

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ФАХОВИЙ КОЛЕДЖ ІНФОРМАЦІЙНИХ**  
**ТЕХНОЛОГІЙ**

---

**КУРСОВИЙ ПРОЕКТ**

на тему:

Виготовлення на платі Arduino сонячного трекера з цифровим  
вольтметром на LCD дисплеї із сервоприводом SG90 та  
фоторезистором GL5506.

Виконав студент групи МТ-41

Колодій Назарій

Керівник проекту:

Остап ЮНАК

Курсовий проект

перевірений і

допущений до захисту

“\_\_\_” \_\_\_\_\_ 20\_\_ р.

Курсовий проект при захисті оцінений

---

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ  
«ФАХОВИЙ КОЛЕДЖ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

**ЗАТВЕРДЖЕНО**

на засіданні циклової комісії  
«Фундаментальної підготовки»

Протокол № \_\_\_ від \_\_\_\_\_ 2025 р.

Голова комісії

\_\_\_\_\_ Богдан ПЕЛЕЩАК

**ЗАВДАННЯ**

на курсовий проєкт

*Колодію Назарію Андрійовичу*

(прізвище, ім'я та по батькові)

**з навчальної дисципліни: ПРОГРАМНО-АПАРАТНІ ЗАСОБИ INTERNET-РЕЧЕЙ**

**Студент групи: MT-41**

**1. Тема проєкту:** Виготовлення на платі Arduino сонячного трекера з цифровим вольтметром на LCD дисплеї із сервоприводом SG90 та фоторезистором GL5506.

**2. Дата видачі завдання:** "22" вересня 2025 р

**3. Термін здачі курсового проєкту:** "15" грудня 2025 р.

**4. Вихідні дані до проєкту:**

4.1. Провести аналіз предметної галузі і вимог (огляд аналогів, функціональні й нефункціональні вимоги).

4.2. Скласти технічне завдання (ТЗ) з переліком функцій, інтерфейсів і обмежень(див. ДОДАТОК А).

4.3 Підібрати ВОР (Bill of Materials) для виконання проєкту (див. ДОДАТОК Б).

4.4 Розробити таблицю з'єднань пінів (Pin Connection Table) використаних компонентів для реалізації пристрою (див. ДОДАТОК В).

4.5 Реалізувати прошивку на Arduino (коментарі в коді, структура проєкту, використані бібліотеки).

4.6. Зробити на монтажній платі безпечною монтажу прототип розробленого пристрою на базі Arduino.

4.7. Провести налаштування, калібрування і тестування.

4.8. Підготувати користувацьку інструкцію та технічний звіт (див. ДОДАТОК Г).

**5 Перелік обов'язкових демонстраційних креслень:**

5.1 Принципова електрична схема підключення

6. Склад розрахунково – пояснювальної записки (перелік питань до розробки):

ВСТУП

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

2 ТЕХНІЧНЕ ЗАВДАННЯ

3 АПАРАТНА ЧАСТИНА

4 ПРОГРАМНА ЧАСТИНА

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

ВИСНОВКИ

ПЕРЕЛІК ПОСИЛАНЬ

### Календарний план

Назва етапів	Термін виконання	Примітка
Вступ		
1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ		
2 ТЕХНІЧНЕ ЗАВДАННЯ		
3 АПАРАТНА ЧАСТИНА		
4 ПРОГРАМНА ЧАСТИНА		
5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ		
Висновки		
Перелік посилань		

<b>Студент</b>			<b>Назарій КОЛОДІЙ</b>
	( підпис )		(імя та прізвище )
<b>Керівник проекту</b>			<b>Остап ЮНАК</b>
	( підпис )		(імя та прізвище )

## ЗМІСТ

ЗМІСТ	2
ВСТУП	3
1. ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ	5
2. ТЕХНІЧНЕ ЗАВДАННЯ	10
3. АПАРАТНА ЧАСТИНА	12
4. ПРОГРАМНА ЧАСТИНА	16
5. ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ	19
6. ВИСНОВКИ	22
7. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	24
ДОДАТКИ	25

## ВСТУП

**Мета роботи:** Розробити та реалізувати апаратно-програмний пристрій на основі мікроконтролерної платформи Arduino відповідно до заданої теми. У межах роботи передбачається створення функціонального прототипу, що виконує визначений набір завдань, а також підготовка звіту з описом конструктивних рішень, принципів роботи пристрою, програмного забезпечення та інструкції з його експлуатації.

### Актуальність теми дослідження

Актуальність даної курсової роботи обумовлена стрімким розвитком сучасних інформаційних та інженерних технологій, зокрема систем, у яких тісно поєднуються апаратні компоненти та програмне керування. У сучасних умовах мікроконтролерні платформи відіграють важливу роль як у промислових автоматизованих системах, так і в навчальному процесі підготовки фахівців технічного профілю.

### 1. Технологічний та інженерний аспект

- **1.1. Розвиток екосистеми IoT (Internet of Things).** Сучасні системи автоматизації, моніторингу та дистанційного керування базуються на використанні компактних, енергоефективних і доступних обчислювальних модулів. Платформа Arduino є одним із найпоширеніших рішень для створення прототипів таких систем, оскільки дозволяє легко взаємодіяти з фізичними об'єктами, датчиками та виконавчими механізмами. Це повністю відповідає глобальним тенденціям розвитку Інтернету речей, де ключову роль відіграють вбудовані системи.
- **1.2. Демократизація процесу розробки.** Однією з головних переваг Arduino є її відкрита архітектура та підтримка концепції open-source. Завдяки цьому значно знижується поріг входження для початківців у сфері електроніки та програмування. Розробник може зосередитися на логіці роботи пристрою та його функціональних можливостях, не витрачаючи значний час на налаштування низькорівневих апаратних параметрів.

- **1.3. Міждисциплінарна інтеграція.** Створення апаратно-програмних комплексів потребує поєднання знань з кількох галузей: електроніки, програмування мікроконтролерів мовами сімейства C/C++, теорії автоматичного керування та проєктування інтерфейсів взаємодії з користувачем. Робота над подібними проєктами сприяє формуванню комплексного інженерного мислення, що є надзвичайно важливим для майбутніх спеціалістів у галузях автоматизації, робототехніки та вбудованих систем.

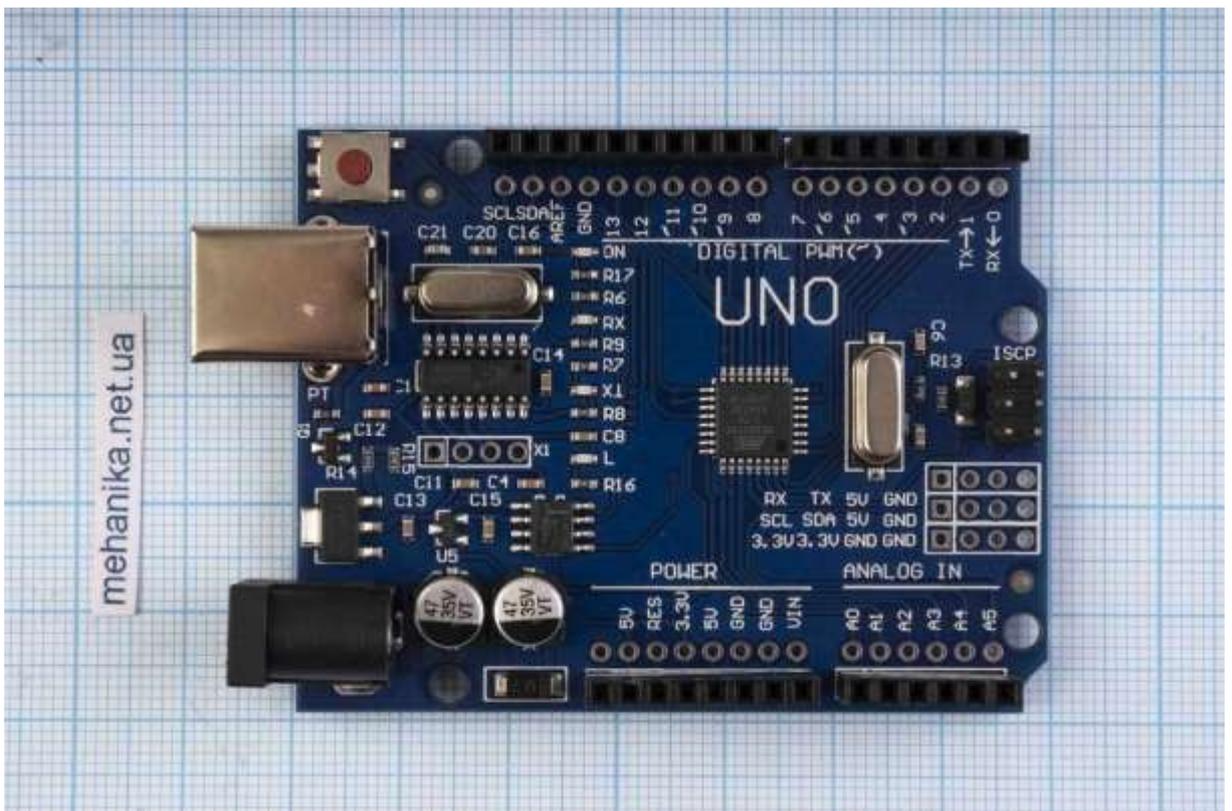
## 2. Прикладний та практичний вимір

- **2.1. Реалізація прикладних рішень.** Дана курсова робота має прикладний характер і передбачає створення реального фізичного пристрою, здатного виконувати функції збору та обробки даних, а також автоматизованого керування. Це дозволяє продемонструвати практичне застосування теоретичних знань, отриманих під час навчання.
- **2.2. Економічна ефективність.** Використання платформи Arduino дає змогу реалізувати необхідний функціонал із мінімальними фінансовими та часовими витратами. Завдяки доступності компонентної бази такі рішення є привабливими для освітніх проєктів, прототипування стартапів і створення систем «розумного дому».
- **2.3. Потенціал масштабування.** Розроблений у межах курсової роботи пристрій може слугувати основою для подальшої модернізації та розширення функціональних можливостей. Це відкриває перспективи використання отриманих результатів у дипломному проєктуванні, наукових дослідженнях або практичних інженерних розробках.

# 1. ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

**1.1 Апаратно-програмна платформа Arduino** — це відкрита апаратно-програмна платформа, призначена для розробки та прототипування електронних пристроїв різного рівня складності. Вона поєднує у собі мікроконтролерну плату з уже інтегрованими базовими вузлами живлення та введення/виведення, а також програмне середовище для створення та завантаження керуючого коду. Такий підхід дозволяє швидко переходити від ідеї до практичної реалізації пристрою.

Рис. 1.1 Вигляд плати Arduino з мікроконтролером ATmega328P



**Історія створення:** Проект Arduino було започатковано у 2005 році в Інституті дизайну взаємодії (Interaction Design Institute Ivrea) в Італії. Ініціаторами розробки стали Массімо Банці, Девід Куартільєс, Том Іго, Джанлука Мартіно та Девід Мелліс. Основною метою створення платформи було надання студентам доступного

та зручного інструменту для реалізації інтерактивних електронних проєктів без необхідності глибокого занурення в складні аспекти схемотехніки та низькорівневого програмування.

**Причини вибору платформи:** Для реалізації даної курсової роботи було обрано плату Arduino Uno R3, побудовану на мікроконтролері ATmega328P. Такий вибір пояснюється низкою переваг, які роблять цю платформу оптимальною для створення сонячного трекера:

**Доступність:** Низька вартість плати та компонентів;

**Кросплатформеність:** середовище Arduino IDE функціонує під управлінням операційних систем Windows, macOS та Linux;

**Наявність АЦП:** вбудований аналого-цифровий перетворювач дозволяє безпосередньо зчитувати сигнали з фоторезисторів і виконувати вимірювання напруги;

**Програмна підтримка:** велика кількість готових бібліотек (зокрема Servo.h та LiquidCrystal\_I2C.h) значно спрощує роботу з сервоприводами та дисплеями.

## 1.2 Мова програмування C++ та середовище Arduino IDE

Програмування мікроконтролерів у межах платформи Arduino здійснюється мовою C++, адаптованою під спрощений фреймворк Wiring. Даний фреймворк надає набір готових функцій, бібліотек і шаблонів, які істотно скорочують час розробки програмного забезпечення та спрощують керування апаратними ресурсами мікроконтролера.

Arduino IDE забезпечує зручний інтерфейс для написання програм, компіляції коду та його завантаження в мікроконтролер через USB-інтерфейс. Розробнику не потрібно вручну налаштовувати таймери, регістри чи переривання, оскільки більшість низькорівневих процесів реалізовані на рівні бібліотек.

Мова C++ у контексті Arduino поєднує високу продуктивність із достатньою простотою синтаксису. Це дозволяє ефективно виконувати математичні обчислення, необхідні для аналізу сигналів з датчиків освітленості та розрахунку значень напруги, зберігаючи при цьому стабільність та швидкодію системи.

Програма для Arduino, яка називається **скетчем**, обов'язково містить дві ключові функції:

- **setup()** — виконується один раз після запуску пристрою та використовується для ініціалізації периферійних модулів, сервоприводу та дисплея;
- **loop()** — виконується циклічно та реалізує основний алгоритм роботи пристрою, зокрема стеження за джерелом світла та вимірювання напруги.

### 1.3 Принципи роботи використаних компонентів

Сервопривід SG90 є компактним виконавчим механізмом, який дозволяє здійснювати точне позиціонування валу в межах визначеного діапазону кутів. Керування його положенням відбувається за допомогою широтно-імпульсної модуляції (ШИМ). Мікроконтролер формує імпульси певної тривалості, які електронна схема сервоприводу перетворює на відповідний кут повороту, зазвичай у діапазоні від 0 до 180 градусів. Це забезпечує можливість точного орієнтування сонячної панелі на джерело світла.

Рис 1.3.1 Вигляд сервоприводу SG90



Фоторезистор GL5506 працює на основі явища фотопровідності, при якому електричний опір напівпровідникового матеріалу змінюється залежно від інтенсивності освітлення. У темряві опір елемента є дуже великим, тоді як при потраплянні світла він істотно зменшується. У схемі сонячного трекера два фоторезистори використовуються як елементи дільників напруги, що дозволяє мікроконтролеру порівнювати рівень освітленості з різних сторін і визначати напрямок на найяскравіше джерело світла.

Рис 1.3.2 Вигляд фоторезистора GL5506



Принцип цифрового вольтметра (Дільник напруги). Для вимірювання напруги, значення якої перевищує допустимий рівень для аналогових входів Arduino, застосовується резистивний дільник напруги. Його робота ґрунтується на законі Ома, відповідно до якого вхідна напруга розподіляється між резисторами пропорційно їх опорам.

Мікроконтролер вимірює знижену напругу на одному з плечей дільника, після чого програмно обчислює реальне значення напруги джерела живлення.

Рідкокристалічний дисплей (LCD 1602) з I2C. Для відображення інформації про поточний кут повороту сервоприводу та виміряну напругу використовується символний LCD дисплей формату 16×2. Застосування інтерфейсу I2C дозволяє скоротити кількість задіяних виводів мікроконтролера до двох сигнальних ліній — SDA та SCL. Це значно спрощує схему підключення та підвищує зручність монтажу пристрою.

Рис 1.3.3 Вигляд рідкокристалічного дисплею (LCD 1602) з I2C



## **2. ТЕХНІЧНЕ ЗАВДАННЯ**

### **1. Назва роботи**

Виготовлення на платі Arduino сонячного трекера з цифровим вольтметром на LCD дисплеї із сервоприводом SG90 та фоторезистором GL5506.

### **2. Мета роботи**

Виготовити на платі Arduino сонячного трекера з цифровим вольтметром на LCD дисплеї із сервоприводом SG90 та фоторезистором GL5506.

### **3. Завдання**

1. Ознайомитись з інструкцією до практичної роботи;
2. Ознайомитись із принциповою електричною схемою охоронний пристрою на Arduino Uno;
3. Зібрати на монтажній платі цифрову схему охоронного пристрою згідно електричної принципової схеми
4. Написати скетч роботи світлофора на Arduino Uno;

### **4. Технічні вимоги**

- Живлення від джерела постійного струму напругою 5 В (USB-порт персонального комп'ютера, PowerBank або автономне джерело живлення);
- Використання мікроконтролерної плати Arduino Uno R3 на базі ATmega328P;
- Застосування безпайкової макетної плати типу breadboard для монтажу електронних компонентів;

- Наявність LCD дисплея формату 1602 з інтерфейсом I2C для відображення текстової інформації;
- Використання сервоприводу SG90 як виконавчого механізму для повороту платформи;
- Застосування фоторезисторів GL5506 для визначення рівня освітленості;
- Використання резисторів необхідних номіналів для реалізації ділників напруги;
- Забезпечення коректного з'єднання компонентів за допомогою з'єднувальних провідників.

## **5. Очікуваний результат**

Результатом виконання курсової роботи має бути працездатний прототип сонячного трекера, який автоматично орієнтується на джерело світла, здійснює вимірювання напруги та відображає відповідні дані на LCD дисплеї, підтверджуючи виконання поставленого технічного завдання.

### 3. АПАРАТНА ЧАСТИНА

#### Підбір BOM

Для реалізації апаратної частини сонячного трекера було сформовано перелік необхідних електронних компонентів та допоміжних матеріалів. **BOM (Bill of Materials)** — це структурований список елементів, які використовуються для складання пристрою та забезпечують його повноцінне функціонування. Такий підхід дозволяє систематизувати компонентну базу, спростити монтаж і подальше обслуговування макета.

Таблиця 3.1.1 Перелік необхідних компонентів

№	Назва компонента	Кількість	Позначення на схемі	Характеристики	Примітка
1	Arduino Uno	1	U1	Мікроконтролер ATmega328P	Основна плата керування
2	Плата breadboard	1	-	830 точок, безпайкова	Для монтажу схеми
3	Servo SG90	1	M1	Кут повороту 0–180°, 5В	Поворотний механізм платформи
4	Фоторезистор GL5506	2	LDR1, LDR2	Опір залежить від освітлення	Датчики світла (Лівий/Правий)

<b>№</b>	<b>Назва компонента</b>	<b>Кількість</b>	<b>Позначення на схемі</b>	<b>Характеристики</b>	<b>Примітка</b>
<b>5</b>	<b>LCD 1602 (I2C)</b>	<b>1</b>	<b>DIS1</b>	<b>16x2 символів, I2C модуль</b>	<b>Відображення кути та напруги</b>
<b>6</b>	<b>Резистор</b>	<b>2</b>	<b>R1, R2</b>	<b>10 кОм</b>	<b>Підтягуючі резистори для LDR</b>
<b>7</b>	<b>Резистор</b>	<b>1</b>	<b>R3</b>	<b>30 кОм</b>	<b>Верхнє плече дільника напруги</b>
<b>8</b>	<b>Резистор</b>	<b>1</b>	<b>R4</b>	<b>7.5 кОм</b>	<b>Нижнє плече дільника напруги</b>
<b>9</b>	<b>Джерело живлення</b>	<b>1</b>	<b>BAT1</b>	<b>Батарея 9В або Сонячна панель</b>	<b>Об'єкт вимірювання вольтметром</b>
<b>10</b>	<b>Провідники</b>	<b>15</b>	<b>-</b>	<b>Male-Male, Male-Female</b>	<b>Комутація компонентів</b>

Таблиця 3.1.2 Таблиця з'єднань пінів

№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
1	Servo SG90	Signal (Помаранчевий)	Digital Pin 9	ШИМ-сигнал керування
		VCC (Червоний)	5V	Живлення сервоприводу
		GND (Коричневий)	GND	Спільна земля
2	Фоторезистори	LDR1 (Лівий) + R1	Analog Pin A0	Дільник напруги (сигнал)
		LDR2 (Правий) + R2	Analog Pin A1	Дільник напруги (сигнал)
		Вільні виводи LDR	5V	Живлення датчиків

<b>№</b>	<b>Компонент</b>	<b>Вивід компонента</b>	<b>З'єднано з піном Arduino</b>	<b>Примітка / Функція</b>
<b>3</b>	<b>Вольтметр</b>	<b>Точка з'єднання R3/R4</b>	<b>Analog Pin A2</b>	<b>Вхід вимірювання (0-5В)</b>
		<b>Вхід R3 (30к)</b>	<b>Джерело вимірювання (+)</b>	<b>"Плюсовий щуп"</b>
		<b>Вхід R4 (7.5к)</b>	<b>GND</b>	<b>Спільна земля</b>
<b>4</b>	<b>LCD 1602 (I2C)</b>	<b>SDA</b>	<b>Analog Pin A4 (або SDA)</b>	<b>Шина даних I2C</b>
		<b>SCL</b>	<b>Analog Pin A5 (або SCL)</b>	<b>Шина тактування I2C</b>
		<b>VCC / GND</b>	<b>5V / GND</b>	<b>Живлення дисплея</b>

## 4. ПРОГРАМНА ЧАСТИНА

Програмне забезпечення для керування розробленим пристроєм було створене з використанням інтегрованого середовища розробки **Arduino IDE**. Як мову програмування обрано **C++**, адаптовану для роботи з мікроконтролерними платформами Arduino. Для спрощення взаємодії з апаратними компонентами застосовано стандартні бібліотеки **Servo.h** для керування сервоприводом та **LiquidCrystal\_I2C.h** для роботи з рідкокристалічним дисплеєм через інтерфейс I2C.

Основна логіка програми полягає в аналізі сигналів, отриманих з двох фоторезисторів, та визначенні напрямку найбільшої інтенсивності освітлення. На основі цих даних здійснюється корекція положення сервоприводу, який орієнтує платформу в напрямку джерела світла. Одночасно з цим виконується вимірювання напруги за допомогою аналогового входу мікроконтролера та її відображення на LCD дисплеї.

### Опис алгоритму роботи:

1. **Оголошення змінних:** На початку програми задаються номери пінів для підключення сервоприводу, фоторезисторів та входу вольтметра. Також вводиться змінна чутливості (*tolerance*), яка використовується для створення гістерезису та запобігання частим незначним коливанням сервоприводу при мінімальних змінах освітлення.
2. **Ініціалізація (setup):** У функції `setup()` відбувається підключення сервоприводу до відповідного цифрового піна, ініціалізація LCD дисплея, увімкнення підсвітки та виведення стартового повідомлення. Сервопривід встановлюється у початкове нейтральне положення (90 градусів).
3. **Основний цикл (loop):**
  - зчитування аналогових значень з лівого та правого фоторезисторів;
  - визначення різниці між отриманими значеннями та порівняння її з порогом чутливості;
  - зміна кута повороту сервоприводу у напрямку більш освітленого датчика;
  - контроль допустимого діапазону кута повороту (від 0 до 180 градусів);
  - зчитування значення напруги з аналогового входу, перетворення його у фізичне значення та корекція з урахуванням коефіцієнта дільника напруги;
  - виведення поточних значень кута повороту та напруги на LCD дисплей.

## Код програми

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

const int pinServo = 9;
const int pinLDRLeft = A0;
const int pinLDRRight = A1;
const int pinVolt = A2;

Servo myServo;
LiquidCrystal_I2C lcd(0x20, 16, 2);

int angle = 90;
int tolerance =
20; float vout =
0.0; float vin =
0.0;

float R1 =
30000.0; float R2
= 7500.0;

void setup() {
  myServo.attach(pinServo);
  myServo.write(angle);

  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Solar
Tracker"); delay(1000);
  lcd.clear();
}

void loop() {
  int valLeft = analogRead(pinLDRLeft);
```

```
int valRight =
analogRead(pinLDRRight); int diff =
valLeft - valRight;

if (abs(diff) > tolerance) {
  if (valLeft > valRight) {
    angle = angle - 1;
  } else {
    angle = angle + 1;
  }
}

if (angle > 180) angle = 180;
if (angle < 0) angle = 0;

myServo.write(angle);

int sensorValue = analogRead(pinVolt);
vout = (sensorValue * 5.0) / 1024.0;
vin = vout / (R2 / (R1 + R2));

if (vin < 0.1) vin = 0.0;

lcd.setCursor(0,
0);
lcd.print("Ang:");
lcd.print(angle);
lcd.print("  ");

lcd.setCursor(0, 1);
lcd.print("Volt:");
lcd.print(vin);
lcd.print("V ");

delay(50);
}
```

## 5. ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

Перевірка працездатності розробленого пристрою «Сонячний трекер з цифровим вольтметром» проводилася з метою підтвердження коректності роботи як апаратної, так і програмної частин системи. Процес тестування складався з двох основних етапів: попередньої перевірки апаратного монтажу та практичного функціонального випробування програмного забезпечення.

**1. Підготовка до тестування** Перед підключенням живлення було виконано ретельний візуальний контроль змонтованої схеми на макетній платі. Особливу увагу приділено правильності підключення основних компонентів та відповідності їх з'єднань принциповій схемі.

- коректність підключення сервоприводу SG90, зокрема сигнального проводу до цифрового піна 9 та ліній живлення до виводів 5V і GND;
- правильність під'єднання LCD дисплея через інтерфейс I2C (лінії SDA та SCL), а також відсутність коротких замикань;
- відповідність номіналів резисторів, використаних у дільниках напруги для фоторезисторів GL5506 та вхідного вольтметра;
- надійність усіх контактів і з'єднувальних провідників на платі breadboard.

**2. Процес тестування** Після підключення плати Arduino Uno до персонального комп'ютера через USB-інтерфейс було виконано компіляцію програмного коду та його завантаження у пам'ять мікроконтролера. Подальше тестування здійснювалося шляхом моделювання реальних умов експлуатації пристрою. Для перевірки роботи системи стеження за світлом використовувалося штучне джерело освітлення (ліхтарик), яке по черзі розташовувалося з лівого та правого боків фоторезисторів. Перевірка роботи цифрового вольтметра проводилася шляхом подачі різних значень напруги на вхід

дільника напруги.

В ході тестування зафіксовано наступний алгоритм роботи:

1. **Ініціалізація:** Після подачі живлення сервопривід встановлюється у початкове положення ( $90^\circ$ ), а на LCD дисплеї відображається стартове повідомлення, після чого виводиться основна інформація про кут повороту та виміряну напругу.
2. **Реакція на світло зліва:** При збільшенні рівня освітленості лівого фоторезистора (пін A0) сервопривід плавно змінює положення, повертаючи платформу у відповідному напрямку до моменту вирівнювання сигналів з обох датчиків.
3. **Реакція на світло справа:** При переміщенні джерела світла до правого фоторезистора система коректно визначає зміну різниці сигналів, унаслідок чого сервопривід повертає платформу в протилежний бік.
4. **Робота вольтметра:** Значення напруги відображається на другому рядку LCD дисплея та оновлюється в режимі реального часу. При зміні вхідної напруги показання змінюються плавно, без затримок і з допустимою похибкою.
5. **Стабілізація:** За умови однакового рівня освітленості на обох фоторезисторах або при різниці значень, меншій за заданий поріг чутливості, сервопривід припиняє рух. Це свідчить про ефективну роботу механізму гістерезису та відсутність небажаних коливань.

### 3. Результати

У ході проведених випробувань було встановлено, що:

- алгоритм керування сервоприводом адекватно реагує на зміну освітленості та забезпечує правильне орієнтування платформи на джерело світла;

- вимірювання напруги здійснюється з достатньою точністю, а результати коректно відображаються на LCD дисплеї;
- робота сервоприводу SG90 є плавною та стабільною, без ривків і надмірних механічних вібрацій;
- обмін даними через інтерфейс I2C відбувається без збоїв, інформація на дисплеї відображається чітко та стабільно.

Отримані результати підтверджують, що розроблений пристрій повністю відповідає вимогам технічного завдання та може використовуватися як функціональний макет системи автоматичного стеження за джерелом світла з можливістю моніторингу напруги.

## 6. ВИСНОВКИ

У процесі виконання курсової роботи було розроблено та реалізовано працездатний макет сонячного трекера з функцією цифрового вимірювання напруги, побудований на базі мікроконтролерної платформи Arduino Uno. Поставлену мету роботи досягнуто в повному обсязі, а всі завдання, визначені технічним завданням, успішно виконано.

### **Висновки:**

**Апаратна реалізація:** Було опрацьовано принципи взаємодії мікроконтролера з аналоговими датчиками освітленості на базі фоторезисторів GL5506, а також з виконавчим механізмом у вигляді сервоприводу SG90. За допомогою безпайкової макетної плати реалізовано схему підключення всіх компонентів, включно з резистивними дільниками напруги для вимірювання вхідної напруги та підключенням рідкокристалічного дисплея з інтерфейсом I2C.

**Програмна реалізація:** У середовищі Arduino IDE розроблено програмний код мовою C++, який забезпечує автоматичне порівняння сигналів з двох датчиків освітленості та керування положенням сервоприводу відповідно до напрямку найбільш інтенсивного світлового потоку. Також реалізовано алгоритм обробки аналогового сигналу з вольтметра, його математичний перерахунок та виведення результатів на LCD дисплей.

**Працездатність системи:** Проведені випробування підтвердили стабільну роботу розробленої системи. Сонячний трекер коректно орієнтується на джерело світла, забезпечуючи потенційне підвищення ефективності роботи сонячної панелі, а цифровий вольтметр відображає актуальні значення напруги в режимі реального часу без помітних затримок чи збоїв.

## **Перспективи розвитку:**

Розроблений макет сонячного трека є базовою моделлю, яку можна вдосконалювати та розширювати. До можливих напрямів подальшої модернізації належать:

- реалізація двовісної системи стеження шляхом додавання другого сервоприводу та додаткових датчиків освітленості, що дозволить відслідковувати положення сонця як по горизонталі, так і по вертикалі;
- забезпечення енергонезалежної роботи системи шляхом інтеграції акумулятора та контролера заряду для живлення безпосередньо від сонячної панелі;
- розширення функціоналу вимірювань шляхом додавання датчика струму з можливістю розрахунку споживаної потужності;
- впровадження режимів енергозбереження, зокрема автоматичного переходу в режим сну в нічний час та повернення платформи у початкове положення зранку;
- розробка захисного корпусу та додаткових сенсорів для підвищення надійності роботи пристрою в умовах зовнішнього середовища.

## 7. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Блум Д. Вивчаємо Arduino: інструменти та методи технічного чарівництва / Джеремі Блум. — К.: БХВ, 2016. — 336 с.

Шилдт Г. С++: базовий курс / Герберт Шилдт. — 3-тє вид. — М.: Вільямс, 2019. — 624 с.

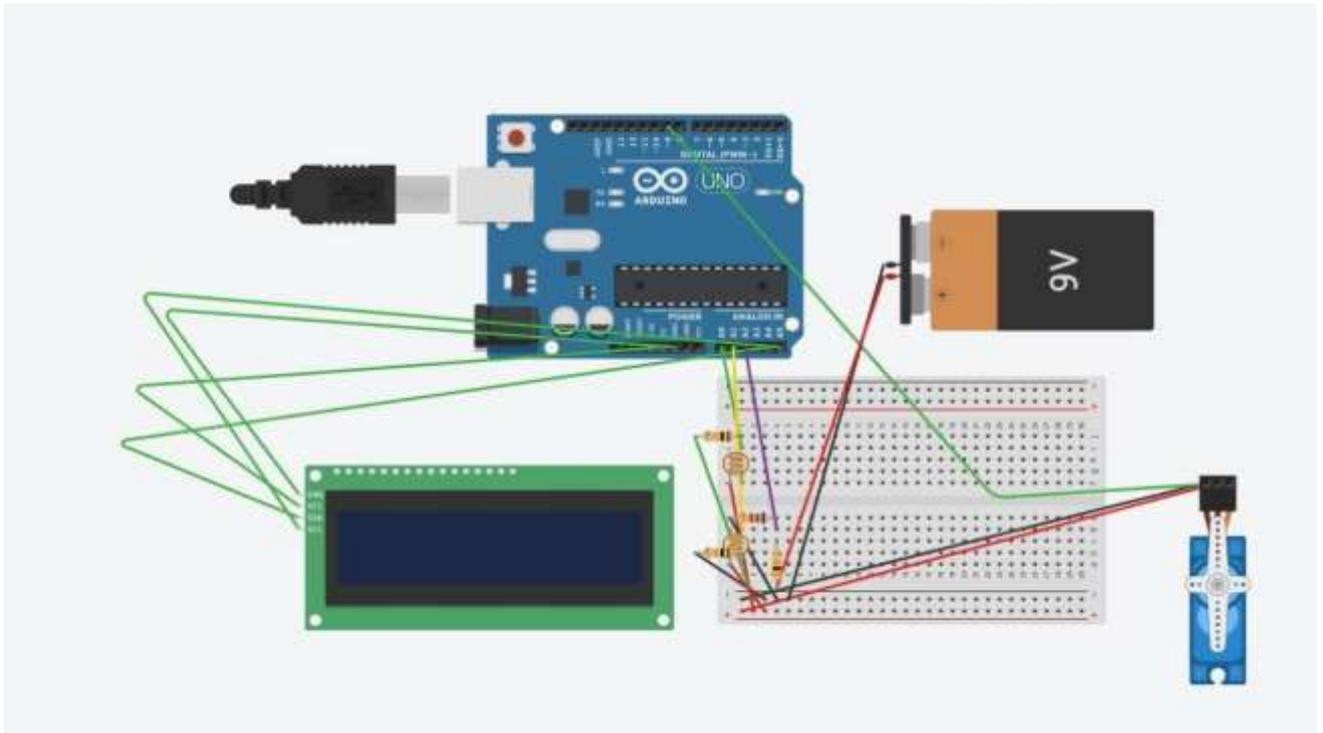
Петренко А. І. Основи мікропроцесорної техніки та програмування мікроконтролерів: навчальний посібник / А. І. Петренко. — К.: Каравела, 2020. — 288 с.

Рюмик С. М. Мікроконтролери AVR: практикум для початківців / С. М. Рюмик. — Х.: Ранок, 2018. — 112 с.

Галкін В. І. Промислова електроніка та мікросхемотехніка: підручник / В. І. Галкін. — К.: Вища школа, 2017. — 320 с.

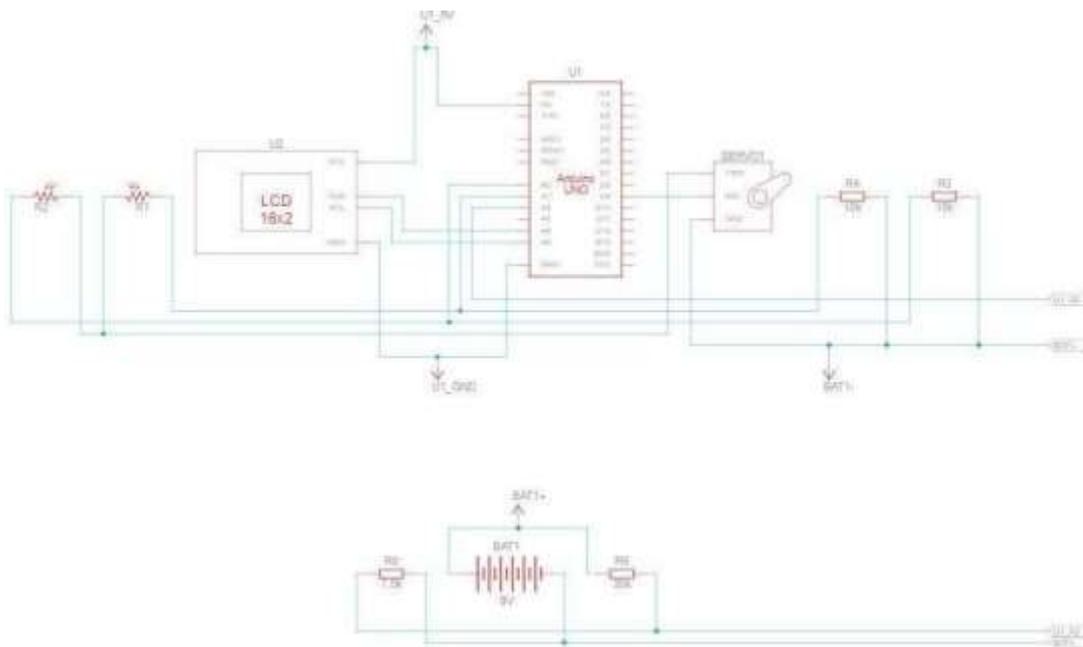
Офіційна документація Arduino [Електронний ресурс]. – Режим доступу: <https://www.arduino.cc/reference/en/>

## ДОДАТОК 1



Структурна схема реалізації моделі

## ДОДАТОК 2



Принципова електрична схема підключення