

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій  
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Розроблення системи автоматизації моніторингу стану

продукції в промислових холодильниках

(тема)

Виконав:

здобувач 4 року навчання,

групи АКТАКІТ-21-2

Нікіта ЛИТОЧКІН

(власне ім'я прізвище)

Спеціальності 151 Автоматизація та  
комп'ютерно-інтегровані технології

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Автоматизація та  
комп'ютерно-інтегровані технології

(повна назва освітньої програми)

Керівник доцент Кирило ХРУСТАЛЬОВ

(посада, власне ім'я прізвище)

Допускається до захисту

Зав. кафедри КІТАР

Ігор НЕВЛЮДОВ  
(підпис) (власне ім'я прізвище)

2025р.

# ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Автоматики і комп'ютеризованих технологій  
Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки  
Рівень вищої освіти перший (бакалаврський)  
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології  
Тип програми освітньо-професійна  
Освітня програма Автоматизація та комп'ютерно-інтегровані технології  
(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри КІТАР \_\_\_\_\_  
(підпис)

« 28 » квітня 2025 р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Литочкіну Никіті Олеговичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення системи автоматизації моніторингу стану продукції в промислових холодильниках

затверджена наказом по університету від “ 19 ” травня 2025р.№ 390 Ст.

2. Термін подання роботи “ 20 ” липня 2025р.

3. Вихідні дані до роботи 3.1 Датчик температури DS18B20 – 3.3-5 В;

3.2 Датчик освітлення – ВН1750 3.3 В; 3.3 Датчик газу – MQ-4, 5 В;

3.4 Мікроконтролер – ESP32 на базі CH340C; 3.5 Програмне забезпечення – Arduino IDE; 3.6 Серверна частина – Flask; 3.7 База даних – SQLite;

3.8 Клієнтський веб-інтерфейс – HTML, CSS, JavaScript, Chart.js;

3.9 Протокол передачі даних – HTTP; 3.10 Електроживлення – через USB (5В);

3.11 Wi-Fi мережа для передачі даних у локальній мережі;

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

4.1 Вступ;

4.2 Аналіз сучасних рішень систем моніторингу на базі IoT;

4.3 Вибір електронних компонентів для побудови системи моніторингу;

4.4 Розробка програмного забезпечення системи моніторингу на базі ESP32;

4.5 Заходи і розрахунки для забезпечення безпечних умов праці;

4.6 Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій Демонстраційний матеріал представлений у форматі презентації PowerPoint (\*.ppt) – 10 с. формату А4

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз існуючих систем моніторингу параметрів середовища	28.04 – 04.05.25	виконано
2	Вибір апаратних компонентів для реалізації системи на базі ESP32	05.05 – 14.05.25	виконано
3	Розробка програмного забезпечення для збору та передачі даних з сенсорів	15.05 – 22.05.25	виконано
4	Розробка серверної частини та веб-інтерфейсу на Flask для візуалізації	23.05 – 31.05.25	виконано
5	Тестування системи та усунення помилок	01.06 – 05.06.25	виконано
6	Заходи і розрахунки для забезпечення безпечних умов праці	06.06 – 10.06.25	виконано
7	Оформлення пояснювальної записки	11.06 – 14.06.25	виконано
8	Подання роботи на перевірку Інтернет-системою StrikePlagiarism	15.06 – 17.06.25	виконано
9	Подання роботи на рецензію	18.06 – 20.06.25	Виконано
10	Подання роботи на підпис зав. кафедри	21.06 – 23.06.25	Виконано
11	Подання кваліфікаційної роботи в ЕК	24.06.2025	Виконано

Дата видачі завдання 28.04.2025р.

Здобувач \_\_\_\_\_ Нікіта ЛИТОЧКІН  
(підпис)

Керівник роботи \_\_\_\_\_ доцент Кирил ХРУСТАЛЬОВ  
(підпис) (посада, власне ім'я прізвище)

Я, Литочкін Нікіта Олегович, як здобувач вищої освіти ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

«20» липня 2025 р.



Нікіта ЛИТОЧКІН

## РЕФЕРАТ

Пояснювальна записка: 62 с., 5 табл., 7 рис., 6 дод., 11 джерел.

ESP32, MONITORING, FLASK, BH1750, MQ-4, DS18B20, IOT, WEB-ІНТЕРФЕЙС, СИСТЕМА ЗБОРУ ДАНИХ.

Мета роботи – проєктування та розробка лабораторного макету системи моніторингу параметрів навколишнього середовища в промислових холодильниках на базі мікроконтролера ESP32, що забезпечує збір, передачу та візуалізацію даних у реальному часі, з метою демонстрації можливостей автоматизованого контролю та підвищення ефективності зберігання продукції.

Об'єкт розробки – це автоматизація контролю параметрів навколишнього середовища у промислових холодильниках.

Предмет розробки – лабораторний макет апаратно-програмного комплексу моніторингу температури, освітлення та концентрації газу в промислових холодильниках на базі мікроконтролера ESP32.

У роботі проведено аналіз існуючих рішень для моніторингу середовища з використанням мікроконтролерів. Обґрунтовано вибір електронних компонентів: датчик температури DS18B20, цифровий датчик освітлення BH1750 та газовий сенсор MQ-4. Розроблено архітектуру апаратної частини та створено програмне забезпечення для ESP32 і Flask-серверу на Python. Реалізовано веб-інтерфейс з можливістю перегляду таблиць та графіків показників. Також виконано моделювання безпечних умов праці при експлуатації холодильних установок.

Отримані результати роботи можна віднести до Цілі сталого розвитку 9 «Промисловість, інновації та інфраструктура».

## **ABSTRACT**

Explanatory note: 62 pages, 5 tables, 7 figures, 6 appendix, 11 references.

**ESP32, MONITORING, FLASK, BH1750, MQ-4, DS18B20, IOT, WEB INTERFACE, DATA ACQUISITION SYSTEM.**

Purpose of the work – to design and implement a monitoring system for temperature, lighting, and gas concentration in industrial refrigeration units based on the ESP32 microcontroller, which ensures timely data collection, wireless transmission, and real-time visualization to improve safety and efficiency of storage processes.

Object of the development – a system for monitoring environmental parameters in industrial refrigeration chambers.

Subject of the development – a hardware and software complex for collecting, transmitting, and visualizing sensor data in real time.

The paper analyzes existing solutions for environmental monitoring using microcontrollers. The selection of electronic components is substantiated: DS18B20 temperature sensor, BH1750 digital light sensor, and MQ-4 gas sensor. The hardware architecture was developed, and software was created for both the ESP32 and the Python-based Flask server. A web interface with the ability to view tables and data graphs was implemented. Measures were also taken to simulate safe working conditions for operating refrigeration systems.

The results of the work can be attributed to Sustainable Development Goal 9 “Industry, Innovation and Infrastructure”.

## ЗМІСТ

Перелік скорочень .....	9
Вступ.....	10
1 Аналіз сучасних систем моніторингу в промислових холодильниках.....	12
1.1 Загальні вимоги до систем моніторингу холодильних установок.....	12
1.2 Системи на основі ПЛК (програмованих логічних контролерів).....	12
1.3 Перспективи впровадження IoT у холодильні системи .....	13
1.4 Порівняння типових сенсорів і протоколів .....	13
1.5 Роль вебінтерфейсів у сучасних системах.....	13
1.6 Огляд стандартних протоколів обміну даними в автоматизованих системах моніторингу .....	14
1.7 Висновки щодо вибору технології .....	17
2 Вибір компонентів .....	18
2.1 Системи моніторингу для промислових холодильників .....	18
2.2 Температурний контроль за допомогою сенсора DS18B20 .....	18
2.3 Вимірювання освітленості за допомогою BH1750.....	19
2.4 Контроль концентрації газу за допомогою датчика MQ-4 .....	20
2.5 Аналіз доступних мікроконтролерів та вибір ESP32 .....	22
2.6 Порівняння обраної архітектури з аналогами .....	23
2.7 Висновок до розділу .....	23
3 Розробка програмного забезпечення системи.....	24
3.1 Програмування мікроконтролера ESP32 .....	24
3.2 Створення серверної частини на Flask.....	25
3.3 Веб-інтерфейс .....	25
3.4 База даних .....	28
3.5 Структура проєкту .....	29
4 Теорія автоматичного управління .....	31
4.1 Вступ до теорії автоматизованого управління.....	31
4.2 Компоненти системи автоматизованого управління.....	31

4.3 Місце сенсорної системи у структурі АСУ .....	32
4.4 Замкнені та розімкнені системи управління .....	32
4.5 Основні характеристики системи управління.....	33
4.6 Висновки до розділу .....	33
5 Тестування та аналіз результатів.....	34
5.1 Методика тестування.....	34
5.2 Тестування сенсорів.....	35
5.3 Передача даних на Flask-сервер .....	36
5.4 Робота вебінтерфейсу .....	37
5.5 Аналіз надійності та стабільності.....	38
6 Заходи для забезпечення безпечних умов праці .....	39
6.1 Аналіз умов праці на робочому місці .....	39
6.2 Промислова безпека при роботі з електронними компонентами .....	39
6.3 Виробнича санітарія в приміщенні .....	40
6.4 Електробезпека.....	41
6.5 Протипожежні заходи.....	41
Перелік джерел посилання .....	44
Додаток А програмний код «index.html» для ПЗ.....	46
Додаток Б програмний код «style.css» для ПЗ .....	48
Додаток В програмний код «script.js» для ПЗ.....	50
Додаток Г програмний код «sketch.ino» для ПЗ .....	55
Додаток Г програмний код «app.py» для ПЗ.....	58
Додаток Д демонстраційний варіант.....	61

## ПЕРЕЛІК СКОРОЧЕНЬ

API – інтерфейс програмування додатків

BH1750 – цифровий датчик освітлення

CSS – каскадні таблиці стилів

DS18B20 – цифровий сенсор температури

ESP32 – мікроконтролер із вбудованим Wi-Fi та Bluetooth

GPIO – порти введення/виведення

HMI – інтерфейс людина-машина

HTML – мова розмітки гіпертексту

HTTP – протокол передачі гіпертексту

JSON – формат обміну даними

MQ – газовий сенсор (наприклад, MQ-4)

UI – інтерфейс користувача

Wi-Fi – бездротова передача даних

## ВСТУП

Сучасна промисловість висуває підвищені вимоги до умов зберігання продукції, особливо в харчовій, фармацевтичній та біотехнологічній сферах. Одним із ключових чинників ефективної логістики та виробничих процесів є забезпечення стабільного контролю параметрів середовища, таких як температурний режим, рівень вологості, освітлення та контроль за наявністю потенційно небезпечних газів у приміщеннях. Порушення цих параметрів може призводити до псування продукції, матеріальних втрат, зниження безпеки праці та, відповідно, до значних фінансових збитків підприємства.

Ураховуючи вищезазначене, особливо актуальним є розроблення надійної та доступної системи моніторингу й контролю параметрів середовища у промислових холодильних камерах. Значущість цієї проблематики посилюється в умовах модернізації існуючих систем та впровадження концепції Інтернету речей (IoT), що передбачає здатність кожного елемента системи зберігання передавати дані, взаємодіяти з іншими пристроями та оперативно інформувати відповідальних осіб про потенційні загрози.

Метою даної бакалаврської кваліфікаційної роботи є створення автоматизованої системи моніторингу температури, освітлення та наявності горючих газів у промислових холодильниках на базі мікроконтролера ESP32. Така система має забезпечувати передачу даних у реальному часі до вебінтерфейсу, де користувач може відслідковувати зміну параметрів середовища, аналізувати статистику та своєчасно реагувати на відхилення від норми.

Для досягнення поставленої мети було сформульовано низку задач:

- проаналізувати існуючі рішення у сфері моніторингу середовища;
- обґрунтувати вибір мікроконтролера, датчиків та інших компонентів;
- розробити схему підключення сенсорів;
- створити програмне забезпечення для збору, обробки та передачі даних;

- реалізувати вебінтерфейс для візуалізації інформації;
- провести тестування системи та оцінити її ефективність;
- оформити звіт за вимогами ДСТУ та науковими вказівками[1][2];
- розглянути питання безпеки та гігієнічних умов під час експлуатації[11].

Об'єкт розробки – це автоматизація контролю параметрів навколишнього середовища у промислових холодильниках.

Предмет розробки – лабораторний макет апаратно-програмного комплексу моніторингу температури, освітлення та концентрації газу в промислових холодильниках на базі мікроконтролера ESP32.

Практична цінність полягає в можливості використання розробленого рішення на реальних об'єктах для підвищення рівня контролю за умовами зберігання та своєчасного виявлення небезпек.

# 1 АНАЛІЗ СУЧАСНИХ СИСТЕМ МОНІТОРІНГУ В ПРОМИСЛОВИХ ХОЛОДИЛЬНИКАХ

## 1.1 Загальні вимоги до систем моніторингу холодильних установок

У сучасному світі контроль кліматичних параметрів у холодильних камерах, особливо у промислових умовах, є критично важливою задачею. Температурний режим, освітлення, рівень загазованості – усе це має безпосередній вплив на якість зберігання продукції, безпеку працівників і довговічність обладнання.

Основними завданнями таких систем є:

- безперервний контроль температури в реальному часі;
- фіксація та збереження даних для подальшого аналізу;
- виявлення та сигналізація про відхилення від допустимих значень;
- забезпечення надійності, автономності й простоти обслуговування;
- інтеграція з іншими системами (наприклад, SCADA, ERP);
- відповідність стандартам охорони праці та санітарним нормам[11].

## 1.2 Системи на основі ПЛК (програмованих логічних контролерів)

Промислові системи автоматизації зазвичай реалізуються на основі ПЛК. Наприклад, Siemens SIMATIC S7, Mitsubishi FX, Schneider Modicon, які застосовуються у складських комплексах, м'ясокомбінатах, морозильних цехах тощо.

Такі системи мають високу надійність, довгий термін служби, можливість масштабування. Вони інтегруються з панелями оператора (HMI), забезпечують точність збору даних і можуть бути частиною великої SCADA-системи.

Однак ці рішення мають високу вартість, вимагають фахівців з програмування ПЛК (мови LAD, FBD, STL), складного введення в експлуатацію та тривалого процесу налагодження.

### 1.3 Перспективи впровадження IoT у холодильні системи

Розвиток Інтернету речей (IoT) дав змогу значно спростити створення систем моніторингу. Мікроконтролери з підтримкою Wi-Fi або Ethernet, такі як ESP32, можуть слугувати центральними вузлами для зчитування сенсорної інформації та передавання її на локальний сервер чи хмарне сховище [8].

До основних переваг таких систем належать:

- модульність (можна додавати нові сенсори або вузли);
- низька вартість (ціна плати ESP32 – кілька доларів);
- гнучкість у створенні вебінтерфейсу;
- збереження даних на сервері або в базі даних;
- можливість автономної роботи;
- підтримка популярних протоколів (HTTP, MQTT, WebSocket).

Сучасні ESP32 мають 2 ядра, великий обсяг оперативної пам'яті (до 512 КБ), підтримку різноманітних бібліотек і просту мову програмування (C++ через Arduino IDE або ESP-IDF).

### 1.4 Порівняння типових сенсорів і протоколів

Для цілей кваліфікаційної роботи обрані недорогі та доступні сенсори:

- DS18B20 – цифровий температурний сенсор з 1-Wire протоколом;
- MQ-4 – сенсор виявлення метану, пропану, природного газу;
- BH1750 – цифровий сенсор освітлення з I<sup>2</sup>C інтерфейсом.

Вони забезпечують достатню точність для моніторингу умов у промисловому холодильнику та легко інтегруються з ESP32 без потреби в додаткових модулях [9].

### 1.5 Роль вебінтерфейсів у сучасних системах

Сучасні користувачі очікують, що дані будуть доступні з будь-якого пристрою – комп'ютера, планшета чи смартфона. Саме тому важливу роль відіграє вебінтерфейс.

Створення вебінтерфейсу на основі Flask (Python) є одним із найпоширеніших рішень. Flask дозволяє обробляти HTTP-запити від ESP32, зберігати дані в базу (наприклад, SQLite або MySQL) та виводити їх у вигляді таблиць, графіків, звітів.

Такий підхід дає змогу:

- спостерігати за змінами температури, рівнем газу або освітлення в реальному часі;
- переглядати історію значень;
- автоматично оновлювати вебсторінку без перезавантаження (через JavaScript);
- реалізувати керування пристроями (наприклад, вмикання\вимикання вентиляції).

## 1.6 Огляд стандартних протоколів обміну даними в автоматизованих системах моніторингу

Передавання даних між пристроями в системі моніторингу є критично важливою складовою. Існує багато протоколів, які дозволяють надійно і стабільно обмінюватися даними між сенсорами, контролерами, серверами та користувацькими інтерфейсами. Вибір протоколу залежить від завдань системи, кількості пристроїв, швидкості передавання даних та вимог до безпеки.

HTTP (Hypertext Transfer Protocol) – найпоширеніший протокол обміну даними у вебсередовищі. Він використовується для взаємодії між клієнтом (наприклад, ESP32) та сервером (наприклад, Flask). У системах моніторингу цей протокол застосовується для:

- надсилання даних у форматі JSON з мікроконтролера на сервер;
- отримання відповіді або команд;
- оновлення вебінтерфейсу.

Порівняння наведено у таблиці 1.1.

Таблиця 1.1 – Переваги та недоліки протоколу HTTP

Переваги	Недоліки
Простота реалізації	Відсутність підтримки реального часу
Велика кількість бібліотек	Кожен запит ініціюється клієнтом
Зрозуміла структура запитів (GET, POST, PUT, DELETE)	Низька ефективність при великій кількості клієнтів або частих запитах

MQTT (Message Queuing Telemetry Transport) – легковаговий протокол публікації/підписки, який широко використовується в системах IoT.

Принцип дії:

- пристрої (ESP32) підписуються на певні "топіки";
- інші пристрої або сервери публікують повідомлення у ці топіки;
- брокер MQTT (наприклад, Mosquitto) розсилає повідомлення всім підписаним клієнтам.

Опис переваг та недоліків наведено у таблиці 1.2.

Таблиця 1.2 – Переваги та недоліки протоколу MQTT

Переваги	Недоліки
Низьке споживання	Необхідність використання окремого MQTT-брокера
Підтримка реального часу	Складніша логіка реалізації у порівнянні з HTTP
Ефективність при великій кількості вузлів	Не призначений для передавання великих обсягів даних

Modbus – один з найстаріших і найпопулярніших промислових протоколів. Існує декілька варіантів: Modbus RTU (RS-485), Modbus ASCII, Modbus TCP/IP.

Особливості:

- орієнтований на зчитування/запис регістрів;
- використовується в ПЛК, HMI, SCADA;
- надійний і перевірений десятиліттями протокол.

Опис переваг та недоліків наведено у таблиці 1.3.

Таблиця 1.3 – Переваги та недоліки протоколу Modbus

Переваги	Недоліки
Стандартизований протокол	Складність реалізації в мікроконтролерах без готових бібліотек
Підтримується більшістю промислових пристроїв	Відносно повільна передача даних
Можливість створення каскадних мереж	Обмежена адресація

WebSocket – це двосторонній протокол зв'язку, який дозволяє відкривати постійне з'єднання між клієнтом і сервером.

У таблиці 1.4 наведено порівняння переваг та недоліків протоколу.

Таблиця 1.4 – Переваги та недоліки протоколу WebSocket

Переваги	Недоліки
Оновлення даних у режимі реального часу без постійного опитування сервера	Складніша реалізація, ніж HTTP
Ефективна передача інформації	Іноді не підтримується в обмежених середовищах
Підходить для відображення графіків, повідомлень, попереджень	Постійне з'єднання може навантажувати сервер при великій кількості клієнтів

### 1.7 Висновки щодо вибору технологій

У межах цієї кваліфікаційної роботи було обрано HTTP для взаємодії між мікроконтролером ESP32 і сервером Flask, що зберігає дані у базу та виводить їх у вебінтерфейс. Це забезпечує достатню швидкодію, простоту реалізації, підтримку JSON-формату та легку інтеграцію з HTML/JS.

У разі масштабування системи до кількох десятків чи сотень сенсорів доцільно розглянути перехід на MQTT або WebSocket, щоб забезпечити реальний час, зменшити навантаження на сервер та уникнути постійних HTTP-запитів.

## 2 ВИБІР КОМПОНЕНТІВ

У цьому розділі здійснено аналіз сучасних технологій моніторингу параметрів навколишнього середовища, що застосовуються у промислових холодильних установках. Розглянуто існуючі технічні рішення для вимірювання температури, рівня освітленості та концентрації горючих газів. Також обґрунтовано вибір відповідних сенсорів та мікроконтролера ESP32 як основного обчислювального модуля системи контролю.

### 2.1 Системи моніторингу для промислових холодильників

У промислових холодильних установках забезпечення належного контролю температурного режиму, своєчасного виявлення витоків горючих газів та рівня освітленості є надзвичайно важливими завданнями. Ці параметри мають суттєвий вплив на якість зберігання продукції, енергоефективність функціонування обладнання та загальний рівень безпеки експлуатації системи.

Більшість наявних рішень з моніторингу параметрів докілья реалізовані на базі комерційних систем із закритою архітектурою, високою вартістю впровадження та обмеженою адаптивністю до змін користувацьких потреб. Крім того, їх обслуговування зазвичай потребує спеціалізованих знань і додаткових витрат. У контексті стрімкого розвитку мікроконтролерної техніки, сенсорних модулів та бездротових засобів передачі даних з'явилася можливість створення економічно доступних і гнучких систем з відкритою архітектурою, здатних до інтеграції як у локальну, так і в хмарну інфраструктуру.

### 2.2 Температурний контроль за допомогою сенсора DS18B20

Одним із ключових параметрів, що підлягають контролю в холодильних системах, є температура. Для її вимірювання було обґрунтовано вибір

цифрового сенсора DS18B20, що функціонує за протоколом OneWire. Це дає змогу підключати кілька сенсорів до одного порту GPIO мікроконтролера, спрощуючи побудову масштабованих вимірювальних систем. Зображення сенсора наведено на рисунку 2.1.



Рисунок 2.1 – Датчик DS18B20

Переваги DS18B20:

- висока точність ( $\pm 0.5^{\circ}\text{C}$  в діапазоні  $-10\dots+85^{\circ}\text{C}$ );
- можливість роботи на великій довжині проводу;
- цифровий вихід, що не потребує калібрування;
- широкий температурний діапазон ( $-55^{\circ}\text{C}\dots+125^{\circ}\text{C}$ );
- простота підключення (лише один пін даних з підтягувальним резистором).

Цей датчик чудово підходить для вбудованих систем, зокрема в умовах холодильників, де стабільна робота при низьких температурах є критично важливою.

### 2.3 Вимірювання освітленості за допомогою BH1750

Для контролю внутрішнього освітлення було використано цифровий сенсор освітленості BH1750. На відміну від аналогових фоторезисторів,

ВН1750 видає значення в люксах (lux) та підключається через I<sup>2</sup>C-шину, що забезпечує високу точність і стабільність вимірювань [9]. Датчик зображено на рисунку 2.2.



Рисунок 2.2 – Зображення датчику ВН1750

Основні характеристики ВН1750:

- діапазон: від 1 до 65535 люкс;
- цифровий вихід без необхідності обробки сигналу;
- підтримка режиму високої роздільної здатності (1 lx);
- низьке енергоспоживання (0.12 мА);
- нечутливість до температурних змін порівняно з аналоговими сенсорами.

Завдяки цифровому інтерфейсу, ВН1750 легко інтегрується в системи на базі ESP32, дозволяючи передавати точні дані без втрат та спотворень, які часто зустрічаються в аналогових схемах.

#### 2.4 Контроль концентрації газу за допомогою датчика MQ-4

MQ-4 – напівпровідниковий сенсор для визначення наявності метану (CH<sub>4</sub>) та інших горючих газів. Він має як аналоговий, так і цифровий вихід, що

дозволяє здійснювати як кількісний, так і пороговий контроль концентрації. Датчик зображено на рисунку 2.3.



Рисунок 2.3 – Зображення датчику MQ-4

Характеристики MQ-4:

- виявлення:  $\text{CH}_4$ , LPG, бутану, пропану;
- напруга живлення: 5 В;
- час прогріву: до 24 годин при першому використанні;
- чутливість до низьких концентрацій метану;
- вихід: аналоговий сигнал пропорційний концентрації газу, цифровий вихід – при перевищенні встановленого порогу.

MQ-4 є недорогим, легко доступним сенсором для побудови системи сигналізації витоків газу. У нашому проєкті він використовується для попередження про небезпечні концентрації метану в середовищі холодильного агрегата, що може бути пов'язано з витоким або неправильною вентиляцією [2].

## 2.5 Аналіз доступних мікроконтролерів та вибір ESP32

Для збору даних з вищенаведених сенсорів і передачі їх на сервер потрібен потужний мікроконтролер із підтримкою бездротового зв'язку. Серед варіантів було розглянуто:

1. Arduino Uno – недостатньо ресурсів для одночасної роботи з Wi-Fi і декількома сенсорами;
2. ESP8266 – обмежена кількість пінів, менше оперативної пам'яті;
3. ESP32 – оптимальний вибір за всіма критеріями.

У таблиці 2.1 наведено порівняння цих трьох мікроконтролерів.

Таблиця 2.1 – Порівняння мікроконтролерів

Характеристика	Arduino UNO	ESP8266	ESP32
Тактова частота	16 МГц	80 МГц	До 240 МГц
ОЗУ	2 КБ SRAM	50 КБ SRAM	520 КБ SRAM
Wi-Fi	Немає	Вбудований	Вбудований
Bluetooth	Немає	Немає	Вбудований
Придатність до проекту	Недостатньо ресурсів	Обмежена кількість пінів	Оптимальний вибір

ESP32 має ряд наступних переваг:

- 2 ядра, до 240 МГц;
- 520 КБ SRAM;
- вбудований Wi-Fi та Bluetooth;
- підтримка протоколів OneWire, I<sup>2</sup>C, SPI, UART;
- багато GPIO-пінів;
- низьке енергоспоживання.

Це дозволяє будувати складні та гнучкі системи моніторингу без потреби в додатковому апаратному забезпеченні. Крім того, ESP32 легко інтегрується з веб-сервером або базою даних через Wi-Fi.

## 2.6 Порівняння обраної архітектури з аналогами

У порівнянні з комерційними системами моніторингу, запропоноване рішення має такі переваги:

- відкритість архітектури;
- низька вартість компонентів;
- можливість гнучкої модернізації;
- локальна або хмарна обробка даних;
- простота розгортання веб-інтерфейсу;
- відсутність потреби в ліцензіях.

Недоліками можуть бути відсутність промислового корпусу та потреба у початковому налаштуванні, але ці фактори не критичні для дослідницької або прототипної системи.

## 2.7 Висновок до розділу

У результаті проведеного аналізу сучасних технічних рішень було обґрунтовано вибір трьох сенсорних модулів – DS18B20 для контролю температури, BH1750 для вимірювання рівня освітленості та MQ-4 для виявлення концентрації горючих газів. У поєднанні з мікроконтролером ESP32 ці компоненти утворюють функціонально завершену апаратну архітектуру системи. Запропоноване рішення вирізняється простотою реалізації, невисокою вартістю та можливістю масштабування, що забезпечує ефективність його застосування для моніторингу параметрів навколишнього середовища у промислових холодильних установках [8].

## 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

Одним із фундаментальних етапів розробки системи моніторингу параметрів у промислових холодильних установках є створення програмного забезпечення. Саме програмна складова забезпечує обробку даних із сенсорних пристроїв, передачу інформації на сервер, збереження у базі даних, а також реалізацію візуалізації через веб-інтерфейс у браузері. У даному розділі розглядається логіка функціонування мікроконтролера ESP32, архітектура серверної частини, особливості структури бази даних, а також принципи реалізації клієнтського інтерфейсу.

### 3.1 Програмування мікроконтролера ESP32

Програмна логіка, що виконується на мікроконтролері, реалізована із використанням середовища розробки Arduino IDE, яке підтримує мову програмування C++. Для зчитування температурних показників застосовується бібліотека DallasTemperature, що функціонує поверх бібліотеки OneWire. Сенсор освітленості BH1750 підключений через інтерфейс I2C і взаємодіє з відповідною бібліотекою BH1750. Газовий сенсор MQ-4 опитується як по аналоговому, так і по цифровому каналу.

У основному циклі програми (loop) здійснюється опитування сенсорів з інтервалом у 5 секунд. Отримані значення (температура, рівень освітленості, концентрація газу) формуються у вигляді JSON-структури та надсилаються на сервер за допомогою HTTP POST-запиту. Перед цим мікроконтролер встановлює з'єднання з мережею Wi-Fi.

Також передбачено виведення діагностичної інформації в серійну консоль, що забезпечує можливість налагодження програмного забезпечення в процесі розробки. Повний текст коду наведено у додатку Г.

### 3.2 Створення серверної частини на Flask

Серверна частина реалізована на мові Python із використанням мікрофреймворку Flask. Сервер відповідає за приймання POST-запитів із даними від ESP32, обробку отриманої JSON-інформації та її збереження в БД.

Для взаємодії з базою даних використовується бібліотека SQLAlchemy, яка надає ORM (об'єктно-реляційне відображення) для роботи з таблицями. Записи зберігаються у базі даних SQLite, що є простим і зручним рішенням для локального або невеликого серверного проєкту.

Основні функції сервера:

- обробка маршруту /data для приймання даних з мікроконтролера;
- збереження записів у таблиці з полями: час, температура, освітленість, концентрація газу;
- надання API для клієнтської частини (вивантаження останніх даних, отримання всіх записів, очищення бази).

Також реалізовано механізм CORS для взаємодії з клієнтом, налаштовано автоматичне оновлення сторінки через JavaScript, та передбачено кнопку очищення бази. Повний код наведено у додатку Д.

### 3.3 Веб-інтерфейс

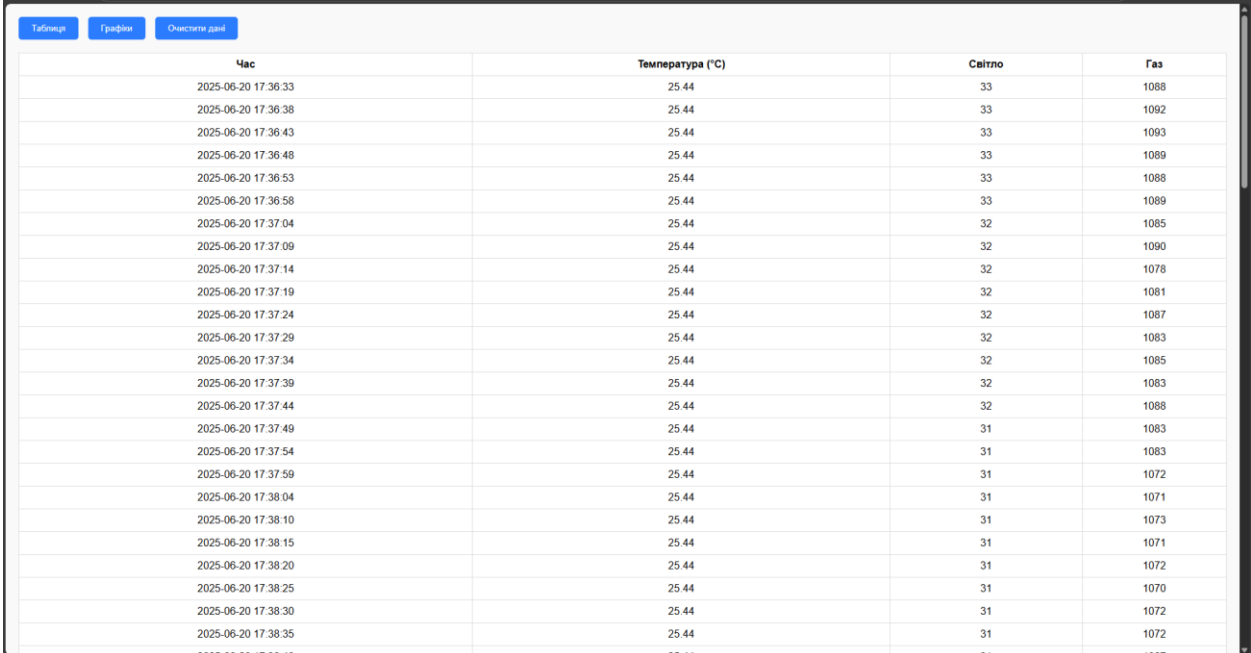
Веб-інтерфейс користувача розроблено з використанням технологій HTML, CSS та JavaScript. Він забезпечує можливість перегляду останніх показників із датчиків у режимі реального часу, а також доступ до історії вимірювань. Повний код HTML наведено у додатку Б, а повний код JavaScript – у додатку Г. Інтерфейс складається з двох вкладок:

- таблиця значень (з останніми записами);
- графіки (побудовані за допомогою бібліотеки Chart.js).

Таблицю значень відображає останні записи, отримані з сенсорів, у вигляді таблиці. Кожен рядок таблиці містить часову мітку, температуру (з

DS18B20), рівень освітлення (з BH1750) та рівень метану (з MQ-4). Дані автоматично оновлюються через певні інтервали часу за допомогою AJAX-запитів, що дозволяє відображати свіжу інформацію без перезавантаження сторінки. В таблиці також реалізована можливість прокрутки, що дозволяє переглядати більше записів у компактному вигляді.

Вигляд цієї вкладки подано на рисунку 3.1. На ньому можна побачити чітку структуру представлення даних, збереження їхньої читабельності та зручну навігацію.



Час	Температура (°C)	Світло	Газ
2025-06-20 17:36:33	25.44	33	1088
2025-06-20 17:36:38	25.44	33	1092
2025-06-20 17:36:43	25.44	33	1093
2025-06-20 17:36:48	25.44	33	1089
2025-06-20 17:36:53	25.44	33	1088
2025-06-20 17:36:58	25.44	33	1089
2025-06-20 17:37:04	25.44	32	1085
2025-06-20 17:37:09	25.44	32	1090
2025-06-20 17:37:14	25.44	32	1078
2025-06-20 17:37:19	25.44	32	1081
2025-06-20 17:37:24	25.44	32	1087
2025-06-20 17:37:29	25.44	32	1083
2025-06-20 17:37:34	25.44	32	1085
2025-06-20 17:37:39	25.44	32	1083
2025-06-20 17:37:44	25.44	32	1088
2025-06-20 17:37:49	25.44	31	1083
2025-06-20 17:37:54	25.44	31	1083
2025-06-20 17:37:59	25.44	31	1072
2025-06-20 17:38:04	25.44	31	1071
2025-06-20 17:38:10	25.44	31	1073
2025-06-20 17:38:15	25.44	31	1071
2025-06-20 17:38:20	25.44	31	1072
2025-06-20 17:38:25	25.44	31	1070
2025-06-20 17:38:30	25.44	31	1072
2025-06-20 17:38:35	25.44	31	1072

Рисунок 3.1 – Вкладена сторінка із таблицею значень

Для забезпечення візуального аналізу динаміки змін параметрів реалізовано вкладку з графічним відображенням даних, побудованих із використанням бібліотеки Chart.js. Ця широко застосовувана бібліотека дозволяє створювати адаптивні та інтерактивні графіки з сучасним дизайном. На відповідній вкладці відображається окремий графік для кожного типу параметрів: температури, освітленості та рівня газу. Графіки формуються на основі останніх 50 або 100 вимірювань (залежно від налаштувань) та автоматично оновлюються при надходженні нових даних. Інтерфейс вкладки з графіками наведено на рисунку 3.2. Завдяки такому відображенню можна

чітко простежити зміни параметрів у часовому розрізі, що сприяє своєчасному виявленню відхилень або потенційно небезпечних тенденцій у роботі холодильного обладнання.

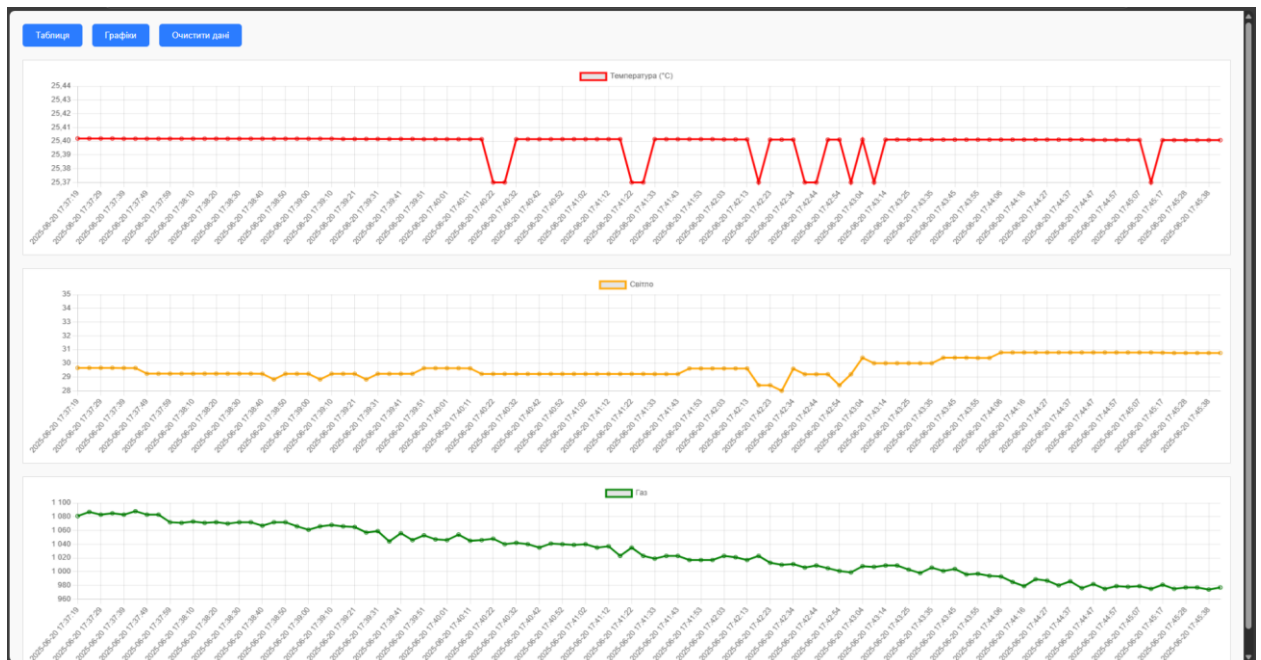


Рисунок 3.2 – Вкладена сторінка із графіками

CSS-оформлення винесено у окремий файл, що сприяє спрощенню підтримки та масштабування дизайну. Повний код стилів наведено у додатку В. Для зручності користувача вгорі сторінки розташовано меню для перемикання між вкладками, а також кнопку для очищення всіх даних у базі. Запити до серверної частини реалізовано за допомогою AJAX (fetch), що забезпечує динамічне оновлення контенту без необхідності повного перезавантаження сторінки.

З метою покращення взаємодії користувача з веб-інтерфейсом реалізовано автоматичне оновлення даних на сторінці, а також функцію очищення бази даних. Це дає змогу відображати актуальні значення сенсорів у режимі реального часу без необхідності ручного оновлення сторінки.

Побудова графіків здійснюється з використанням бібліотеки Chart.js, при цьому дані для візуалізації отримуються через REST API Flask.

### 3.4 База даних

База даних побудована на SQLite. Вона містить одну основну таблицю, у якій зберігаються всі отримані значення. Поля таблиці: ID запису, час, температура, освітленість, рівень газу.

SQLite обрана за її простоту, мінімальні вимоги до налаштувань та повну сумісність із Python через SQLAlchemy. У майбутньому базу легко масштабувати до більш продуктивної системи (наприклад, MySQL).

Інформація зберігається в SQLite – це легка реляційна база даних, яка чудово підходить для локальних проєктів. Структура таблиці складається з:

- temperature – температура в градусах Цельсія;
- light – освітленість у люксах;
- gas – значення з аналогового виходу MQ-4;
- timestamp – час надходження даних

Збереження даних відбувається автоматично – кожен новий запис, що надходить із ESP32 через HTTP-запит, обробляється Flask-сервером і додається до бази. Це дозволяє не лише відображати значення в реальному часі, але й накопичувати історичну інформацію для подальшого аналізу або експорту.

На рисунку 3.3 показано приклад кількох записів у таблиці. Видно, як кожному показнику відповідає свій точний час фіксації, що є важливим для коректного побудування графіків та ведення журналу подій. База даних є основою для зворотного запиту інформації у веб-інтерфейсі: саме з неї витягуються останні записи для таблиці значень та історія для побудови графіків.

The screenshot shows a SQL database browser window with a table named 'sensor\_data'. The table has the following columns: id, temperature, light, gas, and timestamp. The data is as follows:

id	temperature	light	gas	timestamp
1	25.44	30	1409	2025-06-20 17:12:31.928209
2	25.5	29	1393	2025-06-20 17:12:37.007024
3	25.5	29	1380	2025-06-20 17:12:42.095964
4	25.5	29	1376	2025-06-20 17:12:47.169916
5	25.5	29	1360	2025-06-20 17:12:52.243633
6	25.5	28	1360	2025-06-20 17:12:57.322903
7	25.5	28	1355	2025-06-20 17:13:02.404323
8	25.5	26	1346	2025-06-20 17:13:07.492811
9	25.5	25	1344	2025-06-20 17:13:12.567068
10	25.5	24	1345	2025-06-20 17:13:17.639108
11	25.5	28	1328	2025-06-20 17:13:22.713328
12	25.5	30	1331	2025-06-20 17:13:27.796096
13	25.5	30	1305	2025-06-20 17:13:32.879025
14	25.5	30	1306	2025-06-20 17:13:37.940635
15	25.5	30	1295	2025-06-20 17:13:43.009629
16	25.44	30	1290	2025-06-20 17:13:48.093112
17	25.5	30	1277	2025-06-20 17:13:53.166002
18	25.5	30	1270	2025-06-20 17:13:58.238034
19	25.44	29	1271	2025-06-20 17:13:03.310188
20	25.5	29	1254	2025-06-20 17:13:08.385106
21	25.5	29	1254	2025-06-20 17:13:13.464131
22	25.5	30	1247	2025-06-20 17:13:18.544965
23	25.5	30	1248	2025-06-20 17:13:23.623692
24	25.44	30	1242	2025-06-20 17:13:28.701895
25	25.5	30	1227	2025-06-20 17:13:33.784999
26	25.5	30	1230	2025-06-20 17:13:38.855164
27	25.5	30	1217	2025-06-20 17:13:43.924995
28	25.44	30	1220	2025-06-20 17:13:48.996043
29	25.5	30	1207	2025-06-20 17:13:54.077885
30	25.44	30	1214	2025-06-20 17:13:59.169102
31	25.5	30	1210	2025-06-20 17:13:04.247586
32	25.5	30	1200	2025-06-20 17:13:09.331723
33	25.5	30	1195	2025-06-20 17:13:14.424430

Рисунок 3.3 – Приклад роботи бази даних

Таким чином, реалізована база даних виконує роль надійного сховища сенсорної інформації та забезпечує повну інтеграцію між апаратною частиною системи та програмною логікою веб-інтерфейсу.

### 3.5 Структура проєкту

Реалізацію файлової системи наведено на рисунку 3.4. Така організація дозволяє розділити логіку системи за функціональними модулями, що полегшує підтримку, тестування та подальший розвиток.

У межах проєкту структура була розроблена з урахуванням принципів модульності та масштабованості. Основні елементи проєкту були винесені у окремі директорії та файли. Наприклад, основний серверний скрипт `app.py` відповідає за запуск Flask-серверу, прийом даних з пристрою на ESP32, збереження їх у базу даних і передачу у веб-інтерфейс. Статичні файли, такі як таблиці стилів (CSS), JavaScript-логіка для оновлення сторінки та побудови

графіків, а також HTML-шаблони, були організовані в окремі каталоги `static` і `templates`.

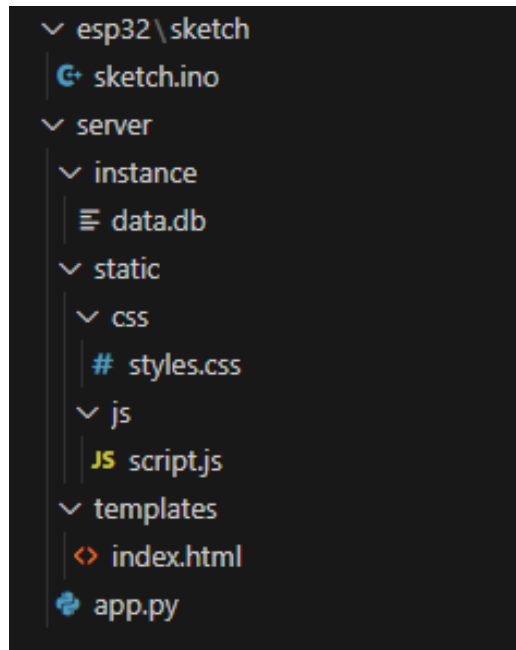


Рисунок 3.4 – Реалізація файлової системи

## 4 ТЕОРІЯ АВТОМАТИЧНОГО УПРАВЛІННЯ

### 4.1 Вступ до теорії автоматизованого управління

Автоматизоване управління (АУ) – це процес впливу на об'єкт за допомогою технічних засобів з метою досягнення або підтримання заданого режиму функціонування без безпосередньої участі людини. У промисловості такі системи використовуються для підвищення продуктивності, якості, безпеки та ефективності.

Теорія автоматизованого управління (ТАУ) – це наукова дисципліна, яка досліджує структуру, властивості та принципи побудови систем управління. Основними об'єктами її дослідження є системи з замкненим або розімкненим циклом, які можуть бути описані за допомогою математичних моделей, структурних схем та характеристик.

### 4.2 Компоненти системи автоматизованого управління

Типова система АУ складається з таких основних елементів:

- Об'єкт керування – система або процес, яким необхідно управляти (у моєму випадку – холодильна камера);
- Задане значення – параметри, які потрібно підтримувати (наприклад, температура до  $+4^{\circ}\text{C}$ , освітленість  $<100$  лк, відсутність газу);
- Регулятор – пристрій, який порівнює фактичне значення параметра з еталонним і формує керуючий вплив (у моєму випадку – мікроконтролер ESP32).
- Сенсори (датчики) – пристрої, які вимірюють поточні параметри середовища та передають їх у регулятор;
- Виконавчі механізми – елементи, що реалізують дію (наприклад, реле для вмикання охолодження чи вентиляції);

– Зворотний зв'язок – канал, який повертає в систему інформацію про результат впливу, щоб система могла скоригувати свою дію.

#### 4.3 Місце сенсорної системи у структурі АСУ

У дипломному проєкті реалізована система моніторингу, що базується на збиранні та аналізі даних від трьох типів датчиків: температурного (DS18B20), освітленості (BH1750) та газового (MQ-4). Всі ці сенсори забезпечують зворотний зв'язок, який у системі АУ є критично важливим.

Регулятор (ESP32) зчитує дані з сенсорів і формує подальші дії: наразі – лише передача на сервер, але потенційно – і активація виконавчих пристроїв (наприклад, подача сигналу на охолодження або тривогу).

Таким чином, навіть у спрощеній реалізації система вже містить ознаки автоматизованого контролю, а при додаванні реле та алгоритмів реагування – може перетворитися на автоматизовану систему управління замкненого типу.

#### 4.4 Замкнені та розімкнені системи управління

Система, яка просто зчитує дані та відображає їх, без впливу на об'єкт, вважається розімкненою. У моїй роботі поки реалізовано саме таку структуру. Але в класичному варіанті ТАУ найчастіше використовуються замкнені системи, в яких є:

- Регулятор – мікроконтролер або ПЛК;
- Зворотний зв'язок – сенсори;
- Алгоритм керування – наприклад, PID-регулятор, або логіка порогового спрацювання;
- Дія – наприклад, включення реле при перевищенні температури.

Така система може не лише вимірювати, а й керувати, і це дозволяє автоматично підтримувати оптимальні параметри.

#### 4.5 Основні характеристики системи управління

У контексті ТАУ, важливо враховувати такі характеристики систем:

- Стійкість – здатність системи повертатися до нормального режиму після збурення;
- Швидкодія – час, за який система реагує на зміну параметра;
- Похибка – різниця між бажаним і фактичним значенням.

У системі, яку було розроблено, хоч поки немає автоматичного керування, вже можна оцінити запізнення в передачі, точність вимірювання, а також частоту оновлення даних. Це стане основою для майбутнього переходу до замкненої схеми з контролем.

#### 4.6 Висновки до розділу

У цьому розділі було розглянуто теоретичні основи автоматизованого управління та їх застосування в межах мого проєкту. Хоча реалізація системи поки має розімкнену структуру, її архітектура повністю відповідає вимогам побудови автоматизованих систем. У майбутньому її легко розширити до замкненого контуру з повноцінним регулюванням і адаптивною поведінкою, що є прямим застосуванням принципів ТАУ в практичному середовищі.

## 5 ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

Розділ присвячений процесу тестування розробленої системи моніторингу температури, освітлення та концентрації газу, побудованої на базі мікроконтролера ESP32. Окрім цього, наведено результати випробувань та сформульовано висновки щодо функціональності, надійності та перспектив подальшого розвитку проєкту.

### 5.1 Методика тестування

Тестування є невід'ємною частиною життєвого циклу будь-якої автоматизованої системи. Основна мета – підтвердити працездатність і відповідність системи технічним вимогам, визначеним на етапі проєктування.

Тестування проводилося за наступними критеріями:

- функціональне тестування сенсорів – визначення точності, стабільності показань, швидкості реакції;
- тестування передачі даних – перевірка з'єднання ESP32 із сервером через Wi-Fi, правильність формату POST-запиту;
- база даних – перевірка запису, читання та очищення таблиці значень;
- інтерфейс користувача – відповідність макету, відображення таблиці та графіків, коректність автооновлення;
- умови роботи – перевірка в середовищах з різними рівнями освітлення, температури та наявності газу.

Технічне середовище тестування:

- ESP32 із чипом CH340C;
- датчики: DS18B20, BH1750, MQ-4;
- локальна Wi-Fi мережа;
- сервер на Flask з базою SQLite;
- перегляд вебінтерфейсу через браузер Chrome, Firefox.

## 5.2 Тестування сенсорів

Мета тестування – перевірити стабільність і точність температурних показників. Для цього було:

- підключено датчик до ESP32 через шину OneWire;
- у циклі зчитувалися дані з інтервалом 5 секунд;
- порівнювалися з показами побутового цифрового термометра.

Результати:

- діапазон вимірювань: від +16 до +28 °C у кімнатних умовах.
- відхилення не перевищувало  $\pm 0.5$  °C.
- температура оновлювалася кожні 5 секунд без пропусків.
- у разі від'єднання датчика система повідомляла про помилку, не перериваючи роботи інших частин.

Переваги:

- висока точність;
- простота інтеграції;
- стабільна робота в довготривалих тестах (більше 8 годин).

ВН1750 – цифровий сенсор освітлення. Ціль – перевірити роботу I2C-сенсора освітлення. Тест проводився в умовах різного світла:

- природне освітлення (відкриті вікна);
- штучне освітлення (лампи денного світла);
- темрява (вимкнене світло).

Результати:

- у темному середовищі: 0–5 lux.
- при денному світлі: 200–400 lux.
- при прямому освітленні: до 1000 lux.
- динаміка зміни значень – плавна, без стрибків.

Переваги:

- цифровий інтерфейс з мінімальною похибкою;
- інтуїтивно зрозумілий діапазон люксів для користувача;

MQ-4 – сенсор метану (газу). Мета тесту – перевірити реакцію на наявність горючого газу, зокрема метану. Для цього:

- проводились виміри при нормальному повітрі (базовий рівень);
- далі тестувалося реагування сенсора після піднесення запальнички (без полум'я).

Результати:

- аналогове значення у нормальних умовах: 250–350 одиниць (на 10-бітному АЦП).
- при виявленні газу: сплески до 900+ одиниць.
- цифровий вихід переключався у «1», коли концентрація перевищувала встановлений поріг.
- відновлення до нормальних значень після провітрювання – близько 20–30 секунд.

Недоліки:

- потрібна калібровка;
- вплив температури та вологості;
- нестабільність перших 2–3 хвилин після вмикання (період прогріву сенсора).

### 5.3 Передача даних на Flask-сервер

Сервер розгорнуто локально на ПК з Windows. ESP32 надсилає JSON-запит через HTTP методом POST. Протестовано:

- підключення до локальної Wi-Fi мережі;
- стабільність з'єднання при безперервній роботі (до 8 годин);
- поведінка при втраті мережі (автоматичне повторне з'єднання);
- правильність структури JSON.

Типовий JSON-пакет наведено нижче:

```
{  
  "temperature": 23.50,  
  "light": 350,  
  "gas": 710  
}
```

Результати:

1. Відповідь сервера: HTTP 200 OK;
2. Запис у SQLite відбувався одразу після надходження запиту;
3. Затримка між передачею та появою даних в інтерфейсі – менше 1 секунди.

#### 5.4 Робота вебінтерфейсу

Вебінтерфейс має дві вкладки: «Таблиця» – список останніх 100 записів, автооновлення кожні 5 с та «Графіки» – візуалізація даних із кожного сенсора з динамічним масштабом. Реалізовано з використанням HTML/CSS/JavaScript + Chart.js.

Перевірено:

- адаптивність сторінки для мобільних пристроїв;
- швидкість відображення нових записів;
- коректність форматування таблиці;
- функціональність кнопки «Очистити дані».

Після натискання кнопки всі записи з бази видаляються, інтерфейс миттєво оновлюється.

## 5.5 Аналіз надійності та стабільності

Тестування в реальних умовах (домашнє середовище з варіативною температурою, освітленням та наявністю газу) засвідчило:

- високу надійність роботи ESP32;
- нульовий рівень збоїв при передачі даних;
- стабільність вебінтерфейсу протягом тривалого часу;
- відсутність утрат або спотворення даних.

## **6 ЗАХОДИ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕЧНИХ УМОВ ПРАЦІ**

Під час розроблення, тестування та експлуатації системи відстеження температури, освітленості та рівня газу на базі ESP32 необхідно дотримуватися нормативів з охорони праці, правил техніки безпеки, санітарно-гігієнічних вимог, а також передбачити заходи щодо електробезпеки та пожежної безпеки. Дотримання вимог охорони праці сприяє запобіганню травм, аварій, виходу обладнання з ладу та забезпечує комфортні умови праці розробника[11].

### **6.1 Аналіз умов праці на робочому місці**

Розробка та тестування розглянутої системи проводилися у лабораторних та офісних умовах, які відповідають вимогам до приміщень із застосуванням електронно-обчислювальної техніки. Робоче місце було обладнане персональним комп'ютером, джерелом живлення 220 В, мікроконтролерами та відповідними датчиками.

Приміщення забезпечувало природне та штучне освітлення, належну вентиляцію, а також доступ до засобів пожежогасіння. Температурний режим підтримувався в межах нормативних значень, зокрема від +18 до +25 °С, рівень шуму не перевищував допустимі норми (до 50 дБ). Організація робочого місця здійснювалася з урахуванням мінімізації навантаження на зір, зменшення впливу шкідливих факторів, таких як електромагнітні випромінювання, а також забезпечення ергономічних умов праці.

### **6.2 Промислова безпека при роботі з електронними компонентами**

При використанні мікроконтролера ESP32, датчиків (DS18B20, BH1750, MQ-4) та інших елементів слід дотримуватись загальних правил безпечної

роботи з низьковольтною електронікою:

- підключення проводиться лише при знеструмленому обладнанні;
- уникати короткого замикання під час монтажу макетної плати;
- не залишати електронні компоненти під напругою без нагляду;
- використовувати стандартні засоби захисту від статичної електрики при роботі з мікросхемами;
- не перевищувати допустимі рівні напруги і струму для підключених датчиків та ESP32.

Особливу увагу слід приділити роботі з датчиком газу MQ-4. Він нагрівається під час роботи, тому не можна торкатися його металевої сітки, слід забезпечити вентиляцію навколо модуля та уникати надмірного накопичення горючих речовин поруч.

### 6.3 Виробнича санітарія в приміщенні

Вимоги виробничої санітарії охоплюють організацію повітрообміну, рівень освітленості, температурний режим, вологість та чистоту на робочому місці. Робоче місце має бути забезпечене достатнім рівнем природного світла або ламп денного освітлення. Поверхня робочого столу повинна регулярно очищуватись від пилу, а також виключається накопичення монтажних відходів, таких як проводи, залишки флюсу тощо.

Відповідно до санітарних норм, час безперервної роботи за комп'ютером не повинен перевищувати 2 години, після чого необхідно робити перерву тривалістю 10–15 хвилин. У цей період рекомендовано виконувати фізичні вправи для очей та кистей рук з метою профілактики перенавантаження та збереження працездатності.

## 6.4 Електробезпека

Обладнання, що використовується, живиться від джерел з напругою до 5 В (через USB або стабілізовані блоки живлення на 3,3/5 В), що не становить значної загрози для життя. Водночас всі підключення до мережі змінного струму 220 В мають виконуватися виключно через справні блоки живлення із подвійною ізоляцією. Рекомендується застосовувати мережеві фільтри або джерела безперебійного живлення для захисту обладнання від перепадів напруги.

Кабелі слід надійно закріплювати з метою запобігання механічним пошкодженням або коротким замиканням. Категорично забороняється експлуатувати несправні електроприлади.

## 6.5 Протипожежні заходи

У разі короткого замикання або перегріву компонентів можливе займання, особливо при тривалому живленні MQ-4. Необхідно:

- тримати поруч вогнегасник (вуглекислотний або порошковий);
- не залишати працююче обладнання без нагляду;
- не використовувати легкозаймісті матеріали поблизу макетів;
- здійснювати регулярну перевірку блоків живлення, кабелів та контактів на наявність перегріву або іскріння.

Усі співробітники, які працюють з електронними пристроями, мають пройти первинний інструктаж з техніки безпеки та бути ознайомленими з правилами евакуації у разі пожежі.

## ВИСНОВКИ

У ході виконання бакалаврської кваліфікаційної роботи було розроблено систему моніторингу параметрів промислового холодильного обладнання на базі мікроконтролера ESP32. Запропонована система забезпечує безперервне вимірювання температури, рівня освітленості та концентрації горючих газів у камері холодильника, передавання отриманих даних через Wi-Fi з подальшим їх збереженням на сервері та відображенням у веб-інтерфейсі.

Система складається з апаратної частини, до якої входять датчики, мікроконтролер та елементи з'єднання, а також програмної частини, що включає скетч для ESP32, серверне програмне забезпечення на базі Flask та клієнтський інтерфейс, реалізований за допомогою HTML, CSS та JavaScript. Застосування температурного датчика DS18B20, світлового датчика BH1750 і газового сенсора MQ-4 дозволяє отримувати достовірну інформацію про параметри середовища всередині холодильного приміщення. Дані зчитуються мікроконтролером ESP32 кожні 5 секунд та передаються у форматі JSON на сервер Flask, який обробляє отримані значення, зберігає їх у базі даних та надає доступ до них через веб-інтерфейс.

Інтерфейс користувача реалізовано з використанням сучасних вебтехнологій, що дозволяє візуалізувати інформацію у вигляді таблиць та графіків. Система підтримує автоматичне оновлення даних та надає можливість очищення збережених показників за допомогою відповідної кнопки, що підвищує зручність використання та робить її придатною для застосування в реальних промислових умовах.

У процесі розробки було проведено аналіз існуючих рішень для моніторингу стану холодильного обладнання, обґрунтовано вибір апаратних компонентів, створено прототип системи, протестовано її працездатність та порівняно отримані результати з очікуваними. Особливу увагу приділено дотриманню безпечних умов праці, аналізу потенційних ризиків при експлуатації пристрою та відповідності розробленого рішення чинним

санітарно-технічним нормам.

Отже, поставлені у роботі завдання було виконано повністю, а визначена мета досягнута. Розроблена система є економічно доцільною, технічно реалізованою та може слугувати основою для подальшої модернізації або масштабування, зокрема шляхом додавання нових сенсорів, інтеграції з мобільними додатками або побудови розподіленої системи моніторингу.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. – Введ. 2015-06-22. – К. Держстандарт України, 2017. – 29 с.
2. Навчальний посібник з підготовки кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології»: Навчальний посібник / І. Ш. Невлюдов, В.А. Андрусевич, О. В. Токарева, С. П. Новоселов, О. В. Сичова. – Харків : Видавництво Іванченка І. С., 2022. – 151 с.
3. Методичні вказівки з підготовки кваліфікаційної роботи бакалавра для студентів усіх форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / Упоряд.: І. Ш. Невлюдов, А. О. Андрусевич, О. В. Токарева, С. П. Новоселов, О. В. Сичова. – Харків: ХНУРЕ, 2022. – 55 с.
4. Основи наукових досліджень : підручник / І. Ш. Невлюдов, Ю. М. Олександров, А. О. Андрусевич, О. О. Чала ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Prague : OKTAN PRINT, 2024. – 468 с.
5. Невлюдов І.Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 1: підручник. Харків: ФОП Панов А.М., 2021. – 604 с.
6. Невлюдов І.Ш. Виробничі процеси та обладнання об'єктів автоматизації. Збірник задач: Навчальний посібник / І.Ш. Невлюдов, А.О. Андрусевич, Г.В. Пономарьова, А.О. Функендорф. Кривий Ріг: КК НАУ. 2018. – 332 с.
7. Орієнтири розвитку ІоТ в Україні: технології та практики впровадження / SHOP GSM.UA [Електронний ресурс]. – Львів, 2019. – 8 с.

[https://shop-gsm.ua/ru/blog/internet-veschej-iot-perspektiva-v-ukraine/?srsltid=AfmBOooumAd282V1YAVnoGg4KJkhswVjYL0WVmQhtZE\\_oXtZQZEPN6kQ&utm\\_source=chatgpt.com](https://shop-gsm.ua/ru/blog/internet-veschej-iot-perspektiva-v-ukraine/?srsltid=AfmBOooumAd282V1YAVnoGg4KJkhswVjYL0WVmQhtZE_oXtZQZEPN6kQ&utm_source=chatgpt.com)

8. Використання датчика BH1750 з ESP32 для вимірювання освітленості / Random Nerd Tutorials [Електронний ресурс]. – 2022. – 1 с. [https://randomnerdtutorials.com/esp32-bh1750-ambient-light-sensor/?utm\\_source=chatgpt.com](https://randomnerdtutorials.com/esp32-bh1750-ambient-light-sensor/?utm_source=chatgpt.com)

9. Моніторинг температури в холодильних камерах та основні вимоги до приміщень зберігання відповідно до вимог GDP / Центр Валідації [Електронний ресурс]. – Київ, 2015. – 12 с. [https://val-center.com/ua/post/temperature-monitoring?utm\\_source=chatgpt.com](https://val-center.com/ua/post/temperature-monitoring?utm_source=chatgpt.com)

110. Методичні вказівки до самостійної роботи з дисципліни «Організація керування умовами праці» / ХНУРЕ; уклад.: Т. Є. Стищенко, Г. В. Пронюк, Н. М. Сердюк. – Харків, 2016. – 340 с. [https://os.nure.ua/wp-content/uploads/2019/05/met\\_zao\\_keruvannja\\_2016.pdf?utm\\_source=chatgpt.com](https://os.nure.ua/wp-content/uploads/2019/05/met_zao_keruvannja_2016.pdf?utm_source=chatgpt.com)

11. Комплекс навчально-методичного забезпечення навчальної дисципліни «Безпека праці в індустрії ІТ-технологій» підготовки освітнього рівня бакалавр усіх спеціальностей та усіх напрямів університету [<http://catalogue.nure.ua/knmz>] / ХНУРЕ; розроб.: Т. Є. Стищенко, Г. В. Пронюк, Н. М. Сердюк. – Харків, 2017. – 122 с.