**wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwww**

|  |
| --- |
| Sistem de stocare a energiei solare  MURESAN BIANCA-AFINIA |

Cuprins

1. Descriere.................................................................2
2. Componenete.......................................................2
3. Explicarea codului.................................................7
4. Schema electrica generala...............................10
5. Bibliografie............................................................11

* *Descriere*

Consumul excesiv de curent electric trebuie eliminat prin alternative mult mai econome si eficiente. Acest lucru il putem face utilizand sisteme ce transforma energia solara în curent electric.

Acest sistem automatizat de transformare a energiei luminoase în energie electrică are ca scop alimentarea diferitelor dispozitive .

Programarea placutei Arduino UNO se bazează pe limbajul C++ realizând deplasarea panoului solar în functie de intensitatea luminii.

Tensiunea de lucru este 5V iar curentul maxim de iesire este 1.5 A.

* Componente
* Laptop
* Telefon/tableta/dispozitiv electronic
* Cablu de încarcare cu capat USB
* Suruburi, stalpi de cupru,piulite
* Placi acrilice
* \*Lanterna
* Acumulator(baterie cu litiu)

|  |  |
| --- | --- |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.16.jpeg | * Cablu de conectare a placutei Arduino |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.17 (4).jpeg | * Placuta Arduino UNO |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.15 (2).jpeg | * Suport baterie (acumulator) |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.16 (1).jpeg | * Fire |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.18.jpeg | * Buton |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.15 (1).jpeg | * Buzzer |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.18 (3).jpeg | * Modul de intensitate a luminii |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.18 (2).jpeg | * Senzor de temperatura si umiditate |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.18 (1).jpeg | * Fotorezistenta \*4 |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.15 (5).jpeg | * Servomotor \*2 |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.15 (3).jpeg | * Display LCD |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.15 (4).jpeg | * Modul de încarcare a telefonului   ( mufa USB) |
| C:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.17 (3).jpegC:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.17 (1).jpegC:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.17.jpegC:\Users\Bianca\Downloads\WhatsApp Image 2023-12-30 at 01.16.15.jpeg | * Panou solar |

* Explicarea codului

**Declaratiile de bibleoteci si obiecte**: Se includ diferite biblioteci necesare pentru funcționarea componentelor precum LCD-uri, senzori de lumină, senzori de temperatură și umiditate, servo-motoare și altele. De asemenea, se definesc obiectele asociate cu aceste biblioteci pentru a le utiliza mai târziu în cod.

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 16, 2);

#include <BH1750.h>

BH1750 lightMeter;

#include <dht11.h> //include codul bibliotecii:

dht11 DHT;

#define DHT11\_PIN 7 //definiți DHT11 ca pin digital 7

#include <Servo.h>

Servo lr\_servo;//definește numele servomotorului care se rotește la dreapta și la stânga

Servo ud\_servo;//definește numele servomotorului care se rotește în sus și în jos

const byte interruptPin = 2; //pinul butonului; corupția este întreruptă

int lr\_angle = 90; //setați unghiul inițial la 90 de grade

int ud\_angle = 10;//setați unghiul inițial la 10 grade; păstrați panourile solare în poziție verticală pentru a detecta cea mai puternică lumină

int l\_state = A0;//definiți intrarea analogică de tensiune a fotorezistorilor

int r\_state = A1;

int u\_state = A2;

int d\_state = A3;

const byte buzzer = 6; //setați buzzerul la pinul digital 6

const byte lr\_servopin = 9;//definiți semnalul de control al servo-rotative dreapta și stanga

const byte ud\_servopin = 10;//definește semnalul de comandă al servomotorului rotativ în sensul acelor de ceasornic și în sensul invers acelor de ceasornic

unsigned int light; //salvați variabila intensității luminii

byte error = 15; //Definiți intervalul de eroare pentru a preveni vibrațiile

byte m\_speed = 10; //setați timpul de întârziere pentru a regla viteza servo; cu cât timpul este mai lung, cu atât viteza este mai mică

byte resolution = 1; //setați precizia de rotație a servomotorului, unghiul minim de rotație

int temperature; //salvați variabila de temperatură

int humidity; //salvați variabila de umiditate

**Inițializarea și configurarea hardware-ului**: În funcția setup(), se inițializează și configurează toate componentele folosite. Aici sunt stabilite vitezele de comunicare, conexiunile cu servo-motoarele, pinii și modurile de funcționare pentru diferitele componente.

void setup() {

Serial.begin(9600); //definiți rata de baud serială

// Inițializarea I2C (Biblioteca BH1750 nu face acest lucru automat)

Wire.begin();

lightMeter.begin();

lr\_servo.attach(lr\_servopin); // setaţi pinul de control al servo

ud\_servo.attach(ud\_servopin); // setaţi pinul de control al servo

pinMode(l\_state, INPUT); //setați modul de pin

pinMode(r\_state, INPUT);

pinMode(u\_state, INPUT);

pinMode(d\_state, INPUT);

pinMode(interruptPin, INPUT\_PULLUP); //pinul butonului este setat la modul pull-up de intrare

attachInterrupt(digitalPinToInterrupt(interruptPin), adjust\_resolution, FALLING); //xternal interrupt touch type is falling edge; adjust\_resolution is interrupt service function ISR

lcd.init(); // inițializați ecranul LCD

lcd.backlight(); //set lumina de fundal LCD

lr\_servo.write(lr\_angle);//reveniți la unghiul inițial

delay(1000);

ud\_servo.write(ud\_angle);

delay(1000);

}

**Bucla principală (loop()):** În bucla principală a programului, sunt apelate funcțiile care efectuează acțiunile dorite. Aceste funcții includ acționarea servo-motoarelor pentru a urmări lumina solară, citirea nivelului de lumină folosind senzorul BH1750, citirea datelor de temperatură și umiditate cu ajutorul senzorului DHT11 și afișarea acestor date pe un ecran LCD.

void loop() {

ServoAction(); //servo efectuează acțiunea

read\_light(); //citiți intensitatea luminii de bh1750

read\_dht11(); //citiți valoarea temperaturii și umidității

LcdShowValue(); //Lcd afișează valorile intensității luminii, temperaturii și umidității

}

**Funcții specifice:**

* **ServoAction():** Această funcție controlează mișcarea servo-motoarelor în funcție de nivelurile de lumină detectate de senzorii din diferite direcții.
* **LcdShowValue():** Afișează valorile citite (lumină, temperatură, umiditate) pe un ecran LCD.
* **read\_light():** Citeste nivelul de lumină utilizând senzorul BH1750.
* **read\_dht11():** Citeste datele de temperatură și umiditate de la senzorul DHT11.
* **adjust\_resolution():** Această funcție permite modificarea preciziei de rotație a servo-motoarelor printr-un buton, folosind o funcție de tip interrupt.
* Schema electrică generală:

Conexiuni hardware:

* **Senzorul de lumină BH1750**: Conectat prin interfața I2C la pinii SDA și SCL ai placii Arduino. Se utilizează biblioteca BH1750.h pentru a citi datele de la acest senzor.
* **Senzorul de temperatură și umiditate DHT11**: Conectat la pinul digital 7 al plăcii Arduino (potrivit cu #define DHT11\_PIN 7 în cod). Se utilizează biblioteca dht11.h pentru a citi datele de la acest senzor.
* **Servo-motoare**: Două servo-motoare pentru mișcarea sistemului de urmărire solară, conectate la pinii digitali 9 și 10 ai plăcii Arduino pentru control.
* **Buzzer**: Conectat la pinul digital 6 pentru a emite semnale acustice.
* **Senzori de lumină:** Patru senzori de lumină conectați la pinii analogici A0, A1, A2 și A3 pentru a detecta nivelurile de lumină din diferite direcții.
* **Buton interruptor**: Conectat la pinul digital 2 pentru a ajusta precizia de rotație a servo-motoarelor.
* **LCD:** Afișajul LCD1602 conectat prin interfața I2C (SDA și SCL) și adresa 0x27. Este utilizată biblioteca LiquidCrystal\_I2C.h pentru a controla acest afișaj.

Conexiunile fizice:

* Arduino SDA și SCL conectate la senzorul BH1750 și afișajul LCD1602.
* Pinii digitali 7, 6, 9 și 10 ai plăcii Arduino conectați la senzorul DHT11, buzzer și servo-motoare.
* Pinii analogici A0, A1, A2 și A3 ai plăcii Arduino conectați la senzorii de lumină.
* Pinul digital 2 al plăcii Arduino conectat la butonul interruptor pentru ajustarea preciziei de rotație a servo-motoarelor.
* BIBLIOGRAFIE

<https://roboromania.ro/manuale/Arduino-2-Starter-Kit-manual-roboromania.pdf> ,

<https://chat.openai.com/> ,

<https://github.com/>,

<https://biblioteca.utcluj.ro/files/carti-online-cu-coperta/341-7.pdf> , <https://users.utcluj.ro/~rdanescu/pmp-lab05.pdf> ,

<https://playground.arduino.cc/> ,

https://wiki.keyestudio.com/KS0530\_Keyestudio\_Solar\_Tracking\_DIY\_Kit\_for\_Arduino