

Universitatea Tehnică a Moldovei
Facultatea Calculatoare Informatică și Microelectronică
Departamentul Ingineria Software și Automatică

RAPORT

Lucrarea de laborator nr. 4.2
La disciplina “Internetul Lucrurilor”
Tema: Actuatori – DC Motor

A efectuat: st. gr. SI-211
A verificat:

Adrian Chihai
Valentina Astafi

Chișinău – 2024

Definire Problema

Sa se realizeze o aplicatie in baza de MCU care va controla dispozitivele de actionare cu comenzi receptionate de la interfata seriala si raportare catre LCD.

Dispozitivele de actionare vor fi urmatoarele:

- un motor in curent continui cu comenzi de setare a puterii motorului intre (-100% .. 100%) adica inainte si inapoi, si viteza prin intermediul driverului L298

Descrierea funcțiilor

1. void setup()

- **Scop:** Configurează pinii hardware, inițializează LCD-ul și Servo-ul, și pregătește comunicația serială.
- **Activități:**
 - Setează pinii trigPin, echoPin, ledGreen, și ledRed ca OUTPUT sau INPUT.
 - Atașează servo-ul la pinul specificat (servoPin).
 - Inițializează LCD-ul cu iluminarea de fundal activată.
 - Afișează textul "Actuator:" pe ecranul LCD.

2. void turnOnLEDGreen()

- **Scop:** Activează LED-ul verde și dezactivează LED-ul roșu.
- **Activități:**
 - Setează pinul ledGreen la HIGH (aprins).
 - Setează pinul ledRed la LOW (stins).

3. void turnOnLEDRed()

- **Scop:** Activează LED-ul roșu și dezactivează LED-ul verde.
- **Activități:**
 - Setează pinul ledRed la HIGH (aprins).

- Setează pinul ledGreen la LOW (stins).

4. void moveServo(int angle)

- **Scop:** Controlează mișcarea servo-motorului către un anumit unghi.
- **Parametri:**
 - angle: Unghiul în care trebuie să se deplaseze servo-ul (0° - 180°).
- **Activități:**
 - Trimite semnalul necesar servo-motorului pentru a se roti la unghiul specificat.

5. void displayState(const char* state)

- **Scop:** Afișează starea curentă a actuatorului pe rândul al doilea al LCD-ului.
- **Parametri:**
 - state: Textul care trebuie afișat pe ecran.
- **Activități:**
 - Plasează cursorul pe rândul 2, coloana 1.
 - Scrie textul specificat și elimină caracterele suplimentare rămase de la afișările anterioare.

6. void loop()

- **Scop:** Asigură funcționalitatea principală a sistemului: controlul actuatorului bazat pe comenzi seriale și măsurători ale distanței.
- **Activități principale:**
 1. **Citire comenzi seriale:**
 - Verifică dacă există date disponibile pe portul serial.
 - Dacă se primește comanda ON, activează LED-ul verde, rotește servo-ul la 90° , și afișează mesajul "Deschis".
 - Dacă se primește comanda OFF, activează LED-ul roșu, rotește servo-ul la 0° , și afișează mesajul "Inchis".

2. Măsurarea distanței cu senzorul HC-SR04:

- Trimite un impuls ultrasonic prin pinul trigPin.
- Măsoară durata semnalului returnat folosind pulseIn.
- Convertește durata în distanță folosind formula: $\text{distance} = (\text{duration} * 0.034) / 2$.

3. Controlul actuatorului pe baza distanței:

- Dacă distanța este mai mică sau egală cu 5 cm, activează LED-ul verde, rotește servo-ul la 90° și afișează "Deschis".
- Dacă distanța este mai mare, activează LED-ul roșu, rotește servo-ul la 0° și afișează "Inchis".

4. Pauză de stabilitate:

- Introduce un delay(100) pentru a preveni suprasolicitarea senzorului și a altor componente.

În figura 1 și figura 2 putem observa cum servo motorul este actionat controlat cu ajutorul comenzilor utilizatorului

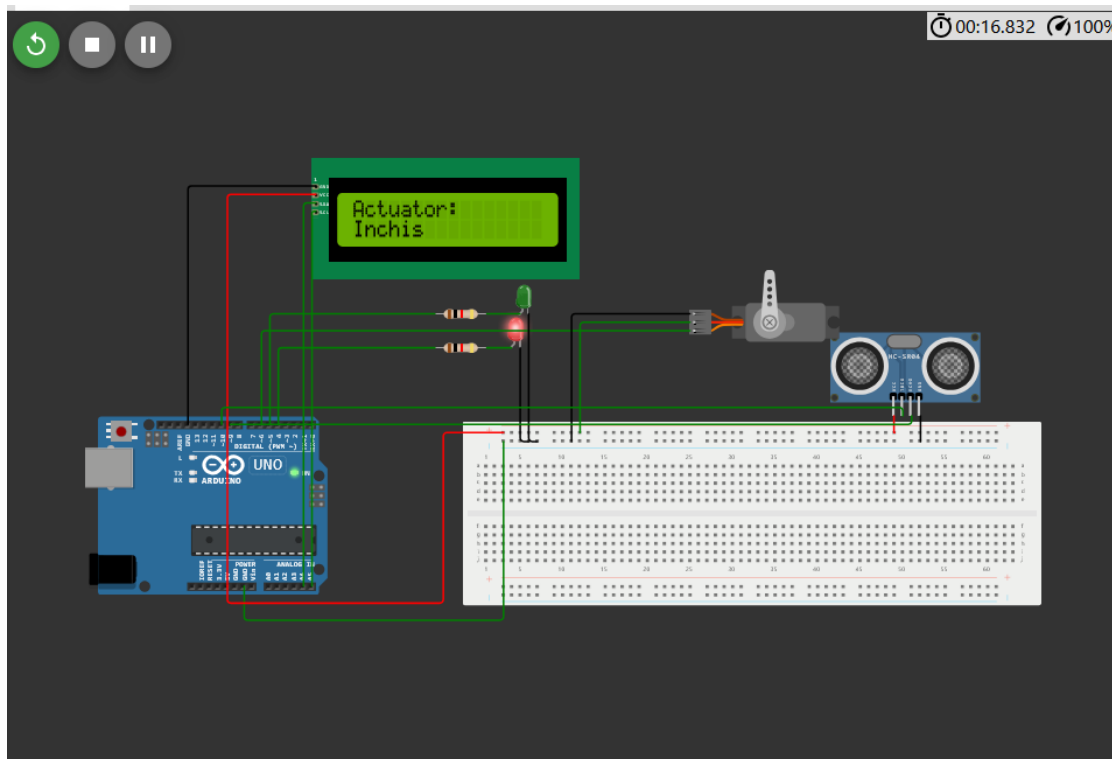


Figura 1 Circuitul închis cu ajutorul comenzilor

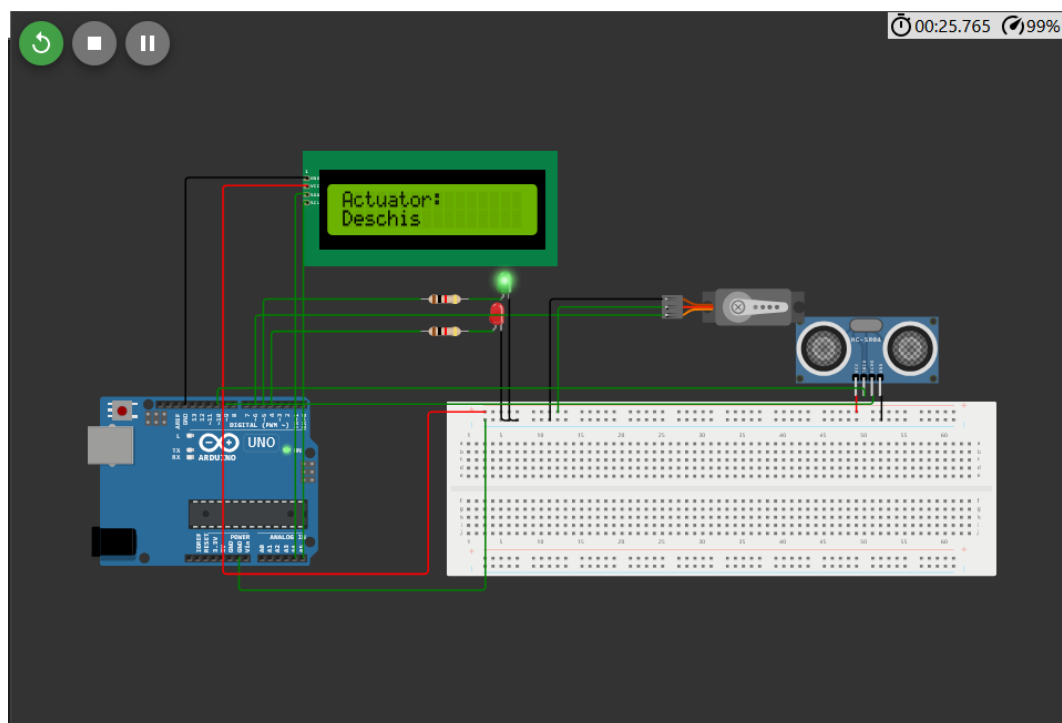


Figura 2 Circuitul deschis cu ajutorul comenzilor

În figura 3 și figura 4 putem observa cum servo motorul este acționat cu ajutorul datelor senzorului de distanță.

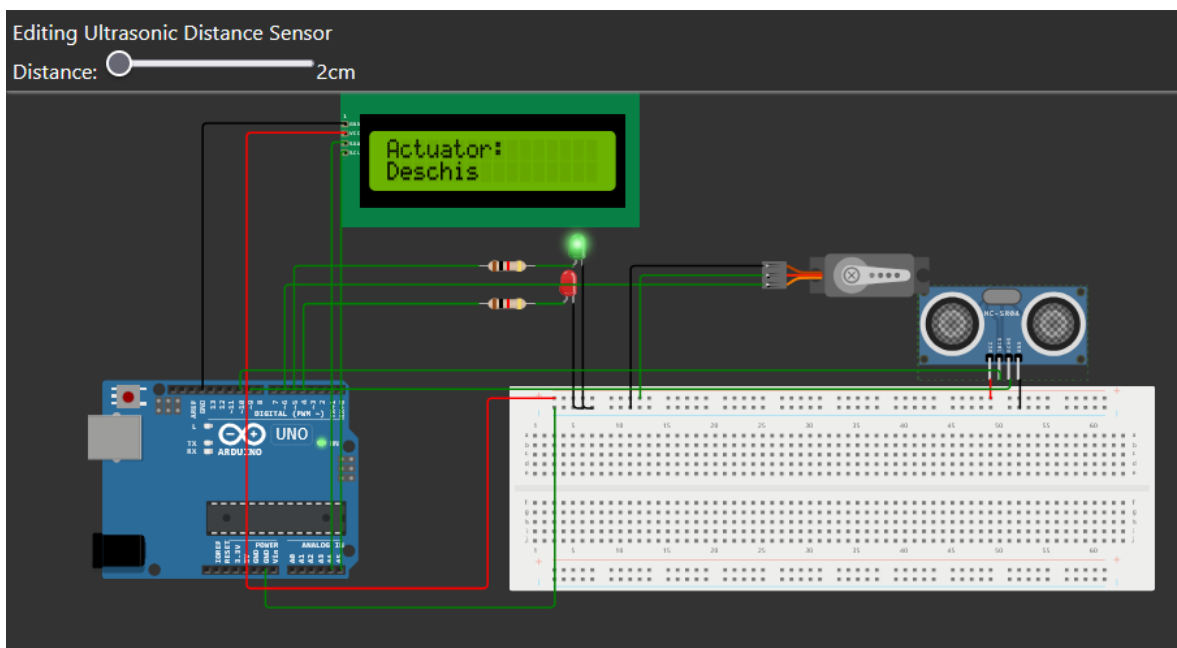


Figura 3 Circuitul deschis cu ajutorul senzorului

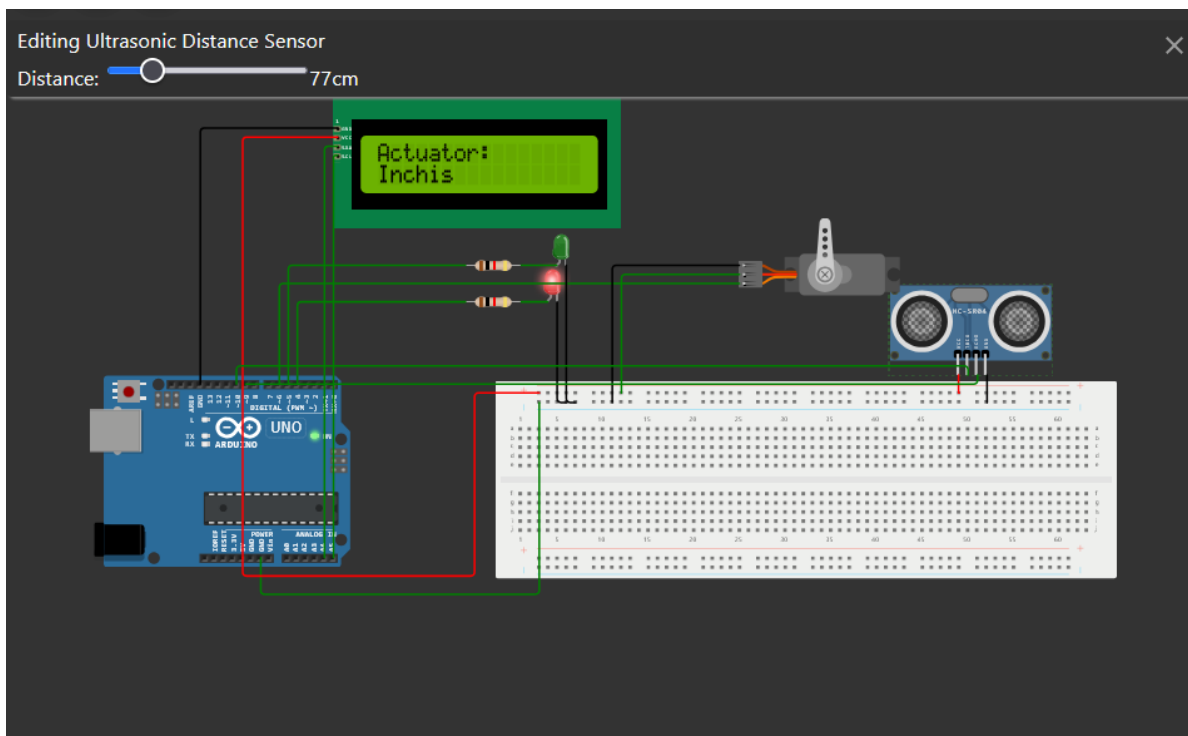


Figura 4 Circuitul închis cu ajutorul senzorului

CONCLUZIE

Această sarcină ilustrează un exemplu de integrare hardware-software care permite controlul unui actuator cu ajutorul unui senzor de distanță, afișaj LCD și interfață cu utilizatorul prin portul serial. Prin implementarea acestui proiect, sistemul poate răspunde atât la comenzi manuale primite de la utilizator, cât și la stimuli din mediu (distanța măsurată), realizând astfel o soluție semi-automatizată de control. Această flexibilitate îl face util în aplicații care necesită un control ușor de operat și reacții rapide la schimbările din mediu, cum ar fi în sisteme de siguranță sau automatizări industriale.

Importanța acestei sarcini constă în familiarizarea cu concepte esențiale de integrare între componente electronice și codul necesar pentru a le coordona. Astfel, este un exemplu de proiect hands-on care dezvoltă atât competențe de programare, cât și de înțelegere a circuitelor electronice și a funcționării senzorilor și actualelor. În plus, proiectul contribuie la dezvoltarea abilităților de gândire critică și problem-solving, necesare în proiectarea unor sisteme eficiente, fiabile și adaptabile în funcție de cerințele utilizatorului sau de mediul înconjurător.

Referințe bibliografice

1. <https://wokwi.com/projects/374644935912458241>