

# VLSI Implementation of a Different Types of Multiplier Unit

**Nitesh Kumar Sharma**

Electronics & Communication

Dr. C.V. Raman Institute of science & Technology Bilaspur, cg India

[sharma786.nitesh@gmail.com](mailto:sharma786.nitesh@gmail.com)

**Mrs. Jigyasa Maru**

Electronics & Communication

C.V. Raman Institute of science & Technology Bilaspur, cg India

[jigyasha.maru21@gmail.com](mailto:jigyasha.maru21@gmail.com)

**Dr. M.R. Khan**

Electronics & Telecommunication

GEC, Jagdalpur, cg India

[mrkhan@cgdterapur.ac.in](mailto:mrkhan@cgdterapur.ac.in)

**Abstract**—This project is primarily deals the construction of 16 bit high speed on Error resilient Multiplier. The motivation behind the investigation is that a multiplier is a very basic building block of Arithmetic Logic Unit (ALU) and would be a limiting factor in performance of Central Processing Unit (CPU). In the past, thorough examination of the algorithms with the respect to particular technology has only been partially done.

**Keywords**— vedic maths; tree; multiplier; HDL.

## I. INTRODUCTION

The merit of the new technology is to be evaluated by its ability to efficiently implement the computational algorithms. In the other words, the technology is developed with the aim to efficiently serve the computation. The reverse path; evaluating the merit of the algorithms should also be taken. Therefore, it is important to develop computational structures that fit well into the execution model of the processor and are optimized for the current technology. In such a case, optimization of the algorithms is performed globally across the critical path of its implementation. In this research article, we will present fast 16 bit multiplier with some approximation technique which is used in arithmetic application. For application analysis I am using 2D Gaussian smooth filter, where I am using my proposed multiplier. Using this application I will prove that proposed multiplier having very less error which is accepted by human eye. This project is design on Xilinx-14.1 and simulated on Modelsim. Application analysis will be done on Matlab for the application of 2D Gaussian smooth filter. Image quality analysis will be done by PSNR, SSIM.

### A. Challenges in the world of computer

There are two important challenges that the world of computing is facing. Currently, the first challenge is due to the increasingly ubiquitous nature of the presentday portable electronics ranging from mobile phones to GPS-based navigation devices. Portability demands lower energy consumption without compromising on the functionality. Also, demand for low energy consuming, also referred to as green design, electronics [1] is gaining a lot of momentum.

The second challenge is manufacturing reliable and predictable electronic devices. Moore's Law predicts that the number of transistors on a single die is going to increase at an exponential rate. This has been accomplished by decreasing

the size of an individual transistor up to 20nm where particular layers such as the gate oxide layer are about 1.2 nm (equivalent to 5 atoms!). But engineering considerations on lithography have limitations of designing these tiny elements precisely which leads to hindrances like thermal noise, parametric variations and other device perturbations which leads to unreliable computing. Again the 2008 ITRS report states as a long term challenge "Dealing with fluctuations and statistical process variations". Also the report mentions that "Increasing yield loss due to non-visual defects and process variations requires new approaches in methodologies, diagnostics and control". [2]

### B. Need of Multiplier Architecture

The core of every microprocessor, digital signal processor (DSP), and data processing application-specific integrated circuit (ASIC) is its data path. It is often the crucial circuit component if die area, power dissipation, and especially operation speed are of concern. At the core of data-path and addressing units in turn are arithmetic units, such as comparators, adders, and multipliers. Finally, the basic operation found in most arithmetic components is the binary addition. Besides of the simple addition of two numbers, adders are also used in more complex operations like multiplication and division. But also simpler operations like incrementing and magnitude comparison base on binary addition. Therefore, binary addition is the most important arithmetic operation. It is also a very critical one if implemented in hardware because it involves an expensive carry-propagation step, the evaluation time of which is dependent on the operand word length. The efficient implementation of the addition operation in an integrated circuit is a key problem in VLSI design. Productivity in ASIC design is constantly improved by the use of cell based design techniques — such as standard cells, gate arrays, and field programmable gate arrays (FPGA) — and by low- and high-

level hardware synthesis. This asks for multiplier architectures which result in efficient cell-based circuit realizations which can easily be synthesized. Furthermore, they should provide enough flexibility in order to accommodate custom timing and area constraints as well as to allow the implementation of customized multiplier. There is lots of application in image processing where we use multiplier as a core architecture [13, 14, 15, 16, 17].

## II. PREVIOUS WORK

Inexact circuit design is a design philosophy where the conventional constraint of requiring 100% accuracy in circuits is relaxed. Fundamentally, this philosophy adds a fourth dimension of accuracy to the current 3-dimensional circuit design space spanning around power consumption, area and delay. This methodology is applicable in the following two situations.

□ The first situation is where the circuits are inherently "unreliable" and "probabilistic". Increasing parameter variations, noise susceptibility and decreasing process sizes are causing CMOS devices to be non-deterministic. To address these issues and precisely model the effect of these probabilistic circuit elements, the metric of accuracy needs to be introduced into the entire circuit design framework.

□ The second situation is where the circuits themselves are not probabilistic in nature but are deterministic, but the application does not demand 100% accuracy. In such cases, relaxing the very rigid constraint of accuracy can be used to decrease energy consumption which is one of the leading challenges in current day circuit design[2, 18].

There are many researchers provide a many types of Accurate Vedic & Multipliers.

### A. Tree Multiplier

The tree multiplier reduces the time for the accumulation of partial products by adding all of them in parallel, whereas the array multiplier adds each partial product in series. The tree multiplier commonly uses CSAs to accumulate the partial products.

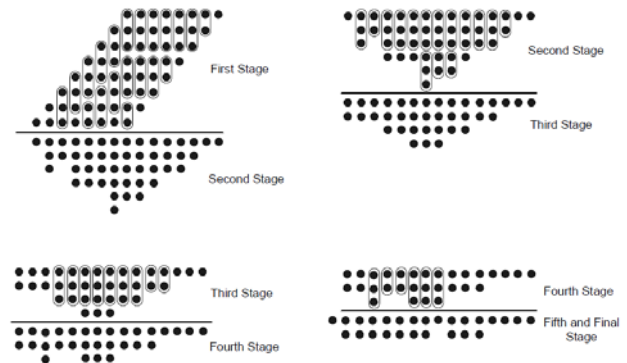
#### 1. AN EFFICIENT HIGH SPEED WALLACE TREE MULTIPLIER [7]:

In this paper author use the existing Wallace tree approach but here the are using carry select adders which will reduce the latency of previous existing approach. A Wallace tree multiplier is an improved version of tree based multiplier architecture. It uses carry save addition algorithm to reduce the latency. This paper aims at further reduction of the latency and power consumption of the Wallace tree multiplier. This is accomplished by the use of 4:2, 5:2 compressors and a proposed carry select adder.

#### 2. COMPARATIVE ANALYSIS FOR HARDWARE CIRCUIT ARCHITECTURE OF WALLACE TREE MULTIPLIER [8]:

The reduction of partial products using full adders as carry-save adders (also called 3:2 counters) became generally

known as the Wallace Tree". According to this paper author used existing Wallace tree method but the use the small size full adder which will reduce the hardware consumption. This architecture reduces the partial products at a rate of  $\log_3 2$  ( $N/2$ ). Figure shows an example of tree reduction for an 8\*8-bit partial product tree. The ovals around the dots represent either a full adder (for three circled dots) or a half adder (for two circled dots). This tree is reduced to two rows for a carry-propagate adder after four stages. There are many ways to reduce this tree with CSAs, and this example is just one of them.



### B. VEDIC MULTIPLICATION ALGORITHM

Vedic mathematics is part of four Vedas (books of wisdom). It is part of Sthapatya- Veda (book on civil engineering and architecture), which is an upa-veda (supplement) of Atharva Veda. It covers explanation of several modern mathematical terms including arithmetic, geometry (plane, co-ordinate), trigonometry, quadratic equations, factorization and even calculus. His Holiness JagadguruShankaracharyaBharati Krishna Teerthaji Maharaja (1884-1960) comprised all this work together and gave its mathematical explanation while discussing it for various applications. Swahiji constructed 16 sutras (formulae) and 16 Upa sutras (sub formulae) after extensive research in Atharva Veda. Obviously these formulae are not to be found in present text of Atharva Veda because these formulae were constructed by Swamiji himself. Vedic mathematics is not only a mathematical wonder but also it is logical. That's why VM has such a degree of eminence which cannot be disapproved. Due these phenomenal characteristic, VM has already crossed the boundaries of India and has become a leading topic of research abroad. VM deals with several basic aswell as complex mathematical operations. Especially, methods of basic arithmetic are extremely simple and powerful [4].

The word „Vedic“ is derived from the word „veda“ which means the store-house of all knowledge. Vedic mathematics is mainly based on 16 Sutras (or aphorisms) dealing with

various branches of mathematics like arithmetic, algebra, geometry etc. These Sutras along with their brief meanings are enlisted below alphabetically [4]

- 1) (Anurupy) Shunyamanyat – If one is in ratio, the other is zero.
- 2) Chalana-Kalanabyham – Differences and Similarities.
- 3) EkadhikinaPurvena – By one more than the previous One.
- 4) EkanyunenaPurvena – By one less than the previous one.
- 5) Gunakasamuchyah – The factors of the sum is equal to the sum of the factors.
- 6) Gunitasamuchyah – The product of the sum is equal to the sum of the product.
- 7) NikhilamNavatashcaramamDashatah – All from 9 and last from 10.
- 8) ParaavartyaYojayet – Transpose and adjust.
- 9) Puranapurabyham – By the completion or noncompletion.
- 10) Sankalana- vyavakalanabyham – By addition and by subtraction.
- 11) ShesanyankenaCharamena – The remainders by the last digit.
- 12) ShunyamSaamyaSamuccaye – When the sum is the same that sum is zero.
- 13) Sopaantyadvayamantyam – The ultimate and twice the penultimate.
- 14) Urdhva-tiryakbhyam – Vertically and crosswise.
- 15) Vyashtisamanstih – Part and Whole.
- 16) Yaavadunam – Whatever the extent of its deficiency.

These methods and ideas can be directly applied to trigonometry, plain and spherical geometry, conics, calculus (both differential and integral), and applied mathematics of various kinds. As mentioned earlier, all these Sutras were reconstructed from ancient Vedic texts early in the last century. Many Sub-sutras were also discovered at the same time, which are not discussed here.

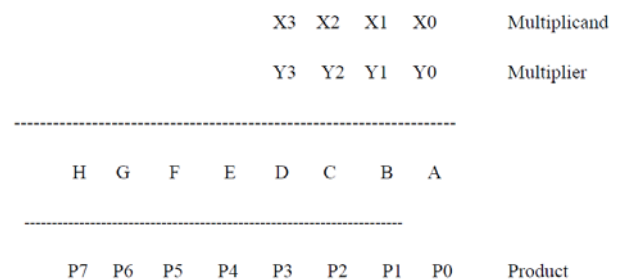
C. Multiplier design based on ancient Indian Vedic Mathematics [4]

Vedic mathematics is the name given to the ancient Indian system of mathematics that was rediscovered in the early twentieth century from ancient Indian sculptures (Vedas). It mainly deals with Vedic mathematical formulae and their application to various branches of mathematics. The algorithms based on conventional mathematics can be simplified and even optimized by the use of Vedic Sutras. These methods and ideas can be directly applied to trigonometry, plain and spherical geometry, conics, calculus (both differential and integral), and applied mathematics of various kinds. In this paper new multiplier and square architecture is proposed based on algorithm of ancient Indian Vedic Mathematics, for low power and high speed applications.

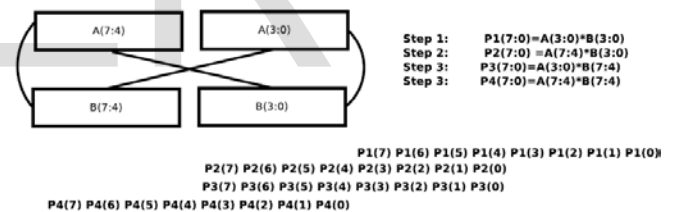
Urdhvatiyakybhyam Sutra is a general multiplication formula applicable to all cases of multiplication. It literally means “Vertically andCrosswise”. The conventional methods already know to us will require 16 multiplications and 15

additions. An alternative method of multiplication using Urdhvatiyakybhyam Sutra is shown in below Fig. 1. The numbers to be multiplied are written on two consecutive sides of the square 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. Thus, each digit of the multiplier has a small box common to a digit of the multiplicand. These small boxes are partitioned into two halves by the crosswise lines. Each digit of the multiplier is then independently multiplied with every digit of the multiplicand and the two-digit product is written in the common box. All the digits lying on a crosswise dotted line are added to the previous 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.

CP = Cross Product (Vertically and Crosswise)



Algorithm for 4 x 4 bit Vedic multiplier Using UrdhvaTiryakbhyam (Vertically and crosswise) for two binary number.



Algorithm for 8 X 8 Bit Multiplication Using UrdhvaTiryakbhyam (Vertically and crosswise) for two Binary numbers

III. FUTURE OBJECTIVE

According to previous existing design, there is lots of issues in terms of power, area, and speed. In this research area, still there is lots of future work is require. The main focus is on performance and accuracy, but we do provide some numbers for the arithmetic units relating to energy and power. This is to provide an estimate of the amount of energy and power consumed by the units we choose to implement. The priorities of the future research objectives are, in order of importance, are:

- 1) Robust and safe circuits.
- 2) Design time
- 3) Area/speed balance

#### IV. CONCLUSION

As we know in present era everyone need fast applications which is basically based on computation. If computation unit require heavy amount of time so automatically overall application will require large amount of power and delay. Due to large power there is need of large battery unit. So according to this paper we study about the previous existing techniques of different kind of multiplier, because multiplier is a main computation unit for any kind of processor unit. According to our study we find there is lots of issues are there which can still a open box for further research area.

#### References

- [1] Garcia, Oscar N., Harvey Glass, and Stanley C. Haimes. "An approximate and empirical study of the distribution of adder inputs and maximum carry length propagation." *Computer Arithmetic (ARITH)*, 1978 IEEE 4th Symposium on. IEEE, 1978.
- [2] Cho, Hyungmin, Larkhoon Leem, and Subhasish Mitra. "ERSA: Error resilient system architecture for probabilistic applications." *Computer-Aided Design of Integrated Circuits and Systems*, IEEE Transactions on 31.4 (2012): 546-558.
- [3] Palem, Krishna V., et al. "Sustaining moore's law in embedded computing through probabilistic and approximate design: retrospects and prospects." *Proceedings of the 2009 international conference on Compilers, architecture, and synthesis for embedded systems*. ACM, 2009.
- [4] Tiwari, Honey Durga, et al. "Multiplier design based on ancient Indian Vedic Mathematics." *SoC Design Conference, 2008. ISOC'08. International*. Vol. 2. IEEE, 2008.
- [5] Lomte, Rashmi K., and P. C. Bhaskar. "High Speed Convolution and Deconvolution Using Urdhva Triyagbhyam." *VLSI (ISVLSI)*, 2011 IEEE Computer Society Annual Symposium on. IEEE, 2011.
- [6] Saokar, Sandesh S., R. M. Banakar, and Saroja Siddamal. "High speed signed multiplier for digital signal processing applications." *Signal Processing, Computing and Control (ISPCC)*, 2012 IEEE International Conference on. IEEE, 2012.
- [7] Sureka, N., R. Porselvi, and K. Kumuthapriya. "An efficient high speed Wallace tree Multiplier." *Information Communication and Embedded Systems (ICICES)*, 2013 International Conference on. IEEE, 2013.
- [8] Gandhi, Dhaval R., and Nehal N. Shah. "Comparative analysis for hardware circuit architecture of Wallace tree multiplier." *Intelligent Systems and Signal Processing (ISSP)*, 2013 International Conference on. IEEE, 2013.
- [9] Kirubaveni, S. "Performance evaluation of FFT processor using conventional and Vedic algorithm." *Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN)*, 2013 International Conference on. IEEE, 2013.
- [10] Itawadiya, Akhalesh K., et al. "Design a DSP operations using vedic mathematics." *Communications and Signal Processing (ICCSP)*, 2013 International Conference on. IEEE, 2013.
- [11] Kulkarni, Parag, Puneet Gupta, and Milos Ercegovac. "Trading accuracy for power with an underdesigned multiplier architecture." *VLSI Design (VLSI Design)*, 2011 24th International Conference on. IEEE, 2011.
- [12] Stine, James E. *Digital computer arithmetic datapath design using verilog HDL*. Springer Science & Business Media, 2012.
- [13] Canny, John. "A computational approach to edge detection." *Pattern Analysis and Machine Intelligence*, IEEE Transactions on 6 (1986): 679-698.
- [14] Marr, David, and Ellen Hildreth. "Theory of edge detection." *Proceedings of the Royal Society of London B: Biological Sciences* 207.1167 (1980): 187-217.
- [15] Unser, Michael, and Murray Eden. "Multiresolution feature extraction and selection for texture segmentation." *Pattern Analysis and Machine Intelligence*, IEEE Transactions on 11.7 (1989): 717-728.
- [16] Reinhard, Erik, et al. "Photographic tone reproduction for digital images." *ACM Transactions on Graphics (TOG)*. Vol. 21. No. 3. ACM, 2002.
- [17] Guo, Zhengyang, Wenbo Xu, and Zhilei Chai. "Image edge detection based on FPGA." *Distributed Computing and Applications to Business Engineering and Science (DCABES)*, 2010 Ninth International Symposium on. IEEE, 2010.
- [18] Lee, Kyoungwoo, et al. "Error-exploiting video encoder to extend energy/qos tradeoffs for mobile embedded systems." *Distributed Embedded Systems: Design, Middleware and*
- [19] G. Eason, B. Noble, and I.N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *Phil. Trans. Roy. Soc. London*, vol. A247, pp. 529-551, April 1955. (references)
- [20] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
- [21] I.S. Jacobs and C.P. Bean, "Fine particles, thin films and exchange anisotropy," in *Magnetism*, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.
- [22] K. Elissa, "Title of paper if known," unpublished.
- [23] R. Nicole, "Title of paper with only first word capitalized," *J. Name Stand. Abbrev.*, in press.
- [24] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
- [25] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.

# IJSER