

Irigator Automat

Introducere

Proiectul meu implementeaza un sistem automat de irigare.

Scopul lui este de a asigura apa necesara unei planta pentru a creste.

Proiectul meu este util pentru persoanele care isi doresc plante dar care nu vor sa adauge un pas nou in rutina lor zilnica.

Descriere generală

Proiectul este format din 5 elemente:

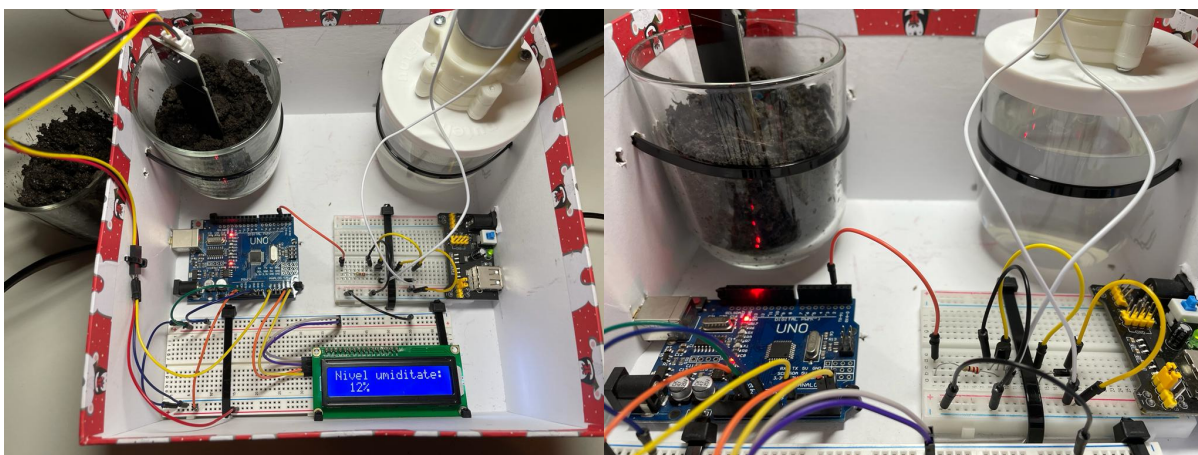
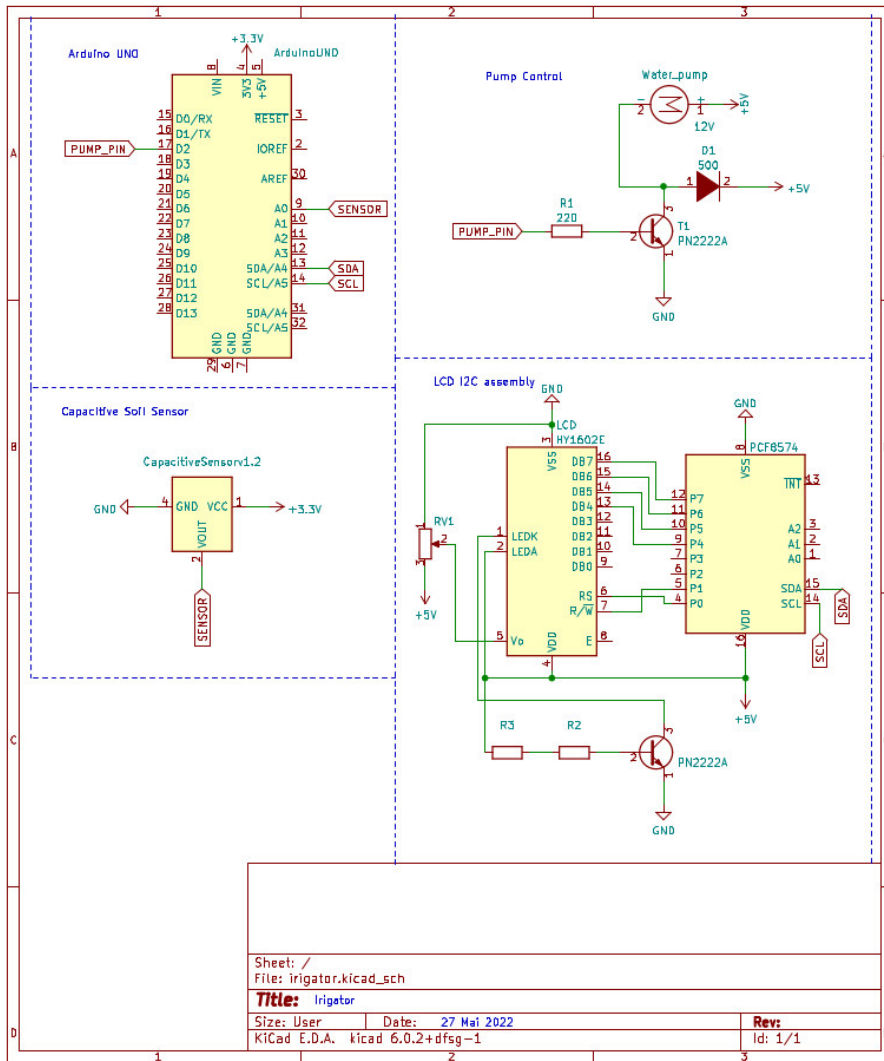
- Placa Arduino Uno care controleaza tot sistemul
- O pompa care asigura circulatia apei
- Un circuit de alimentare pentru pompa
- Un display LCD care arata informatii in timp real
- Un senzor higrometru capacitiv

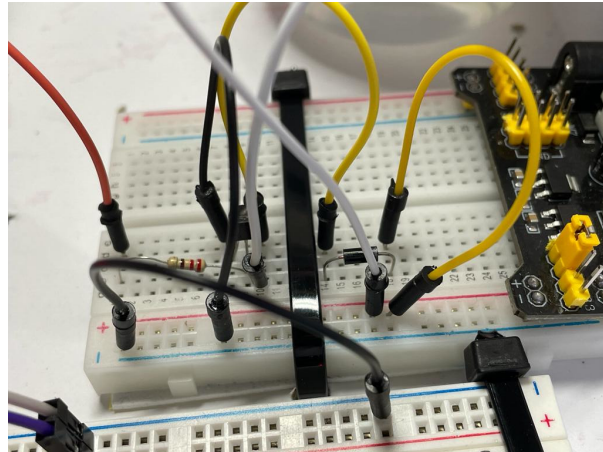
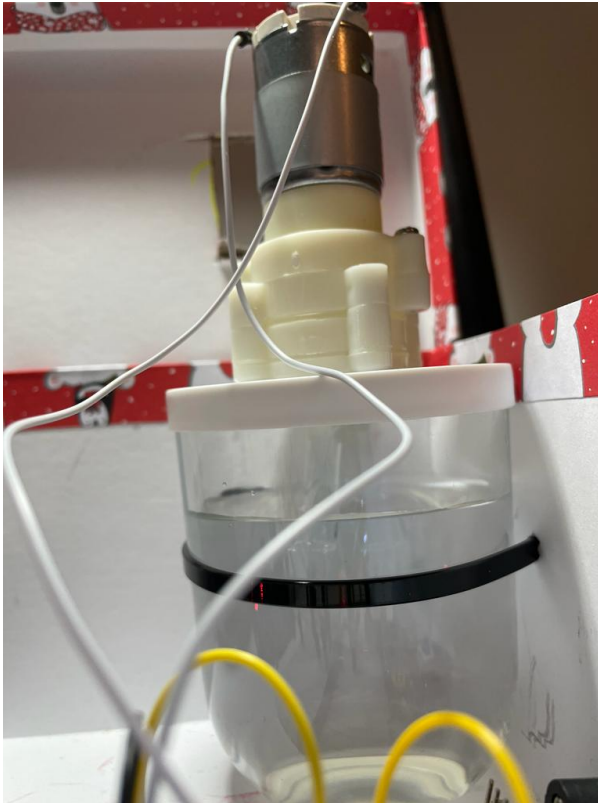


Hardware Design

Lista de componente:

- Arduino UNO
- Pompa DC 12V
- Adaptor AC la DC cu output de 9V si 2A
- [Sursa de alimentare 3.3V/5V](#)
- [Senzor higrometru capacitiv](#)
- [Display LCD cu modul de control I2C](#)
- Rezistenta 220 Ohm
- Tranzistor NPN 2p2222
- Dioda 1N4007





Software Design

Codul implementeaza o logica automata pentru irigare:

- Cand umiditatea solului este suficient de mica, pompa porneste pentru a iriga solul.

In code se folosesc bibliotecile:

- Wire pentru comunicare I2C
- LiquidCrystal I2C pentru a avea o clasa primitiva pentru LCD

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

#define PUMP_PIN 2
#define SENSOR_PIN A0

int soil_saturation_value = 0;
int soil_saturation_percentage = 0;

LiquidCrystal_I2C lcd(0x27,16,2); // set the LCD address to 0x27 for a 16
chars and 2 line display

void setup()
{
  pinMode(PUMP_PIN, OUTPUT);
  digitalWrite(PUMP_PIN, LOW);
}
```

```
lcd.begin();
lcd.backlight();

Serial.begin(9600);
}

void read_soil_saturation() {
  soil_saturation_value = analogRead(SENSOR_PIN);
  soil_saturation_percentage = map(soil_saturation_value, 450, 226, 0, 100);

  if (soil_saturation_percentage > 100) {
    soil_saturation_percentage = 100;
  } else if (soil_saturation_percentage < 0) {
    soil_saturation_percentage = 0;
  }
}

void display_soil_saturation() {
  lcd.setCursor(0,0);
  lcd.print("Nivel umiditate:");
  lcd.setCursor(2,1);
  lcd.print(soil_saturation_percentage);
  lcd.print("%");
}

void water() {
  if (soil_saturation_percentage < 40) {
    digitalWrite(PUMP_PIN, HIGH);
  } else {
    digitalWrite(PUMP_PIN, LOW);
  }
}

void loop() {
  delay(1000);

  read_soil_saturation();
  display_soil_saturation();

  water();
}
```

Rezultate Obținute

În urma realizării proiectului am ajuns la un prototip funcțional potrivit pentru un demo tehnic. Următoarele etape ale dezvoltării pot fi:

- pregati pachetul pentru a fi mai compact
- pregatirea proiectului sa permita un sistem de irigare mai complex pentru mai multe plante

Concluzii

In concluzie, proiectul meu cred ca poate fi foarte util, iar dupa ce este impachetat mai facil pentru un spatiu mic, si cu un pret atractiv, poate fi produs pentru vanzare

Download

[irigator_dutica_maria-diana.zip](#)

Bibliografie/Resurse

https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0193_Web.pdf
<https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>

[Export to PDF](#)

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

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2022/arosca/irigator-automat>



Last update: **2022/05/30 21:23**