ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

VEDIC MATHEMATICS FOR VLSI DESIGN:A REVIEW

Anuradha Singh*, Neetu Gupta
* School of Engineering Sharda University, Greator Noida India
School of Engineering The North Cap University, Gurugram India

DOI: 10.5281/zenodo.400813

ABSTRACT

Multiplication and division are the most critical arithmetic operation carried out in any digital logic algorithm such as digital signal processing, in cryptography for encryption and decryption algorithm, ALU design and in other logic computation. Thrust in higher processing speed and low power has not only tendered the need for improvements in processor technologies but also in exploring new algorithms. Vedic Mathematics' potential can be unleashed in an effective way to design and implement the algorithm for multiplication and division. Through this paper Vedic mathematics application in VLSI design is reviewed and it is found that Vedic sutras based algorithms in digital logic design has resulted in simpler architecture, better speed and higher power efficiency.

KEYWORDS: Vedic Mathemetic; Vedic Multiplier; Vedic Divisor ; cryptography; convolution.

INTRODUCTION

With development in technology, it has become the need to have a multiplier and divisor that should be robust in design and logic yet ensuring speed and accuracy that accomplish the demands of VLSI technology. Conventional method used for multiplication operation such as array method, Wallace tree or booths etc are not very efficient in speed and power consumption. This paper presents the various Vedic multiplication techniques like Nikhilam, Urdhva Tiryakbhayam and Anurupye for multiplication that are not only fast and accurate as compared to traditional methods but also provide the benefit of regularity in layout and hence the less area consumption[29]. It has been found that Urdhva Tiryakbhayam Sutra is most efficient Sutra (algorithm) which gives minimum delay for multiplication of all types of numbers [28].

Division is the expensive and most time consuming operation of the modern processors amongst all elementary operations [24]. Vedic Mathematics proposed various sutras [1] that can be applied for binary division. Paravartya, Nikhilam and Dhvajanka (which means on the top of the flag) are explained in this paper and their various applications are discussed. The study shows that it is time efficient and well optimized method as compared to conventional one.

This paper is organized is as, Section II describes the Vedic mathematics and different sutras for multiplication and division. VLSI design implementation of Vedic Sutras is presented in section III. Conclusion and scope for future work is offered in Section IV.

VEDIC MATHEMATICS

According to *Carl Friedrich Gauss*, "Mathematics is the Queen of the Sciences". Mathematics is the origin and forms the basis of all sciences. The Vedic mathematics is derived from the Veda which provides one line and superfast solution to any problem along with quick cross checking mechanism [1]. The Veda is a Sanskrit word derived from the root Vid, which means to know without limit and it covers all Veda-sakhas known to humanity. The Veda is a depository of all knowledge, fathomless, informative as it is delved deeper. Sri Bharati Krishna Tirthaji reintroduced the concept of Vedic mathematics to this world. After extensive study of Atharva Veda, he constructed sixteen simple mathematical formulae called sutras and thirteen Up-sutras or corollaries [1]. The system of sutras is very simple and powerful as it covers almost every branch of mathematics [27].It looks like magic as one line faster solution to arithmetic problems but the application of the Sutras is perfectly logical and rational. All elementary operations the addition, subtraction, multiplication and division for any processor can be

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

performed efficiently by the sixteen sutras. The sutras mostly used in multiplication and division can be described as:

Multiplication Technique

Urdhva Tiryakbhayam

Urdhva Tiryakbhayam means vertically and crosswise is the most commonly used method. Urdhva Tiryakbhayam having universal application for all cases and can be used for decimal numbers as well as binary number multiplication [25]. Fig. 1 illustrates the multiplication steps in Urdhva Tiryakbhayam sutra for two decimal numbers illustrating the technique.

Multiplication is performed to the numbers at the end of line and if previous carry exists it is also added. More than one multiplication of any step are also added together with previous carry also. Result bit is defined as unit place digit and tens place digit is termed as carry for the next step. Because of the partial product and sums being calculated in parallel, clock frequency of processor does not affect multiplier [40]. Therefore microprocessors need not to operate on increasingly high frequency and thus optimizes the processing power. Due to its regular structure, it can be easily layout in silicon chip. Thus, it is time, power and space efficient technique as compared to conventional method of multiplication.

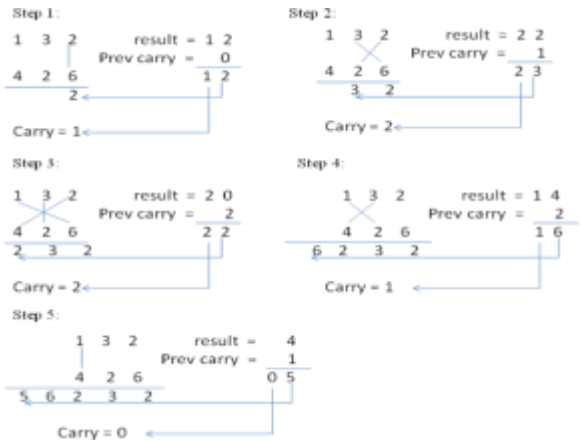


Fig 1: Multiplication of two decimal numbers using Urdhva Tiryakbhayam sutra

Nikhilam Sutra

The Nikhilam Sutra simply means “all from 9 and the last from 10”. The definition itself describes that it is beneficial for multiplication of numbers which are closer to base 10 i.e. to the powers of 10[17]. It is space and time efficient. Consider an example of 96*93. Here the base should be 100 which are near to numbers. Fig. 2 shows the multiplication steps in Nikhilam sutra.

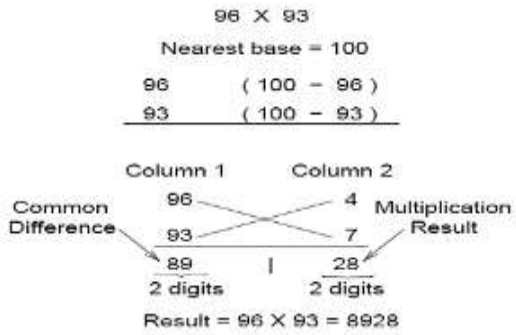


Fig 2: Multiplication using Nikhilam sutra

First, the two numbers are subtracted from the base i.e. the deviation from the base is found and are written in column 1. The result will have two parts. R.H.S. of the answer is the product of the deviations of the numbers. Cross deviation of column 2 from the original number in the first column produces L.H.S of the answer. i.e. (96

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

– 7) or (93 – 4). This requires only two multiplication steps and is less complex [29]. The Nikhilam sutra is efficient when the large numbers are involved in multiplication.

Karatsuba Ofman

Karatsuba Ofman algorithm is adequate to be used with very large numbers [15]. It is based on the divide and conquer rule. It is come out as the fastest technique for multiply larger numbers and is mostly used in processors [18]. If we have to multiply two n digit numbers then it will reduce three n/2 digit multiplications and few shifts and addition [19]. If x and y be considered as n digit strings for any base B, and integer m ,the two numbers x and y may be defined as

$$x = x_1 B^m + x_0$$
$$y = y_1 B^m + y_0$$

Where m < n,

Where x_0 , y_0 < B^m. Substituting x and y,

$$xy = (x_1 B^m + x_0)(y_1 B^m + y_0)$$
$$xy = z_2 B^{2m} + z_1 B^m + z_0)$$

Where

$$z_2 = x_1 y_1$$
$$z_1 = x_1 y_0 + x_0 y_1$$
$$z_0 = x_0 y_0$$
$$z_1 = (x_1 + x_0)(y_1 + y_0) - z_2 - z_0$$

By rearranging z_0, z_1, z_2 it can be concluded that xy can be calculated with three multiplications as compared to four multiplications in conventional method. The impact of reduction in multiplication is for two n-digit numbers as $n^{\log_2 3} \approx n^{1.585}$ single-digit multiplications and therefore it is faster to the conventional algorithm [19].

Division Technique

Nikhilam Sutra

Nikhilam Sutra can be applied more efficiently when divisor is closer and slightly smaller than the power of 10 [31].

Step 1: In this sutra, the dividend digits are rearranged in to two parts: Quotients and Remainder such that the digits in remainder are equal to the digits in divisor.

Dividend : 3483
Divisor : 99

	Quotient	Remainder
Dividend	34	83

Step 2: Similar to Nikhilam process divisor has to be complemented i.e. subtract all digits by 9 and the last digit from 10. This new divisor is called deficit.

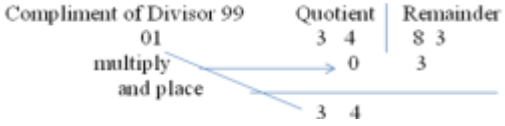
Compliment of Divisor 99	Quotient	Remainder
01	34	83

Step 3: Take first digit down as it is.

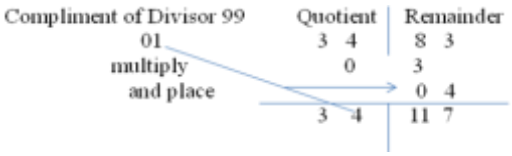
Compliment of Divisor 99	Quotient	Remainder
01	34	83
	3	

Step 4: Deficit is multiplied with first digit ,shifted one place to right and then to be written below the dividend. Now add the first column.

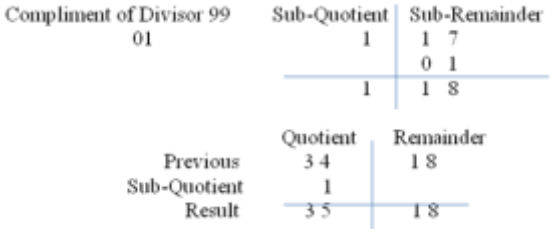
ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7



Step 5: The above step is repeated and added column wise till the number is filled in last column and then added column wise.



Step 6: In few cases it may be possible that remainder comes out to be greater than divisor as shown above which is not possible. In this case divide the remainder with the divisor that results in sub-quotient and sub-remainder. Add sub-quotient with original quotient and sub-remainder becomes the final remainder.



Here it is observed that no subtraction step has been performed. The division operation has been done with multiplication and addition steps. This results in faster and less complex operation. The Simplicity of this sutra is that the largest multiplication required is 9 by 9.

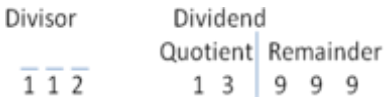
Paravartaya Sutra

If Divisor is more than power of 10 (10, 100, 1000, etc) and starts with 1 then the sign is changed and the changed number is applied.

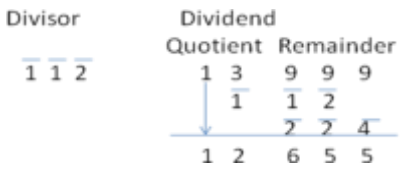
The Remainder Theorem describes the sutra[30].The division with Paravartya sutra is carried out as follows:

Step 1: First digit of the divisor is left out and transpose of remaining digits is taken.

Step 2: Dividend is splitted in two parts Quotient and remainder maintaining remainder with same number of digits as divisor acquired in step 1.



Step 3: Carry the same procedure as done in Nikhilam Sutra.



Step 4: If the result carry any bar digit or complimented digit, it is to be converted to normal digit using vinculum. Paravartya sutra produces same efficient results whether applied to large or small divisors [27] [31].

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

Dhvajanka Sutra

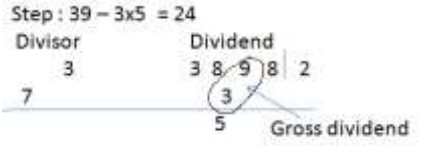
Dhvajanka Sutra is a direct Flag method [27] that is used to carry out the division of any types of numbers [35]. Division by this method can be summarized as follows:

Example: Divisor: 73 and Dividend: 38982

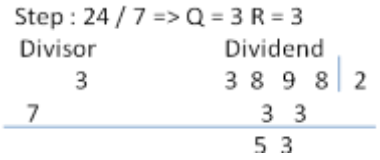
Step 1: Split the divisor in two parts: Leftmost digit is left aside which forms the new divisor and the division will be carried out with this only and second part is called the flag. Write the divisor down and flag on top. Split the dividend in to two with right part consisting same numbers of digits as in divisor.

Step 2: New divisor divides the first digit of dividend, quotient and remainder of this step are written down.

Step 3: In next iteration the remainder from previous step and next dividend digit is used as gross dividend now. Multiplication is performed between quotient from previous step and the flag digit and then subtraction is performed from the gross dividend.



Step 4: The number left after subtraction is now divided by new divisor. The Quotient is written down and remainder is prefixed with next digit of dividend to form new gross dividend.



Step 5: Process is repeated till the last digit of the dividend. Final quotient will have the same number of digits as the left part of the dividend. Final remainder is obtained by subtraction of right part of dividend prefixed by last remainder and cross multiplication of quotient and divisor.

VLSI DESIGN USING VEDIC MATHEMATICS

32×32 Bit Vedic Multiplier

Neeraj Kumar and Subodh Wairya proposed a 32x32 bit multiplier based on Urdhva Tiryakbhayam sutra. Decomposition of numbers to multiplication modules is performed by grouping multiplier and multiplicand each. For two 32-bit numbers, grouping is performed to decompose into 16×16 multiplication modules. It is also illustrated that the further hierarchical decomposition of 8×8 modules into 4×4 modules and then 2×2 modules will have a significant VHDL coding of for 32x32 bits multiplication. The algorithm for two bit number can be described as shown in fig 3.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

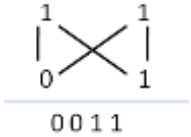


Fig3: Two Bit number Vedic multiplication

By decomposing the multiplier and multiplicand this algorithm may be extended for higher bit numbers as well. The logic is extended such that for 4 bit multiplication the bits are grouped in two bits than the two bit multiplication is applied and grouped for 4 bits. Similarly higher bits multiplication is arranged and calculated.

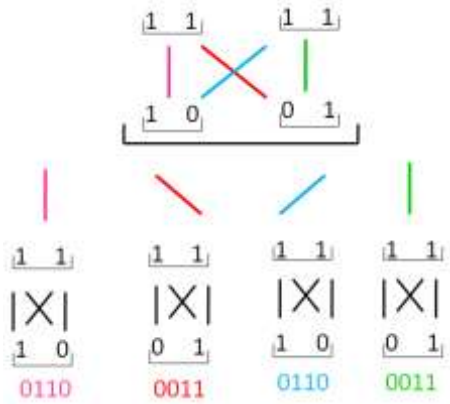


Fig4: Four bit number Vedic Multiplication

They found that the proposed Vedic multiplier proves to be highly efficient in terms of the speed. The increase in delay is very slow with respect to increase in the input bits. Higher speed and greater power efficiency is obtained through this multiplier algorithm.

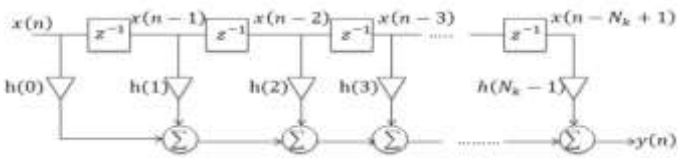
Convolution using Vedic Mathematics

Karishma P Dighorikar and S. L. Haridas, in their paper "Area efficient architecture for convolution using Vedic Mathematics", International Journal of Science and Research ISSN (online):2319-7064, present a direct method of reducing convolution processing time using hardware computing and implementation of discrete linear convolution of two finite length sequences (NxN). Convolution [7] is the mathematical way that relates the output, y (t), of a linear, time invariant system to its input, x (t) and the impulse response of the system, h (t) [8].

$$y(n) = \sum_{k=-\infty}^{\infty} \mathrm{x(k)h(n-k)}$$

The above equation can be expanded and can be written as

$$y(n) = h(0).x(n) + h(1).x(n-1) + \cdots + h(N_k - 1).x(n - N_k + 1)$$



They found that using Vedic multiplier employed using Urdhva sutra [9] instead of conventional multiplication method improves the speed and complexity. Then they redesign the original design with reduced number of multiplier and adders required using cyclic data flow and verify that this optimizes the speed and area. Also they found that reorganizing design with pipeline structure further improve the system performance [6].

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

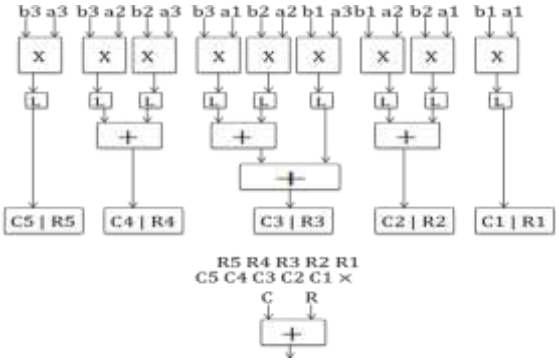


Fig 5: Binary multiplication using Urdhva Tiryakbhayam sutra [9]

Reduction in the number of multiplier in design significantly improves power and area efficiency.

High Speed DSP Algorithm

Leonard Gibson Moses S and Thilagar M in their paper "VLSI Implementation of High Speed DSP algorithms using Vedic Mathematics", Singaporean Journal Scientific Research (SJSR) ISSN: 2231 - 0061Vol.3, have found that Urdhva Tiryakbhayam algorithm is the most efficient multiplication algorithm and is applicable to all types of numbers. They have designed an NxN multiplier using Urdhva Tiryakbhayam sutra and then used this multiplier in various DSP algorithms to implement the cube algorithm, linear convolution [10] and single precision floating point multiplication [12]. Single precision floating point multiplication design is explained in Figure 7.

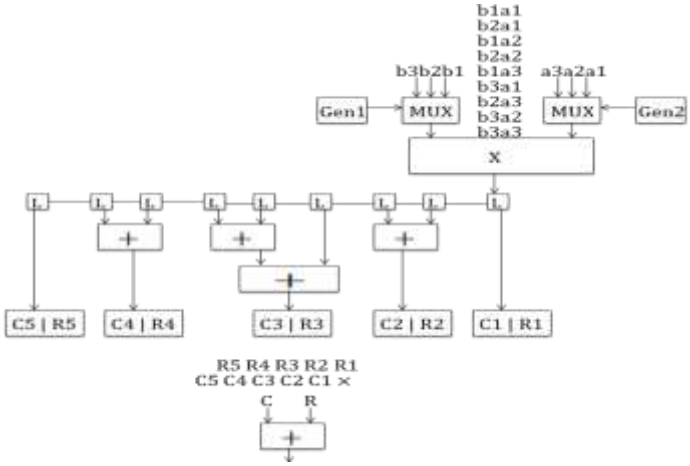


Fig 6: Binary multiplier design having only one number Multiplier

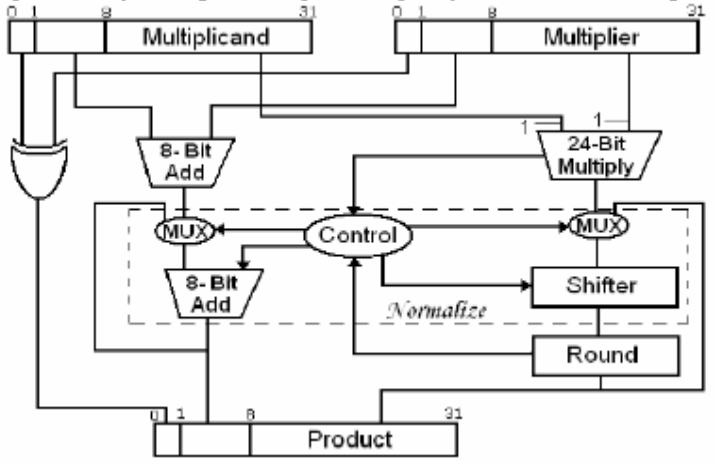


Fig7: Design of single precision floating point multiplier

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

The algorithm consists of three parts each corresponding to the different parts of the single precision format. Exclusive OR function of the two input signs determines the sign and forms the first part of the product. The exponent of the product is calculated by adding the two input exponents. The third part which is the significand of the product is determined by multiplying the two input significands each with a "1" concatenated to it. This shows that the application of Vedic mathematics reduce the efforts and time required to compute any algorithm thus speed up the operation. Also reduces the area by using less number of logic gates.

High Speed Karatsuba Multiplier

Dharmendra Madke and Assoc.Prof. Sameena Zafar in their Review Paper on High Speed Karatsuba Multiplier and Vedic Mathematics Techniques, IJARCSSE, Vol3, Issue 12, December 2012, analyzed the design of high speed Karatsuba Multiplier and Ancient Indian Vedic Multiplication techniques that improves the overall performance [13]. They utilizes the fact that in Urdhva Tiryakbhayam algorithm intermediate products are generated in parallel, which reduce unwanted multiplication steps with zeros and then the multiplier is scaled to higher bit levels using Karatsuba algorithm with the compatibility to different data types. Karatsuba-Ofman algorithm is one of the fastest methods to multiply long integers and works on the principle of divide and conquer [14]. The Vedic multiplication hardware model basic structure based on Urdhva Tiryakbhayam algorithm is shown in figure 8. As the multiplication is performed in single shift the speed of processors is increased significantly [17]. But this algorithm is not efficient for large number because of lot of propagation delay is involved. In order to overcome this problem Nikhilam Sutra can be used which is efficient method for large number multiplications [3].

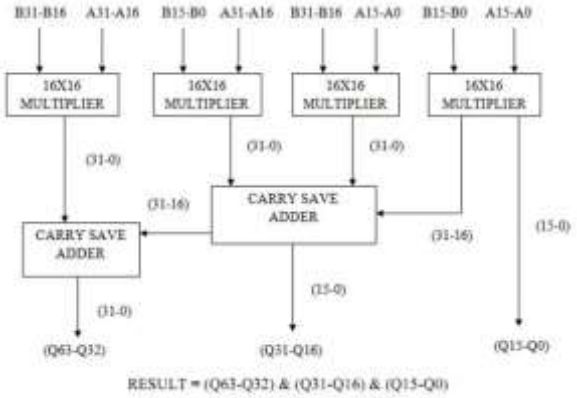


Fig 8: Vedic Multiplier Hardware

Further they analyzed that Karatsuba Multiplication Algorithm [16] can also form efficient multiplier in terms of speed, space and less power consumption because this method required only three multiplications and most of time this method is recursive in nature. Speed of throughput is considerably improved when Karatsuba and Vedic technique was applied in tandem rather than there individual application.

High Speed Reconfigurable FFT

In their paper "High Speed Reconfigurable FFT Design by Vedic Mathematics", Journal of computer science and engineering, Vol. 1, Issue 1, May 2010, Ashish Raman, Anvesh Kumar and R. K. Sarin, offered Vedic mathematics based reconfigurable FFT design with higher speed and lesser area. Urdhva Tiryakbhayam, being a general multiplication formula, is equally applicable to all cases of multiplication and is used to design high speed power efficient reconfigurable FFT [21][22]. The basic formula for FFT computation is

$$X(k) = \sum_{n=0}^{N-1} x(n) W_{N^{nk}} \qquad 0 \leq k \leq N \quad 1$$

They apply the Urdhva Tiryakbhayam algorithm to compute the multiplication of the binary number system to make the proposed algorithm compatible with the digital hardware. The basic architecture of the reconfigurable FFT is to be shown in Figure 9. It can calculate Fourier transform either 2-point or 4-point depending upon the select line and without changing the hardware of the system. This particular reconfigurable FFT is designed using Vedic adder, Vedic subtractor, and Vedic multiplier [20].

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

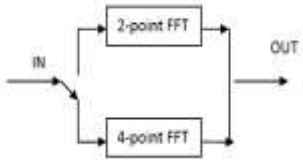


Fig 9: Reconfigurable Architecture

It has been found that the delay produced by the Vedic reconfigurable FFT is smaller than the delay produced by the conventional reconfigurable FFT and also provides the high performance (throughput), dynamic range, and small area.

Vedic Multiplier

Krishnaveni D. and Umarani T.G., in their paper "VLSI Implementation of Vedic multiplier with reduced delay", (IJATER) Volume 2, Issue 4, July 2012, designed a 4 X 4 binary multiplier using Urdhva Tiryakbhayam. They have proposed a new 4 bit Adder which when used in multiplier, reduces its delay [23]. The binary multiplication using this sutra is shown in Figure 10. Bit wise multiplication is performed independently (logical AND) and the product is marked to the common box. Crosswise dotted line bits are added to the previous carry.Least significant bit is the result and another one is defined as carry for next step[4].

The proposed 4-bit adder gives a sum and two bits of carry as output. Its block diagram is given in Figure 11.

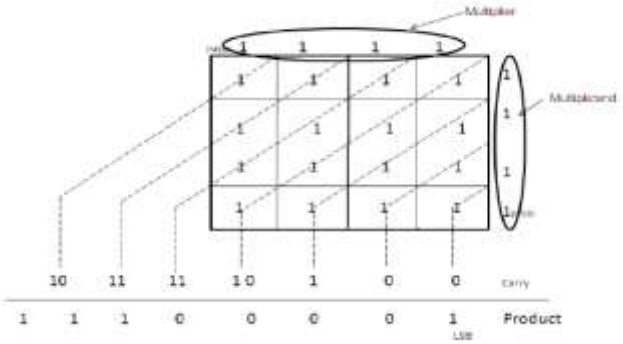


Fig 10: Two 4-bit binary numbers multiplication

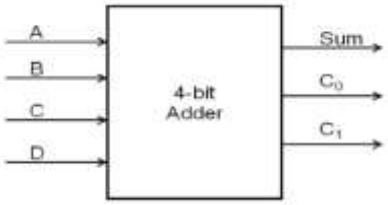


Fig 11: Proposed 4-bit Adder

$$Sum = \sum m(1,2,4,7,8,11,13,14) = A \oplus B \oplus C \oplus D$$

$$C_0 = \sum m(3,5,6,7,9,10,11,12,13,14) = \bar{B}D + C\bar{D} + B\bar{C}$$

$$C_1 = \sum m(15) = ABCD$$

where, A,B,C,D are four inputs; C0 and C1 are LSB and MSB of carry outputs respectively and Sum is the sum of four inputs.

They have designed a 4X4 multiplier using half adder, full adder and the proposed 4-bit adder as shown in Fig 12. and found that with this design the delay has been drastically reduced.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

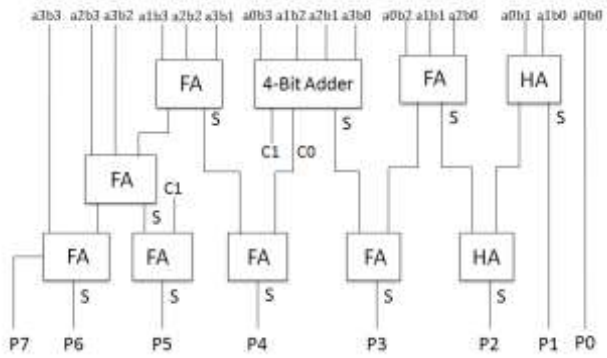


Fig 12: 4X4 Multiplier using 4-bit adder

Vedic Binary Division

Diganta Sengupta, Mahamuda Sultana and Atal Chaudhuri in their paper "Vedivision – A Fast BCD Division Algorithm Facilitated by Vedic Mathematics", (IJCSIT) Vol 5, No 4, August 2013, used the ancient Indian mathematics system to deduce a generalized BCD division algorithm in a much time efficient and optimized manner than the conventional methods in literature. They have used Nikhilam sutra and Paravartya sutra in their algorithm with some modification done to generalize the algorithm for all possible divisors. The algorithm performs the calculations on the number of digits in the divisor and the dividend rather than on the number of bits representing them [26]. Non Restore Type Division Algorithm consumes time in the division proportional to the number of bits [31] [32]. But in the Vedic Division Algorithm, the time requirement is based mainly on the number of normalizations of the intermediate remainders. Thus algorithm produces remarkable results on divisions involving big numbers. Further they have proven that the proposed algorithm performs better with respect to Non Restore Type Division Algorithm in terms of speed and memory requirement.

Floating Point Division

Najib Ghatte, Shilpa Patil, and Deepak Bhoir in their paper on "Single precision floating point division", Proceedings of Fifth IRF International Conference, 10th August 2014, Goa, proposed the division algorithm for single precision floating point division and they have verified it by writing Verilog Code implemented on Virtex-5 FPGA series. They observed the reduction in power dissipation, area -utilization and latency bounds. The proposed design uses ancient Math technique (Nikhilam Sutra) which pad zeroes for dividend. By means of logical shift, the estimated results were obtained. Synthesis results obtained provides better results in terms of computational speed and area utilization thereby reducing the size of silicon dye used which in turn reduces the size and cost of the device. Moreover, they have observed that less time delay has incurred which leads to fast computation and speedy calculations [33].

RSA cryptography

Greeshma Liz Jose and Sani John presented a paper on "VLSI Implementation of Vedic Mathematics and Its Application in RSA Cryptosystem" (IJIRD) Vol. 2, Issue 10, October 2013. In this paper performance comparison is made between Vedic multiplication and division algorithm and the conventional algorithm based on speed, power and area. Then Vedic algorithms are used to implement the RSA encryption and decryption system and the Vedic RSA enabled the RSA hardware to work as fast as its software counterparts [36]. Fig 13 shows the block diagram of the RSA architecture.

In Fig. 13 'm' is the message, which ciphered to 'c' using the encryption algorithm. 'c' is the cypher text input to decryption module , public key'd' and private key 'e' as per RSA algorithm. Public key is chosen and private key is generated. The 'n' is product of the two prime numbers 'p' and 'q'. 'phy' is defined to be (p-1* q-1). The Vedic encryption module describes modified encryption and decryption algorithms by Vedic mathematics. They concluded that the RSA algorithm implemented using Vedic multiplication and division is improved in speed, power, and in area utilization. The Vedic multiplication improves speed of hardware so as it can work in synchronization with the software counterparts.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

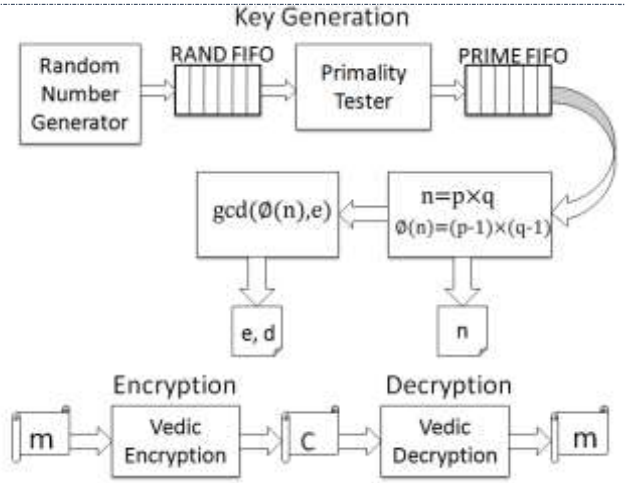


Fig 13: The block diagram of RSA crypto system

CONCLUSION

In this paper, the application of Vedic mathematics in the VLSI design implementation is described. Different multipliers and divisors based on Vedic mathematical sutras which are useful in applications like digital signal processing, image processing, and cryptography are discussed and it can be concluded that employing these sutras in the computation algorithm of the digital system will reduce the complexity of architecture, execution time, area, and power consumption.

Further with fabrication technology reaching at saturation, there arises immense scope in the field of VLSI design to delve much faster and efficient algorithm based on Vedic sutras.

REFERENCES

[1] Jagadguru Swami Sri Bharati Krisna Tirthaji Maharaja, Vedic Mathematics: Sixteen Simple Mathematical Formulae from the Veda, Delhi (1965).
[2] Neeraj Kumar Mishra, Subodh Wairya, "Low power 32x32 bit multiplier architecture based on Vedic mathematicsusing Virtex 7 low power design"in IJRREST Vol. 2, Issue-2, June 2013.
[3] Honey Durga Tiwari, Ganzorig Gankhuyag, Chan Mo Kim, Yong Beom Cho, Multiplier design based on ancient Indian Vedic Mathematics, International SoC Design Conference 2008.
[4] S.S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and Girish V A Implementation of Vedic Multiplier For Digital Signal Processing, International conference on VLSI communication & instrumentation (ICVCI) 2011.
[5] 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.
[6] Karishma P Dighorikar, S. L. Haridas, "Area Efficient Architecture for Convolution Using Vedic Mathematics" in IJSR Vol. 3 Issue 3 March 2014.
[7] K Mohammad, S Agaian, "Efficient FPGA implementation of convolution", Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics San Antonio, TX, USA - October 2009.
[8] Chao Cheng, "Low-Cost Fast VLSI Algorithm for Discrete Fourier Transform", IEEE,. IEEE transaction on circuits and systems, VOL. 54, 2007.
[9] P.Mehta,D.Gavli ,"Conventional versus Vedic mathematical method for Hardware implementation of a multiplier", 2009 International onference on Advances in Computing, Control, and Telecommunication Technologies.
[10] Madihalli J. Narasimha, "Linear Convolution using Skew Cyclic Convolutions", IEEE Transactions on Signal Process, vol. 14, no. 3, pp. 173-176, Mar. 2007.
[11] Leonard Gibson Moses S and Thilagar M, "VLSI Implementation of High Speed DSP algorithms using Vedic Mathematics", Singaporean Journal Scientific Research (SJSR) ISSN: 2231 - 0061Vol.3, 2010.
[12] Anveshkumar,Ashishraman,"Low Power, High Speed ALU Design by Vedic Mathematics "publish in National conference organized by NIT,hamirpur,2009.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

[13] Dharmendra Madke and Assoc.Prof. Sameena Zafar, "High Speed Karatsuba Multiplier and Vedic Mathematics Techniques", IJARCSSE, Vol3, Issue 12, December 2012.
[14] Wallace, C.S., "A suggestion for a fast multiplier," IEEE Trans. Elec. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.
[15] Mrs. M. Ramalatha, Prof. D. Sridharan, "VLSI Based High Speed Karatsuba Multiplier for Cryptographic Applications Using Vedic Mathematics", IJSCI, 2007.
[16] 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.
[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] Knuth, D. E. (1981). The art of computer programming, volume 2. Addison-Wesley, 2 editions, 1981.
[19] Karatsuba, A.; Ofman, Yu. (1962). Multiplication of multidigit numbers on automata. Sov. Phys. Dokl., 7:595-597.
[20] Ashish Raman, Anvesh Kumar and R. K. Sarin, "High Speed Reconfigurable FFT Design by Vedic Mathematics", IJARCSSE, Vol. 1, Issue 1, May 2010.
[21] Liu Zhaohui, Han Yueqiu. "Research of Implementing FFT in FPGA," Journal of Beijing Institute of Technology, 1999, 19(2). 234-238.
[22] Anvesh kumar,Ashish Raman,R K Sarin,Arun Khosla "Small Area Reconfigurable FFT Design by Vedic Mathematics " in Proc IEEE ICAAE'10,Singapoure, vol 5,pp. 836-838,Feb. 2010.
[23] Krishnaveni D. and Umarani T.G., "VLSI Implementation of Vedic multiplier with reduced delay", IJATER Volume 2, Issue 4, July 2012.
[24] Charles. Roth Jr., "Digital Systems Design using VHDL",Thomson Brooks/Cole, 7th reprint, 2005.
[25] Harpreet Singh Dhillon and Abhijit Mitra, " A Reduced-bit Multiplication Algorithm for Digital Arithmetic", International Journal of Computational and Mathematical Sciences, Febrauary 2008, pp.64-69.
[26] Diganta Sengupta, Mahamuda Sultana and Atal chaudhuri, "Vedivision – A Fast BCD Division Algorithm Facilitated by Vedic Mathematics", IJCSIT Vol 5, No 4, August 2013.
[27] Swami Bharati Krishna Tirtha's Vedic mathematics [Online]. Available: http://en.wikipedia.org/wiki/Vedic mathematics.
[28] HimanshuThapliyal, R.V Kamala and M.B Srinivas "RSA Encryption/Decryption in Wireless Networks Using an Efficient High Speed Multiplier", Proceedings of IEEE International Conference On Personal Wireless Communications (ICPWC-2005) , New Delhi, pp-417-420, Jan 2005.
[29] ManoranjanPradhan, Rutuparna Panda and Sushanta Kumar Sahu, "Speed Comparison of 16x16 Vedic Multipliers", International Journal of Computer Applications (0975 – 8887), Volume 21– No.6, May 2011.
[30] Shuangching Chen and ShugangWei,"A High-Speed Realization of Chinese Remainder Theorem", Proceedings of the 2007 WSEAS Int. Conference on Circuits, Systems, Signal and Telecommunications, Gold Coast, Australia, January 17-19, 2007.
[31] Stuart F. Oberman, and Michael J. Flynn, "Division Algorithms and Implementations", IEEE Transactions on Computers, vol. 46, no. 8, August 1997.
[32] J. O'Leary, M. Leeser, J. Hickey and M. Aagaard, "Non-Restoring Integer Square Root: A Case Study in Design by Principled Optimization", LNCS.
[33] Najib Ghatte, Shilpa Patil, and Deepak Bhoir, "Single precision floating point division", Proceedings of Fifth IRF International Conference, 10th August 2014.
[34] Sherif Galal, Dung Pham, "Division Algorithms and Hardware Implementations" in EE 213A: Advance DSP Circuit Design, 2000.
[35] R. Tamil Chelvan and S. Roobini Priya, "Implementation of Fixed and Floating Point Division using Dhvanjanka Sutra" in International Journal of VLSI and Embedded Systems, pp. 234-237, Apr. 2013.
[36] Greeshma Liz Jose and Sani John, "VLSI Implementation of Vedic Mathematics and Its Application in RSA Cryptosystem", IJIRD Vol. 2, Issue 10, October 2013.
[37] Chiranth E,Chakravarthy H,"implementation of rsa Cryptosystem using verilog" international Journal for Scientific and Engineering Research,May 2011.
[38] Vibhor Garg, V. Arunachalam,"Architectural analysis of RSA cryptosystem on fpga" international Journal for computer application, July 2011.
[39] Vaijyanath Kunchigi, Linganagouda Kulkarni, Subhash Kulkarni, "High Speed and Area Efficient Vedic Multiplier,"in IEEE nternational Conference on Devices, Circuits and Systems (ICDCS), 2012.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

[40] A.P. Nicholas, K.R Williams, J. Pickles, "Application of Urdhava Sutra", Spiritual Study Group, Roorkee (India), 1984.