2016 Conference on Advances in Signal Processing (CASP)
Cummins College of Engineering for Women, Pune. Jun 9-11, 2016

Low Power And Area Efficient Wallace Tree Multiplier Using Carry Select Adder With Binary To Excess-1 Converter

R. Bala Sai Kesava[1],
M. Tech Student,Department of ECE,
SRKR Engineering College, Bhimavaram, India
balasaikesava@gmail.com

K. Bala Sindhuri[3],
Assistant Professor, Department of ECE,
SRKR Engineering College, Bhimavaram, India
k.b.sindhuri@gmail.com

B. Lingeswara rao [2],
B. E Student, Department of ECE,
SRKR Engineering College, Bhimavaram, India
balasaikesava@gmail.com
b.lingeswararaoece@gmail.com

N. Udaya Kumar[4]
Professor, Department of ECE,
SRKR Engineering College, Bhimavaram, India
n_uk2010@yahoo.com

Abstract— **Multipliers are major blocks in the most of the digital and high performance systems such as Microprocessors, Signal processing Circuits, FIR filters etc. In the present scenario, Fast multipliers with less power consumption are leading with their performance. Wallace tree multiplier with carry select adder (CSLA) is one of the fastest multiplier but utilizes more area. To improve the performance of this multiplier, CSLA is replaced by binary excess-1 counter(BEC) which not only reduces the area at gate level but also reduces power consumption. Area and power calculations for the Wallace tree multiplier using CSLA with BEC are giving good results compared to regular Wallace tree multiplier**

Keywords— Carry Select Adder(CSLA),Binary to Excess one converter(BEC), Square Root Carry Select Adder(SQRT CSLA),Field Programmable Gate Array(FPGA),Half Adder (HA),Full Adder(FA),Ripple Carry Adder(RCA).

I. INTRODUCTION

A processor is the substantial block, in which the performances accelerate along with the processor speed of the system. The performance of the processor mainly depends on the multiplier as most of the processors time depends on the multiplication process. Major applications like VLSI, digital signal processing requires high performing processors to obtain the processing of huge amount of data. Techniques involved in most of the multipliers are computation of partial products and then the summation of partial products. In parallel multipliers number of partial products to be added is the main parameter that determines the performance of the multiplier. One of the general multiplier is serial/parallel multiplier. One of the operand is fed to the circuit in parallel while the other is serial. N partial products for M Bits of multiplicand and multiplier are formed for each cycle. On successive cycles, each cycle does the addition of one column of the multiplication table of M*N partial products. The final results are stored in the output register after N+M cycles. While the area required is N-1unitsfor M=N.

The common multiplication method is add and shift algorithm depending on the value LSB bit of multiplier, value of the multiplicand is added and accumulated. At each clock cycle the multiplier is shifted one bit to the right and its value is tested. If it is a 0, then only a shift operation is performed. If the value is a 1, then the multiplicand is added to the accumulator and is shifted by one bit to the right. After all the Multiplier bits have been tested the product is in the accumulator. The accumulator is 2N (M+N) in size and initially the N, LSBs contains the Multiplier. The delay is N cycles maximum. This circuit has several advantages in asynchronous circuits. But the main disadvantage is delay.

Array multiplier is one of the well known multiplier due to its regular structure. Multiplier circuit is designed based on add and shift algorithm. Partial products are generated by the multiplication of the multiplicand with one multiplier bit. The partial product are shifted according to their bit orders and then added. The addition can be performed with normal carry propagate adder. N-1 adders are required where N is the multiplier length. Till now we have discussed various multipliers in which delay is the major parameter which affects the speed.

Multiplier architecture consists of three stages, partial products generation stage these are generated by AND operation, partial products addition stage carried by different adders and final addition stage. Speed plays the major role in many of the multipliers. To reduce the delay and to accelerate the multiplication, the number of partial products is reduced. Wallace tree multiplier is designed with CSLA in order to increase the speed. This gives the solution for the problem of carry propagation delay by independently generating multiple carries and then selects a carry to obtain the sum. As we know CSLA is not area efficient as it uses multiple pairs of Ripple Carry Adders(RCA) partial sum and carry by considering cin=0 and cin =1,then final sum and carry are selected by multiplexers, this disadvantage made a reason to replace RCA

in regular CSLA with binary to excess-1 converter(BEC).BEC is replaced instead of RCA with cin = 1 in regular CSLA to achieve the lower area, power consumption and uses less number of logic gates, then final sum and carry is selected through MUX[1]. The importance of BEC logic stems from the large silicon area reduction when the CSLA with large number of bits are designed. The organization of this paper is as follows: In section II , the preliminaries delay evaluation methodology for full adder, two bit BEC and Wallace tree multiplier for four bit are presented . Section III deals with area and delay evaluation of 4 *4 Wallace tree multiplier .In section IV Wallace tree multiplier with CSLA is presented. Section V deals with the Wallace tree multiplier using CSLA with BEC .The simulated results is presented in section VI. The conclusion of the present work is discussed in section VII.

II. PRELIMINARIES

AND, OR and Inverter (AOI) gates each having delay equal to 1 unit and area equal to 1 unit are considered for calculation of power , delay and area . The gate count is done by adding the total number of AOI gates. This process is involved in low power and area efficient carry select adder [2]-[4]. The area evaluation is done by counting the total number of AOI gates required for each logic block. By this approach the accumulated area and delay count for XOR is Five units and Three units, for HA area count is Six units and delay count is Three units, while FA has Thirteen units area count and Six units day count,2:1MUX with Three units area count and Three units delay count. The implementation of full adder is shown in Fig.1.

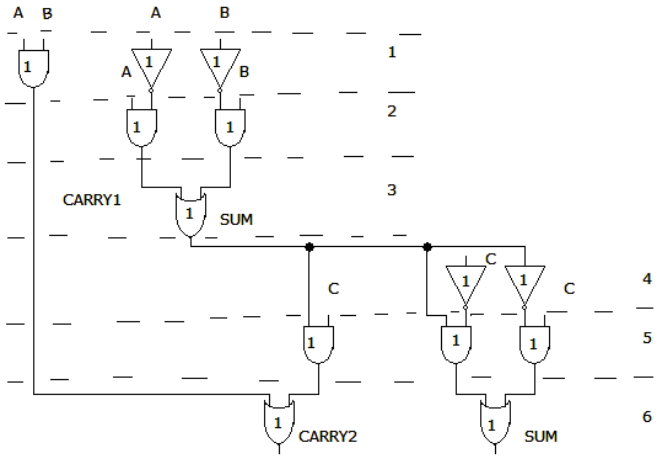

Fig.1.Delay Evaluation methodology for Full Adder

A. Two bit BEC

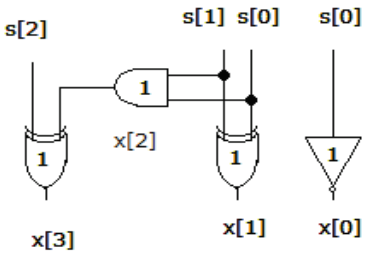

Fig.2.Binary to Excess-1 Converter

In regular CLSA as the utilization of gates are more due to this power consumption also increases. Now by using BEC in regular CSLA we can reduce number of gates and simultaneously decreases power consumption [5]-[7] .For two bit BEC first s (0) is given to the NOT gate to obtain x (0) . Next s (0), s (1) are given input for the XOR and AND gate to obtain sum x(1) , carry x(2). Next x (2), s(2) are given to XOR gate to generate sum x(3).

B. Wallace tree multiplier for 4-bit

Step by step procedure for multiplying two four bit integers according to Wallace multiplier.

Wallace multiplier consists of three steps.

1. Multiply each bit of one arguments with each bit of another argument, which results in n^2 products.
2. Consider the first three rows of the multiplied products and reduce them into two rows by using full adders and half adders as per the requirements. Repeat this process until two rows of multiplied products are obtained.
3. Normally in the case of four bit additions of two integers a sum of four bits and carry one bit is formed. So, in the last step of layer we first have two rows of products half adder to add last two bits and the carry of the half adder is connected to the next layer. By following the same procedure add all the bits of two rows. At last the sum of four bits can be obtained

Coming to the solution of 4*4 Wallace tree multiplier, In first stage we obtain four rows of the multiplied products as shown in the Fig. 2.

```
                          a3  a2  a1  a0
                          b3  b2  b1  b0
-------------------------------------------
                    a3b0  a2b0  a1b0  a0b0
              a3b1  a2b1  a1b1  a0b1
        a3b2  a2b2  a1b2  a0b2
  a3b3  a2b3  a1b3  a0b3
```

Fig.3.Partial Products Generation

Now in the second stage choose the first three rows and reduce them into two rows by using half adders and full adders. As per the requirement it is needed two half adders and two full adders, sum and carry are generated as a0b0, s(0)c(0), s(1)c(1), s(2)c(2), s(3)c(3), a3b2 as in the Fig.3.

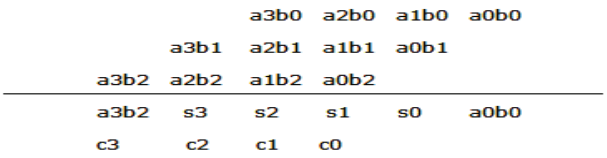

Fig.4.Partioning and Addition of the Partial Products for Group1

These obtained products are added to the fourth row of the multiplied products in the third stage by this operation the results of sum and carry as a0b0, s(0), s(4)c(4), s(5)c(5), s(6)c(6), s(7)c(7), a3b3 are obtained here one half adder and three full adders are used, It is shown in Fig.4 .

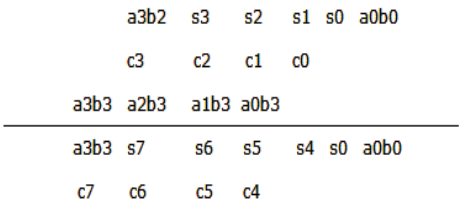

Fig.5. Partioning and Addition of the Partial Products for Group2

Finally in the fourth stage again add this sum and carry to obtain the last stage products as a0b0, s(0), s(4), s(8), s(9), s(10), s(11)c(11),two half adders and three full adders are used in this stage as shown in the below Fig.5.

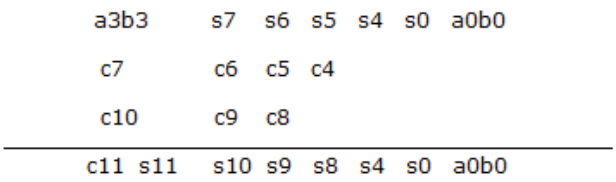

Fig.6. Partioning and Addition of the Partial Products for Group3

III Area and delay evaluation of 4*4 Wallace tree multiplier

Area is calculated by counting the number of components in the operation, while the delay is calculated by the number of stages in the operation involved. For example, for half adder, area count is six and delay count is three and in the case of full adder area count is thirteen and delay count is six. As the addition process starts from the stage two, the area count is thirty eight and the maximum delay is six. Now in stage three area counts is forty five and maximum delay is twelve. Finally in stage four area count is forty five and maximum delay is twenty nine .So total area count in entire multiplication for 4 bit is one hundred and twenty nine and maximum delay is twelve.

Theoretical calculation[5] for Wallace using CSLA uses four half adders and ten full adders and four multiplexers , area count for half adder is six , full adder is thirteen and mux is four by multiplying area count with number of adders used then total area count for Wallace using CSLA is obtained as one seventy units. Wallace using BEC uses four half adders , eight full adders , one NOT gate , two XOR gates , one AND gate and four multiplexers hence the final area count is one fifty six . By comparing Wallace using CSLA and Wallace using BEC , Wallace using BEC reduces the area count . practical results for area count are shown in the section VI.

IV.WALLACE TREE MULTIPLIER USING CSLA

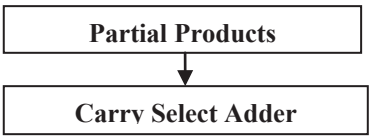

Fig.7.Block diagram for Wallace tree multiplier using CSLA

Partial products obtained from the group 2 were given to the carry select adder, as this addition process reduces the delay as compared with the normal Wallace tree multiplier to generate the final multiplication products

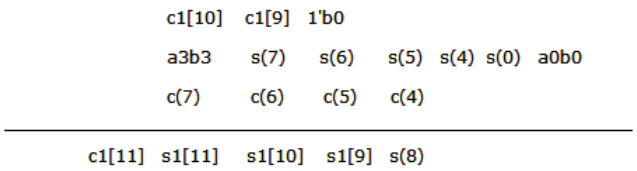

Fig.8. Partioning and Addition of the Partial Products when input carry is zero

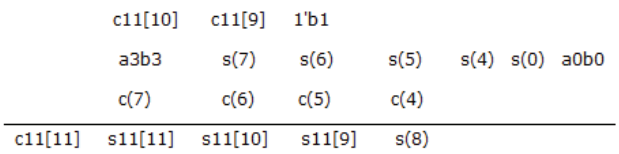

Fig.9. Partioning and Addition of the Partial Products when input carry is one

The CSLA [6] operation takes place by assuming one of the carry bit as 1'b0 and 1'b1. here in this case we assume 1'b0 as c(8) and 1'b1.In the case of 1'b0 we get the results as c1(11) , s1(11) , s1(10) , s1(9) , s(0) , s(4) , s(8) , a0b0.
In the next case c(8) as 1'b1 is assumed to get the result as a0b0 , s(0) , s(8) , s11(9) , s11(10) , s11(11) , c11(11).

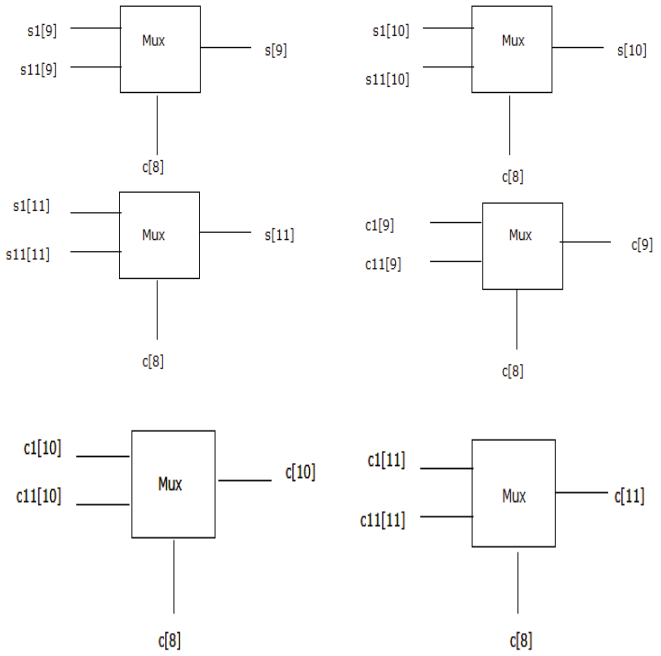

Fig.10.Selection of final results by using Multiplexers

Depending on the value of carry bit c(8) the selection of the bits either from the case of 1'b0 or from the case of 1'b1 takes places and it is the required output. The selection of outputs using carry bit c(8) is clearly mentioned below using multiplexers.

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	2	4656	0%
Number of 4 input LUTs	4	9312	0%
Number of bonded IOBs	8	232	3%

Fig.11.Simulation results for Wallace tree multiplier using CSLA for area count

CSLA uses multiple ripple carry adders in order to increase speed but the area is very high .In order to reduce area Ripple carry adder with cin =1 is replaced by means of BEC

V. Wallace tree multiplier using CSLA with BEC

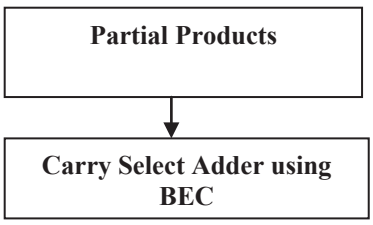

Fig.12. Block diagram for Wallace Tree Multiplier using BEC

While using the carry select adder with BEC method [7], Partial products obtained from the group 2 were given to the carry select adder using BEC in order to reduce the delay and it uses less number of gates when compared to the Wallace tree multiplier using CSLA, and obtain the final products of the multiplication.

```
                c1(10)  1'b0    c(8)
                a3b3    s(7)    s(6)   s(5)
                c(7)    c(6)    c(5)   c(4)
-------------------------------------------
        c1(11)  s1(11)  s1(10)  s(9)   s(8)
```

Fig.13. Partioning and Addition of the Partial Products when input carry is zero

The main reason for using the carry select adder with BEC is to reduce the no. of gates when compared to normal Wallace multiplier. The 1'b1 case in normal CSA is replaced by BEC. The result from 1'b0 case is given as inputs to the BEC adder.

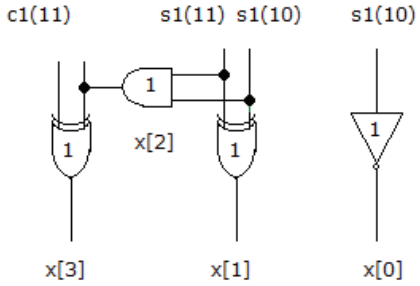

Fig.14 BEC with partial products as inputs

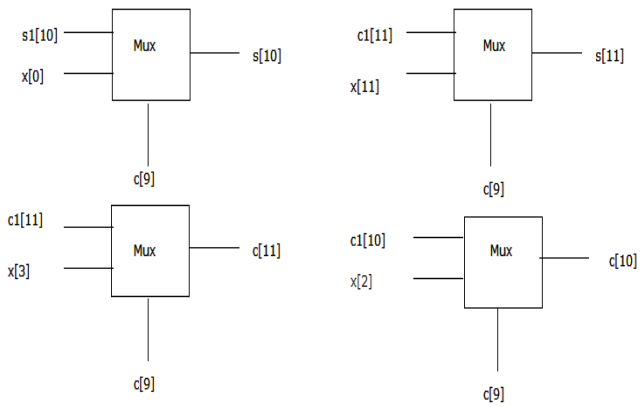

Fig.15. Selection of final results by using Multiplexers

Based on the selection line i.e., carry bit c (9) to the MUX the outputs are obtained. The inputs to the MUX are the outputs obtained from the cases of 1, b0 and BEC.

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	2	4656	0%
Number of 4 input LUTs	3	9312	0%
Number of bonded IOBs	6	232	2%

Fig.16.Simulation results for Wallace tree multiplier using CSLA with BEC for area count

VI. RESULTS

Wallace Multiplier is synthesized using XILINX ISE Design Suite 12.2 and is implemented on FPGA device xc3s500-5fg320 of Spartan 3E family.

The Input output waveforms which are generated by using XILINX software and device utilization summary are shown.

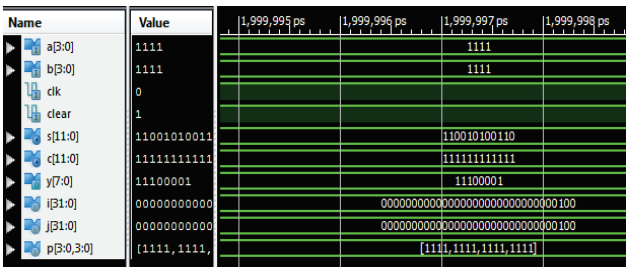

Fig.17 .simulation result for Wallace tree multiplier for 4bit

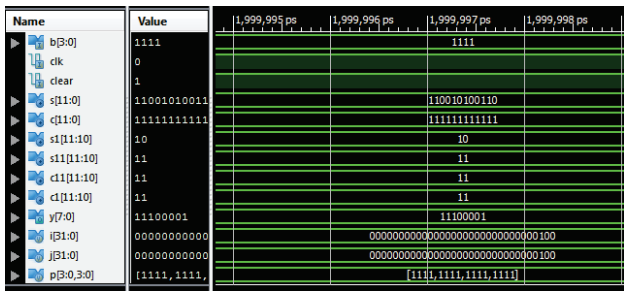

Fig.18. simulation result for Wallace tree multiplier using CSLA

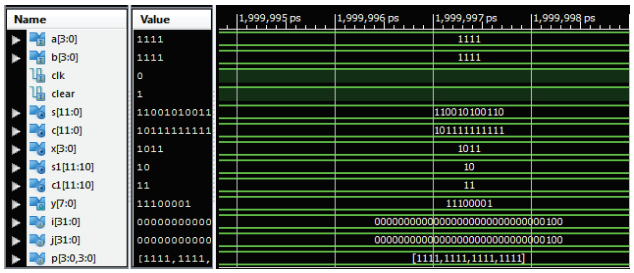

Fig.19. simulation result for Wallace tree multiplier using CSLA with BEC

Parameter	Wallace tree multiplier	Wallace Tree Multiplier using CSLA	Wallace tree multiplier using CSLA with BEC
Memory(KB)	13,7896	13,7768	103448
Delay(ns)	8.486	8.337	8.732
Power(mW)	83.22	82.88	80.98

Table.1. Comparison of Wallace tree multiplier using CSLA and BEC

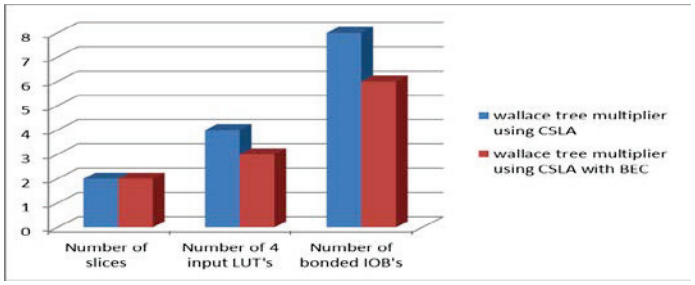

Fig.20 . Chart indicating area parameters for Wallace tree multiplier using CSLA and Wallace tree multiplier using CSLA with BEC

Number of slices for RCA in CSLA is two , where as number of slices in BEC is two .Number of 4 input LUT's for RCA in CSLA is four and number of input LUT's in BEC is three .Number of bonded IOB's for RCA in CSLA is eight and in case of using BEC is six. From these observations of area parameters it is clear that area utilization in Wallace tree multiplier using CSLA with BEC is area efficient .

VII CONCLUSION

A Simple approach is proposed in this paper to reduce the area of Wallace tree multiplier using CSLA. From the above results it is observed that the Wallace tree multiplier using CSLA with BEC is occupying less area, memory and consuming less power when compared to Wallace tree multiplier using CSLA and Wallace tree multiplier. This approach is showing slightly higher delay when compared to the other two approaches.

VIII REFERENCES

[1] Naveen Kr. Gahlan ,Prabhat Shukla,Jasbir Kaur ,"Implementation of Wallace Tree Multiplier Using Compressor ",Naveen Kr.Gahlan et al ,Int.J.Computer Technology & Applications,Vol 3 (3), 1194-1199.

[2] Jagadeshwar Rao M,Sanjay Dubey, "A High Speed Wallace Tree Multiplier Using Modified Booth Algorithm for Fast Arithmetic Circuits", IOSR Journal of Electronics and Communication Engineering (IOSRJECE) Volume 3, Issue 1(Sep-Oct 2012).

[3] Himanshu Bansal, K. G. Sharma, Tripti Sharma," Wallace Tree Multiplier Designs: A Performance Comparison Review",Innovative Systems Design and Engineering, Vol.5, No.5, 2014.

[4] Damarla Paradhasaradhi, M. Prashanthi and N.Vivek "Modified wallace tree multiplier using efficient square root carry select adder" (ICGCCEE/2014), s.no.61 ,pp.1-5,ISBN NO:978-1-4799-4982-3.

[5] B.Ramkumar and Harish M Kittur "Low-Power and Area-Efficient Carry Select Adder"371-375, VOL. 20, NO. 2, FEBRUARY 2012

[6] K.Gopi Krishna, B.Santhosh,V.Sridhar, " Design of Wallace Tree Multiplier using Compressors", International Journal of Engineering Sciences & Research Technology. ISSN: 2277-9655 .September 2013.

[7] E. Prakash, R. Raju, Dr.R. Varatharajan, "Effective Method For Implementation of Wallace Tree using Fast Adders", Journal of Innovative Research and Solutions (JIRAS) ,Volume.1,Issue No.1,July –Dec 2013.

[8] Y. He, C. H. Chang, and J. Gu, "An area efficient 64-bit square root carry-select adder for low power applications," in Proc.IEEE Int. Symp.Circuits Syst., 2005, vol. 4, pp. 4082–4085.

[9] K.BalaSindhuri, N.UdayKumar, D.V.N.Bharathi, B.Tapasvi "128-Bit Area–Efficient Carry Select Adder" IJRASET.

[10] B.Tapasvi, K.Balasindhuri, I.Chaitanya varma, N.udaya kumar, "Implementation of 64 Bit KoggeStone Carry Select Adder with BEC for Efficient Area", IJRECE Vol. 3 Issue 1 Jan –Mar 2015.

[11] G. Challa Ram, D.Sudha Rani, Y. Rama Lakshmanna, K. Bala sindhuri,"Area Efficient modified Vedic multiplier", International conference on circuit power and computing technologies.