

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень

Інформаційна система газового детектора на базі Ардуіно
Назва теми

КвРІСТ 2101035.21.01.07 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 126 «Інформаційні системи та технології»

Шифр, назва

Освітня програма «Інформаційні системи та технології»

Назва

Виконав: студент IV курсу, група ІСТ-21-1

Підпис

Софія КУЛИК

Ініціали, прізвище

Керівник

Підпис, дата

Олег ВОЙЧУР

Ініціали, прізвище

Нормоконтролер

Підпис, дата

Тетяна КИСІЛЬ

Ініціали, прізвище

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

Підпис

Ольга ПАВЛОВА

Ініціали, прізвище

« » червня 2025 р.

Хмельницький 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 126 ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ

Освітня програма «ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

“ 10 ” 01 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Софії КУЛИК

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Інформаційна система газового детектора на базі Ардуіно

Керівник проекту (роботи) Олег ВОЙЧУР, асистент

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

Затверджена наказом ректора університету від 07.02.2025 р. № 5

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)

Інформаційна система моніторингу шкідливих газів на базі ESP32 та постановка задачі щодо її вдосконалення

Проектування та реалізація апаратної частини інформаційної системи моніторингу шкідливих газів

Програмна реалізація інформаційної системи та відправлення даних у хмару

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

Апаратне забезпечення проекту

Архітектура ПЗ для інформаційної системи

Схема електрична принципова

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 напрямку 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 проєктування інформаційної системи газового детектора на базі Ардуіно	01.04.2025	виконано
5	Робота над розділом 3 – проєктування системи газового детектора на базі Ардуіно	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

Керівник роботи

ПідписСофія КУЛІК
Ініціали, прізвищеОлег ВОЙЧУР
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		КВРІСТ 2101035.21.01.07 ПЗ	Пояснювальна записка	60		
			<u>Графічні матеріали</u>			
2		КВРІСТ 2101035.21.01.07 Е8	Апаратне забезпечення проекту	1		
3		КВРІСТ 2101035.21.01.07 Е8	Архітектура ПЗ для інформаційної системи	1		
4		КВРІСТ 2101035.21.01.07 Е8	Схема електрична принципова	1		

					КВРІСТ 2101035.21.01.07 ПЗ ВП			
Зм	Ар к	№ докум	Підпи с	Дат а	Відомість проекту	Літера	Арку щ	Аркуш ів
Розроби в		Кулик		02.06.25		У	1	1
Перевір.		Войчур		02.06.25		ХНУ, ІСТ-21-1		
Н. контр.		Кисіль		02.06.25				
Затв.		Павлова		02.06.25				

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Інформаційна система газового детектора на базі Ардуіно».

Автор роботи: Софія КУЛИК.

Керівник роботи: Войчур Олег Юрійович.

Пояснювальна записка: 55 с., 31 рис., 3 табл., 4 дод., 37 джерел.

Графічна частина: 3 креслення.


МІКРОКОНРОЛЕР, ІоТ (ІНТЕРНЕТ-РЕЧЕЙ), ПЛАТА, ДАТЧИК ГАЗУ, ХМАРА, GOOGLE SHEETS.

Метою дипломного проєкту є розробка системи виявлення газу з використанням звукового та візуального оповіщення. Розроблений метод дозволяє проводити аналіз значень, отриманих з газового сенсора, та автоматично класифікувати рівень небезпеки. У рамках проєкту виконано інтеграцію з Google Sheets - зручним хмарним сервісом, що дозволяє автоматично зберігати всі отримані показники датчика разом із часовими мітками.

Об'єктом дослідження є функціонування газового детектора.

Предметом дослідження є оцінка режимів виявлення газу в приміщеннях.

Для розробки системи було використано мову програмування C++, а також бібліотеки: Wire.h, LiquidCrystal_I2C.h, WiFi.h, HTTPClient.h. Для реалізації схеми ми використали : мікроконтролер ESP32, газовий датчик MQ-2, п'єзодинамік, потенціометр, РК дисплей I2C, емулятор Wokwi, резистори та проводи..


Підпис студента

30.05.2025

Дата

ЗМІСТ

ВСТУП	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	6
1.1 Аналіз предметної області і виявлення наявних проблем і завдань	6
1.2 Постановка задачі	13
1.3 Висновки до першого розділу	15
2 ОГЛЯД АПАРАТНОЇ СКЛАДОВОЇ	17
2.1 Апаратна платформа ESP32	17
2.2 Апаратна платформа Arduino	18
2.3 Принципи роботи датчиків газу та диму	19
2.3.1 Газовий датчик MQ-2	19
2.3.2. Внутрішня структура датчика газу MQ-2	21
2.3.3 Як працює детектор газу	24
2.3.4 Огляд апаратного забезпечення – модуль датчика газу MQ-2	24
2.3.5 Калібровка модуля датчика газу MQ-2	26
2.3.6 Розпинівка модуля датчика газу	27
2.4 П'єзодинамік	28
2.4.1 Опис та схема роботи зумера	29
2.4.2 Відмінності активного та пасивного зумеру	31
2.5 Потенціометр	32
2.5.1 Принципи роботи потенціометра	32
2.6 РК дисплей	34
2.6.1 РК дисплей Arduino LCD 1602	35
2.6.2 Опис пінів РК дисплею I2C	35
2.7 Макетна плата	36
2.8 Wokwi для Arduino та ESP32	37
2.9 Висновки до другого розділу	38
3 ПІДКЛЮЧЕННЯ ОКРЕМИХ МОДУЛІВ	39
3.1 Підключення модуля датчика газу	39

КВРІСТ 2101035.21.01.07 ПЗ								
Зм.	Арк.	№ док.ум.	Підпис	Дата	Інформаційна система газового детектора на базі Ардуіно	Літера	Аркуш	Аркушів
Виконав		Софія КУЛИК		02.08.25		у	2	72
Перевір.		Олег ВОЙЧУР		02.08.25				
Н.контр.		Тетяна КИСІЛЬ		02.08.25				
Затвер.		Ольга ПАВЛОВА		02.08.25				
						ХНУ ІСТ-21-1		

3.3 Підключення РК дисплею до ESP32	41
3.4 Підключення потенціометра до ESP32	44
3.5 Архітектура ІоТ	45
3.6 Надсилання даних у хмарне середовище.....	49
3.7 Висновки до третього розділу.....	53
ВИСНОВКИ	54
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	56
ДОДАТОК А	61
ДОДАТОК Б	61
ДОДАТОК В	63
ДОДАТОК Г	64

ВСТУП

Дипломна робота спрямована на розробку системи виявлення газу з використанням звукового та візуального оповіщення. Одним із ключових аспектів проекту є організація взаємодії ESP32. Основна мета системи – забезпечення безпеки у приватних будинках шляхом оперативного реагування на перевищення допустимого рівня газу та сповіщати про це користувача. Всю інформацію він зможе перевірити навіть на далекій відстані, завдяки підключенню до хмари

Для моделювання та тестування розроблюваної системи використовується веб-сервіс Wokwi, що є інтерактивною платформою для створення електронних схем та роботи з ESP32. Це середовище дозволяє візуалізувати роботу пристрою та перевіряти його працездатність без необхідності фізичної збірки.

Мікроконтролер ESP32 є ефективним засобом для створення бездротових систем управління, збору та обробки даних у реальному часі. Завдяки наявності вбудованих модулів Wi-Fi та Bluetooth, він дозволяє легко інтегрувати пристрої в мережу, забезпечуючи зручну взаємодію між пристроями та користувачем. Потужний двоядерний процесор, велика кількість вхідних/вихідних портів та підтримка різноманітних інтерфейсів (I2C, SPI, UART тощо) роблять його універсальним рішенням для багатьох застосувань - від розумного дому до промислових систем автоматизації.

Програмування пристрою здійснюється мовою Arduino C++ з використанням фреймворку Wiring. Завдяки простоті синтаксису та зручності у використанні, вона є доступною навіть для початківців, що дозволяє швидко розробляти та тестувати електронні системи.

У процесі виконання роботи буде реалізовано механізм виявлення газу, таких як: нафтовий газ (LPG), дим, алкоголь, пропан, водень, метан і чадний газ, у концентраціях від 200 до 10 000 ppm (мільйонних часток). Він дозволить контролювати його концентрацію в повітрі та вчасно сповіщати користувачів про небезпеку.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 4
Зм.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної області і виявлення наявних проблем і завдань

Сучасний світ стрімко змінюється завдяки розвитку технологій, які проникають у всі сфери життя. Особливу увагу сьогодні приділяють Інтернету речей (IoT), який дозволяє об'єднувати фізичні пристрої в єдину мережу для збору, обробки та обміну даними. Ця технологія відкриває нові можливості для автоматизації, створення розумних систем і підвищення комфорту, чи то в побуті, промисловості, чи навіть у сільському господарстві. IoT стає основою для інновацій, роблячи наше оточення більш інтерактивним і ефективним. Ключовою особливістю IoT є здатність до автоматизації, та віддаленого керування [28].

Технологічною основою IoT є мікроконтролери, такі як ESP32 чи Arduino, які керують датчиками, актуаторами, забезпечують обчислення та зв'язок, а також хмарні сервіси, як AWS IoT чи Google Cloud, що обробляють величезні обсяги даних. Макетні плати, часто використовуються для створення прототипів IoT-пристроїв, дозволяючи швидко зібрати схему з сенсорами, дисплеями та комунікаційними модулями для моделювання, налагодження та подальшої реалізації [14].

У даному проєкті для реалізації функціоналу було обрано плату ESP32, яка значно перевершує Arduino Uno за багатьма характеристиками. Основною перевагою ESP32 є наявність вбудованих модулів Wi-Fi та Bluetooth, що робить її ідеальним вибором для розробки IoT-проєктів, зокрема для передавання даних до хмарових сервісів, таких як Google Sheets [1].

Порівняно з Arduino Uno, ESP32 має більше вхідних/вихідних пінів, вищу тактову частоту, більший об'єм пам'яті, а також підтримує багатозадачність. Це дає змогу одночасно зчитувати дані з датчиків, виводити інформацію на дисплей та передавати її в Інтернет. У проєкті використання ESP32 забезпечує ефективну обробку даних та спрощує зв'язок із хмаровими платформами без потреби у додаткових модулях.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 5
Зм.	Арк.	№ докум.	Підпис	Дата		

Окрім цього, ESP32 підтримує широкий спектр бібліотек та має активну спільноту розробників, що значно полегшує розробку програмного забезпечення. Її компактний розмір та низьке енергоспоживання роблять цю плату зручною для вбудованих систем і автономних пристроїв. Усі ці характеристики роблять ESP32 універсальним і надійним вибором для реалізації систем моніторингу як у домашньому, так і в промисловому середовищі.

Таким чином, вибір ESP32 як головної плати є доцільним і технологічно обґрунтованим рішенням, яке дозволяє реалізувати функціонал сучасного розумного пристрою без значних апаратних ускладнень.

У сучасних умовах техногенного навантаження та широкого використання газових приладів у побуті, зростає потреба в оперативному виявленні витоків газу та забезпеченні своєчасного інформування користувача. Газові витoki можуть стати причиною серйозних аварій, пожеж чи отруєнь, тому розробка простих та ефективних систем виявлення загроз є актуальною задачею у сфері безпеки житла.

Для виявлення подібних ситуацій використовуються системи моніторингу якості повітря, що здатні автоматично виявляти перевищення допустимих концентрацій шкідливих речовин та оперативно інформувати про це користувачів. Проте традиційні системи зазвичай є дорогими, складними у встановленні, обслуговуванні, не завжди мають функції бездротового доступу або гнучкості щодо зміни конфігурації.

З появою доступних мікроконтролерів, таких як ESP32, та універсальних газових датчиків, зокрема MQ-серії, стало можливим розробляти ефективні, компактні та бюджетні рішення для локального або віддаленого моніторингу повітря. Ці рішення можна застосовувати як у побуті, так і в навчальних, виробничих або сільськогосподарських умовах.

У межах даного проекту вирішується комплекс завдань, спрямованих на створення сучасної, доступної та функціональної системи моніторингу якості повітря з можливістю хмарного зберігання даних та віддаленого контролю. Одним із ключових етапів є розробка апаратної частини системи, яка базується на

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 6
Зм.	Арк.	№ докум.	Підпис	Дата		

мікроконтролері ESP32 та газовому датчику MQ-2[11]. Датчик MQ-2, у свою чергу, забезпечує виявлення таких небезпечних компонентів, як метан, бутан, пропан, дим та інші горючі гази, що робить його оптимальним для моніторингу повітря у побутових або напівпромислових умовах [20].

Газовий датчик – це чутливий елемент або вимірювальний перетворювач, призначений для визначення якісного та кількісного складу газових сумішей. Він є ключовим компонентом газоаналізаторів і сигналізаторів, що забезпечує виявлення та контроль концентрації газів у повітрі [6].

Для виявлення небезпечних газів у повітрі широко застосовуються різноманітні типи датчиків газу, які відрізняються принципом дії, точністю вимірювання та спектром виявлюваних речовин. Найпоширенішими типами газових датчиків є напівпровідникові, електрохімічні та оптичні (зокрема інфрачервоні), кожен з яких має свої переваги в залежності від умов використання та типу виявлюваного газу.

Найпоширенішими у практиці побудови систем моніторингу є напівпровідникові сенсори серії MQ. Ці датчики мають високу чутливість до певних груп газів, таких як чадний газ (CO), метан (CH₄), пропан, водень, алкогольні пари, дим тощо. Наприклад, датчик MQ-2 здатен виявляти дим, LPG та інші легкозаймісті гази, що робить його доцільним вибором для побутових систем безпеки.

Окрім MQ-серії, використовуються й інші спеціалізовані сенсори. Так, для точного вимірювання вмісту вуглекислого газу застосовують датчики MH-Z19 або SCD30, які забезпечують цифрову передачу даних і мають високу стабільність. Для більш складних застосувань можуть використовуватись багатоконпонентні модулі, наприклад, MiCS-6814 або CCS811. Вони здатні одночасно виявляти кілька видів шкідливих речовин, включаючи леткі органічні сполуки.

У проектуванні IoT-систем вибір конкретного сенсора залежить від вимог до чутливості, точності, швидкості реакції та доступного інтерфейсу підключення.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 7
Зм.	Арк.	№ докум.	Підпис	Дата		

Важливим фактором також є можливість інтеграції з мікроконтролерами для подальшої обробки та передавання даних у хмарні сервіси [8].

Наступним важливим завданням є розробка програмного забезпечення, яке дозволяє проводити аналіз значень, отриманих з газового сенсора, та автоматично класифікувати рівень небезпеки. Програмний алгоритм реалізує порогову обробку аналогових сигналів із сенсора та відображає статус (SAFE, ALERT, DANGER) залежно від концентрації газу. Такий підхід дозволяє не лише вчасно реагувати на зміну якості повітря, але й забезпечує чітку візуалізацію ситуації для користувача.

Окрему увагу приділено підключенню системи до мережі Інтернет за допомогою Wi-Fi, що дає змогу реалізувати хмарні функції, зокрема автоматичне збереження даних, перегляд історії та аналіз. У рамках проєкту виконано інтеграцію з Google Sheets – зручним хмарним сервісом, що дозволяє автоматично зберігати всі отримані показники датчика разом із часовими мітками. Це надає можливість вести журнал історії змін, аналізувати динаміку концентрації газу та за потреби здійснювати віддалений моніторинг даних.

Протоколи зв'язку в Інтернеті речей (IoT) відіграють вирішальну роль, забезпечуючи надійну та ефективну передачу даних між пристроями, серверами та хмарними платформами. Кожен протокол має свої особливості, які роблять його придатним для певних сценаріїв використання, залежно від вимог до енергоефективності, дальності передачі чи обсягу даних [30].

MQTT (Message Queuing Telemetry Transport) – це легкий протокол, який працює за принципом "видавець-підписник". Він ідеально підходить для IoT-пристроїв із обмеженими ресурсами, таких як датчики в розумних будинках. Наприклад, датчик температури може "публікувати" дані на сервер, а додаток на смартфоні "підписується" на ці повідомлення, отримуючи оновлення в реальному часі. MQTT мінімізує витрати енергії та забезпечує швидку передачу невеликих обсягів даних [26].

CoAP (Constrained Application Protocol) розроблений для пристроїв із низьким енергоспоживанням і працює за моделлю "запит-відповідь", подібно до

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 8
Зм.	Арк.	№ докум.	Підпис	Дата		

HTTP, але є значно легшим. Його часто використовують у системах, де важлива економія ресурсів, наприклад, для керування розумним освітленням. CoAP підтримує передачу даних через UDP, що зменшує накладні витрати порівняно з TCP.

HTTP/HTTPS – це добре відомий протокол, який використовується в IoT для передачі більших обсягів даних, наприклад, у веб-додатках чи системах із складними інтерфейсами. Однак через високе енергоспоживання та більшу складність він менш популярний для простих IoT-пристроїв, таких як датчики, але часто застосовується в розумних камерах або пристроях із доступом до хмарних сервісів.

LoRaWAN – це протокол для мереж із великою дальністю дії та низьким енергоспоживанням, який ідеально підходить для IoT у сільському господарстві чи міських системах. Наприклад, датчики вологості ґрунту на полях можуть передавати дані на відстань кількох кілометрів, використовуючи LoRaWAN, споживаючи мінімум енергії.

Zigbee і Z-Wave - це протоколи для створення локальних мереж із низьким енергоспоживанням, які часто застосовуються в розумних будинках. Вони дозволяють пристроям, як-от розумні розетки чи термостати, створювати сітчасту мережу, де кожен пристрій може передавати сигнали іншим, збільшуючи покриття. Zigbee популярний завдяки своїй універсальності, тоді як Z-Wave має перевагу в простоті налаштування.

I2C (Inter-Integrated Circuit) – це протокол для обміну даними між мікроконтролером ESP32 і LCD-дисплеєм через бібліотеку LiquidCrystal_I2C. Він дозволяє передавати команди для виведення інформації, наприклад, значень газу та статусу ("SAFE", "ALERT", "DANGER"), на екран за допомогою лише двох пінів (SDA і SCL). I2C є енергоефективним і широко застосовується в IoT для підключення периферійних пристроїв, таких як дисплеї чи датчики, завдяки простоті та можливості підключення кількох пристроїв до однієї шини.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 9
Зм.	Арк.	№ докум.	Підпис	Дата		

Wi-Fi – це основний протокол для підключення ESP32 до інтернету, реалізований через бібліотеку WiFi.h. У коді ESP32 підключається до мережі (за допомогою SSID і пароля, отримуючи IP-адресу для подальшої комунікації. Wi-Fi забезпечує бездротовий зв'язок із високою пропускнуою здатністю, що дозволяє вашому пристрою відправляти дані про концентрацію газу та статус до віддаленого сервера.

HTTP (HyperText Transfer Protocol) використовується для передачі даних із вашого пристрою до Google Sheets через бібліотеку HTTPClient.h. У коді формується URL-запит до Google Apps Script із параметрами value (значення газу) і status (статус системи), який відправляється методом GET. HTTP є простим і універсальним протоколом, який добре підходить для відправки даних до веб-сервісів, але він менш енергоефективний для IoT-пристроїв із обмеженим живленням порівняно з такими протоколами, як MQTT чи CoAP. У цьому випадку HTTP забезпечує інтеграцію з Google Sheets, дозволяючи зберігати та аналізувати дані в реальному часі.

Вибір протоколу залежить від конкретного застосування. Для невеликих датчиків із батарейним живленням краще підходять MQTT або CoAP, тоді як для промислових систем із великою кількістю даних можуть використовуватися HTTP/HTTPS. LoRaWAN і Zigbee оптимальні для сценаріїв, де важливі дальність і енергоефективність. Ці протоколи разом створюють гнучку основу для розвитку IoT, забезпечуючи зв'язок у найрізноманітніших умовах [30].

Ще одним важливим завданням стало створення зрозумілого інтерфейсу для локального оповіщення користувача. Для цього у системі використано LCD-дисплей, який виводить актуальний рівень газу та його статус у текстовому вигляді, а також набір кольорових світлодіодів (зелений, жовтий, червоний) для інтуїтивного розпізнавання стану навколишнього середовища. Додатково вбудовано звуковий зумер, який активується при перевищенні критичних значень, забезпечуючи оперативне сповіщення навіть без зорового контакту з пристроєм.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 10
Зм.	Арк.	№ докум.	Підпис	Дата		

Для створення та тестування Arduino-проектів існує кілька популярних емуляторів, серед яких найбільш відомими є Wokwi, Tinkercad Circuits і Proteus. Кожен із них має свої особливості, переваги та недоліки, які важливо враховувати при виборі інструмента для роботи.

Wokwi – зручний онлайн-інструмент, який дозволяє створювати, симулювати та тестувати електронні схеми. Він дозволяє працювати з широким спектром плат (Arduino Uno, Mega, Nano, ESP32, STM32 тощо) та компонентів. Його ключова перевага – підтримка реального Arduino-коду з можливістю використання популярних бібліотек. Wokwi має зручний і зрозумілий інтерфейс, що спрощує процес проектування та програмування мікроконтролерів, а також дозволяє легко моделювати дисплеї, сенсори, модулі зв'язку тощо. Також він підтримує симуляцію реального часу з точністю до мілісекунд, що важливо для перевірки часозалежних процесів, таких як робота з датчиками або таймерами. Це чудовий вибір для тих, хто вже знайомий з основами Arduino та хоче швидко створювати і тестувати проекти. Недоліком є відсутність 3D-моделі макетної плати, а також те, що він менш наочний для абсолютних новачків.

. Вибір Wokwi як платформи для моделювання обумовлений його доступністю, гнучкістю та можливістю створення повноцінних електронних проектів без необхідності використання фізичного обладнання. Це рішення є особливо корисним для тих, хто хоче швидко перевірити роботу коду та підключень. Завдяки Wokwi користувачі можуть зосередитися на логіці свого проекту, заощаджуючи час і ресурси.

Tinkercad Circuits, розроблений Autodesk, більше орієнтований на новачків. Він має простий інтерфейс із візуальним макетуванням на breadboard, що дає змогу зрозуміти принципи підключення компонентів без глибоких знань електроніки. Tinkercad також дозволяє писати базовий Arduino-код, але підтримка складніших бібліотек або плат обмежена. Крім того, відсутня підтримка ESP32 та багатьох сучасних модулів, що обмежує його застосування в складніших IoT-проектах.

Такий підхід ідеально підходить для перших кроків у світі Arduino, проте може бути обмежувачим для серйозніших проєктів.

Proteus – це професійна програма, яка використовується не лише для симуляції, а й для проєктування друкованих плат. Вона підтримує не тільки Arduino, а й інші мікроконтролери, включаючи PIC, AVR тощо. Proteus дозволяє створювати складні логічні схеми, симулювати аналогові та цифрові процеси, і навіть перевіряти роботу зібраних електронних систем. Проте, вона є платною, має складніший інтерфейс і вимагає певного досвіду в роботі з електронікою.

У підсумку, якщо вам потрібен зручний і функціональний інструмент для роботи з Arduino в онлайні без встановлення програм - Wokwi стане найкращим вибором, особливо для середнього рівня та просунутих користувачів, які хочуть створювати сучасні проєкти з використанням інтернету речей та складних модулів.

Проєкт передбачає можливість подальшого розвитку та масштабування, зокрема через підключення мобільних додатків, таких як Blynk або Telegram-боти. Завдяки цьому користувач зможе отримувати сповіщення на свій смартфон у режимі реального часу, переглядати графіки показників, керувати пристроєм дистанційно або отримувати push-повідомлення у разі небезпеки. Такий підхід відповідає сучасним трендам у галузі IoT (Інтернету речей) і забезпечує високий рівень адаптивності, зручності та безпеки для користувача.

1.2 Постановка задачі

У сучасних умовах безпека житлових і промислових приміщень набуває все більшої актуальності, що зумовлює необхідність використання новітніх технологій для моніторингу навколишнього середовища. Одним із важливих рішень у цій сфері є система виявлення газу, яка здатна вчасно ідентифікувати перевищення допустимого рівня газів та оперативно сповіщати користувача для запобігання можливим загрозам.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 12
Зм.	Арк.	№ докум.	Підпис	Дата		

Метою цього проєкту є розробка газового детектора, що здійснює моніторинг складу повітря та попереджає про небезпеку за допомогою світлової

Основні завдання, що необхідно виконати для досягнення поставленої мети:

1. Розробити схему газового детектора, що містить датчик газу, світлодіоди (червоний, жовтий, зелений) та звуковий сигналізатор (пищалку).
2. Підключити всі компоненти до плати ESP32 в середовищі Wokwi та забезпечити їх коректну взаємодію.
3. Створити програмний код для пристрою, який буде аналізувати показники датчика та вмикати відповідні сигнали при виявленні газу.
4. Інтегрувати модуль ESP8266 для передавання отриманих даних у хмарне середовище.
5. Виконати тестування розробленої системи, оцінити її ефективність та перевірити коректність роботи всіх функціональних компонентів.

Розроблена система повинна не лише виконувати локальний моніторинг, а й мати можливість передавати показники в реальному часі у віддалене середовище, що забезпечить зберігання даних та їх подальший аналіз. Це особливо важливо для застосування у промислових об'єктах, де централізований моніторинг дозволяє виявляти небезпечні ситуації на ранніх етапах. Для цього у проєкті передбачено реалізацію функції передавання даних через Wi-Fi-з'єднання до Google Sheets за допомогою HTTP-запитів. Такий підхід забезпечує простий, надійний та економічний спосіб інтеграції з хмарною інфраструктурою, без потреби у створенні окремого сервера або використання сторонніх платформ. Таким чином, реалізація даної задачі сприяє не лише створенню прототипу побутової сигналізаційної системи, але й демонструє можливість побудови розумного пристрою в рамках концепції Інтернету речей (IoT). Проєкт поєднує апаратну частину, програмну логіку та хмарну аналітику, що робить його універсальним рішенням для підвищення безпеки у різних сферах застосування, від домашнього використання до промислових систем моніторингу.

1.3 Висновки до першого розділу

У результаті проведеного аналізу предметної області визначено актуальність проблеми моніторингу якості повітря в побутових умовах, зокрема виявлення витоків небезпечних газів. Визначено ключові проблеми, пов'язані з безпекою та необхідністю своєчасного оповіщення користувача.

На основі аналізу було обґрунтовано доцільність створення недорогого, функціонального пристрою для моніторингу повітря, що базується на мікроконтролері ESP32 та газовому сенсорі MQ-2. ESP32 обрано завдяки його вбудованим модулям Wi-Fi і Bluetooth, високій продуктивності, енергоефективності та багатофункціональності, що робить його ідеальним для реалізації IoT-рішень. Сенсор MQ-2 забезпечує виявлення широкого спектра горючих речовин, включаючи метан, пропан, бутан, дим та інші.

У рамках проєкту також було здійснено обґрунтований вибір програмного середовища – емулятора Wokwi, який надає зручну платформу для моделювання, тестування та налагодження електронних схем без потреби у фізичних компонентах. Завдяки підтримці реального Arduino-коду, широкому вибору електроніки та візуальному інтерфейсу, Wokwi значно спрощує процес розробки і дозволяє пришвидшити реалізацію проєкту.

Особливу увагу приділено інтеграції з хмарними сервісами (Google Sheets), що забезпечують автоматичне збереження вимірянних значень та можливість дистанційного моніторингу. Для локального оповіщення користувача застосовано LCD-дисплей, кольорові світлодіоди (зелений, жовтий, червоний) та звуковий зумер. Такий підхід дозволяє забезпечити оперативне та інтуїтивно зрозуміле інформування про поточний стан повітряного середовища.

Крім того, розроблена система має потенціал до удосконалення з можливістю підключення мобільних додатків або месенджер-ботів, що відкриває перспективи її використання в більш широких сценаріях і забезпечує гнучкість у подальшому розвитку.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 14
Зм.	Арк.	№ докум.	Підпис	Дата		

Таким чином, сформульовано вимоги до майбутньої інформаційної системи: вона має забезпечити автоматизований контроль стану повітряного середовища, оповіщення користувача про потенційну небезпеку, зберігання даних у хмарі та можливість подальшого розвитку, відповідаючи сучасним стандартам безпеки та зручності.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

2 ОГЛЯД АПАРАТНОЇ СКЛАДОВОЇ

2.1 Апаратна платформа ESP32

ESP32 – це мікроконтролерна плата, побудована на основі 32-бітного двоядерного процесора Tensilica LX6 з вбудованим Wi-Fi та Bluetooth. Завдяки своїй продуктивності та універсальності, вона є ідеальним вибором для IoT-проектів, бездротових систем, смарт-пристроїв та підключення до хмарних сервісів.

Плата ESP32 має велику кількість цифрових і аналогових входів/виходів, підтримує PWM, SPI, I2C, UART, а також має датчики сенсорного вводу, вбудований аналого-цифровий перетворювач(АЦП) та цифро-аналоговий перетворювач(ЦАП), таймери та апаратне шифрування [14].

Для запуску ESP32 не потрібно складних налаштувань: достатньо підключити її до комп'ютера через USB-кабель, або живити від акумулятора/адаптера через пін VIN або 3.3V.

Мікроконтролер ESP32 зазвичай використовує USB-UART міст на базі чіпа CP2102 або CH340, що дозволяє легко прошивати плату з комп'ютера через Arduino IDE або PlatformIO.

Таблиця 2.1 – Характеристики плати ESP32

Характеристика	Опис характеристики
Flash-пам'ять	4 МБ (залежно від моделі)
Тактова частота	До 240 МГц (двоядерний процесор)
Максимальний струм виводу	~12 мА (3.3V GPIO)
DAC (ЦАП)	2 канали
Кількість аналогових входів	До 18 (12-бітні ADC)
Цифрові входи / виходи	До 34

Кінець таблиці 2.1

Wi-Fi	IEEE 802.11 b/g/n, 2.4 ГГц
Bluetooth	v4.2 + BLE
Підтримка інтерфейсів	I2C, SPI, UART, CAN, IR, RMT, сенсорний ввід
Рекомендована напруга живлення	5 В (через USB) або 3.3 В (через VIN)
Мікроконтролер	Tensilica LX6 (32-біт, dual-core)

2.2 Апаратна платформа Arduino

Arduino Uno представляє собою пристрій, збудований на основі мікроконтролера ATmega328 (datasheet). В його комплекті міститься весь необхідний функціонал для зручної роботи з мікроконтролером: 14 цифрових входів/виходів (з них 6 можуть використовуватися як ШІМ-виходи), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування (ICSP) та кнопка скидання. Для початку використання пристрою досить просто надати живлення від AC/DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою USB-кабелю. Однак, в відміну від попередніх плат Arduino, Uno використовує мікроконтролер ATmega16U2 (до версії R2 - ATmega8U2) як перетворювач інтерфейсів USB-UART замість мікросхеми FTDI [12].

Основні характеристики плати Arduino Uno наведені в таблиці 2.2.

Таблиця 2.2 – Характеристики плати Arduino Uno

Характеристика	Опис характеристики
Flash-пам'ять	32 КБ, 0.5 КБ використовуються завантажувачем
Тактова частота	16 МГц

Кінець таблиці 2.2

Максимальний струм виводу (3.3V)	50 Ма
Максимальний струм виводу	40 Ма
Кількість аналогових входів	6
Цифрові входи / виходи	14, 6 можуть використовуватися як ШІМ-виходи
Характеристика	Опис характеристики
Рекомендована напруга живлення	7-12В
Мікроконтролер	АТmega328

2.3 Принципи роботи датчиків газу та диму

2.3.1 Газовий датчик MQ-2

Основним завданням датчика MQ-2 є перетворення концентрації певної речовини у відповідний електричний сигнал. Цей сигнал далі може бути оброблений системами індикації або управління. Газові сенсори зазвичай входять до складу більш складних систем вимірювання, які включають також модулі підсилення, фільтрації та відображення інформації [6].

Датчик MQ-2 є одним із найпоширеніших сенсорів у лінійці газових датчиків серії MQ. Він належить до типу метал-оксид-напівпровідникових (МОП, MOS) пристроїв, відомих також як хімічні резистори. Принцип його роботи ґрунтується на зміні електричного опору чутливого матеріалу під впливом газів. Вимірювання концентрації здійснюється за допомогою простого ланцюга дільника напруги, що дозволяє визначати рівень газу в повітрі [8].

На рисунку 3.1 зображений датчик газу MQ-2



Рисунок 2.1 – датчика газу MQ-2[33]

Сенсор функціонує при постійній напрузі 5 В та споживає приблизно 800 мВт потужності. Він здатний виявляти широкий спектр газів, зокрема зріджений нафтовий газ (LPG), дим, алкоголь, пропан, водень, метан і чадний газ, у концентраціях від 200 до 10 000 ppm (мільйонних часток) [8].

Повний список технічних характеристик MQ-2 в таблиці 2.3

Таблиця 2.3 – Технічні характеристики MQ-2

Характеристика	Опис характеристик
Робоча напруга	5 В
Опір навантаження	20 кОм
Опір нагрівача	33 Ом ± 5%

Кінець таблиці 2.3

Споживана потужність	<800 мВт
Опір чутливості	10 кОм - 60 кОм
Вимірювання концентрації	200 - 10000 ppm
Час розігріву	більше 24 годин

2.3.2. Внутрішня структура датчика газу MQ-2

Газовий датчик MQ-2 має конструкцію, що включає два захисні шари з тонкої нержавіючої сітки, які виконують функцію так званої "антивибухової сітки" (anti-explosion network). Основне призначення цих шарів – запобігання можливому займанню у разі контакту нагрівального елемента з легкозаймистими газами [6].

На рисунку 2.2 зображений зовнішні компоненти газового датчика



Рисунок 2.2 – Зовнішні компоненти газового датчика MQ-2[33]

Окрім вибухозахисної функції, сітка виконує ще одну важливу роль - вона захищає сенсор від механічних пошкоджень та фільтрує тверді частинки, пропускаючи лише газоподібні речовини всередину камери. Фіксація сітки здійснюється за допомогою мідного затискного кільця, яке забезпечує її надійне з'єднання з корпусом.

На рисунку 2.3 зображена внутрішня структура датчика

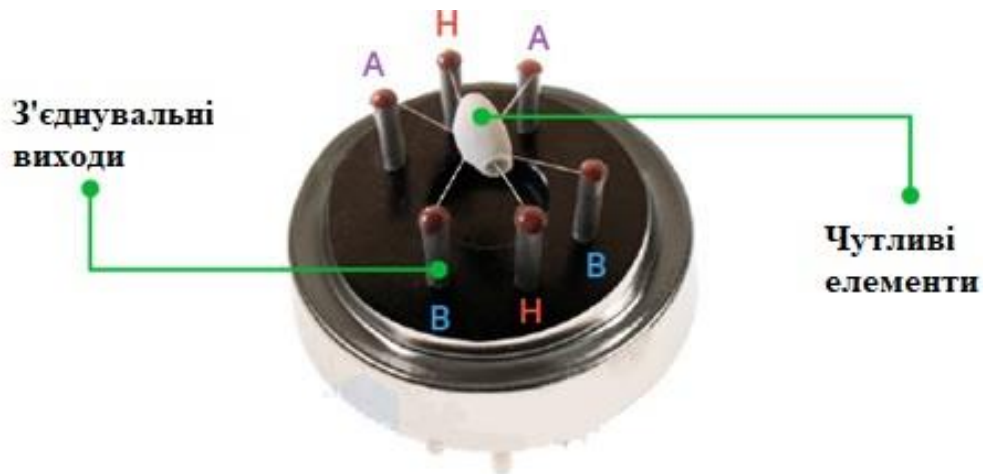
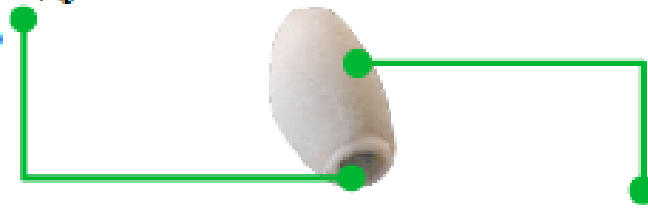


Рисунок 2.3 – Внутрішня структура MQ-2

Після зняття зовнішньої сітки можна побачити характерну зіркоподібну структуру чутливого елемента, яка розміщена на бакелітовій основі(рис.2.4). Така форма забезпечує збільшену площу контакту з навколишнім повітрям, що сприяє швидкому реагуванню на зміну концентрації газів. Датчик має шість з'єднувальних ніжок, дві з яких (позначені як Н) забезпечують нагрівання сенсорного матеріалу за допомогою вбудованого нагрівального елемента. Постійне підігрівання необхідне для підтримки робочої температури, за якої відбуваються хімічні реакції між газами та сенсорним матеріалом. Інші чотири ніжки (маркуються як А та В) відповідають за передавання вихідних сигналів. Вони підключені до корпусу чутливого елемента через платинові дроти, які фіксують зміни електричного струму залежно від концентрації газів у навколишньому середовищі.

**Кераміка на
основі оксиду
алюмінію**



**Оболонка з
діоксиду**

Рисунок 2.4 – Чутливий елемент[34]

Чутливий елемент датчика має трубчасту форму та виготовлений на основі кераміки з оксиду алюмінію (Al_2O_3). Він покритий діоксидом олова (SnO_2), який є основним чутливим матеріалом, здатним реагувати на горючі гази. Керамічна основа, у свою чергу, сприяє ефективному нагріванню та підтримує стабільну робочу температуру датчика [6].

На рисунку 2.5 зображена внутрішня структура чутливого елемента

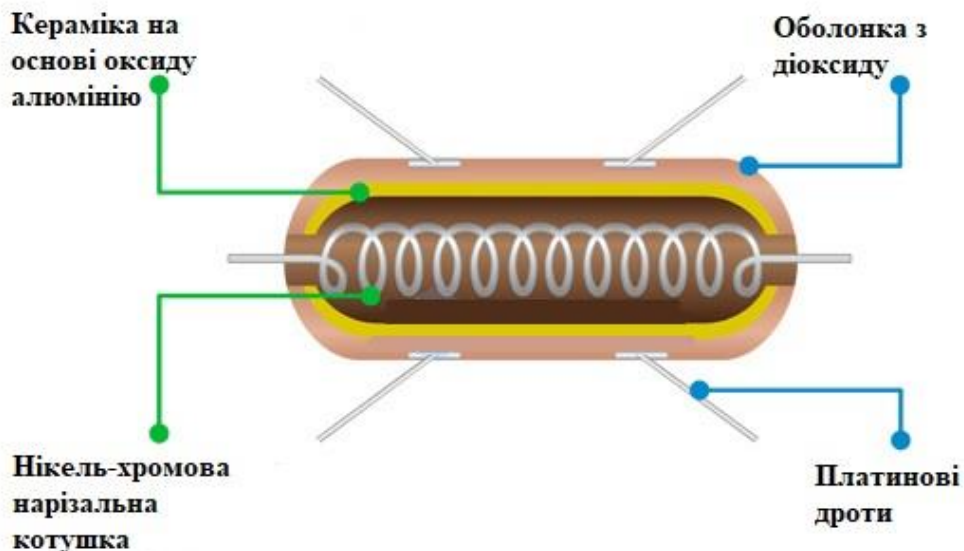


Рисунок 2.5 – Внутрішня структура чутливого елемента [34]

Система нагрівання складається з нікель-хромової котушки, розташованої всередині керамічного корпусу. Це дозволяє рівномірно розподіляти тепло та

підтримувати необхідний температурний режим для стабільної роботи сенсора. Платинові дроти разом із чутливим покриттям із діоксиду олова формують сенсорний елемент, який визначає концентрацію газів та передає відповідні сигнали. Завдяки такій конструкції датчик забезпечує високу точність і надійність у реальних умовах експлуатації.

2.3.3 Як працює детектор газу

При нагріванні напівпровідникового матеріалу, такого як діоксид олова (SnO_2), на відкритому повітрі відбувається процес адсорбції кисню на його поверхні. У чистому повітрі донорні електрони SnO_2 зв'язуються з адсорбованими молекулами кисню, що створює бар'єр для проходження електричного струму через сенсор.

Однак у присутності відновлювальних газів (наприклад, чадного газу або метану) молекули кисню вступають із ними в хімічну реакцію. У результаті цього процесу кисень звільняє захоплені електрони, що призводить до зменшення електричного опору сенсорного шару та збільшення струму в електричному колі. Величина змінного струму залежить від концентрації газу, що дозволяє оцінити його рівень у навколишньому середовищі [8].

2.3.4 Огляд апаратного забезпечення – модуль датчика газу MQ-2

Оскільки датчик MQ-2 не підтримує прямого підключення до макетних плат, для тестування доцільно використовувати його модульну версію. Цей модуль є компактним і зручним у використанні, оскільки оснащений двома типами виходів. Він може не лише сигналізувати про наявність горючих газів за допомогою цифрового виходу, а й передавати аналоговий сигнал, пропорційний їх концентрації у повітрі, що особливо важливо для точніших вимірювань..

На рисунку 2.6 зображений модуль датчика газу.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 23
Зм.	Арк.	№ докум.	Підпис	Дата		

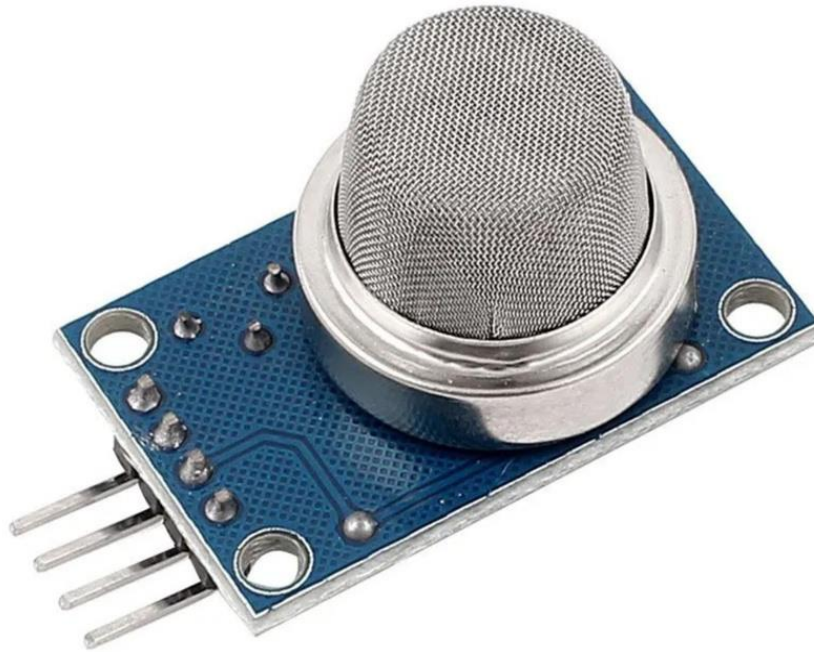


Рисунок 2.6 – Модуль датчика газу [35]

Напруга на аналоговому виході змінюється відповідно до концентрації газу чи диму: зі збільшенням вмісту газів у повітрі вихідна напруга зростає, а при зниженні – падає. Ця залежність добре ілюструється на рисунку 2.7, де представлено типову характеристику чутливості сенсора для різних типів газів, таких як метан, бутан, водень, пропан, дим тощо..

Аналоговий сигнал, отриманий від сенсора, надходить на компаратор LM393, який розташований у нижній частині модуля і виконує його цифрове перетворення. Також на модулі передбачений потенціометр для регулювання чутливості сенсора. Це дає змогу налаштувати порогове значення, при якому датчик буде реагувати на присутність газу в навколишньому середовищі [32].

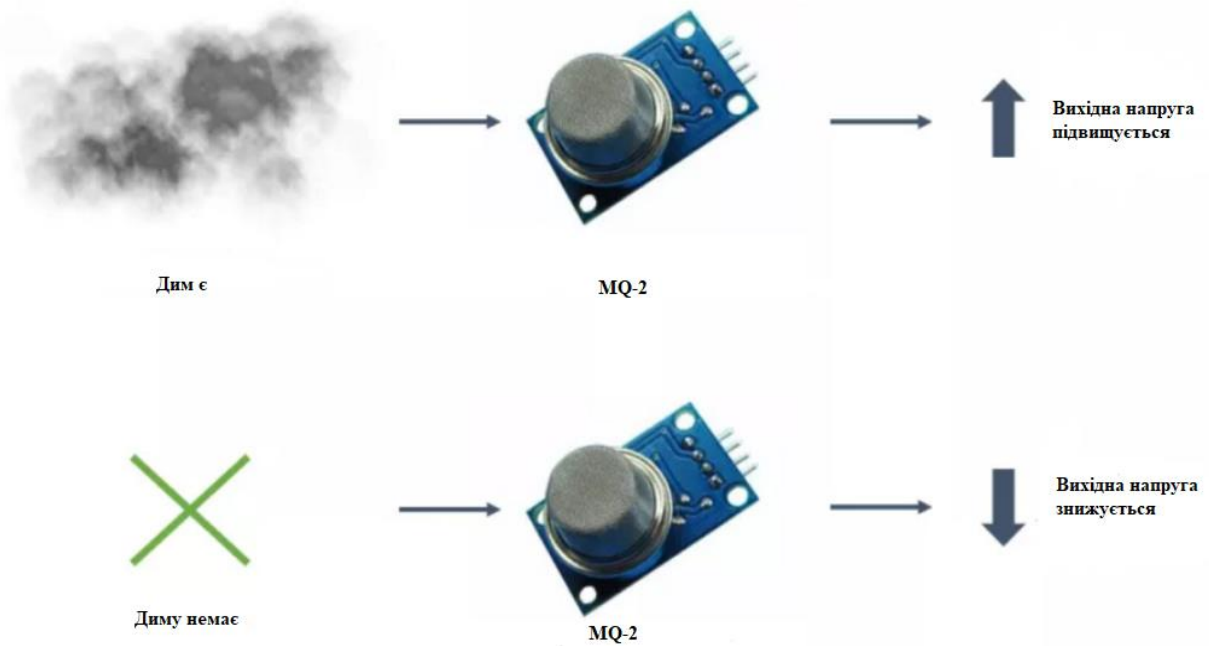


Рисунок 2.7 – Дія датчика газу[34]

2.3.5 Калібровка модуля датчика газу MQ-2

Для налаштування чутливості датчика MQ-2 необхідно розташувати його поруч із джерелом диму або газу, який потрібно виявити, і поступово обертати потенціометр до моменту загоряння червоного світлодіода на модулі. Це свідчитиме про те, що рівень концентрації газу перевищив заданий поріг, і сенсор переходить у стан тривоги. Обертання потенціометра за годинниковою стрілкою підвищує чутливість сенсора, тоді як обертання проти годинникової стрілки її знижує [24].

Такий механізм ручного налаштування дозволяє адаптувати модуль до різних умов експлуатації, від побутових приміщень із можливим витоком газу до лабораторних або виробничих середовищ з високою запиленістю чи змінною вентиляцією.

На рисунку 2.8 показано потенціометр, призначений для регулювання чутливості модуля MQ-2.

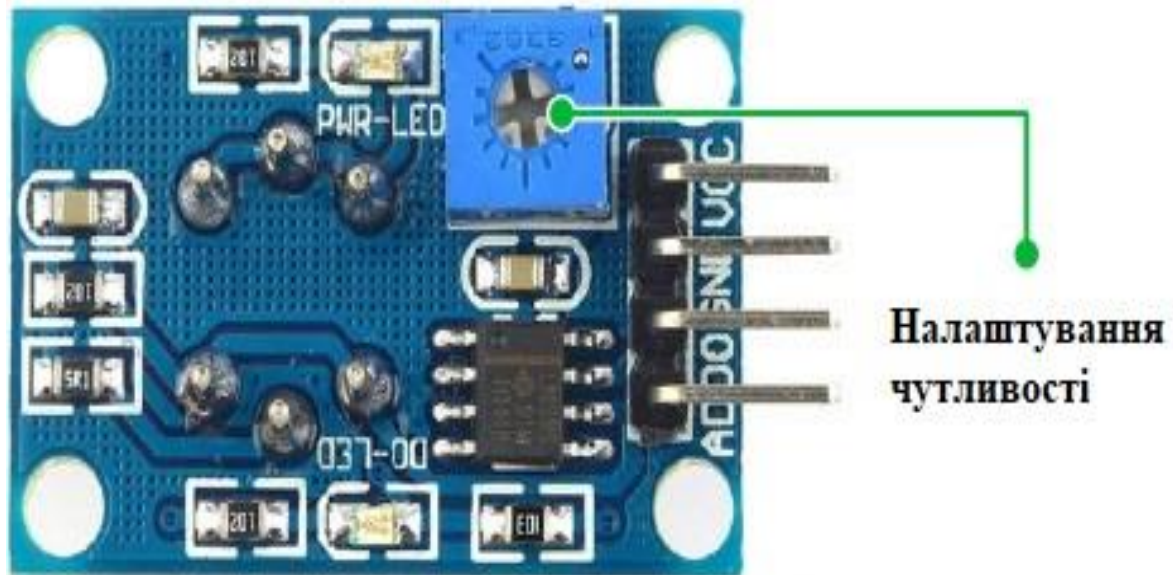


Рисунок 2.8 – Потенціометр регулювання чутливості модуля датчика газу MQ-2[35]

Вбудований компаратор постійно аналізує аналоговий вихідний сигнал (A0) і порівнює його з граничним значенням, встановленим потенціометром. Якщо цей поріг перевищено, цифровий вихід (D0) переходить у високий логічний стан, активуючи світлодіодний індикатор. Це налаштування є корисним для автоматизації певних процесів, таких як увімкнення реле, запуск вентиляційної системи або активація протипожежного обладнання при перевищенні допустимого рівня концентрації газу чи диму. Таке рішення значно спрощує автоматизацію побутових і промислових процесів, зокрема [24]:

2.3.6 Розпінівка модуля датчика газу

Для підключення модуля MQ-2 до мікроконтролера необхідно враховувати розташування його виводів(рисю2.9).



Рисунок 2.9 – Розпинівка модуля датчика газу [35]

- VCC:вхід живлення модуля, підключається до виходу 5 В.
- GND:загальний провід, який необхідно з’єднати з GND.
- D0:цифровий вихід, який змінює свій стан при перевищенні встановленого порогового значення концентрації газу.
- A0:аналоговий вихід, який надає сигнал, пропорційний рівню концентрації диму або газу в повітрі.

2.4 П’єзодинамік

П’єзоелектричний динамік, або зумер, є одним із найпоширеніших компонентів для створення звукових сигналів у проектах. Він широко використовується для подачі звукових оповіщень, індикації подій або навіть відтворення простих мелодій..

Ми розглянемо відмінності між активними і пасивними зумерами, розберемо схему підключення п’єзоелемента до плати ESP32.

2.4.1 Опис та схема роботи зумера

Зумер – це пристрій, призначений для створення звукових сигналів у різних електронних системах. Він використовується в побутовій техніці сигналізаціях, таймерах, медичних приладах, а також у мікроконтролерних проектах на базі ESP32, надаючи їм можливість відтворювати звукові сигнали, що базуються на двійковій системі числення 1 і 0 [11].



Рисунок 2.10 – П'єзоелемент зумер[36]

Зумер складається з металеві пластини, на яку нанесено напилення із струмопровідної кераміки. Вона є полярним елементом і має позначення «+» та «-»(рис. 2.11).

Принцип дії зумера ґрунтується на п'єзоелектричному ефекті, відкритому братами Кюрі. Коли на п'єзоелемент подається змінна напруга, керамічний шар починає періодично деформуватись, викликаючи механічні коливання металеві пластини. Ці коливання спричиняють удари об металеву пластину, створюючи звуковий сигнал певної частоти. Частота звуку залежить від швидкості перемикавання логічного сигналу, що дозволяє генерувати як короткі звукові імпульси, так і цілі мелодії.

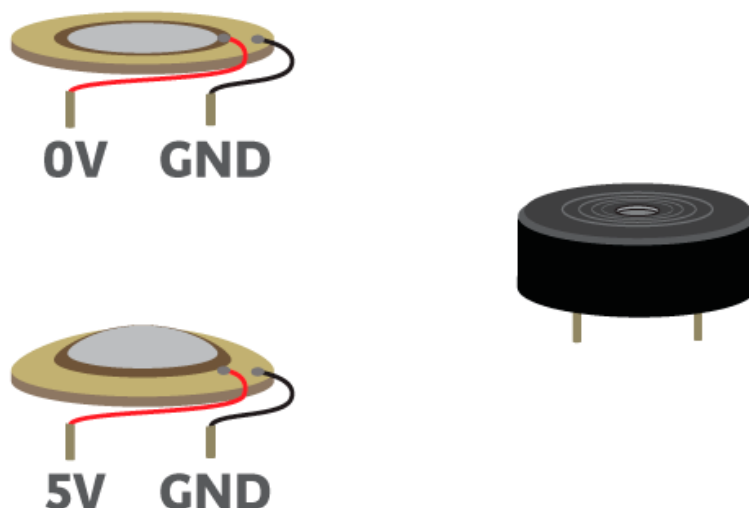


Рисунок 2.11 – Пристрій п'єзодинаміка зумера [36]

Існують два основні типи п'єзоелектричних зумерів: активний і пасивний, і вони працюють за схожим принципом, але мають деякі відмінності.

Активний зумер має фіксовану частоту звучання, яку визначають при виготовленні. Його підключення просте, оскільки для роботи достатньо підключити його до живлення, і він починає працювати без додаткових налаштувань. Після підключення він автоматично генерує звук без необхідності додаткового керування з боку мікроконтролера. Активні зумери зазвичай мають вищу гучність, але не дозволяють змінювати частоту звучання, що обмежує їх у створенні мелодій або складних сигналів.

Пасивний зумер, з іншого боку, дозволяє змінювати частоту звучання за допомогою програмного коду. Це дозволяє гнучко керувати звучанням, змінюючи тональність, частоту та ритм сигналу за допомогою програмного коду. Завдяки цьому пасивні зумери активно застосовуються у проєктах, де необхідно генерувати мелодії, послідовності звуків або реагувати на події різними звуковими ефектами. Проте їх гучність зазвичай нижча, ніж у активних аналогів, що потрібно враховувати при виборі [11].



Рисунок 2.12 – Модуль зумера для Ардуїно[36]

Що стосується підключення п'єзоелементів до ESP32, найкраще використовувати модулі з уже вбудованою обв'язкою. Це значно спрощує підключення і робить процес швидким і зручним. Модулі п'єзоелектричних зумерів для ESP32 зазвичай містять необхідну електроніку, що дозволяє легко інтегрувати звукові сигнали в ваші проекти без зайвих труднощів. Ви можете швидко підключити їх до плати ESP32 та налаштувати звук через програмний код.

Завдяки своїй простій конструкції п'єзоелектричні зумери є економічно вигідними в порівнянні з традиційними електромагнітними перетворювачами звуку, і їх гнучкість дозволяє варіювати звукові сигнали залежно від потреб вашого проекту.

2.4.2 Відмінності активного та пасивного зумеру

Основна відмінність між активним та пасивним зумером полягає в тому, як вони генерують звук. Активний зумер може самостійно генерувати звук, просто коли на нього подається напруга. Тобто для його роботи достатньо лише вмикати або вимикати подачу напруги. У свою чергу, пасивний зумер потребує зовнішнього

джерела сигналу, яке визначає параметри звукового сигналу, таких як плата ESP32. Активні зумери часто відтворюють більш гучний звук, з частотою близько 2,5 кГц (плюс-мінус 300 Гц), а їх напруга живлення може варіюватися від 3,5 до 5 В.

Активні зумери мають перевагу в тому, що їх не потрібно програмувати для генерації звуку. Це означає, що можна уникнути створення додаткового коду для затримок, що може спростити роботу в проекті. Окрім того, визначити наявність активного зумера можна за допомогою вимірювання опору між проводами, оскільки опір буде вищим, ніж у пасивного зумера.

Геометрично п'єзоелементи активного та пасивного зумерів дуже схожі, тому за їх формою буває складно визначити тип зумера. Проте активний зумер можна впізнати за наявністю резистора і підсилювача на платі, в той час як пасивний зумер зазвичай має тільки малий п'єзоелемент без додаткової електроніки.

2.5 Потенціометр

Потенціометр – це змінний резистор, який використовується для налаштування різних параметрів електричних схем. У робототехніці та електроніці він застосовується для регулювання гучності звуку, потужності, напруги, яскравості дисплея тощо. Підключаючи потенціометр до плати ESP32, можна використовувати його для зміни рівня сигналу, що надходить у мікроконтролер [13].

2.5.1 Принципи роботи потенціометра

Потенціометр складається з резистивного шару, вздовж якого рухається повзунок. Два крайніх виводи підключені до фіксованої напруги (наприклад, 5В і GND), а середній вивід (повзунок) дозволяє отримати змінну напругу залежно від його положення(рис. 2.13). Це дозволяє використовувати потенціометр для аналогового керування, наприклад, у схемах регулювання гучності чи яскравості.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 31
Зм.	Арк.	№ докум.	Підпис	Дата		

Завдяки простоті використання потенціометри широко застосовуються в електроніці та автоматизації, оскільки дозволяють зручно змінювати параметри без необхідності перепрограмування пристрою [13].



Рисунок 2.13 – Піни потенціометру [37]

Цифрові та аналогові потенціометри виконують однакову функцію – зміну опору, але реалізують це по-різному.

Цифровий потенціометр працює за принципом резистивних сходів, де кожен рівень опору перемикається електронними ключами. Це дозволяє задавати точні значення опору дискретними кроками. Основні переваги цифрових потенціометрів це: висока точність та стабільність, керування через мікроконтролер (наприклад, ESP32), відсутність механічного зносу, що підвищує надійність.

На відміну від цифрового, аналоговий потенціометр має плавну зміну опору, що робить його зручним для налаштувань у реальному часі. Його основні характеристики це: безперервне регулювання, простота використання, використовується у схемах керування гучністю, яскравістю та аналоговими сигналами

Обидва види потенціометрів знаходять широке застосування в електроніці, залежно від потреб конкретного проекту [13].

На рисунку 2.14 зображено потенціометр.



Рисунок 2.14 – Потенціометр [37]

Цифрові потенціометри в проектах часто використовуються через їхню точність, стабільність та можливість програмного керування. Вони зазвичай представлені у вигляді інтегральних схем, таких як MCP41010, AD5241, DS1803, які працюють через інтерфейси SPI або I2C.

2.6 РК дисплей

LCD 1602 є популярним рідкокристалічним дисплеєм, який широко застосовується в проектах на платформі Arduino. Він має дворядковий екран із 16 символами в кожному рядку, що дозволяє відобразити текстову інформацію, наприклад, параметри датчиків або повідомлення для користувача.

Цей дисплей працює на основі контролера Hitachi HD44780 та може підключатися до Arduino як у 4-бітному, так і в 8-бітному режимі. Однак, стандартне підключення може займати значну кількість контактів на платі. Для економії портів використовується спеціальний I2C-перетворювач (PCF8574), що дозволяє підключити дисплей через інтерфейс I2C. Це значно спрощує монтаж, оскільки

замість восьми проводів потрібно лише чотири. Крім того, дисплей оснащений підсвічуванням і можливістю регулювання контрасту, що робить його зручним у використанні [2].

2.6.1 РК дисплей Arduino LCD 1602

Рідкокристалічний дисплей (LCD) 1602 часто використовується для виведення текстової інформації в проєктах на Arduino завдяки своїй доступності, підтримці бібліотек та наявності модифікацій із різним кольором підсвічування. Попри зручність, цей дисплей вимагає підключення щонайменше шести цифрових виводів, що може стати незручністю при використанні контролерів із обмеженою кількістю пінів, таких як Arduino Uno чи Nano. Щоб уникнути перевантаження контактів, доцільно використовувати інтерфейс I2C, який дозволяє підключити дисплей лише через кілька пінів, значно спрощуючи схему [2].

2.6.2 Опис пінів РК дисплею I2C

На рисунку 2.15 зображені піни та їх назви РК дисплею I2C

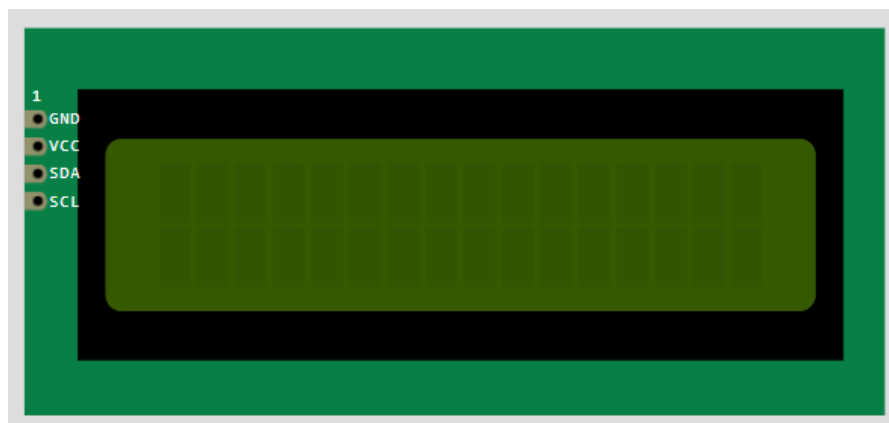


Рисунок 2.15 – Піни та їх назви РК дисплею I2C

Кожен із пінів має своє призначення:

1. GND – земля.

2. VCC – живлення.
3. SDA – Лінія передачі даних I2C.
4. SCL – Лінія синхронізації I2C.

Характеристики дисплея:

- символний тип відображення є можливість завантаження символів;
- світлодіодна підсвітка;
- контролер HD44780;
- напруга живлення 5В;
- формат 16x2 символів;
- Діапазон робочих температур від -20°C до $+70^{\circ}\text{C}$, діапазон температур зберігання від -30°C до $+80^{\circ}\text{C}$;
- кут огляду 180 градусів.

2.7 Макетна плата

Макетна плата – це універсальний інструмент в електроніці, який використовується для швидкого створення та тестування електричних схем без необхідності паяння. Вона являє собою пластикову основу з безліччю отворів, у які вставляються компоненти, такі як резистори, конденсатори, мікросхеми чи дроти, для створення тимчасових з'єднань. Всередині плати розташовані металеві контакти, які забезпечують електричний зв'язок між отворами.

Зазвичай макетна плата поділена на кілька зон. Центральна частина складається з рядів отворів, з'єднаних вертикально, що дозволяє підключати компоненти в схему. Крайні смуги, які називають шинами живлення, з'єднані горизонтально і використовуються для подачі напруги, наприклад, 5 В, та заземлення. Це дає змогу легко організувати живлення для всієї схеми. Отвори на платі розташовані на стандартній відстані, сумісній із ніжками більшості електронних компонентів, таких як інтегральні мікросхеми.

Макетні плати ідеально підходять для новачків, студентів або інженерів, які хочуть швидко перевірити ідею чи протестувати схему(рис 2.16). Наприклад, можна зібрати простий ланцюг із резистором і світлодіодом, підключивши їх до джерела живлення через шину, і одразу побачити результат. Такі плати дозволяють легко змінювати компоненти чи схему, просто виймаючи та вставляючи дроти чи деталі [25].



Рисунок 2.16 – Макетна плата

2.8 Wokwi для Arduino та ESP32

Wokwi – це сучасний онлайн-емулятор Arduino, який став чудовою альтернативою фізичним платам для розробки, тестування та налагодження мікроконтролерних проектів. Він повністю безкоштовний і працює напряму у браузері, тож не потрібно встановлювати жодного додаткового програмного забезпечення. Цей інструмент особливо зручний для тих, хто хоче швидко перейти до програмування та логіки роботи схем, не витрачаючи час на підключення фізичних компонентів. Wokwi підтримує не лише Arduino, а й ESP32, Raspberry Pi Pico та інші мікроконтролери, дозволяючи моделювати досить складні схеми, використовуючи готові бібліотеки та справжній Arduino-код. Це робить платформу надзвичайно ефективною як для навчання, так і для створення повноцінних прототипів.

Крім того, Wokwi дозволяє працювати з великою кількістю реалістичних компонентів, включаючи LCD-дисплеї, модулі Wi-Fi (ESP8266, ESP32), сенсори

газу, температури, вологості, серводвигуни та інші поширені елементи. У середовищі можна писати код на мові Arduino (C/C++) або використовувати MicroPython, що відкриває ширші можливості для розробників. Завдяки високій швидкості компіляції та зручному налагодженню, розробка проєктів стає інтуїтивно зрозумілою навіть без досвіду роботи з фізичним обладнанням.

2.9 Висновки до другого розділу

У результаті аналізу апаратної складової було визначено оптимальні технічні рішення для створення доступної та ефективної системи моніторингу газів у побутовому середовищі. Обґрунтовано вибір мікроконтролера ESP32, який завдяки вбудованому Wi-Fi-модулю забезпечує бездротову передачу даних, а також підтримує підключення додаткових модулів. Вивчено особливості роботи датчика MQ-2, що дозволяє виявляти низку небезпечних газів, таких як: метан, бутан, пропан і дим, що є критично важливим для забезпечення безпеки.

Розглянуто периферійні компоненти, включаючи потенціометр, LCD-дисплей для текстового відображення стану, а також звуковий зумер для оперативного попередження у разі критичних перевищень. Усі ці елементи забезпечують ефективний зворотний зв'язок із користувачем та підвищують зручність у використанні пристрою.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 37
Зм.	Арк.	№ докум.	Підпис	Дата		

3 ПІДКЛЮЧЕННЯ ОКРЕМИХ МОДУЛІВ

3.1 Підключення модуля датчика газу

Після ознайомлення з принципом роботи датчика MQ-2 можна переходити до його підключення до мікроконтролера. У нашому випадку використовується плата ESP32, яка забезпечує більшу функціональність у порівнянні з Arduino Uno.

Підключення здійснюється наступним чином:

Вивід VCC датчика підключається до 3.3V або 5V на платі ESP32 (залежно від сумісності модуля).

Вивід GND з'єднується з землею (GND) ESP32.

Вивід A0 підключається до аналогового входу ESP32 для зчитування рівня газу.

Якщо використовується цифровий вихід D0, його можна під'єднати до будь-якого цифрового входу ESP32 для визначення порогового рівня концентрації.

Таке підключення дозволяє пристрою зчитувати дані з датчика, аналізувати їх і, за необхідності, відправляти до хмарного сервісу чи відобразити на дисплеї(рис. 3.1). У Wokwi можна легко змоделювати цю схему, а результат візуально нагадує стандартне фізичне підключення на макетній платі.

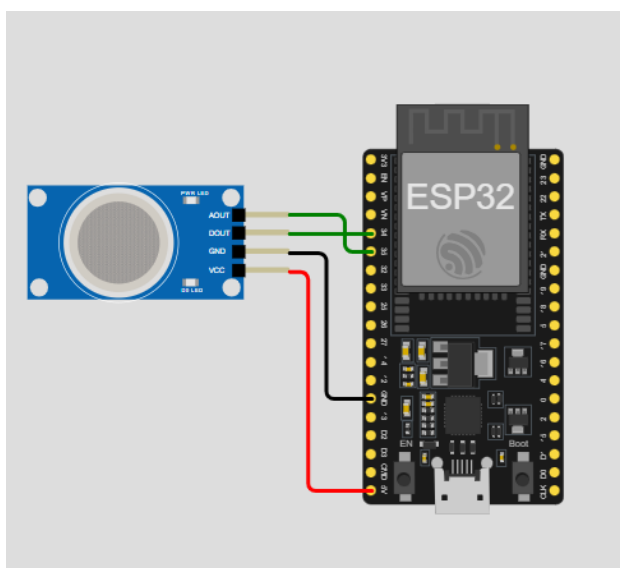


Рисунок 3.1 – Підключення датчика газу

3.2 Підключення зумера до ESP32

Підключення п'єзоелемента (зумера) до ESP32 є досить простим завданням, що не вимагає великої кількості компонентів. Завдяки низькому енергоспоживанню, зумер можна безпосередньо під'єднати до цифрового виходу мікроконтролера.

Для реалізації один із виводів зумера (зазвичай червоний) підключається до цифрового піну ESP32, що підтримує PWM (широтно-імпульсну модуляцію), а інший (зазвичай чорний або темного кольору) – до GND(рис. 3.2).

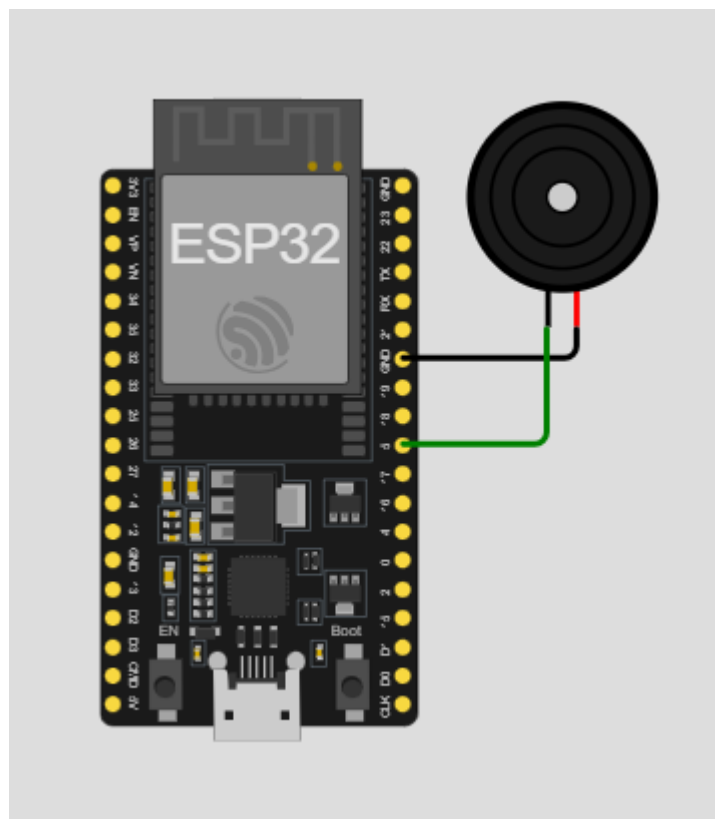


Рисунок 3.2 – Підключення зумера до ESP32

На схемах, змодельованих у Wokwi, чітко видно електричне з'єднання без додаткових модулів. Також іноді рекомендується використання резистора на сигнальному піні, що дозволяє стабілізувати рівень напруги та уникати сторонніх перешкод.

У деяких варіантах корпусів зумера можна знайти отвір для фіксації плати за допомогою гвинта. Зумер Arduino має два виходи, і важливо враховувати їхню полярність. Темний провід слід підключити до "землі", а червоний - до цифрового піна з PWM. Один з виводів налаштовується у програмі як «вхід». Arduino відстежує коливання напруги на виводі, на який подається напруга з кнопки, резистора та датчиків.

На рисунку 3.3 зображено зумер з назвою контактів

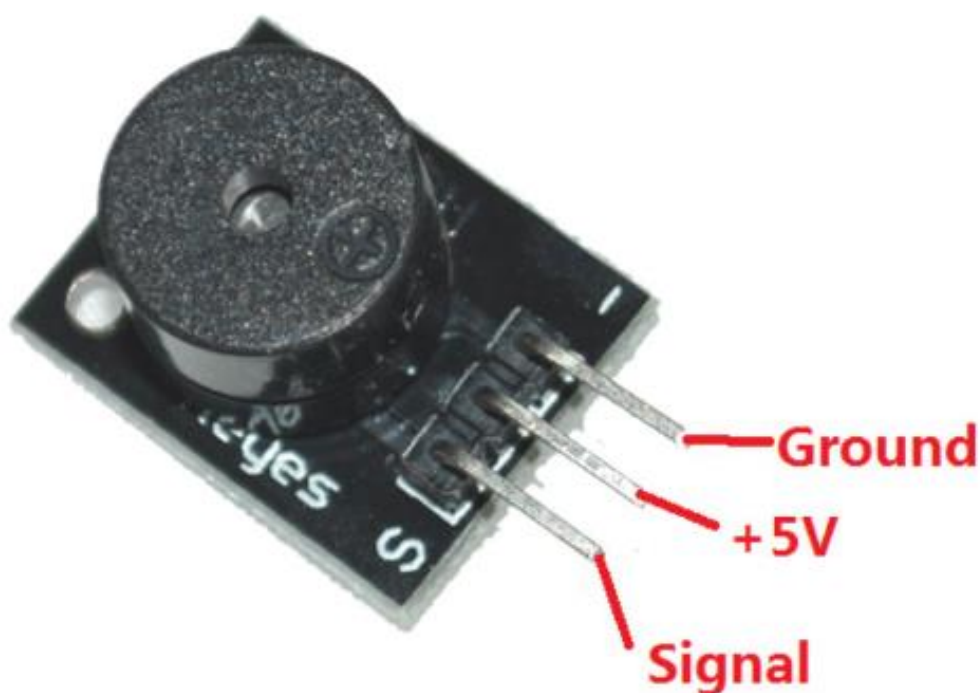


Рисунок 3.3 – Зумер з назвою контактів

Напруга, яка подається на вхід, має різні значення, але система чітко розпізнає лише два стани - логічні "0" та "1". Логічна одиниця відповідає напрузі від 2,3 до 5 В. У режимі виводу генерує логічний "0" або "1". Якщо обрати режим логічного "0", то напруга настільки мала, що недостатня для запалювання світлодіода.

На рисунку 3.4 зображена схема підключення зумера

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 40
Зм.	Арк.	№ докум.	Підпис	Дата		

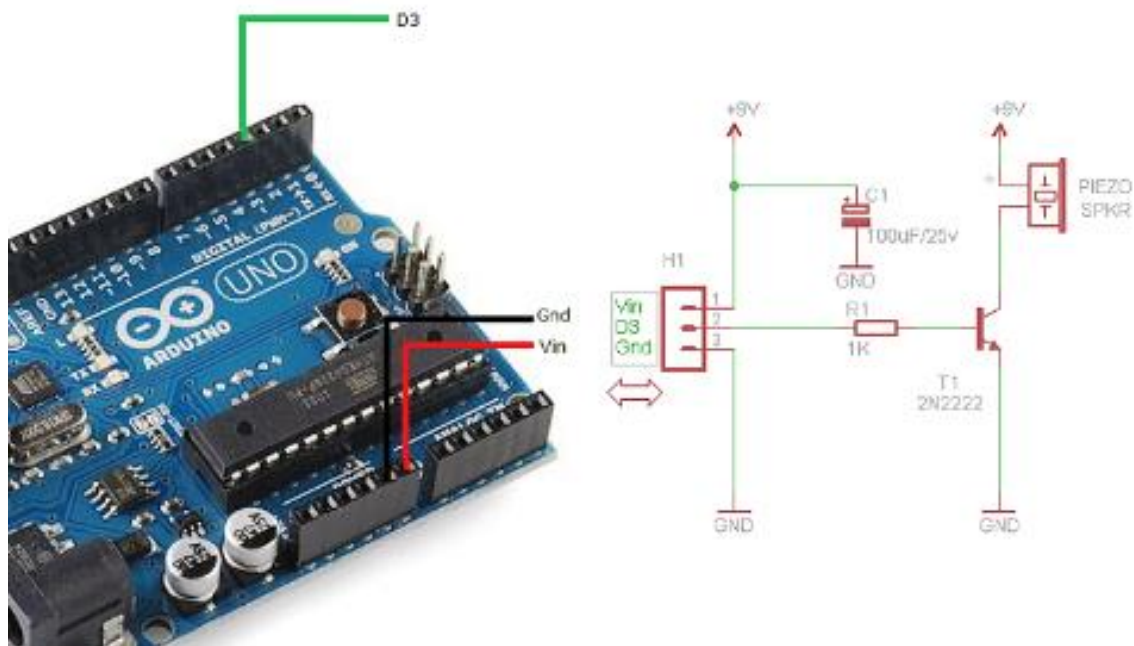


Рисунок 3.4 – Схема підключення зумера до Ардуїно[36]

Обережно використовуйте підключення, оскільки входи є досить чутливими до зовнішніх перешкод. Рекомендується підключати ніжку п'єзоелемента до висновку через резистор, що дозволить досягти високого рівня напруги на цій ніжці.

3.3 Підключення РК дисплею до ESP32

Підключення LCD-дисплея до ESP32 можна реалізувати двома способами: безпосередньо через паралельний інтерфейс або з використанням I2C-модуля, що значно спрощує монтаж. Стандартне підключення без I2C вимагає щонайменше 6 цифрових пінів, що може спричинити нестачу доступних портів на платі, особливо при використанні кількох модулів одночасно.

Стандартна схема приєднання монітора безпосередньо до мікроконтролера Ардуїно без I2C виглядає так(рис. 3.5).

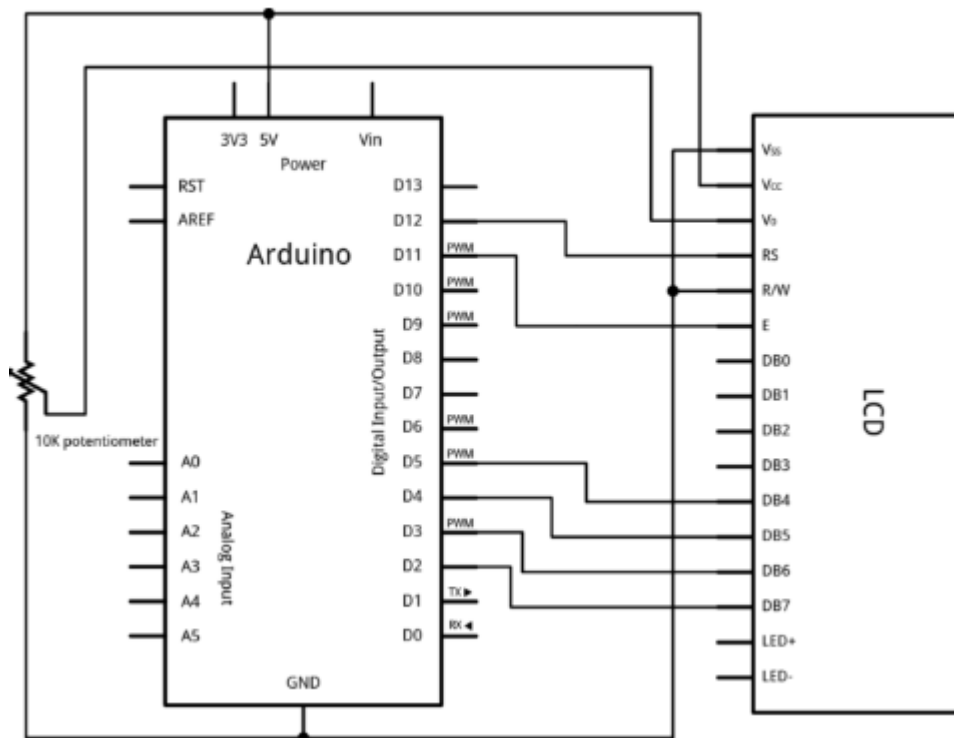


Рисунок 3.5 – Підключення дисплею без I2C

У проєкті було обрано підключення через I2C, яке потребує всього двох пінів ESP32(рис. 3.6).

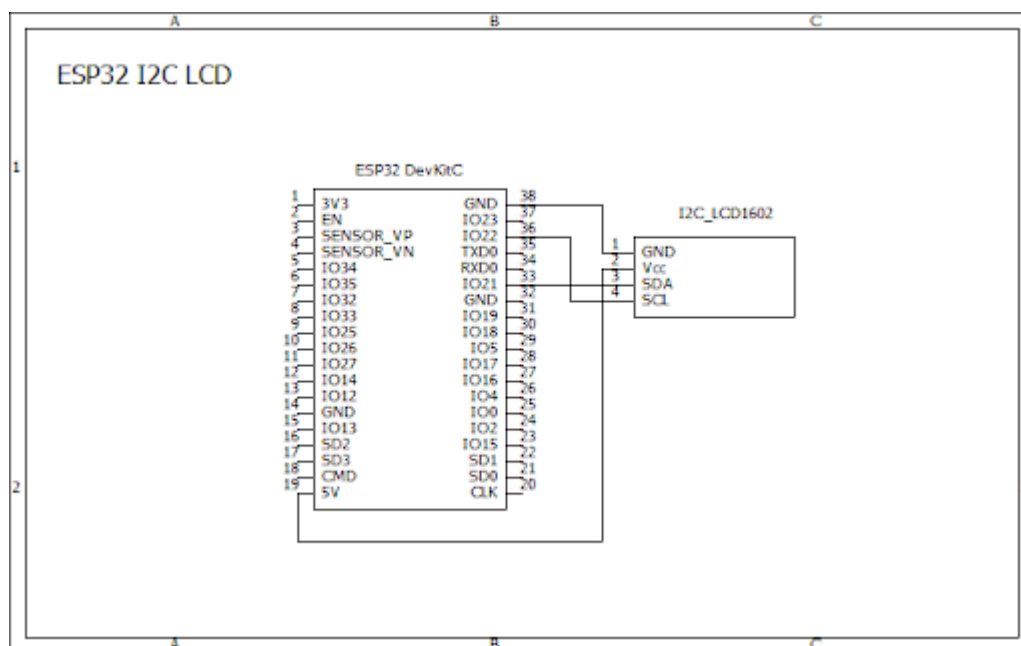


Рисунок 3.6 – Монітор з підтримкою I2C

Зм.	Арк.	№ докум.	Підпис	Дата

Монітор із рідким кристалом та підтримкою і2с приєднується до плати за допомогою чотирьох дротів: два для передачі даних та два для живлення.

- GND підключаються з GND плати.
- VCC підключаються до 5В.
- SDA (дані) підключається до одного з GPIO (наприклад, GPIO21).
- SCL (тактування) підключається до іншого GPIO (наприклад, GPIO22).

На рисунку 3.7 зображено підключення I2C з РК екраном

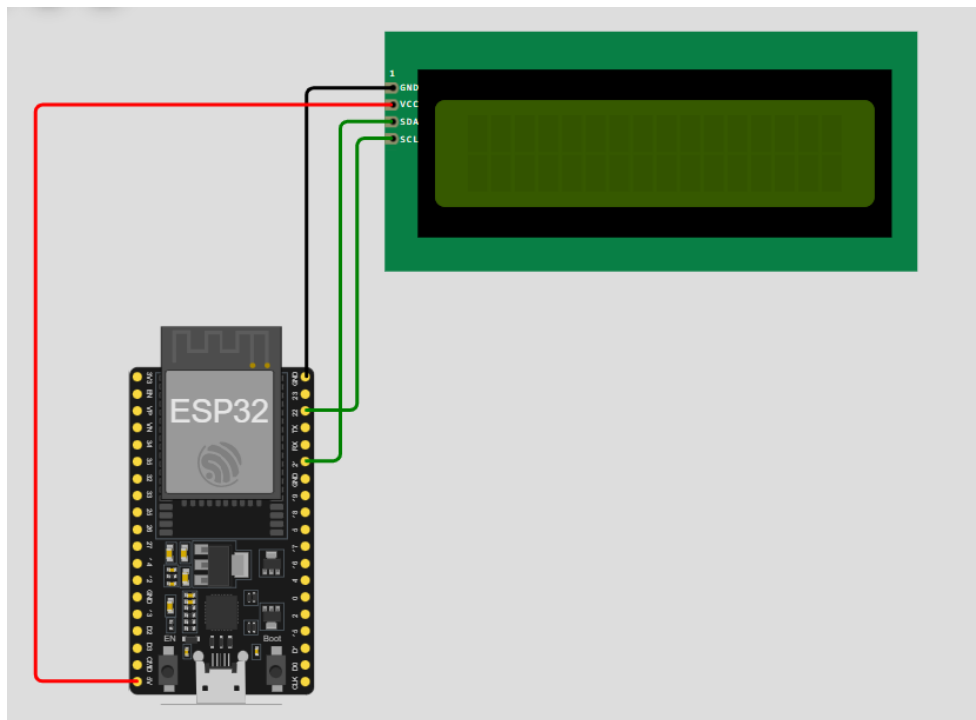


Рисунок 3.7 – Підключення I2C з РК екраном

Немає заплутаних проводів, які легко сплутати. Завдяки використанню інтерфейсу I2C підключення LCD-дисплея до ESP32 стає простим, охайним та зручним. Усього чотири дроти забезпечують повну функціональність дисплея, водночас звільняючи інші пін-контакти мікроконтролера для додаткових модулів. Вся складність протоколу I2C делегується бібліотекам, що значно спрощує написання коду. Це рішення забезпечує стабільну роботу системи, полегшує налагодження та дозволяє легко масштабувати проєкт у майбутньому.

3.4 Підключення потенціометра до ESP32

Підключення потенціометра до плати ESP32 виконується згідно зі схемою, зображеною на рисунку 3.8.

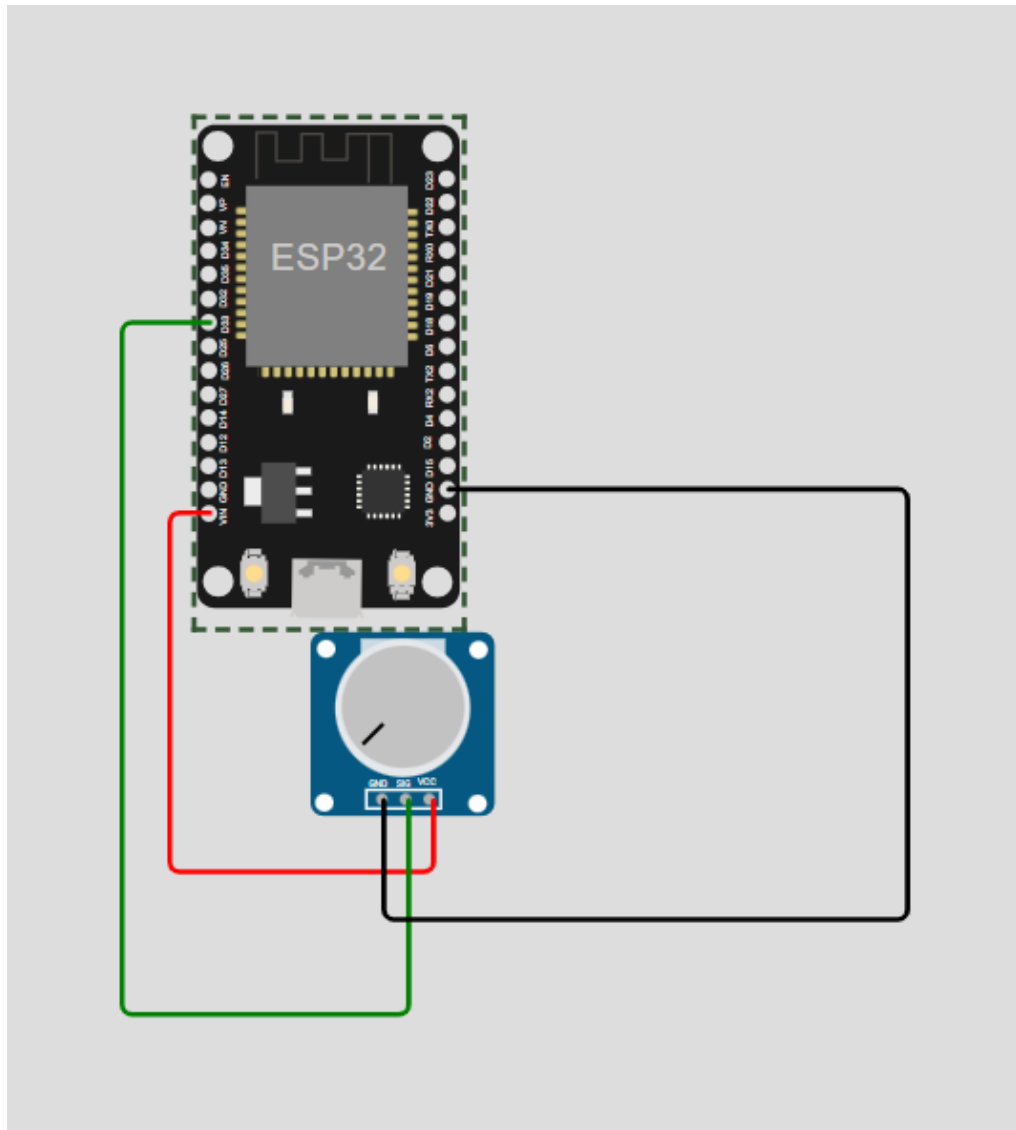


Рисунок 3.8 – Підключення потенціометра

Для цього необхідно з'єднати три виводи потенціометра з пін-контактами плати ESP32 наступним чином:

- чорний провід - до GND (земля);
- червоний провід до 3;

– зелений провід від центрального виводу потенціометра до одного з аналогових входів ESP32, наприклад, GPIO34.

Змінюючи положення ручки потенціометра, змінюється його опір, що, у свою чергу, змінює напругу на аналоговому вході мікроконтролера.

У платі ESP32 вбудовано аналого-цифровий перетворювач (ADC), який зчитує напругу на аналоговому вході та перетворює її у цифрове значення. Однак, на відміну від Arduino Uno, який має 10-бітний ADC (значення від 0 до 1023), ESP32 використовує 12-бітний ADC, що дозволяє отримувати значення в діапазоні від 0 до 4095.

При обертанні ручки потенціометра в одному напрямку напруга на аналоговому вході зменшується до 0 В, що відповідає цифровому значенню 0. При обертанні в протилежному напрямку напруга зростає до максимуму 3.3 В (оскільки ESP32 працює з 3.3-вольтовими рівнями), що відповідає значенню 4095.

3.5 Архітектура IoT

Хмарні сервіси набувають все більшої популярності у сфері зберігання та обробки даних, особливо в рамках проєктів Інтернету Речей (IoT). У цьому проєкті розглядається використання мікроконтролера ESP8266 для зчитування показників із підключеного датчика та подальшої передачі цих даних у хмарне середовище, зокрема - у Google Таблиці.

ESP8266 – це недорогий, але потужний мікроконтролер з вбудованим Wi-Fi, який широко використовується в IoT-застосуваннях. У нашому випадку він слугуватиме засобом збору інформації з датчика та передачі її через інтернет.

Для надсилання даних до хмари використовується Google Sheets як доступний і зручний хмарний інструмент. За допомогою спеціальної бібліотеки, яка дозволяє ESP8266 взаємодіяти з Google Sheets API, можна легко надсилати дані напряму в онлайн-таблицю, що дає змогу зберігати й аналізувати показники в реальному часі.

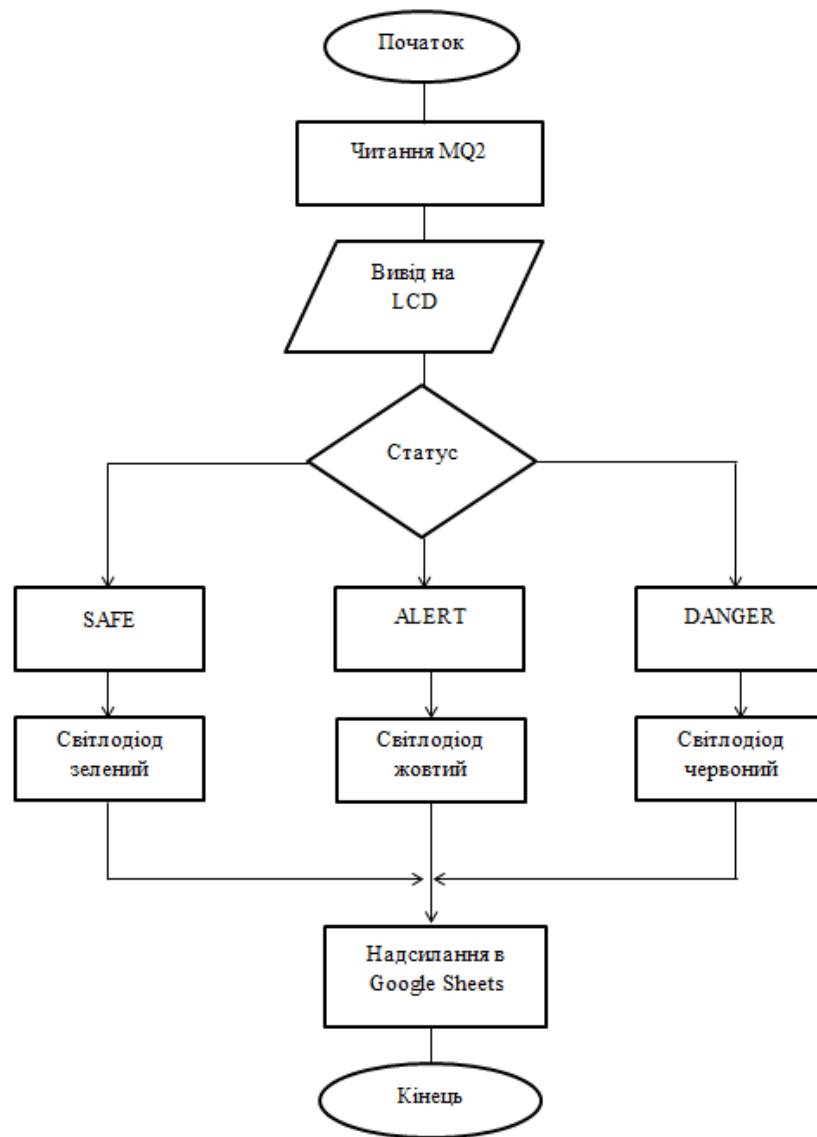


Рисунок 3.9 – Блок-схема логіки

У програмному кодї для ESP32 використовується спеціалізована бібліотека для зчитування даних з підключеного датчика, а також бібліотека для інтеграції з Google Sheets. Після отримання показників сенсора ці значення не лише виводяться на LCD-дисплей, але й надсилаються до онлайн-таблиці в Google. Для взаємодії з Google Sheets API використовується бібліотека GoogleSheets, яка значно спрощує процес інтеграції. Щоб усе працювало коректно, необхідно попередньо створити скрипт у Google Apps Script, який прийматиме та оброблятиме запити від мікроконтролера. Такий підхід демонструє типову архітектуру у сфері Інтернету

Речей (IoT), де взаємодіють різні рівні архітектури. Основні рівні включають рівень краю (Edge), рівень вузла (Endpoint) та рівень хмари (Cloud) [19].

Давайте розглянемо кожен з цих рівнів:

1. Рівень краю (Edge):

Рівень Edge охоплює пристрої, що працюють безпосередньо біля джерела даних, тобто на межі мережі. До цього рівня належать різноманітні IoT-пристрої, датчики, мікроконтролери (наприклад, ESP8266), вбудовані системи та шлюзи. Їхнє головне завдання – це первинна обробка інформації безпосередньо на місці її збору. Завдяки цьому вдається зменшити навантаження на мережу, забезпечити швидку реакцію на події, а також реалізувати локальне керування та базову аналітику ще до передачі даних, далі по мережі або в хмару.

2. Рівень вузла (Endpoint):

До рівня Endpoint належать окремі IoT-пристрої, що забезпечують збір, обробку та передачу даних. Це можуть бути датчики, актуатори, контролери або інші пристрої, які взаємодіють як з рівнем краю, так і з хмарною інфраструктурою. Основною їхньою функцією є отримання інформації з навколишнього середовища, виконання базових обчислень і передача зібраних даних далі до краю або безпосередньо в хмару. Вони є ключовими елементами в побудові IoT-систем, забезпечуючи зв'язок між фізичним середовищем та цифровими процесами.

3. Рівень хмари (Cloud):

Рівень хмари (Cloud) охоплює потужні обчислювальні ресурси та системи зберігання даних, які знаходяться в хмарному середовищі. Це можуть бути як публічні, так і приватні хмарні платформи, що надають високий рівень доступності та масштабованості для обробки даних. Основними функціями цього рівня є централізована обробка та аналіз великих обсягів даних, зберігання інформації, а також масштабування інфраструктури в залежності від потреб користувачів. Хмара дозволяє ефективно управляти даними, надавати послуги та забезпечувати високу надійність та доступність системи.

Зв'язок між рівнями в архітектурі IoT забезпечує ефективну обробку та передачу даних між пристроями та хмарними платформами. На рівні краю, дані можуть бути попередньо оброблені, що дозволяє зменшити обсяг переданих до мережі даних, передаючи лише найважливіші інформаційні фрагменти на рівень вузла. Рівень вузла, в свою чергу, забезпечує обмін даними між рівнем краю та хмарою, а також виконання локальних обчислень на пристроях, таких як датчики або актуатори. Дані з рівня вузла передаються на рівень хмари, де відбувається їх більш глибока обробка, аналіз, зберігання та масштабування обчислень для подальшого використання.

Архітектура, яка об'єднує ці три рівні, надає значну гнучкість і ефективність для реалізації IoT-проектів, дозволяючи вибір оптимального рівня обробки даних залежно від конкретних вимог кожного застосування. Вона забезпечує масштабованість і адаптивність, що є важливими аспектами для успішного впровадження різноманітних рішень в Інтернеті Речей [17].

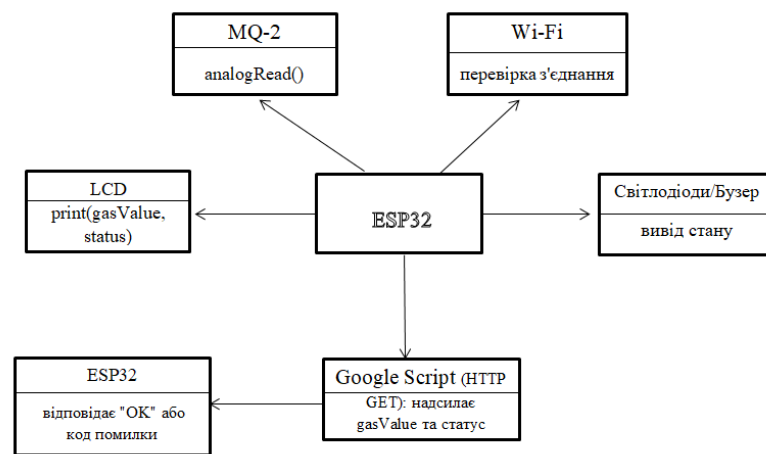


Рисунок 3.10 – Діаграма послідовності

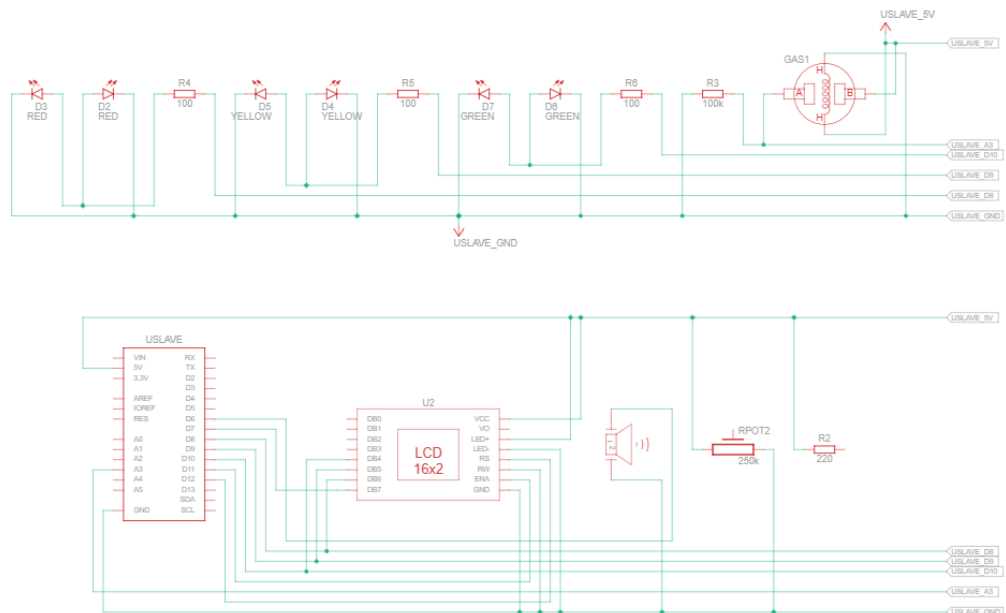


Рисунок 3.11– Схема електрична принципова

3.6 Надсилання даних у хмарне середовище

Одним із ключових етапів у створенні системи моніторингу стало забезпечення можливості автоматичного передавання зібраних даних до хмари для подальшого зберігання та аналізу. Це дозволяє не лише спростити доступ до історичних записів, а й забезпечити інтеграцію з іншими інструментами. Замість зберігання даних локально, було обрано варіант використання Google Таблиць як хмарного середовища, що доступне онлайн і підтримує автоматичну обробку запитів.

У системі використано кілька протоколів, які забезпечують зв'язок між компонентами та передачу даних до зовнішніх сервісів. Основними є I2C для локальної взаємодії з LCD-дисплеєм, Wi-Fi для підключення до мережі та HTTP для відправки даних до Google Sheets.

```
#include <LiquidCrystal_I2C.h>
#include <WiFi.h>
#include <HTTPClient.h>
```

Ці протоколи разом створюють основу для роботи IoT-пристрою.

Для забезпечення зв'язку з хмарною платформою пристрій підключається до мережі Wi-Fi для передачі інформації до Google Apps Script. Обробка даних на стороні хмари реалізована за допомогою простого скрипта на JavaScript, що приймає GET-запити, зчитує параметри та додає відповідні записи до Google Таблиці.

```
function doGet(e) {  
  if (!e || !e.parameter) {  
    return ContentService.createTextOutput("Error: No  
parameters provided");  
  }  
  var value = e.parameter.value || "0";  
  var status = e.parameter.status || "UNKNOWN";  
  var sheet =  
SpreadsheetApp.getActiveSpreadsheet().getActiveSheet();  
  sheet.appendRow([new Date(), value, status]);  
  return ContentService.createTextOutput("Success");  
}
```

Таким чином, кожне нове вимірювання фіксується у вигляді окремого рядка, що містить мітку часу, числове значення концентрації газу та відповідний статус. Це дозволяє здійснювати подальший аналіз змін рівня забруднення повітря в динаміці та забезпечує простий інтерфейс для моніторингу через веб-інтерфейс Google Sheets.

Використання Google Apps Script та HTTP-запитів дозволяє обійти потребу у складних брокерах повідомлень або зовнішніх базах даних, що є перевагою для невеликих проєктів або прототипів. Завдяки такій архітектурі дані стають доступними з будь-якого пристрою, підключеного до Інтернету, а простота реалізації не потребує глибоких знань веброзробки чи адміністрування серверів. Крім того, цей підхід є гнучким і масштабованим. При необхідності обробку можна автоматизувати за допомогою Google Apps Script або інтегрувати з іншими сервісами Google, такими як Google Data Studio для побудови дашбордів. У поєднанні з можливостями Wi-Fi модуля ESP32 така система формує основу для

побудови повноцінного IoT-рішення з низькими витратами на інфраструктуру [31].

На наступних зображеннях можете переглянути роботу інформаційної системи.

	A	B	C
1	Timestamp	Gas Value	Status
2	09.04.2025 20:00:24	3628	DANGER
3	09.04.2025 20:00:39	1438	ALERT
4	09.04.2025 20:00:59	1438	ALERT
5	09.04.2025 20:05:53	948	SAFE
6	11.04.2025 19:30:34	1251	ALERT
7	11.04.2025 19:31:04	1251	ALERT
8	11.04.2025 19:31:18	1712	ALERT
9	11.04.2025 19:45:33	3628	DANGER

Рисунок 3.12 – Таблиця Google Sheets

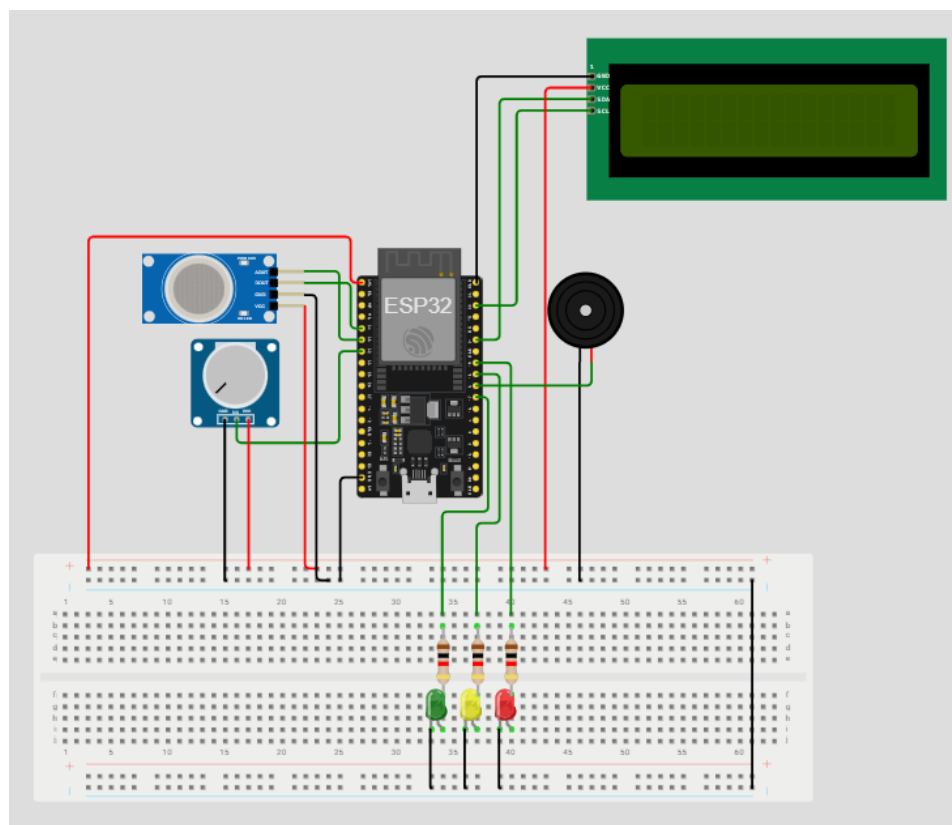


Рисунок 3.13 – Схема готового проєкту

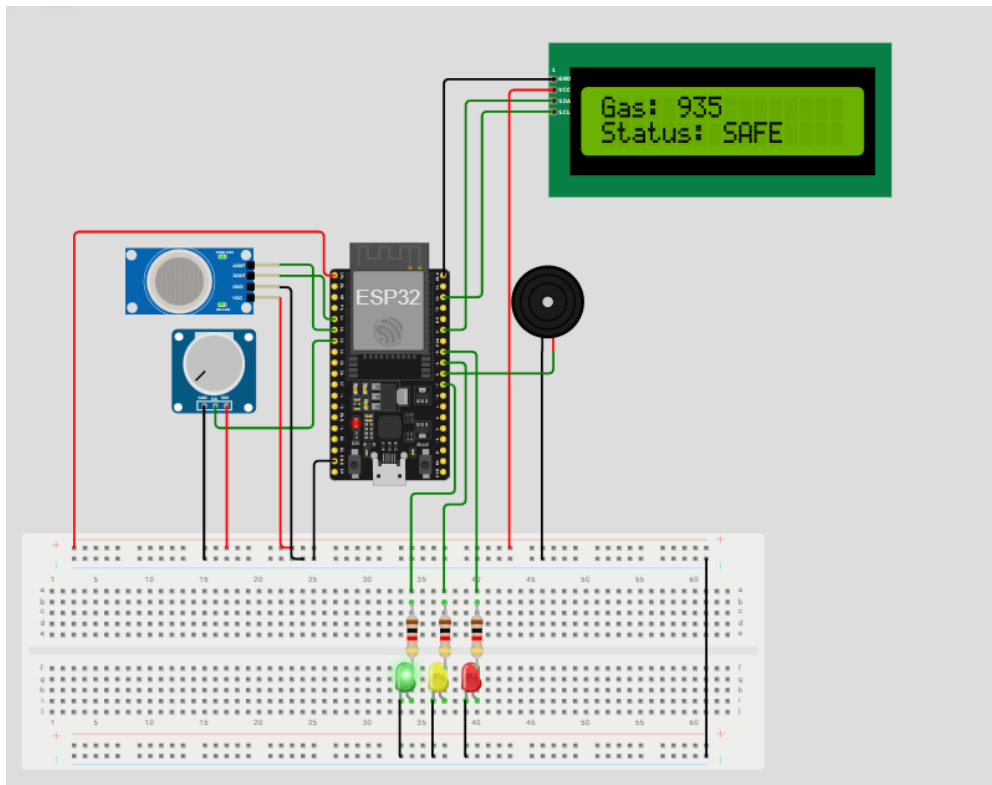


Рисунок 3.14 – Робота системи

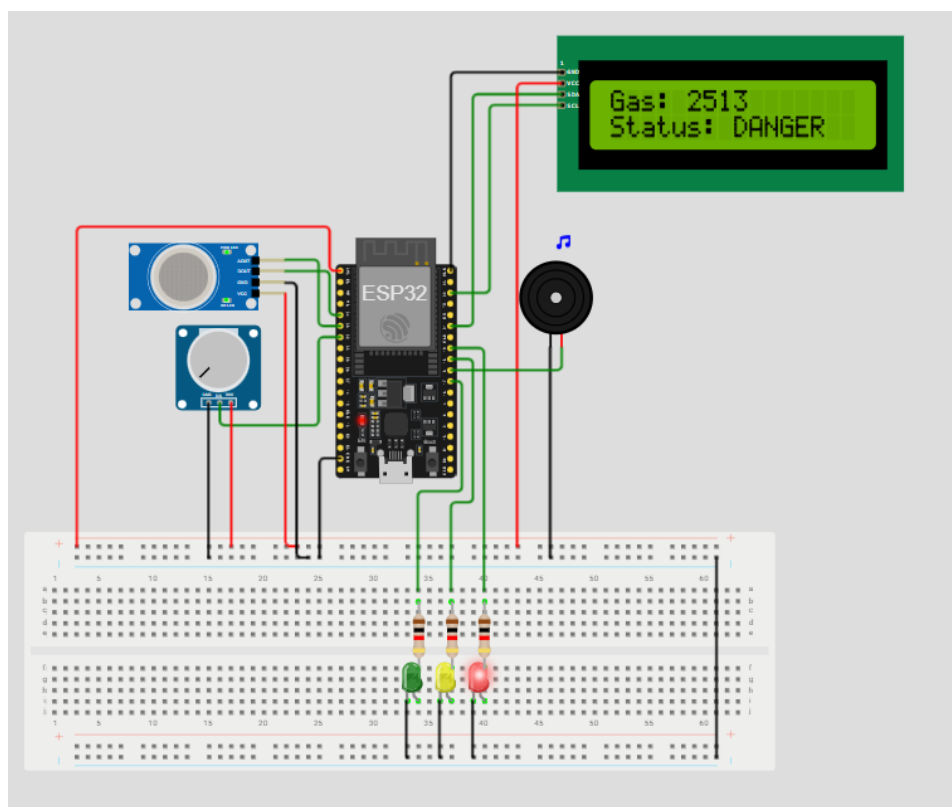


Рисунок 3.15 – Робота зумера при критичних показниках

Зм.	Арк.	№ докум.	Підпис	Дата

3.7 Висновки до третього розділу

У цьому розділі особливу увагу приділено практичній реалізації IoT-системи, яка ґрунтується на мікроконтролері ESP, датчику MQ-2, LCD-дисплеї та модулі Wi-Fi. Також розглянуто архітектуру інтернету речей, яка включає три основні рівні: крайовий (Edge), кінцевих пристроїв (Endpoint) та хмарний (Cloud). Детально описано підключення та принцип дії датчика газу MQ-2, п'єзоелемента (зумера), LCD-дисплея з інтерфейсом I2C і потенціометра. Кожен модуль підключено з урахуванням електричних характеристик ESP32 та особливостей роботи відповідного пристрою. Візуалізація підключень у симуляторі Wokwi дала змогу перевірити правильність з'єднань та попередити можливі помилки при фізичному монтажі.

Зібрані дані передаються до Google Sheets за допомогою HTTP-запиту, що ілюструє один із варіантів інтеграції IoT-пристроїв з хмарними сервісами. Такий підхід забезпечує постійний доступ до даних з будь-якої точки світу та можливість їх подальшого аналізу.

Отримані знання та практичні навички стануть основою для подальшої інтеграції компонентів у повноцінну функціональну систему.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 53
Зм.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У межах даної дипломної роботи було спроектовано та реалізовано прототип системи моніторингу газу у приміщенні з використанням мікроконтролера ESP32. Для візуалізації даних використовувався LCD-дисплей, а зчитування концентрації газів здійснювалося за допомогою сенсора MQ-2. Додатково до системи було інтегровано потенціометр, як приклад аналогового керування. Особливістю проєкту є передача зібраних даних до Google Sheets, що дозволяє зберігати показники в хмарному середовищі та відстежувати їх у режимі реального часу.

Розробка та тестування проєкту відбувалися у Wokwi – сучасному онлайн-середовищі моделювання для мікроконтролерів, що є зручною альтернативою популярному Tinkercad Circuits, оскільки забезпечує ширшу підтримку платформ, таких як ESP32. На відміну від Arduino Uno, який часто використовується для навчання, ESP32 обрано через його розширені функціональні можливості, зокрема підтримку Wi-Fi, що й дозволило реалізувати обмін даними з Google Sheets.

Також у роботі було розглянуто архітектуру Інтернету Речей (IoT), яка складається з трьох основних рівнів: краю (Edge), вузла (Endpoint) та хмари (Cloud). Зібрані з сенсорів дані обробляються на рівні вузла та передаються на хмарний рівень, де відбувається їх збереження та подальший аналіз. Такий підхід дозволяє забезпечити не лише локальне сповіщення про небезпеку, але й віддалений моніторинг, що є важливим у контексті промислового використання або для користувачів, які перебувають поза межами приміщення.

У проєкті використано декілька протоколів, а саме: I2C, Wi-Fi та HTTP. I2C забезпечує локальну взаємодію з дисплеєм, Wi-Fi підключає систему до інтернету, а HTTP дозволяє передавати дані до хмарного сервісу.

Зв'язок із хмарою реалізовано за допомогою скрипта на JavaScript у Google Apps Script, що приймає GET-запити, зчитує параметри та додає відповідні записи до Google Таблиці.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 54
Зм.	Арк.	№ докум.	Підпис	Дата		

У підсумку, реалізована система демонструє, як сучасні мікроконтролери, сенсорні модулі та хмарні сервіси можуть взаємодіяти для створення ефективних та розширюваних IoT-рішень. Отримані знання та досвід можуть бути використані в подальших розробках у сфері автоматизації, екологічного моніторингу, безпекових систем та смарт-систем, сприяючи підвищенню якості життя.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		55

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Aghenta L. O., Iqbal M. T. Low-cost, open source IoT-based SCADA system design using thinger. IO and ESP32 thing. *Electronics*. (2019).Vol.8(8). P. 1-24.
2. Akinwole O. O. Design, simulation and implementation of an Arduino microcontroller based automatic water level controller with I2C LCD display. *International Journal of Advances in Applied Sciences (IJAAS)*. 2020. Vol. 9, No. 2. P. 77–84.
3. Albulayhi K., Smadi A. A., Sheldon F. T., Abercrombie R. K. IoT intrusion detection taxonomy, reference architecture, and analyses. *Sensors*. 2021. Vol. 21, No. 19. P. 1-30.
4. Alwahedi F., Aldhaheer A., Ferrag M. A., Battah A., Tihanyi N. Machine learning techniques for IoT security: Current research and future vision with generative AI and large language models *Internet of Things and Cyber-Physical Systems*. 2024. Vol. 4. P. 167–185.
5. Anusha M., Kumar P. B., Akhil V., Gouthami M., Chinnaiah M. C., Shaik S. Internet of Things (IOT) based energy monitoring with ESP 32 and using Thingspeak. 2024 *10th International Conference on Communication and Signal Processing (ICCSP)* (April 2024). IEEE, 2024. P. 5–20.
6. Daruwati I., Hatika R. G., Mardiansyah D. MQ-2 gas sensor using micro controller arduino uno for LPG leakage with short message service as a media information. *Journal of Physics: Conference Series*. 2021. Vol. 2049, No. 1. P. 012068.
7. Dedeoglu V. et al. A trust architecture for blockchain in IoT. *Proceedings of the 16th EAI international conference on mobile and ubiquitous systems: computing, networking and services*. 2019. C. 190-199.
8. Easterline L. M., Putri A. A. Z. R., Atmaja P. S., Dewi A. L., Prasetyo A. Smart Air Monitoring with IoT-based MQ-2, MQ-7, MQ-8, and MQ-135 Sensors using NodeMCU ESP32. *Procedia Computer Science*. 2024. Vol. 245. P. 815–824.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 56
Зм.	Арк.	№ докум.	Підпис	Дата		

9. Fatema T., Hakim M. A., Mim T. K., Mitu M. J., Paul B. IoT cloud based noise intensity monitoring system. *Indonesian Journal of Electrical Engineering and Computer Science*. 2023. Vol. 30, No. 1. P. 289–298.

10. Gurski J., Quach L. M. Display technology overview. *Lytica White Paper*. 2005. No. 37. P. 1-37.

11. Hercog D., Lerher T., Truntič M., Težak O. Design and implementation of ESP32-based IoT devices. *Sensors*. (2023). Vol.23(15). P. 5-20. .

12. Jamil N. A. A., Jumaat S. A., Salimin S., Abdullah M. N., Nor A. F. M. Performance enhancement of solar powered floating photovoltaic system using arduino approach . *International Journal of Power Electronics and Drive Systems*. 2020. Vol. 11, No. 2.P. 1-7.

13. Kumar G. N., Bhavya A., Balaji J., Dhanunjay S., Vikasitha P., Rafi V. Smart Home Light Intensity Control using Potentiometer method for Energy Conservation. *2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*. August 2021. IEEE, 2021. P. 371–375.

14. Kurniawan A. Internet of Things Projects with ESP32: Build exciting and powerful IoT projects using the all-new Espressif ESP32. Packt Publishing Ltd, 2019. P. 200-231.

15. Li C., Wang J., Wang S., Zhang Y. A review of IoT applications in healthcare *Neurocomputing*. 2024. Vol. 565. P. 1-12.

16. Ray A. K., Bagwari A. IoT based Smart home: Security Aspects and security architecture. *2020 IEEE 9th international conference on communication systems and network technologies (CSNT)*. IEEE, 2020. C. 218-222.

17. Shahinzadeh H., Moradi J., Gharehpetian G. B., Nafisi H., Abedi M. IoT architecture for smart grids. *International Conference on Protection and Automation of Power System (IPAPS)*. 22–30 January 2019. IEEE, 2019. P. 22–30.

18. Vasan D., Alazab M., Venkatraman S., Akram J., Qin Z. MTHAEL: Cross-architecture IoT malware detection based on neural network advanced ensemble learning. *IEEE Transactions on Computers*. 2020. Vol. 69, No. 11. P. 1654–1667.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 57
Зм.	Арк.	№ док.ум.	Підпис	Дата		

19. Villa V. et al. Iot open-source architecture for the maintenance of building facilities. *Applied Sciences*. 2021. Vol. 11. No 12. С. 5374.

20. Величко Д., Олійник М. Можливості використання газових аналізаторів для дослідження якості повітря. *Природничі та гуманітарні науки. Актуальні питання* : матеріали III Міжнародної студентської науково-технічної конференції. 2020 р. С. 133–134.

21. Домрачева К. О., Довженко Н. М., Дмитренко В. В. Аналіз технологій та стандартів зв'язку для мережі ІОТ. *Наукові записки Державного університету інформаційно-комунікаційних технологій*. 2019. (№ 3). С. 54–62.

22. Жидка О. В., Андрійченко Т. Р. Інформаційна безпека систем ІоТ Зв'язок. 2024. (№ 4). С. 65–69.

23. Захарова О. О. Стандарти безпеки для пристроїв ІоТ. *Free and Open Source Software: матеріали XV-ої Міжнародної науково-практичної конференції*. Харків: Харківський національний економічний університет імені Семена Кузнеця. (13–14 лютого 2024 р., Харків). 2024. С. 22-150.

24. Клепцов А., Гусєва-Божаткіна В. А. Розробка апаратно-програмного комплексу з використання штучного інтелекту для калібрування датчику газу MQ-2. *Вісник Херсонського національного технічного університету*. 2024. № 4 (91). С. 280–285.

25. Кулакевич О. Р., Грудецький Р. Я., Сацик В. О., Маркіна Л. М. Розробка ІоТ: iOS-застосунок для управління обладнанням. *Перспективні технології та прилади*. 2024. Т. 1, (№ 25). С. 88–92.

26. Назар О. Ч. Протокол MQTT та його застосування у сфері ІоТ. *Молодь в науці: дослідження, проблеми, перспективи (МН-2025)* : матеріали інтернет-конф. 21 Листопада 2024 р.

27. Павлюк І. Системи кібербезпеки для ІоТ-пристроїв в домашніх умовах: матеріали конференцій МЦНД (Полтава, 8 листопада 2024 р.). Полтава, 2024. С. 310–312.

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 58
Зм.	Арк.	№ докум.	Підпис	Дата		

28. Пилявець А. І. Інтернет речей (IoT) та його вплив на автоматизацію промисловості. *Science of the 21st century: searches, problems, development prospects*. The 27th International scientific and practical conference. (9–12 липня 2024 р., Париж, Франція). Paris: International Science Group. 2024. С. 210-285.

29. Станько А. А., Дудирев Д. Ю., Козачок М. В. Інтелектуальна система управління трафіком на основі IoT. *Актуальні задачі сучасних технологій: бірник тез доповідей XIII Міжнародної науково-практичної конференції молодих учених та студентів*. 2024 р. С. 503–504.

30. Тимошук В., Василишин В., Мудрий І., Луцик Н. Огляд та порівняння протоколів передачі інформації в IoT. *Інформаційні моделі, системи та технології*. Матеріали XI науково-технічної конференції. 2023 р. С. 188-189.

31. Шабала Є., Корнійчук Б. Методологія оцінювання безпеки IoT на промислових об'єктах. *Управління розвитком складних систем*. 2024. (№ 60). С. 146–155.

32. Шинкарук Х. М., Чеховський С. А. Особливості роботи термокаталітичного давача в середовищі природного газу. *Перспективні технології та прилади*. 2020. (№ 16). С. 155–159.

33. MQ2 Gas sensor works on 5V DC and draws around 800mW. It can detect LPG, Smoke, CO and other gases. URL: https://www.researchgate.net/figure/MQ2-MQ2-Gas-sensor-works-on-5V-DC-and-draws-around-800mW-It-can-detect-LPG-Smoke_fig1_385585793 (дата звернення: 27.05.2025)

34. Принцип роботи сенсора MQ-2. Zhihu Zhuanlan. URL: <https://zhuanlan.zhihu.com/p/340078614> (дата звернення: 27.05.2025)

35. How to Use MQ2 Gas Smoke Sensor with Arduino. Arduino Expert. URL: <https://arduinoexpert.com/blog/how-to-use-mq2-gas-smoke-sensor-with-arduino/> (дата звернення: 27.05.2025)

36. Francesco de Palma. Arduino lezione 3. Uni Salento & INFN Lecce. URL: https://www.dmf.unisalento.it/~fdepalma/lezioni_arduino/Arduino_lezione3.pdf (дата звернення: 27.05.2025)

					КВРІСТ 2101035.21.01.07 ПЗ	Арк. 59
Зм.	Арк.	№ докум.	Підпис	Дата		

37. Потенціометр(логарифмічний, 50 кОм). RCS Components. URL:
<https://www.rcscomponents.kiev.ua/search?q=A503> (дата звернення: 27.05.2025)

					КВРІСТ 2101035.21.01.07 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		60

Додаток А
(обов'язковий)

**КОПІЯ КРЕСЛЕННЯ «АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ
ПРОЄКТУ»**

КАРСТ 2101035.21.01.07

Інформаційна система датчика газу

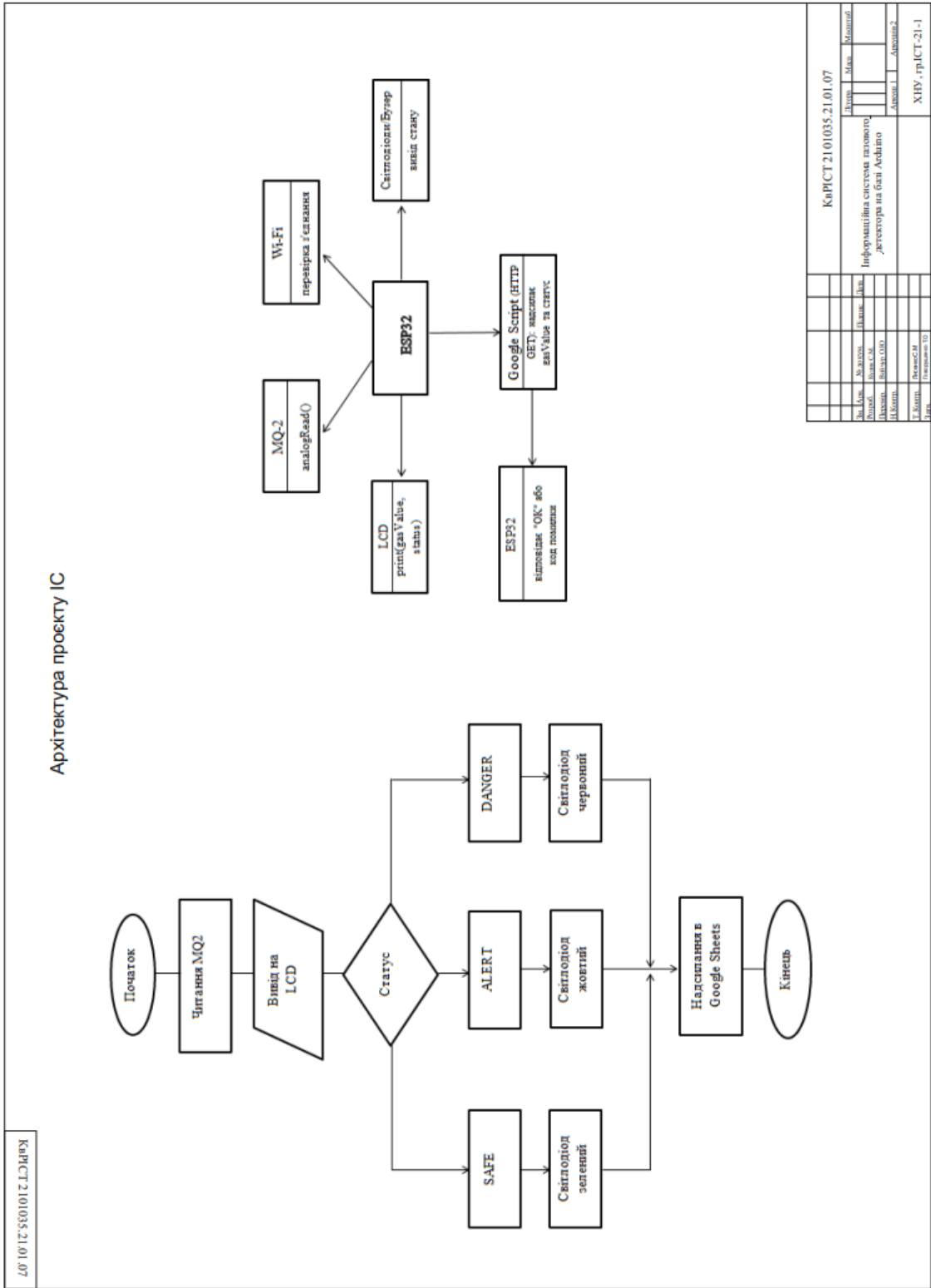
КАРСТ 2101035.21.01.07

	A	B	C
1	Timestamp	Gas Value	Status
2	09.04.2025 20:00:24	3628	DANGER
3	09.04.2025 20:00:39	1438	ALERT
4	09.04.2025 20:00:59	1438	ALERT
5	09.04.2025 20:05:53	948	SAFE
6	11.04.2025 19:30:34	1251	ALERT
7	11.04.2025 19:31:04	1251	ALERT
8	11.04.2025 19:31:18	1712	ALERT
9	11.04.2025 19:45:33	3628	DANGER

КАРСТ 2101035.21.01.07			
Ім'я:	Пізнак:	Дата:	Категорія:
			Інформаційна система газового детектора на базі Arduino
Ім'я:	Пізнак:	Дата:	Версія:
Ім'я:	Пізнак:	Дата:	Версія:
XIV, гр.ІСТ-21-1			

Додаток Б (обов'язковий)

КОПІЯ КРЕСЛЕННЯ «АРХІТЕКТУРА ПЗ ДЛЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ»



Додаток Г (обов'язковий)

ЛІСТИНГ КОДУ

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <WiFi.h>
#include <HTTPClient.h>

// Wi-Fi дані (для Wokwi або власної мережі)
const char* ssid = "Wokwi-GUEST";
const char* password = "";

// Ваша Google Script ID
const char* googleScriptId = "AKfycbybVfW6d-
b3maxWBc8jv_qB3zq8LgbgqpBft4gtd7Evcd6RHpryODvFz6hiFoSbD91zrQ";

// Піни ESP32
#define GREEN_LED 17
#define YELLOW_LED 18
#define RED_LED 19
#define BUZZER 5
#define MQ2_PIN 35

LiquidCrystal_I2C lcd(0x27, 16, 2);
String status;

void setup() {
  Serial.begin(115200);
  lcd.init();
  lcd.backlight();

  pinMode(GREEN_LED, OUTPUT);
  pinMode(YELLOW_LED, OUTPUT);
```

```

pinMode(RED_LED, OUTPUT);
pinMode(BUZZER, OUTPUT);
digitalWrite(GREEN_LED, LOW);
digitalWrite(YELLOW_LED, LOW);
digitalWrite(RED_LED, LOW);
noTone(BUZZER);

connectToWiFi();
}

void connectToWiFi() {
  Serial.println("Connecting to Wi-Fi...");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWi-Fi connected");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
  int gasValue = analogRead(MQ2_PIN);
  Serial.print("Gas value: ");
  Serial.println(gasValue);

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Gas: ");
  lcd.print(gasValue);

  if (gasValue < 1000) {
    digitalWrite(GREEN_LED, HIGH);
    digitalWrite(YELLOW_LED, LOW);

```

```

    digitalWrite(REDD_LED, LOW);
    noTone(BUZZER);
    lcd.setCursor(0, 1);
    lcd.print("Status: SAFE");
    status = "SAFE";
} else if (gasValue >= 1000 && gasValue < 2000) {
    digitalWrite(GREEN_LED, LOW);
    digitalWrite(YELLOW_LED, HIGH);
    digitalWrite(REDD_LED, LOW);
    noTone(BUZZER);
    lcd.setCursor(0, 1);
    lcd.print("Status: ALERT");
    status = "ALERT";
} else {
    digitalWrite(GREEN_LED, LOW);
    digitalWrite(YELLOW_LED, LOW);
    digitalWrite(REDD_LED, HIGH);
    tone(BUZZER, 1000);
    lcd.setCursor(0, 1);
    lcd.print("Status: DANGER");
    status = "DANGER";
}

if (WiFi.status() != WL_CONNECTED) {
    Serial.println("Reconnecting to Wi-Fi...");
    connectToWiFi();
}

HTTPClient http;
String url = "https://script.google.com/macros/s/" +
String(googleScriptId) + "/exec?value=" + String(gasValue) +
"&status=" + status;
Serial.println("Sending to: " + url);
http.begin(url);
int httpCode = http.GET();

```

```

if (httpCode > 0) {
    String response = http.getString();
    Serial.println("Google Sheets response: " + response);
} else {
    Serial.println("Error: " + String(httpCode));
}
http.end();
delay(5000);
}

```

Apps Script:

```

function doGet(e) {
    if (!e || !e.parameter) {
        return ContentService.createTextOutput("Error: No
parameters provided");
    }

    var value = e.parameter.value || "0";
    var status = e.parameter.status || "UNKNOWN";

    var sheet =
SpreadsheetApp.getActiveSpreadsheet().getActiveSheet();
    sheet.appendRow([new Date(), value, status]);

    return ContentService.createTextOutput("Success");
}

```

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:

Співавтор:

Назва: Кулик_Інформаційна система газового детектора на базі Ардуіно

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1: 7.9%

Коефіцієнт подібності 2: 3.4%

Мікропробіли: 8

Заміна букв: 2

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-01 14:20:36.0

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 процедур. Таким чином робота не приймається.

Обґрунтування:

2025-06-01

Дата



Доцент Андрій Нічепорук

експерт

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 6.0%

Dictionaries check: en_US, ru_RU, ua_UA. Errors in the documents: 10%

ID: 242703 Title: БКР Інформаційна система газового детектора на базі Ардуіно Added in a DB: 2025-06-01 Authors: Софія КУЛИК Heads: Олег ВОЙЧУР Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	61796	493	7189 (12%)	65 (13%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

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

Софії КУЛИК

ІІБ здобувача вищої освіти

ФІТ, 4 курсу, групи ІСТ-21-1

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2025, згідно з яким виявлення плагіату, є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомена. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщена 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.

01.06.2025 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: Інформаційна система газового детектора на базі Ардуіно

Автор: Софія Кулик

Спеціальність: 126 – Інформаційні системи та технології

Освітня програма: освітньо-професійна

Науковий керівник: Олег Войчур, 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, які містяться в розділах аналізу, присвячених опису апаратних компонентів не стосуються авторських технічних чи програмних рішень, реалізованих у проєкті;
- 2) усі запозичення є фрагментарними, загальноживаними або мають відповідне посилання на джерело, де це доцільно з методичної точки зору;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з 10-40 джерелами на один фрагмент речення;
- 4) деякі збіги стосуються форматів коду або фрагментів програм, які є типовими для пристроїв, зокрема зчитування даних із датчиків та взаємодії з хмарними сервісами, що не підлягає авторському праву як уніфіковані технічні інструкції;
- 5) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості StrikePlagiarism, складає 7.85% і адресується до 401 першоджерела, що з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС

Олег ВОЙЧУР

Єлизавета ГНАТЧУК

Ольга ПАВЛОВА

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Кулик Софія Михайлівна

Тема: Інформаційна система газового детектора на базі Ардуіно

Спеціальність: 126 «Інформаційні системи та технології»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3 Кількість сторінок записки 55

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є розробка інформаційної системи газового детектора на базі Ардуіно
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 дослідження предметної області (проаналізовано основи технології Інтернету речей (IoT), структуру IoT-систем, типові сенсорні модулі та сучасні підходи до побудови систем моніторингу навколишнього середовища). Виявлено актуальні проблеми, пов'язані з безпекою житлових приміщень, що зумовлюють необхідність своєчасного виявлення газових загроз, та виконано постановку задачі дослідження. В другому розділі кваліфікаційної роботи виконано огляд апаратної складової системи. Проаналізовано принципи роботи основних апаратних компонентів: мікроконтролера ESP32, датчика MQ-2, п'єзоелемента, LCD-дисплея з інтерфейсом I2C та потенціометра. Пояснено їх взаємодію в контексті реалізації завдань моніторингу та сигналізації. Описані апаратні можливості ESP32 як основи для реалізації IoT-проектів з бездротовою передачею даних. В третьому розділі кваліфікаційної розглянуто архітектуру IoT-системи та реалізацію передачі даних у хмарне середовище, а також виконано підключення апаратних модулів, що забезпечують роботу всієї системи. Окрім локальної взаємодії компонентів, важливою частиною реалізації стала передача зчитаних даних у хмару. Для цього

було використано протокол Wi-Fi для з'єднання ESP32 з мережею Інтернет та HTTP-запити до Google Apps Script

4. Позитивні сторони роботи: висока практична цінність роботи.

5. Негативні сторони роботи: у межах роботи симуляція системи проводилася лише в середовищі Wokwi, що не дозволяє повністю перевірити стабільність роботи системи за умов фізичного середовища та реального електроживлення.

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 належному науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: відмінно

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

Звідувач кафедри інженерного програмного забезпечення
професор Ведролюк Леонід Ієтрович

"06" 06 2025 р.

 (підпис)