Documentație

Programare cu microprocesoare

ROBOT DANCER

COVACI ALEXANDRA

Grupa: 30239

Cuprins

[1. Cerință 3](#_Toc29760038)

[2. Schema electrică 3](#_Toc29760039)

[3. Specificații 4](#_Toc29760040)

[3.1. Arduino UNO 4](#_Toc29760041)

[3.2. Punte H 4](#_Toc29760042)

[3.3. Motoare DC 6](#_Toc29760043)

[3.4. Bluetooth 6](#_Toc29760044)

[4. Manual de utilizare 7](#_Toc29760045)

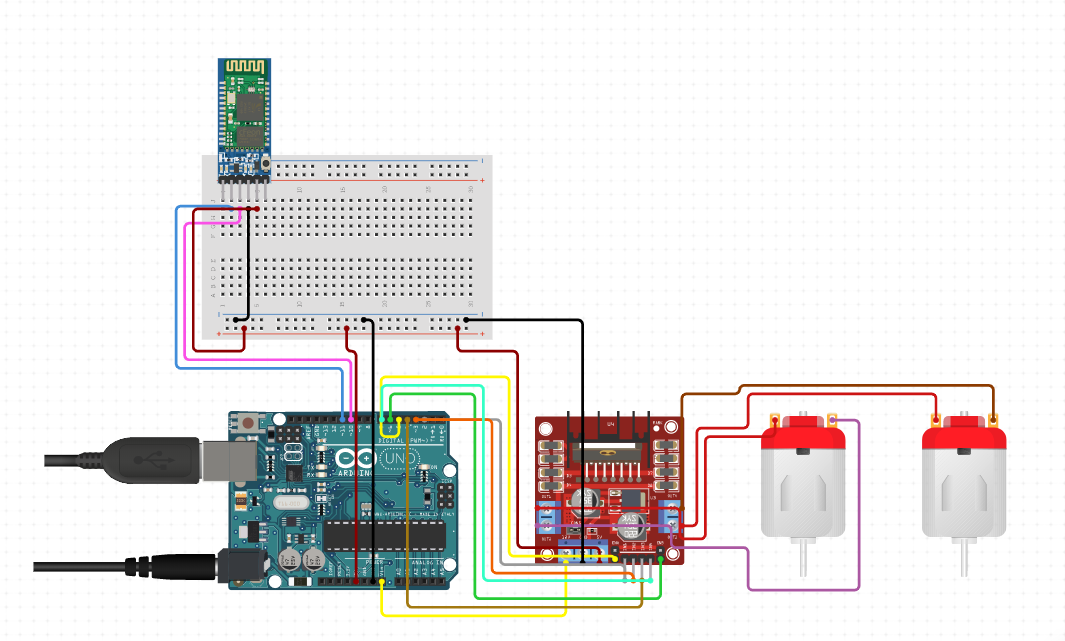
[5. Explicații 8](#_Toc29760046)

[6. Bibliografie 13](#_Toc29760047)

# Cerință

Acest proiect surpinde 2 roboți arduino care simuleaza picioarele unui dansator pentru a învăța să dansezi. Cei doi roboți comunică între ei prin bluetooth, unul dintre ei fiind master, iar al doilea slave-ul. Stilul de dans se transmite către master prin telefonul mobil. În momentul dat avem 2 stilul de dans: salsa și bachata. Roboții simulează pașii de bază pentru a te introduce în lumea dansului.

# Schema electrică



# Specificații

## Arduino UNO

Arduino este o companie [open-source](https://ro.wikipedia.org/wiki/Open-source) care produce atât plăcuțe de dezvoltare bazate pe [microcontrolere](https://ro.wikipedia.org/wiki/Microcontroler" \o "Microcontroler), cât și partea de [software](https://ro.wikipedia.org/wiki/Software) destinată funcționării și programării acestora.

Aceste plăci pun la dispoziția utilizatorului pini [I/O](https://ro.wikipedia.org/w/index.php?title=I/O&action=edit&redlink=1), digitali și analogici, care pot fi interfațați cu o gamă largă de plăcuțe numite scuturi (shield-uri) și/sau cu alte circuite. Plăcile au interfețe de comunicații seriale, inclusiv [USB](https://ro.wikipedia.org/wiki/USB) pe unele modele, pentru a încărca programe din [calculatorele personale](https://ro.wikipedia.org/wiki/Computer_personal" \o "Computer personal). Pentru programarea microcontrolerelor, Arduino vine cu un [mediu de dezvoltare integrat](https://ro.wikipedia.org/wiki/Mediu_de_dezvoltare" \o "Mediu de dezvoltare) ([IDE](https://ro.wikipedia.org/wiki/IDE_(dezambiguizare))) bazat pe proiectul [Processing](https://ro.wikipedia.org/w/index.php?title=Processing&action=edit&redlink=1), care include suport pentru [limbaje de programare](https://ro.wikipedia.org/wiki/Limbaj_de_programare" \o "Limbaj de programare) ca [C](https://ro.wikipedia.org/wiki/C_(limbaj_de_programare)) și [C++](https://ro.wikipedia.org/wiki/C%2B%2B).

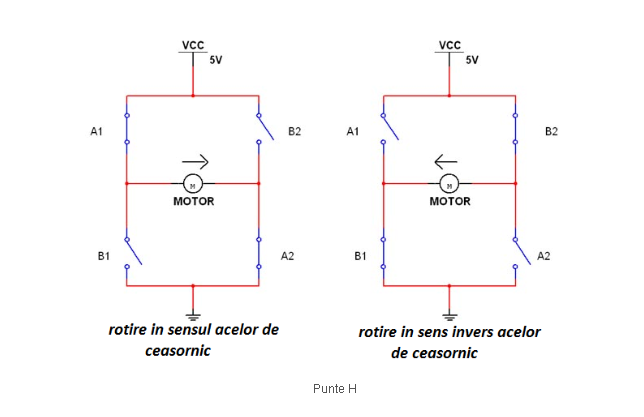
Arduino Uno R3 este o placă de dezvoltare bazată pe microcontrollerul ATmega328. Are 14 pini de intrare/ieșire (dintre care 6 pot fi folosiți ca ieșiri PWM), 6 intrări analog, un oscilator de 16MHz, o conexiune USB, mufă de alimentare, și un buton de reset.

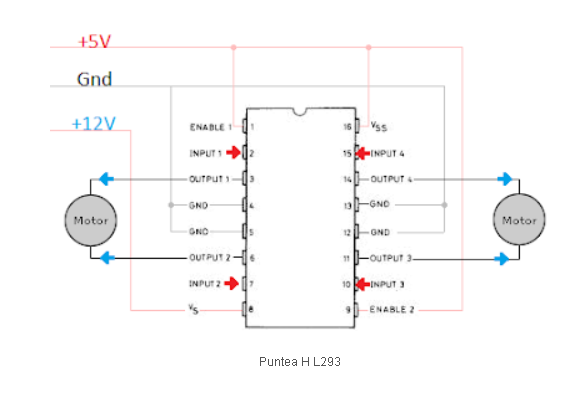
Poate fi alimentat direct de la calculator, de la portul USB, prin intermediul unei baterii de 9V sau a unui alimentator de 9V.

## Punte H

O punte H este un circuit construit din 4 tranzistoare care acționează ca niște întrerupătoare, iar poziția acestora determină sensul de rotire al motorului. În stânga figurii de mai jos comutatoarele A1 și A2 sunt închise și curentul va trece prin ele, iar sensul va fi prin A1-motor-A2 și motorul se va roti în sensul acelor de ceasornic. În dreapta figurii de mai jos celelalte două comutatoare B1 și B2 sunt închise și curentul va trece prin ele și prin motor, iar motorul se va roti în sens invers acelor de ceasornic. În total există 2^4 = 16 poziții posibile ale comutatoarelor. In general se folosesc doar 4:

* toate deschise --> motorul este în standb
* A1,A2 închise, B1,B2 deschise --> motorul se învârte în sensul acelor de ceas
* A1,A2 deschise, B1,B2 închise --> motorul se învârte în sens invers acelor de ceas
* cele de sus închise, cele de jos deschise sau invers --> motorul este frânat.

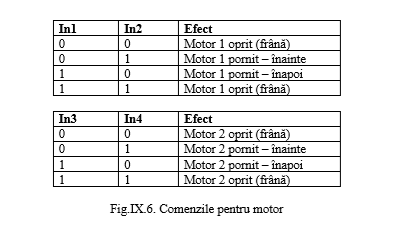




În figură se observă că integratul are 4 intrări (input) logice (0V sau 5V) și 4 ieșiri (output) de putere (0V sau 12V). Fiecare ieșire corespunde fiecărei intrări. Astfel, dacă Input\_1 = 0V -> Output\_1 = 0V, iar dacă Input\_1 = 5V -> Output\_1 = 12V. Practic se obține o translatare a nivelului logic 0-5V în nivelul logic 0-12V. Acesta din urmă poate oferi atat o tensiune mai mare (12V), cât și un curent mai mare (0.6A---1A în funcție de capsula). De asemenea, există și doi pini de Enable (EN1 pentru Output1 și 2, EN2 pentru Output3 și 4) folosiți pentru a valida cele 4 ieșiri. Acești 2 pini, împreună cu cei 4 pini de Input se conectează la un microcontroller care gestionează acționarea acestora. Sunt două modalități de folosire:

1. Pinii de ENable conectați fizic la +5V --> Ieșirile sunt validate permanent și controlate doar prin cei 4 pini de input conectați la microcontroller.

2. Pinii de ENable sunt conectați la microcontroller. Avantajul este că prin acești 2 pini se poate comanda turația motoarelor prin [PWM](https://en.wikipedia.org/wiki/Pulse-width_modulation).



## Motoare DC

Motoarele de curent continuu (DC motors) clasice convertesc energia electrică în lucru mecanic. Viteza de rotaţie a unui motor este proporţională cu tensiunea de alimentare de la bornele acestuia, iar direcţia de rotaţie depinde de polaritate (conectarea celor 2 fire de alimentare ale motorului la +Vcc şi Gnd, sau vice-versa). Motoarele au cutie de viteze (reductor de turaţie) cu raport de 1:48, ceea ce înseamnă că pentru o rotaţie completă a axului extern se efectuează de fapt 48 de rotaţii ale motorului electric. Folosirea unui reductor are avantajul că măreşte forţa de acţionare, cu costul vitezei.

## Bluetooth

Bluetooth este un protocol standardizat pentru comunicarea de date folosind frecvența radio de 2.4 GHz. Banda de 2.4 GHz nu necesită licență pentru operare, și este folosită și de alte protocoale de comunicare wireless, precum ZigBee sau WiFi. Rețelele Bluetooth (numite și piconet) folosesc un aranjament master/slave pentru a controla unde și când dispozitivele pot transmite date. Un dispozitiv master poate fi conectat la până la șapte dispozitive de tip slave, dar un dispozitiv de tip slave poate fi conectat la un singur master. Dispozitivul master coordonează comunicația în rețea, putând transmite date la orice slave, sau cerând date de la un slave. Dispozitivele slave pot să comunice doar cu dispozitivul master, nu și cu alte dispozitive slave din rețea. HC-05 este un modul ușor de folosit în proiectele cu Arduino. Configurarea lui implicită este în mod Slave, cu numele HC-05, și setările pentru interfața serială compatibile cu setările implicite ale Arduino (Baud Rate 9600, fără paritate, 8 biți de date, 1 bit de stop). Modulul poate fi comercializat în variantele cu 4 pini (VCC, GND, RX și TX), sau în varianta cu 6 pini, care permite configurarea și verificarea stării prin program. Pinii disponibili ai dispozitivului din laborator sunt:

1. EN (Uneori cu numele KEY): Dacă e conectat la nivel logic 1 înainte de a se alimenta modulul, acesta intră în modul de configurare (LED-ul clipește lent, o dată la 2 secunde). În acest mod modulul așteaptă comenzi de tip AT prin interfața serială, la un Baud Rate de 38400 biți pe secundă. Pentru operare normală, lăsați acest pin neconectat!

2. VCC: +5v – pin de alimentare, se conectează la Arduino la un pin de 5 V.

3. GND: pin de masă, se conectează la un pin GND al Arduino.

4. TXD: Pin de transmisie al interfeței seriale, se conectează la un RXn al Arduino (de exemplu la RX1, pentru lucrarea de față).

5. RXD: Pin de recepție al interfeței seriale, se conectează la un TXn al Arduino (de exemplu la TX1, pentru lucrarea de față).

6. STATE: spune dacă este conectat sau nu. Acest pin este conectat la LED-ul de pe modul. În funcție de modul în care clipește acest LED, modulul este în una din cele trei stări:

- Clipire odată la 2 secunde: Modulul este în modul de configurare (de comandă).

- Clipire rapidă: Modulul așteaptă conexiuni – acest mod ar trebui să fie activ la pornirea sistemului.

- Clipire de două ori în 1 secundă: Conexiune stabilită, se pot transmite și recepționa date.

# Manual de utilizare

După ce roboții sunt construiți, se încarcă cele două coduri pentru fiecare robot în parte. Se așează pe o suprafață fină și se pornesc de la comutator sau se bagă în priză (depinde de alimentare). Se deschide o aplicație pentru Bluetooth de pe telefon, de exemplu ”Serial Bluetooth Terminal”. Se conectează telefonul cu robotul master pentru a începe să transmită date. Din momentul în care suntem conectați, dacă trimitem roboțiilor cuvintele ”salsa” sau ”bachata”, aceștia vor începe pașii de bază din dansul selectat, după ce ai finalizat mișcarea se poate tasta din nou stilul de dans, dacă nu s-a înțeles primul dans se va transmite iar numele dansului anterior, altfel se va scrie numele altui dans.

# Explicații

#include <Servo.h>

// Pinii motor 1

#define mpin00 6

#define mpin01 9

// Pinii motor 2

#define mpin10 3

#define mpin11 5

Servo srv;

void setup() {

// configurarea pinilor motor ca iesire, initial valoare 0

digitalWrite(mpin00, 0);

digitalWrite(mpin01, 0);

digitalWrite(mpin10, 0);

digitalWrite(mpin11, 0);

pinMode (mpin00, OUTPUT);

pinMode (mpin01, OUTPUT);

pinMode (mpin10, OUTPUT);

pinMode (mpin11, OUTPUT);

Serial.begin(38400); // Interfata Serial pentru master

}

// Funcție pentru controlul unui motor

// Intrare: pinii m1 și m2, direcția și viteza

void StartMotor (int m1, int m2, int forward, int speed)

{

if (speed==0) // oprire

{

digitalWrite(m1, 0);

digitalWrite(m2, 0);

}

else

{

if (forward) // forward = 0 mers in fata, forward = 1 mers in spate

{

digitalWrite(m2, 0);

analogWrite(m1, speed); // folosire PWM

}

else

{

digitalWrite(m1, 0);

analogWrite(m2, speed);

}

}

}

// Funcție de siguranță

// Execută oprire motoare, urmată de delay

void delayStopped(int ms)

{

StartMotor (mpin00, mpin01, 0, 0);

StartMotor (mpin10, mpin11, 0, 0);

delay(ms);

}

String text;

bool enable = false;

void loop() {

if (Serial.available()) // Citire de la Bluetooth

text = "";

char c = Serial.read(); //Citeste cate un caracter

while(c<=122 && c>=65)

{

text = text + c; // se formeaza cuvantul transmis prin bluetooth

c = Serial.read();

}

Serial.write(text);

// if(text != "");

// Serial.println(text+"1");

if(text == "salsa") // se verifica daca textul primit de masina este ”salsa”

{

enable = true;

salsa(); // se apeleaza codul pentru primul dans

}

else if(text == "bachata") // se verifica daca textul primit de masina este ”bachata”

{

enable = true;

bachata();// se apeleaza codul pentru al doilea dans

}

else

enable = false;

// if(enable == true)

// Serial.println("start1");

delay(500);

}

void salsa()

{

// Acum se pornesc motoarele DC, robotul merge in fata

StartMotor (mpin00, mpin01, 0, 128);

StartMotor (mpin10, mpin11, 0, 128);

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

// Acum se pornesc motoarele DC, robotul merge in spate

StartMotor (mpin00, mpin01, 1, 112);

StartMotor (mpin10, mpin11, 1, 112);

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

// Acum se pornesc motoarele DC, robotul merge in spate

StartMotor (mpin00, mpin01, 1, 16);

StartMotor (mpin10, mpin11, 1, 16);

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

}

void bachata(){

// Acum se pornesc motoarele DC, robotul merge in fata

StartMotor (mpin00, mpin01, 0, 128);

StartMotor (mpin10, mpin11, 0, 128);

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

// Acum se pornesc motoarele DC, robotul merge in fata

StartMotor (mpin00, mpin01, 0, 128);

StartMotor (mpin10, mpin11, 0, 128);

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

// Acum se pornesc motoarele DC, robotul merge in spate

StartMotor (mpin00, mpin01, 1, 128);

StartMotor (mpin10, mpin11, 1, 128);

delay (500); // Cât timp e motorul pornit

delayStopped(500); // Cât timp e oprit

// Acum se pornesc motoarele DC, robotul merge in spate

StartMotor (mpin00, mpin01, 1, 128);

StartMotor (mpin10, mpin11, 1, 128);

}

# Bibliografie

1. Radu Dănescu, Mircea Paul Mureșan, Răzvan Itu, Proiectare cu Microprocesoare, Indrumător de laborator, UTPRESS, Cluj-Napoca, 2018
2. Radu Dănescu, Laborator 8 – Utilizarea comunicației Bluetooth
3. <https://ro.wikipedia.org/wiki/Arduino>
4. <http://nedelcugheorghe.blogspot.com/2015/08/controlul-unui-motor-folosind-o-punte-h.html>