Demo 7: Microphone input
Exercises

DSP Lab (EE 4163 / EL 6183)

Fall 2017

1   Demo files

```
filter_play_mic.py
filter_play_mic_FPB.py
myfunctions.py
author.wav
device_info.py
```

The demo program `filter_play_mic.py` takes audio input from the microphone, filters it using a bandpass filter, and plays the output audio signal to the speakers or headphones (the default output audio device as set in the operating system). When reading the input signal from the microphone it is recommended that headphones be used to avoid feedback problems (when the output audio passes from the speaker back into the microphone).

2   Exercises

1. Modify the demo program so it plays a stereo signal. In one channel, play the input audio signal. In the other channel, play the output of the bandpass filter. The two signals should play simultaneously.

2. The demo program `filter_play_mic_FPB.py` specifies the optional PyAudio stream parameter called `frames_per_buffer` as in the following code fragment. Run the program with FPB set to each of the values: 1024, 256, 64. What is the effect of this parameter? Comment on your observation.

```
stream = p.open(format = p.get_format_from_width(WIDTH),
                channels = CHANNELS,
                rate = RATE,
                input = True,
                output = True,
                frames_per_buffer = FPB)
```

3. Modify the demo program `filter_play_mic.py` to write the input signal to a WAV file instead of to the speaker.

4. Modify the demo program `filter_play_mic.py` to process the input signal $x(t)$. The output   SUBMIT
   signal should be

$$y(t) = x(t)\cos(2\pi f_0 t) \tag{1}$$

   there $f_0 = 400$ Hz. The output signal $y(t)$ should both be played to the speaker and save to a
   WAV file. This is amplitude modulation. What is the effect of this on the voice signal? Submit
   your WAV file of yourself talking, as well as your code.