Novel Binary divider architecture for high speed VLSI applications

Ratiranjan Senapati, Bandan Kumar Bhoi, Manoranjan Pradhan

VSS University of Technology, Burla, Odisha, IN

Abstract— **Vedic Mathematics is the ancient methodology of Indian mathematics which has a unique computational technique for calculations based on 16 Sutras (Formulae). Novel Binary divider architecture for high speed VLSI application using such ancient methodology is presented in this paper. Propagation delay and dynamic power consumption of a divider circuitry were minimized significantly by removing unnecessary recursion through Vedic division methodology. The functionality of these circuits was checked and performance parameters like propagation delay and dynamic power consumption were calculated by Xilinx ISE using 90nm CMOS technology. The propagation delay of the resulting 8-bit binary dividend by an 4-bit divisor circuitry was only ~19.9ns and consumed ~34mW power for a LUT Utilization of 23/1536. By combining Boolean logic with ancient Vedic mathematics, substantial amount of iteration were eliminated that resulted in ~46% reduction in delay and ~27% reduction in power compared with the mostly used (Repetitive subtraction method) architecture**.

Index Terms **- Paravartya sutra, Repetitive subtraction, Vedic Divider, Vedic mathematics**

I. INTRODUCTION

BINARY division operation is of immense importance in the field of engineering science. Inherently, division operation is a sequential type of operation, thereby it is more costly in terms of computational complexity and latency (propagation delay) compared with other mathematical operations like multiplication and addition[5].

Many recursive techniques[8] have so far been proposed by various researchers to implement the high speed divider[7] , such as digit recurrence implementation methodology ,division by convergence method (Newton-Raphson method), division by series expansion (Goldschmidt algorithm). The cost in terms of area and computational complexity of digit recurrence algorithms is low due to the large number of iterations; therefore, latency (propagation delay) becomes high. While, some of the investigator rely on higher radix implementation of digit recurrence algorithm to reduce the iteration, therefore the latency becomes improved from the earlier reports, but these scheme additionally increases the hardware complexity. Some other attractive ideas are based on functional iterations, like Newton-Raphson and Goldschmidt's algorithm, that utilize multiplication techniques alongwith the series expansion, where the amount of quotient digits obtained per iteration is doubled. The drawback of these methods is operands should be previously normalized, most used primitive are multiplications, and the remainder is not directly obtained[5].

In algorithmic and structural levels, a lot of division techniques had been developed to reduce the latency of the divider circuitry; which reduces the iteration aiming to reduction of latency but the principle behind division was same in all cases. Vedic Mathematics is the ancient system of Indian mathematics which has a unique technique of calculations based on 16 Sutras (Formulae). In this paper we report on a division algorithm and its architecture based on such ancient Indian mathematics. "Paravartya Method" is a Sanskrit term indicating "all Transpose and apply", is adopted from Vedas; formula is encountered to implement the division circuitry. By employing the Vedic methodology division has been implemented by multiplication and addition, therefore reduces the iteration, owing to the substantial reduction in propagation delay[4].

Transistor level implementation of such division circuitry was carried out by the combination of Boolean logic with Vedic mathematics, performance parameters like propagation delay, dynamic power (switching), static power (leakage), consumption[6] calculation of the proposed method was calculated by Xilinx ISE using 90nm CMOS technology. The calculated results showing a 8-bit binary dividend by an 4-bit divisor circuitry having propagation delay of only ~19.9nS with ~34mW power consumption.

II. VEDIC MATHEMATICS

The gifts of the ancient Indian mathematics in the world history of mathematical science are not well recognized. The contributions of mathematician in the field of number theory, 'Sri Bharati Krisna Thirthaji Maharaja', in the form of Vedic Sutras (formulae) are significant for calculations[1]. He had explored the mathematical potentials from Vedic primers and showed that the mathematical operations can be carried out mentally to produce fast answers using the Sutras (Formulae). In this paper we report only Paravartya formula to implement the division algorithm and its architecture[1].

A. Role of Vedic Mathematics

Ancient Indian Vedic Mathematics deals mainly with sixteen Sutras and their applications for carrying out tedious and cumbersome arithmetical operations, and to a very large extent, executing them mentally. Paravartya Sutra is one of these sixteen Sutras used for division, in Vedic Mathematics. Synthetic division, which is a shortcut method for dividing a

978-1-4673-5758-6/13/$31.00 © 2013 IEEE

polynomial by a linear divisor of the form general division can be considered as a special case of the Paravartya Sutra[1].

III. Vedic Division Methodology

A. Algorithm

This paper presents an algorithm to perform binary number division based on the Paravartya Sutra. An enhancement in this algorithm has led to process optimization with respect to number of intermediate operations involved.

The word "Vedic" is derived from the word **"**Veda" which means the store-house of all knowledge. Vedic mathematics is mainly based on 16 Sutras (or aphorisms) dealing with various branches of mathematics like arithmetic, algebra, geometry etc. These Sutras along with their brief meanings are enlisted below alphabetically[1].

1) (Anurupye) Shunyamanyat – If one is in ratio, the other is zero.
2) Chalana-Kalanabyham – Differences and Similarities.
3) Ekadhikina Purvena – By one more than the previous One.
4) Ekanyunena Purvena – By one less than the previous one.
5) Gunakasamuchyah – The factors of the sum is equal to the sum of the factors.
6) Gunitasamuchyah – The product of the sum is equal to the sum of the product.
7) Nikhilam Navatashcaramam Dashatah – All from 9 and last from 10.
8) Paraavartya Yojayet – Transpose and adjust.
9) Puranapuranabyham – By the completion or noncompletion.
10) Sankalana- vyavakalanabhyam – By addition and by subtraction.
11) Shesanyankena Charamena – The remainders by the last digit.
12) Shunyam Saamyasamuccaye – When the sum is the same that sum is zero.
13) Sopaantyadvayamantyam – The ultimate and twice the penultimate.
14) Urdhva-tiryagbhyam – Vertically and crosswise.
15) Vyashtisamanstih – Part and Whole.
16) Yaavadunam – Whatever the extent of its deficiency.

B. Application of the sutra to Decimal number Division

These methods and ideas can be directly applied to trigonometry, plain and spherical geometry, conics, calculus (both differential and integral), and applied mathematics of various kinds. As mentioned earlier, all these Sutras were reconstructed from ancient Vedic texts early in the last century. Many Sub-sutras were also discovered at the same time, which are not discussed here. The beauty of Vedic mathematics lies in the fact that it reduces the otherwise cumbersome-looking calculations in conventional mathematics to a very simple one. This is so because the Vedic formulae are claimed to be based on the natural principles on which the human mind works.

Here an example of the ancient division methodology using Paravartya sutra is demonstrated. We have taken 13905 as dividend and 113 as our divisor. Using regular repetitive subtraction method it takes around 124 cycles of calculations but in our proposed vedic division it takes only around 8 cycles of calculations, which is a significant reduction in computation.

Divisor	Dividend	
1 1 3	1 3 9 0 5	
-1 -3	-1 -3	
	-2 -6	
	-4 -12	
	1 2 4	-10 -7
	1 2 3	06
	Q=124-1=123	R=113-107=06

Fig. 1. Application of Paravartya Sutra to Decimal Number System

This is a very interesting field and presents some effective algorithms which can be applied to various branches of engineering such as computing and digital signal processing. Following the representation similar to the case of polynomial division, Figure 1 illustrates the Paravartya Sutra's application to Arithmetical Operations[1].

C. Extension to Binary number System

Divisor	Dividend	
1 1 0 1	1 1 1 1 0 1 1 0 1	
-1 0 -1	-1 0 -1	
	0 0 0	
	-1 0 -1	
	1 0 1	
	-1 0 -1	
	1 0 1	
	1 0 1 -1 1 -1	11 -1 10
	Q = 101010-101 = 100101	R= 1100-10+10 = 1100

Fig. 2. Extention of Paravartya Sutra to Binary Number System

The Paravartya Sutra as described in is applied to decimal, i.e., Radix 10 system only. We have extended the Sutra to perform Binary Number Division[3] as illustrated in Figure 2.

IV. Hardware Implementaton

The binary form of Paravartya Sutra is implemented using Dedicated Register Bank Structure. Figure 4 shows one such register bank. At each stage subtract contents of RegN bit wise vertically and add contents of RegP bit wise vertically. This explains hardware representation of -1[3].

TABLE 1
REPRESENTATION OF 0, 1,-1

	0:	1:	-1:
RegN	0	0	1
RegP	0	1	0

TABLE 2
TRUTH TABLE FOR HDL IMPLEMENTATION

Divisor bit	Sign bit	Result bit	Interpret
0	X	X	0
1	0	0	0
1	0	1	-1
1	1	1	1

	1	1	1	1	0	1	1	0	1
RegN		1	0	0					
RegP		0	0	0					
			0	0	0				
			0	0	0				
				1	0	1			
				0	0	0			
					0	0	0		
					1	0	1		
						1	0	1	
						0	0	0	
							0	0	0
							1	0	1
Result bit	1	0	1	1	1	1	1	1	0
Carry bit							1	0	1
Sign bit	0	0	0	1	0	1	0	1	

Fig. 3. Expanded explanation of the register contents for the above example with the quotient being divided into result bit, carry bit and sign bit.

The hardware can be reduced further by retaining only two of the shown registers at the Result position, one being the current register and the other being the Result Register which is continuously refreshed at every stage of computation. The Result Register will show final result when computations for all bits have been done.

V. PROCESS ENHANCEMENT

The number of operations performed, in binary version of the Paravartya Sutra are reduced significantly by modifying its process algorithm[3]. The new enhanced algorithm, which remarkably simplifies the process, is represented in Figure 4.

If the divisor is of n1 bits and the dividend is of n2 bits, then here first the divisor is compared with the first n1 bits of the dividend and if (n1)<(n2) then we directly subtract the divisor from the dividend by taking the quotient as 1. Else we continue as told earlier.

Divisor	Dividend		Quotient
1 1 0 1	1 1 1 1 0 1 1 0 1		
-1 0 -1	1 1 0 1 0 0 0 0 0		Q1=100000
	1 0 0 1 1 0 1		
	-1 0 -1		
	1 0 1		
	-1 0 -1		
	1 0 1		
	1 -1 1 -1	11 -1 10	Q2=101
	Q = 1010-101 = 101	R= 1100-10+10= 1100	Q=Q1+Q2= 100101

Fig. 4. Example of Process Enhancement for the previous division

VI. RESULTS

The algorithm discussed above is implemented in verilog HDL and is synthesized and simulated using Xilinx ISE using FPGA family as Spartan 3E. The HDL code is written for a 8-bit by 4-bit divider, and all the corresponding results are shown here. This vedic divider is giving a total delay of 19.9ns and total power consumption of 34mW. Which is a sharp reduction of 46% in total delay and 27% in total power consumption compared to the general repetitive subtraction method.

We simulated and synthesized the proposed algorithm in Xilinx ISE 14.1 package with Spartan 3E family and XC3S100E device. Both the results are shown here below. The compared synthesis report is also given here for the proposed method versus the repetitive subtraction method.
The results are given as,

A. Simulaton result
B. Synthesis result
C. RTL Schematic

A. Simulation Result

The simulation results for three divisions using Xilinx ISE simulator is taken as an example. The results are not limited to only these divisions. The examples taken are 106/13=8, 236/15=15, 161/10=16. Here we have taken the divisor width as 4 bits and the dividend width as 8 bits, which can be extended to any number of bits. But the algorithm and the logic will be the same.

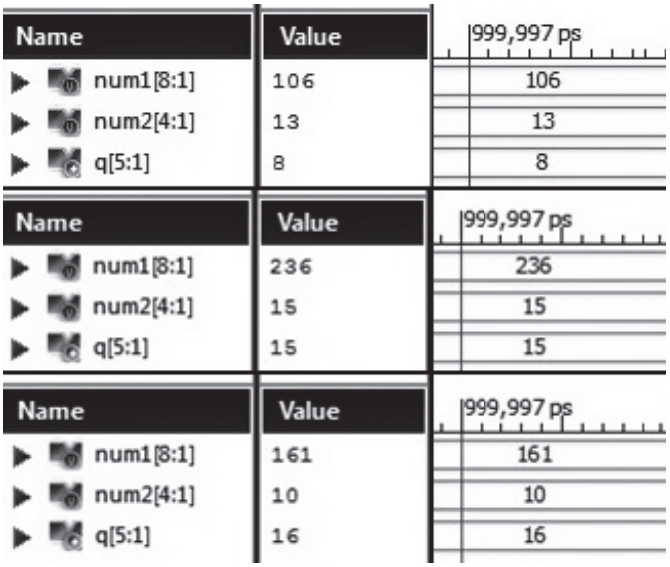

Fig. 5. Simulation results

B. Synthesis Result

We implemented the proposed architecture using Verilog and syntheses the flow, this approach is appropriate because we can change parameters and re-synthesize quickly. In this paper our design primarily consists of five major modules. They are: adder, subtractor, comparator, multiplexer and xor. Each part of the circuit is enabled based on the decode selection. Using this approach, the system can take advantages of using a combination of these functions to perform fast division[2].

Both the delay[9] and power constraints[10] are calculated and the report is given here comparing the result with the conventional division algorithm in Table 3.

TABLE 3
SYNTHESIS RESULT COMPARISON FOR 8BY4 BIT DIVISION

Normal Repetitive subtraction Division Method	Proposed Vedic Divider
Family: Spartan 3E Device: XC3S100E Total Delay: 41.5 ns Power: 46 mW LUT Utilization:39/1536 (2.6%)	Family: Spartan 3E Device: XC3S100E Total Delay: 19.9 ns Power: 34 mW LUT Utilization:23/1536 (1.5%)

For the conventional division method we have taken the repetitive subtraction method. For the same example taken (8bit dividend by 4bit divisor) for both the cases, we calculated the total delay, total dynamic power consumption and the LUTs Utilization for the FPGA device XC3S100E with family Spartan 3E[6]. We can observe that we got a significant reduction in all the parameters discussed above by using our proposed vedic divider.

TABLE 4
LOGIC FUNCTIONS USED

Adders/Subtractors	27
1-bit adder carry out	6
2 bit adder	6
2 bit adder carry out	5
3 bit adder	4
33 bit adder	4
5 bit adder	1
5 bit subtractor	1
Comparators	11
32 bit comparator grater	7
33 bit comparator equal	4
Multiplexers	7
32-bit 4-to-1 multiplexer	7
Xors	8
1 bit xor3	4
1 bit xor5	2
1 bit xor7	2

Also from the synthesis report we have shown the total logic gates used in the proposed divider hardware in Table 4 and the detailed device utilization[6] summery in the Table 5..

TABLE 5
DEVICE UTILIZATION SUMMERY

Selected Device : 3s100evq100-5		
No. of slices	17 out of 960	1% utilization
No. of 4 input LUTs	30 out of 1920	1% utilization
No. of IOs	17	
No. of bonded IOBs	13 out of 66	19% utilization

The FPGA chips used are: Spartan 3E from Xilinx. The critical length in the circuit implemented on the Xilinx FPGA had the minimum period for the proposed subsystem is 19.9 ns.

C. RTL Schematic

The logic circuit diagram for the FPGA implementation[6] of the proposed divider circuit is given above. This circuit diagram is taken from the Xilinx RTL Schematic block.

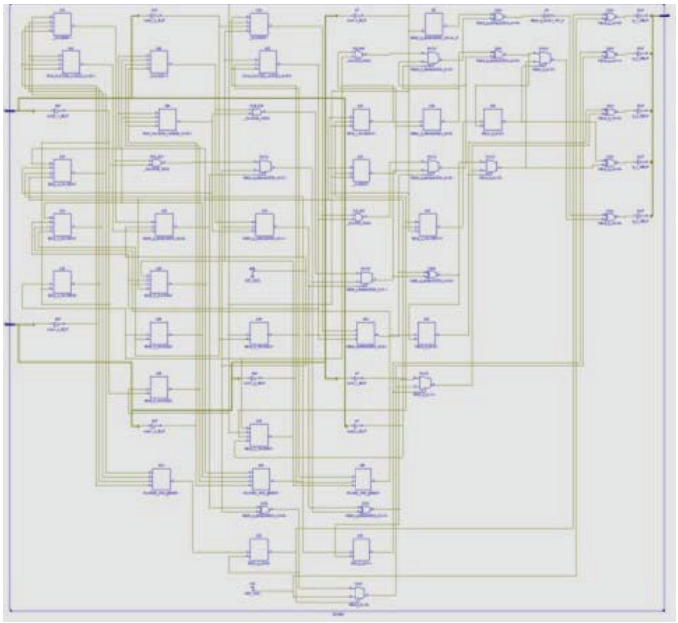

Fig. 6. RTL Schematic of the proposed 8by4 Vedic divider

VII. CONCLUSION

The Paravartya Sutra is successfully extended to implement binary division and also partially implemented in verilog. We have here only proposed for the quotient part of the division and not the remainder part. The proposed vedic divider is

implemented with a 46% reduction in total delay (19.9ns for the proposed divider) and 27% reduction in power consumption(34mw for the proposed divider) as compared to the normal repetitive subtraction division method. This enables applications in various digital systems, like scientific calculators, mobile calculators etc. thereby equipping them with another useful tool.

REFERENCES

[1] Jagadguru Swami Sri Bharati Krisna Tirthatji Maharaja, *Vedic Mathematics,* Motilal Banarsidass Publication, 1992.

[2] Khader Mohammad, Sos Agaian, Fred Hudson, *Implementation of Digital Electronic Arithmetics and its application in image processing* in computers and electrical engg., Elsevier 2010, pp. 424-434

[3] A.Joglekar1, S.Vaidya1, A.Kale, *A Novel Binary Division Algorithm based on Vedic Mathematics and Applications to Polynomial Division* in WORLDCOMP CSC - 2008 Las Vegas, Nevada. pp. 99-104

[4] Himanshu Thapliyal and Hamid R. Arabania , *High Speed Efficient N Bit by N Bit Division Algorithm And Architecture Based On Ancient Indian Vedic Mathematics*, Proceedings of VLSI04, Las Vegas, U.S.A, June 2004, pp. 413-419(CSREA Press).

[5] Shaunak Vaidya, Ashish Joglekar, Vinayak Joshi, *An Efficient Binary Multiplication Algorithm based on Vedic Mathematics*, Proceedings National Conference on Trends in Computing Technologies 2008, Pg 37-43.

[6] Neil H.E Weste, David Harris, Ayan anerjee, *CMOS VLSI Design, A Circuits and Systems Perspective* ,Third Edition, Published by Person Education

[7] P. Saha, A. Banerjee, P. Bhattacharyya, A. Dandapat, *Vedic Divider: Novel Architecture (ASIC) for High Speed VLSI Applications, IEEE* International Symposium on Electronic System Design, 2011, pp 67-71

[8] H. R. Srinivas and K. K. Parhi, *A Fast Radix 4 Division Algorithm*, in *Proc. IEEE Int. Symp. on Circuit & Systems*, London, May 30-June. 02, 1994,vol. 4, pp. 311-314

[9] Sung-Mo Kang, Yusuf Leblebici, *CMOS Digital Integrated Circuits: Analysis and Design*, Third Edition, Published by TMH

[10] Debaprasad Das, *VLSI Design*, First Edition, Published by OXFORD University Press