

# Ventilator inteligent

- Autor: Calafeteanu Tudor-Alexandru
- Grupa: 336CC

Prezentare video : <https://youtu.be/zDpJHCvYeNk>

## Introducere

- Ventilatorul inteligent are viteza de rotație dependentă de temperatura camerei, fiind activ doar atunci când senzorul de proximitate detectează mișcare. Senzorul de proximitate poate fi montat după bunul plac, însă, din motive ergonomice, în prezentarea video a proiectului, senzorul a fost fixat pe breadboard. Temperaturile sunt calculate la un interval de timp prestabilit, fiind salvate pe un card microSD, realizând, astfel, un data logger. Datele sunt apoi introduse în Excel, care generează un grafic ce surprinde evoluția temperaturii camerei în timp.
- Scopul ventilatorului inteligent, față de unul normal, este să își schimbe singur viteza de rotație în funcție de temperatura camerei și să funcționeze doar atunci când detectează mișcare, pentru a eficientiza consumul de energie electrică.
- Am plecat de la ideea de a crea o dependență între procesul de ventilație și temperatura camerei.
- Utilitatea lui derivă din diferențierea sa față de un ventilator clasic de birou.

## Descriere generală



- Senzorul de temperatură, precum și cel de proximitate, transferă date către Arduino.
- Arduino controlează motorul prin PWM, folosind un driver de control al turației motorului. Controlul turației se realizează având în vedere datele primite de la cei doi senzori.
- Servomotorul este controlat direct de Arduino.
- Datele recepționate de Arduino sunt stocate într-un fișier aflat pe cardul SD, care se comportă ca un data logger pentru senzorul de temperatură.
- Cardul este citit de laptop, apoi se generează un grafic ce evidențiază fluctuația temperaturii în timp.

## Hardware Design

- **Listă de componente:**

1. Arduino UNO
2. Breadboard
3. Senzor de temperatură DHT11
4. Motor cu elice
5. Servomotor
6. Card microSD + adaptor
7. Senzor PIR HC-SR505
8. Modul sursă de alimentare + alimentator
9. Driver (1 diodă, 1 rezistență 1kOhm, 1 tranzistor NPN)
10. Fire

#### • Schema electrică



## Software Design

Mediul de dezvoltare folosit este Arduino IDE.

Bibliotecile externe folosite au fost:

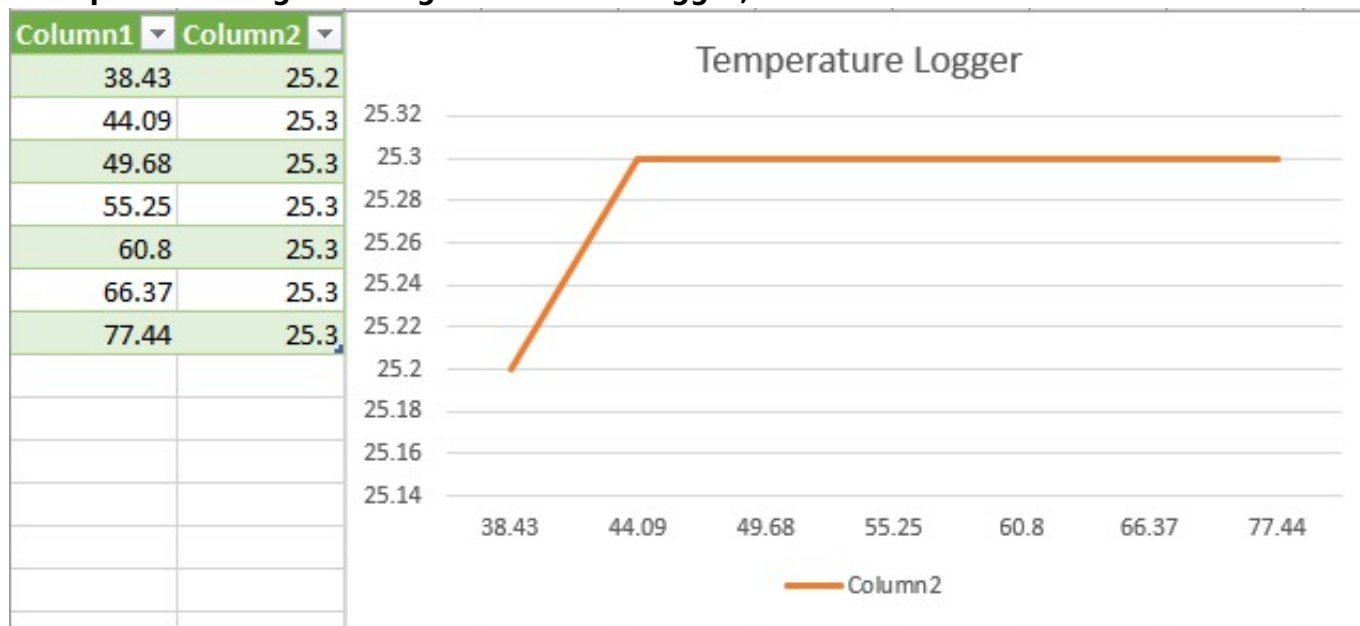
1. **SPI.h.h** (pentru adaptorul microSD)
  2. **Servo.h** (pentru Servomotor)
  3. **SD.h** (pentru cardul microSD)
  4. **DHT.h** (pentru modulul de temperatură DHT11)
- Detecția mișcării se face o dată după fiecare ciclu de mișcare a servomotorului. Un ciclu reprezintă trecerea servomotorului din poziția 0, la poziția 180, apoi din nou la poziția 0 (1 ciclu = 0° → 180° → 0°). Dacă senzorul PIR nu detectează mișcare (output 0) și starea ventilatorului este *running* (flag-ul stopMoving este setat pe 0), se incrementează un counter (counterMovement). Dacă acest counter ajunge la valoarea 3, atunci ventilatorul își schimbă starea din *running* în *blocked*, revenind la starea *running* abia când detectează, din nou, mișcare. De fiecare dată când senzorul detectează mișcare, iar ventilatorul este activ, counter-ul se resetează.
  - Mișcarea motorului DC se face prin PWM, mapând range-ul de temperaturi care poate fi detectat de senzorul de temperatură (0-50), la range-ul de viteze (100 - 255). Astfel, elicea se va învârti cu o viteză direct proporțională cu temperatura camerei.
  - După fiecare ciclu de mișcare a servomotorului, se scrie valoarea temperaturii detectate, împreună cu timpul trecut, în secunde, de la pornirea ventilatorului, într-un fișier *logger.txt*. Punctele rezultate sunt apoi trecute în Excel pentru generarea graficului.

## Rezultate Obținute

## Poză proiect de ansamblu:



## Exemplu rezultat generare grafic din data logger, folosind Excel:



## Concluzii

A fost prima oară când am lucrat cu hardware. Am întâmpinat unele dificultăți în conectarea unelor componente pe breadboard, însă au fost rezolvate rapid. Consider o lucrare foarte interesantă pentru un student care nu a mai avut tangențe cu proiecte în Arduino, am învățat lucruri noi și sunt mândru de rezultatul final.

## Download

Sursa: [ventilatorinteligent.rar](#)

Documentația în format [PDF](#)

## Bibliografie/Resurse

- <https://www.instructables.com/Controlling-a-Motor-an-NPN-Transistor-on-the-Ardui/>
- <https://www.youtube.com/watch?v=-AvF2TsB2GI>
- [https://youtu.be/8TnrozK\\_384](https://youtu.be/8TnrozK_384)
- <https://youtu.be/5Dp-XatLySM>
- <https://www.arduino.cc/reference/en/language/functions/math/map/>

From:

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

Permanent link:

[http://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/ventilator\\_intelligent](http://ocw.cs.pub.ro/courses/pm/prj2021/apredescu/ventilator_intelligent)



Last update: **2021/05/30 16:44**