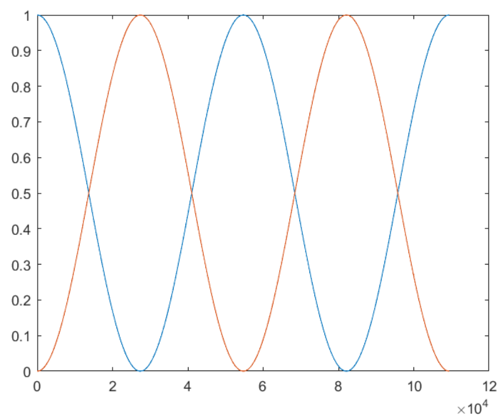
Alonso Herreros Copete
February 28th, 2023
Systems and Circuits (15369)
Lab 1 report

Signals Lab: Report

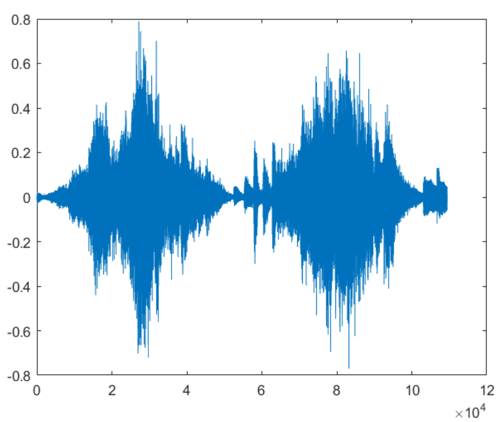
Project 1: mixing music with fading

In this project, we make a song fade into a different one. This effect is achieved by adding together two signals with intermittent fading. While the first one fades out, the second one fades in, and vice versa. This process is repeated 4 times, going back and forth twice.

First, the longest song is cut to match the length of the shortest song. Then, we apply fading by multiplying each sample by an attenuation factor, which varies over time. For the first song, it starts at 1 (full volume) and goes to 0 (silent). For the second song, it starts at 0 (silent) and goes to 1 (full volume). This is repeated a total of 4 times. In the following graph, the blue line represents the attenuation factor for the first song, while the orange line is for the second song.



After changing the volume of each track, the final audio is created by adding the values one by one. This is a plot of the final song.

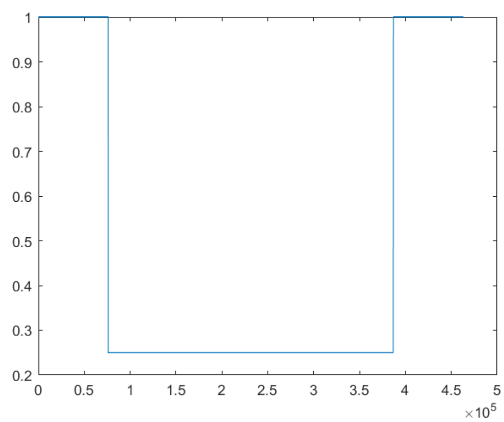


After this, I also calculated the total energy of the final signal, which is 1131, and its average power, which is 0.0103.

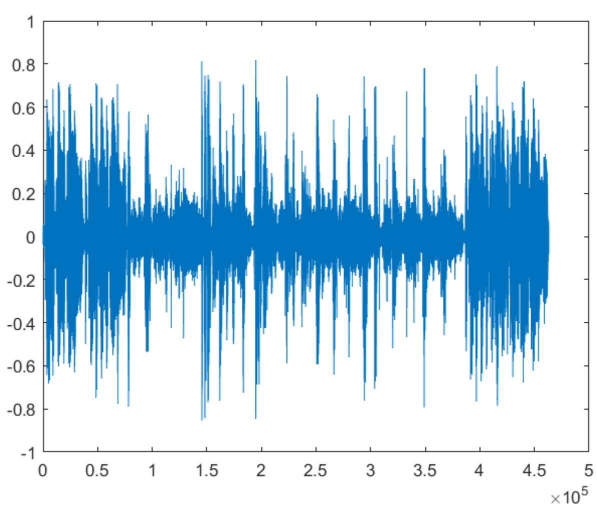
Project 2: Add background music

Here, we added a background song to a track of a speech by applying attenuation to the song during the speech and then adding both signals together.

First, the music track is extended to three times its original length by copying it three times and saving it to another variable, in order to have enough audio for the whole duration of the voice track. Since we need the voice track to be centered, the global midpoint is calculated, and we also calculate the half of the voice track's length so we can offset the voice from the midpoint. Using these values, we create the following attenuation signal, which is 25% volume in the middle.



After applying attenuation, a new track is created with the final length, and the speech is placed in the middle using the previously calculated values, and it is added sample by sample to the music track.



After this, I also calculated the total energy of the final signal, which is 7424, and its average power, which is 0.016.

Code for project 1:

```
% Project 1: mixing music with fading
TheSting = audioread("audio\thesting.wav");
StarWars = audioread("audio\starwars.wav");

% Cut both to the same length
minLength = min(length(TheSting),length(StarWars));
TheSting = TheSting(1:minLength);
StarWars = StarWars(1:minLength);

n = (0:minLength-1)'; % Using transposition to make it vertical
M = length(n)/2; % By setting this to 2 there will be 2 cycles of back and forth fading (4 transitions)

%% Apply fading
s1 = (0.5 + 0.5*cos(2*pi*n/M));
TheSting_faded = TheSting .* s1;

s2 = (0.5 + 0.5*cos(2*pi*n/M + pi));
StarWars_faded = StarWars .* s2;

plot(n, s1, n, s2); %For testing purposes

x = TheSting_faded + StarWars_faded; %% Mix the two songs

playaudio(x); % Play the mixed song

% Plot the mixed song
figure;
%stem(n, x);
plot(n, x); % Plot looks better visually

energy = sum(x.^2) % Calculate the energy of the mixed song
power = mean(x.^2) % Calculate the power of the mixed song. It's equal to energy/length(x)

audiowrite("audio\project1.wav", x, 11025); % Write the mixed song to a file
```

Code for project 2:

```
% Project 2: Add background music
bladerunner = audioread("audio\bladerunner.wav");
thirdman = audioread("audio\thirdman.wav");

tm = repmat(thirdman, 3, 1); % Repeat thirdman 3 times

midpoint = floor(length(tm)/2); % Find the midpoint of the new audio file
half_br_length = floor(length(bladerunner)/2); % Find half the length of bladerunner (offset from midpoint)

attenuation = ones(length(tm), 1);
% Set the attenuation to 0.25 for the bladerunner section
attenuation(midpoint-half_br_length:midpoint+half_br_length) = 0.25;
plot(attenuation);

%plot(1:length(tm), attenuation); %Testing purposes

% Voice alignment
voice = zeros(length(tm), 1);
voice(midpoint-half_br_length:midpoint+half_br_length-1) = bladerunner(1:2*half_br_length);
% 1 sample was cut due to having an odd number of samples and not being able to split it evenly

finalAudio = tm.*attenuation + voice; % Add the two together

playaudio(finalAudio);

% Plot the mixed song
figure;
%stem(0:length(finalAudio)-1, finalAudio);
plot(0:length(finalAudio)-1, finalAudio); %Plot looks better visually

energy = sum(finalAudio.^2) % Calculate the energy of the mixed song
power = mean(finalAudio.^2) % Calculate the power of the mixed song. It's equal to energy/length(x)

audiowrite("audio\project2.wav", finalAudio, 11025); % Write the mixed song to a file

```