

# Sistem alimentare combustibil

## Introducere

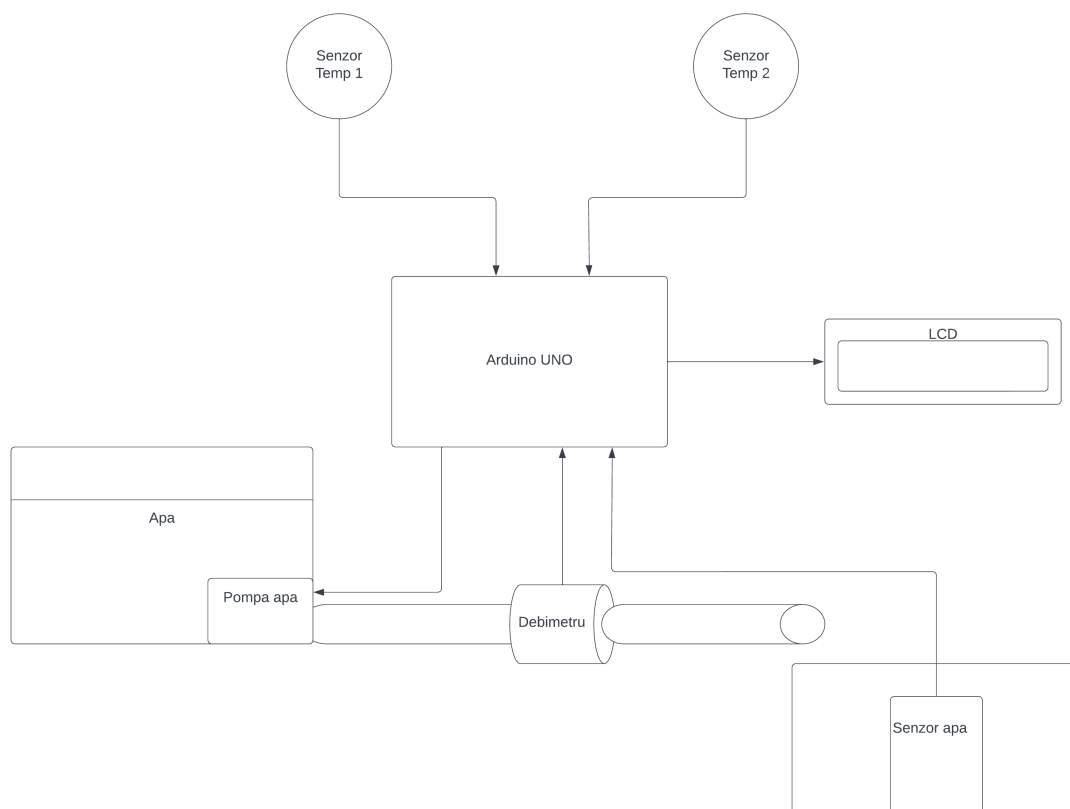
Prezentare concept proiect:

- Ideea proiectului este de a realiza un sistem de alimentare pentru un vehicul cu motor cu ardere internă. Sistemul folosește 2 senzori de temperatura, unul pentru mediul exterior și unul pentru temperatura motorului, astfel în funcție de valorile furnizate de cei 2 senzori combustibilul o sa fie eliminat din rezervor de o pompa de apa în mod diferit. Pentru o verificare a funcționalității sistemului exista și un debimetru conectat la pompa, dar și un senzor de apa pentru a verifica dacă lichidul furnizat de pompa ajunge la destinație. Pentru o monitorizare mai ușoară a întregului sistem exista și un lcd care va afișa modul în care se afla pompa de apa și valoarea indicată de debimetru.
- Scopul sistemului este de a muta combustibilul din rezervor către punctul în care trebuie folosit într-un mod eficient și în funcție de nevoia motorului cu ardere internă, adică o sa avem o rata de transfer variabila.
- Initial am pornit de la fântâna de apa a motanului meu, fântână care foloseste o simpla pompa de apa de 12 v și câteva tuburi pentru a direcționa apa, iar în lipsa altei idei de proiect am încercat sa mă gândesc cum ar arata o evoluție aplicată pentru aceasta fântână de apa în domeniul auto și cum urmăresc Formula 1 mi-am adus aminte de faptul ca motoarele din Formula 1 nu folosesc mereu aceeași cantitate de combustibil pentru ca au nevoie uneori de un amestec diferit în funcție de mai multi factori, astfel am ajuns la concluzia ca pot influenta comportamentul unei pompe de apa cu PWM după datele furnizate de senzori de temperatura și sa verific în timp real comportamentul pompei de apa cu un debimetru și un senzor de ploaie.
- Nevoia de a muta un lichid dintr-un loc în altul cu un debit variabil și în funcție de datele obținute de niște senzori nu este o aplicație strict auto și se poate aplica în alte scenarii cu ușurință precum industria agricolă unde plantele pot avea nevoie de cantități diferite de apa în funcție de temperatura și sezon, astfel un proiect precum al meu se poate adapta ușor unor noi cerințe pentru a furniza o soluție eficientă atât energetic, cât și financiar. Un alt aspect legat de proiect îl reprezintă scalabilitatea, având în vedere conectivitatea de care dispune un Arduino se pot adauga ușor noi senzori sau un nou Arduino care sa se folosească de datele utilizate de acest sistem pentru a dicta comportamentul altui sistem, rămânând în sfera auto un Arduino care controlează un termostat ar avea nevoie de datele furnizate de senzorii de temperatura și debitul pe care îl are pompa de combustibil.

## Descriere generală

\* Senzorul 1 de temperatura va furniza temperatura pentru mediul exterior, iar senzorul 2 de

temperatura o sa fie un senzor de temperatura de tip sonda și va furniza temperatura variabila a unui motor cu ardere interna sau a apei dintr-un recipient pentru simulare. Arduino-ul va prelua informațiile de la cei 2 senzori și va procesa datele pentru a alege unul din cele 3 moduri de control pentru pompa de apa. Denumirea modului de funcționare o sa fie afișată de LCD și Arduino-ul va controla prin PWM pompa cu posibilitatea de a efectua modificări pe baza datelor preluate de la debimetru, iar aceste date se vor afișa pe LCD. Senzorul de apa funcționează ca un sistem de diagnoza, de la 10 secunde de la pornirea pompei de apa se verifica dacă apa a ajuns la destinație prin datele furnizate de acest senzor și în cazul în care nu exista apa după 10 secunde se trece într-un mod de sleep pentru a proteja pompa întrucât asta înseamnă ca nu mai exista lichid în rezervor pentru pompa.



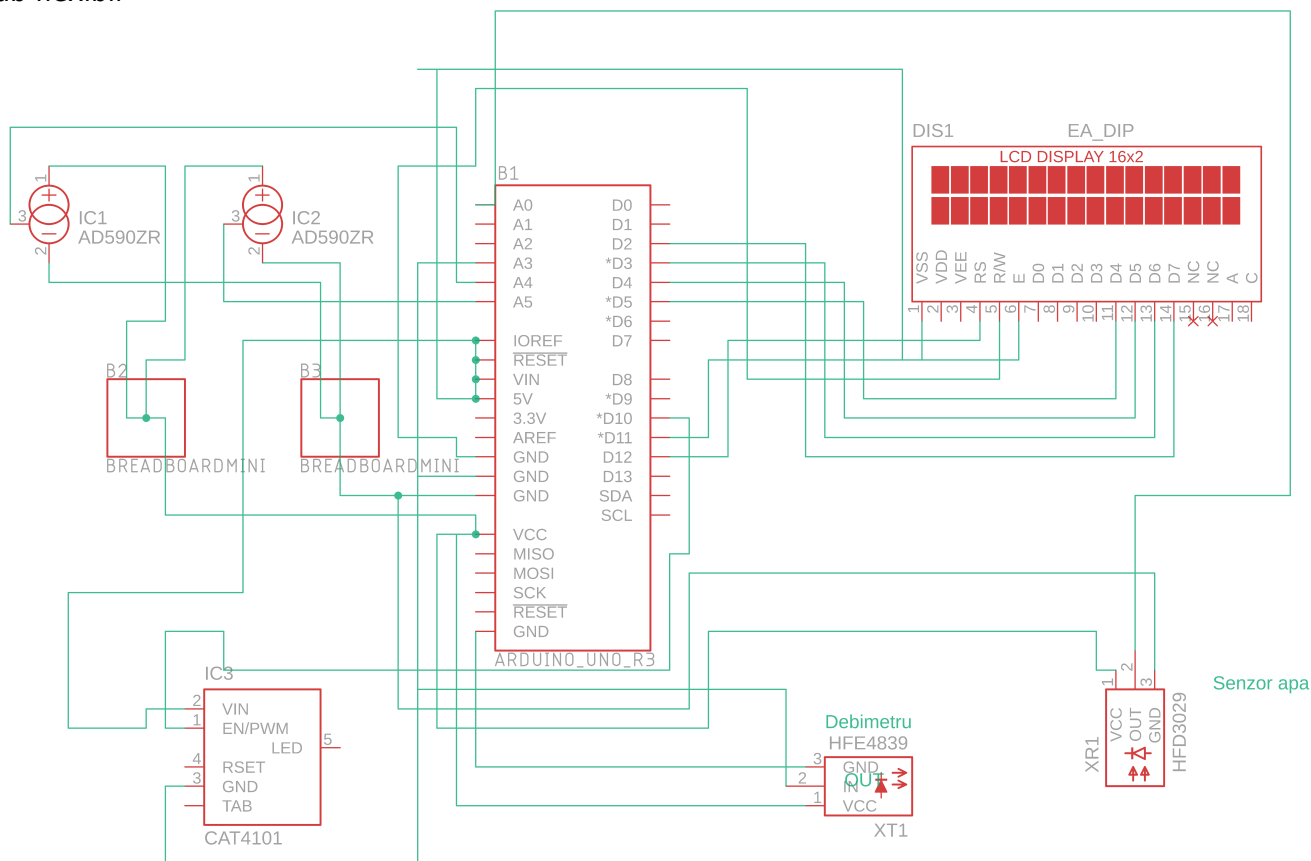
\*

## Hardware Design

Componente și schema electrica EAGLE:

- Arduino Uno
- Breadboard
- Modul senzor de temperatura
- Senzor de temperatura MAX6675
- Debimetru
- LCD

- Senzor picaturi de ploaie
- 2 \* Recipient apa
- Tub flexibil



## Software Design

Descrierea codului aplicației (firmware):


- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuți să le implementați
- (etapa 3) surse și funcții implementate

## Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

## Concluzii

## Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume\_student** (dacă este cazul). **Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru\_alin**.

## Jurnal

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

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

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

Permanent link:  
[http://ocw.cs.pub.ro/courses/pm/prj2022/german/sistem\\_alimentare\\_combustibil](http://ocw.cs.pub.ro/courses/pm/prj2022/german/sistem_alimentare_combustibil) 

Last update: **2022/05/11 21:31**