Module 3 changement dans le code : rajout de Delay pour organiser le rendu du fichier csv

//From Arduino to Processing to Txt or cvs etc.

//import

import processing.serial.\*;

//import une bibliothèque

//declare

PrintWriter output;

//déclare que l'on va créer un fichier

Serial udSerial;

// déclare la variable udSerial

void setup() {

udSerial = new Serial(this, Serial.list()[0], 9600);

//établie une connection entre le port et l'ordinateur pour faire passer la variable

output = createWriter ("battements.csv");

// créer le fuchier csv

}

void draw() {

delay(1000);

String SenVal = udSerial.readString();

delay(1000);

//si la variable udSerial est supérieur a 0, elle est alors converie en chaine de caratère

if (SenVal != null) {

delay(1000);

output.println(SenVal);

delay(1000);

// si une valeur n'est pas égal a 0 elle est alors écrit dans le fichier csv

}

}

void keyPressed(){

output.flush();

output.close();

//stop et ferme le programme lorsqu'on appuie sur une touche

exit();

}

Module 1 : on a deux fonctions : la première se nomme pulsation et permet de nous donner la valeur lue par le capteur

La seconde fonction, s’appel cardiaque nous permet de mesurer le bpm de l’utilisateur.

Il y a des problèmes dans le code du module 1, car la valeur lue par le capteur s’affiche correctement, mais le calcul du bpm est décrit comme une constante dans le fichier battements.csv

Dans le code du module 1, nous avons 3 fichiers cardio.h, cardio.c et main.c

Dans cardio.h

#ifndef CARDIO\_H

#define CARDIO\_H

int pulsation();

//fonction qui va nous permettre de lire la pulsation

int cardiaque(int t1, int t2);

//fonction qui va nous permettre de calculer le bpm

Dans cardio.h

#include "cardio.h"

#define pin A0

#define valeur\_au\_repos 45

int pulsation() {

int valeur\_capteur;

int maxi=0;

int t1=0;

while (analogRead(pin)>valeur\_au\_repos) {

valeur\_capteur = analogRead (pin);

if (valeur\_capteur>maxi){

maxi=valeur\_capteur;

t1=millis();

delay(10);

return maxi

//va nous donner la valeur lu par le capteur et va la définir comme valeur maximal

}

else if (valeur\_capteur<maxi){

delay (10);

}}

return t1;

}

int cardiaque(int t1, int t2)

{

if (t1-t2>0){

t1=millis();

t2=millis();

delay(10);

return(60000/(t1-t2));

//calcul qui va nous permettre de trouver le bpm

}

}

Et main.c

#include "cardio.c"

int LED1;

void setup() {

pinMode(LED1, OUTPUT);

Serial.begin(9600);

}

void loop() {

int t1;

int t2;

int sensorValue = analogRead(A0);

pulsation();

Serial.print(sensorValue);

Serial.print(";");

Serial.println(cardiaque(t1, t2));

t1=t2;

}