

Implementation of an Efficient Multiplier Architecture over a Conventional Methods using Ancient Indian Vedic Sutra

Bhushan M. Shelke
 Lecturer
 NUVA C.E.T.,R.T.M.N.U.
 Nagpur,Maharashtra (INDIA)

Shubhangi A. Wakode
 Lecturer
 NUVA C.E.T.,R.T.M.N.U.
 Nagpur,Maharashtra (INDIA)

ABSTRACT

Fast multiplication is very important in processing of digital signals like DSP for convolution, Fourier Transform, etc. Many conventional methods are used for designing a multiplier for processing a digital signal. In this paper, A fast method for multiplication based on Ancient Indian Vedic mathematics is proposed. The whole of Vedic mathematics is based on 16 sutras (formulae). Among the various methods of multiplication in Vedic mathematics, Urdhava Tiryakbhyam (Vertically and Crosswise) is discussed in detail. This is the general multiplication formula applicable to all cases of multiplication. For implimentation of a efficient Architecture simple Boolean logic is combined with Vedic formulas, which reduces the partial products and sums generated in one step. The coding is done in VHDL and synthesis is done using Xilinx ISE 9.1i simulator. Results are compared with several conventional techniques.

Keywords

Vedic mathematics, Sutra,Urdhava Tiryakbhyam, VHDL.

1. INTRODUCTION

The Vedic Mathematics approach is totally different and considered very close to the way a human mind works. In this work, try to present multiplication operations and the implementation of these using both conventional, as well as Vedic mathematical methods in VHDL (VHSIC Hardware Description Language). Vedic mathematics is the name given to the ancient system of mathematics, a unique technique of calculations based on simple rules and principles with which any mathematical problem can be solved like arithmetic, algebra, geometry or trigonometry. Vedic Mathematics (1965), which is considered the starting point for all work on Vedic mathematics. It was rediscovered from the ancient Indian scriptures between 1911 and 1918 by Sri Bharati Krishna Tirthaji (1884-1960), a scholar of Sanskrit, mathematics, history and philosophy [1]. Conventional mathematics is an integral part of engineering education since most engineering system designs are based on various mathematical approaches. The need for faster processing speed is continuously driving major improvements in processor technologies, as well as the search for new algorithms. A multiplier is one of the key hardware blocks in most digital signal processing systems [2]. Performing atemathical calculations especially multiplication, a computer spends a considerable amount of its processing time, an improvement in the speed of a math coprocessor for performing multiplication will increase the overall speed of the computer. The multiplier is a fairly large block of a

computing system. The amount of circuitry involved is directly proportional to the square of its resolution i.e. A multiplier of size ‘N’ bits has ‘N²’ gates. Apply Vedic Sutras to binary multipliers using carry adders. In particular, develop an efficient binary multiplier architecture that performs partial product generations and additions. Here, the computation time involved is less. The combinational delay and the device utilizations obtained after synthesis is compared. The proposed Vedic multiplier based Circuit seems to have better performance in terms of speed.

1.1 Urdhava Tiryakbhyam Sutra

The meaning of this sutra is “Vertically and crosswise” and it is applicable to all the multiplication operations. Figure 1 represents the general multiplication procedure of the 4×4 decimal digit multiplication. Let us consider the multiplication of two decimal numbers (1234 × 8765).

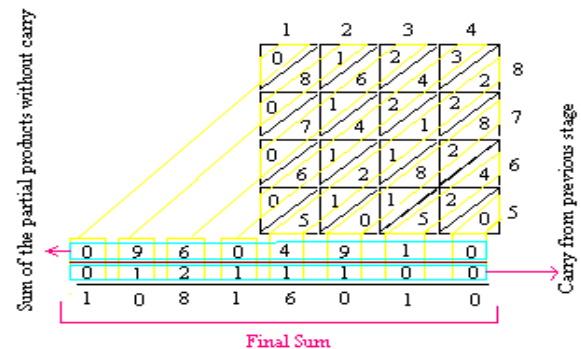


Figure 1 : Multiplication implementation using Urdhava Tiryakbhyam Sutra [3].

Multiplication using ‘Urdhava tiryakbhyam’ Sutra is shown in Figure 1. The numbers to be multiplied are written on two consecutive sides of the square as shown in the figure. Each of the small squares is partitioned into two equal 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 yellow boxes are added and producing sum and carry digits. Finally results are obtained by the addition of sum digits and the previous carry digits. Carry for the first step is taken to be zero [4]. The multiplication of two 2-digit decimal numbers 21 and 32 is shown in Figure 2. The least significant digit 1 of multiplicand is multiplied vertically by least significant digit 2 of the multiplier, get their product 2 and set it down as the least significant part of the answer. Then 2 and 2, 1 and 3 are multiplied crosswise, add the two,

get 7 as the sum and set it down as the middle part of the answer. Then 2 and 3 is multiplied vertically, get 6 as their product and put it down as the last the left hand most part of the answer.

$$21 \times 32 = 672$$

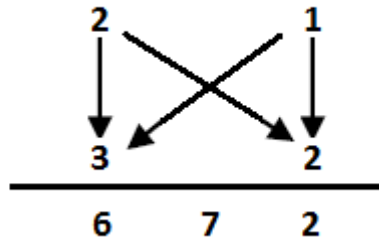


Figure 2 : Multiplication of 21 X 32 using Urdhva Tiryakbhyam Sutra.

1.1.1 Mathematical representation of Urdhava Tiryakbhyam sutra

Assume that X and Y are two numbers, to be multiplied. Mathematically X and Y can be represented as:

$$A = \sum_{i=0}^{N-1} A_i 10^i, \quad B = \sum_{j=0}^{N-1} B_j 10^j$$

Assume that, their product is equal to Z. Then Z can be represented as:

$$Z = \sum_{i=0}^{N-1} A_i \sum_{j=0}^{N-1} B_j 10^{i+j}$$

Where $(A_i, B_j \in \{0,1,2,\dots,9\})$ and 'N' may be any number [4]. From the above expression, it can be observed that each digit is multiplied consecutively and shifted towards the proper positions for partial product generation. Finally the partial products are added with the previous carry to produce the final results.

1.1.2 Binary realization of Urdhava Tiryakbhyam sutra

Consider a 8X8 bit binary number as A and B. So, bit multiplication will be as follows.

$$A = A7A6A5A4A3A2A1A0$$

$$B = B7B6B5B4B3B2B1B0$$

Make the partition into equal part of A and B.

$$AH = A7A6A5A4,$$

$$AL = A3A2A1A0.$$

$$BH = B7B6B5B4,$$

$$BL = B3B2B1B0.$$

Finally, according to Urdhava Tiryakbhyam Sutra,

$$A \times B = AL \times BL + AH \times BL + AL \times BH + AH \times BH$$

The final equation shows the result multiplication using Vedic Sutra.

2. DESIGN & IMPLEMENTATION

The Vedic multiplier is implemented using VHDL and also other multipliers like booth multiplier and shift-add multiplier are also implemented. VHDL code is completely synthesizable. The synthesis is done using Xilinx Synthesis

Tool (XST) available with Xilinx ISE 9.1i. The design starts first with Multiplier design, that is 2X2 bit multiplier as shown in figure 3. Here, "Urdhva Tiryakbhyam Sutra" Algorithm for multiplication has been effectively used to develop digital multiplier architecture. This algorithm is quite different from the traditional method of multiplication, which is to add and shift the partial products.

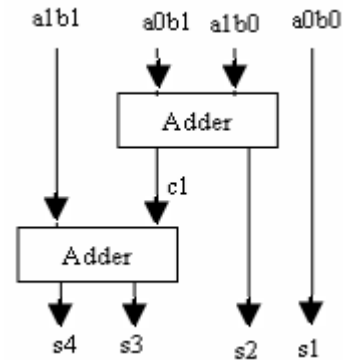


Figure 3 : Multiplication of 2X2 bit using Urdhva Tiryakbhyam Sutra.

For Multiplier, first the basic blocks, that are the 2x2 bit multipliers have been made and then, using these blocks, 4x4 block has been made by adding the partial products using adder circuits and then using this 4x4 block the 8x8 bit multiplier block is implemented. The figure 4 shows implementation of 8X8 bit multiplier using Urdhava Tiryakbhyam sutra. The 8 bit input A and B are given to the 4X4 vedic multiplier block which generate the output to be manipulate into adder circuit to obtained the output given as,

$$X \times Y = (Z_{15}-Z_8) \& (Z_7-Z_4) \& (Z_3-Z_0)$$

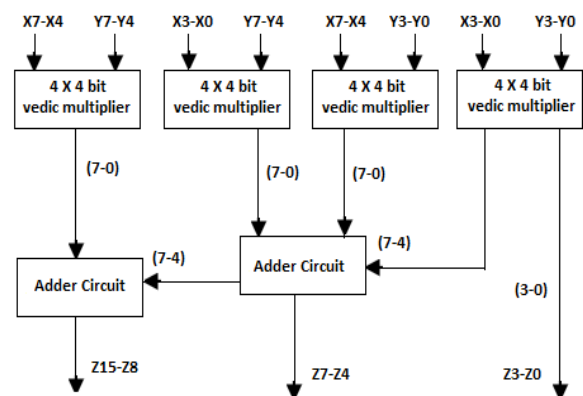


Figure 4 : Hardware realization of 8x8 bit multiplication using Urdhava Tiryakbhyam Sutra.

The architecture of 16X16 Vedic multiplier using Urdhava Tiryakbhyam Sutra is shown in Figure 5. The 16X16 Vedic multiplier architecture is implemented using four 8x8 Vedic multiplier modules and two 16 bit binary adder stages. The resultant output is given as,

$$X \times Y = (Z_{31}-Z_{16}) \& (Z_{15}-Z_8) \& (Z_7-Z_0)$$

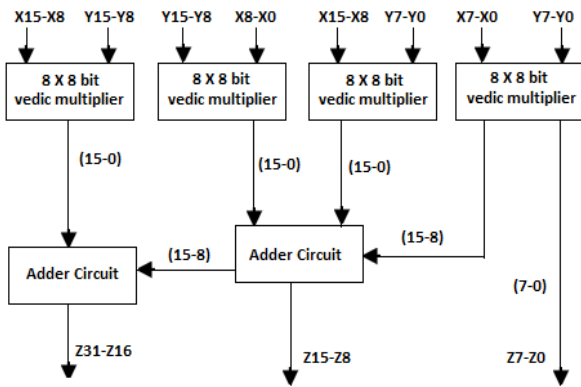


Figure 5 : Hardware realization of 16x16 bit multiplication using Urdhava Tiryakbhyam Sutra.

Finally, 32X32 bit Vedic Multiplier as shown in figure 6 has been made using the four 16X16 bit Vedic multiplier modules and two 32 bit binary adder stages. The resultant output is given as,

$$X \times Y = (Z_{63} - Z_{32}) \& (Z_{31} - Z_{16}) \& (Z_{15} - Z_0)$$

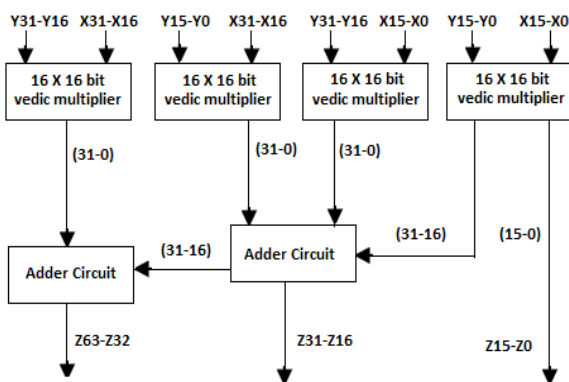


Figure 6 : Hardware realization of 32x32 bit multiplication using Urdhava Tiryakbhyam Sutra.

2.1 Design steps in FPGA implementation

FPGA (Field Programmable Gate Array) device is used for implementation of Vedic Multiplier designed using HDL (Hardware Description Language), here it is VHDL. As shown in figure 7, there are four basic building blocks in FPGA implementation.

1. The design starts with VHDL Entry level refers to RTL level.
2. The design compilation produces Netlist upto the Gate level.
3. Optimization process designed the optimized view of optimized Gate structure.
4. This stage helps to prove the manifests of the design structure by simulation process.
5. The combination of step (2) and (3) equally called as Synthesis Process.
6. Final stage is the implementation on physical device with the help of Place and Route process.
7. As the FPGA device is reconfigurable, the entire procedure can be done several times.

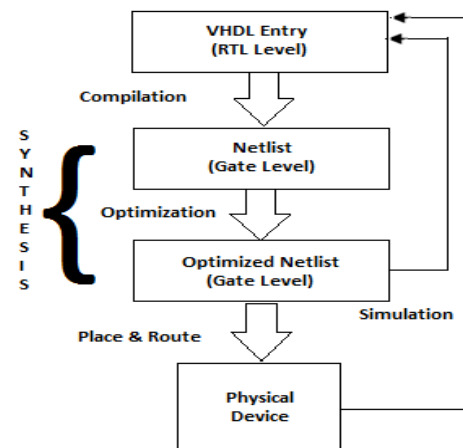


Figure 7 : Design steps in FPGA implementation.

3. RESULTS AND DISCUSSION

In this work the algorithms are implemented in VHDL and logic simulations are done using Xilinx project navigator vession 9.1i. The Xilinx families are used for simulation are as given below,

Xilinx : Spartan3 3s50pq208-5.

Xilinx : Vertex2P 2vp2fg256-7.

Table 1 : Vedic multiplier using Urdhava Tiryakbhyam Sutra – Combinational Delay in Various Devices (ns).

Device	8 X 8	16X16	32X32
Spartan3 s50pq208-5.	23.422	43.398	79.848
Vertex2P vp2fg256-7.	12.963	23.231	42.445

Table 2 : Shift-Add multiplier-Combinational Delay in Various Devices (ns).

Device	8 X 8	16X16	32X32
Spartan3 3s50pq208-5.	27.133	50.968	96.352
Vertex2P 2vp2fg256-7.	14.875	27.240	51.182

Table 3 : Booth multiplier-Combinational Delay in Various Devices (ns).

Device	8X8	16X16	32X32
Spartan3 3s50pq208-5.	25.756	59.238	117.843
Vertex2P 2vp2fg256-7.	15.815	36.071	63.741

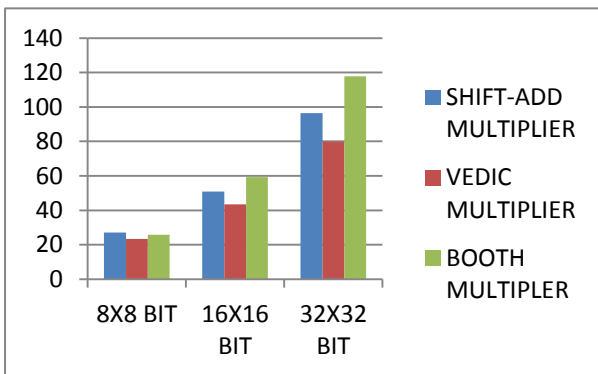


Figure 8 : Grafical representation of combinational delay (ns) for Xilinx Spartan3 (3s50pq208-5) family from Table 1,2,3.

From figure 8, it is seen that the vedic multiplier requires less time than conventional multipliers. The efficiency of Vedic Multiplier for each method is obtained, as shown in table 4.

Table 4 : Efficiency of Vedic multiplier over a conventional method for Xilinx Spartan3 (3s50pq208-5) family from table 1,2,3.

Technique	8X8	16X16	32X32
Shiftadd Method	13.67%	14.85%	17.12%
Booth Method	12.85%	14.71%	17.07%

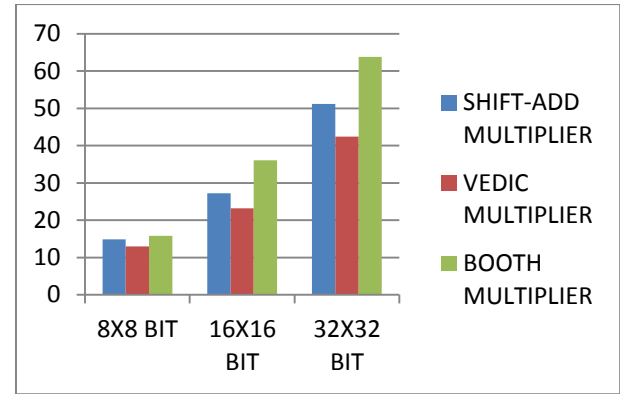


Figure 9 : Grafical representation of combinational delay (ns) for Xilinx Vertex2P (2vp2fg256-7) family from Table 1,2,3.

Similarly, Figure 9 shows the efficient result of Vedic Multiplier for a Combinational delay over a conventional methods and it is numerically shown in table 5.

Table 5 : Efficiency of Vedic multiplier over a conventional methods for Xilinx Vertex2P (2vp2fg256-7) family from table 1,2,3.

Technique	8X8	16X16	32X32
Shiftadd Method	09.06%	26.73%	32.24%
Booth Method	18.03%	35.57%	36.41%

From table 4 and 5, it is seen that as increase in the number of input bits, the performance of the Vedic Multiplier is gradually increases. As shown in table 4 and 5, the Shift-Add multiplier is more efficient than Booth multiplier in time utilization. This is also observed that the utilization of space is effectively efficient in Vedic Multiplier over a Conventional Multipliers. Validation of space efficiency is proven by Xilinx simulator. The obtained results are shown in table 6,7,8 for Xilinx Spartan3 (3s50pq208-5) family.

Table 6 : Device Utilization Summary of 8X8 bit Vedic and Conventional Multiplier for Xilinx Spartan3 (3s50pq208-5) family.

No. of	Vedic Multiplier	ShiftAdd Multiplier	Booth Multiplier [5]
Slices	66 out of 768(8%)	68 out of 768(8%)	96 out of 768(12%)
4 input LUTs	114 out of 1536 (7%)	119 out of 1536(7%)	178 out of 1536 (12%)
Bonded IOBs	32 out of 124(25%)	32 out of 124(25%)	32 out of 124(25%)

Table 7 : Device Utilization Summary of 16X16 bit Vedic and Conventional Multiplier for Xilinx Spartan3 (3s50pq208 -5) family.

No. of	Vedic Multiplier	ShiftAdd Multiplier	Booth Multiplier[5]
Slices	310 out of 768 (40%)	367 out of 768 (47%)	499 out of 768(65%)
4 input LUTs	541 out of 1536(35%)	641 out of 1536 (42%)	923 out of 1536(60%)
Bonded IOBs	64 out of 124(51%)	64 out of 124(51%)	65 out of 124(52%)

Table 8 : Device Utilization Summary of 32X32 bit Vedic and Conventional Multiplier for Xilinx Spartan3 (3s50pq208 -5) family.

No. of	Vedic Multiplier	ShiftAdd Multiplier	Booth Multiplier
Slices	1218 out of 768 (158%)	1458 out of 768 (190%)	2367 out of 768 (308%)
4 input LUTs	2119 out of 1536 (137%)	2814 out of 1536 (183%)	4348 out of 1536 (283%)
Bonded IOBs	128 out of 124 (103%)	128 out of 124 (103%)	129 out of 124 (104%)

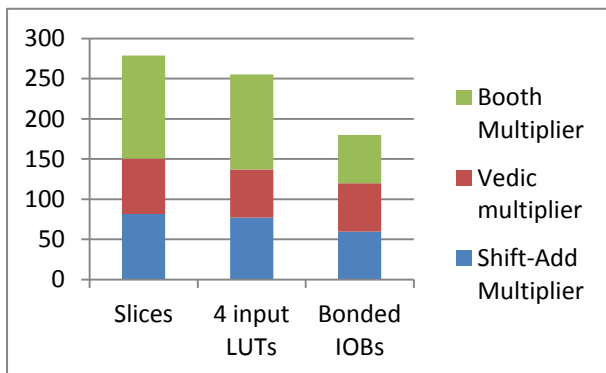


Figure 10 : Grafical representation of average device utilization (%) for Xilinx Spartan3 (3s50pq208-5) family from Table 6,7,8.

Similar results are obtained for the validation of space efficiency of Vedic Multiplier over a Shift-Add and Booth Multiplier by using Xilinx Vertex2P (2vp2fg256-7) family and it is shown in table 9,10,11.

Table 9 : Device Utilization Summary of 8X8 bit Vedic and Conventional Multiplier for Xilinx Vertex2P (2vp2fg256-7) family.

No. of	Vedic Multiplier	ShiftAdd Multiplier	Booth Multiplier
Slices	76 out of 1408 (5%)	81 out of 1408 (6%)	117 out of 1408 (8%)
4 input LUTs	133 out of 2816 (5%)	173 out of 2816 (6%)	198 out of 2816 (7%)
Bonded IOBs	32 out of 140 (22%)	32 out of 140 (25%)	33 out of 140 (23%)

Table 10 : Device Utilization Summary of 16X16 bit Vedic and Conventional Multiplier for Xilinx Vertex2P (2vp2fg256-7) family.

No. of	Vedic Multiplier	ShiftAdd Multiplier	Booth Multiplier
Slices	310 out of 1408 (22%)	385 out of 1408 (27%)	417 out of 1408 (29%)
4 input LUTs	541 out of 2816 (19%)	595 out of 2816 (21%)	639 out of 2816 (23%)
Bonded IOBs	64 out of 140 (45%)	64 out of 140 (45%)	65 out of 140 (46%)

Table 11 : Device Utilization Summary of 32X32 bit Vedic and Conventional Multiplier for Xilinx Vertex2P (2vp2fg256-7) family.

No. of	Vedic Multiplier	ShiftAdd Multiplier	Booth Multiplier
Slices	1218 out of 1408 (86%)	1958 out of 1408 (139%)	2367 out of 1408 (168%)
4 input LUTs	2119 out of 2816 (75%)	3314 out of 2816 (118%)	4348 out of 2816 (154%)
Bonded IOBs	128 out of 140 (91%)	128 out of 140 (91%)	129 out of 140 (92%)

The graphical representation of average device utilization is shown in figure 11 for the simulation results shown in table 9,10,11.

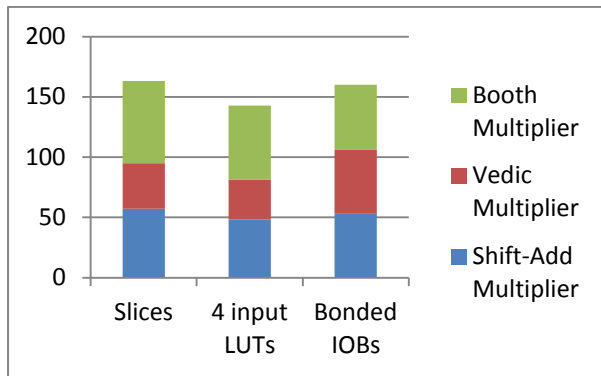


Figure 11 : Grafical representation of average device utilization (%) for Xilinx Vertex2P (2vp2fg256-7) family from Table 6,7,8.

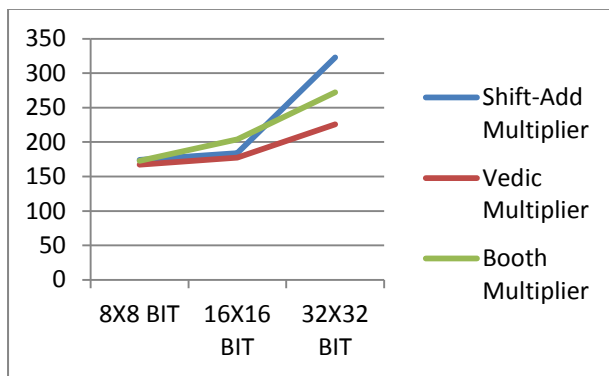


Figure 12 : Grafical representation of total memory usage (Megabyte) for Xilinx Spartan3 (3s50pq208-5) family.

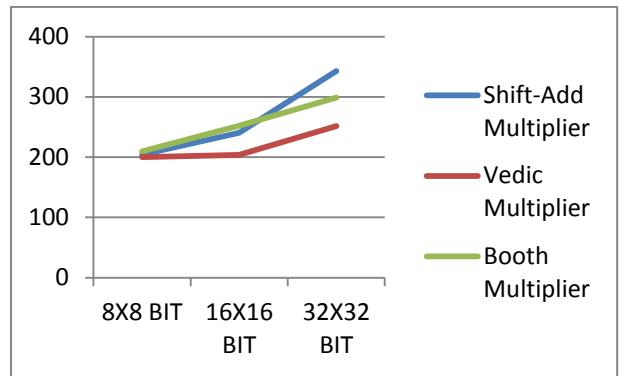


Figure 13 : Grafical representation of total memory usage (Megabyte) for Xilinx Vertex2P (2vp2fg256-7) family.

Figure 12 and 13 are based on the result obtained for the total memory usages by FPGA family used for synthesis using Xilinx Simulator.

The simulation results for a 8X8 bit,16X16 bit, 32X32 bit Vedic Multipliers using Vedic Sutra Urdhava Tiryakbhyam is shown in figure 14,15,16 respectively.

x[7:0]	8'hBA	8'hFF	8'hEE	8'h06	8'hCD	8'hBA
y[7:0]	8'hCA	8'hFF	8'hAB	8'h09	8'hFE	8'hCA
z[15:0]	1...	16'hFE01	16'h9EFA	16'h0036	16'hCB66	16'h92C4

Figure 14 : The results of 8X8 bit multiplication by Vedic Multiplier using Urdhava Tiryakbhyam Sutra.

x[15:0]	1...	16'h1111	16'hAB16	16'h1010	16'hF765	
y[15:0]	1...	16'h1111	16'h9124	16'hAACC	16'h3288	
z[31:0]	3...	32'h01234321	32'h60FF8518	32'h7224F188	32'h032BA880	32'h30D527A8

Figure 15 : The results of 16X16 bit multiplication by Vedic Multiplier using Urdhava Tiryakbhyam Sutra.

P[63:0]	6...	64'h000C3799AA42D208	64'h0000002720BF7564
X[31:0]	3...	32'h12345678	32'h023456FD
Y[31:0]	3...	32'h00ABCDEF	32'h00001234

Figure 16 : The results of 32X32 bit multiplication by Vedic Multiplier using Urdhava Tiryakbhyam Sutra.

4. CONCLUSIONS

The performance of the proposed multiplier using Vedic Sutra proved to be efficient in terms of speed, space and power. As the input bits increases, relatively the efficiency of the circuit is also increases. The main advantage is delay increases slowly as input bits increase. It is also demonstrated that this

design is quite efficient in terms of silicon area/speed. By using these ancient Indian Vedic mathematics techniques, the world can reach new heights of performance and excellence for the cutting edge technology devices.

5. REFERENCES

- [1] Vedic mathematics or sixteen simple mathematical formulae from the Vedas by Jagadguru Swami Shri Bharati Krishna Tirthaji Maharaja, General Editor – Dr. V. S. Agrawal , First Edition : Varanasi ,1965.
- [2] Kabiraj Sethi , Rutuparna Panda : “An Improved Squaring Circuit for Binary Numbers”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No.2, 2012.
- [3] G.Ganesh Kumar, V.Charishma : “Design of High Speed Vedic Multiplier using Vedic Mathematics Techniques”, International Journal of Scientific and Research Publications, Volume 2, Issue 3, March 2012 ISSN 2250-3153.
- [4] P. Saha, A. Banerjee, A. Dandapat, P. Bhattacharyya :” Vedic Mathematics Based 32-Bit Multiplier Design for High Speed Low Power Processors”, international journal on smart sensing and intelligent systems vol. 4, No. 2 June 2011.
- [5] S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur, Girish V A : “Implementation of Vedic Multiplier for Digital Signal Processing”, International Conference on VLSI, Communication & Instrumentation (ICVCI) 2011 Proceedings published by International Journal of Computer Applications® (IJCA).