

Dozator de apa

Proiect D.E.M.

Realizat de Cotoc Daniel și Dobra Mihai
Facultatea de Automatică si Calculatoare, UPT

Scopul lucrării

1

Scopul acestui proiect este de a crea un dozator automat de apă care să ofere o soluție eficientă, igienică și precisă pentru umplerea controlată a recipientelor cu apă.

2

Unul dintre obiectivele principale ale dozatorului de apă este să faciliteze obținerea rapidă a cantităților dorite de apă, eliminând nevoia de a turna manual apă dintr-un recipient în altul.

3

Dozatorul de apă are și rolul de a asigura o utilizare economică a apei, evitând risipa și eliminând supra umplerea recipientelor.

Etapele de realizare

Cercetare și proiectare:

În această etapă, s-au identificat nevoile și cerințele utilizatorilor și s-a realizat un design general al dozatorului de apă, luând în considerare aspecte precum dimensiuni, materiale și interfețe de control.

Asamblare hardware:

Pentru construcția dozatorului de apă, s-a utilizat o placă Arduino ca sistem central de control, care a fost conectată la senzorii ultrasonici pentru a măsura nivelul de apă din rezervor și pentru a detecta prezența recipientului.



Etapele de realizare

Programare software:

S-a dezvoltat codul necesar în limbajul Arduino pentru a asigura funcționalitatea dozatorului de apă. Acesta a inclus algoritmi de măsurare și control, precum și gestionarea interacțiunii cu utilizatorul prin intermediul afișajului LED.

Testare și optimizare:

S-au efectuat teste riguroase pentru a verifica funcționalitatea și performanța dozatorului de apă. Pe baza rezultatelor obținute, s-au realizat ajustări și optimizări pentru a asigura un sistem eficient și precis.

A.

PARTEA SOFTWARE

Software-ul dozatorului de apă a fost dezvoltat în limbajul Arduino, oferind o interfață de comunicare între componente și un control precis al funcționalității.

Principalele funcții ale software-ului includ măsurarea nivelului de apă în rezervor utilizând datele furnizate de senzorii ultrasonici, detectarea prezenței recipientului și determinarea capacității acestuia, controlul pompei de apă pentru umplerea controlată a recipientului și afișarea stării de umplere prin intermediul afișajului LED.

Prezentarea codului folosit:

```
#define ECHO_SENS1 A2 // iesire semnal senzor ultrasonic
#define TRIG_SENS1 A3 // intrare semnal senzor ultrasonic
#define ECHO_SENS2 A5 //iesire semnal senzor ultrasonic
#define TRIG_SENS2 A4 // intrare semnal senzor ultrasonic
#define W_LED_R 3 // led de functionare a aparatului
#define W_LED_G 5 // led de functionare pompa activa
#define W_LED_B 6 // led albastru
#define BUZZ 4 // pin pt buzzer
#define GLASS_ECHO_SENS A0 // iesire senzor pahar
#define GLASS_TRIG_SENS A1 //intrare senzor pahar
#define PUMP 2 // pin pompa apa

LiquidCrystal lcd(7, 8 , 9, 10, 11, 12);//pentru conectarea ecranului LCD

// UNITATE DE MASURA: centimetrii
// EROARE SENZOR: 3mm = 0.3cm

long duration; // citire sezor
float distance; // distanta de la senzor la nivelul apei
float distance2; // distanta de la senzorul cand se umple paharul
int volume; // volum actual de apa in ml si vol anterior

const int min_dist = 3; // distanta minima pana cand e umplut paharul
const int dist_glass = 7; // distanta de activare cand e paharul pus in zona de umplere
const int min_water = 3; // minimul de apa la care se activa rezervor gol
const int bottle_height = 23; //cm inaltimea sticlei
const int bottle_width = 7.50; //cm latura sticlei (patrat)
const float sens_dist = 2; // cm distanta la care e amplasat senzorul
const float sens_error = 0.3; // cm, eroarea senzorului
const float water_lvl; //cm de apa din sticla
```

Selectarea pinilor de
pe placa Arduino

Atribuire valori
măsurate

Prezentarea codului folosit:

Verificare distanța
de la senzor la
nivelul apei

Afișare cantitate de
apă din rezervor

```
        son = readUltrasonicDistance(TRIG_SENS1, ECHO_SENS1); // citim senzorii
        distance = duration / 58; // calculam distanta
        water_lvl = bottle_height - (distance - sens_dist); // cantitate actuala de apa
        volume = bottle_width * bottle_width * water_lvl; // volumul de apa in ml (convertire din cm in ml)
        // volumul = aria * inaltime (cm cub = ml)
        // aria = 1 * 1;

        // verificare nivel de apa

        if(water_lvl <= min_water)
        {
            lcd.setCursor(0,1);
            lcd.print(" REZERVOR GOL ");
            analogWrite(W_LED_R, LOW); //oprire LED ROSU
            tone(BUZZ, 262, 150);
            delay(2000);
            noTone(BUZZ);
        }
        else if (water_lvl < bottle_height - sens_error)
        {
            analogWrite(W_LED_R, HIGH); //PORNIRE LED ROSU
            lcd.setCursor(0,1);
            lcd.print(" ");
            lcd.print(volume);
            lcd.print(" ml ");
        }
        else
        {
            lcd.setCursor(0,1);
            lcd.print(" REZERVOR PLIN");
            analogWrite(W_LED_R, HIGH); //PORNIRE LED ROSU
```

Prezentarea codului folosit:

```
while ((distance <= dist_glass) && (water_lvl > sens_error + min_water) && (distance2 > min_dist))
{
    analogWrite(W_LED_R, LOW); // oprire led rosu
    lcd.setCursor(0,0);
    lcd.print("  ALIMENTARE  ");
    lcd.setCursor(0,1);
    lcd.print("  RECIPIENT  ");
    analogWrite(W_LED_G, HIGH); // LED VERDE FUNCTIONARE POMPA
    delay(500);
    digitalWrite(PUMP, HIGH); // pornim pompa
    delay(150);

    duration = readUltrasonicDistance(TRIG_SENS1, ECHO_SENS1); // citim senzorii
    distance = duration / 58; // calculam distanta
    water_lvl = bottle_height - (distance - sens_dist); // cantitate actuala de apa

    duration = readUltrasonicDistance(GLASS_TRIG_SENS, GLASS_ECHO_SENS); // citim sensorul
    distance = duration / 58;

    duration = readUltrasonicDistance(TRIG_SENS2, ECHO_SENS2);
    distance2 = duration / 58;

}
digitalWrite(PUMP, LOW); // oprim pompa
delay(100);
analogWrite(W_LED_G, LOW); //oprim led
analogWrite(W_LED_R, HIGH); // PORNIM LED ROSU - APARAT FUNCTIONAL
lcd.setCursor(0,0);
```

Dacă este detectat
un pahar și
rezervorul conține
apa va porni
pompa

Dacă nu mai este
apa în rezervor, sau
dacă s-a umplut
paharul, pompa se
va opri

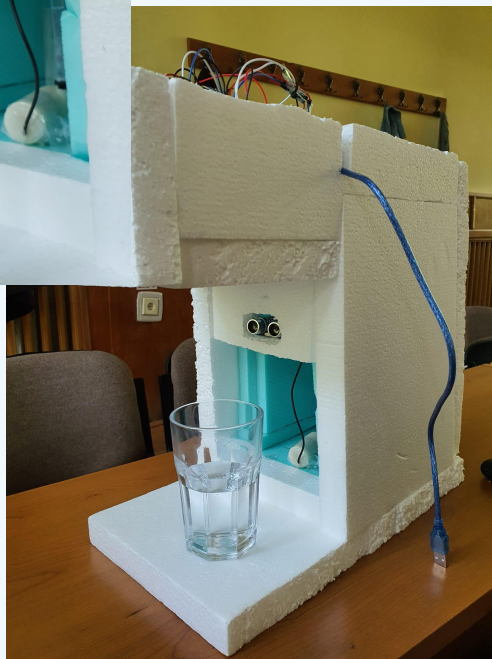
B.

PARTEA HARDWARE

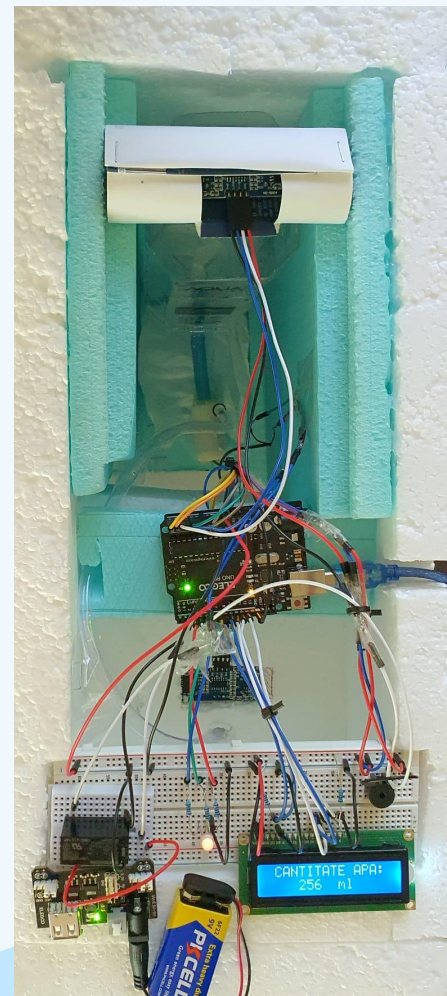
- Placa Arduino reprezintă componenta centrală a sistemului, asigurând controlul și comunicarea între toate celelalte componente hardware.
- Senzorii ultrasonici sunt utilizați pentru a măsura nivelul de apă din rezervor, furnizând date precise pentru a determina momentul umplerii sau golirii acestuia.
- Pompă de apă este responsabilă de furnizarea apei în recipientul dorit. Aceasta este controlată de placa Arduino în funcție de instrucțiunile primite din software.
- Afișajul LED oferă o indicație vizuală a stării de umplere a recipientului, permițând utilizatorului să monitorizeze procesul de umplere.



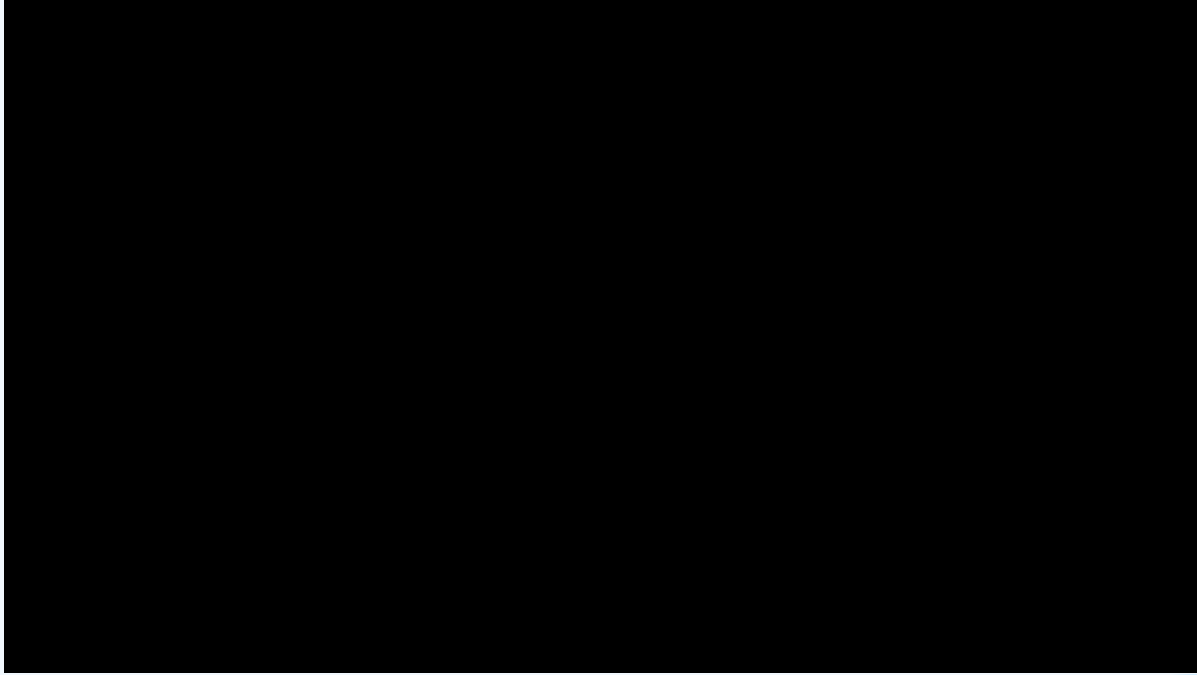
Cei doi senzori
care detectează
dacă este prezent
un recipient și
dacă paharul este
plin



Componentele
folosite:
Placa Arduino, 3
senzori, un releu
(comutare pompa
on/off), sursă
alimentare pompă,
ecran LCD, buzzer.



Video



Concluzii

Scop

Dozatorul automat de apă creat reprezintă o soluție eficientă și practică pentru a satisface nevoia de umplere controlată a recipientelor cu apă.



Ce am învățat

Utilizarea tehnologiei Arduino și a senzorilor ultrasonici oferă un sistem precis și fiabil. Obișnuirea cu mediul de lucru. Cunoștințe hardware.

Îmbunătățiri

În viitor, se pot adăuga îmbunătățiri, cum ar fi integrarea unei interfețe de utilizator mai avansate, cu ecrane tactile și opțiuni de programare personalizate. De asemenea, se poate dezvolta o aplicație mobilă pentru control și monitorizare la distanță a dozatorului de apă.

Aceste îmbunătățiri vor crește funcționalitatea și comoditatea sistemului.

Mulțumesc pentru atenție

