FPGA USING LMS ALGORITHM

A MINOR PROJECT - III REPORT

Submitted by

SUBASH P	**927621BEC213**
TEJNITHISH S	**927621BEC226**
VENKAT RAMAN P	**927621BEC238**
YOGESHWARAN S	**927621BEC246**

BACHELOR OF ENGINEERING

in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous)

KARUR – 639 113

OCTOBER 2023

M.KUMARASAMY COLLEGE OF ENGINEERING, KARUR

BONAFIDE CERTIFICATE

Certified that this **18ECP105 - Minor Project III** report "**FPGA USING LMS ALGORITHM"** is the bonafide work of "**SUBASHP (927621BEC213),TEJNITHISH S (927621BEC226), VENKAT RAMAN P (927621BEC238), YOGESHWARAN S (927621BEC246)** who carried out the project work under my supervision in the academic year **2023-2024 - ODD**.

SIGNATURE	**SIGNATURE**
Dr.A.KAVITHA B.E., M.E., Ph.D.,	**Mr.K.SUDHAKAR.,M.E.,**
HEAD OF THE DEPARTMENT,	**SUPERVISOR,**
Professor,	Assistant Professor,
Department of Electronics and	Department of Electronics and
Communication Engineering,	Communication Engineering,
M.Kumarasamy College of Engineering,	M.Kumarasamy College of Engineering,
Thalavapalayam,	Thalavapalayam,
Karur-639113.	Karur-639113.

This report has been submitted for the **18ECP105 – Minor Project-III** final review held at M. Kumarasamy College of Engineering, Karur on________________________.

PROJECT COORDINATOR

INSTITUTION VISION AND MISSION

Vision

To emerge as a leader among the top institutions in the field of technical education.

Mission

M1: Produce smart technocrats with empirical knowledge who can surmount the global challenges.

M2: Create a diverse, fully -engaged, learner -centric campus environment to provide quality education to the students.

M3: Maintain mutually beneficial partnerships with our alumni, industry and professional associations

DEPARTMENT VISION, MISSION, PEO, PO AND PSO

Vision

To empower the Electronics and Communication Engineering students with emerging technologies, professionalism, innovative research and social responsibility.

Mission

M1: Attain the academic excellence through innovative teaching learning process, research areas & laboratories and Consultancy projects.

M2: Inculcate the students in problem solving and lifelong learning ability.

M3: Provide entrepreneurial skills and leadership qualities.

M4: Render the technical knowledge and skills of faculty members.

Program Educational Objectives

PEO1: Core Competence: Graduates will have a successful career in academia or industry associated with Electronics and Communication Engineering

PEO2: Professionalism: Graduates will provide feasible solutions for the challenging problems through comprehensive research and innovation in the allied areas of Electronics and Communication Engineering.

PEO3: Lifelong Learning: Graduates will contribute to the social needs through lifelong learning, practicing professional ethics and leadership quality

Program Outcomes

PO 1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO 2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO 4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO 5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO 6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO 7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO 8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO 11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes

PSO1: Applying knowledge in various areas, like Electronics, Communications, Signal processing, VLSI, Embedded systems etc., in the design and implementation of Engineering application.

PSO2: Able to solve complex problems in Electronics and Communication Engineering with analytical and managerial skills either independently or in team using latest hardware and software tools to fulfill the industrial expectations.

Abstract	Matching with POs, PSOs
FPGA **LMS**	**PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10, PO11, PO12, PSO1, PSO2**

ACKNOWLEDGEMENT

Our sincere thanks to **Thiru.M.Kumarasamy, Chairman** and **Dr.K.Ramakrishnan, Secretary** of **M.Kumarasamy College of Engineering** for providing extraordinary infrastructure, which helped us to complete this project in time.

It is a great privilege for us to express our gratitude to **Dr.B.S.Murugan., B.Tech., M.Tech., Ph.D., Principal** for providing us right ambiance to carry out this project work.

We would like to thank **Dr.A.Kavitha**,**M.E., B.E., Ph.D., Professor and Head**, **Department of Electronics and Communication Engineering** for his unwavering moral support and constant encouragement towards the completion of this project work.

We offer our wholehearted thanks to our **Project Supervisor, Mr.K.Sudhakar**, **M.E, Assistant Professor** Department of Electronics and Communication Engineering for his precious guidance, tremendous supervision, kind cooperation, valuable suggestions, and support rendered in making our project successful.

We would like to thank our **Minor Project Co-ordinator**, **Dr.K.Karthikeyan, B.E., M.Tech., Ph.D., Associate Professor**, Department of Electronics and Communication Engineering for his kind cooperation and culminating in the successful completion of this project work. We are glad to thank all the Faculty Members of the Department of Electronics and Communication Engineering for extending a warm helping hand and valuable suggestions throughout the project. Words are boundless to thank our Parents and Friends for their motivation to complete this project successfully.

ABSTRACT

In this paper, the fundamental algorithm of noise cancellation, Least Mean Square (LMS) algorithm is studied and enhanced with adaptive filter. **The simulation of the noise cancellation using LMS adaptive filter algorithm is developed**. The noise **corrupted speech signal** and the **engine noise signal** are used as inputs for LMS adaptive filter algorithm. The filtered signal is compared to the original noise-free speech signal in order to highlight the level of attenuation of the noise signal. The result shows that the noise signal is successfully cancelled by the **developed adaptive filter**. The difference of the noise-free speech signal and filtered signal are calculated and the outcome implies that the filtered signal is approaching the noise-free speech signal upon the adaptive filtering. **The frequency range of the successfully cancelled noise by the LMS adaptive filter algorithm is determined by performing Fast Fourier Transform (FFT) on the signals. The LMS adaptive filter algorithm shows significant noise cancellation at lower frequency range.** The LMS algorithm is a widely used adaptive filter technique in noise cancellation. It operates by iteratively adjusting filter coefficients to minimize the mean square error between the output of the filter and the desired signal. This approach makes it highly effective for reducing various types of noise in signals, including both stationary and non-stationary noise. To assess the effectiveness of the LMS adaptive filter, we designed a simulation environment. We utilized a noise-corrupted speech signal as our input, representing the signal that requires noise cancellation. The engine noise signal served as the reference for the noise source. The LMS adaptive filter algorithm was applied to the noisy signal to produce a filtered output.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACRONYM		ABBREVIATION
FPGA	-	Field Programmable Gate Array
FIR	-	Finite Impulse Response
LES	-	Linear Equally Spaced
DDC	-	Digital Down Converts
RLS	-	Recursive Least Square
LEC	-	Line Echo Cancellation
SNR	-	Single-to- Noise Ratio
MSE	-	Mean Squared Error
LMS	-	Least Mean Square

CHAPTER 1

INTRODUCTION

Field-Programmable Gate Arrays (FPGAs) have emerged as versatile hardware platforms for implementing real-time digital signal processing algorithms due to their reconfigurable nature and parallel processing capabilities. This project report presents an FPGA-based implementation of the Least Mean Squares (LMS) algorithm, a fundamental adaptive filtering technique, which has a wide range of applications in areas such as communications, audio processing, and control systems. The LMS algorithm, at its core, is a dynamic method for adjusting filter coefficients in real-time to minimize the mean squared error between a desired signal and the filter's output. Its versatility and adaptability render it invaluable in various fields, making it an essential choice for numerous real-time signal processing tasks. From enhancing communication quality to refining audio outputs and optimizing control systems, the LMS algorithm has proven its worth repeatedly. FPGAs are well-suited for the implementation of the LMS algorithm for several compelling reasons. Their inherent parallel processing capabilities enable them to execute complex tasks more efficiently. Instead of processing data sequentially, FPGAs distribute the workload across multiple logical elements, dramatically reducing processing time. This feature is particularly beneficial for real-time applications where speed is of the essence. One of the most distinctive advantages of FPGAs is their reconfigurable nature. This means that their hardware configuration can be adapted on the fly to cater to specific processing needs. In the context of the LMS algorithm, this means that the FPGA can continually adjust its configuration to optimize filter coefficients as the input data changes. In dynamic environments, where signal characteristics may vary over time, this adaptability is invaluable.

1.1 PROJECT DETAILS

In this project, we will implement the LMS algorithm to perform adaptive noise cancellation on an audio signal. The goal is to design and develop a system that can remove unwanted noise from a corrupted audio signal. This project involves several key components:

1.Signal Generation:

You may need to generate a synthetic noisy audio signal as your input data, or you can use real-world audio recordings with added noise.

2.LMS Algorithm Implementation:

Implement the LMS algorithm, which is a gradient-based optimization technique. The LMS algorithm aims to minimize the mean squared error between the clean signal and the filtered noisy signal. You'll need to design and code the LMS filter.

3.Noise Estimation:

The LMS algorithm requires an estimate of the noise. You'll need to develop a mechanism to estimate and adaptively update the noise characteristics in the input signal.

4.Evaluation:

Measure the performance of your noise cancellation system. Common evaluation metrics include signal-to-noise ratio (SNR) improvement, mean squared error (MSE), and audio quality assessment metrics.

5.User Interface (Optional):

If the project allows, you can create a user-friendly interface for users to upload their audio recordings and apply the noise cancellation.

1.2 DESCRIPTION

Objective of this project aims to develop a real-time noise cancellation system using the LMS algorithm. The system will be capable of reducing background noise from an audio signal, improving the signal's quality for various applications like voice communication, audio recording, and more.

1.2.1 Data Acquisition:

Obtain audio input from a microphone or audio source. You may consider implementing this on a microcontroller, smart phone app, or desktop application.

1.2.2 LMS Algorithm Implementation:

Develop the LMS algorithm for adaptive noise cancellation. The core of this project is the implementation of the LMS filter that adjusts filter coefficients in real-time to minimize the mean squared error between the noisy input signal and the estimated noise component.

1.2.3 Adaptive Noise Estimation:

Design a method for estimating the characteristics of the background noise. The LMS algorithm requires an adaptive estimate of the noise to perform effective cancellation. You may use a reference microphone or an initial noise estimation.

1.2.4 Real-time Processing:

Implement real-time signal processing to ensure that noise cancellation is performed on-the-fly. This is crucial for applications where immediate noise reduction is needed.

1.2.5 Performance Metrics:

Define appropriate metrics for evaluating the performance of the noise cancellation system. Common metrics include Signal-to-Noise Ratio (SNR) improvement, Mean Squared Error (MSE), and audio quality assessment metrics.

1.2.6 User Interface:

Create a user-friendly interface to control the noise cancellation system. This interface could allow users to adjust settings, visualize the input and output signals, and monitor system performance.

CHAPTER-2

LITERATURE SURVEY

2.1 Models for adaptive noise cancellation problem:

The most commonly-used algorithm to design an adaptive filter is the least mean square (LMS) algorithm, originally developed by Widrow and Hoff [5]. The LMS algorithm is based on the principle of the steepest descent algorithm with minimum mean square error. The most important benefit of the LMS algorithm is that it does not require exact measurements of the gradient vector, nor does it require matrix inversion. The LMS algorithm is used to solve the Wiener-Hoff equation by searching for the optimal coefficients' weights for an adaptive filter. Another main advantage of the LMS algorithm is its computational simplicity, ease of implementation, and unbiased convergence. A block diagram of an adaptive noise cancellation system is shown in Fig. 2.1.

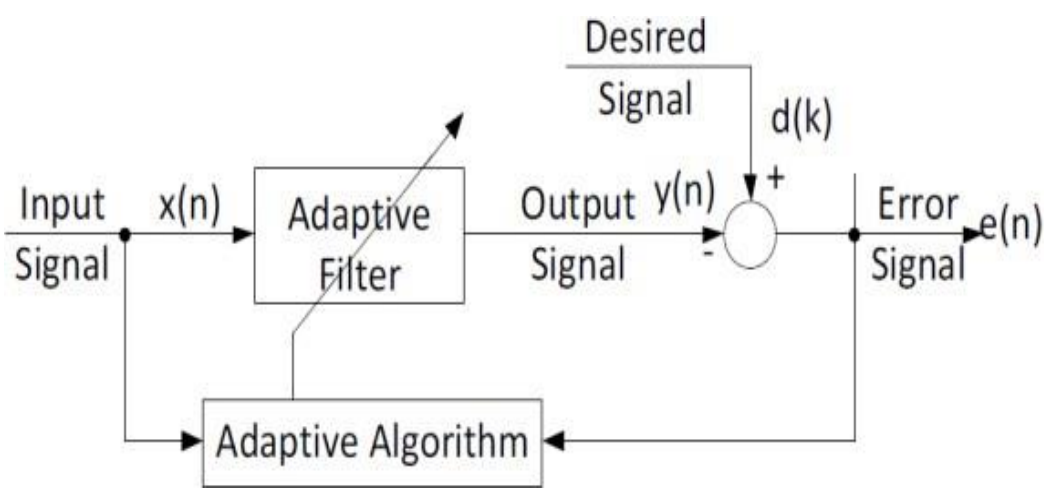

Fig. 2.1 Block diagram of a typical adaptive noise cancellation system

The vector $XX(kk)$ denotes an input vector with time delay and $xx(kk)$ is the input value at time kk,

i.e.: $XX(kk) = [xx(kk)\ xx(kk-1)\ \ldots\ xx(kk-NN+1)]TT$ (1)

Where [ ]TT denotes the transpose operation. The vector $WW(kk)$ is used to represent the weights applied to the filter coefficients at at time kk and is given as

i.e.: $WW(kk) = [WW0(kk)\ WW1(kk)\ \ldots\ WWNN-1(kk)\ ]\ T$----- (2)

In Fig. 1, the step size parameter $\mu\mu$ is the step size of the adaptive filter, and $ee(kk)$ is the error between the desired response $dd(kk)$ and the output of the filter $yy(kk)$, i.e., the filtered signal, at time kk. The pseudo code of an LMS algorithm is described in Table 1. The algorithm is adopted to update the coefficients of a finite impulse response (FIR) filter.

The LMS algorithm updates its coefficients through the minimization of the mean of the instantaneous squared error denoted by $EE[ee2(kk)]$. While $XX(kk)$ and $WW(kk)$ are assumed to be independent, the LMS algorithm assumes that $xx(kk)$ and $dd(kk)$ are wide-sense stationary ergodic processes, and therefore their means and variances are constant. The iterative weight update procedure of the LMS algorithm is given as:

$$WW(kk+1) = WW(kk) + 2\mu\mu\mu\mu(kk)XX(kk)\text{------------ (3)}$$

The selection of the step size parameter$\mu\mu$ plays an important role in updating of the system coefficients and thus can affect the system performance. While a relatively small $\mu\mu$ value could result in longer convergence time to find an optimal solution, selecting a larger $\mu\mu$ value may lead to unstable convergence and an output that may diverge.

For the consideration of stable behavior and convergence, the step size must be a small positive value ($\mu\mu$ << 1) and meet the following criteria

$$0 < \mu < 12*NN*RR \text{------------------------------}(4)$$

where NN is the number of filter taps and RR is the input signal covariance matrix defined as

$$RR = EE(XX(nn) * XXTT(nn)) \text{-----------------}(5)$$

2.2 Critical component in applications

Noise Cancellation:

LMS algorithms can be employed to suppress unwanted noise in audio signals, improving the quality of audio communication and recording systems.

Channel Equalization:

In wireless communication systems, LMS algorithms are used to mitigate the effects of multipath fading, thus improving signal quality and reliability.

Adaptive Beamforming:

LMS algorithms are utilized in array processing for optimizing the directivity of antennas, enhancing the reception of desired signals while suppressing interference.

CHAPTER-3

EXISTING SYSTEM

3.1 FPGA IMPLEMENTATION

Developing a hardware architecture and control logic to execute the LMS algorithm on an FPGA.

1. Performance Optimization:

Exploring techniques for optimizing the FPGA design to meet real-time processing requirements.

2. Validation:

Testing the FPGA implementation for its effectiveness in adaptive filtering applications, such as noise cancellation, echo cancellation, and equalization.

3. Resource Efficiency:

Analyzing resource utilization and power consumption to ensure the design is practical for deployment.

4. Documentation:

Providing a detailed project report that outlines the design process, challenges faced, and results obtained.

By achieving these objectives, this project aims to contribute to the field of real-time signal processing, demonstrating the feasibility and advantages of FPGA-based LMS algorithms.

The subsequent sections of this report will delve into the methodology, implementation details, experimental results, and conclusions, providing a comprehensive understanding of the FPGA-based LMS algorithm and its practical implications.

3.2 Traditional Echo Cancellation Techniques:

Traditional echo cancellation techniques are the early methods and approaches used to mitigate echo in telecommunications and audio communication systems. These methods were primarily used in analog communication systems and have evolved over time with the advent of digital signal processing. Here are some of the key traditional echo cancellation techniques

3.2.1 Line Echo Cancellation (LEC):

LEC is a technique used to mitigate echo in analog telephone networks. It typically employs techniques like echo suppressors and echo cancellers to reduce line echoes caused by impedance mismatches and signal reflections in long-distance telephony.

3.2.2 Adaptive Echo Cancellation Techniques:

Generate a reference signal by sending a known signal through the speaker and capturing it with the microphone. This reference signal will be used to adapt the filter coefficients.

- ❖ Apply the LMS algorithm to estimate the echo by comparing the reference signal with the received signal.
- ❖ Adjust the filter coefficients in real-time to minimize the error between the reference and received signals, effectively canceling out the echo.

3.2.3 Key Algorithms and Approaches:

Highlight some of the key algorithms and approaches used in existing systems, such as the Least Mean Squares (LMS) algorithm, Recursive Least Squares (RLS), and other adaptive filtering techniques.

3.2.4 Least Mean Squares (LMS) Algorithm:

The LMS algorithm is one of the most commonly used algorithms in echo cancellation. It adapts the coefficients of an adaptive filter to minimize the mean squared error between the received signal and the estimated echo. LMS is widely used due to its simplicity, low computational requirements, and effectiveness in many scenarios.

3.2.5 Recursive Least Squares (RLS) Algorithm:

The RLS algorithm is another adaptive filtering method that estimates and cancels echo by recursively updating the filter coefficients. It is known for its ability to adapt quickly to changing conditions and its high convergence speed. However, it tends to be more computationally intensive compared to the LMS algorithm.

3.2.6 Frequency-Domain Algorithms:

Algorithms such as the Frequency-Domain Adaptive Filter (FDAF) and the Partitioned Block Frequency-Domain Adaptive Filter (PB-FDAF) operate in the frequency domain. They can be effective in dealing with echo that varies across different frequency bands.

3.3 Summary of Existing System:

Echo cancellation is a critical technology in modern communication systems, aimed at improving audio quality and enhancing user experience. Existing echo cancellation systems encompass a range of traditional and digital signal processing methods. These systems are designed to mitigate the disruptive effects of echo in diverse communication scenarios, including telephony, video conferencing, voice over IP, and live public address systems.

The existing echo cancellation systems employ various techniques, including acoustic echo cancellation, line echo cancellation, and digital signal processing-based methods. They utilize adaptive filtering algorithms, such as the Least Mean Squares (LMS) and Recursive Least Squares (RLS) algorithms, to estimate and cancel echo in real-time. While these techniques have strengths, they also exhibit notable limitations and challenges.

Key limitations and challenges in existing systems include issues related to signal processing complexity, adaptability to changing acoustic environments, handling of double talk scenarios, adaptation speed, and dealing with non-linear acoustics. Additionally, network-induced latency, long echo tails, and privacy concerns pose significant challenges. Users often have limited control over system settings, and compatibility and integration issues can affect performance.

The need for innovations in echo cancellation is evident, especially in the face of dynamic communication scenarios and evolving technologies. This understanding of the limitations and challenges in existing systems serves as a compelling motivation for the development of a proposed system that leverages advanced algorithms, such as the LMS algorithm, to offer enhanced echo cancellation performance, adaptability, and user experience.

CHAPTER-4

PROPOSED SYSTEM

4.1 INTRODUCTION OF BEAMFORMING

Beamforming is an alternative name for spatial filtering where, with appropriate analog or digital signal processing, an array of antennas, can be steered in a way to block the reception of radio signals coming from specified directions. While a filter in the time domain combines energy over time, the beamformer combines energy over its aperture, obtaining a certain antenna gain in a given direction while having attenuation in others. Beamforming has been used for many years in different radio applications such as communications, surveillance, radar and, with different array sensors, in sonar and audio fields. The traditional analog way to perform beamforming was very expensive, and it was sensitive to component tolerances and drifts, while modern technology offers high speed A/D converters and Digital Down Converters (DDCs), fundamental blocks for digital beamforming. In both analog and digital domains the most common methods used to create directional beams are the time delay (time shift) and phase shift ones. The time delay approach allows to form and steer the beam by adding adjustable time delay steps that are independent from the operating frequency and bandwidth.

The normalised gain depends on the bandwidth B and the delay Δt (different time of arrival of the front wave at the antenna elements due to the physical dimension of the array) as reported in the following expression (normalised to 1):

$$G = \frac{\pi \sin(\pi B \Delta t)}{\pi B \Delta t} \text{ --------------(1)}$$

The linear equally spaced (LES) array has to be considered the most suitable configuration for a basic beamforming implementation. It can be easily noted that a LES is a spatial domain form of a temporal Finite Impulse Response (FIR) filter.The wave plane arrives at each of the antenna elements 1, 2, 3….N (here N=8) at different times, depending on the incidence angle α. In the beamforming phase, the individual delays have to be adjusted to make the signal arrive simultaneously on each antenna from angle α [1]. The required time delay for every antenna of the array is given by:

$$\Delta t = \frac{(n-1)\, d \sin \alpha}{c}$$

where ..,

c = is the speed of light

d = the element spacing

He final concept definition of multi beaming in the context of SKA is still under discussion. We underline that, at the present point of the discussion (ISAC comments, Nov.2002), two possible
definitions of multi beaming are accepted:

- Multi beaming : Multiple beams within the FOV (Fig.4.1) that can be obtained:

a) On line: Multiple beams obtained with analog or digital beamforming blocks.

b) Off Line: Multiple beams obtained with correlators and so, called synthesized beams

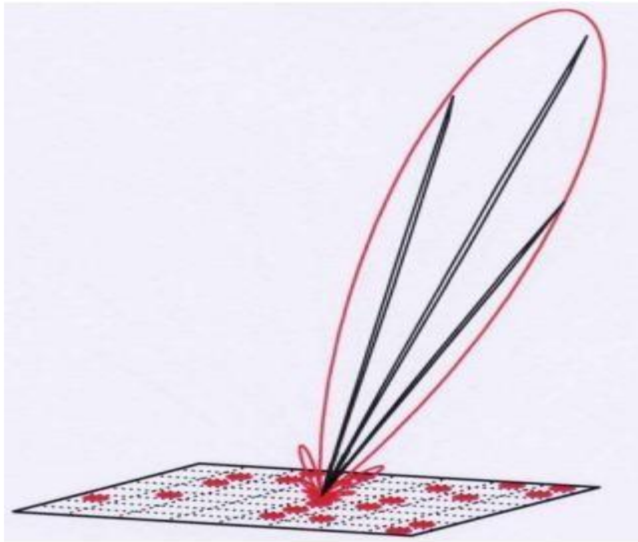

Fig.4.1Multiple beams within the FOV

Multibeaming (Multiple field of View): Multiple simultaneous FOV spaced on the sky (Fig.4.2) . In the Australian lens idea of SKA this is achieved by means of multiple feeds. In the European concept (tile design), multibeaming means splitting in N parallel parts every receiver channel. These are then used to form N independent beams. Multibeaming
conceived as multiple fields of view, leads to a more natural concept of multi-user.

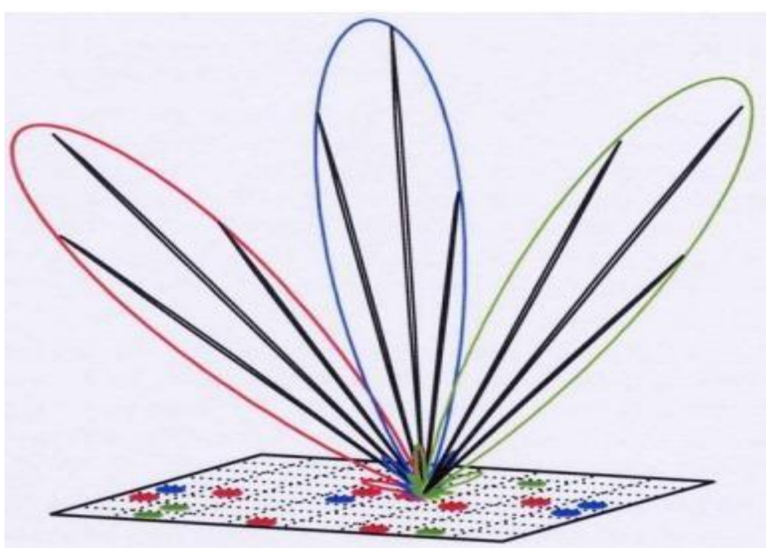

Fig.4.2 Multiple simultaneous FOV spaced on the sky

4.2 Program coding

```
%**********************************************************
% Beamforming_linear.m
%**********************************************************
% It is a MATLAB function that simulates beamforming for linear arrays.
%**********************************************************
% Credits:
% S. Bellofiore
% Zhiyong Huang
%_____________________________________________________

% Set output data format
format short e

% Generate output filename based on current time
cur_time = datevec(datestr(now));

ERR = 1;
save_in_file = input('\n Do you want to save reusults in file? ([n]/y):','s');
while(ERR ~= 0)
  if (isempty(save_in_file) | save_in_file == 'n')
     fprintf('------------Results will not be saved in file ----------- \n\n');
     ERR=0;
  elseif(save_in_file =='y')
     file_string = input('Input the desired output filename: ','s');
     ERR=0;
     fprintf('\n');
  else
     save_in_file=input('Do you want to save the results in file or not? ([n]/y):','s');
  end;
end
% Start timer
tic;

% Start recording
if (save_in_file=='y')
   out_file = sprintf('%s.txt',file_string);
   diary(out_file);
end
```

```
% User input
[N,d,sig,noise,type,nn,NN,AF_thresh,Mu,E_pattern] = linear_data_entry;

% Parameters initialization
FIG          = 'figure(1)';            % Figure to record
SKIP_STEP     = 40;                    % Plot every SKIP_STEP iterations
w            = zeros(N,1);             % iteration initialization

% Generate signals
[dd, X, fm] = linear_sig_gen(N,d,nn,NN,type,sig,noise,E_pattern);

for i = 1 : length(dd)
   w0 = w;
   [w, err(i)] = LMS(w,Mu,X(:,i),dd(i));
   mse(i) = sum(abs(err(i))^2);
   w_err(i) = norm(w0 - w);
   if i>1
      array_factor = linear_AF(N,d,w,sig(1:size(sig,1),2),E_pattern);
      if (abs(w_err(i) - w_err(i-1)) < eps) | (array_factor <= AF_thresh)
         linear_plot_pattern(sig,w,N,d,E_pattern,AF_thresh,'half',4,'-');
         break;
      end;
   end;
   if rem(i,SKIP_STEP) == 0            % Plot every SKIP_STEP iterations
      linear_plot_pattern(sig,w,N,d,E_pattern,AF_thresh,'half',4,'-');
      if i == SKIP_STEP
         FIG_HANDLE = eval(FIG);
      if (isunix)
        pause;
      end;
      end;
   end;
end;

% Final weights and betas
W    = abs(w);
beta = angle(w);
iterationnumber=i;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Output
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf('\n\n');
```

```
disp('*********************************************************
***********');
fprintf('\n');
disp('=== Signal Infomation ===');
string = sprintf('       Amplitude   Theta(degrees)'); disp(string);

[rowofsig,colofsig]=size(sig);
fprintf('SOI     %2.4f        %3d\n',sig(1,1),sig(1,2));
for i=2:rowofsig
   fprintf('SNOI%d   %2.4f        %3d\n',i-1,sig(i,1),sig(i,2));
end

if(isempty(noise))
   fprintf('\n');
   disp('NO noise.');
else
   fprintf('\n');
   disp('=== Noise Information ===');
   fprintf('Mean: %f, Variance: %f\n', noise(1), noise(2));
end

fprintf('\n');
disp('=== Iterations Number of Beamforming ===');              % Displays number
of iterations
fprintf('       %d\n',iterationnumber);

fprintf('\n');
disp('=== Weights (Amplitude) of Each Element ===');              % Displays
computed weights
disp('   Exact Value        Normalized');
for i=1:length(W)
   fprintf('%d   %f           %f\n',i,W(i),W(i)/W(1));
end

fprintf('\n');
disp('=== Beta (Phase in degrees) of Each Element ===');           % Displays
computed beta [Ensemble]
disp('      Exact Value          Normalized');
nbeta=unwrap(beta)*180/pi;
for i=1:length(beta)
   fprintf('%d   %12f         %12f\n',i,nbeta(i), mod((nbeta(i)-nbeta(1)),360));
end
```

```
fprintf('\nThe end.\n');
disp('********************************************************************
************');
fprintf('\n');

% Stop recording

if (save_in_file=='y')
   diary off;
   fprintf('Data saved in file %s.txt\n\n', file_string);
end


% Plot results
figure; plot(W/W(1),'*-'); grid;
xlabel('Antenna Element (n)');
ylabel('Excitation of Antenna Element');
title('Normalized Magnitude Distribution');

figure; plot(unwrap(beta)*180/pi,'*-'); grid;
xlabel('Antenna Element (n)');
ylabel('Phase of Antenna Element (degrees)');
title('Phase Distribution');

figure; plot(10*log10(mse)); grid;
xlabel('Iteration Number');
ylabel('MSE in dB');
title('Learning Curve');

figure; plot(10*log10(w_err)); grid;
xlabel('Iteration Number');
ylabel('Weight Error in dB');
title('Weight Estimation Error');

% Stop timer
disp('Elapsed Time (min):'); disp(toc/60);
```

CHAPTER-5

RESULT AND DISCUSSION

RESULT

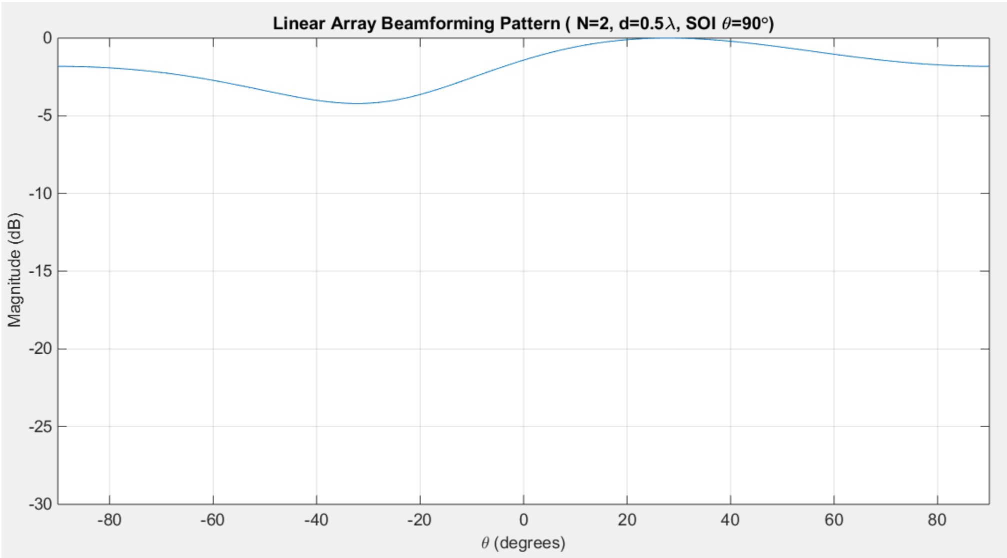

Fig 5.1 Linear Array Beamforming Pattern

X-Axis (Angle): The horizontal axis typically represents the angle in degrees or radians. It covers the range of directions from which signals can arrive at the linear array.

Y-Axis (Amplitude): The vertical axis represents the amplitude of the response of the linear array to signals arriving from different angles. It quantifies the strength of the array's response in each direction.

Main Lobe:

The main lobe is the central and typically strongest part of the beamforming pattern. It represents the direction to which the array is most sensitive. The width of the main lobe depends on the design and spacing of the array elements.

Side Lobes:

Side lobes are smaller, secondary lobes that appear on both sides of the main lobe. They indicate the sensitivity of the array to signals arriving from directions other than the main lobe direction.

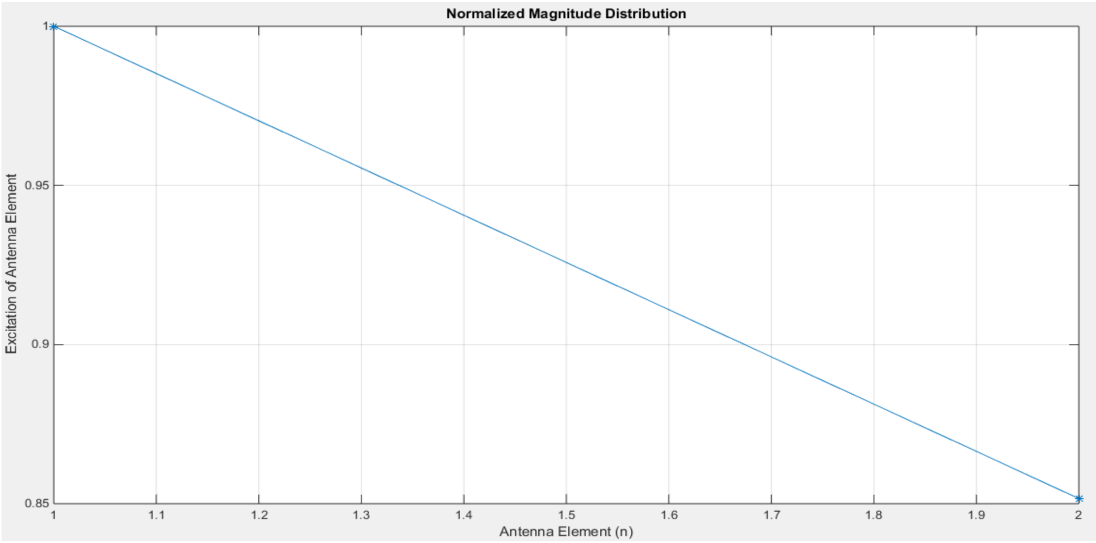

Fig 5.2 Normalized Magnitude Distribution

Filter Coefficients: In the LMS algorithm, filter coefficients (often represented as "w" for weights) are updated iteratively to minimize the error between the filter's output and a desired signal. These coefficients determine how the input signal is filtered or processed.

Normalized Magnitude: To create a normalized magnitude distribution, you calculate the magnitude of each filter coefficient (the absolute value) and then normalize it by dividing by the sum of all magnitudes. This normalization step ensures that the values lie within the range of 0 to 1.

Normalized Magnitude (M_norm) = |M_i| / Σ |M_i|, where M_i is the magnitude of the ith coefficient, and Σ |M_i| is the sum of the magnitudes of all coefficients.Distribution Graph: The distribution graph is a visual representation of the normalized magnitudes of the filter coefficients. Each coefficient's normalized magnitude is plotted on the graph, and it shows how the distribution evolves over time as the LMS algorithm adapts.

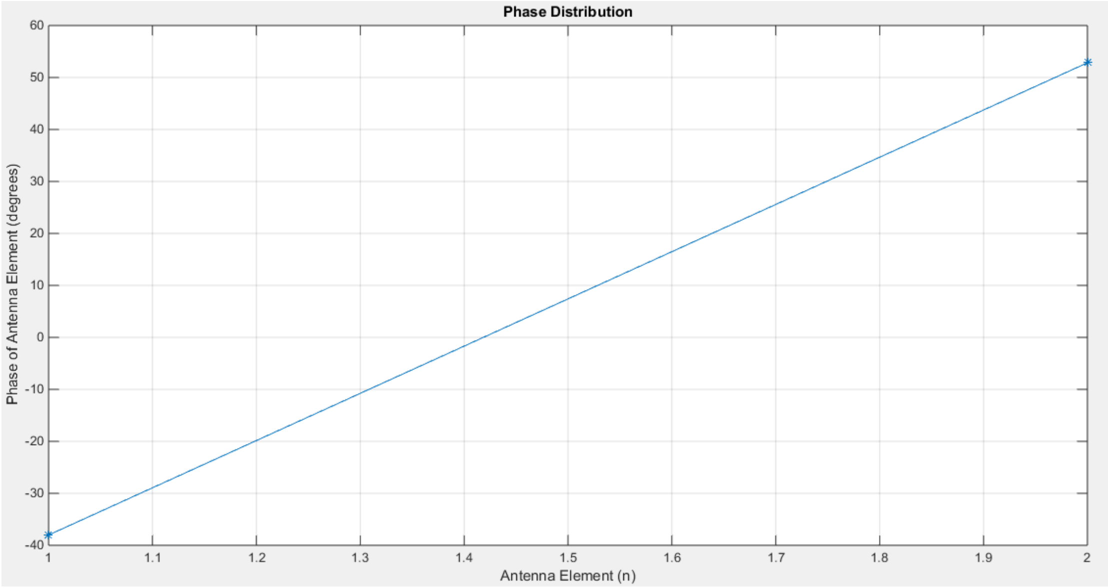

Fig 5.3 Phase Distribution

X-Axis (Iteration or Time): The horizontal axis typically represents the iteration number or time. It shows the progression of the LMS algorithm as it adapts its filter coefficients over time. Early iterations are on the left side of the graph, and as you move to the right, you're observing later iterations.

Y-Axis (Phase Angle): The vertical axis represents phase angles, often measured in radians or degrees. Phase angles describe the time shift or phase difference between the complex-valued filter coefficients and a reference signal. The reference signal could be the desired signal or another relevant reference point.

Phase Distribution Curve: The phase distribution curve on the graph shows the distribution of phase angles for the filter coefficients at each iteration or time step. The curve can help you understand how the filter weights are changing their phase relationships over time. A uniform distribution indicates that the coefficients are evenly distributed in terms of phase angles.

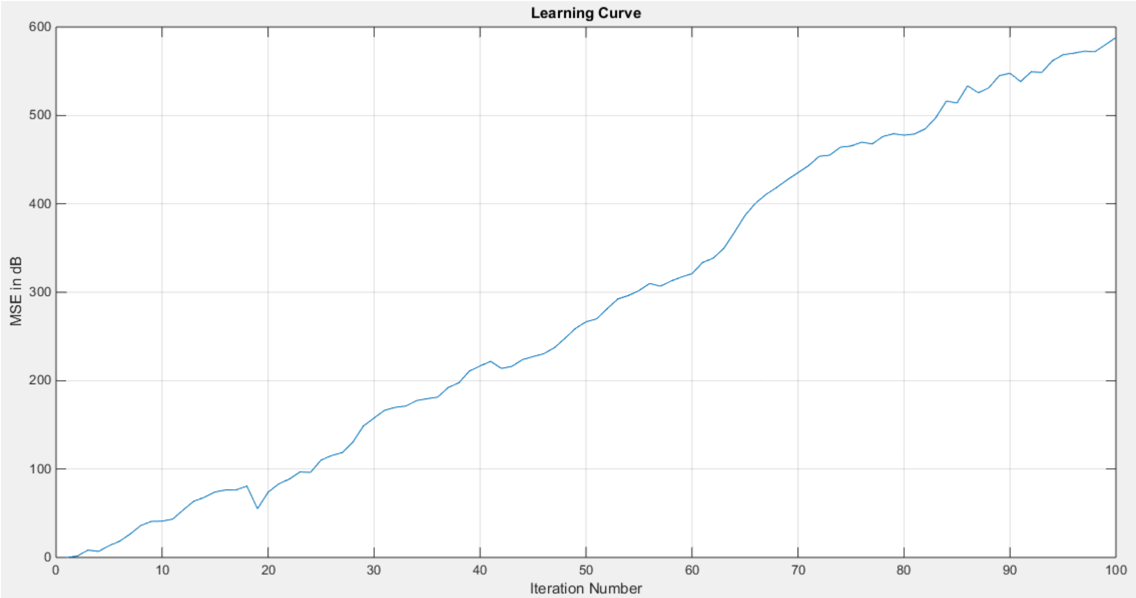

Fig 5.4 Learning Curve

X-Axis (Time or Iterations): The horizontal axis of the graph represents time or the number of iterations. It essentially shows the progression of the LMS algorithm as it updates its filter coefficients. Early iterations are on the left side of the graph, and as you move to the right, you're observing later iterations.

Y-Axis (Error Metric): The vertical axis typically represents an error metric, which is a measure of how well the filter output matches the desired signal. The error metric could be mean squared error (MSE) or some other relevant measure of signal quality. The goal of the LMS algorithm is to minimize this error, so you want the curve to trend downward.

Learning Curve: The learning curve itself is the line or curve on the graph that depicts how the error metric changes over time or iterations. It starts at some initial error level and should ideally decrease over time. The curve represents the learning process of the LMS algorithm as it iteratively updates its filter coefficients to minimize the error.

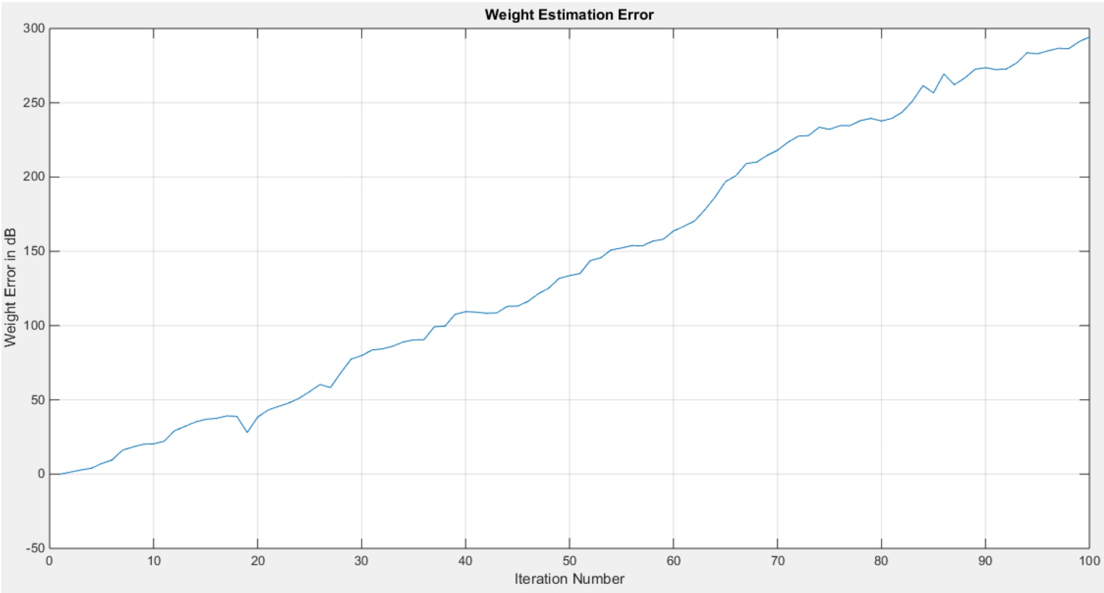

Fig 5.5 Weight Estimation Error

MS Algorithm: The Least Mean Squares (LMS) algorithm is a widely used adaptive filtering algorithm used in various signal processing applications. It is primarily used for updating the coefficients or weights of a linear filter to minimize the mean squared error (MSE) between the filter's output and a desired signal.

Filter Coefficients: In LMS, you have a set of filter coefficients (often represented as "w" for weights) that determine how the input signal is filtered or processed.

dB (Decibels): Decibels are a logarithmic unit of measurement used to express the ratio of two values, often with respect to signal power, amplitude, or intensity. When working with filter coefficients, expressing them in decibels can be useful for understanding the dynamic range of the coefficients and the scale of their adjustment.

Weight Graph in dB: To create a weight graph in dB, you'll typically perform the following steps.

CHAPTER-6

CONCLUSION AND FUTURE WORK

In this paper an adaptive filter system was successfully designed, and deployedwith software/hardware co-design for FPGA based systems. The adaptive filter system was analyzed using the MATLAB/Simulink model, and it later was automatically converted from floating point to fixed point for an Intellectual Property Core. This IP Core was placed in Vivado Synthesis Design for synthesis and implementation. Finally, the debugger was run before the audio file was fed in Zedboard. The design method can be applied to any type of FPGA under the Zynq family as long as this design is supported by the DSP-HDL Tool Support. The LMS Filter was processed and implemented to the FPGA board since it is supported by HDL. Experimental results show that the proposed hardware implementation method has a high degree of noise cancellation performance.

REFERENCES

[1] Wagdy H. Mahmoud and Nian Zhang, Software/Hardware Implementation of an Adaptive Noise Cancellation System, 120th ASEE Annual Conference & Exposition, Atlanta, GA, June 23-26, 2013.

[2] Nian Zhang, Investigation of Fault-Tolerant Adaptive Filtering for Noisy ECG Signals, 2007 IEEE Symposium on Computational Intelligence in Image and Signal Processing (CIISP), Honolulu, HI, pp. 177-182, April 1-5, 2007.

[3] M. I. Troparevsky, C. E. D'Attellis, On the convergence of the LMS algorithm in adaptive filtering, Signal Processing Vol. 84, pp. 1985-1988, October 2004.

[4] Ahmed Elhossini, Shawki Areibi, Robert Dony, An FPGA Implementation of the LMS Adaptive Filterfor Audio Processing, Proceedings of IEEE International Conference onreconfigurable Computing and FPGS's (ReConFig 2006), pp. 1-8, 2006.

[5] A. Rosado-Muñoz, M. Bataller-Mompe, E. Soria- Olivas, C. Scarante, J. F. Guerrero-Martínez, FPGA Implementation of an Adaptive Filter Robust to Impulsive Noise: Two Approaches, IEEE Transactions on Industrial Electronics, Vol. 58, No. 3, pp. 860-870, March 2011.

[6] P.Subash ,S.TejNithish,P. Venkat Raman,S.Yogeshwaran., FPGA USED LMS ALGORITHM , M. Kumarasamy College Of Enginerring, Electronics and Communication Engineering , Karur ,October 2023.

OUTCOME

PPG INSTITUTE OF TECHNOLOGY

INSTITUTION'S INNOVATION COUNCIL

INTERNATIONAL CONFERENCE

on

INNOVATIVE ENGINEERING AND TECHNOLOGY

05^{th} & 06^{th} October 2023

Organized by :

DEPARTMENT OF BIOMEDICAL ENGINEERING

3 GOOD HEALTH AND WELL-BEING

The SDG Accord

TÜV SÜD

CERTIFICATE OF PRESENTATION

This is to certify Dr/Mr/Ms Subash. P from M. Kumarasamy college of Engineering has presented a paper entitled FPGA using LMS algorithm. in the International conference on " Innovative Engineering and Technology " held at PPG institute of technology, Coimbatore on 05.10.2023 & 06.10.2023.

Organising Secretary Convenor Principal

INTERNATIONAL CONFERENCE

on

INNOVATIVE ENGINEERING AND TECHNOLOGY

05th & 06th October 2023

The SDG Accord

Organized by :

DEPARTMENT OF BIOMEDICAL ENGINEERING

CERTIFICATE OF PRESENTATION

This is to certify **Dr/Mr/Ms** Tejnithish. S from M. Kumarasamy college of Engineering has presented a paper entitled FPGA using LMS algorithm. in the International conference on " **Innovative Engineering and Technology** " held at **PPG** institute of technology, **Coimbatore** on **05.10.2023** & **06.10.2023.**

Organising Secretary

Convenor

Principal

PPG INSTITUTE OF TECHNOLOGY

INSTITUTION'S INNOVATION COUNCIL

INTERNATIONAL CONFERENCE

on

INNOVATIVE ENGINEERING AND TECHNOLOGY

05th & 06th October 2023

3 GOOD HEALTH AND WELL-BEING

The SDG Accord

Organized by :

DEPARTMENT OF BIOMEDICAL ENGINEERING

CERTIFICATE OF PRESENTATION

This is to certify **Dr/Mr/Ms** Venkat Raman. P from M. Kumarasamy college of Engineering has presented a paper entitled FPGA using LMS algorithm. in the International conference on " **Innovative Engineering and Technology** " held at **PPG institute of technology, Coimbatore** on **05.10.2023** & **06.10.2023.**

Organising Secretary	**Convenor**	**Principal**

PPG INSTITUTE OF TECHNOLOGY

INTERNATIONAL CONFERENCE

on

INNOVATIVE ENGINEERING AND TECHNOLOGY

INSTITUTION'S INNOVATION COUNCIL

05th & 06th October 2023

3 GOOD HEALTH AND WELL-BEING

The SDG Accord

Organized by :

DEPARTMENT OF BIOMEDICAL ENGINEERING

CERTIFICATE OF PRESENTATION

This is to certify **Dr/Mr/Ms** Yogeshwaran.s from M. kumarasamy college of Engineering has presented a paper entitled FPGA using LMS algorithm. in the International conference on " **Innovative Engineering and Technology** " held at **PPG institute of technology**, **Coimbatore** on **05.10.2023** & **06.10.2023.**

Organising Secretary

Convenor

Principal

FPGA USING LMS ALGORITHM

[1]Subash P, [2]Tejnithish S, [3]Venkat Raman P, [4]Yogeshwaran S

[1,2,3,4] Department of Electronics and Communication Engineering , M.Kumarasamy College of Engineering, Karur ,Tamil nadu, India.

[1]subashsivamani9595@gmail.com

Abstract:In this paper, the fundamental algorithm of noise cancellation, Least Mean Square (LMS) algorithm is studied and enhanced with adaptive filter. The simulation of the noise cancellation using LMS adaptive filter algorithm is developed. The noise corrupted speech signal and the engine noise signal are used as inputs for LMS adaptive filter algorithm. The filtered signal is compared to the original noise-free speech signal in order to highlight thelevel of attenuation of the noise signal. The result shows that the noise signal is successfully cancelled by the developed adaptive filter. The difference of the noise-free speech signal and filtered signal are calculated and the outcome implies that the filtered signal is approaching the noise-free speech signal upon the adaptive filtering. The frequency range of the successfully cancelled noise by the LMS adaptive filter algorithm is determined by performing Fast Fourier Transform (FFT) on the signals. The LMS adaptive filter algorithm shows significant noise cancellation at lower frequency range.