

Міністерство освіти і науки України
Державний заклад
«Луганський національний університет імені Тараса Шевченка»

Факультет технологій та інформаційних систем

Кафедра інформаційних технологій та систем

Возняк Сергій Віталійович

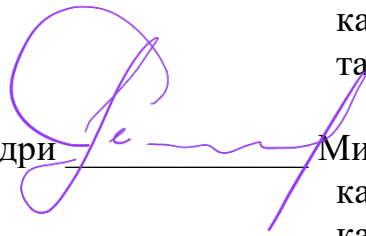
**АВТОМАТИЗОВАНА СИСТЕМА НА ПЛАТФОРМІ ARDUINO
КЕРУВАННЯ ОСВІТЛЕННЯ ТА ОПАЛЕННЯ ПОБУТОВОГО
ПРИМІЩЕННЯ**

кваліфікаційна робота

**здобувача вищої освіти першого (бакалаврського) рівня
освітньої програми «Комп'ютерна інженерія»
за спеціальністю 123 Комп'ютерна інженерія**

Особистий підпис _____  Сергій ВОЗНЯК

Науковий керівник _____  Володимир ДОНЧЕНКО,
старший викладач
кафедри інформаційних технологій
та систем

Завідувач кафедри _____  Микола СЕМЕНОВ,
кандидат педагогічних наук, доцент
кафедри інформаційних технологій
та систем

Міністерство освіти і науки України
Державний заклад «Луганський національний університет
імені Тараса Шевченка»

Факультет (інститут)	Факультет технологій та інформаційних систем (повна назва)
Кафедра	Інформаційних технологій та систем (повна назва)
Освітній ступень	Бакалавр (код, назва)
Напрямок підготовки	123 Комп'ютерна інженерія (код, назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри ІТС
М.А. Семенов

(підпис)

(ініціали, прізвище)

“ ” 2025 р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТУ

Возняку Сергію Віталійовичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Автоматизована система на платформі
Arduino керування освітлення та опалення побутового приміщення

Керівник кваліфікаційної роботи

Донченко В.Ю. старший викладач

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджена наказом по університету

ВІД

2. Строк подання студентом проекту (роботи)

3. Вихідні дані до роботи (проекту)

Розроблено електричну схему

для «розумного» дому на основі емулятора на базі Arduino, а саме на

контролері ESP32.

(визначаються кількісні або (та) якісні показники, яким повинен відповідати об'єкт розробки)

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) НАУКОВО-ДОСЛІДНА ЧАСТИНА. РОЗРОБКА ПРИНЦИПОВОЇ СХЕМИ ПРИСТРОЮ.

(визначаються назви розділів або (та) перелік питань, які повинні увійти до тексту ПЗ)

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів проекту/роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання „_____” _____ 2025р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
	Вибір теми роботи, вивчення наукової літератури, затвердження теми та керівника.	15 жовтня 2025	
	Аналіз літературних джерел за темою роботи. Розробка та апробація методики дослідно-експериментальної роботи. Подання структури теоретичної частини роботи та плану експериментальних досліджень.	Другий тиждень листопада (10 листопада)	
	Робота над теоретичною частиною. Подання теоретичної частини роботи для першого читання науковим керівником.	До 25 грудня	
	Усунення зауважень, урахування рекомендацій наукового керівника. Подання теоретичної частини роботи на друге читання.	Третій тиждень січня	
	Проведення експериментальної роботи. Поетапний аналіз та обговорення її результатів. Перевірка стану виконання роботи.	Перший тиждень березня	
	Урахування рекомендацій наукового керівника, усунення недоліків, підготовка варіанта роботи до передзахисту. Розробка презентації.	До 31 березня	
	Попередній захист роботи на кафедрі	квітень	
	Доопрацювання роботи з урахуванням рекомендацій після передзахисту. Подання роботи науковому керівникові та рецензентові на підготовку відгуку та рецензії	За 10 днів до державної атестації	
	Подання на кафедру остаточного варіанта роботи, переплетеного та підписаного автором, науковим керівником і рецензентом.	За 5 днів до державної атестації	

Студент


підпис

С. В. Возняк

(ініціали, прізвище)

Керівник проекту (роботи)

підпис

В. Ю. Донченко

(ініціали, прізвище)

АНОТАЦІЯ

Возняк С. В.

Тема: Автоматизована система на платформі Arduino керування освітлення і опалення побутового приміщення.

Спеціальність: 123 «Комп'ютерна інженерія»

Установа: ЛНУ імені Тараса Шевченка, 2026 р.

Бакалаврська робота містить: 73 с., 28 рис., 6 табл., 17 джерел.

Об'єкт дослідження - процес управління системою «розумного дому» на прикладі побутового приміщення.

Предмет дослідження - управління «розумним домом» з автономним середовищем на основі Arduino IDE.

Мета дослідження - підвищення ефективності енерговитрат в побутових приміщеннях за допомогою мікроконтролерної системи на основі платформи Arduino.

Для вирішення поставлених задач використовувались методи моделювання та програмування; положення теорії автоматичного керування; методи створення автоматизованих вимірювальних мікропроцесорних систем;

Результати роботи.

У результаті проведених досліджень були розроблені алгоритми для функціонування автоматичного підтримання заданого рівня температури в приміщенні, задання світлових режимів та інтерфейс за допомогою IOT Blynk, наведені рисунки з елементами коду з поясненнями. Створено електричну схему в Web-середовищі на базі Arduino IDE.

Розроблено принципіальну електричну схему для «розумного» акваріуму на основі Arduino, а саме на контролері ESP32_webkit_4.

За допомогою безкоштовної Web-платформи на базі Arduino IDE, було створено схему та написано програмне забезпечення для функціонування системи керування світлом та опаленням.

Ключові слова. Blynk, ESP32, Arduino, IOT, Sensor.

ABSTRACT

Vozniak Serhii

Theme: An Arduino-based automated system for controlling lighting and heating in a residential space.

Speciality: 123 "Computer Engineering"

Institution: Luhansk Taras Shevchenko National University (LTSNU), 2026.

Diploma work contains: 73 pages, 28 Fig., 6 Table, 17 source.

A research object is the process of managing a "smart home" system, using a residential space as an example.

The subject of this study is the control of a "smart home" with an autonomous environment based on the Arduino IDE.

An aim of work is Improving energy efficiency in residential spaces using a microcontroller-based system built on the Arduino platform.

To solve the tasks, modeling and programming methods were used; the principles of the theory of automatic control; methods for creating automated measuring microprocessor systems;

Job performances.

As a result of the research, algorithms were developed for automatically maintaining a set temperature in a room and configuring lighting modes, along with an interface using the Blynk IoT platform; the figures below show code snippets with explanations. An electrical schematic was created in a web-based environment using the Arduino IDE.

A basic electrical schematic has been developed for a "smart" aquarium based on Arduino, specifically using the ESP32_webkit_4 controller.

Using a free web platform based on the Arduino IDE, a schematic was created and software was written to operate the lighting and heating control system.

Keywords. Blynk, ESP32, Arduino, IOT, Sensor.

ІТС.4КІ.0126.01-ВП
АВТОМАТИЗОВАНА СИСТЕМА НА ПЛАТФОРМІ ARDUINO
КЕРУВАННЯ ОСВІТЛЕННЯМ І ОПАЛЕННЯМ ПОБУТОВОГО
ПРИМІЩЕННЯ

[illegible]

					ІТС.4КІ.0126.01-ВП								
Змн.	Арк.	№ докум.	Підпис	Дата									
Розроб.		Возняк С. В.			ВІДОМІСТЬ ПРОЄКТУ				Літ.	Арк.	Акрушів		
Керівник		Донченко В.Ю.									1	1	
Реценз.		Козуб Ю.Г.							ЛНУ Кафедра ІТС, Гр.4КІ				
Н. Контр.													
Зав. каф.		Семенов М.А.											

Міністерство освіти і науки України
Державний заклад «Луганський національний університет
імені Тараса Шевченка»

Факультет (інститут)

Факультет технологій та інформаційних
систем

(повна назва)

Кафедра

Інформаційних технологій та систем

(повна назва)

(код, назва)

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання програмної розробки (ПР):

"АВТОМАТИЗОВАНА СИСТЕМА ПЛАТФОРМИ ARDUINO
КЕРУВАННЯ ОСВІТЛЕННЯ ТА ОПАЛЕННЯ ПОБУТОВОГО
ПРИМІЩЕННЯ"

ІТС.4КІ.0126.02-ТЗ

ПОГОДЖЕНО

Керівник кваліфікаційної роботи

Донченко В.Ю.

“ ” 2026р

ВИКОНАВЕЦЬ

Студент групи 4КІ

Возняк С. В.

“ ” 2026р

ЗМІСТ

ВСТУП.....	3
1. ПРИЗНАЧЕННЯ ПРОДУКЦІЇ.....	3
2. ТЕХНІЧНІ ХАРАКТЕРИСТИКИ	4
3. АЛГОРИТМ РОБОТИ ПРИСТРОЮ	5
4. ТЕХНІКО-ЕКОНОМІЧНІ ВИМОГИ ДО КІНЦЕВОГО ПРОДУКТУ	6
5. ВИМОГИ ДО МАТЕРІАЛІВ ТА КОМПЛЕКТУЮЧИХ	6
6. ЕТАПИ ВИКОНАННЯ ПР.....	6
7. ПРИЙМАННЯ	7
8. ПОРЯДОК ВНЕСЕННЯ ЗМІН ДО ТЕХНІЧНОГО ЗАВДАННЯ, ЩО ЗАТВЕРДЖЕНО.....	8

					ІТС.4КІ.0126.02-ТЗ					
Змн.	Арк.	№ докум.	Підпис	Дата	ТЕХНІЧНЕ ЗАВДАННЯ			Літ.	Арк.	Акрушів
Розроб.		Возняк С. В.								
Керівник		Донченко В.Ю.							2	7
Реценз.		Козуб Ю.Г.						ЛНУ Кафедра ІТС, Гр.4КІ		
Н. Контр.										
Зав. каф.		Семенов М.А..								

ВСТУП

- 1.1. **Найменування:** Автоматизована система на платформі Arduino керування освітлення та опалення побутового приміщення.
- 1.2. **Шифр ПР:** АДР-5
- 1.3. **Підстава до виконання ПР:** Підставою для виконання даної розробки є завдання на дипломний проєкт.
- 1.4. **Терміни розробки:**
 - 1.4.1. Початок 1 жовтня 2025р.
 - 1.4.2. Закінчення 20 квітня 2026р.
- 1.5. **Фінансується** за рахунок коштів замовника.

1. ПРИЗНАЧЕННЯ ПРОДУКЦІЇ

Цей пристрій призначений для вимірювання, моніторингу та налаштування таких параметрів навколишнього середовища:

- температура;
- освітленість;
- наявність руху;

Система створюється з метою:

- відображення виміряних параметрів в реальному часі на дисплеї пристрою;
- бездротової передачі даних через Wi-Fi на смартфон користувача за допомогою додатку Blynk;

					ІТС.4КІ.0126.02-ТЗ	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдання, що вирішуються:

- визначення апаратної платформи та необхідних датчиків;
- розробка схеми підключення компонентів;
- написання програмного забезпечення для мікроконтролера ESP32;
- реалізація передачі даних на смартфон через хмарний сервіс Blynk;
- проведення тестування;
- розробка технічної документації.

2. ТЕХНИЧНІ ХАРАКТЕРИСТИКИ

Розроблено систему на базі мікроконтролера ESP32. Програмне забезпечення написано на базі середовища Arduino IDE. Пристрій забезпечує одночасне зчитування даних з датчиків, їх відображення на дисплеї та передачу через Wi-Fi до мобільного додатку Blynk.

Перелік компонентів та їх призначення наведено в таблиці нижче.

Номер	Компонент	Призначення у схемі
1	ESP32 DevKit-C V4	Головний мікроконтролер, що зчитує дані з датчиків, обробляє логіку та керує Wi-Fi зв'язком.
2	DHT22	Цифровий датчик для точного вимірювання температури та вологості повітря.
3	PIR Motion Sensor	Інфрачервоний датчик для фіксації руху людини в зоні моніторингу.
4	LDR Sensor Module (ldr1)	Чотирипіновий модуль фоторезистора для вимірювання рівня світла через аналоговий вихід АО.
5	LCD1602 (I2C)	Дисплей для локального виводу інформації про стан системи та показники датчиків.
6	Relay Module (Heater)	Електромагнітне реле для дистанційного керування живленням обігрівача.
7	Relay Module (Light)	Електромагнітне реле для автоматичного ввімкнення та вимкнення освітлення.

					ІТС.4КІ.0126.02-ТЗ	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

Номер	Компонент	Призначення у схемі
8	Blue LED (Heat)	Світлодіодний індикатор синього кольору для візуалізації роботи режиму опалення.
9	Yellow LED (Light)	Світлодіодний індикатор жовтого кольору для візуалізації роботи ламп освітлення.
10	Resistor (res1)	Обмежувальний резистор на 220 Ом для безпечного підключення синього світлодіода.
11	Resistor (res2)	Обмежувальний резистор на 220 Ом для безпечного підключення жовтого світлодіода.

3. АЛГОРИТМ РОБОТИ ПРИСТРОЮ

Алгоритм системи керування базується на циклічному опитуванні пристроїв та автоматичному порівнянні отриманих даних із заданими вручну константами. Для керування мікрокліматом контролер кожні дві секунди зчитує показники температури з датчика DHT22 і, у разі їх падіння нижче встановленого порогу, активує реле обігрівача через цифровий вихід GPIO 2.

Паралельно реалізована логіка інтелектуального освітлення, яка використовує датчик руху PIR та фоторезистор для аналізу навколишнього середовища. Система фіксує момент останньої активності людини та підтримує живлення ламп на піні 4 протягом заданої кількості секунд, проте автоматично блокує ввімкнення світла, якщо рівень природної освітленості перевищує межу в 2000 одиниць. Усі поточні параметри, включаючи статус руху та фактичні значення сенсорів, у реальному часі виводяться на LCD-дисплей для локального моніторингу стану об'єкта.

					ІТС.4КІ.0126.02-ТЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

Передає усі шість параметрів на хмарний сервіс Blynk через Wi-Fi кожні 2-5 секунд. Користувач переглядає дані у мобільному додатку Blynk у режимі реального часу.

4. ТЕХНІКО-ЕКОНОМІЧНІ ВИМОГИ ДО КІНЦЕВОГО ПРОДУКТУ

Вартість робіт по розробці даної ПР визначається згідно з договором на розробку. Собівартість апаратної частини пристрою повинна бути нижчою за вартість комерційних аналогів із аналогічним функціоналом, що забезпечує економічну доцільність використання розробленого рішення.

5. ВИМОГИ ДО МАТЕРІАЛІВ ТА КОМПЛЕКТУЮЧИХ

5.1. Вимоги до екологічної безпечності під час експлуатації.

Не пред'являються.

5.2. Спеціальні вимоги до кінцевого продукту.

Не пред'являються.

5.3. Вимоги до безпеки для населення під час експлуатації продукції.

Не пред'являються.

6. ЕТАПИ ВИКОНАННЯ ПР.

Етапи виконання ПР можуть уточнюватися згідно з календарним планом робіт за погодженням між замовником та виконавцем.

№	Етапи виконання роботи	Термін виконання та обсяг робіт	Звітні матеріали
1	Аналіз вимог. Вибір апаратної платформи та компонентів. Розробка структурної та принципової схеми пристрою. Попереднє тестування.		Структурна схема, схема підключення, частковий програмний комплекс та звітна документація.
2	Розробка програмного забезпечення. Реалізація передачі даних через Blynk. Розробка кінцевої версії ПЗ. Тестування.		Готовий програмний комплекс та звітна документація.

№	Етапи виконання роботи	Термін виконання та обсяг робіт	Звітні матеріали
3	Доопрацювання окремих модулів. Оформлення пояснювальної записки, технічного завдання та іншої документації.		Звітні матеріали згідно з пунктом 7 цього ТЗ.

7. ПРИЙМАННЯ

7.1. Необхідні вимоги для впровадження ПР та завершення робіт.

Оцінка результатів розробки і доцільність її продовження здійснюється замовником по поданні наступних матеріалів:

- завантажений програмний комплекс на мікроконтролері замовника;
- перелік файлів на резервному носії;
- стислий опис роботи ПР та опис усіх файлів, необхідних для роботи ПР;
- перелік документів:
 - Технічне завдання;
 - Пояснювальна записка.

7.2. Перелік звітних документів, необхідних для прийняття етапів роботи:

- стислий опис результатів етапу у вигляді анотованого звіту (для 1-го та 2-го етапів);
- частковий або повний програмний комплекс на мікроконтролері замовника згідно з календарним планом робіт;
- акт приймання продукції.

7.3. Загальний перелік до приймання звітних документів, макетів, експериментальних зразків.

До приймання пред'являються: акт здачі-приймання продукції, акт впровадження ПР.

					ІТС.4КІ.0126.02-ТЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

7.4. Тестування ПР

Тестування виконується згідно з «Програмою та методикою тестування», яка розробляється виконавцем та затверджується замовником.

8. ПОРЯДОК ВНЕСЕННЯ ЗМІН ДО ТЕХНІЧНОГО ЗАВДАННЯ, ЩО ЗАТВЕРДЖЕНО.

Дане технічне завдання може уточнюватися в процесі розробки ПР при узгодженні сторін з оформленням доповнень до ТЗ.

					<i>ІТС.4КІ.0126.02-ТЗ</i>	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЗ «ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА»

Факультет технологій та інформаційних систем

(назва факультету, інституту)

Інформаційних технологій та систем

(назва кафедри)

Пояснювальна записка
до дипломного проекту (роботи)
БАКАЛАВРА
(освітньо-кваліфікаційний рівень)

на тему:

АВТОМАТИЗОВАНА СИСТЕМА НА ПЛАТФОРМІ ARDUINO
КЕРУВАННЯ ОСВІТЛЕННЯ ТА ОПАЛЕННЯ ПОБУТОВОГО
ПРИМІЩЕННЯ

Виконав: студент 4 курсу, групи ____
напряму підготовки (спеціальності)
123 «Комп'ютерна інженерія»
(шифр і назва напряму підготовки, спеціальності)

Возняк С. В.
(прізвище та ініціали)

Керівник Донченко В. Ю.
(прізвище та ініціали)

Рецензент Козуб Ю. Г.
(прізвище та ініціали)

Лубни – 2026 року

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ СИСТЕМ АВТОМАТИЗАЦІЇ КЕРУВАННЯ ПРИМІЩЕННЯМИ.....	6
1.1. Сучасні системи керуванням світлом та опаленням	6
1.2. Порівняння контролерів для побудови систем керування освітленням та опаленням і обґрунтування вибору платформи Arduino	9
Висновки до розділу	14
РОЗДІЛ 2. ПРОЕКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ КЕРУВАННЯ ОСВІТЛЕННЯМ ТА ОПАЛЕННЯМ	16
2.1. Постановка задачі та опис функцій системи.....	16
2.2. Алгоритм роботи системи	17
2.3. Вибір та характеристика основних компонентів системи	22
2.4 Структурна схема системи	29
Висновки до розділу	31
РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ	33
3.1. Структура програмного забезпечення	33
3.2. Проектування схеми в Arduino IDE	36
3.3. Алгоритм автоматичного керування освітленням і опаленням	42
3.4. Розробка користувацького інтерфейсу (Bluetooth / веб-керування).....	44
3.5. Тестування та результати роботи системи	50
Висновки до розділу	53
ВИСНОВКИ.....	55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	57
ДОДАТОК А. Код Sketch.ino	59
ДОДАТОК В. Код diagram.json	64
ДОДАТОК С. libraries.txt.....	69

					<i>ІТС.4КІ.0126.02-ПЗ</i>		
Змн.	Арк.	№ докум.	Підпис	Дата	<i>ЗМІСТ</i>		
Розроб.		Возняк С.В.					
Керівник		Донченко В.Ю.					
Реценз.		Козуб Ю.Г.					
Н. Контр.							
Зав. каф.		Семенов М.А.			ЛНУ		
					Кафедра ІТС, Гр.4КІ		

ВСТУП

Сучасні тенденції розвитку інформаційних технологій, цифрової автоматизації та мікроконтролерних систем відкривають нові можливості для підвищення комфорту та енергоефективності побутових приміщень. Зростання вартості енергоресурсів, потреба в раціональному використанні електроенергії та тепла, а також вимоги до безпеки життєвого середовища обумовлюють актуальність розробки розумних систем керування. Одним із найбільш доступних і гнучких рішень у сфері домашньої автоматизації є системи, побудовані на платформі Arduino, яка завдяки простоті програмування, широкій підтримці сенсорів і модулів та невисокій вартості дозволяє створювати функціональні та масштабовані проекти.

Особливої уваги потребують невеликі приватні будинки та квартири, де відсутні дорогі комерційні рішення з управління температурою та освітленням, але існує потреба в підвищенні енергетичної ефективності та комфорту. Типовий український одноповерховий двокімнатний будинок, що має вісім вікон і складається з вітальні, спальні, кухні, коридору та ванної кімнати, становить реалістичну модель середньостатистичного житла, де автоматизація дозволяє значно зменшити витрати на електроенергію та опалення.

У роботі розглядається задача створення комплексної автоматизованої системи, яка на основі даних температурних датчиків, датчиків освітленості та датчиків присутності здатна автономно керувати освітленням і опаленням будинку, а також забезпечувати дистанційний контроль через Wi-Fi. Система має реагувати на зміну зовнішніх умов, оптимізувати роботу освітлювальних приладів і обігрівачів, підтримувати комфортний мікроклімат та мінімізувати споживання енергії.

					ІТС.4КІ.0126.02-ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	ВСТУП	Лім.	Арк.	Акрушів
Розроб.		Возняк С.В.						
Керівник		Донченко В.Ю.					3	4
Реценз.		Козуб Ю.Г.				ЛНУ Кафедра ІТС, Гр.4КІ		
Н. Контр.								
Зав. каф.		Семенов М.А.						

Використання Arduino як центрального контролера забезпечує можливість швидкої інтеграції, модифікації та подальшого розвитку системи відповідно до потреб користувача.

Запропонований комплекс автоматизації вирізняється доступністю, простотою реалізації та можливістю встановлення у звичайних побутових умовах без значних фінансових затрат. Актуальність дослідження зумовлена також тим, що впровадження розумних систем є важливою складовою розвитку енергоефективного житла та шляхом до зменшення залежності від традиційних систем керування.

Таким чином, створення автономної системи керування світлом і температурою на базі Arduino є не лише технічним завданням, а й практично важливим інструментом підвищення якості життя, рівня комфорту та енергоефективності житлових будівель. Саме це визначає необхідність детального дослідження принципів побудови, алгоритмів роботи та технічних рішень для впровадження системи у реальних умовах.

Автоматизація побутових процесів є одним із ключових напрямів розвитку сучасних технологій. У контексті підвищення вартості енергоресурсів та зростаючої потреби в енергоефективності системи розумного керування побутовими приладами набувають особливої ваги. В Україні значна частина житлового фонду представлена невеликими приватними будинками та квартирами, які не обладнані сучасними системами автоматизації. Розробка доступної системи на базі Arduino дає змогу підвищити енергозбереження, комфорт і безпеку житла, не потребуючи значних фінансових інвестицій. Таким чином, тема роботи відповідає сучасним тенденціям розвитку "Smart Home" та є соціально й економічно значущою.

					ІТС.4КІ.0126.02-ПЗ	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

Оскільки, метою роботи є розроблення та практична реалізація автоматизованої системи керування освітленням та опаленням у типовому одноповерховому двокімнатному будинку, заснованої на платформі Arduino. Для досягнення мети необхідно виконати такі завдання:

- Проаналізувати існуючі рішення комерційних та DIY систем “розумного дому”;
- Визначити функціональні вимоги системи;
- Розробити структурну схему та алгоритм роботи системи;
- Підібрати оптимальні апаратні компоненти (Arduino IDE, датчики, модулі реле, Wi-Fi тощо);
- Створити програмне забезпечення для збору даних, обробки сигналів та керування;
- Реалізувати систему та протестувати її в умовах реального або модельованого середовища;
- Оцінити ефективність роботи та можливості масштабування;

Під час виконання проекту у роботі було використано методи:

- Аналітичний - для дослідження існуючих систем автоматизації;
- Порівняльний - для вибору технічних компонентів;
- Методи моделювання - для тестування логіки роботи системи;
- Інструментальні методи - для вимірювання параметрів мікроклімату та навантажень.

Практичне значення роботи полягає у створенні доступної та функціональної системи автоматизації, яка може бути встановлена у звичайному побутовому будинку без необхідності залучення дорогих комерційних рішень. Розроблений комплекс дозволяє зменшити споживання електроенергії, забезпечити комфортне освітлення та підтримання температури, а також підвищити рівень безпеки житла. Система може бути масштабована, доповнена новими датчиками або інтегрована в ширші IoT-архітектури.

					ІТС.4КІ.0126.02-ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ СИСТЕМ АВТОМАТИЗАЦІЇ КЕРУВАННЯ ПРИМІЩЕННЯМИ

1.1. Сучасні системи керуванням світлом та опаленням

Системи керування освітленням та опаленням є важливою частиною сучасних автоматизованих систем житлових приміщень. Вони забезпечують централізований контроль роботи освітлювальних приладів та систем обігріву, що дозволяє підвищити рівень комфорту проживання та оптимізувати використання енергоресурсів. Основою таких систем є контролер, який взаємодіє з датчиками, виконавчими пристроями та засобами керування.

Автоматизовані системи освітлення дозволяють регулювати рівень освітленості приміщень залежно від умов навколишнього середовища. Для цього використовуються датчики освітленості, які вимірюють інтенсивність природного світла. Отримані дані передаються до контролера, який приймає рішення щодо вмикання або вимикання освітлювальних приладів. Завдяки цьому забезпечується ефективно використання електроенергії та підтримується необхідний рівень освітлення в приміщенні.

У сучасних системах освітлення також широко застосовуються датчики руху. Вони дозволяють автоматично вмикати світло при появі людини у приміщенні та вимикати його після завершення активності. Такий підхід значно знижує споживання електроенергії та підвищує зручність користування освітлювальними приладами. Особливо ефективним є використання подібних систем у коридорах, сходових клітинах та допоміжних приміщеннях, але додають зручності та комфорту і в житлових кімнатах.

					ІТС.4КІ.0126.02-ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	РОЗДІЛ І. АНАЛІЗ ІСНУЮЧИХ СИСТЕМ АВТОМАТИЗАЦІЇ КЕРУВАННЯ ПРИМІЩЕННЯМИ	Літ.	Арк.	Акрушів
Розроб.	Возняк С.В.						7	10
Керівник	Донченко В.Ю.					ЛНУ Кафедра ІТС, Гр.4КІ		
Реценз.	Козуб Ю.Г.							
Н. Контр.								
Зав. каф.	Семенов М.А.							

Системи керування опаленням призначені для підтримання комфортного температурного режиму у житлових приміщеннях. Основними елементами таких систем є температурні датчики, контролер та виконавчі пристрої, які керують роботою нагрівальних елементів або систем опалення. Температурні датчики постійно вимірюють температуру повітря у приміщенні та передають ці дані до контролера.

На основі отриманих даних контролер визначає необхідність увімкнення або вимкнення системи опалення. Якщо температура знижується нижче заданого рівня, система автоматично активує нагрівальні елементи. У разі досягнення встановленої температури опалення вимикається. Такий принцип роботи дозволяє підтримувати стабільний мікроклімат у приміщенні та уникати надмірного використання енергії.

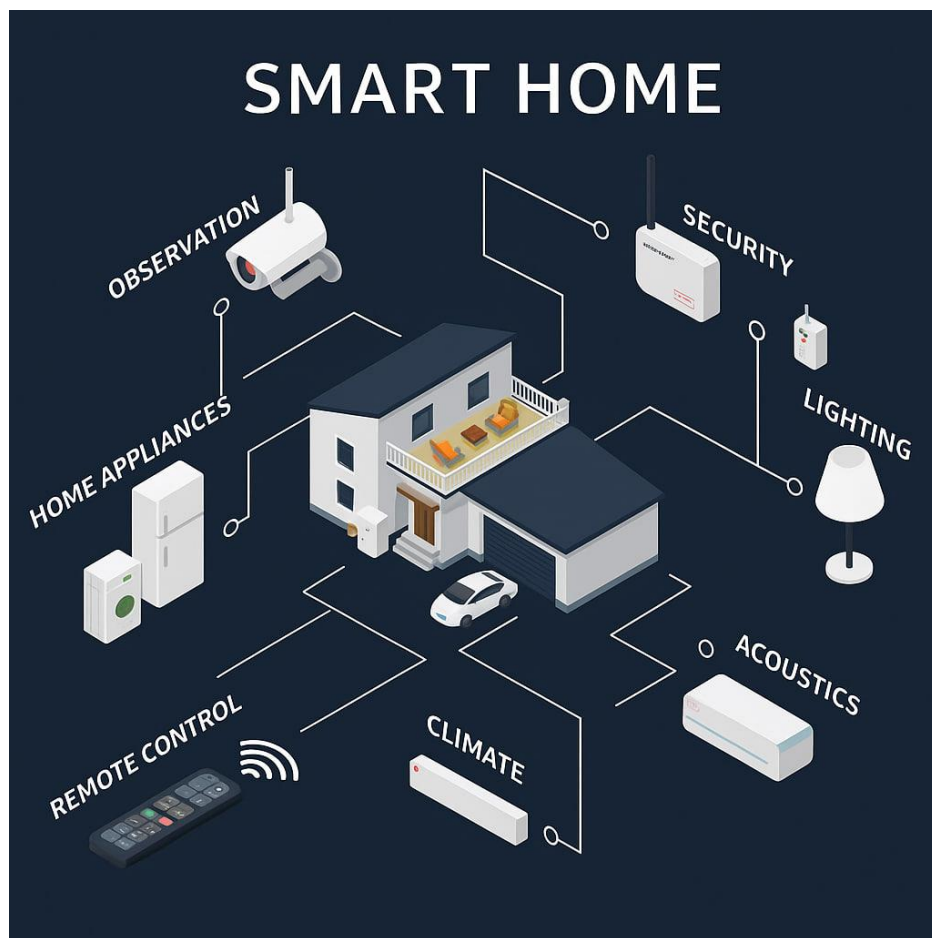


Рис. 1.1. Приклад функціоналу розумної системи

Додатковою перевагою сучасних систем є можливість програмування температурних режимів. Користувач може задавати різні температурні значення для різних періодів доби або окремих приміщень. Наприклад, у нічний час температура може бути знижена, а перед поверненням мешканців додому система автоматично підвищує її до комфортного рівня.

Поєднання систем керування освітленням та опаленням дозволяє створити ефективну автоматизовану систему управління житловим приміщенням. Використання датчиків, контролерів та виконавчих пристроїв забезпечує автоматичне реагування на зміни умов середовища та потреб користувачів. У результаті підвищується енергоефективність будівлі, зменшуються витрати електроенергії та створюються комфортні умови проживання.

Отже, сучасні системи керування освітленням та опаленням є важливим напрямом розвитку автоматизації житлових будівель. Вони дозволяють реалізувати ефективний контроль енергоспоживання, автоматизувати процеси підтримання освітлення та температури, а також підвищити загальний рівень комфорту та безпеки у приміщенні.

Тепер детальніше розглянемо переваги та недоліки розумних систем.

Таблиця 1.1.

Огляд розумних систем

Переваги	Недоліки
1. Енергозбереження - автоматичне вимкнення освітлення, оптимізація опалення зменшує споживання електроенергії та тепла.	1. Висока початкова вартість - встановлення обладнання та налаштування може бути дорогим, залежно від призначення та масштабування.
2. Зручність користування - можливість дистанційного керування	2. Складність налаштування - вимагає технічних знань для

Переваги	Недоліки
освітленням, температурою, технікою через смартфон чи комп'ютер.	інтеграції різних пристроїв і протоколів.
3. Підвищення безпеки - контроль доступу, оповіщення про несправність чи проблему з температурами.	3. Залежність від інтернету - при відсутності мережі деякі функції можуть бути недоступні.
4. Комфорт і автоматизація - система самостійно підтримує оптимальні умови в приміщенні.	4. Можливі збої або конфлікти пристроїв - при оновленнях чи збоях у мережі.
5. Гнучкість і масштабованість - можна додавати нові модулі (датчики, реле, контролери) без повної заміни системи.	5. Ризики безпеки даних - при некоректному налаштуванні існує ймовірність несанкціонованого доступу.
6. Моніторинг у реальному часі - користувач може переглядати стан приміщення будь-коли.	6. Залежність від електроживлення - у разі відключення електрики система може стати неактивною.
7. Підвищення вартості нерухомості - будинки з «розумними» системами мають більшу привабливість.	7. Потреба в регулярному обслуговуванні - оновлення програмного забезпечення, заміна датчиків.

1.2. Порівняння контролерів для побудови систем керування освітленням та опаленням і обґрунтування вибору платформи Arduino

Сучасні системи автоматичного керування освітленням та опаленням будуються на базі мікроконтролерних платформ, які виконують функції центрального контролера. Такі контролери забезпечують обробку сигналів від

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

датчиків температури, освітленості та руху, а також керують виконавчими пристроями, зокрема реле або електронними ключами. Вибір апаратної платформи є важливим етапом проєктування системи, оскільки від характеристик контролера залежить стабільність роботи, можливості розширення та складність реалізації алгоритмів керування.

Однією з відомих платформ для створення систем автоматизації є Arduino. Дана платформа базується на мікроконтролерах сімейства AVR та використовує власне програмне середовище Arduino IDE. Основною перевагою Arduino є відкритість архітектури та простота програмування. Проте обмежена обчислювальна потужність та відсутність вбудованих засобів мережевої взаємодії роблять її менш ефективною для сучасних концепцій "розумного будинку", де ключовою вимогою є дистанційний моніторинг та керування через інтернет.

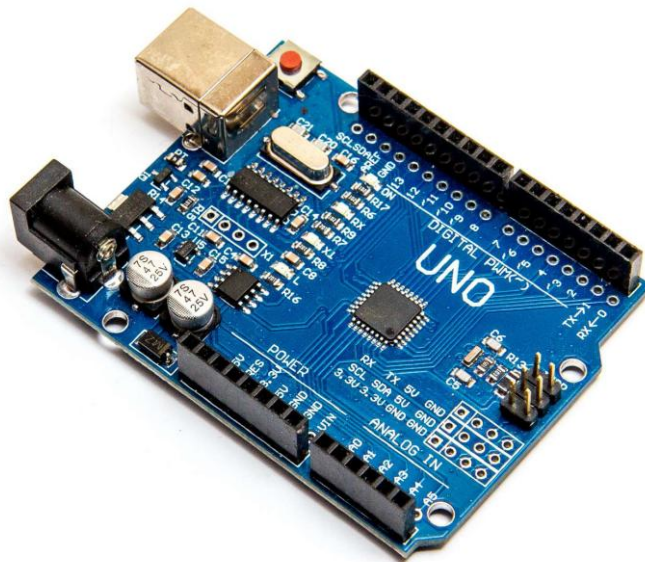


Рис. 1.2. Мікроконтролер Arduino

Найбільш перспективним рішенням для побудови подібних систем є використання контролерів на базі платформи ESP32. Ці мікроконтролери

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

мають значно більшу обчислювальну потужність, ніж класичні плати Arduino, а також вбудовані модулі бездротового зв'язку Wi-Fi та Bluetooth. Працює мікроконтролер на C/C++, в сукупності, це дозволяє створювати високоефективні системи керування, які передають дані до хмарних сервісів або взаємодіють з мобільними пристроями в режимі реального часу. Крім того, ESP32 має великий обсяг оперативної пам'яті та значну кількість апаратних інтерфейсів (включаючи велику кількість АЦП та сенсорних пінів) для підключення датчиків і периферійних модулів.

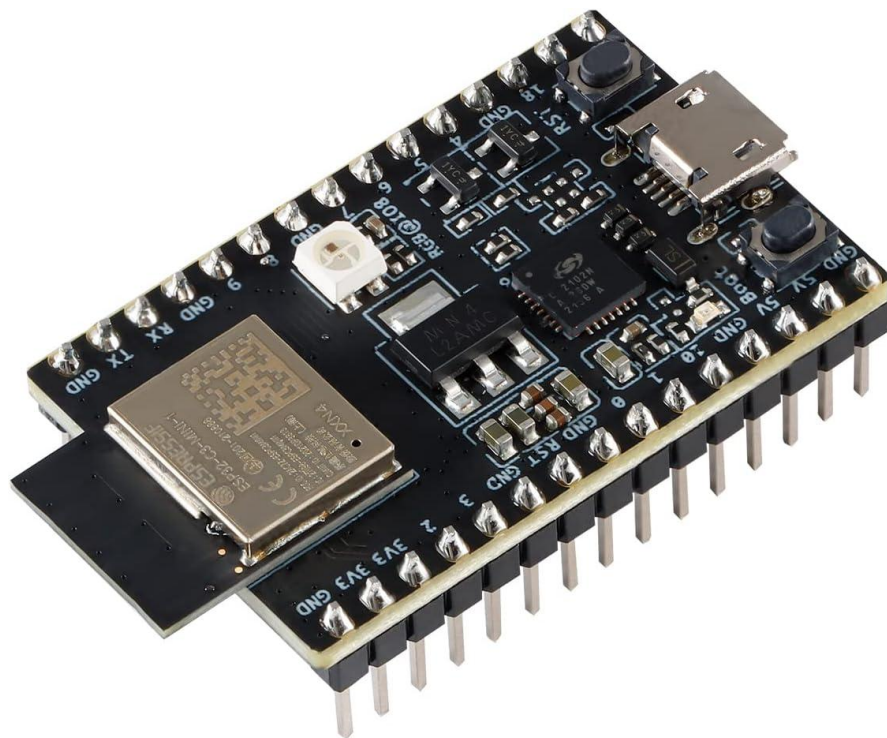


Рис. 1.3. Мікроконтролер ESP32

Ще одним поширеним рішенням є використання мікроконтролерів ESP8266. Ці контролери також мають вбудований модуль бездротового зв'язку та можуть використовуватися для створення недорогих систем автоматизації. Проте кількість доступних входів і виходів у таких контролерах є обмеженою, а наявність лише одного ядра знижує стабільність роботи при одночасному виконанні алгоритмів керування та підтримці мережевого з'єднання.

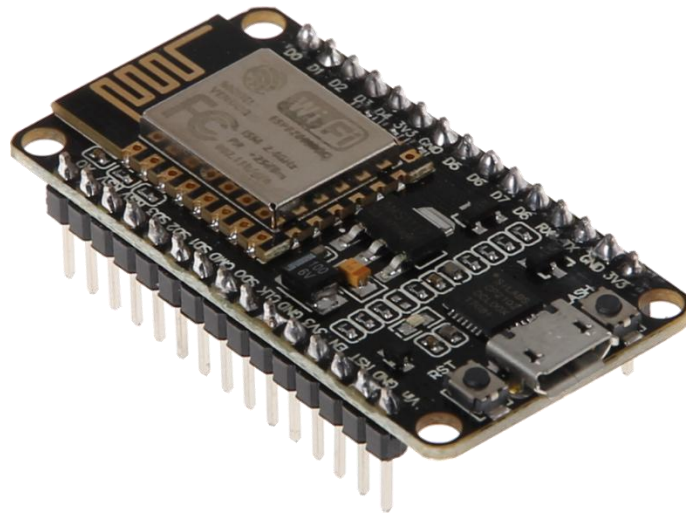


Рис. 1.4. Мікроконтролер ESP8266

Порівнюючи наведені платформи, можна зазначити, що саме контролер ESP32 забезпечує найкращий баланс між продуктивністю та функціональністю. Він ідеально підходить для складних систем, які потребують постійного підключення до мережі та швидкої обробки великих масивів даних від багатьох датчиків. На відміну від базових плат Arduino, ESP32 дозволяє реалізувати повноцінний IoT-функціонал без використання додаткових громіздких модулів зв'язку.

У сучасних умовах платформа ESP32 має ряд критичних переваг. Вона дозволяє створювати системи керування, які поєднують автономну логіку з можливостями глобального керування через веб-інтерфейси. Велика кількість входів і виходів ESP32 є достатньою для підключення датчиків температури, освітленості, руху та багатоканальних модулів реле. Завдяки наявності двох ядер, одне з яких може бути виділене суто під завдання зв'язку, забезпечується висока надійність функціонування системи без "зависань" основного алгоритму.

Важливою перевагою ESP32 є підтримка сучасних стандартів безпеки передачі даних, що є критичним для систем домашньої автоматизації. Висока тактова частота дозволяє використовувати складніші методи обробки сигналів, наприклад, цифрову фільтрацію даних з датчиків для усунення

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

хибних спрацьовувань. Це забезпечує точніше регулювання температури та ввімкнення освітлення лише за необхідності, що веде до суттєвої економії енергоресурсів.

Таким чином, аналіз сучасних мікроконтролерних платформ показує, що для побудови систем керування освітленням та опаленням найбільш доцільним є використання контролера ESP32. Він поєднує в собі високу продуктивність, вбудовані мережеві інтерфейси та широкі можливості розширення. Саме тому платформа ESP32 є оптимальним вибором для реалізації автоматизованої системи керування освітленням та обігрівом у даному проєкті.

Таблиця 1.2.

Порівняння мікроконтролерів

Контролер	Коротка характеристика	Переваги	Недоліки
Arduino Uno	Мікроконтролер ATmega328P, 16 МГц, 14 цифрових входів/виходів, 6 аналогових входів	Простота використання, велика кількість бібліотек, стабільна робота без мережі, доступна ціна, велика спільнота користувачів	Обмежена обчислювальна потужність, невеликий обсяг пам'яті, відсутність вбудованого бездротового зв'язку
ESP8266	Мікроконтролер з вбудованим Wi-Fi, тактова частота до 80 МГц	Вбудований бездротовий зв'язок, більша продуктивність ніж Arduino Uno, можливість	Обмежена кількість входів і виходів, складніша конфігурація, більша

Контролер	Коротка характеристика	Переваги	Недоліки
		створення мережевих систем	чутливість до стабільності живлення
ESP32	Двоядерний мікроконтролер до 240 МГц, Wi Fi та Bluetooth	Висока продуктивність, велика кількість периферійних інтерфейсів, підтримка бездротового зв'язку	Складніш е програмування, вища вартість, надлишкова продуктивність для простих систем

Висновки до розділу

У першому розділі було розглянуто сучасні підходи до побудови систем автоматичного керування освітленням та опаленням у житлових приміщеннях. Встановлено, що автоматизація даних процесів дозволяє підвищити рівень комфорту проживання, забезпечити раціональне використання електроенергії та створити стабільний мікроклімат у приміщеннях. Використання датчиків освітленості, температури та руху дає можливість системі автоматично реагувати на зміни умов навколишнього середовища та відповідно керувати освітлювальними приладами і системами обігріву.

Проведений аналіз показав, що для реалізації подібних систем можуть використовуватися різні мікроконтролерні платформи. Найбільш поширеними серед них є Arduino, ESP8266 та ESP32. Кожна з цих платформ має свої технічні особливості, які визначають сферу їх використання. Проте саме контролери сімейства ESP32 відзначаються найвищою продуктивністю, наявністю вбудованих модулів бездротового зв'язку Wi-Fi та Bluetooth, а

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

також підтримкою сучасних протоколів безпеки, що є критично важливим для інтеграції у сучасні екосистеми "розумного будинку".

Разом з тим для створення систем керування освітленням та опаленням у сучасних проєктах ключовими вимогами є не лише локальна автоматизація, а й можливість дистанційного моніторингу та хмарного керування. У цьому випадку платформа ESP32 має суттєві переваги над застарілими аналогами, оскільки вона забезпечує стабільну мережеву взаємодію без використання додаткових громіздких модулів, має більший обсяг пам'яті для реалізації складних алгоритмів та підтримує багатопотоковість завдяки двоядерній архітектурі.

Також важливою характеристикою платформи ESP32 є велика кількість універсальних портів введення-виведення з розширеним функціоналом. До контролера можуть безпосередньо підключатися цифрові та аналогові датчики температури, освітленості, руху, а також виконавчі реле. Наявність вбудованих сенсорних інтерфейсів та ємнісних каналів дозволяє розширювати можливості системи, створюючи сучасні панелі керування та інтегруючи велику кількість виконавчих елементів без суттєвого ускладнення апаратної частини.

Таким чином, проведений аналіз сучасних рішень для побудови систем автоматизації показав, що для реалізації системи керування освітленням та опаленням доцільно використати мікроконтролерну платформу ESP32. Вона поєднує в собі високу обчислювальну потужність, вбудовані засоби бездротового зв'язку та широку функціональність, що робить її оптимальною основою для розробки сучасної інтелектуальної системи керування освітленням та обігрівом житлового приміщення.

					ІТС.4КІ.0126.02-ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

**РОЗДІЛ 2. ПРОЕКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ
КЕРУВАННЯ ОСВІТЛЕННЯМ ТА ОПАЛЕННЯМ**

2.1. Постановка задачі та опис функцій системи

Метою розробки є створення автоматизованої системи керування освітленням та опаленням житлового приміщення на базі мікроконтролерної платформи ESP32. Система повинна забезпечувати підтримання комфортного рівня освітлення та температури у типовому житловому приміщенні, а також сприяти зменшенню споживання електроенергії та теплових ресурсів завдяки можливостям дистанційного моніторингу через інтернет.

У типовому житловому будинку часто виникають ситуації неефективного використання освітлення та систем опалення. Освітлювальні прилади можуть залишатися увімкненими у приміщеннях, де немає людей, а система опалення може працювати навіть після досягнення необхідного температурного режиму. Це призводить до зайвих витрат енергії та підвищення експлуатаційних витрат.

Основною задачею розроблюваної системи є створення автоматизованого IoT-рішення (Internet of Things), яке забезпечить контроль параметрів освітлення та температури у приміщенні з можливістю автоматичного та дистанційного керування виконавчими пристроями.

Для реалізації поставленої задачі система повинна виконувати низку функцій:

- автоматичне керування освітленням залежно від рівня природного освітлення та наявності людини у приміщенні;
- автоматичне керування системою опалення на основі показників температурних датчиків;

					ІТС.4КІ.0126.02-ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	РОЗДІЛ II. ПРОЕКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ КЕРУВАННЯ ОСВІТЛЕННЯ ТА ОПАЛЕННЯ		
Розроб.		Возняк С.В.					
Керівник		Донченко В.Ю.					
Реценз.		Козуб Ю.Г.					
Н. Контр.							
Зав. каф.		Семенов М.А.					
					Лім.	Арк.	Акрушів
						18	18
					ЛНУ		
					Кафедра ІТС, Гр.4КІ		

- підтримання заданого температурного режиму шляхом вмикання та вимикання нагрівальних елементів;
- автоматичне вимикання освітлення у разі відсутності руху у приміщенні протягом визначеного часу;
- можливість дистанційного керування освітленням та опаленням користувачем через мобільний додаток;
- збір, обробка та передача даних з датчиків температури, освітленості та руху до хмарного сервісу для візуалізації та керування.

Для реалізації зазначених функцій система передбачає використання центрального контролера ESP32, до якого підключаються сенсори та виконавчі модулі. Основними елементами системи є:

- датчики температури для контролю температурного режиму у приміщенні;
- датчики освітленості для визначення рівня природного освітлення;
- датчики руху для визначення присутності людини у кімнаті;
- релейні модулі для керування освітлювальними приладами та елементами системи опалення.

Використання такої структури системи дозволяє забезпечити автоматичне інтелектуальне керування освітленням та опаленням житлового приміщення. Завдяки застосуванню датчиків та сучасного мікроконтролера з вбудованим Wi-Fi модулем, система може самостійно приймати рішення щодо активації ресурсів та інформувати користувача про стан середовища. Це сприяє підвищенню комфорту проживання та суттєвому зменшенню енергоспоживання.

2.2. Алгоритм роботи системи

Алгоритм роботи автоматизованої системи керування освітленням та опаленням на базі платформи ESP32 передбачає послідовну взаємодію між

					ІТС.4КІ.0126.02-ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

апаратними компонентами, хмарним сервісом та мобільним додатком користувача. Програмна частина реалізує логіку обробки даних та прийняття рішень щодо керування приладами. Основною метою системи є поєднання автономності роботи з можливістю повного контролю через мережу інтернет.

1. Ініціалізація системи

Після подачі живлення мікроконтролер ESP32 виконує процедуру ініціалізації, під час якої відбувається підготовка системи до роботи. На цьому етапі виконуються такі дії:

- запуск програмних бібліотек для роботи датчиків, дисплея та стеку Wi-Fi;
- встановлення з'єднання з бездротовою мережею та авторизація на IoT-платформі (Blynk);
- налаштування цифрових та аналогових (ADC) входів і виходів контролера;
- завантаження встановлених користувачем порогових значень температури та освітленості з пам'яті або хмарного сервера.

Після успішного підключення до мережі система переходить до основного циклу роботи.

2. Збір і обробка даних з датчиків

Контролер постійно зчитує дані з усіх підключених сенсорів:

- датчик руху визначає активність у приміщенні;
- датчик освітленості визначає рівень природного світла (з високою точністю завдяки 12-бітному АЦП ESP32);
- датчик температури вимірює поточний тепловий стан повітря.

Отримані значення передаються до алгоритму керування та миттєво відправляються на сервер для відображення у додатку користувача.

					ІТС.4КІ.0126.02-ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Прийняття рішень

Програмна логіка аналізує отримані дані. Основні умови роботи системи такі:

- якщо зафіксовано рух і рівень освітленості нижчий за норму, активується реле освітлення;
- якщо рух не фіксується протягом заданого інтервалу, освітлення вимикається для економії;
- якщо поточна температура нижча за встановлений поріг, вмикається реле опалення;
- якщо температура досягла заданого значення, нагрівання припиняється.

Крім автоматичного режиму, система підтримує пріоритетне ручне керування через фізичні кнопки або віртуальні кнопки у мобільному додатку.

4. Моніторинг та контроль роботи системи

Під час кожного циклу контролер перевіряє стан підключення до мережі Wi-Fi. У разі втрати зв'язку система переходить в автономний режим роботи, продовжуючи керувати пристроями на основі локальних даних датчиків. Стан усіх виконавчих пристроїв відображається на LCD-дисплеї та у інтерфейсі користувача.

5. Енергозбереження та режими роботи

Завдяки енергоефективній архітектурі ESP32, система може оптимізувати власне споживання енергії. Основні методи економії:

- автоматичне вимкнення світла за відсутності людей (Presence Detection);
- точне підтримання температури без перегріву приміщення;

					ІТС.4КІ.0126.02-ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

- можливість віддаленого переведення системи в економний режим під час тривалої відсутності мешканців.

6. Безпека та стабільність роботи

Алгоритм передбачає захист від збоїв:

- перевірка цілісності даних від датчиків (захист від помилкових спрацювань);
- автоматичне вимкнення нагрівачів у разі виходу показників температури за критичні межі;
- використання Watchdog-таймера для автоматичного перезавантаження контролера у разі програмного зависання.

					ІТС.4КІ.0126.02-ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

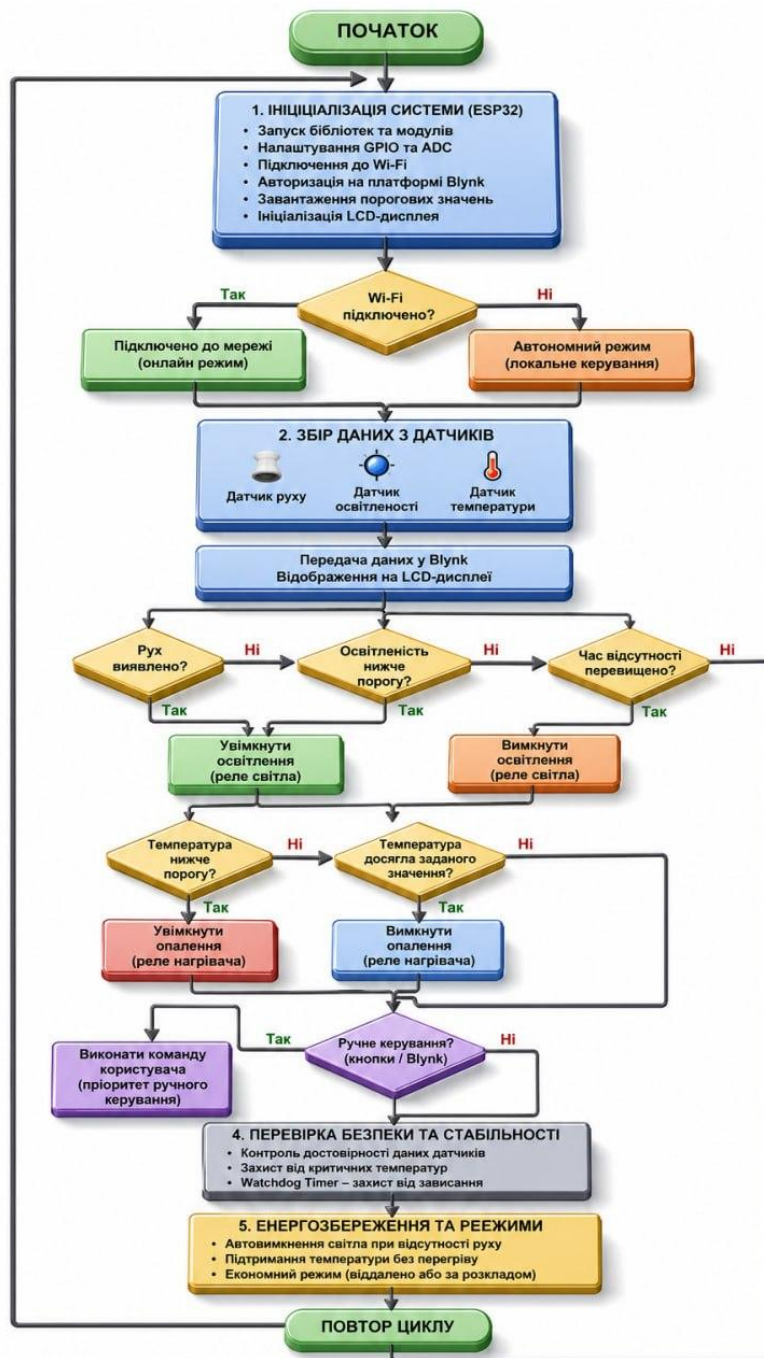


Рис. 2.1. Алгоритм роботи

Отже, алгоритм роботи системи на базі ESP32 забезпечує інтелектуальну взаємодію між середовищем, контролером та користувачем, що гарантує високий рівень комфорту та безпеки при мінімальних витратах ресурсів.

2.3. Вибір та характеристика основних компонентів системи

Для створення системи на базі платформи ESP32 обрано апаратні компоненти, що працюють з логічними рівнями 3.3 В та підтримують високу швидкість обміну даними.

Контролер ESP32

ESP32 – це потужна апаратна платформа з вбудованими модулями Wi-Fi та Bluetooth, яка спеціально розроблена для створення пристроїв інтернет-речей. Вона складається з двоядерного мікропроцесора та розвиненої периферії. На відміну від застарілих рішень, ESP32 пропонує значно вищу тактову частоту (до 240 МГц) та обсяг оперативної пам'яті, що дозволяє виконувати складні задачі без затримок. Основним елементом системи є плата ESP32 DevKit, яка має велику кількість цифрових та аналогових входів/виходів, вбудований роз'єм USB для програмування та стабілізатор напруги. Саме ця модель є оптимальною для побудови сучасних автоматизованих систем керування з можливістю хмарної інтеграції.

ESP32 DevKit - це базова модель для розробки, що має стандартні роз'єми для підключення до макетних плат, вбудований регулятор напруги та інтерфейс USB-UART.

ESP32-S - компактні версії модулів, призначені для інтеграції в готові друковані плати, де велику роль грає саме розмір.

ESP32-WROOM/WROVER - потужні модифікації з додатковою пам'яттю (PSRAM), які мають велику кількість підключень та обчислювальних ресурсів. Ці плати інтегрують у складні проекти з великою кількістю модулів, де потрібна обробка аудіо, зображень або складне криптографічне шифрування даних. Сфера використання цих плат - робототехніка, промислова автоматизація та професійні системи "розумного дому".

Переваги ESP32:

					ІТС.4КІ.0126.02-ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

- понад 20 цифрових входів/виходів, значна кількість аналогових входів (АЦП) - цього більш ніж достатньо для підключення всіх необхідних сенсорів та реле;
- робоча напруга 3.3 В, що є сучасним стандартом для енергоефективних електронних модулів;
- наявність двох ядер, що дозволяє розділяти завдання керування датчиками та підтримку мережевого з'єднання;
- вбудовані модулі Wi-Fi та Bluetooth (Classic + BLE), що виключає потребу у додаткових комунікаційних платах;
- підтримка сучасних середовищ розробки, включаючи Arduino IDE, та сумісність із потужними бібліотеками для IoT.
- ESP32 забезпечує стабільну роботу в умовах домашньої мережі, підтримує режими глибокого сну (Deep Sleep) для економії енергії та дозволяє гнучко змінювати логіку системи через хмарні оновлення. Завдяки функціональності "все в одному", цей контролер є оптимальним варіантом для реалізації повноцінної системи «розумного дому».

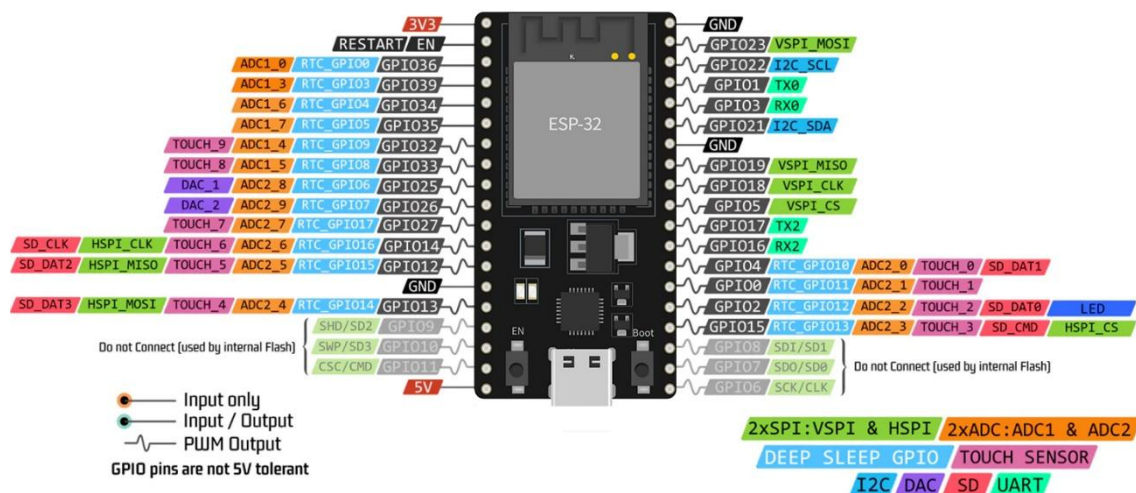


Рис. 2.2. Піни ESP32

Перелік виходів ESP32

Група / Назва піна	Технічні характеристики	Опис та приклади функціонального призначення
3V3	Вихід регулятора (3.3 В)	Живлення малопотужної периферії. Наприклад, датчиків DHT22, LDR або PIR.
5V (Vin)	Вхід/Вихід живлення (5 В)	Пряме живлення від USB або зовнішнього джерела. Наприклад, для модулів реле.
GND	Спільний вивід (0 В)	Спільна точка («земля») для вирівнювання потенціалів усіх компонентів схеми.
GPIO 34, 35, 36, 39	АЦП (ADC1); Тільки вхід	Аналогові входи. Наприклад, для зчитування фоторезисторів (LDR) чи датчиків вологості.
GPIO 32, 33	ADC1; Touch; Crystal Osc	Підключення зовнішнього кварцу або сенсорів. Наприклад, для годинника реального часу (RTC).

GPIO 21 (SDA)	Цифровий I/O; I2C Data	Лінія даних послідовної шини. Наприклад, для підключення дисплеїв LCD 1602.
GPIO 22 (SCL)	Цифровий I/O; I2C Clock	Лінія тактування шини I2C. Наприклад, для синхронізації передачі даних з OLED-екраном.
GPIO 2, 4, 12-15	Цифровий I/O; PWM; ADC2	Універсальні виходи. Наприклад, для керування модулями реле або яскравістю світлодіодів.
GPIO 25, 26	ЦАП (DAC); ADC2; PWM	Генерація аналогового сигналу. Наприклад, для відтворення звуку або керування напругою.
GPIO 27	Цифровий I/O; ADC2; Touch	Універсальний пін. Наприклад, як вхід для ємнісної сенсорної кнопки на корпусі пристрою.
GPIO 0, 2, 5, 12, 15	Strapping Pins	Системні піни. Наприклад, GPIO 0 визначає режим прошивки при

		завантаженні контролера.
GPIO 18, 19, 23	Hardware SPI (SCK, MISO, MOSI)	Швидкісна шина даних. Наприклад, для підключення SD-карт або зовнішніх Ethernet-модулів.
TX / RX (GPIO 1, 3)	Hardware UART	Інтерфейс зв'язку. Наприклад, для прошивки чипа та виводу налагоджувальної інформації.
GPIO 6 – 11	Внутрішня шина Flash-пам'яті	Зарезервовані. Наприклад, для внутрішнього зв'язку з пам'яттю; не для зовнішнього використання.
EN (Reset)	Вхід (Pull-up)	Скидання системи. Наприклад, для підключення фізичної кнопки перезавантаження пристрою.

Датчики температури та освітленості

Для контролю температури обрано датчик DHT22 або DS18B20. DHT22 (модернізована версія DHT11) забезпечує вищу точність та ширший діапазон вимірювання температури і вологості. DS18B20 забезпечує виняткову

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

точність ($\pm 0,5^{\circ}\text{C}$), цифровий вихід та можливість підключення кількох датчиків по шині OneWire, що зручно для контролю температури у різних зонах будинку.

Для вимірювання рівня освітленості використовується фоторезистор (LDR) - аналоговий датчик, який змінює свій опір залежно від інтенсивності світла. ESP32 зчитує ці зміни через високоточний 12-бітний АЦП та визначає необхідність увімкнення штучного освітлення. Цей датчик є недорогим, надійним та забезпечує миттєву реакцію системи на зміну зовнішніх умов.

Модулі реле

Виконавчі елементи системи представлені багатоканальними модулями реле, які дозволяють керувати подачею живлення до освітлювальних приладів і систем опалення. Кожне реле має гальванічну розв'язку між низьковольтною частиною (ESP32) та силовим колом 220 В, що забезпечує безпечну роботу.

Модулі реле дозволяють:

- дистанційно вмикати/вимикати прилади через хмарний інтерфейс;
- автоматично реагувати на зміну показників датчиків за програмним алгоритмом;
- працювати в режимі реального часу з відображенням статусу пристрою в додатку.

Реле є критично важливими елементами, оскільки вони виконують фізичну комутацію навантаження, перетворюючи логічні сигнали контролера на реальні дії.

Комунікаційні можливості

Однією з головних переваг обраної платформи є вбудована підтримка бездротових протоколів. На відміну від Arduino Uno, ESP32 не потребує зовнішніх модулів типу HC-05 чи ESP8266. Вбудований Wi-Fi дозволяє підключити систему безпосередньо до домашнього роутера, забезпечуючи

					ІТС.4КІ.0126.02-ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

зв'язок із сервером Blynk або іншими IoT-платформами. Це створює можливість керування системою з будь-якої точки світу через мобільний додаток. Крім того, наявність Bluetooth дозволяє здійснювати локальне налаштування системи навіть у разі відсутності інтернет-з'єднання.

Таблиця 2.2.

Bluetooth та Wi-Fi в ESP32

Характеристика	Bluetooth (Вбудований)	Wi-Fi (Вбудований)
Дальність зв'язку	до 15-20 м	до 100 м (залежно від роутера)
Швидкість передачі даних	до 2-3 Мбіт/с	до 150 Мбіт/с
Підключення до Інтернету	Відсутнє (пряме з'єднання)	Є (через точку доступу)
Можливість віддаленого доступу	Лише в межах приміщення	Доступ з будь-якої точки світу
Сумісність з мобільними додатками	Висока (для локального контролю)	Повна (хмарна інтеграція)
Енергоспоживання	Низьке (особливо BLE режим)	Середнє (активний обмін даними)
Безпека з'єднання	Стандартне парування	Шифрування WPA2/WPA3, SSL/TLS
Перевага	Локальне керування без мережі	Повна IoT-функціональність та хмари

2.4 Структурна схема системи

Автоматизована система керування освітленням та опаленням побутового приміщення побудована на базі мікроконтролера Arduino і призначена для забезпечення комфортних умов у типовому українському одноповерховому двокімнатному будинку, який включає вітальню, спальню, кухню, коридор та ванну кімнату. Основна ідея полягає у створенні єдиної інтелектуальної системи, здатної автоматично реагувати на зміну зовнішніх умов і потреб користувача, керуючи освітленням та опалювальними приладами. Нижче наведено блок-схему взаємодії системи.

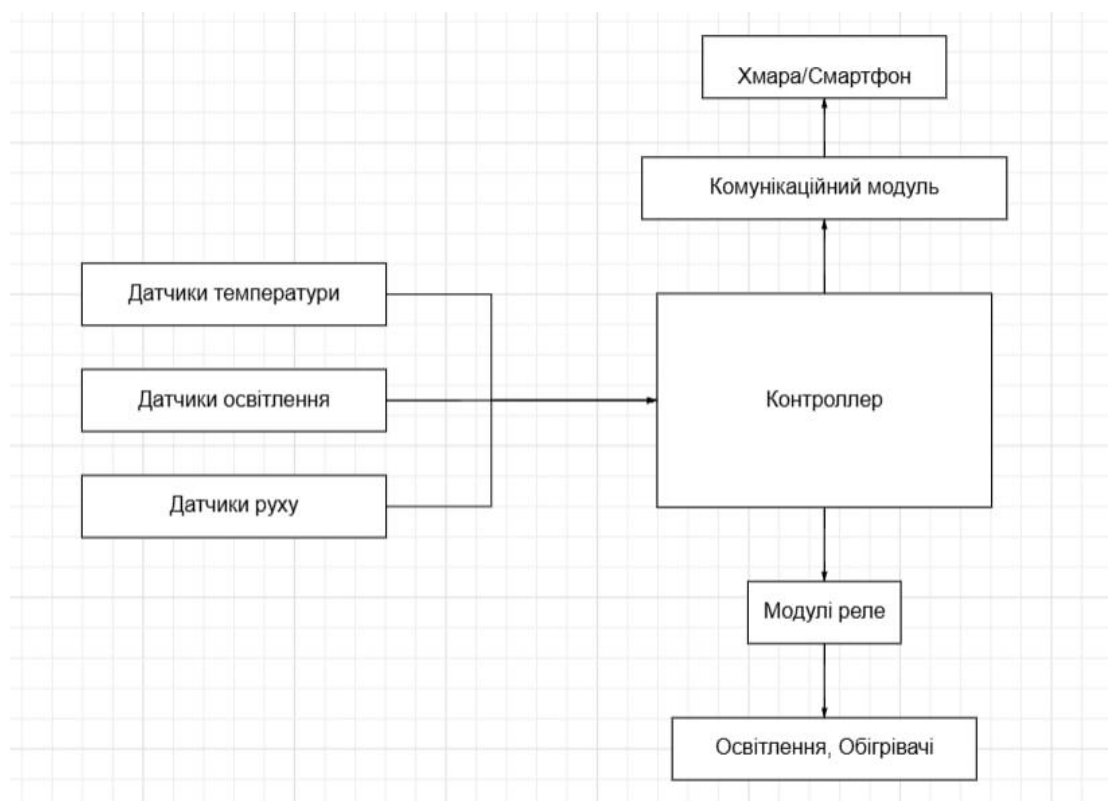


Рис. 2.3. Блок-схема системи

У структурі системи виділяються чотири основні підсистеми: сенсорна, виконавча, керуюча та комунікаційна. Центральним елементом є контролер ESP32, який виступає «мозком» системи. Він здійснює обробку сигналів від датчиків, приймає рішення відповідно до алгоритму роботи та керує виконавчими пристроями через модулі реле. Контролер також взаємодіє з комунікаційним модулем, забезпечуючи обмін даними з мобільним додатком

або веб-інтерфейсом користувача. ESP розміщується у зручному місці біля електрощита або в окремому корпусі, що забезпечує захист від механічних пошкоджень і стабільне живлення.

Сенсорна підсистема системи складається з кількох типів датчиків. Датчики температури розташовуються у кожному приміщенні для контролю мікроклімату. Вони дозволяють системі автоматично вмикати або вимикати опалення при досягненні заданих порогових значень температури. Датчики руху (PIR) дозволяють виявляти присутність людей у приміщеннях, що робить систему енергоефективною - світло активується лише тоді, коли користувач знаходиться у кімнаті.

Виконавча підсистема включає модулі реле, які керують живленням освітлювальних приладів і нагрівальних елементів. Кожен канал реле може бути прив'язаний до окремої групи освітлення чи опалювального пристрою. Наприклад, одне реле керує світлом у вітальні, інше - у спальні, третє - електронагрівачем або конвектором. Для безпеки реле підключаються через гальванічну розв'язку, а лінії живлення захищаються плавкими запобіжниками або автоматами.

Комунікаційна підсистема базується на модулі Wi-Fi (ESP8266 або ESP32), який дозволяє контролеру обмінюватися даними з користувачем через мережу. Модуль з'єднаний з Arduino через інтерфейс UART і забезпечує двосторонній зв'язок: користувач може дистанційно переглядати поточні параметри системи, а також надсилати команди для керування освітленням чи температурою. Це дає можливість інтеграції з мобільними додатками або хмарними платформами, такими як Blynk чи ThingSpeak, що дозволяє створити повноцінний інтернет-керований «розумний дім».

Інформація від усіх сенсорів надходить до Arduino через аналогові або цифрові входи. Контролер періодично опитує датчики, обробляє отримані дані та формує керуючі сигнали для реле. Ці сигнали активують або деактивують

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

відповідні пристрої - лампи, нагрівачі чи інші елементи побутової інфраструктури. Одночасно дані можуть передаватися до Wi-Fi модуля для відображення в інтерфейсі користувача.

Живлення системи організовано двома рівнями: 5 В постійного струму для електронних компонентів (Arduino, Wi-Fi модуль, датчики) та 220 В змінного струму для силових пристроїв (реле, освітлення, опалювальні елементи). Для безпеки в систему інтегровані елементи електрозахисту - розмикачі, УЗО та запобіжники.

Таким чином, структурна схема системи демонструє взаємодію між усіма її складовими: сенсори передають дані до контролера, Arduino аналізує інформацію та формує сигнали на модулі реле, які вмикають або вимикають побутові прилади. Через Wi-Fi модуль користувач може спостерігати за станом системи, змінювати параметри та керувати функціями дистанційно. Така архітектура забезпечує надійну роботу, модульність, простоту інтеграції та можливість подальшого розширення системи - наприклад, додавання функцій вентиляції, контролю доступу чи енергомоніторингу.

Висновки до розділу

У другому розділі було проведено детальну постановку задачі та описано функціональні можливості системи автоматизованого керування побутовими пристроями на базі мікроконтролерної платформи ESP32. Як об'єкт дослідження було обрано типовий український двокімнатний одноповерховий будинок, для якого розроблено концепцію інтелектуальної системи керування освітленням, температурним режимом і безпекою.

Було визначено основні завдання системи, серед яких - автоматичний контроль температури, регулювання рівня освітлення, виявлення руху в приміщеннях та дистанційне керування пристроями через вбудований модуль Wi-Fi. Сформовано алгоритм роботи системи, що включає етапи зчитування даних із датчиків, обробку отриманої інформації потужним двоядерним

					ІТС.4КІ.0126.02-ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

мікроконтролером ESP32 та формування відповідних керуючих сигналів для виконавчих елементів (реле, лампи, нагрівачі тощо) з можливістю одночасної передачі даних у хмарне сховище.

Вибрано та охарактеризовано основні компоненти системи: центральний контролер ESP32 DevKit, датчик температури та вологості DHT22, датчик освітленості LDR, інфрачервоний датчик руху PIR, а також модулі реле для керування силовими навантаженнями. Завдяки наявності вбудованого в контролер модуля Wi-Fi, було обґрунтовано відсутність потреби у додаткових комунікаційних платах, що спрощує апаратну архітектуру та підвищує надійність системи. Проведено аналіз бездротових можливостей платформи, у результаті чого зроблено висновок про перевагу використання інтегрованого Wi-Fi стеку для забезпечення стабільного дистанційного доступу з будь-якої точки світу.

На основі проведеного аналізу було розроблено структурну схему системи, яка відображає взаємозв'язок між усіма компонентами: датчиками, контролером, виконавчими пристроями та мобільним додатком користувача на базі IoT-платформи. У підсумку, другий розділ сформував повне уявлення про архітектуру майбутньої системи, принципи її функціонування та технічну базу. Отримані результати створюють підґрунтя для розробки програмного забезпечення, налагодження зв'язку між компонентами та реалізації експериментального зразка на наступному етапі роботи.

РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ

3.1. Структура програмного забезпечення

Для розробки й тестування програмної частини дипломного проекту доцільно використовувати онлайн-емулятор Wokwi – веб-платформу, яка дозволяє моделювати роботу плат Arduino, ESP32/ESP8266 і низки периферійних пристроїв прямо в браузері. Wokwi дає змогу швидко прототипувати алгоритми, перевіряти логіку взаємодії з датчиками та відлагоджувати послідовну діагностику без фізичного монтажу, що пришвидшує розробку й зменшує ризики на етапі проектування. Також платформа надає найважливішу перевагу – перевірка правильності підключення та реакції на різні критичні ситуації без шкоди для самої системи.

Першим та основним елементом платформи є вікно симуляції. Це основне робоче поле де додаються, розміщуються та під'єднуються компоненти. Компоненти підтягуються з загального сховища, або ж їх можна додати з великої кількості сторонніх бібліотек. Для підключення компонентів необхідно використовувати сокети та піни що знаходяться на платі.

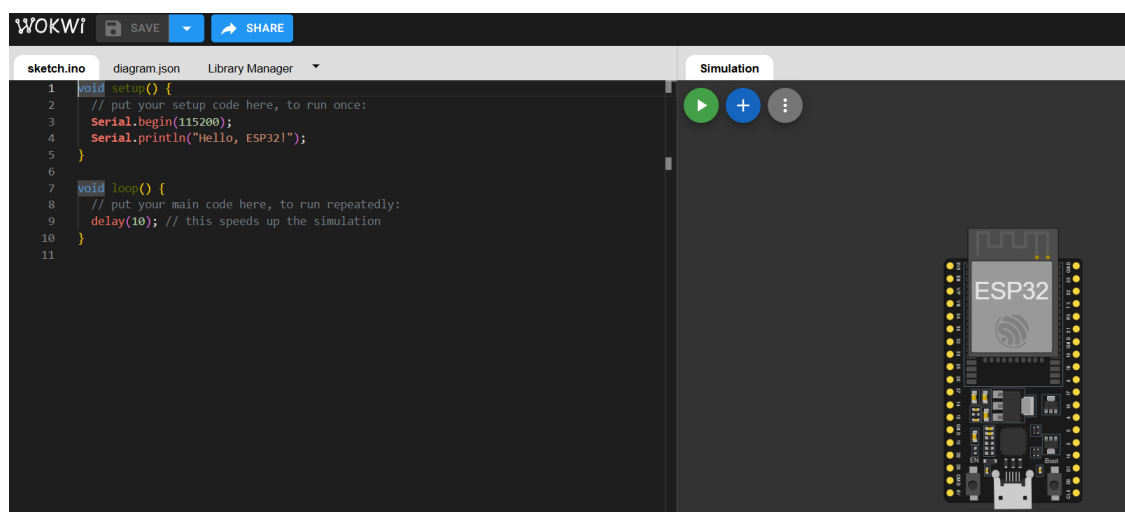


Рис 3.1. Вікно симуляції Arduino

					ІТС.4КІ.0325.03-ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата	РОЗДІЛ ІІІ. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ	Літ.	Арк.	Акрушіє	
Розроб.		Возняк С.В.							
Керівник		Донченко В.Ю.						37	24
Реценз.		Козуб Ю.Г.				ЛНУ Кафедра ІТС, Гр.4КІ			
Н. Контр.									
Зав. каф.		Семенов М.А.							

Живлення на плату ESP32 подається у стандартному режимі 5V через роз'єм Micro-USB або USB-C (залежно від версії плати), після чого вбудований регулятор напруги знижує її до робочих 3.3V.

Другим елементом роботи з ESP32 є вікно редактора коду. Найчастіше для цього використовується середовище Arduino IDE або VS Code (PlatformIO). Редактор коду - це головний інструмент для задання логіки мікроконтролера; він дозволяє редагувати, компілювати та завантажувати програми (скетчі) у внутрішню пам'ять плати. Алгоритми взаємодії з датчиками, обробка сенсорних даних та миттєве реагування на події реалізуються саме за допомогою програмної логіки.

З'єднання елементів відбувається у вікні діаграм – це програмований JSON файл конфігурації, який дозволяє вручну реалізовувати з'єднання, прописувати пристрої та унікальні ідентифікатори. Тобто цей файл – альтернатива малюванню схеми за допомогою графічного інтерфейсу, а також візуалізація схеми у вигляді коду.

Як було вказано в першому розділі, програмування ESP32 базується на мовах C/C++, що в поєднанні з двоядерною архітектурою забезпечує надзвичайно швидку обробку даних, стабільну роботу мережевих протоколів та можливість виконання фонових завдань без зупинки основного циклу керування.

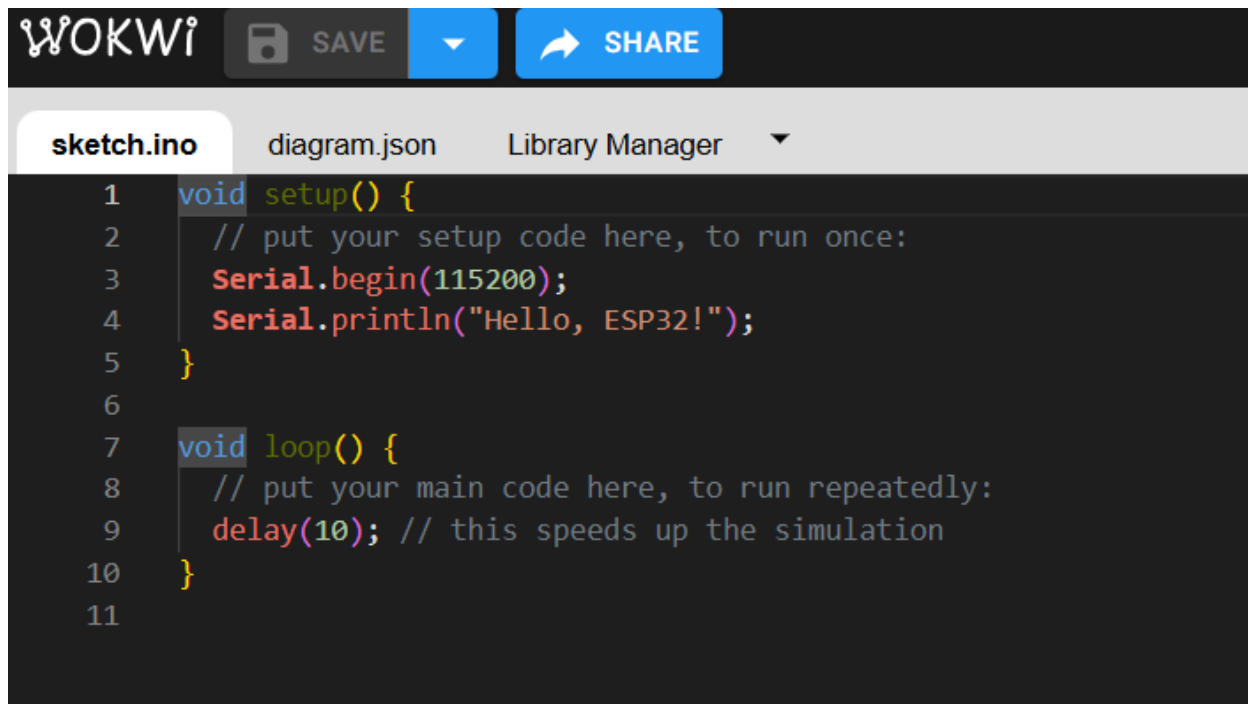


Рис. 3.2. Вікно редактора коду Arduino

Останнім елементом інтерфейсу емулятора є debug меню. Консольний інтерфейс дозволяє відстежувати коректність програмної частини схеми та роботи компонентів, а також керувати апаратною частиною за допомогою команд.

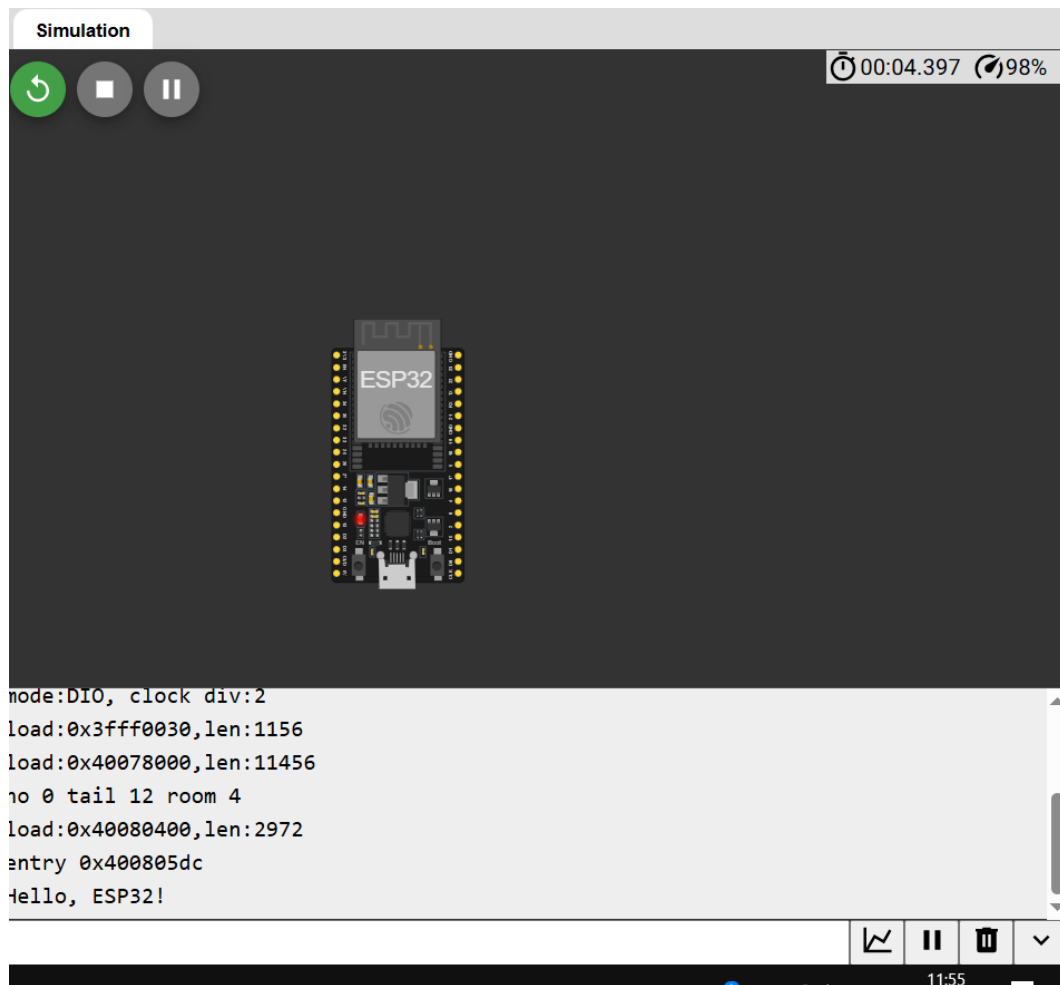


Рис. 3.3. Вікно debug меню

Недоліком проектування в межах онлайн платформи є неможливість точного розрахунку довжини реле, кабелів та маршрутів, але для розгляду та моделювання в рамках можливостей платформи це не потрібно.

3.2. Проектування схеми в Arduino IDE

Першочергово я знаходжу та підключаю необхідні бібліотеки компонентів для заданого функціоналу, тобто бібліотеки для ІОТ взаємодії, бібліотеки датчиків температури, освітлення та реакції на рух.

Серед великої кількості бібліотек я надавав перевагу ресурсам за виробництвом Adafruit. Раніше мені доводилось працювати з різними бібліотеками Adafruit і головними перевагами їх є уніфікована система стандартів, яка ліквідує розбіжності та баги на етапі розробки схем, а також дозволяє змінювати деякі елементи без впливу на ефективність системи.

					ІТС.4КІ.0126.02-ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

Також вони додають велику кількість документації, що ідеально підходить для студентів та людей з невеликим досвідом.

Проте, якщо не вдавалось знайти бібліотеки виробництва Adafruit, я використовував перевірені часом популярні бібліотеки, оглядаючи відгуки в онлайн-ресурсах.

Найважливішою бібліотекою мого проекту можу сміливо назвати «Blynk», оскільки, саме вона відповідає за надсилання інформації на телефон та надання можливості дистанційного регулювання середовища.

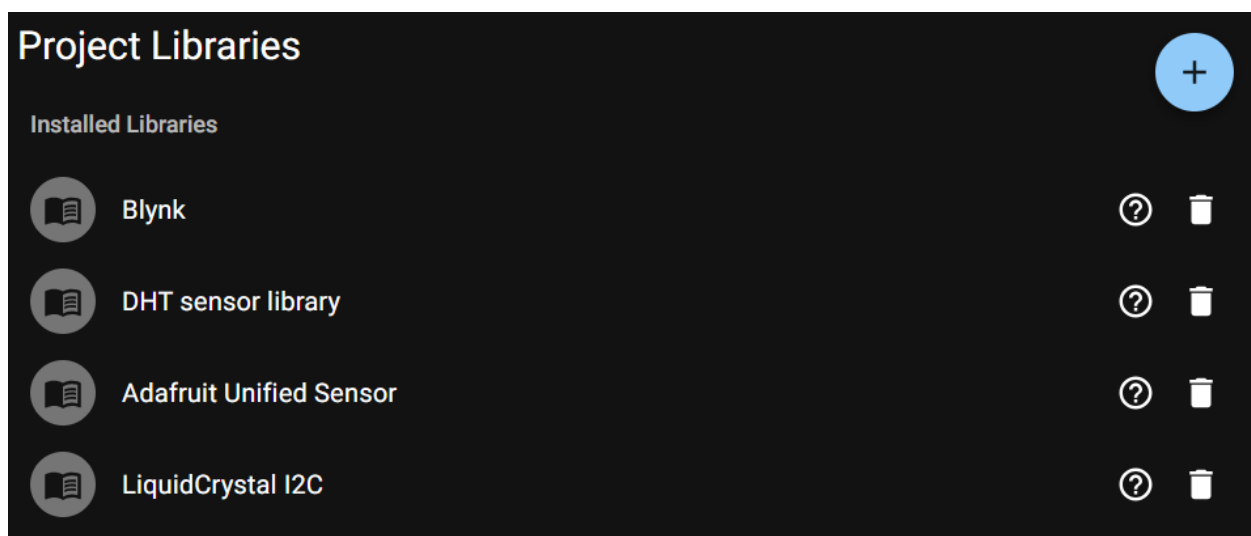


Рис. 3.4. Список використаних бібліотек

Другим етапом розробки є додання на робоче поле елементів схеми та створення з'єднань між ними. Оскільки wokwi надає можливість керувати елементами та підключеннями, як через вікно моделювання, так і через файл JSON – я використовував обидва методи, а саме додавав елементи та виконував з'єднання вручну, а додаткові налаштування ідентифікаторів та кольорів робив в самому файлі.

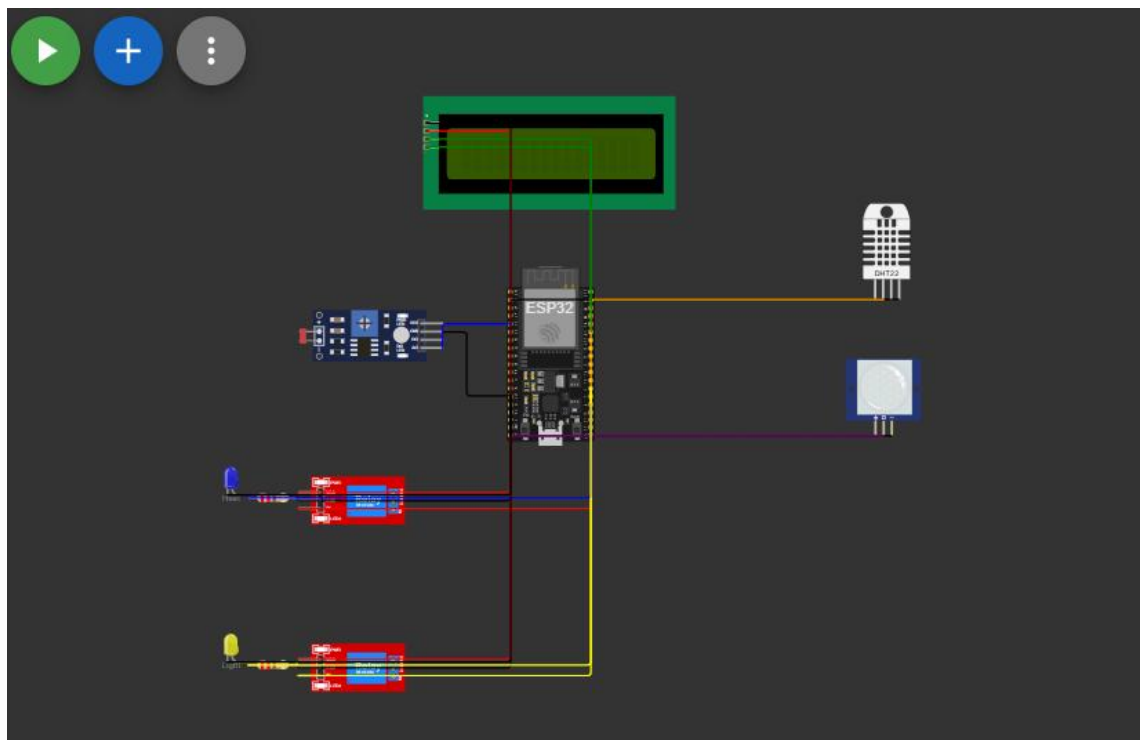


Рис. 3.5. Проектування схеми

Як видно на рис. 3.5, поточна схема складається з 11 компонентів:

Таблиця 3.1.

Опис елементів схеми

Номер	Компонент	Призначення у схемі
1	ESP32 DevKit-C V4	Головний мікроконтролер, що зчитує дані з датчиків, обробляє логіку та керує Wi-Fi зв'язком.
2	DHT22	Цифровий датчик для точного вимірювання температури та вологості повітря.
3	PIR Motion Sensor	Інфрачервоний датчик для фіксації руху людини в зоні моніторингу.

4	LDR Sensor Module (ldr1)	Чотирипіновий модуль фоторезистора для вимірювання рівня світла через аналоговий вихід АО.
5	LCD1602 (I2C)	Дисплей для локального виводу інформації про стан системи та показники датчиків.
6	Relay Module (Heater)	Електромагнітне реле для дистанційного керування живленням обігрівача.
7	Relay Module (Light)	Електромагнітне реле для автоматичного ввімкнення та вимкнення освітлення.
8	Blue LED (Heat)	Світлодіодний індикатор синього кольору для візуалізації роботи режиму опалення.
9	Yellow LED (Light)	Світлодіодний індикатор жовтого кольору для візуалізації роботи ламп освітлення.
10	Resistor (res1)	Обмежувальний резистор на 220 Ом для безпечного підключення синього світлодіода.
11	Resistor (res2)	Обмежувальний резистор на 220 Ом для безпечного підключення жовтого світлодіода.

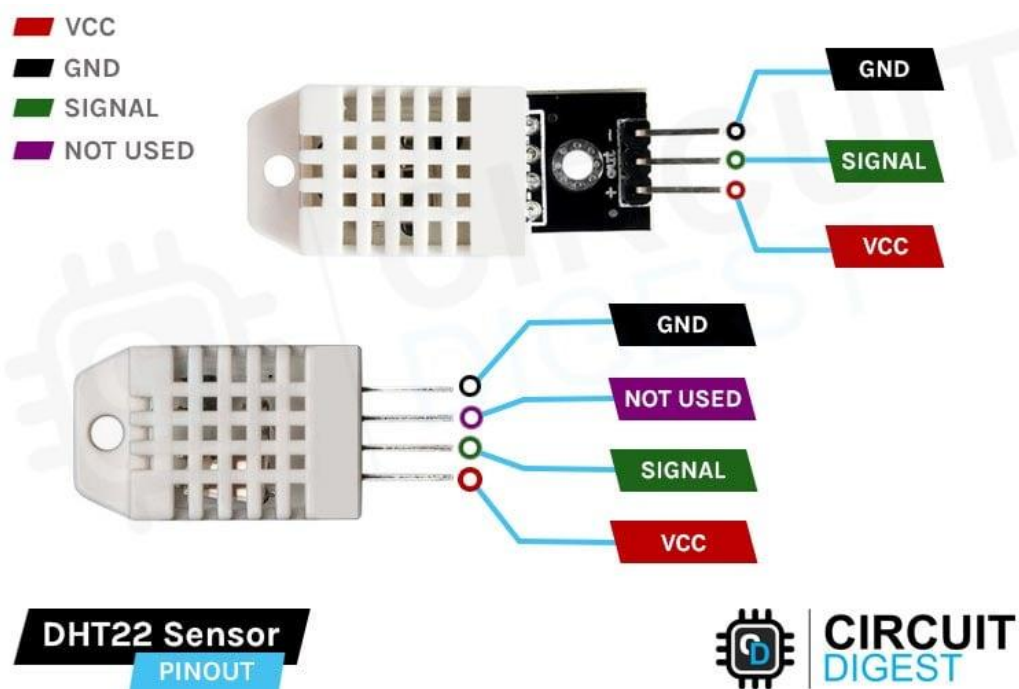


Рис.3.6. DHT22

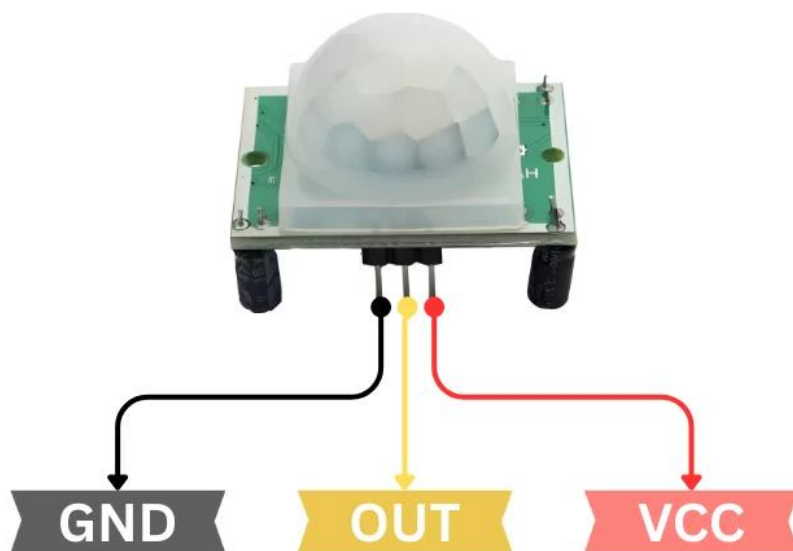


Рис. 3.7. Pir Motion Sensor

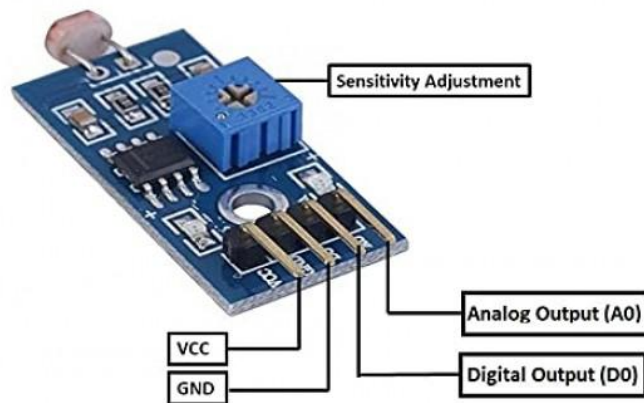


Рисунок 3.8. LDR Sensor Module

Розглянемо як виглядають всередині JSON файлу компоненти та підключення на прикладі одного з компонентів. Для розгляду я обрав датчик температури DHT22.

Спочатку компонент вказується в масиві «parts», що дає емулятору зрозуміти додання нового елементу на робочому полі. В цьому масиві вказується назва елементу, призначається унікальний id, координати на робочому полі та задається базовий атрибут(параметр температури який буде йти стандартом при завантаженні системи до отримання перших даних).

```
{
  "type": "wokwi-dht22",
  "id": "dht",
  "top": -76.5,
  "left": 426.6,
  "attrs": { "temperature": "27.3" }
},
```

Рис. 3.9. DHT22 – parts

Всі з'єднання та їх подробиці знаходяться в масиві «connections». В цьому масиві записи йдуть у форматі «початковий пін з'єднання → кінцевий пін → колір з'єднання → тип з'єднання».

```
[ "esp:GND.1", "dht:GND", "black", [ "v0" ] ],
[ "esp:3.3V.1", "dht:VCC", "red", [ "v0" ] ],
[ "esp:15", "dht:SDA", "orange", [ "v0" ] ],
```

Рис. 3.10. DHT22 – connections

3.3. Алгоритм автоматичного керування освітленням і опаленням

Алгоритм системи керування базується на циклічному опитуванні пристроїв та автоматичному порівнянні отриманих даних із заданими вручну константами. Для керування мікрокліматом контролер кожні дві секунди зчитує показники температури з датчика DHT22 і, у разі їх падіння нижче встановленого порогу, активує реле обігрівача через цифровий вихід GPIO 2.

Паралельно реалізована логіка інтелектуального освітлення, яка використовує датчик руху PIR та фоторезистор для аналізу навколишнього середовища. Система фіксує момент останньої активності людини та підтримує живлення ламп на піні 4 протягом заданої кількості секунд, проте автоматично блокує ввімкнення світла, якщо рівень природної освітленості перевищує межу в 2000 одиниць. Усі поточні параметри, включаючи статус руху та фактичні значення сенсорів, у реальному часі виводяться на LCD-дисплей для локального моніторингу стану об'єкта.

Розглянемо як я це реалізував у коді.

Спочатку ініціалізую константи які будуть орієнтиром для системи керування.

```
float targetTemp = 21;
int lightThreshold = 2000;
bool autoLightMode = false;
unsigned long lastMotionTime = 0;
const unsigned long motionTimeout = 15000; // 15 сек для тесту
```

Рис. 3.11. Задання констант

Як ми бачимо на рис. 3.11, для температури я використовую значення float, для можливості гнучко налаштовувати температурні параметри, для світла ж береться ціле число, булева змінна виконує функцію перемикача розумного режиму освітлення, для тестування реакції освітлення я встановив час 15 секунд.

В подальшому для користувача надається можливість змінювати константи за допомогою програмного засобу на телефоні, що робить систему керування надзвичайно зручною.

```
// Вивід у Serial Monitor
Serial.print("Temp: "); Serial.print(t);
Serial.print(" | Light: "); Serial.print(lightLevel);
Serial.println(motion ? " | [MOTION]" : " | [WAITING]");

// Опалення
if (t < targetTemp) digitalWrite(RELAY_HEATER, HIGH);
else digitalWrite(RELAY_HEATER, LOW);

// Світло
if (motion) lastMotionTime = millis();
bool shouldBeOn = (millis() - lastMotionTime < motionTimeout);
if (shouldBeOn && autoLightMode && lightLevel < lightThreshold) shouldBeOn = false;
digitalWrite(RELAY_LIGHT, shouldBeOn ? HIGH : LOW);

// CD Екран
lcd.setCursor(0, 0);
lcd.print("T:"); lcd.print(t, 1); lcd.print("/"); lcd.print(targetTemp, 0); lcd.print("C ");
lcd.setCursor(0, 1);
lcd.print("L:"); lcd.print(lightLevel);
lcd.print(motion ? " Mov!" : " WAITING ");
```

Рис. 3.12. Логіка розумного керування

Як видно на рисунку, за допомогою команди print я можу редагувати повідомлення які ми будемо отримувати як на фізичному дисплеї, так і всередині системи емулятора.

Опалення реалізоване за дуже простою схемою – якщо значення яке отримує система від DHT22 нижче заданої константи, то подається живлення на реле опалення, інакше живлення вимикається. Така проста схема є надзвичайно ефективною, оскільки система кожні 2 секунди перевіряє приміщення і це допомагає уникнути надлишкової ефективності системи.

Керування світлом реалізоване декілька складніше, оскільки окрім константи може орієнтуватись на рух, за умови перемикавання розумного режиму. Тобто окрім постійної перевірки значень, як в керуванні опаленням, також встановлена перевірка на перемикач режиму роботи та реакція на PIR сенсор.

3.4. Розробка користувацького інтерфейсу (Bluetooth / веб-керування)

При виборі технології зв'язку між Bluetooth та веб-керуванням через Wi-Fi було віддано перевагу останньому через низку суттєвих переваг. Технологія Bluetooth обмежує радіус дії системи кількома десятками метрів і вимагає постійної прямої видимості або близькості до контролера, що нівелює концепцію віддаленого моніторингу. Веб-керування через Wi-Fi дозволяє інтегрувати пристрій у глобальну мережу, забезпечуючи доступ до керування з будь-якої точки світу, де є інтернет. Крім того, протоколи передачі даних через Wi-Fi є більш масштабованими, дозволяють одночасне підключення кількох користувачів та забезпечують стабільне зберігання історії показників на хмарних серверах.

Для реалізації цього проекту було обрано платформу Blynk, оскільки вона пропонує найбільш збалансоване поєднання функціональності та швидкості розробки. Blynk надає готовий набір графічних віджетів, що дозволяє створити професійний інтерфейс користувача без написання складного коду для мобільних додатків Android або iOS. Важливою перевагою є наявність надійного хмарного сервера, який бере на себе обробку пакетів даних між ESP32 та смартфоном, гарантуючи мінімальну затримку відгуку. Вибір цієї платформи обумовлений також її високою сумісністю з

					ІТС.4КІ.0126.02-ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

екосистемою Arduino, що дозволило легко реалізувати дворівневу логіку керування: стабільну автономну роботу алгоритму на рівні мікроконтролера та гнучке корегування параметрів через віртуальні піни хмарного інтерфейсу.

Blynk надає можливість організації дистанційного проектування та керування, як з телефону так і з комп'ютеру, але я буду використовувати його виключно на телефоні.

На початку, необхідно встановити застосунок на телефон та зареєструватись як базовий користувач.

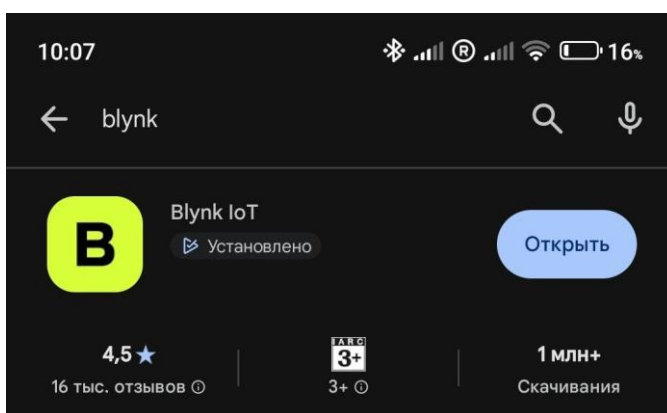


Рис. 3.13. Встановлення Blynk IoT

Далі я створюю новий проект під назвою «System», він буде відповідати за моніторинг та керування світлом та опаленням. Для отримання та передачі даних на ESP32 я додаю Template «System» 6 нових елементів в Datastreams.

Після створення темплейтів Datastreams, потрібно на робочому полі додати відповідні віджети для відображення отриманих даних та задання нових значень в систему ESP32.

Список компонентів «System»

Номер (Pin)	Призначення	Опис функціоналу	Віджет в Blynk
V0	Поточна температура	Передача реальних показників з датчика DHT22 для відображення на віджеті.	Labeled Value
V1	Освітленість (LDR)	Передача аналогового значення з фоторезистора (0– 4095) для відстеження рівня світла в приміщенні.	Gauge
V2	Уставка температури	Отримання значення від користувача через віджет Slider або Step Control для встановлення порогу нагріву (targetTemp).	Vertical Step Slider
V3	Поріг освітленості	Налаштування чутливості автоматичного світла. Визначає, при якій темряві системі дозволено вмикати лампи.	Vertical Step Slider

Номер (Pin)	Призначення	Опис функціоналу	Віджет в Blynk
V4	Автоматичний режим	Віртуальний перемикач (Switch/Button), що активує або деактивує інтелектуальну логіку керування світлом за датчиками.	Switch
V5	Статус освітлення	Зворотний зв'язок від системи. Відображає реальний стан реле світла (ввімкнено/вимкнено) на індикаторі LED у додатку.	Led

Нижче наведено рисунок як це виглядає всередині інтерфейсу Blynk.

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

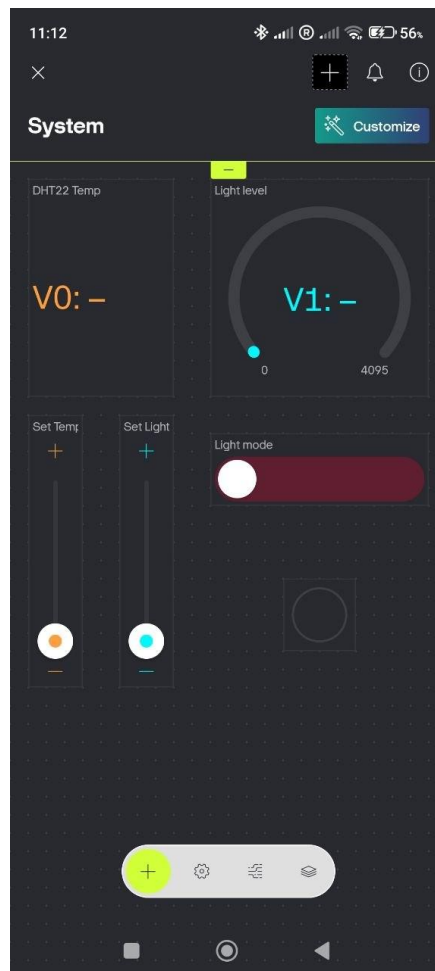


Рис. 3.14. Інтерфейс темплейту в Blynk

Після цього необхідно згенерувати індивідуальний токен автентифікації (Auth Token). У особистому кабінеті копіюю токен, отриманий 32-символьний код є ключем доступу, який дозволяє конкретній моделі в Wokwi підключитися до мого особистого кабінету в Blynk і обмінюватися даними в реальному часі.

Наступним кроком я готую програмне забезпечення в симуляторі. Я копіюю отримані ідентифікатори (ID, Name та Token) і вставляю їх у перші рядки файлу sketch.ino. Це критично важливий момент, оскільки без цих констант бібліотека Blynk не зможе ініціалізувати мережевий стек. Також у налаштуваннях WiFi я вказую Wokwi-GUEST, який симулятор використовує для виходу в реальний інтернет.

```

#define BLYNK_TEMPLATE_ID "TMPL4dY8kHwOy"
#define BLYNK_TEMPLATE_NAME "system"
#define BLYNK_AUTH_TOKEN "tGi8yRGvVQdH2IlvY8KAi

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <DHT.h>
#include <LiquidCrystal_I2C.h>

#define DHTPIN 15
#define DHTTYPE DHT22
#define PIR_PIN 13
#define LDR_PIN 34
#define RELAY_HEATER 2
#define RELAY_LIGHT 4

char ssid[] = "Wokwi-GUEST";
char pass[] = "";

```

Рис. 3.15. Підключення Blynk до ESP32

Як видно на рисунку 3.15, спочатку відбувається ініціація токенів для Blynk, далі йдуть бібліотеки підключені до проекту, після цього призначення пінів та SSID для підключення до віртуального роутеру Wokwi.

Розглянемо як відбувається передача даних з Blynk на мікроконтролер на прикладі змінення рівню температури.

```

BLYNK_WRITE(V2)
{
    targetTemp = param.asFloat();
    Serial.print("Blynk: Target T = ");
    Serial.println(targetTemp);
}

```

Рис. 3.16. Взаємодія Blynk з мікроконтролером

За отримання даних відповідає функція «BLYNK_WRITE()», в параметри якої передається відповідний ідентифікатор Datastream, в моєму випадку за повзунок температури відповідає третій ідентифікатор V2. Далі функція param.asFloat() перетворює отримане значення на дробове число та

					ІТС.4КІ.0126.02-ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

записує в змінну для відстеження нового рівню температури. Команди типу «Serial» відповідають за виведення інформації в консоль для відстеження реакцій системи.

3.5. Тестування та результати роботи системи

Завершальним етапом розробки є проведення комплексного тестування апаратно-програмного комплексу в середовищі симуляції Wokwi з підключенням до хмарної платформи Blynk. Метою тестування є перевірка точності спрацювання логічних алгоритмів та стабільності передачі даних.

Спочатку запускаю систему та очікую підключення до мережі для можливості взаємодіяти з Blynk.

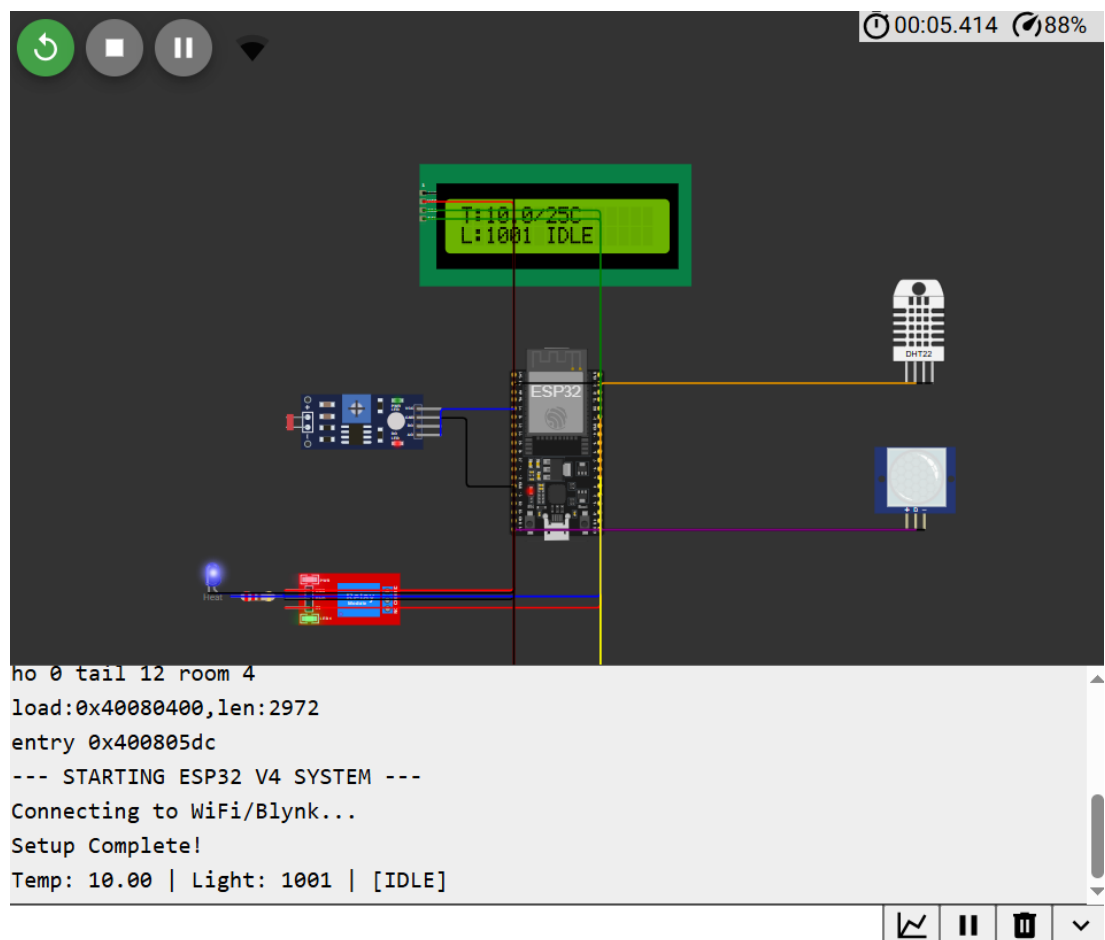


Рис. 3.17. Запуск системи

Завдяки «Serial» повідомленням ми бачимо, що систем встановила зв'язок, почала передавати значення в консоль з датчиків та знаходиться в режимі очікування руху на PIR-сенсорі.

Тепер необхідно підключитись в Blynk до нашої системи.

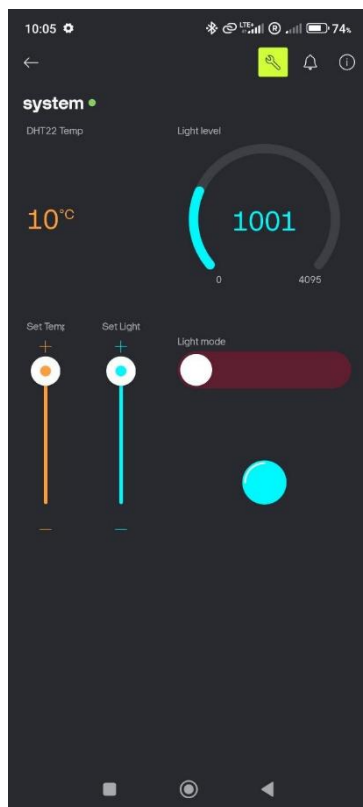


Рис. 3.18. Робота в Blynk

Як видно на рисунку 3.18, система передає температуру з датчику, рівень освітлення з сенсора, та роботу освітлення в Blynk.

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

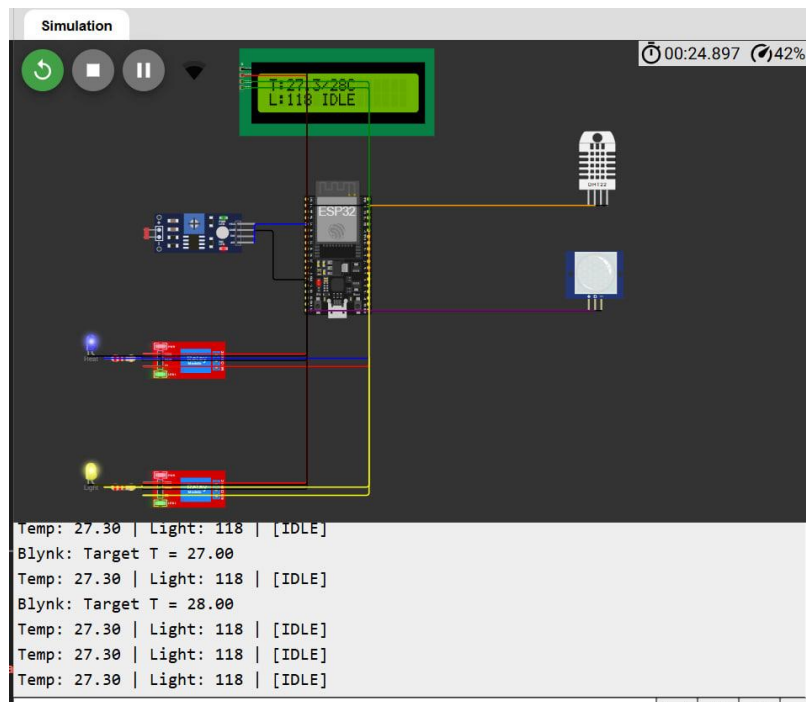


Рис. 3.19. Перевірка роботи датчиків

На рисунку 3.19 візуалізовано реакцію схеми відповідно до параметрів – рівень освітлення в кімнаті занадто низький тому йде живлення на реле світла, рівень опалення також занадто низький(в Blynk я встановив 30 градусів), тому подається живлення на реле опалення.

Тепер я зміню параметри для перевірки роботи системи в режимі «очікування».

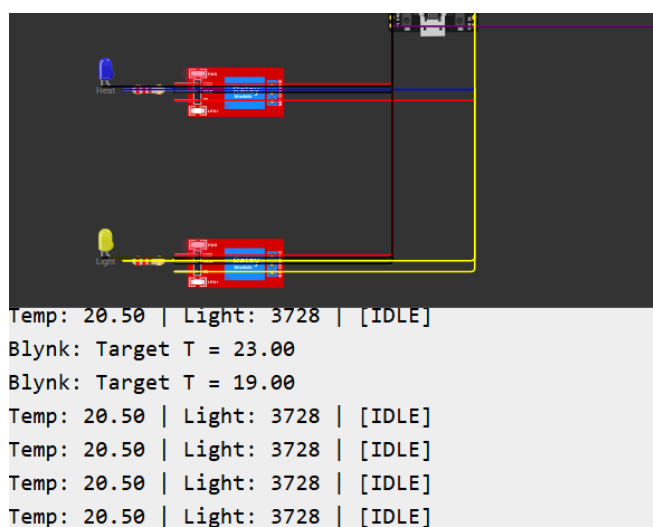


Рис. 3.20. Тестування системи з різними даними

Я додав достатньо освітлення, встановив межу температури в 19 градусів, а температуру в кімнаті на 20. Реакція системи відповідна – живлення не подаються на обидва реле.

Остання перевірка системи – перевірка розумного режиму освітлення, який реалізований за допомогою PIR-сенсору. Для цього необхідно переключити режим роботи в Blynk(switch на робочому полі) та провести імітацію руху.

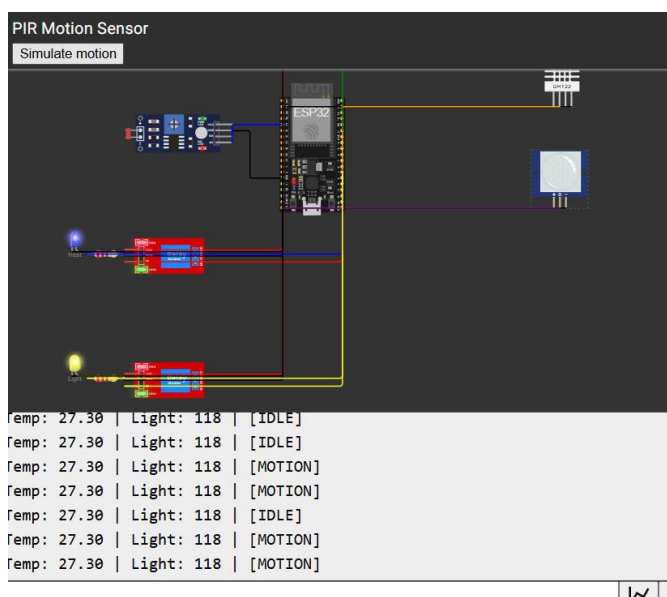


Рис. 3.21. Робота PIR-сенсору

Висновки до розділу

У ході виконання третього розділу було проведено повну програмну реалізацію системи автоматизації на базі мікроконтролера ESP32. Основним результатом роботи є створення стабільного алгоритму, який інтегрує локальне керування периферійними пристроями з хмарними сервісами інтернет-речей.

Проектування структури програмного забезпечення дозволило досягти високої швидкодії системи. Використання неблокуючого забезпечило паралельне виконання завдань: зчитування сенсорів DHT22 та LDR, оновлення інформації на LCD-дисплеї та підтримку активного Wi-Fi з'єднання без затримок у роботі основного циклу програми.

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

Розроблений алгоритм автоматичного керування ефективно поєднує дані з декількох джерел. Логіка роботи підсистеми освітлення враховує не лише наявність руху, а й рівень фонові ілюмінації, що є ключовим фактором енергозбереження. У свою чергу, підсистема терморегуляції реалізує функцію прецизійного підтримки мікроклімату на основі динамічних уставок, що передаються з боку користувача.

Важливим етапом стала розробка користувацького інтерфейсу на платформі Blynk. Застосування віртуальних пінів (Virtual Pins) дозволило створити гнучку систему дистанційного налаштування робочих параметрів. Це забезпечило можливість змінювати порогові значення температури та світла в режимі реального часу, що значно підвищує адаптивність системи до умов експлуатації.

Етап тестування в середовищі Wokwi підтвердив технічну справність усіх модулів та коректність логічних переходів у коді. Аналіз результатів симуляції показав стабільну синхронізацію між віртуальною апаратною частиною та мобільним інтерфейсом, що свідчить про готовність розробленого програмного продукту до впровадження в реальні системи автоматизації будівель.

ВИСНОВКИ

У ході виконання дипломної роботи було повністю досягнуто поставленої мети - розробки та дослідження інтелектуальної системи автоматизованого керування освітленням та опаленням на базі сучасних IoT-технологій. Результати проведеної роботи дозволяють зробити наступні підсумки:

На першому етапі було проведено ґрунтовний аналіз сучасних підходів до автоматизації житлових приміщень. У ході теоретичного дослідження було розглянуто та порівняно існуючі системи керування, що дозволило виявити їхні переваги та недоліки. Особливу увагу приділено вивченню мікроконтролерних платформ, за результатами чого було обґрунтовано вибір контролера ESP32 як найбільш продуктивного рішення, що має вбудовані інтерфейси Wi-Fi та Bluetooth, необхідні для реалізації концепції «розумного будинку».

У процесі проектування було проведено детальне планування архітектури системи для типового житлового об'єкта. Сформовано функціональні вимоги та обрано оптимальний набір сенсорів (DHT22, LDR, PIR), що дозволило створити комплексну систему моніторингу навколишнього середовища. Розроблена структурна схема підтвердила можливість побудови надійної системи з мінімальною кількістю додаткових комунікаційних модулів.

Програмна реалізація проекту дозволила втілити розроблені алгоритми в енергоефективний код. Було успішно реалізовано логіку автоматичного підтримання температурного режиму та інтелектуального керування світлом, що базується на комбінованому аналізі присутності людини та рівня природної освітленості. Інтеграція з хмарною платформою Blynk забезпечила

					ІТС.4КІ.0126.02-ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата					
Розроб.		Возняк С. В.			ВИСНОВКИ		Літ.	Арк.	Акрушіє
Керівник		Донченко В.Ю.						61	4
Реценз.		Козуб Ю.Г.					ЛНУ Кафедра ІТС, Гр.4КІ		
Н. Контр.									
Зав. каф.		Семенов М.А.							

виконання завдання щодо створення зручного користувацького інтерфейсу для дистанційного моніторингу та корекції параметрів системи в реальному часі.

Експериментальна частина роботи, проведена в середовищі симуляції Wokwi, повністю підтвердила теоретичні розрахунки. Тестування продемонструвало стабільність мережевої взаємодії, точність спрацювання виконавчих реле та коректність відображення даних на локальному дисплеї та в мобільному додатку.

Таким чином, усі завдання дипломної роботи виконано в повному обсязі. Розроблена система є функціонально завершеним прототипом, який доводить доцільність використання обраних технічних засобів для створення сучасних, доступних та енергоефективних систем автоматизації будівель.

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке розумний дім і як він працює [Електронний ресурс]. – Режим доступу: <https://ds-electronics.com.ua/support/blog/ymniy-dom/chto-takoe-i-kak-rabotaet-ymniy-dom/> (дата звернення: 20.10.2025).
2. ESP32 Guide – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/guides/esp32> (дата звернення: 25.10.2025).
3. LCD1602 – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/parts/wokwi-lcd1602> (дата звернення: 30.10.2025).
4. Photoresistor Sensor – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/parts/wokwi-photoresistor-sensor> (дата звернення: 05.11.2025).
5. DHT22 Sensor – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/parts/wokwi-dht22> (дата звернення: 15.11.2025).
6. PIR Motion Sensor – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/parts/wokwi-pir-motion-sensor> (дата звернення: 20.11.2025).
7. Relay Module – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/parts/wokwi-relay-module> (дата звернення: 25.11.2025).
8. LED – Wokwi Docs [Електронний ресурс]. – Режим доступу: <https://docs.wokwi.com/parts/wokwi-led> (дата звернення: 30.11.2025).
9. Arduino Documentation [Електронний ресурс]. – Режим доступу: <https://docs.arduino.cc/> (дата звернення: 10.12.2025).
10. Blynk Documentation [Електронний ресурс]. – Режим доступу: <https://docs.blynk.io/en> (дата звернення: 20.12.2025).
11. Adafruit DHT Sensor Library – GitHub Repository [Електронний ресурс]. – Режим доступу: <https://github.com/adafruit/DHT-sensor-library> (дата звернення: 05.01.2026).

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

12. Smart Home Automation Using Blynk ESP32 – Instructables [Електронний ресурс]. – Режим доступу: <https://www.instructables.com/Smart-Home-Automation-Using-Blynk-ESP32-IoT-Projec/> (дата звернення: 15.01.2026).
13. ESP32 DHT11/DHT22 Web Server – Random Nerd Tutorials [Електронний ресурс]. – Режим доступу: <https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-web-server-arduino-ide/> (дата звернення: 25.01.2026).
14. Smart Home via Blynk ESP32 – Hackster.io [Електронний ресурс]. – Режим доступу: <https://www.hackster.io/dimasamya/smart-home-via-blynk-esp32-2cdf90> (дата звернення: 05.02.2026).
15. ESP32 PIR Motion Sensor – Random Nerd Tutorials [Електронний ресурс]. – Режим доступу: <https://randomnerdtutorials.com/esp32-pir-motion-sensor-interrupts-timers/> (дата звернення: 20.02.2026).
16. ESP32 Projects – Circuit Digest [Електронний ресурс]. – Режим доступу: <https://circuitdigest.com/esp32-projects> (дата звернення: 10.04.2026).
17. Wokwi ESP32 Simulator [Електронний ресурс]. – Режим доступу: <https://wokwi.com/esp32> (дата звернення: 16.05.2026).

ДОДАТОК А. Код Sketch.ino

```
#define BLYNK_TEMPLATE_ID "TMPL4dY8kHWOy"

#define BLYNK_TEMPLATE_NAME "system"

#define                                     BLYNK_AUTH_TOKEN
"tGi8yRGvVQdH2IlvY8KAirpm_GsQyY8S"#include <WiFi.h>


#include <BlynkSimpleEsp32.h>


#include <DHT.h>


#include <LiquidCrystal_I2C.h>


// Визначення пінів (згідно зі схемою)

#define DHTPIN 15

#define DHTTYPE DHT22

#define PIR_PIN 13

#define LDR_PIN 34

#define RELAY_HEATER 2

#define RELAY_LIGHT 4

char ssid[] = "Wokwi-GUEST";

char pass[] = "";
```

					ІТС.4КІ.0126.02-ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	ДОДАТОК А	Літ.	Арк.	Акрушіє
Розроб.		Возняк С. В.						
Керівник		Донченко В.Ю.					65	5
Реценз.		Козуб Ю.Г.				ЛНУ Кафедра ІТС, Гр.4КІ		
Н. Контр.								
Зав. каф.		Семенов М.А.						

```

DHT dht(DHTPIN, DHTTYPE);

LiquidCrystal_I2C lcd(0x27, 16, 2);

BlynkTimer timer;

float targetTemp = 25.0;

int lightThreshold = 2000;

bool autoLightMode = false;

unsigned long lastMotionTime = 0;

const unsigned long motionTimeout = 30000; // 30 секунд для тесту

// Отримання налаштувань з Blynk

BLYNK_WRITE(V2) {

targetTemp = param.asFloat();

Serial.print("Blynk: Target T = ");

Serial.println(targetTemp);

}

BLYNK_WRITE(V3) {

lightThreshold = param.asInt();

}

BLYNK_WRITE(V4) {

autoLightMode = param.asInt();

}

void updateSystem() {

```

					ITC.4KI.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

```

float t = dht.readTemperature();

int lightLevel = analogRead(LDR_PIN);

bool motion = digitalRead(PIR_PIN);

if(isnan(t)) {

    Serial.println("Error: DHT22 not found!");

    return;

}

// Вивід у Serial Monitor

Serial.print("Temp: ");

Serial.print(t);

Serial.print(" | Light: ");

Serial.print(lightLevel);

Serial.println(motion ? " | [MOTION]" : " | [IDLE]");

// 1. Опалення

if(t < targetTemp) digitalWrite(RELAY_HEATER, HIGH);

else digitalWrite(RELAY_HEATER, LOW);

// 2. Світло

if(motion) lastMotionTime = millis();

bool shouldBeOn = (millis() - lastMotionTime < motionTimeout);

if(shouldBeOn && autoLightMode && lightLevel < lightThreshold) shouldBeOn
= false;

digitalWrite(RELAY_LIGHT, shouldBeOn ? HIGH : LOW);

// 3. LCD Екран

```

					ІТС.4КІ.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

```

lcd.setCursor(0, 0);

lcd.print("T:");

lcd.print(t, 1);

lcd.print("/");

lcd.print(targetTemp, 0);

lcd.print("C ");

lcd.setCursor(0, 1);

lcd.print("L:");

lcd.print(lightLevel);

lcd.print(motion ? " Mov!" : " IDLE ");

// 4. Blynk Cloud

if(Blynk.connected()) {

  Blynk.virtualWrite(V0, t);

  Blynk.virtualWrite(V1, lightLevel);

  Blynk.virtualWrite(V5, shouldBeOn);

}

}

void setup() {

  Serial.begin(115200);

  Serial.println("--- STARTING ESP32 V4 SYSTEM ---");

  pinMode(PIR_PIN, INPUT);

  pinMode(RELAY_HEATER, OUTPUT);

```

					ITC.4KI.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

```

pinMode(RELAY_LIGHT, OUTPUT);

lcd.init();

lcd.backlight();

dht.begin();

Serial.println("Connecting to WiFi/Blynk...");

Blynk.config(BLYNK_AUTH_TOKEN);

WiFi.begin(ssid, pass);

timer.setInterval(2000 L, updateSystem);

Serial.println("Setup Complete!");

}

void loop() {

if(WiFi.status() == WL_CONNECTED) Blynk.run();

timer.run();

}

```

ДОДАТОК В. Код diagram.json

```
{  
  
  "version": 1,  
  
  "author": "Vozniak Serhii",  
  
  "editor": "wokwi",  
  
  "parts": [  
  
    { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {} },  
  
    {  
  
      "type": "wokwi-dht22",  
  
      "id": "dht",  
  
      "top": -76.5,  
  
      "left": 426.6,  
  
      "attrs": { "temperature": "27.3" }  
  
    },  
  
    { "type": "wokwi-photoresistor", "id": "ldr", "top": 200, "left": 250, "attrs": {} },  
  
    { "type": "wokwi-pir-motion-sensor", "id": "pir", "top": 109.6, "left": 405.42,  
      "attrs": {} },  
  
    {  
  
      "type": "wokwi-lcd1602",  
  
      "id": "lcd",  
  
      "top": -204.8,
```

					ІТС.4КІ.0126.02-ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	ДОДАТОК В		
Розроб.		Возняк С. В.					
Керівник		Донченко В.Ю.					
Реценз.		Козуб Ю.Г.					
Н. Контр.							
Зав. каф.		Семенов М.А.					
					Лім.	Арк.	Акрушів
						70	5
					ЛНУ		
					Кафедра ІТС, Гр.4КІ		

```

"left": -100,

"attrs": { "pins": "i2c" }

},

{

"type": "wokwi-relay-module",

"id": "rel1",

"top": 250,

"left": -250,

"attrs": { "name": "Heater" }

},

{

"type": "wokwi-relay-module",

"id": "rel2",

"top": 450,

"left": -250,

"attrs": { "name": "Light" }

},

{

"type": "wokwi-led",

"id": "led_h",

"top": 230,

"left": -350,

```

					ITC.4KI.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		65

```

"attrs": { "color": "blue", "label": "Heat" }

},

{

"type": "wokwi-led",

"id": "led_1",

"top": 430,

"left": -350,

"attrs": { "color": "yellow", "label": "Light" }

},

{

"type": "wokwi-resistor",

"id": "res1",

"top": 270,

"left": -310,

"attrs": { "value": "220" }

},

{

"type": "wokwi-resistor",

"id": "res2",

"top": 470,

"left": -310,

"attrs": { "value": "220" }

```

					ITC.4KI.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		66

```

},

{
  "type": "wokwi-photoresistor-sensor",
  "id": "ldr1",
  "top": 51.2,
  "left": -248.8,
  "attrs": {}
}

],

"connections": [

[ "esp:TX", "$serialMonitor:RX", "", [] ],

[ "esp:RX", "$serialMonitor:TX", "", [] ],

[ "esp:GND.1", "dht:GND", "black", [ "v0" ] ],

[ "esp:3.3V.1", "dht:VCC", "red", [ "v0" ] ],

[ "esp:15", "dht:SDA", "orange", [ "v0" ] ],

[ "esp:GND.1", "pir:GND", "black", [ "v0" ] ],

[ "esp:3.3V.1", "pir:VCC", "red", [ "v0" ] ],

[ "esp:13", "pir:OUT", "purple", [ "v0" ] ],

[ "esp:21", "lcd:SDA", "green", [ "v0" ] ],

[ "esp:22", "lcd:SCL", "green", [ "v0" ] ],

[ "esp:5V", "lcd:VCC", "red", [ "v0" ] ],

[ "esp:GND.1", "lcd:GND", "black", [ "v0" ] ],

```

					ITC.4KI.0126.02-ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		67

```

[ "esp:2", "rel1:IN", "red", [ "v0" ] ],
[ "esp:4", "rel2:IN", "yellow", [ "v0" ] ],
[ "esp:GND.1", "rel1:GND", "black", [ "v0" ] ],
[ "esp:GND.1", "rel2:GND", "black", [ "v0" ] ],
[ "esp:5V", "rel1:VCC", "red", [ "v0" ] ],
[ "esp:5V", "rel2:VCC", "red", [ "v0" ] ],
[ "esp:2", "res1:1", "blue", [ "v0" ] ],
[ "res1:2", "led_h:A", "blue", [ "v0" ] ],
[ "esp:GND.1", "led_h:C", "black", [ "v0" ] ],
[ "esp:4", "res2:1", "yellow", [ "v0" ] ],
[ "res2:2", "led_l:A", "yellow", [ "v0" ] ],
[ "esp:GND.1", "led_l:C", "black", [ "v0" ] ],
[ "ldr1:GND", "esp:GND.1", "black", [ "h28.8", "v76.4" ] ],
[ "ldr1:VCC", "esp:3.3V.1", "red", [ "h10", "v-40" ] ],
[ "ldr1:AO", "esp:34", "blue", [ "v0" ] ]
],
"dependencies": {}
}

```

ДОДАТОК С. libraries.txt

Blynk

DHT sensor library

Adafruit Unified Sensor

LiquidCrystal I2C

					ІТС.4КІ.0126.02-ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата	ДОДАТОК С			Лім.	Арк.	Акрушів		
Розроб.		Возняк С. В.									75	1
Керівник		Донченко В.Ю.						ЛНУ Кафедра ІТС, Гр.4КІ				
Реценз.		Козуб Ю.Г.										
Н. Контр.												
Зав. каф.		Семенов М.А.										