

МАШИНСКИ ФАКУЛТЕТ
БАЊА ЛУКА



СТУДИЈСКИ ПРОГРАМ
МЕХАТРОНИКА



УПУТСТВО ЗА УПОТРЕБУ ПРОЈЕКТА

Предмет:	Аутоматско управљање 1
Тема:	Пројекат регулације нивоа течности у посуди
Студенти:	Немања Радумило, 9997 Слободан Карановић, 9986 Ратко Марјановић, 9789

САДРЖАЈ

1. Опис пројекта.....	1
2. Програмски код	3

1. ОПИС ПРОЈЕКТА

Пројекат се састоји од двије посуде постављене на дрвеној плочи, од којих је једна посуда (шира) у директном додиру са плочом, док је друга постављена са горње стране доње посуде.

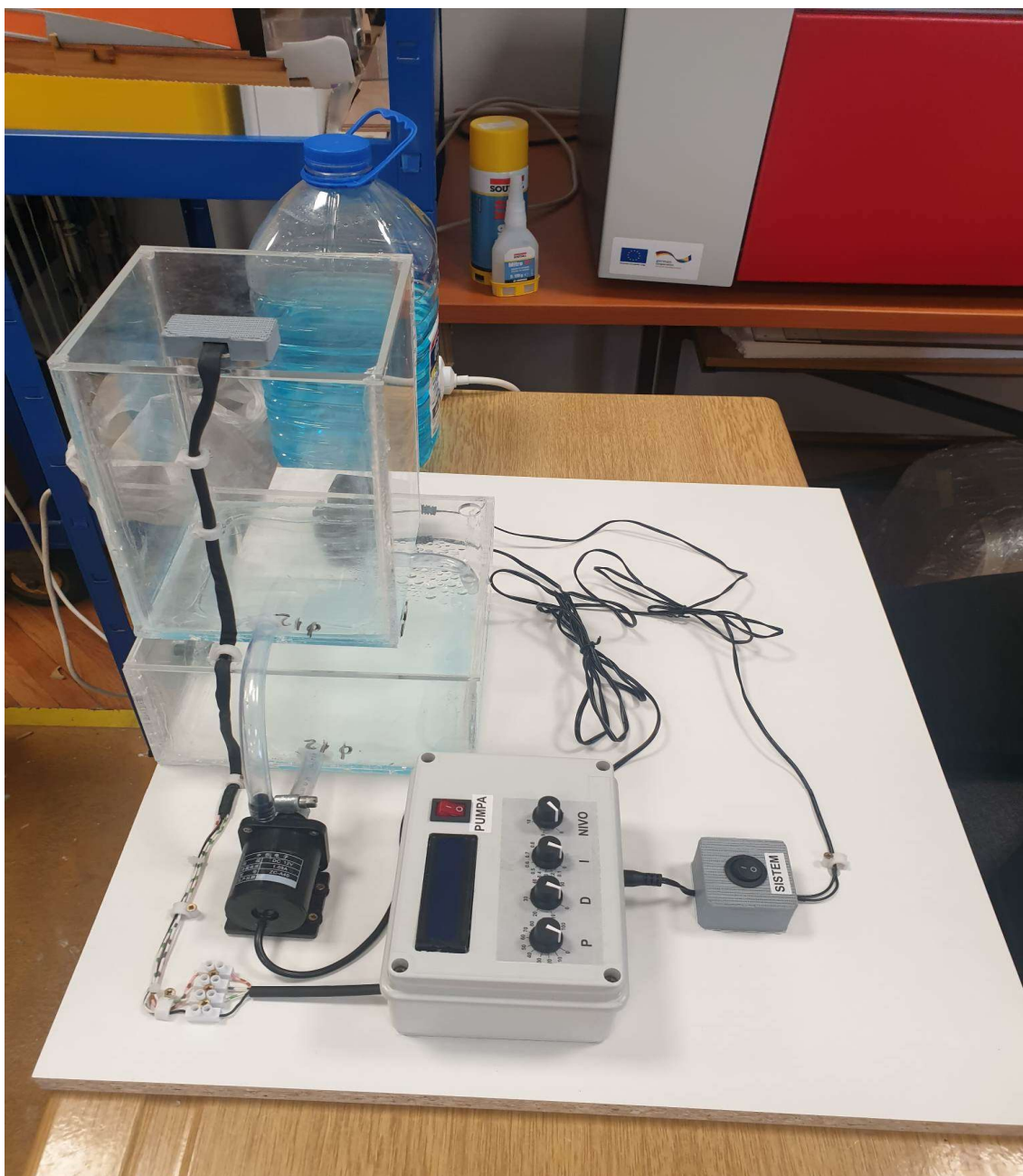
У погледу са предње стране лијево на плочи налази се разводна кутија у којој су смјештени Ардуино-Уно и драјвер на који су повезани потенциометри за управљање параметрима и самим нивоом течности. Такође са лијеве стране налази се и пумпа која је вагрес цријевима директно повезана са посудама.

На сл. 1 приказан је пројекат са његове предње стране.



Слика 1.

На сл. 2 налази се пројекат приказан из профила.



Слика 2.

На задњој страни посуде налази се заптивни вијак који је потребно одвртнути при испуштању течности из доње посуде. Аналогија томе било би испуштање уља из картера. Строго је препоручљиво да се пројекат не преноси нити транспортује док се вода налази у њему и да се приликом покретања пројекат прво постави на одговарајуће мјесто гдје ће бити презентован, а потом да се наспе вода путем отвора на доњој посуди. Такође је препоручљива употреба лијевка током сипања воде.

2. ПРОГРАМСКИ КОД

```
#include <AFMotor.h>
#include <LiquidCrystal_I2C.h>

AF_DCMotor motor(3);
LiquidCrystal_I2C lcd(0x3F, 16, 2);

#define trigPin 10 // Povezivanje UZV senzora
#define echoPin 13 // Povezivanje UZV senzora
#define PREKIDAC 7

int offset = 128;
int potPin = A0;
int potPin1 = A1;
float potPin2 = A2;
int potPin3 = A3;
float lasterror = 0;
float lasti = 0;
float currenti;
unsigned long r;
unsigned long t;
unsigned long t_prev = 0;
float dt;

//Serial plotter
unsigned long current_time = millis() - r;
float min_level = 0.0;
float max_level = 15.0;

void setup() {
    Serial.begin (9600);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(12, OUTPUT);
    pinMode(A0, INPUT);
    pinMode(A1, INPUT);
    pinMode(A2, INPUT);
    pinMode(A3, INPUT);
```

```
pinMode(PREKIDAC, INPUT); // PIN7 prekidac

Serial.print("Y,");
Serial.print(min_level);
Serial.print(',');
Serial.print(max_level);
Serial.println();

lcd.begin();
lcd.setCursor(4, 0);
lcd.print("PROJEKAT");
lcd.setCursor(0,1);
lcd.print("REGULACIJA NIVOVA");
delay(3000); // Prikazivanje poruke 3 sekunde
lcd.clear(); // Brisanje LCD ekrana

lcd.setCursor(0, 0);
lcd.print("ZN:");

lcd.setCursor(7, 0);
lcd.print("SN:");

lcd.setCursor(0, 1);
lcd.print("P:");

lcd.setCursor(6, 1);
lcd.print("D:");

lcd.setCursor(11, 1);
lcd.print("I:");
}

void loop(){
    int potValue = analogRead(A0);
    int mappedValue = map(potValue, 0, 1023, 100, 0); // Mapiranje
na opseg 0-100
    int roundedValue = round(mappedValue / 10.0) * 10; //
Zaokruživanje na deset
    int kp = constrain(roundedValue, 0, 100); // Ogranicavanje
vrijednosti na opseg 0-100
```

```
int potValue1 = analogRead(A1);
int mappedValue1 = map(potValue1, 0, 1023, 50, 0);
int roundedValue1 = round(mappedValue1 / 10.0) * 10;
int kd = constrain(roundedValue1, 0, 100);

float potValue2 = analogRead(A2);
float mappedValue2 = map(potValue2, 0, 1023, 10, 0); //
Mapiranje na opseg 0-10
float ki = mappedValue2 / 10.0; // Podjela sa 10 za dobijanje
decimalnih vrijednosti od 0 do 1

int potValue3 = analogRead(A3);
int setpoint = map(potValue3, 0, 1023, 20-12, 20-8);

float duration, distance, error, errororderiv, errorint, output;
digitalWrite(trigPin, LOW);
delayMicroseconds(2);

digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

duration=pulseIn(echoPin, HIGH);
distance=(duration/2)*0.0344;
error=distance-setpoint;
errororderiv=(error-lasterror)/0.5;
r=millis();
t = r* 1000;
dt=t-t_prev;
currenti=lasti+(dt*(error+lasterror)/2);

if ( (error > 1) || (error <- 1)){
    output=offset+kp*error+kd*errororderiv;
    lasterror=error;
    t_prev=t;
}

else{
    output=offset+kp*error+kd*errororderiv+ki*currenti;
    lasterror=error;
```



```
t_prev=t;
    }

//NIJE DOZVOLJENO PREKORACENJE BRZINE
if(output>255){
    output=255;
}

//NE DOZVOLJAVA DA NEGATIVNI OUTPUT PREDJE U POZITIVNU
VRIJEDNOST
if(output<0){
    output=0;
}

//KADA JE PREKIDAC UKLJUCEN PUMPA SE GASI
if (digitalRead(PREKIDAC) == HIGH) {

    output=0;
    }
    else {
    output=output;
    }

//ISPIS NIVOА KOJI SE ZADAJE NA POTENCIOMETRU
lcd.setCursor(3, 0);
char strN[6]; // Kreiranje prazanog niza karaktera
int nivo = 20 - setpoint;

if (nivo >= 10) {
    snprintf(strN, sizeof(strN), "%d", nivo);
} else {
    snprintf(strN, sizeof(strN), " %d", nivo);
}
lcd.print(strN); // Ispisu konvertovane vrijednosti

//ISPIS NIVOА KOJI MJERI SENZOR
lcd.setCursor(10, 0);
lcd.print(19.4-distance);

//ISPIS KP
lcd.setCursor(2, 1);
```



```
char strKp[6];
if (kp == 100) {
    snprintf(strKp, sizeof(strKp), "%d", kp);
} else {
    snprintf(strKp, sizeof(strKp), "%d ", kp);
}
lcd.print(strKp);

//ISPIS KD
lcd.setCursor(8,1);
char strKd[6];
if (kd == 10) {
    snprintf(strKd, sizeof(strKd), "%d", kd);
} else {
    snprintf(strKd, sizeof(strKd), "%d ", kd);
}
lcd.print(strKd);

//ISPIS KI
lcd.setCursor(13,1);
lcd.print(ki);

//Serial plotter
float level = constrain(19.4 - distance, min_level, max_level);
// Slanje podataka preko serijske veze ka racunaru
Serial.print("D,");
Serial.print(current_time);
Serial.print(20-setpoint);
Serial.print(',');
Serial.print(level);
Serial.println();

//    Serial.print("Greska:");
//    Serial.print("\t");
//    Serial.print(error);
//    Serial.print("\t");
//    Serial.print("\t");
//    Serial.print("Udaljenost:");
//    Serial.print("\t");
//    Serial.print(19.4-distance);
//    Serial.print("\t");
```

```
//    Serial.print("\t");  
//    Serial.print("Brzina motora:");  
//    Serial.print("\t");  
//    Serial.print(output);  
//    Serial.print("\n");  
  
    digitalWrite(12, HIGH);  
    analogWrite(3, output);  
    motor.run(FORWARD); // Zadajem funkciju koja okreće motor  
    naprijed *NAPOMENA: Ne može raditi bez funkcije setSpeed.*  
    motor.setSpeed(output); // Funkcija kontrolise brzinu motora.  
    Brzina se kreće od 0 do 255, pri čemu je 0 isključeno, a 255  
    maksimalna moguća brzina.  
    delay(500);  
}
```