International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 6 ǁ 2022 ǁ PP. 284-289

Design of Vedic Mathematics Based On Mac Unit for Power Optimization

Suma Nair[1], K. Sai Naveen[2], M. Nagamani[3], M. Sushma Nivasini[4]
Assistant Professor, Department of Electronics and Communication Engineering
Vel Tech Rangarajan Dr. Sagunthala R&D Institute of
Science and Technology
Avadi, Chennai-600062

ABSTRACT:
A significant part of machine learning accelerator operations is composed of multiply accumulate calculations. Among the most popular common procedures Signal processing and other applications avail use out of multiply and accumulate. Digital signal processors (DSPs) require a multiplier to function. The performance of a DSP is determined by its factors such as power, LUT use, and latency. As a result, a multiplier featuring high power and delay efficiency must be designed. A 32 bit multiply accumulate is developed utilising a a multiplier having high power and delay economy in this work. When compared to the standard pipelined MAC, the simulation outcomes reveal that suggested multiply accumulate unit saves vitality and takes up less space. This proposed system was written in Verilog HDL, Modalism 6.4 c was used to simulate it, and Xilinx was used to synthesise it.

Keywords: *Modalism software and Xilinx ISE.*

Date of Submission: 25-05-2022 Date of acceptance: 05-06-2022

I.INTRODUCTION

A wide range of digital signal processing (DSP) applications are supported by the multiplier-accumulator (MAC) device. It also gives the microcontroller signal processing capabilities for servo and audio control, among other things. Deep neural networks (DNNs) have shown to be effective in a wide range of outcomes, include image classification and speech recognition. Because an example of a DNN solution necessitates an large number of vector-matrix multiplication calculations, a range of specialised machine learning hardware has been offered to speed up the process.

For simultaneous calculations, a machine learning accelerator contains a large number of multiply–accumulate (MAC) units Calculations, or the unit frequently contains the system's timing-critical pathways. A multiplier is made up of various computing components, like as production process in parts, adding a column, and last addition. The carry-propagation adder makes up an accumulator.

Long critical routes via these phases result in overall system performance decrease. Various solutions have been investigated to reduce this difficulty. Wallace and Dadda are multipliers. well-known instances of quick The carry-lookahead (CLA) adder is a column addition device. frequently used to shorten the accumulator's critical state or the multiplier's latest addition step. Meanwhile, the machine learning algorithm performs a MAC operation to create a partial expansion, which is the accumulation the quantity multiplied by the input.

To minimise the In a MAC unit, the multiply and accumulate operations decrease the number of carry-propagation steps from two to one. frequently combined. However, Even though, such a structure exists. has a considerable Delay on critical path, about comparable to a multiplier's critical path delay. Pipelining is generally understood as one of the most popular common methods for boosting the operating time Although pipelining is a feasible alternative, effective approach to decrease important line difficulties, the insertion of multiple flip-flops leads a growth in space and usage of energy

II. LITERATURE REVIEW

Sheetal N. Gadakh and Amitkumar Khade [1], The most popular method is multiplication. fundamental in operation arithmetic. Signal processing applications frequently include multiplication-based procedures including Convolution, Fast Fourier Transform(FFT), multiply and accumulate unit (MAC) , and filtering. Due to the fact that multiplication takes up the majority of the DSP processing time systems, rising multipliers are required. The solution is helped to some degree by ancient Vedic mathematics. In this study, the Urdhwa-Tiryagbhyam principle is employed to create a 1616 Bit Vedic multiplier employing vertical and transverse multiplication, with optimization accomplished using carry saving adders.

Srinivasa Akella [2], One of the most important common processes a signal processing concept and other applications is multiply and accumulate (MAC). Digital signal processors (DSPs) require a multiplier to function. The performance of a DSP is determined by its factors such as power, LUT use, and latency. As a result, a multiplier with excellent power and delay efficacy must be designed. A carry-save adder and an 8-bit rig veda multiplier are used to create a 16-bit MAC unit in this study. A comparison of the Square-Root (SQR) Carry with the current Vedic multiplier (8 bits).

S Yaswanth [3], We conceived and investigated an(RC)Ripple carry adder, (CSL)Carry select adder, (CSl)Carry select adder utilising Binary to excess- 1(BEC) converter, Ling Carry Select Adder (CS), Brent Kung (BK), and Carry Select Adder (CSL) in the work below. ModelSim-Altera 6.3g was used to model the designs, and 12.1 Xilinx ISE Design Suite was used to synthesise them. The performance and area of the (VM)vedic multiplier are compared using various carry choose adder designs. It was discovered that the (VM)vedic multiplier is 8 bits with the (BK)Brent

Kung (CSL) Carry the selected adder produced excellent results in terms of area and latency.

L. Ranganath [4], A framework for parallel data processing made up number of processors is known as a synthetic neural network (ANN). The processing unit determines whether or not the network is efficient. As a result, it is necessary to build a processing unit that is both efficient and effective. The MAC (Multiplication and Accumulation) and Activation units make up the processing unit. Carry look ahead multiplier and booth multiplier were used to create the processing MAC unit in a system that already exists. The current processing unit has a delay and uses more space and electricity. To address the shortcomings, a new unit in processing, the Select adder Vedic multiplier with scale factor, was developed (SQRT-CSLA). The suggested architecture addresses the shortcomings of the present system while improving overall network performances.

R Anjana [5], Because of the ever-increasing need for high-performance processors, innovation in technology in this area is a hot topic. When it comes to parameter speed, the first thing that comes to mind is multiplication. Multiplication Most throughput determination execution time are influenced by A system's CPU cycle time since it is a crucial All mathematical computations depend on this function.. We provide a revolutionary architecture for doing multiplication at a high percentage utilising ancient vedic mathematics in this study. Urdhva Triyakbhyam, one of the most effective sutras in vedic mathematics, makes a difference in the multiplication procedure itself Valentina BianchiIlaria [6], In the majority all signal and image processing applications, multiplication is a basic operation. A novel a Vedic multiplier's architecture based on the 'Urdhava-tiryakbhyam' technique is suggested in this study. The architecture given here is totally modular and intended to be used where configurability crucial in model-based designs critical. To suit the required frequency clock, this design is prone to have been used in both pure combinational and pipelined systems form.

Can Eyupoglu [7], Multiplication is one of the most crucial activities in computer math. processes. Multiplication is utilised in a variety of operations, square root, division, and reciprocal computation. Furthermore, multiply is an important mathematical operation in a variety of Convolution, correlation, frequency analysis, and image processing are example of signal processing applications. Multiplication's efficiency operations is critical for these applications' processing times. One of the algorithms created to improve efficiency is the Nikhilam algorithm.

R. Balakumaran [8], Combining reversible logic functions Look ahead with hybrid carry adders, a unique accumulator and multiplier approach is proposed in this study. In comparison to a conventional multiplication procedure, the modified booth method produces reduced time and moderates the quantity of partial products. Controlling the total MAC delay is done Using the Carry look-ahead adder as a multiplier and accumulator Reversible logic principal function design is to decrease circuit complexity, power consumption, and information loss. We look at several reversible logic gates to see how we can construct a whole adder architecture. We can presented a novel hybrid is a CLA based on the current situation hierarchical CLA, which has excellent computation, power consumption, and area performance. The resulting design's area, delay, and power difficulties are detailed.

III.METHODOLOGY

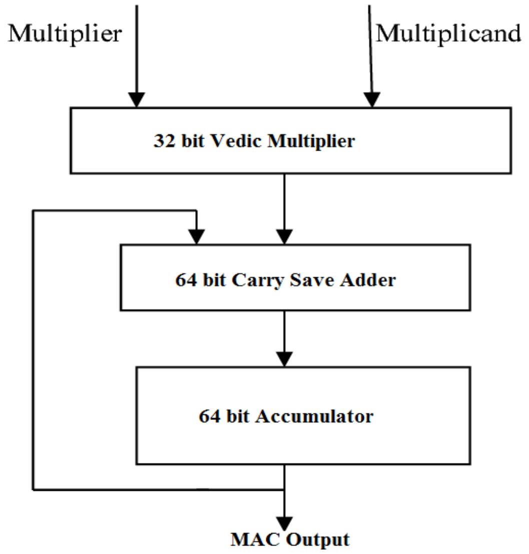


Fig 1: Block diagram

3.1 PROPOSED DESIGN:

A 32-bit Mac may be built in two pieces utilising a Vedic multiplier and a reversible logic gate. The first is the multiplier unit, which uses the Urdhava Triyagbhayam sutra to replace a normal multiplier with a Vedic multiplier. The MAC unit's basic operation is multiplication. The multiplier unit's key challenges are Energy consumption, absorption, area, velocity, and delay are all factors to consider. To circumvent them, we use fast multipliers in DSP, networking, and other applications. There will be two options primary criteria which increase MAC unit results: decreasing accumulator weight and partial outcomes. The critical route and latency are determined by the multiplier in a digital system. The partial products are made up of 2N-1 cross products of varying widths for N*N.

3.2 REQUIREMENT ANALYSIS:

The multiply– accumulate method is a frequent a method in which the result of two integers is computed, then added to an accumulator. A multiplier accumulator is the hardware device that executes the operation; A MAC or MAC operation is another name for this operation. The MAC function changes the value of the accumulator a: A= A+(B*C).

3.3 Multiplication of two 2-digits numbers:

To divide a two-digit number by another two-digit number. Three steps are required. The graphic below might help you recall the vertically across pattern needed to multiply two 2-digit values**.**

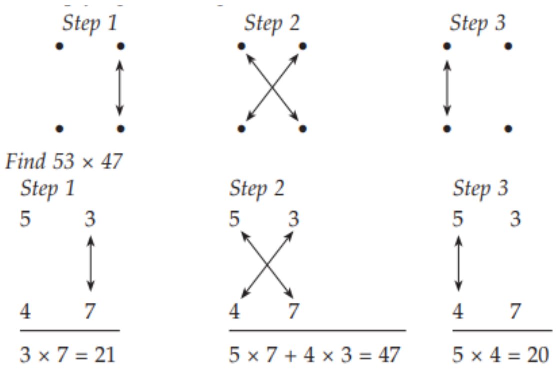


Fig 2: Multiplication of two 2-digits numbers

Multiplication of two 3-digits numbers

We need 5 steps to multiply a three-digit number by another three-digit number. The graphic below might help you recall the vertically across pattern needed to multiply two three-digit values.

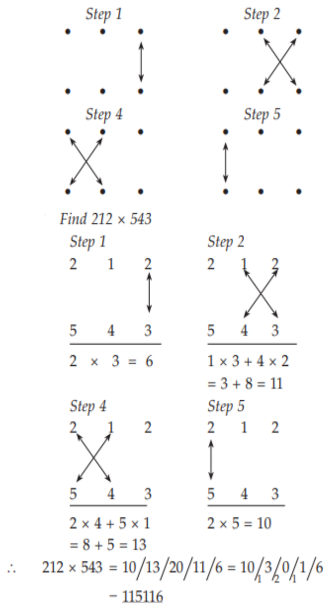


Fig 2: Multiplication of two 3-digits numbers

3.4 SOFTWARE REQUIREMENT:

ModelSim is a handy tool for stimulating your to see both outlets and inside signals from modules. It enables both psychological as well as temporal simulations; However, the focus of The topic of this article will be behavioural simulations. It's important to remember that all these simulators are based on models, therefore the systems accuracy determines the outcome that make up the simulation. Model TechnologyTM Incorporated produces Software /VHDL, ModelSim /VLOG, Simulation /LNL, and Data was needed /PLUS. Without Cloud computing Concept prior permission, unauthorised copying, duplicating, or other reproduction is forbidden. The material these handbook is subject to availability and will not constitute legal advice. imply Model Technology's commitment.

Xilinx is a programmable logic device manufacturer. It is credited with developing the first semiconductor business to adopt electronics fabrication and developing the field programmable gate array (FPGA). Strategy. In 1984, corporation was created in Silicon Valley and has offices in California, the United States, Ireland, Singapore, and Tokyo. Corporate headquarters can be found across North America, Asia, and Europe.

Simulation Tool: ModelSim and Xilinx ISE
Operating System: Windows

3.5 URDHVA TIRYAKBHAYAM MULTIPLIER:

Urdhva Tiryakbhayam signifies vertically, and the most usual approach is across. Urdhva Tiryakbhayam has universal use in all situations and may be used to multiply decimal and binary integers. The multiplication stages in the Urdhva Tiryakbhayam sutra for two decimal integers are illustrated in Figure. If there is a preceding carry, the values at the end of the route are multiplied, it is also added. Multiple multiplications of any step have been combined with the preceding carry. The result bit is the unit place digit, and the carry for the next step is the tens place digit. The multiplier is unaffected by the processor's clock frequency since n parallel, the entire combination and sums are computed.As a result, microprocessors do not need to operate at an increasingly high frequency, maximising processing power. It can be simply laid out in a silicon chip because to its regular structure. In comparison to the traditional way of multiplication, it saves time, energy, and space.

3.6 EXISTING SYSTEM:

The operations multiply and accumulate are are commonly combined in this existing MAC unit to minimise the Carry-propagation steps are reduced from two to one. We'll apply those findings to the construction of a multiply accumulator that accepts two IZ-bit elements A and B as inputs and calculates ((A xB)+C) in just the same time as an ideal n-bit multiplier. The parity of a number of input bits (23) is modified using on the two earliest values, a single half-adder. In this approach, the values from the adding appropriate are combined with the partial products as supplies to the PPRT.

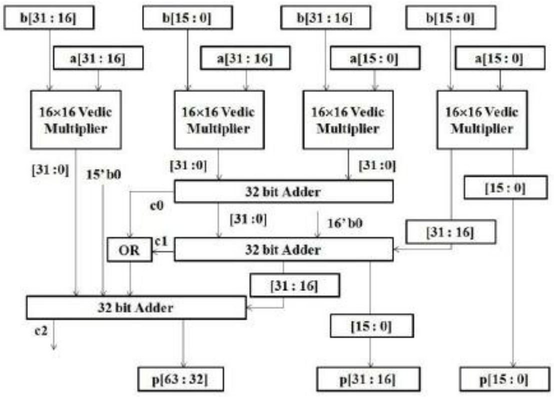
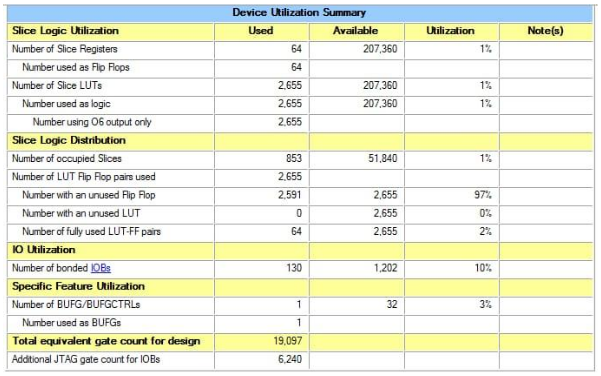


Fig 2: 32bit Vedic Multiplier

3.7 SIMULATION OUTPUT:



Device Utilization Summary				
Slice Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Registers	64	207,360	1%	
Number used as Flip Flops	64			
Number of Slice LUTs	2,655	207,360	1%	
Number used as logic	2,655	207,360	1%	
Number using O6 output only	2,655			
Slice Logic Distribution				
Number of occupied Slices	853	51,840	1%	
Number of LUT Flip Flop pairs used	2,655			
Number with an unused Flip Flop	2,591	2,655	97%	
Number with an unused LUT	0	2,655	0%	
Number of fully used LUT-FF pairs	64	2,655	2%	
IO Utilization				
Number of bonded IOBs	130	1,202	10%	
Specific Feature Utilization				
Number of BUFG/BUFGCTRLs	1	32	3%	
Number used as BUFGs	1			
Total equivalent gate count for design	19,097			
Additional JTAG gate count for IOBs	6,240			

Fig 5: Device Utilization Summary

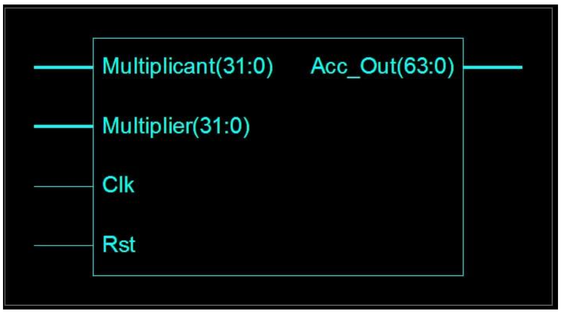


Fig 6: RTL Schematic

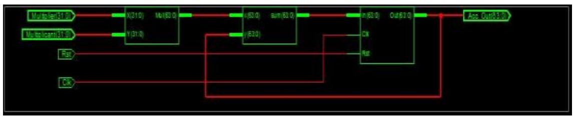


Fig 6: Multiplier, accumulation and register

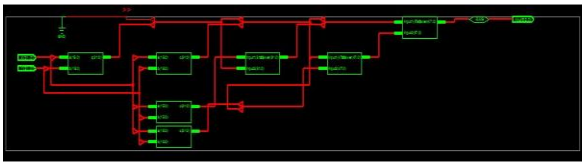


Fig 6: Proposed Multiplier

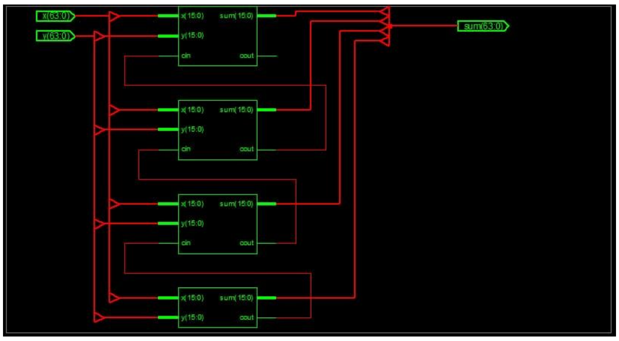


Fig 6: Parallel Prefix Adder

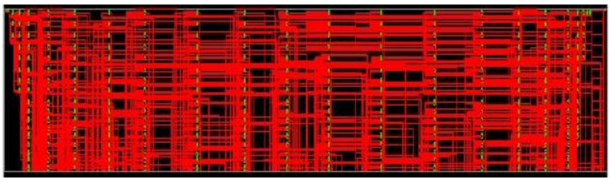


Fig 6: LUTs

IV.CONCLUSION

A 32 bit multiply accumulate unit with a 32bit Vedic was developed a multiplier and a carry-save adder. It was written in verilog HDL and based on the UT sutra. Spartan3 FPGA was used for the implementation. It showed good power reduction, as well as considerable improvements in terms of area or latency. An illustration was made between a traditional multiplier and a multiplier that already exists. The suggested multiplier may be anything employed in Applications for DSP to improve the pace as well as effectiveness of the MAC unit. This work can be expanded in the future by substituting reversible logic gates for multipliers to get even more power savings and improved performance.

REFERENCES:

[1]. Sheetal N. Gadakh and Amitkumar Khade, "Design and Optimisation of 16x16 Bit Multiplier Using Vedic Mathematics", International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), September 2016.

[2]. Srinivasa Akella, Vamsi Krishna and S R Ramesh, "An Efficient Design of 16 sBit MAC Unit using VedicMathematics", International Conference on Communication and Signal Processing, pp. 0319-0322, April 2019.

[3]. S Yaswanth and R.Vishnu Vijeth Nagaraj, "Design and analysis of high speed and low area vedic multiplier using carry select adder", International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), pp. 1-5, February 2020.

[4]. L. Ranganath, D. Jay Kumar and P. Siva Nagendra Reddy, "Design of MAC Unit in Artificial Neural Network Architecture using Verilog HDL", International Conference on Signal Processing Communication Power and Embedded System (SCOPES), pp. 607-612, 2016.

[5]. R Anjana, B Abishna, M. S Harshitha, E Abhishek, V Ravichandra and M S Suma, "Implementation of vedic multiplier using Kogge-stone adder", International Conference on Embedded Systems (ICES), pp. 28-31, July 2014.

[6]. Valentina BianchiIlaria and De Munari, "A modular Vedic multiplier architecture for model-based design and deployment on FPGA platforms" in Microprocessors and Microsystems, ScienceDirect, vol. 76, July 2020.

[7]. Can Eyupoglu, "Investigation of the Performance of Nikhilam Multiplication Algorithm", World Conference on Technology Innovation and Entrepreneurship, vol. 195, pp. 1959-1965, July 2015.

[8]. R. Balakumaran and E. Prabhu, "Design of high speed multiplier using modified booth algorithm with hybrid carry look-ahead adder", International Conference on Circuit Power and Computing Technologies (ICCPCT), 2016.