



# Roboti Mobili

Profesor coordonator : Trăsnea Bogdan

Nume : Condrea Laurentiu Daniel

Programul de studii : Robotica

Grupa : 4LF811

An : 2023-2024



## Cuprins

1.Introducere.....	3
2.Arhitectura.....	3-5
3.Realizarea părţii HW.....	6-8
4.Realizarea părţii SW.....	9-14
5.Ansamblu final.....	14
6.Anexe.....	15

## 1. Introducere

Proiectul presupune asamblarea si programarea unui robot mobil cu braţ robotic care sa sorteze deşeurile manual, controlat de o telecomanda cu senzor infraroşu.

## 2. Arhitectură

Am folosit un braţ robotic deja existent. Bratul robotic are 5 motoare: baza care se roteste, cupla1, cupla2, cupla3 si efectorul. Motoarele funcţionează între 5-6.8 V. Am renunţat la suportul de lemn al braţului robotic.



Fig.2.1 Braţ robotic

După care, am asamblat şasiul tip tank, am găsit nişte mini tomberoane, am conectat toate motoarele si Arduino-ul la o sursa , am scris codul si controlam robotul folosind telecomanda de la senzorul infraroşu.

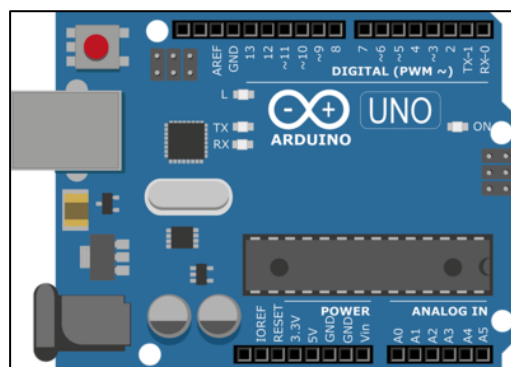


Fig.2.2 Plăcuță Arduino



Fig.2.3 Sursa

La şasiu am schimbat tijele cu unele mai lungi pentru a avea mai mult loc pentru baza braţului robotic, si am orientat motoarele senilelor sa nu frece.

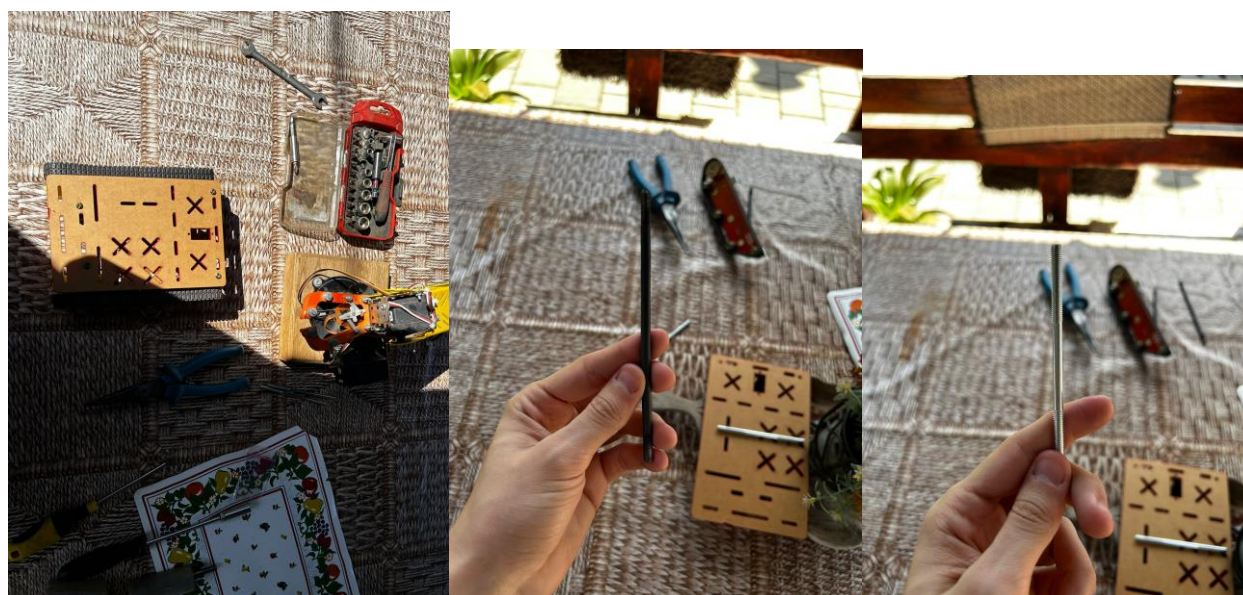


Fig.2.4 Şasiu



*Fig.2.5 Pubele de gunoi*

Inițial am vrut sa control robotul cu un shield de joystick dar nu mai aveam loc sa-l poziționez.  
Pentru a controla motoarele este nevoie de un senzor infraroșu (2.6).



*Fig.2.6 Senzor infraroșu*



### 3. Realizarea părţii HW

Ca prim pas, am asamblat şasiul tip tank, motoarele fiind cu tot cu modul ,iar locomoţiia se face pe senile. Singurul lucru ce l-am modificat au fost tije(Fig 2.4) dintre motoare pentru a avea mai mult loc pentru baza braţului robotic (3.1).

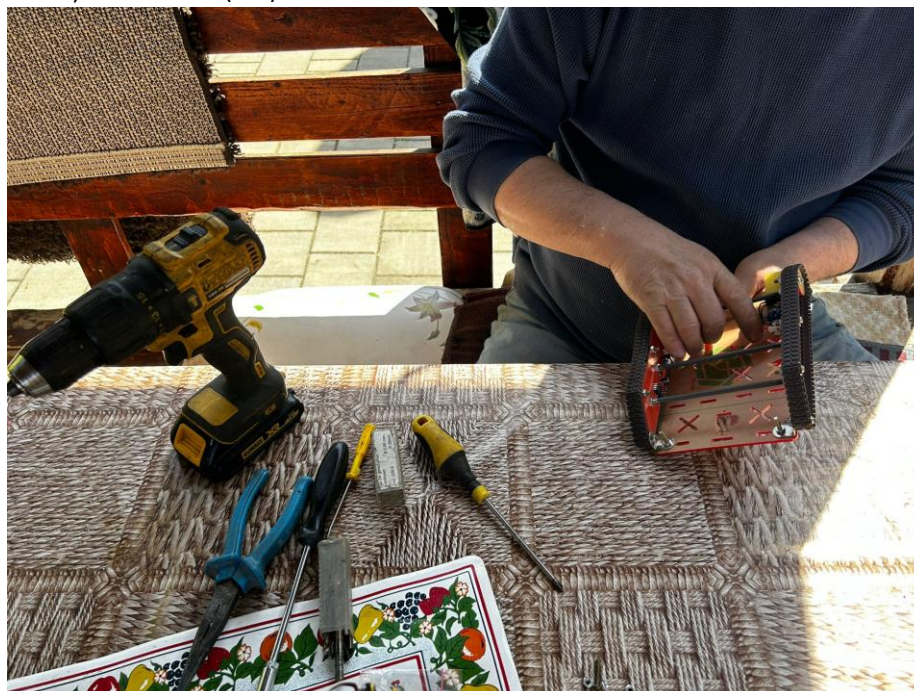


Fig3. 1

În figura (3.2) se pot vedea toate porturile folosite de pe plăcuţa Arduino.

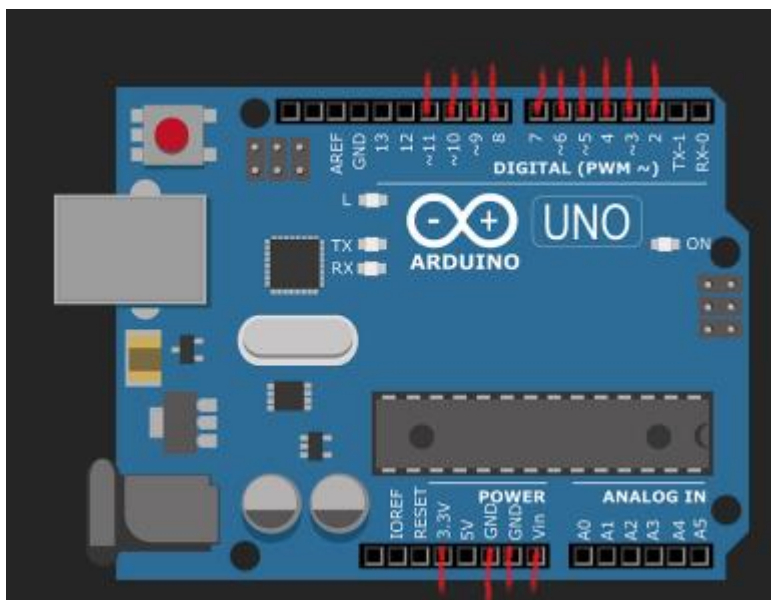


Fig3. 2

Porturile 3.3V, 7, GND, se conectează la senzorul infraroşu (3.3).



Fig3. 3

Porturile 2, 3, 4 şi 9 se conectează la cele doua drivere de motoare. Cele doua drivere sunt poziţionate in spate, stânga si dreapta. Fiind conectata senila de pe partea dreaptă la un driver iar senila de pe partea stânga la celălalt driver. Porturile 2 şi 4 sunt conectate la porturile A1A si B1A, iar porturile 3 şi 9 sunt conectate la porturile A1B şi B1B. Conexiunea pentru ambele drivere se poate vedea in imaginea (3.4).Cablurile(MA,MB,GND,VCC) sunt conectate la o sursa de 5V pentru alimentare care mai apoi va fi înlocuita cu transformatorul de mai sus si cu mufe corespunzătoare(Fig 3.7) deoarece mufarea cablurilor a fost destul de dificila.(Fig 2.3).De asemenea, a trebuit sa folosesc un cablu de dimensiuni mai mari,2x0,5, deoarece cel 6x0,22 nu făcea fata la amperajul motoarelor de pe braţul robotic. La sursa am folosit codul de culori: albastru-nul, galben cu verde-nul de protecţie, maro-Faza L1. (Fig 3.5)

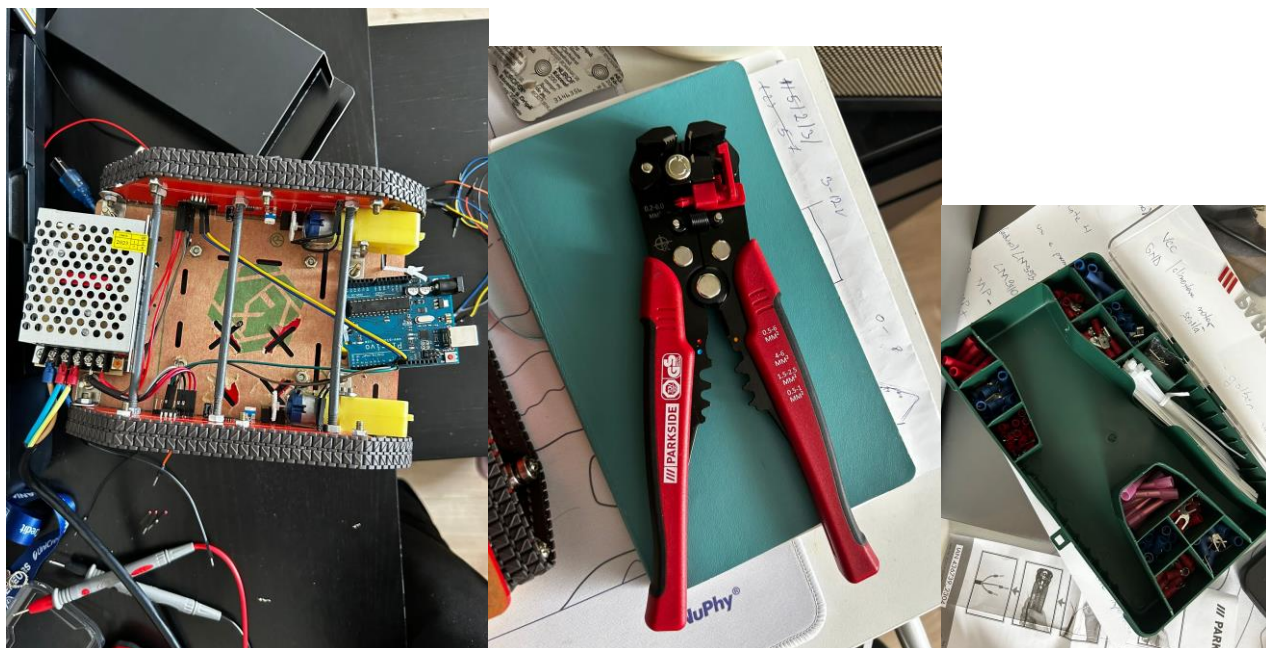


Fig3.4



Fig3.5 Codul de culori

La motoarele de la braț plusul și minusul sunt comune la toate motoarele și mai există un pin de control pentru fiecare servo.

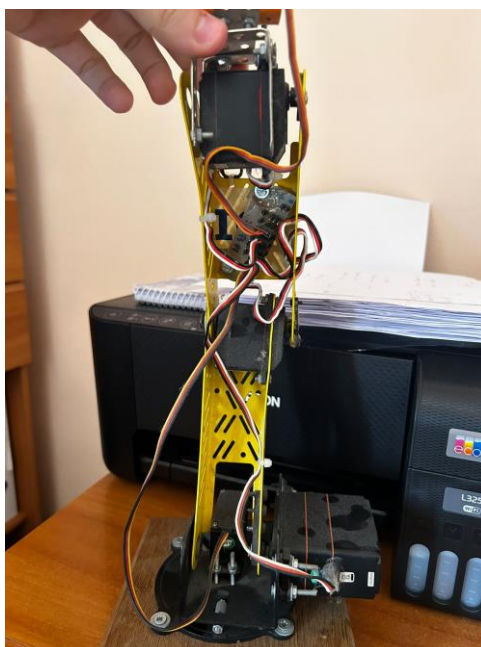


Fig3.6 Control braț



Fig3.7 Mufe alimentare



## 4. Realizarea părții SW

În următoarele figuri se vor putea vedea fragmente din cod

```
#include <Servo.h>
#include <DIYables_IRcontroller.h> // DIYables_IRcontroller library
|
const int IR_RECEIVER = 7;
DIYables_IRcontroller_17 irController(IR_RECEIVER, 200); // debounce time is 200ms

const int A1A = 2; //define pin 2 for A1A
const int A1B = 3; //define pin 3 for A1B

const int B1A = 4; //define pin 8 for B1A
const int B1B = 9; //define pin 9 for B1B

const int PIN_BAZA = 10;
const int PIN_CUPLA1 = 11;
const int PIN_CUPLA2 = 5;
//const int PIN_CUPLA3
const int PIN_PREHENSOR = 6;

const int PIN_LED = 8;

Servo servo_baza;
Servo servo_cupla1;
Servo servo_cupla2;
// //Servo servo_cupla3;
Servo servo_prehensor;

enum motoare {
    DIRECTIE,
    BAZA,
    CUPLA1,
    CUPLA2,
    CUPLA3,
    PREHENSOR,
    NONE
};
```

Fig4.1

```
int pozitie_baza = -60;
int pozitie_cupla1 = 0;
int pozitie_cupla2 = 0;
int pozitie_cupla3 = 0;
int pozitie_prehensor = 0;

motoare motorCurent = NONE;

void setup() {
    motorA('0'); // Turn motor A OFF
    motorB('0'); // Turn motor B OFF
    // put your setup code here, to run once:
    pinMode(IR_RECEIVER, INPUT);
    pinMode(B1A, OUTPUT); // define pin as output
    pinMode(B1B, OUTPUT);

    pinMode(A1A, OUTPUT);
    pinMode(A1B, OUTPUT);

    pinMode(PIN_LED, OUTPUT);
    digitalWrite(PIN_LED, LOW);

    Serial.begin(9600);
    irController.begin();

    // pinMode(PIN_BAZA, LOW);
    // pinMode(PIN_CUPLA1, LOW);
    // pinMode(PIN_CUPLA2, LOW);
    // pinMode(PIN_PREHENSOR, LOW);
```

Fig4.2

```
69  uint8_t status = servo_baza.attach(PIN_BAZA);
70  if (status > 0) {
71      Serial.println("Servo baza attached on channel" + status);
72  } else if (status == 0) {
73      Serial.println("Servo baza was NOT attached. Status = " + status);
74      digitalWrite(PIN_LED, HIGH);
75  }
76  servo_baza.write(pozitie_baza);
77
78  servo_cupla1.attach(PIN_CUPLA1);
79  servo_cupla1.write(pozitie_cupla1);
80
81  servo_cupla2.attach(PIN_CUPLA2);
82  servo_cupla2.write(pozitie_cupla2);
83
84  // // servo_cupla3.attach(PIN_CUPLA3);
85  // // servo_cupla3.write(pozitie_cupla3);
86
87  servo_prehensor.attach(PIN_PREHENSOR);
88  servo_prehensor.write(pozitie_prehensor);
89  }
90
91  void loop() {
92      // put your main code here, to run repeatedly:
93      Key17 key = irController.getKey();
94      if (key != Key17::NONE) {
95          switch (key) {
96              case Key17::KEY_1:
97                  Serial.println("1");
98                  motorCurent = DIRECTIE;
99                  break;
```

Fig4.3

```
126     case Key17::KEY_UP:
127     {
128         Serial.println("UP");
129         switch (motorCurent) {
130             case DIRECTIE:
131             {
132                 motorA('R');
133                 motorB('L');
134                 delay(400);
135                 motorA('O'); // Turn motor A OFF
136                 motorB('O'); // Turn motor B OFF
137                 break;
138             }
139             case CUPLA1:
140             {
141                 pozitie_cupla1 += 15;
142                 servo_cupla1.write(pozitie_cupla1);
143                 break;
144             }
145             case CUPLA2:
146             {
147                 pozitie_cupla2 += 15;
148                 servo_cupla2.write(pozitie_cupla2);
149                 break;
150             }
151         }
152
153         break;
154     }
```

Fig4.4

```
259 void motorA(char d) {  
260     if (d == 'R') {  
261         Serial.println("Motor A - right");  
262         digitalWrite(A1A, LOW);  
263         digitalWrite(A1B, HIGH);  
264     } else if (d == 'L') {  
265         Serial.println("Motor A - left");  
266         digitalWrite(A1A, HIGH);  
267         digitalWrite(A1B, LOW);  
268     } else {  
269         //Robojax.com L9110 Motor Tutorial  
270         // Turn motor OFF  
271         Serial.println("Motor A - off");  
272         digitalWrite(A1A, LOW);  
273         digitalWrite(A1B, LOW);  
274     }  
275 } // motorA end  
276  
277
```

Fig4.5

Am inclus toate librăriile necesare(Servo, senzor infraroşu), am declarat pinii si m am folosit de “enum” pentru a selecta fiecare motor (1,2,3...).Pentru control folosim săgeţile sus, jos, stânga, dreapta. (Fig4.5) este modul de funcţionare(tip tank) al motoarelor de la şasiu.



## 5. Ansamblu final

În imaginile de mai jos, se poate vedea asamblarea finală a proiectului. Am mai adăugat un stabilizator de tensiune pentru a remedia tremuratul motoarelor braţului robotic de la fluctuaţiile sursei, si un led pentru a verifica starea, m-am folosit de panduiţi, suruburi si banda dubla adeziva pentru prindere. Am avut probleme la motorul din partea dreapta a saşii deoarece făcea câmp magnetic cu alimentarea de la arduino si pornea singur. Rotile dinţate de la servo-ul efectorului s au tocit si nu se mai învârtea tot angrenajul.

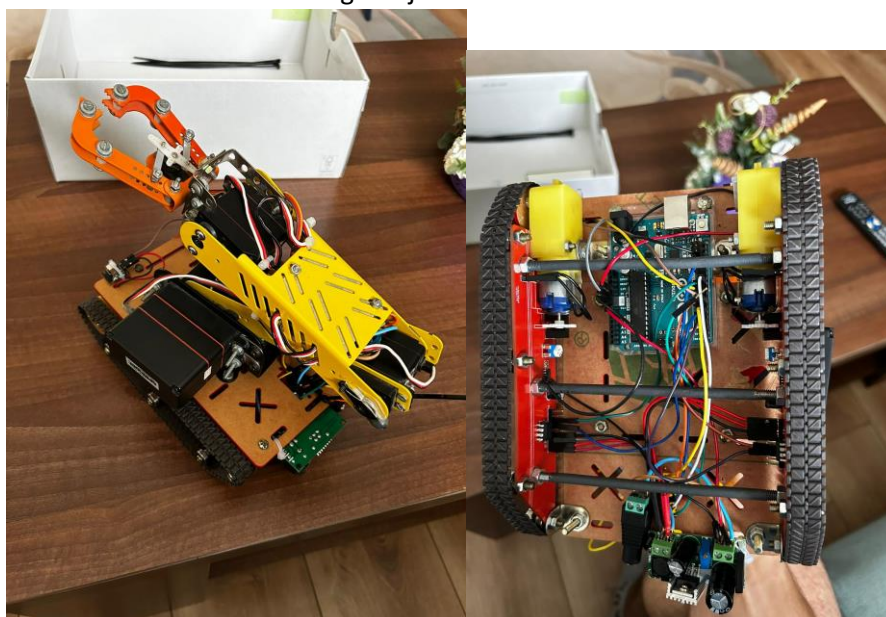


Fig5.1



Fig5.2



## 6. Anexe

1. [How do I use enum? - Using Arduino / Programming Questions - Arduino Forum](#)
2. [Arduino - IR Remote Control | Arduino Tutorial \(arduinogetstarted.com\)](#)
3. [Arduino Code for L9110 Dual DC motor controller module - Robojax](#)
4. [L9110 dc motor control - Using Arduino / Programming Questions - Arduino Forum](#)
5. [Sparkfun's full rotation servo - SpringRC SM-S430R - Hardware / Interfacing - Arduino Forum](#)