

# Design and Implementation of Acoustic Echo Cancellation on Xilinx Zynq for FPGA Co-simulation using LMS Adaptive Filter

<sup>1</sup>Dawit Mamo Alemu (M. Tech), <sup>2</sup>Dr. M. Madhavi Latha, Professor

<sup>1</sup>dave4ece@gmail.com, <sup>2</sup>mmadhavilatha@jntuh.ac.in

Electronics and Communication Engineering, Signal Processing Group  
Jawaharlal Nehru Technological University Hyderabad, Telangana, India

DOI: 10.29322/IJSRP.10.03.2020.p9903

<http://dx.doi.org/10.29322/IJSRP.10.03.2020.p9903>

**Abstract:** An adaptive filter is a filter that can adapt the given environment through self-learning process using optimization algorithm by adjusting the filter coefficient of the input signal. Adaptive filter is the very essence for today's different applications such as Echo Cancellation, Noise Cancellation, System Identification and so on. In this paper we present the design and implementation of acoustic echo cancellation using the art of Xilinx Zynq platform Zed board and by applying a model-based design approach for hardware software co-simulation in Simulink. From the original signal echo generating model is designed for multiple path delay. The echo version of the original signal will be removed by using LMS adaptive filter algorithm which is a simple and robust in performance. RTL (Register Transfer Level) design is generated from the subsystem of echo cancellation design model by converting to intellectual property (IP) core for FPGA implementation using HDL workflow advisor in Matlab Simulink, finally the custom board reference design from Vivado will apply for real time audio output test that integrated with the model design. The hardware software co-simulation shows that the proposed FPGA implementation has cancelled the echo with high performance.

**Key word:** acoustic echo, FPGA, adaptive filter, Matlab®, Simulink®, software/hardware-co-simulation

## I. INTRODUCTION

In the last decades, adaptive filter design and FPGA implementation has been a very active area of research and Innovation. Adaptive filter is a filter that can adapt the given environment through self-learning process using optimization algorithm by adjusting the filter coefficient of the input signal. The basic element in any adaptation filters is filter structure, performance criterion and adaptation rule. Adaptive filters have different real-world application in the area of signal processing and communications such as channel equalization, system identification, active noise cancellation, echo cancellation, speech coding, smart antenna system and so on. From those application acoustic echo cancellation is the most widely used. Since today's the number of hands-free communication users are increasing but still the problem of sound quality due to interference and echo are challenging the industry of hands-free communication. There are two types of echo one is a network echo and the other one is acoustic echo. Adaptive filter that used for acoustic echo cancellation is as shown in Figure.1. Echo is the distorted and delayed version of the original signal reflecting back to the source either from single path or multiple path propagation. Thus, it is desired to eliminate these echoes and the process is called echo cancellation as shown in the below Figure. 2. FPGA implementations are extremely important for the real time acoustic echo cancellation hardware software co-simulation using the Xilinx Zynq 7000 SOC which is a first in its kind that offer hardware, software and IO programmability in the single chip. Such hardware implementation can be realized using the hardware description language such as VHDL or Verilog and the hardware-software co-simulation can be implemented in Matlab-Simulink using model-based design approach that will simplify the work from specification to prototype on single platform. using the attribute of a modern FPGA chip design we can actually use a custom board reference design that will integrate seamlessly with existing intellectual property (IP) such as for audio codec that will allow as to import and play audio file to and through Zed board, this and many other reason makes FPGA software/hardware co-design choice for many real time applications. Therefore, it is crucial and important to design and implement a real time software hardware co-simulation of acoustic echo cancellation based on FPGA co-design.

The rest of the paper is arranged as follows. In Section 2, the proposed adaptive filter algorithm (LMS) and acoustic echo cancellation are discussed briefly. In Section 3, the hardware-software co-design implementation in FPGA Zynq using Zed board will be discussed in details. The hardware software co-simulation experimental result is presented in Section 4. Finally, the paper concluded in Section 5.

## II. ADAPTIVE FILTER FOR ACOUSTIC ECHO CANCELLATION

This Section discusses the adaptive filter LMS algorithm, echo and the processes of echo cancellation

### A. Least Mean Square Algorithm

The most widely used adaptive filter algorithm for acoustic echo cancellation application is the Least Mean Square algorithm which was developed by Widrow and Hoff in 1960. It is a member of stochastic gradient algorithm in which the cost function is minimized by adjusting the filter coefficients and it is a robust and at the same time computationally simple and high-performance during adaptation process. It can be used for solving the Weiner-Hopf equations without finding the inversion matrix and LMS algorithm has two basic processes: those are filtering process and adaptation process. The procedures in the adaptation process of LMS algorithm are described in the table below.

Table.1. Steps to implement LMS algorithm:

Initial condition	Filter length= M Step-size factor = $\mu$ , Input data to the adaptive filter = $x(n)$ $x(0) = w(0) = [0 \ 0 \ 0 \dots]^T$
For each instant of $n = 1, 2, 3, \dots$	
Compute the following	
Filter output	$y(n) = w^T(n) x(n) = x(n)^T w(n)$
Estimation error	$e(n) = d(n) - y(n)$
Tap-weight adaptation	$w(n + 1) = w(n) + 2\mu e(n) x(n)$

For the stability of LMS algorithm  $\mu$  must be chosen such that  $0 < \mu < \frac{2}{\text{tap-input power}}$

Where tap-input power =  $\sum_{k=0}^{M-1} E\{|x(n-k)|^2\}$  which in theory is equivalent to  $\lambda_{\max}$  where  $\lambda_{\max}$  is the maximum Eigen value of Correlation matrix (R) of input signal. Using this stability factor  $\mu$  will be chosen according to  $0 < \mu < \frac{2}{\lambda_{\max}}$ .

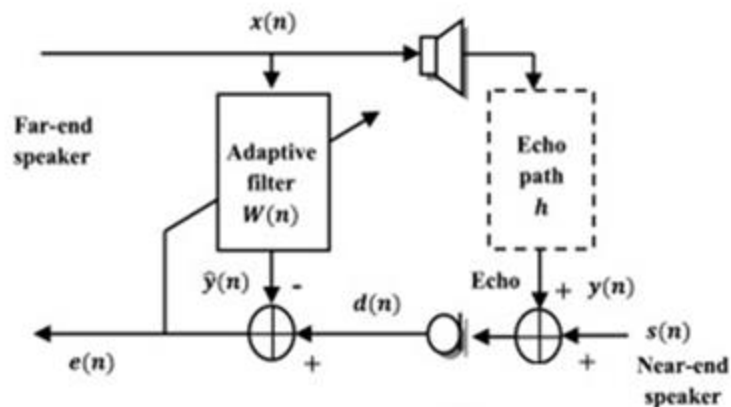


Figure.1. Acoustic Echo Cancellation

Where  $x(n)$  is far voice,  $h(n)$  is echo path  $y(n)$  is far voice plus echo,  $s(n)$  is near voice,  $d(n)$  is the sum of  $y(n)$  and  $s(n)$ ,  $\hat{y}(n)$  is estimated filter output and  $e(n)$  is the error. Note that  $s(n)$  is not considered in this work.

### B. Echo and Echo Cancellation

Echo can define as a phenomenon within which a delayed and distorted version of an original sound or electrical signal is reflected back to the source. The echoes of speech waves will be heard from different sources like floor, wall, and other neighbouring objects. In such a case when the reflected wave arrives after a couple of tens of milliseconds delay after the direct sound, it is heard as a clear echo. These echoes are bothering and will unexpectedly interrupt a conversation. Thus, it's desired to eliminate these echoes and therefore the process is called echo cancellation.

In telephone communication, there are two main kinds of echo: network and acoustic echoes. The network echo results from the impedance mismatch at points along the cable, for instance, at the hybrids of a Public Switched Telephone Network (PSTN) exchange, where the subscriber two-wire lines are connected to four-wire lines. Acoustic echo is because of the acoustic coupling between the loudspeaker and microphone in hands-free telephone systems. When far end speaker speak and that speech signal transmit to loudspeaker when the signal from loudspeaker takes different path and reflected back and picked by near end microphone then it'll transmit back to the source as an echo, we called these phenomena acoustic echo. The multipath propagation echo generation shown in Figure.2

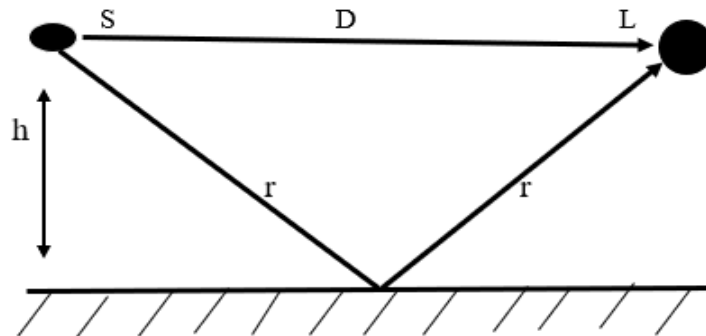


Figure.2. Multipath Propagation

Echo cancellation is as we mentioned in the above it is the process of removing the echo version of audio signal from the original audio. The echo cancellation process is started from knowing the originally transmitted audio that re-appear after some delay in the transmitted or received audio or speech signal. Once the echo is recognized it can easily subtract from the received or transmitted signal. Mostly this technique is implemented on DSP's and FPGA co-simulation.

### III. FPGA IMPLEMENTATION

In this Section we describe the Matlab-Simulink implementation model of acoustic echo cancellation algorithm and the hardware-software co-simulation of FPGA using HDL workflow advisor

#### A. Matlab-Simulink implementation model

##### 1. Echo model:

The given model shown in Figure.3 will generate the echo version of the original signal by considering the delay is due to multiple path scenario. Here we use PCM 16-bit signed, 16000Hz speech signal and from multimedia block, delay block, sum block, gain block, audio device writer and to workspace block are used to design the echo model. the delay parameter which is 4000 sample time for each path is describe the time for which the sound wave takes to travel from the source to the listener after the bouncing back off from reflecting objects whereas the gain is the signal loss due to propagation and reflection.

The transfer function of the single echo filter can be expressed as

$$H(z) = 1 + \alpha z^{-R}, |\alpha| < 1, \text{ Where } \alpha \text{ is the signal loss and } R \text{ is the time delay take by the sound wave.}$$

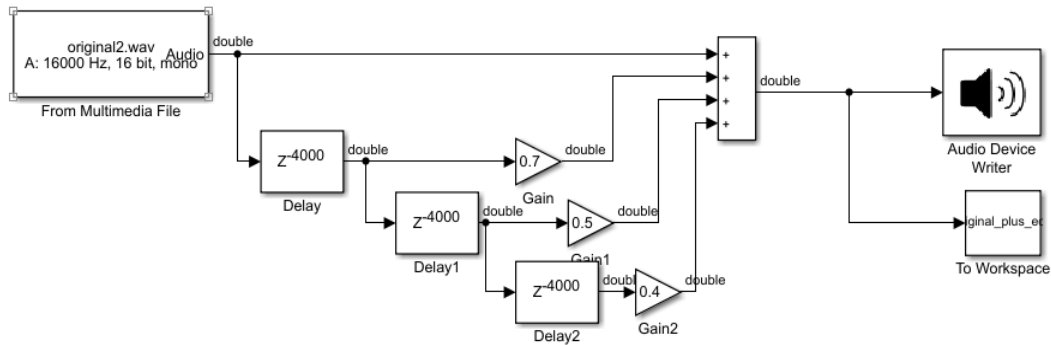


Figure.3. Echo Simulink model

**2. Echo cancellation model:**

We implemented the acoustic echo cancellation using LMS algorithm as shown in Figure.4 below. The only adaptive filter that Xilinx Zynq FPGA hardware support from DSP system toolbox HDL support is LMS filter block. since LMS filter is configurable to fit our design we assign signals  $x(n)$  for input signal which contains the original plus the echo speech signal  $d(n)$  for the desired signal which is the original speech signal from workspace block and  $e(n)$  for error signal which is the result of desired signal minus the input signal.

In order to generate the HDL code for the Simulink for FPGA implementation using HDL coder the input data type must be a fixed-point number by using the data type conversion block. Both the input and desired signal is converted to fixed point. For a visual inspection of a signal the input signal  $x(n)$ , the desired signal  $d(n)$  and the error signal  $e(n)$  is transmitted to time scope.

For audio playback of the output from the echo generation model and from the LMS output port we use two to workspace block and one audio device writer to hear output audio in the Simulink. In the process of updating the filter coefficient the selection of the right step-size is the most crucial part in order to achieve a better performance during echo cancellation using LMS algorithm in adaptive filter. The smaller the step-size the longer it takes to converge to optimal solution the higher the step-size the faster the convergence rate but it leads to unstable. The overall filtering and adaptation process of the LMS algorithm is taken place according to the step wise equation of which is given in the above Table.1 and the complete Simulink model for acoustic echo cancellation is as shown in Figure.5 below.

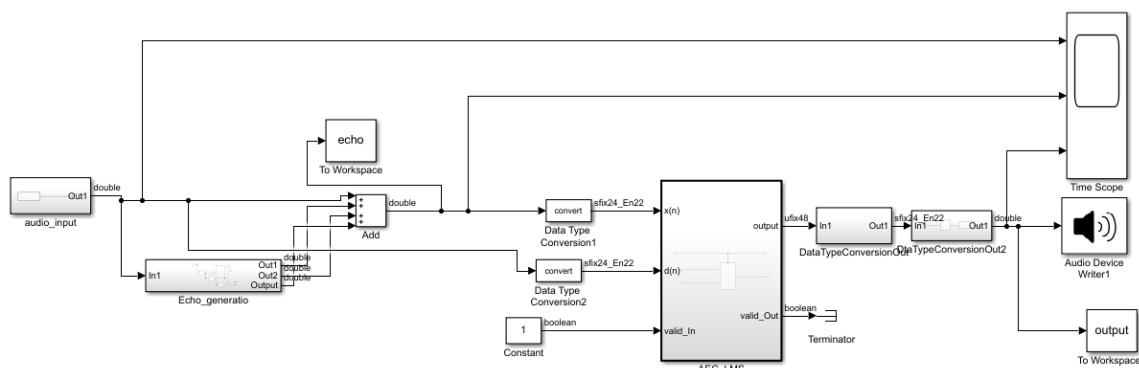


Figure.4 Acoustic Echo Cancellation model for FPGA implementation (Filter length  $M = 32$ , step-size  $(\mu) = 0.013$ )

**B. FPGA co-simulation implementation**

**1. Zynq 7000 SOC evaluation kit overview:**

The overall Simulink model design of echo cancellation is implemented and exported to Xilinx Zynq platform. Xilinx Zynq is a single chip SOC platform that contains two main part Processing system (PS) and a programmable logic (PL). the PL is the equivalent form of FPGA which contains a high-speed interface that will connect to the PS. those interfaces are called advanced extensible interface (AXI) the current version of AXI interface is AXI4.from different types of AXI4 interface we use AXI4-stream interface for this work. PS is the software part or operating system which support implemented by using C programming language to communicate with hardware and it is an ARM cortex A9 chip. So, in Xilinx Zynq platform every system design should be partitioned in to two-part PL side and PS side. The PL part is which is converted in to IP core so we distinguished it as a subsystem of a design model and we

generate Verilog code for implementation. The PL part also a flexible part which can reuse by other designers according to their system requirement and specification but the PS has a fixed architecture.

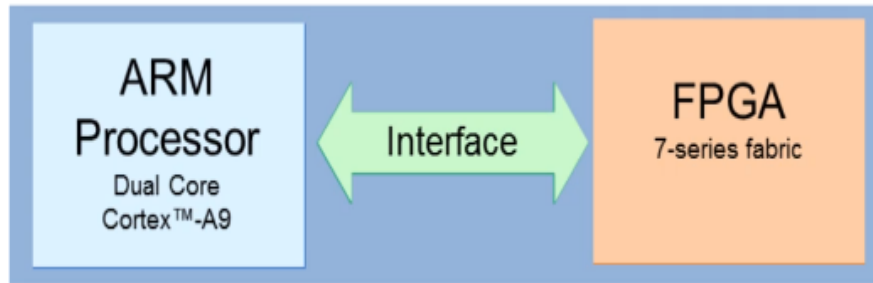


Figure.5. Simplified model of Zynq architecture

For every design project using Xilinx Zynq platform it has a similar design flow starting from system requirement and specification to testing as shown in Figure.7. after deciding in specification and requirement of the design the first thing to do is partitioned the overall design in to software (PS) and hardware (PL) part then defining and use the appropriate interface between PL and PS then if any third party custom reference design is required integrated in to the existing model and create IP from PL and generate VHDL or Verilog code for hardware implementation. For the software side generate a C code and verifying continuously the complete design system after verification the last stage will be building and testing. In present work LMS adaptive filter for echo cancellation subsystem is implemented in the Programmable Logic (PL) and the LMS adaptive filter signal  $x(n)$ ,  $d(n)$  and  $e(n)$  are in software Section which is under processing system (PS) the interface between PL and PS is AXI4-Stream which is integrated with audio codec for audio output in Zed board.

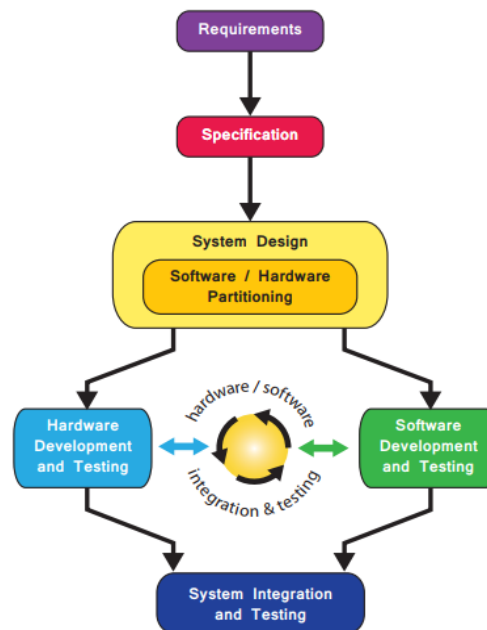


Figure.6. Basic model for design flow of Zynq SoC

## 2. FPGA implementation:

The complete design and implementation work flow is based on a model-based design approach which uses a single environment for the all process of from design specification and requirement to prototype. HDL workflow advisor used the Simulink echo cancellation adaptive filter model to transfer into RTL description which is then converted into IP for hardware implementation. The interface between PS and PL is AXI4-Stream for the purpose of fast and real time data transfer. In order to implement AXI4-stream interface in IP core in our DUT (device under test) we included data in valid signal which is used to control data transmission between AXI4-slave and Master. When the input data is valid the valid signal is asserted. Generally, in this work to implement AXI4-Stream we start by converting the Simulink model of echo cancellation subsystem into an enable subsystem then we add an input control port Valid in

and output control port Valid out finally we use valid into drive both the algorithm subsystem enable port and valid out the input signal  $x(n)$  assigned as data in. The enable subsystem is as shown in Figure.7.

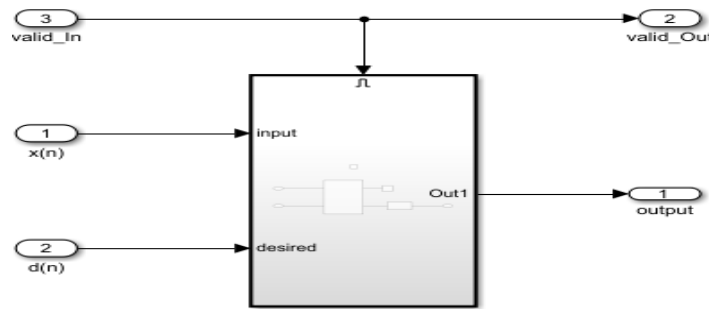


Figure.7. Enabled subsystem for AXI4-stream interface

In order to receive audio input through Zed board line input and process it on FPGA which contain IP core of echo cancellation LMS filter and transmit the echo cancelled audio output through HPN out to speaker we use audio reference design. It uses an audio codec to interface to the peripheral and to convert ADC and vice-versa. The audio codec IP core from the reference design is used to configure the audio codec and for transferring audio data between Zynq board and audio codec. The LMS echo cancellation IP is used for echo cancellation process of the input audio or speech signal. Finally, we integrate the custom reference design with our IP in order to create a single IP core.

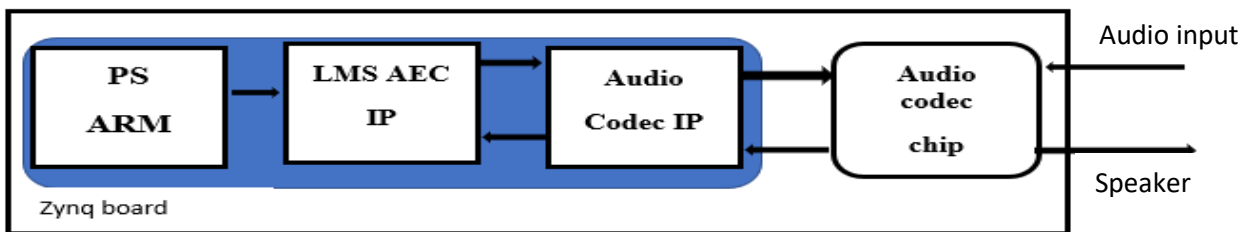


Figure.8. High level architecture of the system on Zed board

Generally, the FPGA hardware co-simulation process involves the following steps.

Step 1: Design adaptive filter with LMS algorithm for acoustic echo cancellation using Simulink DSP toolbox HDL support block as shown in Figure.5.

Step 2: Simulate the design by giving audio input from Matlab workspace and take observation from the output according to the system requirement and specification.

Step 3: Prepare the Simulink echo cancellation model to generate RTL design and create IP core from the subsystem. This will be done by using Xilinx Vivado toolbox as shown in Figure.9.

```
>> hdlsetuptoolpath('ToolName', 'Xilinx Vivado', 'ToolPath', 'C:\Xilinx\Vivado\2018.2\bin\vivado.bat');
Prepending following Xilinx Vivado path(s) to the system path:
C:\Xilinx\Vivado\2018.2\bin
>> addpath(fullfile(matlabroot, 'toolbox', 'hdlcoder', 'hdlcoderdemos', 'customboards', 'ZedBoard'));
>> addpath(fullfile(matlabroot, 'toolbox', 'hdlcoder', 'hdlcoderdemos', 'customboards', 'ipcore'));
```

Figure.9 Matlab command to generate and integrate reference design in Zynq bord

Step 4: Create IP core, generate Verilog code for hardware implementation and target platform interface which is AXI4-Stream using HDL workflow advisor.

Step 5: Generate the software model of echo cancellation design using HDL workflow advisor.

Step 6: Generate the bit stream for implementation of the design in Zynq board using HDL workflow advisor.

Step 7: Prepare the software model and hardware kit for implementation. First connect Zed board with required peripheral device and to host PC and dump the generated bit stream to Zed board then generate the software C code for PS part from the generated software model in step 5 finally run the software model using external mode in Simulink to run on Zed board.

#### IV. EXPERIMENTAL RESULTS

We implemented the FPGA design of the adaptive acoustic echo cancellation system using LMS algorithm by following the steps given under Section 3.

1. This design and implementation require the following software tools those are currently available version of Matlab and Simulink, HDL verifier support package for Xilinx Zynq7000 platform, HDL coder support package for Xilinx Zynq7000 platform and Vivado design suit version 18.2.

2. Simulate the echo cancellation Simulink model using step-size 0.013 and filter length  $M=32$  for LMS adaptive filter block. As expected from the echo generation subsystem we get echo plus original audio signal which is then given to input data in  $x(n)$  port and original is given to the desired port  $d(n)$  after simulation we get the echo cancelled audio output at output port as shown in Figure.10.

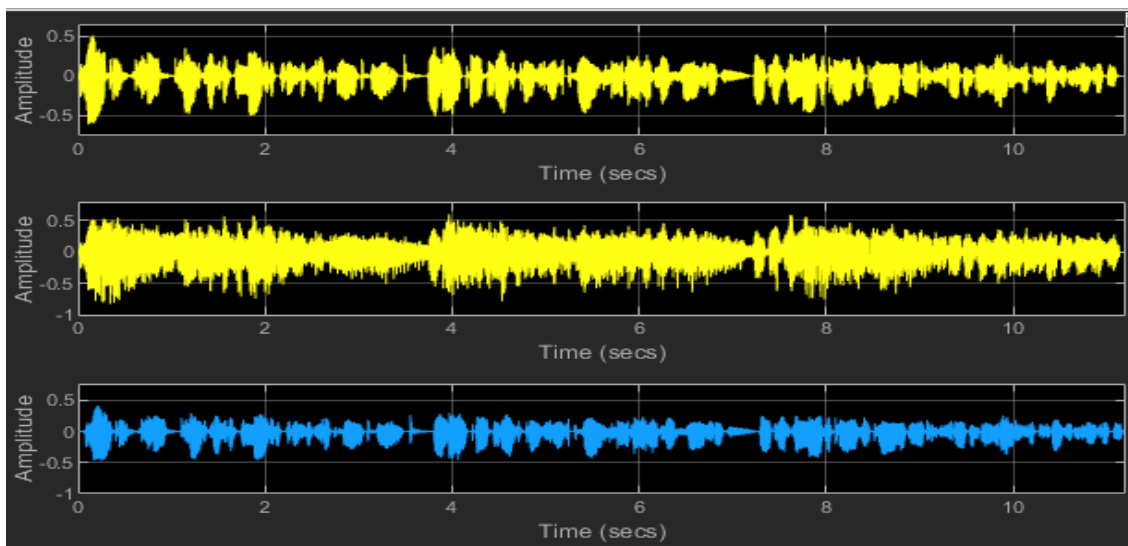


Figure.10. The top signal represents the original input signal (from workspace), the middle signal represents the echo plus original signal the bottom signal is the filtered output signal from LMS system after removing echo, it shows the performance of the LMS algorithm.

3. After the simulation run successfully in Simulink the LMS filter block will be used for RTL by converting it into IP core using HDL coder support in HDL workflow advisor design tasks by specifying the target device for implementation using Xilinx Vivado synthesis tool and selecting reference design in workflow advisor which is audio system with AXI4 stream interface. The most important part is establishing a communication stream or interface between each component and IP block in evaluation Zed board as shown in Figure.12.

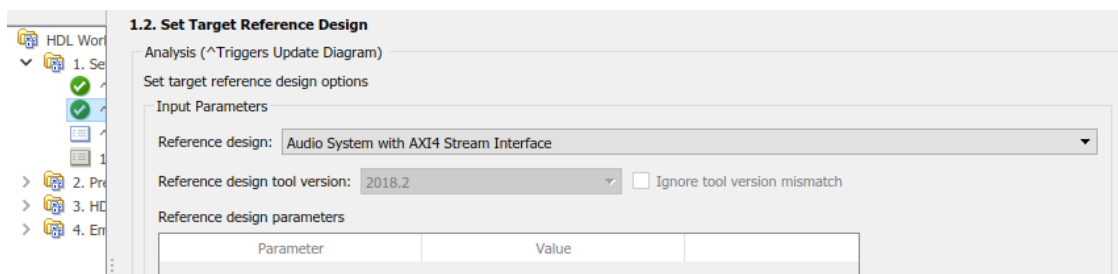


Figure.11 Target reference design for audio codec with AXI4 stream interface



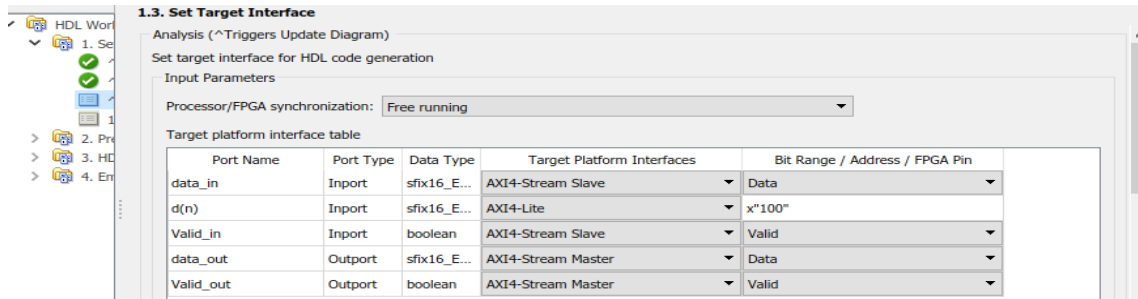


Figure.12 Target interface between PS and PL

4. After setting the target interface by following the design task provided by HDL workflow advisor, we generate Verilog code for hardware implementation in code generation Section and create LMS echo cancellation IP at the same time. Then the software model will be generated which can run on Zed board after bit stream generated and downloaded to Zed board by JTAG cable

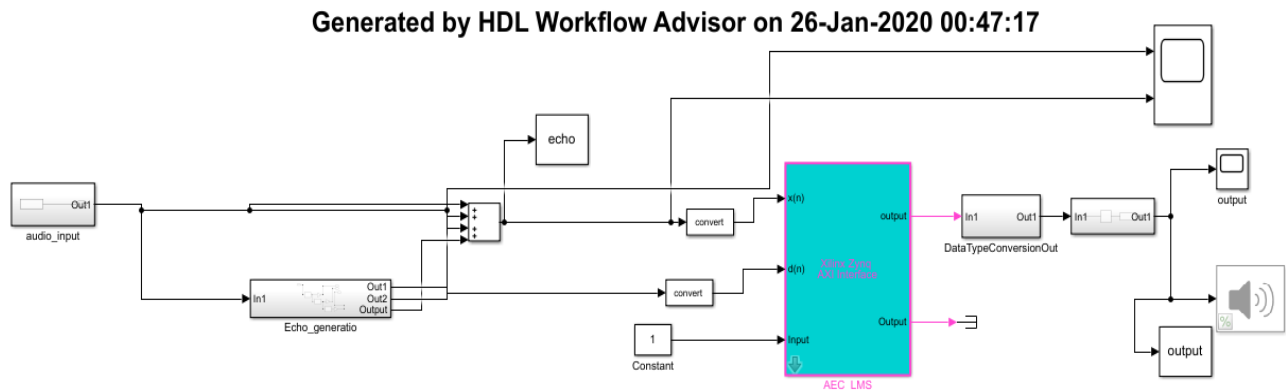


Figure.13. Generated software model of echo cancellation for Zynq evaluation board

5. Finally prepare the generated software model to run on Zed board using external mode and build the program to generate the C code for the software model then run the model by giving audio input from Matlab workspace to Zed board by line in and the output will be heard from the speaker which is echo cancelled audio as we expected.

## V. CONCLUSION

In this paper an adaptive filter with LMS algorithm for echo cancellation application was successfully designed and implemented on Zed board using hardware software co-simulation and using the reference design integration with existing IP core for audio codec has a significant contribution in making possible of getting audio output from Zed board and it solve exporting create IP to Vivado design suite or using another environment for implementation. The experimental result shows that the proposed hardware software co-simulation implementation method has a high performance of echo cancellation.

## REFERENCES:

- [1]. Zhang, Nian, Tam Le and Sasan Haghani. "Design and Implementation of FPGA Based LMS Self-Adjusting Adaptive Filtering System for Audio Signal Processing." (2017).
- [2]. Shilpa, V.J., Jawahar, P.K., & Karthik, S. (2016). Development of Adaptive LMS Filter IP on Zedboard for Hardware-software Co-design.
- [3]. Shaikh, S., & Pujari, S. (2016). Migration from microcontroller to FPGA based SoPC design: Case study: LMS adaptive filter design on Xilinx Zynq FPGA with embedded ARM controller. 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), 129-134.
- [4]. Sung Sue Hwang, Suk Chan Kim and Chae Dong Lee, "Real time acoustic echo cancellation technique based on normalized least mean square method," 2010 2nd International Conference on Computer Engineering and Technology, Chengdu, 2010, pp. V6-12-V6-16.
- [5]. Kumar, Mohan. "Echo Cancellation Algorithms using Adaptive Filters : A Comparative Study." (2014).doi: 10.1109/ICCET.2010.5486265
- [6]. <http://www.zynqbook.com/>
- [7]. <https://in.mathworks.com/help/hdlcoder/examples/basic-hdl-code-generation-with-the-workflow-advisor.html>
- [8]. <https://in.mathworks.com/help/hdlcoder/examples/running-an-audio-filter-on-live-audio-input-using-a-zynq-board.html>



AUTHORS:



Dawit Mamo Alemu did his B.SC in Electronics and Communication in Jimma Institute of Technology, Ethiopia and M.Tech in Systems and Signal Processing in JNTUH India.



Dr. M. Madhavi Latha, Professor in the Department of ECE, JNTUH CEH, guided 25 research scholars in the field of Signal & Image Processing, VLSI and Communications and implementing Deep Learning algorithms on Hardware and published 180 papers in International/National conferences/journals. She is the recipient of "Best Teacher Award" from Govt., 'Exchange Visitor Award' from IEE-Department of States, USA, 'Best paper Award' and 'WIE Appreciation Award' from IEEE. Total of 9 research projects are under her credit.