

## Reverse Parking Sensor

Autor: [Teodora RUMĂNU](#)

### Introducere

- Proiectul constă în atenționarea prin alarmă și lumini atunci când mașina se află la o anumită distanță față de un obstacol.
- Scopul proiectului este acela de a facilita parcare a unei mașini și de a preveni o eventuală coliziune.

### Descriere generală

Pentru a înțelege modul de funcționare, trebuie avute în vedere două praguri pentru distanță - unul aflat la 10cm, iar celălalt aflat la 50cm - care descriu trei zone:



- **Zona verde:** aflată după pragul de 50cm, la o distanță suficient de mare astfel încât să ofere siguranță
- **Zona galbena:** are limitele între cele două praguri, descrie o distanță mai puțin sigură, dar nu alarmantă, însă este nevoie de un avertisment pentru a preveni un potențial accident
- **Zona roșie:** se află sub pragul de 10cm, la o distanță critică, alarmantă, foarte aproape de impact cu obstacolul

### Schema bloc



### Hardware Design

#### Lista de Piese

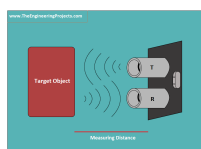
- [Arduino UNO](#)
- [Breadboard](#)
- [Senzor Ultrasonic HC-SR04](#)
- [Buzzer](#)
- [LED-uri](#)
- [Rezistențe](#)
- [Fire tată-tată](#)

### Schema electrică



## Senzor Ultrasonic HC-SR04

- funcționează precum un aparat de sonar - emite ultrasunete la o frecvență de 40000Hz în aer iar dacă acestea lovesc un obiect, sunt reflectate, conform schemei:



- detectează obiecte de la 2 cm la 4m distanță

- are 4 pini - GND, VCC, Trig și Echo
- pinii GND și VCC se conectează la pinii echivalenți de pe Breadboard, care la rândul lor sunt conectați la pinii de GND și VCC de pe placă
- pinul Trig este conectat la pinul 10 de pe plăcuța Arduino, iar Echo, la pinul 9

## Buzzer

- de tip Piezoelectric
- are încorporate două fire, unul negru(GND) și unul roșu
- firul de GND este conectat la pinul corespunzător de pe Breadboard, iar firul roșu este conectat la pinul 2 de pe Arduino

## Software Design

- Mediu de dezvoltare și compilare: Arduino IDE

## Implementare

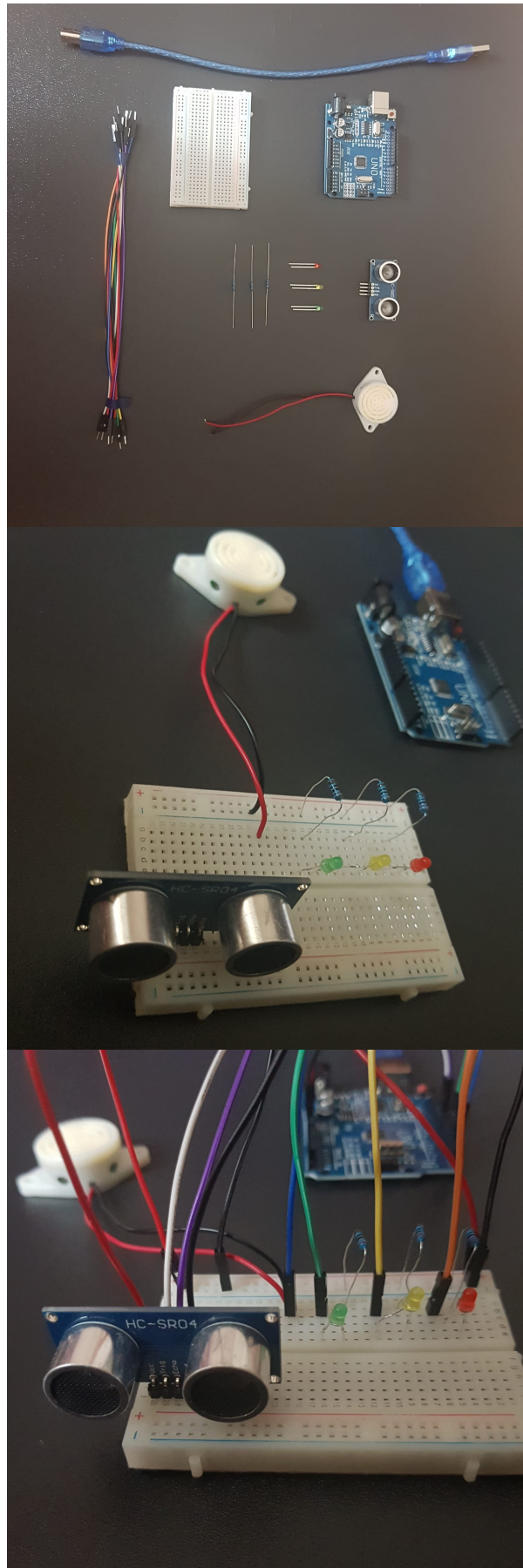
1. Înainte de setup():
  - definire pini ce urmează să fie folosiți
  - definire variabile pentru timp și distanța
2. setup():
  - setare pini pentru OUTPUT/INPUT
  - start interfață seriala
3. loop():
  - curățare pin TRIG prin setarea acestuia la LOW pentru 5 microsecunde
  - TRIG este setat la HIGH pentru 10 microsecunde pentru a putea emite ultrasunetele
  - pentru calcularea distanței, se folosește formula  $d=t*v$ , unde

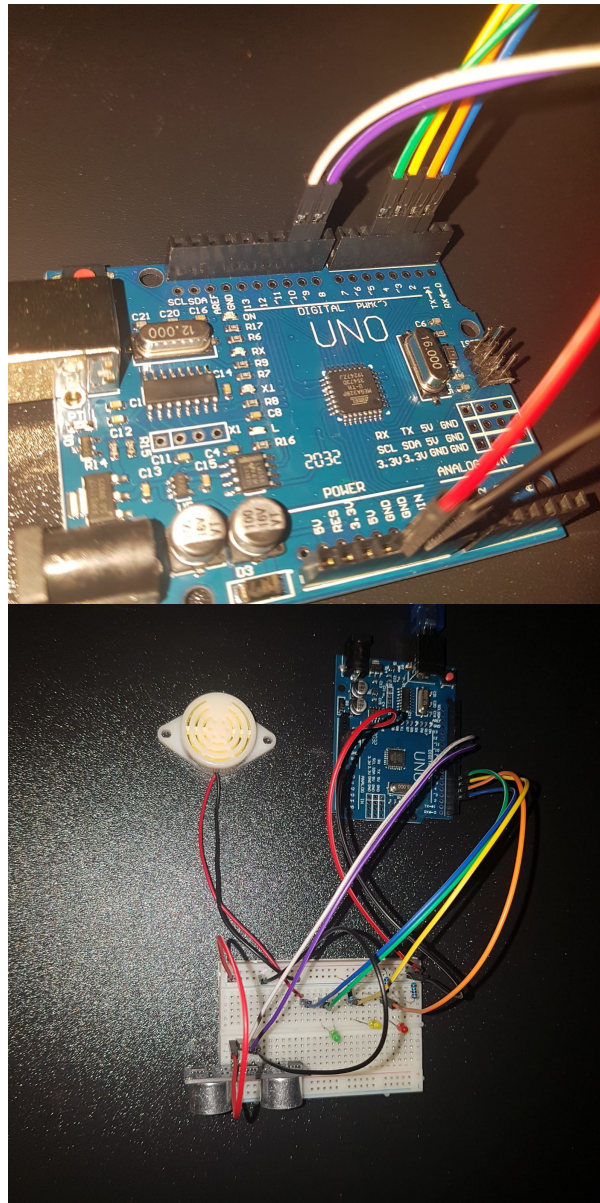
**v**- viteza sunetului, iar **t** - timpul de parcurgere al undelor de la senzor la obiect și înapoi, aflat cu ajutorul funcției *pulseIn()*

- dacă distanța este mai mică de 10 cm, atunci se aprinde doar LED-ul *roșu* și se activează alarma cu ajutorul funcțiilor *tone()* și *delay()*- pentru a produce sunet pentru 2,5s și *noTone()* pentru a face pauza între sunete
- dacă distanța este între 10 și 50 cm, atunci este aprins LED-ul *galben*, iar alarma devine mai rară, având o pauză mai mare între sunete și o frecvența mai puțin gravă decât alarma din zona roșie
- dacă distanța este mai mare decât 50cm, atunci nu se mai declanșează nicio alarmă și se aprinde LED-ul *verde*

## Rezultate Obținute

### Poze circuit:





Video funcționalitate: [Demo proiect](#)

Concluzii

Proiectul funcționează conform celor descrise mai sus. În funcție de distanța dintre senzor(mașină) și obstacol - distanță determinată prin intermediul pinului Echo al senzorului - circuitul are diferite comportamente, aprinzând anumite LED-uri sau declanșând alarme. Per total, a fost un proiect plăcut de realizat, fără creare de probleme și interesant de testat.

Download

[Rev\\_Parking\\_Sensor](#)

[Export to PDF](#)

Bibliografie/Resurse

- <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>
- <https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>
- <https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/>

[https://www.pcbway.com/blog/News/New\\_product\\_in\\_gift\\_shop\\_\\_HC\\_SR04\\_Ultrasonic\\_Ranging\\_Sensor\\_Module.html](https://www.pcbway.com/blog/News/New_product_in_gift_shop__HC_SR04_Ultrasonic_Ranging_Sensor_Module.html)  
[https://create.arduino.cc/projecthub/mudit\\_5246/arduino-car-reverse-parking-alert-system-bd518b?ref=tag&ref\\_id=alarm&offset=21](https://create.arduino.cc/projecthub/mudit_5246/arduino-car-reverse-parking-alert-system-bd518b?ref=tag&ref_id=alarm&offset=21)  
<https://www.arduino.cc/reference/en/language/functions/advanced-io/tone/>  
<https://www.arduino.cc/reference/en/language/functions/advanced-io/pulsein/>  
[https://create.arduino.cc/projecthub/Vijendra/arduino-based-collision-detection-warning-system-d1bee c?ref=search&ref\\_id=arduino%20uno&offset=49](https://create.arduino.cc/projecthub/Vijendra/arduino-based-collision-detection-warning-system-d1bee c?ref=search&ref_id=arduino%20uno&offset=49)

Export to PDF

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2021/avaduva/rev\\_parking\\_sensor](http://ocw.cs.pub.ro/courses/pm/prj2021/avaduva/rev_parking_sensor) 

Last update: **2021/06/02 02:17**