Departmentul Calculatoare Universitatea Tehnic˘a din Cluj-Napoca



**Proiect de laborator la Proiectarea cu Microprocesoare**

***Seră inteligentă***

Nume:Martin Darius Nicolae Grupa:30239

Profesor Coordonator: Mircea Paul Mureșan



# Descrierea problemei

Proiectul are ˆın vedere construirea unei sere inteligente care este comandată cu ajutorul servo-motoarelor.

Proiectul are în vedere rezolvarea a 3 probleme a serelor: problema temperaturii, problema luminii si problema umidității din seră. Prin intermediul servo-motoarelor efectuam deschideri/închideri ale unor uși/geamuri și înclinarea unui recipient pentru a simula problema umidității solului.

Proiectul va fi realizat utilizˆınd pl˘aci Arduino Uno, servo-motoare, senzor de temperatură, senzor de lumină, senzor de

# Solut, ia g˘asit˘a

Am ales să folosesc câte un servo-motor pentru fiecare problemă de rezolvat a proiectului: un servo-motor pentru închiderea/deschiderea unei uși pentru a ajusta temperatura din interiorul serei, alt servo-motor pentru închiderea/deschiderea unui geam pentru a putea intra lumina în interiorul serei, iar al treilea servo pentru a rezolva problema umidității, dacă pământul este prea uscat, servo-ul se va roti la 90 de grade și va vărsa apa dintr-o sticlă în seră.

Senzorii de temperatură au fost legați la pinii analogici 0,1,2 în urmatoarea ordine: temperatură, lumină și umiditate. Servo-motoarele au fost legate la pinii 9,10,11 ai plăcuței.

# Manual de utilizare

Se încarcă programul pe plăcuța Arduino Uno, iar sera va menține o temperatură optimă, un grad de iluminare optim si o umiditate optimă a solului. Utilizatorul nu trebuie să facă teoretic nimic, doar mentenanța serei.

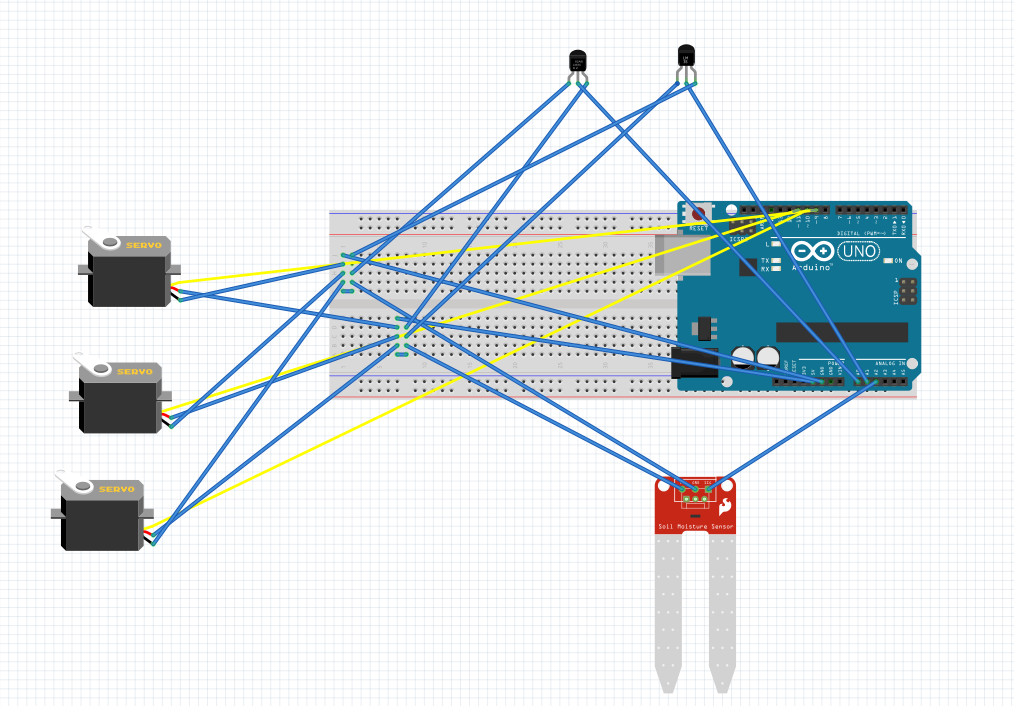
# Explicații Hardware și Software

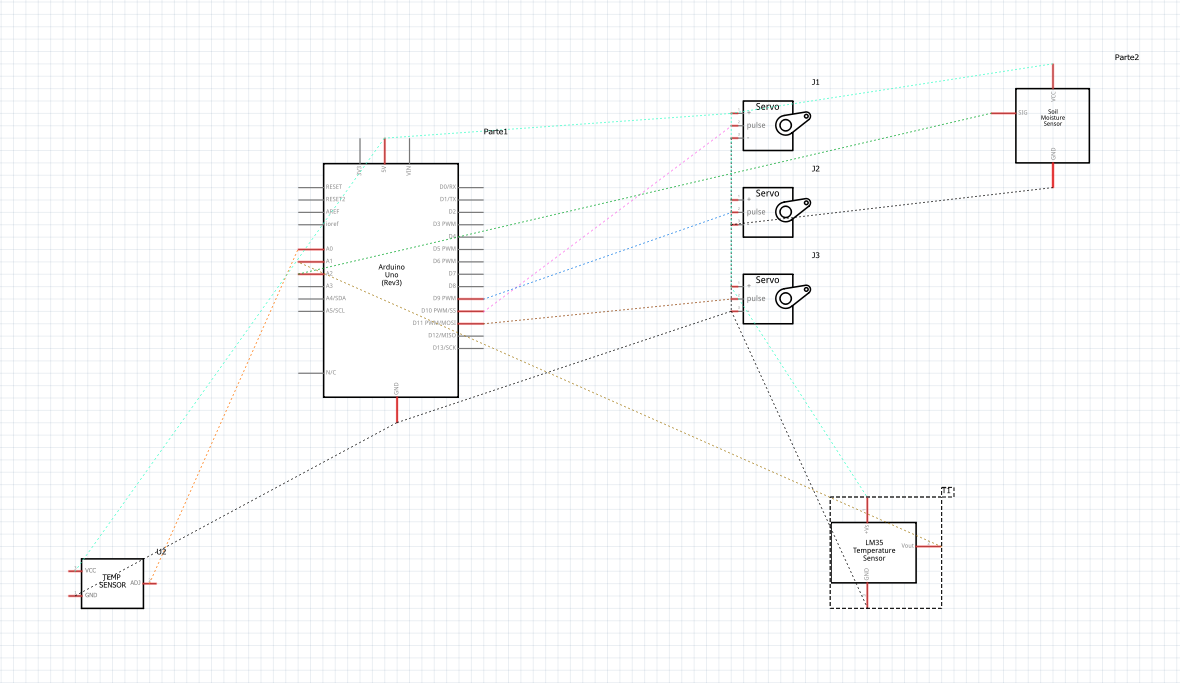
# Ca hardware am folosit plăcuța compatibilă cu Arduino Uno, un breadboard, fire jumper, senzor de temperatură, senzor de lumină și senzor de umiditate; pe lângă acestea am folosit servo-motoare.

# Ca software am folosit aplicația Arduino IDE în care am scris codul pentru seră. Aici am inclus si biblioteca servo pentru a putea manevra motoarele. Am folosit mai multe funcții pentru o mai usoară vizualizare a codului, aceste funcții au rolul de a manevra fiecare servo în parte, iar acestea le apelam pentru fiecare senzor în parte.

# 

# Schema generală

****

****

**Proiectul**

**(poza când e gata macheta)**

**Bibliografie**

1. Proiectarea cu Microprocesoare ˆIndrum˘ator de laborator, https://biblioteca.utcluj.ro/files/carti-online-cu-coperta/336-3.pdf



**Codul pentru seră:**

#include <Servo.h>

float resolutionADC = .0049 ; // rezoluția implicită (pentru

//referința 5V) = 0.049 [V] / unitate

float resolutionSensor = .01 ; // rezoluție senzor = 0.01V/°C

int lightValue=0;

int moistureSensorPin= A2;

int tempSensorPin=A0;

int lightSensorPin=A1;

int moistureValue=0;

Servo srv;

void setup() {

Serial.begin(9600);

}

void closeGateTemp(int pin)

{

srv.attach(pin);

srv.write(0);

delay(1000);

srv.detach();

}

void openGateTemp(int pin)

{

srv.attach(pin);

srv.write(180);

delay(1000);

srv.detach();

}

void closeGateLight(int pin)

{

srv.attach(pin);

srv.write(180);

delay(1000);

srv.detach();

}

void openGateLight(int pin)

{

srv.attach(pin);

srv.write(90);

delay(1000);

srv.detach();

}

void closeGateMoisture(int pin)

{

srv.attach(pin);

srv.write(0);

delay(1000);

srv.detach();

}

void openGateMoisture(int pin)

{

srv.attach(pin);

srv.write(120);

delay(1000);

srv.detach();

}

void loop(){

//temp

Serial.print("Temp [C]: ");

float temp = readTempInCelsius(10, A0); // citeste temperatura

//de 10 ori, face media

// float temp = analogRead(tempSensorPin);

Serial.println(temp);// afisare

if(temp>26){

openGateTemp(9);

}

if(temp<24){

closeGateTemp(9);

}

//light

lightValue=analogRead(lightSensorPin);

Serial.print("Light: ");

Serial.println(lightValue);

if(lightValue<500){

openGateLight(10);

}

else if (lightValue>600){

closeGateLight(10);

}

//moisture

moistureValue=analogRead(moistureSensorPin);

Serial.print("Moisture: ");

Serial.println(moistureValue);

if(moistureValue>400){

openGateMoisture(11);

}

else if (moistureValue<350){

closeGateMoisture(11);

}

delay(200);

}

float readTempInCelsius(int count, int pin) {

// citește temperatura de count ori de pe pinul analogic pin

float sumTemp = 0;

for (int i =0; i < count; i++) {

int reading = analogRead(pin);

float voltage = reading \* resolutionADC;

float tempCelsius = (voltage - 0.5) / resolutionSensor ;

// scade deplasament, convertește în grade C

sumTemp = sumTemp + tempCelsius; // suma temperaturilor

}

return sumTemp / (float)count; // media returnată

}