ISSN 2250-3153

High Speed, Power and Area efficient Algorithms for ALU using Vedic Mathematics

V.Vamshi Krishna, S. Naveen Kumar

Department of ECE, JNTUH College of Engineering, Karimnagar, India

Abstract- Digital signal processing (DSP) is the technology that is omnipresent in almost every engineering discipline. A typical processor devotes a considerable amount of processing time in performing arithmetic operations, particularly multiplication operations. Multiplication is one of the basic arithmetic operations and it requires substantially more hardware resources and processing time than addition and subtraction. In fact, 8.72% of all the instruction in typical processing units is multiplication. The core computing process is always a multiplication routine; therefore, DSP engineers are constantly looking for new algorithms and hardware to implement them.

Vedic mathematics is the name given to the ancient system of mathematics, which was rediscovered, from the Vedas between 1911 and 1918 by Sri Bharati Krishna Tirthaji. The whole of Vedic mathematics is based on 16 sutras (word formulae) and manifests a unified structure of mathematics. This paper presents the design of a low power high speed algorithms for arithmetic logic units using this ancient mathematics techniques and also their hardware implementation. Even convolution algorithms using this technique are discussed along with their FGPA implementation. Employing these techniques in the computation algorithms of the coprocessor has reduced the complexity, execution time, area, power. Further research prospects may include the design and development of a Vedic DSP chip using VLSI technology.

Index Terms- DSP, Vedic mathematics, Urdhva Tiryakbhyam Sutra.

I. INTRODUCTION

In any processor the major units are Control Unit, ALU and Memory read write. Among these units the performance of any processor majorly depends on the time taken by the ALU to perform the specified operation. Multiplication is an important fundamental function in arithmetic operations. Multiplication-based operations such as Multiply and Accumulate (MAC) and inner product are among some of the frequently used Computation Intensive Arithmetic Functions (CIAF) currently implemented in many Digital Signal Processing (DSP) applications such as convolution, Fast Fourier Transform (FFT), filtering and in microprocessors in its arithmetic and logic unit. Since multiplication dominates the execution time of most DSP algorithms, so there is a need of high speed multiplier.

The demand for high speed processing has been increasing as a result of expanding computer and signal processing applications. Higher throughput arithmetic operations are important to achieve the desired performance in many real-time signal and image processing application. One of the key arithmetic operations in such applications is multiplication and the development of fast multiplier circuit has been a subject of interest over decades. Reducing the time delay and power consumption are very essential requirements for many applications. This work presents different multiplier architectures. Multiplier based on Vedic Mathematics is one of the fast and low power multiplier.

Digital multipliers are the most commonly used components in any digital circuit design. They are fast, reliable and efficient components that are utilized to implement any operation. Depending upon the arrangement of the components, there are different types of multipliers available. Particular multiplier architecture is chosen based on the application.

In many DSP algorithms, the multiplier lies in the critical delay path and ultimately determines the performance of algorithm. The speed of multiplication operation is of great importance in DSP as well as in general processor. In the past multiplication was implemented generally with a sequence of addition, subtraction and shift operations. There have been many algorithms proposals in literature to perform multiplication, each offering different advantages and having tradeoff in terms of speed, circuit complexity, area and power consumption.

The multiplier is a fairly large block of a computing system. The amount of circuitry involved is directly proportional to the square of its resolution i.e. A multiplier of size n bits has n^2 gates. For multiplication algorithms performed in DSP applications latency and throughput are the two major concerns from delay perspective. Latency is the real delay of computing a function, a measure of how long the inputs to a device are stable is the final result available on outputs. Multiplier is not only a high delay block but also a major source of power dissipation. That's why if one also aims to minimize power consumption, it is of great interest to reduce the delay by using various delay optimizations.

In this thesis work, Urdhva tiryakbhyam Sutra is first applied to the binary number system and is used to develop digital multiplier architecture. This is shown to be very similar to the popular array multiplier architecture. This Sutra also shows the effectiveness of to reduce the NxN multiplier structure into an efficient 2x2 multiplier structures. Nikhilam Sutra is then discussed and is shown to be much more efficient in the multiplication of large numbers as it reduces the multiplication of two large numbers to that of two smaller ones. The proposed multiplication algorithm is then illustrated to show its computational efficiency by taking an example of reducing a 4x4-bit multiplication to a single 2x2-bit multiplication operation. This work presents a systematic design methodology for fast and area efficient digit multiplier based on Vedic

mathematics. The Multiplier Architecture is based on the Vertical and Crosswise algorithm of ancient Indian Vedic Mathematics.

II. OBJECTIVE

The main objective of this thesis is to design and synthesis of an ALU, which can be used in any processor application. The ever increasing demands in enhancing the ability of processors to handle the complex and challenging processes has resulted in the integration of number of processor cores into one chip still the load on processor is not less. This load is reduced by supplementing the main coprocessor which is designed to work on specific types of functions like numeric computations, signal processing, graphics etc. The speed of ALU depends on the multipliers. In algorithm and structure levels, numerous multiplication techniques have been developed to enhance the efficiency of multiplier which concentrates on reducing the partial product and their additions. But in this case principle behind the multiplication remains same. Use of Vedic mathematics for multiplication strikes difference in actual process.

III. ANCIENT VEDIC METHODS

The use of Vedic mathematics lies in the fact that it reduces the typical calculations in conventional mathematics to very simple ones. This is so because the Vedic formulae are claimed to be based on the natural principles on which the human mind works. Vedic Mathematics is a methodology of arithmetic rules that allow more efficient speed implementation. This is a very interesting field and presents some effective algorithms which can be applied to various branches of engineering such as computing.

3.1. Urdhva Tiryakbhyam Sutra

The given Vedic multiplier based on the Vedic multiplication formulae (Sutra). This Sutra has been traditionally used for the multiplication of two numbers. In proposed work, we will apply the same ideas to make the proposed work compatible with the digital hardware. Urdhva Tiryakbhyam Sutra is a general multiplication formula applicable to all cases of multiplication. It means "Vertically and Crosswise". The digits on the two ends of the line are multiplied and the result is added with the previous carry. When there are more lines in one step, all the results are added to the previous carry. The least significant digit of the number thus obtained acts as one of the result digits and the rest act as the carry for the next step. Initially the carry is taken to be as zero. The line diagram for multiplication of two 4-bit numbers is as shown in figure 1.

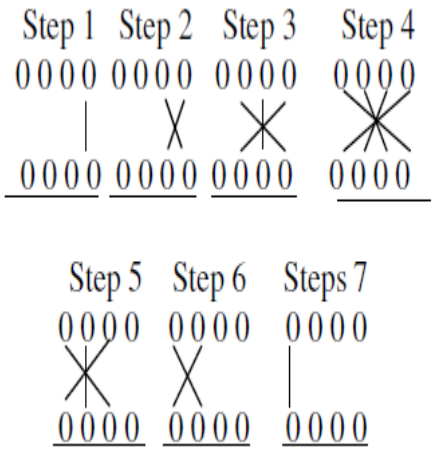


Fig.1: Line diagram for multiplication for two 4-Bit Number

Line diagram for the multiplication is shown in Fig. 1. The digits on the two ends of the line are multiplied and the result is added with the previous carry. When there are more lines in one step, all the results are added to the previous carry. The least significant digit of the number thus obtained acts as one of the result digits and the rest act as the carry for the next step. Initially the carry is taken to be as zero. Now we will extend this Sutra to binary number system.

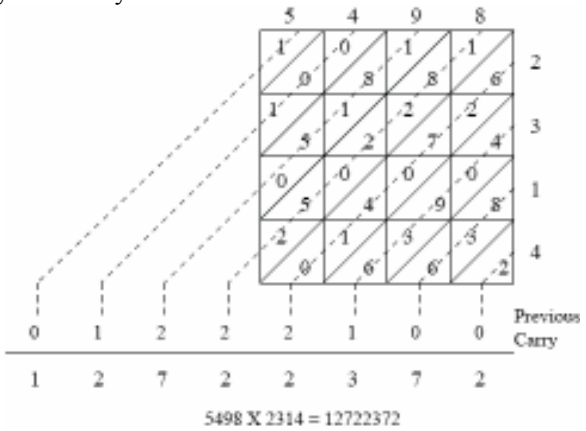


Fig.2: Alternate Method of multiplication by Urdhva tiryakbhyam Sutra

For the multiplication algorithm, let us consider the multiplication of two 4 bit binary numbers a3 a2 a1 a0 and b3 b2 b1 b0. As the result of this multiplication would be more than 4 bits, we express it as … r4 r3 r2 r1 r0. As in the last case, the digits on the both sides of the line are multiplied and added with the carry from the previous step. This generates one of the bits of the result and a carry. This carry is added in the next step and hence the process goes on. If more than one line are there in one step, all the results are added to the previous carry. In each step, least significant bit acts as the result bit and all the other bits act as carry. For example, if in some intermediate step, we will get 011, then1 will act as result bit and 01 as the carry. Thus we will get the following expressions:

r0=a0b0; (1)
c1r1=a1b0+a0b1; (2)
c2r2=c1+a2b0+a1b1 + a0b2; (3)
c3r3=c2+a3b0+a2b1 + a1b2 + a0b3; (4)
c4r4=c3+a3b1+a2b2 + a1b3; (5)
c5r5=c4+a3b2+a2b3; (6)
c6r6=c5+a3b3 (7)

c6r6r5r4r3r2r1r0 being the final product. Hence this is the general mathematical formula applicable to all cases of multiplication. All the partial products are calculated in parallel and the delay associated is mainly the time taken by the carry to propagate through the adders which form the multiplication array. So, this is not an efficient algorithm for the multiplication of large numbers as a lot of propagation delay will be involved in such cases. To overcome this problem, Nikhilam Sutra will present an efficient method of multiplying two large numbers.

3.2. Nikhilam Sutra

Nikhilam Sutra means "all from 9 and last from 10". It is also applicable to all cases of multiplication; it is more efficient when the numbers involved are large. Since it find out the compliment of the large number from its nearest base to perform the multiplication operation on it. Larger the original number, lesser the complexity of the multiplication. We will illustrate this Sutra by considering the multiplication of two decimal numbers (96 × 93) where the chosen base is 100 which is nearest to and greater than both these two numbers. As shown in Fig. 5, we write the multiplier and the multiplicand in two rows followed by the differences of each of them from the chosen base, i.e., their compliments. We can write two columns of numbers, one consisting of the numbers to be multiplied (Column 1) and the other consisting of their compliments (Column 2). The product also consists of two parts which are distributed by a vertical line. The right hand side of the product will be obtained by simply multiplying the numbers of the Column 2 (7×4 = 28). The left hand side of the product will be found by cross subtracting the second number of Column 2 from the first number of Column 1 or vice versa, i.e., 96 - 7 = 89 or 93 - 4 = 89. The final result will be obtained by combining RHS and LHS (Answer = 8928).

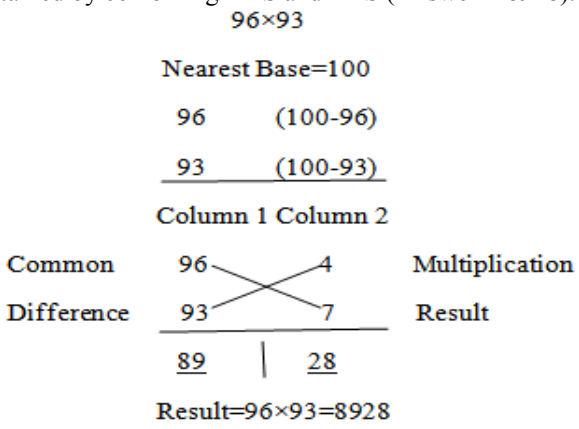


Fig. 3: Multiplication by Nikhilam Method.

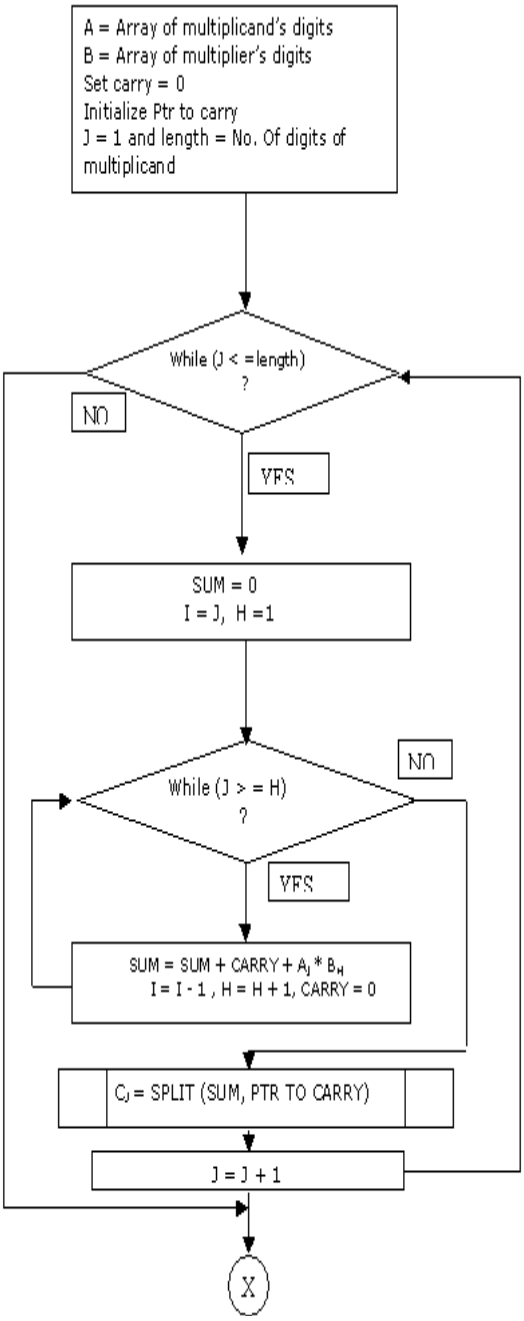
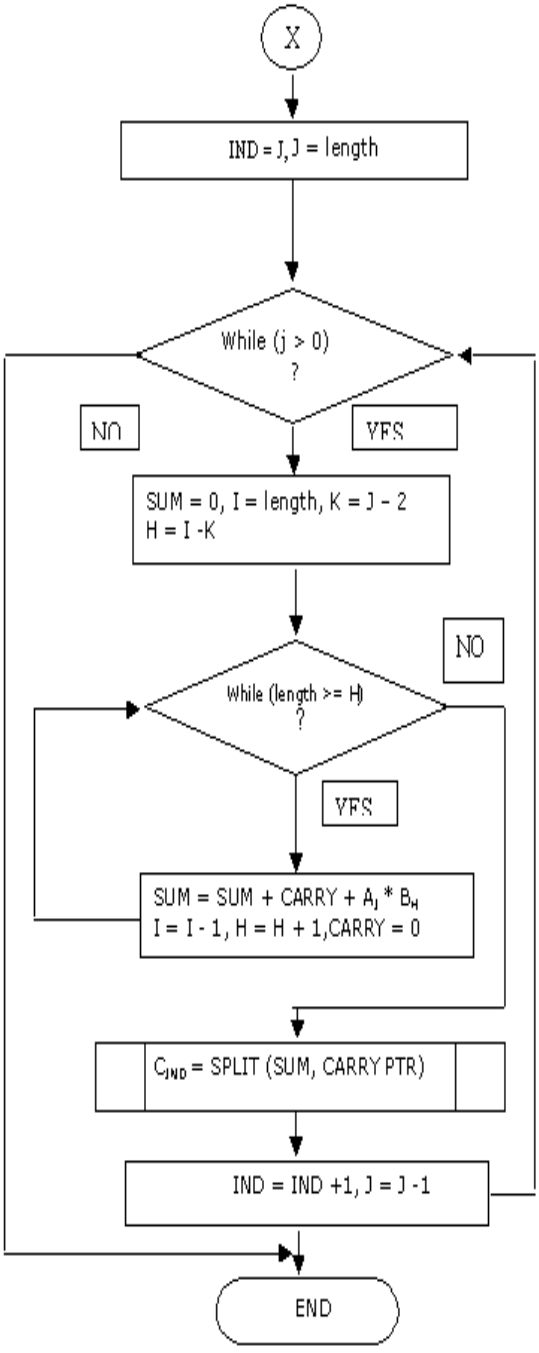


Fig.4: Flow Chart1 for implementing Vedic Multiplication in C language

IV. HARDWARE IMPLEMENTATION

This hardware design of the vedic multiplier using Urdhva tiryakbhyam Sutra is very similar to that of the famous array multiplier where an array of adders is required to arrive at the final product. All the partial products are calculated in parallel and the delay associated is mainly the time taken by the carry to propagate through the adders

Fig.5: Flow Chart2 for implementing Vedic Multiplication in C language



which form the multiplication array. The hardware can be implemented using Verilog HDL and simulation is done in Veriwell simulator and the powe dissipiation time delay tables are provided below.

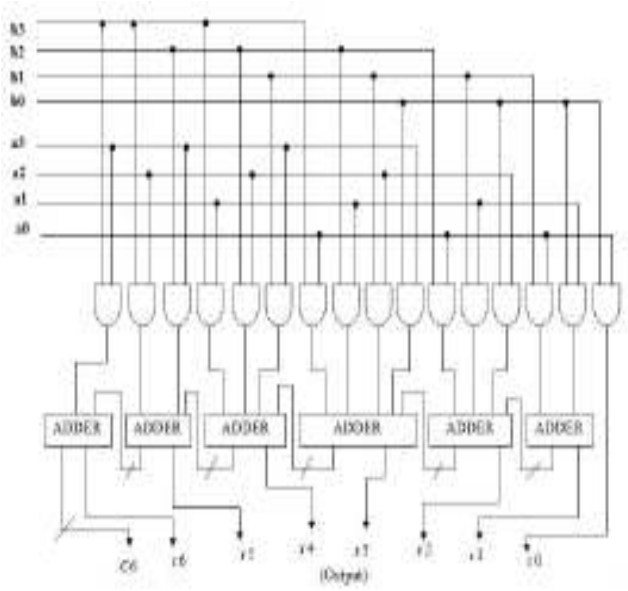


Figure 6: Hardware architecture of the Urdhava tiryakbhyam multiplier.Similarly hardware design for Nikhilam Sutra can also be designed and implemented using HDL.

V. Comparison and Discussion

FPGA implementation results show that multiplier Nikhilam Sutra based on of Vedic mathematics for multiplication of binary numbers is faster than other multipliers. Vedic Multiplier has the advantages as over other multipliers also for power and regularity of structures.

TABLE I: VEDIC AND CONVECTIONAL MULTIPLIER

Type of unit	Cell Area	Total Power (nW)	Dynamic Power (nW)	Leakage Power (nW)
8x8 Conventional Multiplier	4417	466014.411	465984.874	29.536
8x8 Vedic Multiplier	4664	405745.509	405715.454	30.055

TABLE II: CONVENTIONAL AND VEDIC ALU

Type of unit	Operating Freq (MHz)	Total Power (nW)	Cell Area	Timing Slack (ps)
8x8 ALU using conventional Multiplier	150	2004404.604	17676	+1
	300	4614805.708	18871	-1719
8x8 ALU using Vedic Multiplier	150	2858470.659	16881	+2570
	300	4435907.151	17896	+3

There are number of techniques for logic implementation at circuit level that improves the power dissipation, area and delay parameters in VLSI design. Implementation of parallel Multiplier in CPL logic shows significant improvement in power dissipation. CPL requires more number of transistors to implement as compared to the CMOS and provides only a little improvement in speed. Pass Transistor Logic which offers better performance over both the CMOS and CPL in terms of delay, power, speed and transistor count. The PTL outperforms the CMOS implementation in speed and great in power dissipation, with approximately same transistor count. When compared to CPL, PTL is faster and shows improvement in power and transistor count.

TABLE III: Comparison between multiplier designs in three different Logics

Parameters	CMOS	Complementary Pass-Transistor Logic	Pass-Transistor Logic
Number of Transistor	Most	More	Less
Area	Maximum	Medium	Minimum
Power	Most	More	Less
Delay	Most	More	Less
Speed	Less	Medium	High

VI. CONCLUSION

Ancient Indian system of mathematics, known as Vedic mathematics, can be applied to various branches of engineering to have a deeper insight into the working of various formulae. The algorithms based on conventional mathematics can be simplified and even optimized by the use of Vedic Sutras. We have discussed one such possible application of Vedic mathematics to digital signal processing. A simple Vedic multiplier architecture based on the Urdhva Tiryakbhyam (Vertically and Cross wise) Sutra of Vedic mathematics has been presented. The hardware architecture of the Vedic multiplier is also depicted and is found to be very similar to that of the so called array multiplier. This is just one of the many possible applications of the Vedic Mathematics to Engineering and some serious efforts are required to fully utilize the potential of this interesting field for the advancement of Engineering and Technology. Although, Vedic mathematics provides many interesting Sutras, but their application to the field of engineering is not yet fully studied. Knowingly or unknowingly we always use Vedic Sutras in everyday world of technology. As an example, whenever we use i<-- i+1 or i<--i−1 in software routines, we use 'Ekadhikina Purvena' and 'Ekanyunena Purvena' Sutras respectively. Similarly the Vedic algorithms can be directly applied to solve many problems of trigonometry, plain and spherical geometry, conics, calculus (both differential and integral), and applied mathematics of various kinds. Hence, the vast potential of this interesting field should be exploited to solve the real world problems efficiently.

REFERENCES

[1] Swami Bharati Krishna Tirthaji , Vedic Mathematics. Delhi: Motilal Banarsidass Publishers, 1965.

[2] Vedic Mathematics [Online]. Available: http://www.hinduism.co.za/vedic.htm.

[3] P. D. Chidgupkar and M. T. Karad, "The Implementation of Vedic Algorithms in Digital Signal Processing", Global J. of Engg. Edu., Vol.8, No. 2, pp. 153-158, 2004.

[4] M. Ramalatha, K. Deena Dayalan, P. Dharani, " High Speed Energy Efficient ALU Design using Vedic Multiplication Techniques" ACTEA, IEEE pp 600-603.

[5] Shamim Akhter,"VHDL Imp lementation Of Fast NXN Multiplier Based On Vedic Mathematics", Jaypee Institute of Information Technology University, Noida, 201307 UP, INDIA, 2007 IEEE.

[6] Harp reet Singh Dhillon and Abhijit Mitra, "A Reduced- Bit Multip lication Algorithm for Digital Arithmetics", International Journal of Computational and Mathematical Sciences 2;2 © www.waset.org Spring 2008.

[7] E. Abu-Shama, M. B. Maaz, M. A. Bayoumi, "A Fast and Low Power Multiplier Architecture", The Center for Advanced Computer Studies, The University of Southwestern Louisiana Lafayette, LA 70504.

[8] "Low power and high speed 8x8 bit Multiplier Using Non-clocked Pass Transistor Logic" C.Senthilpari, Ajay Kumar Singh and K. Diwadkar, 1-4244-1355-9/07, 2007, IEEE.

[9] Morris Mano, "Computer System Architecture",PP. 346-347, 3rd edition,PHI. 1993.

[10] P. Mehta, D. Gawali, "Conventional versus Vedic Mathematical Method for Hardware Implementation of a Multiplier," Proceeding International Conference on Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT '09, pp-640-642.

AUTHORS

First Author – V.Vamshi Krishna, Department of ECE, JNTUH College of Engineering, Karimnagar, India

V.Vamshi Krishna is pursuing his B.Tech degree in Electronics & Communication from JNTUH College of Engineering Karimnagar, India. His research intrests are low power VLSI

design and DSP. Presented various research papers in national conferences.

Second Author – S. Naveen Kumar, Department of ECE, JNTUH College of Engineering, Karimnagar, India

S.Naveen Kumar is pursuing his B.Tech degree in Electronics & Communication from JNTUH College of Engineering Karimnagar, India. His research intrests are VLSI design.