Embedded Systems Lab 3-A

Abdelrahman Khayri Saad - 1905906

Abdelrahman Adel Abdelfattah - 17012296

Components

1x MPU6050

Code

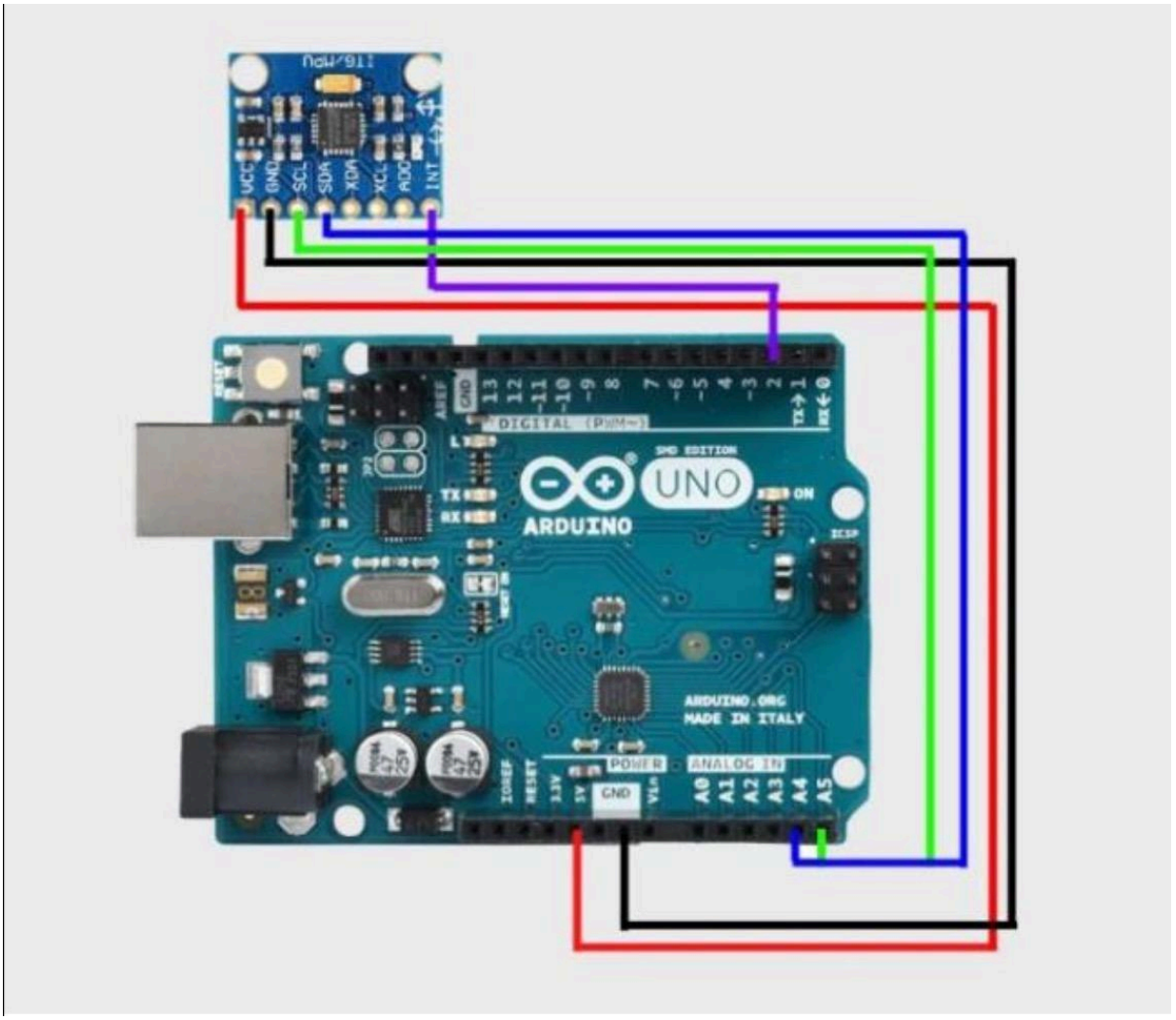
```
// Modified version of (c) Michael Schoeffler 2017,
http://www.mschoeffler.de

#include "Wire.h"  // This library allows you to communicate with I2C
devices.
#include "EEPROM.h"

const int MPU_ADDR = 0x68;  // I2C address of the MPU-6050. If AD0 pin
is set to HIGH, the I2C address will be 0x69.

int16_t accelerometer_x, accelerometer_y, accelerometer_z;  //
variables for accelerometer raw data
int16_t gyro_x, gyro_y, gyro_z;                             //
variables for gyro raw data
int16_t temperature;                                        //
variables for temperature data

double angle_x, angle_y, angle_z;

int OFFSET = 400;
int Z_BIAS = 16000;
int MAX = Z_BIAS * 2;
```

```
int eeprom_start=-1, eeprom_end=0;

double get_angle(int16_t num1, int16_t num2) {
  double angle = RAD_TO_DEG * (atan2(num1, num2));
  if (angle < 0) {
    angle += 360;
  }
  return angle;
}

void setup() {
  Serial.begin(9600);
  Wire.begin();
  Wire.beginTransmission(MPU_ADDR);  // Begins a transmission to the
I2C slave (GY-521 board)
  Wire.write(0x6B);                  // PWR_MGMT_1 register
  Wire.write(0);                     // set to zero (wakes up the
MPU-6050)
  Wire.endTransmission(true);
}
void loop() {
  Wire.beginTransmission(MPU_ADDR);
  Wire.write(0x3B);                          // starting with register
0x3B (ACCEL_XOUT_H) [MPU-6000 and MPU-6050 Register Map and
Descriptions Revision 4.2, p.40]
  Wire.endTransmission(false);               // the parameter indicates
that the Arduino will send a restart. As a result, the connection is
kept active.
  Wire.requestFrom(MPU_ADDR, 7 * 2, true);  // request a total of
7*2=14 registers

  // "Wire.read()<<8 | Wire.read();" means two registers are read and
stored in the same variable
  accelerometer_x = Wire.read() << 8 | Wire.read();  // reading
registers: 0x3B (ACCEL_XOUT_H) and 0x3C (ACCEL_XOUT_L)
  accelerometer_y = Wire.read() << 8 | Wire.read();  // reading
registers: 0x3D (ACCEL_YOUT_H) and 0x3E (ACCEL_YOUT_L)
  accelerometer_z = Wire.read() << 8 | Wire.read();  // reading
registers: 0x3F (ACCEL_ZOUT_H) and 0x40 (ACCEL_ZOUT_L)
  temperature = Wire.read() << 8 | Wire.read();      // reading
registers: 0x41 (TEMP_OUT_H) and 0x42 (TEMP_OUT_L)
```

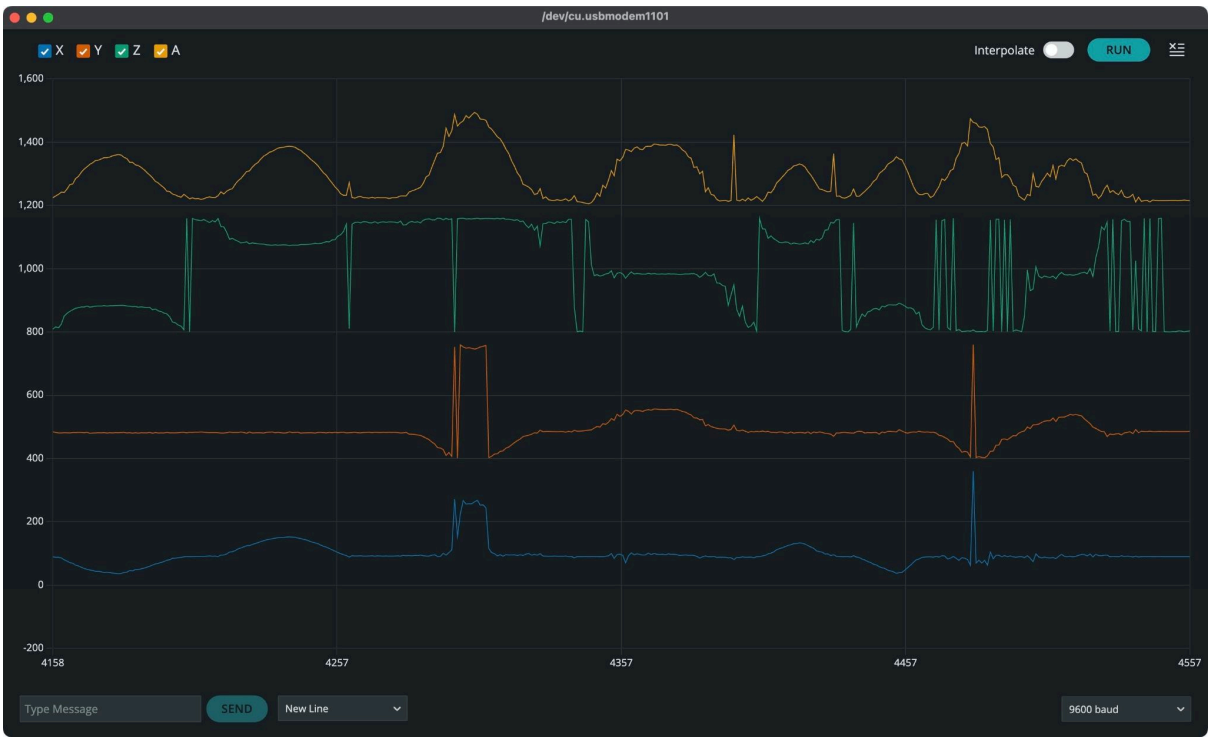
```
  gyro_x = Wire.read() << 8 | Wire.read();           // reading
registers: 0x43 (GYRO_XOUT_H) and 0x44 (GYRO_XOUT_L)
  gyro_y = Wire.read() << 8 | Wire.read();           // reading
registers: 0x45 (GYRO_YOUT_H) and 0x46 (GYRO_YOUT_L)
  gyro_z = Wire.read() << 8 | Wire.read();           // reading
registers: 0x47 (GYRO_ZOUT_H) and 0x48 (GYRO_ZOUT_L)

  //TODO: Your code goes here
  // Calculate acceleration angles
  angle_x = get_angle(accelerometer_z, accelerometer_y);
  angle_y = get_angle(accelerometer_z, accelerometer_x);
  angle_z = get_angle(accelerometer_y, accelerometer_x);

  angle_y += OFFSET * 1;
  angle_z += OFFSET * 2;

  accelerometer_z -= Z_BIAS;

  double acceleration = sqrt(pow(accelerometer_x, 2) +
pow(accelerometer_y, 2) + pow(accelerometer_z, 2));

  acceleration = map(acceleration, 0, MAX, 0, 360) + OFFSET * 3;

  plotter(angle_x, angle_y, angle_z, acceleration);

  //Writing latest reading in EEPROM
  EEPROM.put(eeprom_end, acceleration);
  eeprom_end = (eeprom_end+1) % 10;
  if(eeprom_start == -1 || eeprom_start == eeprom_end) eeprom_start++;
  //If collected 10 or more readings
  if(eeprom_start == eeprom_end)
  {
    //Calculate steady state acceleration
    double ss_acc = 0;
    for(int i=0; i<10; i++)
    {
      double read;
      EEPROM.get(i, read);
      ss_acc += read;
    }
    ss_acc /= 10;
    Serial.print("Steady state acceleration = ");
    Serial.println(ss_acc);
```

```
  }

  // delay
  delay(2);
}

void plotter(double angle_x, double angle_y, double angle_z, double
acceleration) {
  Serial.print("X:");
  Serial.print(angle_x);
  Serial.print("\tY:");
  Serial.print(angle_y);
  Serial.print("\tZ:");
  Serial.print(angle_z);
  Serial.print("\tA:");
  Serial.println(acceleration);
}
```

Circuit Diagram



Outputs



Citations

https://math.stackexchange.com/questions/2874301/transforming-x-y-z-acceleration-into-x-and-y-tilt-angles