Low power Square and Cube Architectures Using Vedic Sutras

Vaijyanath kunchigi
ECE JNTU Hyberabad
Andhra Pradesh, INDIA
e-mail: vaijanath.k@gmail.com

Linganagouda Kulkarni
ECE JNTU Hyberabad
Andhra Pradesh, INDIA
e-mail:
linganagouda@yahoo.co.uk

Subhash Kulkarni
PESIT, Bangalore South
Campus, Bangalore
Karnataka, INDIA
e-mail: subhashsk@gmail.com

***Abstract*— In this paper low power square and cube architectures are proposed using Vedic sutras. Low power and less area square and cube architectures uses Dwandwa yoga Duplex combination properties of Urdhva Tiryagbhyam sutra and Anurupyena sutra of Vedic mathematics. Simulation results for 8-bit square and 8-bit cube shows that proposed architectures lowers the total power consumption by 45% and area by 63% when compared to the conventional architecture. Also the reduction in power consumption increases with the increase in bit width. Comparison is made between conventional and Vedic method implementations of square and cube architecture. Implementation results show a significant improvement in terms of area, power and delay. Proposed square and cube arhitectures can be used for high speed and low power applications. Synthesis is done on Xilinx FPGA Device using, Xilinx Family: Spartan 3E, Speed Grade: -4. Propagation delay of the proposed 8-bit square is 4ns and area consumed in terms of slices is 22 and for 8-bit cube propogation delay is 7.72ns and area consumed in terms of slices is 58. Dynamic power estimation for square and cube are 13mW and 16mW respectively.**

Keywords- Vedic mathematics; Dwandwa yoga; Anurupyena; Square; Cube; low power; high speed.

I. INTRODUCTION *(HEADING 1)*

Square and cube are frequently performed functions in most of the DSP systems. Square and cube are special cases of multiplication. Square and cube architectures forms the heart of the different DSP operations like Image Compression, Decoding, Demodulation, Adaptive Filtering, Least Mean Squaring etc., and also have numerous applications as mentioned in [1] such as cryptography, computation of Euclidean distance among pixels for a graphics processor or in rectangular to polar conversions in several signal processing circuits where full precision results are not required. Traditionally, square and cube were performed using multiplier itself. As the applications evolved demand for the high speed processing increased, special attention was given for square and cube function [9-11].

Most important feature of the Vedic mathematics [2] is its coherence, entire system is wisely interrelated and unified. Simple Squaring Scheme can easily be reversed to produce one-line Square Roots. These methods are very easy to understand.

In this paper algorithms and architectures used to design square and cube of a binary number is exploreed and to create a circuit using the Vedic Sutras. Often times square and cube are the most time-consuming operations in many of digital signal processing applications and computation can be reduced using the vedic sutras and the overall processor performance can be improved for many applications [8]. Therefore, the goal is to create a square and cube architectures that is comparable in speed, power and area than a design using an standard multiplier. The motivation behind this work is to explore the design and implementation of Square and Cube architectures for low power.

This paper is organized as follows. Section II gives the overview of Vedic mathematics and Vedic mathematics sutras and sub-sutras. Section III briefs about square architecture section IV details about cube architecture section V discusses about results and discussion and scetion VI about the conclusion.

II. VEDIC MATHEMATICS

Vedic Mathematics (VM) is an ancient system of mathematics that was re-discovered by Sri Bharati Krishna Tirthaji between 1911 and 1918. Tirthaji (1884-1960) was an Indian scholar well versed in the areas of Sanskrit, English, Mathematics, Astronomy and many other areas of science. He deciphered ancient Indian texts, known as the "Ganita Sutras", (which means mathematics) to discover 16 short verses, known as "Sutras". These sutras, when applied correctly, will enable the user to solve many types of mathematics problems mentally without having to use pencil and paper in a fraction of the time in would take otherwise. Tirthaji wrote sixteen books, one for each sutra, describing the application of each to the solution of math problems. Unfortunately, these books were lost. Tirthaji attempted to re-write all of these books from memory, but was only able to complete the first volume entitled "Vedic Mathematics" before his death. This book, which is available today, is the seminal work on Vedic Mathematics.

Swami Bharati Krishna Tirtha (1884-1960), former Jagadguru Sankaracharya of Puri culled a set of 16 Sutras (aphorisms) and 13 Sub - Sutras (corollaries) from the Atharva Veda. He developed methods and techniques for amplifying the principles contained in the aphorisms and

978-0-7695-5100-5/13 $31.00 © 2013 IEEE
DOI 10.1109/ICSIP.2014.62

their corollaries, and called it Vedic Mathematics. According to him, there has been considerable literature on Mathematics in the Veda-sakhas. Unfortunately most of it has been lost to humanity as of now. This is evident from the fact that while, by the time of Patanjali, about 25 centuries ago, 1131 Veda-sakhas were known to the Vedic scholars, only about ten Veda-sakhas are presently in the knowledge of the Vedic scholars in the country. The Sutras apply to and cover almost every branch of Mathematics. They apply even to complex problems involving a large number of mathematical operations. Application of the Sutras saves a lot of time and effort in solving the problems, compared to the formal methods presently in vogue [3-7]. Though the solutions appear like magic, the application of the Sutras is perfectly logical and rational. The computation made on the computers follows, in a way, the principles underlying the Sutras. The Sutras provide not only methods of calculation, but also ways of thinking for their application.

A. Vedic Mtahematics Sutras

This list of sutra is taken from the book Vedic Mathematics [2], which includes a full list of the 16 main sutras. The following are the 16 main sutras or formulae of Vedic math and their meaning in English.

Ekadhikena Purvena: One more than the previous
Nikhilam Navatascharamam Dastah: All from nine and last from ten
Urdhwa-tiryagbhyam: Criss-cross
Paravartya Yojayet: Transpose and adjust
Sunyam Samyasamuchchaye: When the samuchchaya is the same, the samuchchaya is zero, and i.e. it should be equated to zero.
(Anurupye) Sunyamanyat: If one is in ratio, the other one is zero.
Sankalana-vyavkalanabhyam: By addition and by subtraction
Puranpuranabhyam: By completion or non-completion
Chalana-Kalanabhyam: Differential
Yavdunam: Double
Vyastisamastih: Use the average
Sesanyankena Charmena: The remainders by the last digit
Sopantyadyaymantyam: The ultimate & twice the penultimate
Ekanyunena Purven: One less than the previous
Gunitasamuchachayah: The product of the sum of coefficients in the factors
Gunaksamuchchayah: When a quadratic expression is product of the binomials then its first differential is sum of the two factors

B. Vedic Mathematics Sub-Sutras

Anurupyena: Proportionately
Sisyate Sesasamjnah: Remainder remains constant
Adyamadyenantyamantyena: First by first and last by last
Kevalaih Saptakam Gunyat: In case of seven our multiplicand should be 143
Vestanam: Osculation
Yavdunam Tavdunam: Whatever the extent of its deficiency, lessen it still further to that very extent
Yavdunam Tavdunam Varganchya Yojayet: Whatever the extents of its deficiency lessen it still further to that very extent; and also set up the square of that deficiency.
Antyayordasakepi: Whose last digits together total 10 and whose previous part is exactly the same
Antyayoreva: Only the last terms
Samuchchyagunitah: The sum of the coefficients in the product
Lopanasthapanabhyam: By alternate elimination and retention
Vilokanam: By observation
Gunitsamuchchayah Samuchchayagunitah: The product of sum of the coefficients in the factors is equal to the sum of the coefficients in the product.

III. SQUARE ARCHITECTURE

Square Architecture using dwandwa yoga property of urdhvatiryagbhyam sutra [11, 13]. Yavadunam Sutra is used for Squaring , is limited to the number which are near the base 10,100 etc., The "Ekadhikena Purvena Sutra" is used for Squaring, is limited to number which ends with digit 5 only. The other method "Dwandwa Yoga" or Duplex is used in two different senses. The first one is by squaring and the second one is by cross multiplication. It is used in both the senses (a^2 , b^2 and 2ab)

In order to calculate the square of a number "Duplex" D property of Urdhva Tiryagbhyam is used. In the Duplex, take twice the product of the outermost pair, and then add twice the product of the next outermost pair, and so on till no pairs are left. When there are odd number of bits in the original sequence there is one bit left by itself in the middle, and this enters as its square. Thus for 987654321, D= 2 * (9 * 1)+2 * (8 * 2)+2 * (7 * 3)+ 2 * (6 * 4)+ 5 * 5=165. Further, the Duplex can be explained as follows

For a 1 bit number D is its square.

For a 2 bit number D is twice their product

For a 3 bit number D is twice the product of the outer pair + the square of the middle bit.

For a 4 bit number D is twice the product of the outer pair + twice the product of the inner pair.

Thus D (1) = 1 * 1;
D (11) =2 * 1 * 1;
D (101) =2 * 1 * 1+0 * 0;
D (1011) =2 * 1 * 1+2 * 1 * 0;

The Vedic square has all the advantages of the Vedic multiplier. Further, it is quite faster and smaller than the array, Booth and proposed Vedic multiplier.

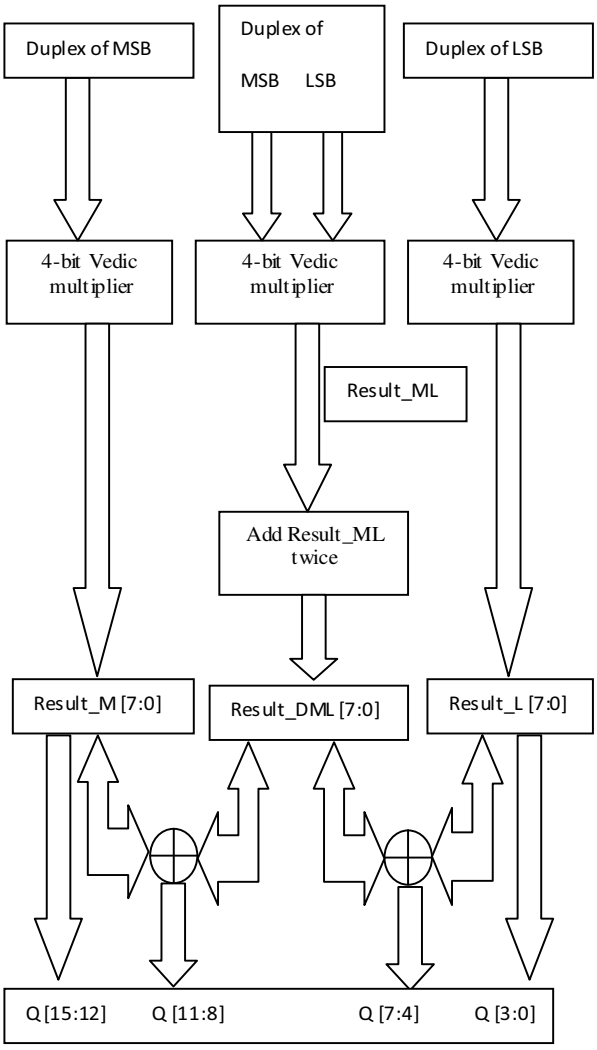

Figure 1. 8-bit Vedic Square Algorithm

IV. CUBE ARCHITECTURE

Yavadunam Sutra can be applied for cubing too. The only difference is tha, take here not the deficit or the surplus but exactly twice the deficit or the surplus as the case. Again this sutra is limited to the number which is near base i.e., 10,100 etc., Cubing of 2-digit numbers can also be performed using another sutra called Anurupyena Sutra [10-11]. To use this, follow the procedure below:

1. Put down the cube of the left digit of the number to be cubed as the left most number in a row of 4 numbers
2. Put down the square of the left digit multiplied by the right digit as the second number in the same row of numbers
3. Put down the square of the right digit multiplied by the left digit as the third number in the same row of numbers
4. Put down the cube of the right digit as the right most number in this row of numbers
5. Under the second number in the row above, put down twice the second number
6. Similarly, under the third number in the first row, put down twice the third number
7. Add them up, making sure to carry over excess digits from right to left. That is the final answer.

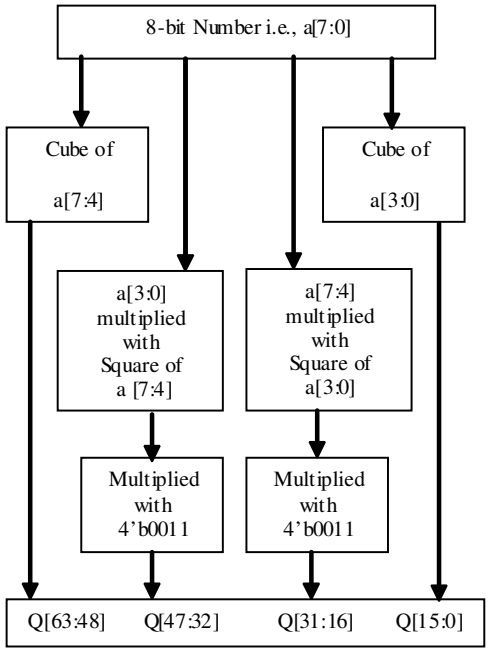

Figure 2. 8-bit Vedic Cube Algorithm

Note that the first row can also be expressed as writing the numbers from the cube of the first digit to the cube of the second digit such that the numbers in between form the same ratio with respect to each other. In other words, the numbers in the first row are in geometric progression from the cube of the first digit to the cube of the second digit. In fact the constant ratio of the geometric progression is the same as the ratio between the first and second digits of the number to be cubed.

Also note that the procedure above is a direct result of the algebraic identity that $(a + b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$. The first line contains the terms a^3, a^2b, ab^2 and b^3. The second row contains the remaining $2a^2b$ and $2ab^2$ (double of the middle two terms of the first row).

To illustrate, let us work through some simple examples:

25^3 = 8|20| 50|125
40|100

15625

Note the carryovers carefully in the example above. From the right most columns, 12 are carried over to the left. This gets added to 150, giving 162, of which 2 remains and 16 is carried over further to the left. This gets added to 60, giving 76. 6 remains and 7 are carried over to the left-most column, giving the final sum of 15. Each column should consist of one digit, with all the excess digits carried over to the left until the left-most column, which obviously will not have any carryover out of it.

V. Results And Discussions

In this work, 8-bit squaring and cube architectures are implemented in Verilog HDL [10]. Logic synthesis and simulation are done in Xilinx - Project Navigator and Xilinx ISE simulator. Synthesis results are compared to conventional method. The results are displayed in Table I and Table II for squaring and cube architecutre of 8-bit size. These Tables show the difference in combinational delays. Device utilization and low power estimation.

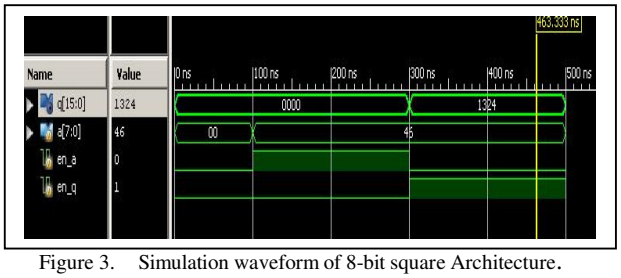

Figure 3. Simulation waveform of 8-bit square Architecture.

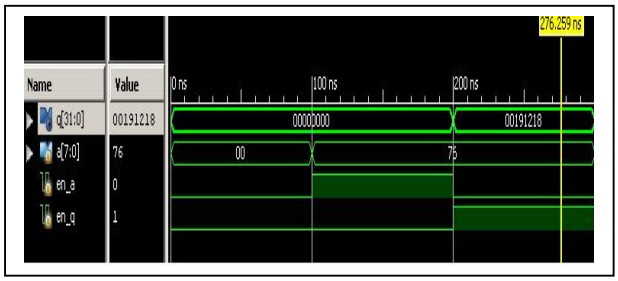

Figure 4. Simulation waveform of 8-bit cube Architecture.

Thus, proposed method outperforms conventional method in terms of speed, area and low power. The proposed squaring architecture may be useful for the design of hardware for computer arithmetic. The result obtained from proposed Vedic square and cube are faster than conventional square and cube. Figure 5 shows the comparison chart of the Vedic and Conventional methods in Square and Cube Architectures.

For the Xilinx, Spartan 3E family it has been found that the gate delay in Vedic square for 8x8 bit number is 4ns while it is 26.825ns for conventional square. The numbers of slices are 22 for Vedic square, 73 for conventional square. For the Xilinx, Spartan 3E family it has been found that the gate delay in Vedic cube for 8x8 bit number is 7.72 ns while the gate delay in conventional cube is 41.34ns . The area i.e the number of slices in Vedic cube are 58 while in conventional cube, they are 128. Figure 3 and Figure 4 shows the comparison of Vedic square and cube with conventional square and cube. Thus the result shows that the proposed Vedic square and cube is smallest, fastest and low powered of the reviewed architectures.

Table I. Area, Speed & Power comparison of cube

8-bit Cube	Area, Speed and Power		
	Area (slices)	*Speed(ns)*	*Power (mW)*
Vedic	58	7.72	69.00
Conventional	128	41.34	151.70

Table II Area, Speed & Power comparison of square

8-bit Square	Area, Speed and Power		
	Area (slices)	*Speed (ns)*	*Power (mW)*
Vedic	22	4.00	65.00
Conventional	73	26.825	97.51

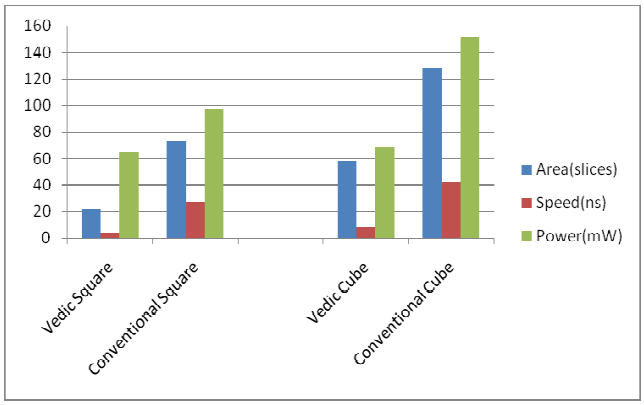

Figure 5. Comparison Chart for Vedic & Conventional methods.

VI. Conclusion

Due to its regular and parallel structure it can be concluded that Vedic Square and cube are faster than conventional square and cube. Due to factors of timing efficiency, speed, low power and less area the proposed Vedic square and cube can be implemented in Arithmetic and Logical Units replacing the traditional square and cube circuits. It is demonstrated that this design is quite efficient in terms of area, speed & low power. Squaring of binary numbers of bit size other than powers of 2 can also be realized easily. For example, squaring of a 24-bit binary number can be found by using 32- bit squaring circuit with 8 MSBs (of inputs) as zero. The idea proposed here may set

path for future research in this direction. Future scope of research is to reduce area requirements.

REFERENCES

[1] Y.Yu Fengqi and A. N.Willson,"Multirate digital squarer architectures," in Proc. 8th IEEE Int. Conf. on Electronics, Circuits and Systems (ICECS 2001), Malta, Sept. 2–5, 2001, pp. 177–180.

[2] Swami Bharati Krisna Tirtha, "Vedic Mathematics," Motilal Banarsidass Publishers, Delhi, 1965.

[3] H. D. Tiwari, G. Gankhuyag, C. M. Kim, and Y. B. Cho, "*Multiplier design based on ancient Indian Vedic Mathematics,*" in Proceedings IEEE International SoC Design Cotiference, Busan, Nov. 24-25, 2008,pp. 65-68

[4] Kunchigi, V.; Kulkarni, L.; Kulkarni, S., "High speed and area efficient vedic multiplier," Devices, Circuits and Systems (ICDCS), 2012 International Conference on , vol., no., pp.360,364, 15-16 March 2012

[5] Vaijyanath Kunchigi, Linganagouda Kulkarni and Subhash Kulkarni. Article: Simulation of Vedic Multiplier in DCT Applications.International Journal of Computer Applications 63(16):27-32, February 2013. Published by Foundation of Computer Science, New York, USA.

[6] Vaijyanath Kunchigi, Linganagouda Kulkarni and Subhash Kulkarni 32-BIT MAC UNIT DESIGN USING VEDIC MULTIPLIER – published at: "International Journal of Scientific and Research Publications (IJSRP), Volume 3, Issue2, Feburary 2013 Edition".

[7] H. Thapliyal, M. B. Srinivas and H. R. Arabnia , "*Design And Analysis of a VLSI Based High Performance Low Power Parallel quare Architecture*", in Proc. Int. Conf. Algo. Math. Comp. Sc., Las Vegas, June 2005, pp. 72–76.

[8] 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.

[9] H. Thapliyal and M. B. Srinivas, "*High Speed Efficient N × N Bit Parallel Hierarchical Overlay Multiplier Architecture Based on Ancient Indian Vedic Mathematics*", *Enformatika Trans.*, vol. 2, pp. 225-228, Dec. 2004.

[10] J.Bhasker, "*Verilog HDL Primer*" BSP Publishers, 2003.

[11] Himanshu Thapliyal, S. Kotiyal and M.B. Srinivas, "*Design and Analysis of a Novel Parallel Square and Cube Architecture Based on Ancient Indian Vedic Mathematics*", *Proceedings on 48th IIEEE International Midwest Symposium on Circuits and Systems* (MWSCAS 2005),

[12] Himanshu Thapliyal and Hamid R. Arabania, "*A Time- Area – Power Efficient Multiplier and Square Architecture Based on Ancient Indian Vedic Mathematics*", proceedings on VLSI04, Las Vegas, U.S.A, June 2004

[13] Kabiraj Sethi, Rutuparna Panada, "*An Improved Squaring Circuit for Binary Numbers*" , International Journal of Advanced Computer Science and Applications, Vol. 3, No. 2, 2012.