

Solar tracker - Andrada Cojocaru ☀

Proiect PM - 2023

Student: Andrada-Ioana Cojocaru

Grupa: 332CA

Asistent: Victor Stoica

Introducere

Proiectul consta intr-un panou solar ce urmareste unda de lumina maxima prin intermediul celor 4 fotorezistori, se poate misca atat pe lungime cat si pe latime prin intermediul a 2 servomotoare. Sistemul va colecta date despre sistem si va determina perioada zilei.

Am ales sa implementez un proiect ce se bazeaza pe un panou solar, deoarece sustin energia alternativa sustenabila, iar cea solara este cea mai raspandita si cunoscuta. Trebuie sa invatam sa profitam de acest tip de energie inepuizabila. Panourile solare sunt destul de costisitoare, de aceea este important sa le folosim la maxim, iar prin acest mecanism propus, captarea energiei creste exponential.

Am pornit de la un video, pe care l-am vazut pe youtube cu un proiect facut pentru un targ stiintific: [proiect_targ_stiintific](#), iar apoi am decis sa continui dupa o alta idee interesanta: [proiect_final](#) si sa includ si o parte de analiza a datelor pentru a studia diversi factori ce influenteaza captarea energiei solare.



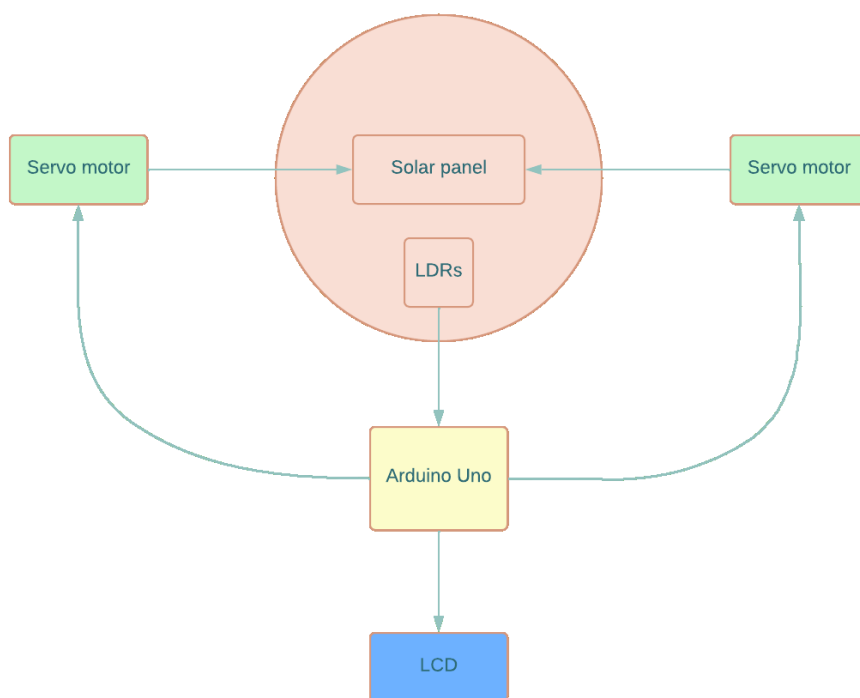
Descriere generală

Panoul solar va fi însoțit de o componentă ce va conține fotorezistoarele care detectează intensitatea luminoasă cea mai mare și transmite microcontroller-ului datele pentru a comanda cele 2 servomotoare să se miște în direcția indicată.

Prin acest tip de ansamblu se maximizează producția de energie solară. Fata de un panou fix, energia captată de acest tip de ansamblu este de 40% mai mult. Panourile solare sunt costisitoare, astfel prin introducerea fotorezistoarelor și celor 2 servomotoare se îmbunătățește randamentul ansamblului.

Lcd-ul va primi detalii despre perioada zilei în funcție de direcția din care este primită lumina sau absența ei.

Schema bloc



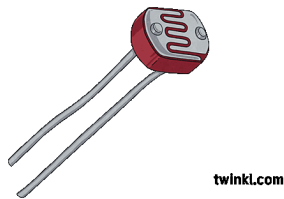
Hardware Design

Componentele utilizate:

- **panou solar** - 6V, 160mA, 0.96W



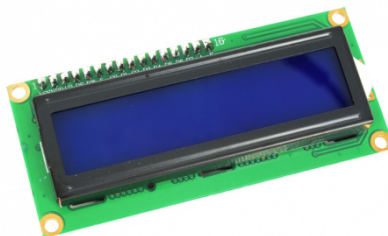
- **4 fotorezistoare** - pentru detectarea sursei de lumina maxima



- **2 servo sg90** - pentru miscarea panoului solar atat pe verticala cat si pe orizontala



- **LCD 1602** cu Interfata I2C si Backlight Albastru - afisarea capacitatii stocate



- **componente 3D** - pentru suportul panoului solar [3dcomponents.zip](#)
- **arduino uno**
- **placuta prototipare**
- **fire**

- **1 led** care se va incarca de la panoul solar
- **rezistente** → 2 * 10K pentru legarea fotorezistentelor si 2 * 1K pentru divizor de tensiune la legarea ledului

Schema electrica:



Software Design

Descrierea codului aplicației (firmware):

- **mediu de dezvoltare:** Arduino IDE 2.1.0
- **biblioteci:** Wire.h, LiquidCrystal_I2C.h, servo.h (initial, dupa am facut propriile functii)

Declararea bibliotecilor

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

Setam lcd -ul

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

</code>

Declararea define urilor

```
#define SERVO_RIGHT_LEFT_PIN 9
#define SERVO_UP_DOWN_PIN 10

#define SERVO_RIGHT_LEFT_REG OCR1A
#define SERVO_UP_DOWN_REG OCR1B

#define LDR_TOP_LEFT_PIN A0
#define LDR_TOP_RIGHT_PIN A1
#define LDR_BOTTOM_LEFT_PIN A2
#define LDR_BOTTOM_RIGHT_PIN A3

#define TOLERANCE 10
```

Declararea variabilelor globale pentru miscarea servo-motoarelor - de aici pornesc

```
int servoRightLeftPos = 35;  
int servoUpDownPos = 76;
```

Functie pentru miscarea servomotoarelor in functie de pin si valoare

```
void setServoAngle(int servoPin, int angle)  
{  
    float pulseWidthMs = 1.0 + (float)angle / 180.0 * 1.0; // Map angle to  
    pulse width between 1ms and 2ms  
    int pulseWidthTicks = pulseWidthMs * 20000 / 20; // Convert pulse width  
    to ticks for 20ms period  
    if (servoPin == SERVO_RIGHT_LEFT_PIN) {  
        SERVO_RIGHT_LEFT_REG = pulseWidthTicks;  
    } else if (servoPin == SERVO_UP_DOWN_PIN) {  
        SERVO_UP_DOWN_REG = pulseWidthTicks;  
    }  
}
```

Setup - configuram lcd ul sa afiseze mesajul de pe primul rand, configuram registri corespunzatori si pozitionam servo-motoarele in pozitia de start

```
void setup()  
{  
    // write message on lcd  
    lcd.init();  
    lcd.backlight();  
    lcd.setCursor(3, 0);  
    lcd.print("Hello");  
  
    analogReference(DEFAULT);  
  
    // Configure Timer1 for servo control  
    // Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM (inverting  
mode)  
  
    TCCR1A = _BV(COM1A1) | _BV(COM1B1) | _BV(WGM11);  
  
    // clkI/O/8 (from prescaler)  
    TCCR1B = _BV(WGM13) | _BV(CS11);  
    ICR1 = 20000; // 20ms period for PWM  
    DDRB |= _BV(PB1) | _BV(PB2); // Set servo pins as outputs  
    PORTB &= ~(_BV(PB1) | _BV(PB2)); // Set servo pins low  
  
    // set where the servo start  
    servoRightLeftPos = 35;  
    servoUpDownPos = 76;  
    setServoAngle(SERVO_RIGHT_LEFT_PIN, servoRightLeftPos);  
    setServoAngle(SERVO_UP_DOWN_PIN, servoUpDownPos);  
}
```

```
    delay(1000);  
}
```

Loop - citim ldr-urile si in functie de valorile obtinute calculam media celor de sus, de jos, dreapta, stanga si apoi diferenta dreapta-stanga, sus-jos si observam daca valoarea e mai mare decat toleranta, caz in care trebuie sa mutam pozitia ansamblului.

Am determinat experimental valorile intre care trebuie servo-motoarele mutate:

- pt cel dreapta-stanga -10 - 60
- pt cel sus-jos 76 - 96

In functie de pozitia soarelui sau absenta lui am afisat mesaje relevante pe lcd:

- Est jos → dimineata
- Est sus → zi
- Vest jos → amiaza
- lipsa soare → noapte

```
void loop()  
{  
    // read data from ldr  
    int topl = analogRead(LDR_TOP_LEFT_PIN);  
    int topr = analogRead(LDR_TOP_RIGHT_PIN);  
    int botl = analogRead(LDR_BOTTOM_LEFT_PIN);  
    int botr = analogRead(LDR_BOTTOM_RIGHT_PIN);  
  
    // calculating average  
    int avgtop = (topr + topl) / 2;      // average of top LDRs  
    int avgbot = (botr + botl) / 2;      // average of bottom LDRs  
    int avgleft = (topl + botl) / 2;      // average of left LDRs  
    int avgright = (topr + botr) / 2;     // average of right LDRs  
  
    // Get the differences  
    int difftopbottom = avgtop - avgbot; // difference between average of  
    // top LDRs and bottom LDRs  
    int diffrightleft = avgright - avgleft; // difference between average  
    // of right LDRs and left LDRs  
  
    // left-right movement of solar tracker  
    if (abs(diffrightleft) >= TOLERANCE) {  
        if (diffrightleft < 0) {  
            if (servoRightLeftPos <= 60) {  
                servoRightLeftPos++;  
                setServoAngle(SERVO_RIGHT_LEFT_PIN, servoRightLeftPos);  
            }  
        } else if (diffrightleft > 0) {  
            if (servoRightLeftPos >= -10) {  
                servoRightLeftPos--;  
                setServoAngle(SERVO_RIGHT_LEFT_PIN, servoRightLeftPos);  
            }  
        }  
    }  
}
```

```
}

// up-down movement of solar tracker
if (abs(difftopbottom) >= TOLERANCE) {
  if (difftopbottom < 0) {
    if (servoUpDownPos <= 96) {
      servoUpDownPos+=5;
      setServoAngle(SERVO_UP_DOWN_PIN, servoUpDownPos);
    }
  } else if (difftopbottom > 0) {
    if (servoUpDownPos >= 76) {
      servoUpDownPos-=5;
      setServoAngle(SERVO_UP_DOWN_PIN, servoUpDownPos);
    }
  }
}

// get time of the day by sun position
String direction, position;
if (topr > topl) {
  direction = "East";
} else {
  direction = "West";
}

if (botr > topr) {
  position = "High";
} else {
  position = "Low";
}

// display to the lcd the day time
lcd.setCursor(0, 1);
lcd.print("          ");
if (direction == "East" and position == "Low") {
  lcd.setCursor(8, 1);
  lcd.println("Morning!");
} else if ( direction == "West" and position == "Low") {
  lcd.setCursor(8, 1);
  lcd.println("Evening!");
} else if (direction == "East" and position == "High") {
  lcd.setCursor(9, 1);
  lcd.println("Midday!");
} else {
  lcd.setCursor(5, 1);
  lcd.println("Good night!");
}

delay(100);
}
```

Rezultate Obținute

Prototipul reuseste sa urmareasca sursa de lumina cea mai puternica cu ajutorul celor 2 servomotoare ce realizeaza miscarea pe orizontala si verticala.

In functie de localizarea soarelui, dispozitivul afiseaza un mesaj sugestiv cu perioada zilei pe lcd.



Conform analizelor descoperite online, solar tracker-ul pe 2 axe prezinta mereu rezultate mai bune decat celelalte 2 variante (varianta fixa si cea pe o singura axa), deoarece urmareste soarele de la rasarit la apus si pe tot parcursul zilei.

Concluzii

Proiectul a fost unul cat se poate de interesant, ce m-a facut sa ma adaptez in diverse situatii.

Initial mi-am dorit sa integrez o analiza de date a valorilor obtinute de panoul solar de-a lungul zilei, insa am avut dificultati cu cardul sd (ce afisa caractere random) si a trebuit sa renunt la idee pentru ca nu m-as fi incadrat in timp, insa am incercat sa compensez prin afisarea pe display a momentului zilei in functie de pozitia soarelui.

O alta problema intampinata a fost pozitionarea ansamblului panou solar - fotorezistoare ce initial a fost pozitionat prea jos, astfel ca servo-motorul sus-jos avea probleme in deplasare blocandu-se in partea de jos.

Pe viitor as vrea sa continui proiectul prin studiul valorilor obtinute de acest timp de ansamblu comparativ cu varianta fixa si sa gandesc o varianta si mai eficienta. De asemenea, mi-ar placea sa conectez niste acumulatori pentru a putea folosi in mod practic.

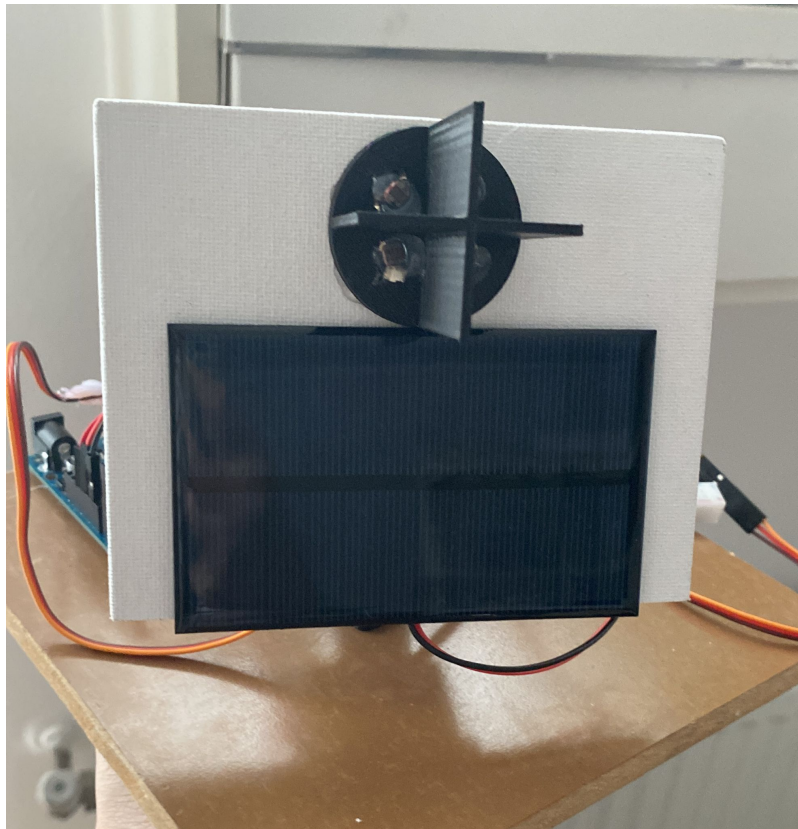
Cred foarte mult in energia alternartiva, desi momentan nu este foarte raspandita din cauza costurilor mai piperate. Este foarte important sa gasim metode prin care sa putem folosi resursele la maxim, mai ales cele ce ne tin planeta protejata de poluare.

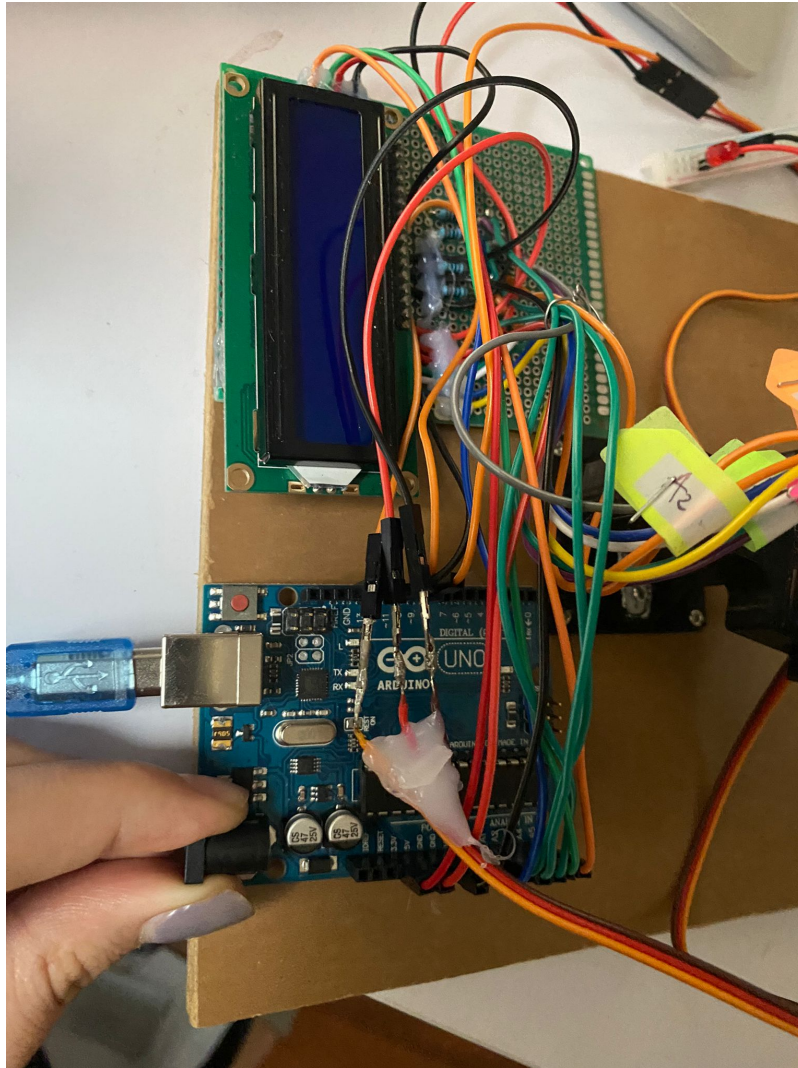
Download

[andrada_solar_tracker.zip](#)

Jurnal

- **26.04** - alegere tema proiect
- **05.05** - documentatie proiect
- **06.05** - testat componente
- **17.05** - schema electrica
- **19.05** - lipit piese pe pcb
- **22.05** - creat cod
- **23.05** - am observat ca unul dintre servo nu functioneaza corespunzator
- **26.05** - schimbat servo
- **27.05** - imbunatatit cod
- **27.05** - documentatie finala





Video functionare:

Bibliografie/Resurse

Resurse software

- https://projecthub.arduino.cc/Aboubakr_Elhammoumi/arduino-solar-tracker-77347b
- <https://arduinogetstarted.com/tutorials/arduino-lcd-i2c>

Resurse hardware

- <https://www.instructables.com/DIY-Miniature-Solar-Tracker/>
- <https://www.engineersgarage.com/solar-tracker-and-data-logger/>

Analiza tipuri de solar tracker

- [Analiza](#)
- <https://digitalcommons.cwu.edu/cgi/viewcontent.cgi?article=1094&context=undergradproj>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/vstoica/andradacojocaru>



Last update: **2023/05/30 06:22**