

# Tea Machine

**Nume:** Bobocu Alexandra-Florentina

**Grupa:** 331CAa

## Introducere

Prezentarea pe scurt a proiectului:

- Proiectul presupune realizarea unui automat de pregătire a ceaiului. Se coboară un pliculeț de ceai în cana cu apă, iar utilizatorul decide timpul potrivit lăsării pliculețului în cană. Când timpul a expirat, pliculețul de ceai iese din cană și revine în poziția inițială.
- Scopul lui este să țină pliculețul de ceai în apă o perioadă de timp specificată.
- Ideea de la care am pornit a fost pregătirea ceaiului perfect, fără a trebui să mai verific ceasul ca să știu când pot scoate pliculețul de ceai din cană; acest aparat face asta automat.
- Utilitate: Întotdeauna mi-am dorit ca pliculețul de ceai să stea în apă pentru o perioadă potrivită de timp, astfel încât ceaiul să fie bun la gust. Cu toate acestea, foarte des mi se întâmplă să las pliculețul de ceai mai mult decât este necesar, astfel că acest automat vine în ajutorul iubitorilor de ceai care se regăsesc în situația mea.

## Descriere generală

La pornirea automatului, se afișează un mesaj sugestiv, după care utilizatorul apasă butonul din dreapta (galben) pentru a alege numărul de minute destinat pregătirii ceaiului. Se va afișa pe ecranul LCD textul: "Select mins: X", unde X reprezintă numărul de minute alese. Acesta nu poate depăși 5 minute, fiind limita maximă recomandată pentru lăsarea ceaiului la infuzat.

După selectarea timpului, utilizatorul apasă butonul din stânga (albastru), moment în care se acționează servomotorul, care va coborî brațul de care este prins pliculețul de ceai. Din acest moment, începe procesul de cronometrare al pregătirii ceaiului, afișându-se textul "Preparing Tea...", alături de timpul rămas, exprimat în minute și secunde. Când timpul s-a scurs, servomotorul va ridica brațul în poziția inițială și pe ecran se va afișa un mesaj sugestiv: "Tea is ready!".

De menționat este că utilizatorul trebuie să aleagă mai întâi cât timp va lăsa pliculețul în apă, după care va începe cronometrarea propriu-zisă. Dacă interschimbă ordinea butoanelor, nu se va întâmpla nimic.

- Pentru a detecta în timp real apăsarea butoanelor, voi folosi întreruperi externe.
- Controlez poziția servomotorului prin semnale PWM.
- Mesajele ce trebuie afișate sunt transmise către ecranul LCD prin protocolul I2C.

## Schema bloc:



## Hardware Design

### Lista componentelor utilizate:

- 1 X Arduino UNO
- 1 X Micro servomotor SG90
- 1 X Ecran LCD 1602 I2C
- 1 X Baterie 9V
- 2 X Butoane
- Fire de legătură (mamă-tată, tată-tată)
- 1 X Breadboard (dimensiune mică)
- 2 X Rezistențe de 10k

### Schema electrică Tinkercad:



### Schema electrică Eagle:



## Software Design

### Biblioteci utilizate:

- Servo.h
- LiquidCrystal\_I2C.h

## Funcții implementate:

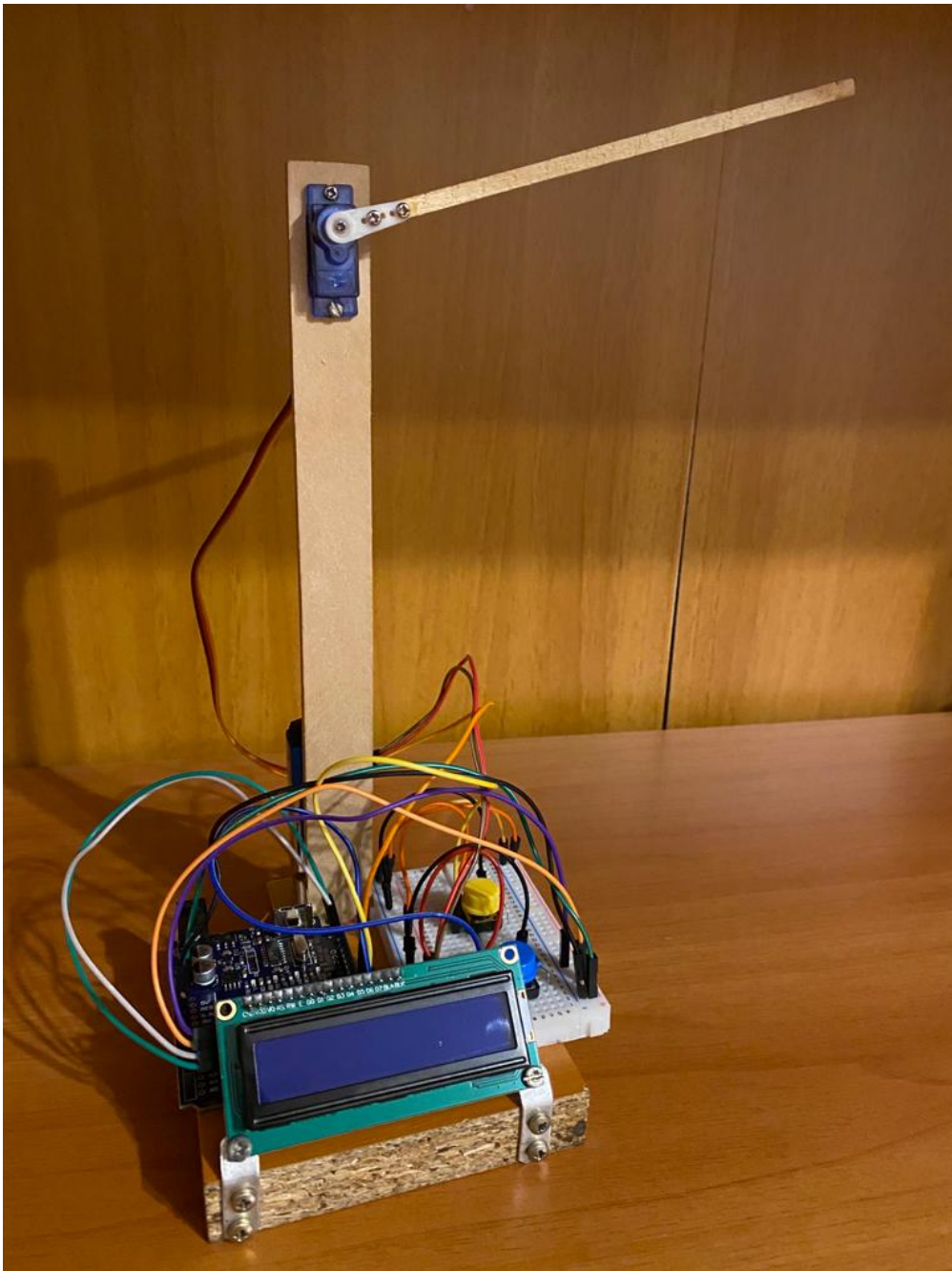
- **void setup():**
  - Configurez pini I/O pentru cele 2 butoane și servomotor;
  - Atașez o întrerupere externă pentru butonul care setează numărul de minute necesar pregătirii ceaiului;
  - Atașez obiectul de tip servo la pinul corespunzător și stabilesc poziția inițială în care să stea brațul cu pliculețul de ceai;
  - Inițializez ecranul LCD;
  - Apelez funcția **initial\_lcd\_message()** pentru afișarea unui mesaj de întâmpinare, urmată de ștergerea textului de pe ecran, pentru a face loc următoarelor mesaje
- **void loop():**
  - Prin apelarea funcției **read\_button()**, verific dacă butonul prin care se setează numărul de minute a fost apăsat, moment în care apelez o altă funcție **add\_minutes()** care se ocupă de logica incrementării minutelor;
  - Verific dacă butonul prin care începe cronometrarea timpului efectiv de pregătire al ceaiului a fost apăsat, ținând cont de ordinea apăsării celor 2 butoane: dacă butoanele au fost apăstate în ordinea recomandată, atunci valoarea unghiului la care stă brațul servomotorului se va decrementa de la valoarea *servo\_high\_position* la *servo\_low\_position*, cu un delay *servo\_speed*, pentru a nu brusca servomotorul. Odată ce brațul servomotorului este în poziția dorită, se apelează funcția **start\_timer()** care începe cronometrul
- **void display\_lcd(String message1, String message2, int delay\_value):**
  - funcție auxiliară care afișează 2 mesaje pe rânduri diferite
- **void initial\_lcd\_message():**
  - funcția apelează **display\_lcd()** pentru afișarea mesajului de întâmpinare
- **void button\_interrupt\_handler():**
  - ISR pentru tratarea întreruperii declanșate de apăsarea butonului *set\_button*, pe frontul crescător al ceasului. Handler-ul este folosit în cadrul funcției **attachInterrupt()** din biblioteca Arduino;
  - Variabila *interrupt\_process\_status* de tip *volatile* este folosită în cadrul acestei funcții pentru a semnaliza dacă a avut loc sau nu întreruperea. Inițial, considerăm că întreruperea nu a avut loc, însă în momentul în care se constată că butonul a fost apăsat, întreruperea externă are loc, motiv pentru care actualizăm corespunzător variabila
- **bool read\_button():**
  - Dacă întreruperea a fost declanșată, verific dacă butonul *set\_button* a fost apăsat sau nu;
  - Dacă da, marchez trecerea butonului în starea "HIGH". Voi considera că a avut loc o singură apăsare de buton în momentul în care trece o perioadă *debounce* de 10 milisecunde. În acest caz, marchez trecerea butonului pe starea "LOW", semn că nu mai este apăsat iar rutina de tratare a întreruperii este gata;
  - Se returnează *switched*, dacă butonul a fost apăsat, sau *!switched*, în caz contrar
- **void add\_minutes():**
  - Funcție care îi permite utilizatorului să aleagă numărul de minute necesar pregătirii ceaiului;
  - Numărul de minute se incrementează doar dacă nu se depășește un prag prestabilit de 5 minute
- **void start\_timer():**
  - Funcție care începe numărătoarea inversă. Afișez pe ecran un mesaj alături de timpul care trece. Pentru a simula faptul că a trecut o secundă, folosesc `delay(1000)`;
  - Verific dacă numărul de secunde este nenul. În caz afirmativ, le decrementez. Altfel, verific dacă numărul de minute este nenul. Dacă da, decrementez numărul de minute și resetez numărul de secunde la 59;
  - Dacă atât minutele, cât și secunde au ajuns la 0, înseamnă că timpul a expirat și ceaiul este gata de servire. În acest caz, se acționează servomotorul. Valoarea unghiului la care stă brațul

servomotorului se va incrementa de la valoarea *servo\_low\_position* la *servo\_high\_position*, cu un *delay servo\_speed*;

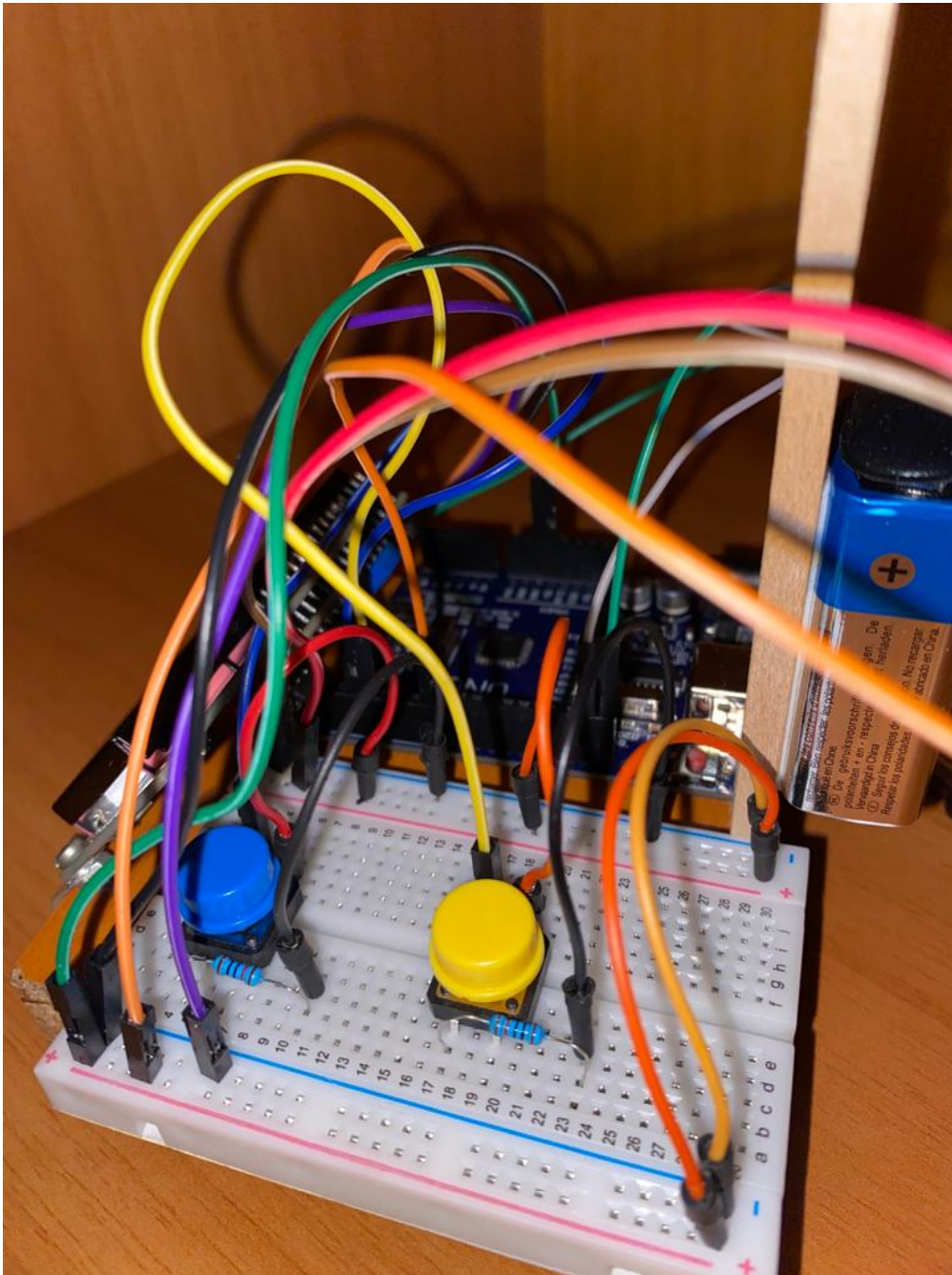
- Se afișează un mesaj care să îl atenționeze pe utilizator că ceaiul este gata. Mesajul va rămâne pe ecran până la o viitoare refolosire a aparatului.

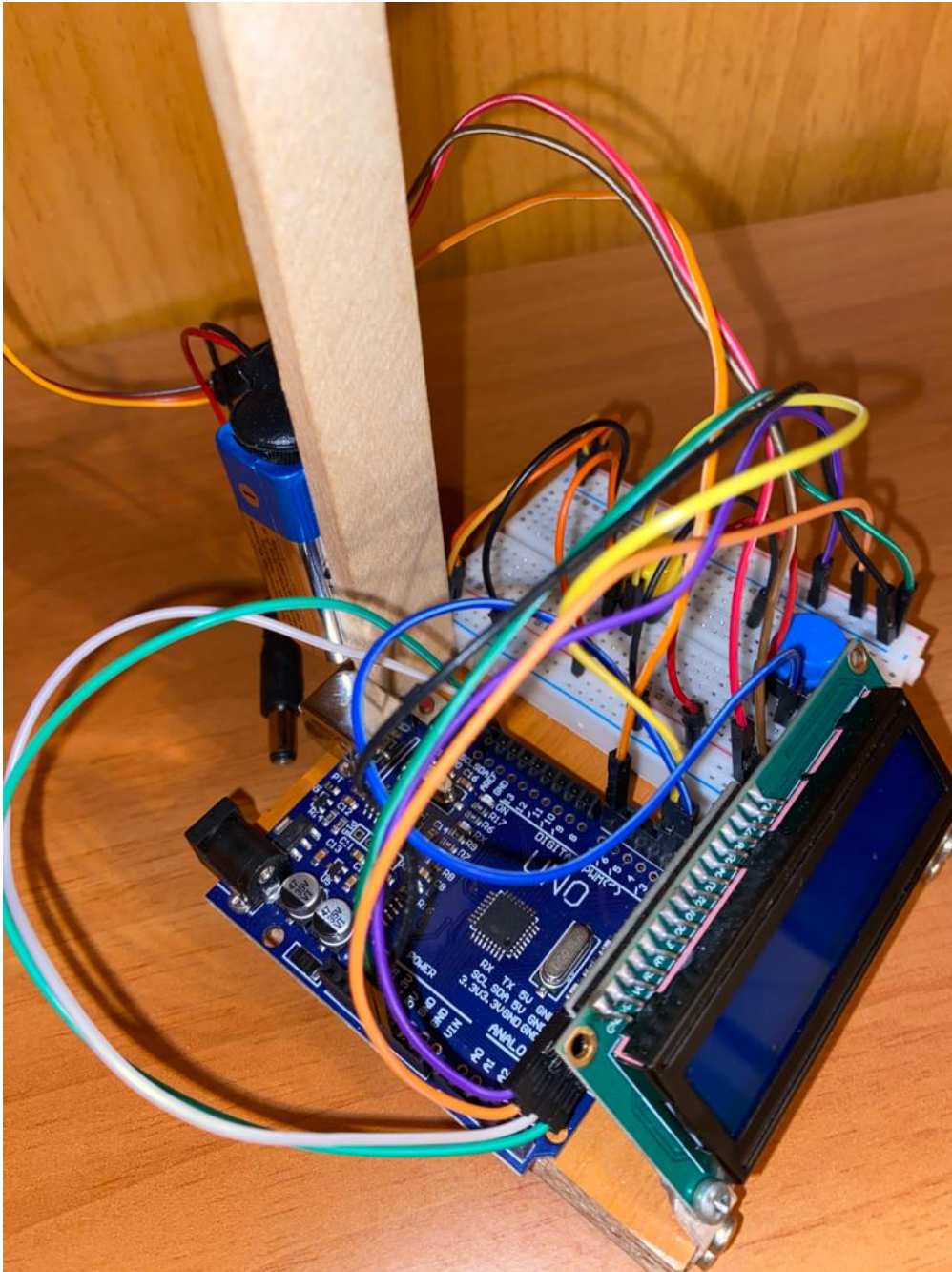
## Rezultate Obținute

### Aparatul în forma finală:

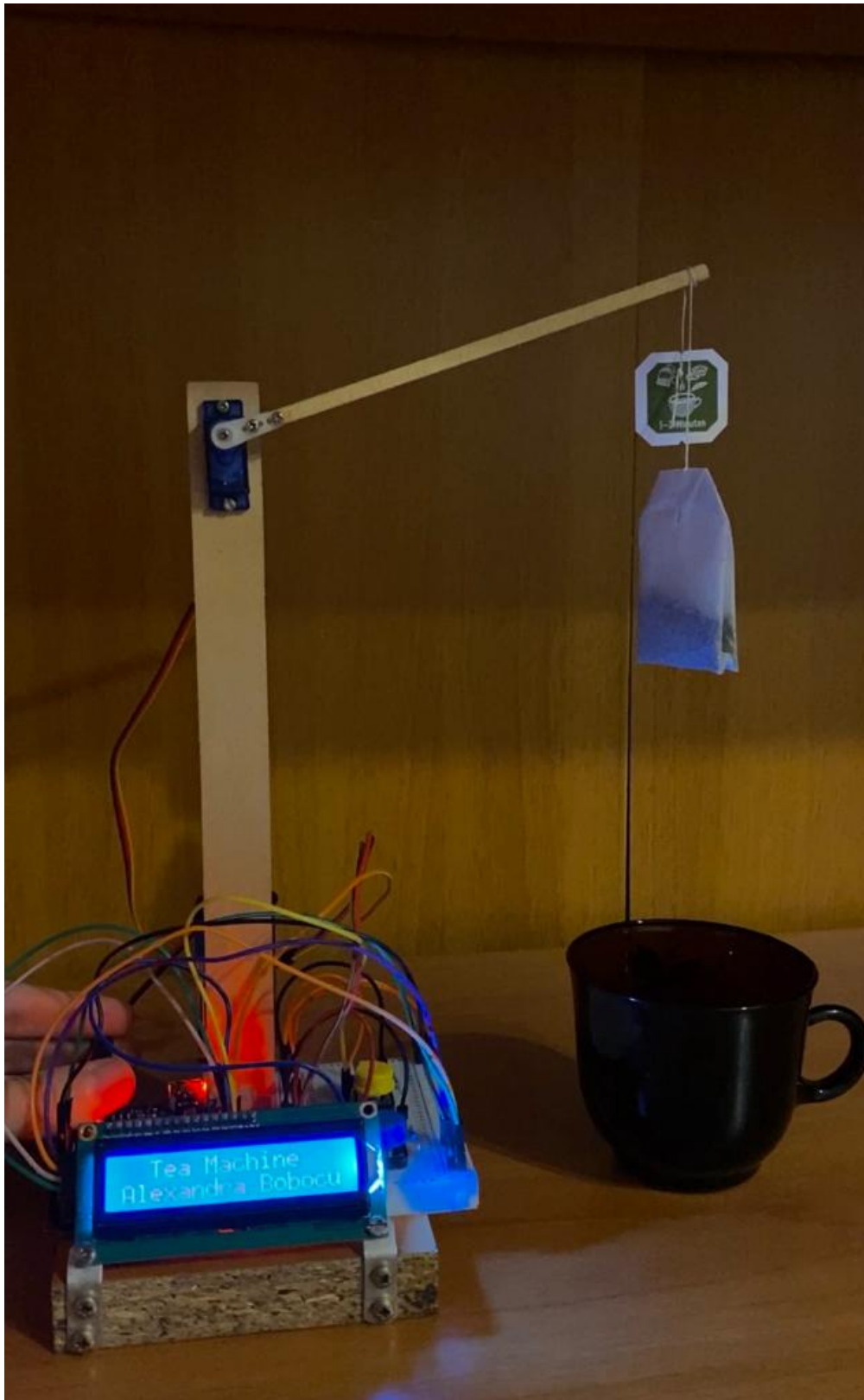


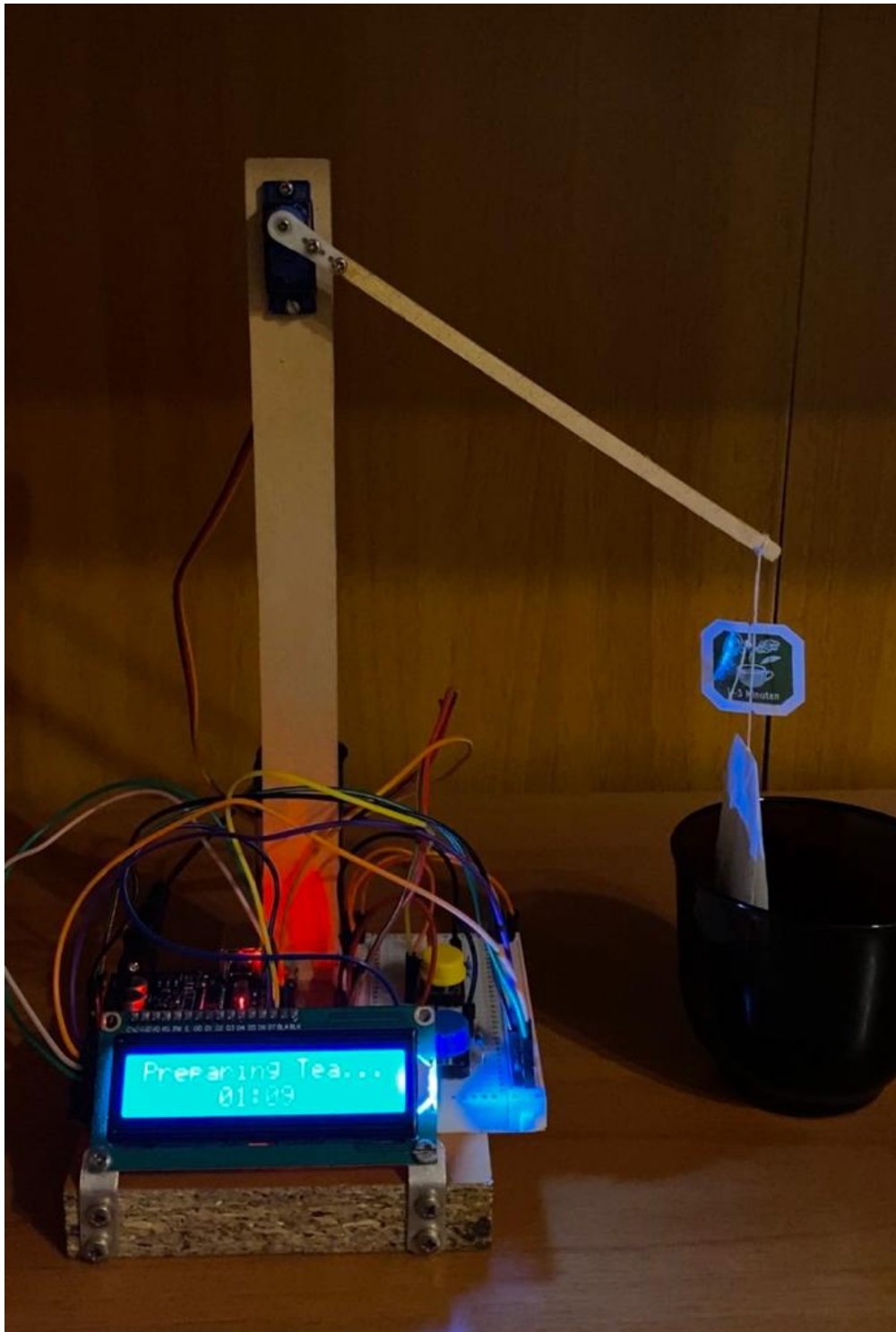
# Circuitul:





## Aparatul în timpul utilizării:






## Demo - 2 minute:

[https://www.youtube.com/watch?v=XJ00SjzfAnQ&ab\\_channel=AlexandraBobocu](https://www.youtube.com/watch?v=XJ00SjzfAnQ&ab_channel=AlexandraBobocu)

## Concluzii

Prin intermediul acestui proiect, am realizat cât de plăcut poate fi să creezi propriile obiecte "inteligente", ce pot fi utile în viața de zi cu zi cu ajutorul unei plăcuțe Arduino. 

Dezvoltarea acestui aparat s-a dovedit a fi o experiență interesantă, de pe urma căreia am dobândit cunoștințe și din zona de Hardware, pe care am reușit să le împletesc cu cele de Software, înțelegând cum comunică componentele între ele și testând funcționalitatea fiecăreia în parte, dar și a întregului ansamblu.

## Download

Arhiva conține codul sursă tea\_machine.ino: [tea\\_machine.zip](#)

## Jurnal


07.05.2023 - documentație inițială

16.05.2023 - update documentație & completare Hardware Design cu schemele electrice

22.05.2023 - completare Software Design

25.05.2023 - încărcarea arhivei ce conține poze cu rezultatele obținute și codul

## Bibliografie/Resurse

- Realizare schemă bloc: <https://app.diagrams.net/>
- Realizare schemă electrică: <https://www.tinkercad.com/>
- Arduino IDE, versiunea 2.1.0: <https://www.arduino.cc/en/software>
- Achiziționare componente: <https://cleste.ro/>
- Folosire ecran LCD cu modul I2C integrat:  
<https://www.diyengineers.com/2021/01/07/learn-how-to-use-the-1602-lcd-with-arduino/>
- Folosire buton: <https://docs.arduino.cc/built-in-examples/digital/Button>
- Folosire servomotor: <https://www.instructables.com/Arduino-Servo-Motors/>
- Sursă de inspirație proiect : <https://www.youtube.com/watch?v=V8ePvT5OjaE>

[Export to PDF](#)

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

Permanent link:  
[http://ocw.cs.pub.ro/courses/pm/prj2023/avaduva/tea\\_machine](http://ocw.cs.pub.ro/courses/pm/prj2023/avaduva/tea_machine) 

Last update: **2023/05/29 20:02**