*Sistem de comanda pentru irigatii*

Studenti: Baesu Sebastian-Flavius

Bociort Dinu

Caloianu Mihai-Alexandru

Voinescu Adriana-Iuliana

An : II Grupa : 1.2

Obiectivul temei şi potenţiala utilitate în lumea reală

Obiectivul temei este ca la măsurarea umidității la un interval de timp să se citească valoarea senzorului de umiditate și să se afișeze pe ecranul LCD , toate acestea efectuându-se în urma apăsării butonului Brick.

Principala utilitate în lumea reală a sistemul de irigație este faptul că poate fi adaptat și folosit cu succes în pomicultură și viticultură, acesta garantând producții constante, indiferent de cantitățile de precipitații căzute în perioada rece și vegetativă a anului.

Componentele montajului

Montajul are următoarele componente:

1. Arduino
2. LED RGB pentru semnalarea stării de umiditate (albastru – umed; verde – umiditate atmosferică; roșu – uscat)
3. Senzor de umiditate (Moisture Sensor)
4. Ecran LCD
5. Buton brick

Schema logică

INITIALIZAM ELEMENTE

NU DA

Se apasa butonul

Citire valori senzor

Val > prag

NU DA

Irigare

Nu are loc udarea

Se aprinde becul albastru

Stop

Plan de testare

Primul obiectiv pentru realizarea proiectului a fost compilarea acestuia şi verificarea greşelilor de sintaxă. După corectarea erorilor apărute în program a fost nevoie de o verificare a plăcii Arduino, care constă în faptul ca fiecare cablu să fie conectat la pinul corespunzător.

Noi pentru că suntem viitori ingineri, ne-am gândit să testăm placa Arduino după cum vedeți și în pozele de mai sus, prin a pune senzorul într-un pahar cu apă și a verifica ce ne afișază și dacă acesta funcționează.

Ultimul obiectiv al nostru a fost verficarea întregului proiect. Dacă se execută sau nu fiecare comandă după cum se cere în proiect, iar în caz de eroare, găsirea erorii și rezolvarea acesteia.

Concluzii

Prin realizarea acestui proiect am reușit să descoperim ceea ce înseamnă munca în echipă, chiar dacă am întâmpinat multe probleme, cum ar fi faptul ca nu ne-am organizat de la început și nu am venit fiecare cu câte o idee, în final, am pus fiecare problemă și am încercat să găsim soltuții și mai apoi sa mergem mai departe, și astfel am reușit să realizăm micul nostru proiect.

Codul sursa al programului

#include <LiquidCrystal.h>

class Pin

{

protected:

int pin;

public:

int Pin\_Query()

{

return pin;

}

};

class LED:public Pin

{

private:

public:

LED(int x)

{

pin=x;

}

};

class Senzor:public Pin

{

private:

int value;

public:

Senzor(int x)

{

pin=x;

}

int State()

{

value=analogRead(pin);

return value;

}

};

class Buton:public Pin

{

private:

int value;

public:

Buton(int x)

{

pin=x;

}

int State()

{

value=digitalRead(pin);

return value;

}

};

LED galben(8); //initializam pin prin clasa LED

LED rosu(10); //initializam pin prin clasa LED

Senzor soil\_moisture(A0); //initializam pin prin clasa Senzor

Buton b1(12); //initializam pin prin clasa Button

int Q[6]={7,6,5,4,3,2}; // RS E DB4 DB5 DB6 DB7

LiquidCrystal lcd(Q[0],Q[1],Q[2],Q[3],Q[4],Q[5]); //declaram pinii pentru LCD

void setup() {

Serial.begin(9600);

lcd.begin(16, 2);

pinMode(b1.Pin\_Query(), INPUT);

pinMode(galben.Pin\_Query(), OUTPUT);

pinMode(rosu.Pin\_Query(), OUTPUT);

lcd.print("Valoare: ");

}

int umiditate,prag=350;

void loop() {

if(b1.State()==HIGH) //daca butonul este apasat atunci porneste sistemul

{

umiditate=soil\_moisture.State();

lcd.setCursor(9, 0); // se seteaza primul rand, col 9

lcd.print(umiditate);

if(umiditate>950) // daca este pamant uscat

{

lcd.setCursor(13, 0);

lcd.print(" ");

lcd.setCursor(0, 1);

lcd.print("Pamant uscat!");

digitalWrite(rosu.Pin\_Query(), LOW);

digitalWrite(galben.Pin\_Query(), LOW);

}

else

if(umiditate<950 && umiditate>=350) // pamant umed

{

lcd.clear();

digitalWrite(rosu.Pin\_Query(), LOW);

digitalWrite(galben.Pin\_Query(), HIGH);

lcd.setCursor(12, 0);

lcd.print(" ");

lcd.setCursor(0, 1);

lcd.print("Irigam putin!");

}

else

if(umiditate<prag) // daca nu este nevoie de irigatie, prag

{

digitalWrite(rosu.Pin\_Query(), HIGH);

digitalWrite(galben.Pin\_Query(), LOW);

lcd.setCursor(12, 0);

lcd.print(" ");

lcd.setCursor(0, 1);

lcd.print("Nu se iriga!");

}

}

}