VHDL Implementation of 8-Bit Vedic Multiplier Using Barrel Shifter

Bhavin D Maru[1] Altaf Darvadiya[2]
[1]P.G. Student [2]Asst. Professor
[1,2] Department of Electronics & Communication Engineering
[1,2] C.U.Shah College Of Engineering And Technology, Wadhwan, Gujarat

Abstract— This paper describe that the propagation delay of 8-bit Vedic multiplier is reduced when compared with conventional multiplier like array multiplier, booth multiplier, wallance multiplier.in our design we use barrel shifter which requires only one clock cycle for 'n' number of shifts. The design is implemented in Xilinx simulator and verified using ISE simulator. The design will implement on Xilinx Spartan-6 family. The propagation delay will take from synthesis report and static timing report. The design may achieve propagation delay approximately 6ns to 9ns using barrel shifter in base selection module and multiplier.

Key words: Vedic Formulas, Nikhilam Sutra, Barrel Shifter, Base Selection Module, Propagation Delay, Power Index Determinant.

I. INTRODUCTION

Arithmetic operations like addition, subtraction and multiplication are essential in different digital circuits to boost the process of computation. Vedic mathematics is the great technique for arithmetic operations. Whereas conventional techniques for multiplication gives significant amount of delay in hardware implementation of n-bit multiplier. This delay degrades the performance of the multiplier.

In this work our aim is to reduce the propagation delay of Vedic multiplier using barrel shifter. The "Nikhilam Sutra" [1] implemented is modified. By using the barrel shifter in [1] the delay will reduce when compared with conventional multipliers.

II. VEDIC SUTRAS

"Vedic Mathematics" refers to a technique of calculation based on a set of 16 Sutras**.** Vedic sutras are the gift of ancient Indian mathematics. For large number of mathematical operations they apply. By using these sutras saves a lot of time compared to conventional computations. The faster processing speed is major improvements in processor technologies. The Vedic mathematics technique is totally different.

Many architectures of multiplier have been reported but the performance of multiplier was improved in proposed design. The architecture in [1] is changed using barrel shifter so significant amount of clock cycles are reduced so speed increases. The performance of the proposed multiplier is compared with the previously implemented multipliers.

"Vedic mathematics" is comprised of sixteen simple mathematical formulae from the Vedas [5].

1) Ekadhikena Purvena
2) Nikhilam navatascaramam Dasatah
3) Urdhva - tiryagbhyam
4) Paravartya Yojayet
5) Sunyam Samya Samuccaye
6) Anurupye - Sunyamanyat
7) Sankalana - Vyavakalanabhyam
8) Puranapuranabhyam
9) Calana - Kalanabhyam
10) Ekanyunena Purvena
11) Anurupyena
12) Adyamadyenantya - mantyena
13) Yavadunam Tavadunikrtya Varganca Yojayet
14) Antyayor Dasakepi
15) Antyayoreva
16) Gunita Samuccayah.

A. "Urdhva-tiryakbyham" Sutra

The meaning of this sutra is "Vertically and crosswise" and it is applicable to all the multiplication operations.

Fig. 1 represents the general multiplication procedure of the 4x4 multiplication. This process is called as array multiplication technique. It is an efficient multiplication technique when the multiplier and multiplicand lengths are small, for the larger length multiplication this technique is not good because a large amount of propagation delays are involved in these cases. To overcome this problem we are describing Nikhilam sutra for calculating the multiplication of two larger numbers.

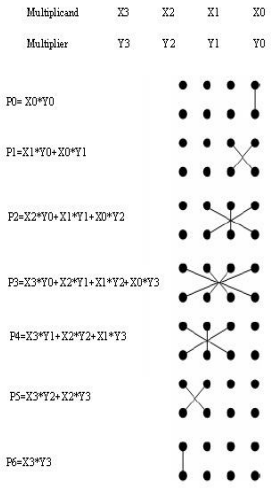

Fig.1 Multiplication procedure using"Urdhva-tiryakbyham" sutra

III. PROPOSED MULTIPLIER ARCHITECTURE DESIGN

Consider two n bit numbers X and Y.k_1 and k_2 are the exponent of X and Y. X and Y can be represented as:

$$X = 2^{k1} \pm z_1 \quad (1)$$

$$Y = 2^{k2} \pm z_2 \quad (2)$$

For the fast multiplication using Nikhilam sutra the bases of the multiplicand and the multiplier should be same, thus the equation (2) can be rewritten as:

$$Y \times 2^{k1-k2} = 2^{k1} \pm z_2 2^{k1-k2} \quad (3)$$

$$X \times Y \times 2^{k1-k2} = (2^{k1} \pm z_1)(2^{k1} \pm z_2 2^{k1-k2}) \quad (4)$$

$$= 2^{2k1} \pm z_1 2^{k1} \pm z_2 2^{2k1-k2} \pm z_1 z_2 2^{k1-k2} \quad (5)$$

$$= 2^{k1}(2^{k1} \pm z_1 \pm z_2 2^{k1-k2}) \pm z_1 z_2 2^{k1-k2} \quad (6)$$

$$= 2^{k1}(X \pm z_2 2^{k1-k2}) \pm z_1 z_2 2^{k1-k2} \quad (7)$$

$$P = XY = 2^{k2}(X \pm z_2 2^{k1-k2}) \pm z_1 z_2 \quad (8)$$

The hardware implementation of the above expression is partitioned into three blocks.

1) Base Selection Module
2) Power index Determinant Module
3) Multiplier.

The base selection module (BSM) is used to select the maximum base with respect to the input numbers. The second sub-module power index determinant (PID) is used to extract the power index of k1 and k2.The multiplier comprises of base selection module (BSM), power index determinant (PID), subtract or, barrel shifter, adder/subtractor as sub-modules in the architecture.

A. Base selection module

The base selection module consists power index determinant (PID) as the sub-module along with barrel shifter, adder, average determinant, comparator and multiplexer.

Operation:

An input 8-bit number is fed to power index determinant (PID) to interpret maximum power of number which is fed to barrel shifter and adder. The output of the barrel shifter is 'n' number of shifts with respect to the adder output and the input based to the shifter. Now, the outputs of the barrel shifter are given to the multiplexer with comparator input as a selection line. The outputs of the average determinant and the barrel shifter are fed to the comparator. The required base is obtained in accordance with the multiplexer inputs and its corresponding selection line.

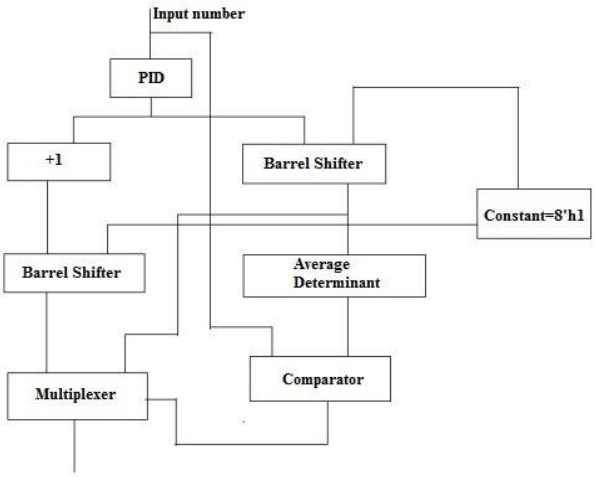

Fig.2 Base Selection Module, *BSM*

B. Power index determinant

The input number is fed to the shifter which will shift the input bits by one clock cycle. The shifter pin is assigned to shifter to check whether the number is to be shifted or not. In this power index determinant (PID) the sequential searching has been employed to search for first '1' in the input number starting from MSB. If the search bit is '0' then the counter value will decrement up to the detection of input search bit is '1'.Now the output of the decremented is the required power index of the input number.

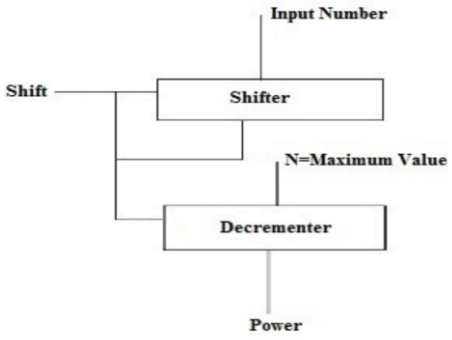

Fig. 3 Power Index Determinant

C. Multiplier Architecture

The base selection module and the power index determinant form subpart of multiplier architecture. The architecture computes the mathematical expression in equation8.Barrel shifter used in this architecture.

The two input numbers are fed to the base selection module from which the base is obtained. The outputs of base selection module (BSM) and the input numbers 'X' and 'Y' are fed to the subtractor. The subtractor blocks are required to extract the residual parts z1 and z2.The inputs to the power index determinant are from base selection module of respective input numbers.

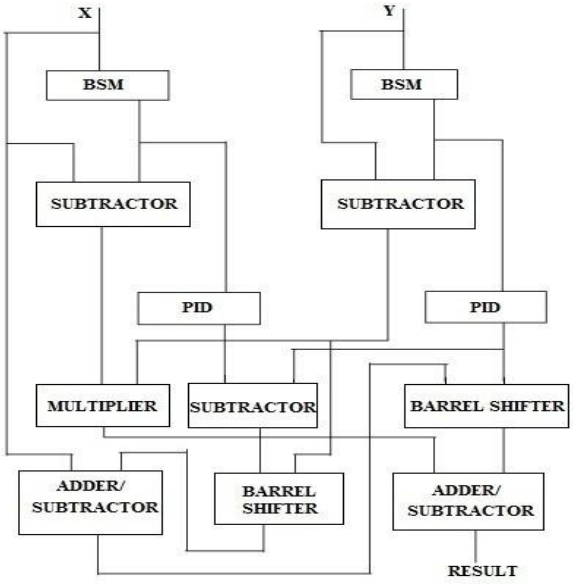

Fig. 4: Multiplier Architecture

The sub-section of power index determinant (PID) is used to extract the power of the base and followed by subtractor to calculate the value. The outputs of subtractor are fed to the multiplier that feeds the input to the second

adder or subtractor. Likewise the outputs of power index determinant are fed to the third subtractor that feeds the input to the barrel shifter. The input number 'X' and the output of barrel shifter are given to first adder/subtractor and the output of it is applied to the second barrel shifter which will provide the intermediate value. The last section of this multiplier architecture is the second adder/subtractor which will provide the required result.

IV. SIMULATION RESULTS AND DESIGN ANALYSIS

Comparison between conventional multipliers and proposed design is been given below. Around 75% of reduction in delay can be observed from the proposed design with respect to array multiplier in Table 1.whereas the conventional Vedic multiplier contributes to 43% of reduction in delay with respect to array multiplier. The analysis provides much in depth coverage between conventional multipliers and modified Vedic multiplier architecture.

Multiplier Type	Conventional Multiplier	Vedic Multiplier	Proposed Multiplier
Delay(ns)	43.42	27	6 to 9

Table 1. Delay Comparison between Various Multipliers Implemented On FPGA [3].

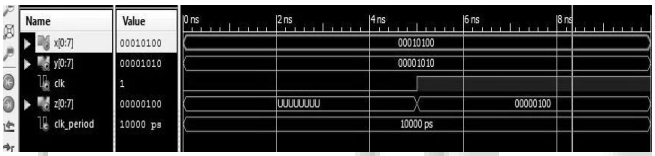

Fig.5 Simulation Result up to Barrel Shifter below subtractor in Multiplier Architecture

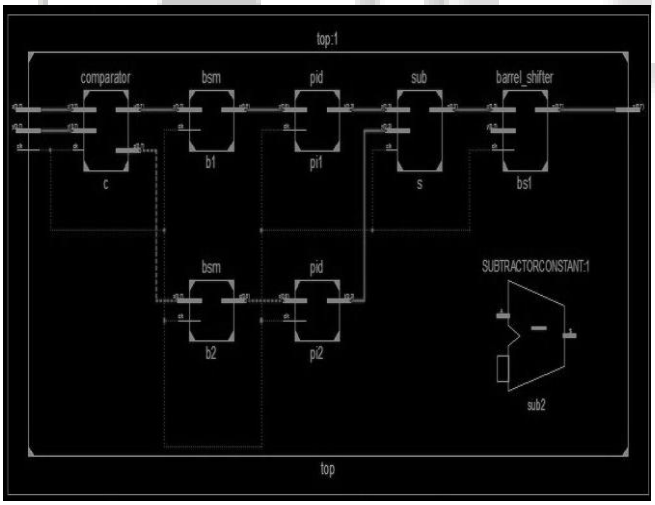

Fig.6 RTL schematic up to Barrel Shifter below subtractor in multiplier Architecture

V. CONCLUSION

In our design, efforts have been made to reduce the propagation delay and may be achieve an improvement in the reduction of delay with 45% when compared to array multiplier, booth multiplier and conventional Vedic multiplier implementation on FPGA [4].The high speed implementation of such a multiplier has wide range of applications in image processing, arithmetic logic unit and VLSI signal processing. The future scope of this particular work can be extended in design of ALU's in RISC processor.

ACKNOWLEDGMENT

I am thankful to my parents, my brother, and my special friends to motivate me and to increase my confidence. My special thanks go to my friend Mr. Rajan Shah for his huge support. I am thankful to Altaf Darvadiya sir to guide me.

REFERENCES

[1] Prabir Saha, Arindam Banerjee, Partha Bhattacharyya, Anup Dandapat, "High speed ASIC design of complex multiplier using Vedic mathematics" , Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January, 2011, lIT Kharagpur, pp. 237-241.

[2] Prabir Saha, Arindam Banerjee, Partha Bhattacharyya, Anup Dandapat, "High speed ASIC design of complex multiplier using Vedic mathematics", Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January, 2011, lIT Kharagpur, pp. 237-241.

[3] Sumit Vaidya and Deepak Dandekar, "Delay-power performance comparison of multipliers in vlsi circuit design", International Journal of Computer Networks & Communications (IJCNC), Vol.2, No.4, July 2010, pp.47-56.

[4] P. Mehta, and D. Gawali, "Conventional versus Vedic mathematical method for Hardware implementation of a multiplier," in Proceedings IEEE International Conference on Advances in Computing, Control, and Telecommunication Technologies, Trivandrum, Kerala, Dec. 28-29, 2009, pp. 640-642.

[5] J. S. S. B. K. T. Maharaja, Vedic mathematics, Delhi: Motilal Banarsidass Publishers Pvt Ltd, (2010).

[6] Krishnaveni D and Umarani T.G. "Vlsi implementation of Vedic multiplier with reduced delay" International Journal of Advanced Technology & Engineering Research (IJATER), Volume 2, Issue 4, July 2012.

[7] G.Vaithiyanathan, K.Venkatesan, S.Sivaramakrishnan and S.Siva "Simulation and implementation of Vedic multiplier using vhdl code" International Journal of Scientific & Engineering Research Volume 4, Issue 1, January-2013.

