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

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.126109

A Review on Design & Implementation of MAC Unit

Rithika[1], Supritha Bekal[2], Pavan[3], Abdul Khadar B M[4], Dr Rashmi Samanth[5]

Student, Electronics & Communication Engineering, MITE, Moodabidri, India[1-4]

Assistant Professor, Electronics & Communication Engineering, MITE, Moodabidri, India[5]

Abstract: One of the most important functional components in any processor is the Multiply-Accumulator (MAC). These functional blocks are currently employed in Fast Fourier Transform (FFT), Finite Impulse Response (FIR) filtering, convolution, and a variety of other DSP and DIP applications. Multiplication and addition are two basic operations that, when compared to other functional blocks, require more hardware resources and computing time. Speed of the multiplier and adder blocks determines the speed of the processor. In this work, a fixed-point power-efficient multiplier and an optimal delay adder block for 2D image convolution have been built and integrated into the MAC unit. Effective multiplication and accumulation units are required due to the demand for high-performance computing systems. The performance of the entire system can be greatly impacted by MAC unit delays, too. In order to find methods that efficiently reduce MAC unit delay, this study compares several adder and multiplier architectures using Xilinx software to handle this difficulty. The results of this study can help researchers and system designers choose optimal designs that balance performance and resource usage, which will ultimately result in increased efficiency and accelerated processing in a variety of computational domains.

Keywords: MAC, Delay, Adders, Multipliers.

I. INTRODUCTION

This research study focuses on the design and analysis of various adders and multipliers to develop a Multiply-Accumulate (MAC) unit with reduced delay. The primary objective is to identify the optimal combination of adders and multipliers that minimizes overall delay while maintaining acceptable performance levels. The study evaluates the trade-off between delay and other design factors through a comprehensive comparison and analysis of different adders and multipliers. By integrating the selected adder and multiplier based on their delay characteristics, a MAC unit is developed to offer improved performance in terms of reduced delay. The research findings provide valuable insights for applications where minimizing latency is critical, such as high-speed data processing and real-time systems.
The study examines various types of adders and multipliers such as,

Adders:

- Ripple carry adder
- Carry save adder
- Carry look ahead adder
- Carry skip adder
- Kogge stone adder

Multipliers:

- Array Multiplier
- Wallace Tree Multiplier
- Booth Multiplier

Overall, by providing a thorough comparative analysis of adder and multiplier architectures for decreasing delay in MAC units, this research adds to the field of high-performance computing. The results of this study can help researchers and system designers choose optimal designs that balance performance and resource usage, which will ultimately result in increased efficiency and accelerated processing in a variety of computational domains.

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

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.126109

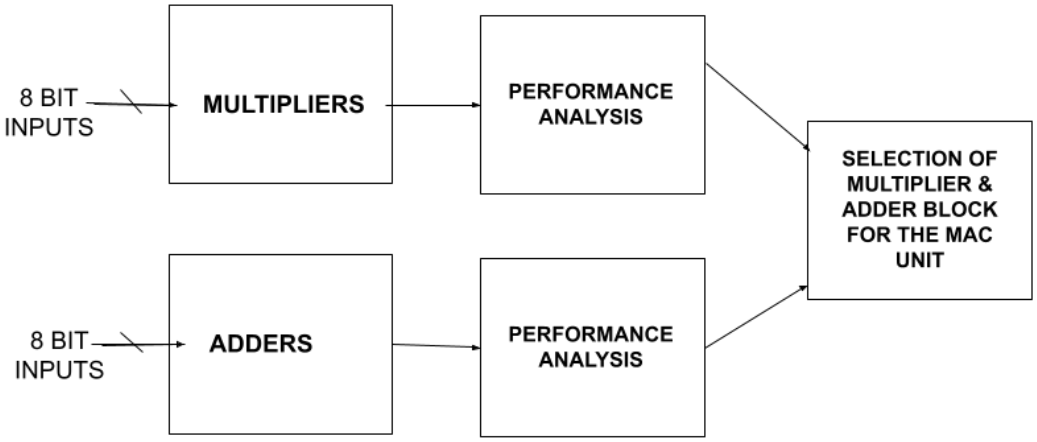

Fig 1: Block Diagram

Traditional design flow of MAC unit is shown in Figure 1. In the first stage, system specification block defines requirements for processing an 8-bit image, ensuring desired functionality and performance targets are met. Accordingly, the optimized logical circuits are designed to meet the architecture functionalities in the next stage. The designer should guarantee the architecture specification evaluation that should meet the necessary performance requirements. If the current architecture requirements are not met, then the designer should consider altering the description of the architecture that is altering the first specification stage of the design flow.

The designer should alter the option of altering the requirements itself. This stage acts as an ultimate sanity test to exact design functionality. After this stage, the physical layout information includes delay, power information from back annotation method which is used to execute post- layout timing analysis. Post-layout synthesis adjusts timing during resynthesis which allows the logic optimization of worsening paths.

Recently, the low power and delay designs rely on efficient implementation of different functional, arithmetic units like multiplier, adder, subtractor and divider circuit design styles to achieve high performance and low power consumption for executing the algorithms. In present era, increasing battery powered devices, consumer products are demand for system performance refinements in terms of quality of visualization, efficient coding, low power, high speed, Low-power designs expand the choice for the designers to apply effective architecture and system level of abstraction for saving the power with additional power aware hardware design.

II. LITERATURE REVIEW

R. Menaka *et al.*, [1] Describes the value of image processing in a variety of industries, including remote sensing, pattern detection, and the medical industry. The location and borders of objects within images are effectively determined using the Sobel filter-based edge detection algorithm, which is presented together with a description of its hardware implementation on an FPGA board. We explain the benefits of the Sobel edge detector, such as adaptability, noise sensitivity, and edge detection without altering grayscale images. The Sobel algorithm's proposed architecture decreases power, area, and delay, and there is room for future improvements to enhance its parameters.

Lei Wu *et al.*, [2] Explains real-time Full-HD Gradient Domain Guided Image Filtering (GDGIF), a well-liked technique for edge-preserving smoothing, VLSI architecture. The suggested architecture makes use of four techniques, including pre-processing, multi-scale down-sampling, parallel structure, and approximated computing, to handle the high computation complexity and additional complex operations required by GDGIF. The implementation findings demonstrate that the suggested architecture, with a design area of 586314 m2 and a power consumption of 18.5mW, is capable of supporting Full-HD image filtering at a rate exceeding 75 frames per second.

Kaiming He *et al.*, [3] Describes a brand-new image filter known as the guided filter computes the filtering result using a local linear model and a guidance image. The guided filter performs better near edges than the bilateral filter when employed as an edge-preserving smoothing operator. As a result, novel filtering applications like dehazing and guided

DOI: 10.17148/IJARCCE.2023.126109

feathering are made possible. It may also transmit the structures of the guidance image to the filtering output. One of the fastest edge-preserving filters is the guided filter, which has a fast and non-approximate linear time algorithm irrespective of kernel size and intensity range. Its usefulness and efficiency in a variety of computer vision and graphics applications are demonstrated by experimental results.

Chandni C.S *et al.,* [4] Explains about Denoising grayscale and colour images with a bilateral filter while preserving edges and using less technology. The filter favours adjacent values over far-off values, blends grey levels based on geometric proximity and photometric analogy, and processes the entire filter window in a single pixel clock cycle. Utilising the distinguishability and evenness of filter components simplifies the filter's design and increases its cost-effectiveness. The final denoised image attributes depend on the chosen filter specs and can support kernels of varying sizes.

Bhavani Koyada *et al.,* [5] Describes the area and speed of 4-bit adders in digital circuits are compared in this study. Ripple Carry, Carry Look Ahead, Carry Save, Carry Skip, Carry Select, Modified Carry Select, and Kogge Stone are some of the adders that have been studied. The results support choosing effective adders based on how well they use resources.

Kalamani. C *et al.,* [6] Describes multipliers are essential in applications that include image processing. Several different metrics are employed to describe the effectiveness of digital analysis, including the circuit's area, potency, and delay analysis. The fractional creation produced by multiplying the 4-bit AXB by a 4X4 Wallace tree results in an increase in delay. By including a 4-2 compressor during the fractional creation generation phase, the delay is significantly decreased. This compressor is utilized to build an 8-bit Wallace multiplier. The simulated outcome demonstrates a reduction in power and latency.

A. Jain *et al.,* [7] Explains about the unique multiplier technique is implemented in HDL in this study and is based on a combination of Vedic mathematics and the Booth-Wallace tree multiplier. In VHDL, an 8X8 multiplier is implemented. ModelSim and Xilinx ISE 14.1 are used to simulate and synthesize the HDL code, respectively. In this study, the performance metrics of 8-bit multipliers developed using different techniques are compared. The comparative findings show that the suggested approach outperforms other multiplier algorithms in terms of speed.

Che- Wei Tung *et al.,* [8] Suggests a pipeline multiply-accumulate (MAC) architecture with high speed and low power consumption. Carry propagations of additions, such as additions in multiplications and additions in accumulations, frequently result in significant power consumption and path delay in a typical MAC. We incorporate a portion of additions into the partial product reduction (PPR) procedure to address this issue. The accumulation and addition of higher significance bits do not take place in the proposed MAC architecture until the PPR phase of the subsequent multiplication. A small-size adder is created to compile the entire number of carries in order to appropriately handle the overflow in the PPR process. Experimental findings demonstrate that the suggested MAC architecture can significantly lower both power consumption and circuit area under the same conditions as prior efforts.

N.M Kumar *et al.,[9]* Describes the development of extremely fast digital devices with little power consumption is a key goal for VLSI circuit designers and manufacturers. The multiplier, which is proven to be a more power-hungry component in electronic circuits, is used for the majority of calculation functions. In the end, the shift-and-add approach has been used to do multiplication operations. The development of various adders opened the door for an increase in the multipliers' execution rate. Combinational circuits are frequently used in parallel multiplication algorithms, which lack feedback mechanisms. VHDL was used to construct the circuit, while Xilinx simulations were used to validate its functions. WTM improvement in this project using the KSA and the Modified

Mansi Jhamb *et al.,* [10] Describes the development of a high-performance processor is a top priority given the developments in the semiconductor industry. One of the most essential components in practically all applications of digital signal processing is the multiplier. The implementation of an 8-bit multiplier using a CMOS full adder, a full adder using a Double Pass Transistor (DPL), and multioutput carry Lookahead logic (CLA) is the subject of this study. The complementary pass transistor (CPL) logic circuits' noise margin issue and speed deterioration at low supply voltage values are avoided by DPL adders. The circuitry's total speed is significantly increased by the multioutput carry lookahead adder.

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

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.126109

III. SUMMARY AND OBSERVATION

Fig 2: Delay, Area of Adders

MULTIPLIERS	DELAY (Nanoseconds)	No. of LUTs
Binary Multiplier	8.315	15
Wallace Tree Multiplier	19.937	128
Booth Multiplier	7.498	17

Fig 2: Delay, Area of Multipliers

ADDERS	DELAY (Nanoseconds)	No. of SLICE LUTs
Ripple Carry Adder	9.688	12
Carry Save Adder	4.107	33
Carry Look Ahead Adder	8.411	12
Carry Skip Adder	9.688	15
Kogge Stone Adder	10.308	15

Fig 3: Delay, Area of Multipliers

Various adders and multipliers were examined in terms of delay and the number of slices required for the design of a Multiply-Accumulate (MAC) unit as mentioned in the fig 2 and fig 3. The analyzed options included Ripple Carry Adder with a delay of 9.688 nanoseconds and 12 slices, the Carry Look Ahead Adder with a delay of 8.411 nanoseconds and 12 slices, the Carry save adder with a delay of 4.107 nanosecond and 33 slices, the Carry Skip Adder with a delay of 9.688 nanoseconds and 15 slices, and the Kogge Stone Adder with a delay of 10.308 nanoseconds and 15 slices.

The Binary Multiplier had a delay of 8.315 nanoseconds and required 15 slices, but the Wallace Tree Multiplier had a latency of 19.937 nanoseconds and required 128 slices. Finally, the Booth Multiplier produced a delay of 7.498 nanoseconds and 17 slices.

The Carry Save Adder and Booth Multiplier were chosen as the best components for the MAC unit after examining performance and resource utilization trade-offs. The Carry Save Adder used 33 slices and had a latency of 4.107 nanoseconds, whereas the Booth Multiplier required 7.498 nanoseconds and 17 slices. These choices aimed to strike a balance between minimizing delay and achieving efficient resource usage in the MAC unit design.

IV. CONCLUSION

After considering performance parameters like delay and area, the Carry-Save Adder (CSA) and Booth Multiplier are the ideal choices for the MAC unit. The parallel addition capability of the CSA reduces delay, while its regular structure enables seamless integration and compact design. The Booth Multiplier reduces area consumption through efficient partial product reduction. This combination optimizes speed as well as resource utilization, resulting in fast computations and compact circuitry. The CSA and Booth Multiplier are suitable for applications that prioritize high-speed operations and limited hardware resources.

REFERENCES

[1]. Menaka, R., Ramadoss Janarthanan, and K. Deeba. "FPGA implementation of low power and high-speed image edge detection algorithm." Microprocessors and Microsystems 75 (2020): 103053.
[2]. Wu, Lei, and Ching Chuen Jong. "A VLSI architecture for real-time gradient guided image filtering." 2016 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC). IEEE, 2016.
[3]. Kaiming, Jian Sun, and Xiaoou Tang. "Guided image filtering." IEEE transactions on pattern analysis and machine intelligence 35.6 (2012): 1397-1409.
[4]. Chandni, C. S., and R. Pushpakumari. "Reduced hardware architecture of bilateral filter for real time image denoising." 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT). IEEE, 2017.
[5]. Bhavani Koyada *et al.*, "A Comparative Study on Adders ." IEEE WISPNET 2017.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023
DOI: 10.17148/IJARCCE.2023.126109

[6]. Kalamani *et al*., "Design of Power Delay Efficient Wallace Multiplier" IEEE ,2023.
[7]. A. Jain, S. Bansal, S. Khan, S. Akhter and S. Chaturvedi, "Implementation of an Efficient N×N Multiplier Based on Vedic Mathematics and Booth-Wallace Tree Multiplier", 2019 International Conference on Power Electronics Control and Automation (ICPECA), pp. 1-5, 2019.
[8]. C.W. Tung and S.H. Huang, "A High-Performance Multiply-Accumulate Unit by Integrating Additions and Accumulations into Partial Product Reduction Process", IEEE Access, vol. 8, pp. 87367-87377, 2020.
[9]. N. M. Kumar, R. S. Adithyaa, B. D. Kumar and T. Pavithra, "Design Analysis of Wallace Tree based Multiplier using Approximate Full Adder and Kogge Stone Adder", 2020 6th International Conference *on* Advanced Computing and Communication Systems (ICACCS), pp. 612-616, 2020.
[10]. Mansi Jhamb *et al*., " Design, implementation and performance comparison of multipliers in power-delay space", Engineering Science and Technology, an International Journal 19 (2016) 355–363.

