

# Design of High Speed Pipelined MAC Unit Using 4:2 Compressor for Signal Processing Applications

S.Maragatharaj<sup>1</sup>, Dr.K.N.Vijeyakumar<sup>2</sup>, Hima Bindu.S<sup>3</sup>, Krithika.N<sup>4</sup>, Madhu Preethi.S<sup>5</sup>

Assistant Professor, Department of ECE, Dhirajlal Gandhi College of Technology, Salem, Tamilnadu, India<sup>1</sup>

Associate Professor, Department of ECE, Dr. Mahalingam College of Engineering & Technology, Pollachi, India<sup>2</sup>

UG Scholars, Department of ECE, Dr. Mahalingam College of Engineering & Technology, Pollachi, India<sup>3,4,5</sup>

**ABSTRACT:** Modern portable electronic gadgets depend on computation intensive digital signal processing (DSP) algorithms for their operations that put onus on design of efficient hardware architectures. Multiply Accumulate (MAC) units form the integral part of many DSP modules that consume more power, making it an important candidate for optimization to achieve efficient VLSI technologies. This work proposes an modified 4:2 compressor adder module that achieves low power and area than its conventional counterpart. The proposed compressor based adder module is implemented in partial product reduction tree of the MAC unit to achieve reduction in power dissipation and area. The proposed efficient MAC module is developed using Verilog HDL, synthesized with 90nm standard CMOS library using cadence SOC encounter tool, and parameters such as: Power, Area and Delay are extracted. Efficacy of the proposed MAC module is confirmed by implementing the proposed MAC unit in a FIR filter using System generator tool of Xilinx. The proposed MAC unit achieves 19% reduction in area and 32% reduction in delay when compared with conventional MAC unit.

**KEYWORDS:** Compressor adder, Wallace tree multiplier, Power optimization, FIR filter.

## I. INTRODUCTION

Modern DSP algorithms used in portable applications demand high performance VLSI data path systems with low area and power. Multiply and accumulate units (MAC) are the most computation intensive component of many DSP operations such as FFT, convolution, filters etc. This puts onus on design of efficient MAC architectures [1] to achieve optimization of DSP modules. MAC unit consists of a 'n'-bit multiplier, which multiplies a set of 'n' bit input operands repeatedly, an adder of width '2n' to add consecutive products and an accumulator of width '2n' to accumulate the result in register [2]. Multipliers are the most time consuming, energy intensive module of a MAC unit which are built using adders. Hence, adder optimization leads to efficient multipliers which in turn results in optimization of MAC unit. Various adder topologies [3] are designed to achieve efficiency in terms of area and delay. Adders such as Ripple carry adder (RCA), Carry select adder (CSLA), Carry save adder (CSA) use traditional addition algorithm to generate sum and carry signal. For an adder of width 'm' bits, RCA computes carry out signal by rippling carry signal of each module from "carry in" to "carry out" with a delay of O(m). CSLA and CSA compute sum similar to RCA, but differs in calculation of carry out signal and reduce critical delay path to O(m<sup>(1+2)/(1+1)</sup>) and O(log(m)) respectively. Carry look ahead adder (CLA) reduce critical delay path to O(log(m)), by computing carry out signal independent of (m-1)<sup>th</sup> carry bit. Compressor based adders are alternate means for implementing multi-bit adders. A traditional 4:2 compressor adder [4] consists of 2 full adder modules and accepts 4 inputs of equal bit weight to produce two output signals. If t<sub>XOR</sub>, t<sub>CY</sub> indicates the delay of XOR gate and carry generator logic then delay of 4:2 compressor can be expressed as: (i\*t<sub>XOR</sub> + j\*t<sub>CY</sub>) where, 'i' is the number of XOR gates and 'j' is the number of carry generation logic stages.

Various multiplier architectures such as array multipliers, booth multiplier, Wallace tree multiplier, dada multiplier etc. [5] use adders for their partial product reduction tree (PP) and PP accumulation. Array multiplier has regular structure, but has the highest delay. Other multiplication algorithms such as booth, Wallace and dada multiplier aim to reduce number of adder stages in PP reduction tree to optimize area and power consumption. Hence, various adder topologies are constituted in multiplier architecture [5] to achieve optimal performance in terms of area and power. Literature for enhanced MAC units through optimized adder and multiplier architectures exist.

Various multipliers such as Array multiplier, Modified booth multiplier, Wallace tree multiplier and Dadda multiplier used to realize MAC architecture in [6]. CSA adders are used in PP reduction tree of MAC unit multipliers in [7]. It ascertained in [6, 7] that Wallace tree multiplier implemented using CSA adder achieves high speed and low power performance and is an ideal candidate for efficient MAC unit. Therefore, wallace tree multiplier is a prime candidate for enhancing efficiency of MAC unit due to less number of stages in its PP reduction tree. Prior work in [11] has proposed a modified wallace tree multiplier that achieves lesser delay and explored the efficacy of compressor adder in PP reduction tree. Counter based wallace multiplier implemented in [12] replaces traditional adder topologies with compressor adders to enhance efficacy of MAC unit. High speed MAC unit design using compressor adders have been explored in [8, 9 & 10]. 3:2 compressor based parallel multiplier is implemented in MAC unit [8] for audio applications. 5:3 compressors for PP reduction in multipliers is proposed in [9] to implement word length MAC unit. [10] propose a merged MAC unit that eliminates the use of separate accumulation unit which is built using traditional 4:3 compressor adder. In this brief, we propose a modified efficient 4:2 compressor to realize high speed MAC unit. Rest of the paper is organized as follows: Section 2 gives an overview of traditional MAC unit. Design of proposed compressor and its subsequent implementation in MAC unit is presented in Section 3. Section 4 discusses the synthesis results of MAC unit and presents the implementation of MAC unit for speech processing applications. Finally, Section 5 concludes the paper.

## II. OVERVIEW OF MAC UNIT

MAC unit is a vital Processing Element (PE) used in many DSP applications to compute running sum of product. It is accomplished by repetitive process of multiplication, addition and accumulation. General expression of a 'n' bit MAC unit is given as:

$$Z = (A * B) + Z \tag{1}$$

where, A and B are two input operands of width 'n' bits each and, Z is the addend whose width is (2n + 1). Block diagram of a basic MAC unit is given in Fig. 1, where a multiplier is implemented in combinational logic followed by an adder and an accumulator register. The multiplier accepts two inputs of width 'n' bits and compute product of width '2n'. Output of the multiplier is fed to a RCA of width '2n' and produce results '2n+1'. Accumulator register is used to accumulate output of the adder on each clock cycle, and save it to produce sum of product.

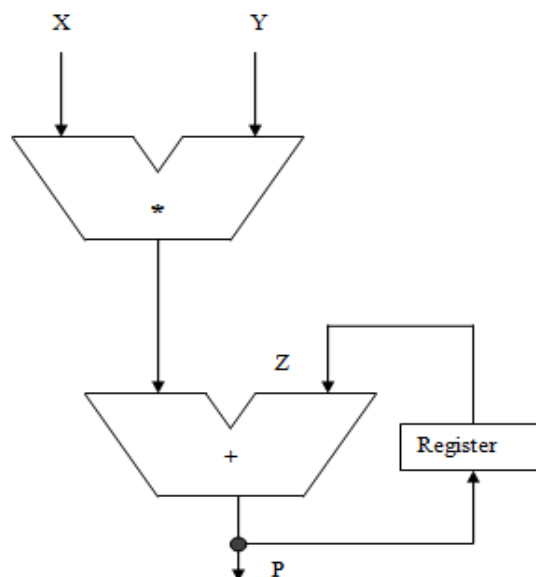


Fig. 1. Block Diagram of MAC Unit

Since multiplication is the most energy intensive process of the three sub-modules, optimization of multiplier determines efficiency of MAC unit.

### III. PROPOSED MAC UNIT USING COMPRESSOR ADDER

The proposed MAC architecture using compressor shown in the above fig 4 reduces the number of stages. In the proposed architecture partial product reduction is accomplished by the use of 4-2 compressor structures and final stage of addition is performed by the carry save adder.

#### Design of Exact 4:2 Compressor

Exact compressor is used to add  $n$  1-bit inputs and produce ' $i$ ' bits result, such that ' $n$ ' is the least integer that satisfy the condition  $n < 2^i$ . A standard 4-2 compressor has four inputs  $X_1, X_2, X_3, X_4$  and carry input  $C_{in}$  from preceding block, and produces three outputs viz., Sum, Carry,  $C_{out}$ . Block level architecture of standard 4:2 compressor is shown in Fig 2 and corresponding Boolean expressions defining the logic are shown in (2)-(4). Following are the functional symbols used to express logic functions viz., | - OR; & - AND; ^ - XOR.

$$S = X_1 \wedge X_2 \wedge X_3 \wedge X_4 \wedge C_{in} \tag{2}$$

$$C = [(X_1 \wedge X_2) \& X_3] \vee [(X_1 \wedge X_2)' \& X_1] \tag{3}$$

$$C_{out} = [(X_1 \wedge X_2 \wedge X_3 \wedge X_4) \& C_{in}] \vee [(X_1 \wedge X_2 \wedge X_3 \wedge X_4)' \& X_4] \tag{4}$$

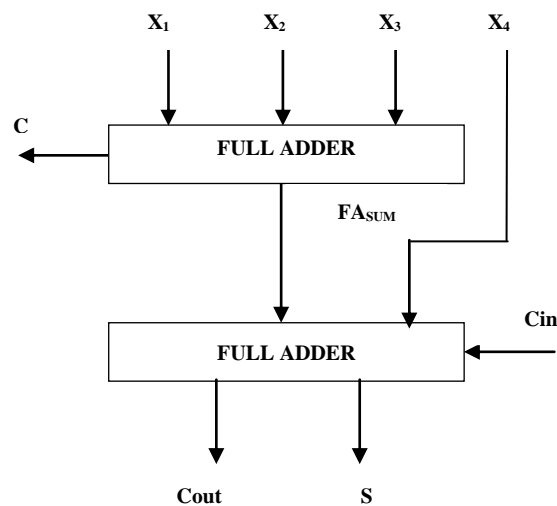


Fig. 2. Block Diagram of Exact 4:2 Compressor

However to improve regularity of arrangement of cells in multiplier modifications to logic expressions of standard 4:2 compressor are proposed with logic low in  $C_{in}$  and neglecting output  $C_{out}[1]$ . Modified 4:2 exact compressor has inputs  $X_1, X_2, X_3, X_4$  and the outputs Sum-S and Carry-C. Elimination of  $C_{out}$  in 4:2 compressor generate error for  $X_1X_2X_3X_4 = "1111"$ .

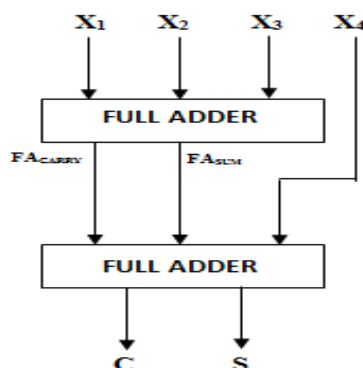


Fig. 3. Block Diagram of modified Exact 4:2 Compressor

Conversely, the approximation logic which produces logic high in *Sum* for  $X_1X_2X_3X_4 = "1111"$  reduces the Max-ED to -1. The block diagram of modified 4:2 exact compressor is shown in Figure 3 and the corresponding logic is shown in (5)-(6).

$$S = X_1 \wedge X_2 \wedge X_3 \wedge X_4 \tag{5}$$

$$C = ((X_1 \wedge X_2 \wedge X_3 \wedge X_4)' \& X_4) \tag{6}$$

However in the multiplier implementation discussed in section 3.2 we account for  $C_{out}$  elimination in the modified 4:2 exact compressor with error compensation bit  $E = X_1 \& X_2 \& X_3 \& X_4$ .

**Proposed MAC Architecture**

The MAC architecture was divided into two blocks one is multiplier and another one is accumulator. The multiplier multiplies the two inputs and the results are stored in the accumulator register.

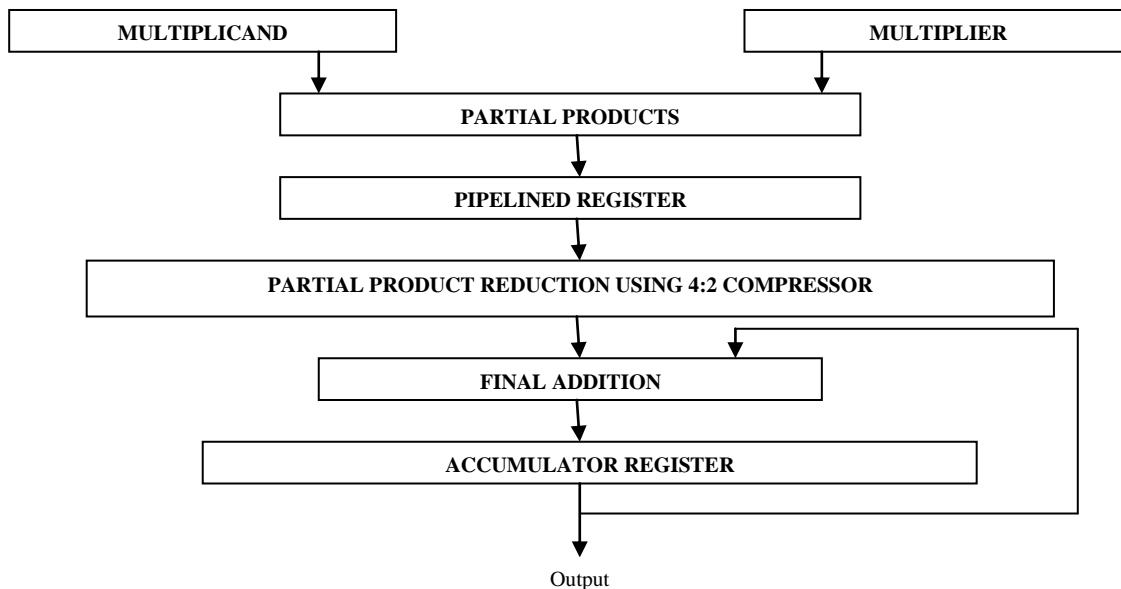


Fig 4. Block diagram of the proposed MAC architecture

In proposed MAC shown in Fig 4, 4:2 compressors are used for PP reduction and hence it reduces the number of PP compression stages. The MAC architecture’s critical path delay still depends on the partial product unit. The proposed architecture aims to reduce the overall latency. This leads to increased speed and reduced power consumption. The design makes use of compressors in place of full adders, and the final carry propagate stage is replaced by carry save adder. The Wallace tree basically multiplies two unsigned integers. The proposed Wallace tree multiplier architecture comprises of an AND array for computing the partial produces so obtained and carry save adder in the final stage of addition. In the proposed architecture partial product reduction is accomplished by the use of 4-2 compressor structures and final stage of addition is performed by the carry save adder. This multiplier architecture comprises of a partial product generation stage, partial product reduction stage and the final addition stage. The latency in the Wallace tree multiplier can be reduced by decreasing the number of adders in the partial product reduction stage. In the proposed architecture, multi bit compressor is used for realizing the reduction in the number of partial product addition stages. The combined factors of low power, low transistor count and minimum delay makes the 4-2 compressor. In this compressor, the outputs generated at each stage are efficiently used by replacing the XOR block with multiplier blocks. The select bits to the multiplexers are available much ahead of the inputs so that the critical path delay is minimized. The various adder structures in the conventional architecture are replaced by compressors.

Fig 5 shows the proposed Wallace tree multiplier partial product reduction. In fig 5, the reduction part uses half-adders, full-adders and 4-2 compressors; each partial product bit is represented by a dot. The Wallace tree multiplier is constructed by considering all the bits in each four rows at a time and compressing them in an appropriate manner. Thus, compressors form the essential requirement of high speed multipliers. The speed, area and power consumption of the multipliers will be in direct proportion to the efficiency of the compressors. In the first stage 4 half-

adders, 4 full adders and 10 compressors are utilized to reduce the partial products into at most four rows. In the second stage or final stage, 6 half-adder, 2 full-adder and 5 compressors are used to compute the two final rows of the partial products.

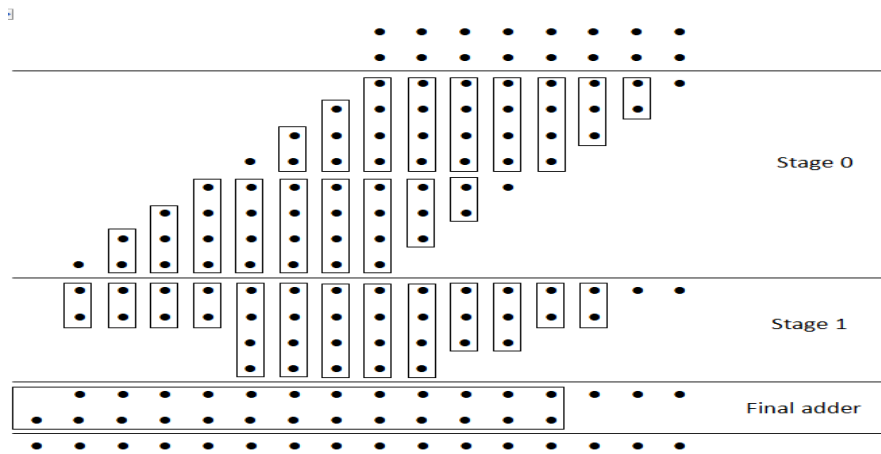


Fig. 5. Partial Product Reduction and accumulation

In the proposed architecture the outputs are utilized efficiently by using multiplexers at select stages in the circuit. Also additional inverter stages are eliminated. This in turn contributes to the reduced of delay, power consumption and gate count (area).The term accumulator or register refers to a flip-flop or latches which is used to store the one bit data. In our proposed work the accumulator register is used to store the output from the multiplier.

The accumulator register consists of D-flip-flop, basic gates that operated by clock synchronization. The register cell has 3 inputs and 1 output. The inputs are D, reset and clock and Q will be the output. The flip-flop will store the input value when clock is 1, if reset signal is 0, then the flip-flop will pass the value that is stored using tristate buffer to the output. In our proposed work the D flip-flop is implemented using a technique called DDFF, which is a hybrid flip-flop that uses sleepy stack technique for less power dissipation and low leakage currents which occurs during the storing of data in the register block. The block diagram is shown in the fig 7.

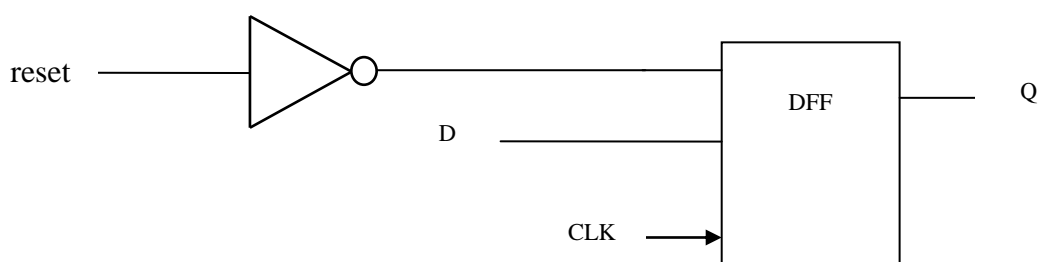


Fig. 6. 1-bit accumulator register

#### IV.RESULTS AND DISCUSSION

The proposed MAC and its RCA counterpart are designed using Verilog HDL, and simulated using ISM in Xilinx environment. Fig. 7 shows the simulation outputs of proposed compressor based MAC for n=8.

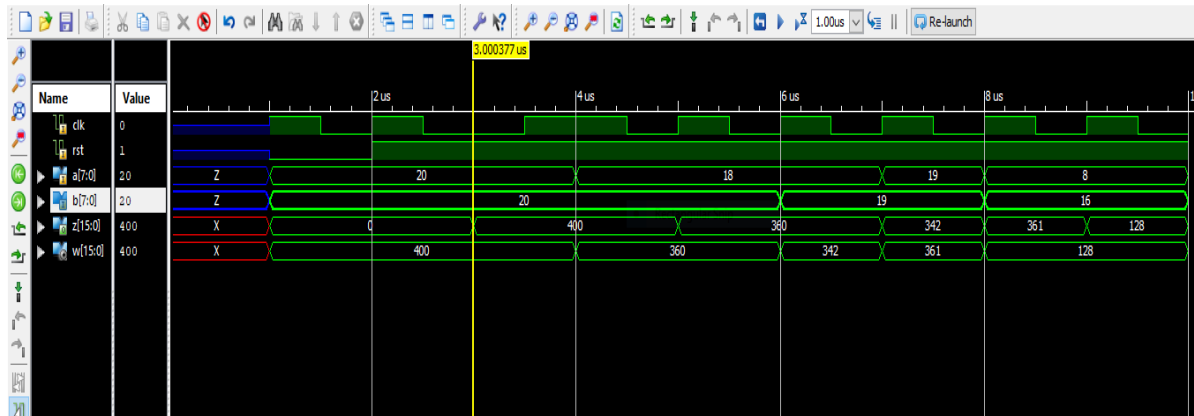


Fig. 3. Simulation result of MAC architecture

At posedge of each clock pulse, with active high on reset, the inputs of the MAC unit are multiplied and is added with the data stored in accumulator. The proposed compressor is employed for partial product reduction. The number of operands in partial product accumulation is reduced due to the usage of compressor unit. Hence, the compressors in proposed MAC unit demonstrate significant speed improvement compared to conventional RCA based MAC.

### Area, Power, Delay Comparison

The proposed MAC and priordesigns are designed structural Verilog HDL and synthesized using Cadence Encounter in 90nm ASIC PDKtechnology .Table 1 compares the area, power , delay of proposed compressor based MAC and conventional design.

Table 1 Comparison Between Conventional and Proposed VLSI Architecture

Designs	Area ( $\mu\text{m}^2$ )	Delay (ps)	Power (mw)	PDP ( $\text{X } 10^{-15}\text{J}$ )	EDP ( $\text{X } 10^{-24}\text{Js}$ )
Conventional MAC	5282	1508	317.75	479.17	722.58
Proposed MAC	4294	1026	263.35	270.19	277.22

Note from Table 1, area of proposed MAC demonstrate 18.7% area reduction and 32% delay reduction compared to conventional MAC. Furthermore proposed MAC exhibit 17.1% power reduction compared to prior MAC unit. Equally, in terms of power-delay product (PDP) and Energy-delay product (EDP) proposed MAC show 43.6% and 61.6% reductions respectively compared to conventional MAC.

### V. FIR FILTER IMPLEMENTATION RESULTS

In order to verify the functionality and driving capability efficacy of proposed MAC, it is implemented in a 27 tap FIR filter [13, 14]. Filter coefficients are selected using REMEEZ command in MATLAB. Speech signal corrupted with uniformly distributed random noise is used as input to the filter. Input signal is fed from MATLAB environment to HDL model of FIR filter in Xilinx environment. Synchronization between MATLAB and Xilinx environment is established using System Generator tool Fig 8 shows the simulink model of the FIR filter .Fig. 9 shows the processed output signals of FIR system

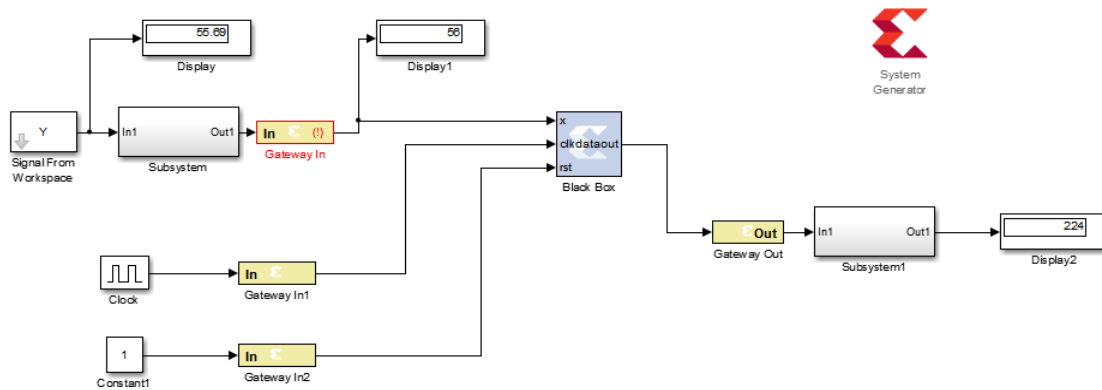


Fig. 8. Simulink model of FIR Filter in System Generator (MATLAB)

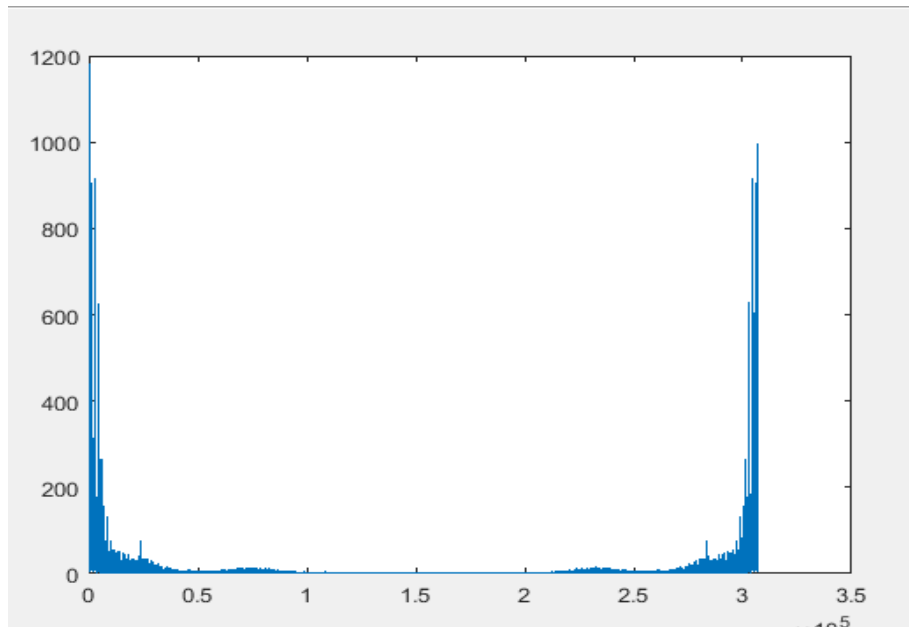


Fig. 9. Output speech signal processed by FIR filter

## VI.CONCLUSION

High speed MAC architecture for FIR filter is proposed in this brief using 4:2 compressors. We chose 4:2 compressor as the optimal choice since modern datapath elements have fixed datapath in multiples of  $2^n$  where  $n=3,4,5$  etc.,. From the obtained results, it is evident that the proposed MAC is able to operate at higher clock frequencies than the conventional schemes. The total delay of the proposed MAC unit is found to be 1026 ps, which is very much less than the conventional MAC. In addition proposed MAC reduces the power requirement to greater extent and suitable for portable designs. Finally, proposed MAC is implemented in FIR filter for speech signal processing. Implementation results proved that the FIR system with proposed MAC perform superior in terms of area and energy reductions compared to conventional MAC based FIR system



#### REFERENCES

- [1] Jithendra Babu N., Sarma R., “A Novel Low Power Multiply–Accumulate (MAC) Unit Design for Fixed Point Signed Numbers”, In: Dash S., Bhaskar M., Panigrahi B., Das S. (eds) Artificial Intelligence and Evolutionary Computations in Engineering Systems. Advances in Intelligent Systems and Computing, 2016, vol 394. Springer, New Delhi
- [2] Seo. Y and Kim. D, “A New VLSI Architecture of Parallel Multiplier–Accumulator Based on Radix-2 Modified Booth Algorithm”, in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2010, vol. 18, no. 2, pp. 201-208.
- [3] Milos D. Ercegovac, Tomás Lang , “Digital Arithmetic” , Elsevier, 2004, Pg: 59-63
- [4] Chang, Chip Hong, Zhang et al., “Ultra Low-Voltage Low-Power CMOS 4-2 and 5-2 Compressors for Fast Arithmetic Circuits” IEEE Transactions on Circuits and Systems I: Regular Papers, 2004, DOI: 10.1109/TCSI.2004.835683.
- [5] Singh. K. N., Tarunkumar. H, “A review on various multipliers designs in VLSI”, Proc. Annual IEEE India Conference (INDICON), 2015, New Delhi, pp. 1-4.
- [6] Patil. P. A., Kulkarni. C, “A Survey on Multiply Accumulate Unit”, Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018, Pune, India, pp. 1-5.
- [7] Sai Kumar. M, D. Kumar. A and Samundiswary. P, “Design and performance analysis of Multiply-Accumulate (MAC) unit”, International Conference on Circuits, Power and Computing Technologies [ICCPCT],2014, Nagercoil, 2014, pp. 1084-1089
- [8] Nagaraju. N, Ramesh. S.M., “Implementation of high speed and area efficient MAC unit for industrial applications”, Journal of Cluster Computing (Springer) 22, Pg.4511–4517, 2019. <https://doi.org/10.1007/s10586-018-2060-z>.
- [9] Kwon. O, Nowka. K, & Swartzlander. E.E, “A 16-Bit by 16-Bit MAC Design Using Fast 5:3 Compressor Cells”, The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology, 2002, vol. 31, pp.77–89, DOI: <https://doi.org/10.1023/A:1015333103608>
- [10] Malleshwari, R. and E. Srinivas,“FPGA Implementation of Low Power and High Speed 64-Bit Multiply Accumulate Unit for Wireless Applications”, International Journal of Science and Research (2016). DOI: 10.21275/v5i4.14041608
- [11] Kapse. V, Jain. A, Pattanaik. M, “Design of an Area Efficient and Low Power MAC Unit”, in: “Smart Trends in Information Technology and Computer Communications”, Unal. A, Nayak. M, Mishra. D, Singh. D, Joshi. A (Eds), Proc. of Smart Trends in Information Technology and Computer Communications: Communications in Computer and Information Science, 2016, vol 628. Springer, Singapore.
- [12] Asif. S and Kong. Y, “Analysis of Different Architectures of Counter Based Wallace Multipliers”, Proc. IEEE Computer Engineering and Systems (ICCES),2015, pp. 139-144.
- [13] Chi-Jui Chou, Satish Mohanakrishnan, Joseph B. Evans “FPGA IMPLEMENTATION OF DIGITAL FILTERS” Proc. ICSPAT’93.
- [14] Ashish B. Kharate, Prof. P. R. Gumble, “VLSI Design and Implementation of Low Power MAC for Digital FIR Filter” International Journal of Electronics Communication and Computer Engineering, Volume 4, Issue (2), REACT-2013.