

Efficient FIR Filter Design Using Modified Carry Select Adder & Wallace Tree Multiplier

¹Gowrishankar V, ²Manoranjitham D, ³Jagadeesh P

Abstract –An area-power-delay efficient design of FIR filter is described in this paper. In proposed multiplier unit high speed is achieved using XOR-XNOR column by column reduction compressors instead of compressors using full adder. The carry propagation delay and area of carry select adder is reduced by splitting carry select adder into equal bit groups. The proposed carry select adder unit consumes less power than conventional carry select adder unit. With proposed multiplier unit and carry select adder unit, the designed FIR Filter consumes 55% power less than the conventional filter without significant increase in area. The power & delay comparison is performed for both existing and proposed method of FIR filter. The design is implemented using 0.18µm technology.

Index Terms - FIR filter design, FDA Tool, Low Power VLSI, Wallace tree multiplication.

I. INTRODUCTION

Advancement in mobile computing and multimedia applications insists for highly efficient VLSI Digital Signal Processing (DSP) systems. Finite impulse response (FIR) filtering is one of the most widely used operations in DSP. The several proposed technique achieves high performance with reduced area & power. Finite Impulse Response (FIR) filter has been designed by considering the power consumption in multiplier unit and adder unit.

The main building block in FIR filter is Multiplier-Accumulator (MAC) unit. In MAC unit full adder mainly affect the efficiency of system. Full Adder circuit power reduction is necessary for low power application.

Manuscript received March 6, 2013.

Gowrishankar V, Asst. Professor, Department of ECE, Velalar College of Engineering & Technology, Erode, India, mobile no: 9965261106,

Manoranjitham D, P.G scholar (M.E-VLSI Design), VCET, Erode, India,

The complete VLSI implementation of the FIR filter is presented in this paper and compares its delay and power performance of both existing and proposed 4-tap and 8-tap filter.

Most of the power is consumed in multiplier unit, so modified Wallace tree multiplier using compressor is proposed. When switching activity of the multiplier is reduced power consumption would reduced statistically.

The following parts of the paper is organized as follows, Section 2 gives the Mathematical concepts of FIR filter, Section 3 describes the designing of modified Wallace tree multiplier which is used in filter design, Section 4 explains the design of adder, Section 5 describes usage of FDA tool for Generating Filter coefficients, Section 6 presents the structural description of the filter. The simulated waveforms and comparison results are discussed in section 7. In Sections 8 the paper is concluded.

II. MATHEMATICAL CONCEPTS USED IN DESIGNING FIR FILTER

The following convolution shows the performance of N-tap FIR filters:

$$Y_j = \sum_{k=0}^{N-1} C_k X_{j-k} \dots\dots (1)$$

Where

c_k is the coefficient of the filter

X_j and Y_j are the j th terms of the input and output sequences, respectively.

The Z- transform of Eq (1) is given below:

$$Y(Z) = H(Z) X(Z) \dots\dots (2)$$

Where $Y(z)$ and $X(z)$ are the z-transforms of the output and input filters respectively and $H(z)$ is the transfer function.

Jagadeesh P, P.G scholar (M.E-VLSI Design), VCET, Erode, India,

By multiplying and dividing the transfer function $H(z)$ the frequency response is not changed.

For example, the z-transfer of the first order low pass FIR filter is given by ($\alpha = -1, \beta = 1, m = 1$) the equation is as follows:

$$Y(Z) = \frac{[1-z^{-1}]H(Z)}{[1-z^{-1}]} X(Z) \dots\dots\dots (4)$$

$$Y(Z)Y(Z)Z^{-1} = H(Z)X(Z) - H(Z)X(Z)Z^{-1} \dots\dots (5)$$

According to Eq (1.0) and (5.0) the transformed filter can be expressed as

$$Y_j - Y_{j-1} = \sum_{K=0}^{N-1} C_K X_{j-K} - \sum_{K=0}^{N-1} C_K X_{j-K-1} \dots\dots\dots (6)$$

Re-arranging the Eq (6.0) we can obtain the following equation for first order ($m=1$) differential coefficients:

$$Y_j = C_0 X_{j-K} + \sum_{K=1}^{N-1} (C_K - C_{K-1}) X_{j-K} - C_{N-1} X_{j-N} + Y_{j-1} \dots\dots\dots (7)$$

For first order differential coefficients, differences between adjacent coefficients are used to obtain the filter outputs are clearly shown in Eq (7.0) (except for first and last coefficients) and the previous filter output. In order to realize the $(-CN-1Xj-N)$ the transformed filter requires one additional multiplication and subtraction operation

III. MULTIPLIER DESIGN

A proposed 8x8 architecture for modified Wallace tree multiplier is based on Wallace Tree logic based on multi operand adders, which is efficient in terms of delay and regularity without significant increase in power and area. Here, AND gates are used to generate the 64 different partial products terms.

The addition of these partial products is done by modified Wallace tree logic which uses full adder. AND gate is used for realizing the parallelism of generating partial products of the first bit (LSB) of the multiplier with the multiplicand bits. ANDing the second multiplier bit with the multiplicand bits the second partial product row is generated preceded by a single zero. ANDing the third multiplier bit with the multiplicand bits the third partial product row is

obtained preceded by two zeros and this process is repeated.

Fig.1 shows 8x8 proposed Wallace tree logic with the hierarchical decomposition. 8 partial products are generated for (8X8) bits and are added in parallel as shown in stage A of Fig.1. The generated 8 partial products are divided into two groups, where each group contains four adjacent rows of partial products. The same parts of the four adjacent partial products are subdivided to two columns and each of 4 bits is represented by dotted lines. Two columns from each adjacent four rows we have four parallel blocks, this blocks working in parallel.

A. DESIGN OF PROPOSED MULTIPLIER

In proposed multiplier, each block addition operation in the column is performed using combination of half adders, full adders, 4:2 compressors and 5:2 compressors instead of full adders and they are selected appropriately based on the number of bits to be added.

The first level (Stage A) computation is represented in the fig 1. Thus the generated partial sums are correctly divided and added again in the same manner which forms the second level (stage B) of computation. In the third level (stage C) the final product will arrive only after generated partial sums in the second level are added.

The speed has been improved when compared with the conventional design without significantly increasing the amount of power in proposed Multiplier tree.

Fig.2 shows the proposed modified Wallace tree multiplier block diagram. The proposed architecture differs from the conventional Wallace tree logic represented by gray boxes. 16x16 bit multiplier can be designed similarly by the same method.

The proposed Wallace tree multiplier is designed using Tanner EDA tool and simulation of the multiplier design is done. The Waveform obtained is correctly checked by giving different input bits combinations. The power & delay for both the conventional & proposed Wallace tree multipliers are compared.

A comparison of power, delay & number of MOSFET counts used in proposed and conventional method of 8x8 bits multiplier is given in table1. It is clear from the table that a significant power is

reduced in a proposed design. 42.5% power can be saved in a proposed design as compared to conventional design without significant increase in area with a low power technique in 8x8 bit multiplier.

B. PROPOSED MULTIPLIER BLOCK DIAGRAM

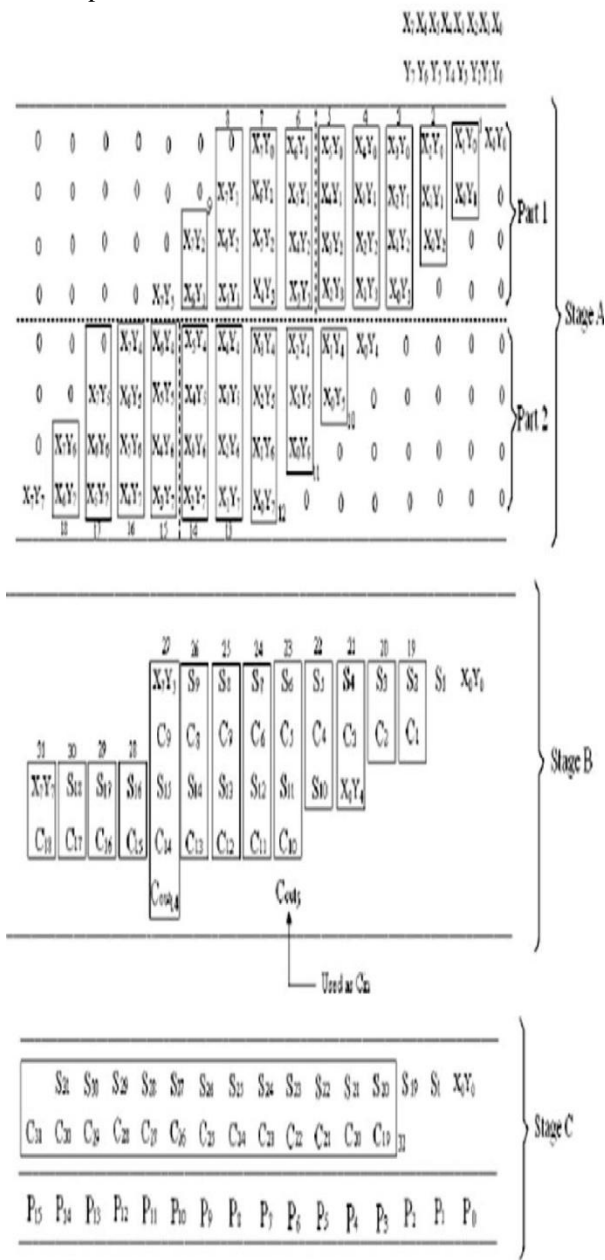


Fig 1: Hierarchical decomposition Proposed Wallace tree logic using compressors

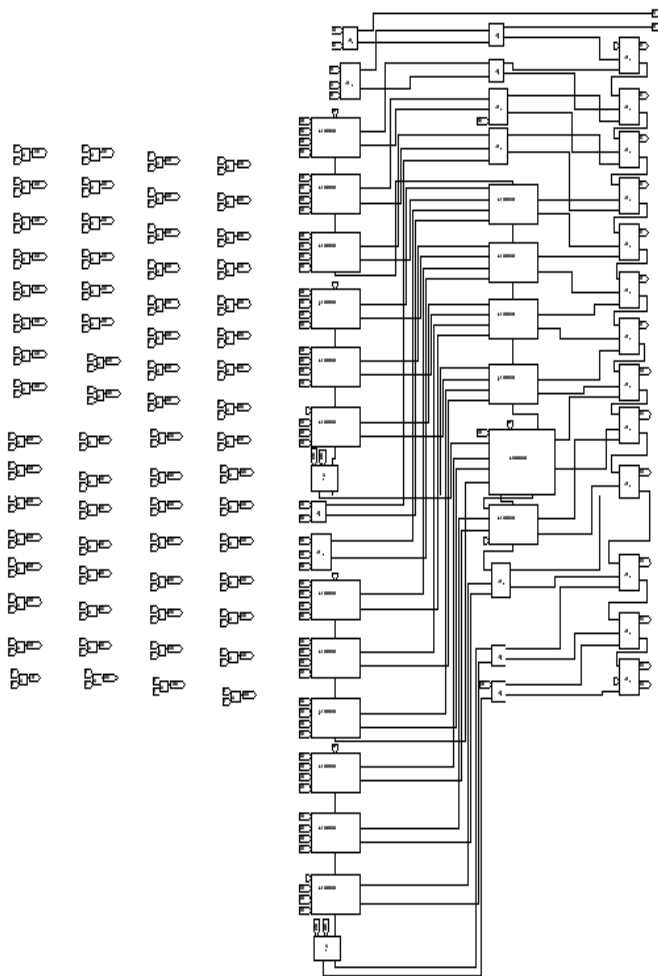


Fig 2: Block Diagram of the Proposed Modified Wallace Tree Multiplier

TABLE I POWER & DELAY COMPARISON OF 8X8 BIT MULTIPLIERS

MULTIPLIER		MOSFET COUNT	DELAY (ns)	POWER (mW)
8x8	Existing	1824	72.5ns	5.77935
	proposed	2652	61.2ns	3.31152

IV. ADDER DESIGN

The addition of numbers is performed by a digital circuit named adders. In arithmetic and logical unit (ALU) other operations like subtraction, multiplication, division and other logical operations are performed. Most commonly adders are constructed for numerical representations like Binary coded decimal or excess-1. Adders operate on binary numbers such as half adder, Full adder and the compressors like 4:2 Compressors and 5:2 Compressors.

A. EXISTING CARRY SELECT ADDER

Parallel computation is the basic operation of Carry Select Adder (CSLA). Many carries and partial sums are generated by CSLA. Multiplexers (mux) select the final sum and carry. In CSLA structure multiple pairs of Ripple Carry Adders (RCA) are used.

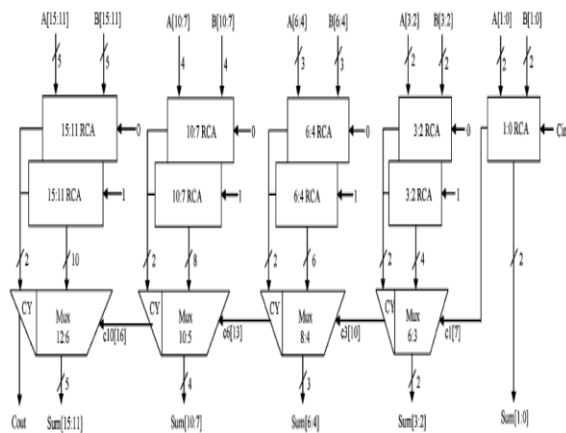


Fig 3: Existing 16-bit Carry Select Adder

The existing method CSLA structure consists of pair of Ripple Carry Adders (RCA). Initially carry $C_{in}=0$ is used in one RCA and carry $C_{in}=1$ is used in another RCA. Using multiple pairs of RCA complexity of the circuit gets increased. The main disadvantage of RCA is its propagation delay. Therefore it is necessary to reduce area, power & delays of existing CSLA structure.

B. PROPOSED 16-BIT CARRY SELECT ADDER

i) BEC Logic

The proposed carry select adder reduces area and power consumed. The delay time of proposed incrementing carry select adder is less when compared with the existing carry select adder. Two Ripple Carry Adders (RCA) used in regular CSLA structure is reduced to single RCA.

BEC logic is used in proposed method. One of RCA use initial carry as $C_{in}=0$ and another RCA works with carry $C_{in}=1$. Instead of RCA with $C_{in}=1$ BEC is used in order to reduce and power consumption of the regular CSLA. An $n+1$ bit BEC is required to replace the n -bit RCA. The 4-bit BEC structure of is shown in Fig. 4.

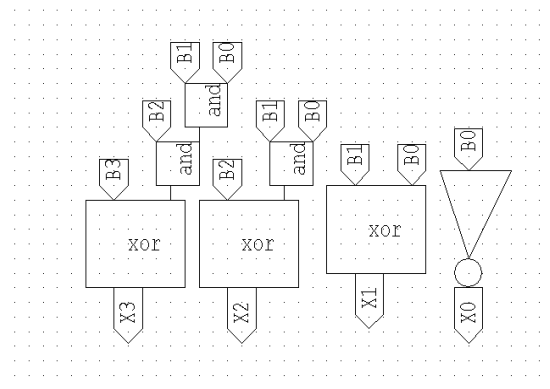


Fig 4: 4-bit Binary to Excess-1 Code Converter

The basic function of CSLA is shown in Fig. 5 consists of 4-bit BEC and an 8:4 multiplexer. Multiplexer receives one input as direct input (B3, B2, B1 and B0) and BEC output as another input of the mux. Thus two possible partial results are produced in parallel and either the BEC output or the direct inputs is selected by mux according to the control signal C_{in} .

The 4-bit BEC Boolean expressions of are shown below (note the functional symbols ~ NOT, & AND, ^ XOR)

$$X0 = \sim B0$$

$$X1 = B0 \wedge B1$$

$$X2 = B2 \wedge (B0 \& B1)$$

$$X3 = B3 \wedge (B0 \& B1 \& B2)$$

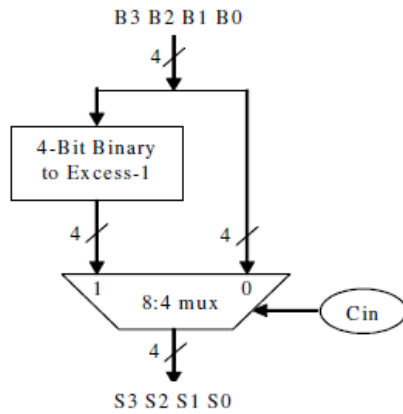


Fig 5: 4-bit Binary to Excess-1 logic with 8:4 multiplexer

ii) Proposed Structure

Multiple pairs of Ripple Carry Adders (RCA) are used in existing CSLA structure. Hence, the CSLA is not area efficient. In this paper, we proposed a new CSLA architecture.

The proposed carry select adder equally divides the word size of the adder into blocks of 4-bit each. This reduces the delay time of incrementing carry select adder when compared with the existing carry select adder. The least 4-bits are added using conventional RCA, while other blocks are added in parallel along with the given incremental block named Binary to excess-1 converter (BEC).

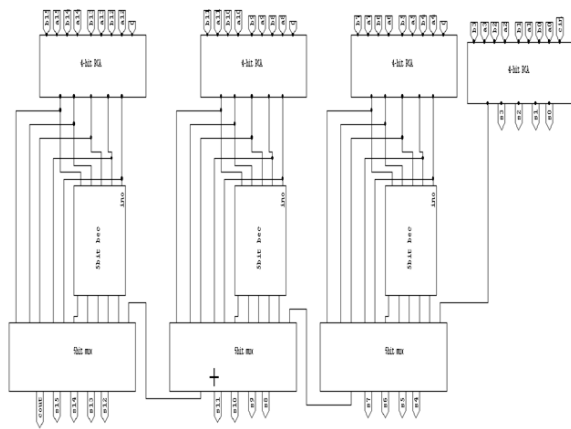


Fig 6 Proposed 16-bit Carry Select Adder

Once all the interim sums and carries are calculated, the final sums are computed using multiplexers having minimal delay. The multiplexer block receives the two sets of 5-bit input (four sum bits and one carry bit each) and selects the final sum based on the select input from the previous stage. By splitting the 16-bit into equal bits achieve fast incrementing action with reduced device count. Thus the proposed carry select adder excels the existing carry select adder circuit in terms of speed by reducing the carry propagation latency.

The delay of the proposed structure will be increased for some input bit combinations. When we go for higher order input adder structure the delay values will be reduced to a considerable amount.

TABLE 2 POWER & DELAY COMPARISON OF 17 BIT ADDERS

ADDER		MOSFET COUNT	DELA Y (ns)	POWER (mW)
16 Bit	Regular	988	12.75	8.06
	Proposed	798	14.57	5.83

V. FILTER COEFFICIENT GENERATION USING FDA TOOL.

The order of the filter determines the number of coefficients generated for designing FIR filter. There will be N+1 coefficient terms in the filter if the order of the filter is N.

The impulse response of the filter is represented by coefficient terms. The mat lab tool named Filter Design and Analysis (FDA) tool is used in this paper for determining filter coefficient.

For quickly analyzing and designing filters powerful Filter Design and Analysis Tool (FDA Tool) is used. By setting filter specifications, FDA Tool enables you to design digital FIR or IIR filters by importing filters from your MATLAB workspace, or by adding, moving or deleting poles and zeros.

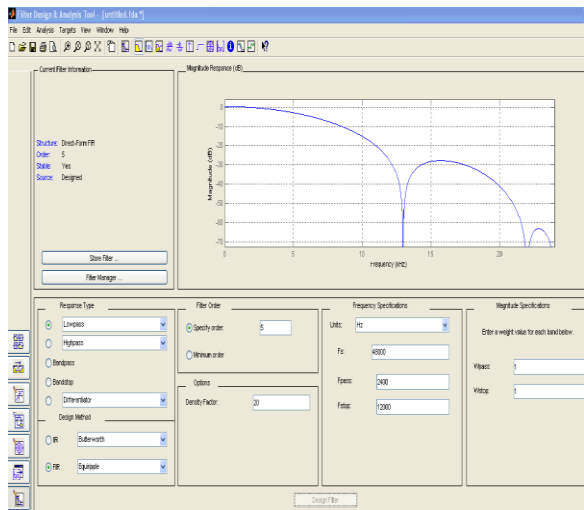


Fig 7: Frequency response of an order 5 FIR filter

FDA Tool consists of tools for analyzing filters, such as phase and magnitude response and pole-zero plots. The frequency response of order 5, FIR equiripple filter with pass band and stop band frequencies of 2400Hz and 12000Hz at a sampling rate 48 Hz is shown in Fig 7.

VI. STRUCTURAL IMPLEMENTATION OF FIR FILTER

Word-level multipliers and adders are used for sequence multiplication and accumulation, treats the FIR as a sequence includes both the direct form and transposed form.

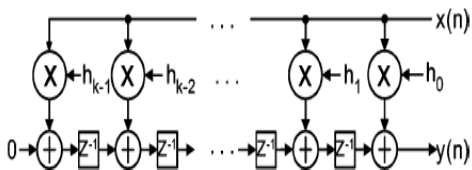


Fig 8: FIR filter structure (Transposed Form)

It is generally believed that DF implementation results in lower area and lower power dissipation while transposed form (TF) offers higher speed but requires larger area.

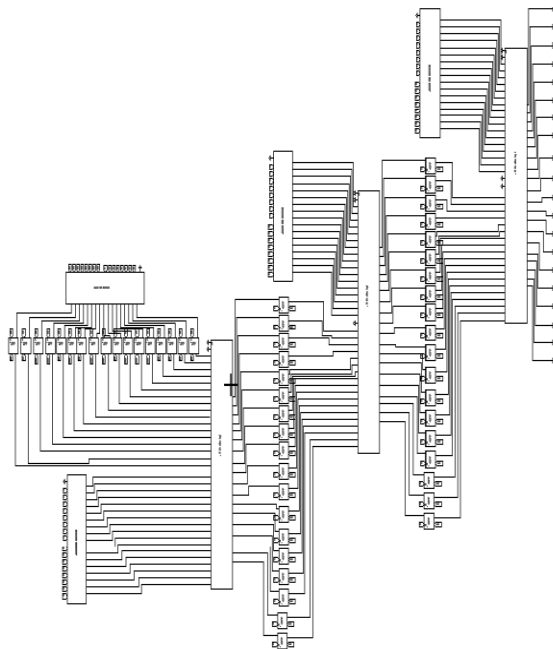


Fig 9: 4-Tap FIR structure (Transposed Form)

The direct form FIR filter needs extra pipeline registers between the adders to reduce the delay of the adder tree and to achieve high throughput.

The FIR filter with transposed structure has registers between the adders and can achieve high throughput without adding any extra pipeline registers. As shown in fig.8 the critical path consists of a multiply and add operation in each stage. The carry select format of the multiply and add output at each stage is converted to regular binary format.

This nearly halves the numbers required for storing intermediate results in contrast to storing carry save outputs. Further it reduces the latency of filter since no final carry propagation is needed beyond the last stage. This increases the area and power dissipation in registers and therefore, reduction of registers are highly desirable.

VII. RESULT ANALYSIS

A.SIMULATED WAVEFORMS

The following waveforms are simulated using Tanner spice of version-7. This is simulated with the model file cmos180n.

1) Modified Wallace Tree Multiplier

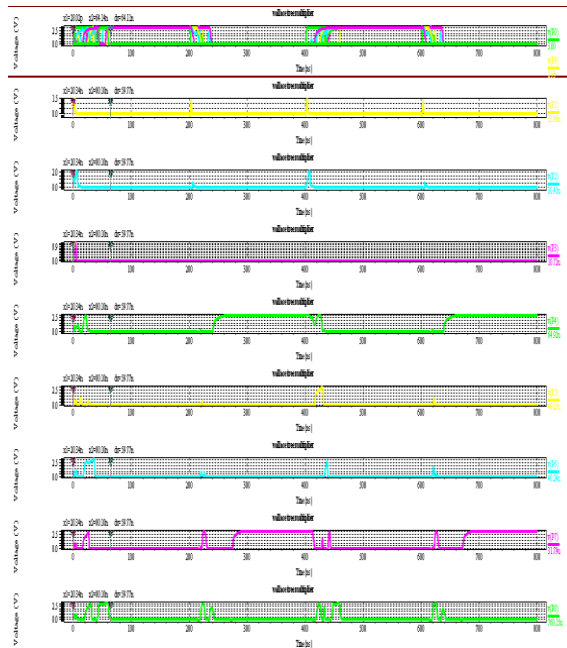


Fig 10: Output Waveform Of Wallace Tree Multiplier In T-SPICE

2) Proposed Carry Select Adder

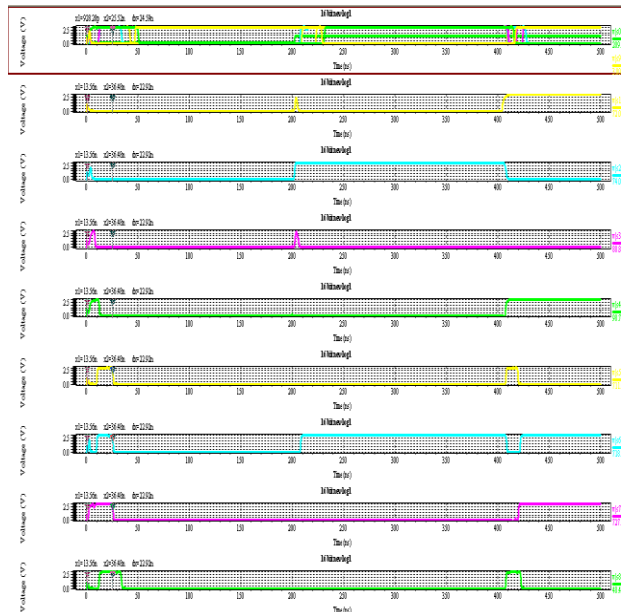


Fig 11: Output Waveform of Proposed Carry Select Adder in T-SPICE

3) 4-Tap FIR Filter

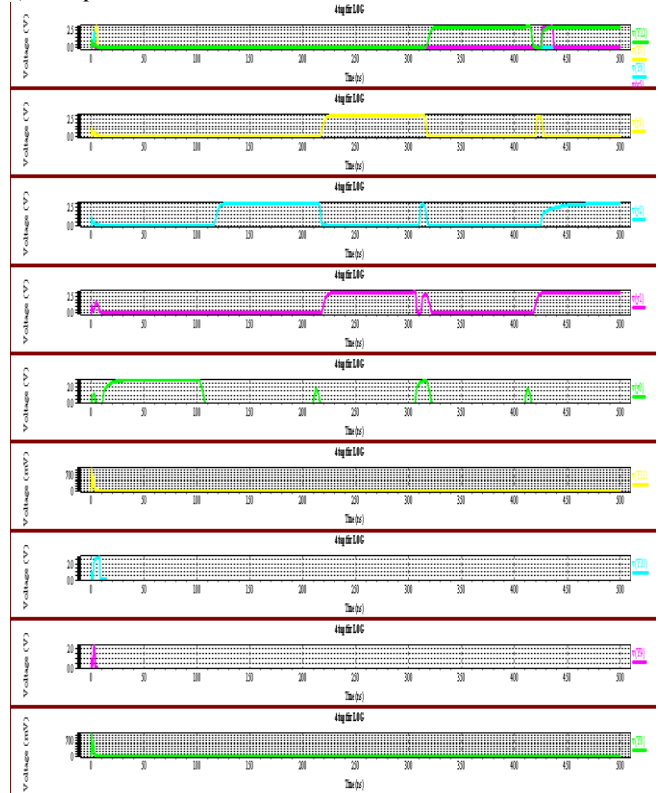


Fig 12: Output Waveform of 4-tap FIR in T-SPICE

VIII CONCLUSION
TABLE 3 AREA- POWER- DELAY
COMPARISONS FIR FILTER

FILTER TAP		MOSFET COUNT	POWER (mW)	DELAY (ns)
4	Existing	17764	148.853	28.23
	Proposed	17052	137.143	34.2
8	Existing	35478	288.94	36.5
	Proposed	23640	244.24	38.8

The comparison charts shows the power, delay and MOSFET count values of both the existing and proposed FIR filter structures. The delay of FIR filter gets increased for lower order tap and reduced statically when go for higher order tap of FIR filter.

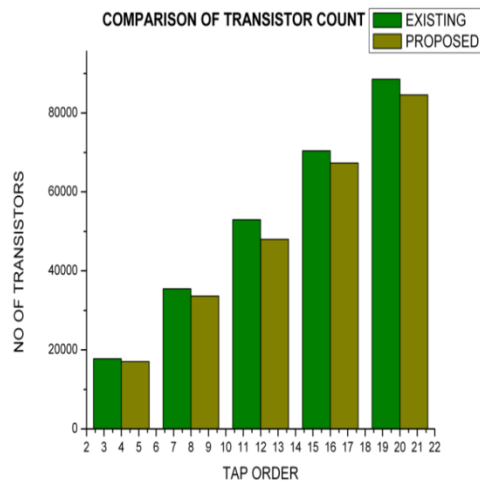


Fig 13: Comparison chart for transistor count

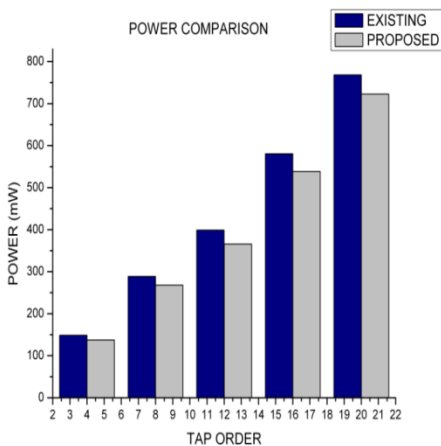


Fig 14: Comparison chart for power

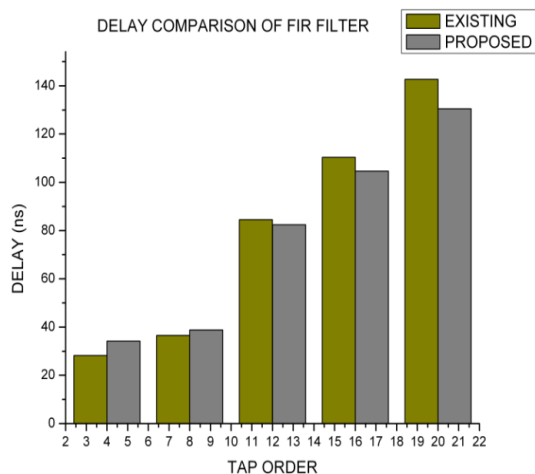


Fig 15: Comparison chart for delay

In this paper the FIR filter is designed to increase the speed of addition and decrease the power taken by the multiplier unit. It has been easily concluded that proposed FIR filter design has consumed less power than the conventional design. The power delay comparison of both Wallace tree multiplier and carry select adders were shown. The compared results prove that proposed carry select adder with binary to excess-1 converter performs faster than conventional carry select adder.

ACKNOWLEDGMENT

The authors acknowledge the contributions of the students, faculty of Velalar College of Engineering and Technology for helping in the design of test circuitry, and for tool support. The authors also thank the anonymous reviewers for their thoughtful comments that helped to improve this paper. The authors would like to thank the anonymous reviewers for their constructive critique from which this paper greatly benefited.

REFERENCES

- [1] B. Ramkumar and Harish M Kittur, "Low Power and Area Efficient Carry Select Adder", IEEE Transaction on Very Large Scale Integration Systems, February 2012.
- [2] A.M. Vijaya Prakash, K.S. Gurumurthy, "A Novel VLSI Architecture for Low Power FIR Filter", Published in International Journal of Advanced Engineering & Application, Jan 2011.
- [3] B. RamKumar, Harish M Kittur and P. Mahesh Kannan, "ASIC Implementation of Modified Carry Save Adder", European Journal of Scientific Research, ISSN 1450-216X Vol.42 no.2, pp.53-58,2010.
- [4] A Low Power, Low Area Multiplier based Shift –and-add Architecture M Mottadhi, A. Afzalikusha and M Pendram. IEEE Transaction on VLSI vol-17 no-2 Feb 2009
- [5] Himanshu Thapliyal, Neela Gopi, K.K Pavan Kumar and M.B. Srinivas, "Low Power Hierarchical Multiplier Carry Look Ahead Architecture", IEEE Transaction on Center for VLSI ,2006.

- [6] 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.
- [7] 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.
- [8] J.M. Rabaey, *Digital Integrated Circuits—A Design Perspective*. Upper Saddle River, NJ: Prentice-Hall, 2001.
- [9] Oklobdzija. V. G, “High-Speed VLSI Arithmetic Units: Adders and Multipliers”, in “Design of High-Performance Microprocessor Circuits”, Book edited by A.Chandrakasan, IEEE Press, 2000.
- [10] Pallavi Devi Gopineedi, Himanshu Thapliyal, Srinivas. M. B and Arabnia1 Hamid, “ Novel and Efficient 4:2 and 5:2 Compressors with Minimum number of Transistors Designed for Low Power Operations”, Center for VLSI and Embedded System Technologies, Hyderabad, India, The University of Georgia, Department of Computer Science.

Gowrishankar V doing Ph.D. and received M.E. degree in VLSI Design from Kongu Engineering College, in 2010. He is currently working as an Assistant Professor in Velalar College of Engineering and Technology, Erode, India. His research includes VLSI architecture for Signal Processing. He is a member of IEEE.

Manoranjitham D received the B.E. degree in Electronics and Communication Engineering from King College of Technology under Anna University of Technology, Tamil Nadu, India, in 2011. She is currently pursuing Master of Engineering (M.E) in Velalar College of Engineering and Technology, Erode, Tamilnadu, India. Her area of interest is Low Power Design.

Jagadeesh P received the B.E. degree in Electronics and Communication Engineering from Park College of Engineering & Technology under Anna University, Tamil Nadu, India, in 2010. He is currently pursuing Master of Engineering (M.E) in Velalar College of Engineering and Technology, Erode, Tamilnadu, India. His areas of interest are Low Power Design and Digital signal processing.