

# Vape

Nume: Butilca Rares-Dumitru

Grupa: 335CA

## Introducere

Dupa cum ii spune numele, acest proiect prezinta constructia unui Vape (aka tigara electronica, aka aia care scoate mult "fum"). Legat de ce face, e simplu, scoate vapori cand apesi pe un buton. Prezinta 2 moduri de functionare ce vor fi descrise mai jos: control in temperatura si control in putere.



Pentru cine nu stie, scopul unui vape este sa iti ofere o alternativa fumatului clasic, avand diverse arome si eliminand din inconvenientele tigarilor "analogice" precum mirosul neplacut sau faptul ca nu este permis fumatul in cladiri publice.

Am ales acest proiect, deoarece sunt pasionat de vape-uri, avand si cativa ani de "practica in domeniu", in sensul ca detin astfel de jucarii de ceva timp si m-am gandit ca ar fi distractiv sa incerc sa confectionez si eu unul.

Acest proiect este util pentru cei care vor sa incerce exemple de DIY, doar pentru experienta in sine.

**DISCLAIMER:** Nu incercati sa vapati(fumati) din asa ceva :) Acest proiect este facut doar pentru fun.

Daca vreti sa vapati, cumparati-va un vape de la producatori.

## Descriere generală

Vape-ul va fi controlat folosind un Joystick din care iti poti seta temperatura/puterea de lucru, modul de functionare(temp vs power) si, prin apasare, poti da drumu la incalzirea rezistentei care iti va vaporiza lichidul pe care il vei "fuma"(inhala).

Valoarea temperaturii/puterii se controleaza miscand joystick-ul pe verticala, iar schimbarea modului de functionare se face prin miscarea acestuia pe orizontala.

Modul de control in temperatura iti va controla curentul prin rezistor astfel incat vaporii sa aiba temperatura dorita, pe cand cel in putere iti va seta puterea pe care o va consuma bateria pentru rezistor. Pentru controlul in temperatura este folosit un senzor de curent pentru a masura rezistenta rezistorului din atomizor.

Controlul intern al vape-ului va fi facut de catre un Arduino UNO care va comanda prin PWM 3 tranzistori MOSFET pusi in paralel pentru controlul tensiunii medii de pe rezistor. Microcontrollerul va fi conectat la joystick pentru a primi comenzi si la un display pentru a se afisa valoarea curenta a puterii/temperaturii cat si modul in care se afla vape-ul.

Intregul ansamblu va fi alimentat prin intermediul a 2 acumulatori 18650.

## Schema bloc



## Hardware Design

### Lista de piese folosite:

- atomizor Exceed D22 <rezistor inclus>
- conector 510 - folosit pentru legarea atomizorului la baterie
- 2 baterii 18650 + suport pentru conectarea in serie
- placuta care integreaza 3 tranzistori AO3400 si rezistente de 10K in baza
- senzor de curent GY169 bazat pe INA169
- joystick
- display cu comunicatie I2C: 1604 + adaptor I2C

- arduino UNO
- conector alimentare arduino UNO

## Schema electrica



Observatii schema:

- Joystickul din aceasta schema a fost inlocuit de 2 potentiometre si de 1 buton. Pinii joystickului sunt cei care ies din chenarul albastru.
- Piesa folosita pentru display in aceasta schema nu este fidela modelului de display folosit in proiect.
  - Pinii de interes sunt doar cei care au fost marcati cu label.
  - Display-ul folosit in proiect are 64 de caractere.

## Software Design

### Mediu de dezvoltare

- Ca mediu de dezvoltare am folosit Arduino IDE.
- Pentru controlul display-ului cu I2C am folosit biblioteca LiquidCrystal\_I2C  
<https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>

### Initializare

#### EEPROM

Partea de initializare include o citire din EEPROM unde sunt stocate permanent detalii de configurare, precum: putere maxima, minima admisa, puterea default, etc. De asemenea, tot aici se gasesc si valorile TFR care vor fi descrise mai jos.

#### Temp control

In general, pentru a putea masura temperatura unui rezistor, se pot folosi 2 metode:

- TCR: calculul  $R(T)$  se face pe baza unui coeficient  
<https://eepower.com/resistor-guide/resistor-fundamentals/temperature-coefficient-of-resistance/#>
- TFR: calculul  $R(T)$  se face pe baza unor factori sub forma de raport  $R(T)/R(20C)$   
[https://www.stealthvape.co.uk/tfr\\_csv\\_files\\_for\\_our\\_wires/](https://www.stealthvape.co.uk/tfr_csv_files_for_our_wires/)

Prima metoda este buna in cazul materialelor cu un profil  $R(T)$  cat de cat liniar pe gama de temperatura 20C - 350C (zona in care se vazeaza). Pentru altfel de materiale, este de preferat o lista de TFR pe care se pot face interpolari. Cu cat lista are mai multe valori, cu atat precizia este mai buna.

Materialele rezistentelor pe care temp controlul are relevanta sunt Ni200, Ti si SS si altele similare. Atomizorul cumparat pentru acest proiect are, din nefericire :(, rezistente din NiCr80, un aliaj al carei valori pentru rezistenta este inerta in raport cu temperatura. Astfel, pentru a masura corect temperatura este nevoie de un senzor foarte fin si de folosirea tehnicii bazata pe TFR.

Neavand un senzor foarte fin, am incercat sa rezolv problema prin folosirea tehnicii TFR cu valori din 20 in 20 C (<https://www.steam-engine.org/tcr>). Cu toate acestea, nu am obtinut rezultate prea fericite, asa ca, pentru cine doreste sa repete aceasta experienta, recomand folosirea unor rezistente confectionate din materialele prezentate mai sus.

ATENTIE: Nu confundati Ni200 cu NiCr80. Nickelul este un material foarte bun pentru temp control, pe cand aliajul de Nickel si Crom strica tot :)

## Alte configurari de initializare

Alte configurari realizate sunt initializarea diferitelor module precum cel de I2C, comunicatiile analogice si digitale cu joystickul si senzorul, etc. Am ales, pentru partea de configurari, sa folosesc cat mai mult lucrul cu registrele.

Avand in vedere ca displayul este format dintr-o componenta de conversie I2C la display 1604 si displayul in sine, am folosit biblioteca externa LiquidCrystal\_I2C pentru o mai simpla comunicare cu acesta.

## Loop

In bucla de rulare a programului, se fac urmatoarele verificari / actiuni:

- In cazul in care butonul este apasat - o simpla citire digitala - marcam faptul ca acesta este apasat si marcam si faptul ca se da drumul la curent pe rezistor.
- In cazul in care nu este apasat - ne asiguram ca este marcat acest lucru, precum si faptul ca nu voltajul va sta pe 0V.
- In cazul in care este marcat faptul ca se da drumul la voltaj, se va calcula voltajul respectiv  $U = \sqrt{P * R}$ , pentru power mode. Pentru temp mode se va aborda o varianta mai simpla in care voltajul se seteaza la o valoare default atunci cand temperatura este cu 10C mai mica decat valoarea dorita si se va pune pe 0V atunci cand trece cu 10C de aceeaasi valoare. Dupa ce se fac

aceste calcule, se va seta voltajul la valoarea respectiva prin folosirea unui semnal pseudo-analogic (PWM) care va deschide si inchide tranzistoarele periodic.

- Daca este marcat faptul ca nu se va da drumu la voltaj, acesta se seteaza la 0V.
- Daca butonul a fost apasat pentru mai mult de 10s, se va taia legatura la tensiune.
- Se va masura rezistenta: daca avem marcat faptul ca se da putere pe rezistenta se calculeaza dupa formula  $R = U / I$ (masurat), altfel se va seta voltajul la 1V, se va calcula rezistenta cu aceeasi formula, dupa care se va seta iar tensiunea la 0V.
- Daca rezistenta atinge valorile 0(SHORT) sau >9.99(HIGH RESISTANCE), se va semnala acest lucru pe display si nu va fi permisa conectarea rezistentei la tensiune.
- Daca joystickul se va misca pe axele xOy, se vor modifica configurarile de putere/temperatura sau modul de lucru.
- Displayul va fi mereu updatat cu valorile pentru putere/temp, modul de lucru, valoarea rezistentei la temperatura camerei si eventualele mesaje precum HIGH RESISTANCE.

## Masurarea rezistentei

In mod empiric am observat un comportament neasteptat la senzorul de curent folosit, avand salturi destul de mari la tensiune constanta. Astfel, nu parea sa masoare foarte bine curentul care trecea prin rezistor. Avand in vedere ca a fost folosit un semnal PWM pentru a genera tensiunea pseudo-analogica pe rezistor, senzorul nu reusea sa masoare curentul mediu cum trebuie.

Pentru calcularea rezistentei la temperatura camerei, putem folosi aceasta metoda, dar pentru calcularea temperaturii rezistorului, nu este o idee buna sa ne bazam pe aceasta metoda, neavand destula precizie.

## Legenda

- R = rezistenta
- T = temperatura
- P = putere
- U = tensiune

## Concluzii

Imbunatatirile care pot fi aduse acestui prototip, astfel incat acesta sa functioneze in parametri optimi, ar putea fi:

- folosirea unei surse comandate care sa genereze o tensiune analogica - ar fi rezolvat problemele intampinate la masurarea rezistentei cu ajutorul senzorului
- folosirea unui atomizor care accepta rezistente din materialele: Ni200, Ti sau SS - ar fi rezolvat problema masurarii temperaturii cu precizie.

ATENȚIE: Tensiune / semnal analogic nu se refera la PWM.

## Download

vape.zip

write\_eeprom\_for\_vape.zip

Aici puteti gasi:

- Proiectul in Arduino IDE pentru Vape
- Proiectul in Arduino IDE pentru scrierea memoriei EEPROM

## Bibliografie/Resurse

- <https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>
- <https://eepower.com/resistor-guide/resistor-fundamentals/temperature-coefficient-of-resistance/#>
- [https://www.stealthvape.co.uk/tfr\\_csv\\_files\\_for\\_our\\_wires/](https://www.stealthvape.co.uk/tfr_csv_files_for_our_wires/)
- <https://www.steam-engine.org/tcr>

[Export to PDF](#)

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

Permanent link:  
<http://ocw.cs.pub.ro/courses/pm/prj2023/alucaci/vape>



Last update: **2023/05/29 22:40**