INTRODUCTION TO ROBOTICS AND INTELLIGENT SYSTEMS LAB

Lab 3

Aryans Rathi

a.rathi@jacobs-university.de

Suraj Giri

s.giri@jacobs-university.de

Riley Edwin Sexton

r.sexton@jacobs-university.de

Instructor's Name **Dr. Fangning Hu**

Task 3.1

In this task, the variables 'rotx' and 'roty' were given to us. These variables are the reason why the box rotates as the cursor is moved around with the mouse. RotateX() and RotateY() are also similar functions which help to rotate the figure around the x and y axis at a specific angle respectively and because of the syntax in processing the angles of rotateX() and rotateY() should be provided in radians.

Task 3.2

In this task we set up the circuit as shown in fig1. For accessing the code Arduino example was opened up from 'file' settings and the code for MMA8451 was selected. After the code was written and compiled, it was ready for being uploaded. Once code had been successfully uploaded, the serial monitor opened up and raw count, acceleration, orientation etc shown up. The accelerometer in the chip detects the orientation and tilt and displays it in the serial monitor with help of the code in Arduino.

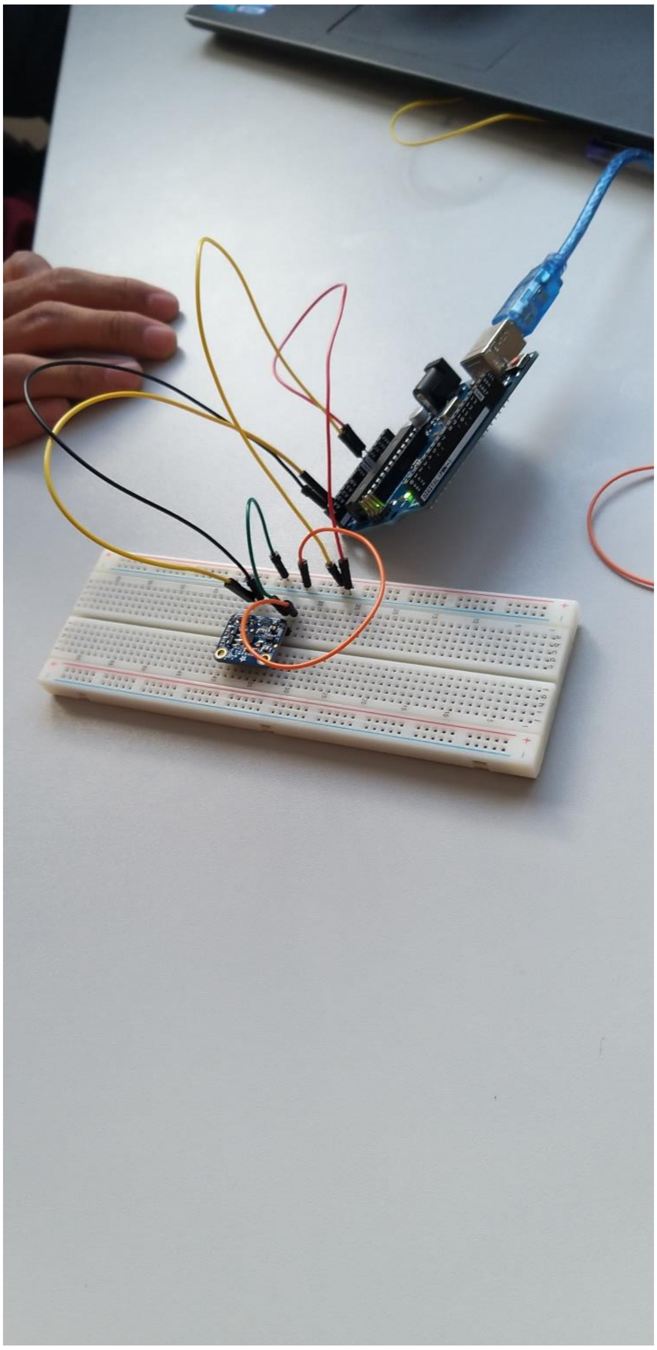


Fig 1 - Circuit for task 3.2

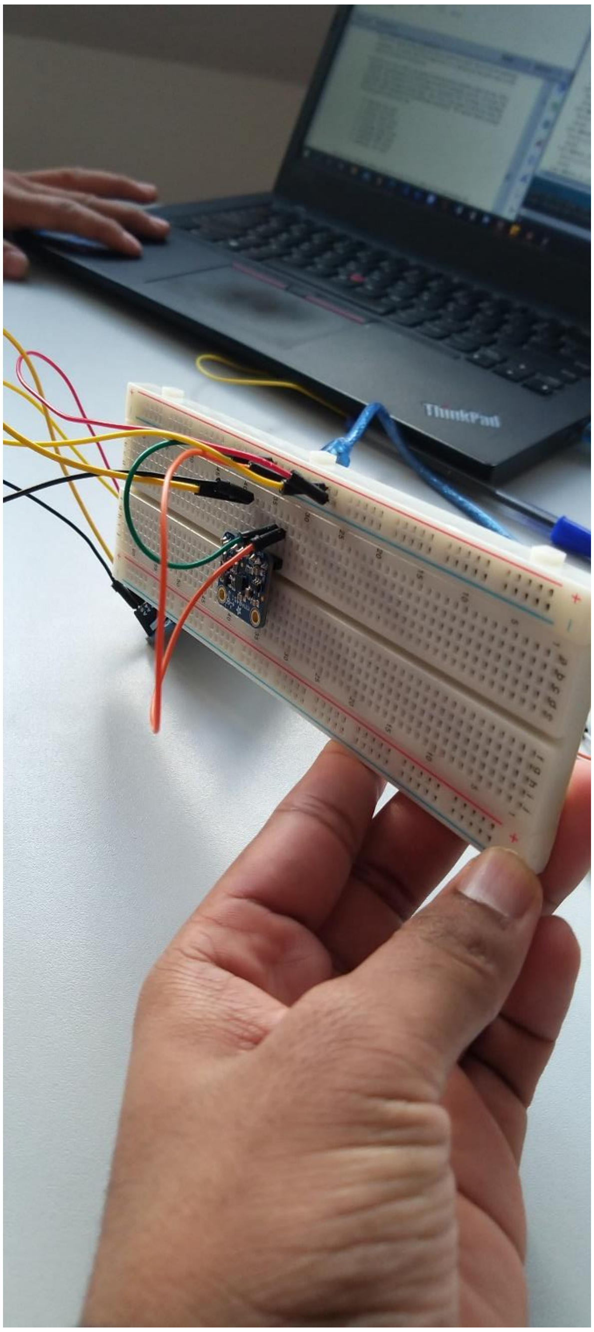
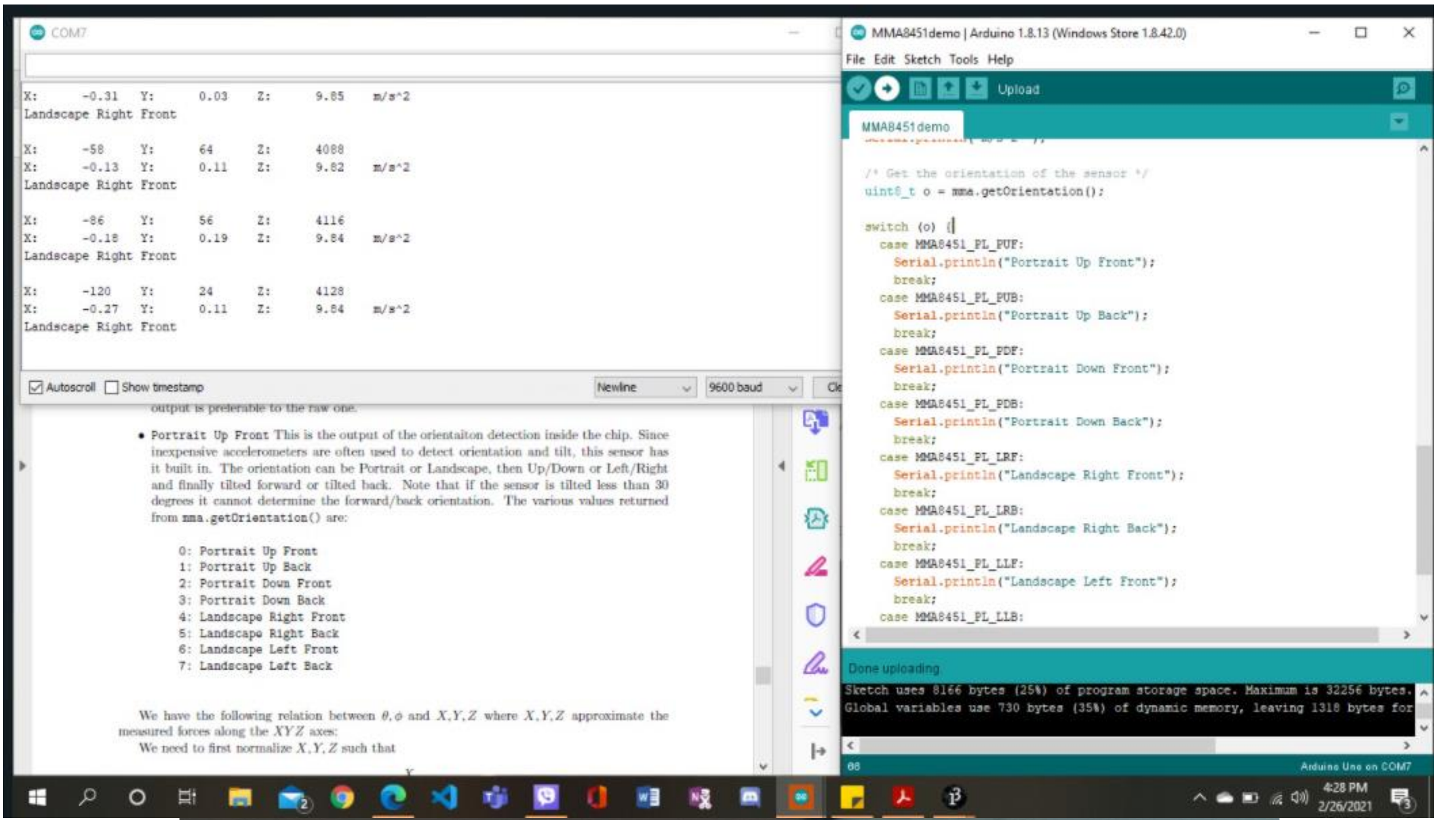
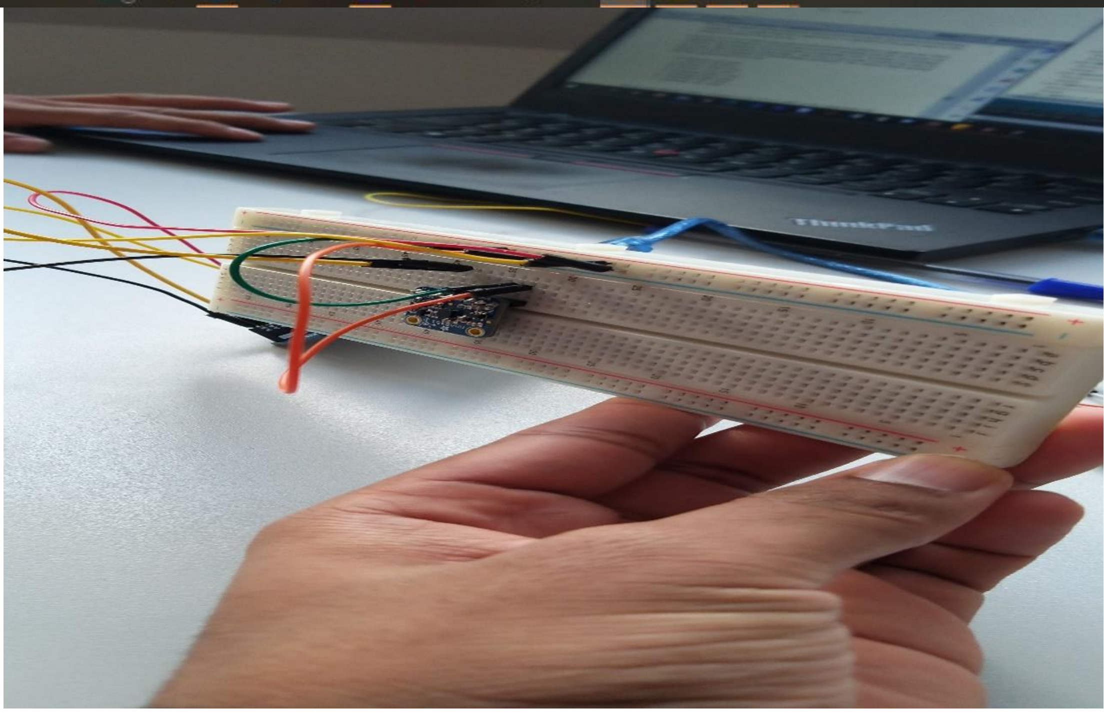


Fig 2- Circuit for task 3.2



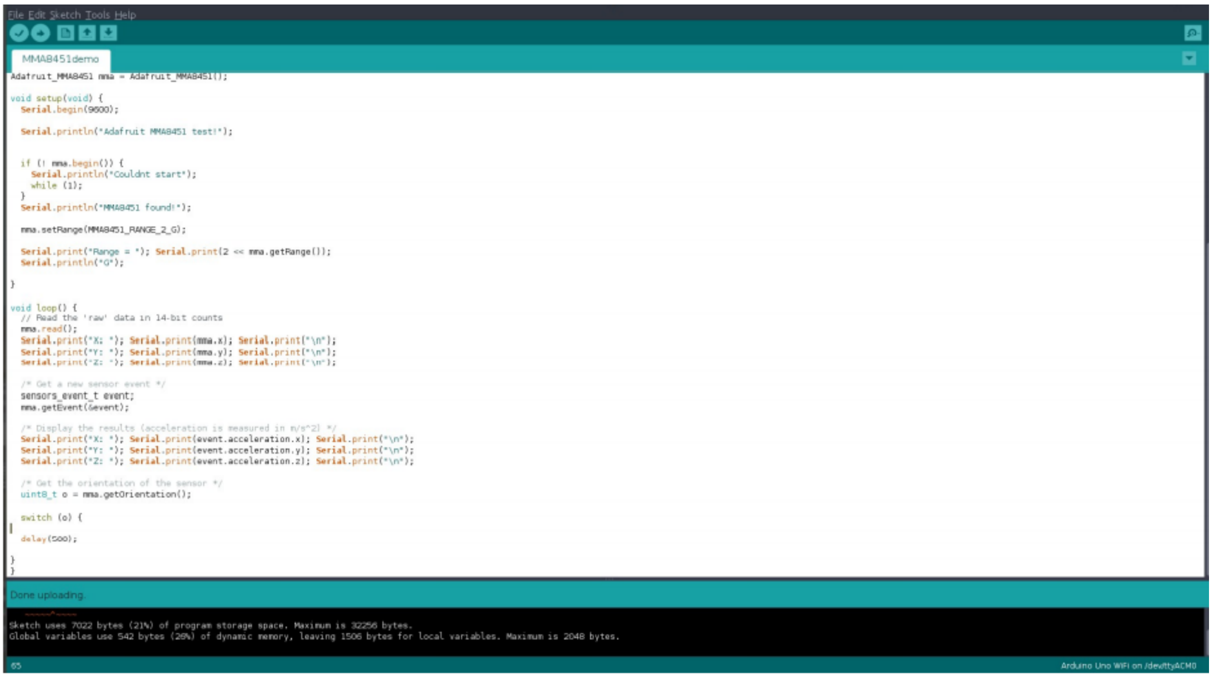


These figures shows the output of orientation given through Serial Monitor

The second line for X,Y and Z are basically Adafruit Sensor'ified which is in m/s² is the SI unit for acceleration. It does not matter what range it is set to, the units will not change and always be the same.

Task 3.3

Modified the code for task 3.1.



Modified code

3.2 Completed all the left out place in the code.

```
void draw()
{
while (myPort.available() > 0){
receivedString = myPort.readStringUntil(lf);
if (receivedString != null)
{
boolean newSample = parseStringForAccelerations(receivedString);
if (newSample){
print("Got acceleration : ");
print(acceleration[0]);
print(", ");
print(acceleration[1]);
print(", ");
print(acceleration[2]);
println(".");
// Compute the roll and pitch angles
float acc_norm = sqrt((acceleration[0] * acceleration[0]) + (acceleration[1] * acceleration[1]) + (acceleration[2] * acceleration[2]));
float ax = acceleration[0] / acc_norm;
float ay = acceleration[1] / acc_norm;
float az = acceleration[2] / acc_norm;
// Fill -in: Compute roll and pitch
float pitch = asin(az / sqrt(ax * ax + ay * ay + az * az));
```

```
float az = acceleration[2] / acc_norm;
// Fill -in: Compute roll and pitch
float pitch = asin(az / sqrt(ax * ax + ay * ay + az * az));
float roll = atan(ay / cos(pitch));
float yaw = 0;
// Draw the rotated object here .
translate(width / 2.0, height / 2.0, -50);
// Fill -in: rotate by roll and pitch
rotateX(radians(pitch));
rotateY(radians(roll));
rotateZ(radians(yaw));
  draw_rgb_cube();
          }
        }
    }

    myPort.clear();
}
boolean parseStringForAccelerations(String str)
{
```

Using the formula giving in the lab manual yaw was set to zero. Rotate function was added.Roatex(pitch)rotates the cube along the x-axis.Pitch is in radians.Rotate(roll) rotates the cube along the y-axis.Roll is also set in radians. Rotate(yaw) rotates the cube along the z-axis.

The value of yaw is set to zero.