**Experiment 10: Multi-rate signal processing**

|  |  |
| --- | --- |
| **Name** | Prathamesh Mane |
| **UID no. & Branch** | **2022200078 (B1)** |
| **Experiment No.** | 10 |

Code :

clear;

clc;

% Step 1: Load the audio file

[input\_signal, input\_Fs] = audioread('prathamesh\_rec.wav'); % Replace with actual file name

Fs = 44100; % Required sampling rate

% Convert to mono if the input signal is stereo

if size(input\_signal, 2) > 1

input\_signal = mean(input\_signal, 2); % Average the two channels

disp('Input signal converted to mono.');

end

% Check if the input sampling rate matches the required rate

if input\_Fs ~= Fs

input\_signal = resample(input\_signal, Fs, input\_Fs); % Resample to 44,100 Hz

end

% Step 2: Define the target sampling rate

Fs\_target = 48000; % Target sampling frequency in Hz

% Step 3: Determine up-sampling (L) and down-sampling (M) factors

[L, M] = rat(Fs\_target / Fs); % Rational fraction of the conversion ratio

fprintf('Up-sampling factor (L): %d\n', L);

fprintf('Down-sampling factor (M): %d\n', M);

% Step 4: Resample the signal

% First, upsample by L

upsampled\_signal = upsample(input\_signal, L);

% Design a low-pass filter to prevent aliasing

Fcutoff = min(Fs, Fs\_target) / 2; % Cutoff frequency for anti-aliasing

h = fir1(128, Fcutoff / (L \* Fs)); % FIR filter design

% Convolve the upsampled signal with the filter

filtered\_signal = filter(h, 1, upsampled\_signal); % Use 'filter' instead of 'conv'

% Then, downsample by M

output\_signal = downsample(filtered\_signal, M);

% Step 5: Play and save the signals

disp('Playing the original signal...');

sound(input\_signal, Fs);

pause(length(input\_signal) / Fs + 1);

disp('Playing the converted signal...');

sound(output\_signal, Fs\_target);

pause(length(output\_signal) / Fs\_target + 1);

% Step 6: Visualization

t\_input = (0:length(input\_signal)-1) / Fs;

t\_output = (0:length(output\_signal)-1) / Fs\_target;

subplot(2, 1, 1);

plot(t\_input, input\_signal);

title('Original Signal');

xlabel('Time (s)');

ylabel('Amplitude');

grid on;

subplot(2, 1, 2);

plot(t\_output, output\_signal);

title('Resampled Signal');

xlabel('Time (s)');

ylabel('Amplitude');

grid on;