Physics

Outline

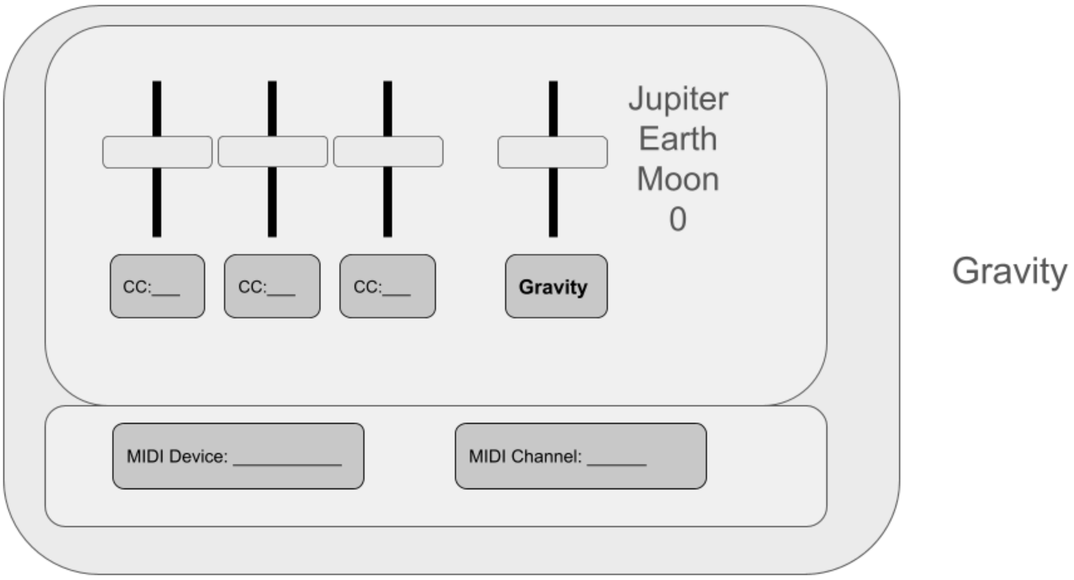
In this project, we're going to build three python 'modules' that will control synthesisers via MIDI continuous controllers (CCs). The modules will be inspired by Physics. Module 1 is called Gravity. Module 2 is called Particle. Module 3 is called Pendulum.
All three modules will share a basic boilerplate, where a user can assign a MIDI device and MIDI channel number (1 to 16).

Technical Stack

The project will use Python, and PyGame, since this combination has worked well in previous projects, and the libraries are available. We'll create a single environment for the development, and create separate python files for each 'module'. The Pymunk library and or pybox2d library could be good for the physics simulations.

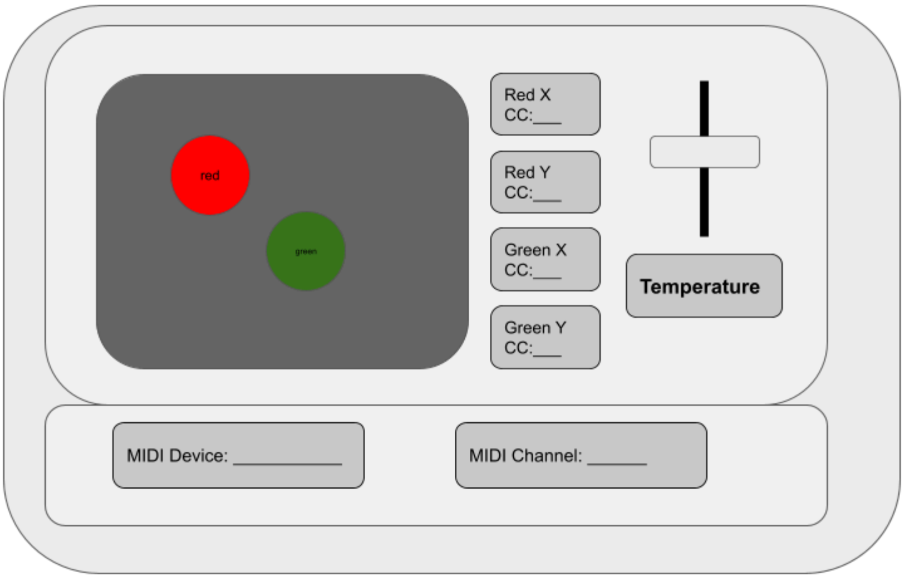
Module 1 - Gravity

Gravity consists of three MIDI CC controllers. The user can assign CC values to each controller. There is a fourth control called 'gravity'. The gravity control determines how quickly the MIDI CC controllers return to zero. When the gravity control is at zero, this represents zero gravity, and the CC controllers don't move once set. As a user increases the gravity, the MIDI CC controllers return to zero. At maximum gravity, the MIDI CC controllers return to zero very quickly. It could be fun to label the maximum gravity 'Jupiter', since this is the largest planet in our solar system. The mid-way point could be labelled 'earth' and simulate the gravity on this planet.



Module 2 - Particle

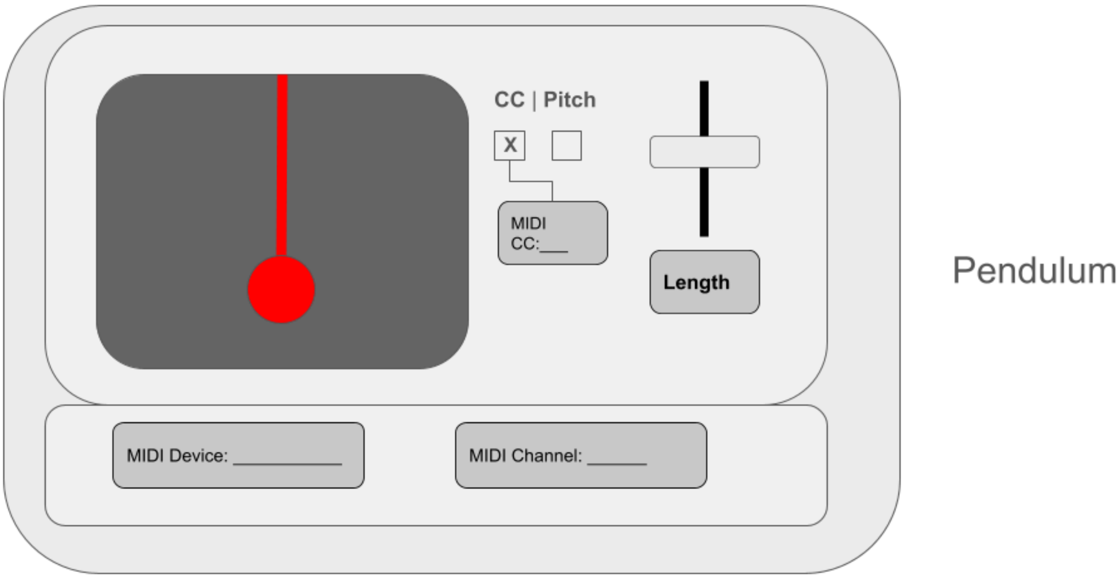
This is a simple particle simulation that sends MIDI CC values. The particle simulation has two particles (red and green). A user can assign MIDI CC values to the X and Y axis for both the Red and Green particles. As the particles bounce around a grid, their position is translated to MIDI CC values that get sent to the MIDI device. Collisions can occur between the boundary of the grid, and the other particle. Collisions cause the particles to change direction. We could even add a temperature control that increases the velocity of the particles (i.e. more collisions). When the program runs, the X and Y coordinates of the Red and Green particles should be set to random positions on the grid and start moving in a random direction.



Particle

Module 3 - Pendulum

This is a simulation of a pendulum. A user can select either pitch shift or MIDI CC control. In pitch shift mode, the position of the pendulum on an XY grid is translated to pitch shift data, similar to a pitch shift wheel on a synth. In MIDI CC mode, the position of the pendulum will be translated to MIDI CC data, centered around the value 64. There is a controller where a user can set the pendulum length. Longer length equals slower swing time. When the program runs, the pendulum starts swinging.



Plan

1. Create the boilerplate and check the MIDI connection
2. Create the Gravity module and conduct some initial tests
3. Create the Particle module and conduct some initial tests
4. Create the Pendulum module and conduct some initial tests
5. Identify and fix bugs
6. Review the UI design and improve/iterate as needed
7. Set-up the Github repo
8. Write some content for the Github wiki pages. This could be background information on how plant sizes affect gravity, how particle simulations are used in research, and how simple pendulums were the basis of our time keeping for many years,
9. Write a blog about the project
10. Record a YouTube video to showcase the project.