**Digital Signal Processing Lab**

**EEL-325**

Lab Journal: 02



* + - 1. Name: M.OMAR
      2. Class: BCE-6
      3. Enrollment No: 01-132182-024

**Lab # 02**

**Speech Processing Using MATLAB**

**Objective:**

* In order to get voice signal in MATLAB use built-in function for recording voice signal.
* Add and subtract both voice signals with duration of your choice.
* Flip and convolve (Correlate) the second signal with the first signal.

**Introduction:**

Audio Recorder

To record audio and generate a vector with the audio samples, we need to go through 3 steps:

1. Create an audiorecorder object;
2. Record the audio;
3. Extract the data from the audiorecorder object.

These three steps are described below. Later, we describe how to reproduce the sound from a processed vector.

Creating an audiorecorder object

In most PCs running Windows, it is necessary to identify the DeviceID of the computer's soundcard before creating the audiorecorder. We may get this information from the audiodevinfo structure. This structure usually has two fields: Input and Output. To access the Input field, which identifies the soundcard that will be used to record the audio, we may use the following command:

getfield(audiodevinfo, 'Input')

A typical output for the command above is:

ans =

Name: 'SoundMAX Digital Audio'

DriverVersion: '5.10'

ID: 0

The audiorecorder function, which is used to create the audiorecorder object, may take different sets of parameters . We are going to use the following command:

arec = audiorecorder(Fs, nbits, channels, id)

* Fs: corresponds to the sampling frequency (n Hz) that should be used. We need to choose one of the standard values: 8000, 11025, 22050, or 44100. Remember that, from the sampling theorem the sampling frequency corresponds to twice the maximum frequency of the signal. We may choose 44100 Hz, if we want to have a wide frequency range to work on.
* nbits: corresponds to the number of bits used to represent each sample. The standard values are 8, 16, 24, 32 bits (the last two are only available 24-bit and 32-bit sound devices). A reasonable choice is 16 bits, which is used in audio cds, for instance. Choosing 16 bits mean that the signal samples will have interger values in the range from (- 2^15) up to (2^15 - 1).
* channels: Number of channels that will be recorded. Possible values are 1 (for mono) and 2 (for stereo). If we are recording using only one microphone, we just need 1 channel.

**Procedure:**

* Open MATLAB Software.
* We create the new project.
* Make a new script and name it on the name of your lab.
* In order to get voice signal in MATLAB use built-in function **“audiorecorder”** for recording Voice signal.
* To read and write the voice signal use **“audiowrite”.**
* To get voice signal as a vector, we use **“getaudiodata”** function.- Record the voice signal for sometimes. I.e. t=10sec- Inter sound and plot command.
* Run the mail file and observe the results.

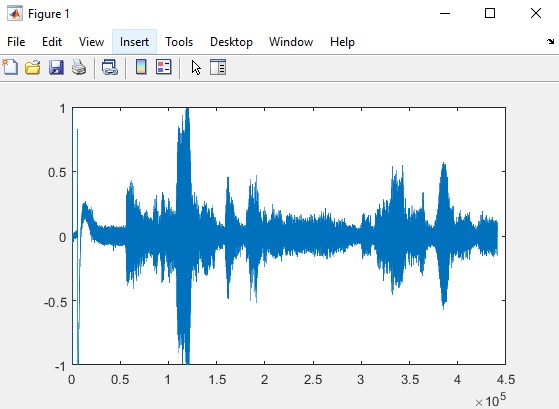
**Problem No. 01:**

**Task 1: In order to get voice signal in MATLAB use built-in function for recording voice signal for 10 seconds.**

MATLAB Code:

recordVoice = audiorecorder(44100,8,1); disp('Recording Started, please speak') recordblocking(recordVoice,10); disp('Recording Ended'); play(recordVoice); recording = getaudiodata(recordVoice); subplot(2,1,1); plot(recording);

Output:



**Task 2: Add and subtract both voice signals with duration of your choice.**

MATLAB Code:

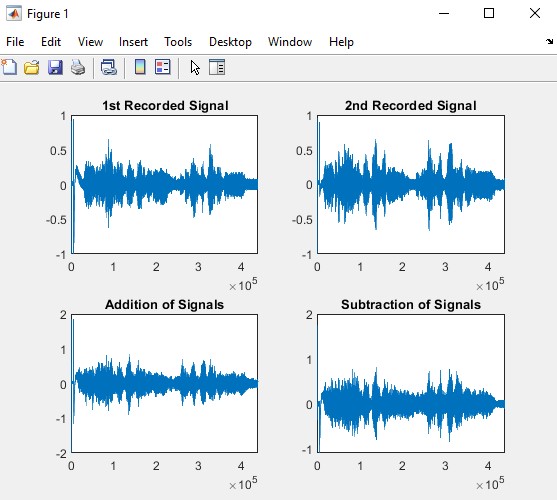
recordVoice = audiorecorder(44100,8,1); disp('Recording Started, please speak') recordblocking(recordVoice,10); disp('Recording Ended'); play(recordVoice); recording = getaudiodata(recordVoice); subplot(2,2,1); plot(recording); title('1st Recorded Signal'); recordVoicetwo = audiorecorder(44100,8,1); disp('Recording Started, please speak') recordblocking(recordVoicetwo,10); disp('Recording Ended'); play(recordVoicetwo); recordingtwo = getaudiodata(recordVoicetwo); subplot(2,2,2); plot(recordingtwo); title('2nd Recorded Signal');

**AddSignal = recording + recordingtwo;** subplot(2,2,3); plot(AddSignal);

title('Addition of Signals');

**SubtractSignal = recording - recordingtwo;** subplot(2,2,4); plot(SubtractSignal); title('Subtraction of Signals');

Output:

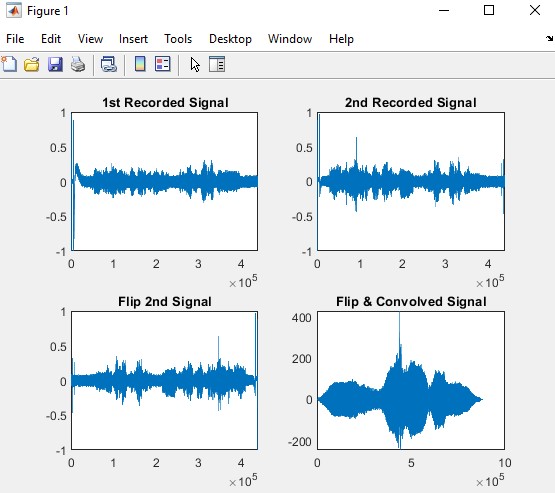


**Task 3: Flip and convolve (Correlate) the second signal with the first signal.**

MATLAB Code:

recordVoice = audiorecorder(44100,8,1); disp('Recording Started, please speak') recordblocking(recordVoice,10); disp('Recording Ended'); play(recordVoice); recording = getaudiodata(recordVoice); subplot(2,2,1); plot(recording);

title('1st Recorded Signal'); recordVoicetwo = audiorecorder(44100,8,1); disp('Recording Started, please speak') recordblocking(recordVoicetwo,10); disp('Recording Ended'); play(recordVoicetwo); recordingtwo = getaudiodata(recordVoicetwo); subplot(2,2,2); plot(recordingtwo); title('2nd Recorded Signal'); **FlipSignal = flip(recordingtwo);** subplot(2,2,3); plot(FlipSignal); title('Flip 2nd Signal'); x=conv(recording,FlipSignal); subplot(2,2,4); plot(x); title('Flip & Convolved Signal');



**Conclusion:**

In this lab we LEARNED about voice signals, performing arithmetic operations on voice signals and executing flip and conv commands in Matlab.