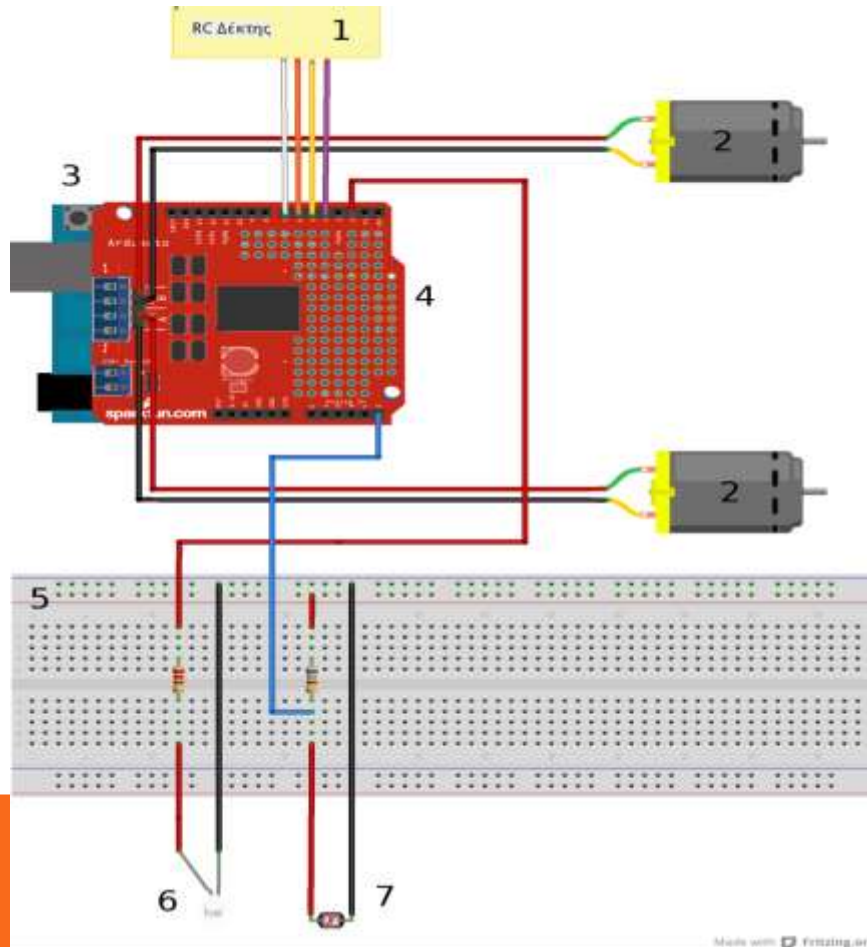


ΜΕΛΕΤΗ ΚΑΙ ΚΑΤΑΣΚΕΥΗ ΑΥΤΟΝΟΜΟΥ ΡΟΜΠΟΤΙΚΟΥ ΟΧΗΜΑΤΟΣ

Γιώργος Βασιλείου
Ιάσοντας Νικολάου

Υπεύθυνος καθηγητής:
Ανδρέας Χατζηορφανός

ΣΥΝΔΕΣΜΟΛΟΓΙΑ



1. RC δέκτης
2. Μοτέρ
3. Μικροελεγκτής Arduino
4. Arduino Shield
5. Breadboard
6. Led
7. Φωτοαντίσταση

Ο ΚΩΔΙΚΑΣ



```

#define THRO_PIN 7
#define AILE_PIN 6
#define ELEV_PIN 5
#define RUDD_PIN 4

#define LIGHT_GEN_PIN 2
#define LIGHT_SENS_PIN A5

#define DIRECTION_A_PIN 12 // Left Wheel
#define SPEED_A_PIN 3
#define DIRECTION_B_PIN 13 // Right Wheel
#define SPEED_B_PIN 11

#define STICK_MIDDLE 1515
#define STICK_RANGE 758

#define STEERING_FACTOR 0.5 //Παίρνει τιμές 0-1
int LIGHT_DARK, LIGHT_BRIGHT;
unsigned long Time0,TimeSetup;
float
Duration_THRO,Duration_AILE,Duration_ELEV,Duration_RUDD,Speed,Speed_A,Speed_B,Steering,LightValue;
boolean Direction;

void setup() {
  Serial.begin(9600);
  pinMode(THRO_PIN, INPUT);
  pinMode(AILE_PIN, INPUT);
  pinMode(ELEV_PIN, INPUT);
  pinMode(RUDD_PIN, INPUT);
  pinMode(DIRECTION_A_PIN, OUTPUT); //Initiates Motor Channel A pin
  pinMode(DIRECTION_B_PIN, OUTPUT); //Initiates Motor Channel B pin
  digitalWrite(LIGHT_GEN_PIN, HIGH);
  LIGHT_DARK=0;
  LIGHT_BRIGHT=900;

  TimeSetup=millis();
  while(millis()-TimeSetup<400){
    digitalWrite(DIRECTION_A_PIN, HIGH);
    analogWrite(SPEED_A_PIN, 200);

    if(analogRead(LIGHT_SENS_PIN)<LIGHT_BRIGHT){
      LIGHT_BRIGHT=LIGHT_SENS_PIN;
    }
    if(analogRead(LIGHT_SENS_PIN)>LIGHT_DARK){
      LIGHT_DARK=LIGHT_SENS_PIN;
    }
  }
  delay(10000);
}

```

```

void loop(){
  Duration_ELEV = pulseIn(ELEV_PIN, HIGH); //Διαβάζει τα stick
  Duration_AILE = pulseIn(AILE_PIN, HIGH);

  if(Duration_ELEV>STICK_MIDDLE){//Αν πάει μπροστά
    Direction=HIGH;
    Speed = (Duration_ELEV-STICK_MIDDLE)/(STICK_RANGE/2)*255; //0 έως 255
    digitalWrite(DIRECTION_A_PIN, HIGH);
    digitalWrite(DIRECTION_B_PIN, HIGH);
  }
  else {
    Direction=LOW;
    Speed = (STICK_MIDDLE-Duration_ELEV)/(STICK_RANGE/2)*255; //0 έως 255
    digitalWrite(DIRECTION_A_PIN, LOW);
    digitalWrite(DIRECTION_B_PIN, LOW);
  }

  Steering = (Duration_AILE-STICK_MIDDLE)/(STICK_RANGE/2); // -1 έως +1

  Speed_A = (1-Steering*STEERING_FACTOR)*Speed;
  Speed_B = (1+Steering*STEERING_FACTOR)*Speed;

  if(Speed_A>255){
    Speed_A=255;}
  else if(Speed_A<0){
    Speed_A=0;
  }

  if(Speed_B>255){
    Speed_B=255;}
  else if(Speed_B<0){
    Speed_B=0;
  }

  analogWrite(SPEED_A_PIN, Speed_A);
  analogWrite(SPEED_B_PIN, Speed_B);

  LightValue = analogRead(LIGHT_SENS_PIN); //Διαβάζει το φώς

  if (LightValue > (LIGHT_DARK + LIGHT_BRIGHT)/2) {
    Time0 = millis();
    Direction=!Direction;
    while ((millis()-Time0)<300){
      Speed_A = 255;
      Speed_B = Speed_A;
      digitalWrite(DIRECTION_A_PIN, Direction);
      digitalWrite(DIRECTION_B_PIN, Direction);
      analogWrite(SPEED_A_PIN, Speed_A);
      analogWrite(SPEED_B_PIN, Speed_B);
    }
  }
}

```

ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

```
TimeSetup=millis();
while(millis()-TimeSetup<400){
  digitalWrite(DIRECTION_A_PIN, HIGH);
  analogWrite(SPEED_A_PIN, 200);

  if(analogRead(LIGHT_SENS_PIN)<LIGHT_BRIGHT){
    LIGHT_BRIGHT=LIGHT_SENS_PIN;
  }
  if(analogRead(LIGHT_SENS_PIN)>LIGHT_DARK){
    LIGHT_DARK=LIGHT_SENS_PIN;
  }

}
delay(10000);
}
```

- Για 4 sec αναγνωρίζει, σύμφωνα με τις συνθήκες φωτισμού του χώρου, το άσπρο(LIGHT_BRIGHT) και το μαύρο(LIGHT_DARK)

ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

```
Duration_ELEV = pulseIn(ELEV_PIN, HIGH); // Διαβάζει τα stick
Duration_AILE = pulseIn(AILE_PIN, HIGH);

if(Duration_ELEV > STICK_MIDDLE) { // Αν πάει μπροστά
    Direction = HIGH;
    Speed = (Duration_ELEV - STICK_MIDDLE) / (STICK_RANGE / 2) * 255;
    digitalWrite(DIRECTION_A_PIN, HIGH);
    digitalWrite(DIRECTION_B_PIN, HIGH);
}
else {
    Direction = LOW;
    Speed = (STICK_MIDDLE - Duration_ELEV) / (STICK_RANGE / 2) * 255;
    digitalWrite(DIRECTION_A_PIN, LOW);
    digitalWrite(DIRECTION_B_PIN, LOW);
}
```

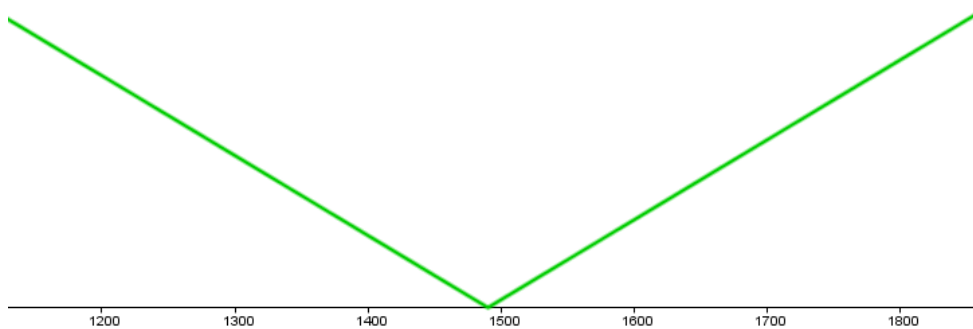
- Διαβάζουμε τις οδηγίες του χρήστη από το χειριστήριο
- Αν ο μοχλός είναι πάνω από το κέντρο δίνουμε εντολή για μπροστά
- Διαφορετικά δίνουμε εντολή για πίσω

ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

```
Steering = (Duration_AILE-STICK_MIDLE)/(STICK_RANGE/2); /
```

```
Speed_A = (1-Steering*STEERING_FACTOR)*Speed;
```

```
Speed_B = (1+Steering*STEERING_FACTOR)*Speed;
```



- Υπολογίζουμε την ταχύτητα για κάθε μοτέρ (Speed_A),(Speed_B), σύμφωνα με την παρακάτω συνάρτηση

ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

```
if(Speed_A>255){  
  Speed_A=255;}  
else if(Speed_A<0){  
  Speed_A=0;  
}
```

```
if(Speed_B>255){  
  Speed_B=255;}  
else if(Speed_B<0){  
  Speed_B=0;  
}
```

```
analogWrite(SPEED_A_PIN, Speed_A);  
analogWrite(SPEED_B_PIN, Speed_B);
```

- Περιορίζουμε το σύνολο τιμών 0-255 (ελάχιστη και μέγιστη ταχύτητα)
- Δίνουμε την ταχύτητα στα μοτέρ

ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

```
LightValue = analogRead(LIGHT_SENS_PIN); //Διαβάζει το φώς

if (LightValue > (LIGHT_DARK + LIGHT_BRIGHT)/2) {
  Time0 = millis();
  Direction=!Direction;
  while ((millis()-Time0)<300){
    Speed_A = 255;
    Speed_B = Speed_A;
    digitalWrite(DIRECTION_A_PIN, Direction);
    digitalWrite(DIRECTION_B_PIN, Direction);
    analogWrite(SPEED_A_PIN, Speed_A);
    analogWrite(SPEED_B_PIN, Speed_B);
  }
}
```

- Διαβάζουμε τις μετρήσεις της φωτοαντίστασης
- Αν το όχημα βρίσκεται στην μαύρη γραμμή κάνουμε όπισθεν.

ΕΥΧΑΡΙΣΤΟΥΜΕ!!!

