

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

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр  
Освітній рівень

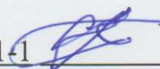
Інформаційна система на базі ультразвукового датчика відстані з п'єзозуметром  
Назва теми


КвРІСТ 2101034.21.01.08 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-21-11  Соломія ПАРЦЕЙ  
Підпис Ініціали, прізвище

Керівник  Олег ВОЙЧУР  
Підпис, дата Ініціали, прізвище

Нормоконтролер  Тетяна КИСІЛЬ  
Підпис, дата Ініціали, прізвище

До захисту допускаю:  
зав. кафедри комп'ютерної  
інженерії та інформаційних  
систем  Ольга ПАВЛОВА  
Підпис Ініціали, прізвище

«\_\_» 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**

Соломії ПАРЦЕЙ

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Інформаційна система на базі ультразвукового датчика відстані з п'єзозумером

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 на базі ультразвукового датчика відстані з п'єзозумером та постановка задачі

Апаратне проектування інфоомаційної системи на базі ультразвукового датчика відстані з п'єзозумером

Апаратна інтеграція компонентів із мікроконтролером ESP32

Реалізація обміну інформацією з хмарним сервісом

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

Архітектура ПЗ проекту

Архітектура ПЗ для інформаційної системи

Апаратне забезпечення проекту

6. Консультанти розділів дипломного проекту (роботи)

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

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 проектування інформаційної системи на базі ультразвукового датчика відстані з п'єзозумером	01.04.2025	виконано
5	Робота над розділом 3 та 4 – підключення всіх компонентів та реалізація обміну з хмарним сервісом	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

Соломія ПАРЦЕЙ  
Ініціали, прізвище

Керівник роботи

Підпис

Олег ВОЙЧУР  
Ініціали, прізвище



## АНОТАЦІЯ

Тема кваліфікаційної роботи: «Інформаційна система на базі ультразвукового датчика відстані з п'єзозумером».

Автор роботи: Соломія Парцей.

Керівник роботи: Войчур Олег Юрійович.

Пояснювальна записка: 58 с., 23 рис., 1 табл., 4 дод., 37 джерел.

Графічна частина: 3 креслення.

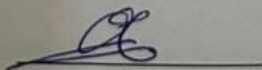
### ІНФОРМАЦІЙНА СИСТЕМА, АПАРАТНЕ ПРОЄКТУВАННЯ, ПІДКЛЮЧЕННЯ МОДУЛІВ, ХМАРА.

Метою дипломної роботи є дослідження умов та особливостей побудови мікроконтролерної інформаційної системи на основі ESP32 з підтримкою моніторингу просторової відстані до об'єктів, а також аналіз механізмів обробки та передачі даних у рамках інформаційного середовища з використанням хмарних технологій для забезпечення адаптивного реагування на зміну ситуацій..

Об'єктом дослідження виступає функціонування інформаційної системи, що включає в себе сенсорні модулі, засоби візуалізації та передавання даних для моніторингу відстані.

Предметом дослідження є методи інтеграції сенсорних пристроїв (зокрема ультразвукового датчика), мікроконтролера ESP32 та периферійних компонентів з алгоритмами збору, обробки, аналізу та передавання інформації до хмарного середовища.

Під час проведення даного дослідження був використаний метод систематичного огляду літератури для вивчення і узагальнення актуальних технічних рішень, методів побудови IoT-систем, а також сучасних підходів до реалізації адаптивного моніторингу з використанням хмарних сервісів.



Підпис студента

30.05.2025

Дата

## ЗМІСТ

<b>ВСТУП</b> .....	4
<b>1 ІНФОРМАЦІЙНА СИСТЕМА НА БАЗІ УЛЬТРАЗВУКОВОГО ДАТЧИКА ВІДСТАНІ З П'ЄЗОЗУМЕРОМ ТА ПОСТАНОВКА ЗАДАЧІ</b> .....	5
1.1 Поняття «Інформаційна система» .....	5
1.2 Структурні і функціональні особливості інформаційної системи на базі ультразвукового датчика відстані з п'єзозумером .....	7
1.3 Аналіз технологічних підходів до вимірювання відстані .....	10
1.4 Постановка задачі.....	11
1.5 Висновки до першого розділу.....	12
<b>2 АПАРАТНЕ ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ НА БАЗІ УЛЬТРАЗВУКОВОГО ДАТЧИКА ВІДСТАНІ З П'ЄЗОЗУМЕРОМ</b> .....	13
2.1 Вибір апаратної платформи .....	13
2.2 Ультразвуковий датчик .....	17
2.2.1 Принцип роботи ультразвукового датчика .....	17
2.2.2 Внутрішня будова та розташування контактів датчика.....	20
2.3 П'єзозумер.....	22
2.3.1 Типи зумерів та роль в інформаційній системі.....	22
2.3.2 Внутрішня структура та принцип роботи п'єзодинаміка .....	23
2.4 LCD-дисплей.....	26
2.5 Потенціометр .....	28
2.6 Світлодіоди та резистори .....	30
2.7 Макетна плата.....	32
2.8 Симулятор Wokwi .....	34
2.9 Висновки до другого розділу .....	34

КвРІСТ 2101034.21.01.08 ПЗ				
Зм.	Арк.	Недокум.	Підпис	Дата
Виконав		Софія ПАРЦЕЙ		
Перевід.		Олег ВОЙЧУР		2.06.21
Н.контр.		Тетяна КИСЛЬ		08.06.21
Затвер.		Ольга ПАВЛОВА		08.06.21
Інформаційна система на базі ультразвукового датчика відстані з п'єзозумером				
		Літера	Аркуш	Аркушів
		у	2	72
ХНУ ІСТ-21-1				

<b>3. АПАРАТНА ІНТЕГРАЦІЯ КОМПОНЕНТІВ ІЗ</b>	
<b>МІКРОКОНТРОЛЕРОМ ESP32 ТА ПЕРЕДАЧА ДАНИХ ДО ХМАРИ .</b>	<b>36</b>
3.1 Підключення ультразвукового датчика .....	36
3.2 Підключення LCD-дисплея .....	37
3.3 Підключення зумера .....	39
3.4 Підключення потенціометра .....	42
3.5 Підключення світлодіодів з резисторами.....	45
3.6 Готова апаратна реалізація та робота системи.....	46
3.7 Загальна архітектура IoT у межах проєкту.....	48
3.8 Взаємодія з Google Sheets як приклад хмарного сервісу .....	51
3.9 Вибір протоколів передачі даних: обґрунтування та реалізація .....	53
3.10 Обмеження хмарного середовища Google Sheets для IoT-систем .....	55
3.11 Висновки до третього розділу.....	56
<b>ВИСНОВКИ .....</b>	<b>57</b>
<b>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....</b>	<b>59</b>
<b>ДОДАТОК А.....</b>	<b>64</b>
<b>ДОДАТОК Б.....</b>	<b>65</b>
<b>ДОДАТОК В.....</b>	<b>66</b>
<b>ДОДАТОК Г.....</b>	<b>67</b>
<b>ДОДАТОК Д.....</b>	<b>68</b>

## ВСТУП

Сучасний світ стрімко трансформується під впливом інформаційних технологій, які проникають у всі сфери людської діяльності, від побутового комфорту до промислових процесів. Вони відіграють вирішальну роль, дозволяючи інтегрувати датчики, виконавчі пристрої та хмарні платформи в єдину екосистему. Такі системи здатні не лише фіксувати параметри, наприклад, відстань до об'єкта, температуру чи рух, але й надавати користувачу зручний доступ до цих даних через локальні дисплеї або віддалені сервіси.

У ході дипломної роботи реалізується інформаційна система на базі ультразвукового датчика відстані. Метою стало створення пристрою, який здатен автономно розпізнавати критичну зміну відстані до об'єкта та відповідно реагувати світловою, звуковою і візуальною індикацією. Ця система не є лише черговою реалізацією популярного "датчика перешкод" – вона розроблена з орієнтацією на варіативне застосування: від безконтактного контролю заповнення ємностей до адаптивної системи попередження наближення в інфраструктурі "розумного дому".

Центральною обчислювальною одиницею виступає мікроконтролер ESP32, побудований на базі 32-розрядного двоядерного процесора Tensilica Xtensa LX6. Його вибір був зумовлений не лише високим рівнем енергоефективності, а й багатим інтерфейсним потенціалом, зокрема наявністю вбудованих модулів Wi-Fi та Bluetooth.

Моделювання системи здійснювалося на базі хмарного емулятора Wokwi, який дозволяє достовірно відтворити поведінку апаратних модулів у віртуальному середовищі без необхідності мати фізичні компоненти. Це дозволило здійснювати гнучке тестування логіки поведінки пристрою на етапах проектування, доопрацювань та перевірки реакції системи на змінні вхідні параметри.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

# 1 ІНФОРМАЦІЙНА СИСТЕМА НА БАЗІ УЛЬТРАЗВУКОВОГО ДАТЧИКА ВІДСТАНІ З П'ЄЗОЗУМЕРОМ ТА ПОСТАНОВКА ЗАДАЧІ

## 1.1 Поняття «Інформаційна система»

Інформаційна система – це не лише сукупність машин чи кабелів. Це впорядкований механізм збирання, обробки, зберігання та передачі даних, який дозволяє людині діяти точніше, швидше й обґрунтованіше. Вперше зародки таких систем можна простежити ще у другій половині ХХ століття, коли великі обчислювальні машини почали застосовуватись для статистики, розрахунків та архівації даних.

Поступово ці системи з громіздких об'єктів, які займали цілі кімнати, трансформувалися у мініатюрні, але набагато потужніші пристрої. Інформаційна система перестала бути чимось винятково офісним або державним – вона стала частиною побуту, охорони, медицини, транспорту й навіть побутових речей, які ми не асоціюємо з «комп'ютером».

Сучасна ера інформаційних систем – це етап інтеграції фізичних об'єктів із цифровим світом, де кожен датчик чи кнопка можуть стати частиною глобальної системи аналізу.

Тобто інформаційна система – це, по суті, зв'язок між тим, хто приймає рішення, і даними, які допомагають це рішення ухвалити. Але це не просто база даних чи комп'ютер з датчиком. Це цілісний механізм, де злагоджено працюють різноманітні пристрої, програми, логіка, реакції й алгоритми.

На рис. 1.1 вказана структура інформаційної системи.



Рисунок. 1.1 – Структура інформаційної системи

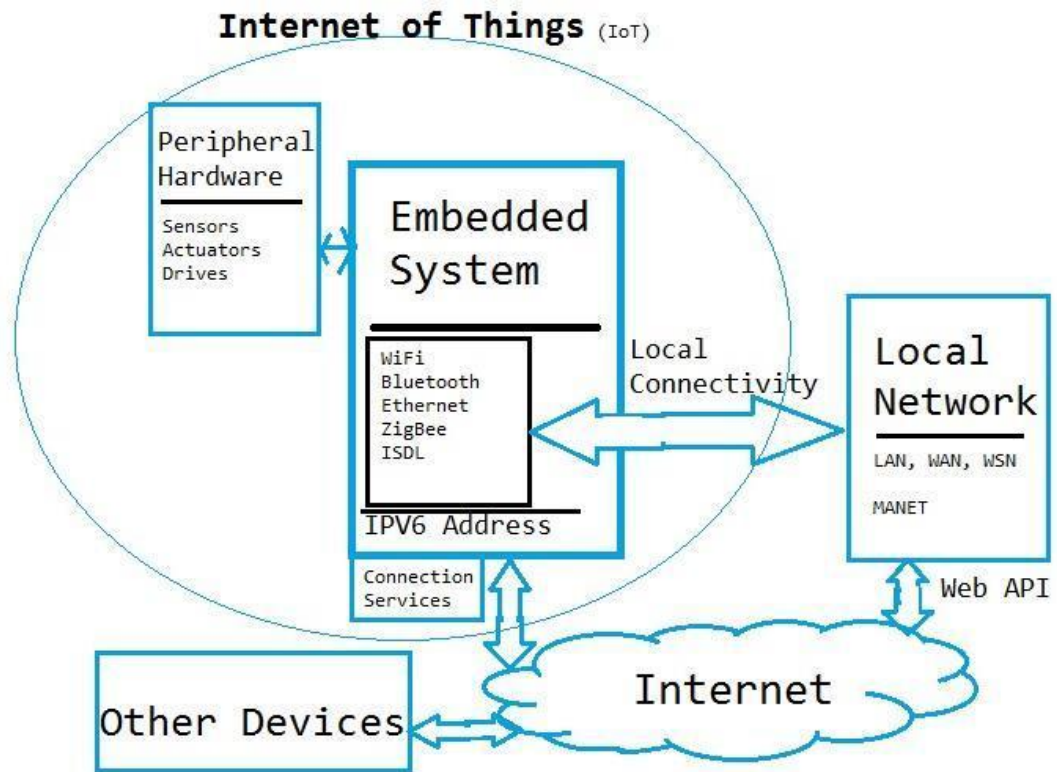
Справжній прорив стався коли до систем додалося нове слово: зв'язок. І отут вже на сцену виходить те, що сьогодні зветься Інтернетом речей, або скорочено - IoT.

Інтернет речей – це не просто "розумні" холодильники чи лампочки. Це підхід, коли фізичні об'єкти - від найпростіших сенсорів до промислових установок – здатні передавати, отримувати й аналізувати дані. Причому не лише "всередині себе", а з іншими подібними пристроями, з хмарними сервісами чи навіть із зовсім іншими системами. Усе це робиться автономно, без постійної участі людини.

У традиційній інформаційній системі пристрої часто збирали дані, але нічого не робили з ними без вказівки. В IoT ж – пристрій не просто фіксує: він розуміє, порівнює, реагує, адаптується. Тобто IoT – це наступний щабель у розвитку інформаційних систем: ті ж принципи, але на порядок ширше, швидше і розумніше.

Основними рисами IoT-систем являється наявність вбудованих сенсорів, які зчитують фізичні параметри (температура, відстань, вологість тощо), обробка даних у мікроконтролерах або в "хмарі", передача інформації бездротовими каналами - Wi-Fi, Bluetooth, LoRa, Zigbee, автоматизована реакція на зміни в середовищі (увімкнення звуку, повідомлення, зміна режиму) [28].

Рисунок 1.2 – класична структура IoT [7]



1.2 Структурні і функціональні особливості інформаційної системи на базі ультразвукового датчика відстані з п'єзозумером

У сучасному технічному середовищі поступово утверджується тренд на створення малогабаритних, адаптивних систем, здатних самостійно реєструвати зміни фізичного середовища й реагувати на них без участі людини. Моя інформаційна система спирається на невеликий, але багатофункціональний мікроконтролер, ультразвуковий сенсор вимірювання відстані, модульний зумер для звукового зворотного зв'язку, та просту візуальну індикацію через світлодіоди й екран.

В основі системи лежить принцип активної дистанційної взаємодії: пристрій самостійно посилає короткі ультразвукові імпульси та, очікуючи на їхнє повернення, обраховує час, за який хвиля повернулась. Це дозволяє точно

визначати відстань до об'єкта навпроти сенсора. Такий підхід, незважаючи на свою технологічну простоту, надає потужні можливості у сфері контролю за простором.

Мікропроцесорна частина побудована на ESP32 – надзвичайно функціональному чипі, що поєднує в собі обчислювальні ядра та бездротову комунікацію (Wi-Fi, Bluetooth). Завдяки цьому пристрій має потенціал бути для подальшого аналізу.

До ESP32 приєднується ультразвуковий модуль HC-SR04, який є чутливим сенсором, що посилає підключеним до хмари або локальної мережі, передаючи дані на сервер або онлайн-таблицю короткі звукові імпульси й чекає на їх повернення після відбиття від об'єкта. Вимірювання часу між посиланням та отриманням імпульсу дає змогу точно обчислити відстань до перешкоди.

Окремо слід зазначити RGB-світлодіод або набір із трьох світлодіодів різного кольору. Це не просто елемент індикації, а спосіб комунікації між машиною та користувачем. Кольори відображають рівень загрози чи наближення об'єкта, даючи миттєвий візуальний сигнал.

П'єзозумер – звуковий компонент, який створює тональний сигнал. Частота або ритм сигналу може змінюватися залежно від ситуації, що дозволяє вловити зміни навіть без візуального контакту з пристроєм.

Не менш важливим структурним елементом є потенціометр, який виступає налаштовувальним вузлом. Він дозволяє вручну змінювати порогові значення відстані, тим самим адаптуючи чутливість системи до середовища: у тісних просторах потрібно менше реагування, а на відкритих територіях – більше.

LCD-дисплей на базі I2C-шини – ще один важливий компонент. Він не просто дублює числові показники – він фіксує стан системи на кожному етапі: від спокою до тривоги. Текстовий інтерфейс на дисплеї дозволяє не лише бачити відстань, а й розуміти причину реакції пристрою.

Усі ці модулі пов'язані між собою через провідники й логічну прошивку, створену мовою Arduino C++, що дозволяє точну синхронізацію між вимірюванням, аналізом та відповіддю.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

Функціональність системи не зводиться до простого «якщо близько - подай звук». Основна мета – адаптивне реагування на зміни в оточенні та надання користувачу зрозумілого, багаторівневого сигналу.

Розглянемо, які саме дії виконує система в режимі роботи й чому це важливо.

У першу чергу це первинний зчитувальний цикл. Система починає з того, що ультразвуковий сенсор надсилає сигнал у навколишнє середовище. Цей сигнал непомітний для людського вуха, але він "відбивається" від будь-якого об'єкта, що стоїть на шляху. Контролер миттєво фіксує час, за який цей імпульс повертається, та розраховує точну відстань.

Наступне це аналіз і класифікація ситуації. Залежно від того, яку відстань зафіксовано, система класифікує ситуацію. Наприклад, якщо об'єкт далеко – система нічого не робить або вмикає зелений індикатор. При середній відстані – жовтий індикатор та попереджувальний звуковий сигнал. Якщо об'єкт критично близько – червоний світлодіод і тривожний, частий звук зумера.

Також система адаптується до середовища через потенціометр, дозволяючи користувачеві змінювати порогові відстані. Наприклад, у невеликому приміщенні налаштування може бути зроблено так, щоб не реагувати на стіни, а лише на рухомі об'єкти.

Дуже важливим є те, що система надає зворотний зв'язок з користувачем та можливість логування. Через індикатори, зумер і дисплей система постійно «пояснює», що саме відбувається: не просто вимірює, а й розповідає про стан простору. Це важливо, особливо у шумних середовищах або для людей з вадами зору чи слуху. У повноцінному використанні система може передавати дані у хмарне середовище (наприклад, Google Sheets), фіксуючи кожен вимір з точною датою й часом. Це відкриває простір для аналітики: можна виявити закономірності або критичні точки часу, коли виникала небезпека.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

### 1.3 Аналіз технологічних підходів до вимірювання відстані

У межах розвитку вбудованих інформаційних систем ключову роль відіграють методи просторової орієнтації та відстеження об'єктів. Вимірювання відстані – це одна з найзатребуваніших задач як у побутових, так і в промислових рішеннях. Залежно від вимог до точності, швидкості обробки сигналу, умов експлуатації та вартості, застосовуються різні типи сенсорів: інфрачервоні, лазерні, оптичні, радарні та ультразвукові.

Ультразвукові датчики, зокрема HC-SR04, стали особливо популярними завдяки поєднанню простоти, доступності та достатньої точності для більшості прикладних задач. Вони ідеально підходять для інтеграції з мікроконтролерами на зразок ESP32, оскільки дозволяють у реальному часі отримувати інформацію про зміну положення об'єктів без потреби в складній обробці сигналу чи дорогих компонентах.

Системи, побудовані на принципах вимірювання відстані, мають широкий спектр застосувань. Вони здатні стежити за простором у складських приміщеннях, коригувати рух роботизованих платформ, використовуються в «розумних» будинках, попереджають про наближення перешкод для людей із вадами зору або виконують контроль доступу до небезпечних зон. Саме тому створення подібних рішень на базі недорогих компонентів, доступних кожному ентузіасту або студенту, не лише актуальне, а й соціально важливе.

Актуальність теми також полягає в її інтеграції з глобальними тенденціями – розвитком Інтернету речей, збиранням великих обсягів даних (Big Data), екологічного моніторингу та інклюзивних технологій. У контексті концепції «розумного простору», де середовище адаптується до потреб людини, навіть такий на перший погляд простий проєкт може стати частиною ширшої екосистеми: передавати дані в хмару, аналізувати поведінкові патерни, реагувати на типові загрози.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 10
Зм.	Арк.	№ докум.	Підпис	Дата		

Крім цього, застосування мікроконтролера ESP32 надає розробці технічної гнучкості: наявність Wi-Fi, Bluetooth, великої кількості вхідних/вихідних портів дозволяє масштабувати систему без повної перебудови. У свою чергу, ультразвукові сенсори, зумери, світлодіоди та дисплеї забезпечують зрозумілий інтерфейс для користувача, що особливо важливо у критичних ситуаціях, коли швидкість реакції є визначальною.

#### 1.4 Постановка задачі

Метою даного проєкту є створення системи моніторингу відстані на базі мікроконтролера ESP32 із виведенням інформації на LCD-дисплей, локальною світловою та звуковою сигналізацією, а також передаванням даних у хмарне середовище для подальшого аналізу. Розроблена система має не лише визначати наближення об'єктів на різні критичні відстані, а й забезпечувати користувача зрозумілою індикацією поточного стану у реальному часі.

Реалізація проєкту поділяється на декілька етапів. Спочатку створюється віртуальна схема в Wokwi з підключенням усіх компонентів до ESP32. Далі розробляється код для ініціалізації компонентів, вимірювання відстані, керування периферійними пристроями та передачі даних. Наступним кроком є налаштування мережевої інтеграції шляхом створення Google Apps Script і тестування HTTPS-запитів. Завершальний етап симуляції, тестування та оптимізації системи для забезпечення стабільної роботи всіх функцій. Використання Wokwi значно спрощує ці етапи, після того як вживає потребу у фізичних компонентах і дозволяє швидко вносити зміни.

Використання Wokwi дає низку переваг для реалізації проєкту. По-перше, наявність потреб у фізичному забезпеченні знижує витрати та робить об'єкт доступним для моделювання в будь-яких умовах. По-друге, графічний інтерфейс і вбудований компілятор прискорюють прототипування та налагодження. По-третє, емуляція Wi-Fi дозволяє тестувати мережеву інтеграцію без реального

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 11
Зм.	Арк.	№ докум.	Підпис	Дата		

підключення до мережі. Однак Wokwi має обмеження, такі як спрощена модель фізичних характеристик (наприклад, відсутність урахування шумів для HC-SR04) і наявні обмеження на складність схеми в безкоштовній версії, що слід слідувати під час розробки.

У висновку методи та засоби реалізації проекту базуються на віртуальному моделюванні в Wokwi, програмуванні ESP32 із використанням Arduino-бібліотек, інтеграції з Google Sheets через Google Apps Script та остаточному тестуванні в симуляційному середовищі.

### 1.5 Висновки до першого розділу

У першому розділі розглянули поняття, що таке інформаційна система та IoT. Було визначено актуальність створення інформаційної системи для вимірювання відстані на базі ультразвукового сенсора HC-SR04. Проведено аналіз технологічних підходів до визначення дистанції, виокремлено переваги ультразвукових рішень, з'ясовано області їх ефективного застосування. Сформульовано мету, завдання та технічні вимоги до майбутньої системи.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 12
Зм.	Арк.	№ докум.	Підпис	Дата		

## 2 АПАРАТНЕ ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ НА БАЗІ УЛЬТРАЗВУКОВОГО ДАТЧИКА ВІДСТАНІ З П'ЄЗОЗУМЕРОМ

### 2.1 Вибір апаратної платформи

Проєктування будь-якої цифрової системи, яка взаємодіє з реальним середовищем, неможливе без чіткого визначення апаратної основи. Саме апаратна платформа виступає своєрідним «скелетом», на якому будується логіка, обчислення, комунікація й інтеграція з периферійними компонентами. Від цієї складової залежить не лише функціональність майбутньої системи, а й простота реалізації, можливості розширення, енергоефективність, стабільність і навіть економічна доцільність проєкту. Сучасний ринок пропонує широкий спектр рішень, однак не кожне з них здатне відповідати всім вимогам, що стоять перед IoT-системою.

Серед найвідоміших представників цього класу можна згадати Arduino UNO, Arduino Mega, ESP8266, Raspberry Pi та ESP32. Кожна з цих платформ має свої сильні та слабкі сторони, а також специфіку застосування

Arduino UNO – це класична платформа для початківців і невеликих проєктів. Її просто програмувати, вона надійна, добре документована, проте її обчислювальна потужність, об'єм пам'яті та функціональність вкрай обмежені. В UNO відсутня підтримка Wi-Fi чи Bluetooth, а це виключає її зі списку кандидатів для реалізації систем, пов'язаних з обміном даними через інтернет.

Arduino Mega вирізняється збільшеною кількістю входів/виходів та більшим об'ємом пам'яті у порівнянні з UNO. Це робить її придатною для проєктів із великою кількістю сенсорів або елементів керування. Проте й вона не має вбудованих бездротових модулів, а для інтеграції з мережею потребує додаткових плат розширення, що збільшує габарити й ускладнює збірку [10].

ESP8266 – перший доступний мікроконтролер з вбудованим Wi-Fi, який став революційним у сфері IoT. Він недорогий, підтримує базову обробку даних, ідеально підходить для простих систем дистанційного моніторингу. Але його

однойдерна архітектура, обмежена кількість GPIO і менша продуктивність – значні мінуси для задач, де паралельно потрібно обробляти сенсорні сигнали, керувати дисплеєм і здійснювати передачу в хмару.

Raspberry Pi – це вже повноцінний мікрокомп'ютер, який працює під керуванням Linux. Його обчислювальні можливості значно перевищують усе вище перераховане. Але за це доводиться платити значно вищою вартістю, більшим енергоспоживанням, складністю налаштування та надмірністю можливостей для систем, де основна функція - зчитування й передавання нескладних даних з датчиків [10].

І врешті – ESP32 (рис. 2.1), що стала золотою серединою між мікроконтролером і мінімальним комп'ютером. Її архітектура заснована на двоядерному процесорі, що дозволяє виконувати кілька задач одночасно. Вбудовані Wi-Fi та Bluetooth-модулі дають змогу реалізувати бездротовий обмін без жодних додаткових пристроїв. Розширений набір портів введення/виведення, підтримка аналогових і цифрових сигналів, можливість роботи з I2C, UART, PWM, SPI – усе це робить ESP32 універсальним інструментом. При цьому її ціна залишається надзвичайно привабливою, а розміри дозволяють інтегрувати її навіть у мініатюрні корпуси [30].

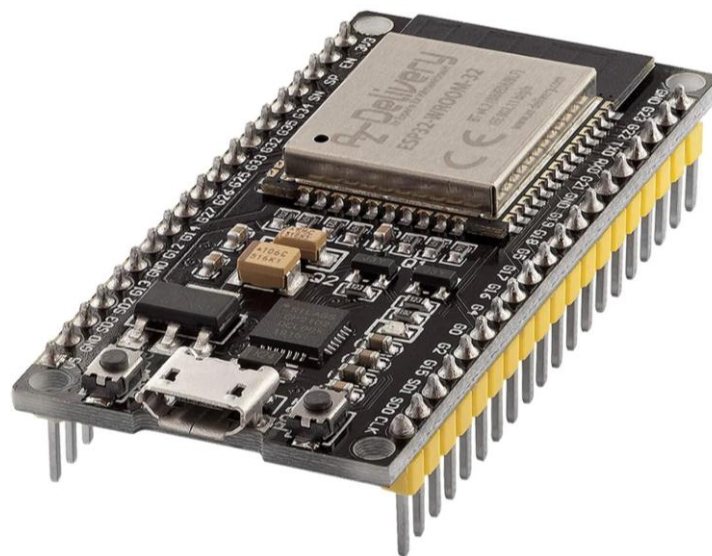


Рисунок 2.1 – Мікроконтролер ESP32 [88]

Нижче наведена таблиця 2.1, що демонструє переваги ESP32 у межах реалізації даної IoT-системи.

Таблиця 2. 1 – Переваги ESP32 у реалізації IoT-системи

Критерій	Вимоги проєкту	Можливості ESP32	Перевага для даної системи
Обробка кількох процесів	Одночасна робота з датчиком, зумером, дисплеєм, Wi-Fi, світлодіодами	Двоядерний процесор (Xtensa LX6), до 240 МГц	Дає змогу розділити логіку між ядрами: одне — обробка сигналів, інше — передача даних
Комунікація з хмарою	Надсилання вимірювань до Google Sheets через Інтернет	Вбудований модуль Wi-Fi (802.11 b/g/n), підтримка HTTPS	Спрощує прямий вихід в Інтернет без додаткових модулів
Локальна індикація	Підключення LCD-дисплея, RGB-світлодіодів, звукового сповіщення	Підтримка I2C, PWM, цифрових/аналогових виходів	Забезпечує просте підключення дисплея, зумера, світлодіодів
Гнучкість підключення	Велика кількість компонентів (HC-SR04, LCD, LED, п'єзозумер, потенціометр, серво)	До 34 GPIO-портів, підтримка ADC, I2C, SPI, UART, PWM	Дозволяє одночасне підключення всіх необхідних периферій

Кінець таблиці 2.1

Передача даних у реальному часі	Оновлення інформації без затримок	Підтримка асинхронного Wi-Fi, апаратна серіалізація	Забезпечує стабільну та швидку передачу без переривання роботи інших модулів
Низьке енергоспоживання	Можливість автономної роботи у разі потреби	Режими енергозбереження, контроль живлення периферії	Зменшує загальне енергоспоживання системи, дозволяє працювати від батарей
Фізичні розміри	Компактність схеми	Мала плата, зручно інтегрується у прототип	Ідеально підходить для проектів з обмеженим простором, не ускладнює макетування
Ціна	Обмежений бюджет	Дуже доступна при широкому функціоналі	Ідеальний вибір для бюджетного, але функціонально насиченого IoT-рішення
Сумісність з Arduino IDE	Простота розробки коду	Повна підтримка Arduino-бібліотек, велика спільнота	Дає змогу швидко створити та налагодити код навіть без глибокого знання системного програмування

## 2.2 Ультразвуковий датчик

### 2.2.1 Принцип роботи ультразвукового датчика

В основі побудови розглянутої інформаційної системи лежить можливість безконтактного вимірювання відстані до об'єкта. Саме цю функцію виконує ультразвуковий датчик, який, завдяки своїй точності, надійності та низькій вартості, став одним із найпоширеніших засобів у задачах дистанційного вимірювання в електроніці та системах автоматизації.

У рамках реалізації проєкту було використано датчик HC-SR04 (рис 2.2), який у середовищі Wokwi представлений у вигляді віртуального компонента з максимально наближеними до реальності характеристиками. Варто, однак, зазначити, що симуляційна модель не завжди точно відтворює всі фізичні параметри реального пристрою, зокрема вплив шумів, відбиттів або завад, які виникають у фізичному середовищі.



Рисунок 2.2 – Ультразвуковий датчик HC-SR04 [22]

Датчик складається з двох основних елементів – передавача (трансдюсера), що генерує звукову хвилю, і приймача, який фіксує її відбиття від перешкоди. Обидва компоненти розміщені в одному корпусі, а робота модуля здійснюється за

допомогою двох цифрових контактів: TRIG (ініціює імпульс) та ECHO (повертає тривалість відбитого сигналу).

Розглянемо основні технічні характеристики:

- дальність вимірювання: від 2 см до 400 см;
- точність:  $\pm 3$  мм;
- робоча напруга: 5 В;
- споживання струму:  $\sim 15$  мА;
- частота ультразвукових хвиль: 40 кГц;
- кут огляду: приблизно  $15^\circ$ .

Датчик досить стабільно працює у внутрішньому середовищі за умови відсутності сильних завад або вібрацій. Його популярність пояснюється не лише вартістю, а й універсальністю: його легко інтегрувати до будь-якої мікроконтролерної платформи, зокрема Arduino, ESP8266 чи ESP32.

Фізичний принцип, на якому базується робота HC-SR04, добре знайомий ще з часів природознавства – це ехолокація, подібна до тієї, яку використовують кажани. Суть у тому, що пристрій генерує короткий ультразвуковий імпульс, який поширюється в повітрі, відбивається від найближчого об'єкта і повертається до датчика. Час, за який відбувається цей цикл, фіксується і використовується для розрахунку відстані [12].

Алгоритм функціонування ультразвукового датчика HC-SR04 (рис. 2.3) ґрунтується на принципі передачі і прийому звукового імпульсу високої частоти, а також на точному вимірюванні часу, що минає між його надсиланням і поверненням. Як тільки на керуючий вивід TRIG подається електричний імпульс тривалістю близько 10 мікросекунд, модуль активується і генерує звукову хвилю з частотою 40 кілогерц, яка починає поширюватися в повітрі з характерною швидкістю звуку. У випадку, якщо на шляху хвилі знаходиться об'єкт, вона відбивається від його поверхні та повертається назад до модуля, де сприймається приймачем. У момент фіксації повернення сигналу на вихідному піні ECHO формується логічний сигнал високого рівня, тривалість якого відповідає часу, що

минує від моменту відправлення імпульсу до його отримання. Власне тривалість цього сигналу і є ключовим параметром для обчислення фактичної відстані до перешкоди. Отримане значення часу (в мікросекундах) підставляється у класичну формулу, що враховує швидкість звуку у повітрі при нормальних умовах (приблизно 343 м/с або 0.0343 см/мкс) [19].

Поділ на два необхідний, оскільки сигнал проходить шлях до об'єкта і назад, а нас цікавить лише одна сторона цієї подорожі. Таким чином, завдяки поєднанню простого апаратного керування та математично обґрунтованої обробки результату, HC-SR04 дозволяє швидко й ефективно отримувати інформацію про розташування об'єктів у межах кількох метрів від пристрою.

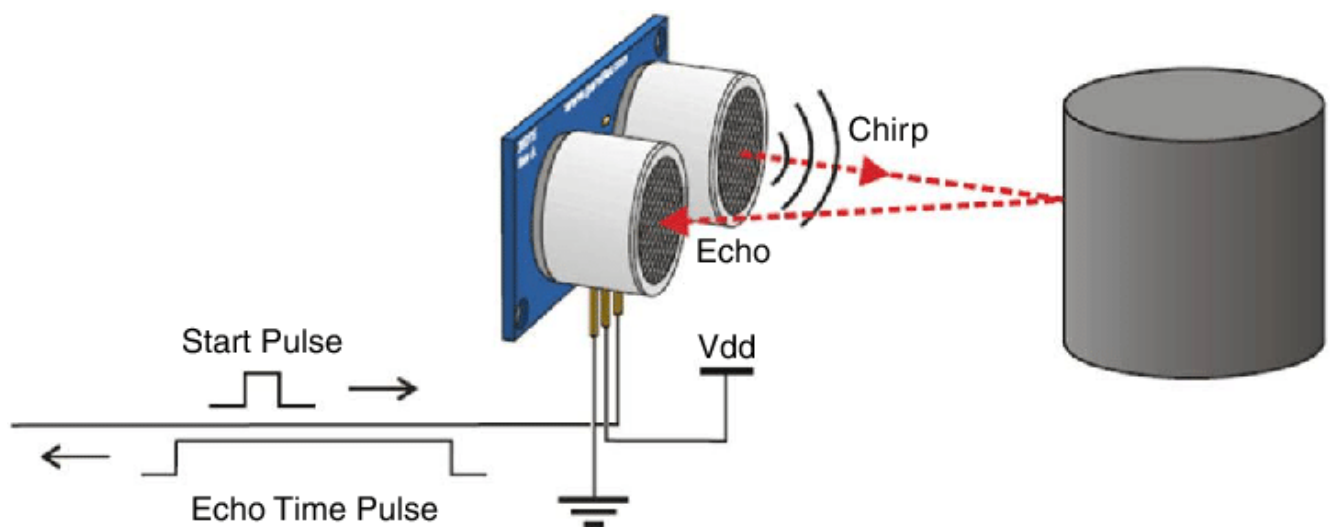


Рисунок 2.3 – Принцип роботи ультразвукового датчика [22]

Ця схема дозволяє з достатньо високою точністю отримувати дані про розташування об'єкта у просторі. Проте в реальних умовах точність може знижуватись через такі фактори, як:

- надто малі або м'які об'єкти, що слабо відбивають сигнал;
- перешкоди між датчиком і об'єктом;
- нахил поверхні відносно напрямку хвилі.

## 2.2.2 Внутрішня будова та розташування контактів датчика

Ультразвуковий датчик HC-SR04, який уже давно закріпився як надійний інструмент у сфері вимірювання відстані, є технічно довершеним елементом, що вражає своєю компактністю й водночас виваженою складністю внутрішньої архітектури. На перший погляд пристрій виглядає просто: невелика прямокутна друкована плата, на якій розміщено дві великі циліндричні конструкції, схожі на очі – передавач і приймач. Саме ці "вічка" є зовнішніми п'єзоелементами, що виконують функцію генерації та фіксації акустичних хвиль, а також створюють характерний вигляд модуля, який часто порівнюють з обличчям роботизованої істоти.

Проте вся справжня інженерна витонченість прихована у внутрішній структурі. Під зовнішньою оболонкою модуля міститься генератор імпульсів, який формує сигнал для передачі, підсилювач прийнятого сигналу, компаратор, що перетворює аналоговий відгук у чіткий цифровий імпульс, а також мікросхема керування (найчастіше на базі NE555 або її аналогів), що координує взаємодію між усіма внутрішніми вузлами. Крім того, варто згадати про додаткові пасивні елементи – резистори, конденсатори та діоди, які забезпечують стабілізацію та захист ланцюгів при роботі на різних рівнях навантаження. Усе це акуратно впаяне в двосторонню друковану плату, де мікросхеми займають нижній бік, а вся оптика та інтерфейс взаємодії – верхній [24].

Розпіновка модуля HC-SR04 (рис. 2.4) – ще один важливий аспект, без якого його інтеграція у будь-яку систему була б неможливою. Модуль має чотири пін-контакти, які розташовані в один ряд і підписані безпосередньо на платі: VCC, TRIG, ECHO, GND. Перший з них – VCC – відповідає за подачу живлення. Стандартна напруга для коректної роботи – 5 В, хоча деякі версії здатні функціонувати і від 3.3 В, проте з меншою стабільністю. Далі йде контакт TRIG – керуючий вхід, через який мікроконтролер подає імпульс активації. Третій пін – ECHO – є вихідним, і саме на ньому з'являється сигнал, який вказує на час

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 20
Зм.	Арк.	№ докум.	Підпис	Дата		

проходження хвилі. І нарешті, GND – загальний мінус, який підключається до землі живлення всієї системи. Таке інтерфейсне рішення забезпечує легку інтеграцію з платформами на кшталт Arduino, ESP32, STM32 тощо.

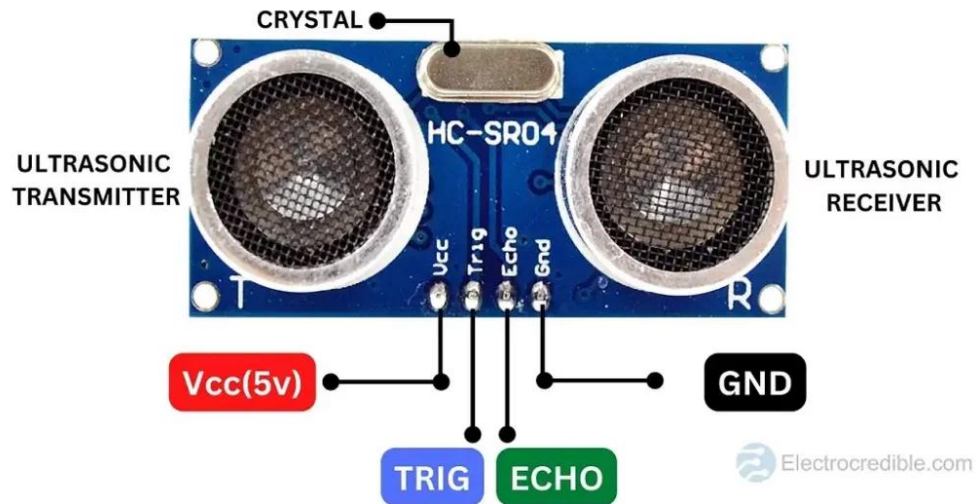


Рисунок 2.4 – Розпіновка модуля HC-SR04 [2]

Крім технічної частини, доцільно відзначити фізичні параметри модуля: середні габарити складають приблизно 45 мм у довжину, 20 мм у ширину й близько 15 мм у висоту. Вага не перевищує декількох грамів, а корпус не має жодного захисту від вологи, що робить його придатним лише для внутрішнього використання або потребує додаткової герметизації.

П'єзокристали, які використовуються в модулях HC-SR04, виготовлені з керамічного матеріалу (найчастіше титанату барію або цирконату титанату свинцю), здатного деформуватися при подачі електричного сигналу. Саме завдяки цьому ефекту досягається випромінювання хвиль, а в режимі прийому – зворотне перетворення в електричний сигнал [25].

## 2. 3 П'єзозумер

### 2.3.1 Типи зумерів та роль в інформаційній системі

П'єзозумер, або як його ще іноді називають – п'єзоелектричний сигналізатор, є одним із найпоширеніших елементів звукової індикації у мікроконтролерних та електронних системах. Його компактність, надійність, простота підключення і вражаюча енергоефективність зробили цей компонент майже незамінним у системах оповіщення, датчиках, будильниках та автоматизованих конструкціях.

Загалом, п'єзозумери поділяються на дві великі категорії: активні та пасивні.

Активний п'єзозумер – це самодостатній пристрій, який вже має вбудований генератор сигналу. Для його функціонування достатньо подати постійний струм або напругу – решту за нього зробить електроніка всередині корпусу. Активні моделі завжди відтворюють один і той самий тон, закладений у мікросхемі.

Пасивний п'єзозумер це більш "ручна" версія, яка не має внутрішнього генератора. Тут вся ініціатива створення частоти лежить на плечах мікроконтролера: потрібно формувати сигнал із необхідною частотою для створення бажаного тону. Саме пасивні варіанти використовуються найчастіше в інтерактивних системах, де потрібна змінна частота, наприклад, різні тони у відповідь на зміну відстані [4].

У нашій системі використовується пасивний п'єзозумер (рис. 2.5), що підключається до одного з цифрових портів ESP32. Такий вибір зумовлений потребою в динамічному керуванні частотою звуку – система реагує на зміну відстані до об'єкта і варіює тональні сигнали відповідно до ступеня небезпеки. Так, при наближенні об'єкта до критичних меж відстані зумер створює частіші сигнали, а на безпечній дистанції – мовчить або працює в низькочастотному режимі.

П'єзозумер виконує роль негайної звукової індикації. Це надзвичайно ефективний спосіб передати користувачу сигнал тривоги або попередження без потреби у візуальному контакті з дисплеєм. У поєднанні зі світлодіодами та

виведенням інформації на екран LCD, він забезпечує комплексну багато-каналну взаємодію з користувачем, яка охоплює як візуальний, так і звуковий канали.



Рисунок 2.5 – П'єзозумер [18]

Цінність п'єзозумера у подібній IoT-системі проявляється не лише у його акустичних властивостях, а і в можливості забезпечити реакцію в реальному часі на зміну вхідних даних з ультразвукового сенсора. Оскільки дані передаються ще й у хмарне середовище, то п'єзозумер виступає не лише як оповіщувач, а як первинний бар'єр безпеки, сигналізуючи раніше, ніж спрацює аналітична частина системи.

### 2.3.2 Внутрішня структура та принцип роботи п'єзодинаміка

Розглянемо конструктивну будову п'єзозумера (рис. 2.6) , в якій видно, що всередині простого на вигляд циліндричного корпусу приховується досить продумана і технологічна структура. Всі основні складові гармонійно взаємодіють між собою, забезпечуючи перетворення електричних імпульсів у звукові хвилі з мінімальними втратами.

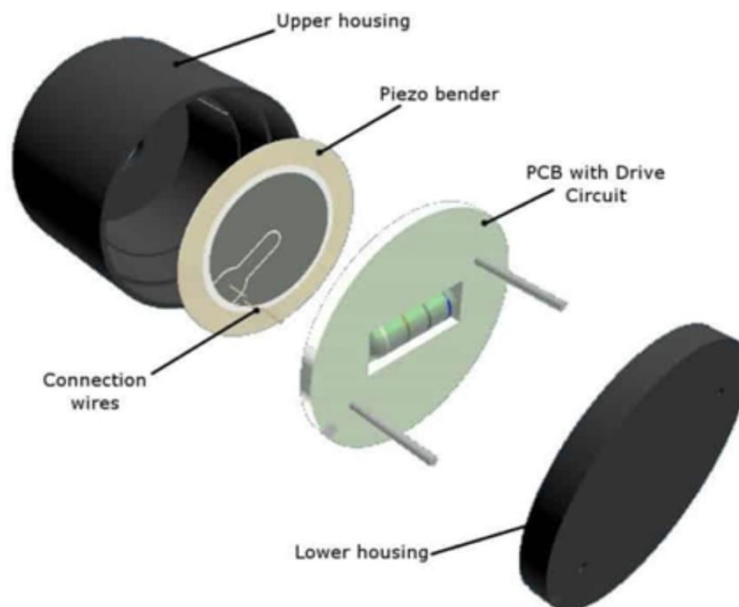


Рисунок 2.6 – Внутрішня структура п'єзодинаміка [21]

Upper housing (верхня кришка) – це зовнішня частина корпусу, що виконує одночасно механічну і акустичну функцію. З одного боку, вона захищає внутрішні компоненти від пилу, вологи та механічних пошкоджень, а з іншого – формує акустичну камеру, яка підсилює і направляє звук назовні. Часто ця частина має невеликий отвір або сітку, через який виходить звукова хвиля.

Piezo bender (п'єзогнучкий диск) – це серце зумера – тонкий диск із п'єзокерамічного матеріалу, закріплений на металевій пластині. При подачі змінної напруги диск періодично згинається і вібрує, передаючи ці коливання мембрані. Саме ці мікроскопічні рухи стають джерелом акустичного сигналу. Диск працює завдяки зворотному п'єзоєфекту – здатності матеріалу змінювати форму під впливом електричного поля [4].

Connection wires (з'єднувальні дроти) – тонкі провідники, що передають електричні імпульси від мікроконтролера або драйвера до п'єзодиска. Вони мають бути достатньо гнучкими та стійкими до вібрацій, адже весь механізм працює в умовах мікроколивань.

PCB with Drive Circuit (плата з драйвером) – друкована плата в активних п'єзозумерах, яка містить генератор, що формує коливання заданої частоти. Тут

розташовані мікросхеми, конденсатори, резистори, які керують п'єзодиском і задають його ритм. У пасивних зумерах цей елемент може бути відсутнім або виконувати роль простої контактної площадки.

Lower housing (нижня кришка) – нижня частина корпусу, що разом з верхньою утворює герметичну оболонку. Вона захищає плату та механізм від зовнішнього впливу. У деяких моделях тут є монтажні отвори для закріплення зумера на платі або панелі пристрою.

Ця модульна структура дозволяє досягти максимальної ефективності при мініатюрних розмірах. Завдяки подібній архітектурі, п'єзозумери можуть працювати стабільно в широкому діапазоні температур, не споживати багато енергії та забезпечувати гучність до 85–100 дБ. Візуальна схема демонструє, наскільки злагоджено взаємодіють електричні й механічні компоненти, щоб створити простий, але надзвичайно корисний звуковий сигнал.

Фізичний процес створення звуку п'єзозумером базується на зворотному п'єзоефекті: при подачі змінної напруги на п'єзодиск, його поверхня починає ритмічно деформуватися. Оскільки він кріпиться до металевої мембрани, остання теж починає вібрувати. Частота цих вібрацій відповідає частоті прикладеної напруги, а амплітуда – її силі. Таким чином, за допомогою прямокутного сигналу з частотою, наприклад, 2000 Гц, ми отримаємо звукову хвилю на 2 кГц [4].

У пасивному п'єзозумері необхідно генерувати такі коливання за допомогою контролера, наприклад, функцією `tone()` в Arduino. У той час активний вже має вбудований генератор і сам підлаштовує частоту.

У типовій схемі підключення пасивний зумер має два контакти – один до цифрового виходу мікроконтролера (GPIO), другий – до GND. За потреби можна використати транзистор як ключ, якщо планується керування зумером від потужнішого джерела або за допомогою ШІМ-сигналу. В активних п'єзозумерах часто передбачено внутрішнє обмеження струму, тому вони можуть підключатися напряму до живлення.

## 2.4 LCD-дисплей

У світі візуалізації даних серед мікроконтролерних систем одне з чільних місць посідають LCD-дисплеї, які перетворилися зі звичайних індикаторів у повноцінні вивідні блоки. Попри невелику фізичну складність, за кожним таким модулем криється вивірена електронна структура, багаторівнева логіка відображення та принцип дії, що базується на фізичних властивостях рідких кристалів.

LCD-модуль складається з кількох взаємопов'язаних компонентів, кожен з яких виконує точково визначену функцію. Основою є дві прозорі пластини, між якими розташовується шар рідкокристалічної речовини. Пластини вкриті прозорими електродами (зазвичай з оксиду індію та олова), які формують сітку комірок – пікселів. У випадку стандартного символного дисплея 16×2 маємо матрицю з 32 символів, кожен з яких формується у сітці 5×8 або 5×10 точок.

На верхній частині дисплея розташовується поляризаційна плівка, яка дозволяє світлу проходити в одному напрямку. Рідкі кристали у звичайному (неактивному) стані повертають площину поляризації світла, дозволяючи зображенню стати видимим або навпаки – непрозорим залежно від прикладеної напруги. Світло в LCD-дисплеях не генерується самостійно, а лише блокується чи пропускається - саме тому у багатьох моделях присутнє підсвічування (наприклад, світлодіодне), яке виводить інформацію на передній план.

Усім цим керує контролер, найчастіше – HD44780 або його сучасні аналоги. Він формує сигнали, що подаються на електроди, та зберігає набір символів у власній ROM-пам'яті, дозволяючи мікроконтролеру передавати лише інструкції, а не малювати кожен піксель окремо.

Коли на електроди подається електричний сигнал, рідкі кристали змінюють орієнтацію, втрачаючи здатність повертати площину поляризованого світла. Таким чином у відповідному місці матриці з'являється темна точка – піксель, або частина символу. Кожна літера, цифра або символ – це набір таких керованих точок.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

Передача даних від мікроконтролера до LCD зазвичай відбувається паралельно – через 4 або 8-бітний інтерфейс. Існують також модулі з інтерфейсом I<sup>2</sup>C, що значно спрощують з'єднання: замість великої кількості проводів використовується лише дві лінії – SDA і SCL. Це особливо корисно, коли на платі обмежена кількість пінів або потрібно підключити багато пристроїв.

Окремо варто зупинитися на віртуальному еквіваленті дисплея, що використовується в середовищі Wokwi (рис. 2.7). Для симуляційної моделі обрано LCD1602 із модулем I<sup>2</sup>C, який повністю відповідає типовому реальному модулю, що базується на контролері HD44780 у парі з PCF8574. У Wokwi модель поводить себе практично ідентично до фізичного пристрою: підтримується той самий обсяг символів (2 рядки по 16 символів), логіка керування за допомогою бібліотеки LiquidCrystal\_I2C.

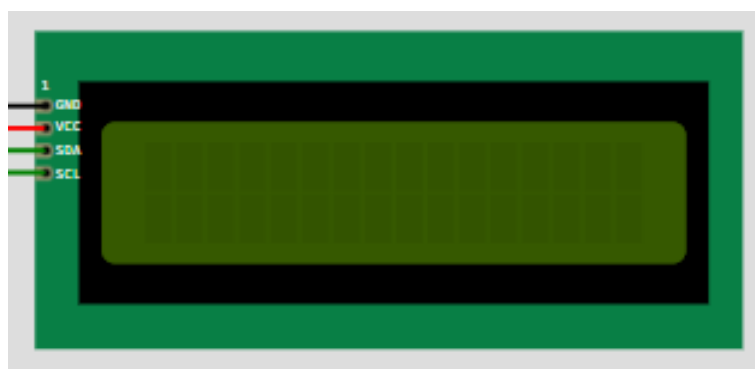


Рисунок 2.7 – LCD-дисплей

Попри цифрову природу, модель у Wokwi відтворює ключові особливості LCD1602 досить точно, включаючи необхідні затримки після ініціалізації, обробку команд позиціонування курсору, очищення екрана та підтримку регістра даних. Це дозволяє на етапі проєктування повністю протестувати поведінку інтерфейсу відображення та своєчасно виявити помилки, що могли б виникнути на реальному пристрої, зокрема в синхронізації даних або неправильному форматуванні.

## 2.5 Потенціометр

У структурі апаратного забезпечення запропонованої інформаційної системи, побудованої на базі ультразвукового датчика, окреме місце займає такий на перший погляд простий, але надзвичайно корисний елемент як потенціометр (рис. 2.8). Це не просто крутилка, що регулює яскравість чи гучність. У контексті автоматизованих систем, потенціометр виступає як важливий інтерфейс взаємодії між користувачем і мікроконтролерним середовищем, надаючи змогу динамічно налаштовувати параметри роботи пристрою без необхідності перепрошивки або змін у програмному коді.

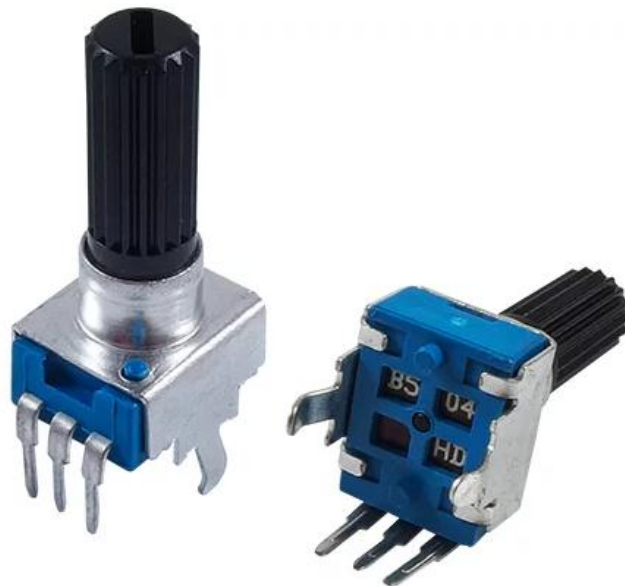


Рисунок 2.8 – Потенціометр [20]

Фізично потенціометр складається з кількох ключових елементів. Розглянемо їх.

Резистивна доріжка – головний провідний шар, зазвичай виготовлений з графіту, металевого сплаву або вуглецевого композиту, по якому ковзає рухомий контакт.

Рухомий контакт (щітка) – механічно з'єднаний із ручкою, дозволяє змінювати точку з'єднання на резистивній доріжці.

Три виводи (пінів):

- перший і третій – кінці резистивного шару (зазвичай GND і VCC);
- середній – вивід із рухомого контакту (змінний сигнал або «wiper»).

Коли користувач обертає ручку потенціометра, змінюється співвідношення опорів між виводами, що відповідно модулює вихідну напругу. Таким чином, потенціометр надає аналоговий сигнал, який може бути зчитаний мікроконтролером через ADC (аналогово-цифровий перетворювач).

Робота потенціометра базується на лінійному (або логарифмічному) розподілі опору вздовж резистивної доріжки. Під час обертання повзунка відбувається зміна положення точки дотику на цій доріжці, що відповідно змінює напругу на середньому виводі. Ця напруга може коливатись у межах від 0 В до напруги живлення (зазвичай 3.3 В або 5 В), утворюючи діапазон аналогових значень, що легко зчитуються мікроконтролером. В ESP32, наприклад, потенціометр підключається до одного з ADC-портів (наприклад, GPIO34), що дозволяє інтерпретувати положення ручки як числове значення в межах від 0 до 4095 (12-бітове перетворення). [1]

У запропонованому пристрої потенціометр несе стратегічну функцію налаштування порогів реагування системи на зміну відстані до об'єкта. Інакше кажучи, користувач отримує змогу у режимі реального часу, без жодних програмних змін, встановлювати умовну межу, при якій починає реагувати п'єзозумер, змінюється колір світлодіодів чи виводиться відповідне повідомлення на LCD-дисплей. Це особливо актуально у ситуаціях, коли пристрій використовується в середовищі з варіативною геометрією або неоднорідною конфігурацією об'єктів, де статичне значення спрацьовування було б неефективним.

Окрім налаштування порогу, потенціометр може виконувати й інші функції: наприклад, регулювання гучності звукового сигналу, встановлення часових

затримок, або навіть вибір одного з кількох режимів роботи системи (якщо це передбачено програмно). Таким чином, потенціометр у нашому випадку – це не просто пасивний елемент, а гнучкий регулятор поведінки системи, своєрідне «аналогове меню» без кнопок і дисплеїв.

Існують різні типи потенціометрів – обертові, слайдерні, цифрові, тримери тощо. У даному проєкті використовується класичний обертовий потенціометр з фіксованим опором у 10 кОм, який є найпоширенішим варіантом для мікроконтролерних систем. Його перевагою є стабільність, простота використання та надзвичайна сумісність із ESP32 без необхідності використання зовнішніх підсилювачів або перетворювачів сигналу.

Також варто згадати, що фізична стійкість потенціометра до зносу залежить від якості матеріалів, частоти використання і ступеня пилозахисності. У реальних промислових системах часто застосовують герметизовані моделі або цифрові потенціометри, керовані по шині I2C або SPI. Але для освітньо-прикладного проєкту, де головна мета – зрозуміти принцип роботи й реалізувати базову взаємодію, обраний елемент є оптимальним за співвідношенням функціональності, ціни та зручності підключення.

Потенціометр у нашій системі – це ключ до гнучкого налаштування, що дозволяє адаптувати пристрій до змін середовища та потреб користувача. Простота цього елемента не повинна вводити в оману, адже його внесок у функціональність і зручність використання системи є фундаментальним [1].

## 2.6 Світлодіоди та резистори

У проектуванні електронних систем, де важлива інтуїтивна, візуально зрозуміла індикація стану, без світлодіодів (LED) (рис. 2.9) не обходиться практично жоден пристрій. У даному проекті світлодіоди виконують не декоративну, а функціонально критичну роль – вони виступають мовчазними інформаторами, які з високою швидкістю сигналізують користувачеві про зміну відстані до об'єкта у полі зору ультразвукового датчика. Вони діють як світлові маркери безпеки: зелений – усе гаразд, жовтий – увага, червоний – небезпека.



Рисунок 2.9 – Світлодіод (LED) [16]

Світлодіод – це напівпровідниковий прилад, який випромінює світло внаслідок явища електролюмінесценції, коли носії заряду (електрони та дірки) рекомбінують у р-п переході. Це відбувається тільки при прямому підключенні: анод отримує плюс, катод – мінус. У середині корпусу знаходиться невеличкий кристал напівпровідникового матеріалу (зазвичай на основі GaAs, InGaN тощо), який і є джерелом світлового випромінювання. Колір світла залежить від енергетичного зазору матеріалу: червоний, зелений, синій, жовтий, білий – кожен має свої фізичні властивості.

У проєкті використано три світлодіоди різного кольору – червоний, жовтий і зелений. Цей набір кольорів є символічно зрозумілим навіть інтуїтивно, бо нагадує сигнал світлофора. Такий підхід дозволяє зчитувати стан системи навіть без дисплея, що особливо важливо в умовах обмеженої видимості або для людей з вадами слуху.

Жоден світлодіод не має працювати напряму від джерела живлення чи цифрового піну мікроконтролера без струмообмежувального резистора (рис. 2.10). Світлодіоди чутливі до перевищення струму – навіть невелике перенапруження може їх миттєво пошкодити. Саме тому в кожному каналі керування LED включається послідовно під'єднаний резистор, який виконує роль обмежувача сили струму, оберігаючи діод і саму мікросхему контролера [6]



Рисунок 2.10 – Резистор [37]

Уся система виявлення відстані спирається на простий, але наочний візуальний зворотний зв'язок. Коли об'єкт перебуває на безпечній відстані – світиться зелений. Як тільки об'єкт наближається до встановленого порогу, активується жовтий – сигнал попередження. При ще більшій близькості вмикається червоний світлодіод, який вказує на критичний стан та супроводжується звуковим сигналом зумера. Такий метод індикації дозволяє навіть без відображення точних значень дистанції зрозуміти ситуацію миттєво.

## 2.7 Макетна плата

Макетна плата (англ. breadboard) (рис. 2.11) – це багаторазовий пристрій для безпаянного монтажу електронних компонентів. Її поверхня – це сітка отворів, упорядкованих у ряди та стовпці. Під пластиковим корпусом прихована металева контактна система, яка об'єднує певні групи отворів електрично.

Стандартна макетна плата поділена на такі зони:

Горизонтальні ряди (power rails) – зазвичай розташовані з боків, позначені символами “+” і “-”. Вони використовуються для розведення живлення та землі.

Центральна сітка – поділена на дві половини, кожна з яких має вертикальні колонки з п'яти з'єднаних отворів. Між половинами – ізоляційна канавка, призначена для розміщення мікросхем типу DIP, щоб кожна ніжка була на окремій контактній смузі.

Контактна основа – виготовлена зі спеціальних пружинних металевих пластин, які забезпечують надійне притискання ніжок компонентів без необхідності паяння.

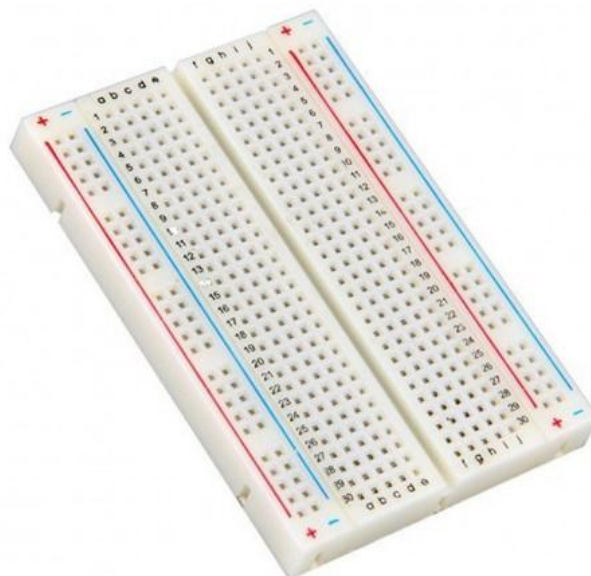


Рисунок 2.11 – Макетна плата [32]

Принцип дії макетної плати полягає у швидкому та зручному з'єднанні компонентів завдяки пружним металевим затискачам, розташованим під кожним рядом отворів. Пристрій уставляється просто: ніжка резистора, транзистора, світлодіода чи провід вставляється в отвір – і одразу входить у контакт із відповідною металевою колодкою.

Таким чином, за лічені хвилини можна зібрати повноцінну схему, змінити конфігурацію, додати або видалити компоненти – все це без жодного паяння, що критично важливо на етапі експериментування.

## 2.8 Симулятор Wokwi

Wokwi – це симуляційна платформа, що працює прямо в браузері та дозволяє моделювати проекти на основі Arduino, ESP32, Raspberry Pi Pico та багатьох інших мікроконтролерів. Інтерфейс інтуїтивно зрозумілий: користувач може розмістити компоненти, з'єднати їх, написати або вставити код і натиснути "Старт" – система одразу почне симуляцію роботи пристрою.

Це не просто онлайн-редактор, це повноцінне середовище, яке дає змогу побачити поведінку схеми в реальному часі, оцінити реакцію пристроїв, а також експериментувати без жодних витрат на обладнання.

Тому для створення та тестування системи вимірювання відстані з ультразвуковим датчиком, світлодіодами, зумером, потенціометром та LCD-дисплеєм було використано саме Wokwi як основний полігон для моделювання.

Завдяки тому, що симуляція в Wokwi відбувається з мінімальними затримками, можна миттєво побачити наслідки будь-якої зміни в кодї чи підключенні. Це дозволяє значно скоротити цикл розробки, зосередитися на функціоналі та логіці, а не на "залізі".

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 34
Зм.	Арк.	№ докум.	Підпис	Дата		

## 2.9 Висновки до другого розділу

У другому розділі було здійснено глибокий огляд апаратної основи системи, розглянуто характеристики кожного з компонентів, що беруть участь у реалізації проєкту: мікроконтролера ESP32, ультразвукового датчика відстані, п'єзозумера, LCD-дисплея, потенціометра, світлодіодів із резисторами та макетної плати. Окрему увагу було приділено середовищу віртуального моделювання Wokwi, яке відіграє важливу роль у процесі розробки, тестування й відлагодження. Проведений аналіз дозволив чітко зрозуміти функціональні можливості елементів, їхню взаємодію в системі, а також визначити оптимальні способи реалізації задуманої логіки.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		35

### 3. АПАРАТНА ІНТЕГРАЦІЯ КОМПОНЕНТІВ ІЗ МІКРОКОНТРОЛЕРОМ ESP32 ТА ПЕРЕДАЧА ДАНИХ ДО ХМАРИ

#### 3.1 Підключення ультразвукового датчика

Одним з центральних сенсорних елементів нашої системи є ультразвуковий датчик HC-SR04, здатний оцінювати відстань до об'єктів шляхом випромінювання звукових хвиль у невидимому для людського вуха діапазоні. Завдяки високій точності та доступності цей модуль є надзвичайно популярним у системах автоматизації, безпеки та робототехніці.

Слід відзначити, що саме даний компонент відіграє фундаментальну роль у всій системі. Він ініціює реакції інших елементів: активує світлодіоди відповідного кольору залежно від відстані, керує частотою пульсацій зумера, подає інформацію на дисплей та, зрештою, забезпечує «сприйняття» довкілля пристроєм.

Для підключення ультразвукового датчика до мікроконтролера ESP32 (рис. 3.1) використано чотири основні контакти модуля. Контакт VCC під'єднано до виводу 3V3 на платі ESP32, що забезпечує живлення сенсора стабільною напругою 3.3 вольт. Це рішення є обґрунтованим, оскільки в даній конструкції HC-SR04 стабільно працює від 3.3 В без потреби у перетворювачах напруги. Контакт GND, відповідальний за загальний «нуль», підключено до одного з GND-виводів ESP32 – через горизонтальну шину на макетній платі. Далі, сигнальний пін Trig, який відповідає за ініціацію вимірювання, приєднано до цифрового порту GPIO5 на ESP32. Контакт Echo, який надсилає назад імпульс з інформацією про час повернення звуку, підключено до виводу GPIO18. Таким чином, ці два пін-контакти забезпечують повноцінний обмін даними між сенсором і мікроконтролером, дозволяючи системі зчитувати точні вимірювання.

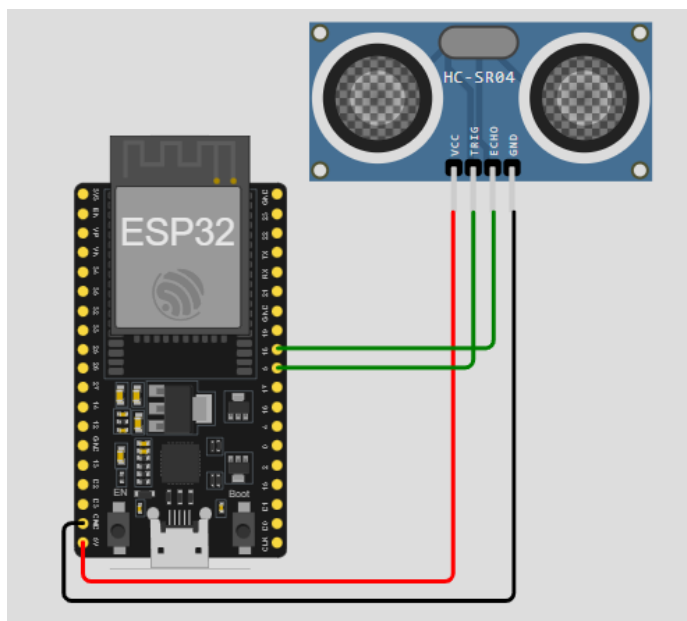


Рисунок 3.1 – Підключення ультразвукового датчика

Фізично модуль HC-SR04 розміщено на правому краї макетної плати, де його зручно інтегрувати з рештою компонентів. Завдяки лаконічному та зрозумілому підключенню цей сенсор можна швидко замінити або протестувати без демонтажу всієї схеми. До того ж, завдяки застосуванню кольорових Dupont-проводів кожен з'єднувальний канал легко ідентифікувати візуально.

### 3.2 Підключення LCD-дисплея

У межах створеної системи виведення інформації на дисплей відіграє не лише допоміжну, а й функціонально критичну роль. Саме LCD-модуль, під'єднаний до ESP32, дозволяє візуалізувати статус системи, миттєво передаючи користувачеві зрозумілу текстову інформацію про поточну відстань до об'єкта, граничні значення та зміну системних станів. У цьому проєкті використано дисплей із інтерфейсом I2C, який значно спрощує підключення і водночас не знижує функціональності.

Перш ніж перейти до технічного аспекту, варто зупинитись на тому, чим дисплей з I2C-шлюзом відрізняється від звичайного LCD 16x02 без адаптера.

Класична версія має цілу низку виводів – зазвичай 16 – серед яких: живлення, контроль контрастності, вибір режиму, сигнальні шини даних (D4–D7), а також керувальні пін-контакти (RS, RW, E). Через це її підключення до мікроконтролера вимагає щонайменше 6-8 цифрових портів, що є критично недоцільним у компактних проєктах з обмеженим числом GPIO. Окрім того, додатково потребується регулятор контрасту потенціометр який займає місце та ускладнює монтаж.

Натомість дисплей із вбудованим I2C-конвертером зводить усе підключення до чотирьох проводів: VCC, GND, SDA та SCL.

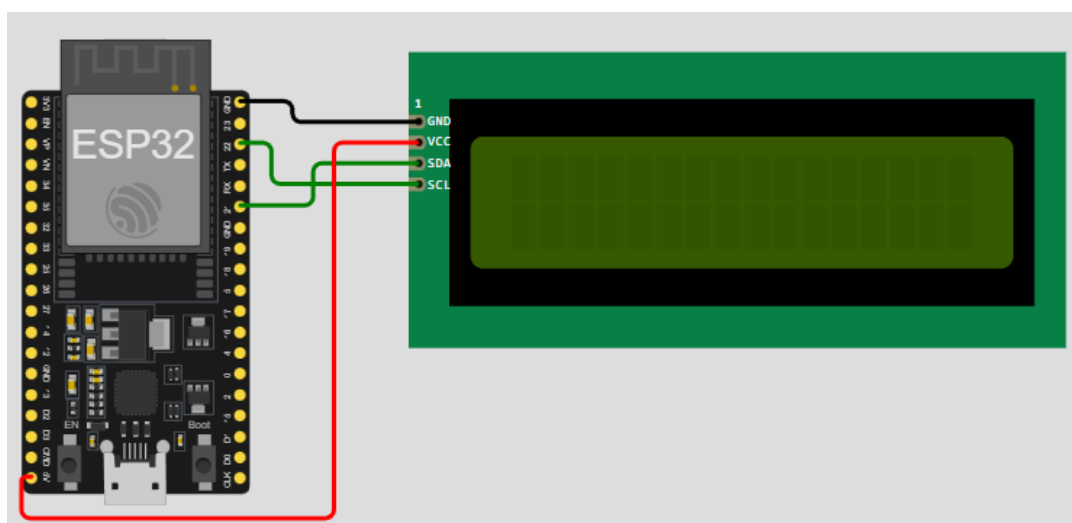


Рисунок 3.2 – Підключення дисплея

Два перші проводи відповідають за живлення VCC підключено до виводу 3V3 ESP32, а GND – до загального «нульового» потенціалу. Щодо двох сигнальних каналів, SDA (Serial Data) та SCL (Serial Clock), то вони забезпечують повноцінний обмін інформацією за допомогою I2C-протоколу. У нашій конфігурації SDA підключено до GPIO21, а SCL – до GPIO22. Така компактна і розумна топологія підключення відкриває можливість легко масштабувати систему, залишаючи інші порти мікроконтролера для додаткових сенсорів чи виконавчих елементів.

Візуально LCD-дисплей розташовано у верхній частині макетної плати, де його екран добре проглядається. Його використання не тільки зручно в роботі, а й

естетично виправдане – всі написи рівномірно впорядковані, символи відображаються чітко, завдяки підсвітці контрастність залишається стабільною незалежно від умов освітлення.

Окремої згадки заслуговує адресація дисплея. Кожен I2C-дисплей має унікальну HEX-адресу, яку потрібно вказувати в коді для правильного з'єднання. У нашому випадку вона – 0x27, що є типовим значенням для більшості модулів із I2C-адаптерами на базі чипа PCF8574. Якщо б ми використовували кілька таких дисплеїв одночасно, то через зміну перемичок на платі розширення можна змінити їхні адреси і використовувати одночасно.

Таким чином, вибір на користь I2C LCD у цьому проєкті був зумовлений не лише зручністю монтажу, а й економією ресурсів мікроконтролера, простотою програмної реалізації та швидкістю розгортання.

### 3.3 Підключення зумера

Звукові сигнали – це один із найефективніших способів оперативного сповіщення користувача про зміну стану системи. У рамках даного проєкту для індикації критичних ситуацій або зміни відстані використовується п'єзозумер, який забезпечує звукове повідомлення у реальному часі. Однак перш ніж перейти до технічного підключення, слід чітко розрізняти два основних типи зумерів: активні та пасивні.

Активний зумер є самодостатнім пристроєм: в його корпусі вмонтовано генератор, який автоматично формує звуковий сигнал при подачі на вхід логічного рівня HIGH (тобто 3.3 В або 5 В). Такий тип зумера зручний у випадках, коли потрібен постійний однотипний звук, без змін частоти. Його використання значно спрощує код, адже достатньо одного цифрового сигналу для запуску та зупинки.

Пасивний зумер, натомість, вимагає зовнішнього генератора – тобто сигналу з певною частотою, що подається із мікроконтролера. Саме цей тип зумера змодельовано в емуляторі Wokwi (рис. 3.3), який використовується в рамках даного

проекту. Пасивний зумер відкриває значно ширші можливості – можна динамічно змінювати частоту сигналу, створювати мелодії або варіювати звук залежно від відстані до об'єкта. Це дає змогу реалізувати гнучку логіку реагування пристрою на зовнішні умови.

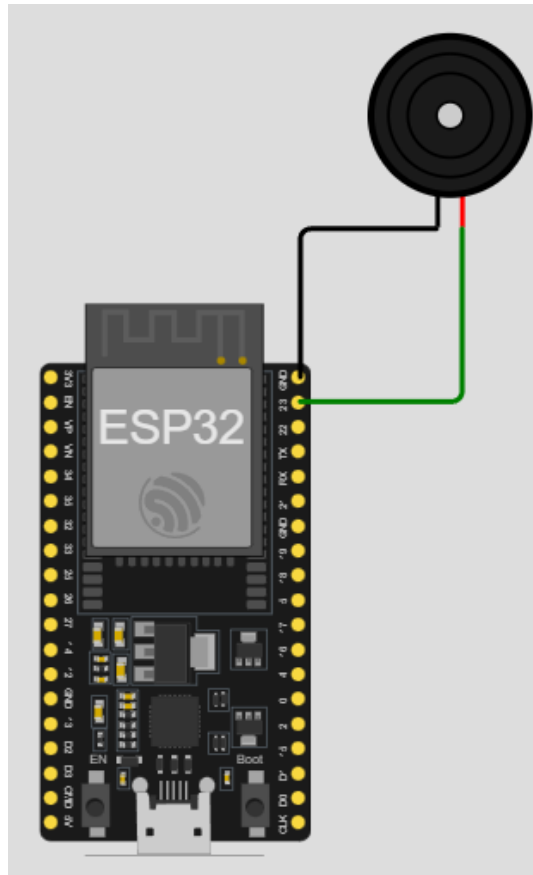


Рисунок 3.3 – Підключення зумера

Зумер, змодельований у Wokwi, має виводи: один – (плюс), інший – катод (мінус). У нашому макеті катод п'єзозумера підключено до GND на макетній дві платі. Анод з'єднано з цифровим піном D23 (GPIO23) плати ESP32, який у коді використовується для генерації сигналу за допомогою функції `tone()`.

Катод зумера підключено до загальної землі (GND) – ця точка є спільною для всіх компонентів системи і забезпечує замкнене коло живлення. Анод зумера приєднано до виводу GPIO23 на ESP32, що призначено для генерування частотного PWM-сигналу.

Саме цей вивід керує зумером за допомогою функцій типу `tone()`, які дозволяють задати точну частоту генерованого звуку. Оскільки ESP32 підтримує апаратне PWM-модуляторне керування, це дозволяє досягати високої точності у відтворенні сигналу.

Для з'єднання зумера з ESP32 використовується стандартна бредборд-плата (макетна плата), яка дозволяє швидко реалізовувати електричні ланцюги без пайки.

Живлення, яке надходить до зумера через цифровий пін, не є постійною напругою, а перемінним сигналом з певною частотою. Звуковий сигнал створюється за рахунок вібрації п'єзокерамічного елемента усередині корпусу зумера під дією змінного електричного поля. Частота коливань визначає тон звуку – чим вона вища, тим голосніший і пронизливіший сигнал. Такий підхід дозволяє дуже точно передавати ступінь критичності ситуації.

На відміну від активного зумера, який вимагає лише подачі постійної логічної одиниці для генерації звуку, пасивний зумер вимагає програмного управління – тобто подачі PWM-сигналу з певною частотою. Це ускладнює реалізацію, однак значно розширює функціонал: дозволяє динамічно змінювати частоту, створювати сигналізації із наростаючим тоном, мелодіями або пульсуючими ефектами.

Суть роботи зумера полягає у подачі на нього сигналу певної частоти. У скетчі Arduino це реалізується за допомогою функції `tone(pin, frequency)`, де `pin` - це номер цифрового виходу, а `frequency` – частота звуку в герцах. Чим менша відстань до об'єкта, тим вища частота – так реалізується звукове попередження наближення.

Оскільки пасивний зумер дозволяє генерувати змінні частоти, його зручно використовувати для створення градації небезпеки: наприклад, на великій відстані звук не подається зовсім, при середній – тихий тон, а при критичному наближенні – гучний і частий звуковий імпульс. Усе це дозволяє зробити взаємодію з пристроєм інтуїтивною і зрозумілою навіть без екранного інтерфейсу.

У реальних умовах, при розробці фізичного пристрою, інженер може сам обирати між активним і пасивним зумером залежно від потреб. Якщо ціллю є

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 41
Зм.	Арк.	№ докум.	Підпис	Дата		

просте попередження - наприклад, постійний звуковий сигнал у разі спрацювання сигналізації – доцільніше використовувати активний зумер. Якщо ж важлива динамічна зміна тону, звучання мелодій або сигналів різної частоти – пасивний є безумовно кращим варіантом. Різниця між активним та пасивним зумером зображено на рисунку 3.4

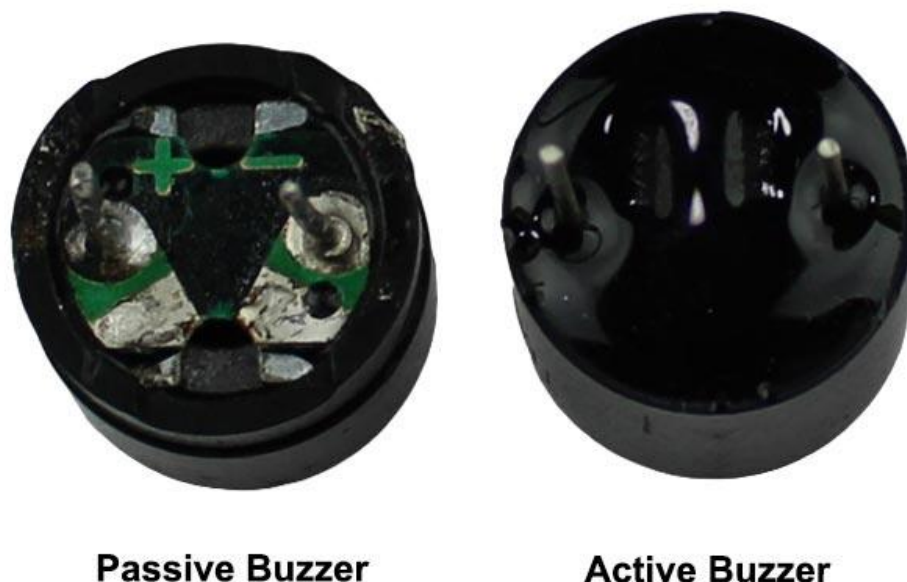


Рисунок 3.4 – Активний та пасивний зумер [23]

Таким чином, у рамках даного проєкту було обрано пасивний п’єзозумер, що відкрив додаткові функціональні можливості для реалізації складнішої логіки інформування користувача про стан об’єкта. Його правильне підключення та програмна реалізація – це ще один приклад того, як простий компонент здатен суттєво вплинути на якість взаємодії з пристроєм.

### 3.4 Підключення потенціометра

Потенціометр – це не просто змінний резистор, як його часто спрощено називають, а надзвичайно зручний елемент керування, який дозволяє тонко регулювати параметри в системі, використовуючи механічне обертання. У нашому

пристрої потенціометр виконує роль регулятора порогових значень для визначення рівня наближення об'єкта. Його використання додає пристрою елемент ручного налаштування – тобто надає користувачеві свободу в зміні поведінки системи без перепрошивки або втручання в код.

По суті, потенціометр складається з резистивного шару з ковзаючим контактом, який рухається по ньому. Це створює поділ резистивного елемента на дві частини, опір яких змінюється залежно від положення повзунка. Один край потенціометра підключається до живлення (3.3 В), другий – до землі (GND), а середній контакт (вивід посередині) – це вихідний сигнал, який формується відповідно до положення ручки. Він передає аналогову напругу, яка плаває між 0 і 3.3 В.

Для взаємодії з ESP32 ми використали тристоронній потенціометр, що ідеально підходить для зчитування аналогових значень. Крайній лівий контакт потенціометра з'єднується з GND – спільною «землею» для всієї схеми. Крайній правий контакт під'єднується до 3V3 ESP32, щоб забезпечити стабільне живлення. Середній (центральний) контакт, також знаний як "wiper", підключається до одного з аналогових входів ESP32, у нашому випадку – виводу GPIO34. Цей контакт подає на ESP32 значення напруги, пропорційне до положення ручки потенціометра. У коді мікроконтролера ці дані зчитуються функцією `analogRead()` і перетворюються на цифрове значення в діапазоні 0-4095, враховуючи 12-бітну розрядність АЦП (аналогово-цифрового перетворювача) ESP32.

На рисунку 3.5 зображено підключення потенціометра до мікроконтролера.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

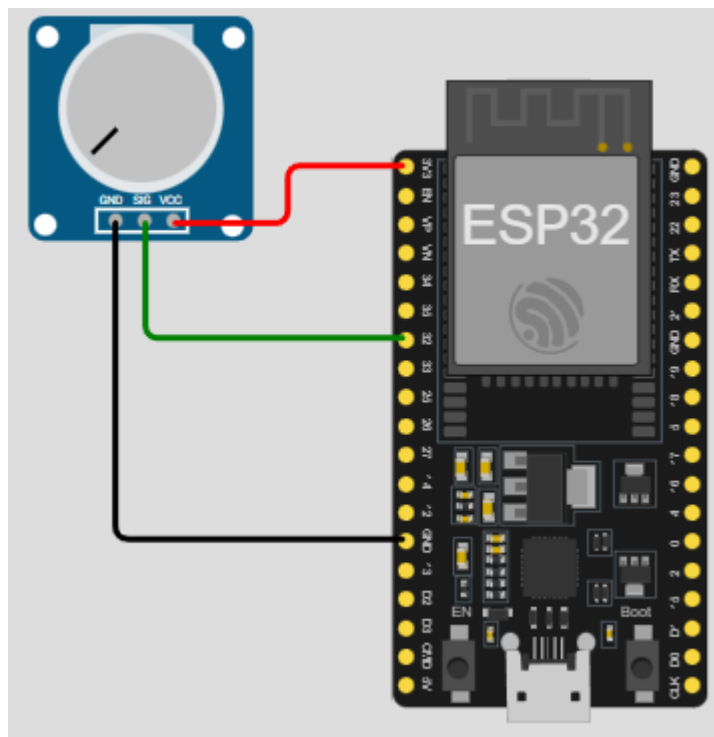


Рисунок 3.5 - Підключення потенціометра

На макетній платі потенціометр закріплюється у верхній частині, його ніжки вставляються у три послідовні отвори так, щоб забезпечити зручність доступу до регулятора. Проводи від GND, 3V3 та аналогового входу проходять до відповідних шин або пінів мікроконтролера, забезпечуючи надійний електричний контакт. Завдяки симетричній формі компонента, його легко вмонтувати навіть у щільно заповнену схему.

Фізичне обертання потенціометра дозволяє регулювати пороги спрацьовування системи сигналізації. Наприклад, обертання ручки може змінювати відстань, при якій активується звуковий сигнал чи змінюється колір RGB-світлодіода. Це дуже зручно в контексті реального середовища, де умови експлуатації можуть змінюватися, і потрібна можливість оперативного переналаштування без підключення до комп'ютера.

Завдяки тому, що потенціометр подає безперервний спектр значень, його можна використовувати не лише для порогового налаштування, а й для створення

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 44
Зм.	Арк.	№ докум.	Підпис	Дата		

адаптивних алгоритмів реагування: наприклад, залежно від положення регулятора система може поводитися агресивніше або, навпаки, толерантніше реагувати на об'єкти, що наближаються.

### 3.5 Підключення світлодіодів з резисторами

Світлодіоди, попри свою зовнішню простоту, є важливими інформаційними носіями в електронних системах. У нашому проєкті вони виконують роль візуальних індикаторів – сигналізують про рівень небезпеки відповідно до відстані до об'єкта, що наближається. Візуальна система оповіщення включає три окремі світлодіоди різного кольору: зелений, жовтий та червоний, які вказують на безпечну, потенційно небезпечну та критичну зони відповідно. Для коректної та надійної роботи кожного світлодіода доцільно реалізувати відповідну електричну схему, в якій особливу роль відіграють захисні резистори.

На рисунку 3.6 зображено підключення світлодіодів з резисторами.

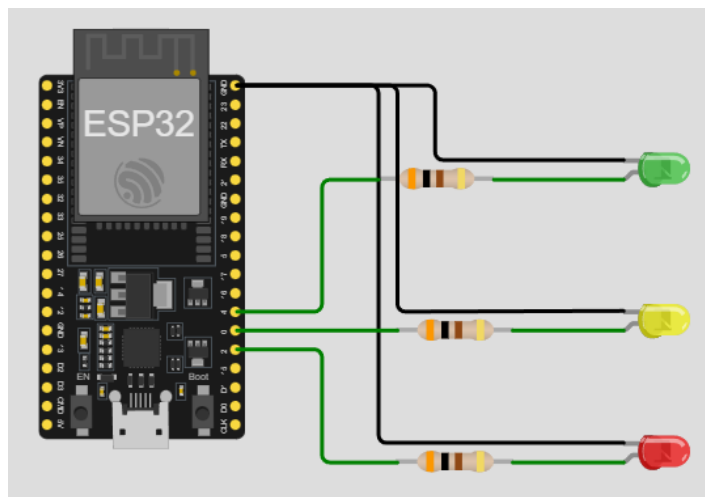


Рисунок 3.6 – Підключення світлодіодів з резисторами

Підключення кожного світлодіода реалізовано за схемою із відкритим колектором: анод (довга ніжка) під'єднується безпосередньо до вихідного цифрового піну ESP32, а катод (коротка ніжка) з'єднується через резистор

номіналом 220 Ом на загальний провід (GND). Такий спосіб підключення дає змогу керувати кожним світлодіодом шляхом подачі логічної одиниці на відповідний пін ESP32. У момент, коли на пині встановлюється рівень 3.3 В, струм починає проходити через світлодіод у прямому напрямку та, пройшовши через обмежувальний резистор, замикається на землю.

Цифрові виходи ESP32 були обрані наступним чином:

- зелений світлодіод керується з пину GPIO4;
- жовтий світлодіод керується з пину GPIO0;
- червоний світлодіод керується з пину GPIO2.

Кожен із трьох ланцюгів ідентичний за принципом дії, але виконує свою логічну функцію в програмній частині. Резистори по 220 Ом підібрані таким чином, щоб забезпечити струм у межах 10 мА, що є безпечним як для ESP32, так і для самих світлодіодів. З урахуванням напруги живлення ESP32 на рівні 3.3 В, падіння напруги на світлодіоді в середньому становить близько 2.0 В для червоного і жовтого кольорів та до 3.0 В - для зеленого. Решта напруги падає на резистор, що і визначає величину струму в колі.

Така схема дозволяє зручно реалізовувати змінну візуалізацію стану системи без затримок або помилкових спрацьовувань. Застосування індивідуальних резисторів для кожного світлодіода дає змогу уникнути перевантаження виходів ESP32 і забезпечити стабільну яскравість світіння, незалежно від кількості одночасно активних індикаторів.

### 3.6 Готова апаратна реалізація та робота системи

На рисунках 3.7, 3.8 та 3.9 зображена готова апаратна реалізація та робота системи. На рис. 3.8 можна побачити роботу інформаційної системи, де об'єкт знаходиться на безпечній відстані, а роботу із критичною відстанню – на рис. 3.9. Також бачимо як відображаються дані на LCD екрані та як спрацьовують світлодіоди і зумер.

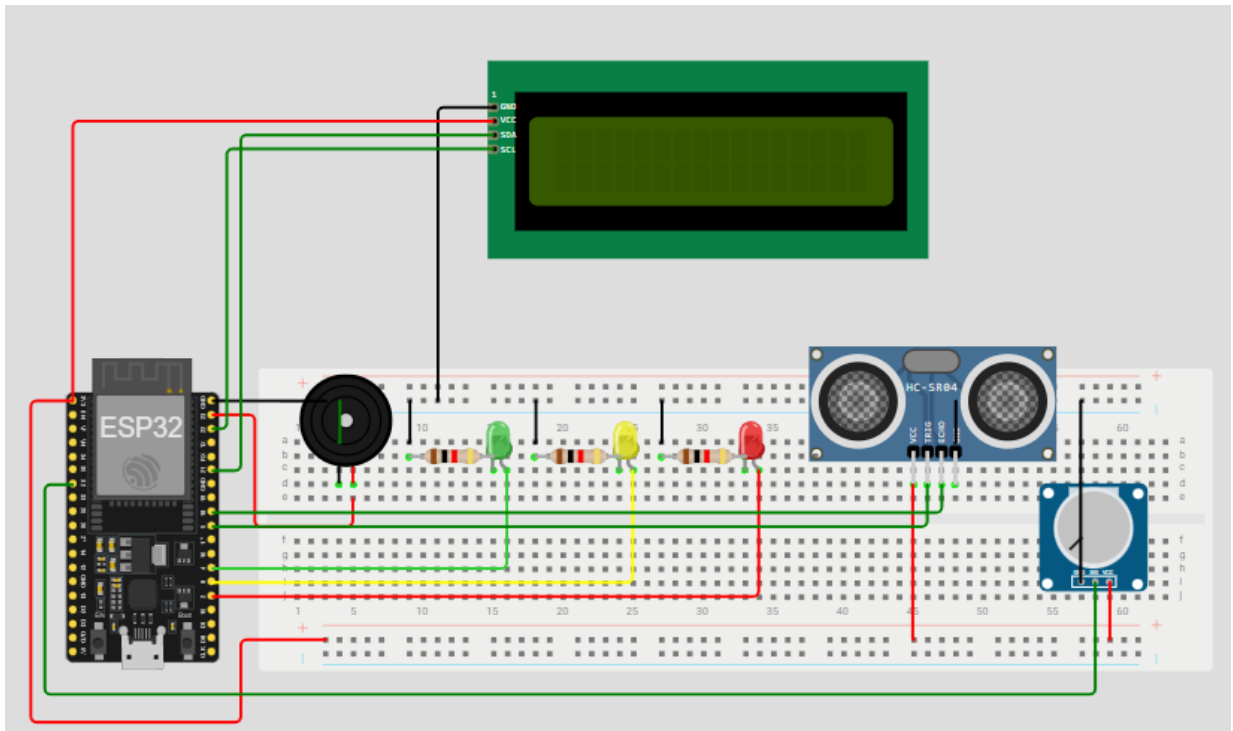


Рисунок 3.7 – Загальна схема збірки пристрою

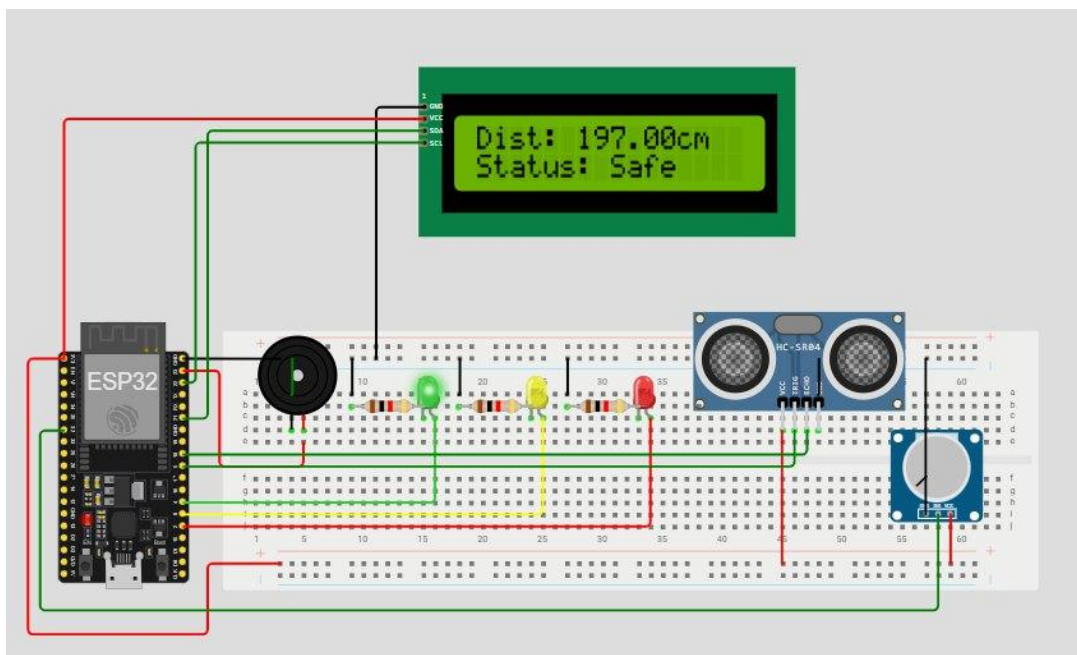


Рисунок 3.8 – Візуалізація функціонування системи в реальному часі

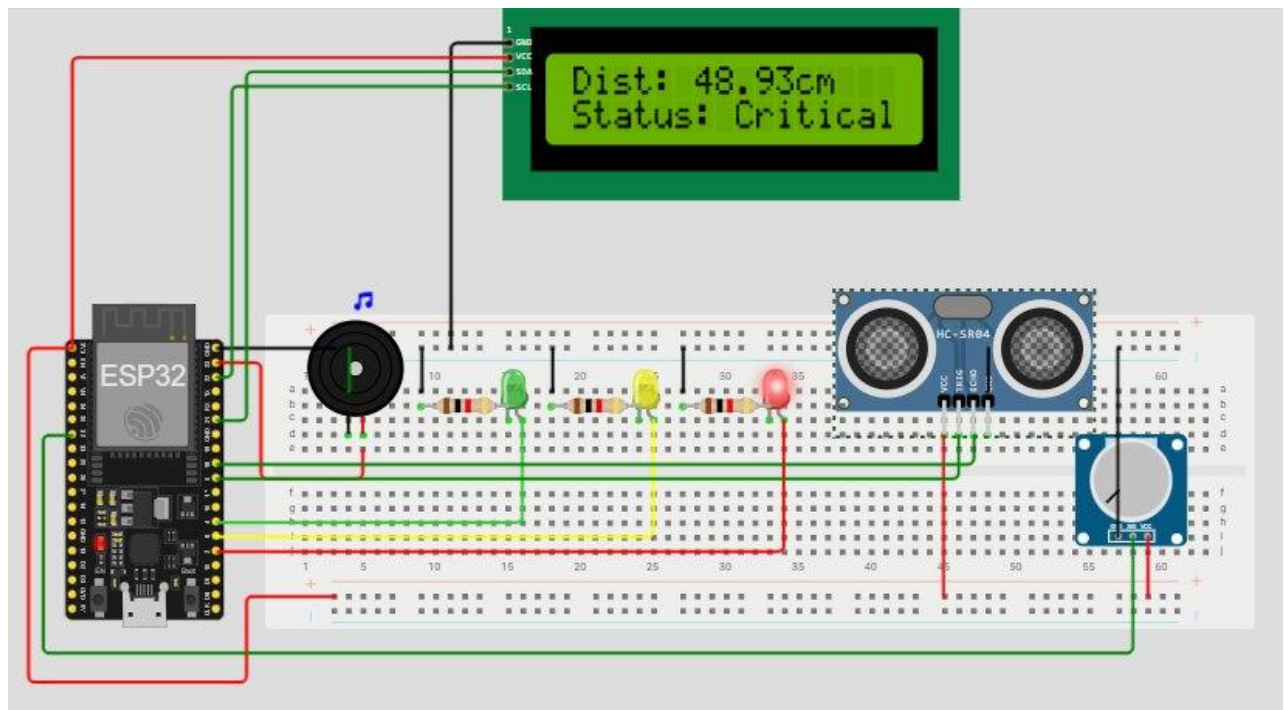


Рисунок 3.9 – Візуалізація результатів роботи пристрою при досягненні небезпечної дистанції

### 3.7 Загальна архітектура IoT у межах проєкту

Поняття "Інтернет речей" (англ. Internet of Things, IoT) у сучасному світі охоплює не лише автоматизацію, а й новий рівень комунікації між пристроями, коли навіть найпростіші електронні модулі отримують змогу спілкуватися з глобальною мережею.

У багаторівневій архітектурі Інтернету речей (IoT) важливою є не лише передача сигналу від сенсора до хмари, а й чітке розуміння ролі кожного структурного рівня в цьому процесі. Класична триврівнева модель – Endpoint, Edge та Cloud – дозволяє максимально ефективно реалізовувати розподілену обробку інформації, мінімізувати затримки, оптимізувати навантаження на мережу та забезпечити масштабованість системи.

Endpoint-рівень – джерело даних. Першим, найнижчим рівнем є Endpoint – кінцеві пристрої, які здійснюють безпосередню взаємодію з навколишнім середовищем. Це можуть бути датчики температури, вологості, тиску,

акселерометри, світлові сенсори, камери тощо. Основне їхнє завдання - захоплення фізичних параметрів і перетворення їх у сигнали, які можна подати на подальшу обробку. Ці пристрої зазвичай обмежені у своїх ресурсах – мають мінімальну обчислювальну потужність, незначний обсяг пам'яті та часто працюють на акумуляторному живленні. Тому вони переважно не здійснюють складної логіки - лише збір і передача даних.

Edge-рівень – попередня обробка та фільтрація. Тут розташовані пристрої, які стоять "на межі" між сенсорами й хмарою. Вони мають більше ресурсів, ніж Endpoint-компоненти, і часто виконують роль локального хабу. Сюди входять мікроконтролери (як-от ESP32, Raspberry Pi, промислові контролери), які збирають дані з кількох джерел, аналізують їх і приймають базові рішення – наприклад, активувати реле при перевищенні певного порогу.

Edge-рівень дозволяє значно розвантажити хмарну інфраструктуру, оскільки не всі дані надсилаються в інтернет – лише ті, що мають аналітичне значення або вимагають зберігання. Крім того, Edge-обробка зменшує затримки при критичних подіях – наприклад, аварійне відключення обладнання може виконуватися миттєво без звернення до хмари.

Cloud-рівень – аналітика, зберігання та взаємодія з користувачем. Останній і найпотужніший рівень. Тут зосереджена глобальна обробка великих обсягів даних, довгострокове зберігання, створення візуалізацій, відправлення повідомлень користувачеві, інтеграція з іншими сервісами. На цьому рівні можуть працювати складні аналітичні алгоритми, системи машинного навчання, контролери доступу та панелі управління.

Хмара часто виступає в ролі центру прийняття стратегічних рішень, які базуються на історичних даних і трендах. Через неї користувач має змогу переглядати, як працює система, отримувати сповіщення, змінювати конфігурацію або віддалено втручатися в її роботу.

Трирівнева модель "Endpoint - Edge - Cloud" є гнучкою основою сучасної IoT-інфраструктури. Вона дозволяє розподіляти навантаження відповідно до

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 49
Зм.	Арк.	№ докум.	Підпис	Дата		

можливостей пристроїв, забезпечує швидкість реакції, економію трафіку та потужну аналітику.

У межах даного проєкту IoT-архітектура реалізована у вигляді багаторівневої моделі, де кожен рівень виконує специфічну, але взаємопов'язану роль у загальному технологічному ланцюгу.

Розглянемо кожен із рівнів.

Перший рівень – сенсорний рівень (рівень збору даних). Цей рівень є першим у логічному ланцюжку і відповідає за безпосередню взаємодію з фізичним світом. Тут функціонують ультразвуковий датчик відстані HC-SR04 та потенціометр, які зчитують змінні параметри навколишнього середовища: відстань до об'єкта та значення напруги, відповідно. Дані з цих пристроїв мають первинний, "сирий" характер і потребують обробки, тому вони передаються на наступний рівень.

Наступний рівень – це рівень обробки (локальний інтелект). Центральною обчислювальною одиницею цього рівня є мікроконтролер ESP32. Саме тут дані отримують осмислення: ESP32 аналізує, порівнює з пороговими значеннями, приймає рішення щодо стану системи та відповідних дій (наприклад, активація світлодіодів або зумера). Також на цьому етапі формується інформаційне повідомлення для виведення на LCD-дисплей, що відображає поточні показники для користувача.

Рівень взаємодії (інтерфейс із користувачем) LCD-дисплей із I2C-модулем служить візуальним каналом комунікації. Саме сюди виводиться інформація про поточну відстань та статус системи у зручній формі, що дозволяє користувачеві оперативно оцінити ситуацію. Окремо варто відзначити роль RGB-світлодіодів, які, залежно від ситуації, змінюють колір, сигналізуючи про зміну дистанції. Вони також входять до цього рівня, оскільки безпосередньо повідомляють людину про стан системи.

Мережевий рівень (передача даних). На цьому етапі ESP32 використовує вбудований Wi-Fi-модуль для встановлення зв'язку з Інтернетом. За допомогою відповідного протоколу (HTTP), сформовані дані надсилаються у зовнішній

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 50
Зм.	Арк.	№ докум.	Підпис	Дата		

хмарний сервіс – Google Sheets. Це дозволяє забезпечити доступ до показників з будь-якої точки світу, що кардинально розширює межі застосування пристрою.

Хмарний рівень (збереження та візуалізація) у даному проєкті виконує функцію цифрового "сховища", де значення відстані зберігаються у вигляді таблиці. Важливо підкреслити, що цей рівень дає змогу зберігати історичні дані, будувати графіки, виявляти тренди та приймати рішення на основі накопиченої інформації. У майбутньому можливе розширення цього рівня за рахунок інтеграції з Google Apps Script або іншими API для розсилки повідомлень чи звітів.

### 3.8 Взаємодія з Google Sheets як приклад хмарного сервісу

У сучасному цифровому середовищі все більше значення набуває можливість інтеграції локальних мікроконтролерних систем з глобальними сервісами обробки та зберігання даних. Одним із найзручніших способів досягти цієї мети без потреби розгортання складної серверної інфраструктури є використання Google Sheets – табличного інструменту, що працює в межах хмарної платформи Google Workspace. Завдяки своїй відкритості до зовнішніх запитів та підтримці Google Apps Script, Google Sheets стає своєрідним «легким хмарним бекендом» для IoT-проєктів.

У межах даного проєкту відбувається зв'язок між мікроконтролером ESP32 та Google Sheets за допомогою HTTP-запитів. Суть методу полягає у відправленні інформації через GET або POST запити до скрипта, що виконується на сервері Google. Цей скрипт – написаний на JavaScript-подібній мові Apps Script – обробляє запит і вносить дані в таблицю.

Такий підхід дозволяє обходитися без проміжних серверів чи баз даних. Інформація передається безпосередньо з пристрою в документ, що відкритий з будь-якої точки світу, має резервне копіювання та зручний інтерфейс для аналізу зібраних показників. Тобто Google Sheets тут виступає не просто як пасивний архів, а як інтерактивний аналітичний інструмент.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 51
Зм.	Арк.	№ докум.	Підпис	Дата		

Щоб реалізувати таку взаємодію, необхідно створити нову Google-таблицю, перейти до редактора скриптів і написати відповідний код на Google Apps Script. Скрипт має бути опублікований як вебдодаток з дозволом на прийом запитів будь-якого користувача. Після цього генерується унікальна URL-адреса, до якої можна звертатися з ESP32.

У коді прошивки мікроконтролера використовується бібліотека WiFi та відповідні функції для надсилання HTTP-запиту. У тілі запиту передається змінна, наприклад, значення відстані, зчитане з ультразвукового датчика. Після обробки запиту скрипт додає дані до відповідного рядка таблиці разом із міткою часу.

На перший погляд, Google Sheets здається нетиповим вибором для IoT-хмари. На відміну від промислових рішень на зразок AWS чи Azure, Google Sheets не потребує реєстрації API-ключів, складного конфігурування або комерційної підписки. Це робить його надзвичайно привабливим для студентських, навчальних або дослідницьких проєктів, де важливо забезпечити наочність і зручність роботи.

Також важливо зазначити, що Google Sheets зберігає дані в реальному часі, дозволяє будувати графіки, виконувати автоматичні обчислення та навіть створювати оповіщення через інтеграцію з іншими Google-сервісами. Таким чином, таблиця перетворюється на керований центр прийняття рішень, де можна відстежувати хід роботи системи та реагувати на зміни.

Застосування Google Sheets як хмарного сховища даних у цьому проєкті доводить, що навіть прості інструменти можуть ефективно виконувати завдання, пов'язані з Internet of Things. Вони дозволяють організувати зв'язок між реальним світом і цифровим середовищем, не вимагаючи великих фінансових чи технічних ресурсів. Такий підхід відкриває широкі горизонти для створення адаптивних і масштабованих IoT-рішень навіть на початковому рівні.

	A	B	C
1	Timestamp	Distance	Status
2	02.05.2025 11:51:29	52.92	Medium
3	02.05.2025 11:52:00	52.92	Medium
4	02.05.2025 11:52:57	02.04	Critical
5	02.05.2025 12:54:40	1.92	Critical
6	02.05.2025 12:55:17	256.95	Safe
7	21.05.2025 10:19:42	196.96	Safe
8	21.05.2025 10:20:16	197.01	Safe
9	21.05.2025 10:20:30	197.00	Safe
10	21.05.2025 10:20:53	37.99	Critical
11	21.05.2025 10:21:07	48.93	Critical

Рисунок 3.10 – Показники у хмарному сервісі Google Sheets

### 3.9 Вибір протоколів передачі даних: обґрунтування та реалізація

Передача даних у середовищі Інтернету речей – не просто технічна дія, а основа повноцінної взаємодії між фізичними пристроями та цифровими хмарними сервісами. Тому критично важливо правильно підібрати протокол комунікації, враховуючи не лише швидкість чи обсяг передаваних даних, а й енергоспоживання, затримку, стабільність з'єднання та доступність інструментів для реалізації.

У контексті розробки IoT-проектів найчастіше застосовують два основні протоколи – HTTP (HyperText Transfer Protocol) та MQTT (Message Queuing Telemetry Transport). Вони представляють дві різні філософії передачі інформації.

HTTP – протокол запит-відповідь, який добре знайомий з веброботки. Він передбачає, що клієнт ініціює з'єднання, надсилає запит, а сервер – відповідає. Це зручно для одноразової або періодичної передачі даних до хмарної системи, наприклад – запис показників у Google Sheets.

MQTT – легкий, подієорієнтований протокол, який працює за принципом публікації та підписки. Пристрої не звертаються один до одного безпосередньо, а

надсилають повідомлення на "брокер", який розподіляє ці повідомлення серед підписників. Цей підхід особливо вигідний для систем, що потребують постійного зв'язку, швидкої реакції на зміни та підтримки багатьох пристроїв. [14]

У межах цього проекту було вирішено використати HTTP як базовий протокол передачі даних між ESP32 та хмарним сервісом Google Sheets. Такий вибір продиктований кількома чинниками. Розглянемо їх.

1. Сумісність із Google Apps Script. Оскільки HTTP-запити легко інтегруються з вебдодатками Google, що значно спрощує взаємодію із таблицею без додаткового посередника.

2. У проєкті достатньо передавати дані з пристрою в хмару – зворотний зв'язок не є обов'язковим.

3. HTTP не потребує встановлення брокера або сторонніх бібліотек. Усе, що необхідно – стандартні функції ESP32.

4. Дані передаються періодично, а не в реальному часі, тому затримки не критичні.

На боці Google Apps Script запит приймається, розшифровується, і дані записуються в таблицю. Таким чином, передача виглядає прозоро, легко налагоджується та не потребує значного програмного навантаження.

Попри обраний шлях, MQTT залишається цінною альтернативою для майбутніх розширень системи. Зокрема, у разі потреби організації зворотного зв'язку з пристроєм, синхронізації кількох ESP32 або використання централізованого брокера – MQTT стає доцільним. Його ефективність при низькій пропускній здатності мережі й підтримка QoS-рівнів дозволяє масштабувати проєкт до промислового рівня.

Отже, обраний для цього проєкту протокол HTTP – це виважене компромісне рішення між зручністю, функціональністю та технічними обмеженнями. Він ідеально підходить для односпрямованої передачі даних із простих пристроїв у хмару. У той же час, усвідомлення можливостей альтернатив, зокрема MQTT, відкриває простір для подальшого розвитку та ускладнення системи.

### 3.10 Обмеження хмарного середовища Google Sheets для IoT-систем

Хмарна платформа Google Sheets часто використовується в ролі базового інструменту для збирання даних з пристроїв Інтернету речей, особливо на етапі розробки або тестування. Вона приваблює простотою використання, безкоштовністю та можливістю інтеграції через Google Apps Script або сторонні сервіси. Проте, попри зручність, існує низка технічних обмежень, які суттєво впливають на масштабованість і стабільність роботи вбудованих систем.

Одним із основних бар'єрів є обмеження на кількість рядків у таблиці. У безкоштовній версії Google Sheets ця межа становить близько десяти мільйонів клітинок, що при постійному потоці даних з сенсорів може бути вичерпано доволі швидко. Особливо це актуально при високій частоті оновлення або використанні великої кількості полів для кожного запису.

Ще однією критичною характеристикою є швидкість обробки HTTP-запитів. Google Apps Script, який часто використовується як шлюз між IoT-пристроєм та таблицею, має обмеження на кількість виконань на добу (залежно від типу облікового запису), а також час виконання одного скрипту. У разі надмірної частоти звернень скрипти можуть повертати помилки або не встигати обробити запити, що призводить до втрати частини даних.

Крім того, Google Sheets не призначений для роботи в режимі реального часу. Між надсиланням запиту та появою запису може пройти декілька секунд або навіть хвилин у разі затримок у сервісі. У критичних додатках, де час реакції є визначальним, така затримка може нівелювати всю ефективність системи.

З розширенням системи, коли збільшується кількість сенсорів, записів або запитів, усі ці обмеження починають накопичуватися і спричиняють нестабільну роботу. Може зростати час відповіді, зменшуватись точність збирання даних, з'являться дублікати або пропуски. У таких умовах Google Sheets перестає бути

надійним рішенням і потребує заміни на спеціалізовані платформи, орієнтовані на роботу з потоковими або масовими IoT-даними.

Таким чином, хоча використання Google Sheets є слушним кроком для перших прототипів або простих освітніх проєктів, подальше розгортання масштабної системи потребує переходу до більш потужних і гнучких інструментів – таких як хмарні бази даних, брокери повідомлень або спеціалізовані IoT-платформи з підтримкою високої пропускну здатності та аналітики в реальному часі.

### 3.11 Висновки до третього розділу

У цьому розділі було послідовно розглянуто процес підключення кожного з апаратних компонентів системи до мікроконтролера ESP32. Детально проаналізовано електричні з'єднання, сигнальні контакти та принципи взаємодії між модулями, що забезпечують повноцінне функціонування системи. З особливою увагою описано особливості підключення дисплея, датчика відстані, зумера, потенціометра та RGB-світлодіодів із резисторами. Також розглянуто особливості реалізації передачі даних між фізичним пристроєм на базі ESP32 та хмарною платформою Google Sheets, що виступає прикладом простого, але дієвого хмарного сервісу. Детально описано загальну архітектуру IoT-системи, рівні її побудови, а також охарактеризовано основні протоколи передачі даних. На основі технічних, функціональних та практичних міркувань було обґрунтовано вибір HTTP-протоколу для реалізації односпрямованої комунікації. Реалізована система продемонструвала надійність, гнучкість та придатність до подальшого вдосконалення, зокрема в напрямку масштабування та переходу до більш складних протоколів.

## ВИСНОВКИ

У роботі, за результатами виконаних теоретичних та практичних досліджень, було реалізовано повноцінну інформаційну систему на основі мікроконтролера ESP32, здатну вимірювати відстань за допомогою ультразвукового сенсора, обробляти результати, сигналізувати користувача про наближення об'єкта засобами візуального та звукового оповіщення, а також здійснювати передачу даних до хмарного середовища для збереження та подальшого аналізу.

У першому розділі було здійснено загальний огляд проблематики автоматизованого вимірювання відстані, обґрунтовано доцільність використання мікроконтролерів у подібних системах, а також наведено порівняння традиційних та сучасних підходів до побудови подібних рішень.

У другому розділі проведено технічний аналіз апаратної складової проєкту. Було детально охарактеризовано всі апаратні компоненти, що складають основу системи: ультразвуковий датчик HC-SR04, активний п'єзозумер, LCD-дисплей з інтерфейсом I2C, потенціометр, світлодіоди різного кольору, резистори та макетна плата. Для кожного елемента розкрито принцип дії, функціональну роль у загальній структурі пристрою та особливості застосування у конкретній реалізації. Також описано інструмент для моделювання – Wokwi, що дозволив протестувати схему без потреби у фізичному збиранні..

У третьому розділі розглянуто поетапний процес підключення кожного з апаратних елементів до плати ESP32. Для кожного модуля описано електричні зв'язки, сигнальні лінії та логіку взаємодії з програмною частиною. Особливу увагу було приділено дотриманню електротехнічних норм, правильному використанню резисторів для захисту світлодіодів, а також обранню портів ESP32 для забезпечення стабільної роботи всієї системи.

У четвертому розділі зосереджено увагу на хмарному аспекті функціонування проєкту. Зокрема, наведено опис загальної архітектури IoT-рішень, визначено типові рівні взаємодії, описано логіку передачі даних у хмару та

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 57
Зм.	Арк.	№ докум.	Підпис	Дата		

обґрунтовано вибір Google Sheets як платформи для зберігання інформації. Додатково було розглянуто протоколи, доступні для обміну даними, і зроблено вибір на користь HTTP-запитів – як найбільш простого та стабільного рішення для даного рівня складності проєкту..

У підсумку, дипломна робота демонструє приклад створення функціональної, ефективної та економічно доцільної системи, здатної обробляти фізичні параметри навколишнього середовища, зокрема відстань до об'єкта, та передавати їх до мережі Інтернет без втрати точності чи швидкодії. Запропонована конструкція може легко масштабуватися, модифікуватися, доповнюватися іншими сенсорами, інтерфейсами або програмними рішеннями.

Практичне значення полягає у можливості застосування такого рішення для систем контролю простору, автоматизованого моніторингу наближення, безпеки в гаражах, складах, робототехнічних пристроях та інших автоматизованих системах.

Таким чином, реалізований проєкт поєднує у собі приклад інженерного підходу, сучасні інструменти розробки, елементи IoT-архітектури та практичні навички інтеграції мікроелектроніки з хмарними платформами.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Aguirre-Manríquez I. V., García-Guzmán M. Á., Razón-González J. P., Cano-Lara M. *Basic operation of rotary potentiometer A, B, C and W. J Innov Eng.* 2022. Vol. 6(19). P. 1-5.
2. Arduino with HC-SR04 Ultrasonic Sensor. Distance Measurement. LastMinuteEngineers. URL: <https://lastminuteengineers.com/arduino-sr04-ultrasonic-sensor-tutorial/> (дата звернення: 29.05.2025).
3. Asadi F. LCD and EEPROM. *In Essentials of Arduino™ Boards Programming: Step-by-Step Guide to Master Arduino Boards Hardware and Software.* 2023. DOI: 10.1007/978-1-4842-9600-4. P. 161-177.
4. Barybin O., Zaitseva E., Brazhnyi V. ТЕСТУВАННЯ БЕЗПЕКИ ПРИСТРОЇВ ІНТЕРНЕТУ РЕЧЕЙ НА БАЗІ МІКРОКОНТРОЛЕРА ESP32. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка».* № 2(6). 2022. С. 71-81.
5. Baumann P. Piezoelectric Buzzer. In *Selected Sensor Circuits: From Data Sheet to Simulation.* Wiesbaden: Springer Fachmedien Wiesbaden. 2022. P. 183-220.
6. Chang B. Y. The effective capacitance of a constant phase element with resistors in series. *Journal of Electrochemical Science and Technology.* 2022. Vol. 13(4). P. 479-485.
7. Classical structure of IoT system ResearchGate. URL: [https://www.researchgate.net/figure/Classical-structure-of-IoT-system\\_fig1\\_319594914](https://www.researchgate.net/figure/Classical-structure-of-IoT-system_fig1_319594914) (дата звернення: 29.05.2025).
8. ESP32: Detalhes internos e pinagem Fernando K Tecnologia. URL: <https://www.fernandok.com/2018/03/esp32-detalhes-internos-e-pinagem.html> (дата звернення: 29.05.2025).
9. Gerodimos A., Maglaras L., Ferrag M. A., Ayres N., Kantzavelou I. IoT: Communication protocols and security threats. *Internet of Things and Cyber-Physical Systems.* 2023. Vol. 3. P. 1-13.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 59
Зм.	Арк.	№ докум.	Підпис	Дата		

10. Gondosubroto R. Internet of Things from Scratch: Build IoT solutions for Industry 4.0 with ESP32, Raspberry Pi, and AWS. *Packt Publishing Ltd.* 2024. DOI: [10.0000/9781837639465-001](https://doi.org/10.0000/9781837639465-001). 438 с.

11. Hercog D., Lerher T., Truntič M., Težak O. Design and implementation of ESP32-based IoT devices. *Sensors*. 2023. Vol. 23(15). P. 6739.

12. Hunter D. Analysis of the Measurement Error From a Low-Cost Ultrasonic Sensor. *Edward Waters University Undergraduate Research Journal*. 2023. Vol. 1(1).

13. Kareem H., Dunaev D. The working principles of esp32 and analytical comparison of using low-cost microcontroller modules in embedded systems design. *International Conference on Circuits, Systems and Simulation (ICCSS)*. 2021. № 4. P. 130-135.

14. Kušević T., Blažević D., Keser T. Comparison Functionalities of HTTP and MQTT Protocols. *Cham: Springer International Publishing*. 2022. DOI: [10.1007/978-3-031-21429-5\\_5](https://doi.org/10.1007/978-3-031-21429-5_5). P. 45-56.

15. Lalhriatpuii Ruchi, Wasson V. Comprehensive Exploration of IoT Communication Protocol: CoAP, MQTT, HTTP, LoRaWAN and AMQP. *Cham: Springer Nature Switzerland*. 2024. DOI: [10.1007/978-3-031-75861-4\\_23](https://doi.org/10.1007/978-3-031-75861-4_23). P. 261-274.

16. LED (light emitting diode) Visual Led. URL: <https://visualled.com/en/glossary/led-light-emitting-diode/> (дата звернення: 29.05.2025).

17. Oner V. O.mDeveloping IoT Projects with ESP32: Automate your home or business with inexpensive Wi-Fi devices. *Packt Publishing Ltd.* 2021. Vol. 1(1). P. 1-17.

18. Passive Electronic Buzzer Piezo buzzer 12085 3V 5V 9V 12V. *Megaeshop.pk*. URL: <https://megaeshop.pk/passive-electronic-buzzer-piezo-buzzer-12085-3v-5v-9v-12v> (дата звернення: 29.05.2025).

19. Ramesh P., Sudheera S., Reddy D. V. Distance measurement using ultrasonic sensor and Arduino. *Journal of Advanced Research in Technology and Management Sciences (JARTMS)*. 2021. Vol. 3(2). P. 1-5.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 60
Зм.	Арк.	№ докум.	Підпис	Дата		

20. Single turn shaft potentiometer; 50K; L=15mm Micros. URL: <https://www.micros.com.pl/en/product/po-r9011-1-b50k115,118374.html> (дата звернення: 29.05.2025).

21. The difference between piezo buzzer and piezo transducer (Part 2). HONGCHANG(CHANGZHOU) ELECTRONIC CO., LTD. URL: <https://www.hcspeakerbuzzer.com/The-difference-between-piezo-buzzer-and-piezo-transducer-Part-2-id3791202.html> (дата звернення: 29.05.2025).

22. Ultrasonik (Ultrasonic) Sensör Nedir? Nasıl Çalışır? Ne İşe Yarar? Maker Robotistan. URL: <https://maker.robotistan.com/ultrasonic-sensor/> (дата звернення: 29.05.2025).

23. Understanding Difference between Active and Passive Buzzer and How to use it with Arduino Circuit Digest. URL: <https://circuitdigest.com/microcontroller-projects/understanding-difference-between-active-and-passive-buzzer-with-arduino> (дата звернення: 29.05.2025).

24. Vukonić L., Tomić M. Ultrasonic sensors in IoT applications. *Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*. 2022. № 45. P. 415-420.

25. Zet C., Foşalău C., Hariton A., Sârbu G. C. Improved Distance Measurement Using Ultrasonic Sensors. *International Conference on Electromechanical and Energy Systems (SIELMEN)*. 2023. № 13. P. 1-5.

26. Барибін О. І., Бражний В. В. Експлуатація типових вразливостей бездротових мереж на прикладі мікроконтролера ESP32. *Збірники наукових праць професорсько-викладацького складу ДонНУ імені Василя Стуса*. 2019. № 1. С. 148-150.

27. Барішніков С. С. Система автоматичного визначення розмірів в процесі токарної обробки деталей з використанням мікроконтролера ESP32-CAM. *Хмельницький національний університет*. дис.: квал. роб. бакал.: Хмельницький, 2024. 89 с.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 61
Зм.	Арк.	№ докум.	Підпис	Дата		

28. Гненний А. П., Гордієнко Ю. Г. Інтернет речей, як головний чинник впровадження іт-технологій на сучасному підприємстві. *Вимірjувальна та обчислювальна техніка в технологічних процесах*. 2018. № 1. С. 94-98.

29. Забігайло О., Терешко О., Панасюк І., Леськів Р. Поєднання «Інтернету речей» та «Великих даних» в проектах класу «Розумне місто». *Збірник тез ІІ Міжнародної наукової конференції молодих учених та студентів „Філософські виміри техніки “*. 2019. № 2. С. 51-51.

30. Зубков О. В. Особливості реалізації Web серверів на модулях ESP8266 ТА ESP32 У Arduino IDE. *IV форум «Автоматизація, електроніка та робототехніка. Стратегії розвитку та інноваційні технології»*. 2022. № 4. С. 54-55.

31. Клубко Д. І. МОЖЛИВОСТІ ЗАСТОСУВАННЯ ІОТ ДЛЯ ОРГАНІЗАЦІЇ ПРОЕКТНОЇ ДІЯЛЬНОСТІ ЗДОБУВАЧІВ ОСВІТИ. *Ц 75 Цифрова трансформація в освіті: виклики та перспективи*. матеріали міжнар. наук.-практ. конф., м. Київ, 15–16 квіт. 2025 р. Київ, 2025. С. 100-324.

32. Макетна плата, що не потребує пайки: принцип роботи, види, застосування BitKit. URL: <https://bitkit.com.ua/maketna-plata-shho-ne-potrebuye-pajki-princip-roboti-vidi-zastosuvannya> (дата звернення: 29.05.2025).

33. Рудий А. О. Робототехнічна система віддаленого спостереження з використанням мікроконтролера ESP32-CAM. *Хмельницький національний університет*. матеріали: квал. роб. бакал., м. Хмельницький, 2022 р. Хмельницький, 2022. С. 85.

34. Слюсарь І.І., Слюсарь, В.І. Слюсар, Р.Д. Інтеграція технологій AR і ІоТ в інтересах перспективних систем . Новітні інформаційні системи та технології. *ПолтНТУ*. 2018. № 10. С. 1-10.

35. Толочко О. О., Семенов М. А. ОСОБЛИВОСТІ ВИБОРУ АЛГОРИТМІВ ШИФРУВАННЯ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ДАНИХ В ІОТ ПРИСТРОЯХ. *«Наукові досягнення сучасного суспільства»*, видавництво Cognit. матеріали: міжнар. наук.-практ. конф., м. Лондон, Велика Британія, 15-17 серпня. 2024 р. Лондон, 2024. С. 170-491.

					КВРІСТ 2101034.21.01.08 ПЗ	Арк. 62
Зм.	Арк.	№ докум.	Підпис	Дата		

36. Щеснюк Т. Особливості застосування гнучких технологій розробки для IoT-систем. *Збірник тез доповідей XIII Міжнародної науково-практичної конференції молодих учених та студентів „Актуальні задачі сучасних технологій“*. 2024. №13. С. 409-409.

37. Що таке резистор простою мовою BitKit. URL: <https://bitkit.com.ua/rezistor> (дата звернення: 29.05.2025).

					КВРІСТ 2101034.21.01.08 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

## Додаток А (обов'язковий)

### КОПІЯ КРЕСЛЕННЯ «АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ »

Інформаційна система на базі ультразвукового датчика

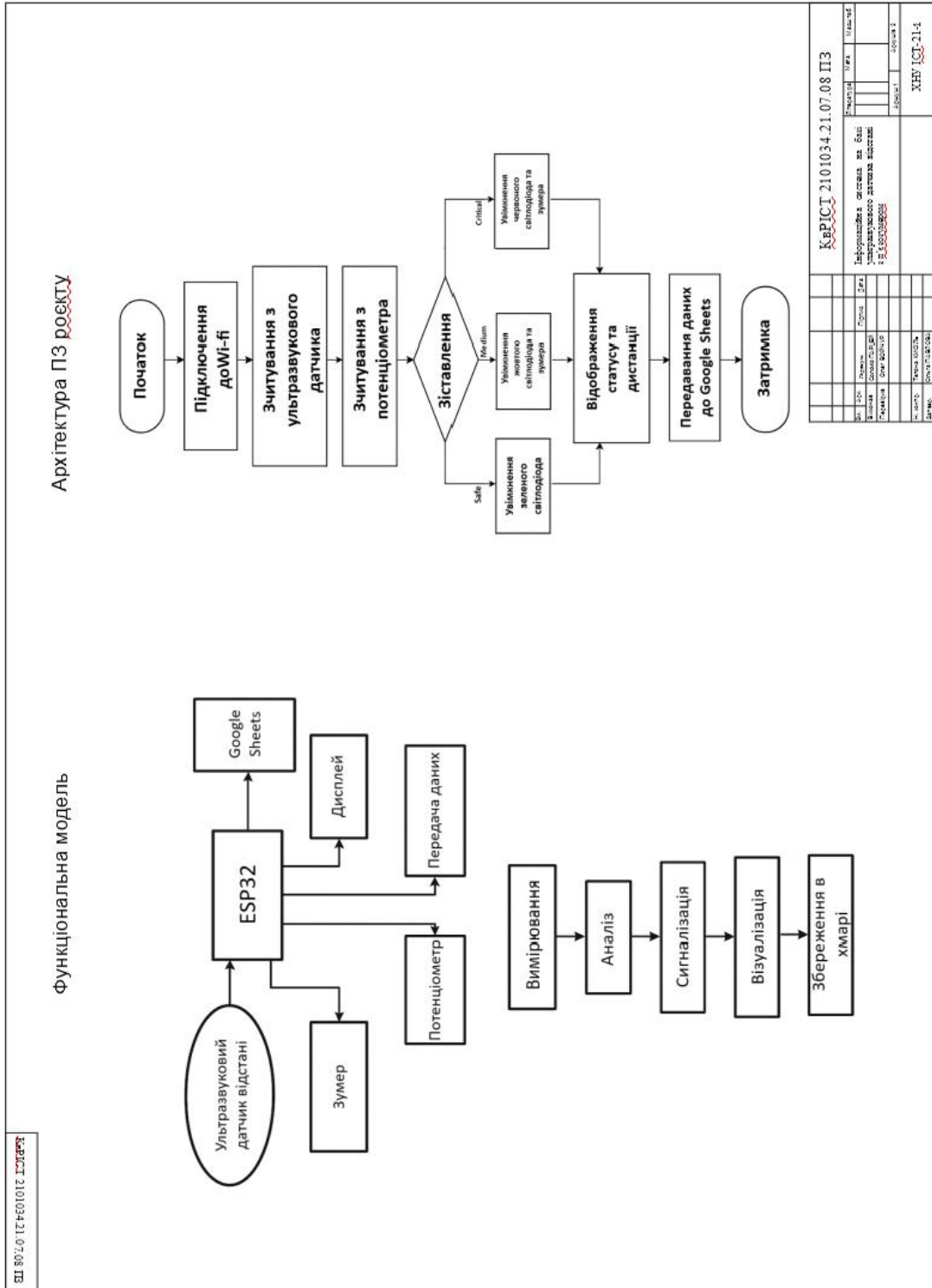
КерісТ 2101034.21.07.08 ПЗ

№	Ім'я	Посада	Дата	Відомості
1	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
2	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
3	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
4	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
5	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
6	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
7	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
8	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
9	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
10	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000
11	Сидоренко	Інженер	21.05.2025	Діагностика системи на базі ультразвукового датчика встановленої в ПЛК UC1000-4000

	A	B	C
1	Timestamp	Distance	Status
2	02.05.2025 11:51:29	52.92	Medium
3	02.05.2025 11:52:00	52.92	Medium
4	02.05.2025 11:52:57	02.04	Critical
5	02.05.2025 12:54:40	1.92	Critical
6	02.05.2025 12:55:17	256.95	Safe
7	21.05.2025 10:19:42	196.96	Safe
8	21.05.2025 10:20:16	197.01	Safe
9	21.05.2025 10:20:30	197.00	Safe
10	21.05.2025 10:20:53	37.99	Critical
11	21.05.2025 10:21:07	48.93	Critical

**Додаток Б**  
(обов'язковий)

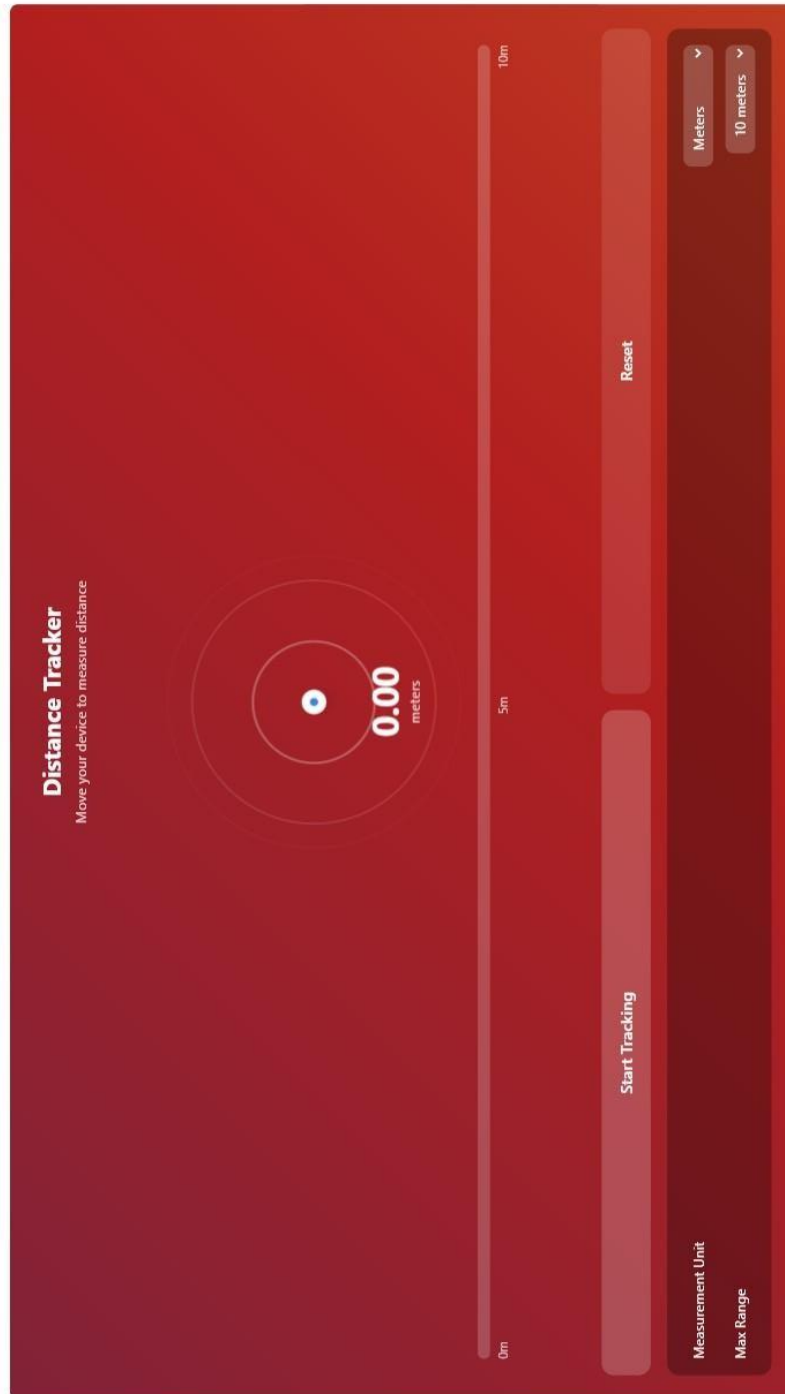
**КОПІЯ КРЕСЛЕННЯ «АРХІТЕКТУРА ПЗ ДЛЯ  
ІНФОРМАЦІЙНОЇ СИСТЕМИ»**





## Додаток Г (обов'язковий)

"Інтерфейс застосунку для виміру дистанції"



## Додаток Д (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
="AKfyxcbx225azA6J3tSylwpFpUOpEwrj6XDV4Gy3RqJ-6tw64PNb4PyVoPIo5u-
NaLb-TR3-lrQ";

#define TRIG_PIN 5
#define ECHO_PIN 18

#define GREEN_LED 4
#define YELLOW_LED 0
#define RED_LED 2

#define BUZZER_PIN 23
#define POT_PIN 32

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

void setup() {
  Serial.begin(115200);
```

```

    lcd.init();
    lcd.backlight();
    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);

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

    pinMode(BUZZER_PIN, OUTPUT);

    connectToWiFi();

}

float readDistanceCM() {
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);

    long duration = pulseIn(ECHO_PIN, HIGH);
    return duration * 0.034 / 2;
}

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() {
    float distance = readDistanceCM();
    int potValue = analogRead(POT_PIN);

    // Динамічне налаштування порогів з потенціометра:
    // Safe > highThreshold
    // Medium = lowThreshold .. highThreshold
    // Critical < lowThreshold

    int highThreshold = map(potValue, 0, 4095, 100, 150); // верхній
поріг
    int lowThreshold = map(potValue, 0, 4095, 50, 80); // нижній
поріг

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Dist: ");
    lcd.print(distance);
    lcd.print("cm");

    if (distance > highThreshold) {
        digitalWrite(GREEN_LED, HIGH);
        digitalWrite(YELLOW_LED, LOW);
        digitalWrite(RED_LED, LOW);
        noTone(BUZZER_PIN);
        lcd.setCursor(0, 1);
        lcd.print("Status: Safe");
        status = "Safe";
    }
    else if (distance > lowThreshold) {
        digitalWrite(GREEN_LED, LOW);

```

```

    digitalWrite(YELLOW_LED, HIGH);
    digitalWrite(REDA_LED, LOW);
    tone(BUZZER_PIN, 1000, 200);
    lcd.setCursor(0, 1);
    lcd.print("Status: Medium");
    status = "Medium";
}
else {
    digitalWrite(GREEN_LED, LOW);
    digitalWrite(YELLOW_LED, LOW);
    digitalWrite(REDA_LED, HIGH);
    tone(BUZZER_PIN, 2000, 500);
    lcd.setCursor(0, 1);
    lcd.print("Status: Critical");
    status = "Critical";
}

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(distance) +
"&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");
}
```

**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 на базі ультразвукового датчика відстані з п'єзозумером

Автор: Соломія Парцей

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, загальноновживаними або мають відповідне посилання на джерело, де це доцільно з методичної точки зору;
- 3) окремі виявлені збіги є загальноновживаними 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 2.85% 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

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

Сооломія ПАРЦЕЙ

ПІБ 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



РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Парцей Соломія Михайлівна

Тема: Інформаційна система на базі ультразвукового датчика відстані з п'єзозумером

Спеціальність: 126 «Інформаційні системи та технології»

Обсяг кваліфікаційної роботи:

Кількість листів креслень   3   Кількість сторінок записки   58  

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 та передових методів роботи

У першому розділі кваліфікаційної роботи було проведено ґрунтовне дослідження предметної області. Розглянуто загальні поняття інформаційних систем та специфіку їх побудови з використанням ультразвукових сенсорів. Окрему увагу приділено сучасним підходам до дистанційного вимірювання відстані, перевагам ультразвукових технологій та їх актуальності у контексті розумних систем. Виконано постановку задачі дослідження з урахуванням реальних потреб у безконтактному моніторингу об'єктів та передачі даних у хмарне середовище.

У другому розділі детально проаналізовано апаратну частину системи. Розглянуто технічні характеристики та принципи роботи основних компонентів: мікроконтролера ESP32, ультразвукового датчика HC-SR04, пасивного п'єзозумера, LCD-дисплея з інтерфейсом I2C, потенціометра, світлодіодів, макетної плати та допоміжних елементів. Пояснено вибір саме ESP32 як сучасного, енергоефективного рішення для IoT-систем, що підтримує бездротову передачу даних. Структуровано схему взаємодії компонентів у межах апаратної реалізації системи.

У третьому розділі описано підключення апаратних модулів до мікроконтролера та реалізацію функціональної взаємодії між ними. Здійснено програмну реалізацію зчитування даних з сенсорів та керування виконавчими елементами на основі Arduino-бібліотек. Система протестована у середовищі Wokwi, що дозволило виконати налагодження без використання фізичних компонентів. Важливою складовою є реалізація передачі зібраних даних у хмарне середовище: ESP32 підключено до Інтернету через Wi-Fi, а для надсилання показників використано HTTP-запити до Google Apps Script з подальшим збереженням даних у Google Sheets. Такий підхід демонструє ефективне застосування сучасних засобів інтеграції мікроконтролерів з веб-сервісами. 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.

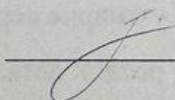
8. Інші зауваження: \_\_\_\_\_

9. Оцінка дипломної роботи: відмінно

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

*Завідувач кафедри інженерно програмного забезпечення*  
*професор Федратюк Леонід Петрович*

"06" 06 2025 р.

 (підпис)

## Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 3.0%

Dictionaries check: en\_US, ru\_RU, ua\_UA. Errors in the documents: 10%

ID: 242739 Title: БКР Інформаційна система на базі ультразвукового датчика відстані з п'єзозуметром Added in a DB: 2025-06-02 Authors: Соломія ПАРЦЕЙ Heads: Олег ВОЙЧУР Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	81454	653	3518 (4%)	33 (5%)

### Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

## 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:** Соломія ПАРЦЕЙ

**Співавтор:**

**Назва:** Парцей\_Інформаційна система на базі ультразвукового датчика відстані з п'єзозуметром

**Експерт:**

**Підрозділ:** Кафедра комп'ютерної інженерії та інформаційних систем

**Коефіцієнт подібності 1:** 2.8%

**Коефіцієнт подібності 2:** 1.1%

**Мікропробіли:** 7

**Заміна букв:** 1

**Інтервали:** 0

**Білі знаки:** 0

**Дата створення звіту:** 2025-06-02 08:35:03.0

**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:**

2025-06-02

Дата



Доцент Андрій Нічепорук

експерт