Rapport de stage N°6. Leonardo Manole.

Pendant la première partie de cette séance, nous avons dû remédier à un petit problème survenu en cours de route. En effet, l'un des câbles de la photorésistance s'est détaché de la soudure avec l'étain. Nous avons donc procédé à la re-soudure avec de l'étain.

Dans la deuxième partie de la séance, nous avons entamé la phase finale de notre projet, qui concerne le code du panneau solaire.

Dans un premier temps, nous avons effectué des recherches sur Internet pour nous inspirer de certaines parties de code. Cependant, nous n'avons trouvé rien de concrètement intéressant, à l'exception de quelques grandes idées et d'explications sur le fonctionnement du code sur des forums.

Finalement, en combinant nos recherches sur Internet et nos connaissances en programmation, nous avons élaboré le code ci-dessous ;

#include <Servo.h>

// Déclaration des objets Servo

Servo servoHorizontal;

Servo servoVertical;

// Broches pour les photoresistances

int photoresistorPin1 = A3;

int photoresistorPin2 = A2;

int photoresistorPin3 = A0;

int photoresistorPin4 = A1;

// Variables pour stocker les valeurs lues des photoresistances

int value1, value2, value3, value4;

void setup() {

  servoHorizontal.attach(2);

  servoVertical.attach(3);

  Serial.begin(9600);

}

void loop() {

// Lecture des valeurs des photoresistances

  value1 = analogRead(photoresistorPin1);

  value2 = analogRead(photoresistorPin2);

  value3 = analogRead(photoresistorPin3);

  value4 = analogRead(photoresistorPin4);

// Affichage des valeurs lues pour débogage

  Serial.print("Value 1: ");

  Serial.println(value1);

  Serial.print("Value 2: ");

  Serial.println(value2);

  Serial.print("Value 3: ");

  Serial.println(value3);

  Serial.print("Value 4: ");

  Serial.println(value4);

// Appel de la fonction pour orienter les servomoteurs en fonction des valeurs lues

  orienterPanneauSolaire(value1, value2, value3, value4);

// Ajout d'un léger délai pour éviter la surcharge du processeur

  delay(50);

}

void orienterPanneauSolaire(int v1, int v2, int v3, int v4) {

// Seuil de déclenchement pour les photoresistances

  int seuil = 500;

// Facteur proportionnel pour adoucir les mouvements (ajustez selon vos besoins)

  float facteurProportionnel = 0.1;

// Angle maximal pour les servomoteurs

  int angleMax = 180;

// Angle minimal pour les servomoteurs

  int angleMin = 0;

// Ajustez ces valeurs en fonction de votre configuration réelle

  int angleHorizontal = map(v1 - v2, -1023, 1023, angleMin, angleMax);

  int angleVertical = map(v3 - v4, -1023, 1023, angleMin, angleMax);

// Limitez les angles pour éviter des mouvements excessifs

  angleHorizontal = constrain(angleHorizontal, angleMin, angleMax);

  angleVertical = constrain(angleVertical, angleMin, angleMax);

// Adoucir les mouvements avec un facteur proportionnel

  int angleHorizontalActuel = servoHorizontal.read();

  int angleVerticalActuel = servoVertical.read();

  int angleHorizontalAjuste = angleHorizontalActuel + facteurProportionnel \* (angleHorizontal - angleHorizontalActuel);

  int angleVerticalAjuste = angleVerticalActuel + facteurProportionnel \* (angleVertical - angleVerticalActuel);

// Affichage des angles pour débogage

  Serial.print("Angle Horizontal: ");

  Serial.println(angleHorizontalAjuste);

  Serial.print("Angle Vertical: ");

  Serial.println(angleVerticalAjuste);

// Contrôle des servomoteurs

  servoHorizontal.write(angleHorizontalAjuste);

  servoVertical.write(angleVerticalAjuste);

}

Toutefois ce code ne marche totalement nous avons rencontrés différents soucis notamment au niveau des servomoteurs ceux-ci fonctionnés de manière trop brutale on doit donc revoir notre code pour rendre le mouvement plus fluide.

Un autre problème est au niveau des photorésistances celles – ci semblent détecter une variation de la luminosité mais le déplacement n’est pas celui escompté donc finalement on doit corriger ça aussi dans le code.