ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

Implementing Vedic Formula for Square Computation in FPGA - A Survey

[1] Subrahmanya Bhat K G, [2] Dr. Jose Alex Mathew
[1] Student , M.Tech VLSI Design and Embedded Systems, [2] Professor
Department of ECE
Sahyadri College of Engineering and Management, Mangalore, India.

Abstract: **The performance of any processor depends upon its speed, area and delay. Vedic mathematics is an ancient mathematics that has a unique method of computation based on simple rule. This paper reviews Vedic mathematics based high speed operations. The Vedic formula Dwandwa Yoga or Duplex squarer is based on duplex property which is used for squaring of numbers.**

Keywords **: Vedic, Square, Dwandwa Yoga ,Duplex squarer, Multiplication, fast computation, FPGA**

I. INTRODUCTION

Digital systems are replacing ol analog systems in almost all the fields due to fast and efficient performance. Digital systems work on digital data. Arithmetic Logic unit (ALU) is the important unit in Digital processors that performs basic arithmetic operations like addition, subtraction, multiplication and logical operations on digital data. Today"s processors operate at very high clock speed and also minimized in silicon area by using VLSI design techniques. Digital systems can be designed to perform operations in the field of image processing, signal processing and many other applications which involve number of equations to be solved. These equations may contain general operations such as addition/subtraction, multiplication/division, squaring, etc. For the faster operations it is required to minimize time delays in each computation. There are much architecture designed for these operations and continuous researches are being done to develop new architectures with less area, less power dissipation and which can compute faster results[1]. Vedic mathematics is an ancient Indian technique and consists number of formulae for fast calculations. It is already proved that multiplier architecture designed using Vedic logic have better performance parameters. Still there are many other formulae in Vedic mathematics which needs to be implemented in digital systems. Usually Square of a number is calculated by direct multiplication. But in Vedic mathematics square of a number can be computed without direct multiplication. Thus this project is to improve data computation performance by implementing Vedic formulae in digital system.

II. LITERATURE REVIEW

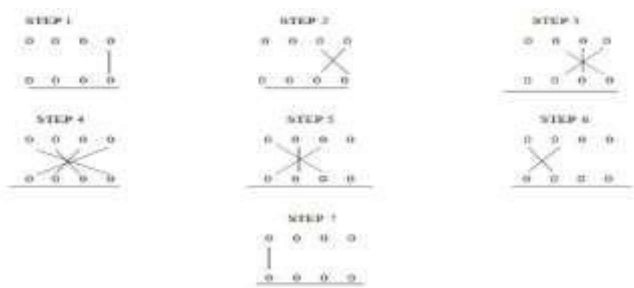
Generally square of a number is computed by multiplying the number to itself. If the number is larger or in terms of digital data 16 bit or 32 bit number multiplication consumes more time, power and silicon area. Existing method for square computation is directly multiplying the data to itself. This is performed by using different multiplier architectures in digital systems. The execution speed of multipliers depends on two factors, one is the semiconductor technology used and the other is the architecture of multiplier. Multiplication process involves a series of repeated additions. Therefore adders are the basic building blocks of digital multipliers. Increasing the speed of the adder, results in an increase in the speed of the multipliers. The area of the multipliers can be minimized by reducing the number of transistors required for implementing full adder circuits. Some of the existing architecture for multiplication are Bit Array Multiplier, Baugh Wooley Multiplier, Wallace Tree Multiplier,etc[2]. The array multiplier works based on the principle of add and shift algorithm. In this Multiplier, the partial products can be generated using AND gates and the summation of partial products can be performed using Full Adders and Half Adders. For an n x n array multiplier, n2 AND gates are required to compute the partial products and the addition of partial products requires n(n-2) full adders and n half adders. In Conventional Multipliers Final product is obtained by adding partial products along with its previous stage carryout. With this great amount of delay exists. As number of bits increases Hardware complexity, delay and power consumption increases.

IFERP
connecting engineers... developing research

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

The proposed approach is square computation using Vedic mathematics. In case of Vedic Multiplier large modules can be divided into sub modules and any arithmetic operations can be performed in parallel. Thus it computes results faster than other conventional multiplier architectures [1][2]. The 4x4 Vedic multiplication technique is shown in the fig.1.



The word „Vedic" was driven from the word „Veda" which is ancient store-house of all knowledge. Vedic mathematics provides the solution to the problem of long computation time by reducing the time delay needed for the operations to be performed. It has originated from "Atharva Vedas" the fourth Veda. Atharva Veda mainly deals with the branches like engineering, mathematics, sculpture, medicines and all other sciences. Vedic mathematics deals with all areas of mathematics either it is pure or applied [3]. It was rediscovered from the Vedas between 1911 and 1918 by Sri Bharati Krishna Tirtha Swamiji.

III.IMPLEMENTATION

Vedic mathematics has many formulae for square computation. One of them is the Dwandwa Yoga or Duplex squarer is based on duplex property[3] which is used for squaring of numbers. This technique is derived from the Vedic multiplier architecture. In this technique for n digit number, 2n-1 derivatives are created by using following formulae [4]. Let „abcd" be a 4 word data. Its derivatives are

$D(d) = d^2$
$D(cd) = 2cd$
$D(bcd) = 2bd + c^2$
$D(abcd) = 2ad + 2bc$
$D(abc) = 2ac + b^2$
$D(ab) = 2ab$
$D(a) = a^2$

By adding these derivatives in required bit position we can calculate the square of the data. If we consider 4 digit hexadecimal number (16 bit data), we require 7 derivatives. All the derivatives can be calculated by using 4x4 bit multipliers, adders and shifters. Thus usage of 16x16 bit multiplier can be avoided. In Vedic logic pipelining can be achieved since all the derivatives are independent to each other.

Ex: considering hexadecimal number 1234

$D(4) = 4^2 = 10$
$D(34) = 2x3x4 = 18$
$D(234) = 2x(2x4)+3^2 = 19$

$D(1234) = 2x(1x4)+2x(2x3) = 14$
$D(123) = 2x(1x3)+2^2 = 0A$
$D(12) = 2x(1x2) = 04$
$D(1) = 1^2 = 01$

```
      10
     18
    19
   14
  0A
 04
01
--------------
 014B5A90
```

This paper compares different methods of implementing verilog code for computing square of 16 bit data in Vedic technique. First case is Square of a 16 bit data is computed using 16x16 Vedic multiplier. In the next cases 16 bit Vedic Squaring Formula „Dwandwa Yoga" is implemented using verilog code in two different types. Vedic square (type1) is designed using 4x4 Vedic multiplier, 2x2 Vedic multiplier, half adders and full adders. Vedic square (type2) is designed using 4X4 array multiplier and full adders. These three cases perform squaring operation and the observed test bench waveform is shown in fig.2

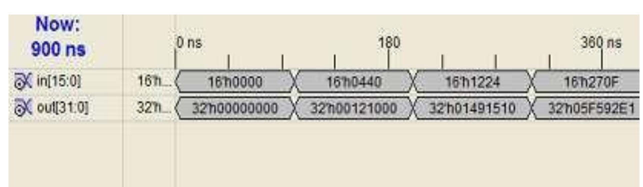


Fig.2: Test bench waveform observed in all three cases of squaring methods.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 5, May 2017

The total delays observed in three cases are listed below in Table.1.

VEDIC SQUARE LOGIC	Delay (ns)
16x16 Vedic Multiplier	31.789
Vedic Square (type 1)	45.748
Vedic Square (type 2)	39.386

Table.1: Delay offered in different type of Vedic square computation of 16 bit data.

It is observed that vedic multiplier produces results with less delay. Vedic square formulae „Dwandwa Yoga" offers different delay for different designs. But code size in Vedic Square(type1) and Vedic Square(type2) is reduced compared to Vedic multiplier code.

IV. FUTURE WORK

Dwandwa Yoga or Duplex Squarer method isapplicable for all data. But Vedic mathematics also contains many other formulae for square computation which can be applied on specific conditions. One such technique is based on the Vedic sutra of "Ekadhikena Purvena". It means "one more than the previous". It is suitable for squaring of numbers ending with „5".
Ex: 352 =1225
Number 35 is ending with digit 5. Thus according to formula right part or last 2 digits of the result is 25 and left part of the result is left digit (3) multiplied with one number more than it. I.e. 3x4=12. Therefore total result is 1225.
Another such formula is Yavadunam sutra. It is a specific and shortcut to square numbers using Vedic Mathematics whenever number is closer to power of 10. In this method the number which is power of 10 is considered as the base. The difference between the base and the number is called deviation. Adding deviation to the number gives left part of the result. Square of deviation forms right part of the result. By concatenating these actual result can be obtained.
Ex: 132 =169
Here base is 10 and deviation is 3. By adding deviation (3) to the number(13) left part of the result is formed. i.e. 13+3=16. Square of the deviation is 32 =9. By concatenating these, the actual result can be obtained as 169. These generalized and condition oriented Vedic formulae require lesser steps for computation than existing methods

V.CONCLUSION

In Vedic technique large modules can be divided into sub modules and any arithmetic operations can be performed in parallel. Squaring operation performed using Vedic multiplier or Vedic squarer (Dwandwa Yoga Formula) computes result faster than other conventional multiplier algorithms.

REFERENCES

[1]. K. R. Gavali and P. Kadam, "VLSI design of high speed Vedic Multiplier for FPGA implementation," 2016 IEEE International Conference on Engineering and Technology (ICETECH), Coimbatore, 2016, pp. 936-939.

[2]. S. Sabeetha, J. Ajayan, S. Shriram, K. Vivek and V. Rajesh, "A study of performance comparison of digital multipliers using 22nm strained silicon technology," 2015 2nd International Conference on Electronics and Communication Systems (ICECS), Coimbatore, 2015, pp. 180-184.

[3]. R. Sharma, M. Kaur and G. Singh, "Design and FPGA implementation of optimized 32-bit Vedic multiplier and square architectures," 2015 International Conference on Industrial Instrumentation and Control (ICIC), Pune, 2015, pp. 960-964.

[4]. V. Kunchigi, L. Kulkarni and S. Kulkarni, "Low Power Square and Cube Architectures Using Vedic Sutras," 2014 Fifth International Conference on Signal and Image Processing, Jeju Island, 2014, pp. 354-358.