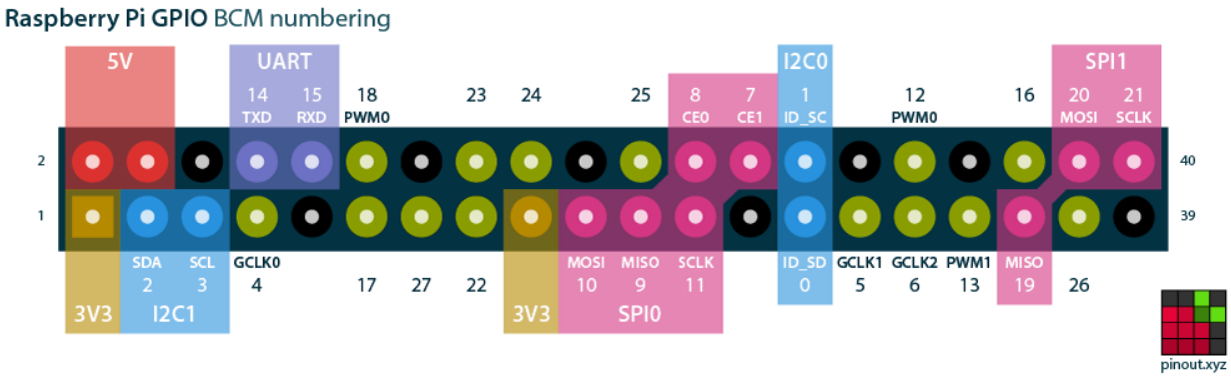
Digital Keyboard Synthesizer

Nathaniel Lawson PY371 Electronics Lab

Abstract:

This project will create a synthesizer. The keyboard registers using touch pads when a key is being pressed and sends a signal in response to the Raspberry Pi. The Raspberry Pi then takes a program that outputs the frequency note associated with the pressed key and sends that note to the speaker. The speaker then outputs the combined waveform of all the notes sent to it.

The pinout map can be found here: <https://pinout.xyz/pinout/piano_hat>. A visual representation of the Raspberry Pi pinout is below. The speaker will simply go into one of the available empty slots.



PARTS LIST:

--1 Piano HAT

--1 Raspberry Pi Circuit Board

--1 Speaker

The keyboard covers one octave C_4, with thirteen keys from 261.626Hz to 523.251Hz. However, the keyboard can shift octaves. This keyboard is attached to the Raspberry Pi, which is essentially a big Arduino. The Raspberry Pi takes Python code rather than C, but is otherwise not particularly different.

The piano key frequencies are in the form $f(n) = 440\ (Hz) * 2^{(n-49)/12}$, where n is the key number from 1 to 88. They key numbers for the chosen octave are 40-52 A full list of frequencies can be found here: <https://en.wikipedia.org/wiki/Piano_key_frequencies>. The notes will initially be a basic sine wave with the given frequency. However, I plan to attempt to add overtones at the harmonics to make the sound more pleasing. I also may fiddle with the shape of the waveform.

The amplitude of the waveforms will depend upon the sensitivity of the speaker. A speaker with a Sensitivity Rating of 87 produces a volume of 87 dB at one meter given 1 watt of power plus 3dB for each extra watt of power applied. The output volume should be greater than 85 dB to obtain a reasonable sound level. Therefore, given an 87 Sensitivity Rating, the amplitude of the waveform should be either sqrt(10) volts (87 dB) or sqrt(20) volts (90 dB).