

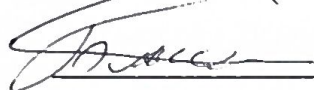
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій

Бакалаврська кваліфікаційна робота на тему:

«Автоматизована система контролю освітленості в навчальних аудиторіях»

Виконав: студент IV курсу, групи АКІТ-21бз
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології

(шифр і назва спеціальності)

 Блашук М. Р.

(прізвище та ініціали)

Керівник: д.т.н., проф. кафедри АІТ

Бісікало О.В.

(прізвище та ініціали)

«17» 06 2025 р.

Рецензент: к.т.н., доц. кафедри САІТ

Козачко О. М.

(прізвище та ініціали)

«18» 06 2025 р.

Допущено до захисту
завідувач кафедри АІТ
Олег БІСІКАЛО

«19» 06 2025 р.

Вінницький національний технічний університет
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій
Рівень вищої освіти перший (бакалаврський)
Галузь знань 15 – Автоматизація та приладобудування
Спеціальність 151 – Автоматизація та комп'ютерно-інтегровані технології
Освітньо-професійна програма Інтелектуальні комп'ютерні системи управління

ЗАТВЕРДЖУЮ

Зав. кафедри АІТ

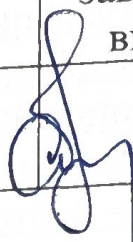
д.т.н., проф. Бісікало О.В.

«18» 06 2025 року

ЗАВДАННЯ
НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Блащуку Максиму Романовичу
(прізвище, ім'я, по батькові)

1. Тема роботи «Автоматизована система контролю освітленості в навчальних аудиторіях»
керівник роботи Бісікало О. В., д.т.н, проф.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу «20» березня 2025 р. №97
2. Термін подання студентом роботи 10.06.2025 р.
3. Вихідні дані до роботи:
 - тип системи – автоматизована система контролю освітленості;
 - сенсорний (фоторезисторний та цифровий датчик освітленості ВН1750);
 - рівень освітленості в навчальних аудиторіях – не менше 200 лк.
4. Зміст текстової частини (перелік питань, які потрібно розробити) Вступ. Аналіз існуючих методів контролю освітленості в навчальних аудиторіях. Розробка структурної схеми та алгоритма роботи системи контролю освітленості в навчальних аудиторіях. Розробка функціонально-апаратної схеми системи контролю освітленості в навчальних аудиторіях. Програмна реалізація системи контролю освітленості в навчальних аудиторіях. Висновки.
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень) Алгоритм роботи системи освітлення. Зовнішній вигляд сервопривода TowerPro SG 5010. Схема під'єднання світлодіодного пристрою. Інформаційна модель системи. Діаграма прецедентів системи.

6. Консультанти розділів роботи		Підпис, дата	
Розділ	Прізвище, ініціали та посада консультанта	завдання видав	завдання прийняв
1-4	Бісікало О.В., д.т.н, проф.		

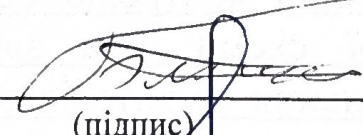
7. Дата видачі завдання 20.03.2025 р.

КАЛЕНДАРНИЙ ПЛАН

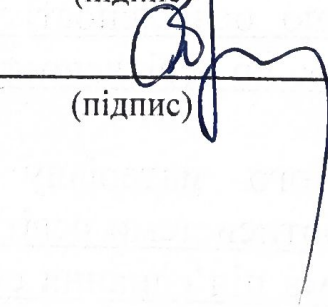
№ з/п	Назва етапів бакалаврської кваліфікаційної роботи	Термін виконання		Примітка
		початок	закінчення	
1	Вибір, узгодження та затвердження теми БКР	03.10.2024	09.10.2024	<i>Вик</i>
2	Аналіз літературних джерел. Попередня розробка основних розділів	12.03.2025	29.03.2025	<i>Вик</i>
3	Розробка технічного завдання (ТЗ)	01.04.2025	26.04.2025	<i>Вик</i>
4	Аналіз вирішення поставленої задачі	29.04.2025	10.05.2025	<i>Вик</i>
5	Розробка структурної схеми	13.05.2025	24.05.2025	<i>Вик</i>
6	Електричні розрахунки	27.05.2025	31.05.2025	<i>Вик</i>
7	Оформлення пояснювальної записки (ПЗ) та графічної частини	03.06.2025	07.06.2025	<i>Вик</i>
8	Перевірка ПЗ на відповідність вимогам	10.06.2025	15.06.2025	<i>Вик</i>
9	Попередній захист БДР, доопрацювання, рецензування БДР	15.06.2025	16.06.2025	<i>Вик</i>
10	Захист БКР ЕК	16.06.2025	23.06.2025	<i>Вик</i>

Студент

Керівник роботи


(підпис)

Блащук М.Р.
(прізвище та ініціали)


(підпис)

Бісікало О.В.
(прізвище та ініціали)

АНОТАЦІЯ

Бакалаврська кваліфікаційна робота складається з 72 сторінок формату А4, на яких є 23 рисунки, 2 таблиці, список використаних джерел містить 19 найменувань.

У кваліфікаційній роботі розроблено автоматизовану систему контролю освітленості в навчальних аудиторіях із використанням сенсорних технологій та мікроконтролера ESP8266. Система забезпечує моніторинг рівня освітленості за допомогою фоторезисторів і цифрового датчика BH1750, обробку даних в режимі реального часу та автоматичне регулювання інтенсивності штучного освітлення шляхом керування RGB-світлодіодами. Основна увага приділена створенню структурної та функціональної схем, моделюванню процесів освітлення, а також розробці програмного забезпечення, що включає веб-інтерфейс для дистанційного налаштування. Представлена система підвищує енергоефективність, комфорт і адаптивність навчального середовища, а також може бути інтегрована до сучасних смарт-рішень освітлення.

Ключові слова: освітленість, автоматизована система, фоторезистор, BH1750, ESP8266, мікроконтролер, RGB-світлодіод, веб-інтерфейс, керування освітленням, навчальна аудиторія.

ANNOTATION

The bachelor's qualification work consists of 72 pages of A4 format, which contain 23 figures, 2 tables, and the list of sources used contains 19 items.

This qualification work presents the development of an automated lighting control system for educational classrooms using sensor technologies and the ESP8266 microcontroller. The system monitors light levels in real time via photoresistors and the digital BH1750 sensor, processes the data, and automatically adjusts artificial lighting by controlling RGB LEDs. The study focuses on the creation of structural and functional diagrams, the modeling of lighting processes, and the development of software including a web interface for remote configuration. The proposed system enhances energy efficiency, comfort, and adaptability of the learning environment and can be integrated into modern smart lighting solutions.

Keywords: illumination, automated system, photoresistor, BH1750, ESP8266, microcontroller, RGB LED, web interface, lighting control, classroom.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ	6
1.1 Методи контролю освітленості.....	6
1.2 Огляд сучасних рішень контролю освітленості.....	9
1.3 Висновки до розділу	15
2 РОЗРОБКА СТРУКТУРНОЇ СХЕМИ ТА АЛГОРИТМА РОБОТИ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ.....	16
2.1 Структурна схема системи контролю освітленості в навчальних аудиторіях	16
2.2 Алгоритм роботи системи контролю освітленості в навчальних аудиторіях	20
2.3 Висновки до розділу 2.....	22
3 РОЗРОБКА ФУНКЦІОНАЛЬНО-АПАРATНОЇ СХЕМИ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ.....	23
3.1 Апаратний склад системи	23
3.2 Технічні характеристики сенсора освітленості BH1750.....	31
3.3 Математична модель освітленості в навчальних аудиторіях	39
3.4 Висновки до розділу 3.....	40
4 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ	41
4.1 Програмна реалізація сенсорної частини системи контролю освітленості.....	41
4.2 Програмна реалізація та програмне забезпечення виконавчої частини системиконтролю освітленості	42
4.3 Висновки до розділу 4.....	51

ВИСНОВКИ.....	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	53
ДОДАТКИ.....	55
Додаток А (Обов’язковий) Автоматизована система контролю освітленості в навчальних аудиторіях. Технічне завдання.....	56
Додаток Б (Обов’язковий) Автоматизована система контролю освітленості в навчальних аудиторіях. Ілюстративна частина.....	59
Додаток В (Обов’язковий) Автоматизована система контролю освітленості в навчальних аудиторіях. Лістинг програми.....	63
Додаток Г (Обов’язковий) Автоматизована система контролю освітленості в навчальних аудиторіях. Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень.....	72

ВСТУП

Актуальність теми. Розробка автоматизованої системи контролю освітленості в навчальних аудиторіях є актуальною з кількох ключових причин. Якість освітлення безпосередньо впливає на зорове навантаження, концентрацію уваги й загальне самопочуття учнів і студентів. Недостатній або надмірний рівень світла погіршує сприйняття матеріалу, спричиняє втому та знижує ефективність освітнього процесу.

Автоматизовані системи дозволяють динамічно регулювати яскравість штучного освітлення залежно від рівня природного світла, що дає змогу зменшити витрати електроенергії.

Впровадження smart-технологій у закладах освіти відповідає сучасним трендам цифровізації, підвищуючи технічний рівень установ і їхню інтеграцію в систему інтелектуального керування будівлями.

Таким чином, створення системи контролю освітленості сприяє покращенню умов навчання, зниженню енергоспоживання та розвитку сучасної освітньої інфраструктури.

Мета дослідження. Метою роботи є підвищення ефективності та енергоощадності освітлення в навчальних аудиторіях шляхом розробки автоматизованої системи моніторингу та регулювання рівня освітленості з використанням сенсорних технологій і мікроконтролера ESP8266.

Для досягнення поставленої мети в роботі сформульовані та вирішені такі **задачі**:

- аналіз вимог до рівня освітленості в навчальних приміщеннях відповідно до нормативних документів;
- обґрунтування вибору сенсорів освітлення (фоторезистор, BH1750) для інтеграції в систему;
- побудова структурної та функціональної схем системи керування;
- моделювання алгоритму автоматичного регулювання освітлення на основі даних із сенсорів;

- розробка програмного забезпечення з використанням мікроконтролера ESP8266 та реалізація веб-інтерфейсу;
- створення прототипу системи та проведення тестування в реальних умовах;
- аналіз точності, стабільності та енергоспоживання системи.

Об'єктом дослідження в роботі є процес керування штучним освітленням у навчальних приміщеннях з урахуванням рівня природного освітлення.

Предметом дослідження є методи та технічні засоби сенсорного контролю освітленості, а також алгоритми автоматичного регулювання на основі мікроконтролера.

Практичне значення одержаних результатів. У результаті дослідження розроблено структурно-функціональну модель автоматизованої системи керування освітленням, що дозволяє в режимі реального часу моніторити рівень освітленості та регулювати яскравість світлодіодного освітлення відповідно до умов середовища. Система забезпечує підвищення комфорту в навчальних аудиторіях, скорочення витрат енергії та може бути легко інтегрована в сучасну інфраструктуру розумних навчальних закладів.

Результати роботи доповідались та обговорювались на LIV Всеукраїнській науково-технічній конференції підрозділів Вінницького національного університету НТКП ВНТУ (2025) [19].

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ

1.1 Методи контролю освітленості

Освітленість є важливим фактором, що впливає на якість навчального процесу [1]. Від рівня освітленості залежить зоровий комфорт студентів, продуктивність навчання та загальний стан здоров'я. Саме тому контроль освітленості в навчальних аудиторіях є необхідним елементом організації освітнього середовища [2].

Існує кілька основних методів контролю рівня освітленості: ручний, напівавтоматичний та автоматизований. Найпростіший метод — ручний контроль, коли викладач або технічний персонал самостійно оцінює освітленість і вмикає або вимикає освітлювальні прилади. Такий підхід є малоефективним і не дозволяє оперативно реагувати на зміну умов освітлення, наприклад, при зміні погоди або часу доби.

Напівавтоматичний контроль передбачає використання простих світлових датчиків, які сигналізують про недостатній або надмірний рівень освітлення. Проте остаточне рішення про зміну освітлення все ще приймає людина. Це покращує точність контролю, але не забезпечує повної автоматизації процесу.

Найбільш ефективним є автоматизований метод контролю освітленості. У цьому випадку в навчальних аудиторіях встановлюються спеціальні сенсори, які постійно вимірюють рівень освітленості. На основі отриманих даних система приймає рішення про ввімкнення або вимкнення джерел світла, а також регулювання їх інтенсивності. Така система може працювати за заданими параметрами, наприклад, підтримувати освітленість на рівні 300–500 люксів, що відповідає державним стандартам.

Для реалізації автоматизованого контролю використовуються різні типи сенсорів: фоторезистори, фотодіоди, фототранзистори та цифрові світ-

лові сенсори. Вони підключаються до мікроконтролера або комп'ютерної системи, яка виконує обробку даних і управління світильниками. Крім того, такі системи можуть включати таймери, датчики присутності, елементи штучного інтелекту для адаптивного управління освітленням.

Впровадження автоматизованої системи контролю освітленості дозволяє досягти оптимальних умов для навчання, знизити енергоспоживання та підвищити рівень комфорту в аудиторіях.

Системи керування освітленням включають в себе апаратні та програмні компоненти, що забезпечують контроль за електроживленням джерел світла відповідно до заданих умов експлуатації. До складових таких систем належать пристрої введення, контролери та виконавчі елементи, які реалізують регулювання світлового потоку.

Однією з базових функцій є вмикання/вимикання освітлення, яке реалізується через класичні або електронні перемикачі. Перемикач встановлюється у ланцюзі між джерелом живлення та навантаженням. У замкнутому стані ланцюг є активним, що дозволяє подачу енергії на джерело світла, а у розімкнутому – подача припиняється (рис. 1.1).

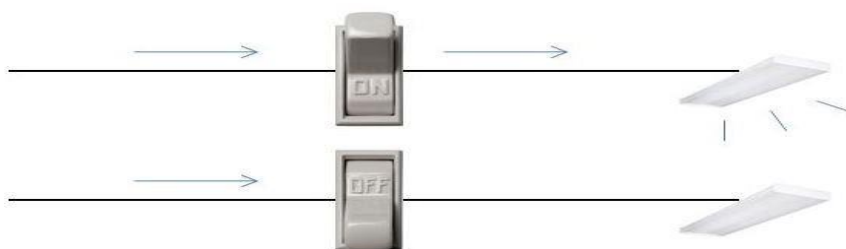


Рисунок 1.1 – Перемикання світла

Ще однією функціональною можливістю є регулювання яскравості (димування). Димери забезпечують зміну рівня напруги або широтно-імпульсної модуляції, що дозволяє змінювати інтенсивність світлового потоку. Це забезпечує більш гнучке використання енергії та комфортні умови освітлення (рис. 1.2).

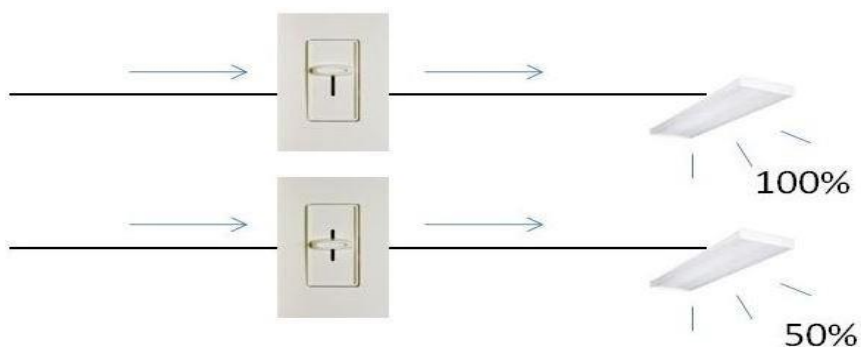


Рисунок 1.2 – Затемнення світла

У сучасних системах управління все частіше впроваджується контроль кольору освітлення, особливо при використанні світлодіодних джерел. Технології керування масивами світлодіодів з різними кольорними температурами дозволяють змінювати відтінки білого світла або реалізовувати повнокольорне освітлення на основі RGB або RGBW-модулів. Завдяки цьому користувач отримує можливість підлаштувати освітлення відповідно до індивідуальних потреб або завдань простору.

Зональне управління є важливою складовою концепції енергоефективного освітлення. Зона освітлення – це визначена група світильників, які одночасно керуються єдиним вихідним сигналом від контролера. Розподіл освітлювальних пристроїв на зони дозволяє оптимізувати споживання енергії, підвищити гнучкість у використанні простору та забезпечити відповідність нормативним вимогам.

У системах із однозонним управлінням уся група джерел світла функціонує одночасно – увімкнення, вимкнення чи регулювання яскравості відбувається єдиним сигналом. Такий підхід є простим у реалізації, але обмежує функціональність.

Багатозональне керування дозволяє незалежно контролювати кілька зон освітлення в межах одного приміщення. Це дає змогу створювати різні

сценарії освітлення, адаптовані до функціональних зон (наприклад, навчальна зона, зона відпочинку, зона демонстрації тощо).

Завдяки сучасним цифровим протоколам зв'язку, таким як DALI або KNX, можливим стало гнучке зонування навіть на рівні окремих світильників. Це забезпечує не лише точне налаштування освітлення, а й можливість динамічного перепланування без зміни фізичної електропроводки – шляхом зміни конфігурації програмного забезпечення.

Таким чином, багатозональне керування освітленням забезпечує адаптивність, енергоефективність та зручність експлуатації в сучасних навчальних закладах.

1.2 Огляд сучасних рішень контролю освітленості

Найбільш доступні за ціною моделі люксометрів, такі як LX-1010BS і Ezodo SP-216, належать до категорії приладів початкового рівня (рис. 1.3 та 1.4). Вони розраховані на виконання базових вимірювальних завдань, де не вимагається висока точність або відповідність професійним метрологічним стандартам. Такі прилади часто застосовуються в умовах, де допустимі відхилення не мають критичного значення – наприклад, у побуті, в аграрному секторі, для попередньої оцінки освітленості в закритих приміщеннях або на відкритому повітрі.

Попри свою простоту, зазначені люксометри оснащені основними функціями, які дозволяють оперативно оцінити рівень освітленості. Вони можуть мати базовий рідкокристалічний дисплей, фіксацію пікового значення, а також зручну форму корпусу, що полегшує експлуатацію в польових умовах. Проте відсутність світлофільтрів та обмежена спектральна чутливість знижують точність вимірювання, що не дозволяє використовувати ці прилади в лабораторних чи високотехнологічних сферах.



Рисунок 1.3 – Зовнішній вигляд LX-1010BS

Через відсутність світлофільтрів дані люксметри мають обмежену точність вимірювань, що зазвичай становить близько 10 Вт/м^2 . Такий рівень точності є прийнятним для загальних оцінок рівня освітленості у сферах, де не висуваються жорсткі вимоги до метрологічних характеристик. Зокрема, це стосується сільськогосподарських потреб (наприклад, у теплицях), метеорологічних спостережень, контролю освітленості у побутових, складських або виробничих приміщеннях, а також у відкритому середовищі, де умови постійно змінюються під впливом природного освітлення.

Подібні прилади орієнтовані на наочну індикацію та попередній моніторинг, що дозволяє користувачам оперативно оцінити умови без необхідності застосування складної та дорогої апаратури.



Рисунок 1.4 – Зовнішній вигляд Ezodo SP-216

До основних переваг люкметрів базового рівня можна віднести наявність великого рідкокристалічного дисплея, що забезпечує чітке відображення результатів вимірювання навіть при слабкому зовнішньому освітленні. Інтерфейс приладу відрізняється простотою та інтуїтивністю, що робить його зручним у користуванні як для фахівців, так і для користувачів без спеціальної технічної підготовки. Ще однією важливою конструктивною особливістю є гнучкий кабель, який з'єднує датчик освітленості з основним блоком приладу. Це дозволяє виконувати вимірювання у важкодоступних або віддалених точках без зміщення самого приладу.

Люкметр моделі ТМ 202 (рис. 1.5) вирізняється розширеним функціоналом. Завдяки вбудованому світлофільтру, прилад може ефективно відсікати випромінювання поза межами видимого спектра, що забезпечує вищу точність у стандартних умовах вимірювання. Наявність півтораметрового з'єднувального кабелю між датчиком і основним модулем підвищує гнучкість його застосування. Також модель підтримує низку корисних функцій: сигналізація перевищення допустимого діапазону вимірювання, попередження про низький заряд батареї, фіксація поточного значення, а також відображення мінімальних, максимальних і усереднених значень освітленості протягом певного періоду.



Рисунок 1.5 – Зовнішній вигляд ТМ 202

Даний люксметр призначений для вимірювання всіх видимих джерел світла в приміщеннях різного призначення.

Люксметр TM-213 UVAB (рис. 1.6 та 1.7) розроблений спеціально для вимірювання потужності ультрафіолетового випромінювання.



Рисунок 1.6 – Зовнішній вигляд TM-213 UVAB



Рисунок 1.7 – Зовнішній вигляд TM-201L

Ці моделі люксметрів оснащені спеціальними світлофільтрами та високочутливими кремнієвими фотодетекторами, спектральна характеристика яких максимально наближена до стандартної спектральної світлової ефективності, встановленої Комісією з освітлення (CIE). Завдяки цьому досягається точність, яка дозволяє використовувати прилади для вимірювань у професійних умовах, зокрема у лабораторіях, медичних закладах чи при стандартизованому освітленні робочих місць.

Крім того, у приладах реалізована система косинусної кутової корекції, яка компенсує похибки, що виникають при косому падінні світлового

поток на сенсор. Це особливо важливо при вимірюваннях у складних просторових умовах, коли джерело світла не розташоване строго перпендикулярно до площини приймача.

Для зручності користування сенсор під'єднується до вимірювального блоку через кабель довжиною 1,5 метра, що дозволяє проводити вимірювання на відстані, не змінюючи положення головного пристрою. Люксметри моделей ТМ-209, ТМ-209N та ТМ-209М (рис. 1.8 та 1.9) мають ідентичний зовнішній вигляд, проте можуть відрізнятися за функціональними можливостями або типом інтерфейсу для передачі даних.



Рисунок 1.8 – Зовнішній вигляд ТМ-209



Рисунок 1.9 – Зовнішній вигляд ТМ-208

Сучасні люксметри, що орієнтовані на професійне використання, вирізняються не лише високою точністю, а й широкими функціональними можливостями. Зокрема, прилади з вбудованим реєстратором даних здатні

накопичувати до 45 000 вимірювань, що особливо зручно для довготривалого моніторингу освітленості у динамічному середовищі. Наявність USB-інтерфейсу дозволяє з легкістю передавати дані на персональний комп'ютер для подальшого аналізу, обробки та створення звітів або графічної візуалізації результатів.

Оптична частина таких люкметрів базується на кремнієвих фотодіодах, спектральна чутливість яких відповідає характеристикам, рекомендованим CIE (Міжнародною комісією з освітлення). Додаткові оптичні фільтри та корекція косинусної похибки забезпечують високу точність незалежно від кута падіння світла. Це робить пристрої придатними для вимірювання у критичних сферах, таких як медицина, фізичні та оптичні лабораторії, а також для тестування фотоелектричних елементів у реальних умовах експлуатації.

Особливої уваги заслуговують професійні моделі від європейської компанії Delta OHM. Завдяки підтримці змінних сенсорів ці прилади можна налаштувати на вимірювання у широкому спектральному діапазоні – від інфрачервоного до ультрафіолетового світла. Така гнучкість робить їх універсальним інструментом для багатьох галузей.

Проведений аналіз існуючих методів контролю освітленості у навчальних аудиторіях показав, що традиційні ручні та напівавтоматичні підходи є обмеженими. Вони не забезпечують ані оперативної реакції на зміну освітлення, ані точного дотримання нормативів упродовж усього навчального дня. Більшість люкметрів на ринку не пристосовані до інтеграції з автоматизованими системами керування, що ще більше ускладнює завдання забезпечення сталого освітлення у класах та аудиторіях.

З огляду на це, виникає актуальна потреба в розробці сучасної автоматизованої системи контролю освітлення, яка повинна забезпечувати постійне вимірювання рівня освітленості, враховувати природне світло та добові коливання, взаємодіяти з датчиками присутності, мати можливість зонального керування і динамічного налаштування параметрів освітлення. Така система

сприятиме створенню комфортного освітлювального середовища, покращенню продуктивності навчального процесу та зниженню енерговитрат.

1.3 Висновки до розділу 1

У першому розділі було проаналізовано існуючі методи контролю освітленості в навчальних аудиторіях, а також технічні засоби, що використовуються для вимірювання рівня освітлення. Розглянуто основні типи систем керування освітленням — від ручних до повністю автоматизованих, їх переваги та недоліки.

Встановлено, що ручні й напівавтоматичні методи не забезпечують необхідного рівня точності, оперативності та енергоефективності, що особливо критично в умовах змінного природного освітлення та вимог до навчального середовища. Наявні автоматизовані рішення не завжди відповідають сучасним вимогам адаптивності, масштабованості та інтеграції з іншими системами управління.

Аналіз сучасних моделей люкметрів показав, що більшість з них призначені для локального контролю освітлення та не можуть бути інтегровані в системи розумного керування [3-10]. Попри наявність високоточних приладів, їх функціональність обмежена з точки зору автоматизованої взаємодії.

На основі проведеного аналізу зроблено висновок, що існує об'єктивна потреба у створенні нової, сучасної автоматизованої системи контролю освітленості в навчальних аудиторіях. Така система має включати: безперервний моніторинг освітленості; інтеграцію з іншими сенсорами (наприклад, датчиками присутності); багатозональне керування; можливість адаптації до умов освітлення та часу доби; зниження енергоспоживання за збереження високої якості освітлення.

Запропонована система сприятиме створенню комфортного, продуктивного та енергоефективного навчального середовища.

2 РОЗРОБКА СТРУКТУРНОЇ СХЕМИ ТА АЛГОРИТМА РОБОТИ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ

2.1 Структурна схема системи контролю освітленості в навчальних аудиторіях

Система автоматизованого контролю освітленості побудована за модульним принципом (рис 2.1) і складається з таких основних блоків: сенсори (фоторезистори), що вимірюють рівень освітленості; блок обробки сигналу, який нормалізує та фільтрує дані; мікроконтролер, що аналізує інформацію і формує керуючі сигнали згідно з заданими параметрами; блок введення для налаштувань користувача; блок живлення, який забезпечує стабільну роботу системи; та вихідний блок, що відображає або передає результат (наприклад, на дисплей або до системи керування світлом). Система працює в реальному часі й дозволяє адаптивно регулювати освітлення відповідно до умов середовища.

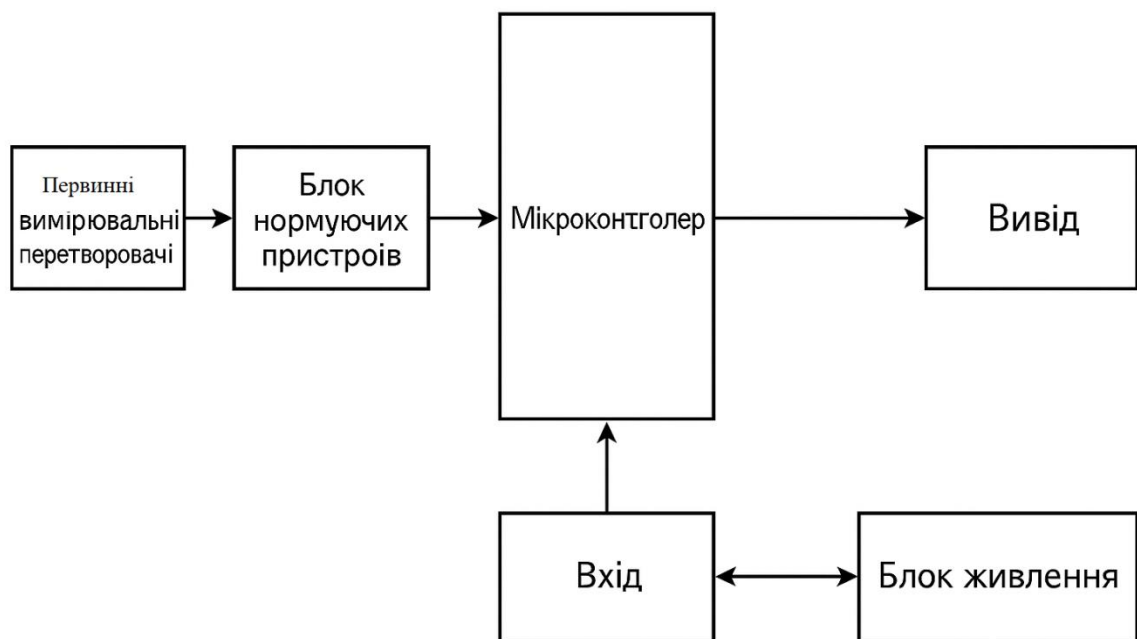


Рисунок 2.1 – Структурна схема системи

На зображеній схемі представлено структурну модель автоматизованої системи контролю освітленості, яка складається з ряду взаємопов'язаних функціональних блоків. Її робота базується на послідовній обробці інформації, починаючи з вимірювання освітленості й закінчуючи формуванням відповідного керуючого або інформаційного сигналу на виході.

Початковим елементом є блок первинних вимірювальних перетворювачів. У його складі використовуються фоторезистори або інші сенсори світла, які реагують на зміни освітленості середовища. Залежно від інтенсивності світлового потоку змінюється опір або вихідна напруга, що й фіксується для подальшої обробки.

Отриманий сигнал надходить до блоку керуючих пристроїв, який відповідає за попередню підготовку інформації — це може бути перетворення сигналу, його посилення або фільтрація шумів. Основне завдання цього блоку — забезпечити стабільний і коректний сигнал для надходження до мікроконтролера.

Мікроконтролер виконує центральну роль у системі. Він приймає дані, аналізує їх за заданим алгоритмом і формує команди на основі отриманої інформації. При цьому мікроконтролер може взаємодіяти з блоком введення, де користувач задає певні параметри роботи або змінює налаштування системи. Для стабільного функціонування вся система живиться від окремого блока живлення, який підтримує необхідні напруги та струми [12].

Оброблені дані або керуючі команди надсилаються на вихідний блок, який може бути у вигляді дисплея, інтерфейсу зв'язку з комп'ютером, або системи керування освітленням (наприклад, ШІМ-регулювання яскравості світлодіодів). Таким чином, система автоматично реагує на зміну умов освітлення та дозволяє ефективно регулювати освітлювальне середовище у приміщенні.

Схема принципова електрична (рис. 2.2). Розробка принципової електричної схеми, обумовлена можливістю уточнення алгоритмів функціонування ІВС та більш детального аналізу складових ІВС.

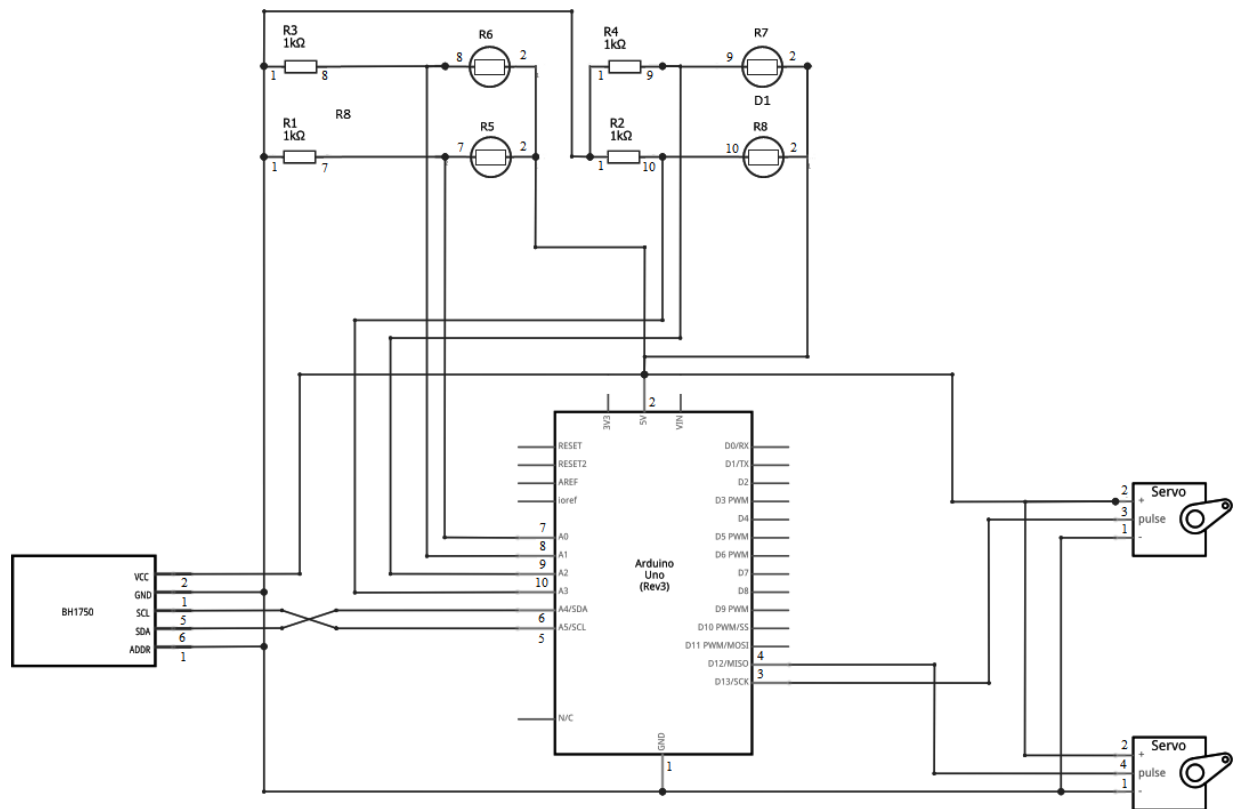


Рисунок 2.2 – Принципова електрична схема

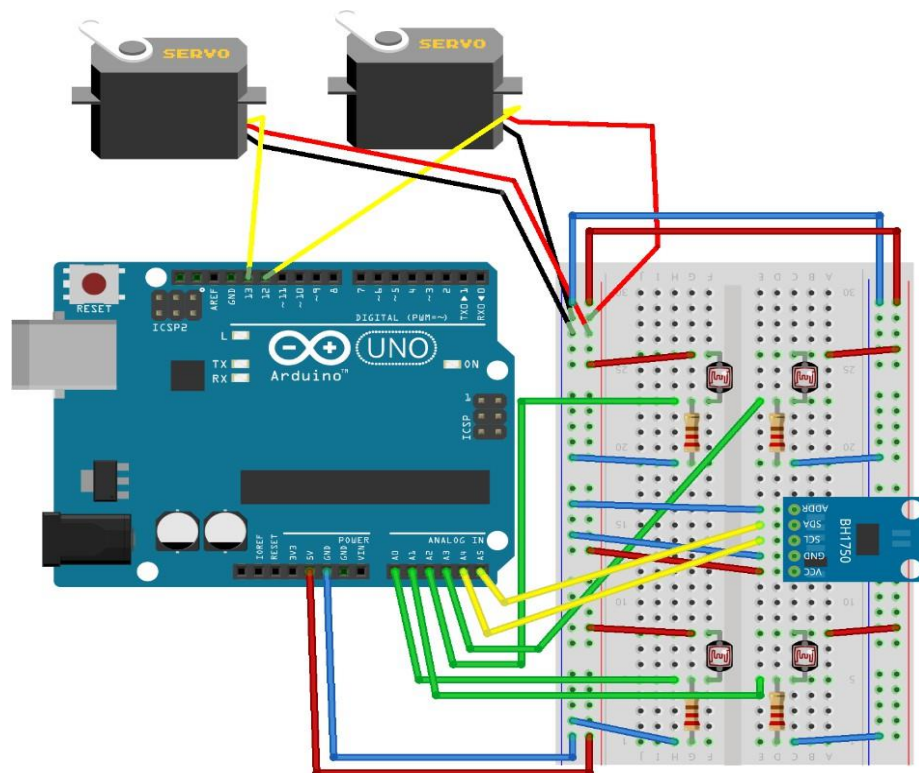


Рисунок 2.3 – Схемасистеми викона у програмі Fritzing

Схема електрична функціональна (рис 2.4)

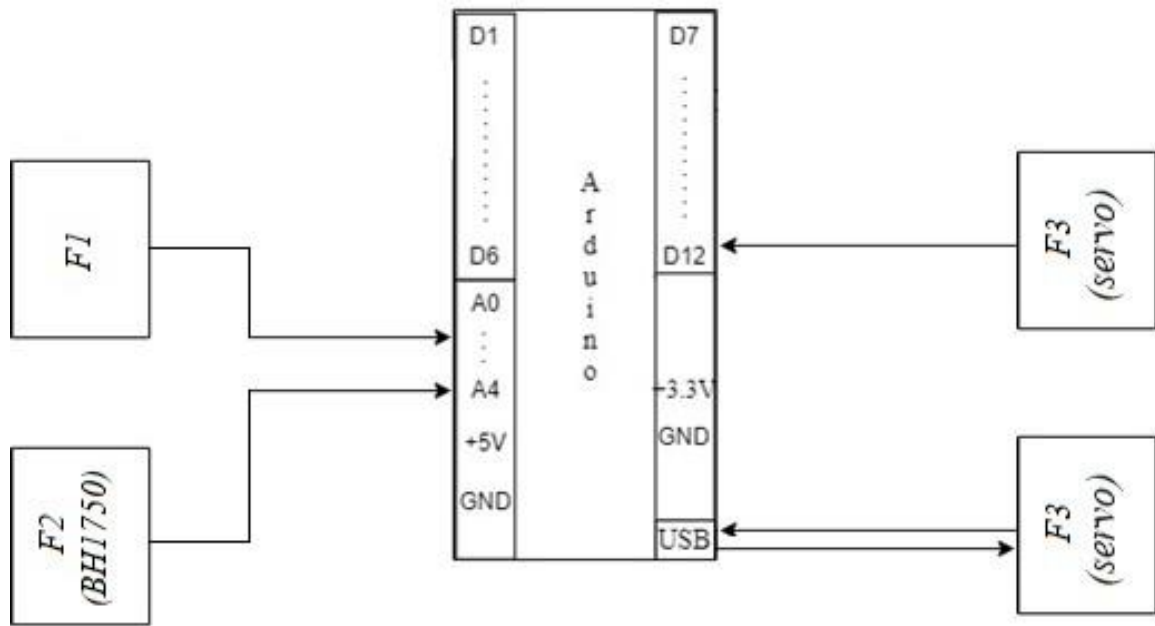


Рисунок 2.4 – Схема електрична функціональна системи

На рисунку 2.4 наведено наступні позначення: F1 – блок з чотирьох фоторезисторів; F2 – цифровий люксметр BH1750; F3 – Сервопривід TowerPro SG 5010; I/O – пристрій вводу/виводу.

На наведеній схемі показано роботу автоматизованої системи, яка регулює положення об'єкта залежно від рівня освітленості [13-15]. Вхідними елементами системи є блок F1, що складається з чотирьох фоторезисторів, і цифровий люксметр F2 (BH1750). Фоторезистори підключені до аналогових входів мікроконтролера Arduino, а BH1750 — до цифрових пінів через інтерфейс I²C. Дані про рівень освітлення надходять до Arduino, де обробляються згідно з алгоритмом визначення найсвітлішого напрямку. На основі отриманих результатів контролер формує керуючі сигнали для двох сервоприводів F3 (TowerPro SG 5010), які змінюють положення конструкції, орієнтуючи її на джерело світла. Вся система живиться через стандартний USB або зовнішнє джерело напруги. Таким чином, схема дозволяє в режимі реального часу адаптувати положення пристрою відповідно до змін рівня освітлення навколишнього середовища.

2.2 Алгоритм роботи системи контролю освітленості в навчальних аудиторіях

Алгоритм роботи системи контролю освітленості в навчальних аудиторіях представлено на рис. 2.5.

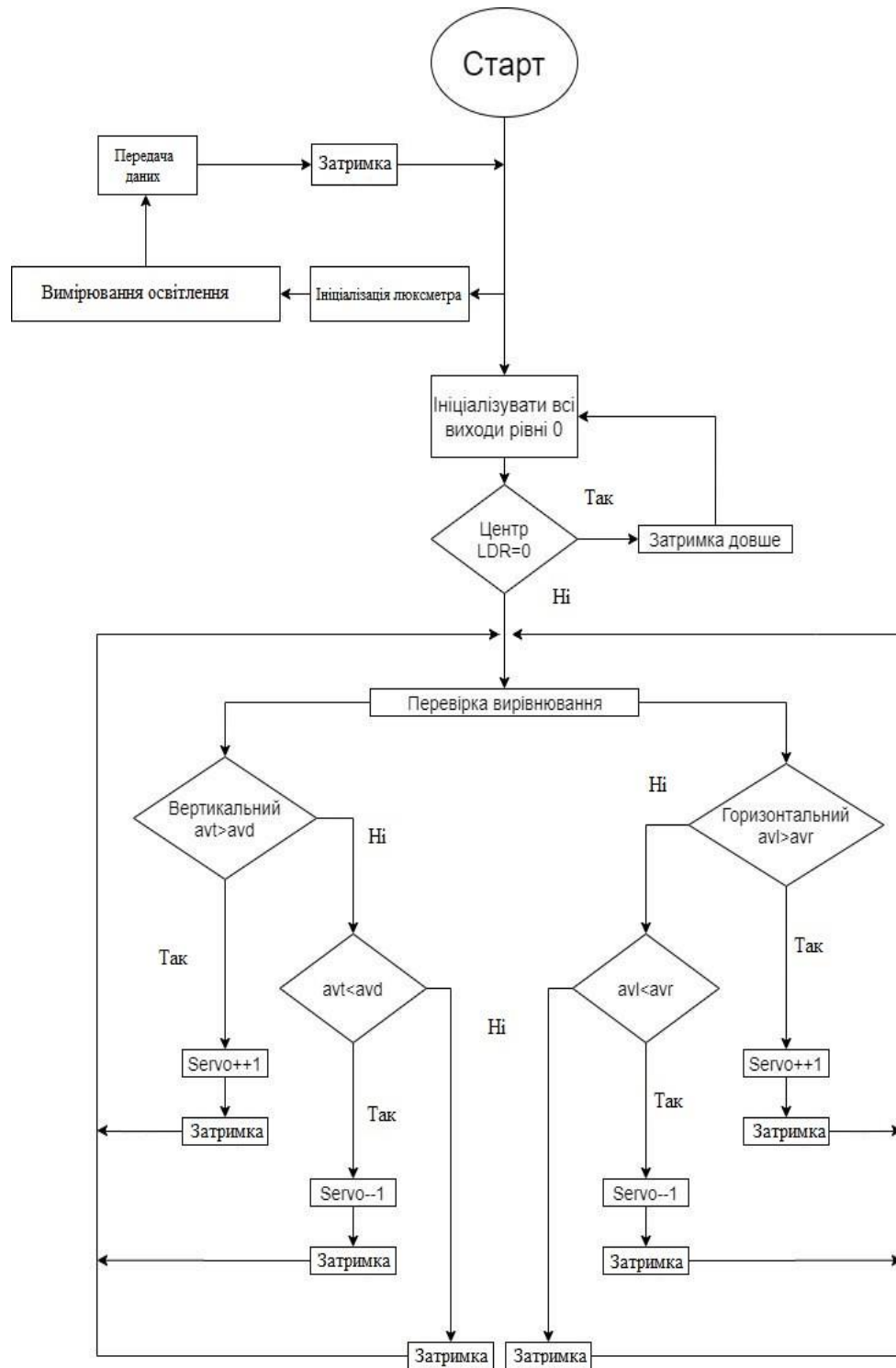


Рисунок 2.5 – Алгоритм роботи системи

На початковому етапі система запускається (блок "Старт"). Одразу після цього виконується передача даних і коротка затримка, що дозволяє стабілізувати роботу компонентів. Далі відбувається ініціалізація люксметра, який відповідатиме за точні вимірювання рівня освітлення.

Після цього запускається процедура вимірювання освітленості через сенсори LDR, і мікроконтролер встановлює усі виходи в нульовий рівень, тобто обнуляє попередні значення керуючих сигналів на сервоприводи.

Далі перевіряється значення центрального LDR (який визначає, чи світло потрапляє рівномірно). Якщо його значення дорівнює нулю (немає світла), система чекає — запускається довша затримка, після чого перевірка повторюється. Якщо значення LDR відмінне від нуля, система переходить до блоку перевірки вирівнювання.

На цьому етапі починається основна логіка роботи вирівнювання:

Визначається різниця між верхнім (avt) і нижнім (avd) сенсорами, що відповідають за вертикальне положення.

Якщо значення верхнього сенсора більше за нижній, це означає, що більше світла потрапляє зверху, і потрібно змінити кут у напрямку вгору. Сервопривод збільшує кут (Servo++1), після чого виконується затримка.

Якщо навпаки — нижній сенсор дає більше світла, сервопривод зменшує кут (Servo--1), і знову затримка.

Аналогічна логіка застосовується для горизонтального вирівнювання:

Якщо значення зліва (avl) більше, ніж справа (avr), сервопривод повертає панель праворуч (Servo++1).

Якщо справа інтенсивність вища — панель повертається вліво (Servo--1). Усі операції супроводжуються паузою (затримкою), щоб система стабільно реагувала на зміни і уникала коливань.

Система в реальному часі аналізує освітлення з чотирьох напрямків і автоматично регулює положення за допомогою двох сервоприводів — вертикального та горизонтального. Такий підхід дозволяє максимально точно

орієнтуватися на джерело світла й ефективно використовувати світлову енергію.

2.3 Висновки до розділу 2

У другому розділі бакалаврської кваліфікаційної роботи було розроблено структурну схему автоматизованої системи контролю освітленості та виконано її опис. Згідно зі схемою, система складається з основних функціональних блоків: блоку первинних перетворювачів (фоторезистори), вимірювального пристрою (люксметр ВН1750), блоку виконавчих пристроїв (сервоприводи) та мікроконтролера, який виконує функції обробки даних та керування.

Розглянуто принцип роботи системи, де фоторезистори визначають напрямок на максимальне освітлення, впливаючи на положення сервоприводів, а люксметр вимірює фактичний рівень освітленості для подальшого контролю. Мікроконтролер аналізує отримані значення та формує команди на зміну положення елементів або передачу даних через UART-USB.

Також розроблено принципову електричну схему, що забезпечує чітке уявлення про логіку взаємодії елементів системи, та алгоритм роботи інформаційно-вимірювальної системи, що демонструє послідовність прийняття рішень на основі отриманих даних.

Отже, розроблена схема є функціональною основою для створення повноцінної автоматизованої системи контролю освітленості, яка здатна реагувати на зміни освітлення, оптимізувати енергоспоживання та забезпечити комфортне освітлення в навчальних приміщеннях.

3 РОЗРОБКА ФУНКЦІОНАЛЬНО-АПАРATНОЇ СХЕМИ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ

3.1 Апаратний склад системи

Мікроконтролер Arduino UNO. У представлений інформаційно-вимірювальній системі в якості головного елемента керування було обрано популярний мікроконтролер Arduino UNO. Вибір саме цієї апаратної платформи зумовлений її простотою в експлуатації, широкою функціональністю та гнучкістю налаштування, що особливо важливо для розробки прототипів та навчальних проектів. Основним завданням цього мікроконтролера у складі системи є керування положенням сервоприводів залежно від даних, отриманих від сенсорів освітленості.

Arduino UNO (рис. 3.1) побудований на базі мікроконтролера ATmega328P, який поєднує у собі необхідний набір апаратних ресурсів для реалізації складних алгоритмів керування. Плата має 14 цифрових входів/виходів, шість з яких підтримують генерацію широтно-імпульсної модуляції (PWM), що дозволяє здійснювати плавне керування, наприклад, яскравістю світлодіодів чи положенням сервомеханізмів. Крім того, передбачено 6 аналогових входів, які можуть бути використані для підключення аналогових сенсорів, таких як LDR, фоторезистори або потенціометри.

Вбудований кварцовий резонатор з частотою 16 МГц забезпечує точну синхронізацію операцій мікроконтролера. Для зручності підключення пристрою до комп'ютера або джерела живлення передбачено роз'єм USB, гніздо живлення (DC barrel jack), а також інтерфейс ICSP для прямого програмування чипа.

На відміну від попередніх версій плат, Arduino UNO використовує мікроконтролер ATmega16U2 (або ATmega8U2 у версії R2) для реалізації перетворення інтерфейсів USB-UART, що дозволяє підключатися до ПК без використання зовнішнього програматора типу FTDI. Така конструкція забез-

печує високу надійність зв'язку та спрощує процедуру завантаження прошивки. До того ж, на версії плати R2 був впроваджений резистор підтяжки до землі для лінії HWB, який полегшує перехід до режиму оновлення прошивки, що важливо для розробників.

Завдяки простій архітектурі та великій спільноті користувачів Arduino UNO є одним з найпоширеніших інструментів у навчальних, дослідницьких та аматорських проєктах.

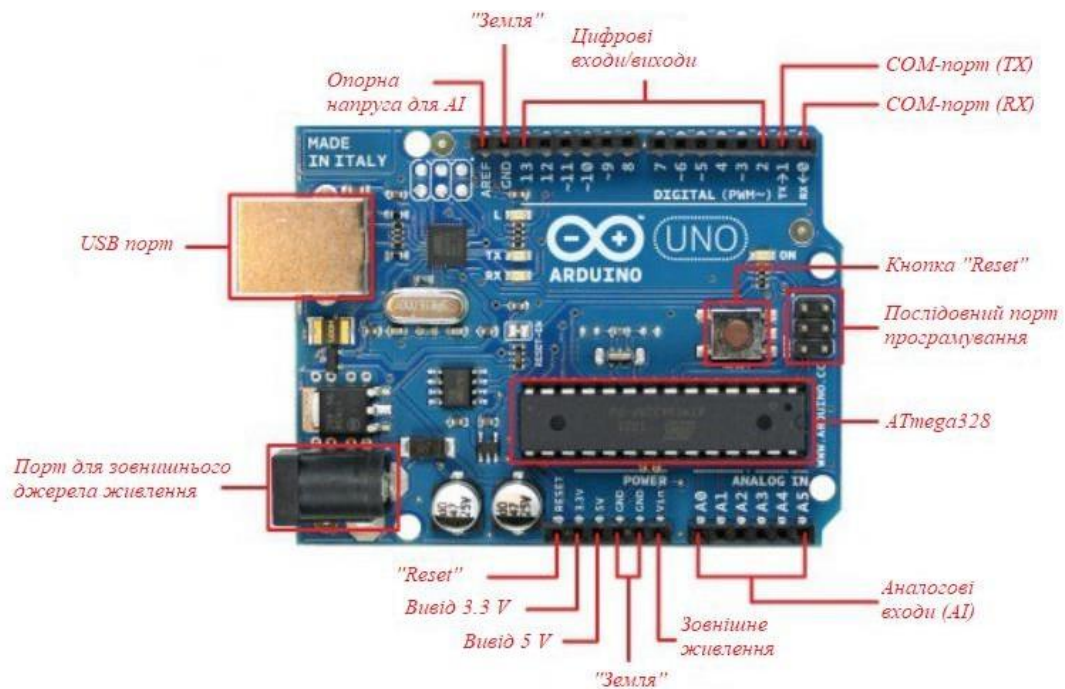


Рисунок 3.1 – Зовнішній вигляд Arduino UNO

Мікроконтролерна плата Arduino UNO підтримує гнучке живлення від кількох джерел, що дозволяє зручно використовувати її як у стаціонарних, так і в автономних умовах. Вибір між живленням через USB або зовнішнє джерело відбувається автоматично завдяки вбудованій схемі перемикачів.

Як зовнішнє джерело живлення (не через USB) може бути використаний мережевий адаптер змінного/постійного струму (AC/DC адаптер) або акумуляторна батарея. Для підключення адаптера передбачено стандартний роз'єм діаметром 2.1 мм з позитивним полюсом у центрі, який вставляється у відповідне гніздо на платі Arduino. Якщо використовується батарея або аку-

мулятор, її дроти під'єднуються до контактів GND (маса) та VIN на роз'ємі живлення POWER.

Arduino UNO працює в межах напруги від 6 до 20 вольт, однак рекомендоване значення для стабільної роботи — від 7 до 12 В. Напруга нижча за 7 В може викликати зниження вихідного рівня напруги до менш ніж 5 В, що потенційно призведе до нестабільної роботи електронних компонентів. Напруга, що перевищує 12 В, створює ризик перегріву стабілізатора на платі, що в свою чергу може спричинити вихід пристрою з ладу.

На платі Arduino UNO реалізовано кілька виходів живлення:

- VIN — вхідна напруга з зовнішнього джерела живлення, яка надходить безпосередньо до плати до стабілізації. Через цей пін можна як подавати живлення, так і брати його, якщо джерело під'єднане через інший роз'єм.

- 5V — стабілізована напруга 5 вольт, яку плата може видавати при живленні від USB або від джерела 7–12 В через VIN. Не рекомендується подавати живлення безпосередньо на цей пін, оскільки при цьому минається внутрішній стабілізатор, що створює ризик пошкодження.

- 3V3 — стабілізований вихід на 3.3 Вольти, з обмеженням струму до 50 мА, що підходить для живлення низьковольтної периферії.

- GND — спільний контакт заземлення для всіх джерел живлення.

- IOREF — спеціальний пін, який передає розширювальним модулям інформацію про логічний рівень напруги (5 В або 3.3 В), що дозволяє їм автоматично адаптуватися під поточний режим роботи Arduino.

У складі цієї системи (рис. 3.2) як виконавчі елементи були використані сервомеханізми TowerPro SG 5010, які забезпечують високоточне позиціонування. Один із сервоприводів відповідає за горизонтальне обертання, інший — за вертикальне. Це дозволяє з великою точністю орієнтувати, наприклад, світлочутливі сенсори або сонячні панелі на джерело світла відповідно до змін зовнішнього освітлення.

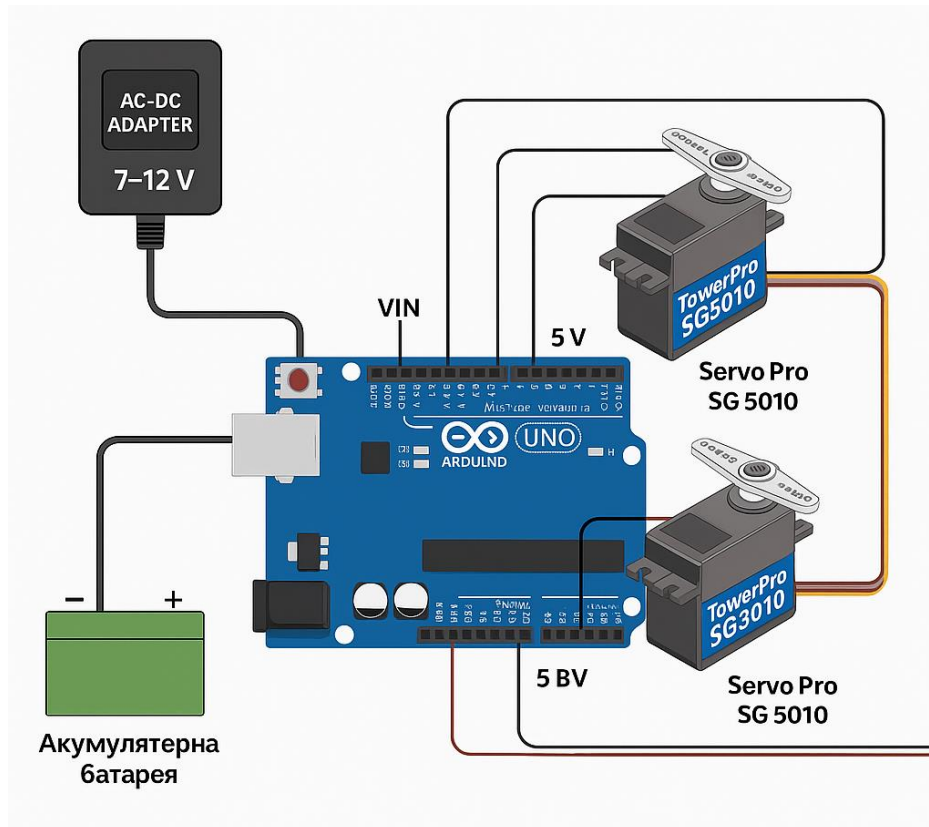


Рисунок 3.2 – Функціонально схема системи контролю освітленості в навчальних аудиторіях

Пам'ять Arduino UNO. Arduino UNO побудований на мікроконтролері ATmega328, який має у своєму розпорядженні 32 кілобайти флеш-пам'яті, з яких 0.5 кілобайта вже зайняті завантажувачем (bootloader), що необхідний для прошивки пристрою безпосередньо через інтерфейс USB. Оперативна пам'ять SRAM становить 2 кілобайти, а постійна пам'ять EEPROM – 1 кілобайт. EEPROM може використовуватись для збереження даних, які не повинні стиратись при вимкненні живлення, причому доступ до неї здійснюється за допомогою спеціальної бібліотеки EEPROM.

Входи та виходи плати. На платі передбачено 14 цифрових пінів, які можуть конфігуруватись як входи або виходи з використанням функцій pinMode(), digitalWrite() та digitalRead() у середовищі програмування Arduino IDE. Кожен з цифрових контактів може обробляти сигнали до 5 вольт, а максимально допустимий струм для одного піну – до 40 мА. Для додаткової

стабільності на всіх входах передбачені внутрішні підтягуючі резистори з опором у діапазоні від 20 до 50 кОм, які за замовчуванням вимкнені.

Окрім базових функцій, частина пінів виконує додаткові спеціалізовані задачі:

- Послідовний зв'язок (Serial): виводи 0 (RX) і 1 (TX) реалізують стандартний UART інтерфейс для обміну даними з ПК або іншими пристроями. Ці пін з'єднані з мікросхемою ATmega16U2, яка виконує роль USB-UART перетворювача.

- Зовнішні переривання: виводи 2 і 3 можуть генерувати переривання за змінами сигналу (зростання, спад або низький рівень), що важливо для швидкого реагування на зовнішні події.

- ШІМ (PWM): пін 3, 5, 6, 9, 10 і 11 підтримують генерацію широтно-імпульсної модуляції (PWM) через функцію `analogWrite()`, що дозволяє керувати, наприклад, яскравістю світлодіодів або швидкістю обертання двигунів.

- SPI інтерфейс: порти 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK) дозволяють здійснювати високошвидкісну синхронну передачу даних із зовнішніми пристроями за допомогою SPI-протоколу, який підтримується через бібліотеку SPI.

- Світлодіод на пині 13: стандартно підключений до пину 13 вбудований світлодіод дозволяє швидко перевіряти базову працездатність плати.

Крім цифрових, UNO має 6 аналогових входів A0–A5, які здатні зчитувати аналогову напругу в діапазоні від 0 до 5 В і перетворювати її в цифрове значення за допомогою вбудованого 10-бітного АЦП (тобто значення в межах від 0 до 1023). Верхню межу напруги вимірювання можна змінити через вивід AREF, встановивши опорну напругу вручну. Аналогові входи A4 та A5 можуть також використовуватись для передачі даних за протоколом I2C (або TWI), що спрощується бібліотекою Wire.

AREF: дозволяє підключити зовнішнє джерело опорної напруги для точнішого аналого-цифрового перетворення.

Reset: зниження напруги на піні Reset ініціює перезавантаження мікроконтролера, часто використовується в платах розширення (shields).

Комунікаційні можливості Arduino UNO. Плата Arduino UNO володіє широким набором інтерфейсів для організації обміну даними як із персональним комп'ютером, так і з іншими мікроконтролерами або сумісними пристроями. Основою для реалізації послідовного обміну інформацією служить апаратний UART-приймач, інтегрований в мікроконтролер ATmega328. Передача та прийом даних здійснюється через цифрові контакти 0 (RX) і 1 (TX), що дозволяє використовувати UART як для локального, так і для віддаленого зв'язку.

З метою забезпечення зручного підключення до ПК, Arduino UNO обладнаний мікроконтролером ATmega16U2, який виконує функції перетворювача USB-UART. Завдяки цьому пристрій легко визначається операційною системою як віртуальний COM-порт, що не потребує встановлення сторонніх драйверів — достатньо стандартного .inf-файлу для Windows. При активному обміні даними через USB, відповідні індикатори RX і TX на платі блимають, сигналізуючи про передавання або прийом інформації.

Якщо використовується передача через стандартні UART-контакти без залучення USB, індикація RX/TX залишається активною, що зручно для моніторингу процесу. Для організації альтернативних каналів зв'язку передбачена бібліотека SoftwareSerial, яка дозволяє реалізовувати послідовний обмін через будь-які інші цифрові контакти.

Крім UART, Arduino UNO підтримує популярні послідовні протоколи I²C (TWI) та SPI, які дозволяють підключати датчики, дисплеї, карти пам'яті та інші модулі. Для роботи з I²C використовується бібліотека Wire, яка суттєво спрощує процес програмування інтерфейсу. В свою чергу, SPI реалізується через бібліотеку SPI, що дозволяє здійснювати швидку синхронну передачу даних між пристроями.

Програмування контролера Arduino UNO. Процес програмування плати Arduino UNO здійснюється через середовище Arduino IDE — інтуїтивно зрозумілий інструмент, спеціально розроблений для роботи з платформами Arduino. Щоб почати програмування, необхідно в меню Tools → Board обрати тип плати “Arduino UNO”, що відповідає мікроконтролеру ATmega328.

Цей мікроконтролер за замовчуванням оснащений завантажувачем (bootloader), який дозволяє перепрошивати його без потреби у зовнішньому програматорі. Передача програмного коду відбувається через USB-інтерфейс за протоколом STK500. Однак у разі необхідності глибшого втручання, наприклад, для відновлення або оновлення прошивки, можна скористатись роз’ємом ICSP для внутрішньосхемного програмування.

Що стосується мікроконтролера ATmega16U2, який використовується як перетворювач інтерфейсів USB-UART, він має можливість оновлення власної прошивки. Прошивка доступна у відкритому доступі і включає так званий DFU-завантажувач (Device Firmware Update), що дозволяє виконати оновлення мікроконтролера без використання сторонніх пристроїв.

Для активації режиму DFU на платах першої ревізії (R1) необхідно закоротити спеціальну перемичку на звороті плати біля контуру карти Італії та натиснути кнопку скидання. У пізніших версіях (R2 і новіших) додано резистор, що автоматично підтягує лінію HWB до землі, спрощуючи вхід у DFU. Для завантаження нової прошивки використовується програмне забезпечення FLIP від Atmel (для Windows) або DFU Programmer (для macOS та Linux). Альтернативним способом є прошивка через ISP-програматор, однак у цьому випадку DFU-завантажувач буде видалено.

Автоматичне скидання мікроконтролера Arduino UNO. Конструкція плати Arduino UNO передбачає можливість автоматичного перезавантаження мікроконтролера без фізичного натискання на кнопку скидання. Це реалізовано через взаємодію між мікроконтролером ATmega8U2 або ATmega16U2 (який відповідає за обробку USB-з’єднання) та основним чипом ATmega328. Один із виводів, що відповідає за керування потоком (DTR), підключено до

лінії RESET через конденсатор ємністю 100 нФ. При появі логічного нуля на DTR, лінія RESET короткочасно переходить у низький рівень, що призводить до перезавантаження ATmega328.

Це дозволяє виконувати завантаження нового скетчу на плату без додаткових дій з боку користувача — достатньо лише натиснути відповідну кнопку в середовищі Arduino IDE. Така реалізація спрощує процес прошивки та дозволяє оптимізувати завантажувач, скорочуючи час очікування при старті завантаження.

Проте існує особливість, яку слід враховувати при роботі з операційними системами macOS або Linux. При кожному підключенні програмного забезпечення до плати, відбувається автоматичне скидання мікроконтролера. Після цього запускається завантажувач, який активний близько 500 мс. У цей проміжок він може перехопити початкові байти інформації, якщо вони надсилаються відразу після встановлення з'єднання. Тому, якщо ваша програма повинна отримувати дані при старті, бажано реалізувати паузу щонайменше в одну секунду перед передачею даних.

Для випадків, коли автоматичне скидання небажане, на платі UNO передбачена спеціальна доріжка з маркуванням "RESETEN". Її можна розрізати, щоб вимкнути функцію автоматичного перезавантаження. Для повернення до початкової конфігурації достатньо спаяти кінці доріжки між собою. Альтернативно, можна підключити резистор на 110 Ом між лінією RESET і 5V, що також блокує автоматичне скидання.

USB-захист Arduino UNO від перенавантаження. Arduino UNO обладнаний вбудованими самовідновлюваними запобіжниками, які забезпечують додатковий захист для USB-порту комп'ютера. Хоча більшість сучасних ПК вже мають вбудовані засоби захисту, присутність такого запобіжника на платі Arduino підвищує загальну надійність системи.

Якщо підключений пристрій перевищує допустиме значення струму (500 мА), захисний елемент автоматично розмикає ланцюг, перешкоджаючи можливому пошкодженню комп'ютера або самої плати. Після усунення при-

чини короткого замикання чи перевантаження з'єднання відновлюється без потреби заміни запобіжника.

Сервопривід TowerPro SG 5010 є компактним електромеханічним пристроєм, призначеним для точного позиціонування вала у заданому кутовому діапазоні. Його конструкція дозволяє здійснювати поворот у межах приблизно від 0° до $170\text{--}180^\circ$, як за годинниковою стрілкою, так і у зворотному напрямку. Такий кут охоплення забезпечує можливість реалізації широкого спектра завдань — від керування положенням елементів освітлення до навігації сенсорів.

В основі приводу використано аналоговий контролер на мікросхемі AA51880, що відповідає за обробку вхідних ШІМ-сигналів (широтно-імпульсної модуляції) та визначення положення вала за допомогою вбудованого потенціометра типу B5K. Цей потенціометр передає актуальну позицію у вигляді аналогового сигналу, який порівнюється із сигналом керування, внаслідок чого мотор автоматично підлаштовується до заданого кута.

Живлення приводу забезпечується через стандартні три дроти: коричневий — заземлення (GND), червоний — позитивне живлення (+V), та оранжевий або жовтий — сигнальний провід для подачі ШІМ-команди. Така конфігурація робить сервопривід сумісним із популярними контролерами, зокрема Arduino, та дозволяє легко інтегрувати його до різноманітних систем автоматики.

3.2 технічні характеристики сенсора освітленості BH1750

Для визначення рівня навколишнього освітлення в системі використовується модуль GY-302, який побудований на основі цифрового світлочутливого сенсора BH1750. Цей сенсор забезпечує високоточне вимірювання освітленості у люксах завдяки вбудованому 16-бітному АЦП та цифровому інтерфейсу I2C, що спрощує підключення до мікроконтролера.

Однією з головних переваг ВН1750 є його спектральна характеристика, наближена до сприйняття світла людським оком. Це означає, що сенсор найбільш точно реагує на видиме світло, майже повністю ігноруючи інфрачервоне випромінювання. Завдяки цьому спотворення даних через сторонні джерела ІЧ-випромінювання (наприклад, сонце чи лампи з ІЧ-компонентом) мінімізуються.

ВН1750 підтримує кілька режимів вимірювання: від високої роздільної здатності з повільним опрацюванням до швидких вимірювань з нижчою точністю. Освітленість можна вимірювати в діапазоні від 1 до 65535 люксів, що дозволяє використовувати сенсор як у темних приміщеннях, так і на відкритому сонячному світлі.

До ключових технічних переваг ВН1750 слід віднести:

- інтерфейс зв'язку I2C, який спрощує інтеграцію у мікроконтролерні системи;
- мінімальне споживання енергії з можливістю відключення живлення;
- нечутливість до частот мережевого освітлення (пригнічення шуму 50/60 Гц);
- робота без потреби у зовнішніх пасивних компонентах;
- два варіанти адреси I2C для підключення декількох сенсорів;
- здатність працювати з різними джерелами світла (лампи розжарювання, світлодіоди, денне світло тощо);
- підтримка компенсації втрат світлового потоку через оптичне вікно.

ВН1750 – це надійний інструмент (рис. 3.3) для точного моніторингу рівня освітленості в автоматизованих системах, що забезпечує адаптивну реакцію на зміну умов довкілля.

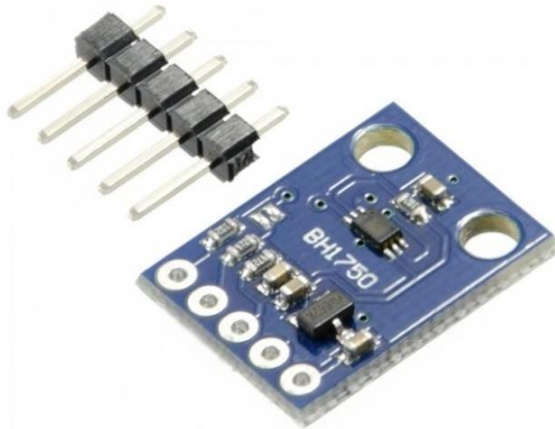


Рисунок 3.3 – Зовнішній вигляд датчика BH1750

Сенсор BH1750 підтримує дві різні I2C-адреси, що визначаються рівнем логічного сигналу на вході ADDR. Якщо цей вхід підключено до низького рівня (LOW), використовується адреса 0x23, а у випадку високого рівня (HIGH) — адреса 0x5C. Це відкриває можливість одночасного підключення двох однакових датчиків BH1750 до однієї I2C-шини без виникнення конфліктів адрес [15-17].

На модулі GY-302, побудованому на основі BH1750, вивід ADDR (іноді позначений як ADR) може бути залишений вільним, приєднаний до землі або до джерела живлення (рис. 3.4). Важливо дотримуватись вимоги щодо максимально допустимої напруги — не більше 3.3 В. Зазвичай на модулі вже встановлено підтягувальний (pull-down) резистор номіналом 1 кОм, що забезпечує фіксацію низького рівня у випадку, якщо контакт залишено непідключеним. У такому разі за замовчуванням використовується адреса 0x23.

Це рішення є простим способом організації кількох точок вимірювання освітленості в одній системі, що особливо корисно в багатозонних освітлювальних системах або системах з автоматичним регулюванням світла у навчальних приміщеннях.

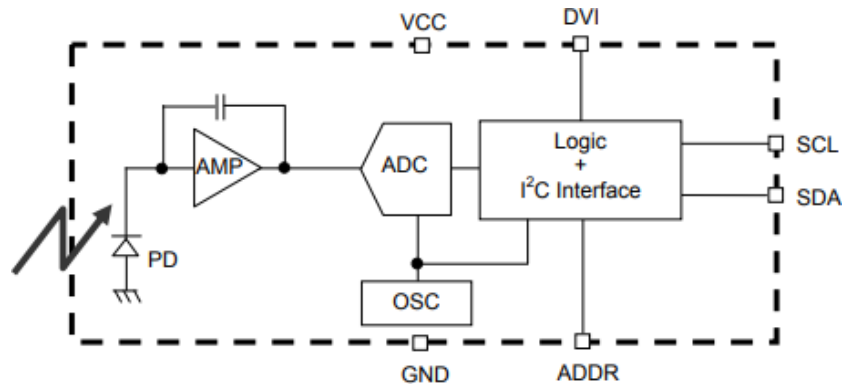


Рисунок 3.4 – Структурна схема BH1750

Опис внутрішньої структури та функціональних блоків модуля на основі BH1750 дозволяє краще зрозуміти принцип його роботи.

Основу датчика складає фотодіод (PD), який характеризується спектральною чутливістю, близькою до спектра сприйняття людського ока. Завдяки цьому BH1750 може точно відображати рівень освітленості в умовах, подібних до сприйняття людиною.

Струм, що виникає в фотодіоді під впливом світла, передається на інтегруючий підсилювач (AMP), реалізований на операційному підсилювачі (OPAMP). Цей підсилювач виконує перетворення фотоструму в аналогову напругу, пропорційну рівню освітленості.

Далі напруга з AMP подається на внутрішній аналогово-цифровий перетворювач (ADC), який виконує її оцифрування з розрядністю 16 біт. Завдяки такій точності забезпечується широкий динамічний діапазон – від дуже слабкого до інтенсивного світла.

Отримані цифрові дані надходять до блоку логіки, який включає в себе логічні елементи для обробки сигналів та інтерфейс шини I2C. Завдяки цій інтеграції модуль може легко взаємодіяти з мікроконтролерами (наприклад, Arduino) через стандартний двопровідний інтерфейс.

Дані про навколишнє освітлення зберігаються у спеціальному регістрі даних. Початкове значення цього регістра дорівнює 0000_0000_0000_0000, що відповідає нульовому рівню освітлення.

Регістр часу вимірювання дозволяє задавати період, протягом якого сенсор здійснює накопичення даних. Початкове значення регістра – 0100_0101, що відповідає певному стандартному значенню тривалості вимірювання, оптимальному для більшості застосувань.

Вся система синхронізується за допомогою внутрішнього генератора (OSC), який формує частоту приблизно 320 кГц і забезпечує тактування внутрішньої логіки пристрою.

Алгоритм роботи BH1750 представлено на рис. 3.5.



Рисунок 3.5 – Алгоритм роботи BH1750

Режими вимірювання BH1750. Ця блок-схема ілюструє логіку роботи живлення та режимів вимірювання датчика освітлення BH1750 у взаємодії з інтерфейсом I2C [18].

Початковий стан системи визначається як "Режим відключення живлення", що активується автоматично після подачі живлення на лінії VCC та DVI. У цьому режимі датчик споживає мінімум енергії та не здійснює жодних вимірювань.

Зі стану "Вимкнення живлення" система може перейти до "Увімкнення живлення" за допомогою відповідної команди запису, переданої через інтерфейс I2C. Цей перехід дозволяє активувати пристрій та підготувати його до вимірювання освітленості (рис. 3.6).

Після включення живлення виконується перевірка на наявність вимірювальної команди. Якщо така команда надійшла, система переходить до одного з двох режимів вимірювання — "Одноразове вимірювання" або "Безперервне вимірювання", в залежності від типу отриманої команди.

У режимі одноразового вимірювання датчик проводить одиничне вимірювання освітленості, після чого повертається до попереднього стану очікування нових команд. Цей режим економить енергію і зручний для періодичних зчитувань.

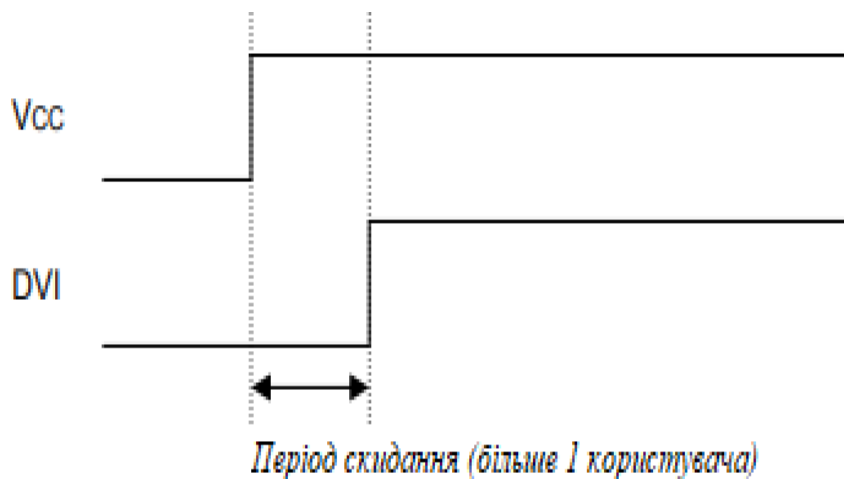


Рисунок 3.6 – Графік часу для подачі VCC та DVI

Графік часу для подачі VCC та DVI. (Якщо DVI піднімається протягом 1 мкс після подачі VCC) (рис. 3.7)

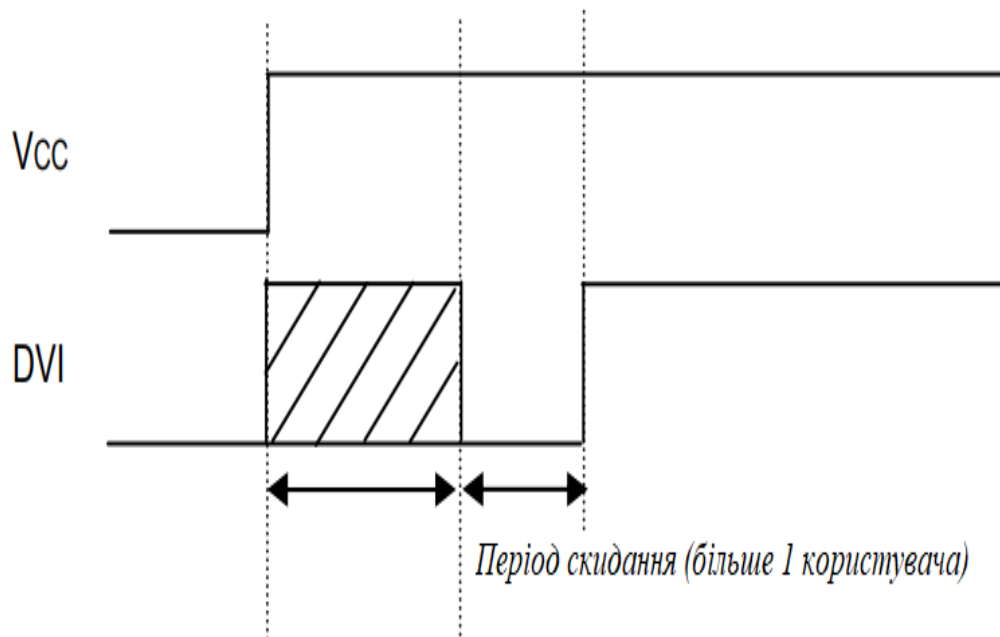


Рисунок 3.7 – Графік часу для подачі VCC та DVI

Режим безперервного вимірювання активується тоді, коли потрібно здійснювати постійний моніторинг рівня освітленості. У цьому стані пристрій безперервно виконує вимірювання і передає дані на мікроконтролер.

Якщо живлення вимикається (наприклад, за командою або внаслідок вимкнення зовнішнього джерела), система автоматично повертається до режиму "Вимкнення живлення". Це також зазначено на схемі пунктирною лінією, яка позначає автоматичні переходи між станами.

Після подачі живлення на пристрій виконується ініціалізація основних компонентів, а також перевірка наявності вхідної команди через інтерфейс зв'язку (наприклад, I2C або UART). Якщо команда відсутня, пристрій залишається в режимі очікування з мінімальним енергоспоживанням. Якщо ж команда виявлена, система аналізує її тип і переходить до одного з двох основних режимів роботи: одноразове або безперервне вимірювання.

У режимі одноразового вимірювання датчик активується на короткий проміжок часу, здійснюючи одне зчитування рівня освітленості. Після завершення вимірювання значення передається до мікроконтролера, а сенсор автоматично повертається до режиму очікування або "сну". Цей підхід сут-

тево знижує загальне енергоспоживання системи, що є критично важливим для автономних або портативних пристроїв, які працюють від батарей.

Графік часу для подачі напруги живлення VCC та цифрового сигналу ініціалізації DVI (Digital Valid Input) визначає послідовність ввімкнення системи. Згідно з технічними характеристиками, якщо сигнал DVI активується протягом 1 мікросекунди після подачі VCC, пристрій розпізнає це як команду до негайного переходу в активний стан. Цей часовий інтервал критично важливий для забезпечення коректного запуску мікросхеми і уникнення некоректної роботи сенсора в разі нестабільного живлення.

У режимі безперервного вимірювання пристрій виконує послідовні вимірювання освітленості з фіксованим часовим інтервалом (наприклад, кожні 120 мс або інші значення, задані конфігурацією). Ці дані постійно передаються на мікроконтролер або зберігаються у внутрішній пам'яті. Цей режим підходить для систем, де необхідний постійний моніторинг, наприклад, у розумному освітленні, системах безпеки або екологічних моніторингових станціях.

У разі відключення живлення пристрій автоматично переходить у режим "Power Down" — повне відключення споживання енергії та виведення всіх ліній у високий імпеданс. Такий перехід може бути ініційований як зовнішньою командою, так і зникненням напруги живлення. На схемі керування станами цей процес відображається пунктирною лінією, що вказує на автоматичний, неконтрольований користувачем перехід.

Таким чином, система керування режимами роботи сенсора забезпечує ефективне використання ресурсів, адаптацію до потреб користувача і можливість інтеграції як у високопродуктивні системи, так і в енергоощадні автономні рішення.

3.3 Математична модель освітленості в навчальних аудиторіях

Освітленість E — це фізична величина, яка визначає, яку кількість світлового потоку Φ падає на одиницю площі S :

$$E = \frac{\Phi}{S}, \quad [E] = \text{лк (люкс)}. \quad (3.1)$$

Для системи з фоторезисторами або цифровим датчиком типу BH1750 модель зчитування освітленості може бути виражена через напругу на аналоговому вході мікроконтролера:

$$E \sim \frac{V_{in}}{R}, \quad (3.2)$$

де V_{in} — напруга, що формується на виході фоторезистора, R — опір у схемі з подільником напруги (залежить від освітленості: при збільшенні освітлення опір фоторезистора зменшується, а напруга на вході Arduino зростає).

У випадку цифрового сенсора BH1750, значення освітленості обчислюється безпосередньо в люксах за внутрішнім каліброваним алгоритмом модуля, і передається в цифровій формі через I²C. Для обробки таких даних використовується проста лінійна модель:

$$E_i = \text{Lux}_i, \quad i = 1, 2, \dots, n, \quad (3.3)$$

де E_i — освітленість, зафіксована -м сенсором або у певному напрямку.

У разі реалізації адаптивного керування, наприклад, обертання платформи на сервоприводах у напрямку максимальної освітленості, використовується логіка порівняння:

$$E_{\max} = \max(E_1, E_2, E_3, E_4), \quad (3.4)$$

де $E_1 \dots E_4$ — значення з чотирьох фоторезисторів або датчиків. Сервомеханізм обертається в напрямку, де зафіксовано $E_{\max}$.

3.4 Висновки до розділу 3

У третьому розділі бакалаврської кваліфікаційної роботи було детально розглянуто склад інформаційно-вимірювальної системи для автоматизованого контролю освітленості, зокрема функціональні можливості основних компонентів – мікроконтролера Arduino UNO, сервоприводу TowerPro SG 5010 і цифрового люксметра BH1750.

Arduino UNO обрано через його зручність, гнучкість та розвинену екосистему бібліотек, що полегшує програмування і інтеграцію системи. Сервоприводи забезпечують точне регулювання положення сенсорів залежно від рівня освітлення, а BH1750 дозволяє з високою точністю і стабільністю вимірювати освітленість у реальному часі.

Функціональна схема системи демонструє здатність до адаптивного керування освітленням, ґрунтуючись на зібраних даних. Це дозволяє оптимізувати витрати енергії, покращити комфорт користувачів і створити основу для подальшого розвитку розумних освітлювальних систем у навчальних закладах.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕНOSTІ В НАВЧАЛЬНИХ АУДИТОРІЯХ

4.1 Програмна реалізація сенсорної частини системи контролю освітленості

Основне завдання цієї частини програми — зчитування сигналу з сенсора освітленості (BH1750 або фоторезисторів), аналіз рівня природного освітлення в аудиторії та ухвалення рішення щодо зміни інтенсивності штучного освітлення. Якщо природного світла недостатньо [19], система автоматично збільшує яскравість світлодіодів. Якщо світла достатньо — підсвічування зменшується або повністю вимикається, що дозволяє економити електроенергію.

Уся система функціонує в режимі реального часу. Додатково вона забезпечує можливість змінювати колір світлодіодного освітлення через Wi-Fi, що дає змогу адаптувати мікроклімат освітлення відповідно до потреб користувача в навчальному або офісному просторі.

Під час ініціалізації в блоці `setup()` встановлюються виходи для RGB-світлодіодів. Запускається Wi-Fi-модуль на основі ESP8266, а також ініціалізується вебсервер на порту 80. Ініціалізується цифровий сенсор BH1750 через інтерфейс I2C. До відповідних пінів прив'язуються два сервоприводи — один для горизонтального, інший для вертикального руху. Початкові положення сервомоторів виставляються на 90° та 45° відповідно.

У кожному циклі виконання програма зчитує значення освітленості у люксах за допомогою BH1750 і виводить ці дані у монітор порту. Додатково використовуються чотири фоторезистори (верхній лівий, верхній правий, нижній лівий, нижній правий), які дозволяють визначити напрямок джерела світла.

Програма обчислює середні значення яскравості по горизонталі й вертикалі. Якщо різниця перевищує встановлену межу похибки ($\text{tol} = 50$),

відповідний сервопривід повільно повертається в бік більшої освітленості. Це дозволяє орієнтувати сенсор або камеру на найкраще освітлену ділянку.

Для зручного керування системою користувач може через вебінтерфейс змінювати колір освітлення, що дозволяє повністю контролювати атмосферу в приміщенні.

4.2 Програмна реалізація та програмне забезпечення виконавчої частини системиконтролю освітленості

Ця програма написана для мікроконтролера ESP8266 (наприклад, NodeMCU) і призначена для керування RGB-світлодіодами через Wi-Fi з використанням веб-інтерфейсу. Основна її функція полягає в регулюванні кольору світлодіодної стрічки у форматі RGB на основі значень HSV-моделі кольору (тон, насиченість, яскравість), що забезпечує плавну і гнучку зміну освітлення залежно від умов або потреб користувача.

RGB-світлодіод — це напівпровідниковий пристрій, який складається з трьох незалежних світлодіодів, об'єднаних в одному корпусі. Кожен з них відповідає за один з базових кольорів — червоний (Red), зелений (Green) і синій (Blue). Шляхом змішування цих трьох кольорів в різних пропорціях можливо створити будь-який відтінок світла, включаючи білий. Це досягається шляхом керування рівнем яскравості кожного каналу окремо, що здійснюється за допомогою широтно-імпульсної модуляції (PWM).

Однією з головних переваг RGB-світлодіодів є їхня гнучкість та універсальність. Завдяки можливості динамічної зміни кольору їх можна застосовувати не лише для освітлення, а й для візуалізації інформації, естетичного дизайну, інтерфейсів користувача. Наприклад, колір підсвічування може змінюватися залежно від подій або стану системи, що є дуже корисним у інтерактивних пристроях.

Використання RGB-світлодіодів у проектах з мобільними додатками особливо доцільне. Завдяки вбудованому Wi-Fi або Bluetooth-модулю

(наприклад, в ESP8266 або ESP32), можлива передача команд з мобільного додатку бездротовим способом. Таким чином користувач може у реальному часі змінювати колір, яскравість або режим світіння через інтерфейс додатку.

З математичної точки зору кожен з кольорів кодується значенням від 0 до 255. Це дає загалом $256^3 = 16\,777\,216$ можливих комбінацій кольору. Кольори можуть бути представлені у форматі RGB або HEX, що дозволяє точно відтворювати потрібні відтінки в інтерфейсах програм та на реальних пристроях. Наприклад, комбінація (255, 0, 0) відповідає чисто червоному кольору, а (255, 255, 255) — білому.

Енергоспоживання RGB-світлодіодів залишається порівняно низьким, навіть при максимальній яскравості. Це робить їх енергоефективними в автономних пристроях, таких як портативні гаджети або розумні лампи. Більшість RGB-світлодіодів керуються за допомогою транзисторів або спеціалізованих драйверів (наприклад, WS2812), які забезпечують точне та синхронізоване керування.

У дизайні мобільного додатку часто реалізується кольорова палітра, повзунки для регулювання R, G, B-каналів, або готові шаблони світлових режимів. Це створює гнучке і зручне керування для кінцевого користувача. Крім того, програміст може реалізувати синхронізацію кольору з зовнішніми подіями: погодою, музикою, таймерами чи повідомленнями.

Таким чином, RGB-світлодіоди є ідеальним вибором для інтерактивних систем освітлення, особливо у поєднанні з мобільними додатками. Вони забезпечують точне управління кольором, низьке енергоспоживання, просту інтеграцію з мікроконтролерами та широку функціональність в інтерфейсах. Їх використання відкриває великі можливості для створення сучасних, адаптивних та інтуїтивно зрозумілих рішень у сфері "розумного" освітлення.

В рамках роботи продумано, що ця частина програмного забезпечення фактично виконує роль зворотного зв'язку в межах повної системи контролю освітленості. Тобто, якщо в іншій частині програми або системи фіксується рівень освітленості (наприклад, за допомогою фоторезистора або цифрового

сенсора BH1750), то блок керування світлодіодами відповідає за адаптивну реакцію на зміну світлових умов. Таким чином, реалізується замкнене управління: спочатку визначається поточний рівень освітлення середовища, а потім, на основі цього, виконується відповідне коригування яскравості або кольору освітлення.

Користувач також має змогу вручну втручатися в роботу системи через веб-інтерфейс, змінюючи параметри HSV, що відкриває можливості для гібридного керування — як автоматичного, так і ручного. Це підвищує гнучкість та комфорт експлуатації, а також дозволяє системі адаптуватися до індивідуальних уподобань.

Підключення до Wi-Fi та ініціалізація системи. У функції `setup()` мікроконтролер:

- Налаштовує пін-коди для каналів червоного, зеленого та синього кольорів як виходи.
- Встановлює максимальне значення широтно-імпульсної модуляції (ШИМ) — 255.
- Підключається до заданої Wi-Fi мережі (SSID — FreeNet, пароль — password1) та конфігурує IP-адресу вручну.
- Після встановлення з'єднання запускається веб-сервер на порту 80.

Головний цикл `loop()`. Програма безперервно:

- Вмикає вбудований індикатор (BUILTIN_LED), сигналізуючи про активність.
- В залежності від вибраного режиму (`modeLED`) виконує відповідну функцію для генерації кольору світла:
 - `timerRunLed()` — плавне переливання кольору по колу.
 - `runningLights()` — зміна між трьома кольорами.
 - `runningLights2()` — переливання через шість відтінків.
 - `runningLights3()` — швидке прокручування кольорового кола з кроком у 10°.

Виводить обраний колір на світлодіоди за допомогою функції `turnRGB()`. Обробка HTTP-запитів. Коли користувач підключається до мікроконтролера через веб-браузер:

- Сервер приймає з'єднання та зчитує HTTP-запит.
- Якщо запит містить параметр `b=`, система змінює режим світіння (`modeLED`).

Якщо запит містить параметр `value=`, зчитуються чотири параметри: відтінок (H), насиченість (S), яскравість (V), швидкість анімації (Sp). Дані парсяться та передаються у функцію `HSVtoRGB()`, яка перетворює HSV у RGB.

У відповідь браузеру повертається проста HTML-сторінка з текстом “Керування LED через WiFi”.

Функція `HSVtoRGB()`. Ця функція реалізує конвертацію кольору з HSV у RGB. В результаті отримані значення RGB масштабуються до діапазону 0–255 та зберігаються у змінних `globalR`, `globalG`, `globalB`.

Режими світіння:

- `timerRunLed()` — постійне обертання по колу відтінків кольору, з урахуванням обраної швидкості.
- `runningLights()` — циклічний перехід між червоним, зеленим та синім кольорами.
- `runningLights2()` — розширене переливання з більшою кількістю кольорів.
- `runningLights3()` — більш динамічний режим переливання з великим кроком.

Ця програма дозволяє дистанційно керувати RGB-світлодіодами за допомогою Wi-Fi з будь-якого пристрою з браузером. Інтерфейс користувача мінімальний, але достатній для основних задач: зміна кольору, яскравості, швидкості анімації та вибір режимів. Таке рішення ідеально підходить для інтерактивних проєктів освітлення, декоративного підсвічування або навчальних цілей.

Після запуску програмного забезпечення користувач потрапляє на головну панель управління, яка виконує роль базового інтерфейсу взаємодії із системою керування освітленням. Даний екран є своєрідним навігаційним хабом, з якого можна здійснювати швидкий перехід до основних функціональних модулів системи.

Інтерфейс головного вікна (рис. 4.1) поділений на два ключові розділи: «Налаштування» та «Зони будівлі». Обидва блоки реалізовані у формі інтерактивного меню, що забезпечує інтуїтивно зрозумілу логіку взаємодії користувача з програмою, зводячи до мінімуму час на освоєння роботи з нею.

У вкладці «Налаштування» реалізовано кілька важливих функцій для персоналізації програмного продукту. Зокрема, користувач має змогу вибрати бажану мову інтерфейсу. Система підтримує три мовні локалізації: українську, англійську. Зміна мови виконується динамічно, без необхідності перезавантаження додатку чи втрати попередніх налаштувань.

Ще однією важливою можливістю є конфігурація параметрів часу. Користувач може обрати між автоматичним визначенням поточного часу на основі системного годинника комп'ютера, або ж задати його вручну через інтерфейс програми. Ця функція є необхідною для коректної роботи автоматизованих сценаріїв освітлення, зокрема таймерів і добових режимів.

Додатково в розділі налаштувань передбачена функція підключення LED-пристроїв. Програма автоматично розпізнає нові світлодіодні стрічки або лампи, підключені до системи через відповідний інтерфейс, і додає їх до переліку доступних пристроїв. Це значно спрощує процес розширення освітлювальної мережі без потреби в ручному конфігуруванні адрес або параметрів.

Ця інструкція описує лише частину функціоналу, доступного в системі керування освітленням. Розроблений інтерфейс поєднує у собі гнучкість налаштувань, підтримку декількох мов і простоту інтеграції пристроїв,

що робить програму універсальним інструментом для керування освітленням в навчальних, комерційних чи побутових приміщеннях.

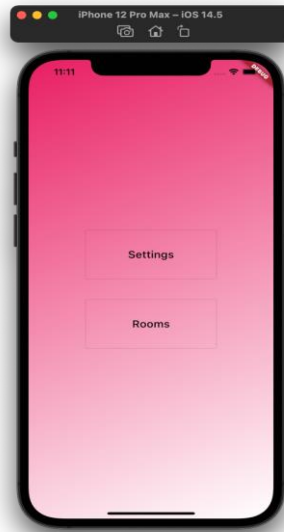


Рисунок 4.1 – Головний екран

Налаштування системи представлено рис. 4.2.

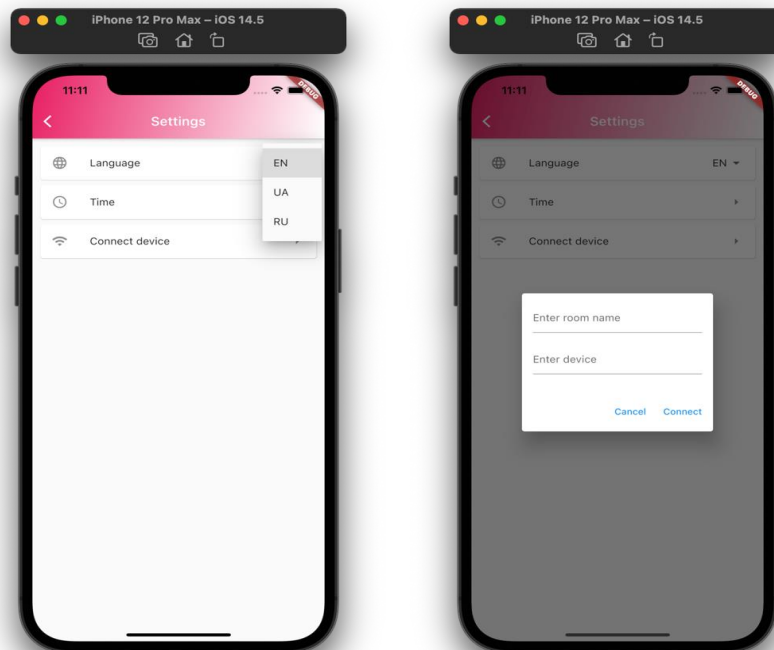


Рисунок 4.2 – Налаштування системи

Меню налаштувань представлено на рис. 4.3.

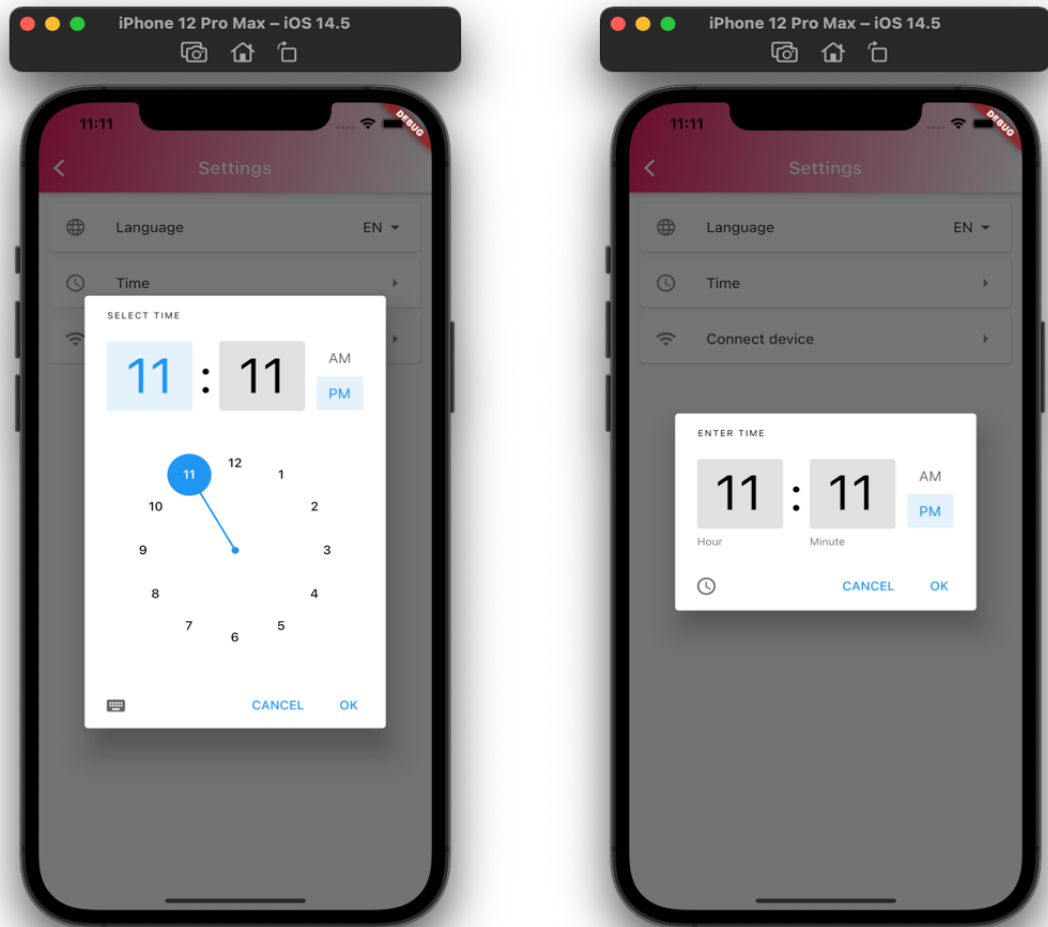


Рисунок 4.3 – Меню налаштувань

Керування зонами аудиторії (рис. 4.4). Повернувшись на головний екран, користувач може перейти до розділу «Зони аудиторії». У цьому розділі відображається структурована карта приміщення з позначеними аудиторіями, в яких встановлено LED-освітлення. Кожна зона має власне позначення та набір світлових пристроїв, прив'язаних до неї.

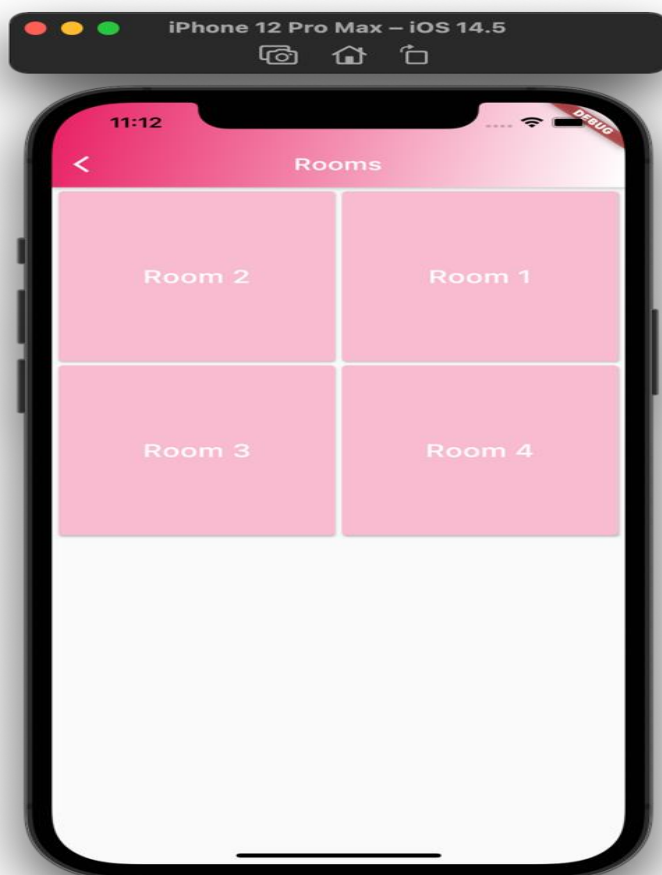


Рисунок 4.4 – Зони аудиторії

Вибравши певну зону, користувач переходить до меню налаштування освітлення (рис. 4.5) в цій області. У ньому можна:

- Обрати конкретний світлодіодний елемент.
- Регулювати яскравість, температуру світла (тепле/холодне), а також колір (при наявності RGB-світильників).
- Перевірити поточний стан пристрою (ввімкнений/вимкнений) та змінити його вручну.
- Використати функцію автоматичного режиму, яка підлаштовує освітлення відповідно до часу доби (ранок, день, вечір, ніч).
- Налаштувати таймер, що дозволяє задати час автоматичного вмикання або вимикання світла.

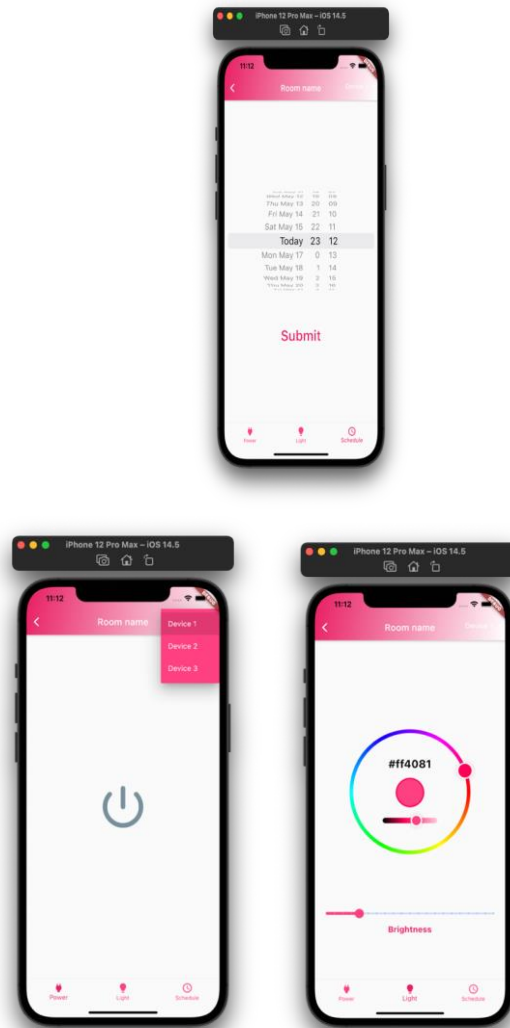


Рисунок 4.5 – Налаштування світла

Загальні навігаційні можливості. Кожен екран додатку має елементи навігації, що дозволяють оперативно повертатись до головного меню або переходити між підрозділами. Таким чином, користувач завжди має доступ до повного функціоналу системи без необхідності перезапуску чи оновлення сторінок.

Завдяки інтуїтивно зрозумілому інтерфейсу, багатомовній підтримці та гнучкому керуванню освітленням, програмне забезпечення дозволяє максимально ефективно управляти світловими параметрами у будівлі, створюючи комфортне та адаптивне середовище.

4.3 Висновки до розділу 4

У четвертому розділі бакалаврської кваліфікаційної роботи було зроблено та описано інтерфейс програмного забезпечення, що забезпечує інтерактивне керування освітленням у приміщеннях на основі отриманих даних з сенсорів. Інтерфейс є зручним, інтуїтивно зрозумілим та підтримує багатомовність, що підвищує зручність користування для широкого кола користувачів.

Програмний продукт дозволяє:

- швидко перемикатися між розділами керування;
- налаштовувати інтерфейс та освітлення згідно з уподобаннями;
- автоматично адаптувати освітлення до часу доби;
- вручну змінювати параметри яскравості, кольору та температури світла;
- керувати зонами приміщення та встановлювати таймери роботи світлових пристроїв.

Завдяки цьому користувач отримує гнучкий інструмент для ефективного контролю світлових умов у приміщенні, що сприяє створенню комфортного середовища та оптимізації енергоспоживання.

ВИСНОВКИ

У бакалаврській кваліфікаційній роботі було розроблено автоматизовану систему контролю освітленості в навчальних аудиторіях, що базується на поєднанні апаратних та програмних засобів. Основною метою стало створення комфортного, енергоефективного та адаптивного середовища освітлення, яке враховує зміну природного світла, час доби та індивідуальні потреби користувачів.

Проведений аналіз існуючих методів контролю освітлення показав, що традиційні підходи є малоефективними в умовах сучасних навчальних закладів. Впровадження сенсорних технологій, цифрових датчиків та мікроконтролерів дозволяє досягти значно вищої точності, оперативності й гнучкості у керуванні освітленням.

Структурна та електрична схеми системи були розроблені з урахуванням простоти реалізації та доступності елементів. Використання Arduino UNO забезпечує легке програмування, а підключення цифрового люксметра BH1750 – точні вимірювання освітленості. Завдяки сервоприводам можливе адаптивне позиціонування сенсорів для підвищення точності в різних умовах освітлення.

Результати розрахунків показали допустимі значення похибок вимірювання, що підтверджує ефективність обраної схеми для контролю освітлення. Програмне забезпечення з графічним інтерфейсом дозволяє зручно керувати світловими параметрами, забезпечуючи зміну мови, налаштування зон, встановлення таймерів та ручне регулювання яскравості й кольору.

Таким чином, створена система є ефективним засобом автоматизованого керування освітленням, який може бути впроваджений у навчальні заклади для підвищення якості освітнього процесу, комфорту користувачів і зниження енергоспоживання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision architectural elements and future directions", Future Generation Computer Systems, 2013. — 45 p.
2. Вступ до засобів управління освітленням [Електронний ресурс] – К.: Ділує, 2019. — 3 с. — Режим доступу до ресурса: <https://lightingcontrolsassociation.org/2017/07/21/introduction-to-lighting-controls>.
3. Neelam Verma, Anjali Jain. " Optimized Automatic Lighting Control in a Hotel Building for Energy Efficiency”, International Conference on Power Energy, 2018. — 170 p.
4. H. Park et al. "Soummotulti-modal Illumimote: Multi-modal and high-fidelity light sensor module for wireless sensor networks", IEEE Sensors Journal, 2006. — 170 p.
5. Hajjaj, M., Miki, M., & Shimohara, K. “The Effect of Using the Intelligent Lighting System to Deduct the Power Consumption at the Office”, IEEE 7th Conference on Systems, Process and Control (ICSPPC), 2019. — 66 p.
6. Smart Wallbox Sensors [Електронний ресурс]. — 2020. — Режим доступу до ресурса: <https://www.leviton.com/en/products/brands/smart-sensors>.
7. Smart lightning for top class schools [Електронний ресурс]. — 2018. — Режим доступу до ресурса: <https://www.interact-lighting.com/global/what-is-possible/interact-pro/schools>.
8. Understanding Building Automation and Control Systems [Електронний ресурс]. — 2013. — Режим доступу до ресурса: http://www.kmcccontrols.com/products/Understanding_Building_Automation_and_Control_Systems.aspx.
9. J. Burke et al. "Embedding expression: Pervasive computing architecture for art and entertainment", Elsevier PMC Journal, 2006. — 17 p.

10. T. Ball and S. Rajamani. "The SLAM Toolkit. In Proceedings of CAV'2001 (13th Conference on Computer Aided Verification) ", volume 2102 of Lecture Notes in Computer Science, Paris, 2001. — 260—264 pp.
11. Payne R. "Beginning App Development with Flutter: Create Cross-Platform Mobile Apps / Rap Payne", Apress, 2019. — 334 p.
12. Michael McRoberts. "Beginning Arduino", Apress, 2010. — 50 p.
13. Huaguo J. "Study and Application of Encoding and Decoding Algorithms for Colored Two-dimensional Code on Mobile Terminals / Jia Huaguo ", Hangzhou, Zhejiang University of Technology, 2009. — 17 p.
14. David Kushner. "The Making of Arduino", IEEE Spectrum, 2015. — 20 p.
15. Adam Gerber, Clifton Craig. "Learn Android Studio. Build Android Apps quickly and effectively", Apress, 2015. — 129 p.
16. Zaccagnino C. "Programming Flutter: Native, Cross-Platform Apps the Easy Way (The Pragmatic Programmers) / Carmine Zaccagnino", Pragmatic Bookshelf, 2020. — 330 p.
17. Grade Booch, James Rumbaugh, Ivar Jacobson. "The Unified Modeling Language Reference Manual", Addison-Wesley, 2001. — 33 p.
18. Philippe Kruchten. "The Rational Unified Process -An Introduction", 2nd edition, Addison-Wesley, 2000. — 170 p.
19. Блащук М. Р. Автоматизована система контролю освітленості в навчальних аудиторіях [Електронний ресурс] / М. Р. Блащук, О. В. Бісікало // LIV Всеукраїнська науково-технічна конференція факультету інформаційних електронних систем. - Електрон. текст. дані. - 2025. - Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2025/paper/view/24303>

Додатки

Вінницький національний технічний університет

(повне найменування вищого навчального закладу)

Кафедра автоматизації та інтелектуальних інформаційних технологій

(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ

д.т.н., професор Бісікало О. В.

(підпис)

«___» _____ 2025 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на бакалаврську кваліфікаційну роботу

Автоматизована система контролю освітленості в навчальних аудиторіях

08-31.БКР.001.02.000 ТЗ

Керівник бакалаврської кваліфікаційної роботи

д.т.н., проф. Бісікало О. В.

«___» _____ 2025 р.

Розробив студент гр. АКІТР-21бз

Блащук М.Р.

«___» _____ 2025 р.

Додаток А
(обов'язковий)
Технічне завдання на бакалаврську кваліфікаційну роботу

1 Підстава для проведення робіт

Підставою для виконання бакалаврської кваліфікаційної роботи на тему: «Автоматизована система контролю освітленості в навчальних аудиторіях» є наказ № 97 від 20.03.2025 р.

Термін виконання робіт:

початок 30.09.2024 р.
кінець 10.06.2025 р.

2 Мета та вихідні дані для проведення робіт

Метою бакалаврської кваліфікаційної роботи є підвищення ефективності та енергоощадності освітлення в навчальних аудиторіях шляхом розробки автоматизованої системи моніторингу та регулювання рівня освітленості з використанням сенсорних технологій і мікроконтролера ESP8266.

Вихідними даними для проведення робіт є індивідуальне завдання на бакалаврську кваліфікаційну роботу від 20.03.2025 р.

3 Етапи виконання робіт

Виконавцем всіх перерахованих в даному розділі етапів є: студент групи **АКІТР-2163** Блащук Максим Романович факультету інтелектуальних інформаційних технологій та автоматизації Вінницького національного технічного університету, а замовником є кафедра автоматизації та інтелектуальних інформаційних технологій.

№ Етапу	Зміст етапу	Строки виконання
Е1	Аналіз існуючих методів контролю освітленості в навчальних аудиторіях	03.09 – 26.02
Е2	Розробка структурної схеми та алгоритма роботи системи контролю освітленості в навчальних аудиторіях	01.03 – 19.03
Е3	Розробка функціонально-апаратної схеми системи контролю освітленості в навчальних аудиторіях	22.03 – 09.04
Е4	Програмна реалізація системи контролю освітленості в навчальних аудиторіях	12.04 – 28.05

3 Призначення і галузь застосування

Система призначена для автоматизованого контролю та регулювання рівня освітленості в навчальних аудиторіях з метою забезпечення комфортних умов освітлення й енергоефективності.

5 Технічні дані

5.1 Тип системи – автоматизована система контролю освітленості;

5.2 Метод контролю – сенсорний (фоторезисторний та цифровий датчик освітленості BH1750);

5.3 Об'єкт контролю – рівень освітленості в навчальних аудиторіях.

6 Джерела розробки

1. Методичні вказівки до виконання бакалаврських дипломних робіт (проектів) для студентів спеціальностей 126 – «Інформаційні системи та технології», 151 – «Автоматизація та комп'ютерно-інтегровані технології» / Уклад. Р. Н. Кветний, О. М. Бевз, О. В. Бісікало. – Вінниця : ВНТУ, 2019. – 26 с.
2. ДБН В.2.5-28:2018. Природне і штучне освітлення : чинні з 01.01.2019. Київ : Мінрегіон України, 2018. 89 с.
3. ДСТУ EN 12464-1:2016. Світло та освітлення. Освітлення робочих місць. Частина 1. Внутрішні робочі місця (EN 12464-1:2011, IDT). Київ : ДП «УкрНДНЦ», 2016. 65 с.
4. ДБН В.2.2-3:2018. Будинки і споруди. Заклади освіти : чинні з 01.09.2018. Київ : Мінрегіон України, 2018. 92 с.
5. Про затвердження Державних санітарних норм та правил «Гігієнічні вимоги до улаштування, утримання і режиму спеціальних загальноосвітніх шкіл (шкіл-інтернатів) для дітей, які потребують корекції фізичного та (або) розумового розвитку, та навчально-реабілітаційних центрів» : Наказ МОЗ України від 20.02.2013 № 144 // Офіційний вісник України. 2013. № 25. Ст. 1045.

Здобувач ст. гр. АКІТР-21бз _____ Блащук М.Р.

Додаток Б
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

**АВТОМАТИЗОВАНА СИСТЕМА КОНТРОЛЮ ОСВІТЛЕНOSTІ В
НАВЧАЛЬНИХ АУДИТОРІЯХ**

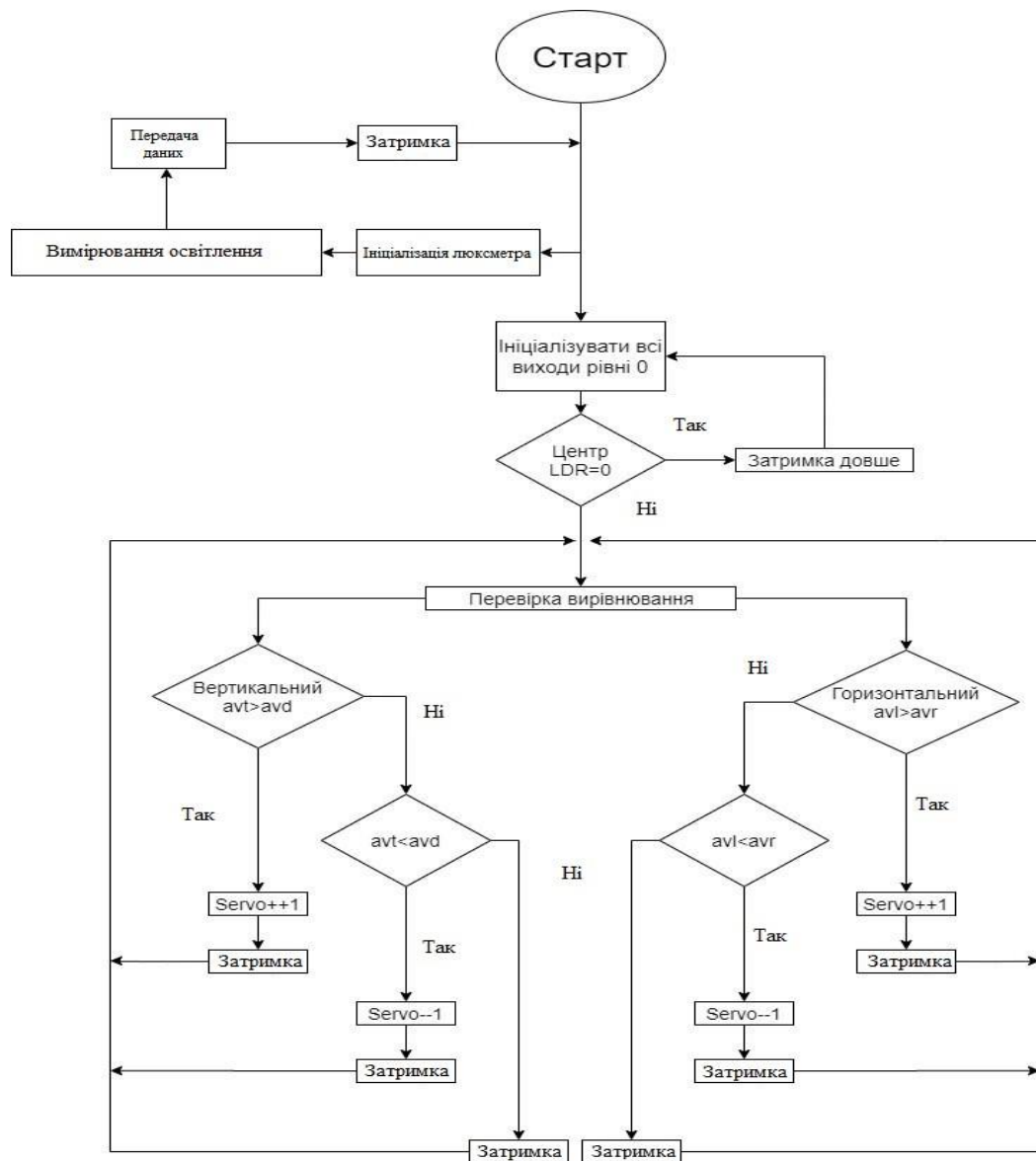


Рисунок Б.1 – Алгоритм роботисистеми освітлення

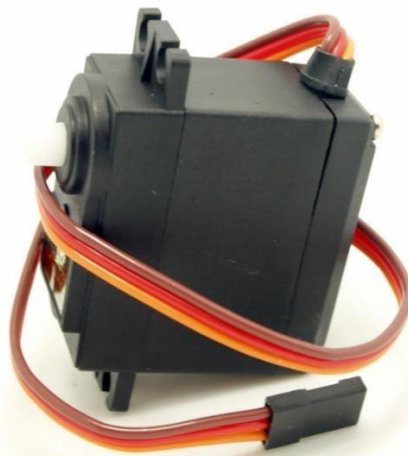


Рисунок Б.2 – Зовнішній вигляд сервопривода TowerPro SG 5010

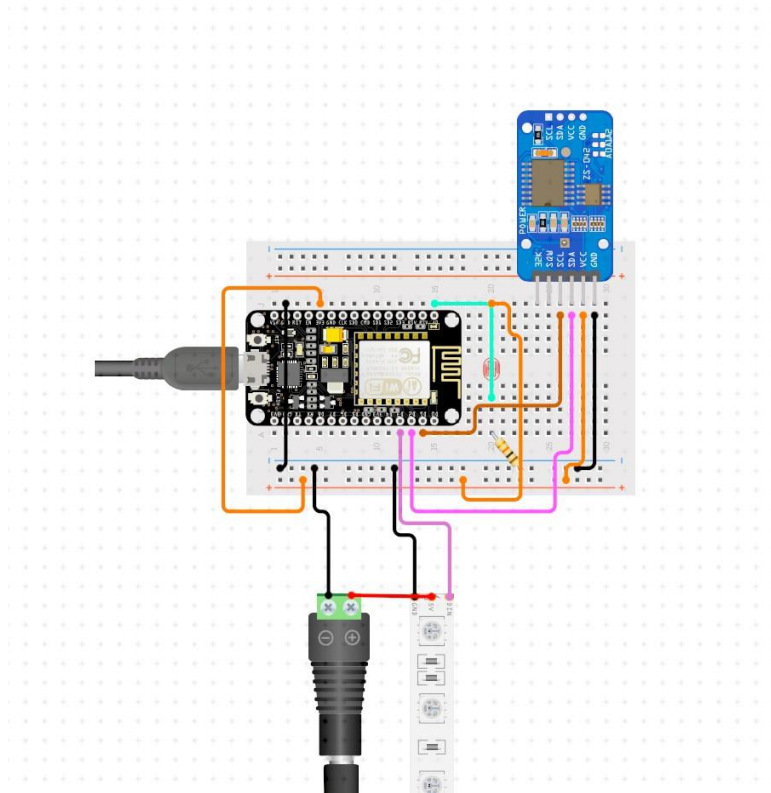


Рисунок Б.3 – Схема під'єднання світлодіодного пристрою

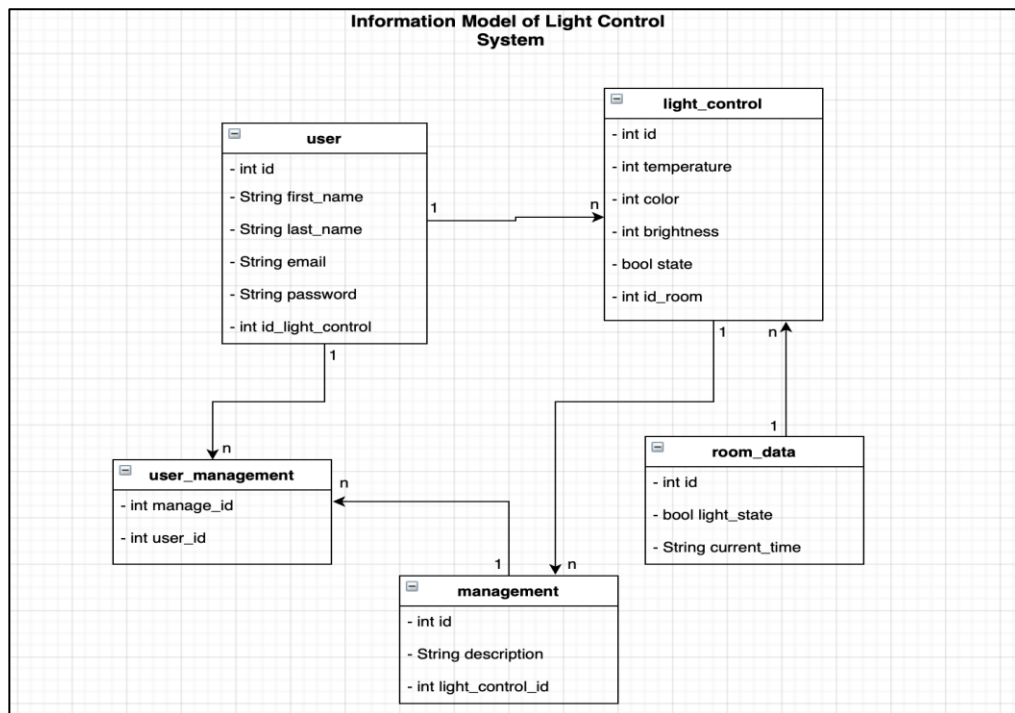


Рисунок Б.4 – Інформаційна модель системи

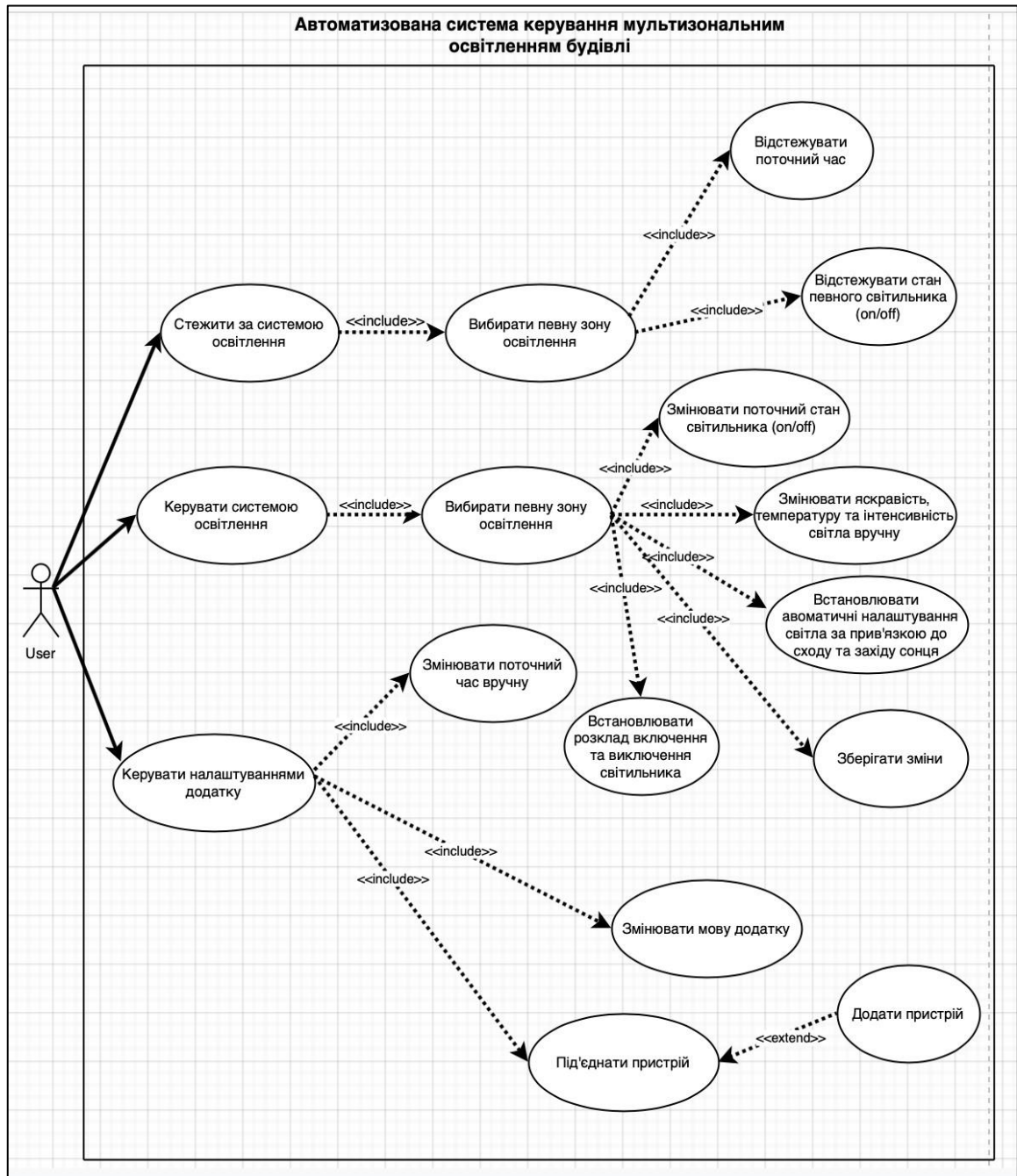


Рисунок Б.5 – Діаграма прецедентів системи

Додаток В

(обов'язковий)

```
#include <ESP8266WiFi.h>

#define maxPMW 255 // Максимальне значення для ШІМ

// Визначення пінів для світлодіодів RGB
#define ledPinR 12 // D6
#define ledPinG 14 // D5
#define ledPinB 13 // D7

// Налаштування IP-адреси
IPAddress ip(192, 168, 8, 201);
IPAddress gateway(192, 168, 8, 254); // Шлюз
IPAddress subnet(255, 255, 255, 0); // Маска підмережі
WiFiServer server(80); // Створення сервера на порту 80

// Дані для підключення до WiFi
const char* ssid = "FreeNet";
const char* password = "password1";

// Змінні для зберігання значень H, S, V та швидкості анімації
String valueStringH = String(100);
String valueStringS = String(100);
String valueStringV = String(100);
String valueStringSp = String(400);
int globalR, globalG, globalB; // Значення кольору у форматі RGB

int modeLED = 3; // Поточний режим світіння
byte modeCurrentColor = 0;
unsigned long runTimer = 0; // Таймер

void setup() {
    // Налаштування пінів як виходи
    pinMode(ledPinR, OUTPUT);
    pinMode(ledPinG, OUTPUT);
    pinMode(ledPinB, OUTPUT);
    pinMode(BUILTIN_LED, OUTPUT);
    analogWriteRange(maxPMW); // Встановлення діапазону ШІМ

    // Підключення до WiFi та налаштування IP
    WiFi.begin(ssid, password);
    WiFi.config(ip, gateway, subnet);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500); // Очікування з'єднання
    }
    server.begin(); // Запуск сервера
}
```

```

void loop() {
    digitalWrite(BUILTIN_LED, 1); // Вмикання допоміжного світлодіода

    // Виклик функцій відповідно до вибраного режиму
    if (modeLED == 2) timerRunLed();
    if (modeLED == 3) runningLights();
    if (modeLED == 4) police();
    if (modeLED == 5) runningLights2();
    if (modeLED == 6) runningLights3();
    turnRGB(globalR, globalG, globalB); // Вивід кольору на стрічку

    WiFiClient client = server.available(); // Очікування клієнта

    if (client) {
        digitalWrite(BUILTIN_LED, 0); // Вимикання допоміжного світлодіода
        String header = "";
        while (client.connected()) {
            if (client.available()) {
                char c = client.read();
                header += c;

                if (c == '\n') {
                    // Обробка запитів клієнта
                    if (header.indexOf("GET /?b=") >= 0) {
                        int pos1 = header.indexOf('=');
                        int pos2 = header.indexOf('&');
                        String parseB = header.substring(pos1 + 1, pos2);
                        modeLED = parseB.toInt();
                    }
                    if (header.indexOf("GET /?value=") >= 0) {
                        int pos1 = header.indexOf('=');
                        int pos2 = header.indexOf('&');
                        String getH, getS, getV, getSp;
                        for (int i = pos1 + 1, bb = 0; i < pos2; i++) {
                            if (header[i] == '_') {
                                bb++;
                                continue;
                            }
                        }
                        switch (bb) {
                            case 0: getH += header[i]; break;
                            case 1: getS += header[i]; break;
                            case 2: getV += header[i]; break;
                            case 3: getSp += header[i]; break;
                        }
                    }
                    if (modeLED == 1) valueStringH = getH;
                    valueStringS = getS;
                    valueStringV = getV;
                    valueStringSp = getSp;
                    HSVtoRGB(getH.toInt(), getS.toInt(), getV.toInt());
                }
            }
        }
    }
}

```

```

        // Відповідь клієнту та передача HTML-сторінки
        client.println("HTTP/1.1 200 OK");
        client.println("Content-type:text/html");
        client.println("Connection: close");
        client.println();
        client.println("<!DOCTYPE html><html><head><meta charset='UTF-8'></head><body>");
        client.println("<h1>Керування LED через WiFi</h1>");
        client.println("</body></html>");
        break;
    }
}
header = "";
client.stop();
}
}

void HSVtoRGB(int H, int S = 100, int V = 100) {
    if (H == 360) H = 0;
    int Hi = H / 60;
    int Vmin = ((100 - S) * V) / 100;
    int a = ((H % 60) * (V - Vmin)) / 60;
    int Vinc = Vmin + a;
    int Vdec = V - a;
    int r = 0, g = 0, b = 0;
    switch (Hi) {
        case 0: r = V; g = Vinc; b = Vmin; break;
        case 1: r = Vdec; g = V; b = Vmin; break;
        case 2: r = Vmin; g = V; b = Vinc; break;
        case 3: r = Vmin; g = Vdec; b = V; break;
        case 4: r = Vinc; g = Vmin; b = V; break;
        case 5: r = V; g = Vmin; b = Vdec; break;
    }
    globalR = (r * maxPMW) / 100;
    globalG = (g * maxPMW) / 100;
    globalB = (b * maxPMW) / 100;
}

void turnRGB(int r, int g, int b) {
    analogWrite(ledPinR, maxPMW - r);
    analogWrite(ledPinG, maxPMW - g);
    analogWrite(ledPinB, maxPMW - b);
}

void timerRunLed() {
    if (millis() > (runTimer + valueStringSp.toInt() / 2)) {
        valueStringH = valueStringH.toInt() + 1;
        if (valueStringH == "361") valueStringH = "0";
        HSVtoRGB(valueStringH.toInt(), valueStringS.toInt(), valueStringV.toInt());
        runTimer = millis();
    }
}

```



```

    }
}

void runningLights() {
    int curModeColor[] = {0, 120, 240};
    if (millis() > (runTimer + valueStringSp.toInt())) {
        runTimer = millis();
        modeCurrentColor++;
    }
    if (modeCurrentColor > 2) modeCurrentColor = 0;
    HSVtoRGB(curModeColor[modeCurrentColor], valueStringS.toInt(), valueStringV.toInt());
}

void police() {
    // Реалізація ефекту "поліцейської мигалки"
    // Логіка чергування кольорів з використанням таймерів
    if (millis() > (runTimer + valueStringSp.toInt() * 10.5)) runTimer = millis();
    // Додати детальну логіку за аналогією до HSVtoRGB()
}

void runningLights2() {
    int curModeColor[] = {0, 60, 120, 180, 240, 300};
    if (millis() > (runTimer + valueStringSp.toInt())) {
        runTimer = millis();
        modeCurrentColor++;
    }
    if (modeCurrentColor > 5) modeCurrentColor = 0;
    HSVtoRGB(curModeColor[modeCurrentColor], valueStringS.toInt(), valueStringV.toInt());
}

void runningLights3() {
    if (millis() > (runTimer + valueStringSp.toInt())) {
        valueStringH = valueStringH.toInt() + 10;
        if (valueStringH.toInt() > 360) valueStringH = "0";
        HSVtoRGB(valueStringH.toInt(), valueStringS.toInt(), valueStringV.toInt());
        runTimer = millis();
    }
}

```

```

#include <ESP8266WiFi.h>
#include <Wire.h>
#include <BH1750FVI.h>
#include <Servo.h>

#define maxPMW 255

#define ledPinR 12 // D6
#define ledPinG 14 // D5
#define ledPinB 13 // D7

IPAddress ip(192, 168, 8, 201);
IPAddress gateway(192, 168, 8, 254);
IPAddress subnet(255, 255, 255, 0);
WiFiServer server(80);

const char* ssid = "FreeNet";
const char* password = "password1";

String valueStringH = String(100);
String valueStringS = String(100);
String valueStringV = String(100);
String valueStringSp = String(400);
int globalR, globalG, globalB;

int modeLED = 3;
byte modeCurrentColor = 0;
unsigned long runTimer = 0;

BH1750 lightMeter;
Servo horizontal;
Servo vertical;
int servoh = 90;
int servohLimitHigh = 180;
int servohLimitLow = 0;
int servov = 45;
int servovLimitHigh = 180;
int servovLimitLow = 0;

int ldrLt = A2;
int ldrRt = A3;
int ldrLd = A1;
int ldrRd = A0;
int tol = 50;

void setup() {

```

```

pinMode(ledPinR, OUTPUT);
pinMode(ledPinG, OUTPUT);
pinMode(ledPinB, OUTPUT);
pinMode(BUILTIN_LED, OUTPUT);
analogWriteRange(maxPMW);

WiFi.begin(ssid, password);
WiFi.config(ip, gateway, subnet);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
}
server.begin();

Serial.begin(9600);
lightMeter.begin();
horizontal.attach(9);
vertical.attach(10);
horizontal.write(servoh);
vertical.write(servov);
}

void loop() {
  digitalWrite(BUILTIN_LED, 1);

  if (modeLED == 2) timerRunLed();
  if (modeLED == 3) runningLights();
  if (modeLED == 4) police();
  if (modeLED == 5) runningLights2();
  if (modeLED == 6) runningLights3();
  turnRGB(globalR, globalG, globalB);

  WiFiClient client = server.available();
  if (client) {
    digitalWrite(BUILTIN_LED, 0);
    String header = "";
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();
        header += c;

        if (c == '\n') {
          if (header.indexOf("GET /?b=") >= 0) {
            int pos1 = header.indexOf('=');
            int pos2 = header.indexOf('&');
            String parseB = header.substring(pos1 + 1, pos2);
            modeLED = parseB.toInt();
          }
          if (header.indexOf("GET /?value=") >= 0) {
            int pos1 = header.indexOf('=');
            int pos2 = header.indexOf('&');
            String getH, getS, getV, getSp;
            for (int i = pos1 + 1, bb = 0; i < pos2; i++) {

```

```

    if (header[i] == '_') {
        bb++;
        continue;
    }
    switch (bb) {
        case 0: getH += header[i]; break;
        case 1: getS += header[i]; break;
        case 2: getV += header[i]; break;
        case 3: getSp += header[i]; break;
    }
}
if (modeLED == 1) valueStringH = getH;
valueStringS = getS;
valueStringV = getV;
valueStringSp = getSp;
HSVtoRGB(getH.toInt(), getS.toInt(), getV.toInt());
}

client.println("HTTP/1.1 200 OK");
client.println("Content-type:text/html");
client.println("Connection: close");
client.println();
client.println("<!DOCTYPE html><html><head><meta charset='UTF-
8'></head><body>");
client.println("<h1>Керування LED через WiFi</h1>");
client.println("</body></html>");
break;
}
}
}
header = "";
client.stop();
}

uint16_t lux = lightMeter.readLightLevel();
Serial.print("Light: ");
Serial.print(lux);
Serial.println(" lx");
delay(1000);

int lt = analogRead(ldrlt);
int rt = analogRead(ldrrt);
int ld = analogRead(ldrld);
int rd = analogRead(ldrrd);
int avt = (lt + rt) / 2;
int avd = (ld + rd) / 2;
int avl = (lt + ld) / 2;
int avr = (rt + rd) / 2;
int dvert = abs(avt - avd);
int dhoriz = abs(avl - avr);

if (dhoriz > tol) {

```

```

    if (avl > avr) {
        if (servoh - 1 >= servohLimitLow) servoh--;
    } else if (avl < avr) {
        if (servoh + 1 <= servohLimitHigh) servoh++;
    }
    horizontal.write(servoh);
}

if (dvert > tol) {
    if (avt > avd) {
        if (servov + 1 <= servovLimitHigh) servov++;
    } else if (avt < avd) {
        if (servov - 1 >= servovLimitLow) servov--;
    }
    vertical.write(servov);
}
}

void HSVtoRGB(int H, int S = 100, int V = 100) {
    if (H == 360) H = 0;
    int Hi = H / 60;
    int Vmin = ((100 - S) * V) / 100;
    int a = ((H % 60) * (V - Vmin)) / 60;
    int Vinc = Vmin + a;
    int Vdec = V - a;
    int r = 0, g = 0, b = 0;
    switch (Hi) {
        case 0: r = V; g = Vinc; b = Vmin; break;
        case 1: r = Vdec; g = V; b = Vmin; break;
        case 2: r = Vmin; g = V; b = Vinc; break;
        case 3: r = Vmin; g = Vdec; b = V; break;
        case 4: r = Vinc; g = Vmin; b = V; break;
        case 5: r = V; g = Vmin; b = Vdec; break;
    }
    globalR = (r * maxPMW) / 100;
    globalG = (g * maxPMW) / 100;
    globalB = (b * maxPMW) / 100;
}

void turnRGB(int r, int g, int b) {
    analogWrite(ledPinR, maxPMW - r);
    analogWrite(ledPinG, maxPMW - g);
    analogWrite(ledPinB, maxPMW - b);
}

void timerRunLed() {
    if (millis() > (runTimer + valueStringSp.toInt() / 2)) {
        valueStringH = valueStringH.toInt() + 1;
        if (valueStringH == "361") valueStringH = "0";
        HSVtoRGB(valueStringH.toInt(), valueStringS.toInt(), valueStringV.toInt());
        runTimer = millis();
    }
}

```

```

}

void runningLights() {
  int curModeColor[] = {0, 120, 240};
  if (millis() > (runTimer + valueStringSp.toInt())) {
    runTimer = millis();
    modeCurrentColor++;
  }
  if (modeCurrentColor > 2) modeCurrentColor = 0;
  HSVtoRGB(curModeColor[modeCurrentColor], valueStringS.toInt(), valueStringV.toInt());
}

void police() {
  // Логіка "поліцейської мигалки" може бути реалізована тут
  if (millis() > (runTimer + valueStringSp.toInt() * 10.5)) runTimer = millis();
}

void runningLights2() {
  int curModeColor[] = {0, 60, 120, 180, 240, 300};
  if (millis() > (runTimer + valueStringSp.toInt())) {
    runTimer = millis();
    modeCurrentColor++;
  }
  if (modeCurrentColor > 5) modeCurrentColor = 0;
  HSVtoRGB(curModeColor[modeCurrentColor], valueStringS.toInt(), valueStringV.toInt());
}

void runningLights3() {
  if (millis() > (runTimer + valueStringSp.toInt())) {
    valueStringH = valueStringH.toInt() + 10;
    if (valueStringH.toInt() > 360) valueStringH = "0";
    HSVtoRGB(valueStringH.toInt(), valueStringS.toInt(), valueStringV.toInt());
    runTimer = millis();
  }
}

```

Додаток Г
(обов'язковий)

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Автоматизована система контролю освітленості в навчальних аудиторіях

Тип роботи: бакалаврська кваліфікаційна робота

(бакалаврська кваліфікаційна робота / магістерська кваліфікаційна робота)

Підрозділ АПТ, ФПТА, АКІТР-21бз

(кафедра, факультет, навчальна група)

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism 2,9 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

- ☒ Запозичення, виявлені у роботі, є законними і не містять ознак плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту
- ☐ У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- ☐ У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як спроб укриття плагіату, фабрикації, фальсифікації, що суперечить вимогам законодавства та нормам академічної доброчесності. Робота до захисту не приймається.

Експертна комісія:

Олег Бісікало, зав. каф. АІІТ

(прізвище, ініціали, посада)

Любов Янчук, секретар каф. АІІТ

(прізвище, ініціали, посада)

(підпис)
(підпис)

Особа, відповідальна за перевірку

(підпис)

Овчинников К. В.
(прізвище, ініціали)

З висновком експертної комісії ознайомлений(-на)

Керівник

(підпис)

Бісікало О. В.

(прізвище, ініціали, посада)

Здобувач

(підпис)

Блащук М. Р.

(прізвище, ініціали)