Abstract—The main purpose of the project is to improve the speed of the digital circuits like multiplier since adder and multiplier are one of the key hardware components in high performance systems such as microprocessors, digital signal processors and FIR filters etc. Hence we always try for good multiplier architecture to increase the efficiency and performance of a system. Vedic multiplier is one such high speed multiplier architecture. This ‘Vedic Mathematics’ is the name given to the ancient system of mathematics or, to be precise, a unique technique of calculations based on simple rules and principles, with which any mathematical problem can done with the help of arithmetic, algebra, geometry or trigonometry can be solved. Multiplication plays an important role in the processors. It is one of the basic arithmetic operations and it requires more hardware resources and processing time than the other arithmetic operations. Vedic mathematics is the ancient Indian system of mathematic. It has a unique technique of calculations based on 16 Sutras. The multiplication sutra between these 16 sutras is the Urdhva Tiryagbhyam sutra which means vertical and crosswise. In this paper it is used for designing a high speed, low power 4x4 multiplier. The proposed system is design using VHDL and it is implemented through Xilinx ISE 14.2.

Keywords—Multiplier, Vedic multiplier, Vedic Mathematics, Urdhva Tiryagbhyam.

I. INTRODUCTION

Multiplication [3] is the most important arithmetic operation in signal processing applications and inside the Processor. As speed is always a major requirement in the multiplication operation, increase in speed can be achieved by reducing the number of steps in the computation process. The speed of multiplier determines the efficiency of such a system. In any system design, the three main constraints which determine the performance of the system are speed, area and power requirement. Vedic mathematics [1] was reconstructed from the ancient Indian scriptures (Vedas) by Swami Bharati Krishna Tirthaji Maharaja (1884-1960) after his eight years of research on Vedas. Vedic mathematics is mainly based on sixteen principles or word-formulae which are termed as sutras. This is a very interesting field and presents some effective algorithms which can be applied to various branches of engineering such as computing and digital signal processing. Integrating multiplication with Vedic Mathematics techniques would result in the saving of computational time.

These are the 16 basic sutra of Vedic mathematic:
1) (Anurupye) Shunyamanyat -If one is in ratio, the other is zero.
2) ChalanaKalanabyham -Differences and similarities.
3) Ekadhikina Purvena- By one more than the previous One.
4) Ekanayunena Purvena - By one less than the previous one.
5) Gunakasamuchyah-Factors of the sum is equal to the sum of factors.
6) Gunitasamuchyah-The product of sum is equal to sum of the product.
7) Nikhilam Navatashcaramam Dashatah -All from 9 and last from 10.
8) Paraavartya Yojayet-Transpose and adjust.
9) Puranapuranabhyham - By the completion noncompletion.
10) Sankalana-vyavakalanabhyham -By addition and by subtraction.
11) Shesanyankena Charamena- The remainders by the last digit.
12) Shunyam Saamyasamuccaye -When the sum is same then 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.

Thus, integrating Vedic mathematics for the multiplier design will enhance the speed of multiplication operation.

The multiplier architecture is based on Urdhva Tiryagbhyam [4] (vertical and cross-wise algorithm) sutra. An illustration of Urdhva Tiryagbhyam sutra is shown in Figure 1[2].
The 4x4 multiplication has been done in a single line in Urdhva Tiryagbhyam sutra [1], whereas in shift and add (conventional) method, four partial products have to be added to get the result. Thus, by using Urdhva Tiryagbhyam Sutra in binary multiplication, the number of steps required calculating the final product will be reduced and hence there is a reduction in computational time and increase in speed of the multiplier.

Consider two 4-bit binary numbers $a_3a_2a_1a_0$ and $b_3b_2b_1b_0$. The partial products [5] $(P_7P_6P_5P_4P_3P_2P_1P_0)$ generated are given by the following equations:

i. $P_0 = a_0b_0$

ii. $P_1 = a_0b_1 + a_1b_0$

iii. $P_2 = a_0b_2 + a_1b_1 + a_2b_0 + P_1$

iv. $P_3 = a_0b_3 + a_1b_2 + a_2b_1 + a_3b_0 + P_2$

v. $P_4 = a_1b_3 + a_2b_2 + a_3b_1 + P_3$

vi. $P_5 = a_1b_2 + a_2b_1 + P_4$

vii. $P_6 = a_3b_3 + P_5$

viii. $P_7 =$ carry of $P_6$

### II. VEDIC MULTIPLIER

An “Urdhva Tiryagbhyam” Sutra is a general multiplication formula applicable to all cases of multiplication such as binary, hex, decimal and octal. The Sanskrit word “Urdhva” means “Vertically” and “Tiryagbhyam” means “crosswise”. Fig 4 shows an example of Urdhva Tiryagbhyam.

**Algorithm:** Multiplication of 101 by 110

1. We will take the right-hand digits and multiply them together. This will give us LSB digit of the answer.
2. Multiply LSB digit of the top number by the second bit of the bottom number and the LSB of the bottom number by the second bit of the top number. Once we have those values, add them together.
3. Multiply the LSB digit of bottom number with the MSB digit of the top one, LSB digit of top number with the MSB digit of bottom and then multiply the second bit of both, and then add them all together.
4. This step is similar to the second step, just move one place to the left. We will multiply the second digit of one number by the MSB of the other number.

Finally, simply multiply the LSB of both numbers together to get the final product.

![Fig-2: Urdhva Tiryakhayam Procedure for Multiplication](image)

The partial products are generated parallel and concurrent additions of the partial products are done using this algorithm. Because of this the speed of the multiplier is increased considerably when compared to other techniques.

### III. EXISTING 4X4 VEDIC MULTIPLIER

This is the existing 4x4 Vedic multiplier, which is having four 2x2 Vedic multiplier and three 4 bit ripple carry adder. [5]
IV. PROPOSED 4X4VEDIC MULTIPLIER

Proposed 4-bit adder performs the function of 4-bit addition that gives two bits of sum and one carry as output. Its block diagram contains one full adder (FA) and two half adders (HF) is given in Fig-4.

Here, A, B, C, D are four inputs. S0 and S1 are LSB and MSB of Sum outputs respectively and Sum is the sum of four inputs. C0 is the carry bit.

To reduce the delay, a 4X4 multiplier is implemented using half adder, full adder and the proposed 4-bit adder as shown in Figure 5.

V. RESULT AND DISCUSSION

The proposed 4x4 multiplier is coded in VHDL, simulated using Xilinx ISim simulator, synthesized using Xilinx XST and verified for possible inputs given below. Inputs are generated using VHDL test bench. The simulation result for 4x4Vedic multiplier is shown in below.
A. Device utilization summary:

---------------------------
Selected Device: 3s500efg320-4
Number of Slices: 18 out of 4656 0%
Number of 4 input LUTs: 31 out of 9312 0%
Number of IOs: 16
Number of bonded IOBs: 16 out of 232 6%
---------------------------

B. Timing Detail:

All values displayed in nanoseconds (ns)
Total number of paths / destination ports: 270 / 8

-----------------------------------
Delay: 12.825ns (Levels of Logic = 8)
Source: a<0> (PAD)
Destination: p<7> (PAD)
Data Path: a<0> to p<7>

Gate     Net
Cell     in->out      fanout   Delay   Logical Name (Net Name)
-----------------------------------

IBUF: I->O  5  1.218  0.808  a_0_IBUF (a_0_IBUF)
LUT2: I0->O  3  0.704  0.706  n6/z1 (s<22>)
LUT4: I0->O  2  0.704  0.622  f1/y1 (s<3>)
LUT3: I0->O  2  0.704  0.482  f4/y1 (s<15>)
LUT3: I2->O  2  0.704  0.622  f5/y1 (s<16>)
LUT3: I0->O  2  0.704  0.451  f6/y1 (s<17>)
LUT4: I3->O  1  0.704  0.420  f7/y1 (p_7_OBUF)
OBUF: I->O  3.272  p_7_OBUF (p<7>)
-----------------------------------
The Proposed 4x4 Vedic multiplier designed is compared with existing architecture in terms of total delay, logic delay, route delay and number of logic levels. The results obtained are tabulated in Table I. From table I, it is evident that there is a reduction in both total delay and logic levels. The routing delay is found to be 4.111 ns, the logic delay is 8.719 ns; thus, giving a total delay of 12.825 ns. The number of logic levels is 8. Thus, it is clear that the proposed design is more efficient than the existing one. The proposed architecture can be used to develop a high speed complex number multiplier with reduced delay.

VI. CONCLUSIONS

This paper presents a novel way of realizing a high speed multiplier [2] using Urdhva Tiryagbhyam sutra. A 4-bit modified multiplier is designed by using the proposed 4 bit adder. The proposed 4x4 multiplier gives a total delay of 12.825 ns which is less when compared to the total delay of existing multiplier architecture. Results also indicate a 13.19% increase in the speed when compared to normal Vedic multiplier. Our design more preferable over all other designs.

REFERENCES