Design and Simulation of High Speed 8-bit Vedic Multiplier Using Barrel Shifter on FPGA

Arnav Gupta
Jaypee Institute Of Information Technology

Harshit Gupta
Delhi Technological University

Abstract

Vedic mathematics[1] is an ancient technique with unique approach and it has got different sutras. Here, in this paper 'Nikhilam Navatascaramam Dasatah' Sutra is been discussed , which is efficient in speed of the multiplier.This paper describes the implementation of an 8-bit Vedic multiplier enhanced in terms of propagation delay when compared with conventional multipliers. In our design we have utilized 8-bit barrel shifter which requires only one clock cycle for 'n' number of shifts. The design is implemented and verified using FPGA and ISE Simulator.The propagation delay comparison was extracted from the synthesis report and static timing report as well. The design could achieve propagation delay of 5.323ns which is significantly less than conventional multipliers.

Keywords– vedic mathematics,multiplier,barrel shifter,delay,fpga

1. INTRODUCTION

A multiplier is one of the key hardware blocks in most of applications such as digital signal processing [2], encryption and decryption algorithms in cryptography and in other logical computations [4]. With advances in technology, many researchers have tried to design multipliers which offer either of the following high speed, low power consumption, regularity of layout and hence less area or even combination of them in multiplier. The Vedic multiplier is considered here to satisfy our requirements. Multipliers are the core component of any DSP applications and hence speed of the processor largely depends on multiplier architecture. A multiplier of size n bits has n^2 gates. For multiplication algorithms performed in DSP applications latency and throughput are the two major concerns

Vedic mathematics [1] is an ancient technique which was used in the time of Vedas. It has got as many as 12 Sutras that can be used for different Arithmetic calculation. Vedic Sutras apply to and cover almost every branch of Mathematics. They apply even to complex problems involving a large number of mathematical operations.

Vedic mathematics has proved to be the most robust technique for arithmetic operations. In contrast, conventional techniques for multiplication provide significant amount of delay in hardware implementation of n-bit multiplier. Moreover, the combinational delay of the design degrades the performance of the multiplier. Hardware-based multiplication mainly depends upon architecture selection in fpga or asic[3][5]. Application of the Sutras saves a lot of time and effort in solving the problems, compared to the formal methods presently in vogue. Though the solutions appear like magic, the application of the Sutras is perfectly logical and rational.

Since the ever growing technology and increased complexity in the design demands for the optimized area and delay. Researchers are constantly working on towards the designing of optimized multiplier architecture. Critical path delay is the key factor in determining the speed of the multiplier, In simpler form multiplication can be developed using successive addition, subtraction and shifting operation as in literature. Different algorithms are implemented for the multiplier and each technique has got its own advantage and trade off in terms speed, area, and power consumption.

Multiplier implementation using Fpga has already been reported using different multiplier architectures but the performance of multiplier was improved in proposed design by employing Vedic multiplier using modified "Nikhilam Navascaramam Dasatah" sutra. The architecture in [1] is modified using barrel shifter by which significant amount of clock cycles are reduced by virtue of which the speed increases. The performance of the proposed multiplier is compared with the previously implemented multipliers on Fpga.

2. Nikhilam Dasatah Sutra

The Nikhilam Sutra literally means "all from 9 and last from 10". It is more efficient when the numbers

involved are large. The Nikhilam Sutra algorithm is efficient for multiplication only when the magnitudes of both operands are more than half their maximum values. For n-bit numbers, therefore both operands must be larger than 2n-1. Nikhilam Sutra is explained by considering the multiplication of two single digit decimal numbers 8 and 7 where the chosen base is 10 which is nearest to and greater than both these two numbers [6].
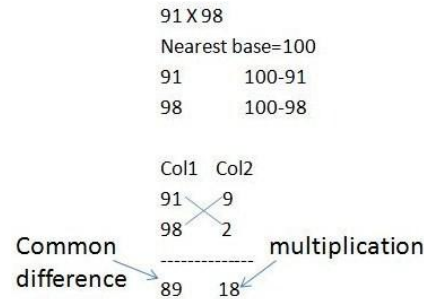

Fig1. Multiplication of Two Digit Decimal Numbers (91 x 98) using Nikhilam Sutra.

As shown in Fig.1, the multiplier and the multiplicand are written in two rows followed by the differences of each of them from the chosen base, i.e., their compliments. There are two columns of numbers, one consisting of the numbers to be multiplied (Column 1) and the other consisting of their compliments (Column 2). The product also consists of two parts which are demarked by a vertical line for the purpose of illustration. The right hand side (RHS) of the product can be obtained by simply multiplying the numbers of the Column 2 i.e., (2x9=18). Common difference is found by subtracting either (91-2=89) or (98-9).

3. PROPOSED MULTIPLIER ARCHITECTURE DESIGN

Assume that the multiplier is A and multiplicand is B. The mathematical expression for nikhilam sutra is given below.

$$P = A.B = 2^{t_2}(A \pm Z_2 * 2^{(t_1 - t_2)}) \pm Z_1 * Z_2 \qquad (1)$$

Where t_1, t_2 are the maximum power index of input numbers A and B respectively. Z_1 and Z_2 are the residues in the numbers X and Y respectively.

For implementing above expression, hardware is divided into three blocks.

1. Power index determinant module(PID)
2. Base selection module(BSM)
3. Multiplier.

The base selection module (BSM) is used to select the maximum base with respect to the input numbers. The 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), subtractor, barrel shifter, as sub-modules in the architecture.

3.1 Base selection module.
It comprises of power index determinant (PID) as the sub-module along with barrel shifter, adder, average determinant, comparator and multiplexer.
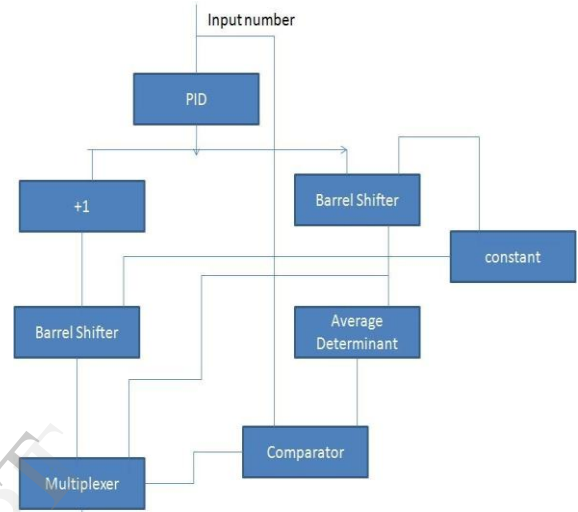

Fig2. Base Selection Module

3.2 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 decrementer is the required power index of the input number.
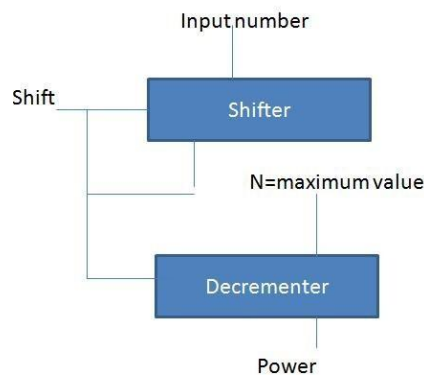

Fig3. Power Index Determinant

3.3 Multiplier Architecture

The architecture implements the equation (1). Base is obtained from BSM when the numbers are provided to it.The output of BSM and the input numbers 'A' and 'B' are fed to the subtractors.Subtractor block provides residual part Z_1 and Z_2.Power Index Determinant(PID)receives values from BSM of respective input numbers. The power of the base is found by sub-section of PID.The outputs of subtractor are fed to the multiplier that feeds the input to the second adder or subtractor[6]. Similarly the outputs of PID are fed to the third subtractor that feeds the input to the barrel shifter. The input number 'A' and the output of barrel shifter are rendered to first adder/subtractor and the output of it is applied to the second barrel shifter which will provide the intermediate value. The last sub-section of this multiplier architecture is the second adder/subtractor which will provide the required result.
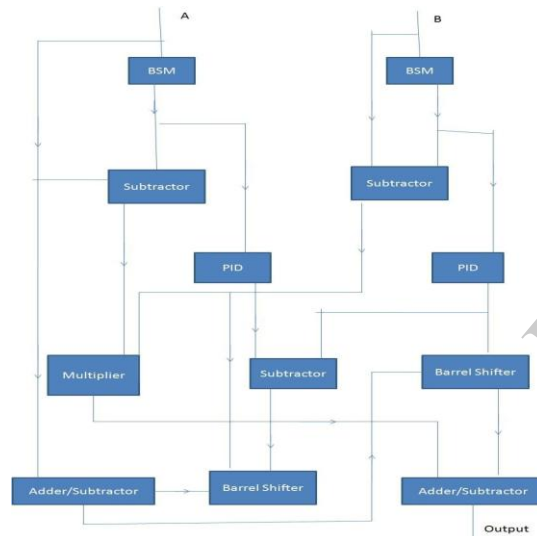

Fig4. Multiplier Architecture

4. RESULTS

Table 1. draws comparison between conventional multipliers and proposed design. Around 89.6% of reduction in delay can be observed from the proposed design with respect to array multiplier and 74.68% of reduction in delay with respect to conventional Vedic Multiplier in Table I.

Table 1. Comparison between various multipliers

Multiplier Type	Vedic Multiplier	Array Multiplier	Proposed Multiplier
Delay(ns)	21.03	51.43	5.323

5. CONCLUSION

Upon completion of the project, we achieve a high percentage of reduction in the propagation delay when compared to array multiplier and conventional Vedic multiplier implementation on FPGA.The wide ranges of applications of multiplier unit can be witnessed in VLSI and signal processing applications.The project can be extended to the power analysis of the multiplier.

6. REFERENCES

[1] Jagadguru Swami Sri Bharati Krisna Tirthaji Maharaja, *Vedic Mathematics: Sixteen Simple Mathematical Formulae from the Veda*, Delhi (1965).

[2] S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and 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), pp.1-6.

[3] Himanshu Thapaliyal and M.B Srinivas, "VLSI Implementation of RSA Encryption System Using Ancient Indian Vedic Mathematics", Center for VLSI and Embedded System Technologies, International Institute of Information Technology Hyderabad, India.

[4] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaja, "Vedic Mathematics: Sixteen simple Mathematical Formulae from the Veda", Delhi(1965).

[5] Devika, K. Sethi and R.Panda, "Vedic Mathematics Based Multiply Accumulate Unit," 2011 *International Conference on Computational Intelligence and Communication Systems*, CICN 2011, pp.754-757, Nov.2011.

[6] Q. LI, G. LIANG, and A. BERMAK, "A High-speed 32-bit Signed/Unsigned Pipelined Multiplier," *IEEE 5th Int. Sym-posium Electronic Design, Test and Application*, pp.207- 211, Jan. 2010.

[7]S.S.Kerur,Prakash.Narchi,Jayashree.C.N,Harish.M. Kittur and 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).*