International Journal of Engineering & Technology, 7 (2.24) (2018) 336-340

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Complex Number Vedic Multiplier and its Implementation in a Filter

N.Saraswathi[1*], Lokesh Modi[2], Aatish Nair[3]

[1]Assistant Professor (S.G), [2,3]UG Scholar, [1,2,3]ECE, SRM Institute of Science and Technology, Chennai, Tamil Nadu

**Corresponding Author E-mail: saraswathy.n@ktr.srmuniv.ac.in*

Abstract

Complex numbers multiplication is a fundamental mathematical process in systems like digital signal processors (DSP). The main objective of complex number multiplication is to perform operations at lightning fast speed with less intake of power. In this paper, the best possible architecture is designed for a Real vedic multiplier based on the ancient Indian mathematical procedure known as URDHVA TIRYAKBHYAM SUTRA i.e. the structure of a MxM Vedic real multiplier architecture is developed. Then, a Vedic real multiplier solution of a complex multiplier is presented and its simulation results are obtained. The MxM Vedic real multiplier architecture, architecture of the Real Vedic multiplier solution for 32 x 32 bit complex numbers multiplication of complex multiplier and the architecture of a FIR filter has been code in Verilog and implementation is done through Modelsim 5.6 and Xilinx ISE 7.1 navigator.

Keywords: *Ancient Indian Mathematics; Urdhva Tiryakbhyam Sutra; Real vedic Multiplier; Complex number multiplication; FIR filter*

1. Introduction

Complex number multipliers are principally used in multiply accumulate operation (MAC) which is often known as a common step that computes the product of two numbers consequently adds the product to an accumulator complex-valued function and also used in a discrete Fourier transform (DFT) to achieve a given frequency. The speed of the processors is largely dependent on the speed of multipliers. Due to the recent advancements in the field of wireless communication, complex number multipliers have got immense significance due to the compound nature of its wireless network. Their application is found unique in systems like Discrete Cosine Transformation (DCT), Discrete Sine Transformation (DST). This leads to the large scale usage of complex multipliers in the hardware modules of various systems. The ancient Indian mathematics is one of the most advanced and proficient mathematical system [2].One of such efficient technique, from the system has been employed to boost the design of the multiplier. This sutra is efficient and can also be applicable to all cases of multiplication. The idea for designing the adder subtraction unit and multiplier is adopted from the" vedas" [3]. On account of those formulas the sums and the partial products are generated in one step, which significantly reduces the propagation of carry from LSB to MSB [3]. Rao et al. has shown in [1] that, in implementation of the Vedic multiplier the path delay plays a very huge role, as it will enable to choose a minimum delay architecture . When an architecture with minimum path delay is chosen, not only does it make the architecture faster but it also makes it more efficient among the other know architectures.

Hence, in this paper, the architectures of a Vedic Complex number multiplier based on Urdhva Tiryakbhyam sutra has been designed. The architecture includes 2x2 bit vedic multiplier, two architectures of 4x4 bit vedic multipliers. The two architectures of 4x4 bit Vedic multiplier has 2x2 bit multiplier as their fundamental module. Then a bigger and efficient architecture of an MxM architecture is designed which has (M/2 X M/2) bit vedic multiplier as their fundamental module. We then design Real vedic Multipliers Solutions of a complex multiplier and then compare the design solutions results of the proposed vedic multiplier with that of the real multiplier solutions using the Booth, and Array multipliers [1]. All the architectures have been designed by the path analysis shown by Rao et al in [1]. This Real vedic multipliers solutions of a complex multiplier is then implemented on an FIR filter to achieve the frequency required for the selective response of a spectrum of the input signal.

2. Urdhva Tiryakbhyam Sutra

In Sanskrit, Urdhva means vertically and Tiryakbhyam means crosswise. Urdhva Tiryakbhyam Sutra is a multiplication formula which can be applied to all cases of multiplication. The multiplier has been designed based on ancient Indian vedic mathematics sutra i.e Urdhva Tiryakbhyam Sutra. There are different steps to the various process in the algorithm. In every step a sum is produced along with a carry, this carry is taken forward to the next step for further processing. The carry and sum are different for the number of bits of the multiplier. This is process is done in all the architecture subsequently and finally put in MxM architecture. One of the main reason why this method is used, because it largely facilitates, even when the number of bit are increased. It's astonishing to find that the speed of the multipliers used in processors is largely dependent on the fixed structure of the sutra not the number of bits of the number.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

2.1. 2x2 Vedic Bit Multiplier

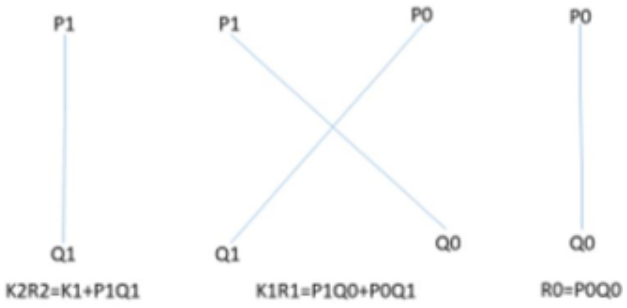


Fig. 1: Multiplication Method in 2x2 bit Vedic Multiplier

Let us, consider two, 2 bit binary numbers, P0 P1 as multiplier and Q0 Q1 as multiplicand respectively as shown in Fig 1. In the given process, the least significant bit R0; of the 2x2 Vedic Bit Multiplier i.e. can be found by (vertical) product of the least significant numbers of both multiplicand and multiplier i.e. P0Q0.

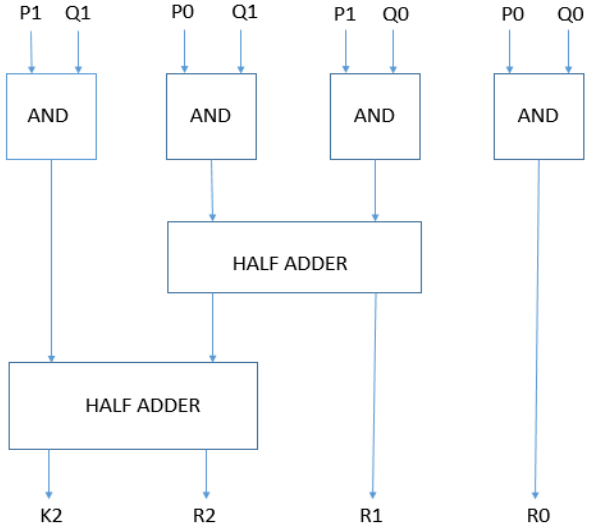


Fig. 2: 2X2 bit Vedic Multiplier Architecture

The middle significant bit R1 can be found out by the addition of the (crosswise) product of the P0 bit of the multiplicand with the Q1 bit of the multiplier and the P1bit of the multiplicand with the Q0 bit of the multiplier. With the middle significant bit there will also be a carry element K1 attached in this process. This K1 will be sent for further processing of the most significant bit R2. The most significant bit R2 will be found out by (vertical) product of Q1 of the multiplier with the P1 of the multiplicand. The final product will be R2 as MSB and K2 as the carry.

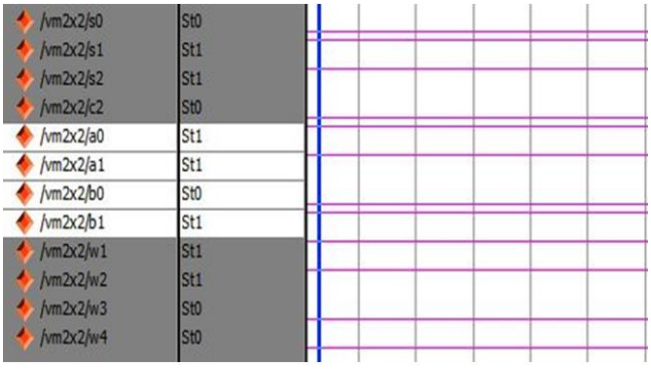


Fig. 3: Simulation Result of 2x2 Bit Vedic Multiplier

This structure of 2x2 bit vedic multiplier, as shown in Fig 2 can be implemented by using 4 AND gates and 2 Half Adders which has P1 Q1, P0 Q1, P1 Q0, P0 Q0 as inputs and output as R0 R1 R2 and carry element as K2. The simulation output of this structure is displayed in Fig 3.

2.2 4x4 Vedic Bit Multiplier

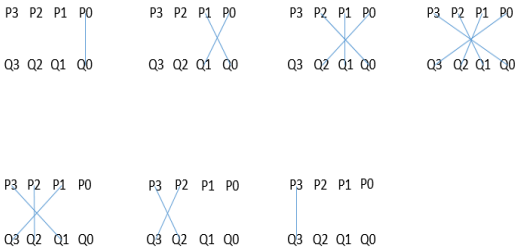


Fig. 4: Multiplication Method in 4x4 bit Vedic multiplier

Considering two, 4 bit binary numbers as P3 P2 P1 P0 and Q3 Q2 Q1 Q0 and dividing them equally into Q3 Q2, Q1 Q0, P3 P2 and P1 P0. The final output is given as R7 R6 R5 R4 R3 R2 R1 R0. The main reason for the division of the numbers is to facilitate 2x2 bit Vedic multiplier as the fundamental module for the 4x4 bit Vedic multiplier. The 4x4 bit vedic multiplier can be implemented in two specific architectures, one which includes, four 2x2 bit Vedic multiplier architectures and three 8-bit adder shown in the Fig 5 and another one which include four, 2x2 bit vedic multiplier architecture and three, 4-bit ripple carry adders shown in Fig 7. The simulation outputs of each of the architecture of a 4x4 bit vedic multiplier is generated alongside in Fig 6 and Fig 8. Among the two architectures, the one which has minimum delay with respect to its sub modules will be considered to be the faster one. According to Rao et al. the architecture 2 has lessor number of AND gates, HALF ADDERS and OR gates hence it has minimum path delay when compared to architecture 1, so it is faster among the two [1].

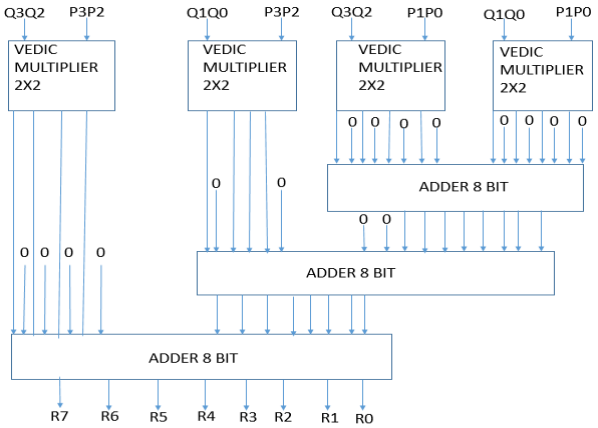


Fig. 5: 4x4 bit Vedic Multiplier Architecture (I)

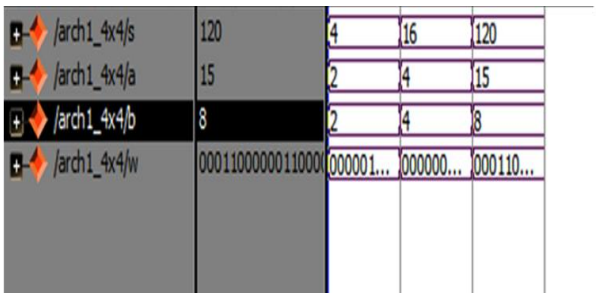


Fig. 6: 4x4 bit Vedic Multiplier Architecture (I) Simulation Output

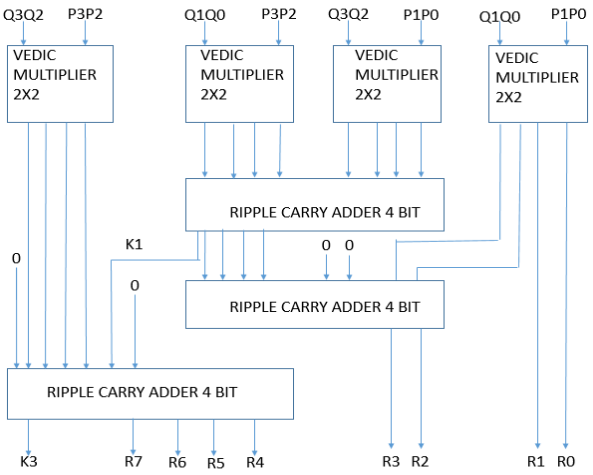


Fig. 7: 4x4 bit Vedic Multiplier Architecture (II)

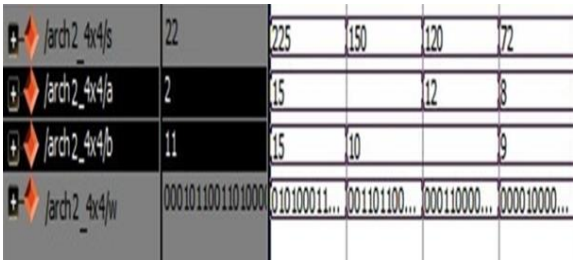


Fig. 8: 4x4 bit Vedic Multiplier Architecture (II) Simulation Output

2.3. MxM Bit Vedic Multiplier

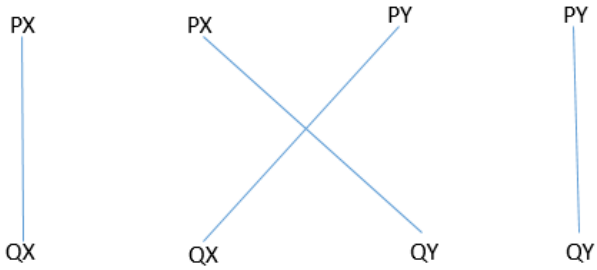


Fig. 9: Multiplication Method in MxM Bit Vedic multiplier

In this section we are going to discuss about the MxM bit Vedic multiplier. This generalized version of the Vedic multi- plier implements the previously discussed architectures. In MxM architecture we use (M/2 x M/2) as the fundamental basis. Here, PX PY QX QY are put in as inputs, i.e. the multiplicand and multiplier are divided in different orders. In which PX is MSB and PY is LSB, similarly for QX and QX.

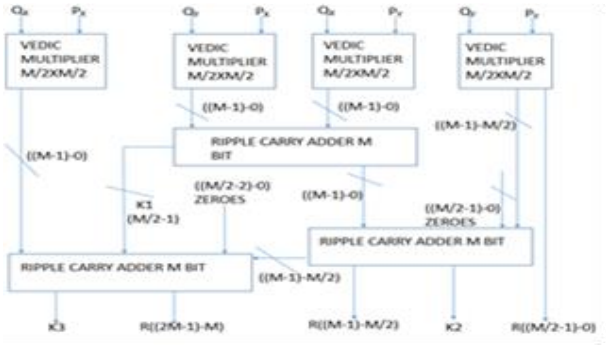


Fig. 10: MxM bit Vedic Multiplier Architecture

The PX contains ((M-1) to (M/2)) bits, and PY consisting of ((M/2-1) to 0) bits, similarly the same is done for QX and QY these values are passed on to four (M/2 x M/2) bit Vedic multiplier. After passing the values through (M/2 x M/2), the values are further processed to three ripple carry adder of M bit. The final output is in terms of R and the carry element is in K terms.

As discussed by Rao et al in [1], the formula through which the path delay can be found out for an MxM bit Vedic multiplier is:

$$\boldsymbol{pd_{NVM} = d_{AND} + 2d_{HA} + \sum_{n=4,8\ldots N} 3[(2n-1)\, d_{HA} + (n-1)d_{OR}]};$$ (1)

where $\boldsymbol{pd_{NVM}}$ is path delay for a MxM bit vedic multiplier

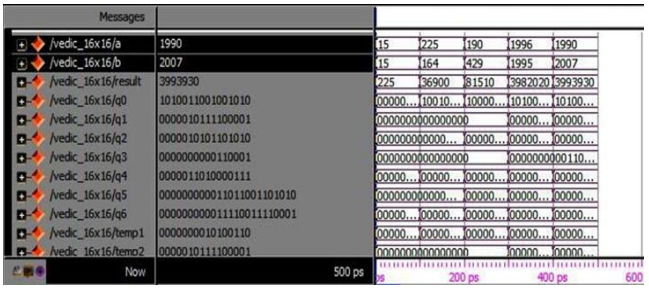


Fig. 11: MxM bit Vedic Multiplier Architecture Simulation Output

2.4 Implementing Vedic Real Multiplier Solution on a Complex Multiplier

In this section, the Vedic real multiplier solutions are implemented on a complex multiplier. For e.g., if we have two complex number let's say:

$$P = Pr + jPi \text{ and } \quad Q = Qr + jQi \tag{2}$$

Then the product of k and t are computed as:

$$R = Pt = Rr + jRi \tag{3}$$

the imaginary part of the product is:

$$Ri = (PrQi - PiQr) \tag{4}$$

and the real part of the product is:

$$Rr = (PrQr - PiQi) \tag{5}$$

The best possible solutions of a real multiplier can be formulated by the following equations according to the above equations:

$$Rr = Pr\,Qr - Pi\,Qi = Pr(Qr + Qi) - (Pr + Pi)Qi \tag{6}$$

$$Ri = Pr\,Qi + Pi\,Qr = Pr\,(Qr + Qi) + (Pi - Pr)\,Qr \tag{7}$$

The best possible real multiplier solutions implement only three MxM vedic multiplier in its architecture but whereas the conventional multipliers use four real multipliers. The main reason why architecture with three multipliers is used because they have less hardware and have a much lower frequency when compared to the architecture which has four multipliers. When an architecture uses less hardware they eventually become faster and light in weight. Therefore, this architecture is preferred.

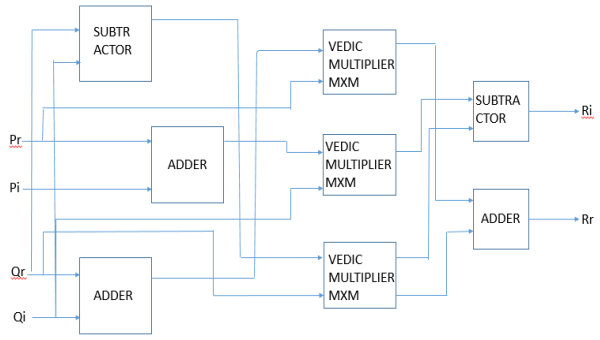
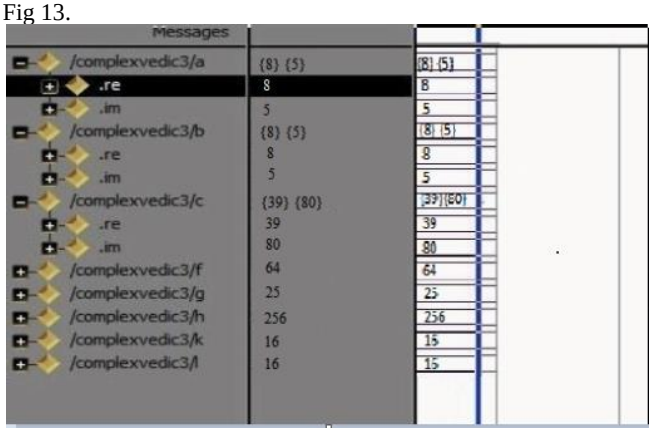


Fig. 12: Real Vedic Multiplier Architecture for Complex Multiplication

The simulation results of the above architecture will now be verified using two complex numbers (5+j8) and (5+j8). The product of this process is (39+80j) which is shown in the simulation output Fig 13.



(5+j8) X (5+j8) = (39+80j)

Fig. 13: Real Vedic Multiplier Architecture for Complex Multiplication

2.5 Comparison of the Vedic Multiplier with Booth and Array Multipliers

PARAMETERS	VEDIC COMPLEX MULTIPLERS	BOOTH COMPLEX MULTIPLIERS	ARRAY COMPLEX MULTILIERS
NUMBER OF LUT's	8096	8621	6971
LOGIC POWER (mW)	7.34	10.47	7.09
PATH DELAY	28.12 (ns)	29.936 (ns)	50.519 (ns)
NUMBER OF BONDED IOB's	256 out 1100 23%	256 out 1100 23%	256 out 1100 23%
POWER (mW)	128.53	123.65	73.88

Fig. 14: Comparision analysis and device utilization summarization of 32x32 bit vedic complex multiplier

From the implementation results shown above, of booth and array complex multiplier has been cited from [1] and it is observed that the Vedic complex multiplier consumes slightly more power than the array complex multiplier but is more efficient having less power consumption and less path delay as compared to Booth complex multiplier.

2.6 Implementation of Vedic Multiplier in Fir Filter

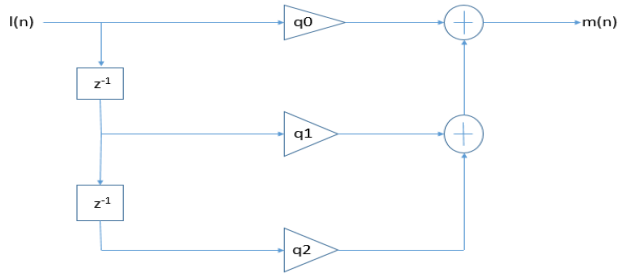


Fig. 15: Vedic multiplier implemented in a FIR filter

The main reason to use a filter is to achieve the kind of frequency selectivity required for the spectrum of the input signal. The reason we use FIR filter instead of IIR filter is to achieve maximum stability and provide linear state response. In order to have a linear-phase FIR filter, we must provide symmetry in the time domain, i.e. b[n] = ±b [M-1-n]. According to the direct form of an FIR filter of order 2(Shown below), assume b0 = b2, hence the equation gives:

$$H(z) = b0 + b1\, e^{-jw} + b0\, e^{-2jw} = e^{-jw}(b1 + 2b0\cos(w)) \quad (8)$$

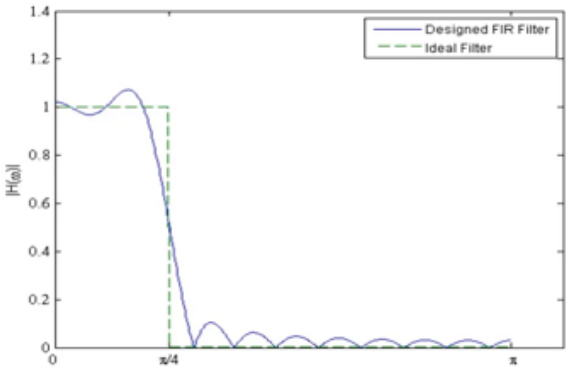


Fig. 16: The frequency response of a low pass FIR filter

3. Conclusion

The designing of Real vedic multiplier based on ancient Indian Vedic Mathematics i.e Urdhva Tiryakbhyam sutra is presented. The structure for MxM architecture and all it sub-modules like have been developed according to the given path delay analysis and simulation output have been obtained for each of the architectures. Further, the best possible architecture of the multipliers real solution is presented for vedic complex multiplier and a comparative analysis to that of Booth and Array complex multipliers have been studied. The given architecture is also implemented on an FIR filter and frequency response have been shown.

Acknowledgement

We would like to deeply express our gratitude to our faculty N.Saraswathi for the indispensable guidance, timely help ,consistent encouragement and providing us with an excellent atmosphere for doing research. All through the work, she has always extended her cordial support for completing this research work , in spite of her busy schedule. We would also be thankful to the ECE department of our college; SRM Institute of Science and Technology for providing research EDA lab during our research work.

References

[1] K Deergha Rao, Ch. Gnagdhar, FPGA Implementation Of Complex Multiplier Using Minimum delay Vedic Real Multiplier Archi-tecture, Uttar Pradesh Section International Conference on

Electrical, Computer and Electronics Engineering (UPCON), Indian Institute of Technology (Banaras Hindu University), Varanasi, India, Dec9-11, IEEE-2016

[2] Sudeep M.C., M. Sharath Bimb, Mahendra Vucha, “Design and FPGA Implementation of High Speed Vedic Multiplier”, International Journal of Computer Applications (0975 – 8887)Volume 90 – No 16, March 2014

[3] P. K. Saha, A. Banerjee, and A. Dandapat, "High Speed Low Power Complex Multiplier Design Using Parallel Adders and Sub-tractors," International Journal on Electronic and Electrical Engineer-ing, (IJEEE), vol 07, no. II, pp 38-46, Dec. 2011.

[4] Tiwari, H. D, Gankhuyag, G., Kim, C. M., and Cho, Y. B. "Multiplier design based on ancient Indian Vedic Mathematics,"Proc. IEEE International SoC Design Conference (ISOCC’08), Vol.2, pp.65-68, 2008.

[5] Agadguru Swami Sri Bharath, Krsna Tirathji, "Vedic Mathematics or Sixteen Simple Sutras From The Vedas", Motilal Banarsidas, Varanasi (India),1986.