ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

Design of a High-Speed, Low-Power, Area-Efficient 8-bit Vedic Multiplier Using Urdhva-Tiryagbhyam Theorem & Modified GDI Cells

Ravi Tiwari[1], Bobby Nelson[2]

Assistant Professor, Dept. of ECE, SSTC, Shri Shankaracharya Group of Institutions, Bhilai, CG, India[1]

PG Student [VLSI Design],Dept. of ECE, SSTC, Shri Shankaracharya Group of Institutions, Bhilai, CG, India[2]

ABSTRACT:The paper presents the implementation of a low-power and area-efficient 8-bit multiplier using the concepts of ancient Vedic Mathematics, more specifically the Urdhva-Tiryagbhyam (UT) theorem. We aim to design the aforementioned multiplier using Modified-Gate-Diffusion-Input cells (Mod-GDI), which facilitate the reduction of transistor count while maintaining a full voltage swing, thereby, consuming even lower power than the CMOS implementation of the Vedic Multiplier. The multipliers are one of the most complex arithmetic functions implemented in processors and hence the simplification of the same becomes essential. The UT theorem is an effective tool that reduces the design complexity of the multiplier. The Mod-GDI cells help to reduce the area constraints of the design with their minimal implementation logic. The designed multiplier has then been compared with a Hierarchical Array Multiplier as well as a Vedic Multiplier based on the UT theorem, both in the CMOS technology.

KEYWORDS:8-bit Multiplier, Vedic Mathematics, Urdhva-Tiryagbhyam Theorem, GDI cells, CMOS

I.INTRODUCTION

The multiplier finds its usage in practically all kinds of processing systems ranging from application specific processors dealing with an infinitely large bit width or a small scale general processor dealing with a comparably smaller subset of data. It also happens to be one of the most time-consuming digital processes and offers a good scope for improvement in terms of area, delay as well as power efficiency.The conventional hierarchical array multiplication is a simple looped method of binary multiplication in which a multiplicand is repetitively multiplied with individual bits of second multiplicand and the partial product terms are then finally added to yield the product. The process, however simple in approach and implementable hierarchically, requires extensive hardware and is also marred by a heavy adder delay. The adder delays are often tackled with complex adder systems that in turn increase the hardware requirements of the system along with the inherent delay.

We aim to tackle these hurdles by using the ancient concepts of Vedic Mathematics. Out of the hundreds of theorems available in the arsenal of Vedic Mathematics, we aim to reduce the mathematical complexity of the binary multiplication process by using the Urdhva-Tiryagbhyam (UT) theorem. It is basically a decimal multiplication theorem which effectively simplifies the multiplication process.The UT theorem can be easily adapted to binary process as well and can be used to reduce the circuit complexity of the multiplier. The UT theorem has been proven to reduce the delay as well as power consumption of the multipliers even in the CMOS process but we further aim to optimize the area constraints as well as the power consumption of the multiplier quite effectively by using a modified circuit for the Gate-Diffusion-Input (GDI) cells. The GDI cells offer a minimalistic technique to reduce the area or the transistor count of a design by the virtue of their capability to implement complex functions in comparably fewer transistors. The use of fewer transistors also leads to fewer switching and thereby conserves a lot of switching energy as well as delay. They also have the added advantage of being easily fabricated with slight modifications to the twin-tub CMOS process.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

II.GENERIC MULTIPLIER MODEL

The design of generic multiplier data-pathscan be reduced to two stages on the basis of their functionality. The first stage comprises of an array of AND gates that generates the product terms using the input multiplicands. The next stage comprises the shifters and adders that finally yield the product as depicted in the Fig. 1. Our methodology aims to improve the AND array using Mod-GDI cells and the ADDERS stage simplification is carried out using the UT theorem.

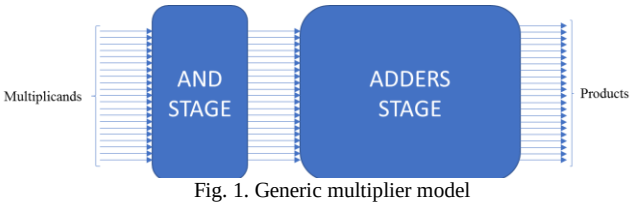


Fig. 1. Generic multiplier model

III. DESIGN METHODOLOGY

A. Urdhva-Tiryagbhyam Theorem

The UT theorem has been traditionally used for the multiplication of two numbers in the decimal number system. In this work, we apply the same ideas to the binary number system to make the proposed algorithm compatible with the digital hardware. The name literally translates to "Vertically and Crosswise" from Sanskrit.It is a general multiplication formula applicable to all cases of multiplication. It aims to simplify the traditional approach of iterative multiplication by replacing it with an optimized combinatorial logic. Even though the UT theorem was developed for the decimal number system, the theorem appropriately fits the binary number system as well. The Fig. 2. shows the generation of the term '*F*' according to the UT theorem and the rest of the terms are shown inthe collective equation (1) below:

$$D = A_0B_0$$
$$E = A_0B_1+A_1B_0$$
$$F = A_0B_2+A_1B_1+A_2B_0$$
$$G = A_0B_3+A_1B_2+A_2B_1+A_3B_0$$
$$H = A_0B_4+A_1B_3+A_2B_2+A_3B_1+A_4B_0$$
$$I = A_0B_5+A_1B_4+A_2B_3+A_3B_2+A_4B_1+A_5B_0$$
$$J = A_0B_6+A_1B_5+A_2B_4+A_3B_3+A_4B_2+A_5B_1+A_6B_0$$
$$K = A_0B_7+A_1B_6+A_2B_5+A_3B_4+A_4B_3+A_5B_2+A_6B_1+A_7B_0$$
$$L = A_1B_7+A_2B_6+A_3B_5+A_4B_4+A_5B_3+A_6B_2+A_7B_1$$
$$M = A_2B_7+A_3B_6+A_4B_5+A_5B_4+A_6B_3+A_7B_2$$
$$N =A_3B_7+A_4B_6+A_5B_5+A_6B_4+A_7B_3$$
$$O = A_4B_7+A_5B_6+A_6B_5+A_7B_4$$
$$P = A_5B_7+A_6B_6+A_7B_5$$
$$Q = A_6B_7+A_5B_6$$
$$R = A_7B_7 \quad (1)$$

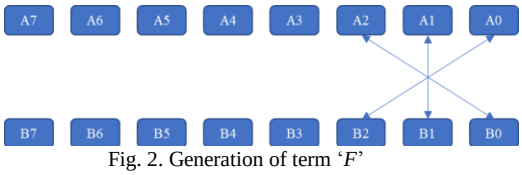


Fig. 2. Generation of term '*F*'

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

The implementation of these partial product terms is rather straightforward and is done by using an array of AND gates and adders.These terms shall have variable bit-widths with '*K*' being the widest, with a bit-width of 4 bits.The Table 1 serves an important tool for easily and minimally realizing the combinatorial logic for the summing of the partial product terms so as to generate the final products. The *C's* represent the carry terms and actually depict a symbolic collective carry from all the previous stages of addition. Finally, the product is obtained as $X_{16\text{-}bits}$.

TABLE1
GRIDREPRESENTATIONOFPARTIALPRODUCT ADDITION

															D
												C	E_1	E_0	
										C	C	F_1	F_0		
									C	G_2	G_1	G_0			
								C	H_2	H_1	H_0				
							C	I_2	I_1	I_0					
					C	C	J_2	J_1	J_0						
					K_3	K_2	K_1	K_0							
				C	L_2	L_1	L_0								
			C	M_2	M_1	M_0									
		C	N_2	N_1	N_0										
		O_2	O_1	O_0											
	C	P_1	P_0												
	Q_1	Q_0													
C	R														
X_{15}	X_{14}	X_{13}	X_{12}	X_{11}	X_{10}	X_9	X_8	X_7	X_6	X_5	X_4	X_3	X_2	X_1	X_0

B. Modified GDI Cells

TheGDI design technique was introduced as a promising alternative to theCMOS logic design style of complimentarily functioning gates. Originally proposed for fabrication in Silicon on Insulator (SOI) and twin-well CMOS processes, GDI methodology allowsthe implementation of a wide range of complex logic functions using merely two transistors. GDI implementation of a design thus helps in reducing its overall transistor count and thereby improves area constraints of the design.

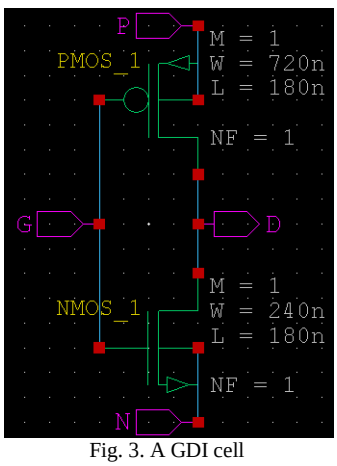


Fig. 3. A GDI cell

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

The Fig. 3. shows the basic construction of a GDI cell. We have implemented our design in 180nm technology with a W_p/W_n ratio of 3. In conventional GDI cells, the gates of the PMOS & NMOS devices are shorted to yield an input *G*, the sources terminals are individually shorted with the substrate to yield the *P*&*N* terminals respectively for PMOS & NMOS. This simple configuration of mere two MOS devices is capable of producing many complex logic functions as shown in the TABLE 2.

TABLE 2
FUNCTION IMPLEMENTATION USING GDI CELLS

N	P	G	D	Functions
0	1	A	A`	Inverter
1	B	A	A+B	OR
B	0	A	AB	AND
C	B	A	A`B+AC	MUX
B`	B	A	A`B+AB`	XOR
B	B`	A	AB+A`B`	XNOR

As opposed to the conventional CMOS implementation GDI cells are extremely versatile and can practically reduce the area constraints by a wide margin as is evident in the TABLE 3 below. Here, inverters for inputs aren't being counted.

TABLE 3
FUNCTION IMPLEMENTATION USING GDI CELLS

FUNCTION	CMOS	GDI
Inverter	2	2
OR	6	2
AND	6	2
MUX	12	2
XOR	16	2
XNOR	16	2

We thus get a general idea about how advantageous it is to use GDI cells in place of the conventional CMOS logic implementations. However, there's a certain caveat of partial swing in GDI cells which renders them practically unusable for any cascade connection with other gates. It also leads to wild harmonics in the output signals which dissipate more power than saved. Hence, the actual utility requires a few modifications to the basic GDI cell design as evident in the Fig. 4. while the functionality remains the same.

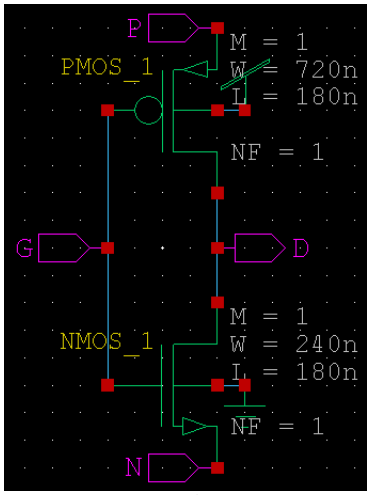


Fig. 4. A modified GDI cell

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

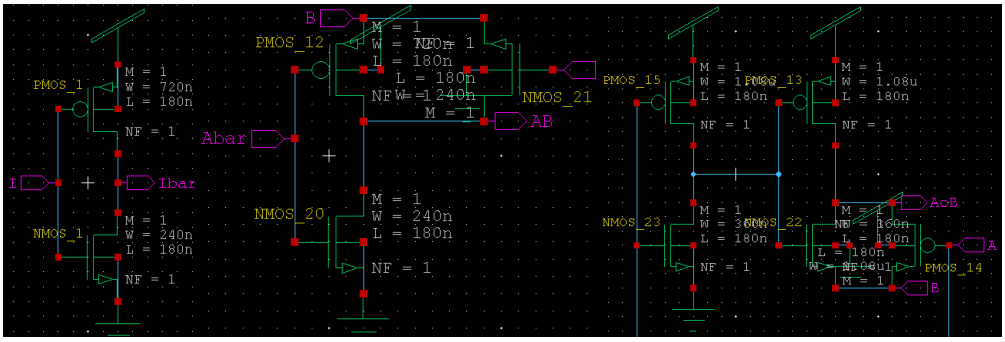
(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

IV. IMPLEMENTATION

Our next task was to implement the sub-circuits required for the multiplier architecture using the Mod-GDI cells. Although the Mod-GDI cells do have a better response and lower harmonics than traditional GDI cells but they still have the partial swing problem which needs to be addressed for the individual sub-circuits. The following figures present the implementation of these sub-circuits using the Mod-GDI cells. The Fig. 5. represents the various basic logic gates used in the multiplier architecture. The gates have been optimized in Mod-GDI logic for minimum transistor count while maintaining a full-swing output.



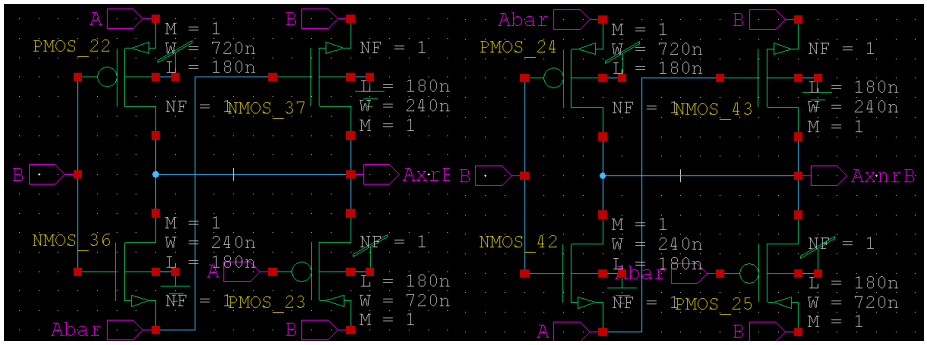
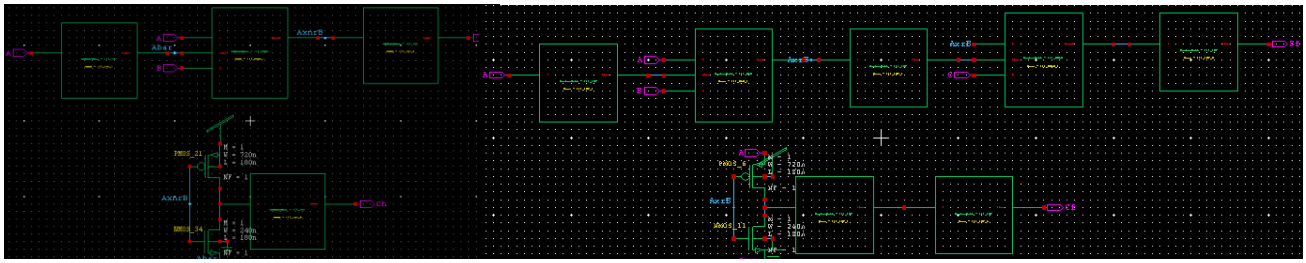


Fig. 5. Implementation of the basic logic gates in Mod-GDI logic
(a) Inverter, (b) AND, (c) OR, (d) XOR, (e) XNOR

The Fig. 6. depicts the implementation of the adder blocks using the basic gates constructed in Fig. 4., in a hierarchical manner. The consideration again is to use minimal transistor count for full-swing at output. Hence, appropriate changes have been made for the full-adder carry out. We have a comparison of the transistor count in CMOS as well as in Mod-GDI technology in Table 4. Again, the extra inverters aren't being considered other than in case of the OR gate.



(a)(b)

Fig. 6. Implementation of adder blocks

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

(a) Half-adder (b) Full-adder

TABLE 4
MINIMAL TRANSISTOR COUNT IN CMOS VS. MOD-GDI

Gate/Block	CMOS	Mod-GDI
Inverter	2	2
OR	6	5
AND	4	3
XOR	12	4
Half-adder	20	12
Full-adder	28	20

The Fig. 7. shows the final 8-bit multiplier architecture completed using all the components designed thus far. The multiplier works in two stages as is evident from the design. The first stage generates all the partial product terms from the collective equations in (1) and the second stage performs the additions from the Table 1 to yield the final product '$X_{16\text{-}bits}$'. The implementation was carried in 180nm technology with a W_p/W_nratio of 3.

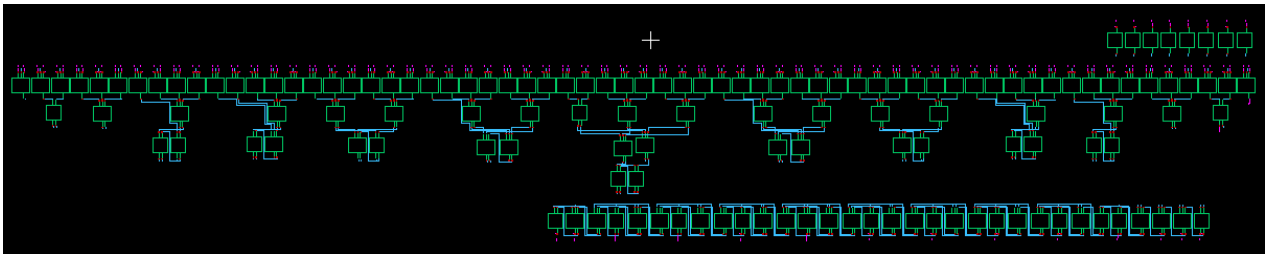


Fig. 7. Proposed 8-bit Multiplier Architecture

V. SIMULATION RESULTS AND DISCUSSION

The architecture was implemented in T-Spice v16.0 at 180nm technology keeping the W_p/W_nratio at 3 for approximately equal rise and fall times. The Fig. 8. shows the simulation output of the proposed multiplier for an input combination of all 1's at 1.8V input supply. The simulation graph traces are highlighted at the high state in blue for ease in reading the output of the multiplier for the said inputs. The output in this case ought to be '*1000000001111111*' as easily verifiable through the highlighted 1.8V level traces in the simulation graph in Fig. 8.

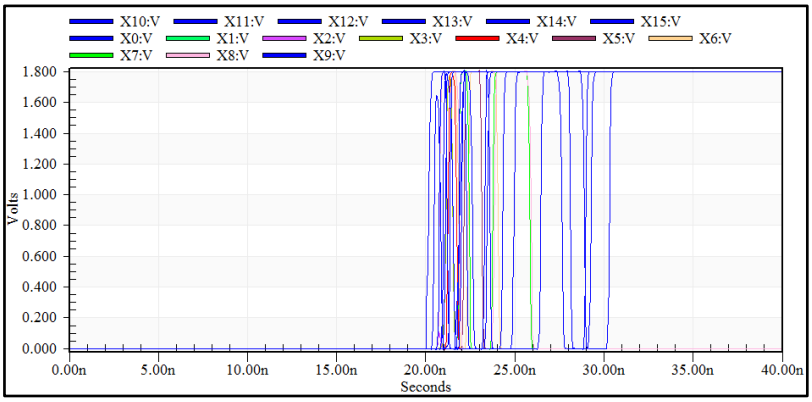


Fig. 8. Simulation results of proposed multiplier for an all 1's input combination

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

The proposed multiplier was tested with a series of pseudorandom inputs and was found to yield conducive results for all the test inputs. The Table 5 presents a detailed comparative analysis of the proposed Mod-GDI UTMultiplier (MDGUTM) with a CMOS UT Multiplier (CUTM) as well as with a CMOS Hierarchical Array Multiplier (CHAM).

TABLE 5
COMPARATIVE ANALYSIS OF THE MULTIPLIERS

Multipliers	MOS-FETs	Power mw	Delay Ns	PDP pJ	EDP 10^{-21}Js
CMOS Hierarchical Array Multiplier (CHAM)	3524	1.2	14.84	17.81	264.27
CMOS UT Multiplier (CUTM)	2880	1.07	14.59	15.61	227.77
Mod-GDI UT Multiplier (MGUTM)*	**1429**	**0.76**	**13.32**	**10.12**	**134.84**

*Proposed design

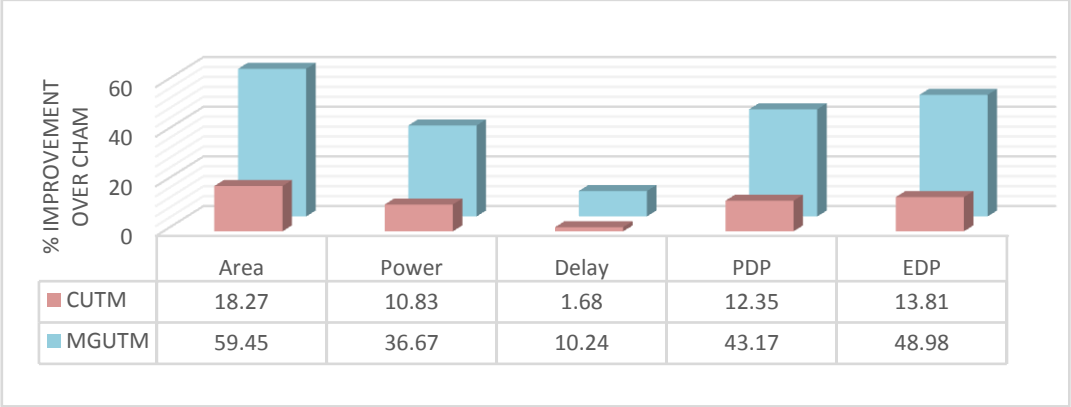


Fig. 9. Comparison of percentage improvements in various parameters of CUTM & MGUTM over CHAM

VI.CONCLUSION

The results obtained for the proposed multiplier design as shown in the Table 5, and Fig. 9. effectively prove that the proposed multiplier design works better than the other designs in terms of all the parameters in comparison. The most prominent improvement achieved over other designs is in termsof the area or transistor count. The use of Mod-GDI cells along with the UT theorem thus effectively reduces the area-constraints, marginally improves the speed of operation and also reduces the power consumption of the multiplier unit.

REFERENCES

[1] Somani, A., Jain, D., Jaiswal, S., Verma, K. and Kasht, S. (2012). Compare Vedic Multipliers with Conventional Hierarchical array of array multiplier. *International Journal of Computer Technology and Electronics Engineering*, 2(6), pp.52-55.

[2] Morgenshtein, A., Fish, A. and Wagner, I. (2002). Gate-diffusion input (GDI): a power-efficient method for digital combinatorial circuits. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, [online] 10(5), pp.566-581.

[3] Narendra, K. and Pandu, S. (2014). Low Power Area-Efficient Adiabatic Vedic Multiplier. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, [online] 03(08), pp.11027-11032.

[4] G.Deshpande, N. and Mahajan, R. (2014). Ancient Indian Vedic Mathematics based 32-Bit Multiplier Design for High Speed and Low Power Processors. *International Journal of Computer Applications*, [online] 95(24), pp.19-22.

[5] Kareem, A., M., V. and Kumar, P. (2017). VLSI Implementation of High Speed-Low Power-Area Efficient Multiplier Using Modified Vedic Mathematical Techniques. *Recent Patents on Computer Science*, [online] 9(3), pp.216-221.

[6] Kaur, S., Singh, B. and D.K, J. (2015). Design and Performance Analysis of Various Adders and Multipliers Using GDI Technique. *International Journal of VLSI Design & Communication Systems*, [online] 6(5), pp.45-56.

[7] Latha, B. and Rao, B. (2014). Design and Implementation of High Speed 8-Bit Vedic Multiplier on FPGA. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, [online] 03(08), pp.11601-11609.

[8] Manikandan, S. and Palanisamy, C. (2016). Design of an Efficient Binary Vedic Multiplier for High Speed Applications Using Vedic Mathematics with Bit Reduction Technique. *Circuits and Systems*, [online] 07(09), pp.2593-2602.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijareeie.com

Vol. 6, Issue 4, April 2017

[9] Manikrao, K. and Shrikant, M. (2016). Analysis of Array Multiplier and Vedic Multiplier using Xilinx. *Communications on Applied Electronics*, [online] 5(1), pp.13-16.
[10] Nicholas, A., Williams, K. and Pickles, J. (2010). *Vertically and Crosswise*. Castle Douglas, Scotland UK: Inspiration Books.
[11] Williams, K. and Gaskell, M. (1997). *The Cosmic Computer*. Skelmersdale: Inspiration Books.
[12] Weste, N. and Harris, D. (2011). *Principles of CMOS VLSI Design*. Pearson Education India.