JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# FPGA Based High Speed Multiplier Design Using Kogge-Stone Adder

**Y. Mamillu[1], M. Sivaramaprasad[2], P. Keerthi[3], P. AdityaVardhan[4], M. Amrutha[5],**
[1]*Y.Mamillu, Assistant Professor,* Seshadri Rao Gudlavalleru Engineering College, *Andhra Pradesh*
[2,3,4,5,6] *M. Sivaramaprasad, UG Student,* Seshadri Rao Gudlavalleru Engineering College, *Andhra Pradesh*

***Abstract*** **- A multiplier is an operation block found in every processing unit in major programs. Within the current framework, numerous multiplication algorithms have been developed. Designing the multiplier structure involves applying the Adder-based multiplication algorithm. Parallel prefix adders (PPAs) handle the last stage of addition for partial products in the current structure. Multiplier structures are implemented using the Kogge stone adder, Sklansky adder, Brent Kung adder, Ladner Fischer adder, and Han Carlson adder. A crucial factor in any application, such as multiplier design is operating speed. In the suggested method, a MAC unit is employed to increase the multiplier's speed. In the Xilinx 14.2 design suite, Verilog HDL is used to create all of the multiplier structures. The ISIM is used to mimic both the proposed and existing systems. The XST synthesizer is used to synthesize the proposed and existent structures, while the ISIM simulator is used for simulation. The suggested designs are examined in comparison to multiplier designs in terms of delay (ns) and number of LUTs.**

***Index Terms*****—Kogge stone adder (KSA), Adder based multiplier, Multiplier and *Accumulator (MAC) Unit.***

## I. INTRODUCTION

A multiplier is a fundamental component in digital circuits used extensively in arithmetic and logic operations. Multiplication is essentially repeated addition, where one number (the multiplicand) is added to itself a certain number of times dictated by another number (the multiplier). Implementing a multiplier efficiently is crucial in digital systems to achieve high-performance computing. One efficient approach to implement a multiplier in hardware is by utilizing a parallel-prefix adder architecture known as the Kogge-Stone adder. The (KSA) is particularly well-suited for this purpose due to its regular structure and ability to perform fast addition of multi-bit numbers in parallel.

The Kogge-Stone adder operates by breaking down the addition process into multiple stages, each of which computes partial sums and carries efficiently. It exploits parallelism to compute these partial sums concurrently, making it highly efficient for multi-bit addition.

A Multiply-Accumulate (MAC) unit is a fundamental component in digital signal processing and arithmetic operations. It performs the combined operation of multiplication and addition in a single processing element. The MAC unit is commonly used in applications such as digital filters, neural networks, and various mathematical computations. the use of PIPO architecture in MAC units enhances the efficiency, speed, and flexibility of digital signal processing systems, making it a well-liked option for a variety of uses.

The use of a "PIPO" (Parallel-In Parallel-Out) architecture in MAC (Multiply-Accumulate) units offers several advantages, particularly in digital signal processing.

## II. ADDER BASED MULTIPLIER

The multiplier is a high-speed multiplication device. A series of adders is used by the multiplier to generate the final outputs. It is a component of combinational logiccircuits that multiplies two binary values and is built with full adders and half adders to efficiently performthe multiplication. The shift-add method is the most common way to accomplish multiplication. The time required to calculate products using the shift-add method grows as the number of bits in the operand increases. In digital design, there are a variety of multipliers to choose from. Because the adder based multiplier performs better in terms of speed, it should take up less slices and LUTs.

This multiplier mainly consists of three steps:
1.Generation of partial products
2.Partial products addition
3.Final addition

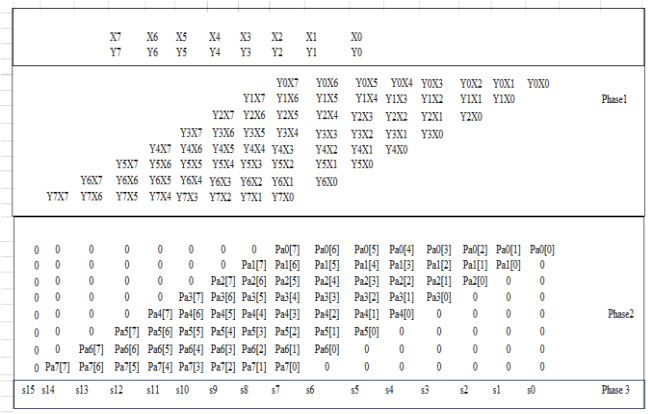



Fig1: Various Phases of Multiplication

Phase 1 consists of multiplying each bit of the input numbers with one another to create partial products. Since the input is only eight bits in size, eight rows of partial products are produced. Phase 2 is made up of numerous subphases where the partial products from Phase 2 are added. The Kogge Stone Adder is used to perform the addition function. Phase 2 begins with the addition operation being applied to the first through last two rows by splitting each pair of partial products produced in phase 2, which results in a result in phase 3 of all rows containing sum terms. Next, the phase 3 result's final addition of partial products is produced.

The addition phase is crucial to the multiplication process as a whole. multiplier designed using Kogge-stone adder was created in fig (a). The primary goal of this suggested design is to achieve a faster high-speed multiplier architecture than that of the current designs.

## III PROPOSING MULTIPLIER USING MAC UNIT

The multiply and accumulate unit, or MAC unit, uses input values from memory to power its multiplier. The multiplier's generated output is then fed to the adder unit, where it is combined with the output from the previous accumulator to produce a new output that is then stored in the accumulator. A parallel in parallel out (PIPO) unit serves as the accumulator. Equation can be used to illustrate MAC operation. (1):

$$Z=A*B+Z \ldots\ldots (1)$$

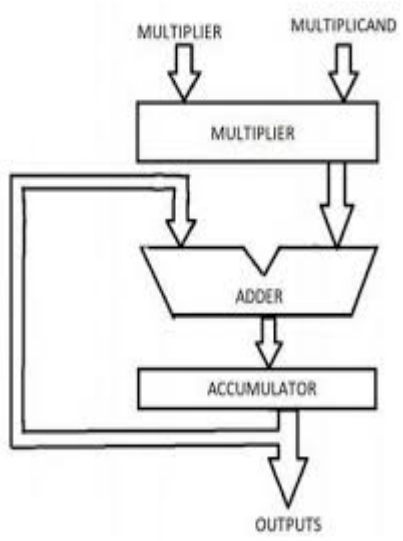



Fig 2: Mac block

If end Z has (2n+1) bits and multiplier A and multiplicand B both have n bits.

The MAC unit's summation unit, which uses up the majority of the delay, is its core.

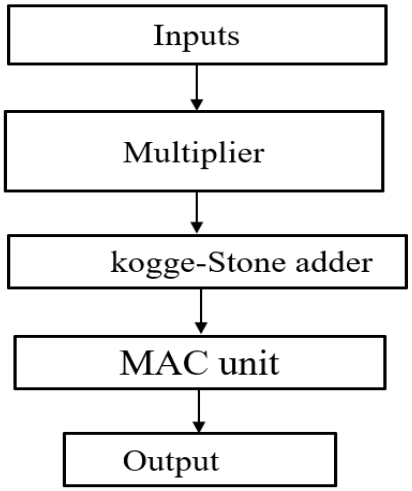



Fig3: Proposing multiplier using Mac unit

To increase speed of the multiplier. we are using Mac unit in the proposing system. In more complex designs, optimizations like parallel processing, and use of dedicated multiplier blocks (e.g., Booth multipliers) can be employed to improve performance and efficiency. Additionally, when targeting specific hardware platforms (such as FPGA or ASIC), considerations for resource utilization, timing constraints, and power consumption become crucial.

## IV. The KOGGE STONE ADDER

The Kogge-Stone adder is a type of parallel prefix adder used in digital circuit design for efficient addition of multi-bit binary numbers. It's named after Peter Kogge and Harold Stone, who introduced it in 1973.

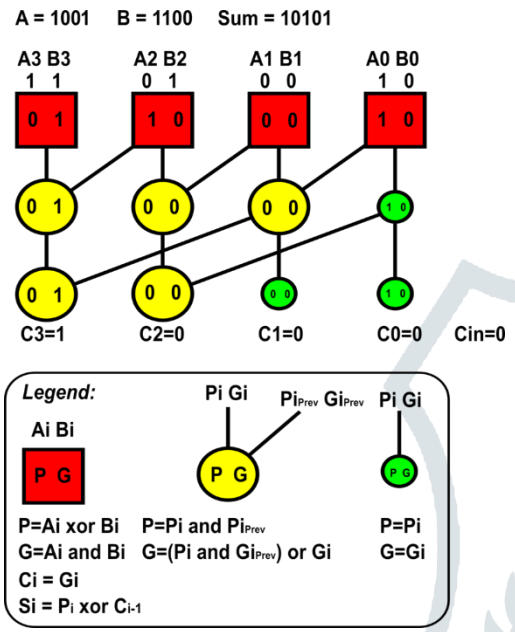



Fig4: 4bit Kogge-stone adder

In a Kogge-Stone adder, each stage performs a carry lookahead operation, allowing for the generation of carry signals across the entire width of the adder in parallel. This means that the carry propagation delay is reduced compared to traditional ripple-carry adders, where the carry must propagate serially through each bits.

Kogge-stone adder have three stages of processing.

- Preprocessing Stage :
  Pi = Ai ⊕ Bi Carry propagate
  Gi = Ai & Bi Carry generate
- Generation of carry:
  Gi = (Pi . Gi*)+Gi
  Pi = (Pi . Pi*)
- Final Processing Stage:
  Ci=Gi
  Si=Pi ⊕Ci-1

Parallel prefix form carry look-ahead adders like the Kogge Stone adder are thought to be the fastest. in the market for high-performance arithmetic circuits. In comparison to other adders, Kogge Stone adders have reduced latency for broad adders. Carry is generated in O (log n) time. The structure's delay is determined by log2n. There are [n (log2n)-n+1] computation nodes in this structure. The area is large, the logic depth is shallow, and the wiring complexity is high. It entails the action of parallel prefix adders in three stages:

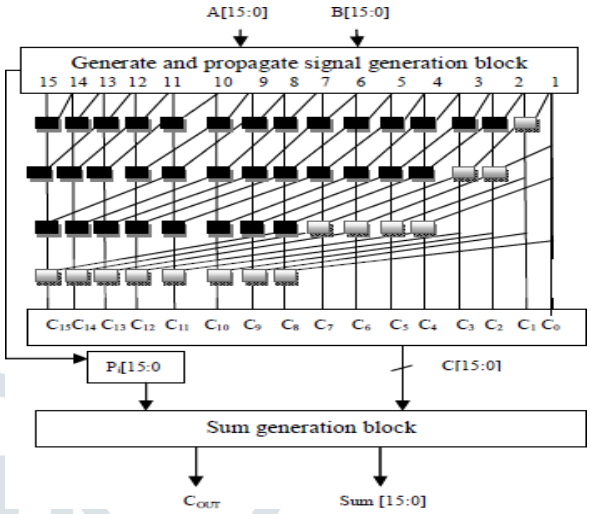



Fig 5: 16bit Kogge-stone adder.

A 16-bit Kogge-Stone adder is a form of parallel prefix adder that is used for quick adding of binary values.. The Kogge-Stone adder is known for its efficient use of hardware resources and its ability to perform addition in logarithmic time complexity. It's particularly suitable for applications requiring high-speed arithmetic operations.

When implementing a 16-bit Kogge-Stone adder, designers need to consider factors such as gate delays, fanout, and routing constraints to optimize performance and minimize area overhead. Additionally, techniques like parallel processing can be employed to further enhance speed and throughput. Overall, a 16-bit Kogge-Stone adder is a sophisticated hardware component suitable for high-performance computing applications where fast arithmetic operations are crucial.

## V. SIMULATION RESULTS:

In the Xilinx ISE Design Suite version 14.7, Multiplier using kogge-stone adder. And Multiplier using MAC are created and compared. All of the designs

created using the Xilinx Synthesis Tool and then tested using the Xilinx ISE simulator.

.

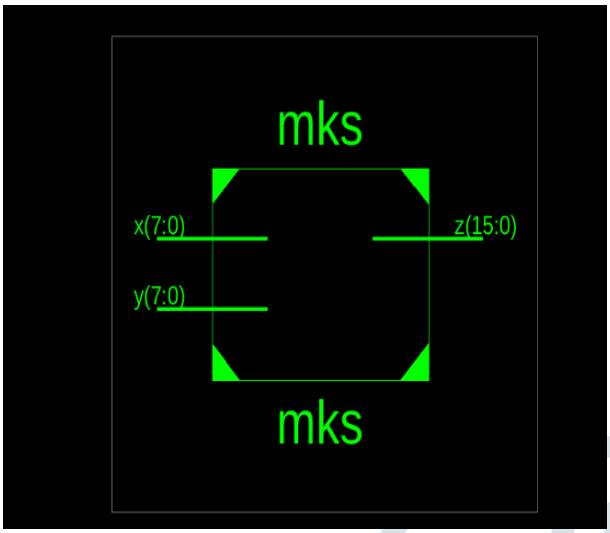



Fig (a):8-bit multiplier using kogge-stone adder

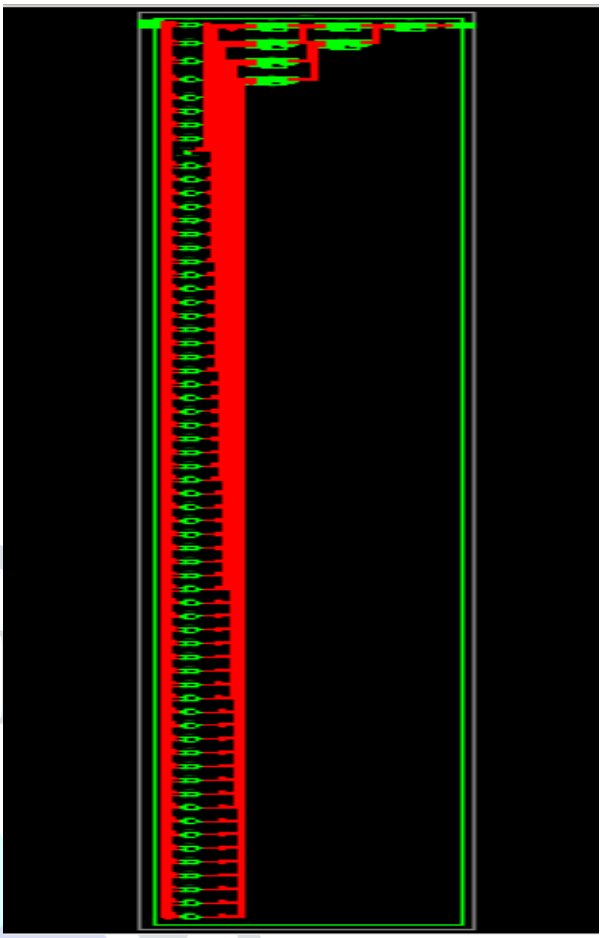


Figure (c): RTL Schematic of 8-bit multiplier using Kogge-stone adder

Fig (b):8-bit mac unit

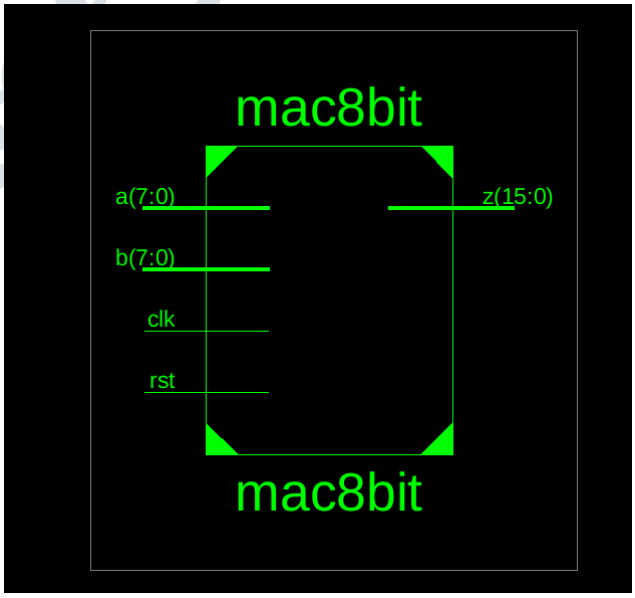


Figure (d): RTL Schematic of MAC

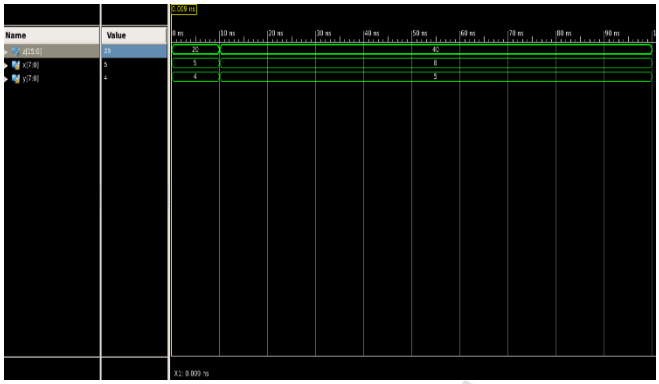
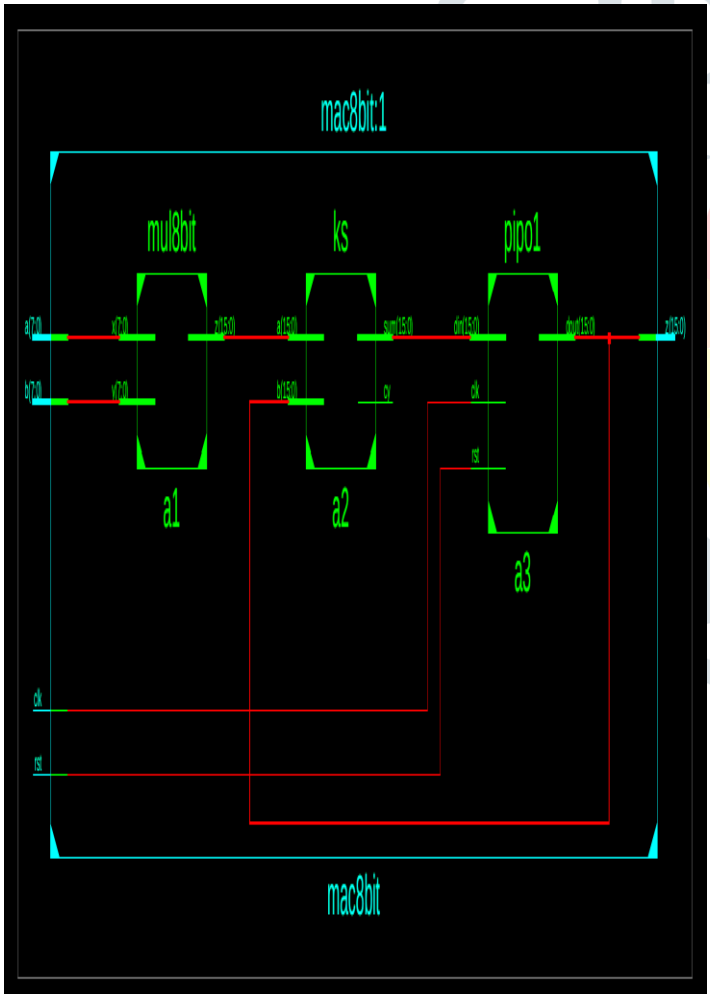


Figure (e): Simulation Wave forms for 8-bit multiplier using kogge-stone adder.




8-bit Multiplier using Kogge stone adder designed, then the RTL schematic is shown in figure (c) and the number of LUT's are used are 82 and number of bonded IOB's are 32. Time delay for this design is 4.810ns and wave forms are shown in figure (e).

Figure (f): Simulation Wave forms for MAC unit

8-bit Multiplier using MAC designed, then the RTL schematic is shown in figure (d) and the number of LUT's are used are 102 and number of bonded IOB's are 34. Time delay for this design is 1.325ns and wave forms are shown in figure (f).

We compared the existing system and proposing system.the multiplier using mac unit has more Speed than the multiplier using kogge-stone adder.

Table 1: Comparison of two Multipliers structures.

| Designs | No. of LUT's | No. of IOBs | Delay |
|---|---|---|---|
| Multiplier designed using Kogge-stone adder (Existing System) | 82 | 32 | 4.810ns |
| Multiplier designed using MAC unit (proposed system) | 102 | 34 | 1.310ns |

In the above table Existing system has 4.810ns and proposing systemhas 1.325ns delays.

## VI . CONCLUSION

FPGA-based high-speed multipliers using Kogge-Stone adders provide a powerful solution for accelerating arithmetic operations in various digital systems. Their parallel processing capability, logarithmic time complexity, and efficient hardware utilization make them well-suited for demanding computational tasks in FPGA-based designs.

## ACKNOWLEDGMENT

We are extremely thankful to my project guide, who assisted us in every way to do our work quickly. We also acknowledge the facilities provided by Seshadri Rao Gudlavalleru Engineering College, which allowed us to complete this Project..

REFERENCES

[1] Yamini devi Ykuntam, Katta Pavani , Krishna Saladi," Design and analysis of High speed wallace tree multiplier using parallel prefix adders for VLSI circuit designs," July 1-3, 2020,

[2] U Penchalaiah and Siva Kumar VG," Design of High-Speed and Energy-Efficient Parallel Prefix Kogge Stone Adder," September 04,2020.

[3] A. M. Bhavani and B. L. Jahnavi , Design and Implementation of Wallace Tree Multiplier Using Parallel Prefix Adders,'' June 2022

[4] Shaik Mahaboob Basha , Y.Mamillu," A Low Power Radix-4 Adder Using Domino Logic," Apr2023 .

[5] R. Arun Sekar, D. Ganeshkumar, A. Rajendran, S. Sasipriya'' High Speed Multiplier Design Using Kogge Stone Adder,'' April 2019.

[6] Goli Praveen, 2gomasani Ram Prasad, 3maleela Sarala, 4gongadi Prabhu Kumar, 5madhava Rao Alla'' An Efficient Design of Mutiplier Using Kogge Stone Adder,'' Volume 9, Issue 3, 2019.

[7] Mohammad Nasiruddin, Dr. Prashant S. Sharma, Dr. Vijay S. Chourasia,'' Implementation of Multiplier-Accumulator (MAC) unit using 16bitVedic Multiplier,'' Vol-05, Issue-08, Nov 2019.

[8] Kalind Mehta, Gaurav Patel,"Design and implement 64 bit MAC(Multiplier and accumulator),'' Vol-2 Issue-3 2016.

erformanc

[9] P. Siva Nagendra Reddy* and M. Saraswathi,'' Design and Implementation of FPGA base64-bit MAC Unit using VEDIC Multiplier and Rever--sible Logic Gates ,'' Vol 10 (3),January 2017.

[10] Aapurva Kaul, Abhijeet Kumar, Simulation of 64- bit MAC Unit using Kogge Stone Adder and Ancient Indian Mathematics, Vol. 6, Issue 6, ( Part -2) June 2016.

[11] Komal Ganer, Dr. S. S. Shriramwar,"FPGA Implementation and Comparative Analysis of an Efficient Multiplier" Volume 9, Issue 8 August 2021.

[12] Neha Shukla, Prof. Deepak Kumar, High Speed Area Efficient Vedic Multiplier using Modified Kogge Stone Adder, Vol. 5, Issue 12, December 2016.

[13] C Ranjit Kumar G Rahul Ram N Chandu Reddy PigiliSuresh, Design of Square and Multiply and Accumulate(MAC) Unit by using Vedic Multiplication Techniques, Volume 4, Issue 12, December-2013.

[14] Mangapathi Vinitha, Kumarganesh.S, FPGA Implementation of Multiplier-Accumulator Unit Using Vedic Multipliers And Reversible Gates, Volume 19 Issue 02 February 2023.

[15] Vegireddi Swapna, Dr. Nihar Ranjan Panda, Bandi Sarada, Performance analysis of 32 bit vedic multiplier design using nikilam and kogge stone adder, November 2019 IJSDR Volume 4, Issue 11.