International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

High Speed Vedic Multiplier Designs Using Novel Carry Select Adder

[1]chintakrindi Saikumar & [2]sk.Sahir

[1](M.Tech) VLSI, Dept. of ECE Priyadarshini Institute of Technology & Management
[2] Associate Professor, Dept. of ECE Priyadarshini Institute of Technology & Management

Abstract:
A system's performance is the major part which is dictated by the speed of the multiplier since multiplier is one of the key hardware components in high performance systems, for example, FIR filters, digital signal processors and microprocessors and so forth. Multipliers have the extensive range, long latency and consume considerable power. High speed is accomplished by diminishing carry propagation delay which might set aside less time for execution. Because of its rapid handling capacity, a multiplier is required. The generally happening issues in a multiplier are power dissipation and more delay. So we utilize Vedic multiplier which brings about less delay, in this manner A few multiplier building design expands the proficiency and execution of a framework. In this paper, a 16-bit X 5-bit multiplier is composed utilizing modified carry select adder which is productive to expand the speed of multiplication.

Keywords: High Speed Vedic; Multiplier Designs; Novel Carry; speed of multiplication

1. INTRODUCTION

Multipliers are key components of many high performance systems such as FIR filters, microprocessors, digital signal processors, etc. A system's performance is generally determined by the performance of the multiplier because the multiplier is generally the slowest clement in the system. Furthermore, it is generally the most area consuming. Hence, optimizing the speed and area of the multiplier is a major design issue. However, area and speed are usually conflicting constraints so that improving speed results mostly in larger areas. As a result, a whole spectrum of multipliers with different area-speed constraints has been designed with fully parallel. Multipliers at one end of the spectrum and fully serial multipliers at the other end. In between are digit serial multipliers where single digits consisting of several bits are operated on. These multipliers have moderate performance in both speed and area. However, existing digit serial multipliers have been plagued by complicated switching systems and/or irregularities in design. Radix 2^n multipliers which operate on digits in a parallel fashion instead of bits bring the pipelining to the digit level and avoid most of 'the above problems. They were introduced by M. K. Ibrahim in 1993. These structures are iterative and modular. The pipelining done at the digit level brings the benefit of constant operation speed irrespective of the size of' the multiplier. The clock speed *is* only determined by the digit size which is already fixed before the design is implemented.

The multipliers are the most important part of all digital signal processors; they are very important in realizing many important functions such as Fast Fourier Transforms and convolutions. Since a processor spends considerable amount of time in performing multiplication, an improvement in multiplication speed can greatly improve system performance. Multiplication can be implemented using many algorithms such as array, Wallace tree, carry save, and booth algorithms. Multipliers require high amount of power and delay during the partial product addition. At this stage, most of the multipliers are designed with different kind of adders that are capable to add a number of bits using compressors at the cost of increased delay.

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

This paper proposes the design of high speed multiplier using Novel carry select adder which has two main features. One is this multiplier involves the carry select addition that increases the speed of multiplication. Second is the compressors used for partial product addition operate simultaneously and are independent of previous stage outputs that also contribute in increasing the speed of multiplication.

2. LITERATURE REVIEW

Young-Ho Seo and Dong-Wook Kim have designed proposed a new architecture of multiplier-and-accumulator (MAC) for high-speed arithmetic. By combining multiplication with accumulation and devising a hybrid type of carry save adder (CSA), the performance was improved. Since the accumulator that has the largest delay in MAC was merged into CSA, the overall performance was elevated. Now, this paper, a new architecture for a high-speed MAC is proposed. In this MAC, the computations of multiplication and accumulation are combined and a hybrid-type CSA structure is proposed to reduce the critical path and improve the output rate. It uses MBA algorithm based on 1's complement number system. A modified array structure for the sign bits is used to increase the density of the operands. A carry look-ahead adder (CLA) is inserted in the CSA tree to reduce the number of bits in the final adder. In addition, in order to increase the output rate by optimizing the pipeline efficiency, intermediate calculation results are accumulated in the form of sum and carry instead of the final adder outputs. Zn = -2* Bn+1 + Bn + Bn-1, PPG is the combination circuit of the product generator and the 5 to 1 MUX circuit. Product generator is designed to produce the product by multiplying the multiplicand A by 0, 1, - 1, 2 or -2. A 5 to 1 MUX is designed to determine which product is chosen depending on the M, 2M, 3M control signal which is generated from the MBE. Now, for product generator, multiply by zero means the multiplicand is multiplied by "0".Multiply by "1" means the product still remains the same as the multiplicand value. Multiply by "-1"means that the product is the two's complement form of the number. Multiply by "-2" is to be shifted left one bit the two's complement of the multiplicand value and multiply by "2" means just shift left the multiplicand by only one place. In this paper, a new MAC architecture to execute the multiplication- accumulation operation, which is the key operation, for digital signal processing and multimedia information processing efficiently, was proposed. By removing the independent accumulation process that has the largest delay and merging it to the compression process of the partial products, the overall MAC performance has been improved almost twice as much as in the previous work.

Shaikh Kalisha Baba and D.Rajaramesh has design and implementation of Advanced Modified Booth Encoding (AMBE) multiplier for both signed and unsigned 32 - bit numbers multiplication. The already existed Modified Booth Encoding multiplier and the Baugh-Wooley multiplier perform multiplication operation on signed numbers only. Whereas the array multiplier and Braun array multipliers perform multiplication operation on unsigned numbers only. Thus, the requirement of the modern computer system is a dedicated and very high speed unique multiplier unit for signed and unsigned numbers. Therefore, this paper presents the design and implementation of AMBE multiplier. The modified Booth Encoder circuit generates half the partial products in parallel. By extending sign bit of the operands and generating an additional partial product the AMBE multiplier is obtained. The Carry Save Adder (CSA) tree and the final Carry Look ahead (CLA) adder used to speed up the multiplier operation. Since signed and unsigned multiplication operation is performed by the same multiplier unit the required hardware and the chip area reduces and this in turn reduces power dissipation and cost of a system. The high speed Booth multipliers and pipelined Booth multipliers are used for digital signal processing (DSP) applications such as for

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

multimedia and communication systems. High speed DSP computation applications such as Fast Fourier transform (FFT) require additions and multiplications. The conventional modified Booth encoding (MBE) generates an irregular partial product array because of the extra partial product bit at the least significant bit position of each partial product row. The requirement of the

3. DIFFERENT MULTIPLIERS

An efficient multiplier should have following characteristics:-

Accuracy:- A good multiplier should give correct result.

Speed:- Multiplier should perform operation at high speed.

Area:- A multiplier should occupies less number of slices and LUTs.

Power:- Multiplier should consume less power.

Multiplication process has three main steps[2]:

1. Partial product generation.
2. Partial product reduction.
3. Final addition.

For the multiplication of an n-bit multiplicand with an mbit multiplier, m partial products are generated and product formed is n + m bits long. Here we discuss about four different types of multipliers which are

1. Booth multiplier.
2. Combinational multiplier.
3. Wallace tree multiplier.
4. Array multiplier.
5. Sequential multiplier.

3.1. Conventional Multiplier

The basic algorithm for multiplication of two numbers, *M*(multiplier) and *N*(multiplicand), makes use of the distributive property of multiplication, i.e., if *M* can be written as sum of smaller numbers $M = M0 + M1 + \ldots + Mm\text{-}1$, then the multiplication $M \cdot N$ can be written as

$$M \cdot N = (M0 + M1 + \ldots + Mm\text{-}1).N$$
$$= M0.N + M1.N + \ldots + Mm\text{-}1.N \quad (1)$$

The terms on the right hand side are called partial products – smaller products, each one representing only a part of the total product. A multiplication algorithm finds a simple way to allow a simple calculation of the partial products. Then the total product is composed by summation of all partial products.

The value of a binary number $M = mm\text{-}1 \ldots m1m0$ can be written as

$$M = \sum_{i=0}^{m-1} m_i \cdot 2^i$$

Where mi are the different bits of M, and m is the total number of bits.

If the partial products are $Pm\text{-}1 \ldots P1P0$, then

$P0 = N \cdot m0$;

$P1 = N \cdot m1$; and embed one zero before least significant bit.

$Pm\text{-}2 = N \cdot mm\text{-}2$; and embed *(m-2)* zeros before least significant bit.

$Pm\text{-}1 = N \cdot mm\text{-}1$; and embed *(m-1)* zeros before least significant bit.

Then, $M \cdot N = P0 + P1 + \ldots + Pm\text{-}2 + Pm\text{-}1$

Special case: If M = 5 bit and N = 14 bit, then partial products are P0, P1, P2, P3 and P4.

A bit-wise addition of partial products for this special case using conventional method can be explained with the help of Table 1. The schematic diagram for the same is shown in Figure 1.

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

Table 1. A bit-wise addition of partial products for this special case using conventional method

Bits to be added	Sum	Carry [0]	Carry[1]
$P_0[0]$	M.N[0]	0	0
$P_0[1],P_1[1]$	M.N[1]	$C_1[0]$	0
$P_0[2],P_1[2],P_2[2],C_1[0]$	M.N[2]	$C_2[0]$	$C_2[1]$
$P_0[3],P_1[3],P_2[3], P_3[3],C_2[0]$	M.N[3]	$C_3[0]$	$C_3[1]$
$P_0[4],P_1[4],P_2[4],P_3[4],P_4[4],C_3[0],C_2[1]$	M.N[4]	$C_4[0]$	$C_4[1]$
$P_0[5],P_1[5],P_2[5],P_3[5],P_4[5],C_4[0],C_3[1]$	M.N[5]	$C_5[0]$	$C_5[1]$
$P_0[6],P_1[6],P_2[6],P_3[6],P_4[6],C_5[0],C_4[1]$	M.N[6]	$C_6[0]$	$C_6[1]$
$P_0[7],P_1[7],P_2[7],P_3[7],P_4[7],C_6[0],C_5[1]$	M.N[7]	$C_7[0]$	$C_7[1]$
$P_0[8],P_1[8],P_2[8],P_3[8],P_4[8],C_7[0],C_6[1]$	M.N[8]	$C_8[0]$	$C_8[1]$
$P_0[9],P_1[9],P_2[9],P_3[9],P_4[9],C_8[0],C_7[1]$	M.N[9]	$C_9[0]$	$C_9[1]$
$P_0[10],P_1[10],P_2[10], P_3[10],P_4[10],C_9[0], C_8[1]$	M.N[10]	$C_{10}[0]$	$C_{10}[1]$
$P_0[11],P_1[11],P_2[11], P_3[11],P_4[11],C_{10}[0], C_9[1]$	M.N[11]	$C_{11}[0]$	$C_{11}[1]$
$P_0[12],P_1[12],P_2[12], P_3[12],P_4[12],C_{11}[0], C_{10}[1]$	M.N[12]	$C_{12}[0]$	$C_{12}[1]$
$P_0[13],P_1[13],P_2[13], P_3[13],P_4[13],C_{12}[0], C_{11}[1]$	M.N[13]	$C_{13}[0]$	$C_{13}[1]$
$P_1[14],P_2[14],P_3[14], P_4[14],C_{13}[0],C_{12}[1]$	M.N[14]	$C_{14}[0]$	$C_{14}[1]$
$P_2[15],P_3[15],P_4[15], C_{14}[0], C_{13}[1]$	M.N[15]	$C_{15}[0]$	$C_{15}[1]$
$P_3[16],P_4[16],C_{15}[0], C_{14}[1]$	M.N[16]	$C_{16}[0]$	$C_{16}[1]$
$P_4[17],C_{16}[0],C_{15}[1]$	M.N[17]	$C_{17}[0]$	$C_{17}[1]$
$C_{17}[0],C_{16}[1]$	M.N[18]	$C_{18}[0]$	0
$C_{18}[0],C_{17}[1]$	M.N[19]	$C_{19}[0]$	0
$C_{19}[0]$	M.N[20]	0	0

IV. INTRODUCTION TO MODIFIED CARRY SELECT ADDER (MSCLA)

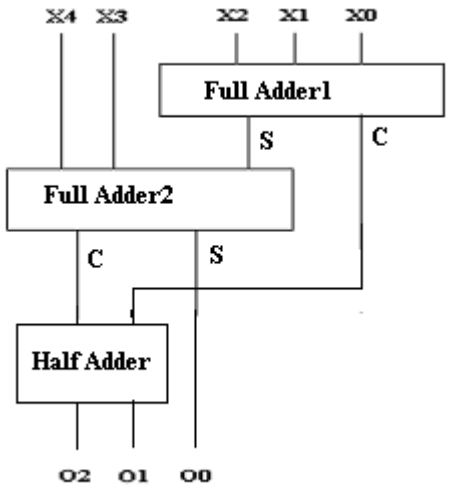
Modified Carry Select Adder is designed to add up to five 16-bit numbers. This MCS*l*A is designed by using two components. First component is compressor that compresses five bits into two bits. Second component is carry select adder that generates the result by using ripple carry adders and multiplexers.

A. Compressor

For compressing five bits into two bits, two compressors are used. One is 5:3 compressor and another one is 3:2 compressor.

(i) 5:3 Compressors

5:3 Compressors is made up of two full adders and one half adder. It is designed to add five bits and generates final sum of three bits [6, 7]. The block diagram of 5:3 compressor is shown in Figure 2(a).



International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

Figure 2(a): Block Diagram of 5:3 Compressor

(ii) 3:2 Compressor

3:2 Compressor is simply a full adder that adds Three bits and generates two bit output as sum and carry. The block diagram of 3:2 compressor is shown in Figure 2(b).

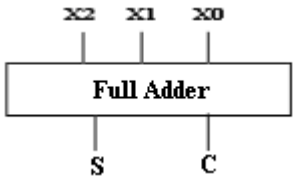


Figure 2(b): Block Diagram of 3:2 Compressor

B. Carry Select Adder (CSlA)

Carry Select Adder architecture consists of independent generation of sum and carry i.e., Cin=1 and Cin=0 are executed in parallel [5]. Depending upon Cin, the external multiplexers select the carry to be propagated to next stage. Further, based on the carry input, the pre calculated sum is selected and delay is reduced. However, the structure is increased due to the complexity of multiplexers [3].The architecture of CS*l*A is shown in Figure 3.

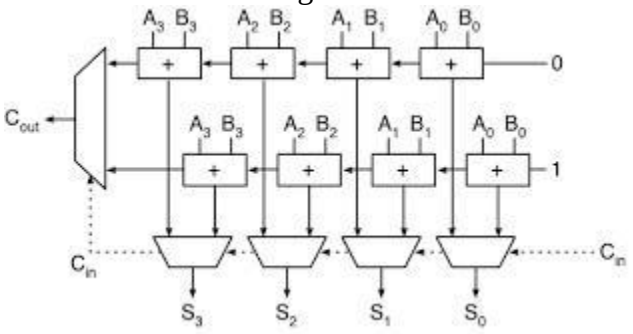


Figure 3: Block Diagram of CS*l*A

IV. DESIGN OF MODIFIED CARRY SELECT ADDER

This Modified Carry Select Adder takes five 16-bit numbers P, Q, R, S and T as input and generates 18-bit sum and a carry. The architecture of MCS*l*A is described step by step with the help of Figure 4(a), Figure 4(b), Figure 4(c), Figure 4(d). The schematic diagram for MCS*l*A is shown in Figure 5.

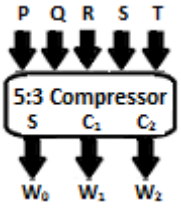


Figure 4(a): 16-bit 5:3 Compressor

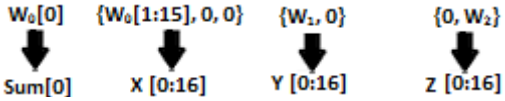


Figure 4(b): Rearrangement of bits

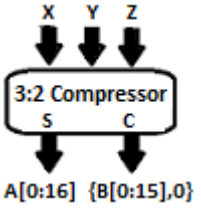
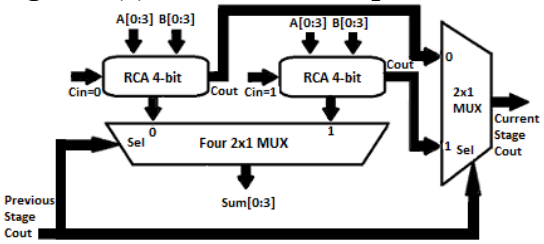


Figure 4(c): 17-bit 3:2 Compressor



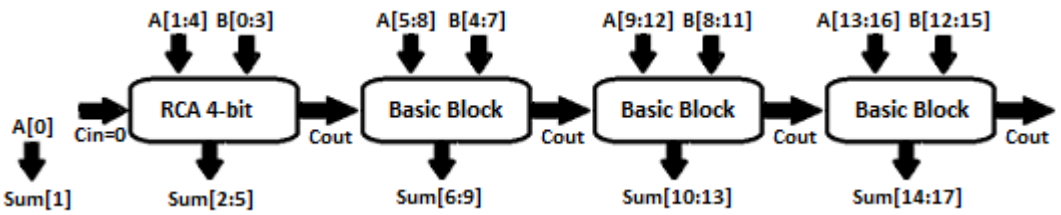


Figure 4(d): Block Diagram of CS*l*A

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

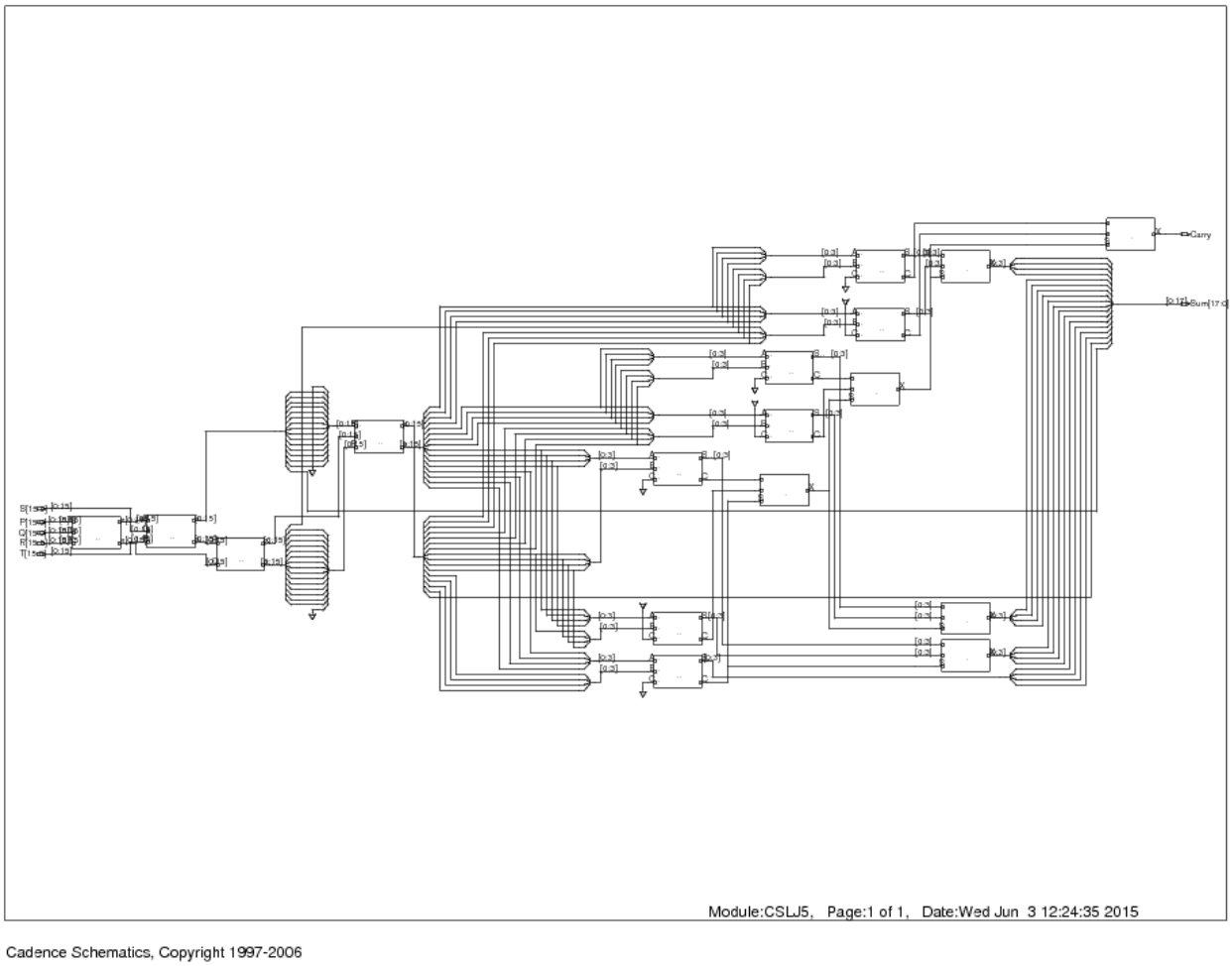


Figure 5: Schematic diagram of MCS*l*A

V. DESIGN OF HIGH SPEED MULTIPLIER USING CARRY SELECT ADDER

The designed high speed multiplier can be divided into two parts- partial product generator and modified carry select adder. The partial product generator involves the logical AND operation and bit shifting. The modified carry select adder generates the result by adding partial product terms.

A. Partial Product Generator

It generates partial product terms P0P1P2P3P4 by using 5-bit multiplier M and 16-bit multiplicand N as per following equations:

P0 = N AND M[0];

P1 = N AND M[1]; and embed one zero before least significant bit.

P2 = N AND M[2]; and embed two zeros before least significant bit.

P3 = N AND M[3]; and embed three zeros before least significant bit.

P4 = N AND M[4]; and embed four zeros before least significant bit.

B. Modified Carry Select Adder

Modified carry select adder takes P0, P1, P2, P3 and P4 as input and generates the result M.N.

The block diagram of designed high speed multiplier using a half adder and modified carry select adder is shown in Figure 6. The schematic diagram for high speed multiplier using MSC*l*A is shown in Figure 6.

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

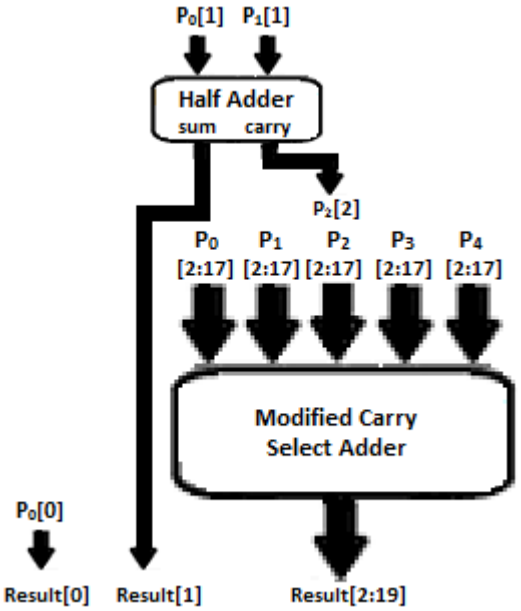


Fig 6. Schematic diagram for high speed multiplier using MSC*l*A

VI. COMPARISON

In this paper the proposed high speed multiplier using modified carry select adder and conventional multiplier have been simulated using Cadence Tool. The comparison is done on the basis of parameters- power consumption, speed of operation and area used. The comparative results of these two multipliers are shown in Table 2.

Table 2. The comparative results of these two multipliers

Parameter	% increase/decrease for proposed multiplier as compared to conventional multiplier
Power	48.22 % increased
Delay	80.27 % decreased
Area	30.20 % increased

VII. CONCLUSIONS

Different types of adder topologies are used to add the partial products [1]. On comparing different adder topologies, it is found that carry select adder is the fastest one among other mostly used adders [2]. The carry select adder is modified to add up to five 16-bit and used it for the addition of partial products generated during multiplication. This designed multiplier is 80.27% faster than conventional multiplier. It can be used for high speed applications where area and power consumption are not major issues.

VIII. REFERENCES

[1] R.Uma, Vidya Vijayan, M. Mohanapriya, and Sharon Paul, "Area, Delay and Power Comparison of Adder Topologies", International Journal of VLSI Design & Communication Systems, vol.3, no.1, pp.153-168,Feb 2012.

[2] Jasbir Kaur, Lalit Sood, "Comparison Between Various Types of Adder Topologies", IJCST vol.6, Issue 1, Jan-March 2015.

[3] Sarabdeep Singh, Dilip Kumar, "Design of Area and Power Efficient Modified Carry Select Adder", International Journal of Computer Applications, vol.33, no.3, pp.14-18,Nov 2011.

[4] Padma Devi, Ashima Girdher and Balwinder Singh, "Improved Carry Select Adder with Reduced Area and Low Power Consumption", International Journal of Computer Application,Vol 3.No.4, June 2010 .

[5] Romana Yousuf and Najeeb-ud-din,"Synthesis of Carry Select Adder in 65 nm FPGA", IEEE.

[6] J. Gu, C. Chang (2003), "Ultra low voltage low power 4-2 compressor for high speed multiplications", Proceedings of IEEE International Symposium on Circuits and Systems", Vol. 5, pp. 321-324.

[7] K. Prasad, K.K. Parhi (2001), "Low power 4-2 and 5-2 compressors", Proceedings of 35th Asilomar Conference on Signals, Systems and Computers, Vol. 1, pp. 129-133.

[8] V.G. Oklobdzija, D. Villeger, S.S. Liu (1996), "A method for speed optimized partial product reduction and generation of fast parallel multipliers using an algorithmic approach", IEEE Transactions on Computers, Vol. 45, No. 3.

[9] P. Stelling, C. Martel, V.G. Oklobdzija, R. Ravi (1998), "Optimal circuit for parallel multipliers", IEEE Transactions on Computers, Vol. 47, No. 3.

[10] V.G. Oklobdzija (2000), "High speed VLSI arithmetic unit: Adders and Multipliers", in Design of High Performance Microprocessor Circuits", Editor A. Chandrakasan, IEEE Press.