Implementation of Systolic Multiplier Using Hybrid Multiplexer Dependent Adder

P. Bhargavi[1], Nandanavanam Mounika[1], Magam Sarika[1], Leburu Sindhuja[1], Kommu Sravani[1]

[1]Department of Electronics and Communication Engineering, Geethanjali Institute of Science and Technology, Nellore.

Abstract: Multipliers are basic building blocks in various integrated circuits like microprocessors, micro controllers, and ALUs. The existing multipliers suffer from high power consumption and inefficient use of hardware resources. They often rely on traditional adder structures that are not tailored for specific operations, leading to suboptimal performance. Additionally, their fixed architectures limit adaptability and scalability in different applications. So, the proposed approach offers enhanced computational efficiency and reduced power consumption compared to conventional multiplier designs. By integrating multiplexer-dependent adders into the systolic array, the proposed method optimizes resource utilization and delivers improved performance for various arithmetic operations. This integration allows for dynamic selection of adder types based on the specific multiplication operation, significantly reducing power consumption and latency. By adapting the hardware resources to computational needs, the method achieves higher efficiency and flexibility, making it suitable for a wide range of applications in digital signal processing and data processing systems.

Keywords: Systolic Multiplier, Razor Flip Flop, Multiplexer, Adders, Very Large-Scale Integration.

1.Introduction

In the ever-evolving landscape of VLSI design, the pursuit of innovative architectures for efficient multiplication has become increasingly pivotal. Systolic multipliers, characterized by their parallel computing capabilities, offer a promising avenue to meet the escalating demands for high-throughput arithmetic operations. This paper introduces a novel approach to systolic multiplier design, incorporating a hybrid multiplexer-dependent adder. The motivation behind this hybridization lies in addressing the challenges posed by traditional multiplication methods, aiming to strike a delicate balance between computational speed, area utilization, and power efficiency. Conventional multiplier architectures grapple with the challenge of reconciling the need for faster computation with the constraints imposed by limited silicon real estate and power budgets. Systolic arrays, which enable parallelism in data processing, present an opportunity to transcend these limitations. The motivation for this research stems from a desire to optimize systolic multipliers by introducing a hybrid multiplexer-dependent adder. This innovative design not only capitalizes on the inherent advantages of parallelism in systolic arrays but also seeks to streamline the critical path through the integration of a flexible and efficient adder structure.

This study is guided by two primary objectives. First, to devise a systolic multiplier that effectively exploits the parallel processing paradigm to achieve rapid multiplication. Second, to seamlessly integrate a hybrid multiplexer-dependent adder into the systolic array, thereby optimizing the critical path and minimizing overall latency. The hybrid adder, enriched with multiplexer-based components, introduces a versatile and efficient mechanism for addition within the systolic multiplier. By pursuing these objectives, this research aspires to contribute significantly to the ongoing advancements in VLSI design, pushing the boundaries of computational efficiency in digital signal processing and other relevant applications.

2. Literature Survey

Sadhineni, Harika, S. Josaph,et.al [1] proposed this paper, we designed the CMOS based multiplier using LFSR for high-speed operation. Basically, the memory cells, multidimensional memory circuits and memory frameworks were referenced with various sub-chips. In [2] authors discussed main objective of this research article was to designed high efficient accurate DL-PO logic multiplier for low power applications. primarily, the execution of DL-PO Multiplier was based on VLSI chips, which were used as critical element. In the multiplier, product information was utilised to create the propagator and generator signals in the same way. Kim, Sunwoong, Cameron J. Norris,et.al [3] discussed IEEE 754 standard for floating-point (FP) arithmetic was widely used for real numbers. Recently, a variant called posit was proposed to improve the precision around 1 and -1 . Since FP multiplication requires high computational complexity, various algorithmic approaches and hardware accelerator solutions were explored.

inaeifar, Atefeh, Ebrahim Abiri,et.al [4] observed multipliers were most frequently employed components in a system, responsible for performing computations, while significantly contributing to power consumption. In that article, a new architecture was presented by removing the least significant bits to implement three multipliers (Mul-1, Mul-2, and Mul-3) with the aim of reducing complexity and power consumption. In [5], authors proposed In-memory computing represents an efficient paradigm for high-performance computing using crossbar arrays of emerging nonvolatile devices. While various techniques have been emerged to implement Boolean logic in memory, the latency of arithmetic circuits, particularly multipliers, significantly increases with bit-width. In this work, we introduce an in-memory Wallace tree multiplier based on majority gates within voltage-gated spin-orbit torque (SOT) magnetoresistive random access memory (MRAM) crossbar arrays. Parmar, Rushik, Khushil Yadav, Gauraangi Anand,et.al [6] introduced myocardial infarction (MI) was a cardiac abnormality in which the coronary artery gets blocked, causing millions of fatalities every year. MI has a very high mortality and disability rate; therefore, with the

detection of MI, it was also imperative to determine the location of the blockage to provide on–time treatment to avoid any fatality.

Beura, Srikant Kumar, Sudeshna Manjari Mahanta, et.al [7] proposed inexact computing, was a modern approach for the development of low power and high-performance digital circuits, which involves approximation stages to get the utmost accurate result and are very much applicable in some error-tolerant applications like image processing. Vakili, Bahareh, Omid Akbari,et.al [8] introduced Approximate computing was one of the promising techniques in error-resilient applications to overcome high-density integration challenges, such as energy consumption and performance. Multipliers constitute a significant portion of computer arithmetic units, leading to considerable energy and time consumption. Haq, Shams Ul, Erfan Abbasian, et.al[9] developed appeal of portable electronics, embedded systems, and other smart devices steadily growing over time. The multi-valued logic (MVL) was primarily introduced to handle the interconnect issues in binary logic. Rohani, Zahra,et.al [10] illustrated the multiplier circuit was considered a significant component of larger circuits, such as the arithmetic and logic unit (ALU), and it was crucial to enhance its energy efficiency. This objective can be easily achieved by utilizing graphene nanoribbon field-effect transistor (GNRFET) devices and adopting ternary logic.

Kuo, Chao-Tsung, et.al [11] developed multi-modulus architecture based on the radix-8 Booth encoding of a modulo $(2n - 1)$ multiplier, a modulo $(2n)$ multiplier, and a modulo $(2n + 1)$ multiplier was proposed in this paper. It uses the original single circuit and shared many common circuit characteristics with a small extra circuit to carry out multi-modulus operations. cc Hao Li,,et.al [12] proposed Modular multiplication plays a crucial role in modern cryptography. Montgomery modular multiplication(MMM), one of the most classic and practical modular multiplication algorithms, has been widely used in cryptographic algorithms such as RSA, Diffie–Hellman algorithm, and Elliptic Curve Cryptography.

Balaji, M., et.al [13] presented in today's digital age, speed, and area are the primary design concerns. Increasing the rate at which multiplication and addition are performed had always been a need for developing cutting-edge technologies. Wallace multipliers, and Dadda multipliers, were among the fastest multipliers used in many processors to accomplish fast arithmetic operations. Malathi, L., et.al [14] illustrated the Low-pass RNS FIR filter was designed for a cut-off frequency of 50 Hz and filter coefficients were generated in MATLAB and implemented to denoise the ECG signal.To overcome this situation, reversible logic has been introduced with reversible gates. Thirumoorthi, Madhan, .,et.al [15] introduced super-resolution (SR) method has been widely used to improve the resolution and quality of natural images, and it was being employed in medical imaging. Recently, many techniques were introduced by researchers to improve the resolution and quality of the image. However, these approaches generate blurring and jagged edges in images.

3.Proposed Methodology

Figure 1 shows the proposed systolic multiplier. A systolic multiplier is a specialized hardware component designed to efficiently perform multiplication operations, particularly in digital signal processing (DSP) applications. It is named after the systolic array, a parallel computing architecture inspired by the human circulatory system's efficient flow of blood. In a systolic multiplier, multiple processing elements (PEs) are arranged in a regular grid-like structure, with each PE responsible for performing a portion of the multiplication operation. These PEs are interconnected in a pipelined fashion, allowing data to flow through the array in a synchronized manner, similar to the rhythmic pumping of blood in the circulatory system.

The operation of a systolic multiplier typically involves a series of multiplication and accumulation steps, commonly referred to as multiply-and-accumulate (MAC) operations. During each clock cycle, input data is fed into the systolic array, where it propagates through the array in a predetermined direction. As the data travels through the array, each PE multiplies its input data with a corresponding coefficient or operand, and the results are accumulated as they propagate through the array. The key advantage of systolic multipliers lies in their ability to exploit parallelism and pipelining to achieve high throughput and efficiency. By breaking down the multiplication operation into smaller, independent tasks and executing them concurrently across multiple PEs, systolic multipliers can significantly accelerate the overall computation speed compared to conventional serial multiplication algorithms.

Systolic multipliers find widespread use in DSP applications such as digital filters, Fourier transforms, and convolutional neural networks (CNNs), where efficient multiplication is a critical performance bottleneck. They are also employed in various other domains, including scientific computing, cryptography, and telecommunications, wherever high-speed and high-throughput multiplication operations are required.

Processing Elements (PEs): PEs are the fundamental computing units within the systolic array. Each PE performs basic arithmetic operations, typically involving multiplication and addition. PEs are arranged in a regular grid or array configuration.

Registers: Registers are used for storing intermediate results and operands. Each PE has its local registers, providing a temporary storage space for data during the computation process. Registers enable the pipelined flow of data through the systolic array.

Systolic Array Operation: The systolic array is initialized with input operands. Data flows through the array in a coordinated, systolic manner. Each PE performs a specific computation on the data it receives and passes the result to its neighbouring PEs. This pipelined and parallel approach allows for efficient multiplication.

Razor Flip-Flops: Razor flip-flops are a type of flip-flop circuit designed to operate at lower supply voltages, enhancing energy efficiency. In the context of a systolic multiplier, razor flip-flops may be employed as storage elements within each PE. These flip-flops contribute to the overall energy-efficient design of the systolic multiplier.

Approximate Correction Error: Approximate correction error refers to the potential inaccuracies introduced during the approximate computation in systolic multipliers. As systolic multipliers may use approximations to reduce power consumption or improve speed, errors may occur in the computed results. Techniques for correcting these errors involve employing error detection and correction mechanisms. Approximate correction error mitigation strategies might include error-tolerant algorithms, redundancy, or feedback loops to refine the results**.**

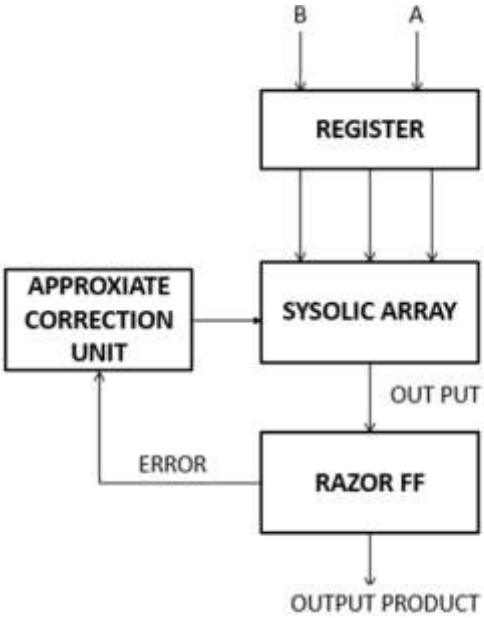

Figure 1. Systolic multiplier.

4.Results and Discussion

Figure 2 shows the simulation results of the proposed systolic multiplier. Figure 3 presents the area analysis of the proposed systolic multiplier. It details the hardware resources required for its implementation, such as the number of logic elements, registers, or other components, indicating the area footprint of the multiplier. Figure 4 depicts the power consumption analysis of the proposed systolic multiplier. It may include metrics such as static power (power consumed when idle) and dynamic power (power consumed during operation), providing insights into the energy efficiency of the multiplier design.

Figure 5 illustrates the setup delay consumption of the proposed systolic multiplier. It may detail the timing characteristics of the multiplier design, including the setup time required for reliable operation, which is crucial for synchronous digital systems. Figure 6 illustrates the hold delay consumption of the proposed systolic multiplier. It may detail the timing characteristics related to the hold time requirement, ensuring that data is stable and valid throughout the entire clock period. Table 1 provides a comparison between the existing method and the proposed method for the systolic multiplier across various metrics or parameters. The comparison may include performance metrics such as throughput, area utilization, power consumption, setup delay, and hold delay, highlighting the differences and improvements achieved by the proposed method compared to the existing one.

Figure 2. Proposed Simulation Result.

Resource	Estimation	Available	Utilization...
LUT	1790	134600	1.33
IO	128	500	25.60

Figure 3. Proposed Area.

Figure 4. Proposed power consumption.

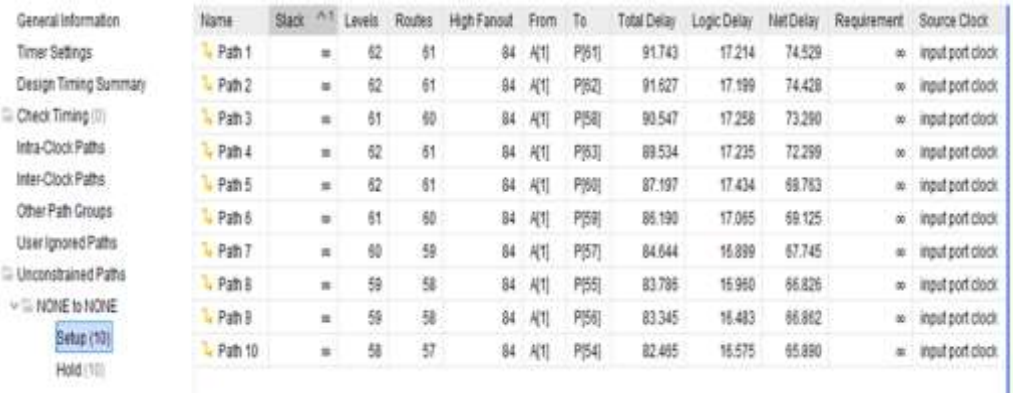

Figure 5. Proposed setup delay consumption.

General Information	Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Sc
Timer Settings	Path 11	∞	3	2	63	B[31]	P[36]	3.484	1.832	1.651	-∞	in
Design Timing Summary	Path 12	∞	3	2	73	B[24]	P[24]	3.544	1.831	1.713	-∞	in
Check Timing (0)	Path 13	∞	3	2	63	B[31]	P[31]	3.555	1.841	1.714	-∞	in
Intra-Clock Paths	Path 14	∞	3	2	63	B[31]	P[32]	3.614	1.838	1.776	-∞	in
Inter-Clock Paths	Path 15	∞	3	2	70	B[30]	P[30]	3.631	1.823	1.808	-∞	in
Other Path Groups	Path 16	∞	3	2	63	B[31]	P[33]	3.691	1.849	1.842	-∞	in
User Ignored Paths	Path 17	∞	3	2	69	B[22]	P[22]	3.695	1.825	1.870	-∞	in
Unconstrained Paths	Path 18	∞	3	2	73	B[26]	P[26]	3.775	1.808	1.966	-∞	in
NONE to NONE	Path 19	∞	3	2	63	B[31]	P[42]	3.791	1.857	1.934	-∞	in
Setup (10)	Path 20	∞	3	2	63	B[17]	P[17]	3.814	1.846	1.968	-∞	in
Hold (10)												

Figure 6. Proposed Hold delay consumption.

Table 1. Comparing existing and proposed methods.

METRIC	EXISTING	PROPOSED
LTU	131	80
IO	32	32
Static	0.154W	0.15W
Dynamic	14.147W	13.262W
Total delay	14.367	21.601
Logic delay	5.640	6.787
Net delay	13.727	14.816

5.Conclusion

In wrapping up our exploration of the systolic multiplier, it's clear that we've uncovered a powerful tool in the realm of computational hardware. Through our journey, we've seen how systolic multipliers efficiently perform multiplication operations by exploiting parallelism and utilizing a regular, structured architecture. In essence, systolic multipliers excel in executing multiplication tasks by breaking down the process into smaller, manageable computations and executing them concurrently. This parallel processing capability leads to significant speed improvements compared to traditional sequential methods, making systolic multipliers invaluable in applications requiring fast arithmetic operations. Moreover, we've delved into the various design considerations and optimization techniques employed to enhance the performance of systolic multipliers. While this work has highlighted the strengths and advantages of systolic multipliers, it's important to acknowledge the challenges and limitations they may encounter. Factors such as resource constraints, power consumption, and complexity of implementation pose potential obstacles that need to be addressed in the design and deployment of systolic multipliers. Looking ahead, the future of systolic multipliers seems promising, with ongoing research focusing on advancements in hardware design, optimization techniques, and integration with emerging technologies such as artificial intelligence and machine learning.

References

[1]. Sadhineni, Harika, S. Josaph, and N. Vaishnavi. "Design and analysis of CMOS based multiplier using LFSR." In AIP Conference Proceedings, vol. 2512, no. 1. AIP Publishing, 2024.

[2]. Srinivas, Chundi Sai, M. S. Manohar, and U. Anusha Rani. "High efficient accurate DL-PO logic multiplier design for low power applications." In AIP Conference Proceedings, vol. 2512, no. 1. AIP Publishing, 2024.

[3]. Kim, Sunwoong, Cameron J. Norris, James I. Oelund, and Rob A. Rutenbar. "Area-Efficient Iterative Logarithmic Approximate Multipliers for IEEE 754 and Posit Numbers." IEEE Transactions on Very Large Scale Integration (VLSI) Systems (2024).

[4]. Minaeifar, Atefeh, Ebrahim Abiri, Kourosh Hassanli, Mehrzad Karamimanesh, and Farshid Ahmadi. "Energy efficient approximate multipliers compatible with error-tolerant application." Computers and Electrical Engineering 114 (2024): 109064.

[5]. Hui, Yajuan, Qingzhen Li, Leimin Wang, Cheng Liu, Deming Zhang, and Xiangshui Miao. "In-Memory Wallace Tree Multipliers Based on Majority Gates Within Voltage-Gated SOT-MRAM Crossbar Arrays." IEEE Transactions on Very Large Scale Integration (VLSI) Systems (2024).

[6]. Parmar, Rushik, Khushil Yadav, Gauraangi Anand, and Gaurav Trivedi. "An SNN Inspired Area and Power Efficient VLSI Architecture of Myocardial Infarction Classifier for Wearable Devices." IEEE Transactions on Circuits and Systems II: Express Briefs (2024).

[7]. Beura, Srikant Kumar, Sudeshna Manjari Mahanta, Bishnulatpam Pushpa Devi, and Prabir Saha. "Inexact radix-4 Booth multipliers based on new partial product generation scheme for image multiplication." Integration 94 (2024): 102096.

[8]. Vakili, Bahareh, Omid Akbari, and Behzad Ebrahimi. "Efficient approximate multipliers utilizing compact and low-power compressors for error-resilient applications." AEU-International Journal of Electronics and Communications 174 (2024): 155039.

[9]. Haq, Shams Ul, Erfan Abbasian, Vijay Kumar Sharma, Tabassum Khurshid, and Hanaa Fathi. "Energy-Efficient High-Speed dynamic logic-based One-Trit multiplier in CNTFET technology." AEU-International Journal of Electronics and Communications 175 (2024): 155088.

[10]. Rohani, Zahra, and Azadeh Alsadat Emrani Zarandi. "Simulation for a low-energy ternary multiplier cell based on Graphene nanoribbon field-effect transistor." Int. J. Nano Dimens 15, no. 1 (2024): 49-62.

[11]. Kuo, Chao-Tsung, and Yao-Cheng Wu. "Area-Power-Delay-Efficient Multi-Modulus Multiplier Based on Area-Saving Hard Multiple Generator Using Radix-8 Booth-Encoding Scheme on Field Programmable Gate Array." Electronics 13, no. 2 (2024): 311.

[12]. Ke, Hongfei, Hao Li, and Peiyong Zhang. "High-performance montgomery modular multiplier with NTT and negative wrapped convolution." Microelectronics Journal 144 (2024): 106085.

[13]. Balaji, M., and N. Padmaja. "Area and delay efficient RNS-based FIR filter design using fast multipliers." Measurement: Sensors (2024): 101014.

[14]. Malathi, L., A. Bharathi, and A. N. Jayanthi. "FPGA design of FFT based intelligent accelerator with optimized Wallace tree multiplier for image super resolution and quality enhancement." Biomedical Signal Processing and Control 88 (2024): 105599.

[15]. Srinivasarao, G., Penchaliah, U., Devadasu, G. et al. Deep learning based condition monitoring of road traffic for enhanced transportation routing. J Transp Secur 17, 8 (2024). https://doi.org/10.1007/s12198-023-00271-3

[16]. Yassin, Hoda, Arash Akhoundi, El-Sayed Hasaneen, and Dante G. Muratore. "A Power-Efficient Oscillatory Synchronization Feature Extractor for Closed-Loop Neuromodulation." IEEE Transactions on Circuits and Systems II: Express Briefs (2024).