MATLAB Audio Libraries

MATLAB provides a library with several easy-to-use functions to both record and play audio. I've played with the various functions that we will likely need to use during the implementation of our acoustic modem. I've summarized the main functions we will be using next semester, however more detail can be found on the MATLAB library website. I've also posted the MATLAB script file I used to learn how to use these functions on GitHub.

Recording Audio

Recording audio centers around an "audiorecorder" object. To start recording any audio, simply instantiate one of these objects.

```
%% Create an audiorecorder object w/ default parameters.
recObj = audiorecorder;
```

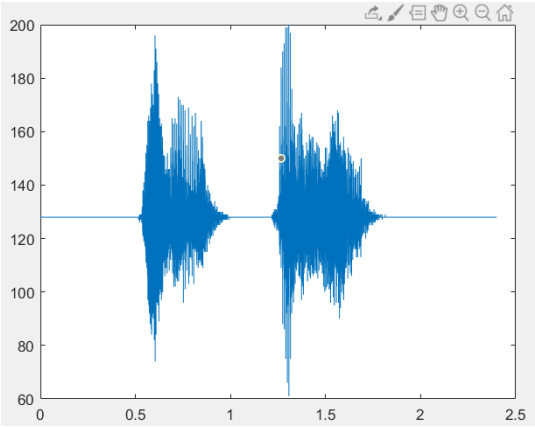
Once the object is instantiated, there are a few functions you can have it do. For our purposes, the most important function is the "record" function. "recordblocking" also does the exact same thing as "record", but also blocks the program from going on until it is done its recording period. To use the record function, simply pass an audiorecorder object and a time in seconds for which you want to record.

```
%% Record Audio
disp('Start Speaking.')
recordblocking(recObj, 1);          %Start a recording block of 1 seconds long, holds control until done
disp('End of Recording.');
```

From here, you can play back the audio recorded and saved in the audiorecorder object using the play() function with a parameter of the audiorecorder object. You can also use the getaudiodata function to convert the audio data to a vector of doubles. From that vector, you can convert to any other desired data type, such as binary.

```
%% Convert to data array
y = getaudiodata(recObj);           %Double as default
x = getaudiodata(recObj, 'uint8');
b = de2bi(x);                       %Convert to binary
b = reshape(b,1,[]);
```

Here's what it looks like when I plot the double vector resulting from me saying "Hello, how are you?"



For our purposes, we will use the audiorecorder object and associated functions on the receiver side if our acoustic modem.

Playing Audio

Playing a signal as audio in MATLAB is very straightforward. The "sound" function will produce a sound from an inputted vector. The inputted vector has to have values in the range of -1.0 to 1.0. It is also optional to pass in the sampling frequency, otherwise it defaults to 8192 Hz.

In the example below, I define a sampling frequency and center frequency as per our project specifications. I then create a sinewave based on these specifications, replicate the signal so it lasts longer than 1 second, and pass it to the sound function.

```
%% Outputting a sine wave as sound with freq of 9kHz and Sampling 44.1kHz
Fs = 44100;                             % Sampling Frequency of 44.1kHz
t  = linspace(0, 1, Fs);                % One Second Time Vector ->LINSPACE(X1, X2, N) generates N points between X1 and X2.%
w = 2*pi*9000;                          % Radian Value To Create 9kHz Tone
s = sin(w*t);                           % Create Sinewave
s_multisecond = repmat(s,1,2);          % Create a 2 second tone instead
sound(s_multisecond, Fs)                % Produce Tone As Sound, sample rate Fs must be >= 2*Tone Frequency to play accurately
```

For our purposes, this is what we will use for our transmitter. We will define a time vector with specified sampling frequency and upconvert it to a carrier wave with desired center frequency. We will then pass in the audio signal to the sound function and ultimately through the speakers.