International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 4 106 - 109

Design of Vedic ALU for 16-Bit Processor

Mugdha Fadnavis
School of Electronics Engineering
(SENSE)
VIT UNIVERSITY CHENNAI CAMPUS
Chennai, India
mugdhashamkant.fadnavis2015@vit.ac.in

Augusta Sophy Beaulet P.
School of Electronics Engineering
(SENSE)
VIT UNIVERSITY CHENNAI CAMPUS
Chennai, India
sophyguna@gmail.com

***Abstract*—** The main objective of this project is to design a VEDIC ALU for a 16-bit processor. The Arithmetic operations in the ALU are performed using the few of the 16 sutras of Vedic Mathematics. Even though, addition and subtraction sutras are similar to the conventional methods, multiplication and division methods are derived and implemented successfully in ALU using the Vedic sutras. The advantage of implementing Vedic sutras for Multiplication and Division is the faster speed and reduced hardware. It also reduces the total power consumption as compared to the conventional ALU which is currently being used. The platform for designing the Vedic ALU is XILINX ISE and the preferred language is Verilog. Vedic Mathematics is the Ancient Indian technique of Mathematics, derived from ancient Vedas, that was rediscovered in 20th century by Swami Sri Bharati Krishna Tirthaji Maharaj.

***Keywords*—** *ALU design, Vedic Mathematics, Urdhva-Tiryagbhyam, Nikhilam Method, Derived Division Method from Nikhilam Sutra, Xilinx ISE.*

*****

I. INTRODUCTION

In this world of high-speed performance, everyone is trying to improve the speed of operation. The digital industry is not an exception in this case. With the evolution of new Semiconductor technologies, the performing speed of the devices such as Microprocessor, Microcontroller and many other digital devices is also augmented. While designing a high-speed processor, the performance of ALU should also be taken into consideration. Thus, enhancing the speed of the ALU, the overall performance of the processor can be amplified. Therefore, to improve the calculation speed of the ALU, Vedic Mathematics Sutras are used in this work [1]. Nowadays, the competition between the size and performance of the processors in the VLSI industry has increased and it is not new. While making a processor to be a good performer, a lot of things are taken into account. Thus, the high performance of a processor depends upon the integral parts that the processor is composed of.

ALU i.e. Arithmetic Logic Unit is also one of the major contributors to the processor's performance which helps it to perform the arithmetic calculations even faster. Thus, the working of the processor can be improved by improving the performance of its ALU. Generally, the conventional methods of calculations are used in the processors. The Arithmetic operations like Addition, Subtraction, Multiplication and Division are executed using the traditional but complex methods of Booth's Algorithm. In this paper, new methods of arithmetic calculations have been proposed, which are derived from the Sutras of Ancient Vedic Mathematics.

II. CONCEPT OF VEDIC ALU

As discussed earlier, performance of the ALU is prime focus for functions such as arithmetic calculations to be executed properly. Hence to improve the performance of the ALU, a new Arithmetic unit is suggested which is completely based on the principles of Vedic Mathematics.

In this Vedic ALU, all the arithmetic functions are performed using Vedic Maths *sutras*. Such as, Addition – "*Ekadhiken Purven*" (One more than the Previous)
Subtraction – "*Ekanyunen Purven*" (One less than the Previous)
Multiplication – "*Urdhva Tiryagbhyam*" (Vertically and Crosswise)
Division – "*Nikhilam Navatashcharamam Dashatah*" (All from Nine and last from ten) and "*Ekanyunen Purven.*

A. *Vedic Multiplication Method*

While performing the multiplication using Vedic Maths Sutras, the Sutra "Urdhva Tiryagbhyam" is used. "Urdhva Tiryagbhyam" means 'vertically and crosswise'. As the name suggests, the given method performs the multiplication first vertically between the two digits and then crosswise between the digits of two numbers.

B. *Working of "Urdhva Tiryagbhyam"*

The method can be explained as follows: **Ex: 426 x 115:**

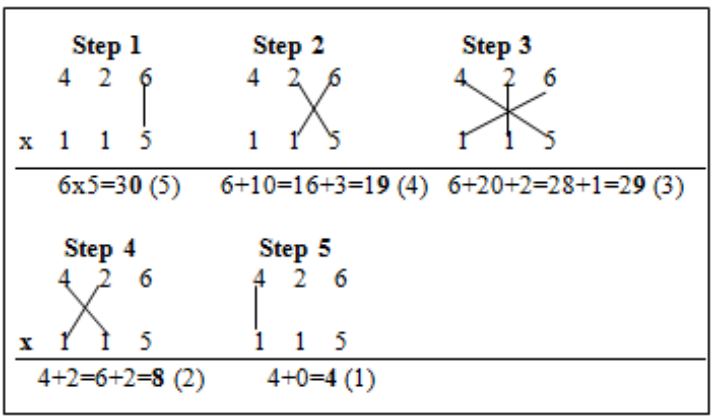

Fig. 1: Algorithm of "Urdhva Tiryagbhyam" Method in Decimal System

This gives the product as: **4 8 9 9 0**

Similarly, multiplication can also be done in Binary system using Vedic Method Urdhva Tiryagbhyam[2]:

Example: 110x111

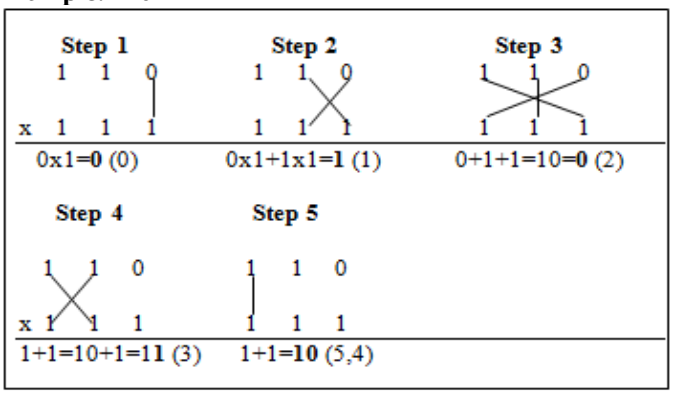

Fig. 2: Algorithm of "Urdhva Tiryagbhyam" Method in Binary System

This gives the product as: **1 0 1 0 10**

In the above method, first the LSB of multiplicand is multiplied with LSB of Multiplier, which gives the LSB of the product. This is 'Vertical' multiplication. In next step, the LSB of multiplicand is multiplied with next higher bit of multiplier and the LSB of multiplier is multiplied with next higher bit of multiplicand. This is the 'crosswise' multiplication which gives the next bit of the product from right to left. This method continues till the MSB of both the multiplicand and the multiplier multiply each other to gives the MSB of the product, which is again a 'vertical' multiplication.

Following the line diagram mentioned above, the algorithm for the Vedic multiplication can be given as:

Algorithm For Urdhva Tiyagbhyam Method for 4x4 Bit Multiplication:

A=A3A2A1A0
B=B3B2B1B0

Then, A3 A2 A1 A0
X B3 B2 B1 B0
P7 P6 P5 P4 P3 P2 P1 P0

Where,
P0=A0.B0
P1=A1.B0+A0.B1
P2=A0.B2+A1.B1+A2.B0+C(P1)
P3=A0.B3+A3.B0+A2.B1+A1.B2+C(P2)
P4=A1.B3+A3.B1+A2.B2+C(P3)
P5=A3.B2+A2.B3+C(P4)
P6=A3.B3+C(P5)
P7=C(P6)

C(P) is the carry of the multiplication result
Similarly, the same algorithm can be implemented for 8x8 bit multiplication as:
A= A7A6A5A4 A3A2A1A0
B= B7B6B5B4 B3B2B1B0

Let, A and B be split into upper and lower nibbles as
C1=A7A6A5A4
C0= A3A2A1A0
D1=B7B6B5B4
D0=B3B2B1B0

Then, C1 C0
X D1 D0
P3 P2 P1 P0

Where,
P0=A0*B0
P1=A1*B0+A0*B1+C(P0)
P2=A1*B1+C(P1)
P3=C(P2)

C(P) is the carry of multiplication result and the symbol '*' follows the 4x4 bit multiplication.

Thus, we can use either 8x8 Bit Multiplication or 4x4 Bit Multiplication for 16x16 Bit multiplication.

C. Vedic Division Method

The Vedic method which is used for Division is *'Nikhilam'* method. The *sutra* is called as "*Nikhilam Navatashcharamam Dashtah*." It means that 'all from nine and last from ten.' While dividing a number and the number being close to the base, the method suggests us to take the complement of the divisor.

D. Working of Nikhilam Method

In this method, there is no need for the division. It is simply done by the multiplication and addition. Here, the complement of the divisor is obtained called Modified Divisor and the rest calculation is done by the MD. The dividend is bifurcated into two parts so that the RHS of the dividend contains equal number of digits to those of the Modified Divisor.

The method can be better explained with the example below.

Example:

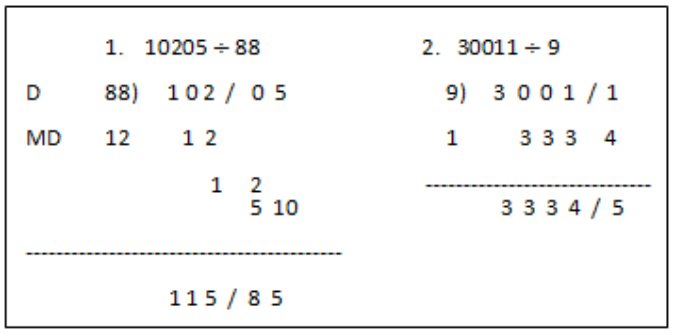

Fig.3: Algorithm of "Nikhilam" Method in Decimal System

III. PROPOSED METHOD OF VEDIC DIVISION

In the given Vedic Division method, i.e. *Nikhilam* method, the method requires frequent multiplications. Also the multiplication with modified divisor becomes difficult when

the MD is not closer to the base but a greater number. The dividend is to be bifurcated into two parts, which requires two registers to store the number. This leads to complicated multiplication as the dividend is split into two parts.

On the other hand, Nikhilam method is most appropriate for the decimal number system. It becomes complex during its execution on the binary number system.

Hence, a new method is derived from the Nikhilam method which is applicable for the binary number system[3]. In this method, only shifting and subtraction operations are needed to be performed.

The algorithm for the proposed method is given in the Fig. below:

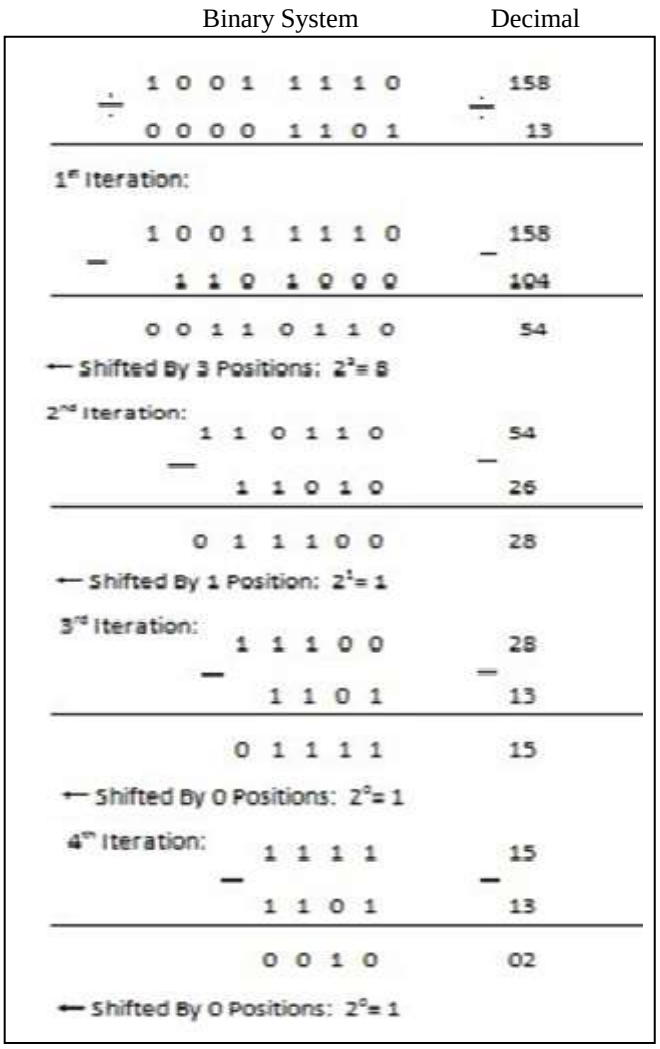

Fig.4: Algorithm of Proposed Vedic Division method in Binary Number System

Results: Quotient= 8+2+1+1=12(1100)

Remainder= 2(0010)

A. *Working of Proposed Method*

In this method, the divisor is shifted to the left by 1-Bit less to the MSB of the dividend. After that, regular subtraction is performed. Result of the subtraction is now the New Dividend. Again we shift the divisor to left up to 1-bit less of the MSB of the New Dividend. This process is continued till we get a number which is not divisible by the divisor. This number is the ultimate Remainder of the Division. While shifting the divisor to the left, we get the sub-quotients as 2^n , where n=number of shift. The ultimate quotient is obtained by adding all the sub-quotients.

B. *Advantages of derived method*

- The derived method can be used for binary systems.
- We can divide any number of bits of dividend by any number of bits of divisor. There is no such restriction on the divisor as such in Booth's Algorithm.
- We are performing Shifting, Subtraction and addition. For which very few registers would be required.
- No carry or borrow are generated during addition or subtraction. Therefore, the flag register is not affected much.

IV. RESULTS

The above mentioned algorithms for Vedic Multiplication and Vedic Division are implemented in Xilinx ISE tool and the simulation results are checked using ISE simulator. The preferred language for implementation is Verilog HDL. The design is implemented on Spartan 3E and the specifications of the device are given as:
Family: Spartan 3E
Device: XC3S250E
Package: PQ208
Speed Grade: -5

Thus, the simulation results of both the algorithms can be shown as:

A. *Simulation Results of Vedic Multiplier*

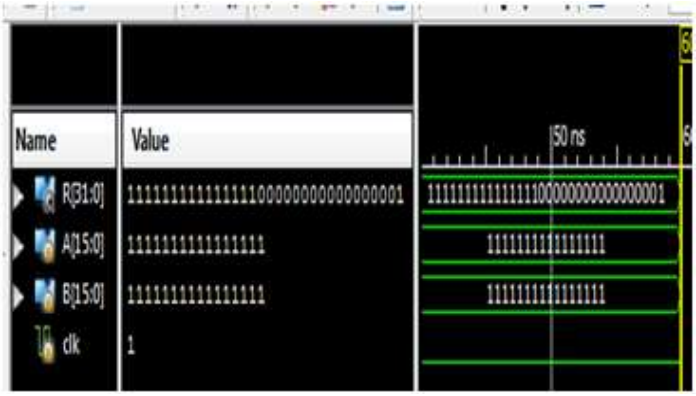

Fig.5: Simulation Results of 16x16 Bit Multiplication using Vedic Multiplication Method

The inputs fed to the Vedic Multiplier and the obtained outputs are as below:
Input A=16-Bit input data= 11111111 11111111 (b)
Input B=16-Bit input data= 11111111 11111111 (b)
Output R= 32-Bit Output Data= 1111 1111 1111 1110 0000 0000 0000 0001 (b)

Here, we can see that the multication result of two 16-Bit numbers is verified in the 32-Bit output.

B. Simulation Results of Vedic Division Method

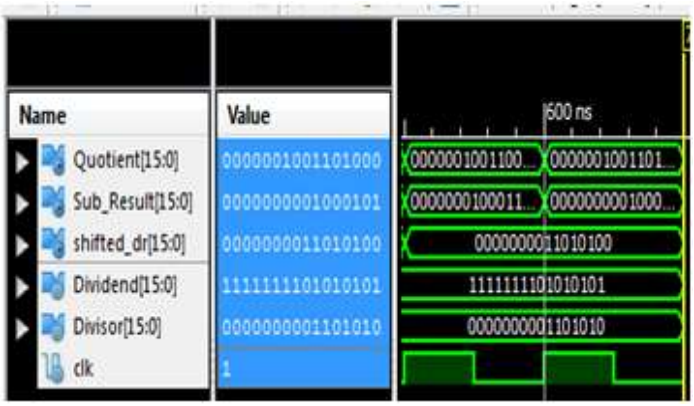

Fig.6: Simulation Results of 16 Bit Division using Proposed Vedic Division Method

The inputs given to the proposed Vedic Division Algorithm and the obtained outputs are as below:

Dividend= 16-Bit input data= 11111111 01010101 (b)
Divisor= 16-Bit Input Data= 00000000 01101010 (b)
Quotient= 16-Bit Output Data= 00000010 01101000 (b)
Remainder= 16-Bit Output Data= 00000000 01000101 (b)

Here, we can verify the division results as Quotient and Remainder. Dividend and Divisor also fed to the simulator. As per the proposed algorithm of Vedic Division method, the divisor is shifted one bit less of the dividend and the same is continued till we obtain the shifted divisor greater than the dividend. Thus, the shifted divisor is also observed in the simulation waveforms.

C. Comparison Results

Comparing the Vedic Multiplier Using "Urdhva-Tiyagbhyam" Method to the conventional Multiplication method and proposed Vedic Division Method to the conventional Division method, we get the comparison results as:

Method	Vedic Multiplication Method	Conventional Method of Binary Multiplication
Number of 4 input LUTs	31 of 1536 2%	678 of 4896 13%
Number of occupied Slices	23 of 768 2%	375 of 2448 15%
Number of bonded IOBs	65 of 124 52%	65 of 158 41%
Delay	6.216ns	15.966ns
Power	24mW	49mW

Table 1: Comparison Results between Conventional and Vedic Multiplication Method

Method	Proposed Vedic Division Method Using "Nikhilam Method"
Number of 4 input LUTs	678 of 4896 15%
Number of occupied Slices	375 of 2448 13%
Number of bonded IOBs	65 of 158 41%
Delay	11.413ns
Power	49mW

Table 2: Results of Proposed Vedic Division Method Derived from Nikhilam Method

CONCLUSION

The proposed Vedic division method derived from "*Nikhilam*" method and the Vedic Multiplication method using "*Urdhva-Tiryagbhyam*" method prove efficient over the conventional methods of binary multiplication and division which are currently being used in the processor hardware. Thus, assembling the Vedic Multiplier[4],[5] and Vedic Divider, an efficient high speed Vedic ALU which performs the arithmetic operations based on the Vedic Mathematics *Sutras* is achieved.

REFERENCES

[1] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaj," Vedic Mathematics", Motilal Banarsidas, Varanasi, India, 1986.

[2] Prof J M Rudagil, Vishwanath Amble, Vishwanath Munavalli, Ravindra Patil, Vinaykumar Sajjan, "Design And Implementation Of Efficient Multiplier Using Vedic Mathematics", Proc. of1nt. Con/, on Advances in Recent Technologies in Communication and Computing 2011

[3] Manish Karandikar, Neha Kawadkar, "Power Efficient High Speed Integer Division Method for Binary System Based on Principles of Vedic Maths", IRAJ International Conference Hyderabad, 2015.

[4] M. Ramalatha, K. Deena Dayalan, P. Dharani, S. Deborah Priya, "High Speed Energy Efficient ALU Design using Vedic Multiplication Techniques", ACTEA 2009, July 15-17, 2009 Zouk Mosbeh, Lebanon

[5] Anju & V.K. Agrawal, "FPGA Implementation of Low Power and High Speed Vedic Multiplier using Vedic Mathematics", IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 2, Issue 5 (May.– Jun.,2013),PP51-57.