Research Journal of Applied Sciences, Engineering and Technology 11(12): 1298-1304, 2015
DOI: 10.19026/rjaset.11.2238
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: May 20, 2015 Accepted: August 5, 2015 Published: December 25, 2015

Research Article

A Comparitive Study of Vedic BCD Multiplier using Reversible Logic Gates

[1]R. Anitha, [1]R. Thenmozhi, [1]M. Madhunila and [2]Sarat Kumar Sahoo
[1]School of Electronics Engineering,
[2]School of Electrical Engineering, VIT University, Vellore, India

Abstract: Hardware implementation for decimal operations are more important rather which is more useful in the field of technical (DSP, Microprocessor, Digital Image Processing, etc.) and non-Technical (banking calculation, currency conversion, even in office ledgers, etc.). The main purpose of this research is to improve the speed of the digital processors such as adders, multipliers, which are the prime factors in digital circuits. Here is the architecture of a BCD multiplier, which can increase the efficiency and performance of the digital systems. The binary and decimal numbers are converted to BCD ad then multiplied using Vedic algorithm. High speed multiplier architecture that we designed here is using Vedic mathematics. Urdhva-Tiryagbyham sutra is used to design Vedic multiplier. Power and area are the other paramount of the VLSI design, to diminish that here reversible logic gates are used. The circuit in this study is constructed using Vedic Multiplier with reversible logic gates. Verilog HDL code has been written to perform the simulation. The area and power consumption of Vedic multiplier and Vedic multiplier using reversible logic are calculated using cadence software and both of the results are compared. Vedic multiplier is used in DSP applications such as IIR and FIR filters, FFT and convolution.

Keywords: BCD, cadence, reversible logic gates, urdhva-triyagbhayam, vedic mathematics, verilog HDL, xilinx

INTRODUCTION

Multiplication is the most common and important arithmetic operation which has wide applications in different areas of science and technology. The rudimentary block of Digital Signal Processing is the Multiplier (Mehta *et al*., 2013; Sriraman and Prabakar, 2012). Multiplication is used in many applications such as instrumentation and measurement, animations, graphics, communication and audio and video processing, etc.

Three different styles (Anitha *et al*., 2012; Anitha and Bagyaveereswaran, 2013) are there to do the multiplication in decimal (Premananda *et al*., 2013). First way is directly multiply the two decimal numbers. Second way is to change the decimal number into the Binary form and then perform the multiplication; solution will be again converting to decimal. Third way is each and every digit will be converted into binary and multiplied; the final product will arrive from partial product generated by binary multiplication. Third way of decimal multiplication is adopted in this study. There are two challenges in designing BCD multiplier (Vaibhav *et al*., 2014). First one is the multiplication process and the second one is the binary to BCD converter. So we are designing a high speed modified Vedic BCD multiplier using the reversible logic (Sriraman and Prabakar, 2012) which has been implemented in this thesis, which is faster than the conventional BCD Multiplier and produces less power consumption (Subudhi *et al*., 2014). We can implement the FIR Filter as which is the important feature of DSP (Krishnaveni and Umarani, 2012).

MATERIALS AND METHODS

Vedic mathematics: Vedic mathematics (Subudhi *et al*. 2014; Pradhan and Panda 2014) are an ancient mathematics which is more efficient than other mathematic techniques. Vedic mathematics is used in many applications such as arithmetic operations, theory of numbers, compound multiplications, squaring, cubing, square root and cube root etc. Totally there are 16 sutras and 14 sub-sutras in Vedic mathematics. Among those sutras, only 3 sutras and 2 sub-sutras are used for multiplication. Urdhva-Triyagbhayam is an universally adopted method for all multiplication. So this method is chosen to design Vedic multiplier. This method is also called as vertical-cross method. The following steps are used in Urdhva-Triyagbhayam multiplication method (Mehta *et al*., 2013):

Step 1: The Least significant bit will be first multiplied and the corresponding product will be assigned as direct product.

Step 2: Next highest bit will cross multiplied with the LSB and the product will be added together. If

Corresponding Author: R. Anitha, School of Electronics Engineering, VIT University, Vellore, India

carry occurs that will be carried out to the next step.

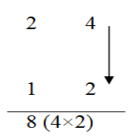
Step 3: Again the next bits will be multiplied in cross and vertically and then availed products will added along with the carry occurred in the last step. Repeat for other higher bits.

Step 4: Partial products are added together with the carry if generated.

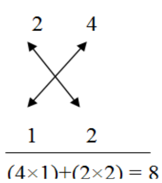
The following is an example of Urdhva-Triyagbhayam method:

$24 \times 12 = 288$

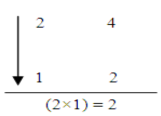
Step 1:



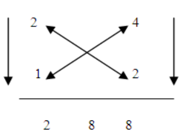
Step 2:



Step 3:



Result:



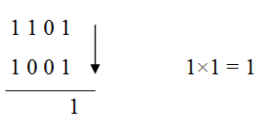
Implementation of 4×4 vedic multiplier: The following method is 4-bit binary multiplication using Urdhva-Triyagbhyam technique. First the decimal value is converted into binary and then the binary multiplication is done using Urdhva-Triyagbhyam method.

Example: $13 \times 9 = 117$

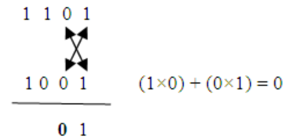
13 = 1101 binary multiplication

9 = 1001 1101×1001 = Step 1

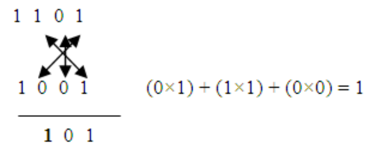
Step 1:



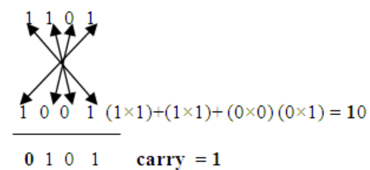
Step 2:



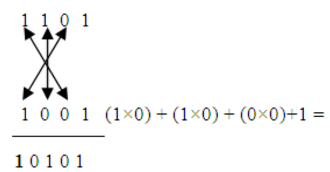
Step 3:



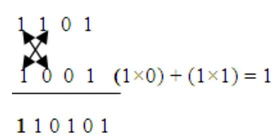
Step 4:



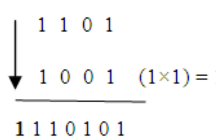
Step 5:



Step 6:



Step 7:



Decimal equivalent of the answer is 1110101 = 117.

The result of 4×4 Vedic multiplier consists of 7-bit binary partial product.

Figure 1 is the block diagram of 4×4 Vedic multiplier. It consists of half adders and full adders (Vaibhav *et al*., 2014).

BINARY TO BCD CONVERTER

Shift-add 3-algorithm is used to perform binary to BCD conversion. This concept can be used on a binary number with any number of bits. For a four bit number, four shifts must be performed. If the number is greater

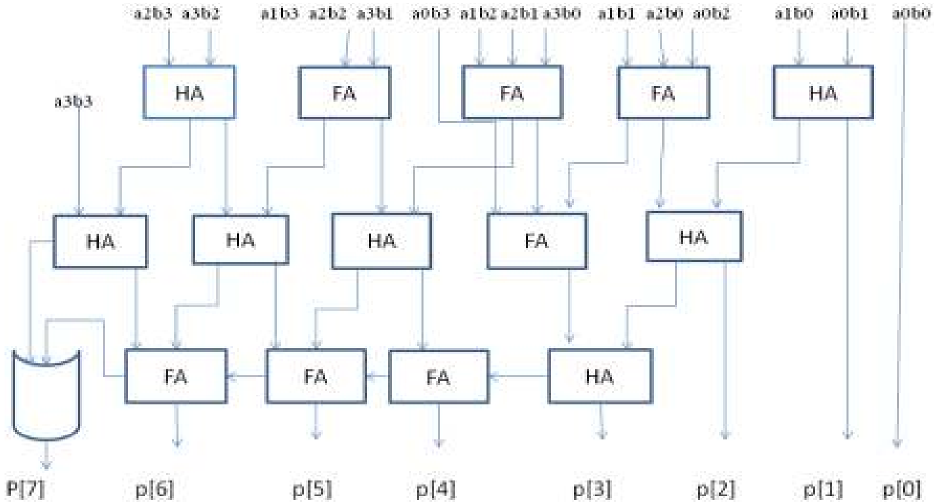


Fig. 1: Block diagram of 4×4 vedic multiplier

Table 1: Binary to BCD conversion

100's	10's	1's	Binary	Operation
			1100000	
		1	100000	<< shift1
		11	00000	<< shift 2
		110	0000	<< shift 3
		110+011 = 1001	0000	Add 3
	1	0010	000	<< shift 4
	10	0100	00	<< shift 5
	100	1000	0	<< shift 6
	100	1000+011 = 1011	0	Add 3
	1001	0110	-	<< shift 7

than four, each shift is performed and then it must be added to three. The final BCD number will be in the uppermost bits of the new number, once all four bits have been shifted (Table 1).

Implementation of 4 bit Vedic multiplier using reversible logic:

Reversible logic: Most of the Low power designs are now-a-days designed using Reversible Logic gates Assarian *et al*. (2012) because which has no internal power.

It's a one to one mapping between the vectors of inputs vs outputs as shown the example Fig. 2.

By the Pigeonhole principle the reversible logic gates will have same number of inputs and outputs. using one not gate and identity gate one input bit can be there which can have two possible reversible gates.

The reversible computation can be carried out in a reversible manner, where there won't be a power dissipation. There are 2 states to perform this:

- Logically reversible-means that unique retrievable of inputs and outputs.
- Physically reversible- the device which can run behind one.

Reversible logic gates:

Feynman gate: 2*2 Feynman gate (Husain *et al*., 2013; Sivakumar and Devi, 2013) has input and output vectors. And it has the quantum cost of 1. The outputs are defined as:

$P = A$

$Q = A \oplus B$

The block diagram and the equivalent quantum diagram is given in Fig. 3a and 3b respectively.

Tofolli gate: Its otherwise called as "controlled-controlled-not", which as 3 inputs and 3 outputs. The first two bits are set and the third bit will do the inversion, otherwise all the input bits will be same (Fig. 4).

It can be described by mapping the inputs a, b and c to the outputs a, b and c XOR (a and b) (Kumar Chunduri *et al*., 2013).

It's a main application of quantum error correction:

Inputs are A, B and C

Outputs

$P = A$

$Q = B$

$R = C$ ^ $(A$ & $B)$

Peres gate: 3*3 Peres Gate is another important gate which has a low quantum cost as compared to other gates. A single Peres gate can work as half adder when the third input C = 0. Let be the input and output vector of a 3*3 Peres gate, where Input = (A, B, C) and Output = ($P = A$, $Q = A \oplus B$, $R = AB \oplus C$) (Husain *et al*., 2013) (Fig. 5).

DKG gate: Let be the input and output vector of a 4*4 Peres gate, where Input = (A, B, C) and Output = ($P = B$, $Q = A'C \oplus AD'$, $R = (A \oplus B)(C \oplus D) \oplus CD$, $S = B \oplus C \oplus D$). A single DKG gate can work as full adder when the first input A= 0 (Kumar Chunduri *et al*., 2013; Devendra and Vidhi, 2012) (Fig. 6).

Using Feynman gate we can calculate the sum of the half adder and using the Tofolli gate we can calculate the carry of the half adder by making it's third

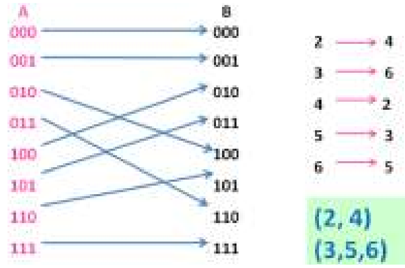


Fig. 2: Example for reversible logic implementation

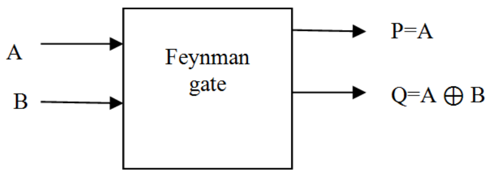


Fig. 3a: Feynman gate

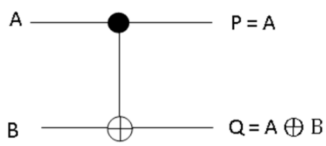


Fig. 3b: Quantum diagram

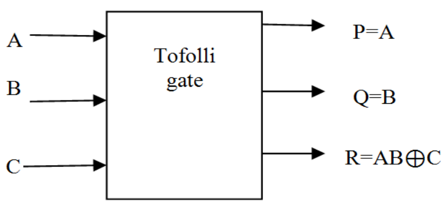


Fig. 4: Tofolli gate

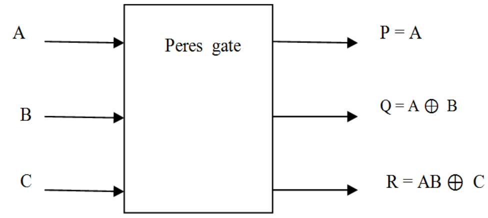


Fig. 5: Peres gate

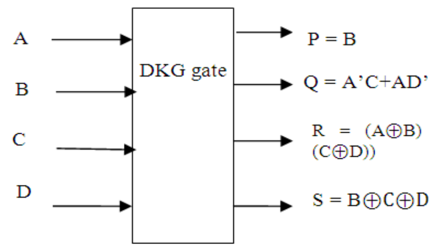


Fig. 6: DKG gate

output as 0. But by using the Peres gate we can calculate the sum and carry of the half adder together by making it's third input as 0.DKG gate can act as full adder when it's first input is 0 (Fig. 7).

Application of Vedic multiplier: Vedic multiplier is used in many DSP applications (Pravin and Dhengre, 2014; Gandewar and Sarde, 2014) such as FFT, Convolution, Fourier transform and Digital filters etc. The two types of digital filters are IIR filter and FIR filter.
The equation of FIR filter is:

$$Y(n) = \sum_{K=0}^{M-1} h(k)\, x(n-k)$$

M = Length of the sequence
y(n) = Filter output

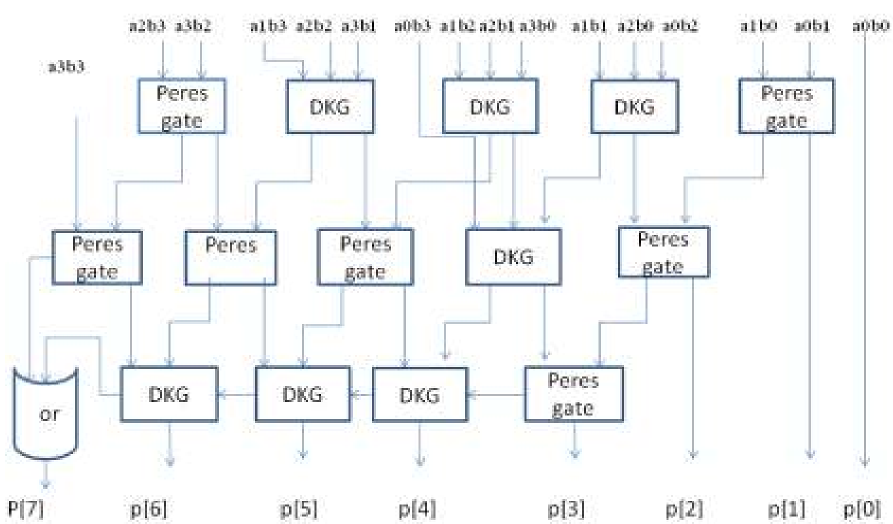


Fig. 7: 4×4 Vedic multiplier using reversible logic gates

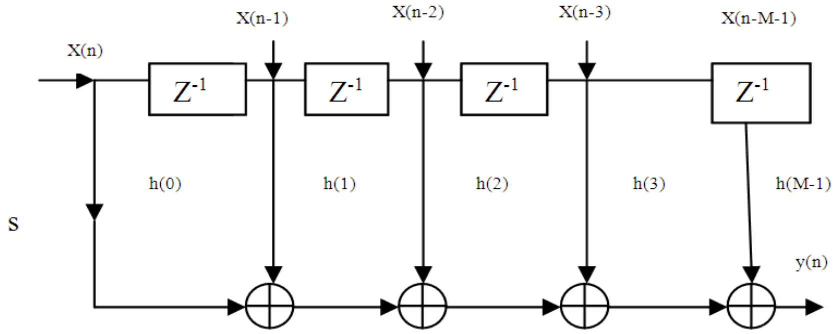


Fig. 8: Block diagram of FIR filter

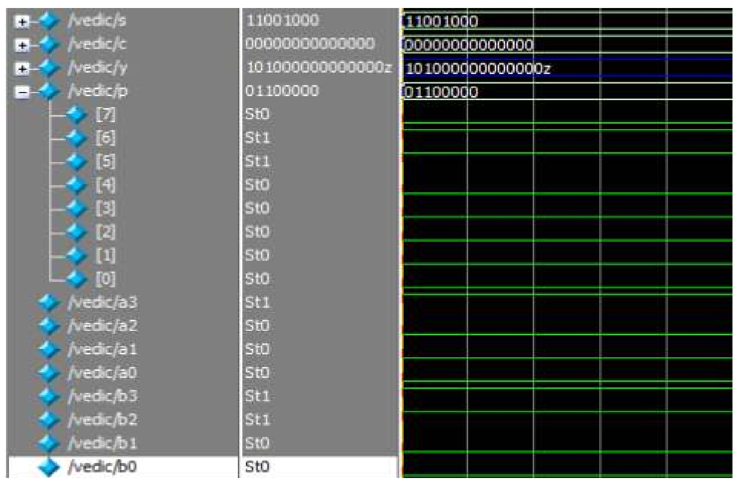


Fig. 9: Simulation result for 4-bit vedic multiplier

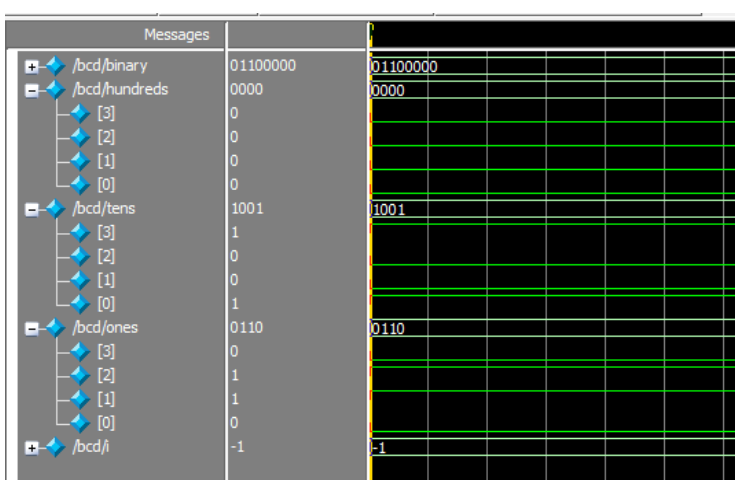


Fig. 10: Simulation result for 7-bit Binary to BCD converter

x(n) = Input
h(k) = Filter co-efficients
Z^{-1} = Delay block

The multiplication of filter co-efficients and delayed signal is performed by using vedic multiplier. By using vedic multiplier the speed of the process increased. The future work may be we can implement the FIR filter using this technique (Fig. 8) (Binu Siva Singh *et al*., 2014).

RESULTS

The Verilog code for vedic 4-bit multiplier in Fig. 1 is simulated in Modelsim 6.5 as well as in xilinx and the results are given below in Fig. 9. Simulation

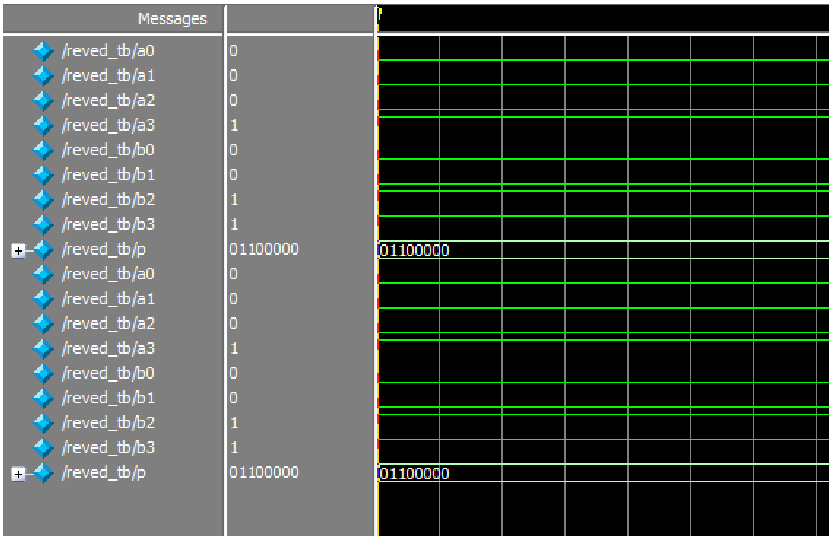


Fig. 11: Simulation result for 4-bit vedic multiplier using reversible logic

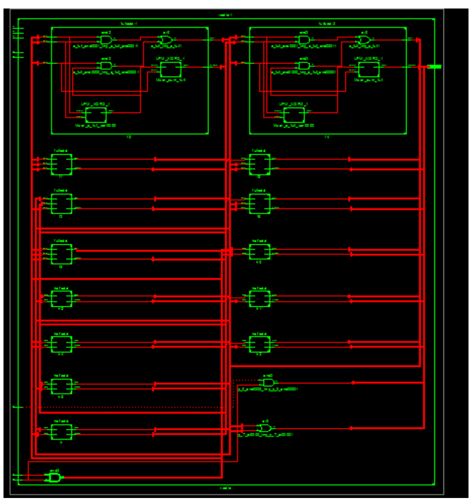


Fig. 12: 4-bit vedic multiplier

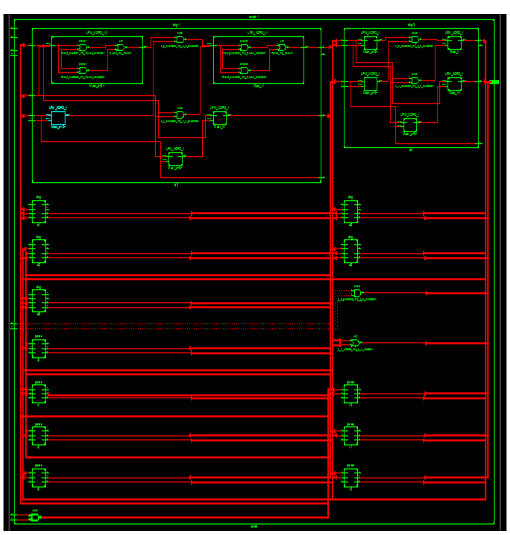


Fig. 13: 4-bit reversible vedic multiplier

Result for 4-bit Vedic multiplier, Fig. 10. Simulation Result for 7-bit Binary to BCD converter and Fig. 11. Simulation Result for 4-bit Vedic multiplier using Reversible logics which is show in the Fig. 3 to 7 is implemented and the results are shown below:

Inputs: $a_3a_2a_1a_0 = 1000$
$b_3b_2b_1b_0 = 1100$ Output: P = 01100000

and the RTL schematic also shown in the Fig. 12. 4-bit vedic multiplier, Fig. 13. 4-bit reversible vedic multiplier.

The code is synthesized using Cadence. Figure 2 showing the synthesized report of Vedic multiplier Vs Reversible vedic multiplier.

CONCLUSION

In this study, we have implemented 4-bit vedic multiplier using reversible logic. The Binary to BCD converter is design and synthesized and the multiplication unit has been implemented. The vedic multiplier is implemented in conventional method and the proposed design i.e., vedic multiplier block is modified using reversible logic gates. So that the comutation time is reduced so far. Quantum cost of the system is reduced further. The number combinational cells have been diminished when compare to the normal conventional binary multiplication.

Table 2: Comparison of 4-bit vedic multiplier with that of 4-bit vedic multiplier using reversible logic gates

Multiplier	Vedic multiplier (4 bit)	Vedic multiplier using reversible logic gates (4-bit)
Power consumption	15217.099 nw	13873.078 nw
Area	109	105
Number of cells	37	31

The comparison of vedic multiplier and vedic multiplier using reversible logic is given in Fig. 2 From the results, We can show that vedic multiplier using reversible logic is more efficient and faster than the normal vedic multiplier. The area and power consumption of vedic multiplier is considerably reduced when reversible logic is applied (Table 2).

Using the reversible vedic multiplier logic we can implement the FIR/IIR filters, Convolution and deconvolution, etc. which are all the essential block in the DSP Processor.

REFERENCES

Anitha, R., N. Alekhya, J.W. Lincy and V. Bagyaveereswaran, 2012. Comparative study of high performance Braun's multiplier using FPGAs. IOSR J. Electron. Commun. Eng., 1(4): 33-37.

Anitha, R. and V. Bagyaveereswaran, 2013. Low power baugh wooley multipliers with bypassing logic. Proceeding of the IEEE International Conference on Research and Development Prospects on Engineering and Technology (ICRDPET, 2013), Vol. 3.

Assarian, M., M. Haghparast and K. Navi, 2012. Delay reduction in optimized reversible multiplier circuit. Res. J. Appl. Sci. Eng. Technol., 4(1): 27-32.

Binu Siva Singh, S.K., V.G. Krishnan and S. Ranjith, 2014. Low power high speed IIR filter design for DSP applications. Int. J. Appl. Eng. Res., 9(24): 30737-30747.

Devendra, G. and S. Vidhi, 2012. VHDL implementation of reversible logic gates. Int. J. Adv. Technol. Eng. Res. (IJATER), 2(3).

Gandewar, S.A. and M. Sarde, 2014. Vedic multiplier for high speed computation using VHDL. J. Res. Elect. Electron. Eng. (ISTP-JREEE), 3(2).

Husain, H., M. Mamun, M. Marufuzzaman and H.A. Shaer, 2013. Design of a high speed low power 2's complement adder circuit. Res. J. Appl. Sci. Eng. Technol., 5(8): 2556-2564.

Krishnaveni, D. and T.G. Umarani, 2012. VLSI implementation of vedic multiplier with reduced delay. Int. J. Adv. Technol. Eng. Res. (IJATER), 2(4).

Kumar Chunduri, V.S., G. Sree Lakshmi and M.J.C. Prasad, 2013. Design and Implementation of multiplier using kcm and vedic mathematics by using reversible adder. Int. J. Modern Eng. Res. (IJMER), 3(5).

Mehta, A.K., M. Gupta, V. Jain and S. Kumar, 2013. High performance vedic BCD multiplier and modified binary to BCD converter. Proceeding of the Annual IEEE India Conference (INDICON, 2013), pp: 1-6.

Pradhan, M. and R. Panda, 2014. High speed multiplier using nikhilam sutra algorithm of vedic mathematics. Int. J. Electron., 101(3): 300-307.

Pravin, Y.K. and K.S. Dhengre, 2014. High speed and low power fir filter implementation using optimized adder and multiplier based on Xilinx FPGA. IORD J. Sci. Technol., 1(3): 46-52.

Premananda, B.S., S.P. Samarth, B. Shashank and S.B. Shashank, 2013. Design and implementation of 8-bit vedic multiplier. Int. J. Adv. Res. Electr. Electron. Instrum. Eng., 2(12).

Sivakumar, G.P. and S.R. Devi, 2013. A comparative study: Multiplier design using reversible logic gates. Int. J. Eng. Adv. Technol. (IJEAT), 2(3).

Sriraman, L. and T.N. Prabakar, 2012. Design and implementation of two variable multiplier using KCM and vedic mathematics. Proceeding of the 1st International Conference on Recent Advances in Information Technology (RAIT, 2012).

Subudhi, A.D., K.C. Gauda, A.K. Pala and J. Das, 2014. Design and implementation of high speed 4x4 vedic multiplier. Int. J. Adv. Res. Comput. Sci. Software Eng., 4(11).

Vaibhav, J., Z.R. Navaid and K.S. Dinesh, 2014. VHDL code of vedic multiplierwith minimum delay architecture. Int. J. Eng. Tech. Res., Special Issue, ISSN: 2321-0869.