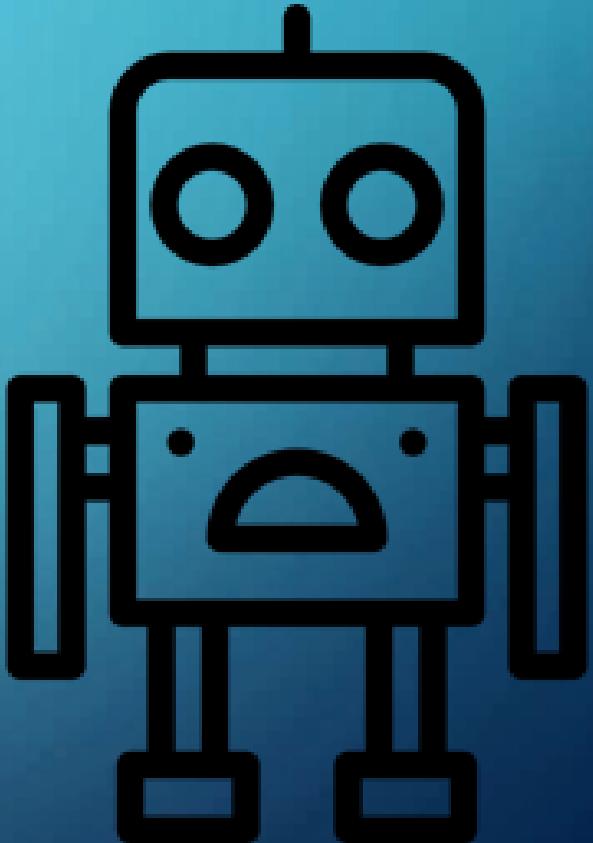




HELLO, WORLD!

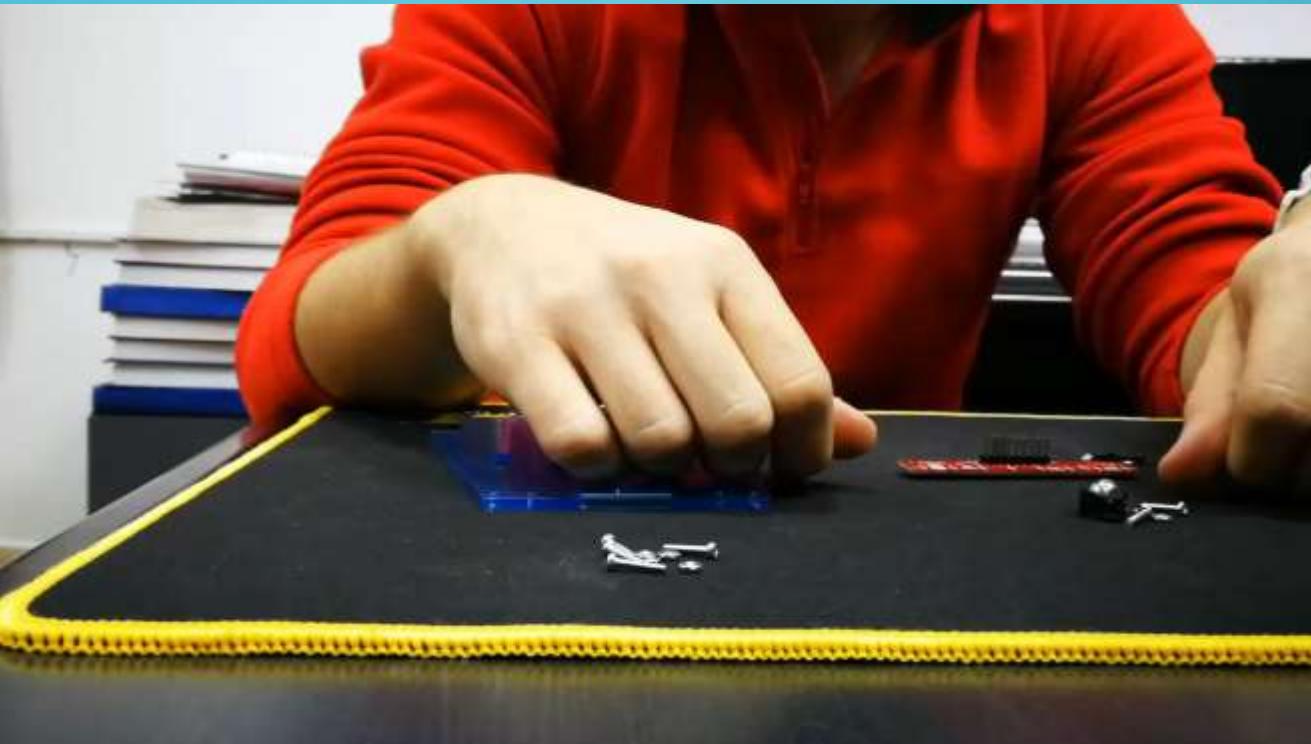
- 
1. CUM SE CONSTRUIESTE ROBOTUL?
  2. CUM SE PROGRAMEAZA PROBOTUL?
  3. EXEMPLE



# COMPONENTE(ROBOT);

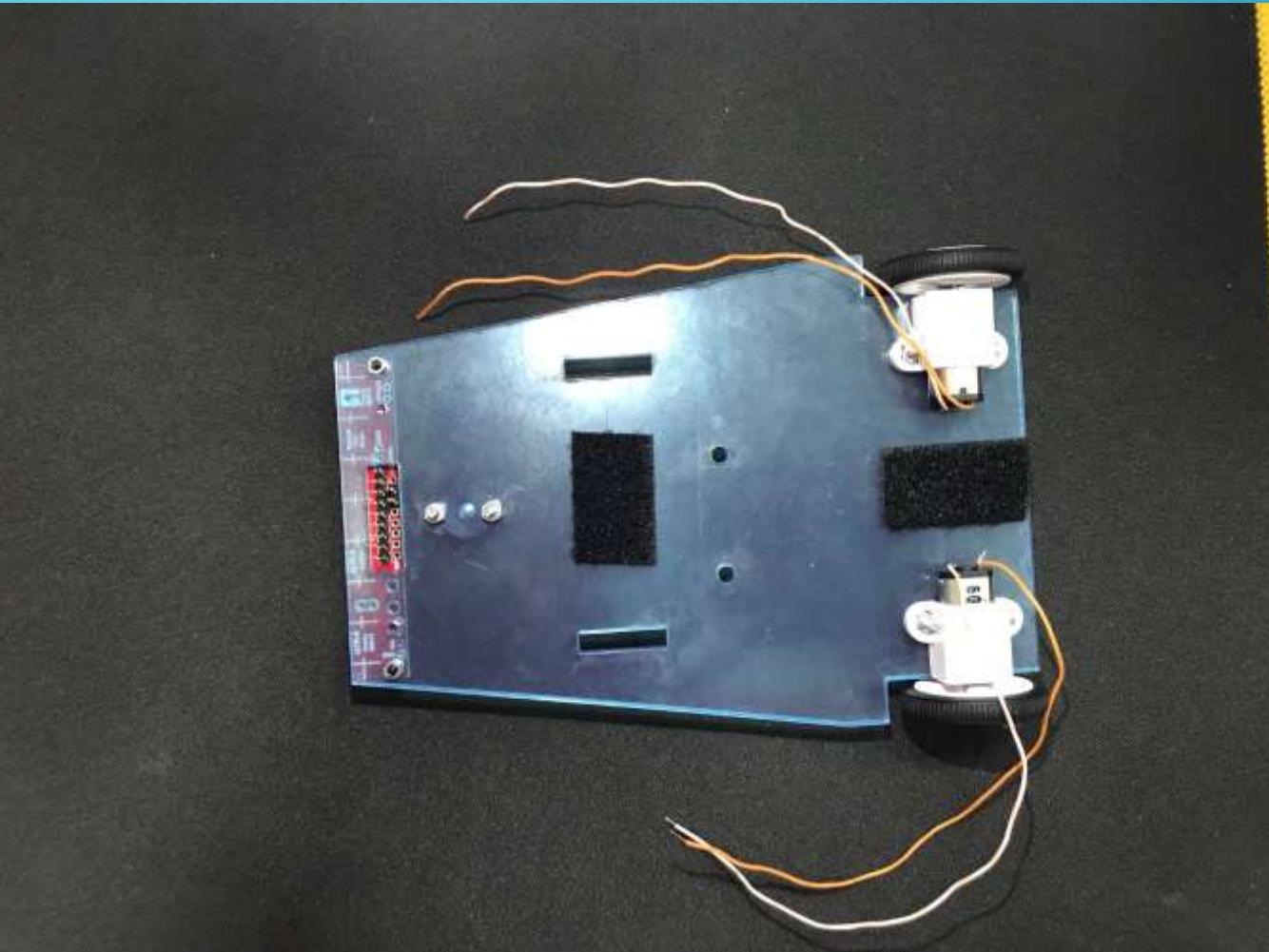
- 1 × sasiu
- 2 × motor cu reductie (1000 rotatii pe secunda)
- 2 × suport motor
- 2 × roata
- 1 × senzor infrarosu
- 2 × senzori de distanta
- 1 × Arduino Uno
- 1 × driver motoare (L298)
- 1 × suport baterii
- 1 × bila control robot
- 1 × fir: mama – mama, tata – tata, mama – tata

# CONSTRUIESTE(ROBOT);

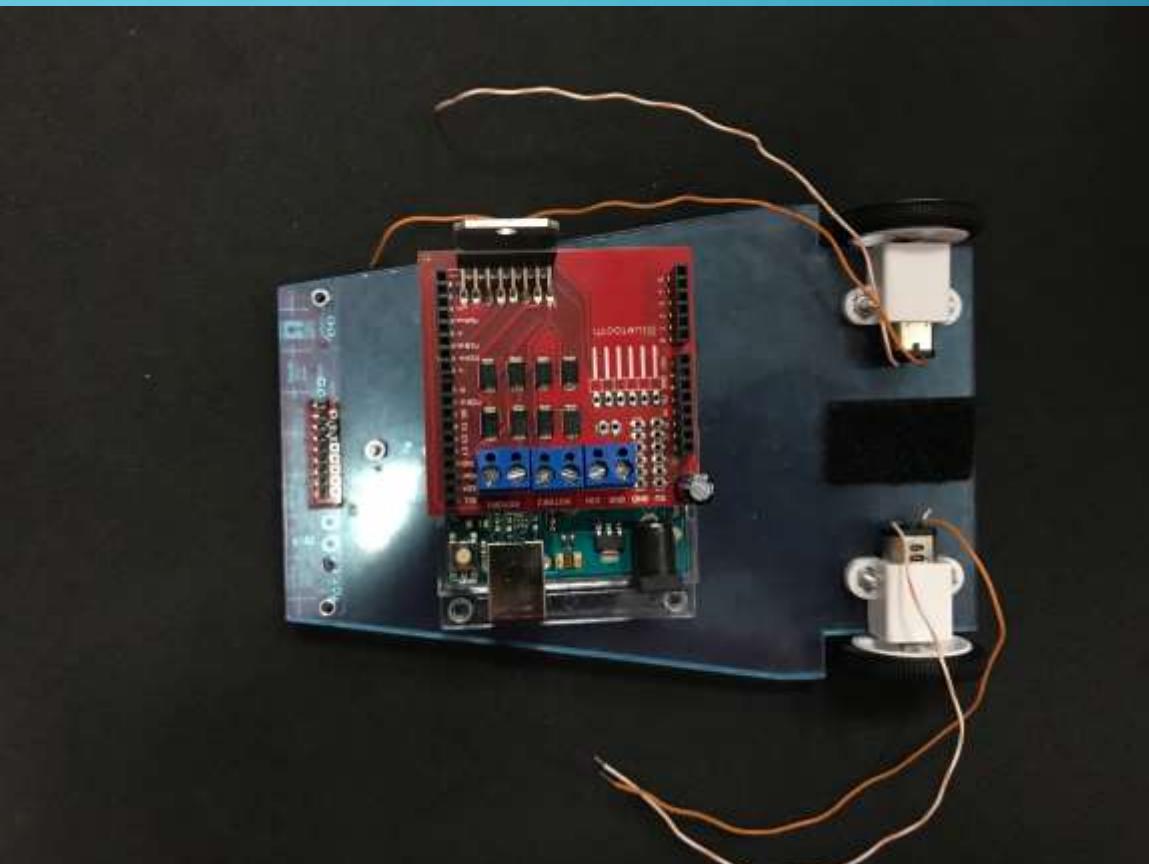
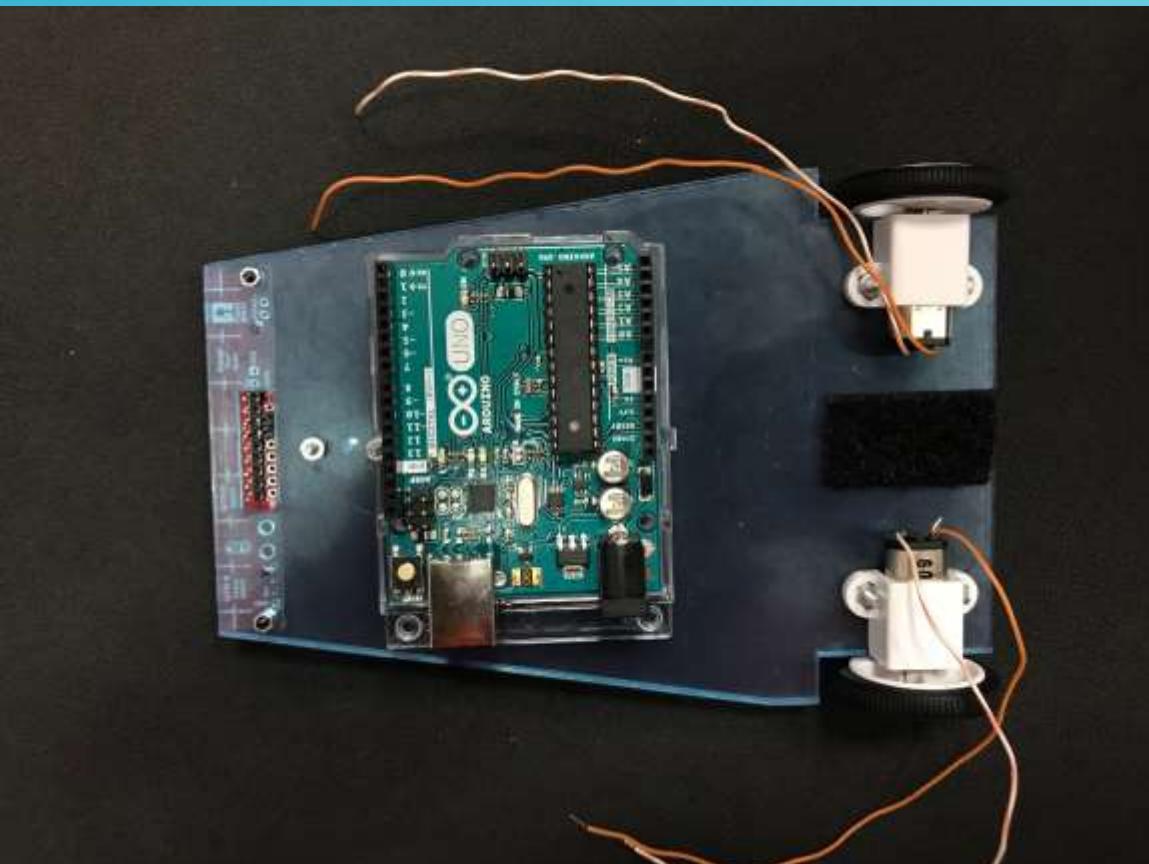


## URMATOR(PAS1);

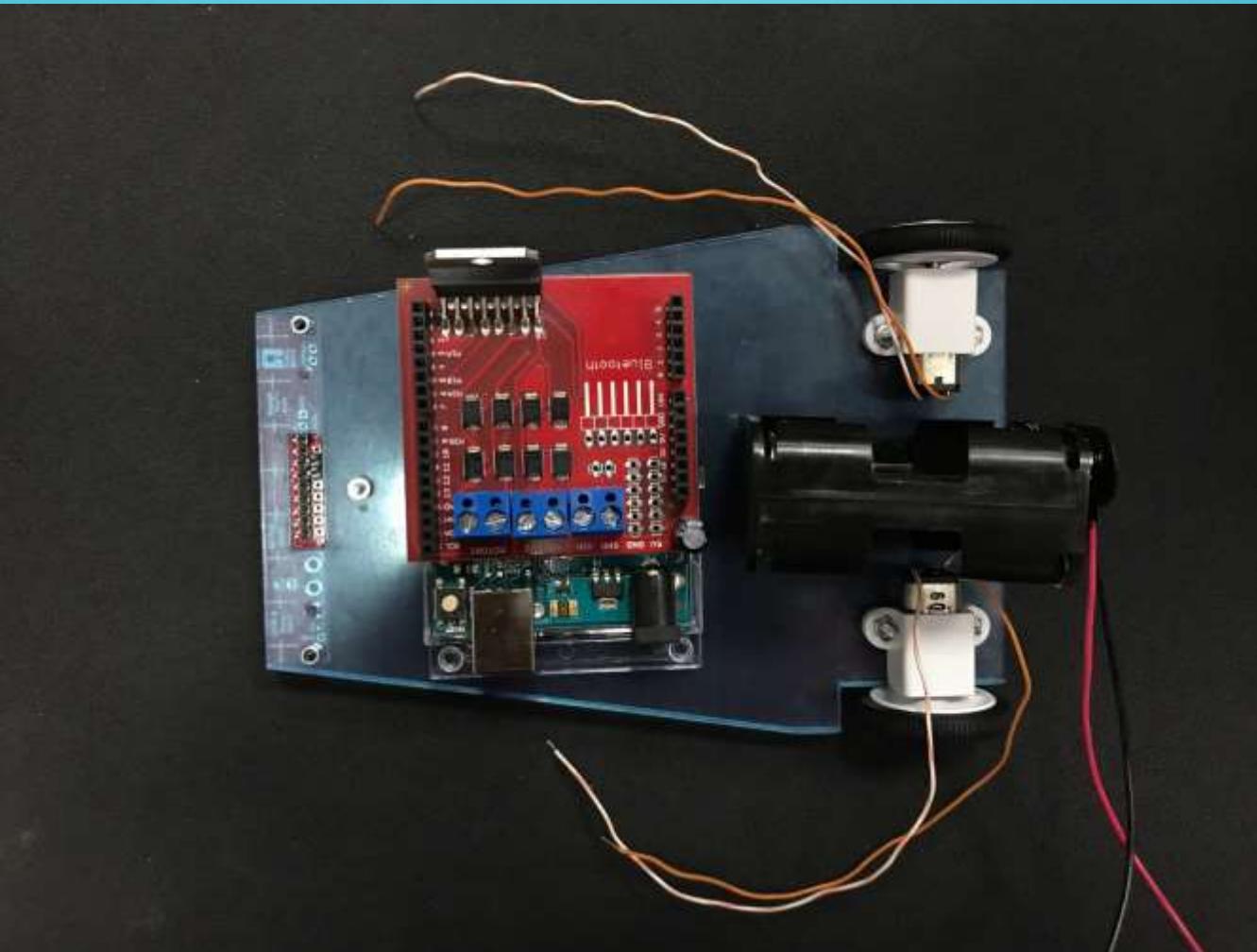
- Lipituri:
  - Pinii de + si - ai motoarelor
  - Pinii senzorului (de lipit inainte de a pune senzorul pe sasiu)



# URMATOR(PAS2);

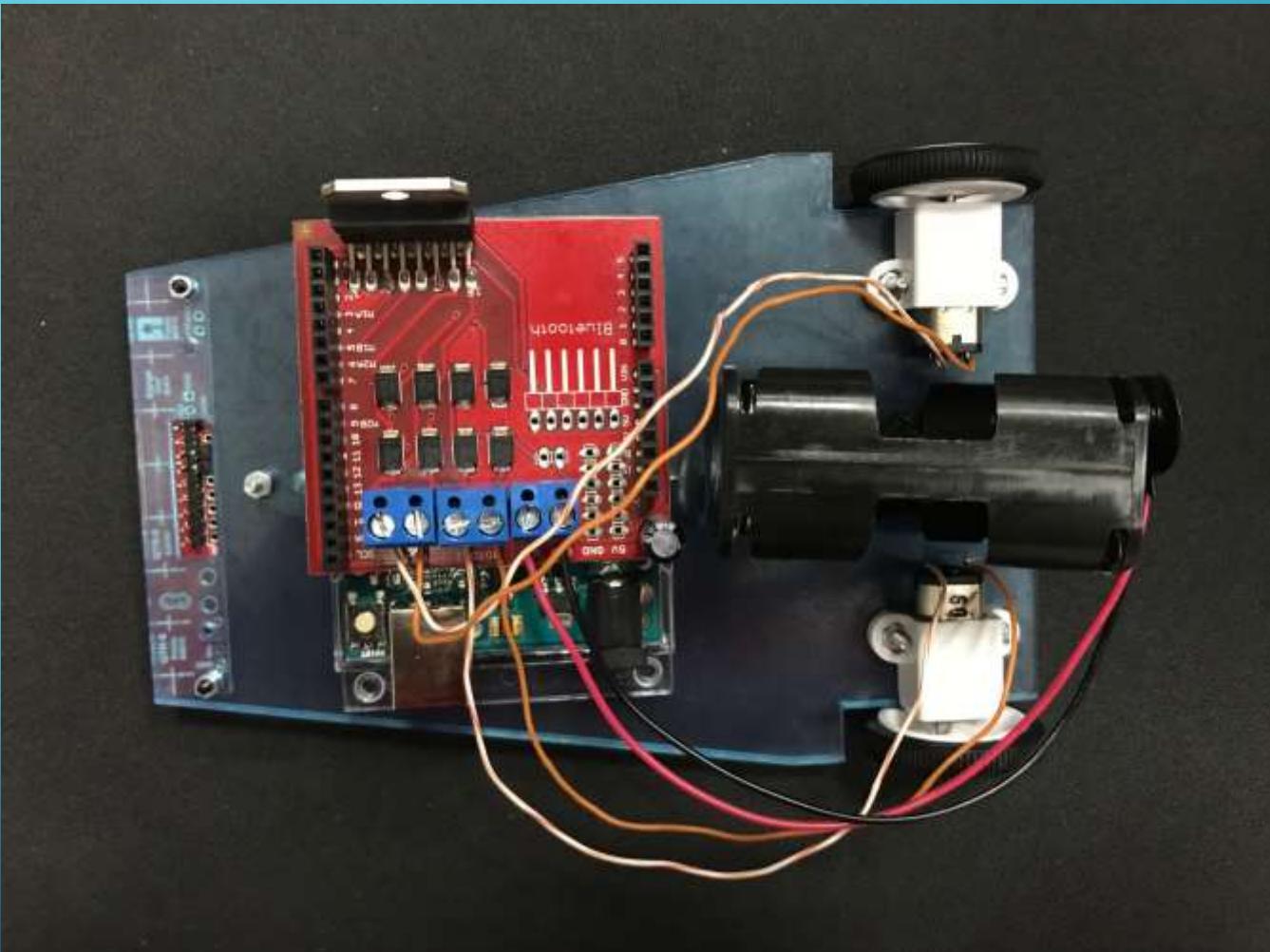


# URMATOR(PAS3);



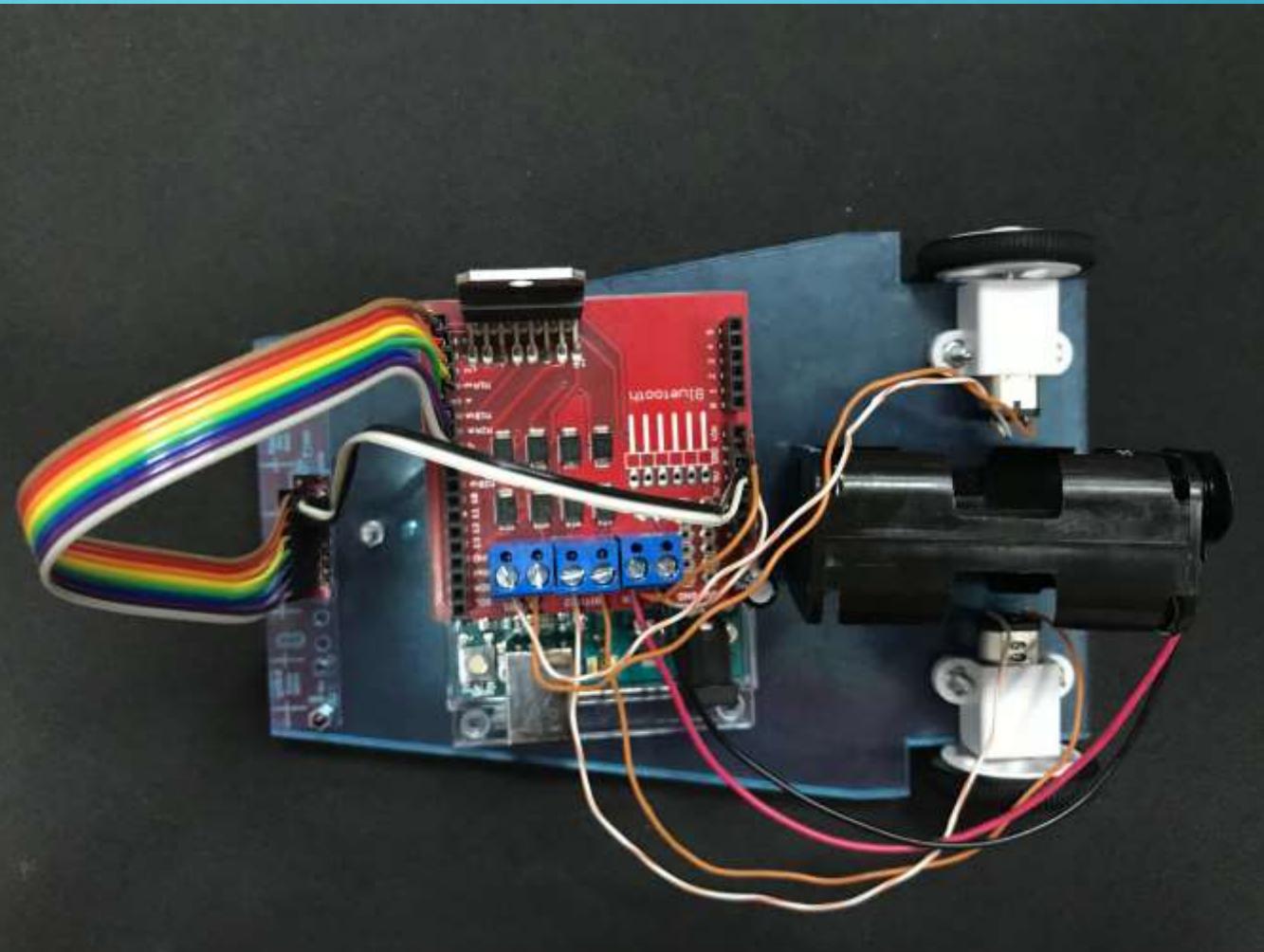
## URMATOR(PAS4);

- Se conecteaza:
  - Pinii de + si - de la fiecare motor in soclul asociat (MOTOR1 si MOTOR2)
  - Firul negru (GND) al bateriilor se conecteaza la intrarea GND a L298
  - Firul rosu (VCC) al bateriilor se conecteaza la intrarea VIN a L298



# URMATOR(PAS5);

- Se conecteaza:
  - De la soclul de alimentare a L298 (GND si VIN) la pinii GND si VIN a L298 (pentru a alimenta microcontroller-ul Arduino)
  - VCC si GND de la senzor la 5V si GND a L298 (firul alb si firul negru din imagine)
  - Pinii senzorului la pinii digitali/analogici a L298 (in imagine sunt conectati la pini digitali)
  - DE RETINUT! Exceptie fac pinii digitali 3, 5, 6 si 9 deoarece sunt folositi implicit la controlul motoarelor



# ARDUINO( IDE);

The image shows the Arduino IDE interface. The title bar reads "sketch\_nov25a | Arduino 1.8.5". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for save, upload, and other functions. The main area contains the following code:

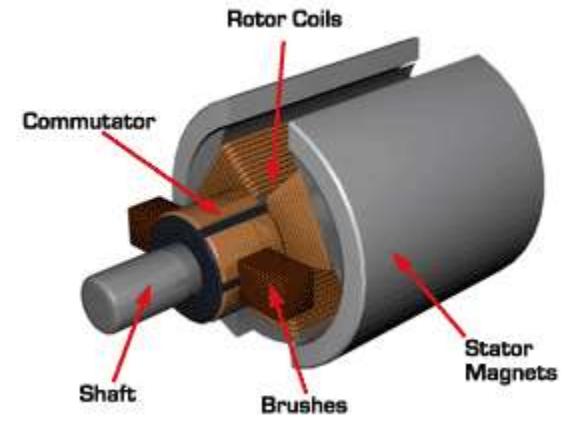
```
// zona 1
void setup() {
    // zona 2
}

void loop() {
    // zona 3
}
```

The status bar at the bottom indicates "Arduino/Genuino Uno on COM12".

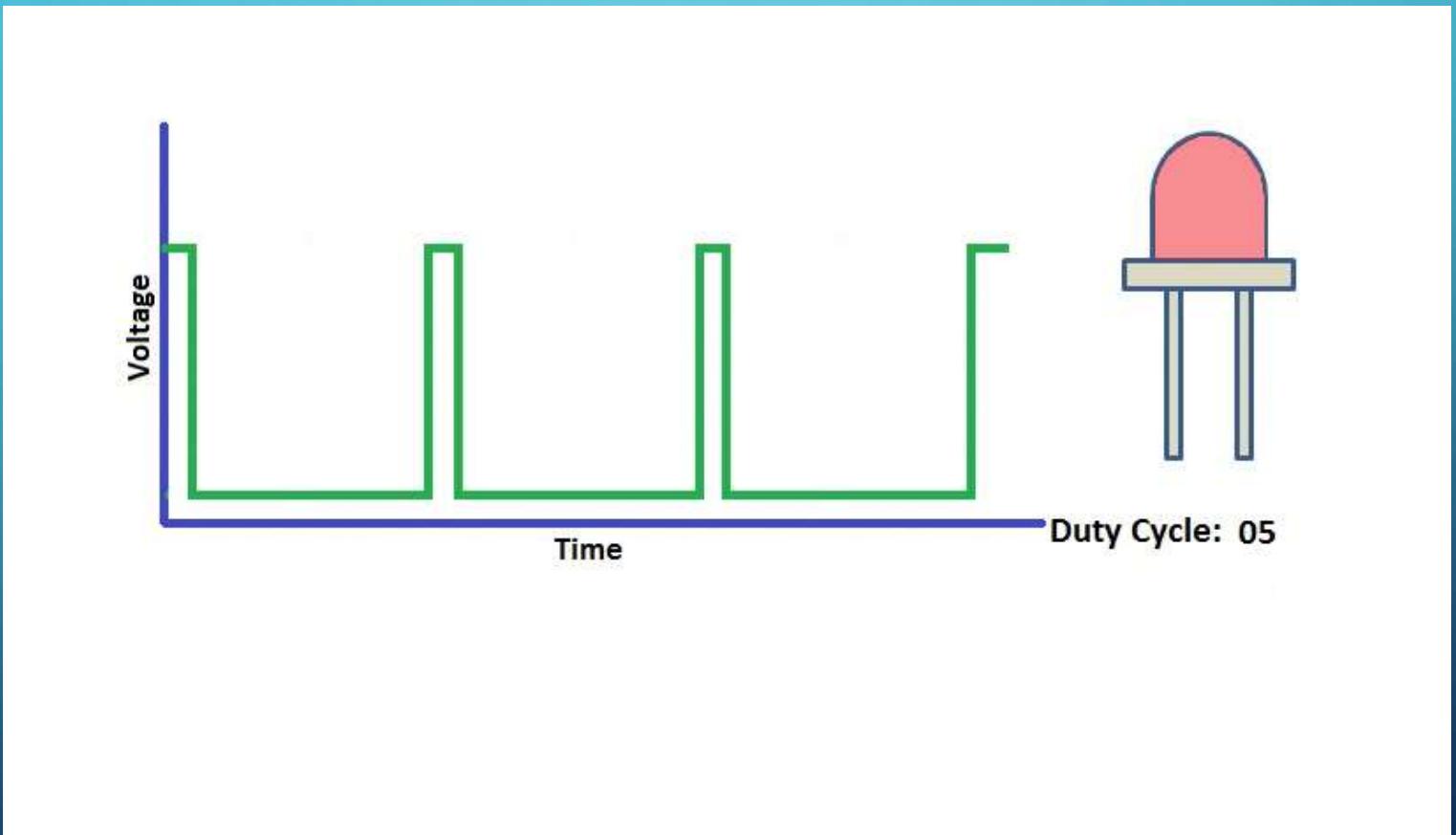
- Zona 1:
  - Declararea variabilelor globale si constantelor care corespund pinilor:
    - int analogInput = A0; //corespunzator pinului A0
    - int digitalOutput = 0; //corespunzator pinului D0
    - #define digitalOutput 0; //corespunzator pinului D0
  - Declararea si initializarea variabilelor globale si constantelor care ne pot ajuta la dezvoltarea programului
- Zona 2:
  - Structura de program care se executa o singura data
  - Initializarea tipului variabilelor:
    - INPUT: pinMode(analogInput, INPUT);
    - OUTPUT: pinMode(digitalOutput, OUTPUT);
  - Serial.begin(9600); //pentru transmitere seriala (folosit pentru debugging)
- Zona 3:
  - Structura de program care se repeta la infinit
  - Programul care va rula pe microcontroller-ul Arduino

# PUTERE(MOTOR);



- Pinii impliciti pentru controlul motoarelor: 3, 5, 6, 9
- Puterea motorului (viteza) este cuprinsa intre -255 si 255 (PWM)
- Rularea inainte a fiecarui motor:
  - Primul pin al motorului primeste valoarea vitezei dorite
  - Al doilea pin al motorului primeste valoarea 0
- Rularea inapoi a fiecarui motor:
  - Primul pin al motorului primeste valoarea 0
  - Al doilea pin al motorului primeste valoarea vitezei dorite
- Se foloseste metoda `analogWrite(pin, viteza)` pentru comanda motoarelor:
  - `analogWrite(motor1_pin1, 255);`
  - `analogWrite(motor1_pin2, 0);`

# PWM;



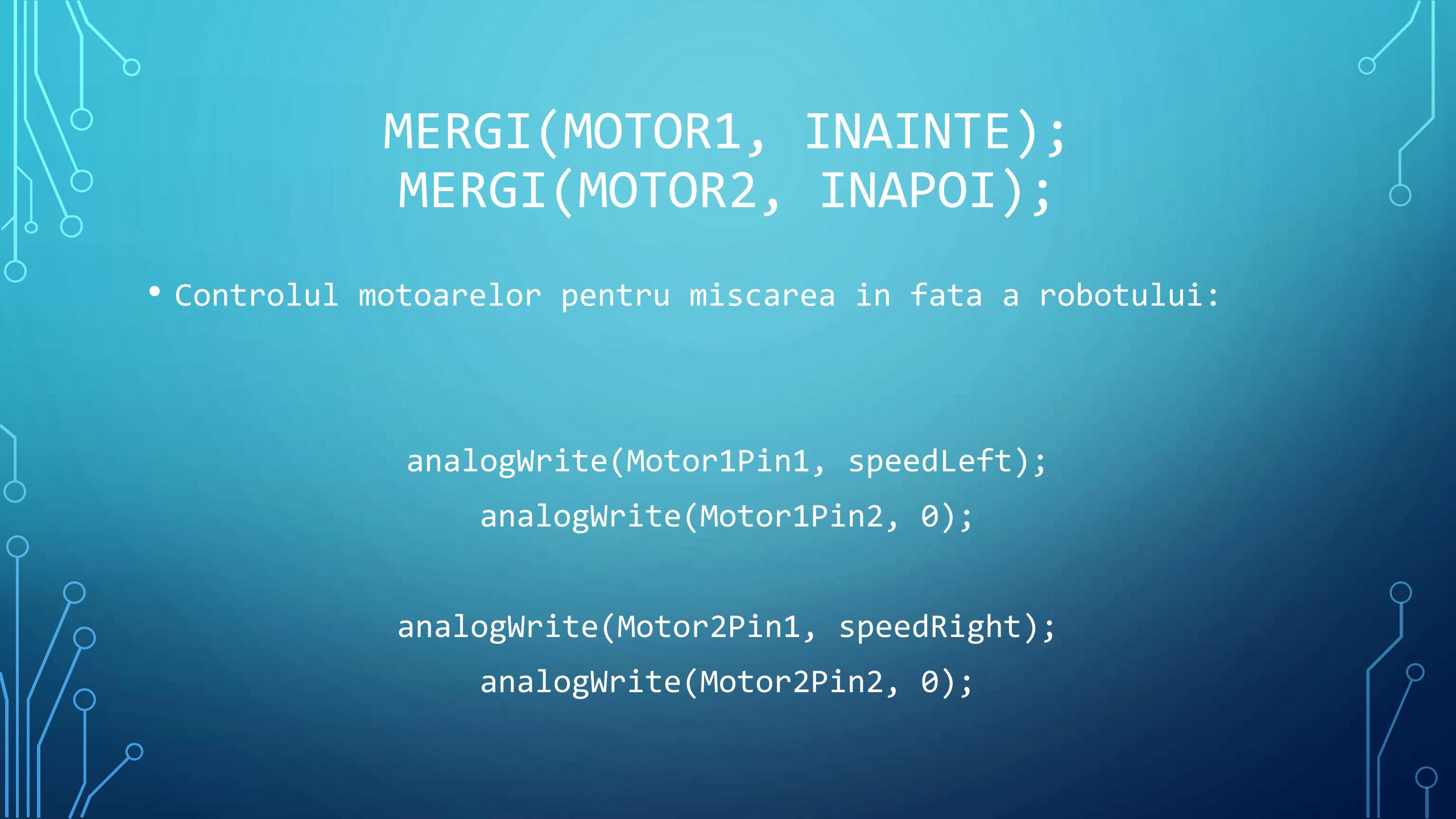
```
int Motor2Pin1 = 3;
int Motor2Pin2 = 5;
int Motor1Pin1 = 6;
int Motor1Pin2 = 9;

void setup()
{
    pinMode(Motor1Pin1, OUTPUT);
    pinMode(Motor1Pin2, OUTPUT);
    pinMode(Motor2Pin1, OUTPUT);
    pinMode(Motor2Pin2, OUTPUT);
}

void loop()
{
    go(150, 150);
}
```

```
void go(int speedLeft, int speedRight)
{
    if (speedLeft > 0)
    {
        analogWrite(Motor1Pin1, speedLeft);
        analogWrite(Motor1Pin2, 0);
    }
    else
    {
        analogWrite(Motor1Pin1, 0);
        analogWrite(Motor1Pin2, -speedLeft);
    }

    if (speedRight > 0)
    {
        analogWrite(Motor2Pin1, speedRight);
        analogWrite(Motor2Pin2, 0);
    }
    else
    {
        analogWrite(Motor2Pin1, 0);
        analogWrite(Motor2Pin2, -speedRight);
    }
}
```



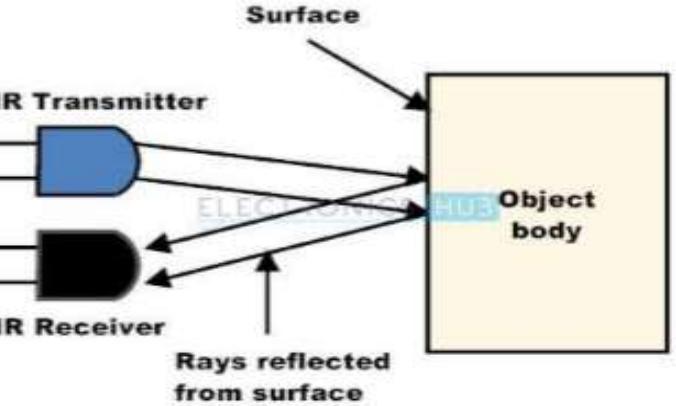
```
MERGI(MOTOR1, INAINTE);  
MERGI(MOTOR2, INAPOI);
```

- Controlul motoarelor pentru miscarea in fata a robotului:

```
analogWrite(Motor1Pin1, speedLeft);  
analogWrite(Motor1Pin2, 0);
```

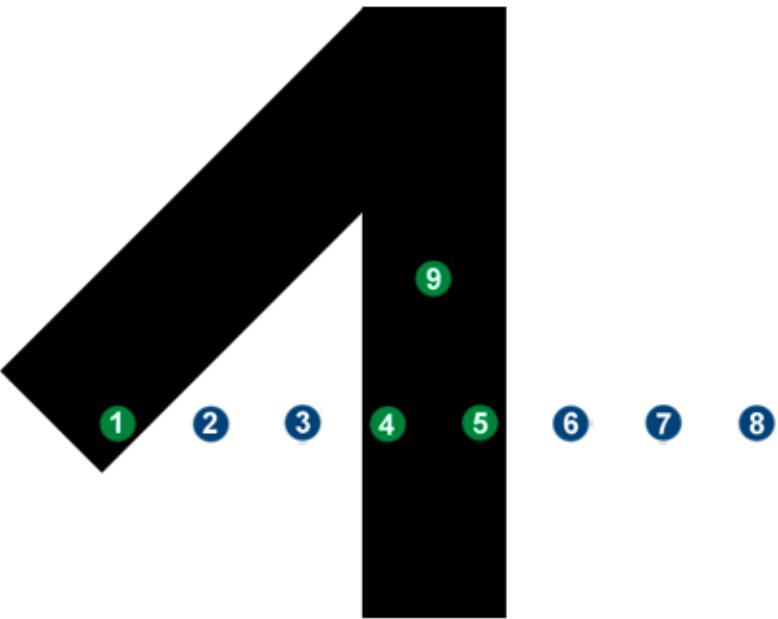
```
analogWrite(Motor2Pin1, speedRight);  
analogWrite(Motor2Pin2, 0);
```

# SENZOR(LINIE);

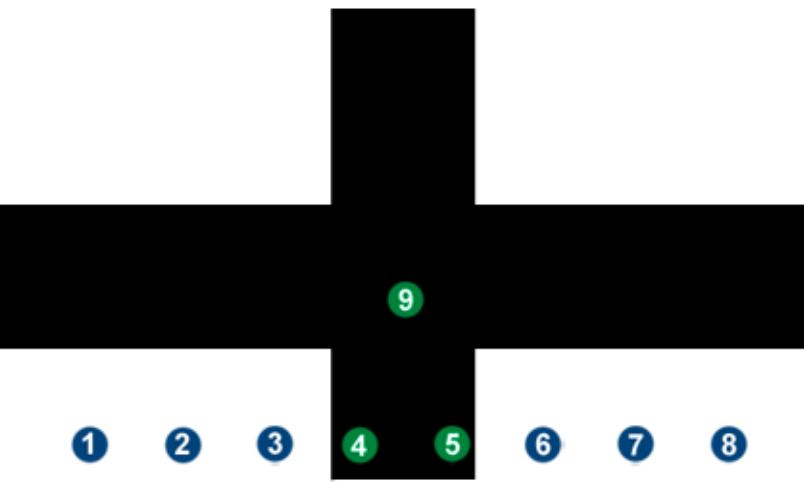


- Senzorul contine 8 pini, fiecare fiind conectat la senzorul de culoare asociat
- Senzorul de linie este compus dintr-un led infraroșu și o fotodiode
- Lumina infraroșie emisă de led loveste suprafața și este reflecată înapoi către fotodioda care scoate la ieșire o tensiune proporțională cu reflexia luminii (pentru suprafete luminoase vom avea o tensiune mare, iar pentru suprafete de culoare închisă o tensiune mică)
- Se pot conecta la pini analogici și returnează un rezultat pozitiv
- Se pot conecta la pini digitali și returnează 0 sau 1

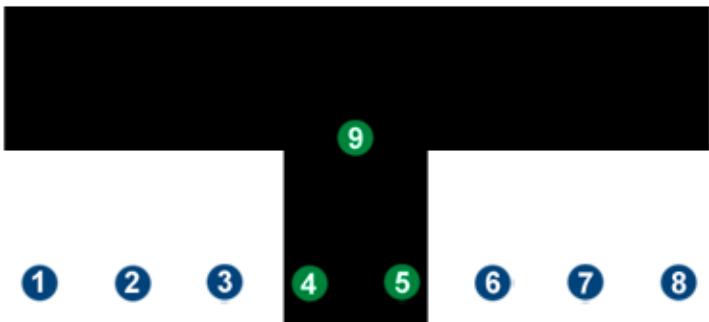
# EXEMPLU(FUNCTIONARE1);



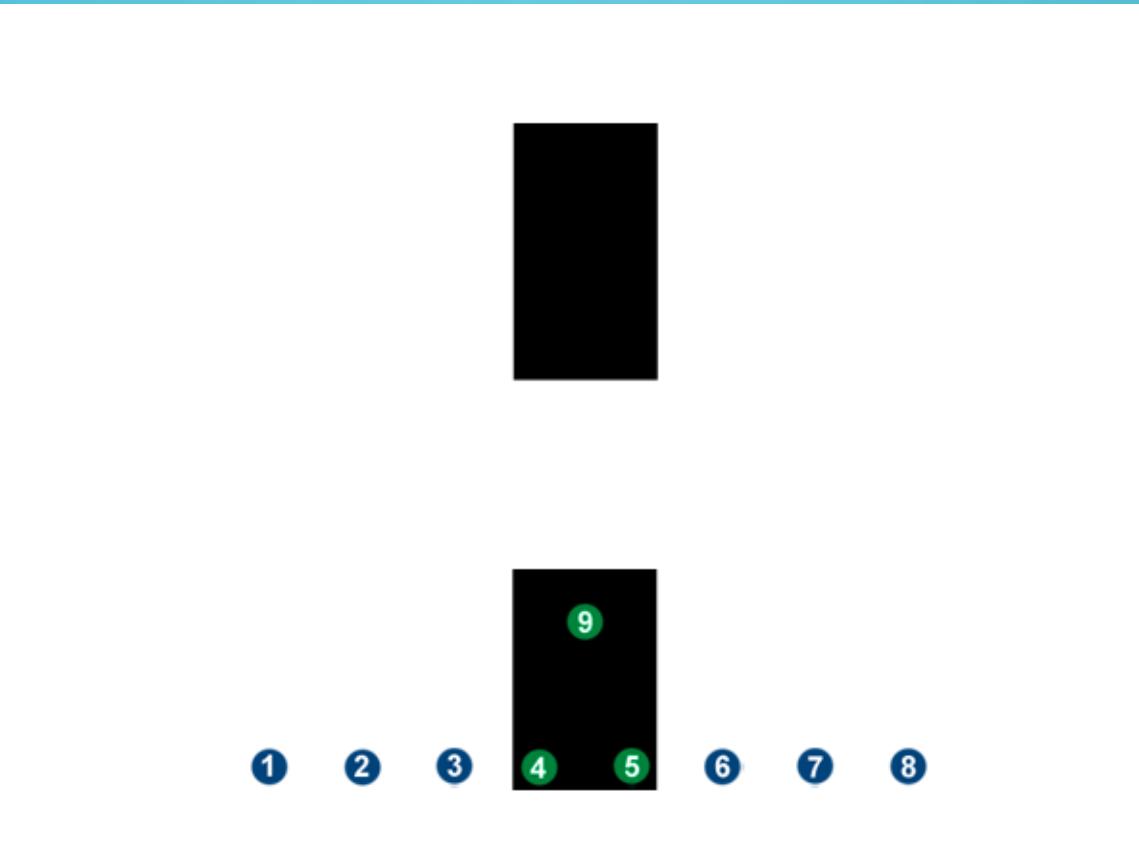
# EXEMPLU(FUNCTIONARE2);



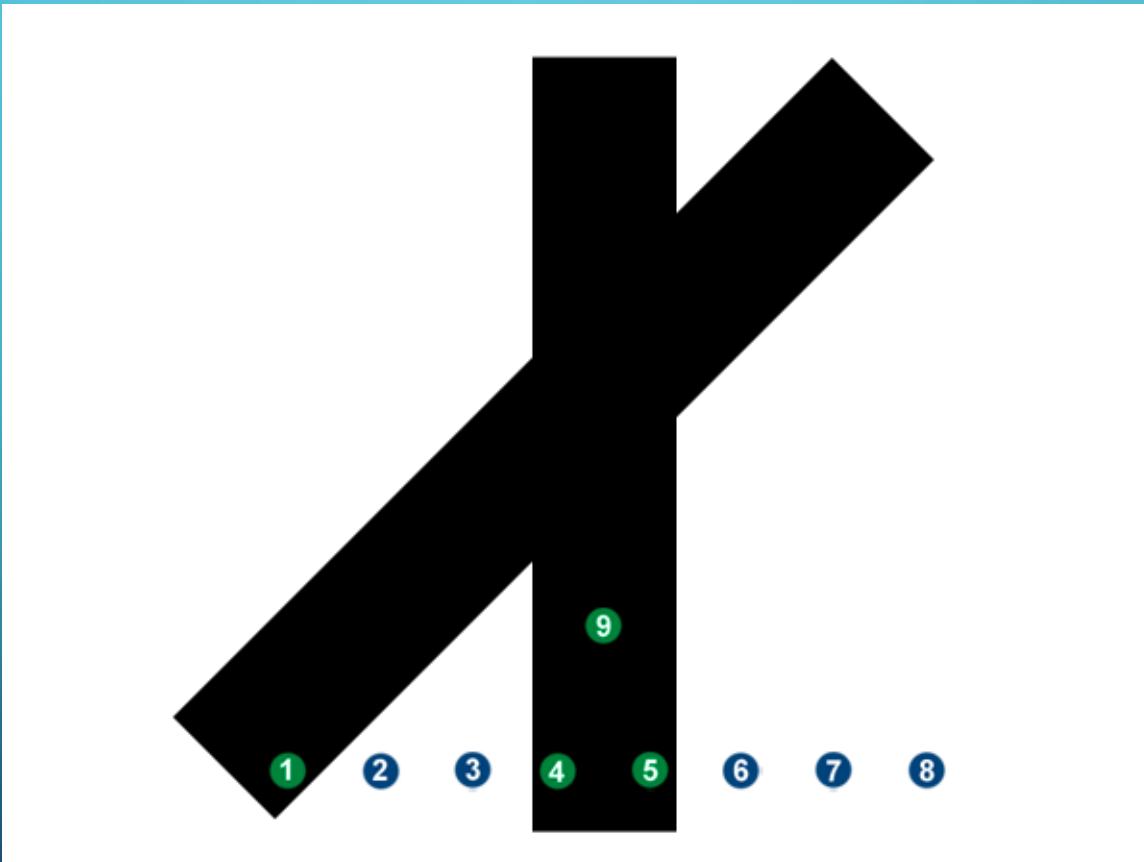
# EXEMPLU(FUNCTIONARE3);



# EXEMPLU(FUNCTIONARE4);



# EXEMPLU(FUNCTIONARE5);



```
int In1 = A0;
int In2 = A1;
int In3 = A2;
int In4 = A3;
int In5 = A4;
int In6 = A5;

int In1Signal;
int In2Signal;
int In3Signal;
int In4Signal;
int In5Signal;
int In6Signal;

void setup()
{
    Serial.begin(9600);

    pinMode(In1, INPUT);
    pinMode(In2, INPUT);
    pinMode(In3, INPUT);
    pinMode(In4, INPUT);
    pinMode(In5, INPUT);
    pinMode(In6, INPUT);
}
```

```
void loop()
{
    In1Signal = analogRead(In1);
    In2Signal = analogRead(In2);
    In3Signal = analogRead(In3);
    In4Signal = analogRead(In4);
    In5Signal = analogRead(In5);
    In6Signal = analogRead(In6);

    Serial.print("A0:");
    Serial.print(In1Signal);
    Serial.print(" A1:");
    Serial.print(In2Signal);
    Serial.print(" A2:");
    Serial.print(In3Signal);
    Serial.print(" A3:");
    Serial.print(In4Signal);
    Serial.print(" A4:");
    Serial.print(In5Signal);
    Serial.print(" A5:");
    Serial.println(In6Signal);
}
```



## SENZOR(ULTRASONIC);

- Senzorul contine 4 pini: VCC, GND, ECHO, TRIG
- ECHO si TRIG se conecteaza la pini digitali
- TRIG declanseaza un impuls, loveste cel mai apropiat obiect, iar ECHO receptioneaza timpul in microsecunde

```
int Trig = 9;
int Echo = 10;

long duration;
int distance;

void setup()
{
    Serial.begin(9600);

    pinMode(Trig, OUTPUT);
    pinMode(Echo, INPUT);
}

void loop()
{
    digitalWrite(Trig, LOW);
    delayMicroseconds(2);

    digitalWrite(Trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(Trig, LOW);

    duration = pulseIn(Echo, HIGH);

    distance = duration * 0.034 / 2;

    Serial.print("Distance: ");
    Serial.println(distance);
}
```





