05_MusicalNoteGeneration

January 10, 2020

1 Generate Musical Notes

- 8Ohm, 2W toy speaker.

[1]:
```
def get_freq(n:int =-21) -> float:
    """ Generate a frequency from an 'n'.

    Based on an equation:
      https://www.intmath.com/trigonometric-graphs/music.php
    """
    return 440.*2.**(n/12.0)

notes = ["C", "C#", "D", "D#", "E", "F", "F#", "G", "G#", "A", "A#", "B"]
def get_note_frequency(note: str, octave: int=0):
    """Get the frequency of a note:

    note: Musical note.
    octave: Setup so octave=0, A = 440 Hz.
    """
    n = notes.index(note)-9.0-octave*12.0
    return get_freq(n)
```

[2]:
```
# A
get_freq(0)
```

[2]:
```
440.0
```

[3]:
```
import mhs5200
inst = mhs5200.MHS5200(port="/dev/ttyUSB0")
```

[4]:
```
get_note_frequency(note="A", octave=0)
```

[4]:
```
440.0
```

1.1 Run Musical Scale

Since A=0, go from -9 to 3 (C to B)

[5]:

```python
import time
```

[6]:

```python
inst.on()
for n in range(-9, 3):
    for chan in inst.channels:
        chan.amplitude=20
        chan.frequency=get_freq(n)
        chan.offset=0
        chan.wave=0
    time.sleep(0.5)
inst.off()
```

[7]:

```python
inst.on()
get_note_frequency("C")
```

[7]:

```
261.6255653005986
```

[8]:

```python
inst.off()
```

[9]:

```python
for note in notes:
    chan = inst.channels[1]
    chan.amplitude=5
    print(f"note: {note}")
    chan.frequency=get_note_frequency(note)
    chan.offset=0
    chan.wave=0
    time.sleep(0.5)
```

```
note: C
note: C#
note: D
note: D#
note: E
note: F
note: F#
note: G
note: G#
note: A
note: A#
note: B
```

2 Twinkle Twinkle Little Star

[10]:

```python
song = "CCGGAAGFFEEDDCGGFFEEDGGFFEEDCCGGAAGFFEEDDC"
inst.on()
for note in song:
    chan = inst.channels[1]
    chan.amplitude=0
    time.sleep(0.0)
    print(f"note: {note}")
    chan.frequency=get_note_frequency(note, -1)
    chan.amplitude=1
    chan.offset=0
    chan.wave=0
    time.sleep(0.1)
inst.off()
```

```
note: C
note: C
note: G
note: G
note: A
note: A
note: G
note: F
note: F
note: E
note: E
note: D
note: D
note: C
note: G
note: G
note: F
note: F
note: E
note: E
note: D
note: G
note: G
note: F
note: F
note: E
note: E
note: D
note: C
note: C
note: G
note: G
note: A
note: A
note: G
note: F
note: F
```

```
note: E
note: E
note: D
note: D
note: C
```

[11]:

```
import re
song_re = re.compile("([A-G]?#?)")
```

[12]:

```
def play_song(inst, song):
    inst.on()
    for note in song:
        chan = inst.channels[1]
        chan.amplitude=0
        # Between beats
        time.sleep(0.1)
        chan.frequency=get_note_frequency(note, -1)
        chan.amplitude=1.0
        chan.offset=0
        chan.wave=0
        time.sleep(0.2)
    inst.off()
```

[13]:

```
twinkle = "CCGGAAGFFEEDDCGGFFEEDGGFFEEDCCGGAAGFFEEDDC"
baby_shark = "CDFFFFFFFFCDFFFFFFFFCDFFFFFFFFFFE"
```

[14]:

```
for song in [twinkle, baby_shark]:
    play_song(inst=inst, song=song)
```