Human Follower Robot

Name: Tanase Paul-Gabriel

Group: 1221 A

Faculty: FILS

Introducere

Prezentarea pe scurt a proiectului:

```
Arduino Human Following Robot is a 4-wheeled robot car meant for education
and entertainment purposes.
```

When it is turned on, it will attempt to scan using a motion sensor combined with two IR sensors to determine the closest moving object next to him (in this case, a human hand). After the target is acquired, it will start following it until the object stops.

Descriere generală

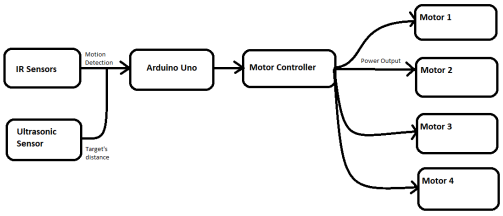
```
Arduino Human Following Robot is a 4-wheeled robot car meant for education
and entertainment purposes.
```

When it is turned on, it will attempt to scan using a motion sensor combined with two IR sensors to determine the closest moving object next to him (in this case, a human hand). After the target is acquired, it will start following it until the object stops.

```
The object scanning is done doing a process similar to that of sonars.
```

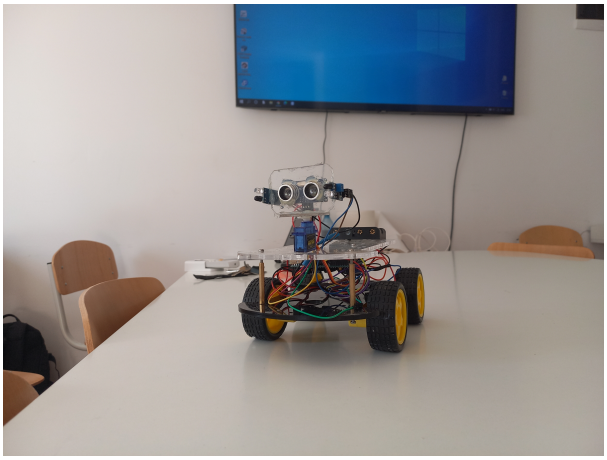
When the robot starts, it's “head” mounted on a servo motor will start rotating, scanning the surrounding enviroment trying to determine the closest object in front of him which emits infrared radiation. once both of the two IR sensors sense the target, the ultrasonic sensor will start pinging the object constatly to measure it's distance. Taking the angle of the servomotor and the distance obtained from the ultrasonic sensor, the motor controler then outputs the power for each of the 4 gear motors to head for the followed object. Once a certain distance treshold is reached (e.g. the robot gets too close or the object ran out of the sensor range) the car will cease it's movement.

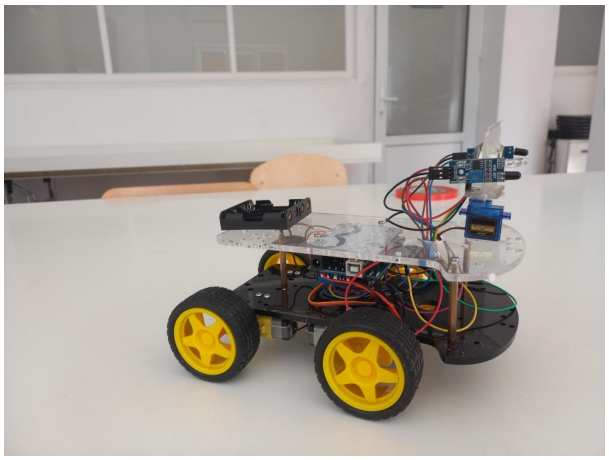
O schemă bloc cu toate modulele proiectului vostru, atât software cât şi hardware însoţită de o descriere a acestora precum şi a modului în care interacţionează.



Schemă bloc:

Poze robot:



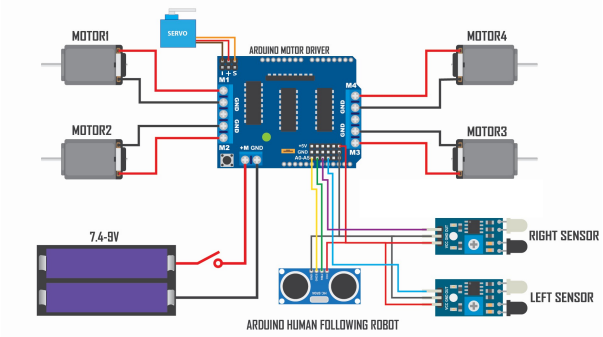


Hardware Design

Lista Piese:

1. Arduino Uno R3
2. 4 Motoare
3. 1 servomotor
4. 1 Arduino Motor Controller
5. 2 senzori IR
6. Jumper Wires

7. 2 Li-Ion Battery *



* Couldn't find Li-ion batteries. Replaced with a 9V battery

Software Design

Descrierea codului aplicaţiei (firmware):

```
Mediu de dezvoltare: **ArduinoIDE**

Librarii folosite: **NewPing**,**Servo**,**AFMotor**
```

Cod:

```
#include<NewPing.h>
#include<Servo.h>
#include<AFMotor.h>

#define RIGHT A2              // Right IR sensor connected to analog pin A2
of Arduino Uno:
#define LEFT A3               // Left IR sensor connected to analog pin A3
of Arduino Uno:
#define TRIGGER_PIN A1        // Trigger pin connected to analog pin A1 of
Arduino Uno:
#define ECHO_PIN A0           // Echo pin connected to analog pin A0 of
Arduino Uno:
#define MAX_DISTANCE 200      // Maximum ping distance:

unsigned int distance = 0;    //Variable to store ultrasonic sensor distance:
unsigned int Right_Value = 0; //Variable to store Right IR sensor value:
unsigned int Left_Value = 0;  //Variable to store Left IR sensor value:


NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);  //NewPing setup of pins
and maximum distance:
```

```
AF_DCMotor Motor1(1,MOTOR12_1KHZ);
AF_DCMotor Motor2(2,MOTOR12_1KHZ);
AF_DCMotor Motor3(3,MOTOR34_1KHZ);
AF_DCMotor Motor4(4,MOTOR34_1KHZ);

 Servo myservo;
 int pos=0;     //variable to store the servo position:

void setup() { // the setup function runs only once when power on the board
or reset the board:

   Serial.begin(9600); //initailize serial communication at 9600 bits per
second:
   myservo.attach(10); // servo attached to pin 10 of Arduino UNO
{
for(pos = 90; pos <= 180; pos += 1){    // goes from 90 degrees to 180
degrees:
  myservo.write(pos);                   //tell servo to move according to
the value of 'pos' variable:
  delay(15);                            //wait 15ms for the servo to reach
the position:
  }
for(pos = 180; pos >= 0; pos-= 1) {     // goes from 180 degrees to 0
degrees:
  myservo.write(pos);                   //tell servo to move according to
the value of 'pos' variable:
  delay(15);                            //wait 15ms for the servo to reach
the position:
  }
for(pos = 0; pos<=90; pos += 1) {       //goes from 180 degrees to 0 degrees:
  myservo.write(pos);                   //tell servo to move according to
the value of 'pos' variable:
  delay(15);                            //wait 15ms for the servo to reach
the position:
  }
}
   pinMode(RIGHT, INPUT); //set analog pin RIGHT as an input:
   pinMode(LEFT, INPUT);  //set analog pin LEFT as an input:
}

// the lope function runs forever
void loop() {

delay(50);                                          //wait 50ms between pings:
distance = sonar.ping_cm();                         //send ping, get distance
in cm and store it in 'distance' variable:
Serial.print("distance");
Serial.println(distance);                           // print the distance in
serial monitor:
```

```
    Right_Value = digitalRead(RIGHT);              // read the value from
Right IR sensor:
    Left_Value = digitalRead(LEFT);                // read the value from
Left IR sensor:

Serial.print("RIGHT");
Serial.println(Right_Value);                       // print the right IR
sensor value in serial monitor:
Serial.print("LEFT");
Serial.println(Left_Value);                        //print the left IR sensor
value in serial monitor:

if((distance > 1) && (distance < 15)){             //check wheather the
ultrasonic sensor's value stays between 1 to 15.
                                                   //If the condition is
'true' then the statement below will execute:
  //Move Forward:
  Motor1.setSpeed(130);  //define motor1 speed:
  Motor1.run(FORWARD);   //rotate motor1 clockwise:
  Motor2.setSpeed(130);  //define motor2 speed:
  Motor2.run(FORWARD);   //rotate motor2 clockwise:
  Motor3.setSpeed(130);  //define motor3 speed:
  Motor3.run(FORWARD);   //rotate motor3 clockwise:
  Motor4.setSpeed(130);  //define motor4 speed:
  Motor4.run(FORWARD);   //rotate motor4 clockwise:

}else if((Right_Value==0) && (Left_Value==1)) {

  //Turn Left
  Motor1.setSpeed(150);  //define motor1 speed:
  Motor1.run(FORWARD);   //rotate motor1 cloclwise:
  Motor2.setSpeed(150);  //define motor2 speed:
  Motor2.run(FORWARD);   //rotate motor2 clockwise:
  Motor3.setSpeed(150);  //define motor3 speed:
  Motor3.run(BACKWARD);  //rotate motor3 anticlockwise:
  Motor4.setSpeed(150);  //define motor4 speed:
  Motor4.run(BACKWARD);  //rotate motor4 anticlockwise:
  delay(150);

}else if((Right_Value==1)&&(Left_Value==0)) {

  //Turn Right
  Motor1.setSpeed(150);  //define motor1 speed:
  Motor1.run(BACKWARD);  //rotate motor1 anticlockwise:
  Motor2.setSpeed(150);  //define motor2 speed:
  Motor2.run(BACKWARD);  //rotate motor2 anticlockwise:
  Motor3.setSpeed(150);  //define motor3 speed:
  Motor3.run(FORWARD);   //rotate motor3 clockwise:
  Motor4.setSpeed(150);  //define motor4 speed:
  Motor4.run(FORWARD);   //rotate motor4 clockwise:
```

```
  delay(150);

}else if(distance > 15) {

  //Stop
  Motor1.setSpeed(0);     //define motor1 speed:
  Motor1.run(RELEASE);    //stop motor1:
  Motor2.setSpeed(0);     //define motor2 speed:
  Motor2.run(RELEASE);    //stop motor2:
  Motor3.setSpeed(0);     //define motor3 speed:
  Motor3.run(RELEASE);    //stop motor3:
  Motor4.setSpeed(0);     //define motor4 speed:
  Motor4.run(RELEASE);    //stop motor4:
}
}
```

Rezultate Obţinute

Care au fost rezultatele obţinute în urma realizării proiectului vostru.

Concluzii

Download

Resurse proiect

O arhivă (sau mai multe dacă este cazul) cu fişierele obţinute în urma realizării proiectului: surse, scheme, etc. Un fişier README, un ChangeLog, un script de compilare şi copiere automată pe uC crează întotdeauna o impresie bună .

Fişierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fişierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul). **Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru_alin**.

Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite

Resurse Software:

- AFMotor GitHub documentation
- NewPing lib wiki
- Arduino Tutorials

Resurse Hardware:

- Arduino Motor Shield L293D General info
- How to connect sensors to Motor Shield (Quora answer)
- How to connect IR sensors to shield (Arduino forums)
- Example to attach Ultrasonic sensor

Export to PDF

From:
http://ocw.cs.pub.ro/courses/ - **CS Open CourseWare**

Permanent link:
http://ocw.cs.pub.ro/courses/pm/prj2022/robert/robotfollower

Last update: **2022/06/01 18:06**