Proceedings of the Fourth International Conference on Computing Methodologies and Communication (ICCMC 2020)
IEEE Xplore Part Number:CFP20K25-ART; ISBN:978-1-7281-4889-2

Vedic Multiplier in 45nm Technology

Chiranjit R Patel*, Vivek Urankar*, Vivek B A*, V Keshav Bharadwaj*
*Electronics and Communication,
RNS Institute of Technology

Email: chiranjitpatel08@gmail.com (Chiranjit R Patel)
Email: vivekurankar@gmail.com (Vivek Urankar)
Email: vivek.adishesha2gmail.com (Vivek B A)
Email: keshavbharadwaj@gmail.com (V Keshav Bharadwaj)
Corresponding authors: Vivek B A, V Keshav Bharadwaj

Abstract—Multipliers in a digital processor remains as a core of mathematical computing paradigm. In ancient times Vedic mathematicians developed basic multiplication algorithms. This study focuses on optimizing area and designing the multiplier in 45 nanometer CMOS technology. Layout design and verification of a 4-bit multiplier is carried out. Operating voltage ranges from 0.9V to 1.1V, this aids in low power operation or the multiplier. Consuming 3.795uW of power in the highest constraint situation. "Layout Versus Schematic" and "Design Rule Check" (LVS & DRC) are the two software verification tools used to verify the integrated circuit design. Delay and power analysis of the multiplier using Cadence virtuoso manager are discussed. Delay of the proposed 4-bit multiplier in 45nm CMOS technology multiplier is 250ps by including all constraints.

Keywords: Vedic multiplier, Vedic mathematics, Urdhva Tiryagbhyam, CMOS, 45 nanometer, CSA (Carry Save Adder).

I. INTRODUCTION

Development of portable embedded electronics with digital signal processors for challenging modern era problems, demands to ameliorate the VLSI technology and designs. Multipliers are fundamental components that are required in high performing systems like mixed signal and digital signal processors. Speed increase is achieved by condensing computational steps in calculation process with help of Vedic mathematical algorithms. The past research and documents (Vedic texts) have helped to develop rapid methods to multiply numbers.

This study presents a novel way to implement the Urdhva Tiryagbhyam sutra in VLSI layouts with efficient power rating. Designs created with 45 nanometer CMOS technology using Cadence Virtuoso.

Different adder topologies have been compared in the works of [1]. Based on the results it can be concluded that Carry Save and Carry Select Adders are the fastest among all the topologies. Urdhva Tiryagbhyam is found to be the most effective algorithm for multiplication, giving minimum delay [2]. An 8x8 multiplier using CMOS logic is shown in the works of [3]. Their proposed design consumes 75% less power. Improvement of the existing proposed design can be done using techniques like GDI [8, 11]. Higher order multipliers can be designed and implemented on the present technology. Karatsuba based multiplier design [10] have provided noticeable reduction in delay.

II. VEDIC MULTIPLIER

Vedic math fundamentally rests on sixteen sutras or mathematical formulas. Vedic math is based on these sixteen sutras which deal with various branches of mathematics such as geometry, algebra and arithmetic. Among these sixteen sutras two are dedicated for multiplication. The first is Urdhva Tiryagbhyam and the second is Nikhilam sutra.

A. Urdhva Tiryagbhyam

It is a generic algorithm applicable to every case of multiplication. This sutra is also applicable for dividing large numbers. This sutra is one of the best known Vedic sutras and has found many applications.

In Fig. 1 A3A2A1A0 and B3B2B1B0 are two numbers of 4-bit each that have to be multiplied. For the least significant bit of the result, A0 and B0 are multiplied and stored. For the second least significant bit, the products [A1B0] and [A0B1] are added and stored. Likewise this process is continued for seven steps till the final result is obtained.

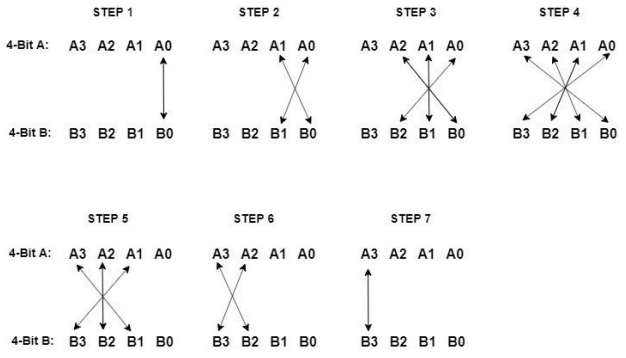

Fig. 1. Urdhva Tiryagbhyam sutra

B. Nikhilam sutra

Nikhilam sutra is based on multiplication of numbers that are closer to the powers of 10. Fig. 2 shows and example of multiplying two numbers with Nikhilam sutra. Nikhilam sutra method is used to multiply or divide numbers closer to the base of 10 in a faster approach.

96x94

Column 1	Column 2
96	(100-4)
94	(100-6)
96	(4)
94	(6)
90 /	24

Result = 9024

Fig. 2. Example using Nikhilam sutra.

III. PREQUISTIES

A. AND gate

Binary multiplication is performed with the help of AND gate. It can have only one output and two or more inputs. The output is latched to VDD when all the inputs applied to the gate are HIGH.

Given a PMOS and NMOS of same W/L ratios, the PMOS will be slower than the NMOS because the mobility of holes is lesser than the mobility of electrons. Therefore, for symmetric switching logic gates the width of the PMOS is made greater than the width of the NMOS for a given fixed length of the channel. Fig. 3 shows the 2-input CMOS AND gate.

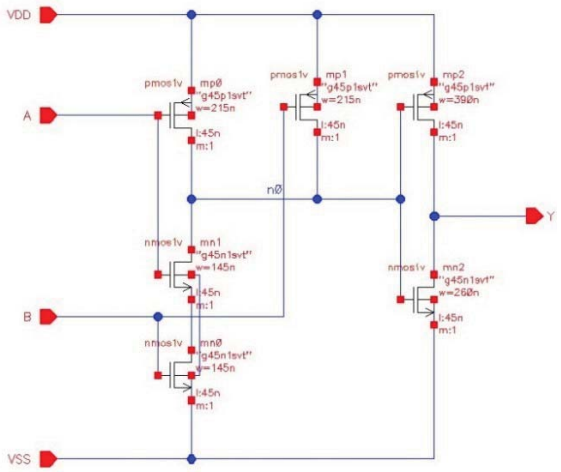

Fig. 3. Schematic of AND gate

B. XOR gate

The output of XOR gate is HIGH only if exactly one of its inputs is HIGH. Fig. 4 shows 2-input CMOS XOR gate. It consists of an XNOR gate in series with an inverter.

C. Half Adder

Addition of two binary bits is done with the help of a half adder. It takes two bits as inputs and produces two output bits,

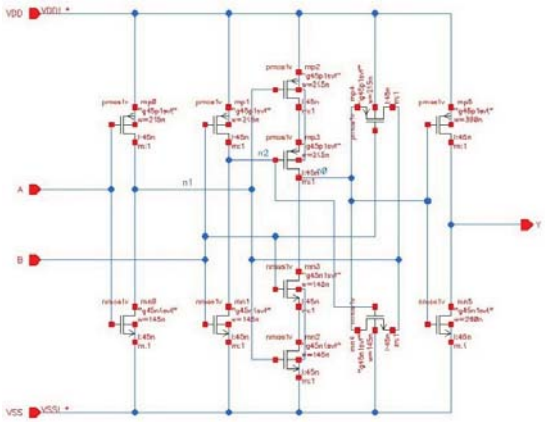

Fig. 4. Schematic of XOR gate

Sum and carry respectively. The sum can be obtained by passing the two inputs through a XOR gate and the carry is obtained by passing the inputs through an AND gate. Fig. 5 shows the CMOS half adder.

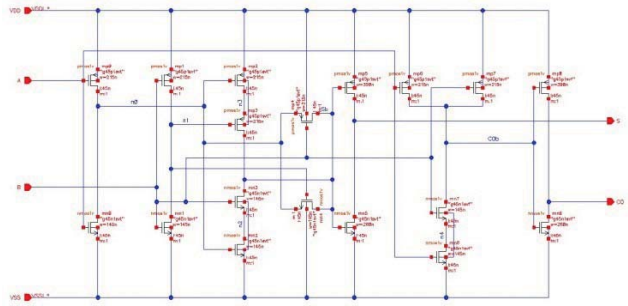

Fig. 5. Schematic of half adder

D. Full adder

Addition of three binary bits is performed using a full adder circuit. Sum and carry are the two outputs generated. Full adder circuit in CMOS 45nm technology is shown is Fig. 6.

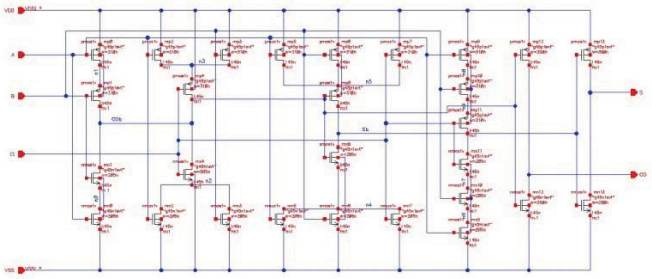

Fig. 6. Schematic of full adder

IV. DESIGN IMPLEMENTATION

A. *Implementation of 2-bit Vedic multiplier*

The implementation of 2-bit multiplier using Urdhva Tiryagbhyam technique is discussed. The two 2-bit numbers is taken as inputs, in binary form A0&A1 and B0&B1.

There are three steps in a 2-bit Vedic multiplier-

1) The LSB bits A0 and B0 are propagated through an AND gate to produce Y0.
2) The product A0B0 and A1B0 are summed using a half adder, whose sum output contributes to Y1. The carry out is transmitted to the next stage.
3) The product A1B1 along with the carry from the previous stage is again added using a half adder. The outputs, sum and carry give Y2 and Y3 respectively.

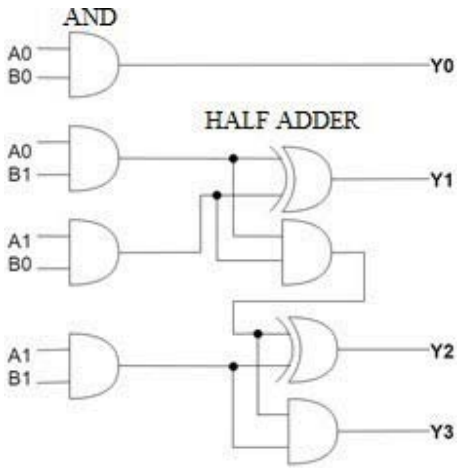

Fig. 7. Realization of 2-bit multiplier using basic logic gates

The same design of the 2x2 multiplier has been realized in cadence virtuoso 45nm technology. The schematic is shown in Fig. 7.

In Fig. 8, we have used bottom up approach to realize the 2-bit multiplier. Symbols of all individual blocks are created and instantiated to form the higher level blocks. VDD and VSS are the two power rails, and the circuit is made to operate at a voltage of 0.9V to 1.1V. Buffers can be used wherever necessary to avoid signal degradation and to ensure a greater fan-out and full rail to rail swing.

Fig. 9 shows the layout of 2-bit multiplier. Areas occupied by the 2-bit multiplier with and without buffers are 21.88μm^2 and 15.08μm^2 respectively. Crosstalk between adjacent metal lines is minimized by proper shielding. The floor plan of the 2-bit multiplier is designed to occupy the least area. Tap cell is used to ensure there are no stamp errors.

B. *Carry Save Adder(CSA)*

CSA is used to add 3-bit or more than 3-bit numbers, also to add up to three numbers at an instant, consists of a set of full

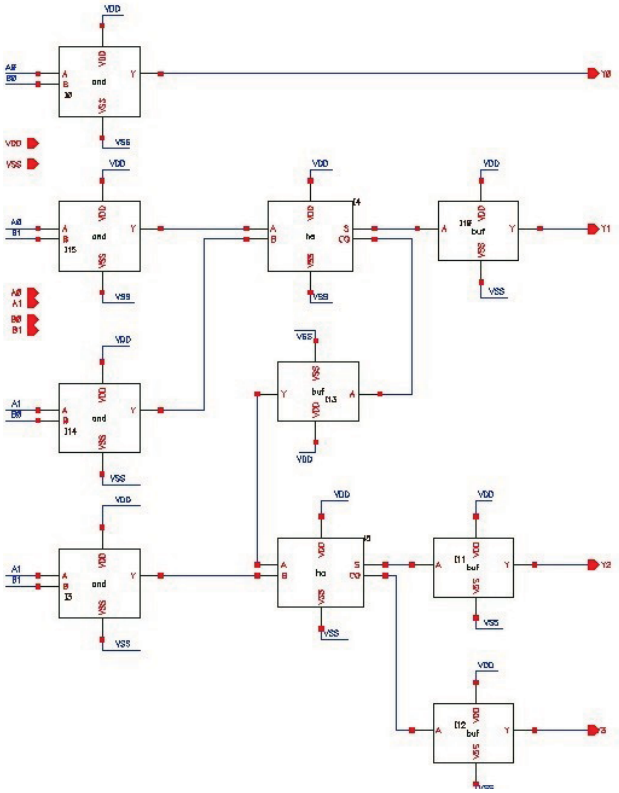

Fig. 8. Realization of 2 bit multiplier using Cadence Virtuoso 45nm technology

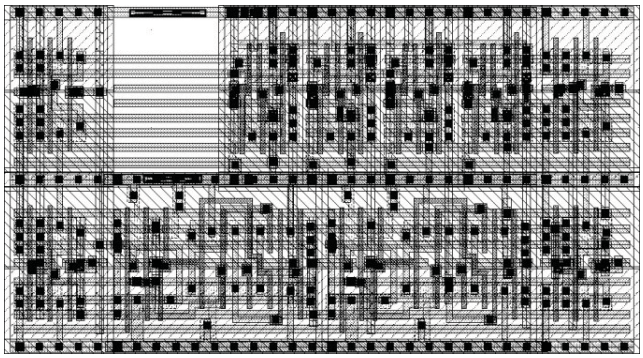

Fig. 9. Layout of 2-bit multiplier with buffers

adders and half adders. The carry out of one stage is the input to the next stage. Major advantage of carry save adder is the delay, when compared to other adders it reduces drastically. Irrespective of the number of bits, the propagation delay of CSA is three gates.

This design of CSA is emulated and modified to suit 4-Bit multiplication. When the carry of the multiplication output is to be added we get two numbers of 4-bit each. The design is refined to accommodate and reduce the use of more gates. Fig. 10 and Fig. 11.

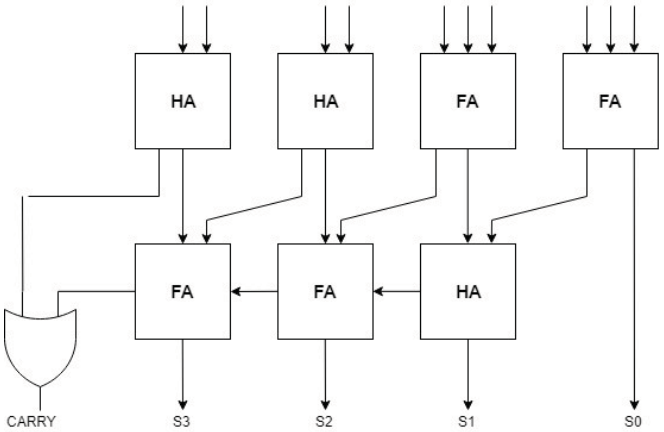

Fig. 10. Carry save adder for three 3-bit numbers

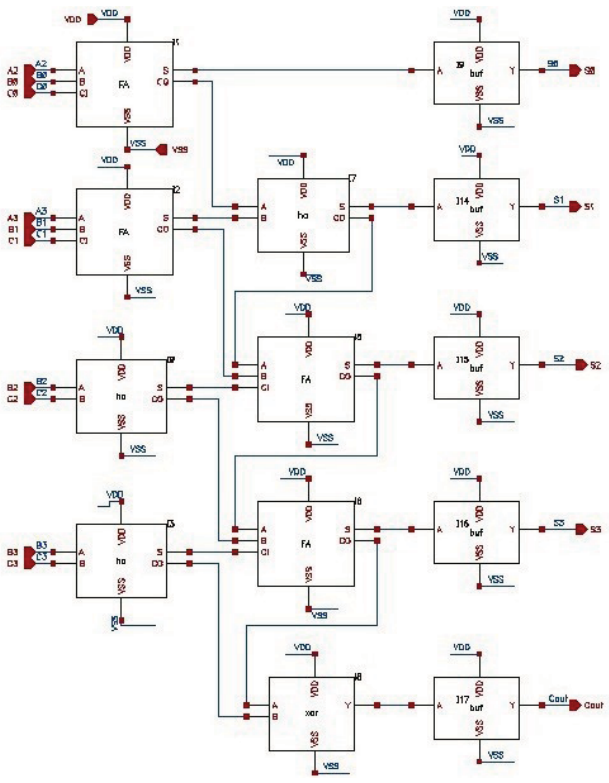

Fig. 11. Schematic of CSA

C. *Implementation of 4-bit Vedic multiplier*

Computation of 4-bit multiplier is done by using the 2-bit multiplier as a basic module. The 4-bit numbers is split in two halves and the 2-bit module is instantiated four times. Fig. 12 and Fig. 13 show the Urdhva Tiryagbhyam sutra being implemented for a 4-bit number.

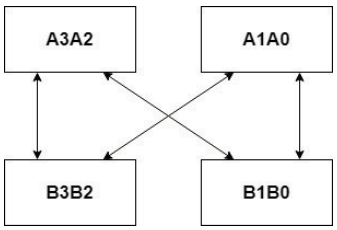

Fig. 12. 4-bit multiplication using Urdhva Tiryagbhyam sutra

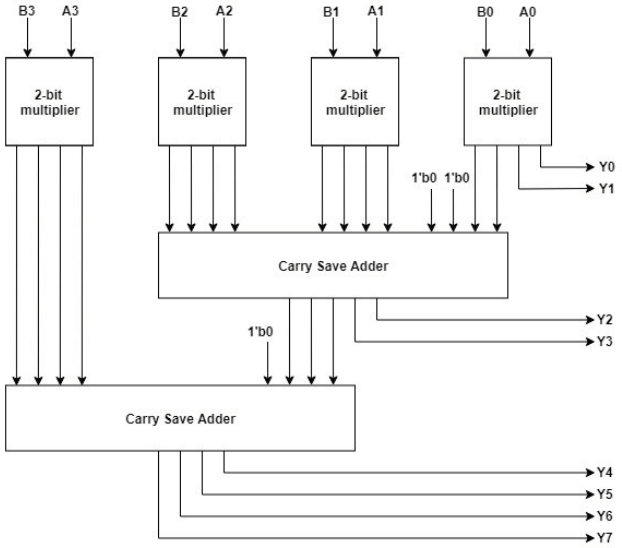

Fig. 13. Design of 4-bit multiplier

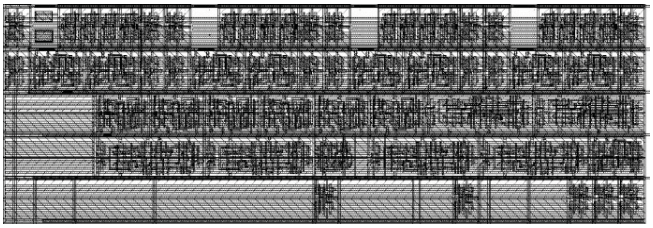

Fig. 14. Layout of 4-bit multiplier

Fig. 14 shows the layout floor plan of the 4-bit multiplier. For better visibility Metal3 and Metal4 have been disabled. For the outputs of the 4-bit multiplier to have greater drive strength and higher fan out, buffers are placed at vital points of the circuit.

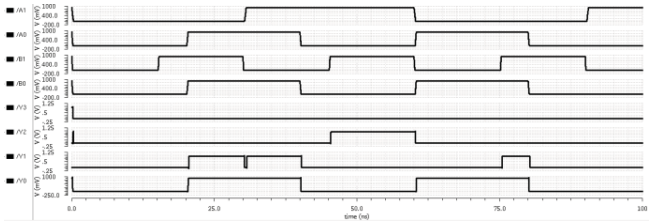

Fig. 15. Simulated waveforms of 2-bit multiplier

V. Simulation and Results

Fig. 15 shows the output waveform of 2-bit multiplier for given inputs of A0A1 and B0B1. The inputs are generated by pulse waveform each with different pulse width and period, which generates various test cases. The outputs correspond to Y3Y2Y1Y0, where Y3 is the MSB and Y0 is the LSB. For example consider the case 01x11. When multiplied the resultant 4-bits will be 0011.

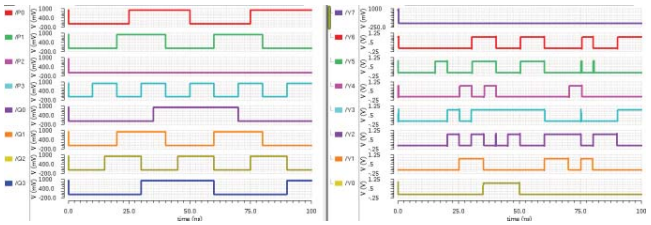

Fig. 16. Simulated waveform of 4-bit multiplier

Fig. 16 shows the simulated waveform of the 4-bit multiplier. P3P2P1P0 and Q3Q2Q1Q0 are the two 4-bit numbers that are to be multiplied. This will result in an 8-bit output (Y7Y6Y5Y4Y3Y2Y1Y0). Left half of Fig. 16 shows the two 4-bit inputs and the right half shows the corresponding 8-bit output.

Proposed design / Analysis	Proposed 2-bit multiplier	Proposed 4-bit multiplier	4-bit multiplier, Ref[5]
Propagation delay	60 ps	250 ps	268 ps
Power	353.4 nW	3.795 uW	2.6 uW
Area	15.08 um^2	218.88 um^2	549.31 um^2

Table. 1. Analysis of proposed design

Area, delay and power consumed are calculated and are shown in Table 1 and are found to occupy lesser area and faster than previous.ly proposed design.

VI. Improvements

The area occupied and power consumed by the multiplier can be improved by using multiplexers based on transmission gate and techniques like gate diffusion input.

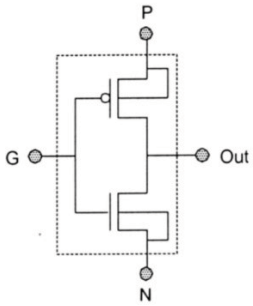

Fig. 18. Basic cell of GDI

Fig. 17 depicts the basic cell of GDI. A GDI cell consists of three inputs. Different functions can be realized by applying different voltages to the three inputs.

Realization of half and full adders can be done using this technique which will help decrease the power consumption and area occupied by the entire circuit.

VII. Conclusion

High speed and area efficient multiplier modules have been designed and developed, which can be implemented in FIR filters and digital signal processors. For future work, higher bit multipliers can be realized and can be compared with conventional multipliers. The multipliers based on this algorithm have high switching speeds even when there is a wide range of temperature change. Higher bit multipliers can be realized and have high operating frequency. Therefore, implementation of high speed DSP's and ASIC's are possible.

This paper presents a 4x4 multiplier with Carry Save Adder using Urdhva Tiryagbhyam sutra. Further, the performance analysis is carried out with voltage ranges from 0.9V to 1.1V. Delay of 250ps and consumption of 3.795uW of power was obtained in the highest constraint cases.

References

[1] R. Uma *et al.*, "Area, Delay and Power Comparison of Adder Topologies," *International Journal of VLSI design & Communication Systems (VLSICS)*, vol. 3, 2012.

[2] G. G. Kumar and V. Charishma, "Design of High Speed Vedic Multiplier using Vedic Mathematics Techniques," *International Journal of Scientific and eResearch Publications*, vol. 2, p. 3, 2012.

[3] D. Krishnaveni *et al.*, "VLSI implementation of Vedic multiplier with reduced delay," *International Journal of Advanced Technology & Engineering Research (IJATER)*, vol. 2, pp. 2250–3536, 2012.

[4] Y. Deodhe *et al.*, "Design and implementation of 8-Bit Vedic multiplier using CMOS logic," *International Conference on Machine Intelligence Research and Advancement*, 2013.

[5] S. Tripathy *et al.*, "Low power multiplier architectures using Vedic mathematics in 45nm technology for high speed computing," *International Conference on Communication, Information and Computing Technology*, 2015.

[6] S. S. Kerur *et al.*, "Implementation of Vedic multiplier for Digital Signal Processing," *International Conference on VLSI, Communication & Instrumentation (ICVCI)*, 2011.
[7] S. P. Pohokar *et al.*, "Design and implementation of 16x16 multiplier using Vedic mathematics," *International Conference on Industrial Instrumentation and Control (ICIC)*, 2015.
[8] R. Kaur and R. Mehra, "Power and Area Efficient CMOS Half Adder Using GDI Technique," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 36, no. 8, 2016.
[9] S. Karunakaran *et al.*, "VLSI architecture of an 8-bit multiplier using Vedic mathematics in 180nm technology," *International journal of advances in engineering and technology*, 2017.
[10] V. Harish and Dr. Kamatchi S, "Comparative Performance Analysis of Karat- suba Vedic Multiplier with Butterfly Unit," *Proceedings of the Third International Conference on Electronics Communication and Aerospace Technology (ICECA)*, 2019.
[11] P. Manju *et al.*, "Design of Vedic Multiplier using GDI Method," *International Journal of Recent Technology and Engineering (IJRTE)*, 2019.
[12] S. Prema *et al.*, "High Performance Reversible Vedic Multiplier Using Cadence 45nm Technology," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 2019.