

ROBOFUN.RO
LECȚIA XXIV

CURS GRATUIT

ARDUINO ȘI ROBOTICĂ

**Senzori Flex Brick
cu Arduino**

**Proiect :
Deget Bionic -
Senzor Flex + Servomotor**

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs
CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

Arduino UNO, senzor de indoire si servomotor

Senzorii de indoire sau senzorii flex au proprietatea de a isi modifica rezistenta atunci cand sunt indoiti. Cu cat gradul de indoire este mai mare, cu atat rezistenta creste mai mult. Senzorii sunt utilizati in manusi pentru a simti miscarile degetelor, exemplu fiind manusa Nintendo Power Glove.

Senzorul poate fi folosit si in alte scopuri, cum ar fi, sa controlezi un servomotor in functie de gradul de indoire.

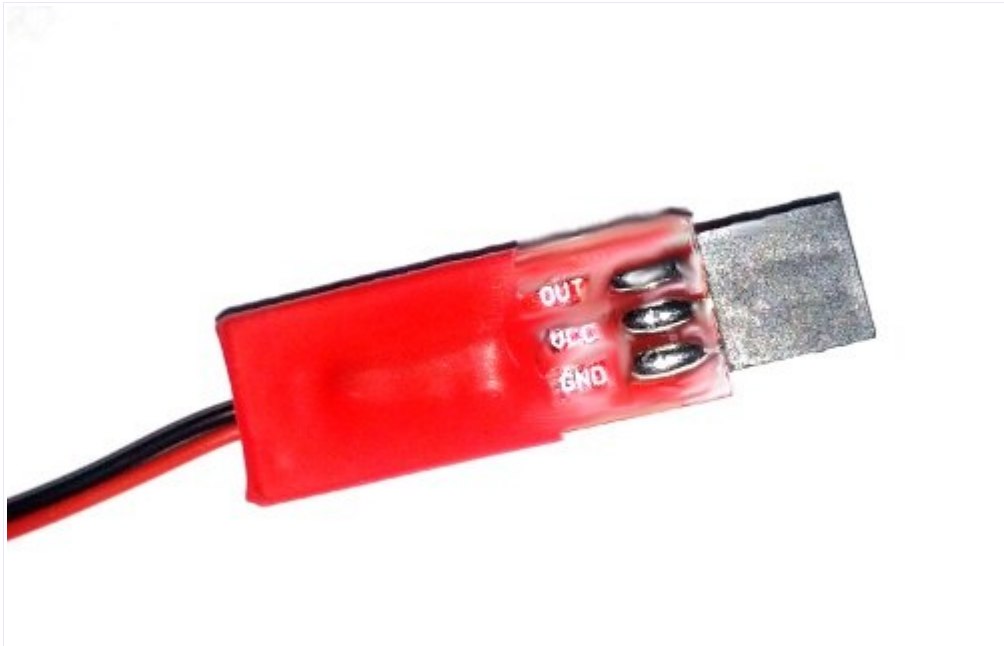
Vei avea nevoie de urmatoarele componente:

- [Arduino UNO](#)
- [Servomotor medium](#)
- [Senzor de indoire 11.4 cm brick](#) (sau [varianta mai mica de 5.5 cm](#))
- [Breadboard](#)
- [Fire pentru conexiuni](#).

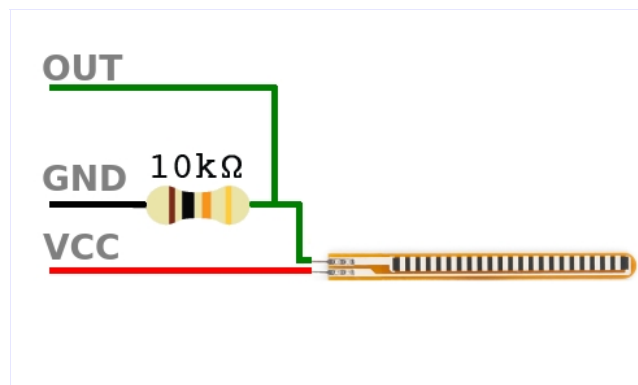
Servomotorul are un consum suficient de mic ca sa il poti alimenta direct din Arduino. Firele servomotorului au urmatoarele semnificatii:

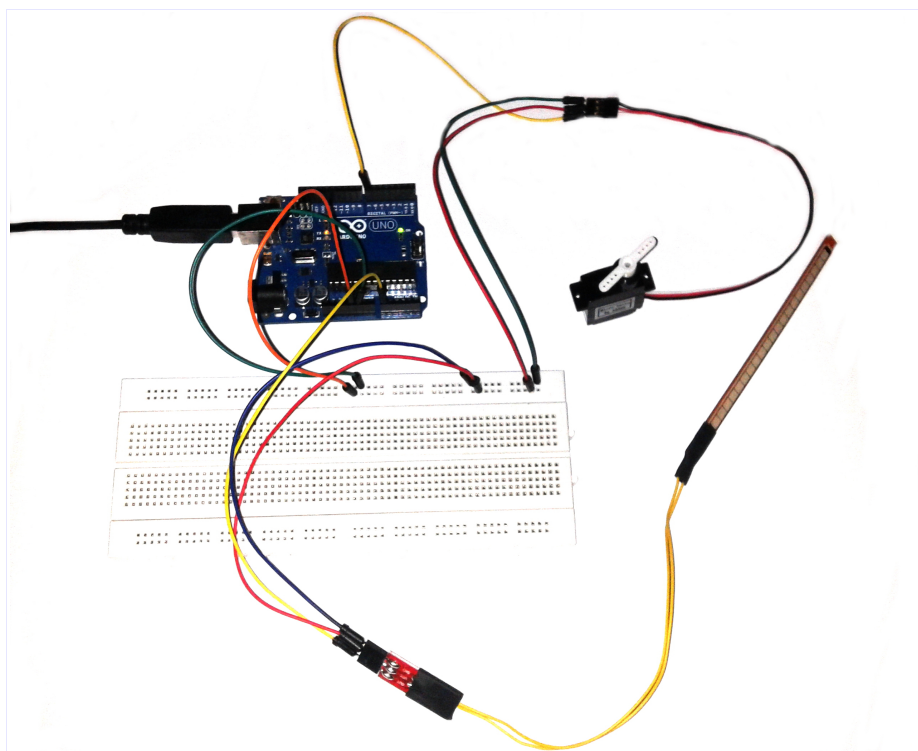
Firul negru - Gnd
Firul rosu - Vcc
Firul alb - PWM

Senzorul brick se conecteaza direct la portul analogic al lui Arduino. Semnificatiile firelor sunt marcate pe placuta de cablaj:



Intern, placuta contine un rezistor de 10 K (sub plasticul rosu), circuitul obtinut formand o configuratie de divizor de tensiune, ca mai jos. Ca sa il utilizezi, tot ce ai de facut este sa conectezi fire la cei trei pini mama (avantajul versiunii brick).





Tabelul de conexiuni:

Senzor indoire pin OUT	Arduino analog A0
Senzor indoire pin Vcc	Arduino 5V
Senzor indoire pin Gnd	Arduino Gnd
Servomotor fir alb	Arduino digital 7
Servomotor fir rosu	Arduino 5V
Servomotor fir negru	Arduino Gnd

Cum functioneaza?

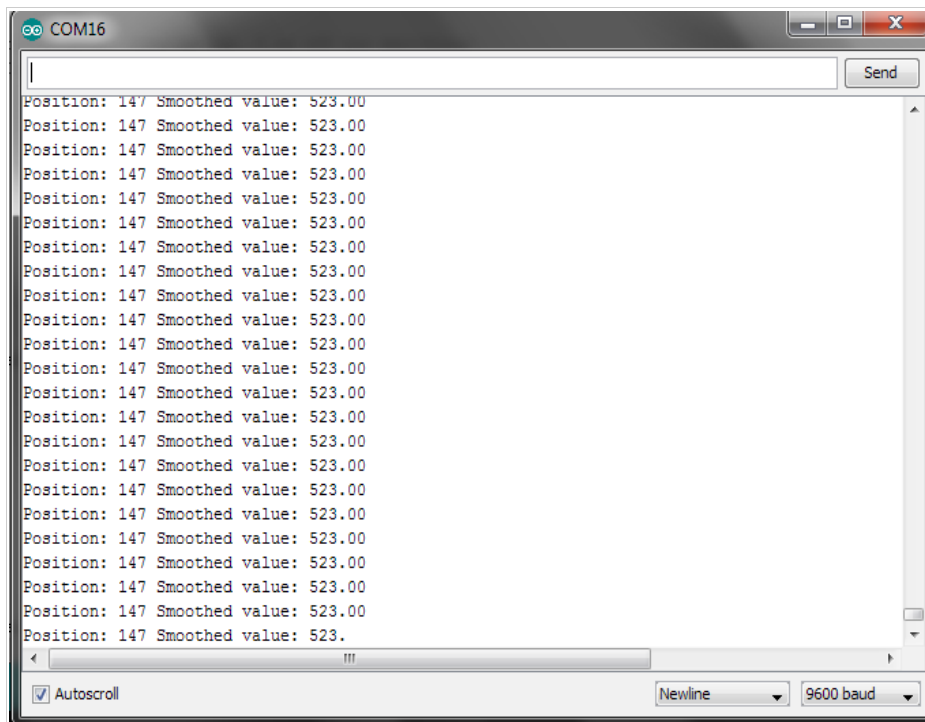
La nivelul sketch-ului, Arduino citeste in mod repetat valori ale senzorului de indoire si le utilizeaza in functia **smooth()** pentru a returna o valoare cat mai aproape de cea reala deoarece senzorul, ca orice sistem electronic, introduce erori - salturi de la o valoare la alta. Aceste salturi pot fi declansate de miscari bruste ale senzorului sau de o indoire incorecta. Gradul de atenuare al erorilor il controlezi din variabila **filterVal**. Pentru atenuarea maxima a erorilor poti sa alegi valoarea 0.0001. Daca servomotorul devine incontrolabil

sau se deplaseaza brusc in pozitii aleatorii inseamna ca este afectat de erori.

Valoarea medie obtinuta (smoothedVal) este redimensionata prin functia **map()** pentru a fi adaptata servomotorului si este mentinuta corect intre valorile (0,179) prin functia **constrain()**.

Valoarea medie si unghiul servomotorului sunt transmise si la monitorul serial ca sa poti vedea usor cum variaza la indoirea senzorului.

Monitorul serial va arata ca mai jos.



In plus, vei vedea ca servomotorul se misca atunci cand indoi senzorul intr-o parte sau in alta.

Daca vei folosi 5 senzori si 5 servomotoare, vei obtine ceva ca in acest film: <http://www.tehnorama.ro/animatronic-hand/> . Tot in link-ul anterior vei vedea si cateva poze din procesul de fabricatie al degetelor.

Codul sursa.

```
#include <Servo.h>

Servo myservo;
int sensVal;           // for raw sensor values
float filterVal = 0.0001; // this determines //smoothness - .0001
                        // is max 1 is off (no smoothing)
float smoothedVal;     // this holds the last loop value just use a
                        //unique variable for every different sensor that needs smoothing

void setup()
{
  Serial.begin(9600);
  myservo.attach(7);
}

void loop()
{
  sensVal = analogRead(0);
  smoothedVal = smooth(sensVal, filterVal, smoothedVal);
  int pos = map(smoothedVal, 440, 545, 0, 179);
  pos = constrain(pos, 0, 179);
  myservo.write(pos);
  Serial.print("Position: ");
  Serial.print(pos);
  Serial.print(" Smoothed value: ");
  Serial.println(smoothedVal);
  delay(50);
}

int smooth(int data, float filterVal, float smoothedVal){
  if (filterVal > 1){      // check to make sure param's are within
range
    filterVal = .99;
  }
  else if (filterVal <= 0){
    filterVal = 0;
  }
  smoothedVal = (data * (1 - filterVal)) + (smoothedVal *
filterVal);
  return (int)smoothedVal;
}
```