Python and Scientific Python Lab

Experiment - 6 Sound Generated By Formula

I Pre-Lab Questions

1. Why the direct Sin wave generated samples can be played without 16 bit data conversion?

Sol: These direct Sin wave generated samples can contain some distortions in them. So without data conversion we may lose some original information at the o/p. We do a 16-bit data conversion so that we get an optimum output. This data conversion is actually done by ADC. The digital o/p from the ADC is helpful in getting the original signal back.

2. Give the Calculation for generating no of samples if one want to do to play audio and Sampling freq is 5000 samples/sec.

Sol:
```
fs = 5000
Seconds = 10 sec
t = np.linspace(0, seconds, seconds * fs, False)
note = np.sin(freq * t * 2 * np.pi)
audio = note * (2**15 - 1) / np.max(np.abs(note))
```

II Post Lab Questions

1. Give the command for starting the playback and explain the syntax and usage

Sol: There are diff commands to start the playback based on the modules.

i) playsound module:

```
playsound('audio.mp3')
```

ii) pydub module:

```
song = AudioSegment.from_wav('sound.wav')
play(song)
```

iii) Native Player

```
import os
file = "file.mp3"
os.system("mpg123 " + file)
```

2. Write the mathematics of piano note frequency generation.

Soln. Pianos are tuned using "equal temperament system" so, the relation between freq of keys is like

note_freq = base_freq * $2^{n/12}$.

In the code, we write it like this;

```
note_freq = {octave[i]: base_freq * pow(2,(i/12))
             for i in range(len(octave))}
```