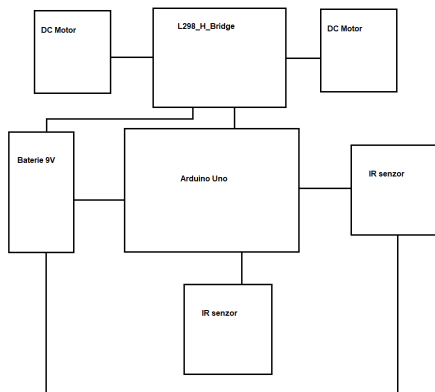


Masina Linie

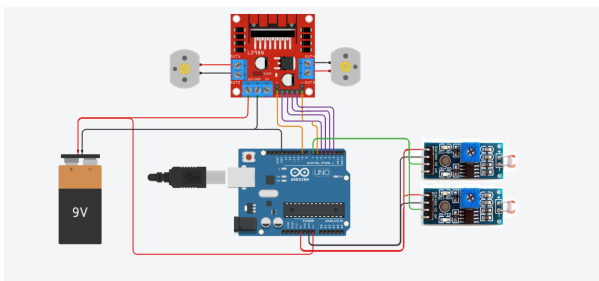
Introducere

Proiectul consta in crearea unei masini care va putea sa urmareasca o linie. Ideea de la care am pornit este ca voiam sa fac un proiect de tip masina, motivatia fiind ca mi se parea interesant sa fac un proiect de tip masina

Descriere generală



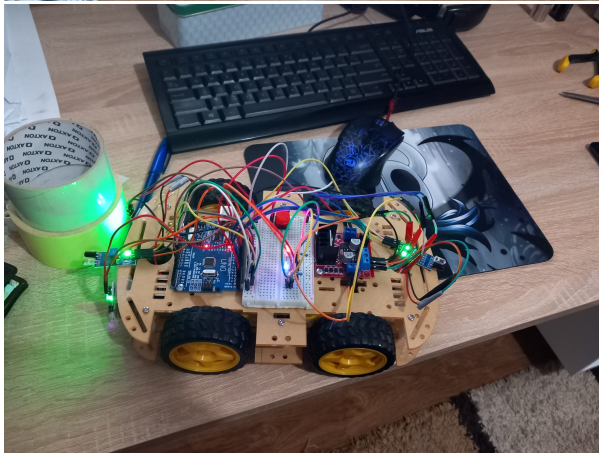
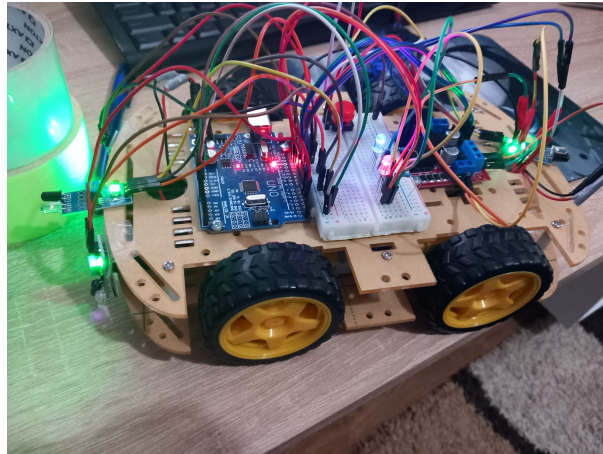
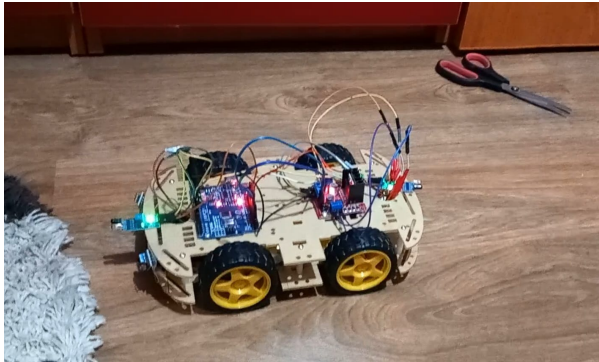
Hardware Design



Componente:

- arduino uno
- 2 x senzori IR
- 2 x motoare DC
- driver motor dc
- baterie 9V

Poze masina:



Software Design

am folosit tinkercad ca sa simulez anumite functii cum ar fii la leduri si buton (intreruperile si timerele)

```
int on = 0;
int time = 0;
int light = 0;
int mode = 0;

ISR(INT0_vect)
{
    on = !on ;
}

ISR(TIMER1_COMPA_vect) {
    // cod intrerupere
    time++;
    if(time == 2)
    {
        if(mode == 0)
        {
            light+=50;
        }
    }
}
```

```
}
else
{
    light-=50;
}
if(light <= -1)
{
    light = 0;
    mode = 0;
}
if(light >= 250)
{
    light = 249;
    mode = 1;
}
analogWrite(12,light);
analogWrite(11,250-light);
time = 0;
}
//Serial.println(x);
}

void configure_timer1() {
    // exemplu de configurare pentru Timer 1 în mod CTC
    // care va genera întreruperi cu frecvența de 2Hz
    TCCR1A = 0;
    TCCR1B = 0;
    TCNT1 = 0;
    OCR1A = 31249;           // compare match register 16MHz/256/2Hz-1
    TCCR1B |= (1 << WGM12); // CTC mode
    TCCR1B |= (1 << CS12);  // 256 prescaler
}

void init_timer1() {
    TIMSK1 |= (1 << OCIE1A); // enable timer compare interrupt
}

void setup_interrups() {

    cli();

    // input
    DDRD &= ~(1 << PD2);
    // input pullup
    PORTD |= (1 << PD2);
    EICRA |= (1 << ISC01);

    EIMSK |= (1 << INT0);
    // activare intreruperi

    sei();
}
```

```
}

int motor11 = PD3;//3;
int motor12 = PD4;//4;
int motor21 = PD5;//5;
int motor22 = PD6;//6;
int senzor1 = PD7;//7;
int senzor2 = PB0;//8;
int senzor3 = PB1;//9;
int senzor4 = PB2;//10;
int countdown = 0;
void moveForward()
{
    PORTD |= (1 << motor11); //digitalWrite(motor11,HIGH);
    PORTD &= ~(1 << motor12); //digitalWrite(motor12,LOW);
    PORTD |= (1 << motor21); //digitalWrite(motor21,HIGH);
    PORTD &= ~(1 << motor22); //digitalWrite(motor22,LOW);
}

void moveLeft()
{
    PORTD |= (1 << motor11); //digitalWrite(motor11,HIGH);
    PORTD &= ~(1 << motor12); //digitalWrite(motor12,LOW);
    PORTD &= ~(1 << motor21); //digitalWrite(motor21,LOW);
    PORTD |= (1 << motor22); //digitalWrite(motor22,HIGH);
}

void moveRight()
{
    PORTD &= ~(1 << motor11); //digitalWrite(motor11,LOW);
    PORTD |= (1 << motor12); //digitalWrite(motor12,HIGH);
    PORTD |= (1 << motor21); //digitalWrite(motor21,HIGH);
    PORTD &= ~(1 << motor22); //digitalWrite(motor22,LOW);
}

void Stop()
{
    PORTD &= ~(1 << motor11); //digitalWrite(motor11,LOW);
    PORTD &= ~(1 << motor12); //digitalWrite(motor12,LOW);
    PORTD &= ~(1 << motor21); //digitalWrite(motor21,LOW);
    PORTD &= ~(1 << motor22); //digitalWrite(motor22,LOW);
}

void change_side()
{
    int aux = motor11;
    motor11 = motor12;
    motor12 = aux;
    aux = motor21;
    motor21 = motor22;
    motor22 = aux;
}
```

```
}

void setup() {
    // put your setup code here, to run once:
    DDRD |= (1 << motor11); //pinMode(motor11,OUTPUT);
    DDRD |= (1 << motor12); //pinMode(motor12,OUTPUT);
    DDRD |= (1 << motor21); //pinMode(motor21,OUTPUT);
    DDRD |= (1 << motor22); //pinMode(motor22,OUTPUT);
    DDRD &= ~(1 << senzor1); //pinMode(senzor1,INPUT);
    DDRB &= ~(1 << senzor2); //pinMode(senzor2,INPUT);
    DDRB &= ~(1 << senzor3); //pinMode(senzor3,INPUT);
    DDRB &= ~(1 << senzor4); //pinMode(senzor4,INPUT);
    change_side();

    pinMode(12, OUTPUT);
    pinMode(11, OUTPUT);

    configure_timer1();
    init_timer1();
    setup_interrupts();
    on = 0;
}

void loop() {
    // put your main code here, to run repeatedly:
    while(on == 0)
    {
        Stop();
        delay(2000);
    }

    int x = PINB & (1 << senzor3); //digitalRead(senzor3);
    int y = PINB & (1 << senzor4); //digitalRead(senzor4);

    if((x == 0 || y == 0) && countdown == 0)
    {
        countdown = 5000;

        if(y == 0)
        {
            change_side();
            Stop();
            delay(2000);
            int l = PINB & (1 << senzor2); //digitalRead(senzor2);
            while(x != 0 || l == 0)
            {
                moveRight();
                x = PINB & (1 << senzor3); //digitalRead(senzor3);
                l = PIND & (1 << senzor1); //digitalRead(senzor1);
            }
            Stop();
        }
    }
}
```

```
        change_side();
        delay(3000);
    }
}

if(countdown != 0)
{
    countdown--;
}

int r = PINB&(1 << senzor2);//digitalRead(senzor2);
int l = PIND&(1 << senzor1);//digitalRead(senzor1);

if( l != 0)
{
    moveRight();
}
else if(r != 0)
{
    moveLeft();
}
else
{
    moveForward();
}
}
```

Rezultate Obținute

masina reuseste sa urmeze o linie, sa se invarte cand ajunge la capat de linie si sa o ia inapoi pe unde a venit

<https://www.youtube.com/watch?v=LBZnS3yhsbY>

Concluzii

Am reusit sa implementez tot ce voiam sa implementez si proiectul a iesit asa cum imi planuiam. Unde nu a iesit cum ma asteptam este la timp si cost unde in ambele cazuri au fost mai mari decat ma asteptasem. Am avut probleme atat cu hardware-ul (piese stricate) cat si cu softwarul (bugs). Am consumat vre-o 3 baterii de 9V pentru a testa si a face probe la masina. All in all a fost fun si destul de diferit de temele/proiectele obisnuite din facultate

Download

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

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele 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

18.05.2023: finalizare motoare + caroseria masina

21.05.2023: finalizare functionalitate de urmarire linie

23.05.2023: finalizare hardware

Bibliografie/Resurse

https://ardushop.ro/ro/home/88-modul-senzor-lumina-intensitate-luminoasa.html?search_query=senzor+ir&results=175

https://ardushop.ro/ro/electronica/84-l298n-punte-h-dubla-dual-h-bridge-motor-dcsteppe.html?search_query=DC+200RPM+3-6v+BO+&results=56

<https://projecthub.arduino.cc/lightthedreams/34b1d3f5-b31e-46fe-b57a-156fbf440cc1>

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/alexau/mihai.ilinca>

Last update: **2023/05/28 15:15**

