Volume 3, Issue 12, December 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Review Paper on High Speed Karatsuba Multiplier and Vedic Mathematics Techniques

Mr. Dharmendra Madke
M.Tech Scholar
(EC DEPARTMENT) P.C.S.T,(R.G.P.V).
Bhopal (M.P.), India

Assoc.Prof. Sameena Zafar
Associate Professor
(EC DEPARTMENT)P.C.S.T,(R.G.P.V.)
Bhopal (M.P.), India

Abstract – This paper analysed the design of high speed Karatsuba Multiplier and Ancient Indian Vedic Multiplication techniques that have been time to time been modified as per the problem requirement to improve performance. Vedic Mathematics is the old formulas of the mathematics which has a unique technique of calculations based on 16 Sutras. Previous work has shown the efficiency of Urdhva Triyagbhyam – Vedic method for multiplication over other multiplication methods which strikes a difference in the actual process of multiplication itself. It generates parallel equations of intermediate products, which eliminates unwanted multiplication steps with zeros and scaled to higher bit levels using Karatsuba algorithm with the compatibility to different data types. Urdhva tiryakbhyam Sutra found to be one of the most efficient Sutra (Algorithm), giving minimum delay for multiplication of all types of numbers.

Keywords— Vedic multiplication, urdhva triyakbhyam sutra (algorithm), Nikhilam Sutra,karatsuba- ofman algorithm

I. INTRODUCTION

Multiplication is an important arithmetic operations which is used frequently in hardware level in digital filtering where currently implementations applied in many Digital Signal Processing (DSP) applications such as convolution, Fast Fourier Transform (FFT), filtering and in microprocessors in its arithmetic and logic unit [1]. Since multiplication dominates the execution time of most DSP algorithms and also they are required many in numbers than the other hardware component, so there is a need of high speed multiplier to increase the speed of the multiplier. Still, multiplication time of the hardware multiplier is the important factor in determining the instruction cycle time of a DSP chip. The demand for high speed DSP has been increasing as a result of expanding computer and signal processing applications. Arithmetic operations like multiplication are important to achieve the desired performance in many real-time digital signal and image processing applications [2]. The development of fast multiplier circuits has been a subject of interest from two decades. Reducing the time delay and power consumption are very essential requirements for any digital applications as they mostly works on battery [2, 3]. This analysis gives different Vedic multiplier architectures with karatsuba algorithm. Multipliers based on Vedic Multiplication are one of the fast and low power multipliers. Minimizing power consumption for digital systems involves optimization at all levels of the design and in Vedic multiplication this is achieved due to less steps to solve the multiplication than the traditional multiplication. This optimization includes the technology used to implement the digital circuits requirements are circuit style, topology, and the architecture for implementing the circuits and at the highest level the algorithms. In Digital designs multipliers are the most commonly used components. They are fast, mostly used and efficient components that are utilized to implement many operations. Depending upon the algorithms of the components, there are so many types of multipliers available. Particular multiplier architecture is selected based on the desired application. In many DSP algorithms, the multipliers are in the critical delay path and ultimately determine the overall algorithm performance. The speed of multiplication operation is of great importance in DSP as well as in general processor.

The 16 Vedic multiplication Sutras along with their brief meanings are enlisted below alphabetically.
1. Ekadhikina Purvena – In this method we have to find one more than the previous sequence.
2. Ekanyunena Purvena – In this method we have to find one less than the previous sequence.
3. (Anurupye) Shunyamanyat – If one multiplicand is in the ratio, the other is zero.
4. Chalana-Kalanabyham – multiplication is found by the Differences and Similarities between multiplier and multiplicand.
5. Gunakasamuchyah – in these method factors of the sum is equal to the sum of the factors.
6. Gunitasamuchyah – in this method product of the sum is equal to the sum of the product.
7. Paraavartya Yojayet – multiplication is found by the Transpose and adjusts.
8. Puranapuranabyham – multiplication is found by the completion or noncompletion.
9. Nikhilam Navatashcaramam Dashatah – – in this method product of all from 9 and last from 10.
10. Sankalana- vyavakalanabhyam – multiplication is found by the addition and by subtraction.

11. Sopaantyadvayamantyam – multiplication is found by the ultimate and twice the penultimate.
12. Urdhva-tiryagbhyam – multiplication is found by the vertically and crosswise.
13. Shesanyankena Charamena – multiplication is found by the remainders by the last digit.
14. Shunyam Saamyasamuccaye – When the sum is the same that sum is zero.
15. Vyashtisamanstih – multiplication is found by the Part and Whole.
16. Yaavadunam – multiplication is found by whatever the extent of its deficiency.

These sutras can be used in various trigonometric as well as the geometric problems in mathematics effectively. These Sutras were reconstructed from ancient Vedic texts. Many Sub-sutras were also discovered till now which gives its distinctive advantages, which are not discussed here. The advantage of Vedic mathematics lies in the fact that it reduces the otherwise complex calculations in conventional mathematics to a very simple one. This is so because the Vedic sutras are claimed to be based on the natural principles on which the human brain works. This is a very remarkable field and presents some effective algorithms which can be applied to various branches of engineering such as computing, digital signal processing and digital image processing [1,4]. The multiplier architectures can be broadly classified into three categories. First one is the serial multiplier which emphasizes on hardware and minimum amount of chip area. Second is parallel multiplier (array and tree) which carries out high speed mathematical operations where the drawback is the relatively larger chip area utilization. Third one is serial- parallel multiplier which serves as a good trade-off between the times consuming serial multiplier and the area consuming parallel multipliers.

II DESIGN OF VEDIC MULTIPLIER

1. Urdhva tiryakbhyam

The most prominently used sutras of the 16 mention sutras are Urdhva tiryakbhyam Sutra which literally means "Vertically and crosswise". To demonstrate this multiplication scheme, let us consider the multiplication of two decimal numbers (5498 × 2314). The conventional method will require 16 multiplications and 15 additions. Multiplication using Urdhva tiryakbhyam Sutra is shown in Fig. 1. The numbers to be multiplied are written on two consecutive sides as shown in the figure. The square is divided into rows and columns where each row/column corresponds to one of the digit of either a multiplier or a Multiplicand. Where, each digit of the multiplier has a small box common to a digit of the multiplicand. These small boxes are separated into two halves by the crosswise lines. Each digit of the multiplier is then autonomously multiplied with every digit of the multiplicand and the two-digit product is written in the common box. All the digits on a crosswise dotted line are added to the subsequent carry. The least significant digit of the obtained number acts as the result digit and the rest as the carry for the next step. Carry for the first step (i.e., the dotted line on the extreme right side) is taken to be zero [9]

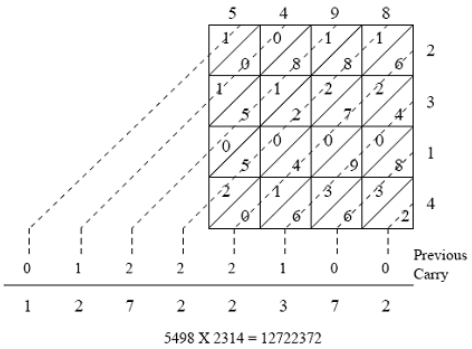

Figure 1: multiplication by Urdhva tiryakbhyam Sutra.

The design starts first with Multiplier design that is 2x2 bit multiplier as shown in figure 2.

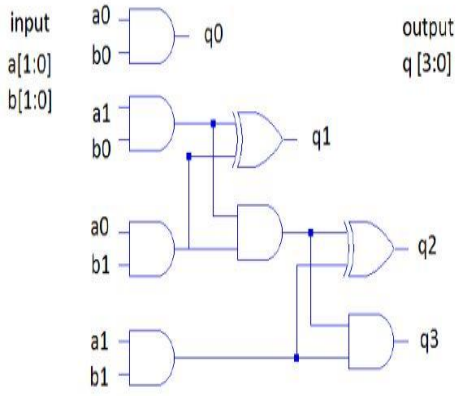

Figure 2: Hardware Realization of 2x2 blocks

Here, "Urdhva Tiryakbhyam Sutra" or "Vertically and Crosswise Algorithm" [4] for multiplication has been effectively used to develop digital multiplier architecture. This algorithm is quite different from the traditional method of multiplication, which is to add and shift the partial products. To scale the multiplier further, Karatsuba – Ofman algorithm can be employed [6]. Karatsuba-Ofman algorithm is considered as one of the fastest ways to multiply long integers. It is based on the divide and conquers strategy [11].

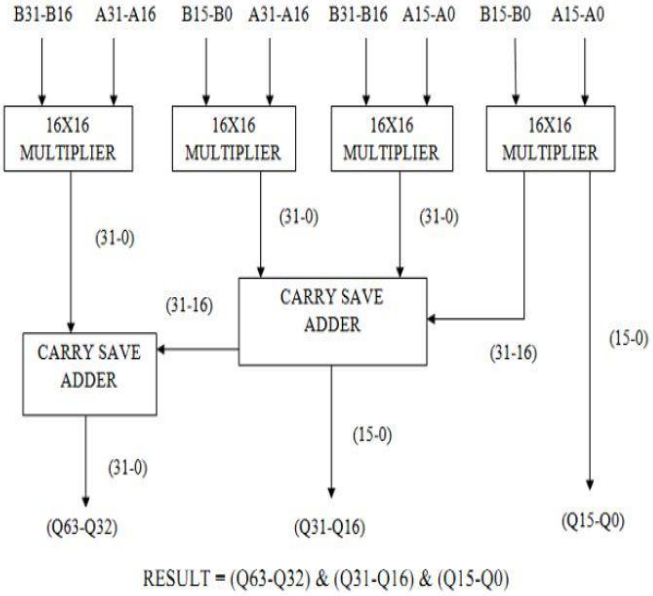

Figure 3: Vedic Multiplier hardware model.

The Vedic multiplication hardware model basic structure is shown in figure 3.The below figure 4 show the line diagram of Urdhva tiryakbhyam.This is the another method of Urdhva tiryakbhyam to perform multiplication. This method is mostly use in most of the implementation of processors.

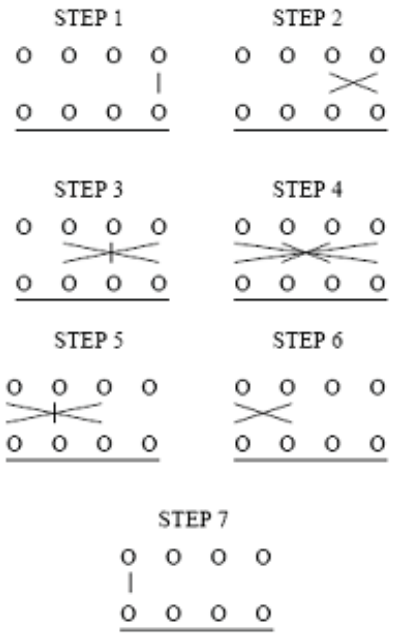

Figure 4.Urdhava Line diagrame

Here we can see that urdhava method done multiplication in single shift which increase the speed of processors [17]. This algorithm is not efficient for large number because of lot of propagation delay is involved. In order to deal with this propagation delay problem Nikhilam Sutra is present which is efficient method for large number multiplication s[18].

2. NIKHILAM SUTRA

Nikhilam Sutra stands for "all from 9 and last from 10"[18].Basically this method is use in all type of multiplication, but most efficiently use in large number multiplications. This method take a nearest base of number, larger the number

lesser the complexity in multiplication[18].Lets take an example of 96x 93 where we take base 100 which is near to number and grater to number as shown below fig.5

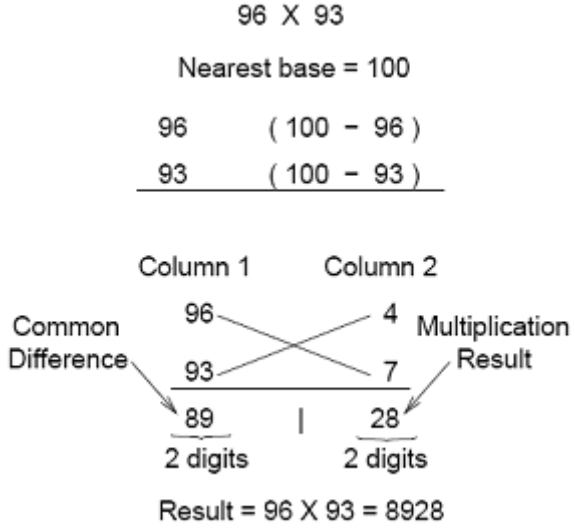

Figure5. Nikhiam Line Diagram of 4 bit number

From above we can see that we take base as 100 which is near to 96 as well as 93, here we subtract number with base. After subtracting number from base we get 4 and 7.we place this number on left hand side we see on column 2,Know multiply this number we get an 28.In column 1 we placed a numbers ,for LSB we have to subtract number with their opposite number remainder which obtain during subtraction. Let's see 96-7=89 or 93-4=89.Know merges this answer with the answer that obtains during first multiplication. from the above we can see that this method required only two multiplication steps as well as less complexity. Nikhilam power will make pupils play with power of base like 10,100...etc

III. THE KARATSUBA MULTIPLICATION

If we want to multiply large numbers, it's possible to use other techniques than the traditional multiplication, which is time expensive. In this case the Karatsuba Multiplication Algorithm [8], [9] is one of the choices. But this method is one of the fastest as well as mostly uses method in processors, because this method required only three multiplications and most of time this method is recursive in nature.For convenience, we will focus on long integers in binary representation [9]. If we have the numbers

$$U = (u_{2n-1}...u_1u_0)_2 \text{ and } V = (v_{2n-1}...v_1v_0)_2 \quad (1)$$

Represented on 2n bits, we can write then

$$U = 2^nU_1 + U_0 \text{ and } V = 2^nV_1 + V_0 \text{ , where,}$$

$$U_1 = (u_{2n-1}...u_n),\ U_0 = (u_{n-1}...u_1u_0),$$
$$V_1 = (v_{2n-1}...v_n),\ V_0 = (v_{n-1}...v_1v_0).$$

Now, we have:

$$UV = (2^nU_1 + U_0)(2^nV_1 + V_0) =$$
$$= 2^{2n}U_1V_1 + 2n(U_1V_0 + U_0V_1) + U_0V_0 =$$
$$= 2^{2n}U_1V_1 + 2^n((U_1 + U_0)(V_1 + V_0) - U_1V_1 - U_0V_0) + U_0V_0 \quad (2)$$

This formula reduce the original 2n bits operands multiplication to three n bits operands multiplication: U_1V_1, $(U_1 + U_0)(V_1 + V_0)$ and U_0V_0, and few simple operations (like shift and addition). The main advantage of using the formula (2) is that we can define a recursive process for multiplication, which is faster than the traditional multiplication. The traditional multiplication has the complexity $O(n^2)$ [9]. Proceeding recursively we obtain the bit complexity $O(n^{\log 3})$, where

$$\log 3 = 1.58... < 2. \text{ [10]}$$

Lets take two numbers Xa Xb and Ya Yb Then

A=Xb. Yb
B=Xa Ya
C=(Xa+Xb)(Ya+Yb)-A-B

From above it is clear that we required three multiplication and thus increase the speed of multiplication.It is similar to Urdhva Tiryakbhyam, but it reduce multiplication steps because A and B multiplication is already performed and we have to take value in to C as an input.When this technique is recursively applied to ultidigit numbers, a point is reached in the recursion when the overhead of addition and subtraction makes it more efficient to use the usual O (n^2) multiplication algorithm to evaluate the partial products. We call this point "Karatsuba threshold". The most efficient overall method therefore relies on a combination of Karatsuba and conventional multiplication [6]. Namely, we'll have $n^{\log 3/\log 2}$ digit products for operands of length n, not n^2 like in the traditional multiplication. (See [9], p. 233)It's convenient to see this algorithm in terms of a ternary tree. Each node has three children that compute the partial products and at each level the input length is divided with two. The leaves perform the classical multiplication. The threshold for switching to the traditional multiplication method is determined experimentally [3].

IV. CONCLUSION

This is found out that Karatsuba Algorithm and Nikhilam Sutra is giving the most efficient multiplier output than the other sutras in terms of speed, space and less power consumption. As the need of low power devices is mounting in digital world due to battery operation and device's operational longevity is the main concern these multipliers are the main focuses of researches than the conventional multipliers.Karatsuba along with the Vedic shows it even more efficient type of algorithm than its individual approaches where speed of throughput is drastically improved over other methods.

REFERENCES

[1] Wallace, C.S., "A suggestion for a fast multiplier," IEEE Trans. Elec. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.
[2] Booth, A.D., "A signed binary multiplication technique," Quarterly Journal of Mechanics and Applied Mathematics, vol. 4, pt. 2, pp. 236– 240, 1951.
[3] Jagadguru Swami Sri Bharath, Krsna Tirathji, "Vedic Mathematics or Sixteen Simple Sutras from the Vedas", Motilal Banarsidas, Varanasi (India), 1986.
[4] A.P. Nicholas, K.R Williams, J. Pickles, "Application of Urdhava Sutra", Spiritual Study Group, Roorkee (India), 1984.
[5] Neil H.E Weste, David Harris, Ayan anerjee,"CMOS VLSI Design, A Circuits and Systems Perspective", Third Edition, Published by Person Education, PP-327-328]
[6] Mrs. M. Ramalatha, Prof. D. Sridharan, "VLSI Based High Speed Karatsuba Multiplier for Cryptographic Applications Using Vedic Mathematics", IJSCI, 2007
[7] Thapliyal H. and Srinivas M.B. "High Speed Efficient N x N Bit Parallel Hierarchical Overlay Multiplier Architecture Based on Ancient Indian Vedic Mathematics", Transactions on Engineering, Computing and Technology, 2004, Vol.2.
[8] Knuth, D. E. (1981). The art of computer programming, volume 2. Addison-Wesley, 2 editions, 1981.
[9] Karatsuba, A.; Ofman, Yu. (1962). Multiplication of multidigit numbers on automata. *Sov. Phys. Dokl.*, 7:595-597.
[10] D. Zuras, On squaring and multiplying large integers, In Proceedings of International Symposium on Computer Arithmetic, IEEE Computer Society Press, pp. 260-271, 1993.
[11] Shripad Kulkarni, "Discrete Fourier Transform (DFT) by using Vedic Mathematics"Papers on implementation of DSP algorithms/VLSI structures using Vedic Mathematics, 2006, www.edaindia.com, IC Design portal.
[12] S.G. Dani, Vedic Maths': facts and myths, One India One People, Vol 4/6,January 2001, pp. 20-21; (available on www.math.tifr.res.in/ dani).
[13] M.C. Hanumantharaju, H. Jayalaxmi, R.K. Renuka, M. Ravishankar, "A High Speed Block Convolution Using Ancient Indian Vedic Mathematics," ICCIMA, vol. 2, pp.169-173, International Conference on Computational Intelligence and Multimedia Applications, 2007.
[14] Himanshu Thapliyal, "Vedic Mathematics for Faster Mental Calculations and High Speed VLSI Arithmetic", Invited talk at IEEE Computer Society Student Chapter, University of South Florida, Tampa, FL, Nov 14 2008
[15] Jeganathan Sriskandarajah, "Secrets of Ancient Maths: Vedic Mathematics", Journal of Indic Studies Foundation, California, pages 15 and 16.
[16] S. Kumaravel, Ramalatha Marimuthu, "VLSI Implementation of High Performance RSA Algorithm Using Vedic Mathematics," ICCIMA, vol. 4, pp.126-128, International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), 2007.
[17] Vaijyanath Kunchigi, Linganagouda Kulkarni, Subhash Kulkarni, "High Speed and Area Efficient Vedic Multiplier,"in IEEE nternational Conference on Devices, Circuits and Systems (ICDCS), 2012.
[18] Honey Durga Tiwari, Ganzorig Gankhuyag, Chan Mo Kim, Yong Beom Cho, "Multiplier design based on ancient Indian Vedic Mathematics," in IEEE International SoC Design Conference, pp. II-65 - II-68, November 2008.

