ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 9, September 2014

Design of Efficient FIR Filter MAC unit Using Parallel Prefix Adder

J.Ravi[1], K. Rama Rao[2], N. Tirumala[3]

Student, Department of ECE, Indur Institute of Engineering and Technology, Siddipet, India[1]

Associate Professor, Department of ECE, Indur Institute of Engineering and Technology, Siddipet, India[2]

Assistant Professor, Department of ECE, Stanley College, Hyderabad, India[3]

Abstract: Digital Signal Processing (DSP) is a field of Utmost importance as it performs the processing of a digital signal. DSP techniques improve signal quality or extract important information by removing unwanted parts of the signal. This extraction of the unwanted parts of the signal is possible with the help of filters. A Finite Impulse Response (FIR) filters play a crucial role in many of the signal processing applications. The output is computed using Multiply and Accumulate (MAC) operations. The functionality of MAC unit enables high-speed filtering and other processing typical for DSP applications. A MAC unit consists of a multiplier and accumulator. In this paper multiplier is designed using modified Wallace multiplier and the adder used is Parallel Prefix Adder and is compared with other adders that is Carry Save Adder and Carry Select Adder. This Paper Presents Design of Low Power and High Speed MAC unit with Modified Wallace Multiplier and Parallel Prefix Adder (Kogge-Stone adder) for FIR filter. And it also gives the Comparison of three adders (Parallel Prefix Adder, Carry Save Adder, and Carry Select Adder) in Case of Power, Delay and Area. Modified Wallace multiplier with three different adders has been written Coded in VHDL and then synthesized and simulated using Xilinx ISE 9.2i. The MAC unit which designed by using Modified Wallace Multiplier and Parallel Prefix adder has consumes less Power and has less delay by when compared with other adders.

Keywords: Modified Wallace multiplier, Parallel prefix adder, Kogge-Stone adder, Carry Select adder, Carry Save adder.

I. INTRODUCTION

MAC unit is an inevitable component in many digital signal processing (DSP) applications involving multiplications and/or accumulations. MAC unit is used for high performance digital signal processing systems. The DSP applications include filtering, convolution, and inner products. Most of digital signal processing methods use nonlinear functions such as discrete cosine transform (DCT) or discrete wavelet transforms (DWT). Because they are basically accomplished by repetitive application of multiplication and addition, the speed of the multiplication and addition arithmetic determines the execution speed and performance of the entire calculation [1]. Multiplication-and-accumulate operations are typical for digital filters. Therefore, the functionality of the MAC unit enables high-speed filtering and other processing typical for DSP applications. Since the MAC unit operates completely independent of the CPU, it can process data separately and thereby reduce CPU load. The application like optical communication systems which is based on DSP, require extremely fast processing of huge amount of digital data. The Fast Fourier Transform (FFT) also requires addition and multiplication. 64 bit can handle larger bits and have more memory.

A MAC unit consists of a multiplier and an accumulator containing the sum of the previous successive products. The MAC inputs are obtained from the memory location and given to the multiplier block. The design consists of 64 bit modified Wallace multiplier, 128 bit three different adders are used at final stage.

This paper is divided into seven sections. In the first section the introduction about MAC unit is discussed. In the second section discuss about the detailed operation of MAC unit. The third and fourth section deals with the operation of modified Wallace multiplier and Parallel Prefix Adder respectively. In the fifth and sixth section deals with carry save adder and carry select adder. In the Seventh section obtained result for the 64 bit MAC unit is discussed and finally the conclusion is made in the eighth section.

II. MAC OPERATION

The Multiplier-Accumulator (MAC) operation is the key operation not only in DSP applications but also in multimedia information processing and various other applications. As mentioned above, MAC unit consist of multiplier, adder and register/accumulator. In this paper, we used 64 bit modified Wallace multiplier. The MAC inputs are obtained from the memory location and given to the multiplier block. This will be useful in 64 bit digital signal processor. The input which is being fed from the memory location is 64 bit. When the input is given to the multiplier it starts computing value for the given 64 bit input and hence the output will be 128 bits. The multiplier output is given as the input to three different adders which performs addition.

The function of the MAC unit is given by the following equation [4]:

$$F = \sum P_j Q_j \quad \text{----------} \qquad (1)$$

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 9, September 2014

The output of three adders is 129 bit i.e. one bit is for the carry (128bits+ 1 bit). Then, the output is given to the accumulator register. The accumulator register used in this design is Parallel in Parallel out (PIPO). Since the bits are huge and also three adder produces all the output values in parallel, PIPO register is used where the input bits are taken in parallel and output is taken in parallel. The output of the accumulator register is taken out or fed back as one of the input to different three adder. The figure 1 shows the basic architecture of MAC unit.

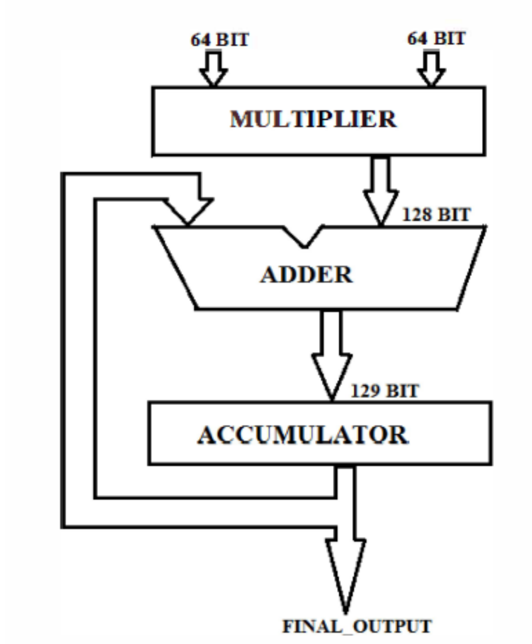


Figure: 1 Basic architecture of MAC unit

III. MODIFIED WALL ACE MULTIPLIER

A modified Wall ace multiplier is an efficient Hardware implementation of digital circuit multiplying two integers. Generally in conventional Wallace multipliers many full adders and half adders are used in their reduction phase. Half adders do not reduce the number of partial product bits. Therefore, minirnizing the number of half adders used in a multiplier reduction will reduce the complexity [2]. Hence, a modification to the Wallace reduction is done in which the delay is the same as for the conventional Wallace reduction. The modified reduction method greatly reduces the number of half adders with a very slight increase in the number of full adders [2].

Reduced complexity Wall ace multiplier reduction consists of three stages [2]. First stage the N x N product matrix is formed and before the passing on to the second phase the product matrix is rearranged to take the shape of inverted pyramid. During the second phase the rearranged product matrix is grouped into non-overlapping group of three as shown in the figure 2, single bit and two bits in the group will be passed on to the next stage and three bits are given to a full adder. The number of rows in the in each stage of the reduction phase is calculated by the formula

rj+ 1= 2[rj/3]+rjmod3 --------- (2)

If rj mod3 = 0, then rj+ 1 = 2r/3 -------- (3)

If the value calculated from the above equation for Number of rows in each stage in the second phase and the number of row that are formed in each stage of the second phase does not match, only then the half adder will be used. The final product of the second stage will be in the height of two bits and passed on to the third stage. During the third stage the output of the second stage is given to the carry propagation adder to generate the final output.

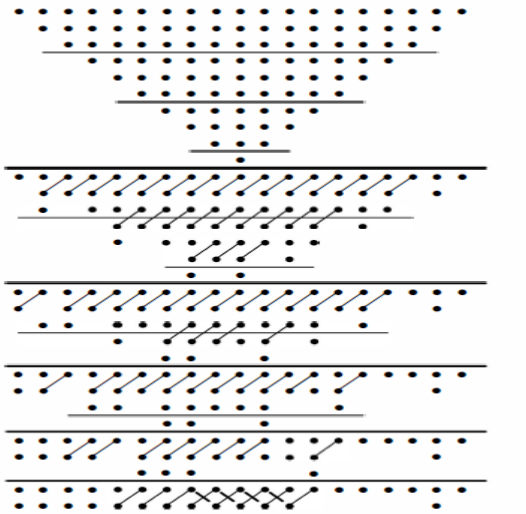


Figure: 2 Modified Wallace 10-bit by 10-bit reduction

Thus 64 bit modified Wallace multiplier is constructed and the total number of stages in the second phase is 10. As per the equation the number of row in each of the 10 stages was calculated and the use of half adders was restricted only to the 10^{th} stage. The total number of half adders used in the second phase is 8 and the total number of full adders that was used during the second phase is slightly Since the 64 bit modified Wallace multiplier is difficult to represent, a typical 10-bit by 10-bit reduction shown in figure 2 for understanding. The modified Wallace tree shows better performance when Parallel prefix adder is used in final stage instead of ripple carry adder. The Parallel Prefix adder which is used is considered to be the critical part in the multiplier because it is responsible for the largest amount of computation. Increased that in the conventional Wallace multiplier. Since the 64 bit modified Wallace multiplier is difficult to represent, a typical 10-bit by 10-bit reduction shown in figure 2 for understanding. The modified Wallace tree shows better performance when three adders are used once at a time in final stage instead of ripple carry adder.

IV. PARALLEL PREFIX ADDER

Parallel Prefix adder is that it is primarily fast when compared with ripple carry adders. Parallel Prefix adders (PPA) are family of adders derived from the commonly known carry look ahead adders. These adders are best suited for adders with wider word lengths. PPA circuits use a tree network to reduce the latency to O (log2 n) where „n" represents the number of bits. Parallel Prefix Adders (PPA) is variations of the well-known carry look ahead adder (CLA). The difference between a CLA and a PPA lies in the second stage which is responsible for the generation of the carry signals of the binary addition. A parallel Prefix Addition is generally a three step process.

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 9, September 2014
ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

The first step involves the creation of generate (gi) and Propagate (pi) signals for the input operand bits. The second step involves the generation of carry signals and finally a simple adder to generate sum. The three stage structure of carry look ahead adder and parallel prefix adder is shown

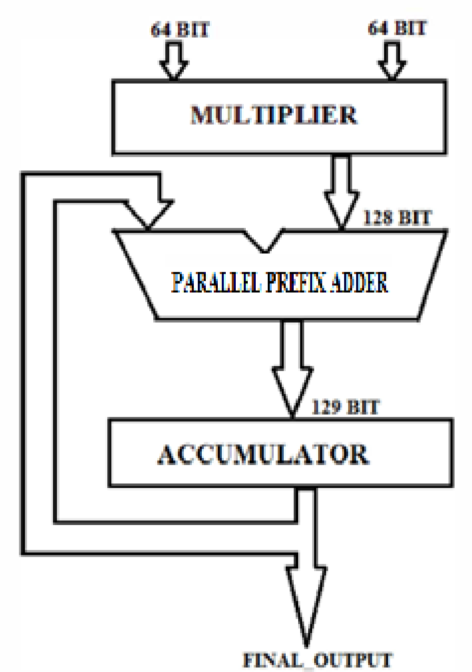


Figure: 3 Modified Wallace Multiplier With Parallel Prefix adder

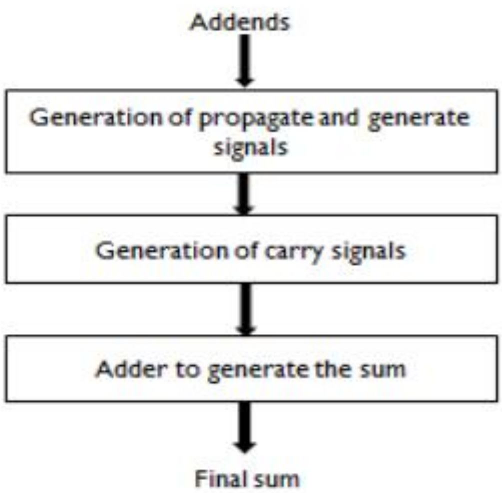
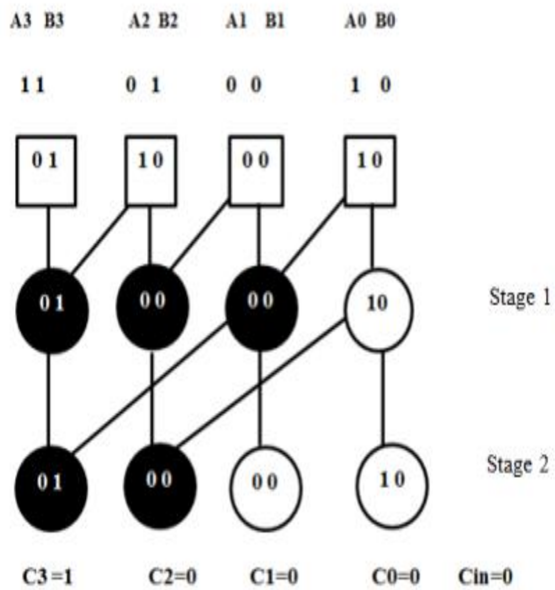


Figure: 4 three stage structure of the carry look ahead and parallel prefix adder.

Parallel-prefix adders, also known as carry-tree adders, pre-compute the propagate and generate signals. These signals are variously combined using the fundamental carry operator (fco). Beside fundamental carry operator also known as black operator or dot operator as shown in figure 2, there is another component called buffer component which translates the generate and propagate signals. The two operators are shown in the figure:



In a 4 bit adder like the one shown in the picture to the right, there are 5 outputs. Below is the expansion:

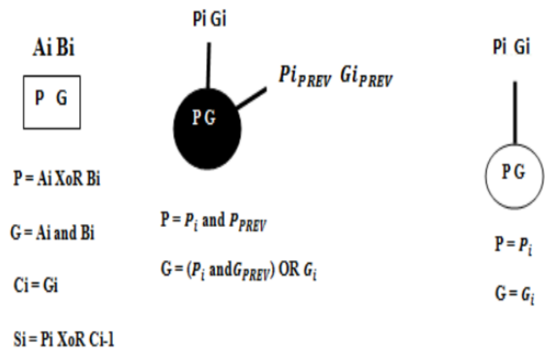


Figure: 5 4 bit kogge-stone adder

S0 = (A0 XOR B0) XOR
S1 = (A1 XOR B1) XOR (A0 AND B0)
S2 = (A2 XOR B2) XOR (((A1 XOR B1) AND (A0 AND B0)) OR (A1 AND B1))
S3 = (A3 XOR B3) XOR ((((A2 XOR B2) AND (A1 XOR B1)) AND (A0 AND B0)) OR (((A2 XOR B2) AND (A1 AND B1)) OR (A2 AND B2)))
S4 = (A4 XOR B4) XOR ((((A3 XOR B3) AND (A2 XOR B2)) AND (A1 AND B1)) OR (((A3 XOR B3) AND (A2 AND B2)) OR (A3 AND B3)))

V. CARRY SAVE ADDER

In this design 128 bit carry save adder [6] is used since the output of the multiplier is 128 bits (2N). The carry save adder minirnize the addition from 3 numbers to 2 numbers. The propagation delay is 3 gates despite of the number of bits. The carry save adder contains n full adders, computing a single sum and carries bit based mainly on the respective bits of the three input numbers. The entire sum can be calculated by shifting the carry sequence left by one place and then appending a 0 to most significant bit of the partial sum sequence. Now the partial sum sequence is added with ripple carry unit resulting in n + 1 bit value. The ripple carry unit refers to the process

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 9, September 2014

where the carryout of one stage is fed directly to the carry in of the next stage. This process is continued without adding any intermediate carry propagation. Since the representation of 128 bit carry save adder is infeasible , hence a typical 8 bit carry save adder is shown in the figure 3[6].Here we are computing the sum of two 128 bit binary numbers, then 128 half adders at the first stage instead of 128 full adder. Therefore , carry save unit comprises of 128 half adders, each of which computes single sum and carry bit based only on the corresponding bits of the two input numbers.

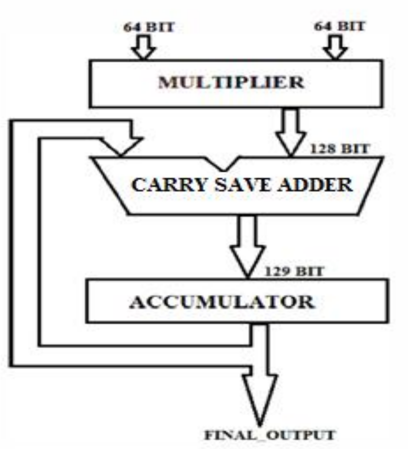


Figure: 6 Modified Wallace Multiplier With Carry Save Adder

If x and y are supposed to be two 128 bit numbers then it produces the partial products and carry as S and C respectively.

$S_i = X_i \wedge Y_i$ ---------- 4

$C_i = X_i \& Y_i$ ---------- 5

During the addition of two numbers using a half adder, two ripple carry adder is used. This is due the fact that ripple carry adder cannot compute a sum bit without waiting for the previous carry bit to be Produced, and hence the delay will be equal to that of n full adders. However a carry-save adder produces all the output values in parallel, resulting in the total computation time less than ripple carry adders. So, Parallel In Parallel Out (PIPO) is used as an accumulator in the final stage.

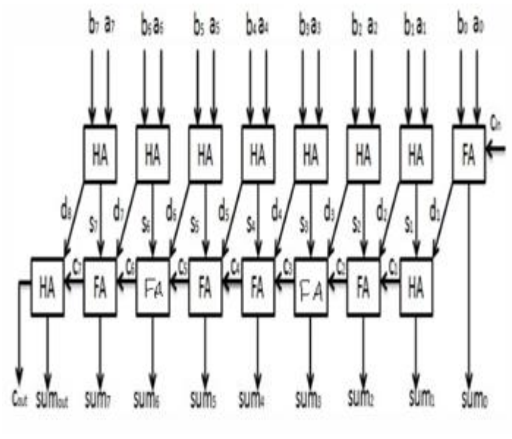


Figure: 7 8 bit carry save adder

VI. CARRY SELECT ADDER

In electronics, a carry-select adder is a particular way to implement an adder, which is a logic element that computes the (n+1) bit sum of two n-bit numbers. The carry-select adder is simple but rather fast, having a gate level depth of O√n.

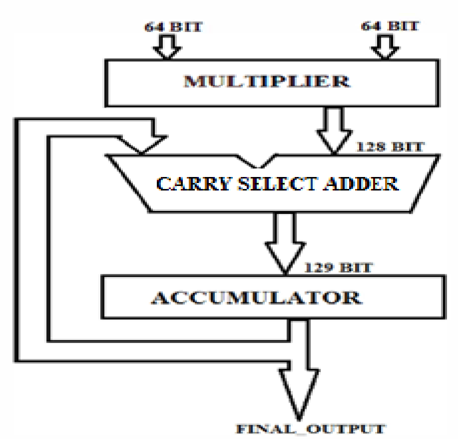


Figure: 8 Modified Wallace Multiplier With Carry Select Adder

The carry-select adder generally consists of two ripple carry adders and a multiplexer. Adding two n-bit numbers with a carry-select adder is done with two adders (therefore two ripple carry adders) in order to perform the calculation twice, one time with the assumption of the carry being zero and the other assuming one. After the two results are calculated, the correct sum, as well as the correct carry, is then selected with the multiplexer once the correct carry is known.

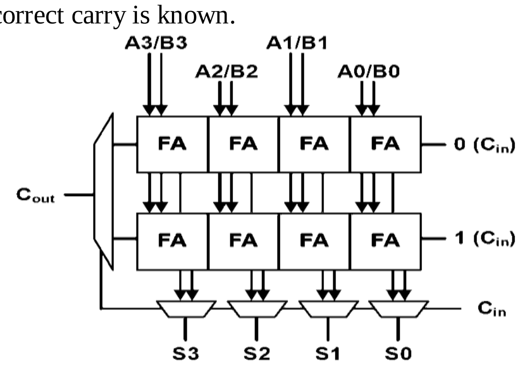


Figure: 9 4 bit carry Select adder

VII. RESULT

The design is developed using in VHDL and then synthesized and simulated using Xilinx ISE 9.2i. And Power Calculations have found for Different MAC Unit (with different adders) using Xilinx Analyse PowerX.

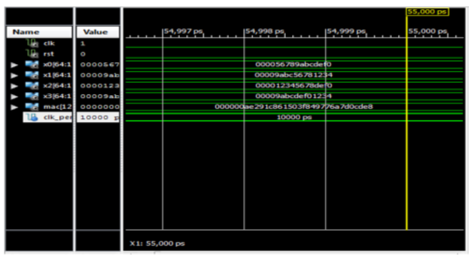


Figure: 10 4 Simulation report of 64 bit 4 Tap fir filter Mac Unit

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

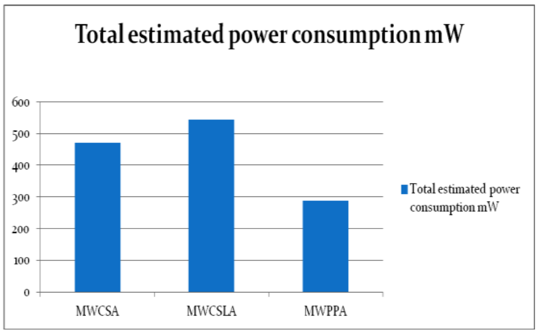
IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 9, September 2014

A Multiplier is designed Modified Multiplier and with Three adders (Carry Save Adder, Carry Select Adder and Parallel Prefix Adder). Area, Power and Delay results are Present Below

POWER CALCULATIONS:

	Total estimated power consumption
MWCSA	471 mW
MWCSLA	543mW
MWPPA	289mW



Graph 1: Comparison Graph between MWCSA, MWCSLA and MWPPA

MWCSA: Modified Wallace Carry save Adder
MWCSLA: Modified Wallace Carry Select Adder
MWPPA: Modified Wallace Parallel Prefix Adder

AREA COMPARISION: (Only Adders)

Device Utilization Summary			
Logic Utilization	Used by CSA	Used by CSLA	Used by PPA
No of Slices	2038	2174	2539
No of Slice Flip Flop	204	203	203
No of $ input LUT's	3683	3944	4643
No of Bounded I/O's	387	387	387

DELAY COMPARISION: (Only Adders)

DELAY COMPARISION BETWEEN ADDERS

ADDER	TIMING DELAY
CSA	151.106ns (82.611ns logic, 68.495ns route) (54.7% logic, 45.3% route)
CSLA	173.758ns (59.971ns logic, 113.787ns route) (34.5% logic, 65.5% route)
PPA	26.699ns (11.858ns logic, 14.841ns route) (44.4% logic, 55.6% route)

VIII. CONCLUSION

Hence a design of High Speed Power Efficient and Less Delay 64 bit 4 TAP FIR Filter Multiplier and Accumulator (MAC) Unit is designed in this paper. The total estimated power consumed by 64 bit MAC unit (Which is designed using Modified Wallace Multiplier and Parallel Prefix Adder) is 289 mW. The total Delay Time for Parallel Prefix Adder when compared to other adders is **26.669 ns.** Since the delay of 64 bit is less, this design can be used in the system which requires high performance in processors involving large number of bits of the operation. The MAC unit is designed using VHDL and then synthesized and simulated using Xilinx ISE 9.2i.

REFERENCES

[1] Young-Ho Seo and Dong-Wook Kim, "New VLSI Architecture of Parallel Multiplier-Accumulator Based on Radix-2 Modified Booth Algorithm," IEEE Transactions on very large scale integration (vlsi) systems, vol. 18, no. 2,february 20 10
[2] Ron S.Waters and Earl E. Swartzlander, Jr., "A Reduced Complexity Wall ace Multiplier Reduction," IEEE Transactions on Computers, vol. 59, no. 8, Aug 20 10
[3] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. ElectronComput., vol. EC-13, no. I, pp. 14-17, Feb. 1964
[4] Shanthala S, Cyril Prasanna Raj, Dr.S.Y.Kulkarni, "Designand VLST Implementation of Pipelined Multiply Accumulate Unit," IEEE International Conference on Emerging Trends in Engineering and Technology, ICETET-09
[5] B.Ramkumar, Harish M Kittur and P.Mahesh Kannan, "ASIC Implementation of Modified Faster Carry Save Adder ", European Journal of Scientific Research, Vol. 42, Issue 1, 2010.
[6] WJ. Townsend, E.E. Swartzlander Jr., and J.A. Abraham, "A Comparison of Dadda and Wall ace Multiplier Delays," Proc. SPIE, Advanced Signal Processing Algorithms, Architectures, and Implementations XIII, pp. 552-560, 2003
[7] Fabrizio Lamberti and Nikos Andrikos, " Reducing the Computation Time in (Short Bit-Width) Two's Complement Multipliers", IEEE transactions on computers, Vol. 60, NO. 2, FEBRUARY 20 1 1
[8] Y. Kim and L.-S. Kim, "64-bit carry-select adder with reduced area," Electron. Lett., vol. 37, no. 10, pp. 614–615, May 2001.
[9] J. M. Rabaey, Digtal Integrated Circuits—A Design Perspective. Upper Saddle River, NJ: Prentice-Hall, 2001.
[10] Y. He, C. H. Chang, and J. Gu, "An area efficient 64-bit square root carry-select adder for lowpower applications," in Proc. IEEE Int. Symp.Circuits Syst., 2005, vol. 4, pp. 4082–4085.
[11] Cadence, "Encounter user guide," Version 6.2.4, March 2008.
[12] P.Ramanathan, P.T.Vanathi, "Novel Power Delay Optimized 32-bitParallel Prefix Adder for High Speed Computing", International Journal of Recent Trends in Engineering, Vol 2, No. 6, November 2009.
[13] R. Zimmermann, Binary Adder Architectures for Cell-Based VLSI and their Synthesis, ETH Dissertation 12480, Swiss Federal Institute of Technology, 1997.
[14] David Harris, "A Taxonomy of parallel prefix networks," Proceedings of the 37th Asilomar Conference on Signals, Systems and Computers Pacific Grove, California, pp.2213-2217, November 2003.