

Design And Implementation Of 64 Bit High Speed Vedic Multiplier For DSP Applications

¹Pooja Krishnamurthy Revankar , ²Dr. H C Hadimani

¹Research Scholar, Department of E & C, G M Institute of Technology, Davangere, India.

²Professor, Department of E & C, G M Institute of Technology, Davangere, India.

Abstract

A multiplier is one of the key hardware blocks in most digital signal processing (DSP) systems. Typical DSP applications where a multiplier plays an important role include digital filtering, digital communications and spectral analysis. Many current DSP applications are targeted at portable, battery-operated systems, so that power dissipation becomes one of the primary design constraints. Since multipliers are rather complex circuits and must typically operate at a high system clock rate, reducing the delay of a multiplier is an essential part of satisfying the overall design. This paper implementation forward a high speed Vedic multiplier, which is efficient in terms of speed, making use of Urdhva Tiryagbhyam, a sutra from Vedic Maths for multiplication of partial products. The most important part of this paper is to reduce the power utilization and give high speed. In these work 64 bit multiplications calculations are performed. The synthesized results are implemented on Genus and Innovus tool GDSII files Cadence tools utilizing 45nm innovation of technology. The recreated comes about for proposed 64 bit Vedic multiplier demonstrates a optimized in area and power utilization against other multiplication techniques.

The code is written in Verilog and results shows that multiplier implemented using Vedic multiplication is efficient in terms of area, power and speed compared to its implementation using Array and Booth multiplier architectures.

Keywords: Vedic Multiplier, Genus, Innovus, cadence, GDSII file

1. Introduction

Multiplication is an important fundamental function in arithmetic operations. Multiplication-based operations such as Multiply and Accumulate(MAC) and inner product are among some of the frequently used computation Intensive Arithmetic Functions(CIAF) currently implemented in many Digital Signal Processing (DSP) applications such as convolution, Fast Fourier

Transform(FFT), filtering and in microprocessors in its arithmetic and logic unit. Multiplication can be implemented using several algorithms such as: array, Booth, modified Booth algorithms. Array multiplier is well known due to its regular structure. Multiplier circuit is based on add and shift algorithm. Each partial product is generated by the multiplication of the multiplicand with one multiplier bit. The partial product are shifted according to their bit orders and then added. Booth Multipliers is a powerful algorithm for signed-number multiplication, which treats both positive and negative numbers uniformly. This method that will reduce the number of multiplicand multiples. For a given range of numbers to be represented, a higher representation radix leads to fewer digits.

The partial-sum adders can also be rearranged in a tree like fashion, reducing both the critical path and the number of adder cells needed. The presented structure is called the Wallace tree multiplier the tree multiplier realizes substantial hardware savings for larger multipliers. The propagation delay is reduced as well. In fact, it can be shown that the propagation delay through the tree is equal to $O(\log_{3/2}(N))$. While substantially faster than the carry-save structure for large multiplier word lengths, the Wallace multiplier has the disadvantage of being vary irregular, which complicates the task of an efficient layout design.

2. Literature survey

Rapidly growing technology has raised demands for fast and efficient real time digital signal processing applications. Multiplication is one of the primary arithmetic operations every application demands. A large number of multiplier designs have been developed to enhance their speed. Active research over decades has lead to the emergence of Vedic Multipliers as one of the fastest and low power multiplier over traditional array and booth multipliers.

Honey DurgaTiwari.et.alltalked about designing a multiplier and square architecture is based on algorithm of ancient Indian Vedic Mathematics, for low power and high speed applications. They explained Urdhvatiyakybhyam and Nikhilam algorithm and found that Urdhvatiyakybhyam, is applicable to all cases of multiplication but due to its structure, it suffers from a high carry propagation delay in case of multiplication of large numbers. This problem has been solved by introducing Nikhilam Sutra which reduces the multiplication of two large numbers to the multiplication of two small numbers.

Prof J M Rudagil.et. all designed a multiplier using Vedic mathematics. They explained Urdhvatiyakybhyam and found that it is efficient Vedic multiplier with high speed, low power and consuming little bit wide area was designed. It was also found that the multiplier based on Vedic sutras had execution delay of almost half of that of binary multiplier.

P Manju.et. all presented a technique that modifies the architecture of the Vedic multiplier by using some existing methods in order to reduce power. They explained Nikhilam sutra and double base number system. Nikhilam sutra method is not valid for negative numbers. They found that Vedic

Multiplier without any Modification has high power consumption. Vedic Multiplier with modified Two's complement block has less power consumption with cost of delay and area.

3. Methodology

A. Methods for multiplication

There are number of techniques that can be used to perform multiplication. In general, the choice is based upon factors such as latency, throughput, area, and design complexity.

a) Array Multiplier b) Booth Multiplier

Booth's multiplication algorithm is a multiplication algorithm that multiplies two signed binary numbers in two's complement notation. The algorithm was invented by Andrew Donald Booth.

1) Booth Multiplier

Conventional array multipliers, like the Braun multiplier and Baugh Woolley multiplier achieve comparatively good performance but they require large area of silicon, unlike the add-shift algorithms, which require less hardware and exhibit poorer performance. The Booth multiplier makes use of Booth encoding algorithm in order to reduce the number of partial products by considering two bits of the multiplier at a time, thereby achieving a speed advantage over other multiplier architectures. This algorithm is valid for both signed and unsigned numbers. It accepts the number in 2's complement form.

2) Array Multiplier

Array multiplier is an efficient layout of a combinational multiplier. Multiplication of two binary number can be obtained with one micro-operation by using a combinational circuit that forms the product bit all at once thus making it a fast way of multiplying two numbers since only delay is the time for the signals to propagate through the gates that forms the multiplication array. In array multiplier, consider two binary numbers A and B, of m and n bits. There are mn summands that are produced in parallel by a set of mn AND gates. n x n multiplier requires n(n-2) full adders, n half-adders and n² AND gates. Also, in array multiplier worst case delay would be (2n+1) td.

B. Vedic Multiplication (Proposed work)

The proposed design uses Vedic mathematics based on Urdhva Tiryakabhyam sutra for the multiplication of the mantissa part in IEEE 754 single precision floating point multiplication. In this proposed multiplier the base block used as first stage implementation is 3*3 blocks which is shown in figure 1. Here we needs 64b it Vedic multiplier for the multiplication of mantissa part. The 3*3 b lock consists of two half adders, one full adder and three 2 bit adders as shown in figure 1. From this 3*3 block, 6*6 multiplier blocks is designed. From this 6*6 multiplier block, 12*12

multiplier blocks is designed and similarly from this 24*24 multiplier block, 64*64 multiplier block is designed and implemented. These blocks require Vedic multipliers and ripple carry adders for getting the final output.

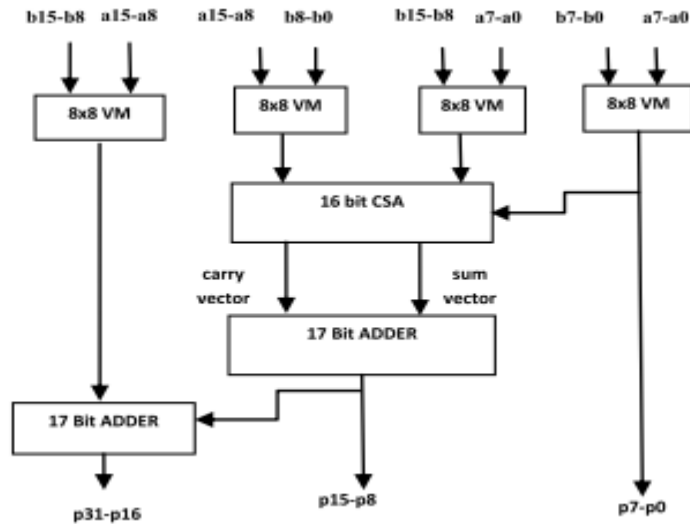


Figure 1: Implementation of 16x16 Bits Vedic Multiplier

The 16X16 bit multiplier structured using 8X8 bits blocks as shown in Figure 2. In this Figure the 16 bit multiplicand A can be decomposed into pair of 8 bits AH-AL. Similarly multiplicand B can be decomposed into BH-BL. The outputs of 8X8 bit multipliers are added accordingly to obtain the 32 bits final product. Thus, in the final stage two adders are also required.

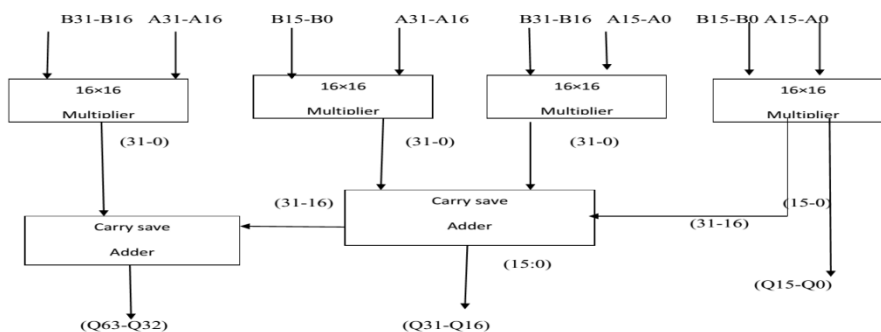


Figure 2: Implementation of 32x32 Bits Vedic Multiplier

The 32 bits multiplicand A is decomposed into pair of 16 bits AH-AL. Similarly multiplicand B can be decomposed into BH-BL. The outputs of 16X16 bit multipliers are added accordingly to obtain the 64 bits final product. Thus, in the final stage two adders are also required.

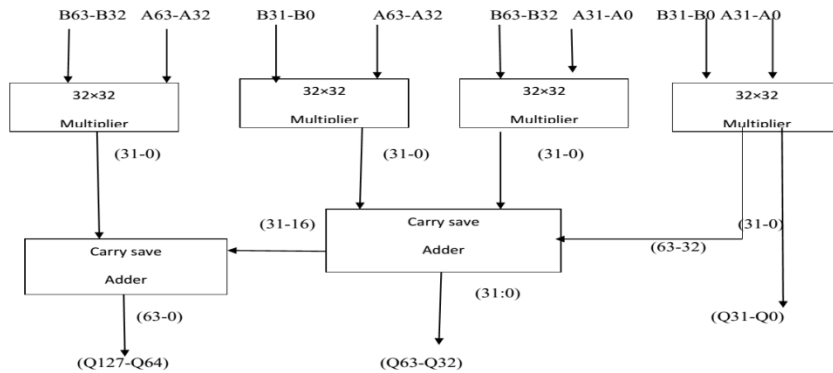


Figure 3: Implementation of 64x64 Bits Vedic Multiplier.

The 64 bits multiplicand A is decomposed into pair of 32 bits AH-AL. Similarly multiplicand B can be decomposed into BH-BL. The outputs of 32X32 bit multipliers are added accordingly to obtain the 128 bits final product. Thus, in the final stage two adders are also required.

3. Simulation results

Above Fig. 5 shows the simulation result of High Speed and Low Power three operand adder (HS and LP 3 operand adder).

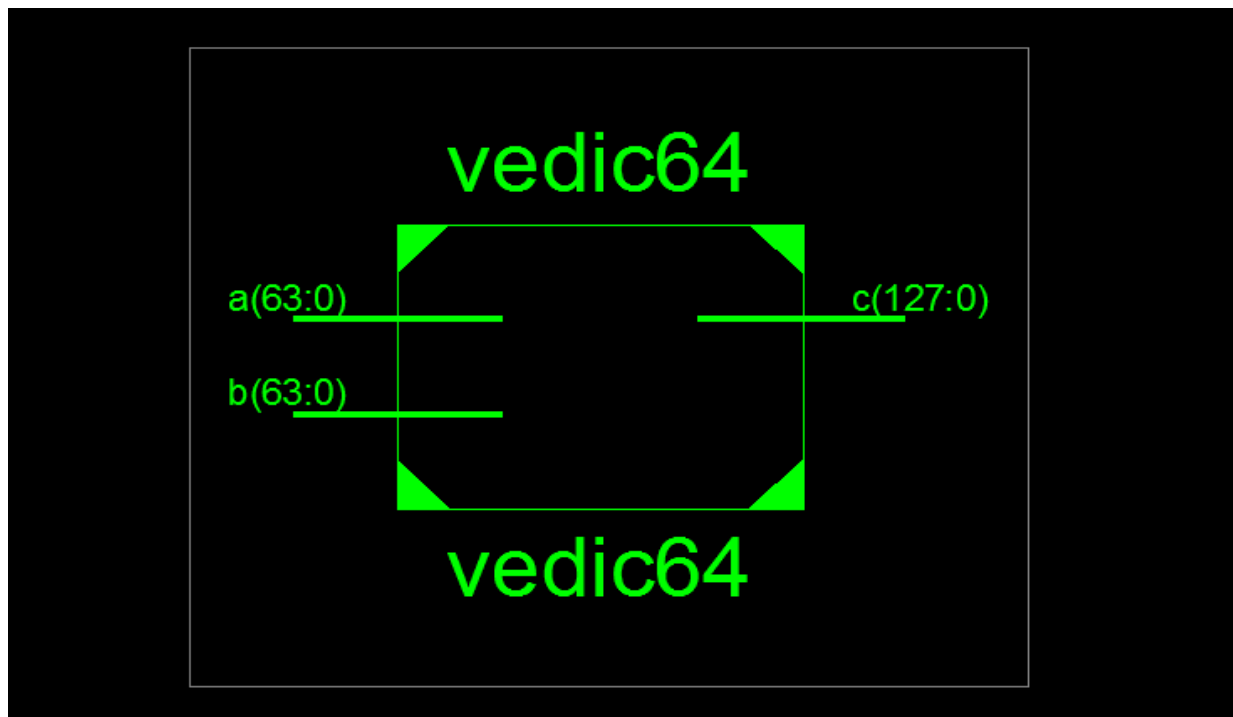


Figure 4: RTL View of 32-Bit HC binary adder



Figure 5: Simulation Result of ‘64 bit Vedic Multiplier

4. Synthesis and Simulation Results

For the fair comparison, the same coding style using Verilog HDL using the Xilinx 14.7 ISE tool is adopted for designing the CS3A and HC3A and the proposed three-operand adders. Further, all these designs are synthesized using Synopsys Design Compiler in same SAED 32nm CMOS technology library to obtain the core area, timing and power for different word size. The physical synthesis analysis metrics comprised of maximum combinational gate delay, core area, power consumption, area-delay-product (ADP) and power-delay-product (PDP) are proved. The estimated results are shown shall vary with adopted verilog coding style and optimization options available in Genus tool.

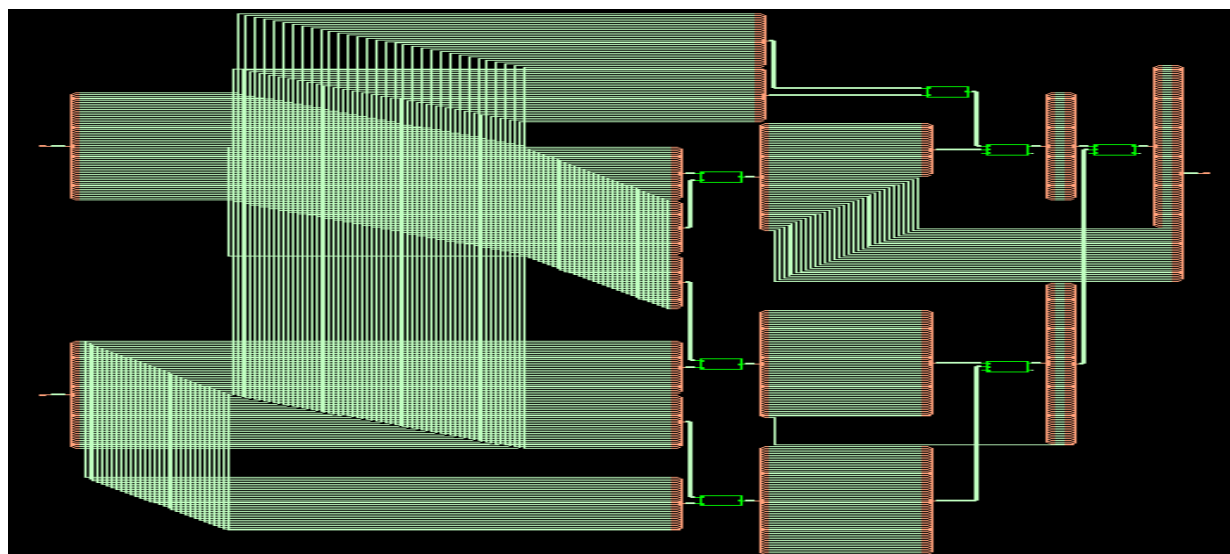


Figure 6: Synthesis Layout

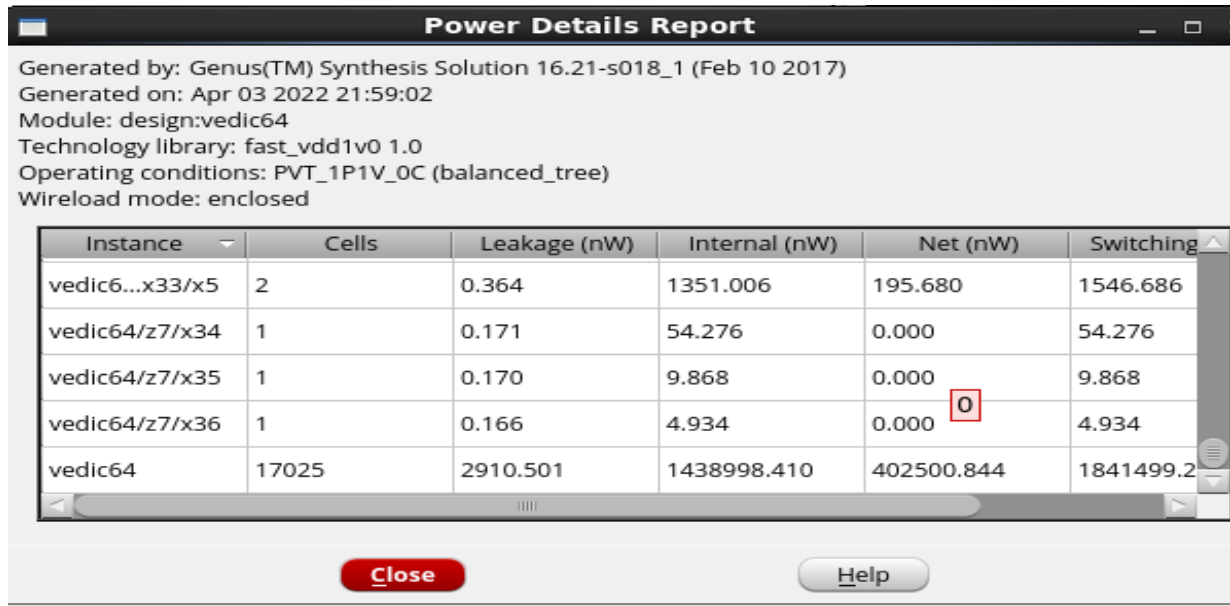


Figure 7: detailed report of power

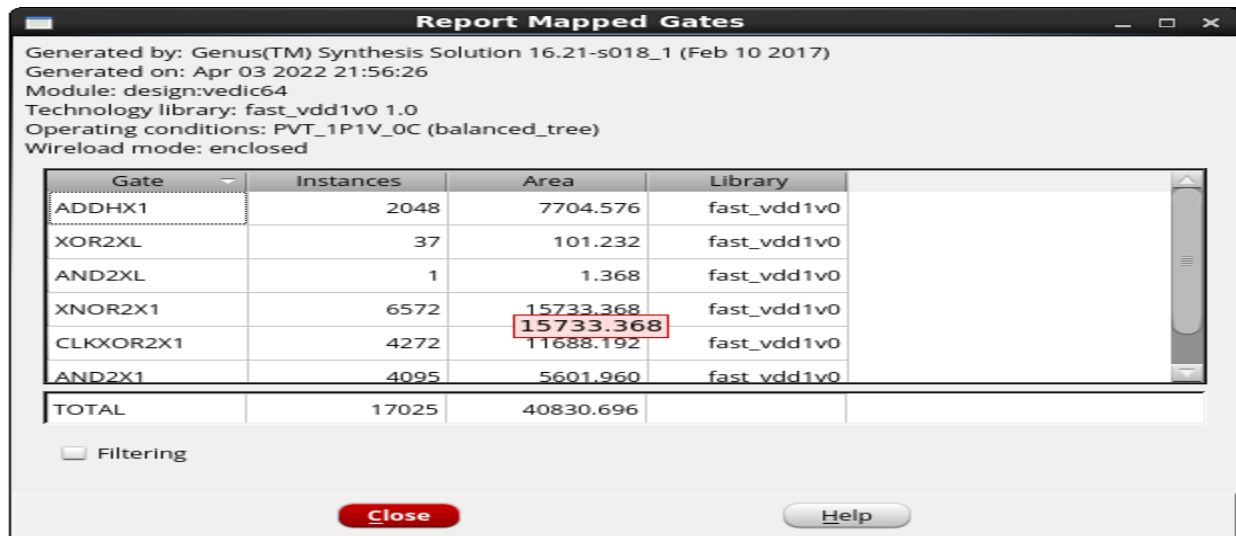


Figure 8: Detailed report of area

5. Physical layout

64 bit length input sequence is taken for the implementation of the described High Speed and Low Power Vedic Multiplication and it is clear that described method acquires less power, less area and less delay which automatically increases the speed, which is suitable for DSP applications. ADP and PDP plots are represented for 64 bit input sequence in Fig. 7 and Fig. 8 respectively. RTL View of 64-Bit Vedic multiplier is shown in below Fig. 9.

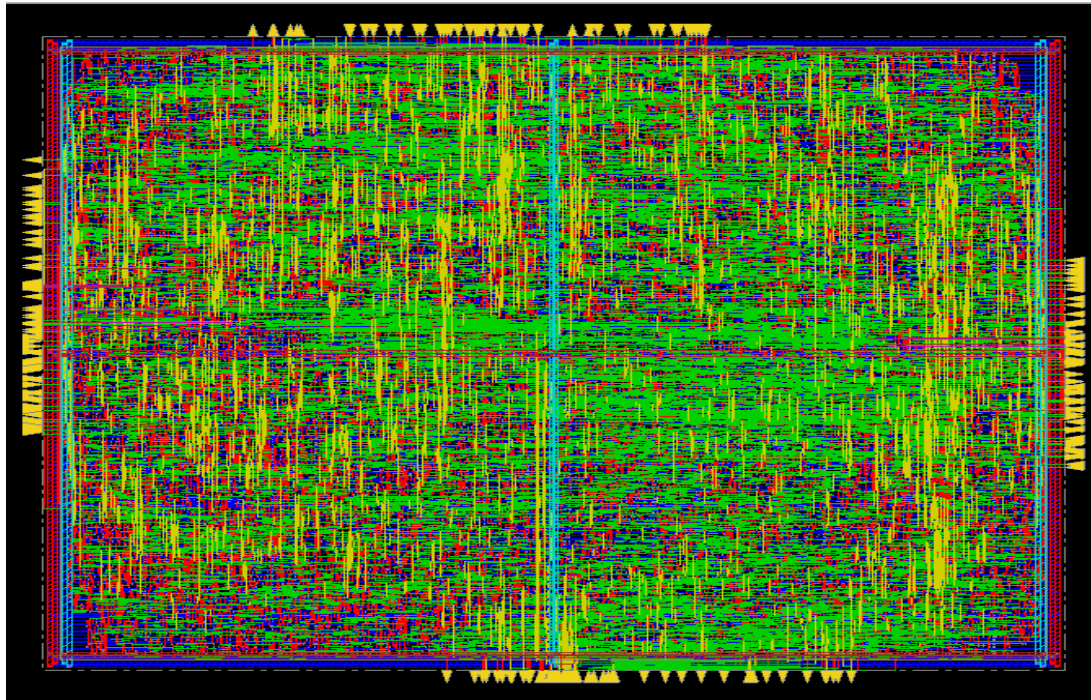


Figure 9: Physical Implementation

6. CONCLUSION

In this paper, 64 bit of Vedic multiplier of High Speed and Low Power VLSI Architecture is implemented. The proposed Vedic technique is a parallel prefix adder that uses different stages structures to compute the multiplication of input operands. For the fair comparison, the same coding style using Verilog HDL using the Xilinx 14.7 ISE tool is adopted for designing the logic blocks and the proposed Vedic Multiplication. The novelty of this proposed architecture is the reduction of power, delay and area in the prefix computation stages in PG logic and bit-addition logic that leads to an overall reduction in area-delay product (ADP) and power-delay product (PDP). From the physical synthesis results, this is clear that the proposed Vedic multiplier architecture is 4 to 10 times faster than the corresponding booth architecture. Concluding that, our Vedic Multiplication was comparatively better than other existing multiplication in terms of the power, area and delay.

References

- [1] Honey Durga Tiwari, Ganzorig Gankhuyag, Chan Mo Kim, Yong Beom Cho, "Multiplier design based on ancient Indian Vedic Mathematics," in IEEE International SoC Design Conference, pp. II-65 - II-68, November 2008.
- [2] Prof J M Rudagil, Vishwanath Amble, Vishwanath Munavalli, "Design and implementation of efficient multiplier using vedic mathematics", Proc. of Int. Con/, on

- Advances in Recent Technologies in Communication and Computing, pp.162-166, Nov 2011.
- [3] P. Manju, P.B. Sreelakshmi, T. Madhavi, "Design of Vedic Multiplier Using GDI method", International Journal of Recent Technology and Engineering (IJRTE), Vol. 7, Issue 6S4, April 2019
- [4] A.D.Booth, "A Signed Binary Multiplication Technique," J. mech. and appl. math, vol 4,no.2, pp. 236-240, Oxford University Press, 1951.
- [5] C. R. Baugh, B. A. Wooley, "A Two's Complement Parallel Array Multiplication Algorithm," IEEE Trans. Computers 22(12), pp. 1045–1047, 1973.
- [6] Koren Israel, "Computer Arithmetic Algorithms," 2ndEd, pp. 141-149, Universities Press,2001.