Logo

Description automatically generatedLogo, company name

Description automatically generated*National University of Sciences and Technology (NUST)*

*College of Electrical and Mechanical Engineering (CEME)*

*DEPARTMENT OF COMPUTER ENGINEERING*

**DE-43 CE A**

**6th Semester**

**EC-313 Digital Signal Processing**

**Lab Mid Report**

**Submitted by:**

**Name:** Abdullah Uppal

**CMS ID:** 381058

**Submitted to**

**Prof Dr. Kiran Khurshid**

**LE Sundas Ashraf**

**Dated**

**March 25, 2024**

**Code:**

close all

[audio, fs] = audioread("LabMidAudio.mp3");

time = input("Enter delay time for echo: ");

alpha = input("Enter value of Alpha for echo:");

echoAud = echo(audio, time, alpha);

sound(audio, fs);

pause(length(audio)/fs + 1);

disp("Original Done")

sound(echoAud, fs);

% Plot the Audio and Echo

figure;

subplot(3,1,1);

plot(audio);

title('Original Audio');

subplot(3,1,2);

plot(echoAud);

title('Audio with Echo');

% Reverberation

timere = input("Enter delay time for reverberation: ");

alphare = input("Enter value of Alpha for reverberation: ");

reverbAud = reverb(audio, timere, alphare);

sound(reverbAud, fs);

subplot(3,1,3);

plot(reverbAud);

title('Audio Reverberation Effect');

%%

% Chorus

delaychor = input("Enter delay time for chorus: ");

delay\_mod = round(0.005\*fs);

voicesNum = input("Enter Number of voices: ");

chorAudio = chorus(audio, delaychor, voicesNum, delay\_mod, fs);

disp("Playing Chorus voice")

sound(chorAudio, fs);

figure;

plot(chorAudio);

title('Audio with Chorus');

function echoAudio = echo(audio, delay, alpha)

echoAudio = zeros(size(audio));

for n = delay+1:length(audio)

echoAudio(n) = audio(n) + alpha \* audio(n-delay);

end

end

function reverbAudio = reverb(audio, delay, alpha)

reverbAudio = zeros(size(audio));

for n = delay+1:length(audio)

reverbAudio(n) = audio(n) + alpha \* reverbAudio(n-delay);

end

end

function chorAudio = chorus(audio, delay, voiceNum, delay\_mod, fs)

chorAudio = audio;

mod\_freq = 2;

mod\_period = fs / mod\_freq;

for voice = 1:voiceNum

% Generate the modulated delay amount

mod\_delay = delay \* ones(size(audio));

for n = 1:length(audio)

mod\_index = mod(n, mod\_period) + 1;

if mod\_index <= mod\_period/2

mod\_delay(n) = delay + delay\_mod;

else

mod\_delay(n) = delay - delay\_mod;

end

end

% Apply delay and modulation to the current voice

audio\_voice = zeros(size(audio));

for n = max(mod\_delay)+1:length(audio)

delay\_index = n - mod\_delay(n);

if delay\_index > 0 && delay\_index <= length(audio)

audio\_voice(n) = audio(delay\_index);

end

end

% Mix the current voice with the chorus signal

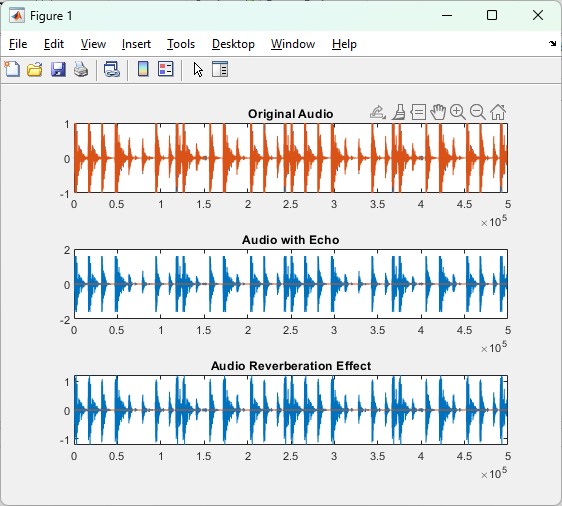
chorAudio = chorAudio + audio\_voice / voiceNum;

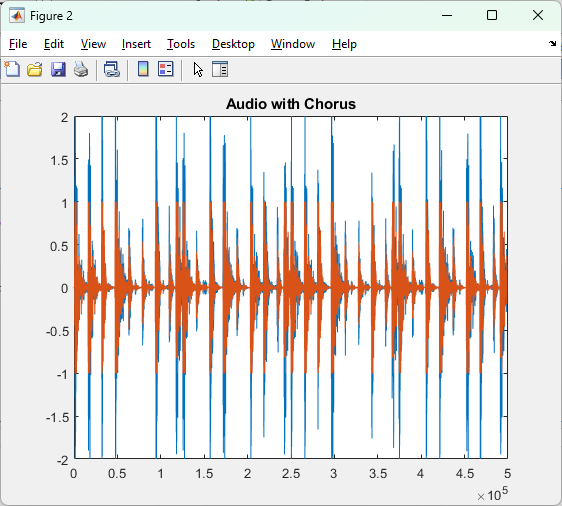
end

end

**Outputs:**

**To check the output in better way give the inputs that are better for checking as we didn’t know about those value.**

****

****