Studies on Design of FIR Filters using Arithmetic Optimization Algorithm

By
KAUSHIK DEBNATH
Registration No. 154081 of 2021 - 2022
Exam Roll No. M4ETC22012

Under the Guidance of:
Prof. P. VENKATESWARAN

Thesis submitted in partial fulfillment of the requirements for the award of the Degree of Master of Engineering in Electronics and Telecommunication Engineering

Department of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata 700032

June 2022

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains a literature survey and original work by the undersigned candidate, as part of his Master of Electronics and Telecommunication Engineering.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name of the Candidate : **Kaushik Debnath**

Exam Roll No. : **M4ETC22012**

Thesis Title : **Studies on Design of FIR Filters using Arithmetic Optimization Algorithm**

Signature of the Candidate : ------------------------------------

Date:

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

I hereby recommend that this Thesis prepared by **Kaushik Debnath** entitled **Studies on Design of FIR Filters using Arithmetic Optimization Algorithm** under my supervision be accepted for partial fulfillment of the requirements for the award of the Degree of Master of Electronics and Telecommunication Engineering.

……………………………………..
Dr. P. Venkateswaran
Professor
Dept. of Electronics and Tele-Communication Engineering
Jadavpur University, Kolkata – 700032. W. Bengal. India.

…………………………………
Dr. Ananda Shankar Chowdhury
Professor and Head
Department of Electronics and
Telecommunication Engineering,
Jadavpur University, Kolkata - 700032.

………………………………………
Prof. Chandan Mazumdar
Dean
Faculty of Engg. & Technology
Jadavpur University
Kolkata 700032. W Bengal.

FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL*

The foregoing Thesis is hereby approved as a creditable study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the Degree for which it has been submitted. It is understood that by this approval, the undersigned don't necessarily endorse or approve any statement made, opinion expressed or the conclusion drawn therein but approve the Thesis only for the purpose for which it is submitted.

Committee on ………………………
Final examination for
Evaluation of the thesis of:
KAUSHIK DEBNATH
Exam Roll No: **M4ETC22012** ……………………….

……………………….
Signature of Examiners

* Only in case the thesis is approved

ACKNOWLEDGEMENT

I would like to take the opportunity to express my thanks of gratitude to my thesis supervisor Prof. Palaniandavar Venkateswaran, Department of Electronics & Tele-Communication Engineering, Jadavpur University for his confidence in me regarding the project and his valuable suggestions, and ideas at every step of the thesis.

I express my special thanks to Dr. Supriya Dhabal, Assistant Professor, Department of Electronics & Communication Engineering, Netaji Subhash Engineering College, Kolkata, for his constant and effortful guidance, prompt inspiration, and timely suggestions with kindness, which have enabled me to prepare my thesis work.

I express my sincere gratitude to Dr. A. S. Chowdhury, Head of the Department of Electronics & Tele-Communication Engineering, and Jadavpur University for his support and motivation. I am also very grateful to all the teaching and non-teaching staff of the department for providing necessary support.

Last but not the least, I would like to thank, my family, parent, and my friends for giving me constant support and motivation in preparing the whole thesis work.

Computer Networking Lab (T-3-16A)

ETCE Dept., Jadavpur University

Kolkata - 700 032. West Bengal.

June 2022

....

KAUSHIK DEBNATH

ABSTRACT

In this paper, an efficient design of three FIR filters low pass, high pass, and band pass filters using the arithmetic optimization algorithm, have been proposed. The values of stop-band attenuation and pass-band ripple are obtained using AOA and are compared to the values obtained by using PM (Park-McClellan) and Cuckoo Search Algorithm (CSA) method. Further, the design is implemented on FPGA using SPARTAN-6 XC6SLX16-2FTG2T6 for different orders and simulated with XILINX ISE Design suite 14.7. The design summary report and power reports are obtained of the filters for different orders of their respective algorithm which are comparable to find the efficiency and robustness of the proposed algorithm.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Preamble

We use various kinds of filters in our daily lives, homes, offices, and schools like the water filter, air filters, etc. This points to the fact that filters are important for us to live a good standard of living. Similarly, in the field of electronics, filters are very essential elements. The term 'filtering' usually refers to the quality or process of blocking something out while allowing some other things to pass through. Electronic filters refer to the devices that upon receiving input signals, allow only the components within a certain range of frequencies to transmit and attenuate the rest of the components outside that frequency range. Electronic filters have been categorized in the literature based on various criteria. Depending upon the type of components used in construction, filters are classified as — active and passive. Based on rating frequency ranges filters may be classified as Audio Frequency (AF) or Radio Frequency (RF). According to the nature of impulse response, filters can be categorized as— Finite Impulse Response (FIR) or Infinite Impulse Response (IIR) [1-3]. As opposed to IIR filters, FIR filters are non-recursive and the response due to impulse input settles down to zero in a finite amount of time. FIR [2] filters are preferred over IIR filters in a wide array of engineering applications because of the linear phase and greater stability offered by the former. Again, depending on the nature of the filtering operation, filters can be classified as— analog and digital. Digital filter [4-8] circuits have to sample the analog input signal, convert the sampled signal into a set of binary numbers, store the numbers in memory, and process them through a processing unit before digitally manipulating them to yield the final output. None of these steps are required in an analog filter. Despite the higher complexities of digital filters, they are more widely preferred over analog filters these days because of the numerous advantages it offers like linear phase response, adaptability, repeatability, data storage ability, etc. Section 1.2 of this chapter presents a literature survey in the field of optimization techniques. Section 1.3 reveals the motivation behind taking up this work. The thesis outline is presented in Section 1.4 and finally, this chapter is summarized in Section 1.5.

1.2 Literature Survey

In the recent era, the application of optimization problems has found an important space in solving many problems in various application fields. Traditional optimization methods require time and costs, so all optimization problem values cannot solve all problems, they are applicable only to a specific field according to their application. The main objective of using novel meta-heuristic population-based algorithms for digital filter [5] design is to minimize or reduce the error between the ideal response and the approximated response (cost function) as much as possible. These algorithms have the least probability of getting trapped in local minima, known as the “Local optima entrapment” problem, algorithms like particle swarm optimization algorithm [8-15], genetic algorithm, artificial bee colony algorithm, sine cosine algorithm, cuckoo search algorithm, etc. are a significant innovation in the optimization algorithm. The theoretical studies have published that no researchers do not use a single algorithm because there is no optimization algorithm to solve all optimization problems the according to **No Free Lunch Theorem**. This theorem logically proves that any one algorithm can only successfully solve a specific set of problems. This motivates researchers to design and develop a newer modified algorithm to obtain better results. In 2020, Laith Abualigah developed Arithmetic Optimization Algorithm [17-25]. This algorithm mainly depends on the properties of arithmetic operators which are multiplication, division, subtraction, and addition, which helps in exploring and exploiting the minima within the search region. This algorithm can be applied to various design problems like pressure vessel design, 3-bar truss design problems, etc. Based on this algorithm, various other algorithms hybridized with AOA have been developed. Few examples are advanced AOA for solving mechanical engineering design problems hybrid SCA-AOA and fuzzy SCA-AOA[18] in information security, AOA in multi-level thresholding segmentation of COVID-19 CT images, an improved artificial neural network using AOA for damage assessment in FGM composite plates, an improved arithmetic optimization with forced switching mechanism for global optimization problems, application of arithmetic optimization in chaotic mapping strategy, AOA in mutation based problems for global optimization, introduction of a new arithmetic optimization problems for solving real world multi-objective constrained optimization problems, introduction of improved arithmetic optimization problem through modified opposition based learning process, again application of arithmetic optimization problem based on particle energy and particle diffusion, are quite successful in achieving specific and accurate results, so this motivates us to use this in designing a FIR filters with less computational effort.

1.3 Thesis Motivation

Digital FIR[16] filters have never been designed using Arithmetic Optimization Algorithm (AOA), a newly developed population-based meta-heuristic optimization-based algorithm developed by Abualigah et al. in 2021. The algorithm is based on the four arithmetic operators namely addition, subtraction, multiplication, and division. The application and working mechanism of AOA is basically quite easier has a straightforward application based on few parameters compared to the application of other optimization algorithms and the result of the values derived by using other algorithms hybridized with AOA are also quite impressive and efficient. Previously no endeavor was taken up to design the three FIR filters that are low-pass, band-pass and high-pass filter using AOA before, so this becomes a motivating factor and provides the challenge to take up the task with success in it. This newly proposed algorithm has been discussed and its output response has been analyzed with the response obtained using other algorithms.

1.4 Thesis Outline

The thesis has been systematically arranged and efforts have been taken to cover the theoretical aspects in detail about the practical work performed in the laboratory.

Chapter 1 presents the introduction about the importance of filtering, as well as the advantages of the FIR and digital filters over IIR and analog filters respectively. Again a literature survey regarding optimization algorithms, the motivation behind the current work, and finally the thesis outline is provided in the chapter.

Chapter 2 gives the theoretical details along with a pseudo code and flowchart of the Arithmetic Optimization Algorithm (AOA).

Chapter 3 presents the simulation and analysis of the design of three FIR filters using AOA compared to other algorithms that are Park-McClellan (PM) and Cuckoo Search Algorithm (CSA). Relevant figures and data in t6abulated form have been provided.

Chapter 4 gives the final conclusion of the complete thesis work. The future scope in the domain of filter design utilizing optimization algorithms is also discussed in this chapter.

Chapter 2

Design Methodologies:

Optimization Algorithm

2.1 Introduction

The finite Impulse Response (FIR) [7] filter is one of the categories of a filter whose impulse response becomes zero after a finite duration of time. FIR filter structure can be used digitally to implement almost any kind of filter response. The circuit consists of a series of delays, adders, and multipliers. Low pass, Band pass [3], and High Pass filters allow only the components within a certain frequency range to pass through and attenuate the rest of components the outside that frequency range. Moreover, the switching between the stop-band and pass-band would be instantaneous i.e. transition width would be zero.

The term ‘optimization’ refers to the process of selecting the best results from a set of other results. An optimization problem generally consists of inputs, and the cost function is evaluated or iterated on each input to find the minima or the best results out of all the inputs, thus minimizing the output of the objective function, which is known as the fitness or cost of the solution. So, population-based optimization algorithms are highly preferred over traditional methods for solving complicated real-world problems as they provide various advantages like non-requirement of continuous and differentiable cost functions and much lesser probability of getting stuck in local optima without reaching global optima.

The next section presents an extensive description of AOA whose pseudo code and flowchart of the algorithm are given.

2.2 Types of Filter

A filter rejects unwanted frequencies from the input signal and allows the desired frequencies. There are two types of filters:-

a) FIR Filter
b) IIR Filter

a) FIR Filter:-

Finite Impulse Response (FIR) filter is a filter whose impulse is of finite duration because it settles to zero in finite time. It is non-recursive type filter. The present output sample depends on the present input samples.

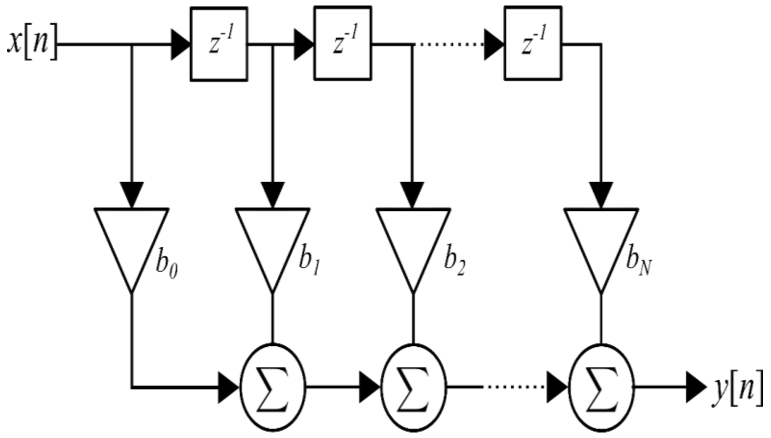


Fig. 1 FIR Filter Block Diagram

b) IIR Filter:-

Infinite Impulse Response (IIR) [9] filter is a filter whose response is of infinite duration because it does not settles to zero in finite time. It is recursive type. The present output sample depends on the present input sample, past sample and output sample.

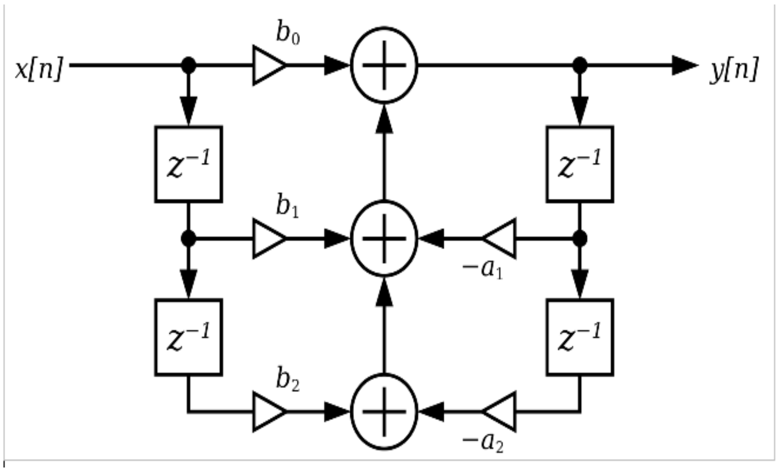


Fig. 2 IIR Filter Block Diagram

2.3 Frequency Selective Filter

The range of frequencies between which the signal can pass through is known as the pass-band and the range between which the signal cannot pass is known as the stop-band.

There are four types of frequency selective filter:-

- i Low-Pass Filer
- ii High-Pass Filter
- iii Band-Pass Filter
- iv Band-Reject Filter

i. Low-Pass Filter

A low-pass filter (LPF) [26-34] is a filter that passes signals with a frequency lower than a selected cut-off frequency and attenuates signals with a frequency higher than the cut-off frequency.

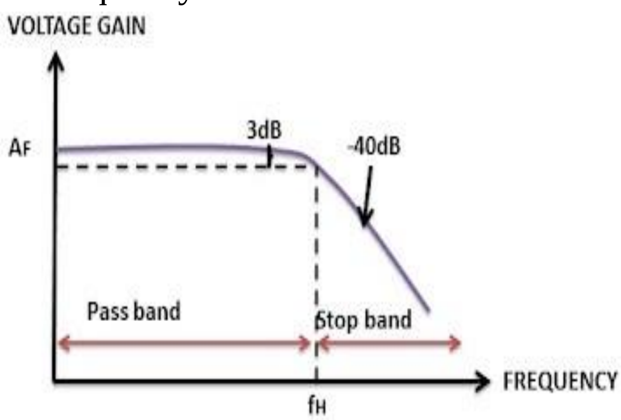


Fig. 3 Low-Pass Filter Magnitude Response

ii. High-Pass Filter

A high-pass filter (HPF) [35-42] is an electronic filter that passes signals with a frequency higher than the cut-off frequency and attenuates signals lower than the cut-off frequency.

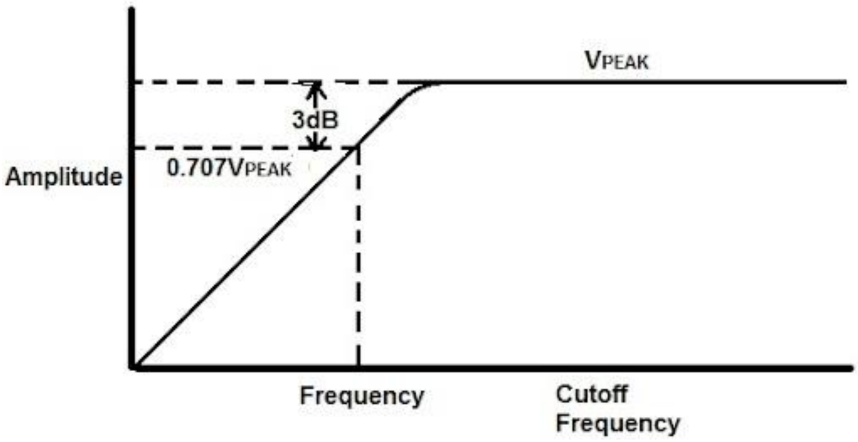


Fig. 4 High-Pass Filter Magnitude Response

iii. Band-Pass Filter

A band pass filter (BPF) [43-49] is a device that passes frequencies within a certain range and rejects frequencies outside that range.

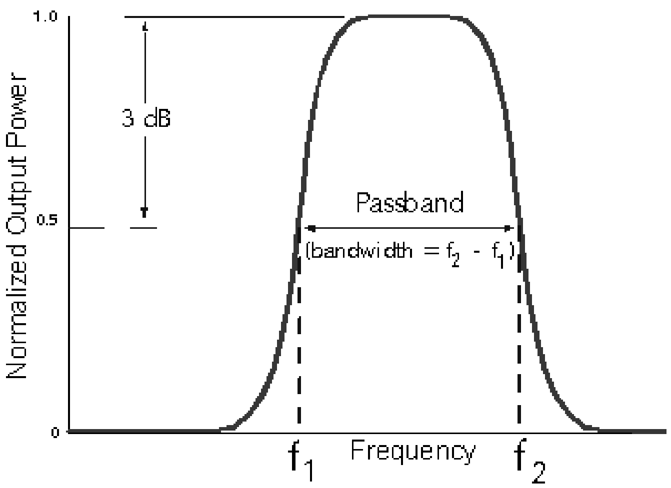


Fig. 5 Band-Pass Filter Magnitude Response

iv. Band-Reject Filter

A band stop or band reject filter is a filter that passes most frequencies unaltered, but attenuates those in a specified range to a very low level.

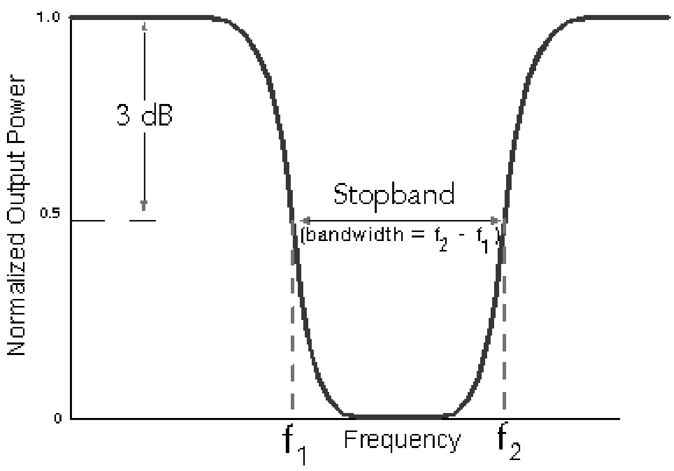


Fig. 6 Band-Reject Filter Magnitude Response

2.4 Arithmetic Optimization Algorithm (AOA)

2.4.1 Overview

Generally, all population-based optimization algorithm is based on some common framework. The algorithms consist of some randomly generated initial solutions called search agents and the fitness value of each solution is determined using the cost function corresponding to the specific algorithm and the process continues repeatedly over the solution until the lowest minima or the desired optimized solution is reached. All meta-heuristic optimization techniques have two main phases – exploration and exploitation. In the former, different sets of solutions are generated randomly, and they impart a large degree of Volatility to zero in on a potential region in the solution space. Whereas, in the latter, the degree of randomness in positioning the updates are reduced which results in exploiting the local optima regions precisely. Generally, all metaheuristic algorithms have four different characteristics, firstly to generate initial random solutions, then present a function to evaluate or any optimization algorithm is applied, thirdly generating positions of random solutions according to the position update formula, and decide which minimum and efficient solutions to be accepted or which have to be ignored using two phases that is exploration and exploitation phase, thus avoiding the problem of local optima entrapment. Thus a balanced optimum approach between the exploitation and exploration phase results in reaching the optimum solution in a much lesser number of iterations.

AOA, a novel population-based meta-heuristic optimization algorithm, was developed recently. The main motivation of AOA [19-22] originates from the use of arithmetic operators i.e. Multiplication, Division, Subtraction, and Addition, which helps in exploring and exploiting the searching region and resulting in finding the least global optimum solution in the searching space. Multiplication and Division operators are used in the exploration phase since they have a very high density of spreading in the solution region. Addition and Subtraction operators are used in the exploitation phase due to their low density of spreading in the solution region.

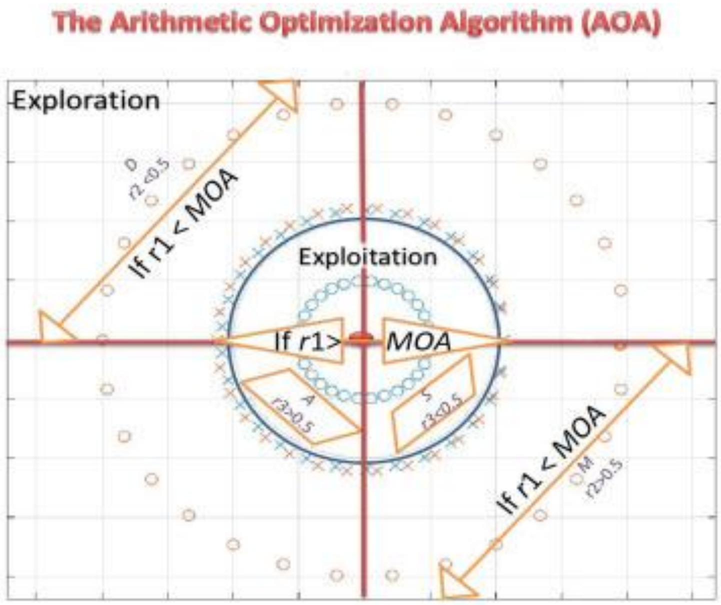


Fig. 7 Exploitative and Exploitatory Mechanism in AOA

2.4.2 Mathematical Calculation and Problem Formulation

Math optimizer accelerated (MOA) co-efficient is calculated as:

$$MOA\ (C\_Iter) = Min + C\_Iter \times \left(\frac{Max - Min}{M\_Iter}\right) \quad (1)$$

where MOA (C_Iter) denotes the function value at t^{th} iteration, which is calculated by Eq. (1). C_Iter denotes the current iteration, which is between 1 and the maximum number of iterations (M_Iter) [10-13]. Min and Max denote the minimum and maximum values of the accelerated function, respectively.

The equation governing the exploration phase is denoted by:

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) \div (MOP + \epsilon) \times \left((UB_j - LB_j) \times \mu + LB_j \right), & r_2 < 0.5 \\ best(x_j) \times MOP \times \left((UB_j - LB_j) \times \mu + LB_j \right), & otherwise \end{cases} \quad (2)$$

where $x_{i,j}(C_{Iter}+1)$ denotes the jth position of the ith solution at the current iteration, and $best(x_j)$ is the jth position in the best-obtained solution so far. ϵ is a small integer number, UBj and LBj denote the upper bound value and lower bound value of the jth position, respectively. This phase is selected if $r_1 > 0.5$.

$$\text{MOP (C\_Iter)} = 1 - \frac{C\_Iter^{1/\alpha}}{M\_Iter^{1/\alpha}} \quad (3)$$

where Math Optimizer probability (MOP) is a coefficient, MOP (C_Iter) denotes the function value at the t^{th} iteration, C_Iter denotes the current iteration and (M_Iter) denotes the maximum number of iterations. α is a sensitive parameter and defines the exploitation accuracy over the iterations.

The equation governing the exploitation phase is given by:

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) - MOP \times \left((UB_j - LB_j) \times \mu + LB_j \right), & r_3 < 0.5 \\ best(x_j) + \left((UB_j - LB_j) \times \mu + LB_j \right), & otherwise \end{cases} \quad (4)$$

This phase is selected when r_1 is not greater than 0.5.

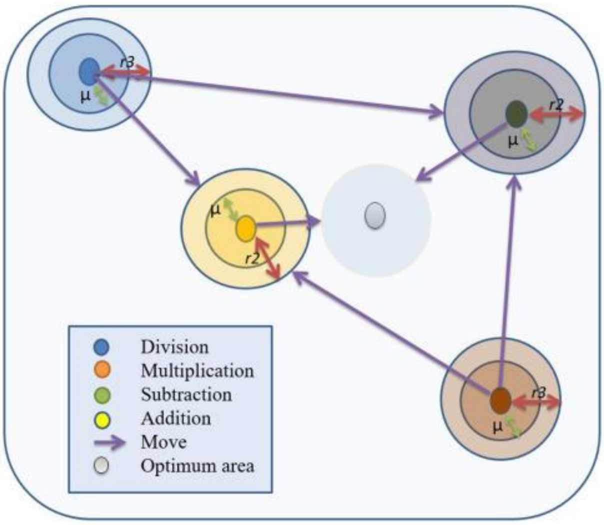


Fig. 8 Working of AOA

Fig.2 shows the working of AOA operators, that is if $r_1 > \text{MOA}$, the exploration phase is chosen where division and multiplication operators are used by the condition if $r_2 < 0.5$ then division operator is employed in the exploration phase and multiplication is neglected and vice-versa and if $r_1 <$ MOA , then the exploitation phase is chosen where subtraction and addition operator are used by the condition if $r_3 < 0.5$, then subtraction operator is applied in exploitation phase and addition operator is neglected and vice-versa.

2.4.3 Pseudo Code of AOA

Initialize the arithmetic optimization parameters α, μ.
Initialize the solution's positions.
while(C_Iter<M_Iter) do
    Calculate the fitness function of each solution.
    The best solution is determined.
    MOA and MOP values are updated.
    for (i=1 to Solutions) do
        for (j=1 to Positions) do
A random value is generated between [0.1] (r_1, r_2, r_3)
    if r_1>MOA then
    Exploration Phase
        if r_2>0.5 then
Apply Division Math Operator.
Update i^{th}solution using the 1st rule
        else
            Apply Multiplication Math Operator
            Update i^{th}solution using 2nd rule .
        endif
            else
                Exploitation Phase
            if r_3>0.5 then
Apply subtraction math operator.
Update i^{th}solutions' position using the first rule.
            else
Apply addition math operator.
Update i^{th}solutions' position using the second rule.
        endif
            endif
        endfor
    endfor
    C_Iter=C_Iter+1
endwhile

The pseudo code shows that at the starting of the AOA algorithm, the population positions are initialized by random values. If the current iteration is less than the total number of iterations then the best solution is determined from the calculated fitness functions. MOA and MOP values are updated. If $r_1$is less than MOA then the exploration phase is selected where the multiplication and division operator is operated depending on r_2 and otherwise the exploration phase is selected where the subtraction and addition operator is selected depending on r_3 and at last the best solution is determined.

2.4.4 Flowchart of AOA

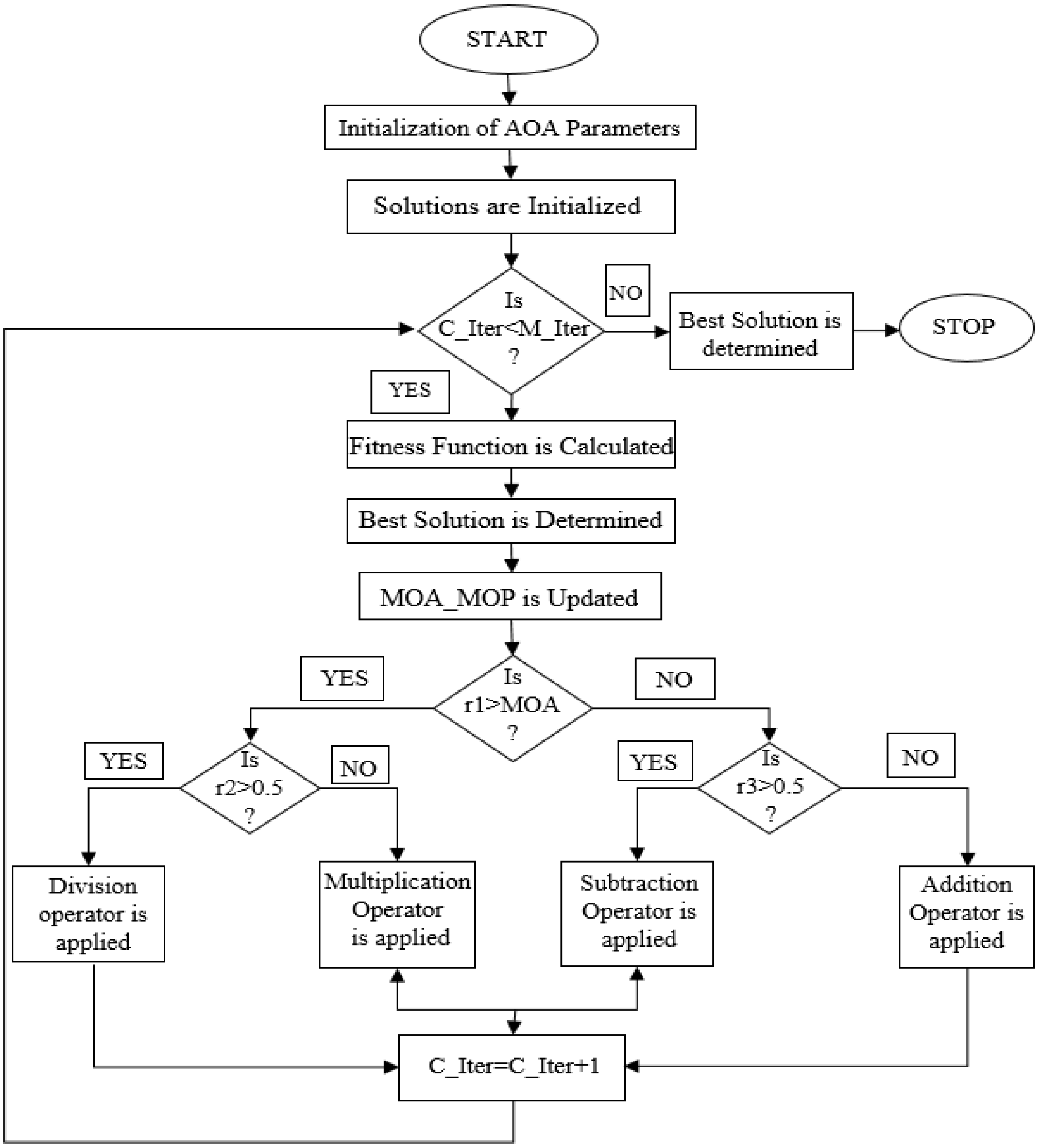


Fig. 9 Flowchart of AOA

2.5 Field Programmable Gate Array (FPGA)

2.5.1 Introduction

FPGA stands for Field Programmable Gate Array. FPGA [49-62] is an array of interconnected digital sub-circuits that implement common functions while also offering very high levels of flexibility [51-56] and they can be programmed as required by the users anytime which makes it more advantageous. Previously microcontrollers were used to serve the various purpose related to electronics, but since they were expensive and highly versatile, the instructions in a microcontroller are executed sequentially i.e. the process is inherently constrained, and so it's not possible to accomplish multiple tasks simultaneously. So to replace this problem FPGA [17] has been introduced.

2.5.2 Hardware Description Language (HDL)

FPGA programming can be implemented using a particular language better known as Hardware description language (HDL) and the two most common are VHDL and Verilog. HDL [57-59] is a computer language that is fundamentally different from the software programming language, as software programming languages are executed sequentially whereas HDL language is more like a schematic that uses text to introduce components and create interconnections.

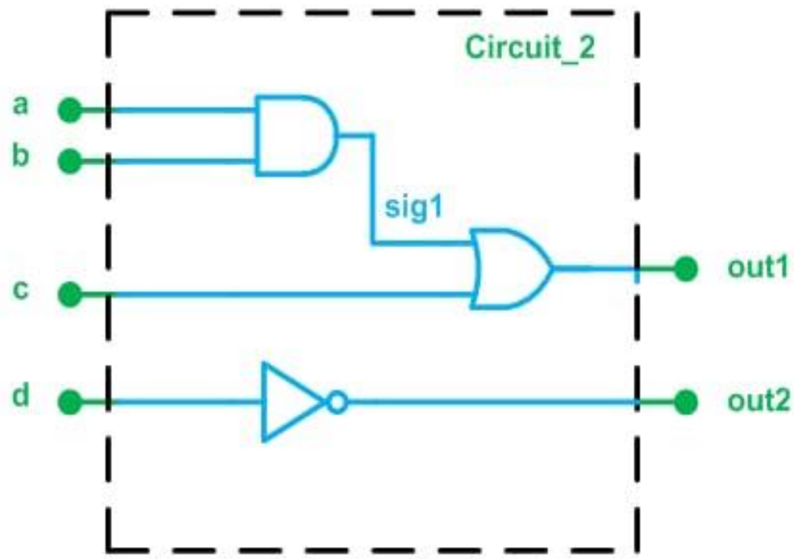


Fig. 10 Schematic diagram of a Boolean function using HDL

The HDL language includes a text description consisting of operators, expressions, statements, inputs, and outputs. The HDL code can be written in three styles [54]:-

i. Structural description language describes the circuit structure in terms of logic gates and the interconnects wiring between the logic gates.

ii. Dataflow description describes the transfer of data from input to output.

iii. The behavioral description describes the behavior of the design in terms of circuit or system behavior using the algorithm.

2.5.3 Architecture

The general FPGA structure consists of three types of modules. They are configurable logic blocks (CLB), I/O blocks and switch matrix/interconnection wires. The functions of an FPGA architecture module can be described as [60-62]:-

i. Configurable Logic Block (CLB) includes digital logic, input and outputs.

ii. Interconnections provide direction between the logic blocks to implement the user logic.

iii. I/O pads used for the outside world to communicate with different applications.

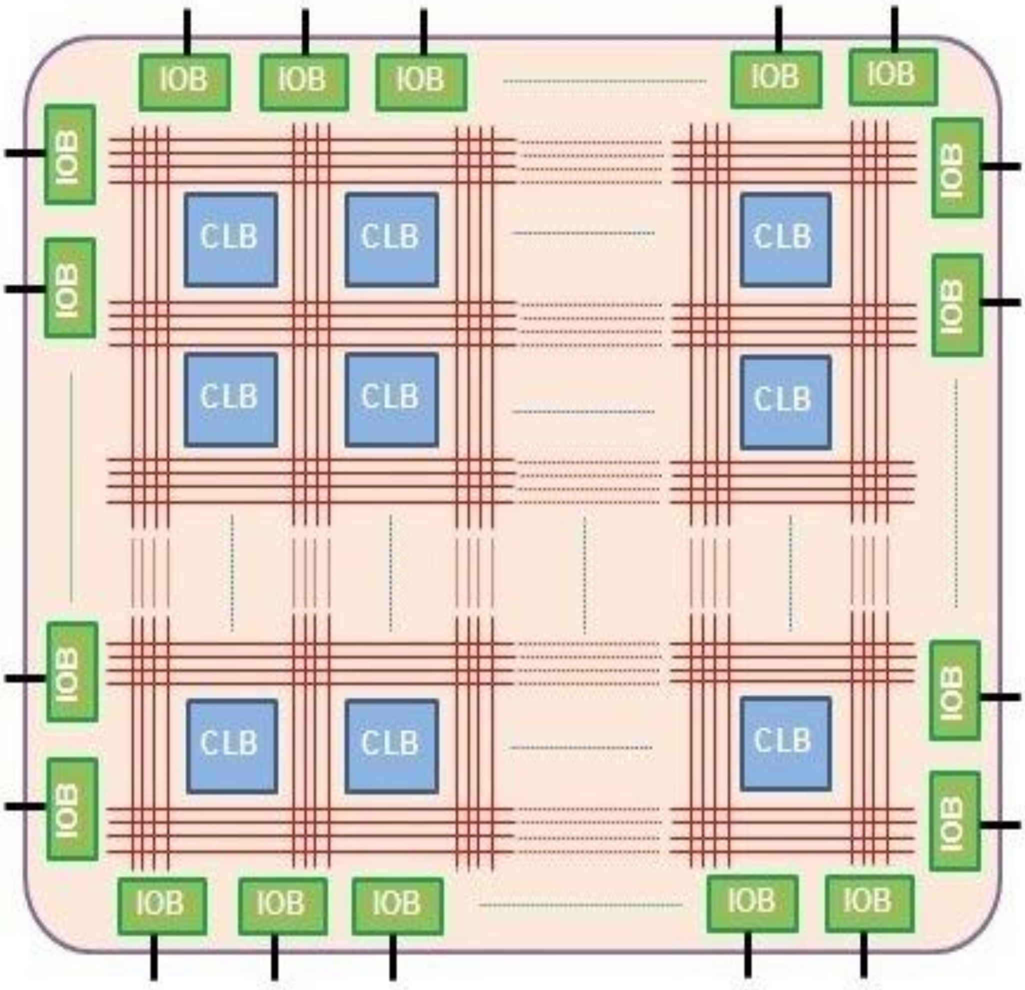


Fig. 11 FPGA Architecture

2.6 Steps of Implementation of the design

Step 1:- The following Matlab Code is compiled and run.

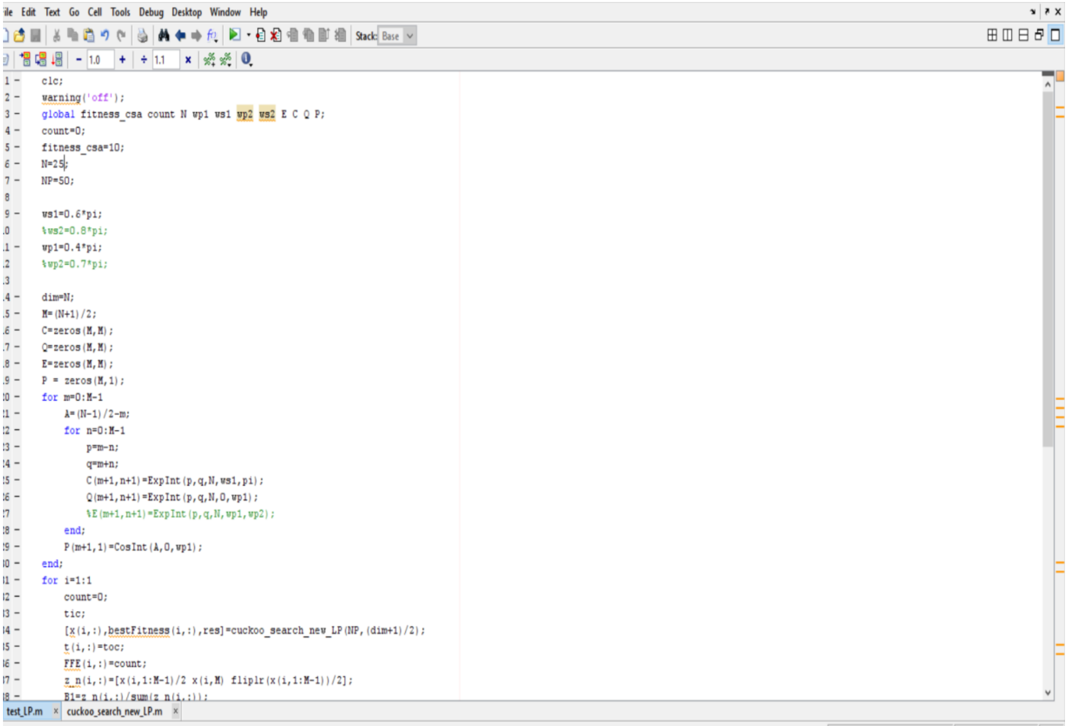


Fig. 12 MATLAB code compiled and Run

Step2:- Matlab coefficients are the obtained from Matlab figure edit tool.

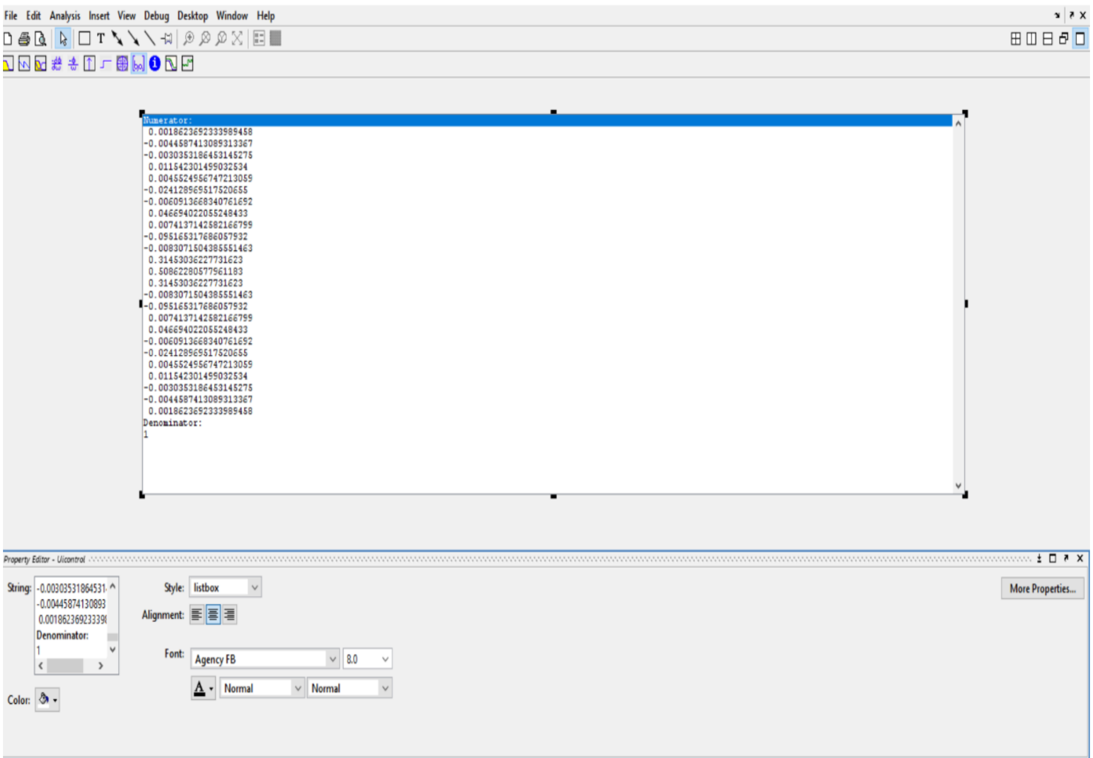


Fig. 13 MATLAB coefficients are obtained

Step3:- The ‘fdatool’ is opened and the coefficients are placed there with denominator as 1 and the style is chosen as Direct Form FIR.

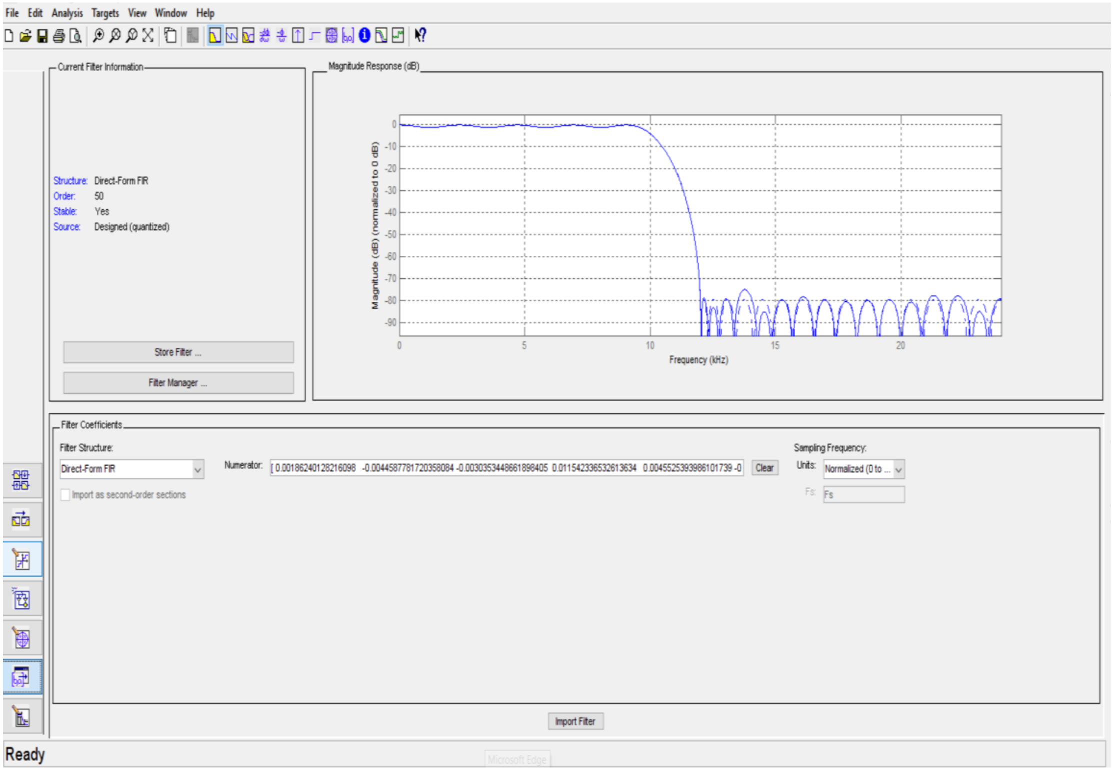


Fig. 14 MATLAB coefficients are placed in FDATOOL and Direct Form FIR is chosen

Step4:- The Quantization tool is opened and the necessary settings are done.

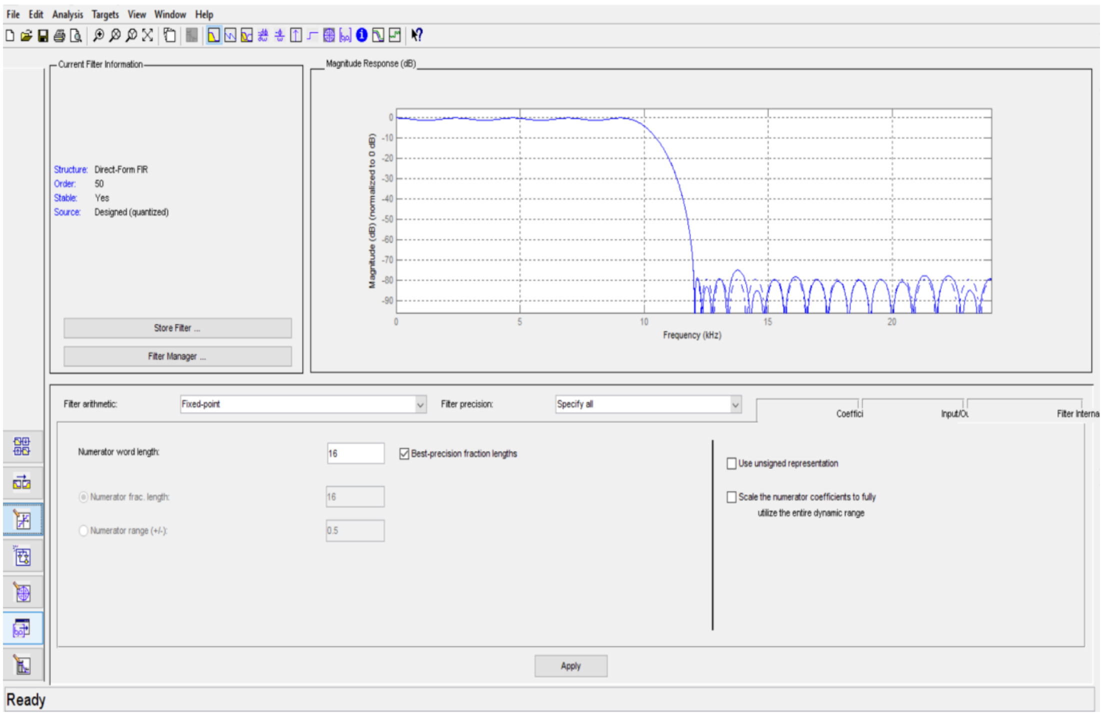


Fig. 15 Quantization Tool is opened and necessary settings are done

Step5:- Then Target is chosen from where HDL coder is selected and the necessary setting are done.

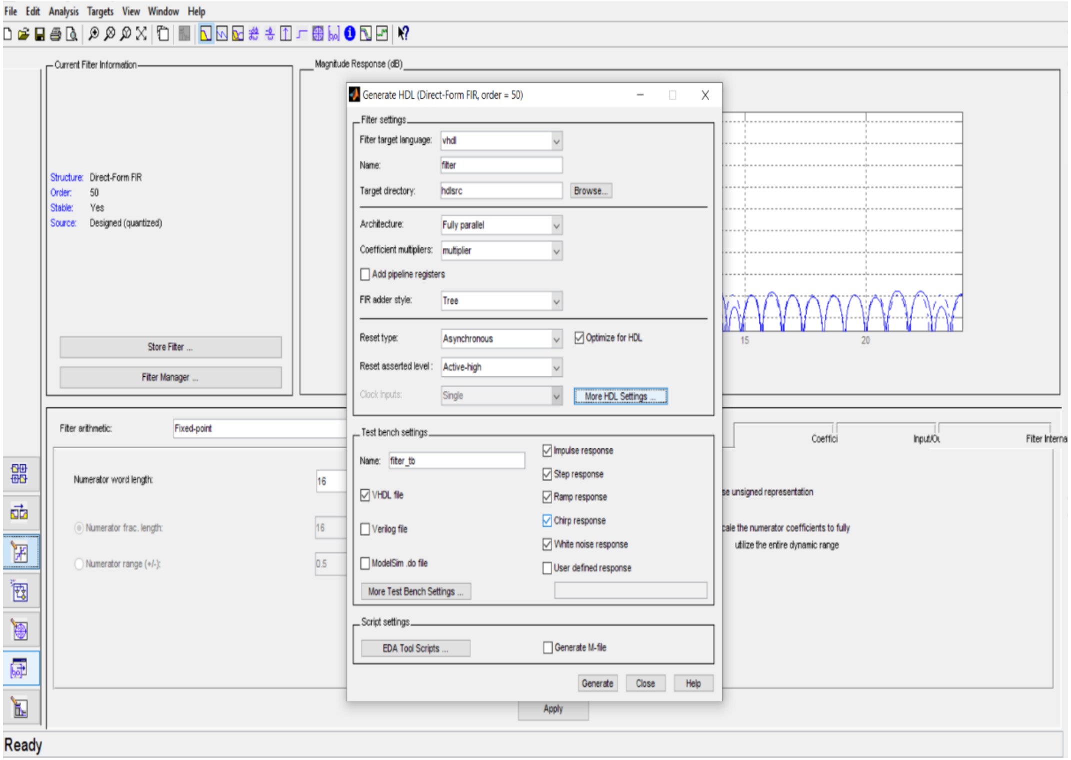


Fig. 16 Target is chosen as HDL Coder

Step6:- The test bench folder is formed which is opened in the Xilinx folder under a new project.

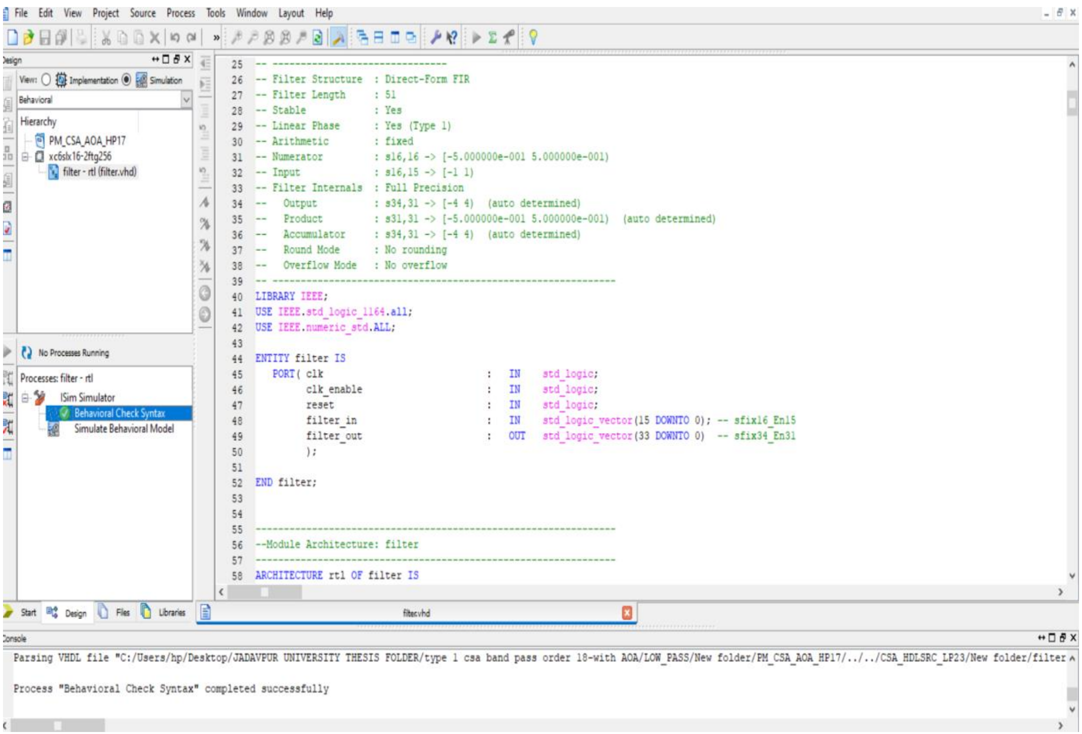


Fig. 17 Test Bench Folder is formed using XILINX Folder

Step7:- There it is simulated using behavioral check syntax and timing diagram is obtained.

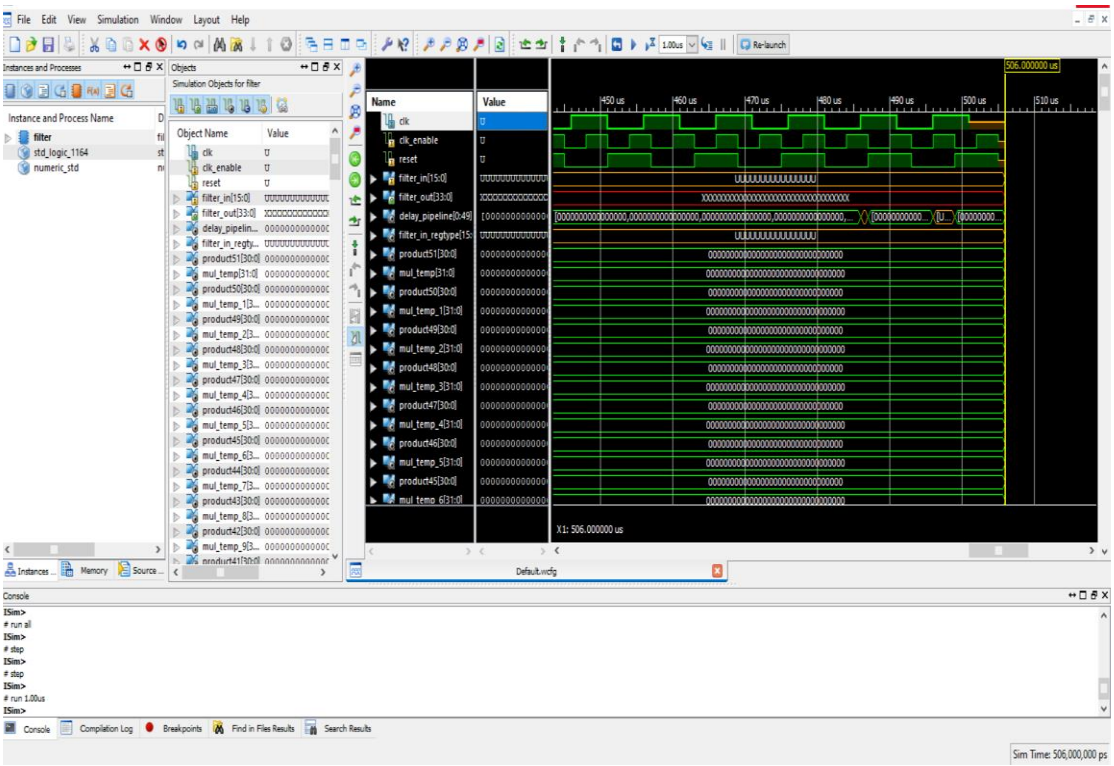


Fig. 18 Simulation is done using Behavioral Check Syntax

Step8:- Then it is simulated and the design summary and the power report is obtained.

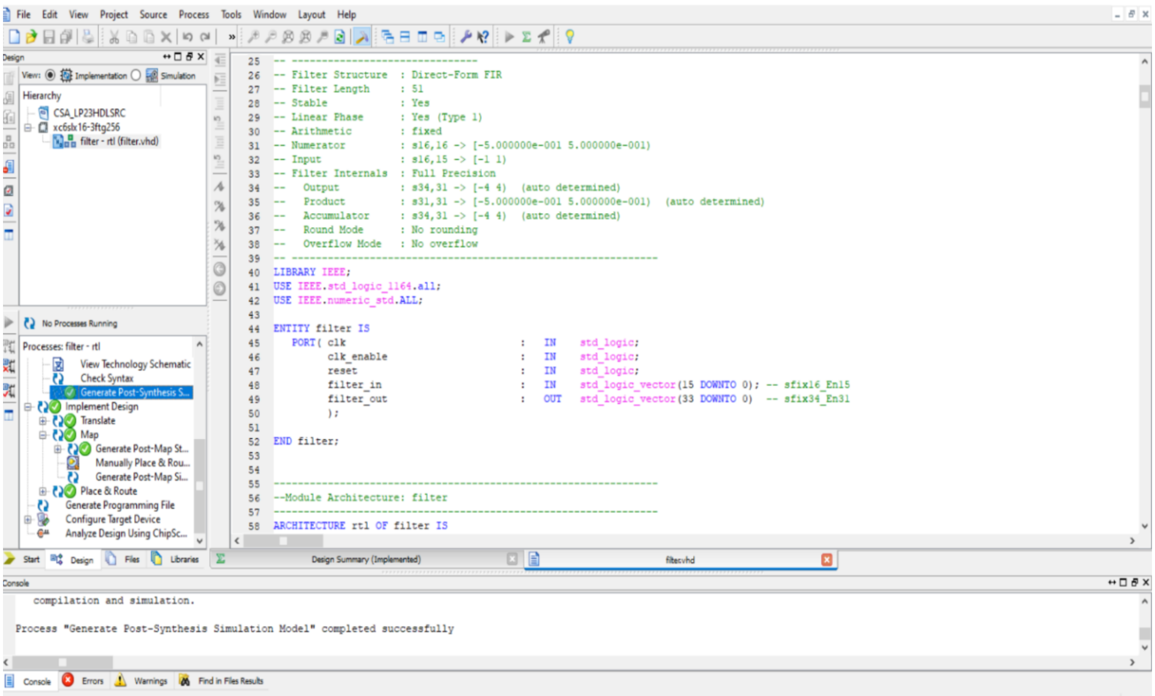
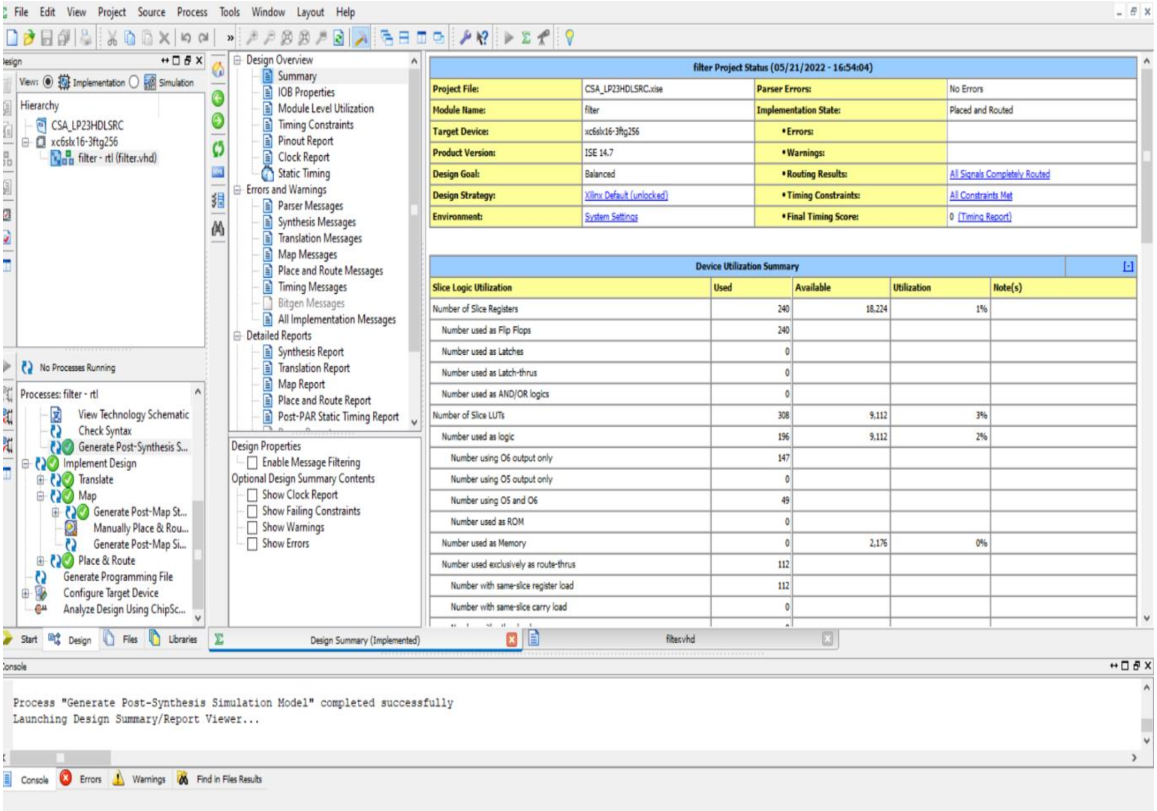


Fig. 19 Design Summary and Power Report is obtained



2.7 Summary

This chapter starts with brief descriptions of FIR filters. Then the chapter proceeds with a sufficiently detailed presentation of the design methodologies of FIR filters in the form of eVolutionary optimization algorithms such as AOA [23-25] and its associated pseudo code and flowchart have also been presented in this section. Again few steps related to the implementation of the design on FPGA have also been presented.

Chapter 3

Simulation Results and Analysis of FIR filter Design using AOA

3.1 Introduction

This chapter presents the problem formulation of Type-1 low-pass, band-pass, and high-pass filter.

3.2 Problem Formulation of FIR Filter

The system function of the FIR [2] filter is represented by:

$$H(z) = \sum_{n=0}^{M-1} h(n)\, z^{-n} \qquad n=0,1\ldots M-1 \tag{5}$$

The equation consists of (M-1) poles at origin and (M-1) zeros. Based on the coefficients of h(n) a filter can exhibit different types of magnitude responses.

3.3 Error Function Representation

Practically the design of the desired filter can be achieved by changing the degrees of the favorable outcomes and minimizing the deviation of the desired magnitude response from the ideal. The weighted difference in stop-band and pass-band primarily defines the error function which is defined as:

$$E(\omega) = W(\omega)[H_d(e^{j\omega}) - H_a(e^{j\omega})] \tag{6}$$

The error function E(ω) , given by Parks-McClellan (PM) is represented in Eq. (6), where W(ω), $H_d(e^{j\omega})$, $H_a(e^{j\omega})$ are the weight vector, desired, and approximated frequency response respectively. Weight function W(ω) modulates the minimization of error.

In error function $E(\omega)$, the ratio between peak ripple at pass-band (δ_p) and stop-band (δ_s) cannot take different values. Thus, in order to overcome the flaws in this function, a modified error function is used:

$$U = \max_{\omega \le \omega_p}(|E(\omega)| - \delta_p) + \max_{\omega \ge \omega_s}(|E(\omega)| - \delta_s) \tag{7}$$

where ω_p, ω_s, δ_p and δ_s are the desired filter specifications.

3.4 Type-1 Linear Phase FIR Filter

The frequency the response function of FIR filter is represented by:

$$H\left(e^{j\omega}\right) = \sum_{n=0}^{M-1} h[n]e^{-j\omega n} \quad , \qquad -\pi \leq \omega \leq \pi \tag{8}$$

Now linear phase constraint is described by:

$$angle\ H\left(e^{j\omega}\right) = -\tau_{\emptyset}\omega\ , -\pi \leq \omega \leq \pi \tag{9}$$

Here $\tau_{\emptyset}$ is a constant-phase delay. Now for type 1 filter h[n] has to be symmetrical:

$$h(n) = h(M-1-n), 0 \leq n \leq (M-1) \text{ with } \tau_{\emptyset} = \frac{M-1}{2} \tag{10}$$

where h(n) shows symmetry about $\tau_{\emptyset}$ is the index of symmetry. The value of M , in Eq.(5), can take even or odd integer values in case of Type 1 and Type 2 filters respectively. The frequency of the Type 1 filter is:

$$H\left(e^{j\omega}\right) = \left[\sum_{n=0}^{(M-1)/2} a(n)cos\omega n\right] e^{-j\omega(M-1)/2} \tag{11}$$

3.5 Objective Function Formulation

3.5.1 Low-Pass Filter

The objective function for the low-pass [29-32] filter is considered as:

$$\emptyset_{LP} = \beta * E_S + (1-\beta) * E_P, \quad 0 \leq \beta \leq 1$$

(11)

where E_p , E_S are calculated following Eqs. (12)-(13).

$$E_P = \frac{1}{\pi}\int_0^{\omega_p}(1-H(\omega))^2 d\omega = \frac{\omega_p}{\pi} - 2b_1^T P_1 + b_1^T Q b_1 \tag{12}$$

$$E_S = \frac{1}{\pi}\int_{\omega_s}^{\pi}(H(\omega))^2 d\omega = b_1^T C_1 b_1 \tag{13}$$

where E_P represents pass-band error, E_S is the stop-band error respectively for low-pass filter.

Here $H(\omega) = b^T C(\omega), \qquad b = [\, b_{1,} b_{2,} \ldots \ldots, b_{N/2}]^T$.

$$P = \frac{1}{\pi}\int_0^{\omega_p} \cos(A\omega)\, d\omega \tag{14}$$

$$Q = \frac{1}{\pi}\int_0^{\omega_p} \cos(A\omega)\cos(B\omega)\, d\omega \tag{15}$$

In case of pass-band error, the value of C can be measured from the formula of the following:

$$C(m,n) = \frac{1}{\pi}\int_{\omega_s}^{\pi} \cos(A\omega)\cos(B\omega)\, d\omega \tag{16}$$

where $A = \frac{N-1}{2} - m$, $\quad m = 0,1, \dots \dots, (M-1)$

and $B = \frac{N-1}{2} - n$, $\quad n = 0,1, \dots \dots, (M-1)$, and H(ω) is the magnitude response of the filter. Thus the objective of our design is to minimize the cost function for low-pass filter as given in Eq. (11)

3.5.2 Band-pass Filter

The objective function for the band-pass [43-46] filter is considered as:

$$\emptyset_{BP} = \alpha * E_S + (1-\alpha) * E_P, \quad 0 \le \alpha \le 1 \tag{17}$$

where E_p , E_S are calculated following Eqs. (18)-(20).

$$E_P = \frac{1}{\pi}\int_{\omega_{p1}}^{\omega_{p2}} (1 - H(\omega))^2 d\omega = \frac{\omega_{p2}-\omega_{p1}}{\pi} - 2b_1^T P_1 + b_1^T Q b_1 \tag{18}$$

$$E_{S1} = \frac{1}{\pi}\int_0^{\omega_{s1}} (H(\omega))^2 d\omega = b_1^T C_1 b_1 \tag{19}$$

$$E_{S2} = \frac{1}{\pi}\int_{\omega_{s2}}^{\pi} (H(\omega))^2 d\omega = b_1^T C_2 b_1 \tag{20}$$

where E_P represents pass-band error, E_{S1} *and* E_{S2} are the first and second stop-band error respectively for band-pass filter.

Here $H(\omega) = b^T C(\omega)$, $\quad b = [\, b_{1,} b_{2,} \dots \dots, b_{N/2}]^T$

$$P = \frac{1}{\pi}\int_0^{\omega_p} \cos(A\omega)\, d\omega \tag{21}$$

$$Q = \frac{1}{\pi}\int_0^{\omega_p} \cos(A\omega)\cos(B\omega)\, d\omega \tag{22}$$

In case of pass-band error, the value of C can be measured from the formula of the following:

$$C(m,n) = \frac{1}{\pi}\int_{\omega_s}^{\pi} \cos(A\omega)\cos(B\omega)\, d\omega \qquad (23)$$

where A $= \frac{N-1}{2} - m$, $\quad m = 0,1, \ldots \ldots, (M-1)$

and $B = \frac{N-1}{2} - n$, $\quad n = 0,1, \ldots \ldots, (M-1)$, and H(ω) is the magnitude response of the filter. Thus the objective of our design is to minimize the cost function for low-pass filter as given in Eq. (17)

3.5.3 High-pass Filter

The objective function for high pass filter [36-39] is considered as:

$$\emptyset_{HP} = \gamma * E_S + (1-\gamma) * E_P, \quad 0 \le \gamma \le 1 \qquad (24)$$

where E_p , E_S are calculated following Eqs. (24)-(25).

$$E_P = \frac{1}{\pi}\int_{\omega_p}^{\pi} (1 - H(\omega))^2 d\omega = \frac{\pi - \omega_{p1}}{\pi} - 2b_1^T P_1 + b_1^T Q b_1 \qquad (25)$$

$$E_S = \frac{1}{\pi}\int_{0}^{\omega_s} (H(\omega))^2 d\omega = b_1^T C_1 b_1 \qquad (26)$$

where E_P represents pass-band error, E_S is the stop-band error respectively for the high-pass filter.

Here $H(\omega) = b^T C(\omega), \quad b = [\, b_{1,} b_{2,} \ldots \ldots, b_{N/2}]^T$

$$P = \frac{1}{\pi}\int_{\omega_p}^{\pi} \cos(A\omega)\, d\omega \qquad (27)$$

$$Q = \frac{1}{\pi}\int_{\omega_p}^{\pi} \cos(A\omega)\cos(B\omega)\, d\omega \qquad (28)$$

In case of pass-band error, the value of C can be measured from the formula of the following:

$$C(m,n) = \frac{1}{\pi}\int_{0}^{\omega_s} \cos(A\omega)\cos(B\omega)\, d\omega \qquad (29)$$

where A $= \frac{N-1}{2} - m$, $\quad m = 0,1, \ldots \ldots, (M-1)$

and $B = \frac{N-1}{2} - n$, $\quad n = 0,1, \ldots \ldots, (M-1)$, and H(ω) is the magnitude response of the filter. Thus the objective of our design is to minimize the cost function for high-pass filter as given in Eq. (23) .

3.6 Simulation Results and Analysis

In this section the implementation of three types of filter with the following parameter settings are discussed:

We assume MOP_Max=1, MOP_Min =0.2, alpha =5.The implementation work is completed using MATLAB 2007B, XILINX ISE Design suite 14.7 tool, and XILINX XC6SLX16-2FTG256C FPGA[49-52] hardware. The FIR filter has been designed as a minimization problem and AOA [18-20] is applied to optimize the cost function. The magnitude responses of the filters of their respective orders of 15 and 23 are shown in the below figures. The magnitude responses of the FIR filter, are for PM, CSA and AOA, which describes how AOA shows better response than CSA and PM. From the same, we can infer the values of attenuation at stop-band (As) and the pass-band ripple (Rp) as provided in respective tables. For the filters of order (N) 15, the population size (NP) is noted as 70 and the number of functional eVolution (NOFE) is 50,000. For order (N) 23, the population size (NP) is noted as 70 and the number of functional eVolution (NOFE) is 150,000.

TABLE 1. SIMULATION RESULTS FOR LPF ($\omega_p = 0.4\pi, \omega_s = 0.6\pi$)

N	NP	NOFE	Method	A_S(in dB)	R_P(in dB)
15	70	50000	PM	31.8331	0.2549
			CSA	32.4507	0.0871
			AOA	34.5861	0.0598
23	70	100000	PM	40.956	0.0956
			CSA	42.3012	0.0795
			AOA	43.9447	0.0233

The data presented in Table 1 represent the orders of the low-pass filter and their corresponding population size using three algorithms that are Park-McClellan (PM) algorithm, Cuckoo search Algorithm (CSA), and Arithmetic Optimization Algorithm (AOA). The attenuation values (A_s) and the stop band ripple values (R) derived by using AOA is found to be much better than those derived by using CSA and PM method.

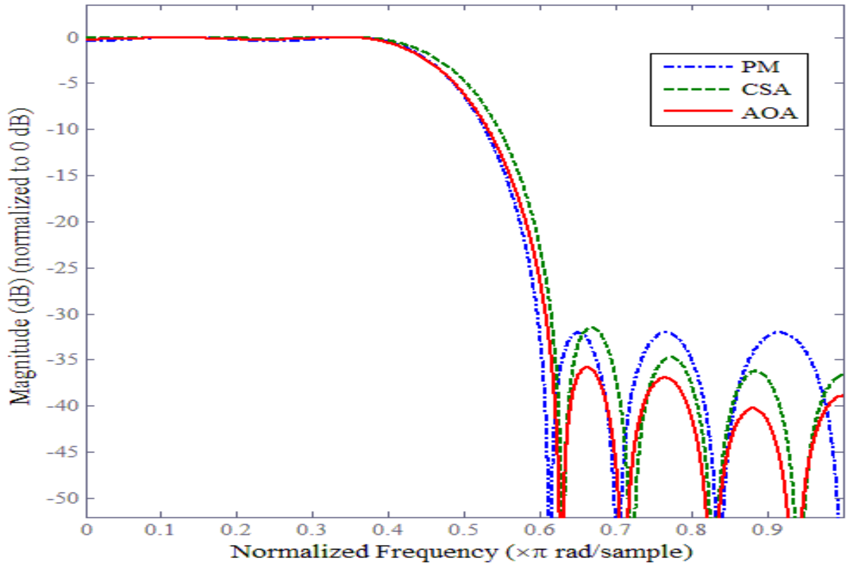


Fig. 20 Magnitude Response of order 15 Low Pass Filter

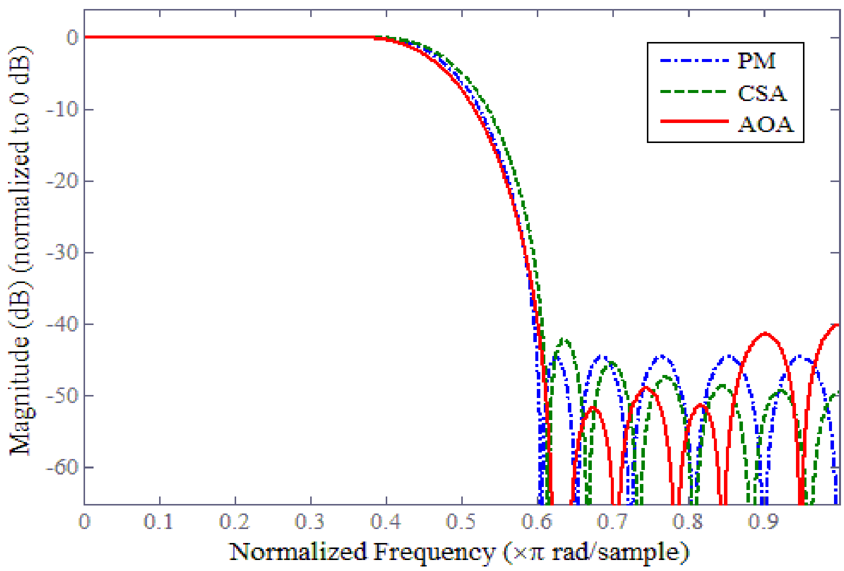


Fig. 21 Magnitude Response of order 23 Low Pass Filter

The plots presented in Fig. 20 and Fig.21 represent the magnitude response of low pass filter using three algorithms namely Park- McClellan method, Cuckoo Search algorithm and Arithmetic Optimization algorithm. The magnitude response determined by AOA is much better than magnitude phase derived by CSA and PM, because in the magnitude phase derived by using AOA there is a sharp rise of the transition band as a result the value of the attenuation band is more and the number of ripples in the stop band is less.

TABLE 2. SIMULATION RESULTS FOR BPF ($\omega_{p1} = 0.4\pi, \omega_{P2=}, \omega_{s1} =, \omega_{s2} = 0.6\pi$)

N	NP	NOFE	Method	A_S(in dB)	R_P(in dB)
15	70	50000	PM	15.878	0.8760
			CSA	19.7680	0.7260
			AOA	23.9699	0.6108
23	70	100000	PM	20.9840	0.6785
			AOA	23.4802	0.5620
			AOA	28.5613	0.3993

The data presented in Table 2 represent the orders of the band-pass filter and their corresponding population size using three algorithms that are Park-McClellan (PM) algorithm, Cuckoo search Algorithm (CSA) and Arithmetic Optimization Algorithm (AOA). The attenuation values (A_s) and the stop band ripple values (R) derived by using AOA is found to be much better than derived by using CSA and PM method.

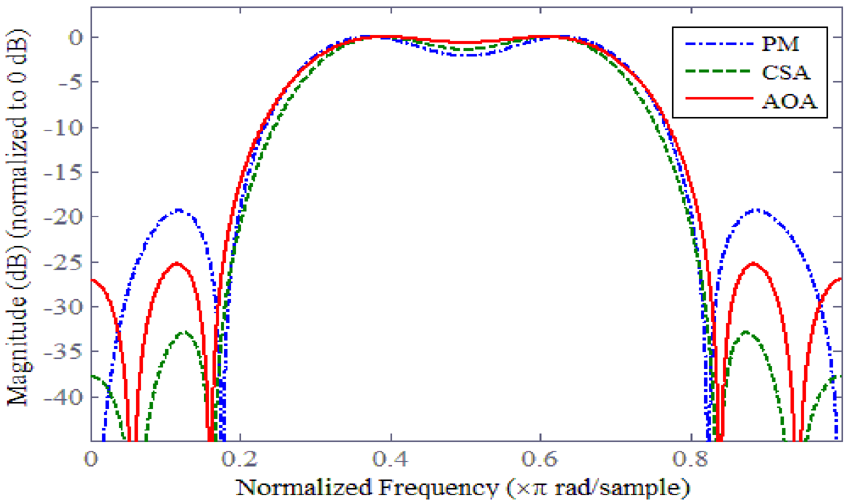


Fig. 22 Magnitude Response of 15 order Band Pass Filter

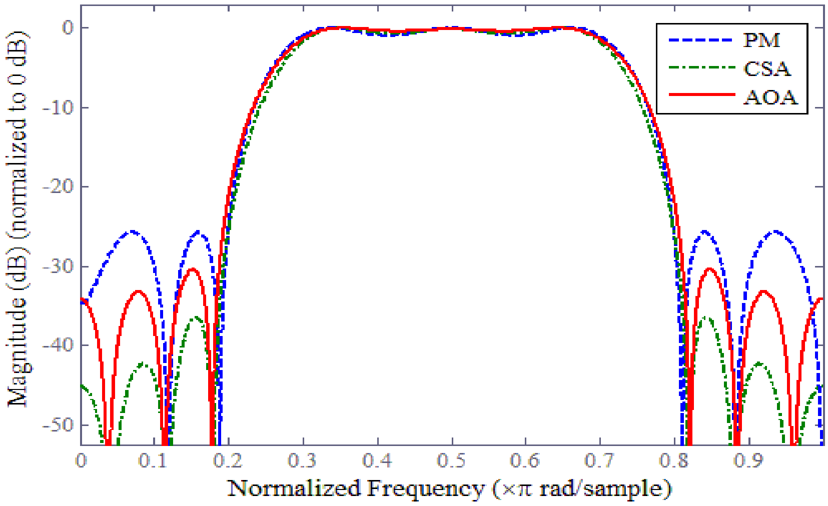


Fig. 23 Magnitude Response of 23 order Band Pass Filter

The plots presented in Fig.22 and Fig.23 represents the magnitude response of band-pass filter using three algorithms namely Park- McClellan method, Cuckoo Search algorithm, and Arithmetic Optimization algorithm. The magnitude response determined by AOA is much better than the magnitude phase derived by CSA and PM, because in the magnitude phase derived by using AOA there is a sharp rise of the transition band as a result the value of the attenuation band is more and the number of ripples in the stop band is less.

TABLE 3. SIMULATION RESULTS FOR HPF ($\omega_p = 0.6\pi, \omega_s = 0.4\pi$)

N	NP	NOFE	Method	A_S(in dB)	R_P(in dB)
15	70	50000	PM	32.9706	0.2547
			CSA	36.0328	0.2032
			AOA	41.2413	0.1679
23	70	100000	PM	43.9873	0.1945
			CSA	46.2630	0.1232
			AOA	49.4328	0.0781

The data presented in Table 3 represent the orders of the high-pass filter and their corresponding population size using three algorithms that are Park-McClellan (PM) algorithm, Cuckoo Search Algorithm (CSA), and Arithmetic Optimization Algorithm (AOA). The attenuation values (A_s) and the stop band ripple values (R) derived by using AOA is found to be much better than those derived by using CSA and PM method.

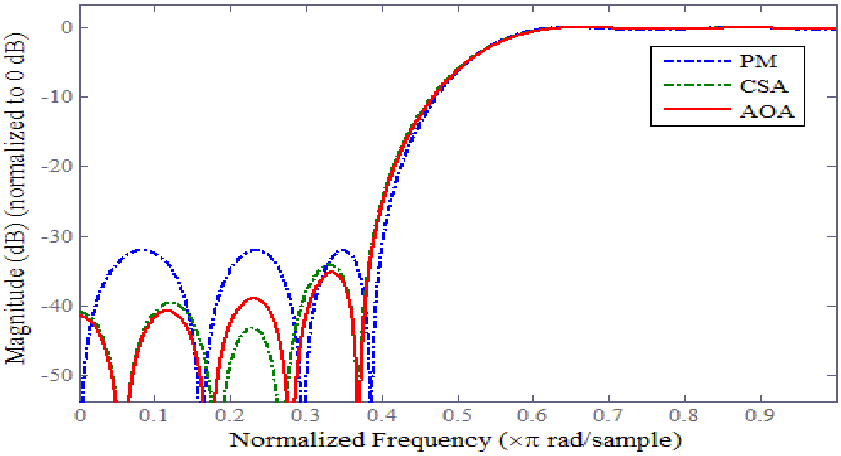


Fig. 24 Magnitude Response of 15 order High Pass Filter

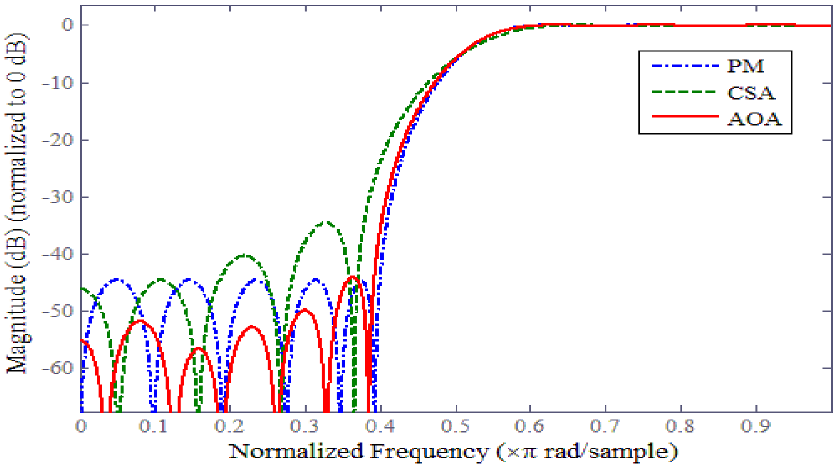


Fig. 25 Magnitude Response of 23 order High Pass Filter

The plots presented in Fig.24 and Fig.25 are representing the magnitude response of the high-pass filter using three algorithms namely the Park-McClellan method, Cuckoo Search algorithm, and Arithmetic Optimization algorithm. The magnitude response determined by AOA is much better than the magnitude phase derived by CSA and PM, because in the magnitude phase derived by using AOA there is a sharp rise of the transition band as a result the value of the attenuation band is more and the number of ripples in the stop-band is less.

TABLE 4. DEVICE UTILIZATION FOR LPF

		Components					
		Slice Registers	Slice UTs	No. of FF	Bonded IOBs	No. of BUFG	No. of DSP481A 's
N=15	Available	18224	9112	854	186	16	32
	PM	368	485	0	35	1	24
	CSA	360	410	75	35	1	24
	AOA	352	395	69	35	1	24
N=23	Available	18224	9112	854	186	16	32
	PM	240	614	0	35	1	24
	CSA	235	450	109	35	1	24
	AOA	224	398	85	35	1	24

The data presented in Table 4 represents the Device Utilization for Low-Pass Filter, which consists of slice registers, slice UT's, no. of FF, bonded IOBs, No. of BUFG, and no. of DSP481A's. The data noted in the corresponding column using PM, CSA, and AOA algorithm for different orders, shows the efficiency and minimum space requirement for AOA than using CSA and PM algorithms.

TABLE 5. DEVICE UTILIZATION FOR BPF

		Components					
		Slice Registers	Slice UTs	No. of FF	Bonded IOBs	No. of BUFG	No. of DSP481 A's
N=15	Available	18224	9112	854	186	16	32
	PM	368	476	0	35	1	24
	CSA	360	350	72	35	1	24
	AOA	352	330	61	35	1	24
N=23	Available	18224	9112	854	186	16	32
	PM	240	597	0	35	1	24
	CSA	235	419	112	35	1	24
	AOA	224	378	95	35	1	24

The data presented in Table 5 represents the Device Utilization for Band-Pass Filter, which consists of slice registers, slice UT's, no. of FF, bonded IOBs, No. of BUFG, and no. of DSP481A's. The data noted in the corresponding column using PM, CSA, and AOA algorithms for different orders, shows the efficiency and minimum space requirement for AOA than using CSA and PM algorithms. Due to the minimum space requirement by Arithmetic Optimization Algorithm, the device components are less which leads to more economical than using the cuckoo search algorithm and Park-McClellan algorithm.

TABLE 6. DEVICE UTILIZATION FOR HPF

		Components					
		Slice Registers	Slice UTs	No. of FF	Bonded IOBs	No. of BUFG	No. of DSP481 A's
N=15	Available	18224	9112	854	186	16	32
	PM	368	485	0	35	1	24
	CSA	360	330	80	35	1	24
	AOA	352	315	71	35	1	24
N=23	Available	18224	9112	854	186	16	32
	PM	240	614	0	35	1	24
	CSA	235	415	105	35	1	24
	AOA	224	366	90	35	1	24

The data presented in Table 6 represents the Device Utilization for High-Pass Filter, which consists of slice registers, slice UT's, no. of FF, bonded IOBs, No. of BUFG, and no. of DSP481A's. The data noted in the corresponding column using PM, CSA, and AOA algorithms for different orders, shows the efficiency and minimum space requirement for AOA than using the CSA and PM algorithms. Due to minimum space requirement by, Arithmetic Optimization Algorithm the device components are less which leads to more economical than using the cuckoo search algorithm and Park- McClellan algorithm.

TABLE 7. POWER UTILIZATION FOR LPF

Type of Power	N= 15			N= 23		
	PM	CSA	AOA	PM	CSA	AOA
Supply Power(mW)	20.38	20.39	19.62	20.47	20.38	20.32
Dynamic Power(mW)	0.49	0.47	0.42	0..49	0.47	0.42
Static Power(mW)	19.89	19.91	19.20	19.98	19.91	19.90

The above Table 7 represents the power utilization table of Low-pass filter using three algorithms, the Park-McClellan method (PM), the Cuckoo Search Algorithm (CSA), and Arithmetic optimization Algorithm (AOA) of orders 15 and 23. The table consists of three rows, supply power (mW), Dynamic Power (mW), and Static Power (mW). The data in the corresponding rows prove that the device needs the least amount of power in the case of AOA than the data derived by using CSA and PM. Thus less is the amount of power required less is the amount of heat generated.

TABLE 8. POWER UTILIZATION FOR BPF

Type of Power	N= 15			N= 23		
	PM	CSA	AOA	PM	CSA	AOA
Supply Power(mW)	20.38	20.39	19.62	20.47	20.38	20.32
Dynamic Power(mW)	0.49	0.47	0.42	0..49	0.47	0.42
Static Power(mW)	19.89	19.91	19.20	19.98	19.91	19.90

The above Table 8 represents the power utilization table of the Band-pass filter using three algorithms, the Park-McClellan method (PM), Cuckoo Search Algorithm (CSA), and Arithmetic optimization Algorithm (AOA) of orders 15 and 23. The table consists of three rows , supply power (mW), Dynamic Power (mW), Static Power (mW). The data in the corresponding rows prove that the deice needs least amount of power in case of AOA than the data derived by using CSA and PM. Thus less is the amount of power required less is the amount of heat generated.

TABLE 9. POWER UTILIZATION FOR HPF

Type of Power	N= 15			N= 23		
	PM	CSA	AOA	PM	CSA	AOA
Supply Power(mW)	20.38	20.39	19.62	20.47	20.38	20.32
Dynamic Power(mW)	0.49	0.47	0.42	0..49	0.47	0.42
Static Power(mW)	19.89	19.91	19.20	19.98	19.91	19.90

The above Table 9 represents the power utilization table of the High-pass filter using three algorithms, the Park-McClellan method (PM), the Cuckoo search Algorithm (CSA), and the Arithmetic optimization Algorithm (AOA) of orders 15 and 23. The table consist of three rows, supply power (mW), Dynamic Power (mW), Static Power (mW). The data in the corresponding rows prove that the device needs the least amount of power in case of AOA than the data derived by using CSA and PM. Thus less is the amount of power required less is the amount of heat generated.

3.6.1 Comparison of FIR filters using their optimized coefficients

The MATLAB co-efficient obtained by using the three algorithms on FIR filters of order 15 and 23 are presented and compared in the following tables.

TABLE 10. Optimized Coefficients for FIR LP Filter of order 15 using PM, CSA, AOA

h(n)	PM	CSA	AOA
h(1)=h(16)	-0.018030	0.006621	-00.01173
h(2)=h(15)	1.12E-11	-0.016790	-00.02362
h(3)=h(14)	0.040688	-0.008330	0.028889
h(4)=h(13)	-0002E-06	0.040223	0.035582
h(5)=h(12)	-0.09062	0.010527	-00.05394
h(6)=h(11)	002E-06	-0.090720	-00.08198
h(7)=h(10)	0.311976	-0.012100	00.145973
h(8)=h(9)	0.499547	0.312948	00.447995
h(0)	0.311976	0.512663	00.447995

The Table 10 represents the optimized coefficients of FIR Low-pass filter of order 15 obtained by using PM, CSA, AOA. The coefficients are symmetrical about h(0).

TABLE 11. Optimized Coefficients for FIR BP Filter of order 15 using PM, CSA, AOA

h(n)	PM	CSA	AOA
h(1)=h(16)	-000.0186	1.951E-09	00000000
h(2)=h(15)	0.058334	3.978E-10	0.077048
h(3)=h(14)	00.08599	0.0813868	00000000
h(4)=h(13)	00.05289	-2.836E-10	0.009856
h(5)=h(12)	0.047316	0.0240265	0000000
h(6)=h(11)	00.17537	-1.098E-09	-0.308980
h(7)=h(10)	00.27968	0.309104	-02.1E-05
h(8)=h(9)	0.334916	4.386E-10	0.490757
h(0)	0.334916	00.475446	-02.1E-05

The Table 11 represents the optimized coefficients of FIR Band-pass filter of order 15 obtained by using PM, CSA, AOA. The coefficients are symmetrical about h(0).

TABLE 12. Optimized Coefficients for FIR HP Filter of order 15 using PM, CSA, AOA

h(n)	PM	CSA	AOA
h(1)=h(16)	0.011730	0.017051	0.01683
h(2)=h(15)	0.028889	0.001110	4.94E-13
h(3)=h(14)	0.035580	0.039620	-0.040200
h(4)=h(13)	0.053940	0.001822	-05.1E-05
h(5)=h(12)	0.081985	0.091181	0.090727
h(6)=h(11)	0.145973	0.001650	5.32E-05
h(7)=h(10)	-0.448000	0.313490	-00.31289
h(8)=h(9)	0.447995	0.500833	0.499929
h(0)	0.145970	0.313490	-00.31289

The Table 12 represents the optimized coefficients of FIR High-pass filter of order 15 obtained by using PM, CSA, AOA. The coefficients are symmetrical about h(0).

TABLE 13. Optimized Coefficients for FIR LP Filter of order 23 using PM, CSA, AOA

h(n)	PM	CSA	AOA
h(1)=h(24)	-0.00320	0.001862	00.00448
h(2)=h(23)	0.006360	0.004460	0.003171
h(3)=h(22)	0.008616	0.003040	0.009812
h(4)=h(21)	0.010849	0.011542	0.004040
h(5)=h(20)	0.015920	0.004553	-0.023900
h(6)=h(19)	-0.021300	0.024130	0.005710
h(7)=h(18)	0.029656	0.006090	0.045019
h(8)=h(17)	0.039785	0.046694	0.007080
h(9)=h(16)	0.056730	0.007414	0.094410
h(10)=h(15)	0.083850	0.095170	0.007840
h(11)=h(14)	0.146902	0.008310	0.314017
h(12)=h(13)	0.448573	0.314530	0.490736
h(0)	0.146902	0.506230	0.314017

The Table 13 represents the optimized coefficients of FIR Low-pass filter of order 23 obtained by using PM, CSA, AOA. The coefficients are symmetrical about h(0).

TABLE 14. Optimized Coefficients for FIR BP Filter of order 23 using PM, CSA, AOA

h(n)	PM	CSA	AOA
h(1)=h(24)	0.008516	0.015757	2.12E-06
h(2)=h(23)	0.026430	00.02370	0.032840
h(3)=h(22)	0.038040	0.039770	0000000
h(4)=h(21)	0.022509	0.002829	-0.004610
h(5)=h(20)	0.018810	0.090452	8.83E-10
h(6)=h(19)	0.059538	-00.00320	0.086351
h(7)=h(18)	0.068054	0.312840	8.62E-15
h(8)=h(17)	0.030770	0.503373	0.004847
h(9)=h(16)	0.049563	0.312840	-001E-29
h(10)=h(15)	0.159270	-0.003200	-00.3114
h(11)=h(14)	0.271810	0.090452	2.62E-10
h(12)=h(13)	0.346524	0.002829	0.495058
h(0)	0.346524	0.039770	2.62E-10

The Table 14 represents the optimized coefficients of FIR Band-pass filter of order 23 obtained by using PM, CSA, AOA. The coefficients are symmetrical about h(0).

TABLE 15. Optimized Coefficients for FIR HP Filter of order 23 using PM, CSA, AOA

h(n)	PM	CSA	AOA
h(1)=h(24)	-0.0032	0.015757	0.004566
h(2)=h(23)	0.006363	0.002370	-0.000400
h(3)=h(22)	0.008616	0.039770	0.011580
h(4)=h(21)	0.010850	0.002829	0.000850
h(5)=h(20)	0.015920	0.090452	0.024361
h(6)=h(19)	0.021297	-0.003200	0.001140
h(7)=h(18)	0.029656	0.312840	0.046850
h(8)=h(17)	0.039780	0.503373	0.001429
h(9)=h(16)	0.056730	0.312840	0.095091
h(10)=h(15)	0.083847	-0.003200	0.001450
h(11)=h(14)	0.146902	0.090452	0.314690
h(12)=h(13)	0.448570	0.002829	0.501416
h(0)	0.448573	0.039770	0.314690

The Table 15 represents the optimized coefficients of FIR High-pass filter of order 23 obtained by using PM, CSA, AOA. The coefficients are symmetrical about h(0).

We know that the number of coefficients of a filter is one more than the order of the filter. So the coefficients of the filters of order 15 and 23 are 16 and 24 respectively and they are symmetrical about the centre coefficient that is h(0) for the filters of both orders 15 and 23.

3.7 Summary

The magnitude responses of the filter of their respective order have been represented in the above figures. The blue color response is denoted as the performance of the filter using PM Algorithm, the red color denotes the performance of the filter using AOA Algorithm. It is clear from the responses that performance of the filter using AOA is much better than PM in terms of pass-band ripples. A slight deviation is noticed in the transition band in case of AOA from PM, and the stop-band attenuation in case of AOA is much better than other two algorithms.

The device utilization and power utilization table are derived using the FPGA kit for the three algorithms that are PM, CSA, and AOA. It is evident from the tables that AOA requires much less power and consumes the minimum device area or components as compared to the PM algorithm. The other figures represent the RTL diagram and timing diagram. The data in the optimized coefficient tables again depicts that the coefficients of the filter design using AOA are much efficient and of minimum order than other coefficients derived by using PM and CSA, thereby avoiding the problem of local minima entrapment. Again the design of three filters done by AOA exhibits much better performance than that derived by using PM algorithm.

Chapter 4

Conclusion and Future Prospect

4.1 Conclusion

In this paper type-1 low pass, band-pass, and high-pass filters have been designed using the AOA algorithm. The simulation results and magnitude responses of the FIR filter justify the betterment of the performance of AOA, CSA, and PM. The attenuation value (As) and pass-band ripple(Rp) provided in the respective table is calculated for the design of low-pass, band-pass, and high-pass filters for different orders proving that the AOA algorithm is much better than the CSA and PM algorithm. Again the coefficients obtained by using AOA for different filters of their respective orders are minimum and more efficient than the coefficients obtained by using CSA and PM, thereby avoiding the problem of local minima entrapment and acquiring minimum space. Further, the FPGA-based design of optimal FIR filter using AOA is implemented and their respective RTL design and timing diagram is given. The reports of the power requirement and device component utilization indicate that the space or the component utilization is more minimum in the case of AOA than the component space required in the case of CSA and PM algorithm. Even the power distribution is also minimum for AOA than the power distribution in the case of PM and CSA, thus generating less amount of heat while implementing on FPGA using the AOA algorithm. The values derived by using AOA in the Xilinx ISE design suite have been presented in the above table and are also compared with the value derived from the PM algorithm.

The findings of this Thesis work has been accepted for Presentation as per the details given below:

K. Debnath, S. Dhabal and P. Venkateswaran, "Design of High-Pass FIR Filter Using Arithmetic Optimization Algorithm and Its FPGA Implementation", IEEE Region 10 Symposium TENSYMP 2022, VICTOR MENEZES CONVENTION CENTRE, IIT BOMBAY, JULY 1 – 3, 2022.

REFERENCES

[1] J. G. Proakis and D. G. Manolakis, “Digital signal processing”, 3rd ed., Pearson Education, 2007.

[2] X. X. Zheng, J. Yang, S. Y. Yang, W. Chen, L. Y. Huang, and X. Y. Zhang, "Synthesis of Linear-Phase FIR Filters With a Complex Exponential Impulse Response", IEEE Transactions on Signal Processing, Vol. 69, pp. 6101-6115, 2021.

[3] R. A. Zitar and A. Al-Dmour, “An Evolutionary FIR Filter Design Method”, in: Evolutionary Image Analysis and Signal Processing, Studies in Computational Intelligence, Springer, Vol. 213, pp.185-200, 2009.

[4] A. Aggarwal, T. Rawat, and D. Upadhyay, “Design of Optimal Digital FIR Filters using Evolutionary and Swarm Optimization Techniques”, AEU - International Journal of Electronics and Communications, Vol. 70, No. 4, pp. 373-385, 2015.

[5] S. Tsutsumi and K. Suyama, “Design of Digital FIR Filters with discrete coefficients using Ant Colony Optimization”, Electronics and communication in Japan, Vol. 97, No. 4, pp. 30-37, 2014.

[6] S. Yadav, R. Yadav, A. Kumar, and M. Kumar, “A novel approach for optimal design of digital FIR filter using grasshopper optimization algorithm”, ISA Transactions, Vol. 108, pp. 196-206, 2021.

[7] R. S. Chauhan and S. K. Arya, “An Optimal Design of FIR Digital Filter Using Genetic Algorithm”, Contemporary Computing. IC3 Communications in Computer and Information Science, Springer, Vol. 168, pp. 51-56, 2011.

[8] P. U. Kumar, G.R.C.K. Sarma, S. M. Das, and M.A.V. Kamalnath, “Design of Optimal Digital Fir Filter Using Particle Swarm Optimization Algorithm”, Advances in Computational Science, Engineering and Information Technology, Advances in Intelligent Systems and Computing, Springer, Vol. 225, pp. 187-196, 2013.

[9] S. K. Saha, R. Kar, D. Mandal, and S. P. Ghoshal, “An Efficient Craziness Based Particle Swarm Optimization Technique for Optimal IIR Filter Design”, Transactions on Computational Science XXI, Lecture Notes in Computer Science, Springer, Vol. 8160, pp. 230-252, 2013.

[10] R. P. M. Sundari and S. N. Deepa, "Variant of the charged system search algorithm for the design of optimal linear phase finite impulse response filters", Journal for Control, Measurement, Electronics, Computing and Communications, Vol. 60, pp. 266-273, 2019.

[11] S. K. Saha, R. Kar, D. Mandal, and S. Ghoshal, "Optimal Stable IIR Low Pass Filter Design Using Modified Firefly Algorithm", Swarm, Evolutionary, and Memetic Computing, SEMCCO, Lecture Notes in Computer Science, Springer, Vol. 8297, pp. 98-109, 2013.

[12] P. Shao, Z. Wu, X. Zhou, and D.C. Tran, " FIR Digital filter design using Improve Particle Swarm Optimization Based On Refraction Principle", Soft Computing, Vol. 21, pp. 2631-2642, 2017.

[13] W. Huang, L. Zhou, J. Qian, and L. Ma, "FIR Frequency Sampling Filters Design Based on Adaptive Particle Swarm Optimization Algorithm", Advances in Natural Computation, ICNC, Lecture Notes in Computer Science, Springer, Vol. 3612, pp. 289-298, 2005.

[14] T. Mittal, "Design of Optimal FIR Filters Using Integrated Optimization Technique", Circuits, Systems, and Signal Processing, Vol. 40, pp. 2895-2925, 2021.

[15] I. Sharma, B. Kuldeep, A. Kumar, and V. K. Singh, "Performance of swarm based optimization techniques for designing digital FIR filter", A comparative study, Engineering Science and Technology, an International Journal, Vol. 19, No. 3, pp.1564-1572, 2016.

[16] S. K. Saha and N. Venkatesan, "Farrow structure based FIR filter design using hybrid optimization", AEU - International Journal of Electronics and Communications, Vol. 114, pp. 1-27, 2020.

[17] L. Abualigah, A. Diabat, S. Mirjalili, M. A. Elaziz, and A. H. Gandomi, "The Arithmetic Optimization Algorithm", Computer Methods in Applied Mechanics and Engineering, Vol. 376, pp. 1-38, 2021.

[18] L. Abualigah, A. Diabat, P. Sumari, and A. H. Gandomi, "A Novel Evolutionary Arithmetic Optimization Algorithm for Multilevel Thresholding Segmentation of COVID-19 CT Images", Vol. 9, pp. 1-37, 2021.

[19] R. Zheng, H. Jia, L. Abualigah, Q. Liu, and S. Wang, "An improved arithmetic optimization algorithm with forced switching mechanism for

global optimization problems”, Mathematical Biosciences and Engineering, Vol. 19, No. 1, pp. 473-512, 2022.

[20] X. D. Li, J. S. Wang, W. K. Hao, M. Zhang, and M. Wang, “Chaotic arithmetic optimization algorithm”, Appl Intell , pp. 1-40, 2022.

[21] S. Chauhan and G. Vashishtha, “Mutation-based Arithmetic Optimization Algorithm for Global Optimization”, in Proc. of International Conference on Intelligent Technologies (CONIT), pp. 1-6, 2021.

[22] M. Premkumar et al., “A New Arithmetic Optimization Algorithm for Solving Real-World Multiobjective CEC-2021 Constrained Optimization Problems: Diversity Analysis and Validations”, IEEE Access, Vol. 9, pp. 84263-84295, 2021.

[23] N. Panga, U. Sivaramakrishnan, R. Abishek, K. Bingi, and J. Chaudhary, “An Improved Arithmetic Optimization Algorithm”, in Proc. of IEEE Madras Section Conference (MASCON), pp. 1-6, 2021.

[24] D. Izci, S. Ekinci, E. Eker, and A. Dündar, “Improving Arithmetic Optimization Algorithm Through Modified Opposition-based Learning Mechanism”, in Proc. of 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), pp. 1-5, 2021.

[25] S. Ma, H. Huang, and Z. Wu, “The Arithmetic Optimization Algorithm Based-on Particle Energy and Particle Diffusion”, in Proc. of International Conference on Wireless Communications and Smart Grid (ICWCSG), pp. 246-253, 2021.

[26] N. Kumari and P. Jaglan, “Design Of FIR Low Pass Filter Using Particle Swarm Optimization Algorithm”, International Advanced Research Journal in Science, Engineering and Technology, Vol. 1, pp. 31-35, 2017.

[27] A. Mukherjee, N. Chakraborty, and B. K. Das, “Whale optimization algorithm: An implementation to design low-pass FIR filter”, in Proc. of Innovations in Power and Advanced Computing Technologies (i-PACT), Vol. 5, pp. 1-5, 2017.

[28] M. Khorshidian, N. Reiskarimian, and H. Krishnaswamy, "A Compact Reconfigurable N-Path Low-Pass Filter Based on Negative Trans-Resistance with <1dB Loss and >21dB Out-of-Band Rejection", in Proc.

of IEEE/MTT-S International Microwave Symposium (IMS), pp. 799-802, 2020.

[29] M. Sababha, and M. Zohdy, “Linear Phase FIR Low Pass Filter Design Based on Firefly Algorithm”, International Journal of Electrical and Computer Engineering (IJECE), Vol. 8, N0. 6, pp: 4356-4365, 2018.

[30] S. K. Sarangi, R. Panda, and A. Abraham, “Design of optimal low-pass filter by new levy Swallow Swarm Algorithm”, Soft Computing - A Fusion of Foundations, Methodologies and Applications, Vol. 24, No. 23, pp. 18113–18128, 2020.

[31] L. Chen, M. Liu, J. Yang, J. Wu, and Z. Dai, “Structure Evolution based Optimization algorithm for Low pass IIR Filter Design”, Vol. 10, No. 1, pp. 1036 – 1055, 2017.

[32] Neha and A. P. Singh, “Design of Linear Phase Low Pass FIR Filter using Particle Swarm Optimization Algorithm”, International Journal of Computer Applications, Vol. 98, No.3, pp. 40-44, July 2014.

[33] P. Gowtham, S. Sowndarya, and N. Pachauri, “Simulated Annelaing Optimization Low-Pass FIR Filter for Biomedical Signals”, Advances in Mechanical Engineering, Springer Singapore, pp. 741-748, 2021.

[34] S. K. Sarangi, R. Panda, and A. Abraham, “Design of optimal low-pass filter by a new Levy swallow swarm algorithm”, Soft Computing, Vol. 24, pp. 18113–18128, 2020.

[35] P. Bertsias and C. Psychalinos, “Differentiator based fractional-order high-pass filter designs”, in Proc. of 7th International Conference on Modern Circuits and Systems Technologies (MOCAST), pp. 1-4, 2018.

[36] S. Mandal, S. P. Ghoshal, R. Kar, and D. Mandal, “Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique”, Journal of King Saud University - Computer and Information Sciences, Vol. 24, No. 1, pp. 83-92, 2012.

[37] R. Islam, R. Kar, D. Mandal, and S.P. Ghoshal, “Design and Simulation of FIR High Pass Filter Using Gravitational Search Algorithm”, Swarm, Evolutionary, and Memetic Computing. SEMCCO, Lecture Notes in Computer Science, Springer, Vol. 8297, pp. 547-557, 2013.

[38] A. Sarangi, R. Lenka, S. K. Sarangi, “Design of Linear Phase FIR High Pass Filter Using PSO with Gaussian Mutation”, Swarm, Evolutionary,

and Memetic Computing. SEMCCO 2014, Lecture Notes in Computer Science (), Springer, Vol. 8947, pp. 471-479, 2015.

[39] A. Sarangi, R. Lenka, and S. K. Sarangi, “Design of High Pass Filter Using PSO with Gaussian Mutation”, Swarm, Evolutionary and Memetic Computing, Springer International Publishing, Vol. 8947, pp. 471-479, 2015.

[40] S. Mandal, S. P. Ghoshal, R. Kar, and D. Mandal, “Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique”, Journal of King Saud University - Computer and Information Sciences, Vol. 24, No. 1, pp. 83-92, 2012.

[41] A. Aggarwal, T. K. Rawat, M. Kumar, and D. K. Upadhyay, “Optimal design of FIR high pass filter based on L1 error approximation using real coded genetic algorithm”, Engineering Science and Technology, an International Journal, Vol. 18, No. 4, pp. 594-602, 2015.

[42] S. Dhabal and P. Venkateswaran, “An efficient gbest-guided Cuckoo Search algorithm for higher order two channel filter bank design, Swarm and Evolutionary Computation”, Vol. 33, pp.- 68-84, 2017.

[43] A. Sikder, P. Bhattacharjee, S. Chowdhury, S. Dhabal, and P. Venkateswaran, “Design of Band-pass FIR Filter using Improved Sine Cosine Algorithm and its Implementation on FPGA”, in Proc. of IEEE Region 10 Symposium (TENSYMP), pp. 310-315, 2019.

[44] V. Kumar, R. Mehra, and Shallu, "Reconfigurable band-pass filter using Kaiser window for satellite communication", in Proc. of 2nd International Conference on Next Generation Computing Technologies (NGCT), pp. 409-413, 2016.

[45] S. K. Saha, R. Kar, D. Mandal, and S.P. Ghoshal, “ Design and simulation of FIR band pass and band stop filters using gravitational search algorithm”, Memetic Computing, Vol. 5, pp. 311-321, 2013.

[46] R. Kar, D. Mandal, S. Bardhan, and S. P. Ghoshal, "Optimization of linear phase FIR band pass filter using Particle Swarm Optimization with Constriction Factor and Inertia Weight Approach," in Proc. of IEEE Symposium on Industrial Electronics and Applications, pp. 326-331, 2011.

[47] M. Sangeeta, P. G. Sakti, and K. Rajib, *et al, “* Optimal linear phase finite impulse response band pass filter design using craziness based particle

swarm optimization algorithm", J. Shanghai Jiaotong Univ. (Sci.) , Vol. 16, pp. 696–703, 2011.

[48] S. Mandal, P. Mallick, D. Mandal, R. Kar, and S.P. Ghoshal, "Optimal FIR band-pass filter design using novel particle swarm optimization algorithm", in Proc. of SHUSER IEEE Symposium on Humanities, Science and Engineering Research, pp. 141-146, 2012.

[49] D. Sasidaran, A. Azam, K. E. Nelson, and M. A. Soderstrand, "FPGA implementation of a tunable band-pass filter using the basic heterodyne block", in Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings, pp. 1093-1096, Vol. 2, 2001.

[50] C. Charoensak and S. S. Abeysekera, "FPGA implementation of efficient Kalman band-pass sigma-delta filter for application in FM demodulation", in Proc. of IEEE International SOC Conference, pp. 137-138, 2004.

[51] C. Guo-wei and W. Feng-ying, "The implementation of FIR low-pass filter based on FPGA and DA", in Proc. of Fourth International Conference on Intelligent Control and Information Processing (ICICIP), pp. 604-608, 2013.

[52] O. KalaiPriya, S. Ramasamy, and D. Ebenezer, "VLSI implementation of nonlinear variable cut-off high pass filter algorithm", in Proc. of 3rd International Conference on Electronics Computer Technology, pp. 275-278, 2011.

[53] K. A. Toker, S. Özen, and A. Arsal, "FPGA implementation of a low-complexity fading filter for multipath Rayleigh fading simulator", in Proc. of XXXth URSI General Assembly and Scientific Symposium, pp. 1-4, 2011.

[54] Y. Zhou and P. Shi, "Distributed Arithmetic for FIR Filter implementation on FPGA", in Proc. of International Conference on Multimedia Technology, pp. 294-297, 2011.

[55] A. Das, S. Dash, A. K. Sahoo, and B. C. Babu, "Design and implementation of FPGA based linear all digital phase-locked loop", in Proc. of Annual IEEE India Conference (INDICON), pp. 280-285, 2012.

[56] A. Singhal, A. Goen, and T. Mohapatra, "FPGA implementation and power efficient CORDIC based ADPLL for signal processing and

application”, in Proc. of 7th International Conference on Communication Systems and Network Technologies (CSNT), pp. 325-329, 2017.

[57] P. K. Meher, S. Chandrasekaran, and A. Amira, "FPGA Realization of FIR Filters by Efficient and Flexible Systolization Using Distributed Arithmetic", IEEE Transactions on Signal Processing, Vol. 56, No. 7, pp. 3009-3017, 2008.

[58] T. Dutta, R. M. Aich, S. Dhabal, and P. Venkateswaran, “Finite Impulse Response Filter Design using Grasshopper Optimization Algorithm and Implementation on FPGA”, in Proc. of IEEE Applied Signal Processing Conference (ASPCON), pp. 313-317, 2020.

[59] J. B. Evans, "Efficient FIR filter architectures suitable for FPGA implementation", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 41, no. 7, pp. 490-493,1994.

[60] M. A. Eshtawie and M. Othman, “FPGA Implementation of an Optimized Coefficients Pulse Shaping FIR Filters”, in Proc. of IEEE International Conference on Semiconductor Electronics, pp. 454-458, 2006.

[61] G. Jinding, H. Yubao, and S. Long, “Design and FPGA Implementation of Linear FIR Low-pass Filter Based on Kaiser Window Function”, in Proc. of International Conference on Intelligent Computation Technology and Automation, pp. 496-498, 2011.

[62] R. Chand, P. Tripathi, A. Mathur, and K. C. Ray, “FPGA implementation of fast FIR low pass filter for EMG removal from ECG signal”, in Proc. of International Conference on Power, Control and Embedded Systems, pp. 1-5, 2010.

[63] X. X. Zheng, J. Yang, S. Y. Yang, W. Chen, L. Y. Huang, and X. Y. Zhang, "Synthesis of Linear-Phase FIR Filters With a Complex Exponential Impulse Response", IEEE Transactions on Signal Processing, Vol. 69, pp. 6101-6115, 2021.

Jadavpur University

Kaushik Debnath <kaushikd.etce.pg@jadavpuruniversity.in>

[TENSYMP2022] Your paper #1570799089 ('Design of High-Pass FIR Filter Using Arithmetic Optimization Algorithm and Its FPGA Implementation')

1 message

TENSYMP2022 (tensymp2022@gmail.com) <tensymp2022=gmail.com@edas.info> Wed, Apr 20, 2022 at 5:54 PM
Reply-To: TENSYMP2022 <tensymp2022@gmail.com>
To: Kaushik Debnath <kaushikd.etce.pg@jadavpuruniversity.in>, Supriya Dhabal <supriya_dhabal@yahoo.co.in>, Palaniandavar Venkateswaran <pvwn@ieee.org>
Cc: tensymp2022@gmail.com

Dear Mr. Kaushik Debnath:

Congratulations - your paper #1570799089 ('Design of High-Pass FIR Filter Using Arithmetic Optimization Algorithm and Its FPGA Implementation') for TENSYMP2022 has been **accepted** for oral presentation during TENSYMP 2022.

Please revise the paper as per the comments of the reviewers (appended below) and also make sure that the similarity score of your paper is below 20 %, failing which your paper is liable to be rejected at any stage.

The registration link and camera-ready paper submission details will be shared soon on the conference website. We look forward to meeting you and your co-authors in person during TENSYMP 2022 at IIT Bombay, India!

The reviews are below or can be found at 1570799089.

Manuscript Review TENSYM 2022 1

Reviewer's Comment: Further comments on the paper

1. Finding of the literature survey is not mentioned.
2. A selection criterion of FPGA is not justified.
3. The dynamic power in Table III is zero in all cases, justification of this is not included.
4. Compare results with published/experimental results.
5. Reference papers of reputed journals like IEEE transactions are not used.
6. Overall paper and references are not in standard IEEE format.

Manuscript Review TENSYM 2022 2

Reviewer's Comment: Further comments on the paper

Author details are not furnished on first page (affiliations and mail ID) References in part IV simulation Results needs correction. Work seems satisfactory!

Manuscript Review TENSYM 2022 3

Reviewer's Comment: Further comments on the paper

1. Abstract must contain all the key words
2. page 1 co 2 para 2, "Based on this theorem......". In place of "problems" Algorithms is better word. In the same para where you have written Few examples, please cite suitable reference for the same.
3. You have provided the Pseudo code of your algorithm, you must provide flow chart for the same.
4. Figures re of poor quality.
5. There is no heading given for the Table II.
6. References are not proper, for the books you havn't mentioned the publisher, in some references name of the conference/ journal is missing.

Regards,
The conference chairs

Dear Authors,

Greetings from TENSYMP2022!

Please find the Detailed Program Schedule of the Conference along with your own Presentation Session and Timing Details etc., attached with this mail. Please note that we would like all our Authors (Presenters) and Participants to be present during the entire duration of the conference and thus take full benefit of this prestigious event. Besides, four keynote addresses by the most distinguished speakers from India and abroad, we have also organised track invited talks by the expert speakers in the respective domains.

If you have already submitted the camera-ready manuscript of your paper and you have already made author registration, but if your paper is missing in the attached program schedule, then please let us know urgently (latest by June 10, 2022) by sending an email to "tensymp2022@gmail.com" and by giving details such as your long paper id (EDAS), first author name, author registration details etc.

Participants' registrations are still open due to many requests, and hence your co-authors and colleagues can still register for the conference as co-authors or participants.

We are looking forward to welcoming you all to the Conference venue – physically or virtually. Please help us make the Conference a grand success.

With Best Regards!

General Chair, TENSYMP2022

IEEE Region 10 Symposium (TENSYMP 2022)
July 1-3, 2022, VMCC, IIT Bombay, INDIA
Conference Program

Day 1: July 1, 2022

Time (IST)	Mode	Activity
08:00 - 09:00	Offline	Registration & Welcome
09:00 - 09:30	Hybrid	Inauguration
09:30 - 10:15	Hybrid	Keynote Talk (Inaugural)
10:15 - 10:30		High Tea
10:30 - 11:00	Hybrid	Parallel Track Invited Talks (IT1/IT2)
11:00 - 13:00	Hybrid	Parallel Technical Sessions (TS1/TS6/TS9/TS15/TS23)
13:00 - 14:00		Lunch
14:00 - 14:30	Hybrid	Parallel Track Invited Talks (IT3/IT4)
14:30 - 16:30	Hybrid	Parallel Technical Sessions (TS2/TS11/TS16/TS20/TS24)
16:30 - 16:45		Tea
16:30 - 18:30	Hybrid	Industry Session (MathWorks Workshop)

Day 2: July 2, 2022

Time (IST)	Mode	Activity
09:00 - 11:00	Hybrid	Parallel Special (SS1/SS4/SS5) & Physical Poster (PP1) Sessions
11:00 - 11:15		Tea
11:15 - 13:15	Hybrid	Parallel Technical (TS3/TS7/TS10/ TS18/TS25) & Virtual Poster (VP1) Sessions
13:15 - 14:00		Lunch
14:00 - 14:30	Hybrid	Parallel Track Invited Talks (IT5/IT6)
14:30 - 16:30	Hybrid	Parallel Technical (TS4/TS12/TS17/TS21/TS26) & Virtual Poster (VP2) Sessions
16:30 - 16:45		Tea
16:45 - 17:30	Hybrid	Keynote Talk (Evening Talk)
17:30 - 19:30	Hybrid	Parallel Special (SS2/SS3/SS6) & Physical Poster (PP2) Sessions
20:00 - 22:00		Gala dinner

Day 3: July 3, 2022

Time (IST)	Mode	Activity
09:30 - 10:15	Hybrid	Keynote Talk (Virtual)
10:15 - 10:45		Parallel Track Invited Talks (IT7/IT8)
10:45 -11:00	Hybrid	Tea
11:00 - 13:00	Hybrid	Parallel Technical Sessions (TS5/TS8/TS14/TS19/TS27)
13:00 - 13:45		Lunch
13:45 - 14:30	Hybrid	Track Invited Talk (IT9) & R10 IRC Session (IT10)
14:30 - 16:30	Hybrid	Parallel Technical Sessions (TS13/TS22/TS28/TS29)
16:30 - 16:45		Tea
16:45 - 17:30	Hybrid	Keynote Talk (Concluding)
17:30 - 18:00	Hybrid	Valedictory Session, Awards and Unveiling of new logo of Bombay Section

TS 12: Low Power VLSI Devices, Circuits and Systems				
Sn.	**Paper ID**	**Paper Title**	**Corresponding Author**	**Registered Author**
1.	1570794489	An RC Charging Based Soft-Start Circuit for Voltage Regulators	Saurabh Goyal (NXP Semiconductors, India)	Saurabh Goyal (NXP Semiconductors, India)
2.	1570798974	A Concurrent Testing Scheme for Muller Circuits Using Reduced Ordered Binary Decision Diagram	Pradeep Kumar Biswal (IIIT Bhagalpur, India)	Pradeep Kumar Biswal (IIIT Bhagalpur, India)
3.	1570799043	Hybrid MoCs for Long-Distance On-Chip Communications	Arijit Karali (NIT Karnataka, India)	Arijit Karali (NIT Karnataka, India)
4.	1570799089	Design of High-Pass FIR Filter Using Arithmetic Optimization Algorithm and Its FPGA Implementation	Kaushik Debnath (Jadavpur University, India)	Kaushik Debnath (Jadavpur University, India)
5.	1570801453	Device Reliability Affecting Coding Schemes in Neuromorphic Circuits	Siona Picardo (BITS Pilani Dubai Campus, United Arab Emirates)	Nilesh Goel (BITS Pilani Dubai Campus, United Arab Emirates)

Design of High-Pass FIR Filter using Arithmetic Optimization Algorithm and its FPGA Implementation

Kaushik Debnath
Dept. of ETCE
Jadavpur University
Kolkata, India
kaushikd.etce.pg@jadavpuruniversity.in

Supriyo Dhabal
Dept. of ECE
Netaji Subhash Engineering College
Kolkata, India
supriya_dhabal@yahoo.co.in

P. Venkateswaran, SMIEEE
Dept. of ETCE
Jadavpur University
Kolkata, India
pvwn@ieee.org

***Abstract*—This paper presents an efficient implementation of high-pass filter using the Arithmetic Optimization Algorithm. The result of this algorithm was found to be better in terms of pass-band ripple and stop-band attenuation in comparison to the Park-McClellan (PM) algorithm. Further, the hardware implementation of the proposed design in the FPGA platform have been discussed in this proposed work.**

Keywords—FIR Filter, High-Pass Filter, AOA, meta heuristic optimization algorithm, genetic optimization algorithm, simulated annealing algorithm, Sine Cosine algorithm, whale optimization algorithm, forced switching mechanism, FPGA.

I. INTRODUCTION

Filters can be broadly classified as analog and digital. Digital filter is an important part of digital signal processing. A digital filter is actually used to serve the purposes like signal separation and restoration. Since a digital filter takes a digital input and gives a digital output, with various digital components mixed up, so signal separation separates the noise and signal restoration restores the signal in its actual form. Digital filters are again classified as Finite Impulse Response (FIR) filter and Infinite Impulse Response (IIR) filter [1]. The output of FIR filters fades away due to impulse input within a limited amount of time and they are non-recursive. Due to greater stability and having a linear phase[2] and finite impulse response, FIR filters are preferred over IIR filters in many applications.

The design of a digital FIR filter in an optimum way requires a set of coefficients so that the magnitude response has a maximum stop-band attenuation and minimum pass-band ripple. Conventionally, various well-known methods have been implemented for the design of digital FIR filters like the window method, frequency sampling method, etc. The window method consists of window functions such as Butterworth, Chebyshev, Kaiser, etc. It consists of truncating an infinite filter impulse response by a window function that will be suitable for the respective filter. The windowing method for the digital filter design is fast, robust, convenient but usually suboptimal. For the design of optimal digital filters, the main objective function involves specific control of various parameters of the frequency spectrum which is highly non-uniform, non-linear, non-differentiable, and multimodal. So optimization of the above objective function cannot be done using the above techniques and convergence to the global minimum solution is also quite difficult. So, various optimization techniques or algorithms [1-3] are used to get rid of these problems.

The most important purpose of these evolutionary or meta-heuristic optimization algorithms for the digital filter is to optimize the cost function by reducing or minimizing the error between the desired response and the actual response as much as possible. Due to this the requirement of conventional gradient-based design processes like continuous and differentiable cost function has been eliminated. Again the problem of "Local Optima Entrapment" that is the chance of getting trapped in local minima is very much less for this algorithm. Algorithms like simulated annealing, genetic algorithm, particle swarm optimization algorithm [4], whale optimization algorithm [5], cuckoo search algorithm [6], etc. are significant optimization problems.

According to the theoretical studies it has been published that a single algorithm cannot solve all optimization problems and this theorem is known as the "**No Lunch Free Theorem**". Based on this theorem, the researchers are motivated to design new problems and get better results. In 2020, Laith Abualigah developed Arithmetic Optimization Algorithm (AOA) [7-9]. This algorithm mainly depends on the properties of arithmetic operators which are multiplication, division, subtraction, and addition, which helps in exploring and exploiting the minima within the search region. According to our literature survey this algorithm is applied in various design problems like pressure vessel design, 3-bar truss design problems, etc [8]. Again an improved version of this algorithm is integrated with forced switching mechanism [6], a chaotic mapping strategy introduced in AOA to boost its convergence speed and accuracy[8], etc. Based on this algorithm, various other algorithms hybridized with AOA have been developed. Few examples are advanced AOA for solving mechanical engineering design algorithms hybridizing Sine Cosine Algorithm (SCA) with AOA and fuzzy SCA-AOA in information security, AOA in multi-level thresholding

978-1-6654-6658-5/22/$31.00 ©2022 IEEE

segmentation of COVID-19 CT images, an improved artificial neural network using AOA for damage assessment in FGM composite plates, are quite successful in achieving specific and accurate results, so this motivates us to use this in designing a FIR high pass filter [10-11] with less computational effort. FPGA (Field Programmable Gate Array) consists of three parts which are complex logical blocks (CLB), interconnects, and I/O ports. These parts are configurable by the user according to their requirement. It consists of flip-flops and two-dimensional logic array blocks. The interconnections between them are programmed or established by programmable switches. The logical interactions and logical functions are controlled by memory cells. FPGA also plays an important role in embedded systems based on their capability to start system software development simultaneously with hardware development and shows parallel execution of hardware characteristics, which shows more efficient than a DSP block in high capacity data processing. Thus, there are various advantages of designing or implementing a digital filter on FPGA at a lower cost, with high sampling rates, and with better flexibility and on the basis of this reason today for implementing real time signal processing algorithms FPGA is used. So to check the reliability and the compatibility of our designed high pass filter using AOA we implemented it on FPGA.

The remaining paper is arranged as follows: The problem formulation of the high pass filter is provided in section II. A short description of Design Methodologies using the Arithmetic Optimization Algorithm is presented in section III. Section IV demonstrates the simulation outputs obtained using the optimization algorithm and its implementation on FPGA. Finally, the inference or conclusion is drawn in section V.

II. PROBLEM FORMULATION

The transfer function of the high pass filter is represented as:

$$H(z) = \sum_{n=0}^{M-1} h(n) z^{-n} \quad n = 0,1,\dots,M-1 \tag{1}$$

which contains $(M-1)$ trivial poles and $(M-1)$ zeroes located anywhere in the z-plane. Here, $h(n)$ denotes the impulse response of FIR filter and based on the coefficients of $h(n)$ a filter can exhibit the different types of magnitude responses like low-pass, high-pass, or band-pass.

A. Error Function Representation

The error function is defined by the difference between the stop-band and pass-band. Practically by minimizing the deviation of the actual magnitude response from the ideal response the implementation of the filter can be done which is defined by the equation:

$$E(\omega) = W(\omega)\left[\, H_d\left(e^{j\omega}\right) - H_a\left(e^{j\omega}\right)\,\right] \tag{2}$$

The error function E (ω), given by Parks-McClellan (PM) is represented in Eq. (2), where W (ω), $H_d\left(e^{j\omega}\right), H_a(e^{j\omega})$ are the weight vector, desired, and approximated frequency response respectively.

978-1-6654-6658-5/22/$31.00 ©2022 IEEE

In error function $E(\omega)$, the ratio between peak ripple at pass-band (δ_p) and stop-band (δ_s) cannot be different values. Thus, in order to reduce the flaws in this function, a modified error function is represented as:

$$U = \max_{\omega \le \omega_p}\left(\left|E(\omega)\right| - \delta_p\right) + \max_{\omega \ge \omega_s}\left(\left|E(\omega)\right| - \delta_s\right) \tag{3}$$

where ω_p, ω_s, δ_p and δ_s are the desired filter specifications.

Now, the high-pass filter's ideal response can be denoted by [9]:

$$H_i(e^{j\omega}) = \begin{cases} 0 & for\ 0 \le \omega \le \omega_c \\ 1 & otherwise \end{cases} \tag{4}$$

where ω_c denotes the cut-off frequency of the HP filter; ω_p, ω_s are the pass-band and the stop-band frequency of the filter.

B. Type 1 Linear Phase FIR Filter

The frequency response function of the FIR filter is described as:

$$H\left(e^{j\omega}\right) = \sum_{n=0}^{M-1} h[n] e^{-j\omega n}, \ -\pi \le \omega \le \pi \tag{5}$$

Now linear phase constraint is imposed by:

$$angle\ H\left(e^{j\omega}\right) = -\tau_{\emptyset}\omega\,, -\pi \le \omega \le \pi \tag{6}$$

Here ω is a constant phase delay. Now for Type 1 filter, $h(n)$ must be symmetric, i.e.

$$h(n) = h(M-1-n)\,, 0 \le n \le (M-1) \text{with}\ \omega = \frac{M-1}{2} \tag{7}$$

Here $h(n)$ is symmetric about ω, which is the index of symmetry. The value of M, in Eq. (5), can take even or odd integer, and hence two types of filters come into the picture namely, Type 1 and Type 2 filter.

Type 1 FIR filter has symmetrical impulse response with M odd. In this case, ω is an integer (from Eq. (7)), and $h(n) = h(M-1-n)$, where $0 \le n \le M-1$. Then the response of the Type 1 filter can be written as:

$$H(e^{j\omega}) = \left[\sum_{n=0}^{(M-1)/2} a(n) \cos \omega n\right] e^{-j\omega(M-1)/2} \tag{8}$$

C. Cost function formulation for High-Pass FIR Filter

The objective function for high pass filter is considered as:

$$\emptyset_{HP} = \gamma * E_S + (1-\gamma) * E_P, \quad 0 \le \gamma \le 1 \tag{9}$$

where E_p, E_S are calculated following Eqs. (10)- (11).

$$E_P = \frac{1}{\pi}\int_{\omega_p}^{\pi} (1 - H(\omega))^2 d\omega = \frac{\pi - \omega_p}{\pi} - 2b_1^T P_1 + b_1^T Q b_1 \tag{10}$$

$$E_S = \frac{1}{\pi}\int_0^{\omega_s}(H(\omega))^2 d\omega = b_1^T C_1 b_1 \quad (11)$$

where E_P represents pass-band error, E_S is the stop-band error respectively for the high-pass filter.

Here

$$H(\omega) = b^T C(\omega), \; b = [\, b_1, b_2, \ldots\ldots, b_{N/2}]^T$$

$$P = \frac{1}{\pi}\int_{\omega_p}^{\pi} \cos(A\omega)\, d\omega \quad (12)$$

$$Q = \frac{1}{\pi}\int_{\omega_p}^{\pi} \cos(A\omega)\cos(B\omega)\, d\omega \quad (13)$$

In case of pass-band error, the value of C can be calculated from the equation which is given below:

$$C(m,n) = \frac{1}{\pi}\int_0^{\omega_s} \cos(A\omega)\cos(B\omega)\, d\omega \quad (14)$$

where $A = \frac{N-1}{2} - m$, $m = 0,1,\ldots\ldots,(M-1)$ and $= \frac{N-1}{2} - n$, $n = 0,1,\ldots\ldots,(M-1)$, and H (ω) is the magnitude response of the filter. Thus the main of our implementation is to optimize the cost function for the high-pass filter as given in Eq. (9).

III. Design Methodologies

A. Arithmetic Optimization Algorithm (AOA)

Generally, all population-based algorithms are based on some common framework. The fitness of the algorithm is defined by its corresponding cost function using some solutions generated randomly which are known as search agents and the process continues repeatedly over the solution until the lowest minima or the desired optimum solution is reached. Generally, there are two main phases in any optimization process - exploration and exploitation. In the exploration phase, different random solutions are generated from where different suitable global minima are selected among the local minima by using the search agents imparting a large degree of volatility to zero in the solution region. In the exploitation phase, the accuracy of the solution or the global minima is improved which are obtained during the exploitation phase. Thus a balanced optimum approach between the exploitation and exploration phase results in reaching the optimum solution in a much lesser number of iterations.

AOA, a novel population-based meta-heuristic optimization algorithm, was developed recently. The main motivation of AOA [12-17] originates from the use of arithmetic operators i.e. Multiplication, Division, Subtraction, and Addition, which helps in exploring and exploiting the searching region and resulting in finding the least global optimum solution in the searching space [13].

The three phases of AOA are as follows:

I. Initialization Phase:

The optimization process of AOA starts with different sets of solutions which are defined by a matrix (X) and the solution obtained in each iteration is considered as the best-obtained solution.

$$X = \begin{bmatrix} x_{1,1} & \cdots & \cdots & x_{1,j} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & \cdots & x_{2,j} & \cdots & x_{2,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & \cdots & x_{N-1,j} & \cdots & x_{N-1,n} \\ x_{N,1} & \cdots & \cdots & x_{N,j} & x_{N,n-1} & x_{N,n} \end{bmatrix} \quad (15)$$

MOA (Math Optimizer Accelerated) co-efficient is calculated as:

$$\text{MOA(C\_Iter)=Min+C\_Iter} \times \left(\frac{Max-Min}{M\_Iter}\right) \quad (16)$$

where MOA (C_Iter) represents the function value at the current iteration, which is calculated by Eq. (16). C_Iter represents the current iteration, bounded between 1 and the maximum number of iterations (M_Iter). Min and Max denote the minimum and maximum values of the accelerated function, respectively.

II. Exploration Phase:

In mathematical calculation, the arithmetic operators like Division (‘ ÷ ’) and multiplication (‘ × ’) produces high dispersion results which helps them to implement the exploration phase. These operators explore the search area randomly in several regions to find a better solution, which is described by Eq. (17). This phase of searching is conditioned by MOA (Math Optimizer Accelerated) function for the condition by $r_1 > \text{MOA}$.

The equation describing the exploration search is defined as:

$$x_{i,j}(C_{Iter}+1) = \begin{cases} best(x_j) \div (MOP+\epsilon) \times \left((UB_j - LB_j) \times \mu + LB_j\right), & r_2 < 0.5 \\ best(x_j) \times MOP \times \left((UB_j - LB_j) \times \mu + LB_j\right), & otherwise \end{cases} \quad (17)$$

where $x_{i,j}(C_{Iter}+1)$ represents the position at the jth term of the ith solution at the current iteration, and $best(x_j)$ represents the best-obtained solution at the jth position. ϵ is a small integer number, UB_j and LB_j represent the upper bound value and lower bound value at the jth position, respectively.

$$\text{MOP(C\_Iter)} = 1 - \frac{C\_Iter^{1/\alpha}}{M\_Iter^{1/\alpha}} \quad (18)$$

where MOP is known as Math Optimizer Probability which is a coefficient, MOP (C_Iter) represents the function value at the current iteration, C_Iter represents the current iteration, and (M_Iter) represents the maximum number of iterations. α is a sensitive parameter that denotes the exploitation accuracy over the iterations.

III. Exploitation Phase:

Again, the arithmetic operators like subtraction (‘-’) and addition (‘+’) produce high dense results which prefer them to implement the exploitation search phase. These operators explore the search area deeply on several dense regions to find a better solution which is described by eq.(19) and this phase of searching is conditioned by the MOA function that is r_1 is not greater than MOA.

978-1-6654-6658-5/22/$31.00 ©2022 IEEE

The equation describing the exploitation phase is defined as:

$$x_{i,j}(C\_Iter+1)= \begin{cases} best(x_j) - MOP \times \left((UB_j - LB_j) \times \mu + LB_j\right), & r_3 < 0.5 \\ best(x_j) + \left((UB_j - LB_j) \times \mu + LB_j\right), & otherwise \end{cases} \quad (19)$$

This equation represents the exploitation of the search space by conducting a deep search. The subtraction operator (S), in this phase (Eq. (19)), is selected when r3<0.5 and the other operator that is the addition operator (A) will not be selected until the previous operator finishes its current task. Otherwise, the second operator (A) will be selected to perform the current task instead of the S.

Arithmetic Optimization Algorithm: Pseudo Code:

Initialize the arithmetic optimization parameters α, μ.
Initialize the solution's positions.
while(C_Iter<M_Iter) do
  Calculate the fitness function of each solution.
  The best solution is determined.
  MOA and MOP values are updated.
  for (i=1 to Solutions) do
    for (j=1 to Positions) do
      A random value is generated between [0.1] (r_1, r_2, r_3)
      if r_1>MOA then
        Exploration Phase
        if r_2>0.5 then
          Apply Division Math Operator.
          Update i^{th}solution using the 1st rule
        else
          Apply Multiplication Math Operator
          Update i^{th}solution using 2nd rule .
        endif
      else
        Exploitation Phase
        if r_3>0.5 then
          Apply subtraction math operator.
          Update i^{th}solutions' position using the first rule.
        else
          Apply addition math operator.
          Update i^{th}solutions' position using the second rule.
        endif
      endif
    endfor
  endfor
  C_Iter=C_Iter+1
endwhile

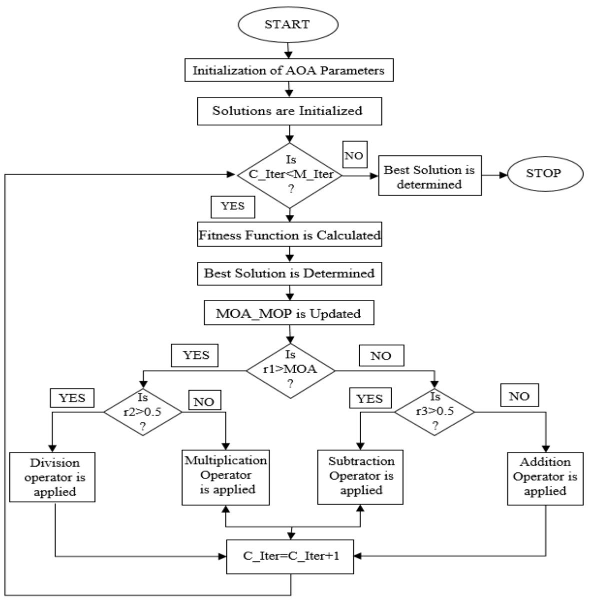


Fig. 1. Flowchart of AOA

The above figure represents the flowchart of the working mechanism of Arithmetic Optimization Algorithm (AOA).

IV. SIMULATION RESULTS

In this section the implementation of high-pass filter with the following parameter settings are discussed:

We assume MOP_Max=1, MOP_Min =0.2, alpha =5.The implementation work is completed using MATLAB 2007B, XILINX ISE Design suite 14.7 tool, and XILINX XC6SLX16-2FTG256C FPGA [18-27] hardware. The FIR filter has been designed as a minimization problem and AOA is applied to optimize the cost function. Fig.1 shows the magnitude response of the high pass filter of order 15. Fig. 2 shows the magnitude response of the high pass filter of order 23. The magnitude responses of the FIR high pass filter are for PM, CSA[5] and AOA, which describes how AOA shows better response than CSA and PM. From the same, we can infer the values of attenuation at stop-band (As) and pass-band ripple (Rp) as provided in Table I. For the high-pass filter of order (N) 15, the population size (NP) is noted as 70 and the number of functional evolution (NOFE) is 50,000. For order (N) 23, the population size (NP) is noted as 70 and the number of functional evolution (NOFE) is 150,000.

TABLE I. SIMULATION RESULTS FOR HPF ($\omega_p = 0.6\pi, \omega_s = 0.4\pi$)

N	NP	NOFE	Method	A_s(in dB)	R_P(in dB)
15	70	50000	PM	32.9706	0.2547
			CSA	40.8878	0.2492
			AOA	41.2413	0.1679
23	70	100000	PM	43.9873	0.1945
			CSA	46.0194	0.2484
			AOA	49.4328	0.0781

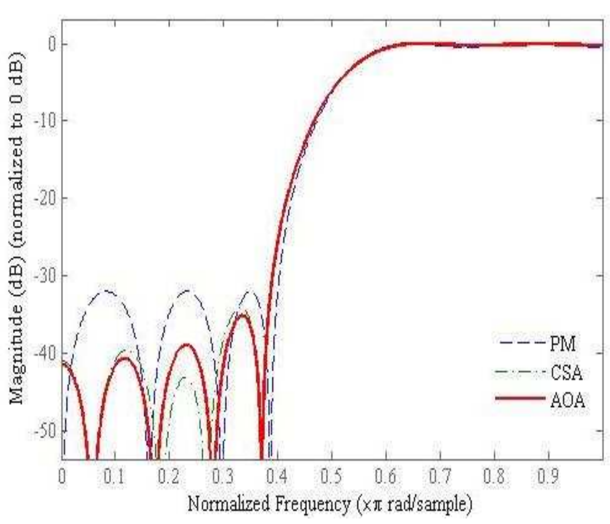


Fig. 2. Response of HPF with order (N)=15

Fig. 2 and Fig. 3 represent the magnitude responses of the high-pass filter of order 15 and 23. The red colour response is denoted as the performance of the filters using AOA algorithm and the blue colour response is denoted as the performance of the filters using PM algorithm and the green colour response is denoted as the performance of the filters using the CSA algorithm. It is clear from the responses that the performance of the filter using the arithmetic optimization algorithm (AOA) is much better than the CSA and PM algorithm. The attenuation band and the pass-band ripple of the filter of their respective orders represent much better value using the AOA algorithm than using the CSA and PM algorithm.

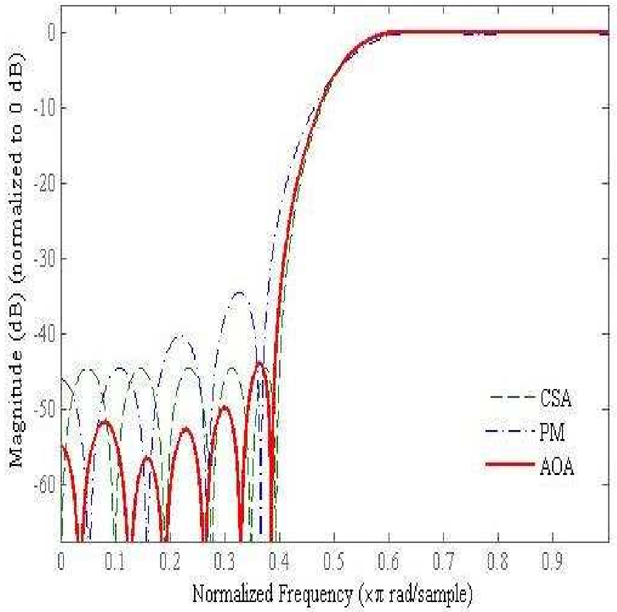


Fig. 3 Response of HPF with order (N)=23

TABLE II. DEVICE UTILIZATION

		Components					
		Slice Registers	Slice UTs	No. of FF	Bonded IOBs	No. of BUFG	No. of DSP481A's
N=15	Available	18224	9112	854	186	16	32
	PM	368	485	0	35	1	24
	CSA	360	330	80	35	1	24
	AOA	352	315	71	35	1	24
N=23	Available	18224	9112	854	186	16	32
	PM	240	614	0	35	1	24
	CSA	235	415	105	35	1	24
	AOA	224	366	90	35	1	24

TABLE III.POWER UTILIZATION

Type of Power	N= 15			N= 23		
	PM	CSA	AOA	PM	CSA	AOA
Supply Power(mW)	20.38	20.39	19.62	20.47	20.38	20.32
Dynamic Power(mW)	0.49	0.47	0.42	0..49	0.47	0.42
Static Power(mW)	19.89	19.91	19.20	19.98	19.91	19.90

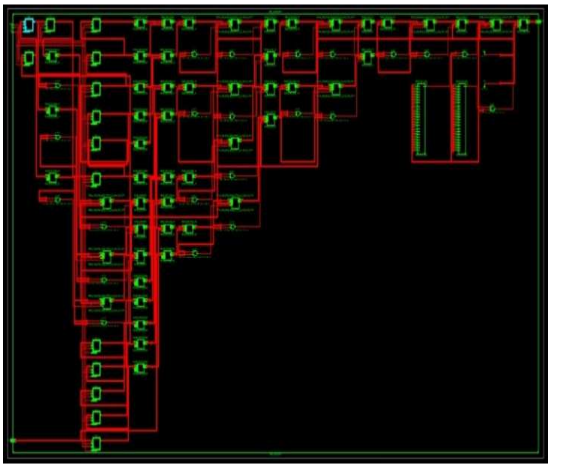


Fig. 4 RTL schematic of 15-tap HPF

Fig. 5 Timing diagram of 15-tap HPF

The data in above Table II and Table III show the device and power utilization of the FPGA kit for PM, CSA and AOA respectively. It can be inferred from both the tables that AOA requires much less power and consumes the minimum device area or components as compared to the CSA and PM algorithms and hence its implementation is on FPGA using AOA is more effective. Fig. 4 and Fig. 5 represent the RTL diagram and timing diagram of the high-pass filter of order

978-1-6654-6658-5/22/$31.00 ©2022 IEEE

15 respectively. Again, the design of the filter done by AOA exhibits much better performance than derived by using the CSA and PM algorithm.

V. CONCLUSION

In this paper a type-1 high-pass filter has been designed using the AOA algorithm. The simulation results and magnitude response of the FIR filter justify the betterment of AOA over CSA and PM algorithm. The attenuation value (As) and the pass-band ripple (Rp) provided in the respective tables calculated for the design of high-pass filter for different orders prove that the values derived by using AOA algorithm are much better than the values derived by using the CSA and PM algorithm. Further, the FPGA-based design of optimal FIR high-pass filter using AOA is implemented and its respective RTL and timing diagram are given. The values derived by using AOA in the Xilinx ISE design suite have been presented in the above tables are also compared with the values derived from CSA and PM algorithm.

VI. REFERENCES

[1] J. G. Proakis and D. G. Manolakis, "Digital signal processing," 3rd ed., Pearson Education, 2007.

[2] X. X. Zheng, J. Yang, S. Y. Yang, W. Chen, L. Y. Huang and X. Y. Zhang, "Synthesis of Linear-Phase FIR Filters With a Complex Exponential Impulse Response," in IEEE Transactions on Signal Processing, vol. 69, pp. 6101-6115, 2021.

[3] A. Sikder, P. Bhattacharjee, S. Chowdhury, S. Dhabal, and P. Venkateswaran, "Design of Band-pass FIR Filter using Improved Sine Cosine Algorithm and its Implementation on FPGA," pp. 310-315, 2019.

[4] N. Kumari and P. Jaglan, "Design Of FIR Low Pass Filter Using Particle Swarm Optimization Algorithm," Vol. 1, No. 3, pp. 31-35, 2017.

[5] A. Mukherjee, N. Chakraborty, and B. K. Das, "Whale optimization algorithm: An implementation to design low-pass FIR filter," Innovations in Power and Advanced Computing Technologies (i-PACT), pp. 1-5, 2017.

[6] S. Dhabal and P. Venkateswaran, "An efficient gbest-guided Cuckoo Search algorithm for higher order two channel filter bank design, Swarm and Evolutionary Computation," vol. 33, pp.- 68-84, 2017.

[7] R. Zheng, H. Jia, L. Abualigah, Q. Liu, S. Wang, "An improved arithmetic optimization algorithm with forced switching mechanism for global optimization problems[J]," Mathematical Biosciences and Engineering, Vol. 19(1), pp: 473-512, 2022.

[8] L. Abualigah, A. Diabat, S. Mirjalili, M. A. Elaziz, and A. H. Gandomi, "The Arithmetic Optimization Algorithm," Computer Methods in Applied Mechanics and Engineering, Vol. 376, 2021.

[9] Li, XD., Wang, JS., Hao, WK. *et al.* "Chaotic arithmetic optimization algorithm," *Appl Intell* , 2022.

[10] S. Mandal, S. P. Ghoshal, R. Kar, and D. Mandal, "Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique," Journal of King Saud University - Computer and Information Sciences, Vol. 24, No. 1, pp. 83-92, 2012.

[11] P. Bertsias and C. Psychalinos, "Differentiator based fractional-order high-pass filter designs," 7th International Conference on Modern Circuits and Systems Technologies (MOCAST), pp. 1-4, 2018.

[12] S. Chauhan and G. Vashishtha, "Mutation-based Arithmetic Optimization Algorithm for Global Optimization," International Conference on Intelligent Technologies (CONIT), pp. 1-6, 2021.

[13] L. Abualigah, A. Diabat, P. Sumari, and A.H. Gandomi, "A Novel Evolutionary Arithmetic Optimization Algorithm for Multilevel Thresholding Segmentation of COVID-19 CT Images," 2021.

[14] M. Premkumar et al., "A New Arithmetic Optimization Algorithm for Solving Real-World Multiobjective CEC-2021 Constrained Optimization Problems: Diversity Analysis and Validations," in IEEE Access, vol. 9, pp. 84263-84295, 2021.

[15] N. Panga, U. Sivaramakrishnan, R. Abishek, K. Bingi, and J. Chaudhary, "An Improved Arithmetic Optimization Algorithm," IEEE Madras Section Conference (MASCON), pp. 1-6, 2021.

[16] D. Izci, S. Ekinci, E. Eker, and A. Dündar, "Improving Arithmetic Optimization Algorithm Through Modified Opposition-based Learning Mechanism," 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), pp. 1-5, 2021.

[17] S. Ma, H. Huang, and Z. Wu, "The Arithmetic Optimization Algorithm Based-on Particle Energy and Particle Diffusion," International Conference on Wireless Communications and Smart Grid (ICWCSG), Hangzhou, China, pp. 246-253, 2021.

[18] C. Guo-wei and W. Feng-ying, "The implementation of FIR low-pass filter based on FPGA and DA," pp. 604-608, 2013.

[19] O. KalaiPriya, S. Ramasamy, and D. Ebenezer, "VLSI implementation of nonlinear variable cutoff high pass filter algorithm," 3rd International Conference on Electronics Computer Technology, pp. 275-278, 2011.

[20] K. A. Toker, S. Özen, and A. Arsal, "FPGA implementation of a low-complexity fading filter for multipath Rayleigh fading simulator," XXXth URSI General Assembly and Scientific Symposium, pp. 1-4, 2011.

[21] Y. Zhou and P. Shi, "Distributed Arithmetic for FIR Filter implementation on FPGA," International Conference on Multimedia Technology, pp. 294-297, 2011.

[22] A. Das, S. Dash, A. K. Sahoo, and B. C. Babu, "Design and implementation of FPGA based linear all digital phase-locked loop," Annual IEEE India Conference (INDICON), pp. 280-285, 2012.

[23] A. Singhal, A. Goen, and T. Mohapatra, "FPGA implementation and power efficient CORDIC based ADPLL for signal processing and application," 7th International Conference on Communication Systems and Network Technologies (CSNT), pp. 325-329, 2017.

[24] P. K. Meher, S. Chandrasekaran, and A. Amira, "FPGA Realization of FIR Filters by Efficient and Flexible Systolization Using Distributed Arithmetic," in *IEEE Transactions on Signal Processing*, Vol. 56, No. 7, pp. 3009-3017, 2008.

[25] T. Dutta, R. M. Aich, S. Dhabal, and P. Venkateswaran, "Finite Impulse Response Filter Design using Grasshopper Optimization Algorithm and Implementation on FPGA," IEEE Applied Signal Processing Conference (ASPCON), pp. 313-317, 2020.

[26] J. B. Evans, "Efficient FIR filter architectures suitable for FPGA implementation," in IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 41, no. 7, pp. 490-493,1994.

[27] M. A. Eshtawie and M. Othman, "FPGA Implementation of an Optimized Coefficients Pulse Shaping FIR Filters," IEEE International Conference on Semiconductor Electronics, pp. 454-458, 2006.