



An optimized MAC based architecture for adaptive digital filter

Britto Pari James^a, Vaithiyanathan Dhandapani^{b*} & Karuthapandian Mariammal^c

^aVel Tech Rangarajan Dr Sagunthala R&D Institute of Science and Technology
Chennai 600 062, Tamil Nadu, India

^bNational Institute of Technology Delhi, Delhi 110 040, India

^cMadras Institute of Technology, Anna University, Chennai 600 044, Tamil Nadu, India

Received: 26 May 2020

Filter design in signal processing field plays a vital role in achieving low power dissipation, which is essential for portable gadgets. This paper proposes an effective flexible FIR filter structure, which is adaptive and utilizes multiply-accumulate (MAC) core. Most common algorithm for filter coefficient optimization includes least mean square (LMS) and recursive least square (RLS). Though the performance of the recursive least square (RLS) algorithm is superior as compared to the least mean square (LMS); because of higher arithmetic complexity in design, it has not been preferred for real time applications. The fundamental filter has used a LMS based tapped delay line filter, which is practically a feasible choice for adaptive filtering algorithm in order to attain lesser computation. In the proposed work, the adjustable coefficient filters using an optimized LMS approach has been implemented for the utilization of determining the unexplored system. The filter tap considered here is a 32-tap and its analysis and synthesis has been carried out using hardware description language (HDL) programming and synthesized in field programmable gate array (FPGA) devices. The placement and post routing design has offered good performance in terms of utilized resources. The implemented filter architecture requires 80% reduction in resources and has enhanced the clock frequency by about five times when examined with the reported architecture.

Keywords: Memory optimisation, Adaptive filter, LMS, FIR, MAC, FPGA

1 Introduction

In recent days, the adaptive filter becomes popular for signal processing and at the same time find solutions for various applications such as echo cancellation and noise removal in different communication techniques. The adaptive filters can be realized with different methods such as real time embedded systems either using digital signal processors or microprocessors or microcontroller and for improving the performance the hardware realization is opted by either using field programmable gate array (FPGA) devices or in ASIC/SoC domain either full-custom or semicustom design. Adaptive filters consists of two sections one is mainly to update the filter coefficients, and other for computing the filter output. The filtering part is alike as conventional digital filter, at the same time as the adaptation part requires a suitable algorithm for updating the filter coefficients¹. In recent years, numerous coefficient updating algorithms have been reported by many researchers. Among the works most of them related to the Delay Least Mean Square

(DLMS) algorithm, which is closer to LMS algorithm²⁻⁵. The DLMS algorithm uses precedent error for revise the filter weight instead of current error this makes the degrading error performance of DLMS algorithm judge against to the LMS algorithm. At the same time DLMS algorithm have the advantages of pipelining, which provides shorten the iteration period compared to the LMS based design⁶⁻¹⁰. There are many reported techniques to improve the performance of the filter architecture¹¹⁻¹⁷.

There are two schemes of memory techniques are used to avoid the embedded multipliers. These techniques are distributed arithmetic (DA) and look up table schemes. The main approach of DA scheme is use the shift and addition operation and avoiding the multiplication, which significantly reduces the area. At the same time LUT scheme stores all feasible values of inner-products. Meher¹⁸ proposed the new approach for LUT design and multiplication based realization of FIR filter, this paper adopted this technique and a parallel pipeline technique for optimizing the multiplier hardware. The main feature of the OPC architecture scheme is to reduce the complexity at the same reduce the delay of operation. By using pipelined and parallel

*Corresponding author (E-mail: dvaithiyanathan@nitdelhi.ac.in)

multiplier is used to improve the speed of the operation. Because of computational complexity, the realization of adaptive systems in hardware is not all the time straightforward method. But at the same in the real time application high performance in terms of power, speed and area will achieve by hardware realization¹⁹⁻³⁹. The hardware implementation of digital signal processor algorithms is by means of ASIC/SoC or digital signal processor or by using general purpose processor. Among these methods the digital signal processor gives more simplicity, flexibility and programmability for realization of any complex algorithms, but at the same time power consumption is high. In recent years, the FPGA devices usage is increases in hardware domain, because of its advantages like short design cycle and time to market, so in this work we adapted the FPGA devices for hardware implementation. In recent years FPGA device contains many resources in terms of intellectual property cores, which supports digital signal processing applications such as floating multipliers or embedded integer operations, multiply and accumulate units and many other cores. In the direct implementation of adaptive finite impulse response filter architecture for N-tap will requires N numbers of MAC operations. This is more costly due to hardware complexity and more area and power consumption. These factors motivated to design the filter architecture with time sharing based multiplier structure across pipelined MAC core. The existing and modified architecture is first verified in MATLAB tool and moreover in order evaluate the hardware requirement it is further implemented using verilog hardware description language and synthesized in Altera and Xilinx FPGA devices. The performance metrics such as area, speed and power dissipation is considered for evaluation of hardware.

The paper flow is as follows. In Section 2 we describe the algorithm description of adaptive filter. In the Section 3 we discussed about the multiplier technique. In the Section 4 pursues with the architecture design of adaptive filter. In the Section 5 the performances of modified and reported structure are discussed and analyzed. In Section 6 the conclusions were discussed.

2 Algorithm description

An effective and adjustable coefficient calculation is done to modify the characteristics of adaptive filters in order to curtail the cost of the work picked out for

the specific application. The algorithm involves iterative procedure for determining the valuable adjustable coefficient which is essentially done with the help of the step index parameter. This LMS based direct filter design composed of two essential procedures.

The design procedure of the filter includes figuring out the yield of a tapped delay line methodology along with several inputs of the tap and obtaining the actual computation error by contrasting the obtained yield with an ideal one. In this processing, the tap coefficients are updated automatically based on the obtained actual computation error. This least means square based calculation begins with the computation of the output the filter while information tests line up in a channel. Further the evaluated output signal is observed and the difference between this signal and original signal $d(i)$ is carried out. This subtraction is quite useful to ascertain the channel coefficients newly. This above mentioned process is carried out for every trained cycle and the error computed is utilized for updating the filter coefficients. For i^{th} iteration, this process can be mathematically given as

$$W_{i+1} = W_i + \mu \cdot e(i) \cdot X_i \quad \dots (1)$$

where

$$e(i) = d(i) - y(i)$$

$$y(i) = X_i W_i^T$$

For every i^{th} training iteration, X_i denotes input vector and W_i denotes weight vector, $d(i)$ denotes the required original response and $y(i)$ indicates the actual response obtained from the filter. Likewise the evaluated error $e(i)$ is mainly applied for changing the weights along with the parameter μ in order to have required convergence. Here the term N indicates the order of the designed filter. Dual MAC core pipelined architecture is proposed in this work.

3 Multiplier structures for adaptive FIR filter

The multiplier complexity is higher than the adder complexity in general. Similarly the speed of operation is decided by the critical path time. Hence most of the DSP algorithms and its performance characteristics are rely on the multipliers as well as critical path. OPC and pipelined parallel multiplier are the two multiplier architectures which are used for the adaptive FIR filter and is discussed in¹⁹. Meher proposed the odd multiple storage schemes¹⁸ in which

the odd multiple of the filter weights scheme is applicable only for unsigned number input. However, this scheme is inappropriate for negative inputs. OPC scheme is used to overcome the disadvantage of odd multiple storage. The OPC scheme takes into account both positive as well as negative sample inputs. The parallel multipliers of OPC are widely employed in most of the high performance DSP processors, but it requires more hardware when examining with the serial multiplier in order to avail enhanced performance. Several methods are available to design the parallel multipliers. All the partial products are combined in the easiest $L \times L_1$ bit parallel multiplier to achieve the required output. The main drawback of

this approach is that reduced speed of operation because of bit based multiplication. In order to improve the speed of operation, pipelined registers are inserted by altering combinational circuit into a sequential one and this is facilitated in Fig. 1.

4 Delayed LMS algorithm based adaptive FIR filter MAC structure

FIR filter with appropriate constant coefficients can be employed for the characterization of well defined signal as well as noise. Nonetheless, the use of this filter is inappropriate for the characterization of non stationary signals. Moreover the desired response was obtained by choosing dynamically varying

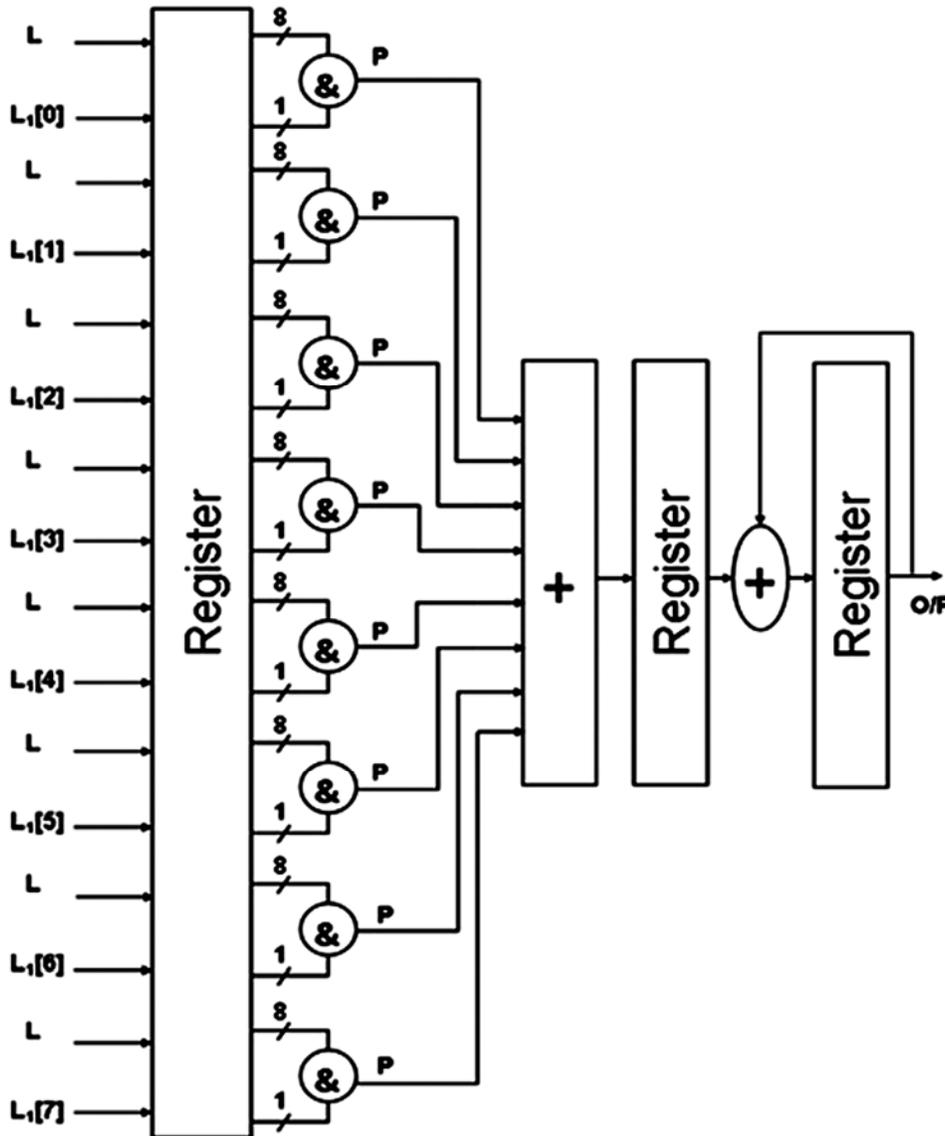


Fig. 1 — Structure of parallel pipelined multiplier.

coefficients of a filter with respect to input. Hence adaptive filter plays a vital role in several applications. There are various algorithms available to achieve required convergence rate as well as mean square error. This work uses LMS algorithm based adaptive filter for better design. The realization of this filter includes adder, multiplier and delay networks which is similar to constant coefficient filter design.

Figure 2 represents the existing conventional adaptive FIR filter architecture. The main drawback of this structure is number of multipliers are also increased with the number of taps. To mitigate this drawback, an improved and an efficient design is suggested in this work. The proposed design involves time division based multiplier which is utilized for the entire dual core multiply and accumulate unit. The required speed of operation of the filter is achieved by the inclusion of delay registers at needed locations such that their functionality remains unchanged.

Figure 3 shows the pictorial representation of suggested pipelined based 4-tap adaptive FIR filter architecture. This architecture involves of two major parts, which are namely filter part or an error computation part and the other one is weight update part or algorithm based block. The weight update block has a pipelined based single MAC algorithm with conventional LMS architecture. To complete the filter operation on N tap filter, needed number of clock cycles is about N. For example, considering a Size of FIR filter is four with the input data rate of about 2 Mega samples per second (MSPS), then the output data rate of the filter is raised to eight times of input MSPS. The multiplier receives the data via delay blocks for every clock cycle in a filter and the result is available in the four clock cycles. The input to the 4:1 multiplexer is 8-bit and is given via a

number of delay elements which delays the sample by one clock cycle and the resultant architecture is depicted in Fig. 3. The multiplier operation carried out with the help of Four to one MUX that choose the data through the Flip-flops. The counter is used to choose the select line of 4:1 MUX. In the first clock cycle, the multiplier operation is done through counter selection.

Likewise, the consecutive inputs are waited in every flip flops and the waited samples are registered and sent to the multiplier by enabling the counter input in every clock cycle. The multiplexer selects the appropriate data via delay elements and then multiplication is carried out using the filter coefficients which are kept in the registers w_0, w_1, w_2 and w_3 and are rounded to fixed point value. After fourth clock cycle, the earlier value is added to the current value by accumulator section and then it is changed to a zero value. The select lines to the multiplexer and the accumulator process are restrained via a common counter. By using appropriate counter, delay registers and multiplexer, single MAC core used to implement N number of taps. Single MAC-LMS based pipelined is implemented in weight update block. The subtraction is carried out between the desired signal ' d_{in} ' and the output of the filter called ' Y_{out} ' is marked as error signal and this signal is taken as the data input for updating the filter weights. The parameter called step index μ is multiplied with an error signal to have appropriate optimal efficiency and then the outcome is multiplied to X_{in} . For each and every clock cycle, the resultant is added and de-multiplexed output data is infused to update the filter coefficients w_0, w_1, w_2 and w_3 simultaneously through delayed registers. Likewise, filter with any size of filter is able to be

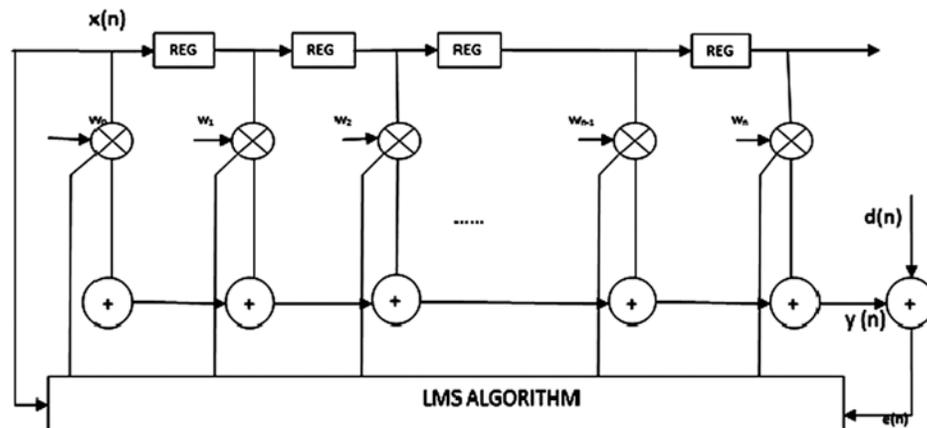


Fig. 2 — General structure of adaptive FIR filter.

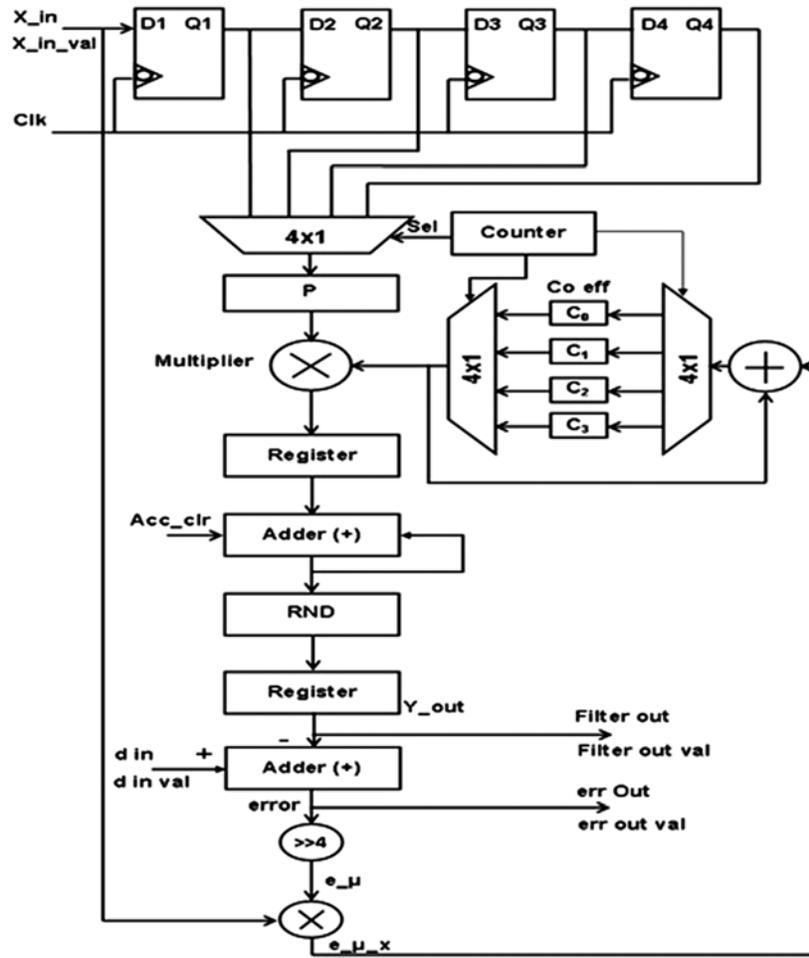


Fig. 3 — Proposed MAC architecture of adaptive FIR filter.

implemented by Dual MAC based multiplier and adders using addition of registers which is used for broadening the operating frequency of a filter. In this suggested architecture, two multipliers employ the error computation as well as weight update sections by time division approach irrespective of filter taps. The inclusion of two multiplier schemes enhances the filter performance in terms of reduction in area and enhances the frequency of operation by minimising the longest path delay. The parallel processing and pipelined multipliers are used to enhance the performance to a significant extent. In this time division based multiplier technique, pipelined registers are added to enhance the operating speed. The Output Product Scheme oriented L-bit multiplier scheme reduces the necessary memory locations from 2^L to 2^{L-2} for storing the partial products. Examining this with the odd schematic based multiplication, in this scheme, memory size is significantly reduced, which results in reduced complexity.

4.1 FIL implementation

FIL simulation is obtained by MATLAB/Simulink model and is implemented in DE2-115 FPGA board. JTAG interface is used as an interconnect between FPGA board and PC. The Simulink model of LMS based adaptive filter is modelled and the performance of the LMS filter is simulated by varying the step index term (μ). A sine wave of frequency 100Hz is taken as input and is mixed with random sequence and this mixing signal is set as the other input signal for the design validation. Simulation is done for the whole model and the response of the LMS filter is depicted in Fig. 4. After the corresponding simulation results, FPGA-in-the-Loop (FIL) Wizard is launched. Then Altera DE2-115 is connected to have essential communication with Simulink via the interface called JTAG. Using HDL coder, the HDL code is generated from Simulink platform. Then the suggested adaptive filter design code is considered instead of generated HDL code in order to perform building process.

Figure 5 illustrated Simulink model of the designed and obtained FIL LMS based filter. The synthesis report is taken from the FPGA software through a build process. Further the Design file is generated by FPGA in Loop and is stored into FPGA through JTAG cable. Both simulink models as well as FIL simulation results are matched.

5 Results and Discussion

The proposed architectures were compiled and synthesis was carried out with Altera Cyclone IV4CE115F23C7 and Xilinx Virtex-5 FPGA platform. To test the model performance for real time operations, FPGA in loop is carried out in MATLAB Simulink tool and is used to support Altera DE2-115

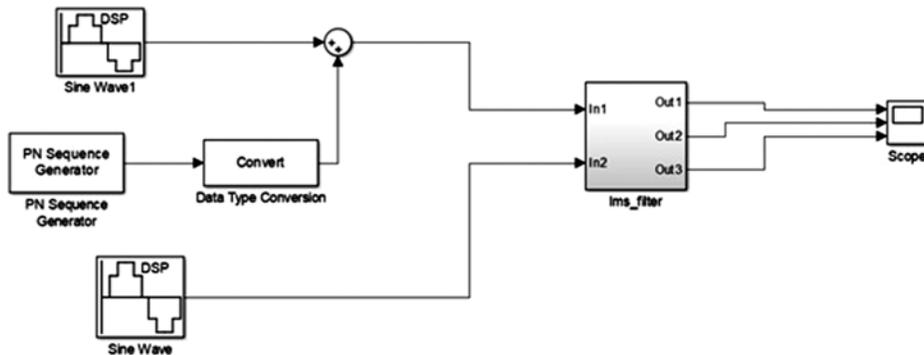


Fig. 4 — LMS adaptive simulink model.

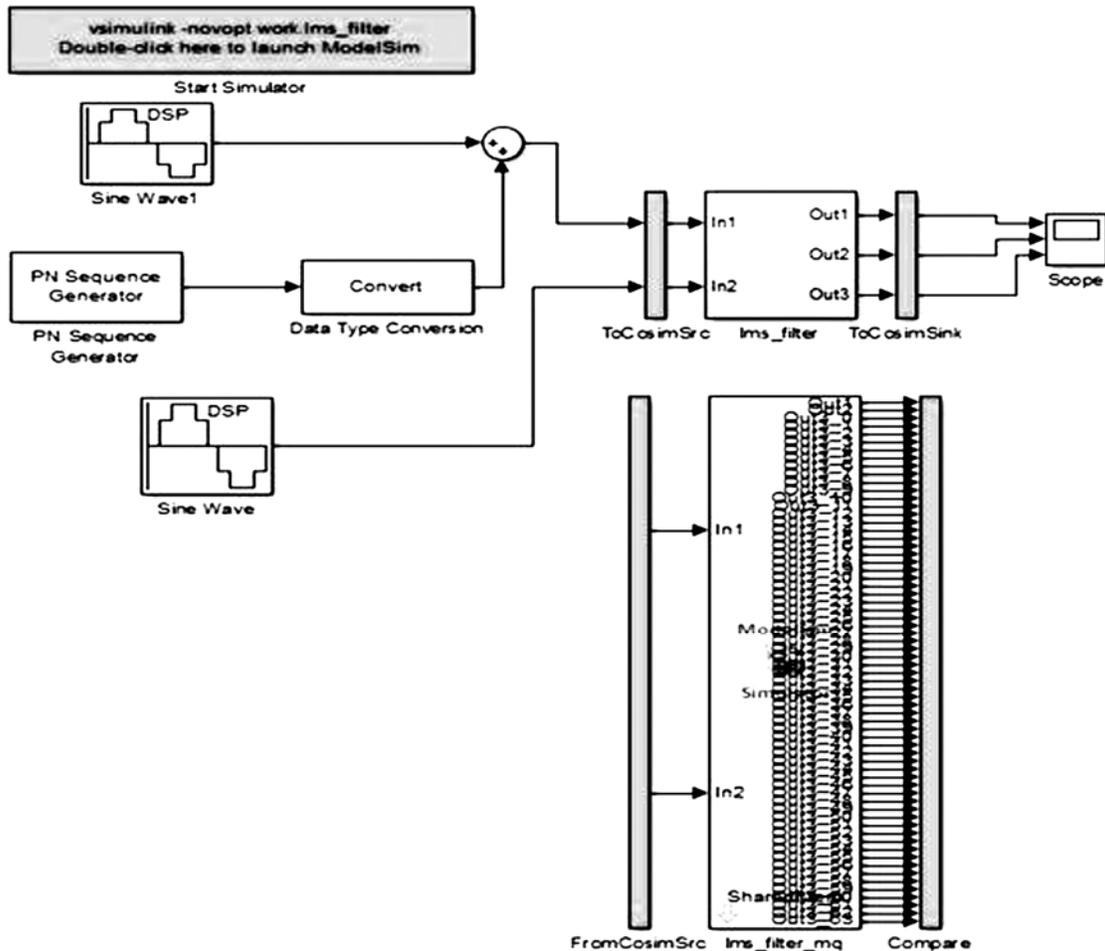


Fig. 5 — FPGA in loop simulink model for LMS adaptive filter.

hardware and time complexity of the designed adaptable filter. The required 6 clock cycles composed of several operations. Four clock cycles are required for MAC operation. Among the 4 clock cycles, one clock cycle is used for multiplication, two cycles are needed for input registering and the remaining one cycle is used for performing accumulate process. The remaining two cycles are utilized for output register and coefficient update process. The needed register involved in this structure is about $N+6$ in number. Here N specifies number of filter taps. Because of time division based multiplication structure, dual MAC is enough for carrying out filtering process, irrespective of any value of taps.

From Table 2, the proposed MAC LMS structure based on time sharing occupies only two multipliers irrespective of number of taps. It has lesser complexity and higher convergence rate. The adaptive FIR architectures performance is analyzed using FPGA devices. The OPC and parallel pipelined multiplier schemes are used to synthesize the proposed architectures, its results are tabulated in Table 3. When related to the general FIR filter structures, the proposed LMS structures reduce area drastically and increases the speed due to usage of MAC core architecture and pipelining. Further analysis of Table 4 shows that the proposed pipelined multiplier performs better than other architectures due to bit product matrix and LUT which leads to reduce area usage.

Table 4 compares the synthesis output of filter structures with size of filter is 16 and the conventional structures by taking parameters like slices, frequency

as well as delay. The suggested structures are examined with carry save adder utilized distributed arithmetic oriented structure proposed by Meher⁶ when compared to the existing architecture, the suggested adaptable coefficient based MAC filter employing OPC structure achieves 82% speed enhancement as well as 78% improvement in slice delay product. The parallel pipelined multiplier structure achieves 60% improvement in speed and 68% improvement in slice delay product. The synthesized outputs arrived using Altera Stratix EP1S80F1508C6 FPGA platforms and the results for the size of 16 and 32 based suggested adaptable filter is compared with the conventional structures and is listed in Table 5. Allred⁵ proposed a secondary look up table based optimal addressing technique in order to update the coefficients. Likewise, the results obtained with various platforms are compared between suggested adaptable filter structure and the conventional structures and are given in Table (6-8).

Table 1 — Hardware implementation of LMS filter.

MAC based LMS filter	Area Usage			
Size of LMS filter	2	8	16	32
Usage of multipliers	2	2	2	2
Number of registers	8	14	22	38
Latency	6	6	6	6

Table 2 — Area complexity of adaptive algorithms.

Algorithm	Computational Multipliers	Computational Adders	Convergence rate
LMS	N	N	HIGHER
RMN	$2N+3$	$2N+2$	LOWER
Robust	$N+3$	$N+5$	LOWER
MRMN	N	N	HIGHER
Proposed	2	$N+1$	HIGHER

Table 3 — Synthesis results of LMS filter.

Parameters	General LMS OPC		MAC based LMS OPC		General LMS pipeline multiplier		MAC based parallel pipeline multiplier	
	VIRTEX-5 XC5VSX95T-1FF1136							
Family								
Size of filter	16	32	16	32	16	32	16	32
Slices	808	1586	210	290	240	412	135	280
Delay(ns)	11.697	16.817	3.12	3.14	19.129	23.684	7.14	8
Frequency (MHz)	85.489	59.465	320	318	52.27	42.22	140	125

Table 4 — Synthesis results of MAC LMS filter with existing architectures.

Parameters	Delay (ns)	Frequency (MHz)	Slices	Registers	LUT	Efficiency of Slice-delay
VIRTEX-5 XC5VSX95T-1FF1136						
Meher ⁶	17.35	57	178	412	267	-
MAC LMS OPC	3.14	318	210	300	1400	78
MAC LMS parallel pipeline multiplier	7.14	140	135	200	350	68

Table 5 — Synthesis comparison of MAC LMS filter with existing architecture.

Parameters	Configurable Logic Blocks	
	Family	Altera Stratix EP1S80F1508C6
Size of filter	16	32
Allred et al., [k=2] ⁵	1309	2244
Allred et al., [k=4] ⁵	915	1429
Allred et al., [k=8] ⁵	798	1073
MAC LMS OPC	1200	1650
MAC LMS parallel pipeline multiplier	600	1100

Table 6 — Synthesis comparison of proposed MAC LMS filter with prevailing architectures.

Parameters	Rosado-Muñoz ³⁸	Parmar ⁸	Proposed MAC -OPC	Proposed MAC – Pipelined
Family	Xilinx Virtex-4	XC4VFX12	FF6618-12	
Number of slices	2586	629	1011	218
Delay(ns)	52.71	35.84	3.631	9.102
Maximum Operating frequency(MHz)	18.97 MHz	27.895 MHz	275.383	109.871

Table 7 — Synthesis results of proposed MAC LMS filter with existing architecture.

Parameters	Rosado-Muñoz ³⁸	Parmar ⁸	Proposed MAC -OPC	Proposed MAC - Pipelined
Family	Xilinx Virtex-5	XC5VLX30	FF324-3	
Number of slices	3906	643	1188	225
Delay(ns)	39.6	31.19	3.524	7.959
Maximum operating frequency(MHz)	25.27 MHz	32.060 MHz	283.767	125.637

Table 8 — Implementation results of MAC LMS filter with existing architecture.

Parameters	Rosado-Muñoz ³⁸	Parmar ⁸	Proposed MAC -OPC	Proposed MAC - Pipelined
Family	Xilinx Spartan 3E	XCS500E-4		
Number of slices	2586	277	739	225
Delay(ns)	52.7	20.05	5.253	17.244
Maximum operating frequency(MHz)	18.97	49.863	190.351	57.993

The results shows that the suggested structures uses less number of logical components because of the time sharing based MAC core filter structure

6 Conclusions

In this work an area efficient MAC is implemented with an adjustable coefficient based FIR filter

employing LMS scheme is proposed. The time division based multiplier structure across the MAC core offers significant reduction in hardware cost. The speed of the proposed designs enhanced about 82% for OPC scheme and 60% for pipelined multiplier scheme when examining with the respective conventional design. The suggested OPC structure improves the slice delay product by 78% and the over the existing structure. Likewise the proposed parallel pipelined multiplier architecture improves the slice delay product by 68% over the existing architecture. The suggested 16-tap adaptive filter implementation works in the input sampling frequency range of up to 320MHz. The proposed architectures offer significant reduction in hardware complexity as well as enhanced speed over the conventional designs. The suggested multiplier architectures are well suited for optimal design of filters with larger order and are used for various signal processing applications.

References

- Chin-Liang Wang, *IEEE Transactions Signal Process*, 42 (1994) 2169.
- Safarian C, Ogunfunmi T, Kozacky W J & Mohanty B K, *2015 IEEE Int Conf Digit Signal Process (DSP)*, (2015) 1251.
- Proakis J G & Manolakis D G, *Digital Signal Processing: Principles, Algorithms and applications* (Pearson India), 4thEdn, ISBN: 9788131710005, 2007.
- Mirchandani G, Zinser Jr R L & Evans J B, *IEEE Trans Circuits Syst II, Analog Digit Signal Process*, 39 (1995) 681.
- Allred D J, Yoo H, Krishnan V, Huang W & Anderson D V, *IEEE Trans Circuits Syst I, Reg Papers*, 52 (2005) 1327.
- Meher P K & Park S Y, in *Proc IEEE/IFIP19th Int Conf VLSI-SOC*, (2011) 428.
- Bergamasco M, Rossa F D, & Piroddi, *J Sound Vib*, (2012) 27.
- Parmar C A, Ramanadham B & Darji A D, *IET Comput Digit Techniques*, 11 (2017) 107.
- Park S Y and Meher P K, *IEEE Transac Circuits Syst-II: Express Briefs*, 61 (2014) 511.
- LogiCORE IP FIR Compiler v5.0, Xilinx, USA, 2010.
- Mahesh R & Vinod A P, *IEEE Trans Computer-Aided Des Integr Circuits Syst*, 27 (2008) 217.
- Abbaszadeh A, Azerbaijan A & Sadeghipour K D, *2011 17th Int Conf Digit Signal Process (DSP)*, (2011) 1.
- Jongsun Park , Woopyo Jeong, Yongtao Wang, Choo H & Roy K, *IEEE J Solid State Cir Sys*, 39 (2004) 348.
- Mahesh R & Vinod A P, *IEEE Transac Computer aided Des Integrat Circuits Syst*, 29 (2010) 592.
- White S A, *IEEE ASSP Mag*, 6 (1989) 5.
- Yoo H & Anderson D V, *IEEE Int Conf Acoustics, Speech, Signal Process*, 5 (2005) 125.
- Jeng S -S, Lin H -C & Chang S -M, *2006 IEEE Intl Symp Circuits Syst ISCAS*, (2006) 875.
- Meher P K, *IEEE Transac Circuits Syst*, 57 (2010) 592.
- BrittoPari J & Joy Vasantha Rani S P, *ARPN J Eng Appl Sci*, 10 (2015) 4964.

- 20 BrittoPari J & Vaithiyathan D, *Int J Appl Eng Res*, 12 (2017) 2209.
- 21 BrittoPari J & Vaithiyathan D, *Proc IEEE Int Conf Wireless Commun Signal Process Networking*, (2019) 978.
- 22 Chan Y H & Siu W C, *IEEE Trans Circuits Syst I, Fundam Theory Appl*, 39 (1992) 705.
- 23 Chen H -C, Guo J -I, Chang T-S & Jen C -W, *IEEE Trans Circuits Syst Video Technol*, 15 (2005) 445.
- 24 Meher P K, Chandrasekaran S & Amira A, *IEEE Trans Signal Process*, 56 (2008) 3009.
- 25 Meher P K, *IEEE Trans Circuits Syst I, Reg Papers*, 53 (2006) 2656.
- 26 Guo J -I, Liu C -M & Jen C -W, *IEEE Trans Circuits Syst II, Analog Digit Signal Process*, 39 (1992) 723.
- 27 Chiper D F, *Proc In IEEE Conf Image Process*, (1999) 764.
- 28 Chiper D F, Swamy M N S, Ahmad M O & Stouraitis T, *IEEE Trans Circuits Syst I, Reg Papers*, 52 (2005) 1125.
- 29 Meher P K & Swamy M N S, *IEEE Trans Circuits Syst II, Exp Briefs*, 54 (2007) 262.
- 30 Meher P K, Patra J C, & Swamy M N S, *IEEE Trans Circuits Syst II, Exp. Briefs*, 54 (2007) 606.
- 31 Marouane H, Kachouri A, Kamoun L, *16th Inter Conf Microelectronics*, (2004) 637.
- 32 Hemantha G R, Varadarajan S & Giriprasad M N, *J Sci Indust Res*, 79 (2020) 135.
- 33 Mahil J, SreeRenga Raja T & SreeSharmila T, *Indian J Pure Appl Phys*, 53 (2015) 274.
- 34 Mehendale M, Sherlekar S D & Venkatesh G, *10th Int Conf VLSI Des*, (1997) 124.
- 35 Xu D & Chiu J, *in Proc. IEEE Southeastcon*, (1993).
- 36 Kha H H, Tuan H D, Vo B -N, & Nguyen T Q, *IEEE Trans Signal Process.*, 55 (2007) 4405.
- 37 Dam H H, Cantoni A, Teo K L, & Nordholm S, *IEEE Trans Circuits Syst I, Reg Papers*, 54 (2007) 1348.
- 38 Rosado-Muñoz A, Bataller-Mompeán M, Soria-Olivas E, Scarante C & Guerrero-Martínez J F, *IEEE Trans Ind Electron*, 58 (2011) 860.
- 39 Meher P K, *IEEE Transac Signal Process*, 56 (2008) 3009.