IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 4, Issue 3, Ver. II (May-Jun. 2014), PP 54-59
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

"FPGA implementation of high speed 8 bit Vedic Multiplier using Fast adders"

Mrs.Toni J.Billore, Prof.D.R.Rotake
Research scholar,GHRIETW,Nagpur.
Assistant professor, GHRIETW,Nagpur

Abstract: *This paper describes the implementation of an 8-bit Vedic multiplier using fast adder enhanced in terms of propagation delay when compared with conventional multiplier. In our design of 8 bit Vedic multiplier using fast adder, we have utilized 8-bit barrel shifter which requires only one clock cycle for 'n' number of shifts. The design of 8 bit Vedic multiplier using barrel shifter is implemented and verified using FPGA and ISE Simulator. The core used here was implemented on Altera Cyclone® II 2C20 FPGA device software. The propagation delay between 8 bit Vedic multiplier using barrel shifter using barrel shifter and using fast adder comparison was extracted from the synthesis report and static timing report as well. The design which is implemented here could achieve propagation delay of 6.781ns using barrel shifter block in base selection module and multiplier of architecture used. In our project, we make a comparison between performance analysis of 8 bit Vedic multiplier using barrel shifter and using fast adder.*

Keywords: *Fast adder, Barrel shifter, base selection module, Propagation delay, power index determinant*

I. Introduction

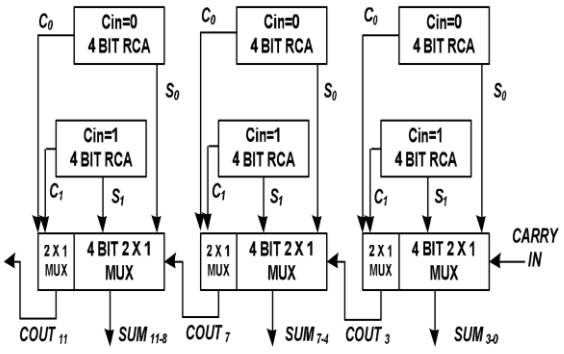
Arithmetic operations such as addition, subtraction and multiplication are used in various digital circuits to speed up the process of computation. Arithmetic logic unit (ALU) is also implemented in various processor architectures like RISC, CISC etc., In general, arithmetic operations are performed using the packed-decimal format. This means that the fields used here are first converted to packed-decimal format prior to performing the arithmetic operation, and then converted back to their specified format prior to placing the result in the result field.

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. However, the combinational delay of the design degrades the performance of the multiplier. Hardware-based multiplication mainly depends upon architecture selection in FPGA or ASIC.

In this work we have put into effect a high speed Vedic multiplier fast adder and barrel shifter. The sutra was implemented by modified design of "Nikhilam Sutra" due to its feature of reducing the number of partial products. The barrel shifter used at different levels of design drastically reduces the delay when compared to conventional multipliers.

The hardware implementation of 8 bit Vedic multiplier using barrel shifter contributes to adequate improvement of the speed in order to achieve high outturn.

In our design, we are using fast adder for reducing the propagation delay, area or power consumption of 8 bit vedic multiplier. In our design, we are use carry select adder & conditional sum adder.



Carry select & conditional sum adder

The sum & carry can be calculated using following formulas:

S0i = ai ex-or bi
C0i = ai.bi
S1i = ai ex-nor bi
C1i = ai + bi

II. Literature Survey

1. Pavan Kumar U.C.S1 , SaiprasadGoud 2 A, A.Radhika ,"FPGA Implementation of high speed 8-bit Vedic multiplier using barrel shifter", In this paper, authors uses a nikhilam sutra , for implementing this sutra they uses the multiplier architecture which contains the Base index Module & Power index Module which is the integral part of this project. In this work we have put into effect a high speed Vedic multiplier using barrel shifter. we have utilized 8-bit barrel shifter which requires only one clock cycle for 'n' number of shifts.

2. Hamid M. Kamboh , Shoab A. Khan, "FPLA Implementation of Fast Adder", This paper was proposes implementation of an high data rate adder on Field Programmable Gate Arrays (FPGA). Digital Signal processing applications are characterized by the data rate or the throughput of the system. Experimental results have shown that traditional hardware optimizations perform adversely after implementation on FPGAs, where the Ripple Carry Adder has shown clock speed gain of a minimum of 18.24% using 50% lesser resources for two operand multipliers.

3. Sumitvaiydha , pramod Dandekar, "Delay-power performance comparison of multipliers in vlsi circuit design" A typical processor central processing unit devotes a considerable amount of processing time in performing arithmetic operations, specially multiplication operations. Multiplication operation is one of the basic arithmetic operations and it requires substantially more hardware resources and processing time than addition and subtraction. In fact, 8.72% of all the instruction in typical processing units is multiplication. In this paper, we studied comparative of different multipliers is done for low power requirement and high speed. The paper gives information of "UrdhvaTiryakbhyam" algorithm of Ancient Indian Vedic Mathematics which is utilized for multiplication of two or more operands for improving the speed, area parameters of multipliers. Vedic Mathematics has been suggests one more formula for multiplication of large number i.e. "Nikhilam Sutra" which can also increase the speed of multiplier by reducing the number of iterations

4.Ramachandran.S,KirtiPande,"Design, Implementation and Performance Analysis of an Integrated Vedic Multiplier Architecture": udhrva sutra performs faster than for small inputs and nikhilam sutra for larger inputs. Here a novel integrated Vedic multiplier is proposed.

III. Multiplier Architecture Design

Assume that the multiplier is 'X' and multiplicand is 'Y'. Though the designation of the numbers is different but the architecture implemented is same to some extent for evaluating both the numbers.

The mathematical expression of modified nikhilam sutra is given below.

P=X*Y=(2^k2)*(X+Z2*2^(k1-2))+Z1*Z2. (1)

Where k1, k2 = the maximum power index of input numbers X and Y respectively.

Z1 and Z2 = the residues in the numbers X and Y respectively.

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

i. Base Selection Module
ii. Power index Determinant Module
iii. 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), subtractor, barrel shifter, adder/subtractor as sub-modules in the architecture.

A. Base selection module.

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

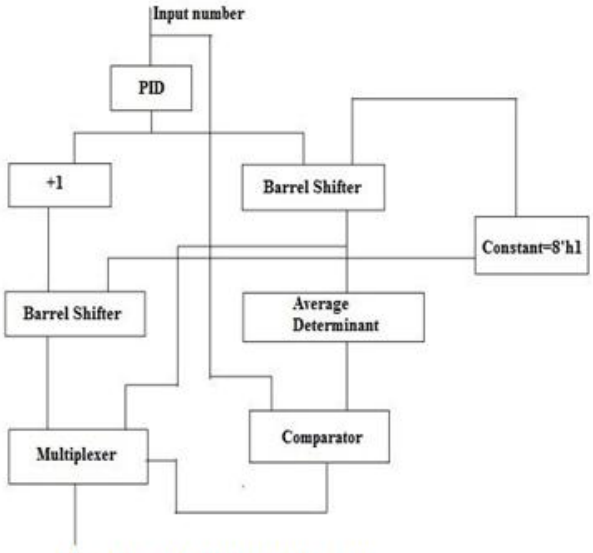
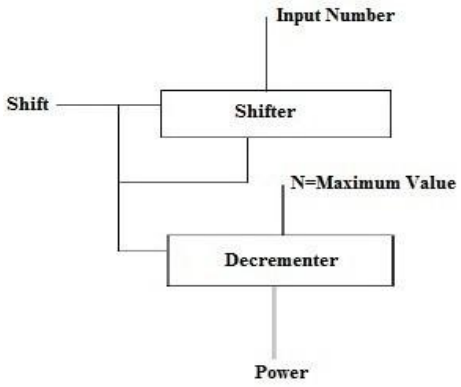


Fig.1 Base Selection Module, *BSM*

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.

B. Power index determinant.



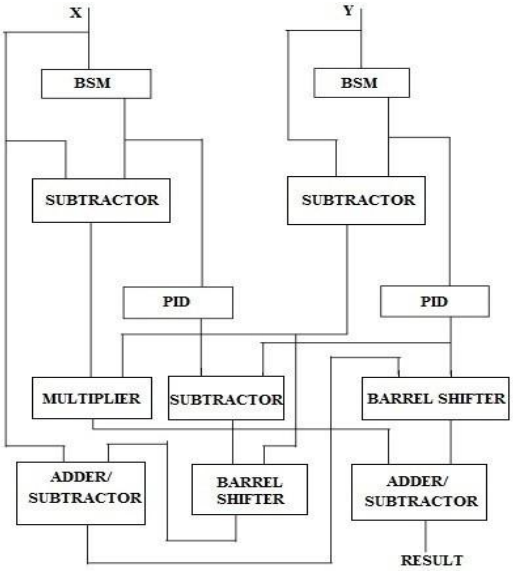
The input number is fed to the shifter which will shift the input bits by one clock cycle. The shifter pin is assigned to shifter to check whether the number is to be shifted or not. In this power index determinant (PID) the sequential searching has been employed to search for first '1' in the input 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 decrementer is the required power index of the input number.

C.MultiplierArchitecture

The base selection module and the power index determinant form integral part of multiplier architecture. The architecture computes the mathematical expression in equation1.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 subtractors. 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.

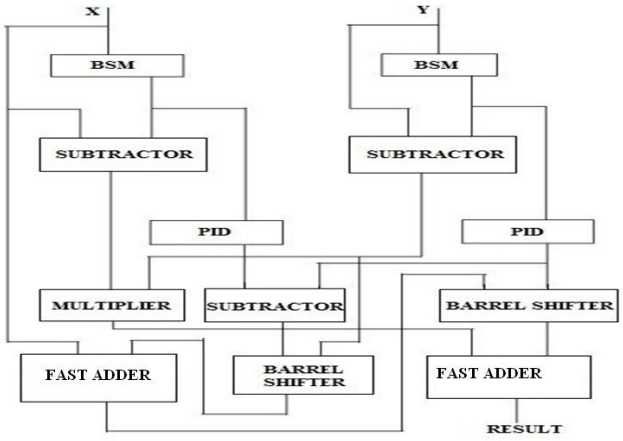
The sub-section of power index determinant (PID) is used to extract the power of the base and followed by subtractor to calculate the value. The outputs of subtractor are fed to the multiplier that feeds the input to the second adder or subtractor. Likewise the outputs of power index determinant are fed to the third subtractor that feeds the input to the barrel shifter. The input number 'X' and the output of barrel shifter are 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 used in this multiplier architecture is the second adder/subtractor which will provide the required result.



Multiplier Architecture

D. Proposed Architecture

In our proposed design, we replace fast adder instead of adder/substractor in the architecture of basic multiplier.



Proposed multiplier architecture

IV. SimulationResultsAndDesignAnalysis

Comparison between conventional multipliers, multiplier using simple adders and proposeddesign is been projected below. Around 75% of reduction in delay can be observed from the proposed design with respect to array multiplier in Table I. 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.

TableI . Delay Comparison Between Various Multipliers Implemented On Fpga[3].

Multiplier type	Conventional Multiplier(Array)	Vedic Multiplier	Vedic Multiplier	Proposed Multiplier
Delay(ns)	43.42	27	6.781	3.382

A. Timing Summary

Speed grade: -3
Minimum period: No path found
Maximum input arrival time before clock: No path found
Maximum output required time after clock: No path found
Maximum combinational path delay: 3.382ns

B. TimingDetails

All values displayed in nanoseconds (ns)
Timing constraint: Default period analysis for clock 'clk'
Clock period: 3.382ns (frequency: 147.47 MHz)
Total no of paths / destination ports: 256 / 16
Delay: 3.382ns (levels of logic=3)
Source: yin<7> (PAD)
Destination: mout<15> (PAD)
Data Path: yin<7> to mout<15>

Table.II Power Report of Proposed Design

	Total	Dynamic	Quiescent
Supply power(W)	0.546	0.00	0.546

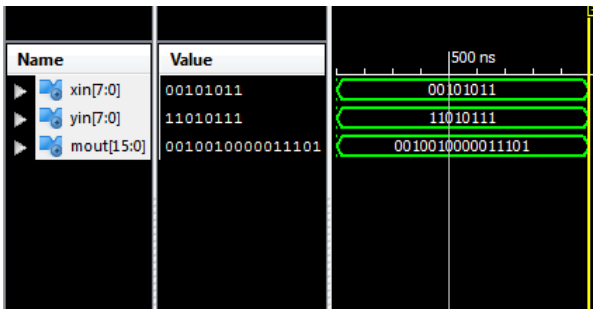
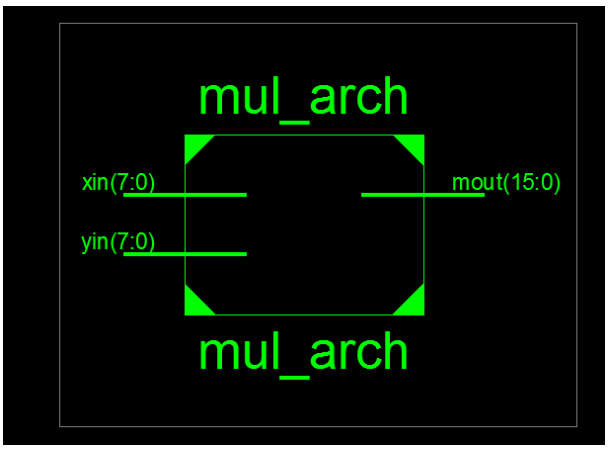


Fig.4 Post-Route Simulation



RTL View of multiplier

V. Conclusion

In our design, efforts have been made to reduce the propagation delay and achieved 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.

References:

[1]. U.C.S.I. P.Kumar, S.Goud A2, A.Radhika,2013: 'FPGA Implementation of high speed bit Vedic multiplier using barrel shifter', ICEETS Journal, 14-17.
[2]. H. M. Kamboh, S. A. Khan,2012 'FPLA Implementation of Fast Adder', ICCCT journal, 1324-1327.
[3]. J. S. S. B. K. T. Maharaja, Vedic mathematics, Delhi: MotilalBanarsidass Publishers Pvt Ltd,(2010).
[4]. M. Ortiz, F. Quiles, J.Hormigo, F.J. Jaime, J. Villalba, E. L. Zapata, 2009 'Efficient implementation of carry-save adders in FPGAs',ICASSAP journal, 207-210.
[5]. P.Saha, A Banerjee, P. Bhattacharyya, A.Dandapat, 2011,'High speed ASIC design of complex multiplier using vedicmathematics',IJERA journal, 3, 237-241.
[6]. P. Mehta, and D. Gawali, 2009, 'Conventional versus Vedic mathematical method for Hardware implementation of a multiplier', ICACCTT journals, 28-29,.
[7]. Suhaili,O.Sidek, 2004, 'Design and implementation of reconfigurable alu on FPGA',ICECE journal, 56-59.
[8]. S.Vaidya, D.Dandekar, 2010, 'delay-power performance comparison of multipliers in vlsi circuit design', IJCNC journal, 2,47-56.