Martí Serra
zcabmse@ucl.ac.uk

COMP101P: Reading Week Coursework Documentation

I. Engduino Application

In order to develop the Engduino application I have used the Arduino and Processing IDEs. I've employed the former to write a simple program "Sensor Print" in C that prints to the serial port each of the sensors' value. Then, thanks to the Processing default library "serial", I can easily read the printed values and use them as needed. In this case, I have created two separate applications which make use of them in different ways.

The **first** one is called "Sensor Panel", and it is a simplistic window where the user can easily see the sensors' data (Figure 1). Three bar graphs show the light, temperature and accelerometer values, a digital compass displays the Engduino heading, taking as pointing direction the "E" shape orientation, and a two phase graph represents the button state.

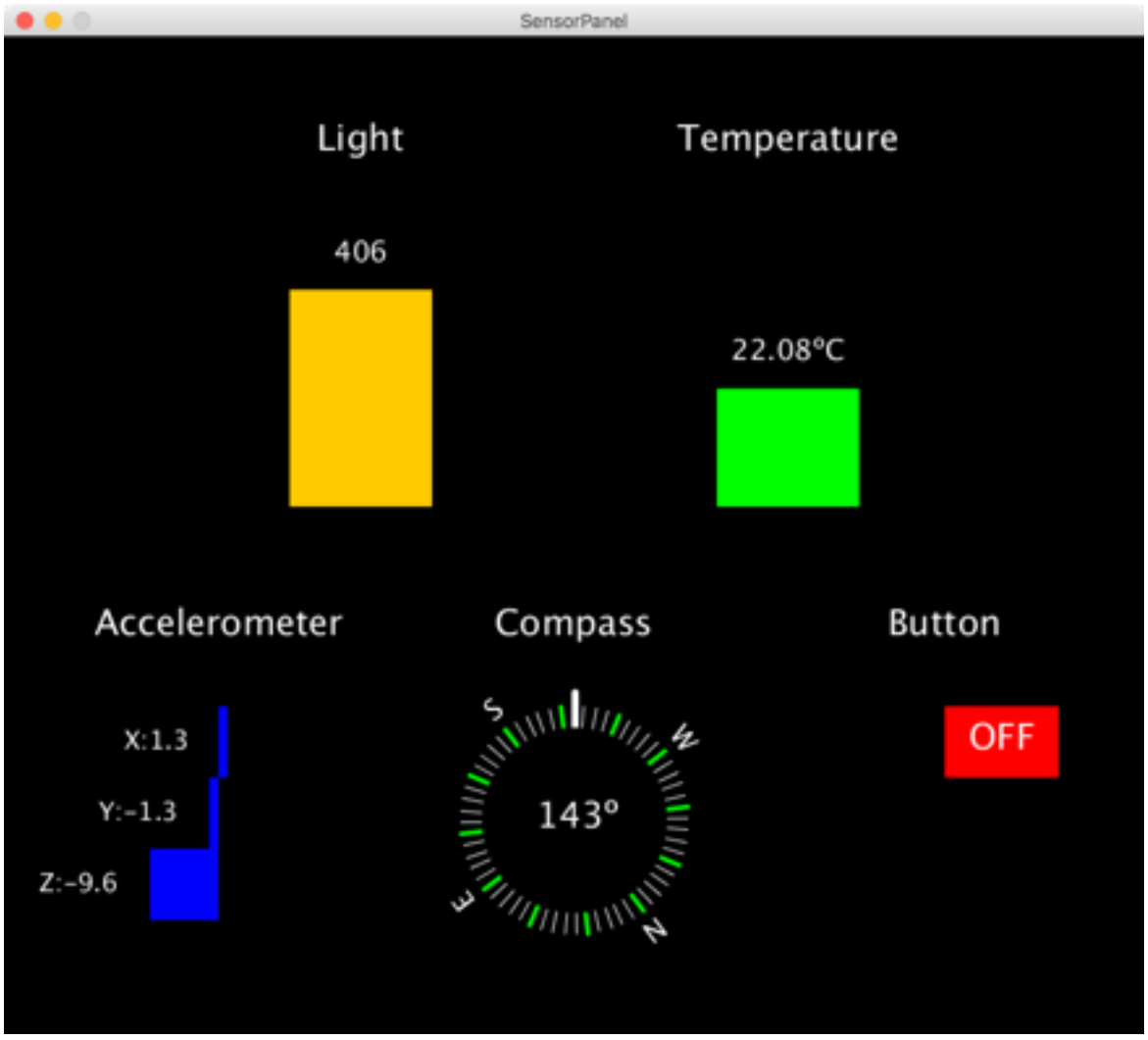

FIGURE 1: SENSOR PANEL SCREENSHOT

The **second** program "Atoms System" uses the sensors' values to approximately recreate the behaviour of gas atoms (Figure 2). I have used the Processing 3D space to create a void with some particles randomly wiggling. The magnitude of this movement depends on the temperature, the higher it is, the more they move around. Given a temperature of -273ºC, the particles won't move at all. The background color also changes according to the light detected by the Engduino. In addition, the user can shift the camera by rotating and tilting the board. The compass heading is used for the y axis rotation and the x component of the accelerometer, which activates when the board button is pressed, rotates the camera around its x axis.

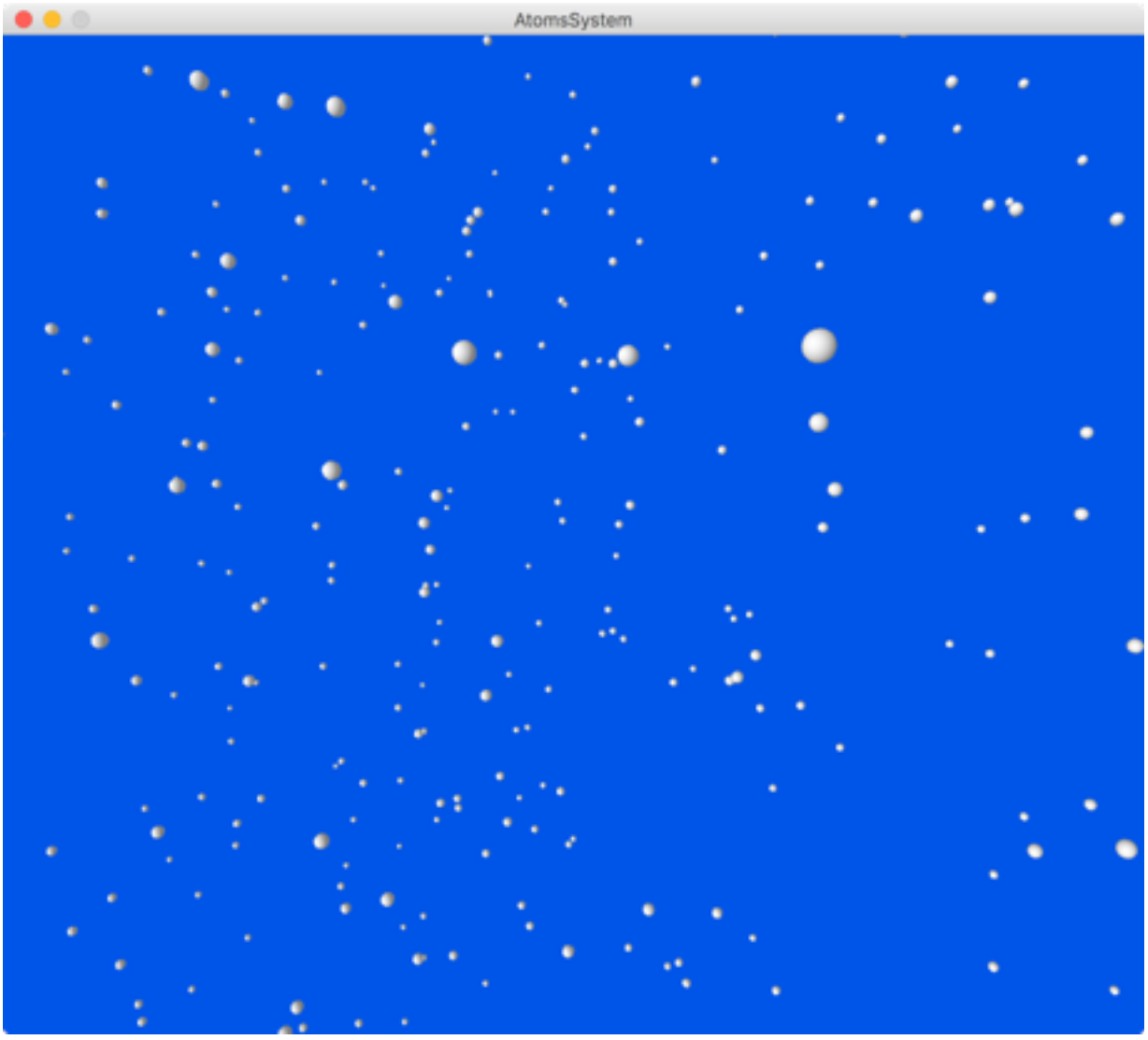

FIGURE 2: ATOMS SYSTEM SCREENSHOT

To run the applications above, first install the Sensor Print code on the Engduino and leave the board connected to the computer. Then run the desired Processing application normally, taking into account that the serial port index in the setup function has to be changed for the port index that your Engduino is connected in. A comment in the source code shows where to perform this adjustment.

II. 3D Printting Program

Using the OpenSCAD software I've modeled a rotating Engduino stand (Figure 3). It' is made out of three separate pieces: the frame, which includes the base and the columns, and the two spinning bars. The Engduino would be fixed using both the USB port and the on-board hole. The USB slot in one of the bars is slightly rotated around the z axis to align the Engduino's anchor points and allow proper rotation. However, this is a design concept and if printed it probably wouldn't function correctly since the rotation axis might not be exact. Some further prototyping and precise calculations would be required.

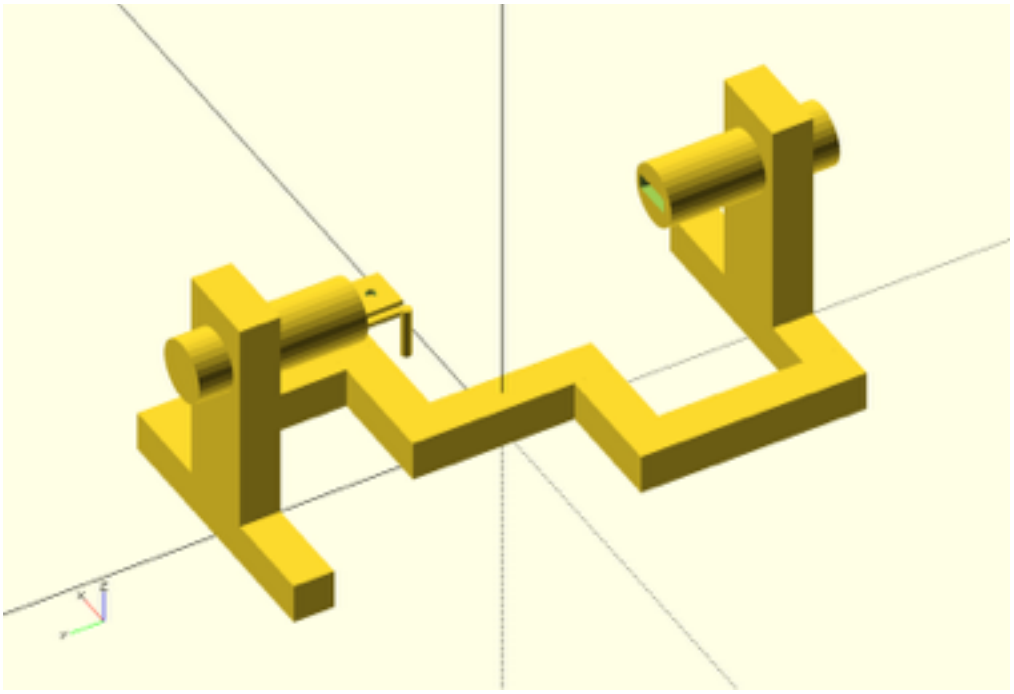

FIGURE 3: ENGDUINO ROTATING STAND 3D MODEL