

# Bunn-care

Vrabiuta Catalin-Constantin 331CC

## Introducere

Acest dispozitiv are grija de iepurasul tau, atunci cand tu esti plecat mai multe zile de acasa. El are grija ca animalutul tau sa aiba o portie de mancare zilnica, sa ii umple bolul cu apa cand e nevoie si sa iti traseze date despre ambientul in care traieste, cum ar fi luminozitatea si temperatura, toate datele se transmit prin intermediul conexiunii Wifi la o baza de date cu care poti interactiona din intermediul unei aplicatii.

## Descriere generală

Dispozitivul foloseste mai multe mecanisme pentru a se asigura ca iepurasul este in siguranta, aceste mecanisme fiind :

1. Mecanismul pentru mancare este constituit dintr-un motor DC controlat de un driver L298, care ridica bolul in partea de sus a custii, un motor stepper controlat de un driver ULN2003, ce roteste mecanismul responsabil de portionarea mancarii, dupa ce mancarea a cazut in bol, bolul este lasat jos pentru ca iepurasul sa se bucure de masa.
2. Pentru a reimplina apa aparatul are un servo ce este conectat la o tija pe care se afla un senzor de nivel al apei si un furtun. Furtunul este conectat la o pompa de apa, pompa care este conectata la un recipient mare de apa potabila. Cand senzorul de nivel al apei detecteaza ca iepurasul nu mai are apa el transmite un semnal ce porneste pompa de apa si o opreste atunci cand nivelul este optim. Pompa de apa este controlata cu ajutorul unui releu cu 1 canal.
3. In varful custii mai exista un senzor DHT11 si un fotorezistor, cu ajutorul acestora analizam temperatura, umiditatea si intensitatea luminii din incapere.
4. Utilizatorul se poate folosi de un ecran si cateva butoane pentru a interactiona cu dispozitivul local, fara a fi nevoie de conexiunea Wifi.

## Schemă bloc



# Hardware Design

Lista de piese:

1. Arduino Uno WiFi rev2
2. Driver L298N
3. Driver ULN2003
4. Modul releu 2 canale
5. Motor DC 3-6V
6. Stepper motor
7. Pompa apa 3-6V
8. Servomotor SG90 180 grade
9. Fotorezistor
10. Senzor DHT11
11. Senzor nivel apa
12. Ecran LCD1602 cu modul I2C
13. Butoane
14. Alimentator priza 9v 3A

## Schema Hardware:



## Alegerea pinilor:

1. Driverul LN298N are nevoie de un port pwm pentru a putea controla viteza motorului, asa ca am ales portul 3 iar celelalte 2 porturi le-am ales sa nu fie pwm, adica 2 si 4.
2. Servomotorul avea nevoie de un port pwm pentru a functiona asa ca am ales portul 5.
3. Senzorul de atingere foloseste un port pwm desi nu are nevoie de unul, era singurul port digital ramas liber in afara de 0 si 1.
4. Senzorul de temperatura si umiditate foloseste un pin digital, nu are nevoie de pwm.
5. Driverul pentru motorul stepper foloseste 3 porturi pwm (9, 10, 11) si portul 8, el are nevoie de 3 porturi pwm.
6. Releul cu 2 canale foloseste 2 pini digitali pentru directionarea curentului pe fiecare canal. La un canal este conectat servo-ul si senzorul de nivel al apei, pentru a le putea dezactiva atunci cand nu este nevoie de ele, iar la celalalt canal este conectata pompa de apa ce primeste curent in momentul in care senzorul de nivel este pornit si transmite ca nu este destula apa in bol.
7. Senzorul pentru nivelul apei foloseste un pin analog(A0) pentru a putea interpreta cu ajutorul ADC-ului daca se afla sau nu suficienta apa in bol.
8. Fotorezistorul este conectat la un pin analog(A1) pentru a putea analiza nivelul de luminozitate, folosind ADC-ul. Astfel putand determina nivelul de luminozitate ce face iepurasul confortabil.
9. Pe portul A2 momentan sunt 2 butoane ce transmit semnale diferite fiind conectate la un divizor de tensiune. Am facut acest lucru pentru a nu irosi porturi, pe viitor posibil sa cresc pana la 5 butoane pe acest port.
10. Ecranul LCD comunica prin protocolul I2C, astfel el foloseste porturile speciale de comunicare, SDA si SCL.

# Software Design

## Mediu de dezvoltare Arduino IDE

### Biblioteci folosite:

- Servo.h
- ThreeWire.h
- RtcDS1302.h
- Wire.h
- LiquiCrystal\_I2C.h
- DHT.h

**Surse:** \*<https://github.com/cata11v1/Bunn-care>

### Cod

Codul este destul de simplu, am folosit biblioteci pentru ecran, modul de timp, servomotorul si senzorul de umiditate si temperatura. Am 2 butoane care sunt conectate la un singur port analogic cu ajutorul unui divizor de tensiune, astfel citesc ce se transmite pe portul analogic si stiu ce trebuie facut. Senzorul de nivel al apei si fotorezistorul sunt conectate tot pe porturi analogice astfel pot interpreta cum vreau datele citite. Motoarele DC sunt conectate cu ajutorul unui driver LN298N, astfel incat unui motor ii controlez viteza cu ajutorul scrierii analogice pe un port digital, PWM. Pentru dozatorul de mancare am scris secvente scurte de mers fata spate pentru a agita mancare astfel sa cada in bol. Pompa de apa este controlat de un canal al releului, iar alimentarea servomotorului si a senzorului sunt conectate de celalalt canal, astfel mai intai alimentez servo si senzorul care analizeaza daca trebuie pornita pompa. De-a lungul utilizarii am adaugat si alte mesaje pe ecran, iar daca putonul de mancare este tinut mai mult apasat poate aparea pe ecran timpul curent.


## Rezultate Obținute

<https://youtu.be/lhSwD9haPoo?feature=shared>

## Concluzii

Mi-a placut, mi-am adus aminte de robotica din liceu, am realizat ca pot automatiza aproape orice din casa cu cateva piese arduino. Pe viitor as mai face si un proiect care sa imi ude plantele si unul in care sa am toate luminile din casa smart, sa se inchida si sa deschida la diferiti stimuli. Am realizat ca un motor stepper nu este deloc puternic si ca este foarte greu sa faci un motor dc precis. Am pierdut cam o zi in incercarea de dezvoltare de un mecanism bun pentru dispenserul de mancare.

## 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

- 06.05 - Infaptuirea planului si datul de comenzi pentru piese
- 15.05 - Au venit piesele si am le-am pus in functiune, sa vad cum se folosesc
- 19.05 - Am realizat mecanismul de ridicare al bolului, urmeaza cel de mancare
- 22.05 - Aparen motorul stepper nu e destul de puternic si am pierdut o zi incercand sa il folosesc, am renuntat si am facut cu un motor dc si alt tip de mecanism
- 24.05 - Finalizat si sistemul pentru apa
- 26.05 - Am terminat partea soft si rezolvatul de buguri (aparent unui servomotor nu trebui sa ii tai curentul si imediat cum il pornesti sa il pui sa se miste, il lasi 2 secunde si dupa merge)

## 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/prj2024/fgul/catalin.vrabiuta> 

Last update: **2024/05/27 13:11**