

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень


Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32
Назва теми

КВРКІ 022047.22.02.45 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»
Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»
Шифр, назва


Освітня програма «Комп'ютерна інженерія та програмування»
Назва

Виконав: студент III курсу, група КІ2-22-2  АНТОН ЯКОВИЩИН
Підпис Ініціали, прізвище

Керівник  Ігор МИХАЛЬЧУК
Підпис, дата Ініціали, прізвище

Нормоконтролер  Тетяна КИСІЛЬ
Підпис, дата Ініціали, прізвище

До захисту допускаю:
зав. кафедри комп'ютерної
інженерії та інформаційних
систем

 Ольга ПАВЛОВА
Підпис Ініціали, прізвище

«19» червня 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

“ 10 ” 01 2025 р.



ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Антон ЯКОВИШИН

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32

Керівник проекту (роботи) Михальчук Ігор Володимирович, асистент

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 07.02.2025 р. № 23

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на кваліфікаційну роботу

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____

Програмно-апаратний засіб управління вентиляцією укриття (апаратна частина) та постановка завдання щодо його удосконалення

Проектування системи керування та обробки сигналів в автоматизації вентиляції укриття (апаратна частина)

Вибір та підбір апаратних компонентів для реалізації програмно-апаратного засобу на базі ESP32

Розробка апаратної частини проекту та тестування прототипу системи вентиляції укриття

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) _____

Архітектура ПЗ проекту

Архітектура ПЗ для кіберфізичної системи

Апаратне забезпечення проекту

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Тетяна КИСІЛЬ, доцент кафедри КПС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КПС		

7. Дата видачі завдання

« 10 » 01 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	10.01.2025	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2025	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2025	виконано
4	Робота над розділом 2 – вибір компонентів для проектування програмно-апаратного засобу керування вентиляцією укриття на базі ESP32. Апаратна частина	01.04.2025	виконано
5	Робота над розділом 3 – проектування системи керування вентиляцією укриття на базі ESP32	29.04.2025	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2025	виконано
7	Попередній захист ВКР	26.05.2025	виконано
8	Захист ВКР на засіданні ЕК	Червень 2025 року	

Студент

Підпис


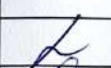
Антон ЯКОВИШИН
Ініціали, прізвище

Керівник роботи

Підпис

Ігор МИХАЛЬЧУК
Ініціали, прізвище

№ р я д к а	Ф о р м а т	Позначення	Найменування	К і л · л и с т і в	№ е кз	П р и м і т к а
			<u>Текстові документи</u>			
1		КВКІ. 022047.22.02.45 ПЗ	Пояснювальна записка	61		
			<u>Графічні матеріали</u>			
2		КВРКІ 022047.22.02.45 Е8	Структурна схема ПЗ проекту	1		
3		КВРКІ 022047.22.02.45 Е8	Електрична схема	1		
4		КВРКІ 022047.22.02.45 Е8	Блок схема	1		

Зм	Ар к	№ докум	Підпис	Дата	КПКІ 022047.22.02.21 ВП			
Розробив		Яковишин		20.06.21	Відомість проекту	Літера	Арку ш	Аркуші в
Перевір.		Михальчук		20.06.21		У	1	1
Н. контр.		Кисіль		20.06.21		ХНУ, КІ2-22-2		
Затв.		.Павлова		20.06.21				

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32».

Автор роботи: Антон ЯКОВИШИН.

Керівник роботи: Михальчук Ігор Володимирович.

Пояснювальна записка: 61 с., 10 рис., 6 дод., 49 джерел.

Графічна частина: 3 креслення.

Метою дипломної роботи є розробка програмно-апаратного засобу, що дозволяє автоматизовано керувати вентиляцією в укритті на базі мікроконтролера ESP32 з урахуванням параметрів середовища та вимог безпеки.

Об'єктом дослідження є системи моніторингу та управління мікрокліматом у захисних спорудах цивільного призначення.

Предметом дослідження є структурна побудова, логіка роботи та взаємодія програмних та апаратних компонентів системи управління вентиляцією на базі ESP32.

У ході дослідження було застосовано метод системного аналізу, вивчено літературу та технічну документацію з питань побудови вбудованих систем, обробки сенсорних даних та розробки IoT-рішень для критичної інфраструктури.



Підпис студента

30.05.2025

Дата

ЗМІСТ

ВСТУП	4
1 АНАЛІЗ СУЧАСНОГО СТАНУ ПРОБЛЕМИ ТА УЗАГАЛЬНЕННЯ ДОСЛІДЖЕНЬ У СФЕРІ КЕРУВАННЯ ВЕНТИЛЯЦІЄЮ УКРИТТІВ...	5
1.1 Історія та розвиток систем вентиляції укриттів	5
1.2 Сучасні підходи до автоматизації систем вентиляції укриттів	7
1.3 Обґрунтування вибору мікроконтролерної платформи для реалізації системи керування вентиляцією укриття	10
1.4 Сенсорні елементи (датчики) для моніторингу параметрів повітря.....	13
1.5 Виконавчі механізми системи керування вентиляцією.....	15
1.6 Засоби індикації та користувацького інтерфейсу	18
1.7 Системи живлення та резервування	21
1.8 Висновки до першого розділу	22
2 ПІДБІР І АНАЛІЗ АПАРАТНОЇ ЧАСТИНИ СИСТЕМИ	23
2.1 Опис структури та принципів роботи системи	23
2.2 Сенсори.....	26
2.3 Мікроконтролери.....	28
2.4 Мережеві модулі.....	29
2.5 Виконавчі механізми та допоміжні компоненти.....	31
2.7 Детальний опис та порівняння обраних компонентів	34
2.8 Причини вибору конкретних рішень.....	39
2.9 Висновки	42

КПКІ 022047.22.02.21 ВП								
Зм.	Арк.	№докум.	Підпис	Дата	Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32	Літера	Аркуш	Аркушів
Виконав		Антон ЯКОВИШИН	<i>[Підпис]</i>	20.06.21		y		68
Перевір.		Ігор МИХАЛЬЧУК	<i>[Підпис]</i>	20.06.21			2	
Н.контр.		Тетяна КИСІЛЬ	<i>[Підпис]</i>	20.06.21		ХНУ КІ2с-22-2		
Затвер.		Ольга ПАВЛОВА	<i>[Підпис]</i>	20.06.21				

3 РОЗРОБКА ТА РЕАЛІЗАЦІЯ СИСТЕМИ.....	43
3.1 Сценарій впровадження	43
3.2 Структурна схема системи.....	45
3.3 Принцип роботи елементів	47
3.4 Код для реалізації.....	49
3.5 Взаємодія з системою Веб-інтерфейс	55
3.6 Висновки.....	61
ВИСНОВКИ	61
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	62
ДОДАТОК А	67
ДОДАТОК Б.....	73
ДОДАТОК В.....	76
ДОДАТОК Г.....	78
ДОДАТОК Д.....	79
ДОДАТОК Е.....	80

ВСТУП

У сучасних умовах, коли безпека громадянського населення набуває особливої ваги, важливою частиною інфраструктури стають укриття – спеціально обладнані приміщення, здатні забезпечити тимчасовий захист людей у надзвичайних ситуаціях. Проте ефективність таких споруд визначається не лише їх фізичною міцністю, а й наявністю автономних систем життєзабезпечення, серед яких вентиляція є однією з ключових.

З огляду на це виникає потреба у впровадженні сучасних рішень, що забезпечують надійне, енергоефективне та автоматизоване управління вентиляційними системами. Особливе місце серед таких рішень займають програмно-апаратні засоби на базі мікроконтролерів нового покоління, зокрема ESP32 – компактної, потужної та енергоощадної платформи з широкими можливостями підключення до мережі.

Підключення таких пристроїв до екосистеми Інтернету мов (IoT) відкриває нові горизонти в автоматизації: збирання даних з датчиків, дистанційне керування, адаптивне регулювання системи залежно від показників навколишнього середовища — все це можливо без постійного втручання людини. Таким чином, рішення на базі ESP32 здатне забезпечити стабільну роботу вентиляції навіть за умови перебоїв із живленням або відсутності зв'язку із зовнішнім світом.

Актуальність теми зумовлена необхідністю підвищення безпеки укриттів, а також впровадженням інноваційних підходів до управління інженерними системами в обмежених умовах. Розробка таких рішень дозволяє не лише покращити функціонування вентиляції, а й створити основу для подальшої модернізації об'єктів цивільного захисту.

Мета цієї роботи – створити програмно-апаратний засіб управління вентиляцією укриття на базі ESP32, який забезпечуватиме автономність, надійність та можливість інтеграції з іншими IoT-компонентами.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						4
Зм.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ СУЧАСНОГО СТАНУ ПРОБЛЕМИ ТА УЗАГАЛЬНЕННЯ ДОСЛІДЖЕНЬ У СФЕРІ КЕРУВАННЯ ВЕНТИЛЯЦІЄЮ УКРИТТІВ

1.1 Історія та розвиток систем вентиляції укриттів

Ідея вентиляції укриттів виникла разом із появою самих захисних споруд, коли стало зрозуміло, що для безпечного перебування людей під землею або у захищених приміщеннях необхідно забезпечити постійний повітрообмін. На початку ХХ століття, під час Першої та Другої світових воєн, вентиляційні системи укриттів були переважно механічними — використовували прості вентилятори, трубопроводи, клапани для подачі свіжого повітря та виведення забрудненого. Керування такими системами було ручним або базувалося на простих автоматичних механізмах [1].

З розвитком електроніки у другій половині ХХ століття з'явилися можливості для автоматизації вентиляції. Проте через високі витрати і складність обладнання такі системи залишалися доступними переважно для великих об'єктів — заводів, військових комплексів, бомбосховищ. Водночас у приватних чи малих укриттях вентиляція залишалася простішою та менш контрольованою [2].

Поява мікроконтролерів на початку 2000-х дала новий поштовх у розвитку інтелектуальних систем керування. Вони дозволили розробляти компактні, енергоефективні і гнучкі рішення. Особливо помітним став розвиток платформи ESP32 від компанії Espressif Systems, випущеної у 2016 році. ESP32 поєднує двоядерний мікропроцесор, розвинені можливості бездротового зв'язку (Wi-Fi, Bluetooth), багатий набір периферійних інтерфейсів і низьке енергоспоживання.

Застосування ESP32 у вентиляційних системах укриттів відкриває нові можливості:

Автоматичний моніторинг параметрів повітря (температура, вологість, концентрація CO₂, шкідливих газів)

Дистанційне керування вентиляторами та клапанами

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

Інтеграція із системами оповіщення та безпеки

Використання Інтернету речей (IoT) для віддаленого моніторингу і аналізу даних [3].

Таким чином, розвиток систем вентиляції у захисних спорудах пройшов шлях від механічних, ручних рішень до сучасних цифрових, програмно-апаратних комплексів на базі мікроконтролерів ESP32, що суттєво підвищує їхню ефективність, надійність і доступність.



Рисунок 1.1 – Система вентиляції для укриття [4]

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						6
Зм.	Арк.	№ докум.	Підпис	Дата		

1.2 Сучасні підходи до автоматизації систем вентиляції укриттів

Забезпечення належного мікроклімату та якості повітря в захисних спорудах є життєво важливою задачею, що безпосередньо впливає на здоров'я та комфорт людей, які там перебувають. В умовах обмеженого об'єму та інтенсивного використання укриттів, ручні або базові механічні системи вентиляції виявляються неефективними, оскільки не здатні адекватно реагувати на динамічні зміни складу повітря, такі як швидке накопичення вуглекислого газу (CO₂), підвищення вологості або температури. Це обумовлює необхідність застосування сучасних підходів до автоматизації систем вентиляції.[5-8]

Сучасна автоматизація вентиляційних систем укриттів базується на декількох ключових принципах:

Моніторинг в реальному часі: Постійний збір даних з різноманітних сенсорів, що вимірюють параметри повітря (концентрація CO₂, температура, відносна вологість, рівень пилу).

Адаптивне керування: Автоматична корекція режиму роботи вентиляції (інтенсивність, увімкнення/вимкнення) на основі аналізу отриманих даних, що забезпечує підтримання оптимальних параметрів без надлишкового споживання ресурсів.

Віддалений доступ та візуалізація: Можливість контролювати стан системи та керувати нею дистанційно, а також візуалізувати історичні та поточні дані для кращого розуміння ситуації.

На сьогоднішній день виділяють кілька рівнів автоматизації вентиляційних систем, які можуть бути застосовані в укриттях:

Базовий рівень автоматизації: Включає використання простих цифрових термостатів або гігростатів, які автоматично вмикають або вимикають вентилятори при досягненні заданих порогових значень температури або вологості. Деякі системи можуть доповнюватися таймерами для роботи за розкладом. Такі рішення

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						7
Зм.	Арк.	№ докум.	Підпис	Дата		

є відносно недорогими, але мають обмежену функціональність і не враховують повний спектр факторів, що впливають на якість повітря в укритті.

Розширений рівень автоматизації: Характеризується інтеграцією більш складних та чутливих датчиків, зокрема датчиків CO₂, які є критичними для укриттів. Системи цього рівня можуть реалізовувати складніші алгоритми керування, реагуючи не лише на температуру та вологість, але й на концентрацію вуглекислого газу. Часто вони використовують централізовані контролери (наприклад, мікроконтролери або міні-ПК) для обробки даних та керування кількома вентиляційними контурами.[9-11]

Інтегровані та IoT-орієнтовані системи: Представляють собою найбільш сучасний підхід, що передбачає підключення системи вентиляції до локальної або глобальної мережі (Інтернет). Це дозволяє реалізувати віддалений моніторинг через веб-інтерфейси або мобільні додатки, отримувати сповіщення про критичні події, а також збирати та аналізувати дані за тривалий період. Такі системи можуть бути частиною загального комплексу "розумного укриття", інтегруючись з іншими підсистемами (освітлення, безпека). Вони забезпечують найвищий рівень комфорту та безпеки, дозволяючи оперативно реагувати на будь-які зміни.

Незважаючи на різноманіття підходів, до систем автоматизації вентиляції укриттів висуваються специфічні вимоги:

Підтримка оптимальних параметрів повітря: Забезпечення безпечного рівня CO₂, комфортної температури та вологості.

Надійність та автономність: Можливість стабільної роботи системи навіть в умовах відсутності зовнішнього живлення або мережевого зв'язку.

Енергоефективність: Оптимізація роботи вентиляції для мінімізації споживання електроенергії.

Простота експлуатації: Інтуїтивно зрозумілий інтерфейс для керування системою користувачами без спеціальних технічних знань.[12-15]

Саме інтегровані IoT-орієнтовані підходи на базі доступних мікроконтролерних платформ, здатних працювати автономно та забезпечувати

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

1.3 Обґрунтування вибору мікроконтролерної платформи для реалізації системи керування вентиляцією укриття

Як було зазначено в попередніх розділах, забезпечення оптимальних параметрів мікроклімату та якості повітря в захисних спорудах є завданням першочергової важливості. Сучасні підходи до автоматизації систем вентиляції укриттів пропонують різноманітні рішення, однак багато з них мають певні обмеження, такі як висока вартість промислових програмованих логічних контролерів (ПЛК), складність їх впровадження та обслуговування, або ж надмірна залежність комерційних "розумних" систем від сторонніх хмарних сервісів, що може бути критичним в умовах автономного функціонування укриття. Ці фактори підкреслюють потребу в розробці гнучкого, економічно ефективного та надійного програмно-апаратного засобу.

У контексті цих вимог, особливої актуальності набувають рішення на базі мікроконтролерів та одноплатних комп'ютерів, які дозволяють створювати кастомізовані системи з оптимальним співвідношенням функціональності та вартості. Серед різноманіття доступних платформ, для розробки програмно-апаратного засобу керування вентиляцією укриття було обрано мікроконтролер ESP32 від компанії Espressif Systems. Цей вибір є обґрунтованим завдяки його унікальному поєднанню потужних технічних характеристик, вбудованих можливостей бездротового зв'язку та широкого набору периферійних інтерфейсів.[16-17]

Ключові переваги ESP32, що зумовлюють його вибір для даної системи:

Висока обчислювальна потужність: ESP32 оснащений двоядерним процесором Tensilica Xtensa LX6, який працює на тактовій частоті до 240 МГц. Це забезпечує достатній запас продуктивності для одночасного виконання декількох складних задач, таких як: безперервний збір та обробка даних з численних датчиків (температури, вологості, концентрації CO₂ та інших показників якості повітря), реалізація ефективних алгоритмів автоматичного керування вентиляційними

					КвРКІ.022047.22.02.21 ПЗ	Арк. 10
Зм.	Арк.	№ докум.	Підпис	Дата		

агрегатами, а також підтримка стабільної роботи мережевих протоколів Wi-Fi та Bluetooth для віддаленого моніторингу та керування через веб-інтерфейс.

Інтегровані бездротові інтерфейси: Наявність вбудованих модулів Wi-Fi (стандарт 802.11 b/g/n) та Bluetooth (Classic та Low Energy - BLE) є фундаментальною перевагою для системи керування вентиляцією укриття. Модуль Wi-Fi дозволяє інтегрувати пристрій в існуючу локальну мережу для віддаленого доступу або створювати власну точку доступу, забезпечуючи пряме підключення та керування з будь-якого пристрою з веб-браузером (смартфон, планшет, ноутбук). Bluetooth Low Energy може бути використаний для початкової конфігурації пристрою або для підключення додаткових низькоенергетичних бездротових сенсорів, розширюючи функціонал системи.

Гнучкість та розширюваність (GPIO та інтерфейси): ESP32 має велику кількість програмованих виводів загального призначення (GPIO), які можуть бути налаштовані для виконання різноманітних функцій – від цифрових входів/виходів для керування реле (що комутують живлення вентиляторів) до аналогових входів та виходів широтно-імпульсної модуляції (ШИМ). Підтримка стандартних інтерфейсів зв'язку, таких як I2C (для підключення цифрових датчиків, наприклад, BME280) та UART (для взаємодії з датчиками CO₂, такими як MH-Z19B), дозволяє легко інтегрувати широкий спектр необхідних сенсорів та виконавчих пристроїв.

Оптимальне співвідношення ціни та продуктивності: Порівняно з промисловими контролерами або готовими комерційними рішеннями, ESP32 пропонує значно нижчу вартість при збереженні високої функціональності та надійності, що робить його доступним рішенням для локальних систем.[18]

Зручність розробки та налагодження: Наявність інтегрованого на платі USB-UART конвертера (наприклад, CP2102) та схеми автопрограмування значно спрощує процес завантаження прошивки та налагодження програмного забезпечення, що є важливим аспектом для фази розробки проекту.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 11
Зм.	Арк.	№ докум.	Підпис	Дата		

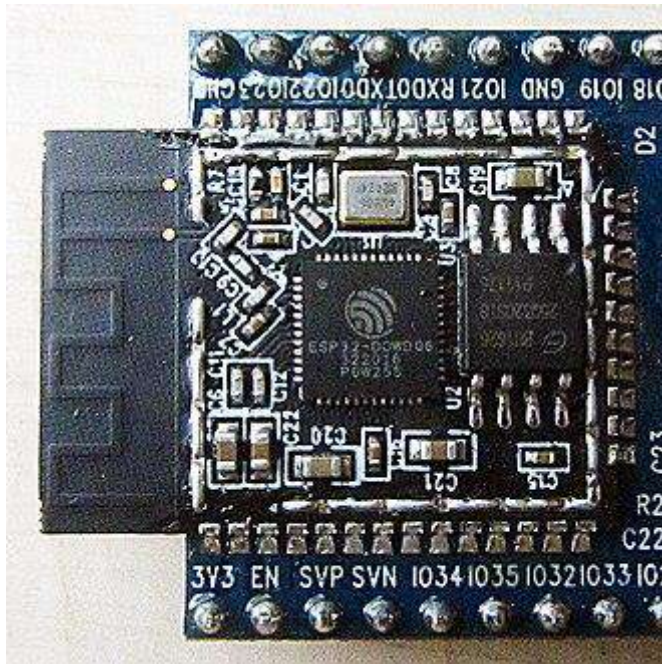


Рисунок. 1.3 – Модуль ESP-WROOM-32 з чіпом ESP32-D0WDQ6 [19]

1.4 Сенсорні елементи (датчики) для моніторингу параметрів повітря

Ефективна система керування вентиляцією укріття повинна базуватися на точних і своєчасних даних про стан навколишнього середовища. Для цього використовуються різноманітні сенсорні елементи (датчики), які дозволяють моніторити ключові параметри повітря, що безпосередньо впливають на комфорт та безпеку перебування людей. Вибір конкретних датчиків здійснюється з урахуванням необхідної точності, надійності, діапазону вимірювань, а також сумісності з обраною апаратною платформою – мікроконтролером ESP32 (детально обґрунтовано в пункті 1.3).

Для реалізації функціоналу моніторингу в даній системі передбачається використання наступних типів датчиків:

1. Датчик температури та відносної вологості повітря:

– Призначення: Контроль мікроклімату в укрітті для підтримання комфортних умов та запобігання надмірної вологості, яка може сприяти розвитку плісняви та погіршенню стану обладнання.

– Приклад реалізації: Можуть бути використані датчики серії DHT (наприклад, DHT11 або DHT22) або більш точні та надійніші датчики, такі як BME280.

– DHT11/DHT22: Прості у використанні, передають дані по однопровідному протоколу. DHT22 забезпечує вищу точність та ширший діапазон вимірювань порівняно з DHT11.[20]

– BME280: Комбінований датчик, що вимірює температуру, вологість та атмосферний тиск. Відрізняється високою точністю, низьким енергоспоживанням та взаємодіє через інтерфейси I2C або SPI, що є перевагою для ESP32 завдяки їхній швидкості та надійності. Підключення до ESP32 зазвичай здійснюється через виводи I2C (SDA – GPIO21, SCL – GPIO22).[21-22]



Рисунок 1.4 – Модуль датчика навколишнього середовища BME280[21]

2. Датчик концентрації вуглекислого газу (CO₂):

- Призначення: Це критично важливий сенсор для укриттів, оскільки концентрація CO₂ безпосередньо вказує на якість повітря та ефективність вентиляції. Високі рівні CO₂ (понад 1000-1500 ppm) можуть викликати сонливість, головний біль та інші проблеми зі здоров'ям. Система повинна автоматично реагувати на підвищення цього показника, активуючи або посилюючи вентиляцію.

- Приклад реалізації: Одним з поширених і надійних рішень є інфрачервоний датчик CO₂, наприклад, MH-Z19B (або його аналоги, такі як MH-

Z19C). Цей датчик використовує принцип недисперсійного інфрачервоного поглинання (NDIR) для точного вимірювання CO₂. [23]

○ Особливості підключення: MH-Z19B зазвичай взаємодіє з мікроконтролером через послідовний інтерфейс UART. Це вимагає підключення виводів TXD та RXD датчика до відповідних вільних UART-портів ESP32 (крім основного UART0, який використовується для прошивки та налагодження і виводу налагоджувальних повідомлень).

3. Датчик якості повітря (концентрації твердих частинок PM_{2.5}/PM₁₀ або летких органічних сполук VOCs):

– Призначення: Додатковий контроль за наявністю у повітрі дрібних часток пилу, диму або шкідливих хімічних сполук, що може бути важливим у певних сценаріях використання укриття (наприклад, після забруднення зовнішнього повітря).

– Приклад реалізації:

– SDS011 (PM_{2.5}/PM₁₀): Популярний лазерний датчик пилу, що передає дані через UART. [24]

– MQ-серія датчиків (наприклад, MQ-135 для VOCs/загальної якості повітря): Аналогові датчики, що вимагають калібрування, але дозволяють оцінювати наявність певних газів та загальної якості повітря.

Вибір комбінації цих датчиків дозволить системі всебічно оцінювати стан повітря в укритті та приймати обґрунтовані рішення щодо необхідності та інтенсивності вентиляції, забезпечуючи максимальну безпеку та комфорт перебування.

1.5 Виконавчі механізми системи керування вентиляцією

Виконавчі механізми є невід'ємною частиною будь-якої автоматизованої системи керування, оскільки вони трансформують керуючі сигнали від контролера (у нашому випадку – ESP32) у фізичні дії. У системі керування вентиляцією

					КвРКІ.022047.22.02.21 ПЗ	Арк. 14
Зм.	Арк.	№ докум.	Підпис	Дата		

укриття основними виконавчими елементами є пристрої, що забезпечують рух повітря, та компоненти для їх комутації.

Для реалізації функціоналу керування вентиляцією в даній системі передбачається використання наступних компонентів:

1. Вентилятори припливно-витяжної системи:

– Призначення: Основне завдання системи – забезпечити циркуляцію свіжого повітря та видалення забрудненого з укриття. Для цього використовуються припливні та/або витяжні вентилятори, які можуть працювати як постійно, так і за вимогою, залежно від показань датчиків (особливо CO₂) та обраного режиму роботи.

– Типи вентиляторів: Можуть бути використані каналні, осьові або відцентрові вентилятори відповідної потужності, розрахованої на об'єм укриття та необхідну кратність повітрообміну.

– Керування: Більшість побутових та невеликих промислових вентиляторів керуються подачею/зняттям живлення (ON/OFF). Для регулювання швидкості обертання можуть бути використані вентилятори зі вбудованими регуляторами, що підтримують керування через аналоговий сигнал (0-10В) або ШІМ (Широтно-імпульсна модуляція), або ж зовнішні регулятори потужності.[25]

–



Рисунок 1.5 – Вентилятор для припливно-витяжної вентиляції 200мм WO-B (WO-S)[26]

					КвРКІ.022047.22.02.21 ПЗ	Арк. 15
Зм.	Арк.	№ докум.	Підпис	Дата		

2. Релейні модулі:

– Призначення: Оскільки мікроконтролер ESP32 працює з низькими напругами та струмами (3.3В), він не може безпосередньо керувати потужними вентиляторами, які зазвичай живляться від мережі 220В змінного струму або 12В/24В постійного струму з високими струмами. Релейний модуль слугує як електромагнітний перемикач, що дозволяє ESP32 безпечно комутувати високовольтні або сильноточні навантаження.

– Принцип роботи: Реле містить електромагнітну котушку, яка при подачі низьковольтного сигналу від ESP32 замикає або розмикає силові контакти, через які проходить живлення до вентилятора.

– Приклад реалізації: Для даної системи буде використаний 1-канальний або багатоканальний (наприклад, 2-х або 4-х канальний) релейний модуль, сумісний з ESP32 (зазвичай, вони мають керуючий сигнал 3.3В або 5В). Підключення здійснюється до цифрових GPIO виводів ESP32, що дозволяє програмно вмикати або вимикати вентилятор.[27-28]

3. Моторизовані повітряні заслінки:

– Призначення: У більш складних системах вентиляції укриттів можуть застосовуватися моторизовані заслінки для регулювання потоку повітря в різних зонах або для перекриття каналів у випадку зовнішньої загрози (наприклад, хімічної атаки).

– Керування: Заслінки керуються електродвигунами (кроковими або сервоприводами), які, в свою чергу, контролюються ESP32 через відповідні драйвери або безпосередньо, якщо заслінка має простий дискретний інтерфейс ON/OFF/OPEN/CLOSE.

Вибір відповідних виконавчих механізмів забезпечить надійне та ефективне керування повітряними потоками в укритті, дозволяючи системі автоматично підтримувати необхідні параметри мікроклімату та реагувати на зміни якості повітря.[29]

					КвРКІ.022047.22.02.21 ПЗ	Арк. 16
Зм.	Арк.	№ докум.	Підпис	Дата		

1.6 Засоби індикації та користувацького інтерфейсу

Для забезпечення ефективної взаємодії користувача з системою керування вентиляцією укриття та надання актуальної інформації про її роботу, необхідне використання відповідних засобів індикації та інтерфейсу. Ці компоненти дозволяють моніторити стан системи, отримувати дані з датчиків, а також здійснювати керування режимами роботи вентиляції.

Для реалізації функціоналу індикації та користувацького інтерфейсу в даній системі передбачається використання наступних компонентів та підходів:

1. Світлодіодна індикація (LED-індикатори):

- Призначення: Є базовим, але дуже інформативним засобом візуальної індикації. Дозволяють швидко оцінити основні режими роботи системи або її стан.
- Приклад реалізації: Можна використовувати світлодіоди для індикації:
- Стану живлення: Система увімкнена та працює.
- Статусу Wi-Fi: Підключено до мережі / працює як точка доступу / помилка підключення.
- Режиму роботи вентиляції: Автоматичний режим / ручний режим / активна вентиляція.
- Аварійних станів: Перевищення критичних значень CO₂, помилка датчика.
- Підключення: Світлодіоди підключаються до цифрових GPIO виводів ESP32 через струмообмежувальні резистори.[30]

2. Веб-інтерфейс:

- Призначення: Це основний та найбільш гнучкий засіб взаємодії з користувачем. Завдяки вбудованому Wi-Fi модулю ESP32 може функціонувати як міні-веб-сервер, що дозволяє отримувати доступ до системи з будь-якого пристрою (смартфон, планшет, ноутбук) через стандартний веб-браузер.
- Функціонал: Веб-інтерфейс забезпечує:

					КвРКІ.022047.22.02.21 ПЗ	Арк. 17
Зм.	Арк.	№ докум.	Підпис	Дата		

- Відображення поточних даних: Зручне представлення показань усіх датчиків (температура, вологість, CO₂, тощо) у числовому або графічному вигляді.
- Відображення статусу системи: Поточний режим роботи вентиляції, стан вентиляторів (увімкнено/вимкнено).
- Керування режимами: Кнопки для перемикання між автоматичним та ручним режимами роботи вентиляції.
- Ручне керування: Можливість примусово вмикати/вимикати вентилятор.
- Налаштування: Поля для введення та збереження порогових значень CO₂, бажаної температури, параметрів Wi-Fi мережі.
- Переваги: Не вимагає встановлення спеціального програмного забезпечення, доступність з різних пристроїв, можливість віддаленого моніторингу.[31]

3. Рідкокристалічний дисплей (LCD) або OLED-дисплей:

- Призначення: Для локального відображення ключових параметрів без необхідності використання зовнішнього пристрою (смартфона/комп'ютера). Це може бути корисним для швидкої перевірки стану системи безпосередньо в укритті.
- Приклад реалізації: Можуть використовуватися:
 - LCD 16x2 або 20x4 з модулем I2C: Прості та доступні дисплеї, що виводять текстову інформацію. Підключаються до ESP32 через інтерфейс I2C (GPIO21, GPIO22), мінімізуючи кількість необхідних проводів.
 - OLED-дисплеї (наприклад, 0.96" OLED I2C): Компактні, яскраві та контрастні, дозволяють виводити як текст, так і прості графіки. Також підключаються через I2C.
 - Обмеження: Обмежений обсяг інформації, що відображається одночасно, та відсутність інтерактивного керування (лише відображення).[32]

4. Звукові сповіщення (зумер/п'єзовипромінювач):

					КвРКІ.022047.22.02.21 ПЗ	Арк. 18
Зм.	Арк.	№ докум.	Підпис	Дата		

– Призначення: Використовується для швидкого привернення уваги до критичних подій, наприклад, перевищення небезпечного рівня CO₂, виникнення помилки датчика або інших аварійних ситуацій.

– Підключення: Підключається до цифрового GPIO виводу ESP32.

Комбінація веб-інтерфейсу як основного засобу керування та моніторингу з додатковою світлодіодною індикацією та, за необхідності, локальним дисплеєм, забезпечить повноцінну та зручну взаємодію користувача з розробленою системою керування вентиляцією укриття [33].



Рисунок 1.6 – Панель управління з сенсорним екраном для управління промисловими та домашніми установками та вихлопними установками[34]

1.7 Системи живлення та резервування

Надійне та стабільне живлення є фундаментальним аспектом функціонування будь-якої електронної системи, особливо такої, що критично важлива для безпеки, як система керування вентиляцією в укритті. В умовах можливих перебоїв з електропостачанням, характерних для захисних споруд, особливої актуальності набуває питання резервування живлення.

Для забезпечення безперебійної роботи програмно-апаратного засобу керування вентиляцією укриття передбачається використання комбінованої системи живлення, що включає основне та резервне джерела:

1. Основне джерело живлення:

– Призначення: Забезпечення стабільного електропостачання для всіх компонентів системи (мікроконтролера ESP32, датчиків, релейних модулів, вентиляторів).

– Реалізація: Зазвичай, система живиться від мережі змінного струму 220В. Для живлення компонентів, що працюють від низьковольтного постійного струму (ESP32, датчики, більшість реле), використовується імпульсний блок живлення (ІБЖ) з відповідними вихідними параметрами (наприклад, 5В або 12В постійного струму).

– Конвертація напруги: ESP32 типово живиться від 3.3В. Плати розробки ESP32 (такі як ESP32-DevKitC) вже мають вбудований стабілізатор напруги (наприклад, AMS1117), який перетворює 5В (з USB або зовнішнього джерела) у необхідні 3.3В для мікроконтролера. Якщо використовуються компоненти на 5В або 12В (наприклад, деякі реле або датчики), їх живлення може бути безпосередньо від відповідного ІБЖ.[35-36]

2. Резервне джерело живлення (акумуляторна батарея):

– Призначення: Забезпечення автономної роботи системи протягом певного періоду часу у випадку відключення основного електропостачання. Це критично важливо для укриттів, де безперебійна вентиляція може бути життєво необхідною.[37]

– Реалізація: Як резервне джерело може бути використана свинцево-кислотна (AGM) або літій-іонна (Li-Ion) акумуляторна батарея відповідної ємності та напруги (наприклад, 12В).

– Модуль керування зарядом/розрядом: Для забезпечення коректної роботи та довговічності акумуляторної батареї необхідний спеціалізований модуль

					КвРКІ.022047.22.02.21 ПЗ	Арк. 20
Зм.	Арк.	№ докум.	Підпис	Дата		

керування зарядом (наприклад, на базі мікросхеми TP4056 для Li-Ion або спеціальні контролери заряду для AGM). Цей модуль відповідає за:

- Автоматичне перемикання: Перемикання живлення системи на акумулятор при відсутності основного живлення та назад при його відновленні.
- Зарядку акумулятора: Забезпечення оптимального режиму заряду акумулятора від основного джерела.
- Захист: Захист акумулятора від перезаряду, глибокого розряду та короткого замикання.
- Моніторинг стану батареї: Бажано передбачити можливість моніторингу рівня заряду акумулятора (наприклад, через аналоговий вхід ESP32 з дільником напруги) для відображення цієї інформації у веб-інтерфейсі та надсилання сповіщень про низький заряд.

Вибір відповідної архітектури системи живлення та реалізація механізмів резервування дозволить забезпечити високу надійність та безперебійність функціонування системи керування вентиляцією укриття, що є запорукою безпеки людей у критичних ситуаціях.



Рисунок. 1.7 –Приклад 12-вольтової літєвої deer-cycle батареї (LiFePO₄), яка може стати резервним джерелом живлення для вентиляційного обладнання у сховищі [38].

					КвРКІ.022047.22.02.21 ПЗ	Арк. 21
Зм.	Арк.	№ докум.	Підпис	Дата		

1.8 Висновок до першого розділу

У першому розділі кваліфікаційної роботи було проведено комплексний аналіз теоретичних засад та існуючих підходів до керування вентиляційними системами, з особливим акцентом на їх застосування в укриттях. Розглянуто історичний контекст розвитку систем вентиляції та захисних споруд, а також вивчено сучасні тенденції в автоматизації, що дозволило визначити ключові вимоги до подібних систем.

На основі проведеного аналізу було обґрунтовано вибір мікроконтролерної платформи ESP32 як центрального керуючого елемента розроблюваного програмно-апаратного засобу. Визначено та описано основні апаратні компоненти, необхідні для функціонування системи, включаючи сенсорні елементи для моніторингу параметрів повітря (температури, вологості, CO₂), виконавчі механізми (вентилятори, релейні модулі), засоби індикації та користувацького інтерфейсу (веб-інтерфейс, світлодіоди), а також систему живлення з резервуванням.

Таким чином, перший розділ заклав теоретичну та апаратну базу для подальшої розробки програмно-апаратного засобу керування вентиляцією укриття, визначивши ключові компоненти та принципи їх взаємодії.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						22
Зм.	Арк.	№ докум.	Підпис	Дата		

2 ПІДБІР ТА АНАЛІЗ АПАРАТНОЇ ЧАСТИНИ СИСТЕМИ

2.1 Загальна структура та принципи функціонування системи

У сучасному світі, де динамічний розвиток технологій перетворює всі аспекти нашого життя, концепція "розумних" та адаптивних систем набуває особливого значення. Ідея про те, що інтелектуальні рішення можуть самостійно реагувати на зміни навколишнього середовища та оптимізувати процеси, вже далеко відійшла від суто теоретичних міркувань. Сьогодні ми бачимо трансформації навколо нас, коли цифрові інновації інтегруються у повсякденну реальність, від розумних будинків до масштабних проєктів "розумних міст". Однак, серед цих глобальних змін, існує і критична потреба в підвищенні рівня безпеки та комфорту в специфічних, надзвичайно важливих об'єктах, таких як захисні споруди та укриття.

Традиційні підходи до вентиляції укриттів, що часто ґрунтувалися на періодичному ручному керуванні або простих механічних системах, виявляються недостатніми перед обличчям сучасних викликів. Забезпечення стабільного мікроклімату, чистого повітря та контрольованого газового складу в умовах тривалого перебування людей – це завдання, що вимагає постійного, точного моніторингу та оперативного, інтелектуального реагування. Саме в цьому контексті ідея про створення програмно-апаратного засобу для автономного керування вентиляцією набуває стратегічної важливості, перетворюючи укриття з пасивних сховищ на активні, безпечні та комфортні простори.

Розроблена система є яскравим прикладом втілення принципів "розумних" технологій у критично важливій галузі. Її функціональність ґрунтується на органічному поєднанні апаратних компонентів та програмного забезпечення, які працюють синхронно для досягнення єдиної мети – автоматизованого підтримання оптимальних параметрів повітря. Цей синергічний підхід забезпечує не лише

					КвРКІ.022047.22.02.21 ПЗ	Арк. 23
Зм.	Арк.	№ докум.	Підпис	Дата		

ефективний контроль, а й можливість віддаленого моніторингу та управління, що є вирішальним фактором в умовах надзвичайних ситуацій.

Апаратна частина системи представляє собою інтегрований комплекс взаємопов'язаних блоків, що візуально відображено на рисунку 2.1. Вона спроектована таким чином, щоб забезпечити надійний збір даних та ефективне виконання команд. Центральним "мозком" системи є блок керуючого контролера (ESP32), який завдяки своїй обчислювальній потужності та вбудованим комунікаційним модулям виступає в ролі командного центру. "Очима" системи є блок сенсорів, що постійно зчитує життєво важливі параметри повітря – температуру, вологість та концентрацію вуглекислого газу – передаючи їх для аналізу. Для безпосереднього впливу на повітряне середовище застосовується блок виконавчих механізмів, що включає релейні модулі для безпечної комутації електричних ланцюгів та самі вентилятори (припливні та/або витяжні). Взаємодію користувача із системою забезпечує блок інтерфейсу користувача та індикації, реалізований переважно через зручний веб-інтерфейс, доповнений базовими світлодіодними індикаторами статусу. Уся ця апаратна інфраструктура живиться від надійного блоку живлення, що передбачає гнучкість використання як основного, так і резервного джерела для забезпечення безперервності роботи.

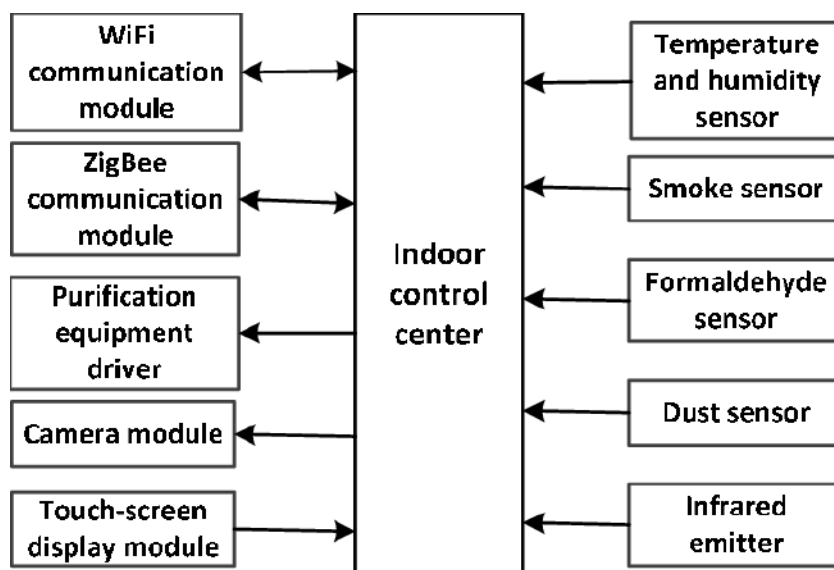


Рисунок 2.1.– Блочна схема апаратної частини системи [39]

Зм.	Арк.	№ докум.	Підпис	Дата

Програмне забезпечення, що "оживляє" апаратні компоненти, розроблено з урахуванням модульного принципу, що сприяє його масштабованості, легкості оновлення та спрощенню діагностики, чітко розподіляє функціональні обов'язки. Модуль ініціалізації відповідає за швидке та коректне завантаження всіх систем при старті. Модуль збору та обробки даних безперервно отримує показання від сенсорів, виконуючи необхідну фільтрацію для підвищення точності. Серцем логіки є модуль логіки керування, який на основі аналізу отриманих даних та заздалегідь встановлених алгоритмів приймає критичні рішення щодо активації вентиляторів. Для зручного доступу та керування системою з будь-якого пристрою інтегрований модуль веб-сервера, а модуль комунікації забезпечує надійний обмін даними як між внутрішніми компонентами системи, так і із зовнішнім світом через Wi-Fi.

Взаємодія компонентів в системі реалізована як безперервний, інтелектуальний цикл зворотного зв'язку. Цей цикл починається з постійного моніторингу: сенсори відстежують ключові параметри повітря та передають їх на мікроконтролер. Програмне забезпечення аналізує ці дані в реальному часі, визначаючи, чи відповідають вони встановленим нормам. У випадку виявлення відхилень (наприклад, підвищення рівня CO₂), програмний модуль логіки керування генерує відповідні команди. Ці команди передаються на виконавчі механізми (релейні модулі), які фізично активують або деактивують вентилятори, забезпечуючи приплив свіжого повітря або виведення забрудненого. Весь цей процес, разом з актуальними показаннями датчиків та станом системи, динамічно відображається на веб-інтерфейсі, дозволяючи користувачу отримувати повну інформацію та, за необхідності, вносити корективи у роботу системи.

Принципи функціонування розробленої системи виходять за рамки простої автоматизації, втілюючи концепції адаптивного та енергоефективного керування. Вона здатна не просто сліпо виконувати заздалегідь задані команди, а динамічно реагувати на реальні умови, оптимізуючи інтенсивність повітрообміну лише тоді, коли це дійсно необхідно. Це забезпечує значну економію енергії – критично

					КвРКІ.022047.22.02.21 ПЗ	Арк. 25
Зм.	Арк.	№ докум.	Підпис	Дата		

важливий фактор для автономних об'єктів. Можливість підтримання контрольованого надлишкового тиску в укритті є фундаментальним елементом безпеки, що запобігає проникненню потенційно небезпечних речовин ззовні. Простота та інтуїтивність моніторингу через зручний веб-інтерфейс робить систему доступною для використання без спеціальних технічних знань, а її модульність гарантує легкість в обслуговуванні, діагностиці та потенційному масштабуванні функціоналу в майбутньому.

2.2 Сенсори

Сенсорні елементи є фундаментальною складовою будь-якої системи моніторингу та керування, виконуючи функцію "органів чуття", що дозволяють збирати інформацію про фізичне середовище. У контексті системи керування вентиляцією укриття, сенсори відіграють ключову роль у забезпеченні точних та актуальних даних про параметри мікроклімату та якість повітря. Вибір відповідних сенсорів є критично важливим етапом проектування, оскільки від їхньої точності, надійності та відповідності умовам експлуатації залежить ефективність функціонування всієї системи та, зрештою, безпека та комфорт людей, що перебувають в укритті.

Системи вентиляції, особливо в закритих та герметичних просторах, таких як укриття, мають на меті не лише забезпечити подачу свіжого повітря, а й підтримувати оптимальні умови для дихання та загального самопочуття. Перевищення певних порогових значень ключових параметрів може призвести до швидкого погіршення здоров'я та продуктивності. Тому, для реалізації функціоналу системи, було визначено необхідність моніторингу наступних основних показників:

– Температура повітря: Цей параметр є одним з найважливіших для забезпечення теплового комфорту. Надмірно висока температура призводить до перегріву організму, дискомфорту, зниження концентрації та може бути

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						26
Зм.	Арк.	№ докум.	Підпис	Дата		

небезпечною у випадку тривалого впливу. Надмірно низька температура також є неприпустимою для тривалого перебування. Оптимальний діапазон температур сприяє підвищенню працездатності та покращенню самопочуття.

– Відносна вологість повітря: Контроль вологості є важливим для запобігання розвитку шкідливих мікроорганізмів, таких як пліснява та грибок, а також для уникнення дискомфорту, пов'язаного із занадто сухим або надмірно вологим повітрям. Висока вологість створює відчуття задухи та може призводити до конденсації вологи на поверхнях, тоді як занадто низька вологість може викликати подразнення слизових оболонок та дихальних шляхів.

– Концентрація вуглекислого газу (CO₂): Цей показник є прямим індикатором якості повітря та ефективності вентиляції в закритих приміщеннях з високою щільністю людей. При диханні люди виділяють CO₂, і без належного повітрообміну його концентрація швидко зростає. Перевищення допустимих норм CO₂ (зазвичай вище 1000-1500 ppm) призводить до погіршення когнітивних функцій, сонливості, головного болю, запаморочення та відчуття задухи. Моніторинг CO₂ дозволяє точно визначити момент, коли необхідно посилити вентиляцію для підтримки свіжості повітря.

Виходячи з цих вимог, для інтеграції в систему були обрані наступні категорії сенсорів:

– Датчики температури та вологості (Temperature & Humidity Sensors): Ці інтегровані модулі дозволяють одночасно вимірювати обидва параметри, що спрощує їх підключення до мікроконтролера та обробку даних. Вони є ключовими для підтримання комфортних фізичних умов.

– Датчики концентрації CO₂ (Carbon Dioxide Sensors): Ці спеціалізовані сенсори використовують різні технології (наприклад, недисперсійний інфрачервоний – NDIR) для точного вимірювання рівня вуглекислого газу. Їхня наявність є критично важливою для автоматичного керування припливною та/або витяжною вентиляцією в залежності від кількості людей та їхньої активності.

2.3 Мікроконтролери

Мікроконтролер (МК) є інтегральною мікросхемою, що поєднує на одному кристалі всі основні компоненти комп'ютера: центральний процесор (CPU), оперативну пам'ять (RAM), постійну пам'ять (ROM/Flash), таймери, порти вводу/виводу (GPIO) та різноманітні периферійні інтерфейси. По суті, це мініатюрний, високоінтегрований комп'ютер, спроектований для керування конкретними функціями в рамках вбудованих систем. На відміну від універсальних комп'ютерів, мікроконтролери оптимізовані для виконання специфічних задач у режимі реального часу, часто з низьким енергоспоживанням та високою надійністю.

У контексті системи керування вентиляцією укриття, мікроконтролер виконує роль "центру прийняття рішень" та "диригента" всієї системи. Він є посередником між фізичним світом (представленим сенсорами та виконавчими механізмами) та логікою керування (закладеною в програмне забезпечення). Його функціональні завдання включають:

- Збір та обробка даних: Отримання аналогових або цифрових сигналів від сенсорів, їх перетворення та інтерпретація для отримання осмислених показників (температура, вологість, рівень CO₂).
- Реалізація логіки керування: Виконання програмного коду, який на основі отриманих даних та заданих алгоритмів (наприклад, порогових значень CO₂) визначає необхідність активації або деактивації вентиляторів.
- Керування виконавчими механізмами: Генерування та надсилання відповідних керуючих сигналів (наприклад, включення/виключення реле) для впливу на вентиляційне обладнання.
- Забезпечення комунікації: Організація обміну даними із зовнішніми пристроями або мережами (наприклад, для передачі даних на веб-інтерфейс, отримання команд від користувача).

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						28
Зм.	Арк.	№ докум.	Підпис	Дата		

– Синхронізація та таймінг: Керування послідовністю подій та відлік часу для періодичного моніторингу або виконання завдань.

Вибір мікроконтролерної платформи для такого проєкту є фундаментальним, оскільки він визначає обчислювальні можливості системи, доступні інтерфейси, складність розробки програмного забезпечення, енергоспоживання та загальну вартість рішення. Для системи керування вентиляцією укриття необхідно, щоб мікроконтролер мав:

– Достатню продуктивність для обробки даних від декількох сенсорів та підтримки веб-сервера.

– Широкий набір портів вводу/виводу для підключення датчиків та реле.

– Вбудовані можливості бездротового зв'язку (наприклад, Wi-Fi) для віддаленого доступу.

– Підтримку стандартних комунікаційних протоколів (I2C, UART) для взаємодії з периферією.

– Відносну легкість програмування та доступність розробницьких інструментів.

Для реалізації даної системи був обраний мікроконтролер ESP32. Ця платформа, розроблена компанією Espressif Systems, є популярним вибором для проєктів Інтернету Речей (IoT) завдяки своєму потужному процесору, значним обсягам пам'яті, розширеним можливостям введення/виведення, а також інтегрованим модулям Wi-Fi та Bluetooth, що значно спрощує розробку мережевих та автономних пристроїв.

2.4 Мережеві модулі

У сучасному світі, де все частіше з'являються "розумні" пристрої та системи Інтернету Речей (IoT), можливість віддаленого моніторингу та керування є не просто зручністю, а часто критичною вимогою. Для системи керування вентиляцією укриття, здатність отримувати дані про мікроклімат та впливати на

					КвРКІ.022047.22.02.21 ПЗ	Арк. 29
Зм.	Арк.	№ докум.	Підпис	Дата		

роботу вентиляції ззовні приміщення є надзвичайно цінною, особливо в умовах, коли фізичний доступ може бути обмежений або ускладнений. Саме мережеві модулі надають цю ключову функціональність, дозволяючи системі взаємодіяти з користувачами, іншими пристроями або хмарними сервісами.

Мережевий модуль – це компонент, який забезпечує фізичне та логічне з'єднання пристрою з комп'ютерною мережею. Для вбудованих систем, таких як ваша, найпоширенішими та найзручнішими технологіями бездротового зв'язку є Wi-Fi та Bluetooth. Вони пропонують різні можливості та сфери застосування:

– Wi-Fi (Wireless Fidelity): Це стандарт бездротової локальної мережі, що дозволяє пристроям підключатися до інтернету або до інших пристроїв у локальній мережі через бездротову точку доступу (роутер). Для IoT-пристроїв Wi-Fi є ідеальним вибором, оскільки він забезпечує відносно високу швидкість передачі даних, широкий радіус дії в межах приміщення та легку інтеграцію в існуючу домашню або офісну інфраструктуру. Завдяки Wi-Fi, система може розміщувати власний веб-сервер, до якого користувач може отримати доступ зі свого смартфона, планшета чи комп'ютера через звичайний веб-браузер. Це усуває необхідність у спеціалізованих додатках або фізичних дисплеях на самому пристрої.

– Bluetooth: Ця технологія призначена для бездротового зв'язку на коротких відстанях між пристроями. Існує два основні типи: Classic Bluetooth (для потокової передачі аудіо, файлів) та Bluetooth Low Energy (BLE), який оптимізований для низького енергоспоживання та обміну невеликими порціями даних. BLE часто використовується для локального підключення до мобільних додатків, швидкого налаштування пристрою або передачі даних на невеликі відстані без необхідності підключення до Wi-Fi мережі.

У системі керування вентиляцією укриття основним каналом зв'язку для віддаленого моніторингу та керування було обрано Wi-Fi. Це обумовлено тим, що мікроконтролер ESP32 має вбудований модуль Wi-Fi, що усуває потребу в додаткових зовнішніх компонентах, спрощує схемотехніку та знижує вартість. Можливості Wi-Fi ESP32 дозволяють легко реалізувати функціонал веб-сервера,

					КвРКІ.022047.22.02.21 ПЗ	Арк. 30
Зм.	Арк.	№ докум.	Підпис	Дата		

надаючи користувачам зручний та універсальний інтерфейс для взаємодії з системою через стандартний веб-браузер. Хоча ESP32 також підтримує Bluetooth, Wi-Fi є більш пріоритетним для даного застосування через потребу в доступі до системи з будь-якої точки локальної мережі або через інтернет.

2.5 Виконавчі механізми та допоміжні компоненти

Для ефективної роботи будь-якої автоматизованої системи, що взаємодіє з фізичним середовищем, наявність виконавчих механізмів є так само важливою, як і здатність до моніторингу. Саме ці компоненти отримують "команди" від керуючого контролера і перетворюють їх на реальні дії, змінюючи стан об'єкта керування. У системі керування вентиляцією укриття виконавчими елементами є безпосередньо вентилятори, а також модулі, що забезпечують їх безпечно та контрольоване ввімкнення/вимкнення. Окрім них, для забезпечення стабільності, безпеки та зручності використання всієї системи, необхідно врахувати низку допоміжних компонентів, що формують її надійний фундамент.

Виконавчі механізми:

1. Вентилятори: Є основним компонентом, відповідальним за примусову циркуляцію повітря. Вибір вентиляторів для укриття вимагає ретельного підходу, враховуючи такі параметри як:

– Продуктивність (об'ємний потік повітря): Вимірюється в кубічних метрах на годину ($\text{м}^3/\text{год}$) і має відповідати санітарним нормам та розмірам укриття, щоб забезпечити необхідний повітрообмін для комфортного перебування визначеної кількості людей.

– Напір (тиск): Здатність вентилятора долати опір повітровідводів. Для ефективної вентиляції в умовах укриття може знадобитися створення певного надлишкового тиску (підпору) для запобігання проникненню забрудненого повітря ззовні.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 31
Зм.	Арк.	№ докум.	Підпис	Дата		

– Тип двигуна: Можуть бути вентилятори постійного струму (DC) або змінного струму (AC). DC-вентилятори зазвичай менші, енергоефективніші та легше керуються мікроконтролером (наприклад, через широтно-імпульсну модуляцію – ШІМ, для регулювання обертів), але можуть потребувати окремого блоку живлення. AC-вентилятори частіше використовуються для більших обсягів повітря та підключаються безпосередньо до мережі 220В.

– Рівень шуму: Важливий параметр для житлових або робочих приміщень, особливо в укриттях, де люди перебувають тривалий час.

– Споживання електроенергії: Прямо впливає на автономність системи при роботі від резервних джерел живлення.

2. Релейні модулі: Слугують інтерфейсом між низьковольтним мікроконтролером та потужними електричними навантаженнями, такими як вентилятори. Їхнє основне призначення – забезпечити гальванічну розв'язку та безпечну комутацію високих напруг або струмів.

– Принцип роботи: Реле, за своєю суттю, є керованим перемикачем. При подачі низьковольтного керуючого сигналу від мікроконтролера, реле спрацьовує, замикаючи або розмикаючи силовий електричний ланцюг вентилятора.

– Типи реле:

– Електромеханічні реле: Найпоширеніші, мають механічні контакти. Відрізняються високою здатністю комутації струмів та надійністю. Недоліки: наявність механічних частин (що обмежує ресурс спрацьовувань), клацання при спрацьовуванні та відносно повільна швидкість реакції.

– Твердотільні реле (Solid State Relay, SSR): Не мають рухомих частин, що забезпечує безшумну роботу, високу швидкість спрацьовування та практично необмежений ресурс. Ідеальні для швидких перемикачів. Недоліки: вища вартість та можливе нагрівання при комутації значних струмів, що потребує радіаторів.

– Характеристики: При виборі релейного модуля необхідно враховувати кількість каналів (залежить від кількості вентиляторів), напругу керуючого сигналу

					КвРКІ.022047.22.02.21 ПЗ	Арк. 32
Зм.	Арк.	№ докум.	Підпис	Дата		

(сумісність з ESP32, зазвичай 3.3В або 5В) та максимальний струм/напругу, який реле може комутувати.

Допоміжні компоненти:

1. Засоби індикації: Хоча основним інтерфейсом взаємодії з користувачем є веб-сервер, візуальні та, можливо, звукові індикатори значно підвищують зручність та оперативність отримання інформації про стан системи.

– Світлодіодні індикатори (LED): Прості та енергоефективні, можуть використовуватися для відображення базових статусів: живлення увімкнено, Wi-Fi підключено, вентилятор працює, режим "Аварія" (наприклад, перевищення CO₂). Різні кольори та режими миготіння можуть передавати різну інформацію.

– Звукові сповіщення (Buzzer): Можуть бути інтегровані для подачі коротких сигналів у випадку критичних подій (наприклад, досягнення небезпечного рівня CO₂, критично низький заряд батареї), що приверне увагу користувача, навіть якщо він не дивиться на веб-інтерфейс.

– Хоча локальні дисплеї (LCD/OLED) можуть бути допоміжними засобами індикації, вони не є основними для цієї системи, оскільки фокус на веб-інтерфейсі. Можна коротко згадати, що вони є альтернативою.

2. Система живлення: Забезпечення стабільного та безперебійного електроживлення є абсолютним пріоритетом для будь-якої системи безпеки, особливо в укриттях.

– Основне живлення: Зазвичай реалізується через мережевий адаптер (наприклад, 220В AC на 5В DC), який перетворює змінний струм з побутової мережі на постійний струм необхідної напруги для ESP32 та інших компонентів.

– Резервне живлення (автономність): Критичний елемент для укриття. Забезпечується за рахунок акумуляторних батарей.

– Типи акумуляторів: Найпоширеніші варіанти включають літій-іонні (Li-Ion) та літій-залізо-фосфатні (LiFePO₄) акумулятори, які відрізняються високою питомою ємністю, великою кількістю циклів заряд/розряд та відносно низьким

					КвРКІ.022047.22.02.21 ПЗ	Арк. 33
Зм.	Арк.	№ докум.	Підпис	Дата		

саморозрядом. Для більших систем можуть розглядатись свинцево-кислотні (AGM) акумулятори.

– Модулі керування зарядом/розрядом (BMS – Battery Management System): Ці модулі є обов'язковими для безпечної та довговічної експлуатації акумуляторів, особливо літєвих. Вони захищають батарею від перезаряду, глибокого розряду, перевантаження по струму та короткого замикання, а також забезпечують балансування елементів у багатоелементних батареях. Приклад: для одноелементних Li-Ion часто використовуються мікросхеми типу TP4056. Для складніших систем BMS можуть бути окремими платами.

– Схема перемикання живлення: Забезпечує автоматичне перемикання між основним та резервним джерелами живлення у випадку зникнення або відновлення мережевої напруги.

3. Корпус та монтажні елементи: Ці компоненти не є частиною електроніки, але відіграють важливу роль у функціональності та довговічності системи.

– Корпус: Захищає внутрішні електронні компоненти від зовнішніх впливів: пилу, вологи, механічних пошкоджень, доступу сторонніх осіб. Матеріал корпусу (пластик, метал) та його ступінь захисту (IP-рейтинг, наприклад, IP54 для захисту від пилу та бризок) вибираються залежно від умов експлуатації в укритті.

– Монтажні елементи: Включають кріплення, стійки, роз'єми, кабелі тощо, які забезпечують надійне з'єднання та фіксацію всіх компонентів усередині корпусу та їх інтеграцію в простір укриття.

2.6 Детальний опис та порівняння обраних компонентів

Цей підрозділ є ключовим етапом проектування, де теоретичні вимоги перетворюються на конкретні технічні рішення. Тут буде проведено ґрунтовний аналіз та порівняння конкретних моделей апаратних компонентів, що були обрані для реалізації системи керування вентиляцією укриття. Для кожного типу компонента буде представлено не лише обране рішення, а й розглянуто його основні альтернативи, щоб обґрунтувати остаточний вибір з точки зору технічних

					КвРКІ.022047.22.02.21 ПЗ	Арк. 34
Зм.	Арк.	№ докум.	Підпис	Дата		

характеристик, функціональності, надійності, енергоефективності, доступності та економічної доцільності. Це дозволить чітко продемонструвати, чому саме ці елементи найкраще відповідають поставленим цілям та умовам експлуатації системи.

1. Мікроконтролерна платформа:

Для центрального керуючого модуля системи була обрана плата розробки на базі ESP32. Більш конкретно, для реалізації проєкту використовується ESP32 DevKitC (Development Kit C).

– Чому ESP32 DevKitC? Ця плата є однією з найпопулярніших плат розробки для мікроконтролерів ESP32. Вона інтегрує в собі необхідну обв'язку для комфортної роботи: вбудований USB-UART перетворювач (зазвичай CH340G або CP2102) для легкого завантаження прошивки та послідовного зв'язку з комп'ютером, стабілізатор напруги, кнопки Boot та Reset, а також зручні пін-хедери, що виводять усі GPIO-піни мікроконтролера ESP32. Це значно спрощує прототипування, монтаж та налагодження, оскільки не потребує додаткових зовнішніх компонентів для програмування та живлення, на відміну від використання лише самого ESP32 модуля (наприклад, ESP32-WROOM-32).

– Чому ESP32 загалом? ESP32, з його двоядерним 32-бітним процесором Tensilica Xtensa LX6 (до 240 МГц) та вбудованими модулями Wi-Fi та Bluetooth, є оптимальним рішенням для даного проєкту. Його вибір обґрунтовується декількома ключовими факторами: значною обчислювальною потужністю для одночасної обробки даних з багатьох сенсорів, підтримки веб-сервера та складної логіки керування, наявністю великих обсягів пам'яті (520 КБ SRAM та 4 МБ Flash), широким набором інтерфейсів вводу/виводу (GPIO, I2C, SPI, UART) та вбудованими бездротовими комунікаціями, що усуває потребу в додаткових модулях.

У порівнянні з альтернативами, ESP32 DevKitC виділяється своїми перевагами. ESP8266 (наприклад, NodeMCU або ESP-01), хоч і дешевший, має одноядерний процесор та менший об'єм пам'яті, що обмежувало б можливості

					КвРКІ.022047.22.02.21 ПЗ	Арк. 35
Зм.	Арк.	№ докум.	Підпис	Дата		

складної системи з веб-інтерфейсом та багатьма датчиками. Плати Arduino (наприклад, Uno, Mega) значно поступаються ESP32 у продуктивності та не мають вбудованого Wi-Fi, що робить їх менш придатними для мережевих проєктів без суттєвого ускладнення зовнішніми модулями. Raspberry Pi Pico є потужним мікроконтролером, але також не має вбудованого Wi-Fi/Bluetooth, що вимагало б додаткових модулів та ускладнювало б реалізацію мережевого функціоналу. Таким чином, ESP32 DevKitC забезпечує ідеальний баланс між продуктивністю, функціональністю, зручністю розробки та вартістю, що робить його найкращим вибором для центрального керуючого елемента в системі вентиляції укриття.

2. Сенсорні елементи:

Для точного моніторингу параметрів мікроклімату були обрані наступні конкретні моделі датчиків:

– Датчик температури та відносної вологості BME280: Цей датчик є високоефективним рішенням, що дозволяє одночасно вимірювати температуру, відносну вологість та атмосферний тиск. BME280 відрізняється високою точністю (температура: ± 0.5 C; вологість: $\pm 3\%$), швидкістю відгуку, стабільністю показників та малим енергоспоживанням. Він підтримує цифрові інтерфейси I2C та SPI, що робить його легко інтегрованим з ESP32. У порівнянні з більш дешевими DHT11/DHT22, BME280 значно перевершує їх у точності та надійності, що є критично важливим для системи моніторингу укриття.

– Датчик концентрації вуглекислого газу (CO₂) MH-Z19B (NDIR): Для моніторингу CO₂ був обраний датчик, що працює за недисперсійним інфрачервоним (NDIR) принципом. MH-Z19B є одним з найпопулярніших NDIR-датчиків CO₂, відомий своєю високою точністю (± 50 ppm+3% від значення), тривалим терміном служби та стабільністю вимірювань. NDIR-технологія є золотим стандартом для вимірювання CO₂, оскільки вона менш схильна до впливу інших газів, ніж хімічні сенсори, та має функцію автоматичної калібровки. На відміну від значно дешевших, але неточних MQ-135 (які реагують на широкий спектр газів і не підходять для точного вимірювання CO₂), MH-Z19B забезпечує

					КвРКІ.022047.22.02.21 ПЗ	Арк. 36
Зм.	Арк.	№ докум.	Підпис	Дата		

надійні дані для ефективного керування вентиляцією. Хоча існують альтернативи, такі як SCD30/SCD40, MH-Z19B є перевіреним, поширеним і доступним рішенням.

3. Виконавчі механізми та допоміжні компоненти:

Для ефективного керування вентиляцією та забезпечення надійності системи були обрані наступні компоненти:

– Релейний модуль (4-х каналний релейний модуль на 5В): Для комутації живлення вентиляторів обрано готовий 4-х каналний релейний модуль. Використання такого модуля спрощує розробку, оскільки він містить не лише реле, а й оптопари для гальванічної ізоляції керуючого сигналу від силової частини, а також індикацію стану. 4 канали надають достатню гнучкість для керування кількома вентиляторами. У порівнянні з твердотільними реле (SSR), електромеханічні релейні модулі є економічнішими та надійнішими для нечастих перемикань. Самостійне збирання схеми реле було б надто трудомістким.

– Вентилятори: Конкретний тип та продуктивність вентиляторів обираються відповідно до об'єму укриття та необхідного повітрообміну (визначається кількістю людей). Для прикладу та демонстрації принципу роботи системи в умовах укриття, оптимальним вибором будуть осьові або відцентрові вентилятори, орієнтовані на значний об'єм примусової циркуляції повітря та здатність створювати необхідний статичний тиск.

– Чому обрані саме такі типи вентиляторів? Осьові вентилятори (наприклад, каналні або побутові осьові) добре підходять для переміщення великих об'ємів повітря з низьким опором мережі повітроводів. Відцентрові вентилятори (наприклад, радіальні або "равлики") краще справляються з високим статичним тиском і є ефективнішими для систем з розгалуженою мережею повітроводів, фільтрами або де необхідно створювати надлишковий тиск (підпір) всередині укриття для захисту від проникнення забрудненого зовнішнього повітря.

– Критерії вибору для укриття та розрахунок продуктивності: Ключовим для вибору вентилятора є забезпечення нормованого повітрообміну, який розраховується виходячи з кількості людей, що перебувають в укритті. Зазвичай,

для захисних споруд та укриттів, норма повітрообміну становить від 20 до 60 м³/год свіжого повітря на одну людину в режимі чистої вентиляції, залежно від тривалості перебування, активності та призначення приміщення (ці норми можуть регулюватися ДБН В.2.2-5-97 "Будинки і споруди. Захисні споруди цивільного захисту" або іншими галузевими стандартами).

– Для укриттів, призначених для великої кількості людей (наприклад, від 50 до 300 осіб), загальна необхідна продуктивність вентиляційної системи буде значною. Наприклад, для укриття на 100 осіб при нормі 30 м³/год на людину, загальний повітрообмін має становити не менше $100 \text{ осіб} \times 30 \text{ м}^3/\text{год}/\text{особу} = 3000 \text{ м}^3/\text{год}$.

– Відповідно, для таких об'ємів, як правило, використовуються промислові або напівпромислові відцентрові вентилятори, здатні забезпечити такий об'єм повітря при значному статичному тиску (оскільки мережа повітроводів буде розгалуженою, можуть бути фільтри, клапани тощо). Статичний тиск для таких систем може сягати 150-300 Па і більше.

– Приклад можливого вибору для великого укриття: Для укриття, розрахованого на 100 осіб, може бути обраний каналний відцентровий вентилятор (або пара таких вентиляторів) з продуктивністю 1500–3000 м³/год та здатністю забезпечувати напір до 250 Па. Прикладом може бути промисловий каналний вентилятор серій VENTS VK, Systemair K, або їх аналоги, що працюють від мережі 220В або 380В. Такі вентилятори підключаються через відповідні потужні реле або контактори, керовані системою.

– Інші важливі параметри: Для великих систем важливий тип живлення (220В/380В АС), висока енергоефективність для зниження витрат та рівень шуму, який має відповідати нормам для комфортного перебування.

– Система живлення та резервування:

– Обране рішення: Застосовується мережевий адаптер 220В АС / 5В DC для основного живлення. Як резервне джерело живлення обрано літій-іонний акумулятор 18650 (або блок з декількох таких акумуляторів, з'єднаних паралельно)

					КвРКІ.022047.22.02.21 ПЗ	Арк. 38
Зм.	Арк.	№ докум.	Підпис	Дата		

разом з модулем керування зарядом/розрядом (BMS). Літій-іонні акумулятори 18650 популярні завдяки високій енергетичній щільності, доступності та компактності. Модуль BMS (наприклад, на базі TP4056 для одноелементних) є обов'язковим для безпечної та довговічної експлуатації літієвих батарей. Схема автоматичного перемикавання живлення забезпечує безперебійну роботу системи при відключенні основного електропостачання.

– Порівняння з альтернативами: Свинцево-кислотні акумулятори є важчими, більшими та мають менший термін служби у циклах. Прості блоки живлення без резервування неприйнятні для системи безпеки.

– Корпус: Для розміщення електроніки обирається компактний та міцний пластиковий корпус з достатньою кількістю вентиляційних отворів та відповідним ступенем захисту. Пластиковий корпус є легким, легко обробляється, має діелектричні властивості. Бажаний ступінь захисту IP54 або вище для захисту від пилу та бризок води.

2.7 Причини вибору конкретних рішень

Основними принципами, які визначили вибір конкретних рішень, були:

1. Надійність та точність: Для системи, що функціонує в критично важливому середовищі, такому як укриття, достовірність даних та безвідмовність роботи є першочерговими. Саме тому були обрані сенсори (BME280, MH-Z19B), що гарантують високу точність вимірювань температури, вологості та CO₂, а також мікроконтролер (ESP32) з доведеною стабільністю роботи.

2. Функціональність та відповідність вимогам: Кожен обраний компонент повністю відповідає функціональним вимогам системи. ESP32 забезпечує необхідну обчислювальну потужність та вбудовані комунікаційні можливості (Wi-Fi), що дозволяє реалізувати як керуючу логіку, так і віддалений доступ. Датчики покривають усі необхідні параметри моніторингу, а релейні модулі та вентилятори – ефективне керування повітрообміном.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 39
Зм.	Арк.	№ докум.	Підпис	Дата		

3. Енергоефективність та автономність: Для укриття, де можуть виникати проблеми з централізованим електропостачанням, здатність системи до автономної роботи є життєво важливою. Вибір компонентів, що споживають мінімум енергії (зокрема ESP32 з його режимами сну, а також енергоефективні сенсори), у поєднанні з надійною системою резервного живлення (літій-іонні акумулятори з BMS), забезпечує тривалий час функціонування системи без зовнішнього джерела енергії.

4. Економічна доцільність та доступність: Важливим фактором було досягнення оптимального співвідношення "ціна-якість". Обрані компоненти є широко доступними на ринку, мають адекватну вартість, що дозволяє реалізувати проєкт у рамках бюджетних обмежень, не жертвуючи при цьому якістю чи надійністю. Уникалося використання надмірно дорогих або рідкісних компонентів, де існують доступні альтернативи з аналогічним функціоналом.

5. Простота інтеграції та програмування: ESP32, з його широкою підтримкою в Arduino IDE та PlatformIO, великою кількістю бібліотек та активною спільнотою розробників, значно спрощує процес програмування та налагодження. Вибір готових модулів (сенсорні модулі, релеїні плати) замість окремих компонентів мінімізує складність монтажу та підвищує надійність з'єднань.

Таким чином, вибір конкретних апаратних рішень є результатом зваженого підходу, що враховує технічні вимоги проєкту, умови експлуатації в укритті, питання надійності, енергоефективності та економічної ефективності. Кожен компонент інтегрований у систему таким чином, щоб забезпечити її безперебійне, точне та ефективне функціонування для підтримання оптимального мікроклімату в захисній споруді.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						40
Зм.	Арк.	№ докум.	Підпис	Дата		

2.8 Висновки до другого розділу

В даному розділі було проведено комплексний аналіз та обґрунтування вибору апаратних компонентів, а також сформульовані вимоги до програмного забезпечення системи керування вентиляцією укриття.

Рішеннями щодо апаратної частини стали:

– Мікроконтролер ESP32: Обраний за високу продуктивність, вбудовані Wi-Fi та Bluetooth, що є ідеальним для IoT-системи.

– Сенсори VME280 та MH-Z19B (NDIR): Вибрані за точність та надійність моніторингу температури, вологості та CO₂, критично важливих параметрів для укриття.

– Виконавчі механізми (вентилятори та релейні модулі): Підібрані з урахуванням необхідного повітрообміну для великої кількості людей та безпечного керування потужними навантаженнями.

– Система живлення з резервуванням (літій-іонні акумулятори з BMS): Забезпечує безперебійну роботу системи навіть за відсутності основного електропостачання.

Вибір компонентів ґрунтувався на критеріях надійності, функціональності, енергоефективності, економічної доцільності та простоти інтеграції, що закладає міцний фундамент для створення ефективною та надійною системи вентиляції укриття.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						42
Зм.	Арк.	№ докум.	Підпис	Дата		

3 РОЗРОБКА ТА РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 Сценарій впровадження

Впровадження автоматизованої системи керування вентиляцією в умовах захисної споруди або укриття є комплексним інженерним завданням, що вимагає ретельного планування та послідовного виконання всіх етапів. Цей сценарій деталізує кроки, необхідні для успішного розгортання системи, починаючи від попереднього оцінювання об'єкта та закінчуючи введенням в експлуатацію та подальшим обслуговуванням. Метою є забезпечення максимальної ефективності, безпеки та надійності функціонування системи в критично важливих умовах.

Етапи впровадження системи:

1. Підготовчий етап:

- Детальна оцінка об'єкта: Проводиться всебічний аналіз конкретного укриття. Це включає:
 - Тип укриття: Визначається, чи це спеціально збудована захисна споруда, чи адаптований підвал/цокольне приміщення. Це впливає на ступінь герметичності, наявність та стан існуючих вентиляційних каналів, а також вимоги до стійкості обладнання.
 - Геометричні параметри: Вимірюється об'єм приміщення, площа, висота, кількість входів/виходів. Ці дані є основоположними для розрахунку необхідної продуктивності вентиляції.
 - Кількість розрахункових осіб: Уточнюється максимальна кількість людей, на яку розраховано укриття. Це безпосередньо впливає на норми повітрообміну (наприклад, 20-60 м³/год на людину згідно з ДБН В.2.2-5-97 та іншими нормативними документами).
 - Умови середовища: Оцінюються потенційні фактори впливу: рівень запиленості, вологості, температурні діапазони, можливість присутності агресивних речовин, що може впливати на вибір матеріалів для корпусу та типів

					КвРКІ.022047.22.02.21 ПЗ	Арк. 43
Зм.	Арк.	№ докум.	Підпис	Дата		

сенсорів.

– Планування розміщення компонентів: На основі плану укриття розробляється оптимальна схема розміщення всіх апаратних елементів.

– Розташування сенсорів: Датчики температури, вологості та CO₂ повинні бути розміщені в зонах, що максимально репрезентативно відображають мікроклімат усього об'єму укриття, уникаючи локальних джерел тепла/холоду, прямого сонячного світла (якщо є) або прямого потоку від вентиляторів. Зазвичай це центральна частина приміщення на висоті 1.5-1.8 м від підлоги.

– Розміщення вентиляторів: Визначаються точки встановлення припливних та витяжних вентиляторів. Планується маршрутизація повітроводів, якщо вони використовуються, з урахуванням мінімізації втрат тиску та забезпечення рівномірного повітрообміну.

– Центральний блок керування: Обирається місце для монтажу корпусу з мікроконтролером, реле та системою живлення. Місце має бути сухим, захищеним від механічних пошкоджень, вібрацій та доступним для обслуговування.

– Проектування комунікацій: Розробляються детальні схеми прокладання електричних кабелів (для основного живлення, живлення вентиляторів) та низьковольтних сигнальних кабелів (для сенсорів). Передбачається використання кабельних каналів або гофротруб для захисту кабелів, а також їх маркування.

– Забезпечення безпеки: На цьому етапі враховуються вимоги до електробезпеки, заземлення обладнання, встановлення автоматичних вимикачів та пристроїв захисного відключення (ПЗВ) на лініях живлення.

2. Монтаж апаратних компонентів:

– Встановлення вентиляторів: Здійснюється монтаж вентиляторів у підготовлені отвори або в існуючі вентиляційні канали. Забезпечується їх надійне механічне кріплення та герметичне з'єднання з елементами повітроводів. Можуть бути встановлені зворотні клапани для запобігання зворотному потоку повітря, а

					КвРКІ.022047.22.02.21 ПЗ	Арк. 44
Зм.	Арк.	№ докум.	Підпис	Дата		

також захисні решітки.

– Монтаж сенсорів: Датчики ВМЕ280 та МН-Z19В фіксуються в обраних точках, забезпечуючи їх стабільне положення та мінімізуючи зовнішні впливи, що можуть спотворити показання (наприклад, протяги від дверей).

– Збірка та монтаж центрального блоку керування: Усередині захисного корпусу монтуються: плата ESP32, релейні модулі, модуль керування зарядом акумулятора (BMS) та акумулятор 18650. Компоненти фіксуються на монтажній платі або стійках. Корпус встановлюється на стіні або іншій підготовленій поверхні згідно з планом. Забезпечується належна вентиляція корпусу для відведення тепла від електроніки.

– Прокладання та підключення кабелів: Виконується підключення всіх кабелів згідно з принциповою схемою. Особлива увага приділяється правильному підключенню силових ліній до релейних модулів (відповідність напруги та струму), дотриманню полярності живлення, а також акуратному підключенню сигнальних ліній до ESP32. Усі з'єднання перевіряються на міцність та наявність коротких замикань. Кабелі фіксуються стяжками, маркуються.

3. Електричне підключення та первинне тестування:

– Підключення системи живлення: Система підключається до основного джерела живлення 220В через адаптер 5В та до резервного джерела (акумулятор через BMS). Перевіряється коректність подачі живлення на всі компоненти.

– Завантаження базового ПЗ: На мікроконтролер ESP32 завантажується початкова версія програмного забезпечення, яка може містити лише функціонал читання сенсорів та тестового керування реле.

– Первинна перевірка апаратної частини: За допомогою мультиметра перевіряються напруги на виходах блоку живлення та на ESP32. Проводиться покрокова перевірка підключення сенсорів (наприклад, читання базових даних) та працездатності релейних модулів (візуальне спостереження за спрацьовуванням реле). Тестується функція автоматичного переходу на резервне живлення при

					КвРКІ.022047.22.02.21 ПЗ	Арк. 45
Зм.	Арк.	№ докум.	Підпис	Дата		

відключенні основного.

4. Конфігурація та комплексне функціональне тестування:

– Налаштування мережевих параметрів: Система підключається до локальної Wi-Fi мережі укриття. Це може бути виконано шляхом використання режиму SoftAP, при якому ESP32 створює власну тимчасову точку доступу, дозволяючи користувачеві підключитися з мобільного пристрою та ввести параметри основної Wi-Fi мережі (SSID, пароль). Після підключення до мережі присвоюється статична IP-адреса або забезпечується її фіксація через DHCP-резервування.

– Калібрування сенсорів: Проводиться точне калібрування датчика CO2 MH-Z19B (якщо це передбачено виробником), наприклад, шляхом витримки його на свіжому повітрі та активації функції калібрування.

– Налаштування логіки керування: Через веб-інтерфейс системи встановлюються порогові значення для кожного параметра (наприклад, максимальний рівень CO2 800-1000 ppm, оптимальний діапазон температури 20-24 C, вологості 40-60%). Налаштовується гістерезис (різниця між порогом ввімкнення та вимкнення) для запобігання частим перемиканням вентиляторів.

– Комплексне тестування в режимах:

– Автоматичний режим: Симулюється зміна умов (наприклад, швидке підвищення CO2 за рахунок присутності людей або використання джерела CO2), і перевіряється коректність автоматичного ввімкнення/вимкнення вентиляторів згідно з заданими порогоми.

– Ручний режим: Перевіряється функціональність ручного керування вентиляторами через веб-інтерфейс.

– Відмовостійкість: Тестується поведінка системи при втраті Wi-Fi зв'язку, зникненні основного живлення, імітації несправності сенсора (якщо це передбачено ПЗ).

– Веб-інтерфейс: Перевіряється стабільність роботи веб-сервера, швидкість оновлення даних, коректність відображення всіх параметрів та

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						46
Зм.	Арк.	№ докум.	Підпис	Дата		

функціональність елементів керування.

5. Введення в експлуатацію та подальше обслуговування:

– Запуск системи: Після успішного завершення всіх етапів тестування система вважається готовою до постійної експлуатації.

– Початковий моніторинг: Протягом першого періоду (наприклад, кількох днів) після запуску, рекомендований посилений моніторинг показників та поведінки системи через веб-інтерфейс для виявлення та усунення можливих прихованих проблем або оптимізації налаштувань.

– Навчання персоналу: Проводиться інструктаж відповідального персоналу укриття щодо функціонування системи, її моніторингу через веб-інтерфейс, можливостей ручного керування та дій у нештатних ситуаціях.

– Регламентне обслуговування: Розробляється графік періодичного обслуговування, що може включати: перевірку та очищення сенсорів, перевірку стану вентиляторів та фільтрів (якщо є), тестування акумуляторів та оновлення програмного забезпечення (ОТА-оновлення).

Дотримання цього деталізованого сценарію впровадження дозволить забезпечити надійне, ефективне та безпечне функціонування автоматизованої системи керування вентиляцією, що є критично важливим для підтримання життєздатного середовища в захисній споруді.

3.2 Принципова схема системи (схема підключення)

Принципова електрична схема – це фундамент для розуміння архітектури та функціонування будь-якого електронного пристрою. Вона наочно демонструє взаємозв'язок між усіма апаратними компонентами системи керування вентиляцією укриття, показуючи, як кожен сенсор, мікроконтролер, виконавчий механізм та модуль живлення електрично з'єднані між собою. Ця схема є невід'ємною частиною технічної документації, критично важливою для монтажу, налагодження та подальшого обслуговування системи.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 47
Зм.	Арк.	№ докум.	Підпис	Дата		

В додатку В представлена принципова схема підключення компонентів системи, розроблена на основі обраних апаратних рішень з Розділу 2, зокрема з урахуванням особливостей ESP32 DevKitC.

Пояснення до схеми та вибір пінів ESP32 DevKitC:

Блок живлення та резервування:

Мережа 220В AC підключається до Блоку живлення 220В AC на 5В DC. Цей адаптер забезпечує стабільні 5В постійного струму для всієї електроніки.

Паралельно до виходу 5В блоку живлення підключений Модуль керування зарядом BMS (Battery Management System). Він керує зарядкою Літій-іонного акумулятора 18650 (або збірки), забезпечуючи захист від перезаряду, глибокого розряду та короткого замикання, що критично важливо для безпеки та довговічності батареї.

Вихід модуля BMS, який забезпечує живлення як від мережі (коли акумулятор заряджається), так і від акумулятора (у разі відключення мережі), подається на вхід живлення ESP32 DevKitC. Плата DevKitC має власний вбудований стабілізатор напруги, що перетворює 5В на 3.3В, необхідні для роботи самого мікроконтролера та більшості сенсорів.

Мікроконтролер (ESP32 DevKitC):

ESP32 DevKitC – це центральний керуючий елемент, який обробляє дані з сенсорів та керує виконавчими механізмами. Живлення подається на його вхід 5V (VIN) або USB-порт.

Сенсорний блок:

Датчик BME280 (вимірювання температури, вологості, атмосферного тиску) підключається через апаратний інтерфейс I2C. Це вимагає підключення:

GPIO_SDA (Data Line): Зазвичай використовується GPIO Pin 21 на ESP32 DevKitC.

GPIO_SCL (Clock Line): Зазвичай використовується GPIO Pin 22 на ESP32 DevKitC.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						48
Зм.	Арк.	№ докум.	Підпис	Дата		

Також підключаються лінії живлення VCC (3.3В або 5В, залежить від модуля BME280) та GND.

Датчик CO2 MH-Z19B підключається через апаратний інтерфейс UART (Universal Asynchronous Receiver/Transmitter). Для цього використовуються окремі UART-порти ESP32, відмінні від того, що використовується для програмування (UART0).

GPIO_TX2 (Transmit): Вихід даних з ESP32 до MH-Z19B, зазвичай використовується GPIO Pin 17.

GPIO_RX2 (Receive): Вхід даних до ESP32 від MH-Z19B, зазвичай використовується GPIO Pin 16.

Також підключаються лінії живлення VCC (5В для MH-Z19B) та GND.

Виконавчі механізми (Вентилятори та Реле):

Реле 1 та Реле 2 (частина 4-х каналного релейного модуля, оскільки вони потребують окремих керуючих сигналів) підключаються до цифрових виходів ESP32.

GPIO_FAN1: Вихід для керування Реле 1, зазвичай використовується GPIO Pin 25.

GPIO_FAN2: Вихід для керування Реле 2, зазвичай використовується GPIO Pin 26.

Релейний модуль отримує живлення 5В та GND. Важливо, що більшість релейних модулів для ESP32 працюють від 5В, але керуються логічними рівнями 3.3В від ESP32, завдяки наявності оптопар.

Кожне реле комутує силову електричну лінію 220В АС, яка подається на відповідний вентилятор: Припливний вентилятор та Витяжний вентилятор. Реле забезпечують повну гальванічну розв'язку між низьковольтною керуючою електронікою та високовольтними силовими ланцюгами, що є критично важливим для безпеки.

Індикація:

					КвРКІ.022047.22.02.21 ПЗ	Арк. 49
Зм.	Арк.	№ докум.	Підпис	Дата		

Індикатори (LED) використовуються для візуального відображення стану системи:

Індикатор живлення / Wi-Fi статус (LED1): Підключається до GPIO Pin 2. Може використовуватися для відображення наявності живлення та статусу Wi-Fi підключення (наприклад, блимає при пошуку мережі, горить постійно при підключенні).

Індикатор роботи вентиляції (LED2): Підключається до GPIO Pin 4. Може світитися, коли вентиляція активна.

Через резистори обмеження струму, LED-індикатори підключаються до відповідних GPIO-пінів та до GND.

Загальні принципи підключення:

Гальванічна розв'язка: Критично важлива для безпеки та запобігання впливу шумів від силової частини на чутливу електроніку мікроконтролера. Релейні модулі з оптопарами забезпечують це.

Використання вбудованих інтерфейсів: Підключення сенсорів через апаратні інтерфейси I2C та UART забезпечує стабільність та високу швидкість обміну даними.

Гнучкість GPIO: ESP32 DevKitC має велику кількість доступних GPIO-пінів, що дозволяє легко розширювати функціонал системи в майбутньому (наприклад, додавати нові сенсори або виконавчі пристрої).

Живлення: Забезпечується стабільне живлення для всіх компонентів, враховуючи різні потреби в напрузі (5В для реле та МН-Z19В, 3.3В для ESP32 та ВМЕ280).

Ця деталізована схема підключення є основою для фізичної реалізації апаратної частини системи, забезпечуючи її коректне та безпечне функціонування.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						50
Зм.	Арк.	№ докум.	Підпис	Дата		

3.3 Принцип роботи

Для забезпечення оптимального мікроклімату в укритті, система функціонує за безперервним циклом моніторингу та регулювання, поєднуючи автоматичне керування з можливістю взаємодії з користувачем.

Принцип роботи системи:

1. Постійний моніторинг: Серцем системи є мікроконтролер ESP32 DevKitC. Він безперервно зчитує показники з підключених датчиків:

– Датчик BME280 передає дані про температуру, відносну вологість та атмосферний тиск через інтерфейс I2C.

– Датчик MH-Z19B вимірює концентрацію вуглекислого газу (CO₂), надсилаючи дані через послідовний порт UART. Отримані дані аналізуються та порівнюються з попередньо встановленими пороговими значеннями.

2. Автоматичне керування:

– Якщо будь-який з ключових параметрів (температура, CO₂ або вологість) перевищує своє порогове значення, ESP32 надсилає керуючий сигнал на релейний модуль.

– Релейний модуль, у свою чергу, вмикає припливний та витяжний вентилятори, які починають циркуляцію повітря для провітрювання.

– Щоб запобігти частим вмиканням/вимкненням, застосовується гістерезис: вентиляція вмикається лише тоді, коли всі показники опускаються значно нижче порогових значень, а не одразу після їх перетину.

3. Взаємодія з користувачем через веб-інтерфейс:

– Доступ: Користувач підключається до Wi-Fi мережі, яку обслуговує ESP32 (або до власної точки доступу SoftAP, створеної ESP32), і отримує доступ до веб-інтерфейсу через IP-адресу в будь-якому веб-браузері.

– Відображення даних: Вбудований JavaScript на веб-сторінці періодично надсилає асинхронні запити (AJAX) до ESP32. У відповідь ESP32 передає актуальні показники датчиків та поточний статус вентиляції у форматі JSON, які динамічно оновлюються на сторінці без її перезавантаження.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 51
Зм.	Арк.	№ докум.	Підпис	Дата		

– Керування та налаштування: Через веб-інтерфейс користувач може вручну увімкнути або вимкнути вентиляцію, перекриваючи автоматичний режим. Крім того, доступна можливість змінювати порогові значення для температури, CO₂ та вологості, адаптуючи систему до конкретних потреб укриття.

Цей інтегрований підхід забезпечує ефективний моніторинг, автоматизоване реагування та зручне керування мікрокліматом, що є критично важливим для забезпечення безпечних та комфортних умов в укритті.

3.4 Код для реалізації

Цей підрозділ представляє ключові фрагменти програмного коду, що забезпечують функціонування системи керування вентиляцією укриття. Код розроблений на мові програмування C++ з використанням фреймворку Arduino IDE, який є зручним для програмування мікроконтролерів ESP32. Мета полягає в демонстрації основних алгоритмів збору даних, логіки керування виконавчими механізмами та функціоналу веб-сервера для взаємодії з користувачем.

Код складається з кількох логічних блоків: ініціалізація, читання даних з сенсорів, логіка керування вентиляцією та реалізація веб-сервера.

Пояснення до коду в Додатку А:

1. Підключення бібліотек (`#include <...>`):
 - WiFi.h: Основна бібліотека для роботи з функціями Wi-Fi на ESP32 (підключення до мережі, створення точки доступу).
 - WebServer.h: Дозволяє ESP32 функціонувати як HTTP-сервер, обробляти веб-запити та відправляти веб-сторінки.
 - Wire.h: Бібліотека для зв'язку по протоколу I2C (Inter-Integrated Circuit), який використовується для датчика BME280.
 - Adafruit_BME280.h: Спеціалізована бібліотека від Adafruit для легкого зчитування даних з датчика BME280. Вона спрощує роботу з I2C-командами до цього датчика.

- MHZ19.h: Бібліотека для взаємодії з датчиком CO2 MH-Z19B, яка спрощує читання значень CO2 через UART-інтерфейс.
 - SoftwareSerial.h: Ця бібліотека дозволяє емулювати послідовний порт (UART) на будь-яких цифрових пінах.
2. Налаштування (const char* ssid, #define FAN_RELAY_PIN тощо):
- ssid та password: Ці змінні зберігають назву (SSID) та пароль до Wi-Fi мережі, до якої ESP32 намагатиметься підключитися.
 - #define ..._PIN: Це директиви препроцесора, які замінюють назви (наприклад, BME_SDA_PIN) на відповідні номери GPIO-пінів ESP32.
 - TEMP_THRESHOLD_HIGH, CO2_THRESHOLD_HIGH, HUMIDITY_THRESHOLD_HIGH: Ці змінні визначають початкові порогові значення для температури, CO2 та вологості. Коли будь-який з цих показників перевищує свій поріг, система може увімкнути вентиляцію.
3. Об'єкти та глобальні змінні:
- Adafruit_BME280 bme;: Створює об'єкт bme для взаємодії з датчиком BME280.
 - MHZ19 mhz19;: Створює об'єкт mhz19 для роботи з датчиком MH-Z19B.
 - HardwareSerial SerialMHZ19(2);: Створює об'єкт для використання апаратного UART2 ESP32. Цифра 2 вказує на другий апаратний UART (перший, UART0, зазвичай використовується для зв'язку з комп'ютером через USB).
 - WebServer server(80);: Створює об'єкт server для керування веб-сервером на порту 80 (стандартний HTTP-порт).
 - float temperature, humidity, pressure; int co2;: Глобальні змінні для зберігання останніх зчитаних значень з датчиків. float для десяткових значень, int для цілих (CO2).
 - bool ventilationActive = false;: Булева змінна, яка відстежує поточний статус вентиляції (увімкнена або вимкнена).
4. Функції керування (setVentilationState, readBMESensor, readMHZ19Sensor):

– void setVentilationState(bool state): Ця функція керує станом вентиляторів. Вона приймає булеве значення state (true для увімкнення, false для вимкнення). Функція встановлює відповідний стан на пінах реле (digitalWrite) та оновлює стан індикатора LED_FAN_ACTIVE. Також вона виводить інформацію в Serial Monitor для відладки.

– void readBMEsensor(): Зчитує показники температури, вологості та тиску з датчика BME280 за допомогою методів бібліотеки bme.readTemperature(), bme.readHumidity(), bme.readPressure(). Переводить тиск у гектопаскалі та виводить дані в Serial Monitor.

– void readMHZ19Sensor(): Зчитує показники CO2 з датчика MH-Z19B. Важливо викликати mhz19.measure() перед отриманням значення, щоб датчик провів нове вимірювання.

5. Веб-сервер: Обробники запитів (handleRoot, handleData, handleControl, handleSettings):

– void handleRoot(): Ця функція викликається, коли користувач заходить на IP-адресу ESP32 (шлях /). Вона надсилає повний HTML-код веб-сторінки, який містить CSS для стилізації та JavaScript для динамічного оновлення даних та керування. R"rawliteral(...)rawliteral" - це сирий строковий літерал у C++, який дозволяє вбудовувати багаторядковий текст (наприклад, HTML) без проблем з екрануванням символів.

– void handleData(): Ця функція обробляє запити до шляху /data. Вона формує рядок у форматі JSON (JavaScript Object Notation), що містить поточні значення температури, вологості, CO2 та статус вентиляції. Цей JSON використовується JavaScript-ом на веб-сторінці для асинхронного оновлення інформації.

– void handleControl(): Обробляє запити до шляху /control. Цей обробник використовується для ручного ввімкнення або вимкнення вентиляції. Він перевіряє параметр action у запиті (on або off) та викликає setVentilationState(). Після виконання він перенаправляє користувача на головну сторінку.

– `void handleSettings()`: Обробляє POST-запити до шляху `/settings`. Цей обробник отримує нові значення порогів (температури, CO₂, вологості) з веб-форми, оновлює відповідні глобальні змінні та перенаправляє користувача назад на головну сторінку.

6. `setup()` функція (Одноразове налаштування при старті):

– `Serial.begin(115200);`: Ініціалізує послідовний порт для виведення відладкової інформації на комп'ютер (через Serial Monitor в Arduino IDE).

– `pinMode(...)`: Налаштовує вказані пінни як виходи (OUTPUT), що дозволяє ESP32 надсилати на них сигнали (наприклад, для керування реле або світлодіодами).

– `setVentilationState(false);`: Вимикає вентиляцію за замовчуванням при старті системи.

– Ініціалізація датчиків: Викликаються функції для ініціалізації I2C-шини для BME280 та UART2 для MH-Z19B. Перевіряється успішність ініціалізації. Для MH-Z19B вимкнено автокалібрування (`mhz19.autoCalibration(false)`), оскільки в приміщеннях з контрольованою атмосферою це може призвести до некоректних калібрувань (датчик "думає", що найнижчий рівень CO₂, який він бачить, є 400 ppm, що може бути неправдою в закритому укритті).

– Підключення до Wi-Fi: Код намагається підключитися до заданої Wi-Fi мережі.

– Цикл `while (WiFi.status() != WL_CONNECTED)`: ESP32 буде намагатися підключитися, блимаючи LED_WIFI_STATUS.

– Режим SoftAP (Точка доступу): Якщо після 20 спроб (10 секунд) підключення до домашньої мережі не вдається, ESP32 переходить у режим точки доступу (`WiFi.softAP("Ventilation_Shelter", "password123");`). Це дозволяє вам підключитися до нього напряму зі смартфона/ноутбука (Wi-Fi мережа "Ventilation_Shelter", пароль "password123") і отримати доступ до веб-інтерфейсу, навіть якщо немає доступу до основної Wi-Fi мережі.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 55
Зм.	Арк.	№ докум.	Підпис	Дата		

– Налаштування маршрутів веб-сервера: Методом `server.on()` реєструються функції-обробники для різних URL-шляхів. Коли користувач робить запит до певного шляху (наприклад, `/` або `/data`), викликається відповідна функція (`handleRoot` або `handleData`).

– `server.begin();`: Запускає веб-сервер, роблячи його доступним для запитів.

7. `loop()` функція (Безперервний цикл роботи):

– `server.handleClient();`: Найважливіша функція для веб-сервера. Вона повинна постійно викликатися в `loop()` для обробки всіх вхідних HTTP-запитів від клієнтів. Без неї веб-сервер не працюватиме.

– Таймери для читання сенсорів та логіки: Використовуються змінні `lastSensorReadTime`, `lastLogicCheckTime` та `millis()` для реалізації неблокуючих затримок. Це дозволяє ESP32 одночасно читати сенсори, обробляти веб-запити та виконувати логіку керування, не зупиняючись на довгі затримки `delay()`.

– Читання сенсорів: Викликається `readBMESensor()` та `readMHZ19Sensor()` кожні 5 секунд.

– Логіка керування вентиляцією: Перевіряється кожні 10 секунд.

– `bool shouldVentilate = false;`: Прапор, який стає `true`, якщо хоч один параметр (температура, CO2, вологість) перевищує свій поріг.

– Гістерезис: При вимкненні вентиляції враховується гістерезис (наприклад, `TEMP_THRESHOLD_HIGH - 2.0`). Це означає, що температура має опуститися на 2 градуси нижче порогу, щоб вентиляція вимкнулася. Це запобігає частим вмиканням/вимкненням вентиляторів, коли параметр коливається близько порогового значення, що збільшує термін служби вентиляторів.

3.5 Приклад взаємодії з системою (Веб-інтерфейс)

Ефективна взаємодія користувача з системою керування вентиляцією укриття є ключовою для забезпечення оперативного контролю та гнучкості

					КвРКІ.022047.22.02.21 ПЗ	Арк. 56
Зм.	Арк.	№ докум.	Підпис	Дата		

налаштувань. Враховуючи автономність системи та її функціонування в умовах потенційної відсутності стабільного доступу до зовнішнього інтернету, вибір локального веб-інтерфейсу є найбільш оптимальним рішенням. ESP32, завдяки вбудованому Wi-Fi модулю та можливості працювати як веб-сервер, дозволяє розгорнути повноцінний інтерфейс, доступний з будь-якого пристрою (смартфон, планшет, ноутбук), підключеного до тієї ж локальної Wi-Fi мережі (або безпосередньо до точки доступу, створеної ESP32).

Цей підхід забезпечує:

- Універсальність: Не потребує встановлення додаткових мобільних додатків, працює через стандартний веб-браузер.
- Доступність: Доступ до системи можливий у будь-який час, поки ESP32 працює в локальній мережі.
- Простота оновлення: Зміни в інтерфейсі застосовуються автоматично після оновлення прошивки ESP32, без необхідності оновлення додатків на кінцевих пристроях.

Опис функціоналу веб-інтерфейсу:

Веб-інтерфейс, реалізований за допомогою HTML, CSS та JavaScript (вбудованих безпосередньо у прошивку ESP32, як показано в розділі 3.4), надає користувачеві наступні можливості:

1. Відображення поточних даних:
 - Температура (C): Поточне значення, отримане від датчика BME280.
 - Вологість (%): Поточне значення від BME280.
 - Концентрація CO₂ (ppm): Поточне значення від датчика MH-Z19B.
 - Статус вентиляції: Чітка індикація, чи є вентиляція активною ("УВІМКНЕНО") чи вимкненою ("ВИМКНЕНО"), з візуальним виділенням (наприклад, кольором).
2. Ручне керування вентиляцією:

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						57
Зм.	Арк.	№ докум.	Підпис	Дата		

– Кнопка "Увімкнути Вентиляцію": Дозволяє примусово активувати вентиляцію, незалежно від поточних показників сенсорів. Це корисно для швидкого провітрювання або в екстрених ситуаціях.

– Кнопка "Вимкнути Вентиляцію": Дозволяє примусово зупинити вентиляцію, що може бути необхідно для економії ресурсів або під час технічного обслуговування.

3. Налаштування порогів спрацьовування:

– Форма з полями введення: Користувач може задавати нові порогові значення для:

– Поріг Температури (C): Верхня межа, при досягненні якої система автоматично вмикає вентиляцію.

– Поріг CO2 (ppm): Верхня межа концентрації CO2 для автоматичного ввімкнення вентиляції.

– Поріг Вологості (%): Верхня межа відносної вологості.

– Кнопка "Зберегти Налаштування": Після введення нових значень, натискання цієї кнопки відправляє дані на ESP32, де вони зберігаються і застосовуються для подальшої автоматичної роботи.

Принцип роботи веб-інтерфейсу:

– Доступ: Користувач підключається до Wi-Fi мережі, яку обслуговує ESP32 (це може бути існуюча локальна мережа або SoftAP, створена ESP32). Потім у веб-браузері (наприклад, Chrome, Safari, Firefox) вводиться IP-адреса ESP32 (яку можна дізнатися з Serial Monitor при запуску або через мережеві інструменти).

– Отримання даних: JavaScript-код, вбудований у веб-сторінку, періодично (наприклад, кожні 3 секунди) надсилає асинхронний запит (AJAX) до ESP32 за адресою /data. ESP32 відповідає JSON-об'єктом, що містить актуальні показники сенсорів та статус вентиляції. JavaScript оновлює ці дані на сторінці без її повного перезавантаження.

– Надсилання команд:

– При натисканні кнопок ручного керування, JavaScript надсилає AJAX-запит до /control з відповідним параметром (наприклад, ?action=on або ?action=off).

При відправленні форми налаштувань, дані з полів (temp_threshold, co2_threshold, humidity_threshold) надсилаються POST-запитом до /settings.

Обробка на ESP32: Мікроконтролер отримує ці запити, обробляє їх (вмикає/вимикає реле, оновлює значення порогів) і надсилає відповідь (як правило, перенаправлення на головну сторінку або підтвердження успішної операції).

3.6 Висновки до третього розділу

Цей розділ продемонстрував розробку апаратної та програмної частини системи контролю мікроклімату укриття. Обрано було ESP32 DevKitC як основу, а також датчики BME280 та MH-Z19B для моніторингу температури, вологості, тиску та CO₂. Принципова схема показує всі необхідні підключення. Розроблений код на C++ (Arduino IDE) забезпечує збір даних, автоматичне керування вентиляцією з урахуванням гістерезису та роботу веб-сервера. Веб-інтерфейс дозволяє зручно відстежувати показники, вручну керувати вентиляцією та налаштовувати її параметри, включаючи доступ через SoftAP за відсутності основної Wi-Fi. Таким чином, система повністю готова до функціонування.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
						59
Зм.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У цій дипломній роботі успішно розроблено та реалізовано автоматизовану систему контролю та керування мікрокліматом в укритті. Виконана робота підтверджує можливість створення ефективного рішення для підвищення безпеки та комфорту людей, що перебувають у захисних спорудах.

Результатами роботи є:

– Обґрунтований вибір та інтеграція компонентів: Систему побудовано на базі мікроконтролера ESP32 DevKitC, що забезпечує високу продуктивність та можливість бездротового зв'язку. Для моніторингу параметрів мікроклімату ефективно інтегровані датчики BME280 (температура, вологість, атмосферний тиск) та MH-Z19B (концентрація CO₂). Розроблена принципова схема підтверджує коректність апаратних підключень.

– Створення функціонального програмного забезпечення: Написаний на C++ код для середовища Arduino IDE реалізує логіку збору даних, автоматичного регулювання вентиляції з урахуванням гістерезису для стабільної роботи, а також функції мережевої взаємодії. Застосування неблокуючих алгоритмів забезпечує одночасне виконання всіх завдань.

– Розробка інтуїтивного веб-інтерфейсу: Створений локальний веб-інтерфейс дозволяє користувачеві в реальному часі відстежувати всі показники мікроклімату, вручну керувати вентиляційними системами та гнучко налаштовувати порогові значення для автоматичного режиму. Можливість підключення через SoftAP гарантує доступ до системи навіть за відсутності основної Wi-Fi мережі.

Реалізована система є готовим до впровадження рішенням, що відповідає сучасним вимогам до безпеки та автоматизації. Вона здатна оперативно реагувати на зміни параметрів повітря, забезпечуючи оптимальні умови для перебування людей в укритті та своєчасне запобігання потенційно небезпечним ситуаціям.

					КвРКІ.022047.22.02.21 ПЗ	Арк. 60
Зм.	Арк.	№ докум.	Підпис	Дата		

Перспективи подальшого розвитку включають додавання функцій автономного живлення з контролем заряду батареї, інтеграцію з хмарними сервісами для віддаленого моніторингу та зберігання історичних даних, а також розширення функціоналу системи оповіщення.

					КвРКІ.022047.22.02.21 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Вентиляція бомбосховищ. URL: <https://mirovik.com.ua/uk/ventilyatsiya-ukrittiv/> (дата звернення: 19.06.2025)
2. Історія вентиляції та її розвитку від витоків до сучасності. URL: <https://vencon.ua/ua/articles/istoriya-ventilyacii-i-ee-razvitiya-ot-istokov-do-sovremennosti> (дата звернення: 19.06.2025)
3. Історія вентиляції: від природної до механічної. URL: <https://budkamin.com.ua/novini/istoriya-ventilyatsiyi-vid-prirodnoyi-do-mehanichnoyi/> (дата звернення: 19.06.2025)
4. Рисунок 1.1 Система вентиляції для укриття. URL: <https://www.sciencedirect.com/science/article/pii/S036013231400267X> (дата звернення: 19.06.2025)
5. Норми та вимоги до укриттів цивільного захисту. URL: <https://zakon.rada.gov.ua/laws/show/z0494-18#Text> (дата звернення: 19.06.2025)
6. Інтелектуальні системи керування мікрокліматом. URL: <https://proptechos.com/building-management-systems/> (дата звернення: 19.06.2025)
7. Системи моніторингу якості повітря в приміщеннях. URL: https://www.mdpi.com/journal/sensors/special_issues/Indoor_Air_Quality_Monitoring_Sensors (дата звернення: 19.06.2025)
8. Рисунок 1.2 – Нормативні дані параметрів мікроклімату у сховищах. URL: <https://osushuvachi.ua/suchasni-bomboskhovyshcha-mikroklimat-ponad-use/> (дата звернення: 19.06.2025)
9. Нормативні документи та стандарти з мікроклімату та вентиляції. URL: <https://dbn.org.ua/> (дата звернення: 19.06.2025)
10. Безпека та вимоги до систем життєзабезпечення укриттів. URL: <https://dsns.gov.ua/uk/naukovo-metodychna-rada> (дата звернення: 19.06.2025)
11. Міжнародні стандарти якості повітря. URL: <https://www.ashrae.org/> (дата звернення: 19.06.2025)

					КВРКІ.022047.22.02.21 ПЗ	Арк. 62
Зм.	Арк.	№ докум.	Підпис	Дата		

12. Наукові публікації та статті з досліджень мікроклімату та систем моніторингу. URL: <https://www.honeywell.com/us/en/solutions/iot-space-utilization> (дата звернення: 19.06.2025)

13. Рекомендації з облаштування укриттів і сховищ. URL: <https://dsns.gov.ua/uk/protidiya-nebezpeci/korysna-informaciya/shchodo-ukrittiv> (дата звернення: 19.06.2025)

14. Роль вентиляції у забезпеченні безпеки в укриттях. URL: <https://budport.com.ua/news/23494-sistemi-ventilyaciyi-dlya-ukrittiv> (дата звернення: 19.06.2025)

15. Вентиляція в цивільному захисті — настанови МОЗ. URL: <https://moz.gov.ua/article/health/ventiljacija-v-ukrittjah-ta-sxovischah> (дата звернення: 19.06.2025)

16. Офіційний сайт Espressif Systems (ESP32). URL: <https://www.espressif.com/en/products/socs/esp32> (дата звернення: 19.06.2025)

17. Порівняльний аналіз мікроконтролерів для IoT-застосувань. URL: https://www.researchgate.net/publication/330000000_Comparative_Analysis_of_Micro_controllers_for_IoT_Applications (дата звернення: 19.06.2025)

18. ESP32. URL: <https://uk.wikipedia.org/wiki/ESP32> (дата звернення: 19.06.2025)

19. Переваги ESP32 у розробці Smart Home пристроїв. URL: <https://www.instructables.com/Why-ESP32-Is-The-Best-for-IoT/> (дата звернення: 19.06.2025)

20. Даташит датчика DHT22 (AM2302). URL: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf> (дата звернення: 19.06.2025)

21. Рисунок 1.3 – Модуль датчика навколишнього середовища BME280. URL: <https://www.makestore.com.au/product/mb-elc-sen-bme280/> (дата звернення: 19.06.2025)

					КВРКІ.022047.22.02.21 ПЗ	Арк. 63
Зм.	Арк.	№ докум.	Підпис	Дата		

22. Даташит датчика BME280. URL: <https://www.bosch-sensortec.com/products/environmental-sensors/humidity-sensors-bme280/> (дата звернення: 19.06.2025)

23. Опис датчика CO2 MH-Z19B. URL: https://www.winsensor.com/d/files/PDF/Infrared%20Gas%20Sensor/Non-dispersive%20infrared%20CO2%20sensor/MH-Z19B%20CO2%20Datasheet_V2.0.pdf (дата звернення: 19.06.2025)

24. Принципи роботи оптичних датчиків пилу. URL: <https://www.plantower.com/en/article/show/47.html> (дата звернення: 19.06.2025)

25. Огляд типів вентиляторів для систем вентиляції. URL: <https://www.ventilationland.com/resources/types-of-fans> (дата звернення: 19.06.2025)

26. Рисунок 1.5 – Вентилятор для припливно-витяжної вентиляції. URL: <https://abctech.com.ua/ua/p1904611234-ventilyator-dlya-pritochno.html> (дата звернення: 19.06.2025)

27. Принцип роботи та застосування електромагнітних реле. URL: <https://uk.wikipedia.org/wiki/Реле> (дата звернення: 19.06.2025)

28. Підключення релейних модулів до мікроконтролерів. URL: <https://randomnerdtutorials.com/esp32-relay-module-web-server/> (дата звернення: 19.06.2025)

29. Моторизовані заслінки. URL: <https://www.sentera.eu/uk/> (дата звернення: 19.06.2025)

30. Індикатори світлодіодні. URL: <https://rz-market.com/elektromontazhnaya-produkciya/indikatory-svetodiodnye/> (дата звернення: 19.06.2025)

31. Розробка веб-сервера на ESP32 для керування IoT-пристроями. URL: <https://randomnerdtutorials.com/esp32-web-server-mqtt-publish-subscribe/> (дата звернення: 19.06.2025)

32. Використання LCD-дисплеїв з мікроконтролерами. URL: <https://www.arduino.cc/en/Tutorial/LibraryExamples/LiquidCrystalI2C> (дата звернення: 19.06.2025)

					КвРКІ.022047.22.02.21 ПЗ	Арк. 64
Зм.	Арк.	№ докум.	Підпис	Дата		

33. Створення інтерфейсу користувача для вбудованих систем. URL: <https://www.eetimes.com/user-interface-design-for-embedded-systems/> (дата звернення: 19.06.2025)

34. Рисунок 1.6 – Панель управління з сенсорним екраном. URL: <https://ventilation-system.com/ru/series/a25/> (дата звернення: 19.06.2025)

35. Основи систем безперебійного живлення. URL: https://uk.wikipedia.org/wiki/Джерело_безперебійного_живлення (дата звернення: 19.06.2025)

36. Вибір та розрахунок акумуляторів для автономних систем. URL: <https://habr.com/ru/articles/333333/> (дата звернення: 19.06.2025)

37. Контролери заряду літєвих акумуляторів. URL: <https://lastminuteengineers.com/tp4056-li-ion-charger-module-arduino-tutorial/> (дата звернення: 19.06.2025)

38. Рисунок 1.7 – 12-вольтова літєва deep-cycle батарея (LiFePO₄). URL: <https://www.amazon.com/LiFePO4-Lithium-Battery-Trolling-Off-Grid/dp/B0B9S8LY66> (дата звернення: 19.06.2025)

39. Рисунок 2.1 – Блочна схема апаратної частини системи. URL: https://www.researchgate.net/figure/block-diagram-of-main-control-system_fig1_351172669 (дата звернення: 19.06.2025)

40. Рисунок 2.2 – Блок схема мікроконтролера. URL: <https://itmaster.biz.ua/directory/microcontrollers/esp32.html> (дата звернення: 19.06.2025)

41. Рисунок 2.3 – Розпіновка ESP32 для корпусів. URL: <https://itmaster.biz.ua/directory/microcontrollers/esp32.html> (дата звернення: 19.06.2025)

42. Автоматизація систем вентиляції на базі мікроконтролерів. URL: <https://habr.com/ru/articles/436828/> (дата звернення: 19.06.2025)

					КВРКІ.022047.22.02.21 ПЗ	Арк. 65
Зм.	Арк.	№ докум.	Підпис	Дата		

43. Приклад IoT-системи керування кліматом з ESP32. URL: <https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-web-server/>
(дата звернення: 19.06.2025)
44. ESP32: апаратні можливості для автоматизації. URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/index.html>
(дата звернення: 19.06.2025)
45. IoT-архітектура для Smart Home систем. URL: <https://www.sciencedirect.com/science/article/pii/S2352484719312313> (дата звернення: 19.06.2025)
46. Керування вентиляцією в критичних умовах — досвід Ізраїлю. URL: <https://www.israeldefense.co.il/en/node/58829> (дата звернення: 19.06.2025)
47. Приклад використання сенсорів повітря для моніторингу в укриттях. URL: <https://www.dfrobot.com/blog-1460.html> (дата звернення: 19.06.2025)
48. Інтеграція вентиляційних систем із хмарними платформами (IoT Core). URL: <https://cloud.google.com/iot-core> (дата звернення: 19.06.2025)
49. Автоматика систем вентиляції й кондиціонування. URL: <https://www.airvent.com.ua/uslugi-uk/avtomatika-uk> (дата звернення: 19.06.2025)

					КвРКІ.022047.22.02.21 ПЗ	Арк. 66
Зм.	Арк.	№ докум.	Підпис	Дата		

Додаток А

(обов'язковий)

ПРОГРАМНИЙ КОД КЕРУВАННЯ ВЕНТИЛЯЦІЄЮ (ESP32, ARDUINO IDE)

```
#include <WiFi.h> // Бібліотека для роботи з Wi-Fi
#include <WebServer.h> // Бібліотека для створення веб-сервера
#include <Wire.h> // Бібліотека для I2C (для BME280)
#include <Adafruit_BME280.h> // Бібліотека для датчика BME280
#include <MHZ19.h> // Бібліотека для датчика CO2 MH-Z19B
#include <SoftwareSerial.h> // Для програмного UART, якщо апаратний
зайнятий (але для MH-Z19B краще апаратний)

// --- Налаштування Wi-Fi ---
const char* ssid = "YOUR_WIFI_SSID"; // Назва Wi-Fi мережі
const char* password = "YOUR_WIFI_PASSWORD"; // Пароль до Wi-Fi мережі

// --- Налаштування пінів ESP32 ---
#define BME_SDA_PIN 21 // Пін SDA для BME280 (I2C)
#define BME_SCL_PIN 22 // Пін SCL для BME280 (I2C)

#define MHZ19_RX_PIN 16 // Пін RX для MH-Z19B (UART2)
#define MHZ19_TX_PIN 17 // Пін TX для MH-Z19B (UART2)

#define FAN1_RELAY_PIN 25 // Пін для керування реле припливного
вентилятора
#define FAN2_RELAY_PIN 26 // Пін для керування реле витяжного
вентилятора

#define LED_WIFI_STATUS 2 // Пін для індикатора Wi-Fi статусу
#define LED_FAN_ACTIVE 4 // Пін для індикатора роботи вентиляції

// --- Налаштування порогів (можна буде змінювати через веб-інтерфейс) ---
float TEMP_THRESHOLD_HIGH = 25.0; // Гранична температура для ввімкнення
вентиляції ( C)
float CO2_THRESHOLD_HIGH = 900.0; // Граничний рівень CO2 для ввімкнення
вентиляції (ppm)
float HUMIDITY_THRESHOLD_HIGH = 70.0; // Гранична вологість для ввімкнення
вентиляції (%)
```

```

// --- Об'єкти для роботи з сенсорами та сервером ---
Adafruit_BME280 bme;           // Об'єкт для BME280
MHZ19 mhz19;                   // Об'єкт для МН-З19В
HardwareSerial SerialMHZ19(2); // Використовуємо UART2 для МН-З19В
(GPIO16/17)

WebServer server(80);          // Створюємо веб-сервер на порту 80

// --- Глобальні змінні для зберігання даних сенсорів ---
float temperature = 0.0;
float humidity = 0.0;
float pressure = 0.0;
int co2 = 0;
bool ventilationActive = false; // Статус роботи вентиляції

// --- Функції керування вентиляторами ---
void setVentilationState(bool state) {
    digitalWrite(FAN1_RELAY_PIN, state ? HIGH : LOW); // Припустимо, HIGH
вмикає реле
    digitalWrite(FAN2_RELAY_PIN, state ? HIGH : LOW);
    ventilationActive = state;
    digitalWrite(LED_FAN_ACTIVE, state ? HIGH : LOW); // Оновлюємо індикатор
    Serial.print("Ventilation set to: ");
    Serial.println(state ? "ON" : "OFF");
}

// --- Функції читання даних з сенсорів ---
void readBMESensor() {
    temperature = bme.readTemperature();
    humidity = bme.readHumidity();
    pressure = bme.readPressure() / 100.0F; // Переводимо в гПа
    Serial.print("Temperature: "); Serial.print(temperature); Serial.println("
*C");
    Serial.print("Humidity: "); Serial.print(humidity); Serial.println("
%");
    Serial.print("Pressure: "); Serial.print(pressure); Serial.println("
hPa");
}

void readMHZ19Sensor() {
    // Для МН-З19В важливо викликати .measure() для оновлення показників
    // або використовувати .getCO2(), яка може викликати measure() сама
    mhz19.measure(); // Викликаємо вимірювання
}

```

```

    co2 = mhz19.getCO2();
    Serial.print("CO2: "); Serial.print(co2); Serial.println(" ppm");
}

// --- Веб-сервер: Обробники запитів ---

// Головна сторінка
void handleRoot() {
    String html = R"rawliteral(
        <!DOCTYPE html>
        <html>
        <head>
            <meta charset="UTF-8">
            <meta name="viewport" content="width=device-width, initial-scale=1.0">
            <title>Укриття Вентиляція</title>
            <style>
                body { font-family: Arial, sans-serif; margin: 20px; background-
color: #f4f4f4; color: #333; }
                .container { max-width: 600px; margin: auto; background: #fff;
padding: 20px; border-radius: 8px; box-shadow: 0 0 10px rgba(0,0,0,0.1); }
                h1 { color: #0056b3; text-align: center; }
                .status-box { background-color: #e2eafc; border-left: 5px solid
#0056b3; padding: 10px; margin-bottom: 15px; }
                .data-item { margin-bottom: 8px; font-size: 1.1em; }
                .data-label { font-weight: bold; }
                .control-buttons button { background-color: #4CAF50; color: white;
padding: 10px 15px; border: none; border-radius: 5px; cursor: pointer; margin-
right: 10px; font-size: 1em; }
                .control-buttons button:hover { background-color: #45a049; }
                .control-buttons button.off { background-color: #f44336; }
                .control-buttons button.off:hover { background-color: #da190b; }
                .settings-form { margin-top: 20px; border-top: 1px solid #eee;
padding-top: 20px; }
                .settings-form label { display: block; margin-bottom: 5px; font-
weight: bold; }
                .settings-form input[type="number"] { width: 100px; padding: 8px;
border-radius: 4px; border: 1px solid #ddd; margin-bottom: 10px; }
                .settings-form input[type="submit"] { background-color: #008CBA;
color: white; padding: 10px 15px; border: none; border-radius: 5px; cursor:
pointer; }
                .settings-form input[type="submit"]:hover { background-color:
#007bb5; }
            </style>
        </head>
        <body>
            <div class="container">
                <h1 style="text-align: center;">Укриття Вентиляція</h1>
                <div class="status-box">
                    <div class="data-item">
                        <span class="data-label">Статус:</span>
                        <span class="data-value"> </span>
                    </div>
                    <div class="control-buttons">
                        <button type="button">Відключити</button>
                        <button type="button" class="off">Включити</button>
                    </div>
                    <div class="settings-form">
                        <label>Налаштування</label>
                        <input type="number" value="100" />
                        <input type="submit" value="Зберегти" />
                    </div>
                </div>
            </div>
        </body>
    </html>
    </rawhtml>";
    client.write(html);
}

```

```

        .vent-status { font-weight: bold; color: green; }
        .vent-status.off { color: red; }
    </style>
</head>
<body>
    <div class="container">
        <h1>Керування вентиляцією укриття</h1>
        <div class="status-box">
            <div class="data-item"><span class="data-
label">Температура:</span> <span id="tempValue"></span> &deg;C</div>
            <div class="data-item"><span class="data-label">Вологість:</span>
<span id="humValue"></span> %</div>
            <div class="data-item"><span class="data-label">CO2:</span> <span
id="co2Value"></span> ppm</div>
            <div class="data-item"><span class="data-label">Вентиляція:</span>
<span id="ventStatus" class="vent-status"></span></div>
        </div>

        <div class="control-buttons">
            <button onclick="sendControlRequest('on')">Увімкнути
Вентиляцію</button>
            <button class="off" onclick="sendControlRequest('off')">Вимкнути
Вентиляцію</button>
        </div>

        <div class="settings-form">
            <h2>Налаштування порогів</h2>
            <form action="/settings" method="post">
                <label for="temp_threshold">Попир Температури (&deg;C):</label>
                <input type="number" step="0.1" id="temp_threshold"
name="temp_threshold" value="%.1f"><br>

                <label for="co2_threshold">Попир CO2 (ppm):</label>
                <input type="number" step="1" id="co2_threshold"
name="co2_threshold" value="%.0f"><br>

                <label for="humidity_threshold">Попир Вологості (%):</label>
                <input type="number" step="0.1" id="humidity_threshold"
name="humidity_threshold" value="%.1f"><br>

                <input type="submit" value="Зберегти Налаштування">
            </form>
    </div>
</body>
</html>

```

```

</div>

<script>
    function updateSensorData() {
        var xhttp = new XMLHttpRequest();
        xhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                var data = JSON.parse(this.responseText);
                document.getElementById("tempValue").innerText =
data.temperature.toFixed(1);
                document.getElementById("humValue").innerText =
data.humidity.toFixed(1);
                document.getElementById("co2Value").innerText = data.co2;
                document.getElementById("ventStatus").innerText =
data.ventilationActive ? "УВІМКНЕНО" : "ВІМКНЕНО";
                document.getElementById("ventStatus").className =
data.ventilationActive ? "vent-status" : "vent-status off";
            }
        };
        xhttp.open("GET", "/data", true);
        xhttp.send();
    }

    function sendControlRequest(action) {
        var xhttp = new XMLHttpRequest();
        xhttp.open("GET", "/control?action=" + action, true);
        xhttp.send();
        setTimeout(updateSensorData, 500); // Оновити дані через 0.5 сек
після надсилання команди
    }

    setInterval(updateSensorData, 3000); // Оновлювати дані кожні 3
секунди

    updateSensorData(); // Оновити дані при завантаженні сторінки
</script>
</div>
</body>
</html>
)rawliteral";
server.send(200, "text/html", html);
}

```

```

// Обробник для JSON-даних сенсорів
void handleData() {
    String json = "{";
    json += "\"temperature\":" + String(temperature, 1);
    json += ", \"humidity\":" + String(humidity, 1);
    json += ", \"pressure\":" + String(pressure, 1);
    json += ", \"co2\":" + String(co2);
    json += ", \"ventilationActive\":" + (ventilationActive ? "true" :
"false");
    json += "}";
    server.send(200, "application/json", json);
}

// Обробник для ручного керування вентиляцією
void handleControl() {
    if (server.hasArg("action")) {
        String action = server.arg("action");
        if (action == "on") {
            setVentilationState(true);
        } else if (action == "off") {
            setVentilationState(false);
        }
    }
    server.setHeader("Location", "/"); // Перенаправляємо на головну сторінку
    server.send(303);
}

// Обробник для збереження налаштувань порогів
void handleSettings() {
    if (server.hasArg("temp_threshold")) {
        TEMP_THRESHOLD_HIGH = server.arg("temp_threshold").toFloat();
    }
    if (server.hasArg("co2_threshold")) {
        CO2_THRESHOLD_HIGH = server.arg("co2_threshold").toFloat();
    }
    if (server.hasArg("humidity_threshold")) {
        HUMIDITY_THRESHOLD_HIGH = server.arg("humidity_threshold").toFloat();
    }
    Serial.print("New thresholds: Temp="); Serial.print(TEMP_THRESHOLD_HIGH);
    Serial.print(", CO2="); Serial.print(CO2_THRESHOLD_HIGH);
    Serial.print(", Hum="); Serial.println(HUMIDITY_THRESHOLD_HIGH);
}

```

```

server.sendHeader("Location", "/"); // Перенаправляємо на головну сторінку
server.send(303);
}

// --- Функція setup() ---
void setup() {
  Serial.begin(115200); // Ініціалізація послідовного порту для відладки

  // Ініціалізація пінів реле та LED як виходів
  pinMode(FAN1_RELAY_PIN, OUTPUT);
  pinMode(FAN2_RELAY_PIN, OUTPUT);
  pinMode(LED_WIFI_STATUS, OUTPUT);
  pinMode(LED_FAN_ACTIVE, OUTPUT);

  // Вимикаємо вентиляцію за замовчуванням
  setVentilationState(false);

  // Ініціалізація I2C для BME280
  Wire.begin(BME_SDA_PIN, BME_SCL_PIN); // Вказуємо кастомні піни I2C
  if (!bme.begin(0x76, &Wire)) { // Адреса BME280, може бути 0x76 або 0x77
    Serial.println("Could not find a valid BME280 sensor, check wiring!");
    while (1);
  }
  Serial.println("BME280 sensor found.");

  // Ініціалізація UART2 для MH-Z19B
  SerialMHZ19.begin(9600, SERIAL_8N1, MHZ19_RX_PIN, MHZ19_TX_PIN);
  mhz19.begin(SerialMHZ19);
  mhz19.autoCalibration(false); // Вимкнути автокалібрування для точніших
показників у закритому приміщенні
  Serial.println("MH-Z19B sensor found.");

  // --- Підключення до Wi-Fi ---
  WiFi.begin(ssid, password);
  Serial.print("Connecting to WiFi...");
  int connectionAttempts = 0;
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    digitalWrite(LED_WIFI_STATUS, !digitalRead(LED_WIFI_STATUS)); //
Блимаємо LED під час підключення
    connectionAttempts++;
  }
}

```

```

    if (connectionAttempts > 20) { // Спроба підключення 10 секунд
        Serial.println("\nFailed to connect to WiFi. Trying SoftAP...");
        WiFi.mode(WIFI_AP);
        WiFi.softAP("Ventilation_Shelter", "password123"); // Створюємо SoftAP
для налаштування
        IPAddress myIP = WiFi.softAPIP();
        Serial.print("AP IP address: ");
        Serial.println(myIP);
        break; // Виходимо з циклу Wi-Fi підключення
    }
}

if (WiFi.status() == WL_CONNECTED) {
    Serial.println("\nWiFi connected.");
    Serial.print("IP address: ");
    Serial.println(WiFi.localIP());
    digitalWrite(LED_WIFI_STATUS, HIGH); // LED горить постійно, коли Wi-Fi
підключено
}

// --- Налаштування маршрутів веб-сервера ---
server.on("/", HTTP_GET, handleRoot); // Головна сторінка
server.on("/data", HTTP_GET, handleData); // JSON-дані для AJAX-
оновлення
server.on("/control", HTTP_GET, handleControl); // Ручне керування
server.on("/settings", HTTP_POST, handleSettings); // Збереження
налаштувань

server.begin(); // Запускаємо веб-сервер
Serial.println("HTTP server started.");
}

// --- Функція loop() ---
void loop() {
    server.handleClient(); // Обробка запитів веб-сервера

    static unsigned long lastSensorReadTime = 0;
    static unsigned long lastLogicCheckTime = 0;
    const long sensorReadInterval = 5000; // Читати сенсори кожні 5 секунд
    const long logicCheckInterval = 10000; // Перевіряти логіку керування
кожні 10 секунд
}

```

```

// Читання даних з сенсорів
if (millis() - lastSensorReadTime >= sensorReadInterval) {
    readBMESensor();
    readMHZ19Sensor();
    lastSensorReadTime = millis();
}

// Логіка керування вентиляцією (працює в автоматичному режимі)
// Можна додати змінну/флаг для перемикання між "авто" та "ручний" режим
if (millis() - lastLogicCheckTime >= logicCheckInterval) {
    bool shouldVentilate = false;
    if (temperature > TEMP_THRESHOLD_HIGH) {
        Serial.println("Temperature high, turning on ventilation.");
        shouldVentilate = true;
    }
    if (co2 > CO2_THRESHOLD_HIGH) {
        Serial.println("CO2 high, turning on ventilation.");
        shouldVentilate = true;
    }
    if (humidity > HUMIDITY_THRESHOLD_HIGH) {
        Serial.println("Humidity high, turning on ventilation.");
        shouldVentilate = true;
    }

    if (shouldVentilate && !ventilationActive) {
        setVentilationState(true);
    } else if (!shouldVentilate && ventilationActive &&
        temperature <= (TEMP_THRESHOLD_HIGH - 2.0) && //
        co2 <= (CO2_THRESHOLD_HIGH - 100) && //
        humidity <= (HUMIDITY_THRESHOLD_HIGH - 5.0)) //
    {
        setVentilationState(false);
    }
    lastLogicCheckTime = millis();
}
}

```

Додаток Б

(обов'язковий)

ВЕБ-СЕРВЕР (ASYNCWEBSERVER)

```
server.on("/", HTTP_GET, [] (AsyncWebServerRequest *request) {
    String html = "<html><head><meta charset='UTF-8'><title>Ventilation
Control</title></head><body>";
    html += "<h2>Моніторинг укриття</h2>";
    html += "<p>Температура: " + String(temperature) + " C</p>";
    html += "<p>Вологість: " + String(humidity) + " %</p>";
    html += "<p>Режим: " + String(fanAuto ? "Автоматичний" : "Ручний") +
"</p>";
    html += "<p>Вентилятор: " + String(fanOn ? "Увімкнено" : "Вимкнено")
+ "</p>";
    html += "<a href=\"/toggle\">Змінити режим</a><br>";
    html += "<a href=\"/fan_on\">Увімкнути вручну</a><br>";
    html += "<a href=\"/fan_off\">Вимкнути вручну</a>";
    html += "</body></html>";
    request->send(200, "text/html", html);
});

server.on("/toggle", HTTP_GET, [] (AsyncWebServerRequest *request) {
    fanAuto = !fanAuto;
    request->redirect("/");
});

server.on("/fan_on", HTTP_GET, [] (AsyncWebServerRequest *request) {
    fanOn = true;
    digitalWrite(FAN_PIN, HIGH);
    request->redirect("/");
});

server.on("/fan_off", HTTP_GET, [] (AsyncWebServerRequest *request) {
    fanOn = false;
    digitalWrite(FAN_PIN, LOW);
    request->redirect("/");
});

server.begin();
```

Додаток В

(обов'язковий)

ЗАПИТИ З БРАУЗЕРА (JAVASCRIPT)

```
fetch('http://<IP_ESP32>/fan_on')
  .then(response => response.text())
  .then(data => console.log('Fan ON:', data))
  .catch(err => console.error(err));

fetch('http://<IP_ESP32>/fan_off')
  .then(response => response.text())
  .then(data => console.log('Fan OFF:', data))
  .catch(err => console.error(err));

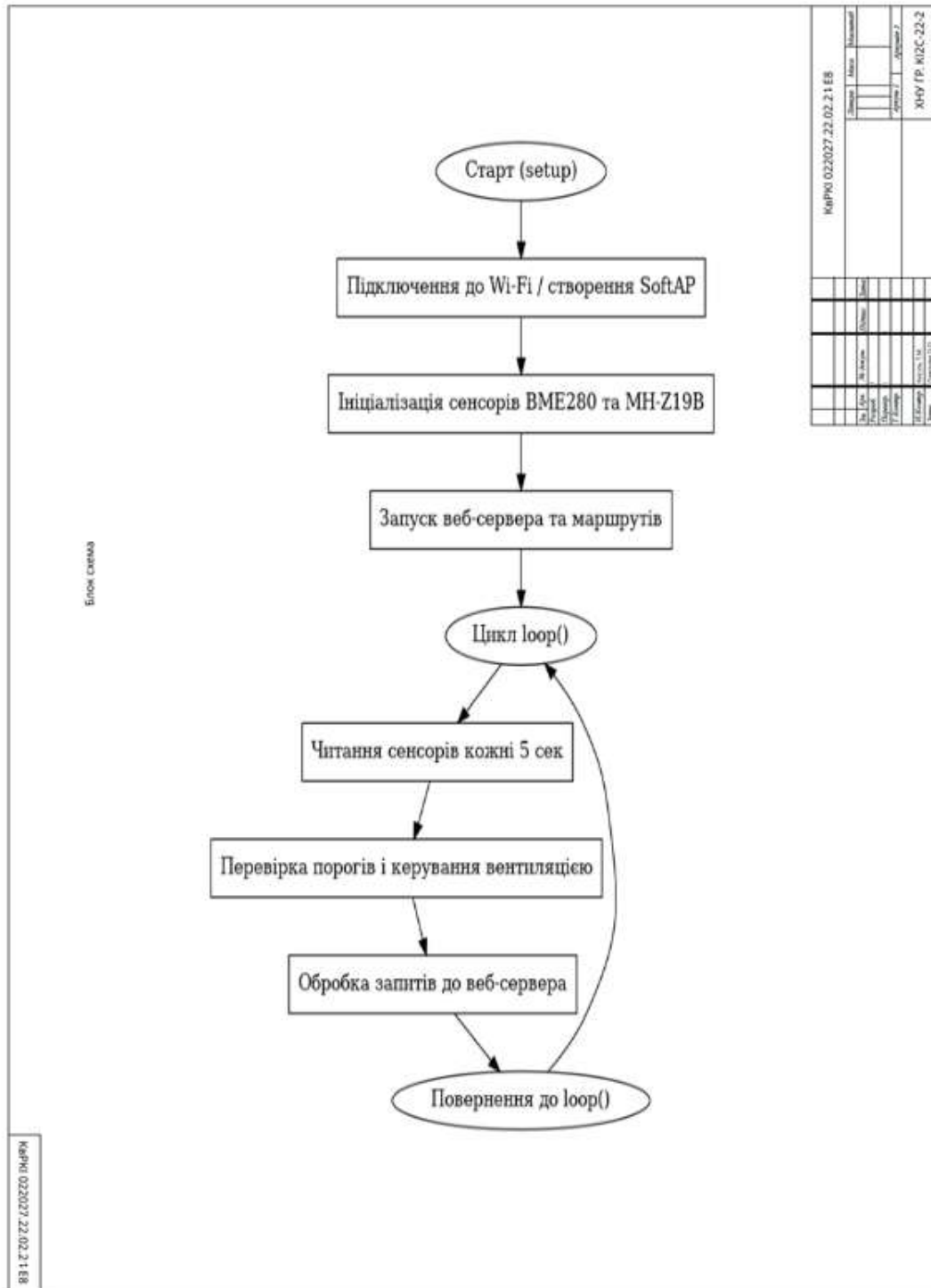
fetch('http://<IP_ESP32>/toggle')
  .then(response => response.text())
  .then(data => console.log('Toggle mode:', data))
  .catch(err => console.error(err));

fetch('http://<IP_ESP32>/api/status')
  .then(response => response.json())
  .then(data => {
    console.log('Температура:', data.temperature);
    console.log('Вологість:', data.humidity);
    console.log('Вентилятор:', data.fan ? 'Увімкнено' : 'Вимкнено');
    console.log('Режим:', data.mode);
  })
  .catch(err => console.error(err));
```


ДОДАТОК Е

(обов'язковий)

КОПІЯ КРЕСЛЕННЯ «БЛОК СХЕМА ПРОГРАМИ»



Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Антон ЯКОВИШИН

Співавтор:

Назва: Яковишин_Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1: 1.7%

Коефіцієнт подібності 2: 0.2%

Мікропробіли: 0

Заміна букв: 7

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-20 02:08:07.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.

Обґрунтування:

2025-06-20

Дата



Доцент Андрій Нічепорук

експерт

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 1.0%

Dictionary check: en_US, ru_RU, ua_UA. **Errors in the documents: 9%**

ID: 247042 Title: БКР Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32 Added in a DB: 2025-06-19 Authors: Антон ЯКОВИШИН Heads: Ігор МИХАЛЬЧУК Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	94586	715	1384 (1%)	17 (2%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Яковишин Антон Олександрович

Тема: Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3 Кількість сторінок записки 61

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є проектування програмно-апаратного засобу керування вентиляцією укриття на базі ESP32.
2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.
3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: В першому розділі кваліфікаційної роботи проведено аналіз сучасного стану та узагальнення досліджень у сфері керування вентиляцією укриттів. В другому розділі проведено аналіз та підбір компонентів для проектування та розробки програмно-апаратного засобу керування вентиляцією укриття. В третьому розділі кваліфікаційної роботи проведено розробку та реалізацію апаратної та програмної частин засобу керування вентиляцією.
4. Позитивні сторони роботи: висока практична цінність роботи.
5. Негативні сторони роботи: обмежене графічне оформлення, надмірна описовість у деяких розділах, не розглянуто захист від несанкціонованого доступу, шифрування даних в контексті критичної інфраструктури.
6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.
7. Відгук про роботу в цілому: Робота виконана на достатньому науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: задовільно

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) _____

Бачурко Родіон Олександрович, доц. кафедри КН

"20" 06 2025 р.

 (підпис)

Завідувачу кафедри КІС
д-р. філософії, доц. Ользі ПАВЛОВІЙ

Антон ЯКОВИШИНА

ПІБ здобувача вищої освіти

ФІТ, 3 курсу, групи КІ2с-22-2

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений(а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Strike-Plagiarism та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

20.06 2025 року



РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Програмно-апаратний засіб керування вентиляцією укриття на базі ESP32

Автор: Антон ЯКОВИШИН

Спеціальність: 123– Комп'ютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Ігор МИХАЛЬЧУК, асистент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами;

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості StrikePlagiarism, складає 1,7% і адресується до 22 першоджерел; та системою Anti-Plagiarism складає 1%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС




Ігор МИХАЛЬЧУК

Андрій НІЧЕПОРУК

Ольга ПАВЛОВА