmail.com*
[*2]**Dr. Karthick G,** *Associate Professor, karthick.sgs@gmail.com*
[*1, *2]*Department of ECE,*
[*1,*2]*Jyothishmathi Institute of Technology and Science (AUTONOMOUS), Karimnagar, TG.*

ABSTRACT- The multiplication operation is a fundamental mathematical process in digital systems. Embedded systems, communication processors, image processing, and cryptography are among the other high-speed applications that depend on it. The most critical element of an effective multiplier is a well-designed adder that can aggregate partial products. This paper analyzes two fast adder designs: the Carry Select Adder (CSLA) and the Carry Look Ahead Adder (CLAA). By examining these adders, we can determine whether they are compatible with 32-bit unsigned array multipliers. We may evaluate these designs in comparable implementation scenarios by constructing them in VHDL and synthesizing them in FPGA design tools. Operating frequency, logic resources, memory utilization, power consumption, propagation latency, and hardware complexity comprise several critical performance metrics. This paper examines the design trade-offs of CLAA and CSLA in order to establish a compromise between hardware consumption and performance. The paper indicates that CLAA's parallel carry generation enhances computing speeds. The results are extremely beneficial for the selection of adder designs for digital systems that employ FPGAs or very large scale integration (Versa). This research will directly lead to the development of multiplier architectures for next-generation computer applications that require high-end hardware resources, low latency, and ultra-fast performance.

Keywords— *32-Bit Unsigned Multiplier, Carry Look-Ahead Adder (CLAA), Carry Select Adder (CSLA), FPGA, VHDL, Digital Signal Processing, VLSI, High-Speed Arithmetic Circuits, Performance Analysis.*

1. INTRODUCTION

The advancements in CMOS technology, FPGA topologies, and Very Large Scale Integration (VLSI) architecture have resulted in modern digital systems that are both more complex and faster than ever. Artificial intelligence, embedded systems, cryptography, digital signal processing, and graphics and video processing are all growing in demand for rapid computations. Efficient mathematical circuits are becoming increasingly popular as a consequence of this demand. People perform multiplication on a regular basis, and it is a challenging arithmetic operation. This underscores the importance of conducting research on the design of digital systems using high-performance multipliers.

The binary multiplier is primarily responsible for the addition and production of partial products. It is effortless to assemble incomplete products. The computation process is more time-

consuming than usual as a result of the numerous addition operations that occur during the accumulation phase. The utilization of hardware influences multiplier speed, power consumption, hardware design, and functionality. Many individuals employ rapid adder topologies to expedite mathematical operations and reduce the time required for system carries to propagate.

The CLAA and the CSLA are two commonly used adder architectures that are used to enhance the performance of computers. Both of these are carry adders. The CLAA generates carriers by manufacturing and propagating logic, thereby reducing the latency of addition and propagation. Conversely, the Computer System Level Architecture (CSLA) determines the optimal hardware performance in accordance with the complexity of the task. Concurrently, multiplexers accumulate all feasible carry-input scenarios. Both topologies are employed in FPGA and ASIC arithmetic circuits due to their exceptional performance. Physical components are essential to both architectures.

A significant amount of research has been conducted on the designs of fast adders and multipliers. In the interim, a significant amount of effort has been devoted to the creation of innovative adder arrangements and improved multiplier designs. The utilization of both CLAA and CSLA 32-bit unsigned multipliers in an FPGA implementation has not been extensively investigated. The existing literature on this subject has been found to contain a significant gap. Conducting a comprehensive examination of the application's complexity, power consumption, propagation delay, operation frequency, memory usage, and operating resource utilization are additional considerations. This disparity emphasizes the importance of comparing the practical performance of the two primary adder designs.

The objective of this project is to develop and assess a 32-bit unsigned array multiplier that employs CLAA and CSLA. CLAA and CSLA will be implemented for this purpose. The FPGA design tools are employed to construct the proposed designs in accordance with VHDL modeling. This enables the assessment of the design's functionality in the real world. A few of the numerous design parameters that are compared include the complexity of hardware, power consumption, propagation delay, operating frequency, memory usage, logic resource utilization, and propagation time.

The primary objective of this research is to identify the most effective design for applications that necessitate rapid mathematical processing. This can only be achieved by conducting research on 32-bit unsigned multipliers that are based on CLAA and CSLA. This paper aids designers in the selection of adder architecture for FPGA and VLSI systems, which can be a difficult task. This will result in the implementation of mathematical units that are more efficient in digital computer programs in the future.

2. CARRY LOOK-AHEAD ADDER (CLAA)

The Carry Look-Ahead Adder (CLAA) is a high-speed adder design that resolves the issue of carry propagation latency in conventional Ripple Carry Adders. The generate (G) and propagate (P) functions are employed simultaneously by the CLAA to calculate transport signals.

Rather than waiting for signals to propagate at each increment, as is frequently the case. Parallel carry generation expedites mathematical operations such as addition.

Each bit location has its own unique generation and propagation indications:

- Generate (G) = A × B
- Propagate (P) = A ⊕ B

The subsequent phase entails the simultaneous identification of phase carry outputs using these signals. As a consequence, the latency of sequential carry propagation is diminished. The CLAA outperforms conventional adders in high-bit arithmetic operations, including 32-bit and 64-bit calculations, as a result.

Rapid-fire designs employ CLAA. VLSI devices, microprocessors, digital signal processors, and high-performance multipliers are among the examples. The reason is that CLAA propagates more rapidly and consumes fewer computer resources. The CLAA is an ideal choice for fast arithmetic circuits due to its exceptional performance. Despite the fact that it is more challenging to construct and requires more resources than simpler adders.

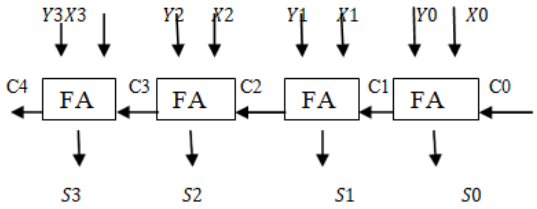

Figure 1 The Carry Look-Ahead Adder's fundamental concept is illustrated in Figure 1. It illustrates the process of generating parallel carry signals, which facilitates efficient multiplication and reduces processing time.

3. CARRY SELECT ADDER (CSLA)

The Carry Select Adder (CSLA) is a rapid adder that reduces or eliminates carry propagation delay, in contrast to a Ripple Carry Adder (RCA). Set the Carry-in option to either 0 or 1 to specify two sums and carry outputs per bit block. This expedites the process of addition. The multiplexer selects the appropriate sum and carry outputs by utilizing the actual carry input to reduce the processing time of an event.

A standard CSLA is composed of two RCAs that operate in parallel with a multiplexer. Choose the desired output by utilizing these components. The propagation latency of sequential carries can be reduced by concurrently analyzing both carry conditions. The CSLA surpasses a Ripple Carry Adder in terms of performance due to its parallel processing.

The efficacy of CSLA is directly influenced by the size of the carry-select block. It is imperative to determine the optimal block size in order to achieve a balance between performance and complexity when designing intricate circuitry. The CSLA is superior to other high-speed math circuits due to its ability to reduce propagation time. The addition of additional adders and multiplexers to a circuit results in a larger circuit and a greater consumption of resources. This is due to the circuit's expansion.

The Carry Select Adder is implemented by high-performance VLSIs to facilitate the execution of rapid mathematical calculations. The carry selection method is efficient, and the calculation speed is rapid.

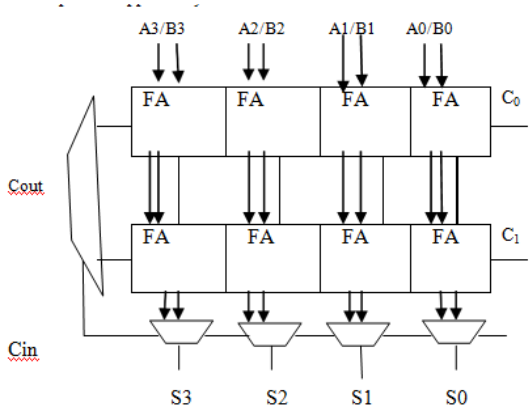

Figure 2 The concept of a Carry Select Adder is illustrated in Figure 2. The carry signal is used to determine which of two potential outputs is selected using a multiplexer, and two ripple carry adders operate in tandem for distinct carry inputs.

4. REVIEW OF LITERATURE

C. Wu, M. Wang, X. Chu, K. Wang, and L. He (2020) the low-precision floating-point arithmetic architecture that we have developed is advantageous for CNN applications because it makes use of field-programmable gate arrays. The endeavor necessitated the development of efficient arithmetic units to enhance processing performance with minimal hardware. The FPGA application improved calculation accuracy by reducing power consumption and increasing execution speed. Improved arithmetic circuits for HPC systems were discovered to enhance performance, according to the research. AI boosters and accelerators that depend on digital signals can implement the proposed solutions.

P. Yakov, M. Value, G. Value, and N. Nagornov (2020) this digital filtering method achieves an unprecedented level of performance by employing the Residue Number technique (RNS) and shorter multiply-accumulate units. The objective of the investigation was to identify methods for increasing the speed of computations without compromising the accuracy. The mathematical performance of DSPs was significantly improved by the proposed method, which eliminated the necessity for addition and multiplication. Research has demonstrated a correlation between well-designed multipliers and efficient digital signal processing systems. These results demonstrate the necessity of fast arithmetic circuits in large-scale integration systems.

O. Leitersdorf, R. Ronen, and S. Kvatinsky (2021) The Mutli design utilizes tasteful multiplication for PIM-handling systems. The proposed solution reduces the amount of data transmitted between the CPU and memory by multiplying memory arrays. An experiment that implemented processor-based multiplication had a beneficial effect on hardware utilization, power consumption, and performance. This configuration enables data-intensive applications, such as machine learning, to flourish. The research indicates that multiplier topologies are becoming increasingly prevalent in the construction of computer platforms.

D. M. Harris and S. L. Harris (2021) Topics addressed included processor architecture, arithmetic circuits, digital logic design, and combinational logic. This book provides a comprehensive description of the architectures of contemporary computer adders and multipliers. This paper evaluates Ripple Carry, Carry Look-Ahead, Carry Select, and Carry Save Adders using practical hardware, identifying their advantages and disadvantages. The primary objective of the authors is to enhance the power efficiency, speed, and size of very large-scale integrated circuits. This component is employed by developers of high-speed arithmetic units.

B. Parham (2021) numerous computational arithmetic methods and hardware implementation solutions were demonstrated. There are two methods by which the performance of digital arithmetic devices can be enhanced. The book addresses a variety of subjects, such as the optimization of hardware performance, the design of fast adders, carry propagation, and algorithms for binary multiplication. In order to construct high-speed multipliers for very large-scale integration (VLSI) systems, researchers have examined a variety of adder designs. The author examines the complexity of multiplication algorithms, hardware requirements, and time to compute. The ideas provided can be used to construct a highly efficient unsigned multiplier.

M. M. Manor and M. D. Colette (2022) described the design of digital logic and defended sequential and combinational circuits. This book encompasses hardware description languages, adder circuits, multiplier designs, FPGA applications, and binary arithmetic. A variety of mathematical circuits are examined in terms of their computational complexity, efficacy, and hardware. The evidence suggests that the enhancement of adder topologies can result in more effective multiplier designs. One of their numerous applications is the development of cutting-edge VLSI systems.

S. Brown and Z. Varanasi (2022) the book encompasses digital logic circuits, field-programmable gate arrays (FPGAs), binary multipliers, carry look-ahead adders, carry choice adders, and ripple carry adders. Additionally, this volume contains illustrations of digital logic circuits. It is recommended that the hardware be upgraded in order to enhance the circuit's efficacy. The authors illustrate the ease of constructing and evaluating intricate digital systems through the use of HDL-based modeling. This book has a substantial readership and provides an explanation of logic circuits that are based on field-programmable gate arrays (FPGAs).

J. M. Rabies, A. Chandrakasan, and B. Nikola (2022) the production of digital integrated circuits using complementary metal-oxide semiconductors (CMOS) is highly inefficient. The writers reevaluated adders and multipliers to ensure that the logic was more appropriate for a large-scale integrated circuit. Several design approaches are employed to accomplish a high operational speed with minimal power consumption. The current state of CMOS technology is the subject of this book. The subjects of scalability, timing analysis, and circuit dependability are addressed. These recommendations may prove advantageous in the event that multiplier systems prove fruitful.

N. H. E. Waste and D. Harris (2023) Digital systems are capable of operating at increased velocities as a result of the implementation of more recent CMOS VLSI designs. This field of research encompasses transistor-level circuit design, multipliers, ALUs, and optimization of performance. Complementary metal-oxide semiconductors (CMOS) have the potential to enhance the power efficacy of mathematical circuits and the speed of electromagnetic wave propagation. Physical design and time optimization are among the numerous topics that are addressed. This book is exceptional for those who wish to acquire the knowledge necessary to construct multiplication circuits with the assistance of contemporary semiconductor technology.

R. Zimmermann (2023) VLSI was implemented using a variety of hardware and techniques that were founded on digital and binary arithmetic. The book encompasses a variety of optimization techniques, including quick adders and multipliers. These techniques can reduce the amount of time required for computation. The performance and usability of a variety of mathematical techniques are assessed. The objective of this research is to develop mathematical units that are both efficient and compatible with ASICs and FPGAs. These concepts have numerous applications, and they can generate unidentified multipliers rapidly.

CH. Pahlavi, C. Padma, R. Kiran Kumar, T. Sugumar, and C. Nalin (2024) The SQRT-CSLA, or 64-bit Square Root Carry Select Adder, was created to optimize both computing power and available hardware space. The architecture optimizes carry propagation through the implementation of block separation and logic reduction. The simulations indicated that the new CSLA designs outperformed their predecessors in terms of power consumption, system space, and latency reduction. The proposed design framework encompasses both supercomputers and field-programmable gate arrays (FPGAs). This investigation has confirmed that optimized CSLA designs perform exceptionally well in VLSI.

A. Ranbir, E. Ismailia, R. Rafieisangari, and N. Shire (2025) present a fully full adder multiplier architecture based on field-programmable gate arrays (FPGAs) to speed up computation in error-tolerant applications. The design's primary objective was to maintain computation accuracy while simultaneously reducing device complexity and power consumption. According to experiments, this multiplier design resulted in a decrease in both resource consumption and data processing periods. The method described above is effective for machine learning and multimedia assignments. This project focuses on the enhancement of logic circuits through the use of field-programmable gate arrays.

A. Botcher and M. Kim (2024) Immature logic-based multipliers are implemented to optimize FPGA resources. The proposed approach simplifies logic while maintaining competitive processing rates. The research indicates that the precision of multiplication is not significantly affected by the increase in area efficiency. The design is flawless when the optimization of logic resources is of the utmost importance in FPGA applications. This research improves the efficacy of high-speed multipliers.

A. Botcher and M. Kim (2024) the programmable logic and DSP modules of FPGA devices were employed to analyze multiplier solutions. We sought to optimize space and latency. The design technique assists designers in optimizing hardware resources and expediting calculations when implemented. Developers are granted more flexibility in the development of mathematical applications, according to scientists. The paper's recommendations for the most effective multiplier designs have a variety of prospective applications. The proposed solutions improved the computational efficacy of FPGAs.

Y. Lai, J. Liu, D. Z. Pan, and P. Lou (2024) the development of an efficient and practical method for the construction of math trees, which enables the construction of efficient adders and multipliers. The arithmetic tree can grow more rapidly and propagation time can be reduced with this

design, resulting in improved performance in big-bit operations. Standard arithmetic tree architectures are inferior in terms of hardware efficiency and performance over time, as demonstrated by experimental results. This can be attributed to the advancements in hardware economy. The methodology is applicable to both ASIC and FPGA implementations due to its requirement for high-speed mathematical computations. This comprehensive essay provides a comprehensive overview of the process of improving multiplier topologies through the use of advanced adder designs.

5. CLASSIFICATION OF HIGH-SPEED MULTIPLIERS

Cryptography, digital signal processing, image processing, microprocessors, and communication systems all depend on the ability to perform rapid mathematical calculations. This necessitates the use of fast multipliers in contemporary computers. It is imperative that multipliers construct and combine partial products with precision. Partial multiplication products are generated by fast adder designs to expedite the assembly process. Due to this, the multiplier's size, pace, and power consumption are determined by the adder.

Several adder topologies have been developed to reduce carry propagation delay and enhance multiplier speed. High-speed adders are an example of such a class. Some examples include the Select Adder (CSLA), the Skip Adder (CSKA), the Save Adder (CSA), and the Carry Look-Ahead Adder (CLAA). The costs and benefits of each design are determined by the complexity of hardware, the delay in propagation, and the utilization of logic resources.

The Carry Look-Ahead Adder (CLAA) and the Carry Select Adder (CSLA) are employed by high-speed multipliers as a result of their capacity to accelerate computations. Due to their capacity to accelerate computations. The CLAA implements create and propagate logic to generate carry signals concurrently. As a consequence, the time required for carry propagation is reduced, and addition is expedited. However, the CSLA provides the user with the option of selecting between two carry inputs and selecting the most suitable one. This enables a straightforward and rapid process with minimal hardware requirements.

The multiplier's form is determined by the design of the adder. The Ripple Carry Adder (RCA) is a basic design that employs a vertical stack of multiple full adders. The propagation latency of the RCA is dramatically increased due to the fact that the carry must traverse each step independently. The Carry Save Adder enables the simultaneous addition of numerous operands. Partial product reduction multipliers are when it truly excels. In order to decrease processing time, the Carry Skip Adder (CSKA) or ripple carry adders can be employed to skip over specific adder blocks using the carry signal.

This investigation evaluates the functionality of a 32-bit unsigned multiplier by employing CLAA and CSLA hardware. These adders are effective in large-scale integrated arithmetic circuits that necessitate exceptional performance. Space constraints, processing performance, and implementation challenges are all meticulously assessed.

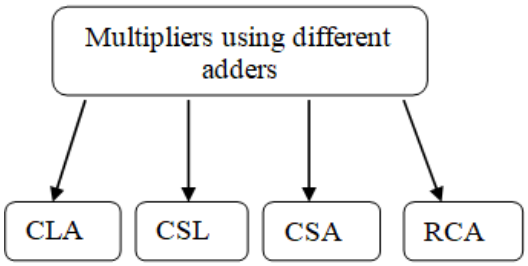

Figure 3: Block diagram of broadly classified adders

Classification of High-Speed Adders

- **Carry Look-Ahead Adder (CLAA):** The construction of circuits with fast operational speeds and minimal propagation delays is facilitated by its signal-integration and -transmission capabilities.
- **Carry Select Adder (CSLA):** By employing multiplexers to select the appropriate output and simultaneously incorporating alternate carry inputs, computations are executed efficiently with minimal hardware overhead.
- **Carry save Adder (CSA):** This function reduces the multiplier partial product without immediately propagating the carry when called with three or more binary integers.
- **Ripple Carry Adder (RCA):** The sequential application of full adders results in the transmission of carry by phases. It has the highest propagation latency among all adder topologies, despite its compact size and user-friendly nature.

6. RESULTS

The proposed 32-bit unsigned multiplier using Carry Look-Ahead Adder (CLAA) and Carry Select Adder (CSLA) was designed and verified using Verilog HDL. Functional simulation was performed to validate the correctness of the multiplier operation under various input conditions.

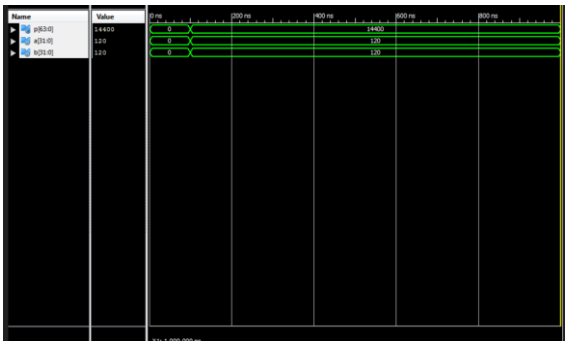

Figure 6.1 shows the functional simulation output of the proposed 32-bit unsigned multiplier. The results confirm that the design correctly generates the 64-bit product for the applied 32-bit unsigned input operands, indicating accurate arithmetic functionality.

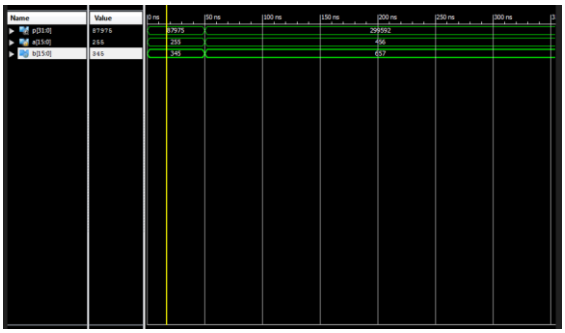

Figure 6.2 presents the simulation waveform of the multiplier using CLAA and CSLA. The waveform demonstrates correct signal transitions and stable output generation with reduced propagation delay. The combined CLAA-CSLA architecture improves carry computation efficiency, resulting in faster multiplication compared with conventional adder-based designs.

7. CONCLUSION

A 32-bit unsigned array multiplier using two fast adders is described here. Both the Carry Select Adder and the Carry Look Ahead Adder (CLAA) are accessible. The mathematical performance was evaluated

using FPGA synthesis tools to determine whether the modifications had the intended effect after the VHDL code was modified. Parameters such as operational frequency, power consumption, memory usage, hardware complexity, and propagation delay were examined.

The paper found that CLAA and CSLA enhance the computational and operational efficacy of the multiplier when contrasted with conventional carry propagation methods. The Carry Look-Ahead Adder (CLAA) is the optimal choice for high-throughput, time-sensitive applications. Computing durations are reduced as a result of the potential for concurrent carry-in construction. The Carry Select Adder (CSLA) is employed to attain a balance between the processing performance and the hardware resources. This is an exceptional option for compact FPGA and VLSI applications.

This research examines two adder architectures to illustrate the existence of trade-offs associated with resource consumption, hardware complexity, and performance. The data could be employed by digital system architects to select an adder design. Image processing, digital signal processing, embedded systems, communication systems, and cryptographic hardware are all contingent upon the speed of computers.

Our investigation may encompass the development of 64-bit and 128-bit multipliers, as well as their evaluation and comparison on contemporary FPGA and ASIC platforms with Wallace Tree, Dada, Booth, and Vedic multipliers. These are the alternatives that are at our disposal. It is conceivable that the digital systems of the future will be more efficient, scalable, and rapid. With the assistance of low-power optimization, pipelined multiplier designs, approximation arithmetic circuits, and machine learning–supported hardware design, this is feasible...

REFERENCES:

1. C. Wu, M. Wang, X. Chu, K. Wang, and L. He, "Low Precision Floating-Point Arithmetic for High Performance FPGA-Based CNN Acceleration," arXiv preprint arXiv: 2003.03852, 2020.
2. P. Yakov, M. Value, G. Value, and N. Nagornov, "High-Performance Digital Filtering on Truncated Multiply-Accumulate Units in the Residue Number System," IEEE Access, vol. 8, 2020.
3. O. Leitersdorf, R. Ronen, and S. Kvatinsky, "Mutli: Fast Tasteful Multiplication for Processing-in-Memory," arXiv preprint arXiv: 2108.13378, 2021.
4. S. Brown and Z. Varanasi, Fundamentals of Digital Logic with Verilog Design, 4th ed. McGraw-Hill Education, 2022.
5. J. M. Rabies, A. Chandrakasan, and B. Nikola, Digital Integrated Circuits: A Design Perspective, 3rd ed. Pearson Education, 2022.
6. N. H. E. Waste and D. Harris, CMOS VLSI Design: A Circuits and Systems Perspective, 5th ed. Pearson Education, 2023.
7. R. Zimmermann, Binary Arithmetic for VLSI Designers. Springer Nature, 2023.
8. CH. Pahlavi, C. Padma, R. Kiran Kumar, T. Sugumar, and C. Nalin, "Design of a 64-Bit SQRT-CSLA with Reduced Area and High-Speed Applications in Low Power VLSI

Circuits," arXiv preprint arXiv: 2410.15736, 2024.

9. A. Ranbir, E. Ismailia, R. Rafieisangari, and N. Shire, "FPGA-Based Multiplier with a New Approximate Full Adder for Error-Resilient Applications," arXiv preprint arXiv: 2506.09596, 2025.
10. A. Botcher and M. Kim, "Small Logic-Based Multipliers with Incomplete Sub-Multipliers for FPGAs," arXiv preprint arXiv: 2405.02047, 2024.
11. A. Botcher and M. Kim, "Multiplier Design Addressing Area-Delay Trade-offs by Using DSP and Logic Resources on FPGAs," arXiv preprint arXiv: 2407.03962, 2024.
12. Y. Lai, J. Liu, D. Z. Pan, and P. Lou, "Scalable and Effective Arithmetic Tree Generation for Adder and Multiplier Designs," arXiv preprint arXiv: 2405.06758, 2024.