

# PC Sensor Control

Autor: [Saceleanu Andrei-Iulian](#)

Grupa: 331CB

## Introducere

Scopul proiectului este de a permite executia unor comenzi asupra PC-ului, cu input-ul generat de cativa senzori sau componente, via un microcontroller.

Ideea de la care am plecat a fost cum sa scot cat de mult se poate de la un numar redus de senzori, intr-un mod care sa fie cat de cat interesant. Chiar daca aceleasi comenzi pe PC pot fi executate prin apasarea unor taste, proiectul meu incearca sa promoveze interactiunea non-tactila.

## Descriere generală

In functie de modul de operare selectat cu un buton, distanta data de un senzor ultrasonic (calibrat cu ajutorul unui senzor de temperatura) va fi folosita pentru a realiza diferite actiuni asupra PC-ului, precum: modificare luminozitate/volum, fast back/forward catre o anumita secunda intr-un video. Un mod aditional va fi realizat cu ajutorul unui fotorezistor, anume luminozitate adaptiva. Un LED RGB in blinking va indica modul de operare curent.

## Schema bloc



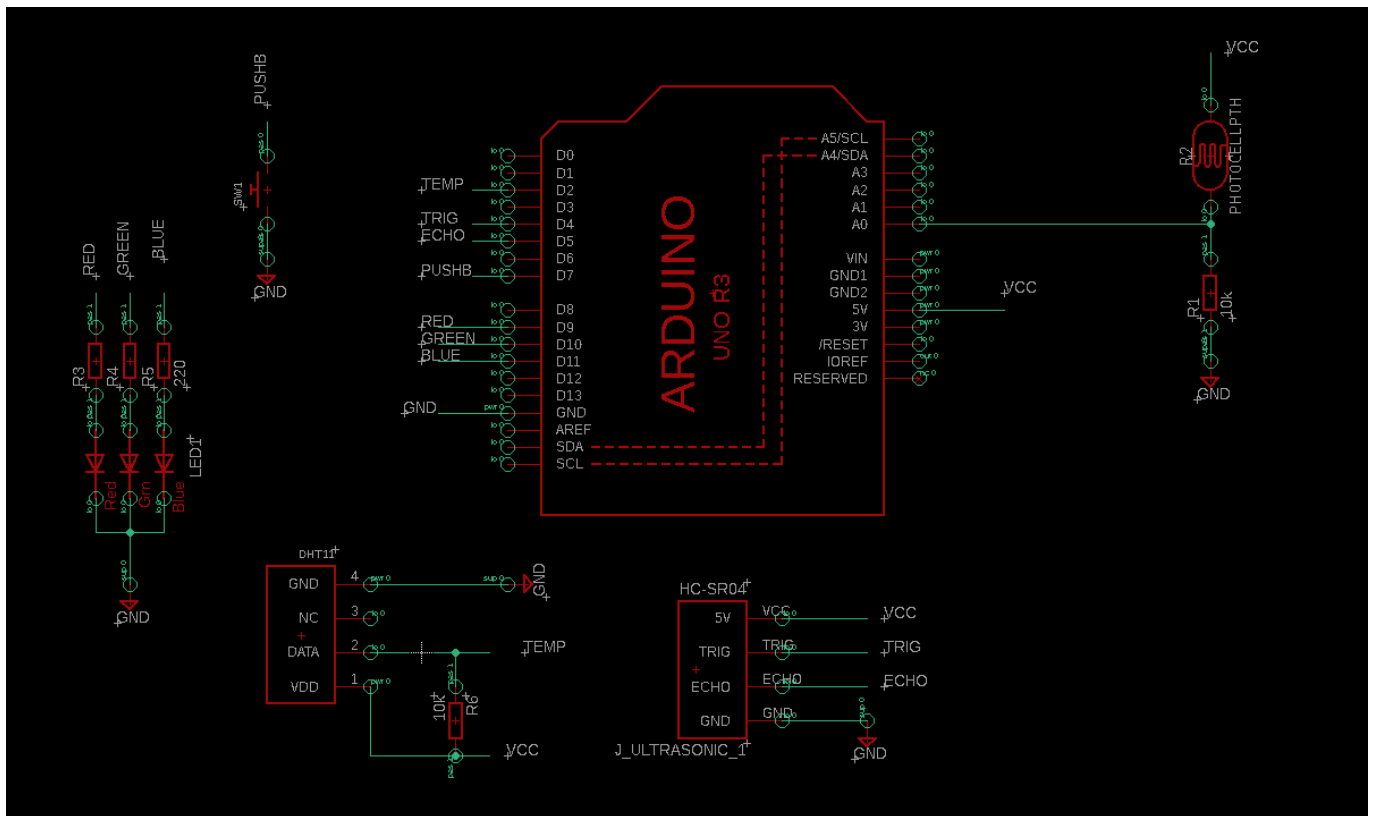
## Hardware Design

## Lista piese

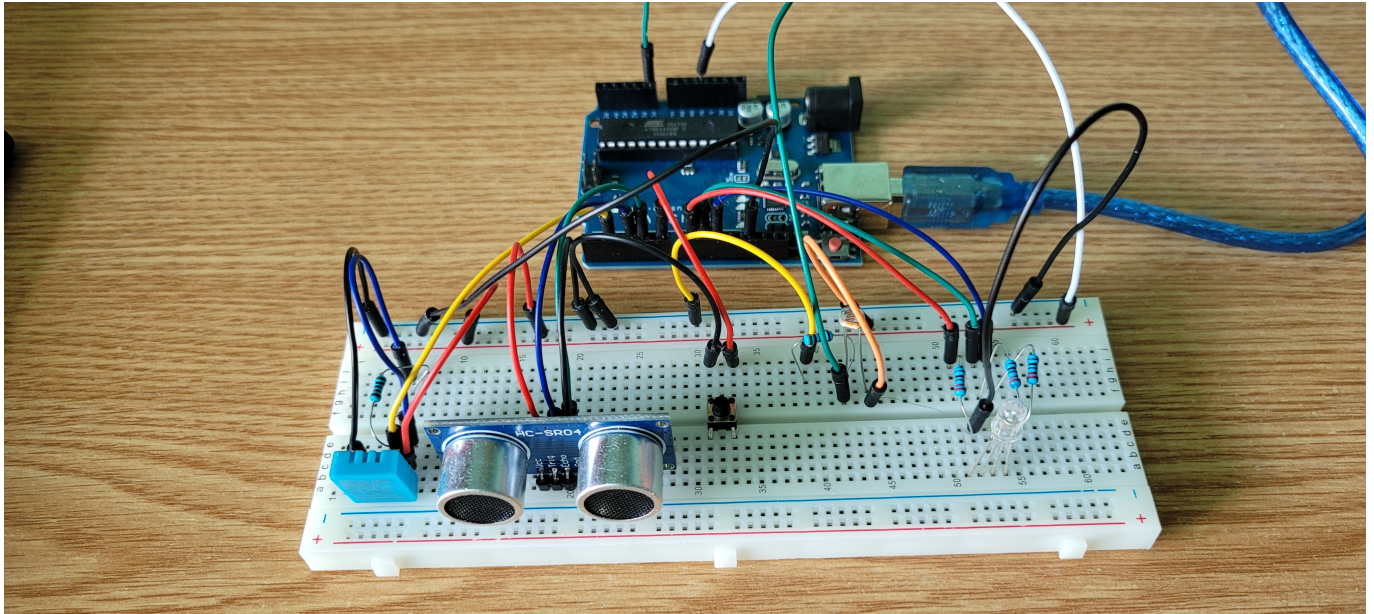
- Arduino UNO

- Breadboard
- Senzor ultrasonic HC-SR04+
- Rezistente,fire
- Push button
- Fotorezistor
- Senzor de temperatura si umiditate DHT11
- LED RGB catod comun

## Schema electrica



Asa arata construita:



## Software Design

### Cod microcontroller

Arduino IDE a fost folosit pentru scrierea/upload-ul codului.

S-au folosit bibliotecile NewPing.h (pentru senzorul ultrasonic) și DHT.h (pentru senzorul de temperatura și umiditate).

### Cod PC

Command runner-ul este implementat în Python.

Pentru interacțiunea cu Arduino, am utilizat modulul pyserial/serial. Calea către device-ul serial este prestabilită la un port Windows (COM3). Se citește linie cu linie comandă de pe interfață și se aplică operațiile corespunzătoare de actualizare ai parametrilor de sistem.

Pentru modificarea:

- luminozității: s-a folosit modulul **screen-brightness-control**
- volumului: s-a folosit modulul **pycaw** (specific Windows)
- timpului de redare video: s-a folosit standardul VLC, interacțiunea realizându-se prin modulul **python-vlc**

## Surse

Sursele si detalii aditionale se regasesc in repository-ul de git:

<https://github.com/andrei-saceleanu/proiectPM>

## Rezultate Obținute

Demo: <https://youtu.be/5ft1jckMolo>

## Concluzii

Proiectul reuseste sa transforme datele de la senzori in comenzi corespunzatoare pentru PC. In realizarea acestuia, consider ca am aprofundat si fixat notiunile prezentate la laborator + idei extra necesare pentru a interactiona cu HC-SR04+, etc. In mod clar, exista multiple posibile imbunatatiri si functii suplimentare care ar contura un sistem destul de versatil.

## Download

Cod git: [andrei\\_saceleanu\\_proiectpm-main.zip](#)

## Jurnal

- **21.04.2022**: creare initiala pagina
- **03.05.2022**: formatare continut si adaugare schematic
- **06-08.05.2022**: actualizare schematic + implementare hardware
- **11-13.05.2022**: definitivare documentatie

## Bibliografie/Resurse

PDF: [Export to PDF](#)

Cod: <https://github.com/andrei-saceleanu/proiectPM>

Software:

- <https://dronebotworkshop.com/hc-sr04-ultrasonic-distance-sensor-arduino/>

- <https://playground.arduino.cc/Code/NewPing/>
- <https://create.arduino.cc/projecthub/pibots555/how-to-connect-dht11-sensor-with-arduino-uno-f4d239>
- <https://www.geeksforgeeks.org/vlc-module-in-python-an-introduction/>
- <https://www.olivieraubert.net/vlc/python-ctypes/doc/vlc.MediaPlayer-class.html>
- <https://github.com/AndreMiras/pycaw>
- <https://www.codestudyblog.com/cs2112pyc/1221175124.html>
- OCW PM Labs

#### Hardware:

- <https://eepower.com/resistor-guide/resistor-types/photo-resistor/#>
- <https://dronebotworkshop.com/hc-sr04-ultrasonic-distance-sensor-arduino/>
- <https://learn.sparkfun.com/tutorials/pull-up-resistors/all>

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/dene/pc-sensor-control>



Last update: **2022/05/13 09:06**