Demo 7 Exercises: Microphone input

DSP Lab (ECE 4163 / ECE 6183)

Fall 2018

Demo files

`mic_filter.py`
`mic_filter_FPB.py`
`myfunctions.py`
`author.wav`
`device_info.py`

The demo program `mic_filter.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 you read the input signal from the microphone, it is recommended that you use headphones to avoid feedback problems (wherein the output audio passes from the speaker back into the microphone).

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 `mic_filter_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 `mic_filter.py` to write the output signal to a wave (wav) file instead of to the speaker.

4. Modify the demo program `mic_filter.py` to process the input signal $x(t)$ so that the output SUBMIT

signal is

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

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