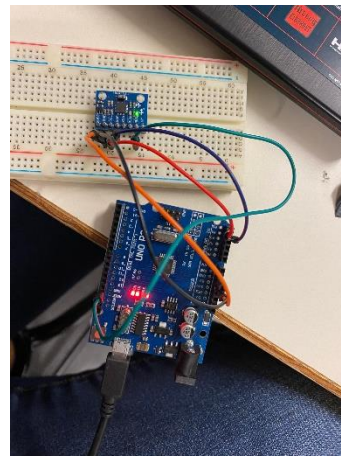
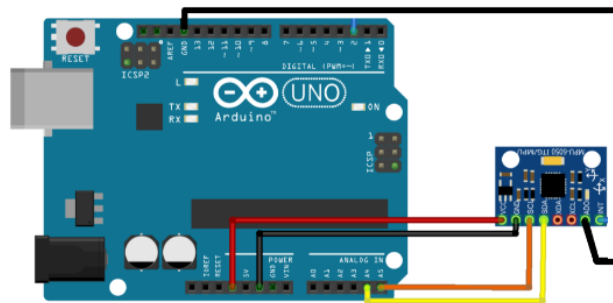


Compte rendu séance 2

Pendant cette séance, je me suis penché sur le fonctionnement de l'accéléromètre et plus particulièrement le code permettant de prendre des mesures.

Pour le branchement de l'accéléromètre, j'ai copié celui du cours d'Arduino :



J'ai ensuite suivi la démarche sur comment obtenir des données dites « brutes » du site <http://gilles.thebault.free.fr/spip.php?article32> (installation bibliothèque puis code) :

Ici les données dites « brutes » :

```
Test_Valeur_msd | Arduino 1.8.15
Fichier Edition Croquis Outils Aide

Test_Valeur_msd
// MPU-6050 Short Example Sketch
// By Arduino User @ohiochi
// August 17, 2014
// Public Domain
#include <Wire.h>
const int MPU_addr=0x68; // I2C address of the MPU-6050
int16_t AccX,AccY,AccZ,Tmp,GyX,GyY,GyZ;
void setup() {
  Wire.begin();
  Wire.beginTransmission(MPU_addr);
  Wire.write(0x6B); // PWR_MGMT_1 register
  Wire.write(0); // set to zero (wakes up the MPU-6050)
  Wire.endTransmission(true);
  Serial.begin(9600);
}
void loop() {
  Wire.beginTransmission(MPU_addr);
  Wire.write(0x3B); // starting with register 0x3B (ACCEL_XOUT_H)
  Wire.endTransmission(false);
  Wire.requestFrom(MPU_addr,14,true); // request a total of 14 registers
  AccX=Wire.read(); // 0x3B (ACCEL_XOUT_H) & 0x3C (ACCEL_XOUT_L)
  AccY=Wire.read(); // 0x3D (ACCEL_YOUT_H) & 0x3E (ACCEL_YOUT_L)
  AccZ=Wire.read(); // 0x3F (ACCEL_ZOUT_H) & 0x40 (ACCEL_ZOUT_L)
  Tmp=Wire.read(); // 0x41 (TEMP_OUT_H) & 0x42 (TEMP_OUT_L)
  GyX=Wire.read(); // 0x43 (GYRO_XOUT_H) & 0x44 (GYRO_XOUT_L)
  GyY=Wire.read(); // 0x45 (GYRO_YOUT_H) & 0x46 (GYRO_YOUT_L)
  GyZ=Wire.read(); // 0x47 (GYRO_ZOUT_H) & 0x48 (GYRO_ZOUT_L)
  Serial.print("AccX = "); Serial.print(AccX);
  Serial.print(" | AccY = "); Serial.print(AccY);
  Serial.print(" | AccZ = "); Serial.print(AccZ);
  Serial.print(" | Tmp = "); Serial.print(Tmp/340.00+36.53); //reguation for temperature in degrees C from datasheet
  Serial.print(" | GyX = "); Serial.print(GyX);
  Serial.print(" | GyY = "); Serial.print(GyY);
  Serial.print(" | GyZ = "); Serial.print(GyZ);
  delay(333);
}
```

```
COM1
AccX = 954 | AccY = 240 | AccZ = 16084 | Tmp = 26.98 | GyX = 66 | GyY = -487 | GyZ = 50
AccX = 880 | AccY = 204 | AccZ = 16116 | Tmp = 26.98 | GyX = 35 | GyY = -484 | GyZ = 22
AccX = 812 | AccY = 340 | AccZ = 16084 | Tmp = 26.88 | GyX = 36 | GyY = -427 | GyZ = 51
AccX = 528 | AccY = 308 | AccZ = 15964 | Tmp = 26.98 | GyX = 61 | GyY = -431 | GyZ = 40
AccX = 940 | AccY = 244 | AccZ = 16014 | Tmp = 26.84 | GyX = 44 | GyY = -485 | GyZ = 34
AccX = 600 | AccY = 204 | AccZ = 16020 | Tmp = 27.02 | GyX = 54 | GyY = -413 | GyZ = 29
AccX = 448 | AccY = 308 | AccZ = 16120 | Tmp = 26.93 | GyX = 69 | GyY = -434 | GyZ = 26
AccX = 608 | AccY = 296 | AccZ = 16092 | Tmp = 27.02 | GyX = 71 | GyY = -421 | GyZ = 31
AccX = 808 | AccY = 288 | AccZ = 15968 | Tmp = 26.93 | GyX = 51 | GyY = -411 | GyZ = 28
AccX = 488 | AccY = 332 | AccZ = 16084 | Tmp = 26.88 | GyX = 45 | GyY = -412 | GyZ = 38
AccX = 540 | AccY = 152 | AccZ = 16020 | Tmp = 26.93 | GyX = 53 | GyY = -416 | GyZ = 42
AccX = 776 | AccY = -332 | AccZ = 16044 | Tmp = 26.93 | GyX = -632 | GyY = -324 | GyZ = -307
AccX = 2156 | AccY = -652 | AccZ = 14672 | Tmp = 27.02 | GyX = 3259 | GyY = -3714 | GyZ = -516
AccX = 2476 | AccY = 408 | AccZ = 15724 | Tmp = 26.88 | GyX = 730 | GyY = 843 | GyZ = -1105
AccX = 2840 | AccY = 872 | AccZ = 16520 | Tmp = 26.93 | GyX = 903 | GyY = 1473 | GyZ = 2004
AccX = 428 | AccY = 112 | AccZ = 16352 | Tmp = 26.88 | GyX = 994 | GyY = -2362 | GyZ = -1627
```

Avant mouvement

Après mouvement

Ici les données obtenues (angle en combinant le gyropode et l'accéléromètre avec un "filtre complémentaire") :



(via la formule : $\text{angle} = 0,98 \times (\text{angle} + \text{Gy} \times 0,01 / 131) + 0,02 \times (\text{angle de l'accéléromètre en } ^\circ)$)

Pour cela il faudra peut-être que je détermine si les valeurs de base sont constantes et ainsi je pourrais avoir un vecteur position en intégrant 2 fois ces composantes (ou d'une autre manière).