

# Parking Sensor

## Introducere

Acest proiect constă într-un senzor de parcare, care măsoară și semnalizează distanța dintre vehicul și un obstacol din spate. Prezentarea pe scurt a proiectului vostru:

- Ce face? - Proiectul detectează distanța până la un obstacol cu ajutorul unui senzor ultrasonic și alertează utilizatorul printr-un buzzer și afișaj LED.
- Scopul - Creșterea siguranței în timpul parcării, prevenind coliziunile.
- Care a fost ideea de la care ați pornit? - Observarea frecventelor dificultăți de parcare în spații înguste fără senzori de distanță.
- De ce credeți că este util pentru alții și pentru voi? - Poate fi folosit în vehicule reale, pentru a ajuta soferii să diminueze coliziunile, mai ales în locuri înguste sau în parcări.

## Descriere generală

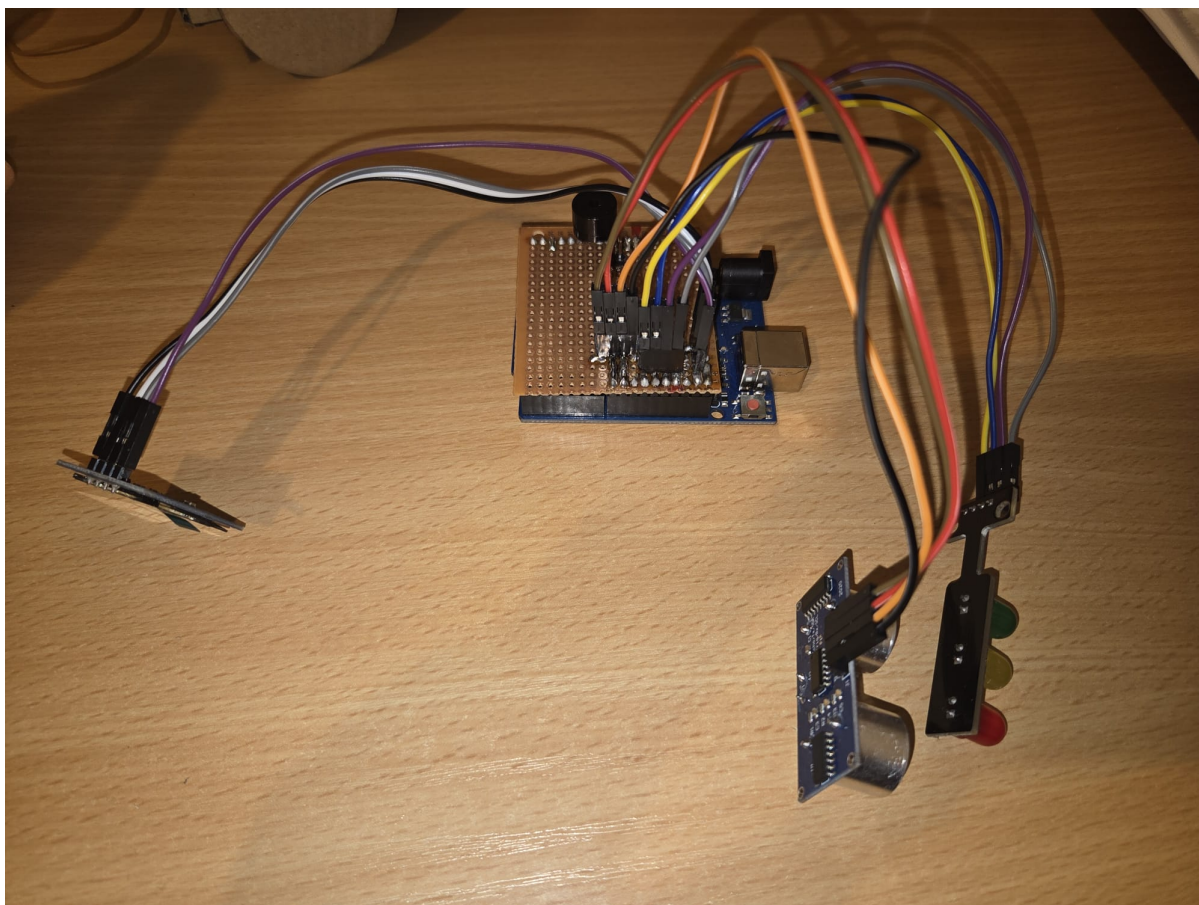
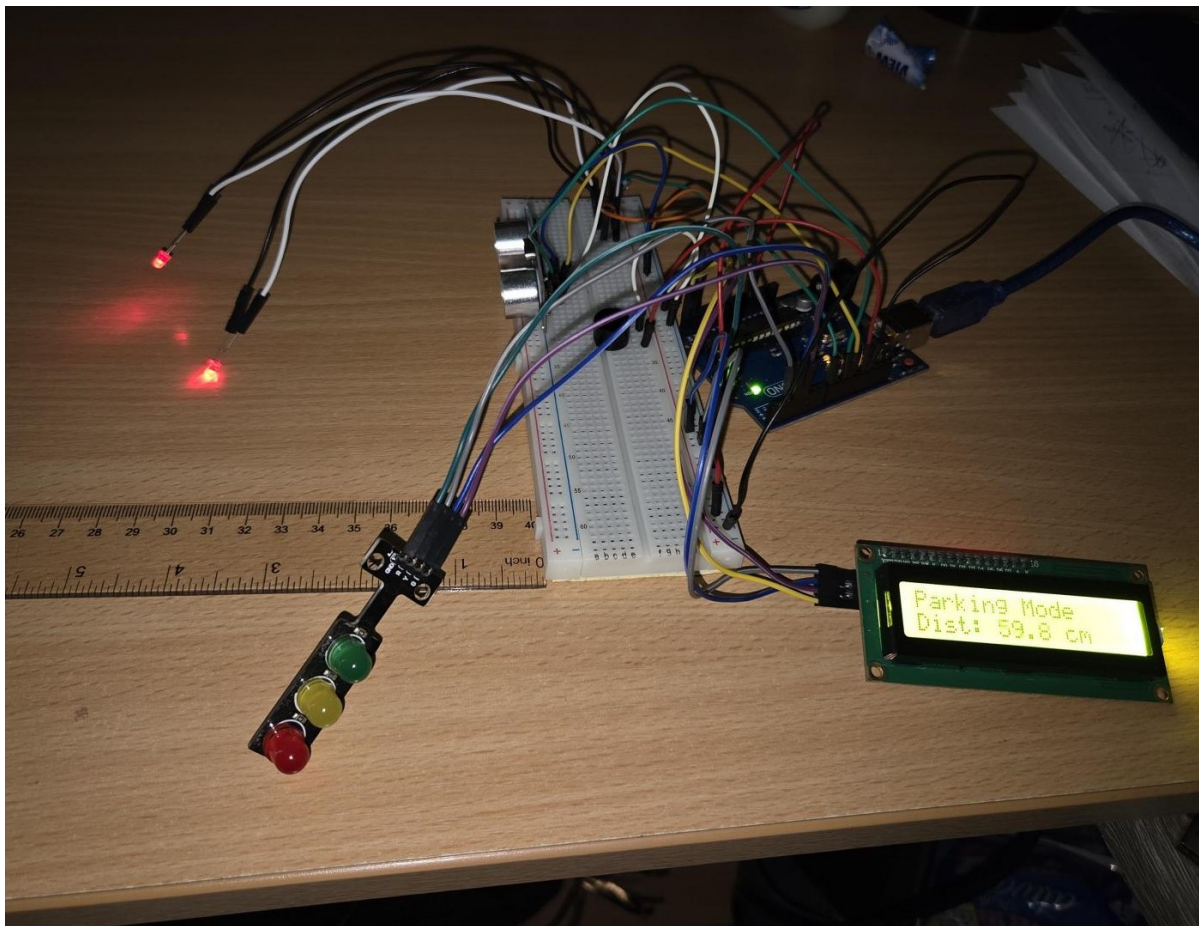


## Hardware Design

Componente:

- Arduino UNO R3
- Breadboard
- Fire
- LED-uri colorate/ modul LED-URI(rosu, galben, verde)
- Rezistente
- Senzor Ultrasonic HC-SR04+
- Buzzer
- Modul LCD





# Software Design

## Mediu de dezvoltare:

- Proiectul este dezvoltat în Arduino IDE, utilizând arhitectura bare-metal AVR C parțial, pentru un control direct asupra pinilor hardware.

## Librării și surse 3rd-party:

- Se utilizează biblioteca Adafruit SSD1306 împreună cu Adafruit GFX pentru afișarea informațiilor pe un ecran OLED I2C. Nu se folosesc funcții Arduino standard (digitalWrite, digitalWrite, etc.) pentru controlul pinilor – s-au folosit registre AVR (PORTx, PINx, DDRx) pentru precizie și performanță.

## Algoritmi și structuri implementate:

- Măsurarea distanței cu un senzor ultrasonic HC-SR04, folosind control direct pe pini și măsurarea duratei impulsului cu micros().
- Afișarea distanței pe un ecran OLED în timp real.
- Activarea unui buzzer și a unor LED-uri RGB în funcție de distanța detectată:
  - Verde = sigur
  - Galben = apropiere
  - Roșu = pericol
  - Beep-uri diferite în funcție de nivelul de risc, imitând un senzor de parcare auto.
- Avertizare “Out of range” dacă distanța depășește 500 cm sau dacă senzorul nu primește ecou.

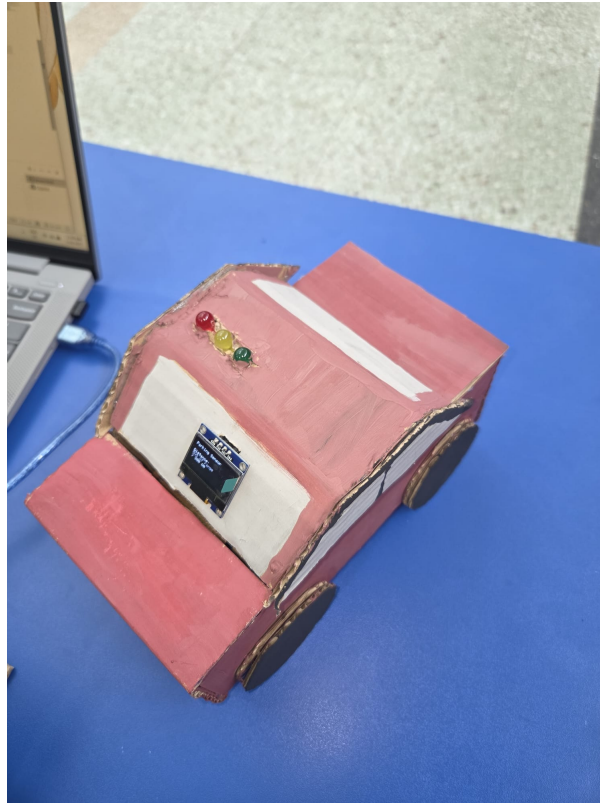
## Surse și funcții implementate:

- Funcții pentru trimiterea și citirea semnalului ultrasonic
- Funcție personalizată carSensorBeep() pentru emiterea de sunete tipice unui senzor de parcare
- Inițializare și afișare grafică pe OLED
- Gestionarea pinilor I/O folosind registre AVR (bare-metal)

# Rezultate Obținute


Proiectul folosește un senzor pentru detectarea distanței, 3 LED-uri(verde, galben, rosu), care se aprind in functie de cat de aproape se afla senzorul fata de un obstacol, un buzzer, care emite sunet mai repede sau mai incet si un ecran OLED, care arata cati cm mai sunt la obstacol.

[https://drive.google.com/file/d/1JCA0JTdGQy4\\_kYNqO7TZx7Rw3xZeELfZ/view?usp=sharing](https://drive.google.com/file/d/1JCA0JTdGQy4_kYNqO7TZx7Rw3xZeELfZ/view?usp=sharing)



## Concluzii

## 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



1. 01.05.2025 - Documentare despre diverse proiecte
2. 03.05.2025 - Alegerea proiectului
3. 04.05.2025 - Comanda componentelor
4. 06.05.2025 - Venirea componentelor
5. 07.05.2025 - Testarea componentelor
6. 11.05.2025 - Implementare hardware v1 - pe breadboard; am lipit doar senzorul, împreună cu becurile și cu buzzer-ul.
7. 14.05.2025 - Implementare hardware v2 - am adăugat un ecran lcd.
8. 14.05.2025 - Comanda de alte componente
9. 16.05.2025 - Implementare hardware v3 - am înlocuit lcd cu un oled.
10. 16.05.2025 - Implementare hardware v4(final) - am lipit componentele pe un pcb.
11. 19.05.2025 - Implementare software v1 - am făcut un cod în mare pentru a vedea dacă totul este în regulă și funcționează cum trebuie.
12. 23.05.2025 - Implementare software v2 - am actualizat codul, adăugând precizarea că senzorul detectează suprafețe >500 cm, deoarece acesta detecta 0.00cm și nu se comporta cum trebuie.
13. 26.05.2025 - Implementare software v3(final) - am făcut codul final, am făcut buzzer-ul să cante mai prietenos.

## Bibliografie/Resurse

**Resurse Hardware:** <https://docs.arduino.cc/>

**Resurse Software:**

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2025/cristi/andreea.serbanica>



Last update: **2025/05/28 09:16**