

# Breathalyzer

## Introducere

Proiectul constă în realizarea unui etilotest digital, care utilizează un senzor de alcool MQ-3 pentru a detecta nivelul alcoolului din aerul expirat. După apăsarea unui buton, senzorul începe să înregistreze date pentru un interval fix de timp, de exemplu, 5 secunde. Aceste date sunt procesate pentru a determina nivelul alcoolului și sunt afișate pe un display OLED.

Scopul principal al acestui proiect este de a oferi o modalitate simplă și eficientă de a determina dacă cineva este apt să conducă în siguranță. Prin detectarea nivelului de alcool din aerul expirat, proiectul ajută la prevenirea accidentelor cauzate de consumul de alcool înainte de a urca la volan.

Pornind de la experiența personală ca șofer, am constatat lipsa unui instrument portabil și accesibil pentru a testa nivelul de alcool înainte de a urca la volan. Ca șofer responsabil, am simțit nevoia de a avea o modalitate rapidă și precisă de a evalua propriul nivel de alcool, mai ales în situații în care consumul de alcool este incert sau în cazul în care am consumat alimente sau medicamente care pot afecta rezultatul testului. De asemenea, am observat că astfel de dispozitive sunt disponibile doar pentru autorități, iar lipsa lor pe piața generală m-a inspirat să dezvolt un astfel de dispozitiv pentru uz personal și pentru ceilalți șoferi responsabili. Astfel, am dorit să creez un etilotest digital care să ofere o soluție practică și accesibilă pentru evaluarea rapidă și precisă a nivelului de alcool înainte de a lua decizia de a conduce.

Un etilotest portabil este indispensabil pentru orice șofer responsabil, furnizând o modalitate rapidă și precisă de a evalua capacitatea de a conduce în siguranță.

## Descriere generală

**Senzorul MQ-3:** Acesta este un senzor de alcool care detectează concentrația de alcool din aer. După ce butonul este apăsat, senzorul începe să înregistreze datele pentru un interval de timp prestabilit.

**Display-ul OLED:** Afișează rezultatele testului, precum și alte informații utile, cum ar fi unele avertismente, în funcție de rezultat. Este interfața principală de comunicare cu utilizatorul.

**LED-ul RGB:** Schimbă culoarea în funcție de rezultatul testului. De exemplu, arată verde pentru un rezultat negativ și roșu pentru un rezultat pozitiv.

**Buzzer-ul:** Produce un sunet în funcție de rezultatul testului. De exemplu, emite un sunet scurt pentru un rezultat negativ și un sunet lung pentru un rezultat pozitiv.

**Mod de functionare:** După ce butonul este apăsat, senzorul MQ-3 începe să măsoare concentrația de alcool din aer. Datele detectate sunt apoi procesate de către microcontrollerul Arduino, iar rezultatul este afișat pe display-ul OLED. În același timp, LED-ul RGB și buzzer-ul sunt activate în funcție de încadrarea rezultatului în limitele admise. Astfel, utilizatorul primește informații clare și imediate despre aptitudinea sa de a conduce în siguranță.

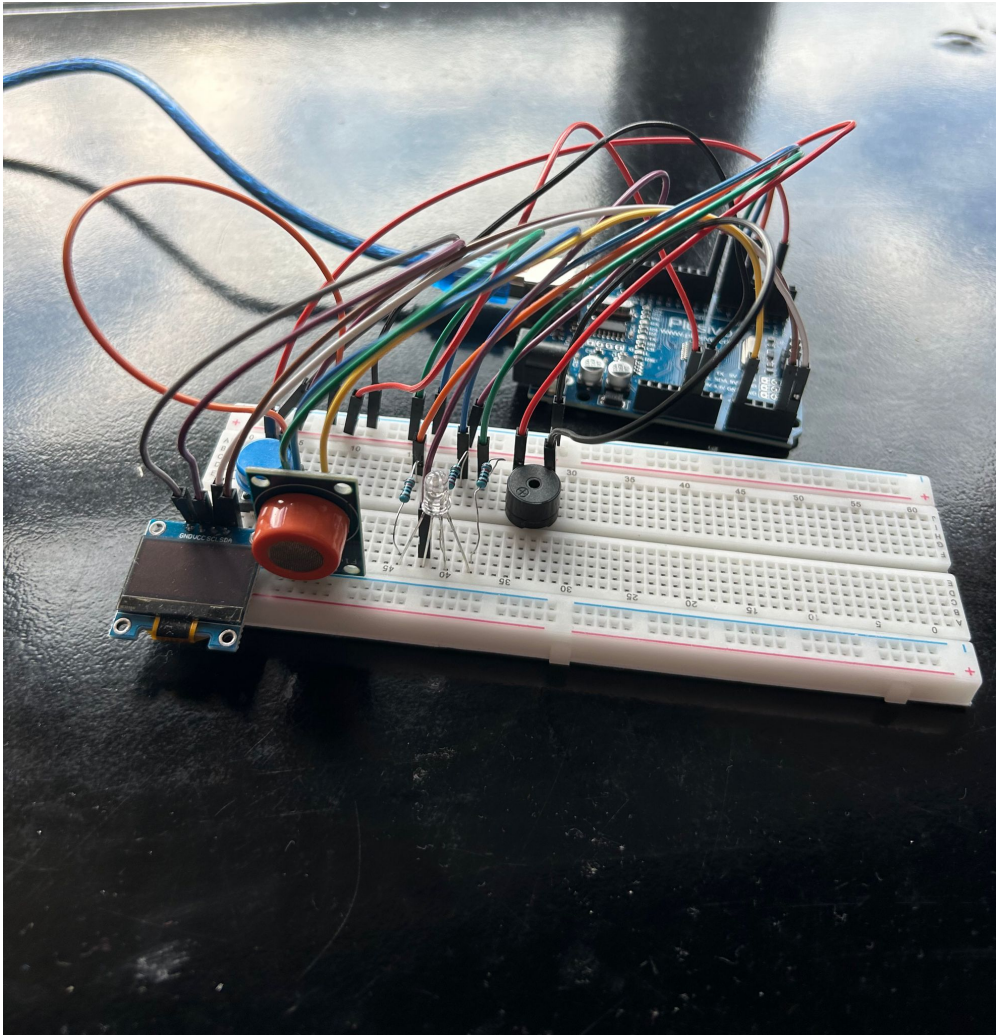


## Hardware Design

### Lista piese:

- Arduino UNO (ATmega328p și CH340)
- Senzor MQ-3
- Ecran OLED 0.96" I2C
- Push-button
- LED RGB
- Buzzer Pasiv 5 V
- Breadboard
- Jumper wires
- Rezistente: 220Ω





## Software Design

### Biblioteci folosite:

- Wire.h - folosită pentru comunicarea I2C cu afișajul OLED.
- Adafruit\_GFX.h - folosită pentru a desena forme și a afișa text pe ecranul OLED.
- Adafruit\_SSD1306.h - folosită pentru a controla și inițializa afișajul OLED SSD1306.

### Laboratoare folosite

- UART → folosit pentru a afișa informații de debug.
- TIMERE → folosit pentru a seta timpul de măsurare a nivelului de alcool.
- PWM → folosit pentru a controla buzzer-ul pasiv.
- I2C → folosit pentru ecranul OLED.

### Descrierea logicii și metodelor folosite

- **setup()** - folosit pentru initializarea componentelor (ecran OLED, buton cu rezistență internă de pull-up, led RGB, buzzer)
- **loop()** - aici este implementată toată logica programului. Inițial se afișează un mesaj de pregătire al senzorului deoarece acesta trebuie să se încălzească pentru a afișa valori corecte. Când timerul expiră, voi afișa pe ecran faptul că senzorul este gata și se poate începe măsurarea prin apăsarea

butonului. După ce este apăsător se va începe măsurarea timp de 7 secunde. După ce s-au terminat cele 7 secunde voi afișa ultima valoare măsurată și voi apela funcția `printResult()` pentru a declanșa acțiunile corespunzătoare în funcție de valoare.

- **`printResult()`**

1. Dacă valoarea este mai mică de 200, se va afișa 'You can drive.' pe ecran și se va face led-ul verde.
2. Dacă valoarea se află între 200 și 280, se va afișa 'Wait one hour.' pe ecran, led-ul va fi galben iar buzzer-ul va transmite sunete întrerupte.
3. Dacă valoarea se află între 280 și 350, se va afișa 'Wait a few hours.' pe ecran, led-ul va fi tot galben, iar buzzer-ul va transmite aceleași sunete întrerupte.
4. Dacă valoarea este mai mare de 350, se va afișa 'Do not drive!', led-ul va deveni roșu iar buzzer-ul va scoate un sunet constant de înaltă frecvență.

- **`printButtonPrompt()`** - funcție pentru a afișa mesajul că senzorul este gata de măsurare.
- **`printTitle()`** - funcție pentru a afișa numele 'Brealyzer' constant în partea de sus a ecranului.
- **`printWarming()`** - funcție pentru a afișa mesajul de pregătire a senzorului.
- **`printAlcohol()`** - funcție care afișează valoarea nivelului de alcool.
- **`readAlcohol()`** - funcție pentru citirea valorilor de la senzor și calibrarea lor.

## Rezultate Obținute

Demos:

- [Exemplu 1](#)
- [Exemplu 2](#)

## Concluzii

Prin combinarea componentelor hardware și software, am reușit să construiesc un dispozitiv capabil să detecteze nivelurile de alcool din respirație și să ofere recomandări corespunzătoare șoferilor.

În timpul dezvoltării, am întâlnit diverse provocări tehnice, cum ar fi gestionarea temporizărilor și a controlului buzzer-ului fără a bloca funcționarea sistemului. Cu toate acestea, prin aplicarea unor soluții eficiente și prin perseverență, am reușit să depășesc aceste obstacole și să obțin rezultatele dorite.

## Download

[breathalyzer.zip](#)

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

Resurse software:

- <https://ocw.cs.pub.ro/courses/pm/lab/lab1-2023>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab3-2023-2024>
- <https://ocw.cs.pub.ro/courses/pm/lab/lab6-2023-2024>

Resurse hardware:

- <https://www.sparkfun.com/datasheets/Sensors/MQ-3.pdf>
- <https://docs.arduino.cc/tutorials/generic/digital-input-pullup/>
- <https://docs.arduino.cc/learn/microcontrollers/analog-output/>
- <https://circuitdigest.com/microcontroller-projects/understanding-difference-between-active-and-passive-buzzer-with-arduino>

[Export to PDF](#)

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

Permanent link:  
<http://ocw.cs.pub.ro/courses/pm/prj2024/sseverin/diana.stanicoaea>



Last update: **2024/05/26 22:16**