

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: *Комп'ютеризована система контролю рівня шуму в
громадських місцях*

Виконав: студент 4 курсу, групи СІ-42
спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

Ханевич В.В.
(підпис) (прізвище та ініціали)

Керівник Шингера Н.Я.
(підпис) (прізвище та ініціали)

Нормоконтроль Тили Є.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри Осухівська Г.М.
(підпис) (прізвище та ініціали)

Рецензент
(підпис) (прізвище та ініціали)

Тернопіль
2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
«27» січня 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студенту Ханевичу Віталію Васильовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютеризована система контролю рівня шуму в громадських місцях

Керівник роботи Шингера Наталія Ярославівна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 27 » січня 2025 року № 4/7-53

2. Термін подання студентом завершеної роботи 20.06.2025

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз технічного завдання

2. Проєктна частина

3. Практична частина

4. Безпека життєдіяльності, основи охорони праці.

Висновки

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

1. Структурна схема системи

2. Схема електрична принципова

3. Модель тестового макету

4. Блок-схема алгоритму роботи системи

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Безпека життєдіяльності, основи охорони праці</i>	<i>д.т.н., професор Пилипець М.І.</i>		

7. Дата видачі завдання 27.01.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
	<i>Розробка технічного завдання</i>	<i>27.01 – 02.02</i>	<i>Викон.</i>
	<i>Аналіз технічного завдання та обґрунтування можливих рішень</i>	<i>10.04 – 15.04</i>	<i>Викон.</i>
	<i>Розробка структурної та електричної принципової схем</i>	<i>17.04 – 21.04</i>	<i>Викон.</i>
	<i>Розробка програмного забезпечення для проектованої системи</i>	<i>23.04 – 08.05</i>	<i>Викон.</i>
	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>10.05 – 13.05</i>	<i>Викон.</i>
	<i>Оформлення пояснювальної записки і графічного матеріалу</i>	<i>16.05 – 04.06</i>	<i>Викон.</i>
	<i>Перевірка на академічний плагіат, перевірка керівником та консультантами</i>	<i>9.06 – 15.06</i>	<i>Викон.</i>
	<i>Попередній захист кваліфікаційної роботи бакалавра</i>	<i>16.06 – 22.06</i>	<i>Викон.</i>
	<i>Захист кваліфікаційної роботи бакалавра</i>	<i>27.06.2025</i>	<i>Викон.</i>

Студент

(підпис)*Ханевич В.В.*

(прізвище та ініціали)

Керівник роботи

(підпис)*Шингера Н.Я.*

(прізвище та ініціали)

АНОТАЦІЯ

Ханевич В.В. Комп'ютеризована система контролю рівня шуму в громадських місцях: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 — комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2025.

Ключові слова: комп'ютеризована система, шумове забруднення, мікроконтролер, Arduino Uno, моніторинг, передача даних.

Кваліфікаційна робота присвячена розробці комп'ютеризованої системи контролю рівня шуму в громадських місцях. Метою роботи є створення системи, що дозволяє в режимі реального часу здійснювати моніторинг шумового забруднення, реагувати на перевищення встановлених норм та передавати дані для подальшого аналізу. В ході виконання роботи проаналізовано технічне завдання, сформульовано вимоги до системи, розглянуто існуючі рішення та обґрунтовано вибір компонентів. Розроблено структурну та принципову електричну схеми, описано алгоритм роботи, реалізовано програмне забезпечення для збору, відображення та передачі даних, а також створено веб-інтерфейс для візуалізації рівня шуму. Після тестування було підтверджено стабільність, надійність та ефективність запропонованої системи для застосування в умовах громадських місць.

ANNOTATION

Khanevych V.V. Computerized system for noise level monitoring in public places: Bachelor's Graduation Thesis: speciality 123 — computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2025.

Keywords: computerized system, noise pollution, microcontroller, Arduino Uno, monitoring, data transfer.

The qualification work is devoted to the development of a computerized system for noise level control in public places. The purpose of the work is to create a system that allows real-time monitoring of noise pollution, to respond to exceeding the established norms and to transfer data for further analysis. In the course of the work, the technical task was analyzed, the requirements for the system were formulated, existing solutions were considered and the choice of components was justified. A structural and schematic electrical diagram was developed, the operating algorithm was described, software for data collection, display and transmission was implemented, and a web interface for noise level visualization was created. After testing, the stability, reliability and efficiency of the proposed system for use in public places were confirmed.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ	9
1.1 Аналіз основних вимог комп'ютеризованої системи	10
1.2 Огляд існуючих засобів моніторингу шуму	11
1.2.1 Класифікація систем контролю шуму	11
1.2.2 Датчик шуму RK300-06	13
1.2.3 Датчик шуму LiveNoise LN2-SE	14
1.2.4 Датчик шуму LoRaWAN IOT-S500NOIS.....	15
1.2.5 Результат порівняльного аналізу засобів контролю рівня шуму	17
1.3 Аналіз можливих рішень поставленого завдання.....	18
1.4 Огляд інструментів для симуляції та розробки (Wokwi).....	19
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА	21
2.1 Розробка структури системи контролю шуму в громадських місцях.....	21
2.2 Обґрунтування вибору апаратного забезпечення системи.....	22
2.2.1 Вибір мікроконтролера Arduino Uno.....	22
2.2.2 Вибір датчика шуму MAX9814.....	25
2.2.3 Вибір Wi-Fi модуля ESP8266-01S.....	26
2.2.4 Вибір динаміка Active Buzzer Module.....	28
2.2.5 Вибір LCD-дисплея 16x2 з інтерфейсом I2C	30
2.3 Опис шин обміну даними	32
2.3.1 Опис шини даних I2C	32

					КС КРБ 123.241.00.00 ПЗ						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Ханевич В.В.			Комп'ютеризована система контролю рівня шуму в громадських місцях			Лім.	Арк.	Аркушів	
Перевір.		Шингера Н.Я.								6	78
Реценз.								ТНТУ, каф. КС, гр. CI-42			
Н. Контр.		Тиш Є.В.									
Затверд.		Осухівська Г.М.									

2.3.2	Опис шини даних UART	33
2.4	Опис електрично-принципової схеми системи	34
2.4.1	Опис середовища EasyEDA.....	34
2.4.2	Опис розробленої електричної схеми	36
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА		38
3.1	Розробка алгоритму роботи системи контролю шуму в громадських місцях	38
3.2	Створення схеми з'єднань в середовищі Wokwi.....	40
3.3	Розробка програмного забезпечення	43
3.3.1	Написання коду для функціонування системи.....	43
3.3.2	Створення веб-сервера для отримання даних	48
3.4	Тестування програми	51
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ		55
4.1	Вплив шуму на організм людини та розробка заходів щодо його зниженню до допустимих величин для обладнання.....	55
4.2	Захист електрообладнання від короткого замикання та перенавантаження .	56
ВИСНОВОК.....		59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....		60
Додаток А Технічне завдання		
Додаток Б Перелік елементів		
Додаток В Код програми контролю рівня шуму		

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

ВСТУП

В сучасних умовах зростання урбанізації та активного використання громадських місць (вулиці, вокзали, освітні заклади) проблема шумового забруднення набуває особливого значення. Наукові дослідження підтверджують, що систематичний вплив шуму на людину спричиняє зниження продуктивності, збільшення ризику розвитку стресових станів, а також несприятливого впливу на серцево-судинну та нервову системи. Саме тому виникає нагальна потреба у створенні ефективних засобів моніторингу рівня шуму в реальному часі.

Ефективним рішенням даної проблеми є створення комп'ютеризованої системи контролю, яка в режимі реального часу слідкує за параметрами шуму в громадських місцях. Завдяки розвитку вбудованих технологій та мікроконтролерів, таких як Arduino, виникла можливість реалізовувати доступні, надійні та автономні системи контролю, які можна інтегрувати в різних громадських місцях.

Метою кваліфікаційної роботи є розробка комп'ютеризованої системи контролю рівня шуму в громадських місцях на базі платформи Arduino Uno, яка повинна визначати рівень шуму і передавати дані в відповідні органи.

Для реалізації поставленої мети були вирішенні такі завдання:

- проаналізувати існуючі методи та системи контролю шуму;
- розробити структурну схему комп'ютеризованої системи;
- обґрунтувати вибір апаратної платформи, сенсорів та допоміжних компонентів;
- реалізувати алгоритми та програмне забезпечення для постійного контролю шуму в громадських місцях;
- провести тестування реалізованої системи для експлуатації її в реальних умовах.

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

Отже, комп'ютеризована система спрямована на підвищення якості акустичного середовища у громадських місцях за допомогою постійного контролю шуму [1].

					КС КРБ 123.241.00.00 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз основних вимог комп'ютеризованої системи

В умовах зростання урбанізації та збільшення кількості людей у громадських місцях, таких як торгово-розважальні центри, вокзали, парки, освітні заклади тощо, актуальним стає питання контролю рівня шуму, який безпосередньо впливає на фізичне та психоемоційне здоров'я людей. Надмірний шум може призводити до втоми, зниження працездатності, підвищеної тривожності, а в довгостроковій перспективі — серйозні порушення в роботі серцево-судинної та нервової систем.

З огляду на це, важливим є створення комп'ютеризованої системи, котра дає змогу здійснювати безперервний моніторинг рівня шуму у громадських місцях та виявляти перевищення допустимих норм [2]. Система повинна забезпечувати точне та своєчасне вимірювання рівня шуму в децибелах, а також фіксувати моменти, коли ці показники перевищують встановлені санітарні або нормативні межі.

Окрему увагу під час розробки такої системи варто приділити чутливості та точності датчиків, котрі мають фіксувати навіть малі зміни в рівні шуму, не реагуючи на випадкові перешкоди або короточасні сигнали. Надійність функціонування системи теж має велике значення — система повинна функціонувати стабільно в умовах можливих зовнішніх перешкод, змін температури, підвищеної вологості або пилу.

Ще однією важливою вимогою є енергоефективність системи, особливо в умовах її розміщення у важкодоступних або відкритих просторах. Система повинна мати можливість автономної роботи протягом тривалого часу без необхідності частого заряджання або заміни джерел живлення.

					КС КРБ 123.241.00.00 ПЗ					
Змн.	Арк.	№ докум.	Підпис	Дата	Аналіз технічного завдання			Літ.	Арк.	Аркушів
Розроб.		Ханевич В.В.								
Перевір.		Шингера Н.Я.							10	78
Реценз.								ТНТУ, каф. КС, гр. СІ-42		
Н. Контр.		Тиш Є.В.								
Затверд.		Осухівська Г.М.								

Крім того, система має забезпечувати постійне відображення поточного рівня шуму в децибелах на моніторі для відповідальних осіб, що дасть змогу миттєво оцінювати ситуацію в режимі реального часу

У разі перевищення встановлених норм система повинна автоматично сповіщати про це, щоб відповідальні особи мали можливість вжити необхідних дій для зменшення шумового навантаження.

Таким чином, комп'ютеризована система контролю рівня шуму в громадських місцях повинна бути високочутливою, точною, стабільною та здатною до автономної енергоефективної роботи. Вона має вчасно розпізнавати перевищення допустимих норм шуму та сприяти підвищенню рівня комфорту, безпеки і здоров'я відвідувачів у публічному просторі [3].

1.2 Огляд існуючих засобів моніторингу шуму

1.2.1 Класифікація систем контролю шуму

Системи контролю шуму в громадських місцях мають велика значення для створення зручного акустичного оточення для людей. Їх застосовують у таких об'єктах, як навчальні заклади, офісні приміщення, лікарні, вокзали, аеропорти, торговельно-розважальні центри та інші місця з підвищеною відвідуваністю. Розрізняють декілька основних видів систем контролю шуму, які відрізняються за способом виявлення шуму, ступенем інтеграції, типом оповіщення та складністю обробки отриманої інформації.

Системи, що використовують мікрофонні сенсори є найпоширенішими. Ці пристрої вимірюють звуковий тиск у децибелах, що дає можливість швидко виявляти випадки перевищення допустимого рівня шуму. Зазвичай вони використовують електретні або MEMS-мікрофони, які мають високу чутливість і компактний розмір. Інформація з цих сенсорів може зчитуватися в реальному часі, зберігатись на носії даних або передаватись на сервер для детальнішого огляду.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

Системи з інтегрованою обробкою цифрових сигналів (DSP) дають змогу не тільки вимірювати рівень гучності, а й проводити аналіз спектру шуму. Вони здатні виявляти характерні частоти, встановлювати джерела шуму або розпізнавати типи звукових сигналів (наприклад, голос, музика, механічні шуми).

Ці системи знаходять застосування в умовах з підвищеними вимогами, де потрібно розрізняти різні типи шумів та реагувати лише на ті, що вважаються небажаними або перевищують встановлені норми.

Інтелектуальні системи, що використовують штучний інтелект (AI), інтегрують модулі машинного навчання[4-7]. Ці модулі навчаються розпізнавати специфічні патерни шумів. Наприклад, це можуть бути гучні голоси в бібліотеках, крики в школах або шум двигунів у житлових зонах. Такі системи мають змогу адаптувати свої алгоритми залежно від часу доби, типу об'єкта чи попередніх інцидентів. Вони також здатні автоматично сповіщати відповідальних осіб через мобільний додаток або систему моніторингу.

Автономні системи контролю шуму функціонують не потребуючи підключення до інтернету чи центрального сервера. Вони обладнані локальними індикаторами, що можуть бути світловими або звуковими, які реагують на перевищення заданого рівня шуму. Ці системи показують свою ефективність в класах, кабінетах або інших приміщеннях, де доступ до централізованої інфраструктури обмежений.

Хмарні та мережеві системи контролю шуму збирають дані з багатьох пристроїв, обробляють їх централізовано та відображають на інформаційних панелях. Завдяки підключенню до Інтернету речей (IoT), ці системи можуть масштабуватися, аналізувати розподіл шуму в просторі, створювати карти шумового забруднення та інтегруватися з іншими інженерними системами будівель.

Отже, класифікація систем контролю шуму охоплює розмаїття технологій — від простих автономних пристроїв до складних інтелектуальних мереж. Кожен різновид має свої переваги й обмеження, що визначають вибір системи на умови використання, необхідний рівень точності, інтеграцію та доступний бюджет.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2.2 Датчик шуму RK300-06

Датчик шуму RK300-06 може розглядатися як ймовірний аналог запланованої комп'ютеризованої системи контролю шуму в громадських місцях.

Він розроблений для безперервного моніторингу рівня акустичного тиску в навколишньому середовищі та може бути розміщений як у приміщеннях, так і на відкритих територіях завдяки герметичному корпусу, що захищає від вологи та пилу (рис. 1.1).



Рисунок 1.1 – Датчик шуму RK300-06

RK300-06 проводить вимірювання у широкому діапазоні – від 30 дБ до 130 дБ, із звичайною похибкою, яка не перевищує $\pm 1,5$ дБ. У приладі використано вбудований електретний мікрофон високої чутливості. Це забезпечує надійне виявлення навіть незначних коливань шуму. Виводити дані можна через інтерфейси RS485 (Modbus RTU), 4–20 мА або 0–10 В, що сприяє легкій інтеграції пристрою у більшість сучасних систем моніторингу.

Ключовою перевагою приладу є надійна робота в екстремальних умовах, зокрема при температурі від -30°C до $+70^{\circ}\text{C}$ та вологості до 100 % RH. Завдяки універсальним комунікаційним можливостям і високій точності вимірювань, датчик ефективно застосовується в промислових та муніципальних системах контролю шуму.

Проте недоліком RK300-06 є потреба в зовнішньому джерелі енергії, а також відсутність можливості автономної роботи або збереження інформації на

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

пристрої. До того ж, апарат не обладнаний засобами візуального чи звукового оповіщення, що знижує його ефективність при використанні без інтеграції у комплексну систему [8].

Загалом, датчик шуму RK300-06 представляє собою точний і надійний інструмент для збору акустичних даних, що чудово підходить для інтеграції в системи моніторингу шуму в громадських місцях. Його конструкція та функціональність повністю відповідають стандартам щодо точності, безпеки та стабільності вимірювань у складних умовах експлуатації.

1.2.3 Датчик шуму LiveNoise LN2-SE

Датчик шуму LiveNoise LN2-SE є зразком високоточного обладнання для моніторингу шумів, що може використовуватись як прототип або аналог при розробці комп'ютеризованої системи контролю шуму в громадських місцях. Пристрій розроблений для безперервного спостереження за рівнем звуку в реальному часі, з функцією передавання даних через мережу Ethernet або Wi-Fi (рис. 1.2).



Рисунок 1.2 – Датчик шуму LiveNoise LN2-SE

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

LiveNoise LN2-SE оснащений інтегрованим звуковим аналізатором класу 1, який відповідає міжнародним стандартам ІЕС 61672, що стосуються вимірювальних мікрофонів. Прилад забезпечує вимірювання рівнів шумів в широкому діапазоні – від 20 дБ до 140 дБ, характеризуючись при цьому високою точністю. Завдяки таким характеристикам, його можна застосовувати в різних умовах: в офісних приміщеннях, навчальних закладах, на підприємствах та міських середовищах, з метою контролю за дотриманням встановлених норм шуму [9].

Основною перевагою датчика є здатність відправляти отримані дані на центральний сервер чи у хмару, позбавляючи потреби в ручному зчитуванні показників. Крім того, прилад передбачає функції сповіщення у випадку, якщо рівень шуму перевищує задані межі – наприклад, надсилаючи повідомлення електронною поштою або через API.

Серед недоліків слід відзначити досить велику ціну самого приладу. Також існує потреба в надійному інтернет-з'єднанні для забезпечення повної функціональності.

Загалом, LiveNoise LN2-SE – це ефективний та професійний прилад для постійного контролю шуму, який поєднує у собі високу точність вимірювань, автоматизований збір інформації та зручність у використанні. Його функціонал може стати у нагоді при створенні більш доступної системи, розрахованої на місцеве або мобільне застосування в громадських місцях.

1.2.4 Датчик шуму LoRaWAN IOT-S500NOIS

Датчик шуму LoRaWAN IOT-S500NOIS – це сучасний пристрій для моніторингу звукового середовища. Сенсор використовує бездротовий зв'язок LoRaWAN, забезпечуючи передачу даних на великі відстані з низьким енергоспоживанням (рис. 1.3).

					КС КРБ 123.241.00.00 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 1.3 - Датчик шуму LoRaWAN IOT-S500NOIS

Датчик вимірює рівень шуму в режимі реального часу в діапазоні від 30 до 120 дБ, з точністю ± 3 дБ, що робить його придатним для використання у освітніх закладах, лікарнях, бібліотеках, офісних приміщеннях, а також у громадських місцях міст. Завдяки можливості підключення до LoRaWAN зібрана інформація може передаватися на шлюз або централізований сервер навіть віддалених зон або в місць зі складною архітектурою, де інші бездротові технології обмежені у функціоналі.

Прилад має вмонтовану батарею великої ємності, що гарантує автономну роботу протягом декількох років без потреби в частому обслуговуванні. Його корпус сертифіковано за стандартом IP65, що дозволяє розміщувати пристрій як усередині будівель, так і на вулиці, гарантуючи безперебійну роботу навіть у складних погодних умовах [10].

Втім, як і більшість приладів, що застосовують LoRaWAN, даний датчик вимагає наявності відповідної інфраструктури (шлюзів), а також початкового налаштування для інтеграції з сервером збору даних або системою сповіщення.

Підсумовуючи, датчик шуму LoRaWAN IOT-S500NOIS – це енергоощадне, просте в установці та надійне рішення для постійного моніторингу шуму. Його можливості можуть бути основою для створення більш доступної чи адаптованої системи контролю шуму у громадських місцях.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2.5 Результат порівняльного аналізу засобів контролю рівня шуму

Результати порівняльного аналізу сучасних приладів контролю рівня шуму показують їхню значну варіативність з огляду на функціонал, спосіб передачі даних, умови використання та рівень інтеграції в інтелектуальні системи. Для аналізу було розглянуто такі пристрої, як RK300-06, NoiseMeters LN2-SE та LoRaWAN IOT-S500NOIS, які відрізняються конструкцією та технічними характеристиками.

Одним з важливих параметрів є точність та діапазон вимірювання шуму. Усі три прилади показують досить високу точність ($\pm 1-3$ дБ) та проводять виміри рівнів шуму в широкому діапазоні (від 20 до 140 дБ), що дає можливість використовувати їх у різноманітних місцях – від внутрішніх приміщень до промислових зон.

Ще одним ключовим критерієм є можливість дистанційного моніторингу та методи обміну даними. Датчик рівня шуму LoRaWAN IOT-S500NOIS забезпечує передачу даних через LoRaWAN, що є практичним рішенням для віддалених місцевостей з низькою щільністю інфраструктури. RK300-06 підтримує типові цифрові інтерфейси, забезпечуючи простоту інтеграції з наявними системами автоматизації, у той час як NoiseMeters LN2-SE має локальний дисплей і USB-підключення для аналізу на місці або передачі даних на ПК.

Також важливим аспектом є вартість і масштабованість. Датчик LoRaWAN вирізняється чудовою масштабованістю, дозволяючи об'єднувати велику кількість пристроїв у мережу з мінімальними витратами на інфраструктуру. RK300-06, завдяки промислому рівню захисту, може працювати у складних умовах, однак його впровадження потребує фінансових ресурсів. LNT-SE краще підходить для індивідуального чи локального використання, проте пропонує зручний аналіз даних безпосередньо на місці.

Загалом, висновки порівняльного аналізу вказують, що вибір конкретного рішення для керування шумом зумовлено цілями системи, умовами експлуатації і доступним фінансуванням.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

1.3 Аналіз можливих рішень поставленого завдання

Розробка комп'ютеризованої системи контролю рівня шуму в громадських місцях або закритих приміщеннях вимагає дослідження доступних технічних варіантів, що гарантують точність вимірювання, стабільність функціонування та можливість взаємодії з системами сповіщення й передачі даних. Враховуючи завдання проекту, необхідно обґрунтувати вибір сенсора шуму, обчислювальної платформи, методів виводу та передачі інформації.

У рамках обраного рішення, за основу взято платформу Arduino. Це один з найпоширеніших мікроконтролерів, який відзначається простотою в роботі, підтримкою різноманітних сенсорів та великою спільнотою розробників. За допомогою Arduino можливо зчитувати аналогові або цифрові показники з сенсорів шуму, обробляти ці сигнали та керувати зовнішніми пристроями — наприклад, світловою або звуковою сигналізацією.

В якості сенсора було обрано мікрофонний модуль типу MAX9814, що забезпечує автоматичне посилення сигналу (AGC), має високу чутливість та здатен реагувати на перевищення рівня шуму. Завдяки цьому модулю Arduino може ефективно визначати зміни гучності, а також та фіксувати моменти, коли значення гучності перетинають задані порогові рівні.

Для віддаленої передачі даних про рівень шуму або сповіщень, Arduino можна спорядити WiFi-модулем (ESP8266, ESP32). Це забезпечить зв'язок з сервером чи хмарною сервісом. Таким чином можна реалізувати інтеграцію з Telegram-ботами, мобільними додатками або системами централізованого контролю.

Програмне забезпечення проекту, адаптоване для Arduino, дає змогу фіксувати та обробляти звукові коливання, визначати рівень шуму у децибелах та передавати ці показники у вигляді графіків. За потреби, зібрані дані можуть бути передані для зберігання та візуалізації на віддаленому сервері або локальній базі даних.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, здійснений аналіз показує, що Arduino – це слушна платформа для втілення комп'ютеризованої системи контролю шуму в громадських місцях, завдяки своїй простоті у використанні, можливості адаптації, великій кількості сенсорів та модулів, що з нею взаємодіють. Поєднання мікрофонного сенсора, можливості сповіщення та модуля бездротового зв'язку дозволяє розробити функціональну, масштабовану та доступну за ціною систему, яка підійде як для локального, так і для віддаленого контролю шуму у визначеному просторі.

1.4 Огляд інструментів для симуляції та розробки (Wokwi)

Під час розробки комп'ютеризованих систем важливою частиною є перевірка та симуляція програмного забезпечення і апаратних складових фізичним втіленням системи. Такий підхід дозволяє зменшити витрати часу й ресурсів, виявити ймовірні помилки на ранніх стадіях та оптимізувати процес розробки. Одним із поширених та доступних інструментів для симуляції та розробки вбудованих систем є Wokwi (рис. 1.4).

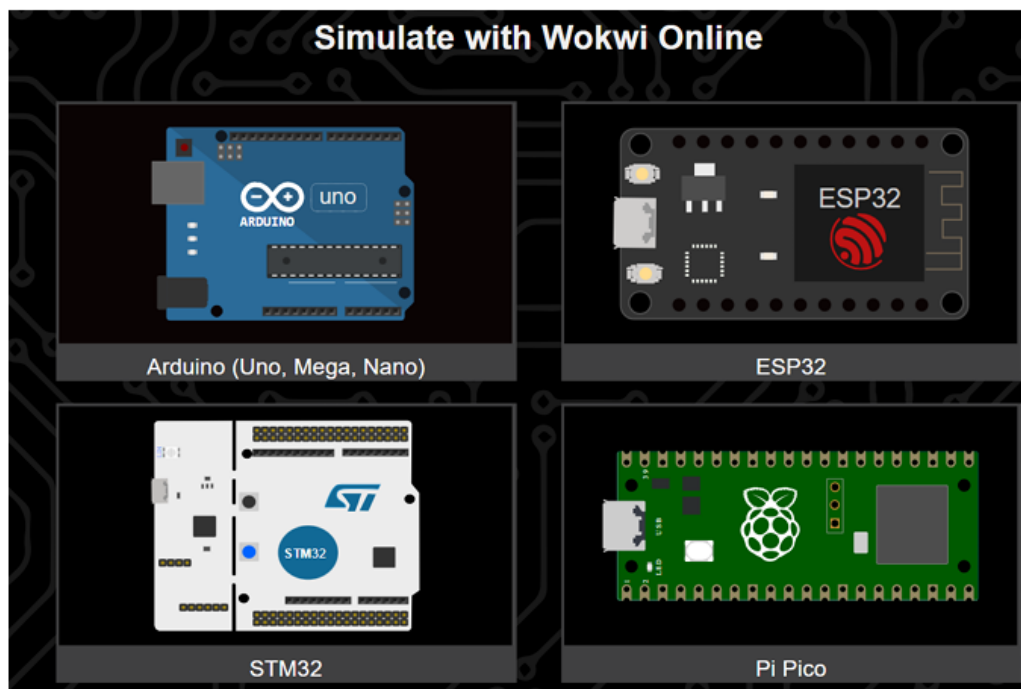


Рисунок 1.4 – Wokwi

Wokwi — це онлайн-середовище, призначене для моделювання та розробки комп'ютеризованих систем. Воно дає змогу користувачам легко розробляти, тестувати та виправляти свої проекти, не маючи потреби у фізичних компонентах. Платформа має підтримку багатьох мікроконтролерів, сенсорів та інтерфейсів, що робить її ефективним засобом для навчання, досліджень та розробки.

Завдяки значному обсягу підтримуваних складників і модулів, Wokwi надає широкий набір компонентів для проведення симуляцій. Це включає в себе не тільки мікроконтролери, а й різноманітні датчики (температури, вологості, звуку), дисплеї, кнопки, реле та багато інших пристроїв. Усе це дає змогу створювати реалістичні моделі для будь-яких вбудованих систем.

Wokwi – це потужний та інтуїтивно зрозумілий засіб для проектування та симуляції комп'ютеризованих систем, що є особливо корисним на етапі проектування і відлагодження. Вона дозволяє випробовувати алгоритми, взаємодіяти з різноманітними компонентами та інтегрувати програми в реальному часі, що є великим плюсом для розробників. Платформа ідеально підходить для навчальних цілей, для оперативного тестування ідей та прототипів, а також для комерційних проєктів, що потребують швидкої та економічно вигідної розробки.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Розробка структури системи контролю шуму в громадських місцях

Структурна схема є ключовим елементом, що відображає всі основні компоненти та їх взаємозв'язки (рис. 2.1). Основою системи є мікроконтролер Arduino, який виступає центральним керуючим елементом.

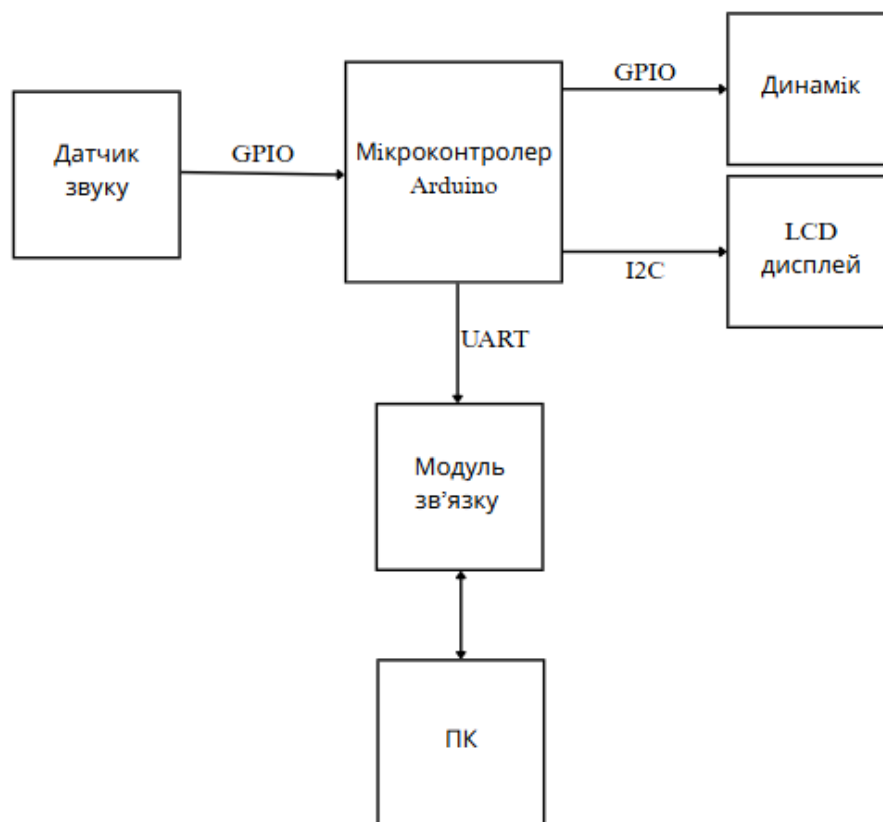


Рисунок 2.1 – Структурна схема системи

Система складається з основних компонентів, що забезпечують її функціональність:

					КС КРБ 123.241.00.00 ПЗ						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Ханевич В.В.			Проектна частина			Літ.	Арк.	Аркушів	
Перевір.		Шингера Н.Я.								21	78
Реценз.								ТНТУ, каф. КС, гр. СІ-42			
Н. Контр.		Тиш Є.В.									
Затверд.		Осухівська Г.М.									

- мікроконтролер Arduino UNO;
- датчик звуку ;
- модуль зв'язку ESP8266 Wi-Fi;
- LCD-дисплей 16x2 (з інтерфейсом I2C);
- динамік (зумер);
- персональний комп'ютер.

Датчик звуку підключається до мікроконтролера Arduino за допомогою аналогового входу. Цей датчик визначає рівень шуму у навколишньому середовищі та надсилає відповідний аналоговий сигнал на Arduino.

Мікроконтролер Arduino UNO аналізує отриманий сигнал і визначає, що робити далі. Якщо рівень шуму перевищує допустиму межу, Arduino активує динамік (зумер), що забезпечує місцеве звукове сповіщення. Водночас на LCD-дисплеї виводиться інформація про поточний рівень шуму та статус підключення до Wi-Fi.

Обробивши дані, мікроконтролер відправляє інформацію через UART-інтерфейс на модуль Wi-Fi ESP8266. Модуль встановлює з'єднання з Wi-Fi-мережею та передає дані на віддалений комп'ютер, що виконує роль центру моніторингу. Це дає змогу відповідальним особам швидко реагувати на перевищення допустимого рівня шуму.

Цей метод дає змогу створити просту систему контролю рівня шуму, яка здатна передавати дані бездротовим зв'язком.

2.2 Обґрунтування вибору апаратного забезпечення системи

2.2.1 Вибір мікроконтролера Arduino Uno

Arduino Uno — це мікроконтролерна плата, заснована на мікроконтролері ATmega328P (рис. 2.2). Ця платформа часто обирається для створення прототипів та реалізації комп'ютерних систем, включаючи систему контролю шуму в громадських місцях [11, 12].

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

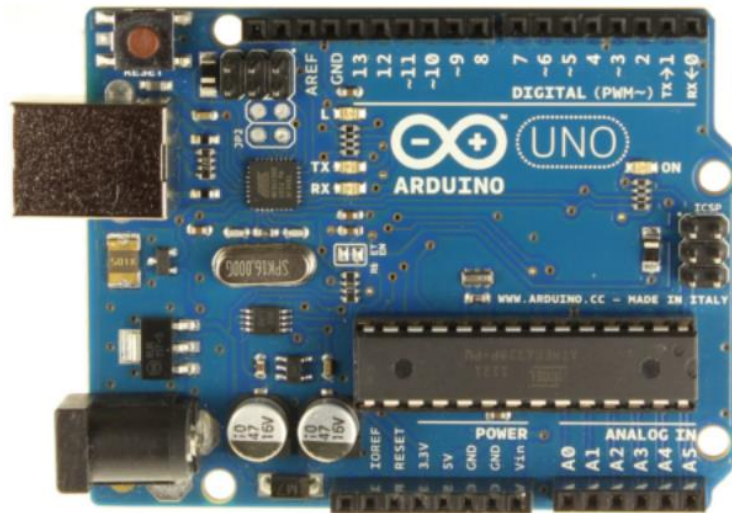


Рисунок 2.2 – Вигляд Arduino Uno

Arduino Uno забезпечує зручний інтерфейс для підключення різних датчиків, модулів зв'язку та пристроїв виконання команд. Вона надає підтримку цифрових та аналогових входів/виходів, а також інтерфейсів UART, SPI, I2C, що дозволяє інтегрувати її з більшістю електронних компонентів (рис. 2.3).

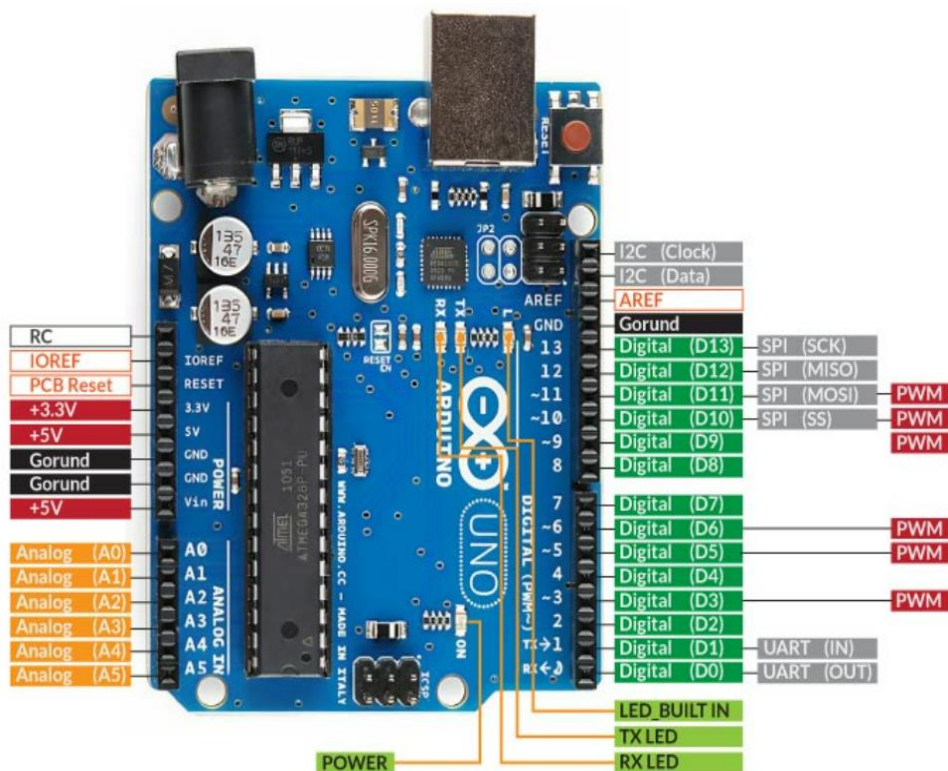


Рисунок 2.3 – Структура плати Arduino Uno

В таблиці 2.1 описані характеристики модуля Arduino Uno

Таблиця 2.1 – Характеристики модуля Arduino Uno

Характеристика	Значення
Мікроконтролер	ATmega328P
Робоча напруга	5 В
Вхідна напруга (рекомендована)	7–12 В
Цифрові входи/виходи	14 (з них 6 підтримують PWM)
Аналогові входи	6
Тактова частота	16 МГц
Оперативна пам'ять (SRAM)	2 КБ
EEPROM	1 КБ
Інтерфейси	UART, SPI, I2C
Розміри	68.6 мм x 53.4 мм

Arduino Uno програмується за допомогою середовища Arduino IDE, яке суттєво полегшує процес розробки та відлагодження систем. Підключення з ПК відбувається через USB-кабель, що водночас забезпечує живлення плати та її програмування.

В рамках проєкту Arduino Uno виступає центральним очислювальним вузлом: обробляє дані, отримані з датчика звуку, передає їх до модуля зв'язку через UART, а також взаємодіє з іншими складовими системи.

Arduino Uno – це бюджетна, відкрита та популярна платформа, що робить її відмінним вибором для розробки функціональних прототипів у проєктах контролю шуму в громадських місцях.

Отож, Arduino Uno було обрано як ключову платформу для створення комп'ютеризованої системи контролю шуму в громадських місцях, враховуючи її зручність, стабільність, широку сумісність з зовнішніми пристроями та підтримку розробників.

2.2.2 Вибір датчика шуму MAX9814

У даному проєкті було обрано датчик шуму MAX9814, що дає можливість вимірювати рівень звукового тиску в навколишньому середовищі. Цей модуль заслужено входить до числа найпопулярніших для визначення звукових коливань в системах моніторингу, завдяки своїй високій чутливості та наявності автоматичного регулювання посилення (AGC).

Датчик MAX9814 (рис. 2.4) оснащений вбудованим електретним мікрофоном, підсилювачем та схемою AGC, яка автоматично регулює коефіцієнт підсилення, орієнтуючись на рівень зовнішнього шуму. Це дозволяє запобігти перевантаженню аналогового входу мікроконтролера при занадто потужному сигналі, а також забезпечити необхідну амплітуду для фіксації слабких звуків.

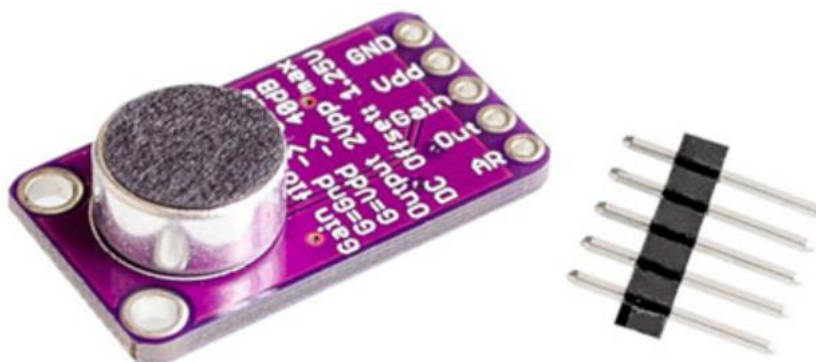


Рисунок 2.4 – Датчик шуму MAX9814

Модуль має аналоговий вихід, який з'єднується з одним з аналогових входів плати Arduino Uno (зазвичай A0). Напруга на цьому виході відповідає рівню гучності в навколишньому середовищі. Це дає змогу проводити цифрову обробку сигналу мікроконтролером для визначення перевищення допустимого рівня шуму [13, 14].

Датчик отримує живлення від стабілізованого джерела напруги 5 В, яке забезпечується самою платформою Arduino. Для гарантування високої точності вимірювань, рекомендується розміщувати модуль на деякій відстані від інших електронних елементів. Це потрібно для мінімізації впливу електромагнітних перешкод.

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

У таблиці 2.2 наведено основні технічні характеристики датчика MAX9814

Таблиця 2.2 – Технічні характеристики датчика MAX9814

Характеристика	Значення
Тип мікрофону	Електретний
Живлення	2.7 – 5.5 В
Вихідний сигнал	Аналоговий (AOUT)
Частотний діапазон	20 Гц – 20 кГц
Автоматичне регулювання посилення	Присутнє (AGC)
Максимальне підсилення	До 60 дБ
Час реакції AGC	~30 мс
Інтерфейс підключення до Arduino	AOUT → A0, VCC → 5V, GND → GND

Модуль MAX9814 був обраний як найкращий варіант для втілення комп'ютеризованої системи контролю шуму в громадських місцях, враховуючи його вдале поєднання необхідної точності, легкості інтеграції, надійності функціонування та економного споживання енергії. Використання цього модуля забезпечує можливість фіксації перевищення заданого рівня шуму та миттєве інформування відповідальних за це осіб про виявлені факти порушень.

2.2.3 Вибір Wi-Fi модуля ESP8266-01S

Для налагодження бездротової передачі даних в системі контролю шуму в громадських місцях було обрано Wi-Fi модуль ESP8266. Цей модуль вирізняється як одне з найбільш поширених рішень для організації бездротового зв'язку, що пояснюється його привабливою ціною, компактними габаритами та багатим функціоналом.

ESP8266-01S (рис. 2.5) — це модуль, що обладнаний мікроконтролером з інтегрованим Wi-Fi інтерфейсом, який підтримує стандарти IEEE 802.11 b/g/n. Завдяки інтегрованому стеку TCP/IP, цей модуль дозволяє без зусиль підключати

обладнання до локальної мережі або глобальної мережі Інтернет, забезпечуючи віддалений контроль і збір даних [15].

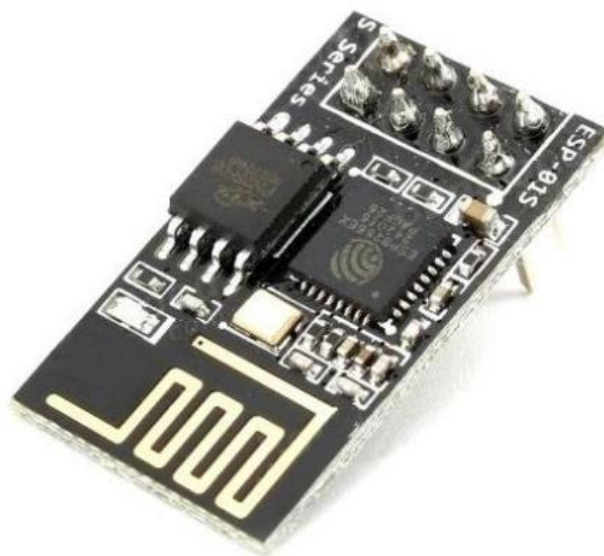


Рисунок 2.5 – Wi-Fi модуль ESP8266-01S

Модуль характеризується низьким споживанням енергії, що є важливим для функціонування від автономних джерел живлення. Крім цього, він підтримує різноманітні робочі режими, включаючи режим сну, призначений для мінімізації енергоспоживання. Інтерфейс підключення ESP8266-01S до плати Arduino Uno здійснюється за допомогою послідовного UART порту, що забезпечує простий і надійний обмін даними. Модуль ESP8266-01S був обраний через його високу сумісність з Arduino, велику кількість підтримки в програмному забезпеченні та легкість налаштування Wi-Fi з'єднання. Це гарантує стабільну передачу даних про рівень шуму в режимі реального часу на віддалений сервер, що дозволяє оперативно контролювати громадські місця.

В таблиці 2.3 наведені основні технічні характеристики модуля ESP8266-01S.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 2.3 – Технічні характеристики Wi-Fi модуля ESP8266-01S

Характеристика	Значення
Тип модуля	Wi-Fi модуль з мікроконтролером
Частота роботи Wi-Fi	2.4 ГГц
Підтримувані стандарти	IEEE 802.11 b/g/n
Інтерфейс підключення	UART (TX, RX), SPI, I2C
Об'єм вбудованої пам'яті	До 4 МБ Flash
Живлення	3.3 В
Споживання у активному режимі	~80 мА
Режими енергозбереження	Так

Отже, застосування ESP8266-01S дає змогу збудувати адаптивну та продуктивну систему контролю шуму, яка поєднує у собі точні вимірювання, автоматизовану обробку даних та швидке інформування відповідальних осіб.

2.2.4 Вибір динаміка Active Buzzer Module

Для реалізації звукових сповіщень у системі контролю рівня шуму в громадських місцях було обрано модуль Active Buzzer Module, як невеликий та практичний компонент для створення попереджувальних сигналів. Головна його перевага полягає у інтегрованому генераторі, що дає змогу генерувати звукові імпульси без необхідності генерування частоти з боку мікроконтролера.

Active Buzzer (рис. 2.6) – це пристрій, який генерує звук, коли на нього подається логічний рівень HIGH. Ця особливість значно полегшує його інтеграцію з мікроконтролерами типу Arduino Uno, ESP8266 або STM32. Не потрібно застосовувати широтно-імпульсну модуляцію (PWM) або таймери для створення потрібної частоти [16].



Рисунок 2.6 – Динамік Active Buzzer Module

Модуль має два виводи – VCC і GND, а також вихідний сигнал I/O, який підключається до цифрового GPIO-порту мікроконтролера. Управління модулем реалізується через подачу високого логічного рівня на конкретний цифровий пін, що активує звуковий сигнал. Завдяки такому рішенню можливо генерувати як короткі, так і тривалі звукові сповіщення, визнаючи тривалість імпульсу. У таблиці 2.4 наведено основні технічні характеристики модуля Active Buzzer.

Таблиця 2.4 – Технічні характеристики модуля Active Buzzer.

Характеристика	Значення
Тип динаміка	Активний зумер
Живлення	3.3–5 В
Робочий струм	~30 мА
Частота звукового сигналу	~2–4 кГц (вбудована)
Гучність	~85–95 дБ
Метод підключення	GPIO (цифровий вихід)
Тип сигналу	HIGH/LOW
Розміри	10–15 мм (діаметр корпусу)

Застосування Active Buzzer Module є виправданим з огляду на простоту підключення, незначне споживання енергії та високу стабільність роботи. Звукові сигнали, які він генерує, добре чутно навіть серед галасу, що дає змогу швидко реагувати на зміни в шумовому фоні. Отже, цей модуль повністю відповідає потребам у складових для створення звукових сповіщень в рамках розроблюваної системи.

1.2.5 Вибір LCD-дисплея 16x2 з інтерфейсом I2C

Для візуального представлення інформації про рівень шуму в реальному часі було обрано рідкокристалічний дисплей LCD 16x2 з інтерфейсом I2C. Цей модуль забезпечує можливість легко відображати текстові дані, зокрема поточне значення гучності в децибелах та статус Wi-Fi з'єднання.

LCD-дисплей 16x2 (рис. 2.7) – це популярний індикатор, що вміщує два рядки по 16 символів кожен. Він працює на основі контролера HD44780, який розпізнає загальноприйняті ASCII-символи, включаючи латинський алфавіт, цифри та базові знаки. Застосування плати I2C-конвертера, дозволяє підключати його до мікроконтролера за допомогою лише двох пінів – SDA (дані) та SCL (синхронізація), що помітно спрощує процес підключення та мінімізує кількість необхідних з'єднань у системі [17].



Рисунок 2.7 – LCD-дисплей 16x2

Дисплей з'єднується з мікроконтролером через I2C інтерфейс, який дозволяє передавати дані послідовно, використовуючи мінімальні ресурси. Живлення дисплея забезпечується від джерела 5 В, а контраст можна регулювати, використовуючи потенціометр, що вбудований у I2C модуль. Завдяки високій сумісності з Arduino, ESP8266 та STM32, цей дисплей легко програмується, застосовуючи стандартні бібліотеки, зокрема LiquidCrystal_I2C. У таблиці 2.5 наведено основні технічні характеристики LCD 16x2 з інтерфейсом I2C.

Таблиця 2.5 – Технічні характеристики LCD 16x2 з інтерфейсом I2C

Характеристика	Значення
Тип дисплея	PK (LCD) 16x2
Кількість символів	2 рядки по 16 символів
Інтерфейс передачі даних	I2C (SDA, SCL)
Контролер дисплея	HD44780
Напруга живлення	5 В
Споживання струму	~20 мА
Адреса I2C	Зазвичай 0x27 або 0x3F
Регулювання контрасту	Так (вбудований потенціометр)
Підсвічування	Присутнє (може вимикатись в коді)

Вибір LCD-дисплея з інтерфейсом I2C є доцільним рішенням під час створення комп'ютеризованої системи контролю шуму, адже він забезпечує зрозуміле та наочне представлення інформації, мінімально використовуючи піни мікроконтролера. Його легкість у застосуванні, доступність і стабільність роботи, роблять цей модуль найкращим варіантом для виведення даних.

2.3 Опис шин обміну даними

2.3.1 Опис шини даних I2C

Шина I2C (Inter-Integrated Circuit) – це серійний інтерфейс, через який мікроконтролери та периферійні пристрої (датчики, дисплеї, пам'ять тощо) взаємодіяти між собою, використовуючи лише дві лінії: лінію даних SDA (Serial Data Line) і лінію синхронізації SCL (Serial Clock Line). Цей інтерфейс знайшов широке застосування у вбудованих системах, адже він простий у використанні, гнучкий і дозволяє підключати багато пристроїв до однієї шини [18].

Основною перевагою шини I2C полягає у можливості підключення до неї до 127 пристроїв водночас. Кожен з яких має унікальну 7-бітну адресу для ідентифікації. Взаємодія відбувається за схемою “ведучий–ведений” (master–slave) (рис. 2.8). Ведучим найчастіше виступає мікроконтролер, наприклад, Arduino UNO, а підлеглими різні периферійні пристрої (наприклад, LCD-дисплей, датчики тощо).

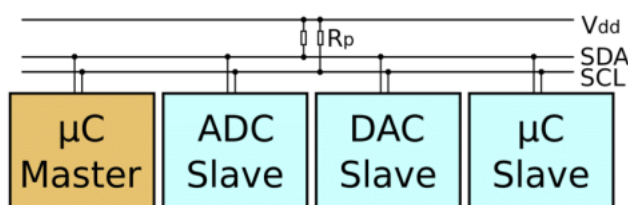


Рисунок 2.8 – Схема з'єднання шини даних I2C

Робота шини I2C складається з кількох ключових етапів. Спершу ведучий пристрій надсилає стартовий сигнал, що інформує всі пристрої на шині про початок передачі даних. Далі ведучий пристрій пересилає адресу потрібного йому пристрою а також вказує напрямок обміну – читання чи запис даних. Після цього відбувається поелементне пересилання даних, де кожен байт підтверджується отримувачем за допомогою спеціального сигналу підтвердження (ACK). У завершенні обміну даними, ведучий пристрій відправляє стоп-сигнал, що свідчить про завершення передачі. Схема

формування передачі шини даних I2C зображена на рисунку 2.9.

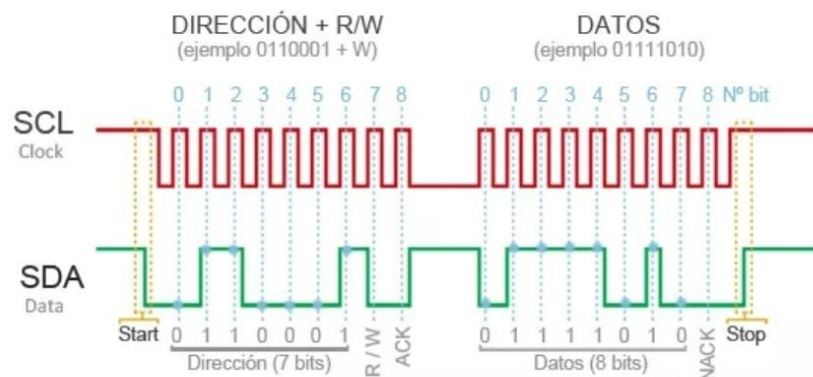


Рисунок 2.9 – Схема формування передачі шини даних I2C

У даній роботі шина I2C застосовується для з'єднання LCD-дисплея з мікроконтролером Arduino UNO. Цей метод дає змогу суттєво скоротити кількість використаних виводів мікроконтролера, адже дисплей взаємодіє з Arduino тільки через SDA та SCL.

2.3.2 Опис шини даних UART

Шина UART (Universal Asynchronous Receiver-Transmitter) – це універсальний асинхронний передавач-приймач, який використовується для послідовного обміну даними між мікроконтролерами та іншими електронними компонентами. На відміну від таких протоколів як I2C або SPI, UART не потребує окремої лінії синхронізації, оскільки обидва пристрої заздалегідь визначають швидкість передачі даних (тобто працюють асинхронно). Обмін даними відбувається лише по лініях TX (transmit) — передача даних і RX (receive) — прийом даних [19].

UART знайшов широке застосування у вбудованих системах, де важливо забезпечити простий і надійний обмін даними, наприклад, між мікроконтролером та модулем Wi-Fi (ESP8266), GPS-пристроєм, Bluetooth-модулем або ПК. У такому випадку один пристрій виконує роль передавача, а інший — приймача (рис. 2.10).

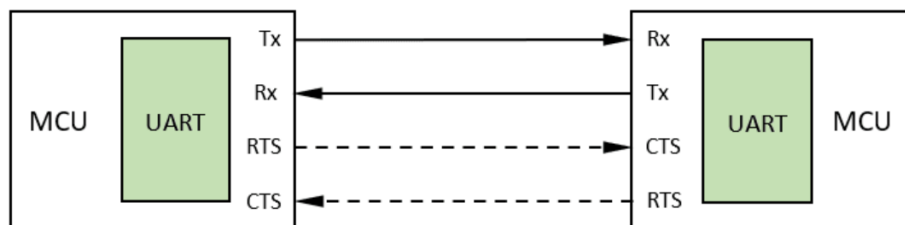


Рисунок 2.10 – Схема з'єднання шини даних UART

Робота шини UART базується на послідовній передачі байтів. Передача починається зі стартового біта, після котрого йде байт даних (8 біт), можливе додавання біта парності (для перевірки цілісності), а також одного або двох стоп-бітів, які повідомляють про закінчення пакету. Для вдалого обміну інформарцією обидва пристрої повинні бути налаштовані на однакову швидкість передачі даних (baud rate) — наприклад, 9600 або 115200 бод.

Асинхронний характер UART полягає в тому, що обмін даними не потребує спільного тактового сигналу, що полегшує з'єднання між різними пристроями. Крім цього, кожне передане повідомлення має чітку структуру, яка дозволяє приймачу правильно відтворити отриману інформацію.

У даній роботі UART застосовується для обміну даними між мікроконтролером Arduino UNO та модулем Wi-Fi ESP8266. Через цю шину здійснюється передача інструкцій для підключення до мережі, а також передавання повідомлень на сервер. Завдяки простоті реалізації та широкій підтримці, UART є надійним інтерфейсом для бездротової комунікації в системі контролю шуму.

2.4 Опис електрично-принципової схеми системи

2.4.1 Опис середовища EasyEDA

EasyEDA — це сучасна онлайн-платформа для створення електричних схем та друкованих плат, яка функціонує в браузері. Вона надає зручний інтерфейс, що дає змогу оперативно розробляти, змінювати та моделювати електронні схеми, не вимагаючи інсталяції додаткових програм (рис. 2.11).

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

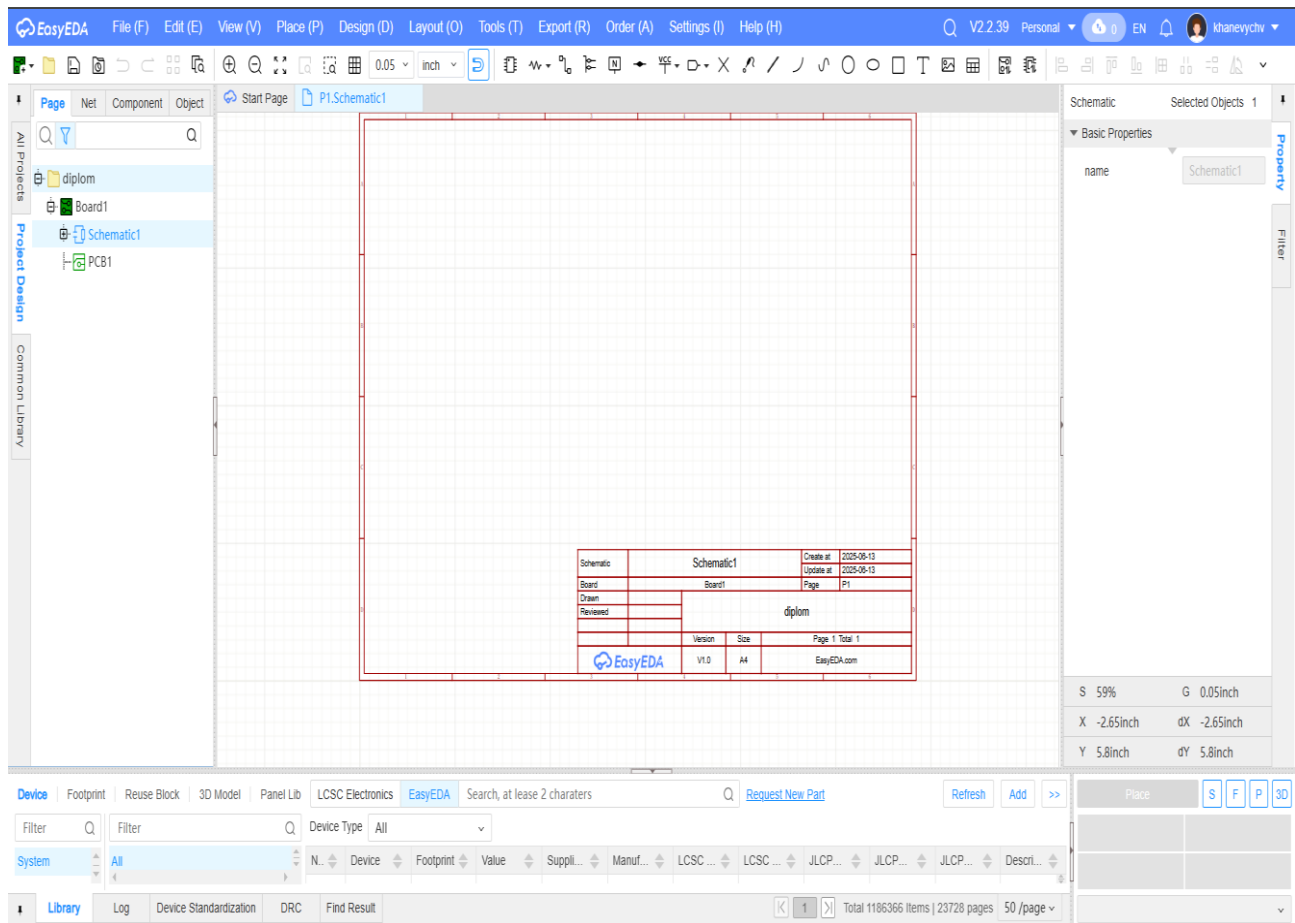


Рисунок 2.11 – Вигляд середовища EasyEDA

Платформа включає в себе значну бібліотеку стандартних електронних елементів: резисторів, діодів, транзисторів, мікросхем, роз'ємів та інших. Це суттєво прискорює процес проектування, адже користувачу не потрібно самостійно конструювати кожен компонент.

EasyEDA дає змогу без зусиль створювати електричні схеми, додаючи компоненти на робоче поле, з'єднуючи їх, підписуючи та додаючи анотації.

Щоб створити друковані плати (PCB), платформа містить потужний редактор, включно з функціями ручного й автоматичного трасування доріжок, перевірки цілісності схеми (DRC), експорту в форматі Gerber та іншими.

Середовище EasyEDA дозволяє імпортувати та експортувати схеми та плати у форматах, таких як PDF, PNG, SVG, а також у форматах Gerber, що дають змогу замовляти виробництво друкованих плат. Крім того, передбачено імпорт проектів з інших програм, зокрема з Altium Designer або KiCad.

EasyEDA – це ефективний інструмент для розробки електронної частини проєктів, що стосуються з автоматизованого контролю, моніторингу чи оповіщення, наприклад, системи контролю шуму в громадських місцях.

2.4.2 Опис розробленої електричної схеми

Електрична принципова схема системи для контролю шуму зображена на рисунку 2.12. У схемі використовуються наступні компоненти: мікроконтролер Arduino ATmega328 (U1), модуль Wi-Fi ESP-01S (U3), датчик звуку MAX9814 (U2), звуковий модуль HX-2212 (U4), а також LCD-дисплей з інтерфейсом I2C (H1) (рис. 2.12).

Мікрофонний модуль MAX9814 (U2) підключено до аналогового входу ADC0 (PC0) мікроконтролера для вимірювання рівня шуму. Звуковий модуль HX-2212 (U4) підключений до цифрового виводу PD6, який активується мікроконтролером при перевищенні допустимого рівня шуму.

LCD-дисплей з інтерфейсом I2C (H1) підключено до виводів PC4 (SDA) та PC5 (SCL) і використовується для виводу поточних вимірних значень шуму або статусу Wi-Fi.

Модуль ESP-01S (U3) використовується для передавання зібраної інформації по Wi-Fi. Він підключений безпосередньо до апаратної послідовної шини UART мікроконтролера – до виводів RXD та TXD що забезпечує стабільну та надійну передачу даних. Для безпечної роботи ESP-01S використано подільник напруги (резистори R2 і R3). Живлення модуля здійснюється стабільним джерелом 3.3 В.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

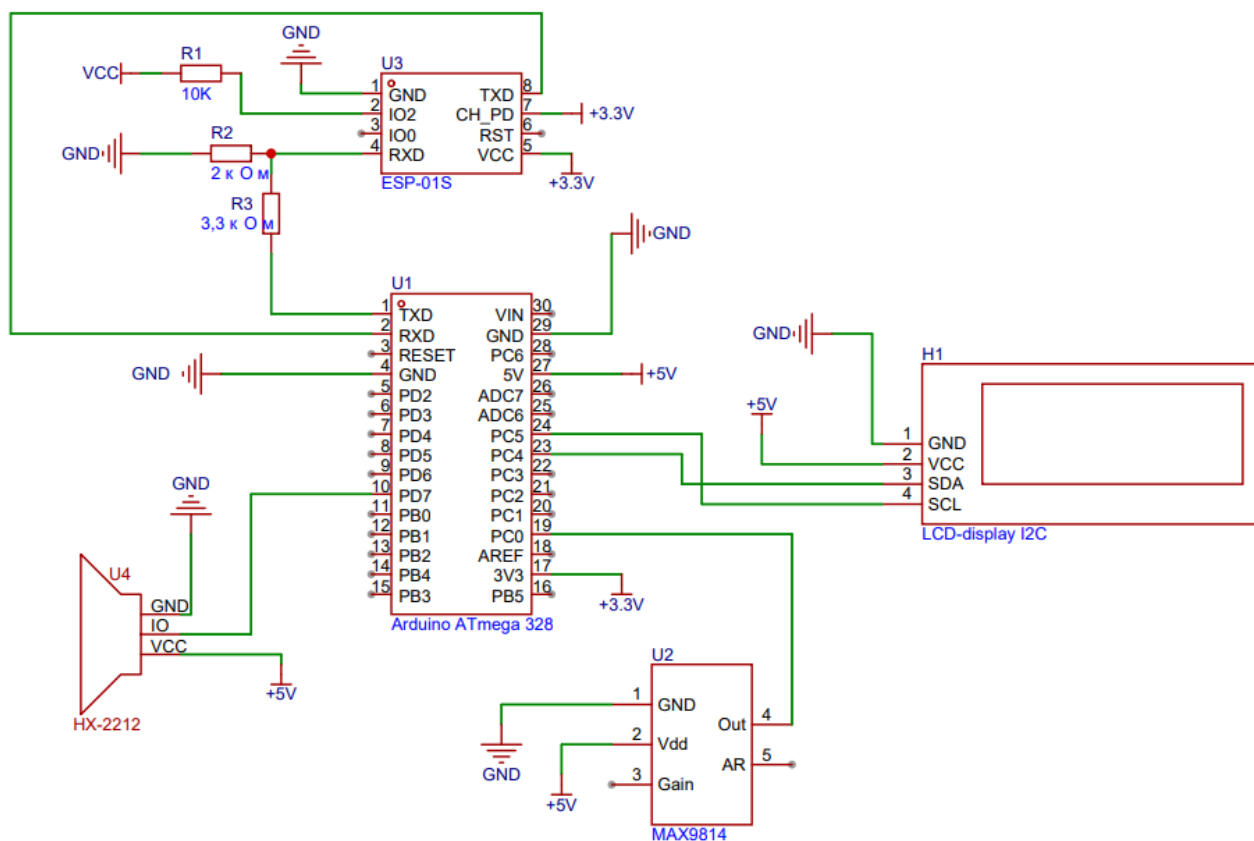


Рисунок 2.12 – Розроблена електрична принципова схема

Розроблена електрична принципова схема забезпечує коректну взаємодію всіх ключових модулів: вимірювання рівня шуму, локальне відображення даних на екрані, сигналізації за допомогою динаміка та передавання даних через Wi-Fi. Елементи обрані таким чином, аби гарантувати простоту реалізації, надійність та можливість розширення функціоналу системи в майбутньому.

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Розробка алгоритму роботи системи контролю шуму в громадських місцях.

Алгоритм роботи комп'ютеризованої системи контролю шуму в громадських місцях показано на блок-схемі (рис. 3.1). Головна ціль алгоритму – безперервне вимірювання рівня шуму, його аналіз, сигналізація при перевищенні встановленого порогу та передача даних на веб-сервер для моніторингу.

Робота системи починається з ініціалізації всіх необхідних компонентів. Після завершення ініціалізації здійснюється зчитування аналогового сигналу з вихідного порту модуля MAX9814, що визначає гучність шуму. Отримане значення обробляється мікроконтролером Arduino UNO, після чого відбувається його перетворення в значення, виражене в децибелах.

Далі система переходить до перевірки чи перевищує шум визначений поріг. Якщо перевищує, відбувається активація звукового сигналу, а також інформація про поточний рівень шуму виводиться на LCD-дисплей. Після цього дані надсилаються на веб-сервер за допомогою модуля Wi-Fi.

Якщо рівень шуму не перевищує допустимий поріг, то система пропускає блок увімкнення сигналізації та виводу на дисплей, і відразу передає дані на веб-сервер для оновлення інформації.

Після передачі даних, система робить коротку затримку для зниження навантаження на мережу та мікроконтролер. Після цього відбувається повернення до початку циклу, і процес повторюється безперервно.

					КС КРБ 123.241.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	Практична частина		
Розроб.		Ханевич В.В.					
Перевір.		Шингера Н.Я.					
Реценз.							
Н. Контр.		Тиш Є.В.					
Затверд.		Осухівська Г.М.					
					Літ.	Арк.	Аркушів
						38	78
					ТНТУ, каф. КС, гр. СІ-42		



Рисунок 3.1 – Блок-схема алгоритму роботи системи

Такий підхід дає змогу забезпечити автоматизований моніторинг рівня шуму, миттєве реагування на перевищення допустимих норм, а також

централізований збір інформації для подальшого аналізу або контролю відповідальними особами.

3.2 Створення моделі тестового макету в середовищі Wokwi

Модель тестового макету для системи вимірювання рівня шуму була створена у середовищі Wokwi. В основі системи лежить мікроконтролер Arduino Uno, до котрого приєднано всі компоненти. Аналоговий мікрофон під'єднаний до аналогового входу A0, що дозволяє зчитувати рівень звуку з навколишнього середовища (рис. 3.2).

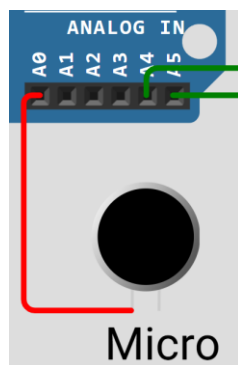


Рисунок 3.2 – Підключення мікрофону до Arduino Uno

Для відображення інформації використовується LCD-дисплей 16x2 з інтерфейсом I2C, котрий приєднано до пінів A4 (SDA) і A5 (SCL). Такий спосіб підключення суттєво спрощує схему та зменшує кількість необхідних проводів (рис. 3.3).

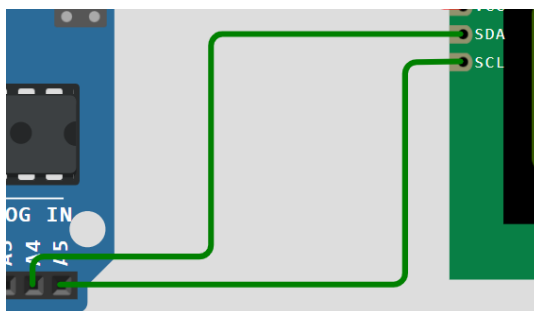


Рисунок 3.3 – Підключення LCD-дисплею до Arduino Uno

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40

Звукова сигналізація реалізована за допомогою зумера, підключеного до цифрового піна D2 (рис. 3.4). Зумер вмикається при перевищенні порогового рівня шуму, що дозволяє швидко повідомляти про небажані звукові рівні.

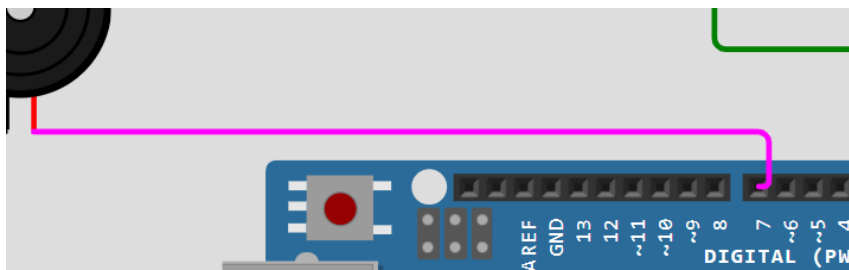


Рисунок 3.4 – Підключення зумера до Arduino Uno

Для передачі даних на віддалений сервер використовується Wi-Fi модуль ESP-01S, який підключений до Arduino через програмний послідовний інтерфейс. Для обміну даними задіяні піни TX і RX. Варто зазначити, що живлення модуля ESP-01S потребує пониження напруги до 3.3 В, тому для цього використовуються резистори (рис. 3.5).

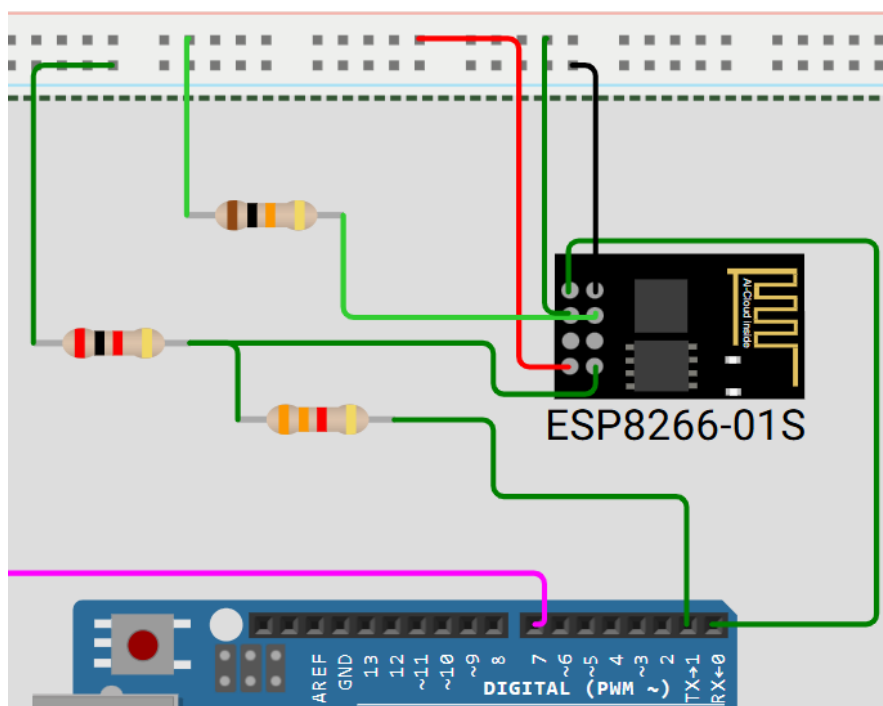


Рисунок 3.5 – Підключення ESP8266-01S до Arduino Uno

Живлення всіх компонентів здійснюється від 5-вольтової шини Arduino. Всі підключення в схемі були ретельно спроектовані з метою забезпечення коректного функціонування системи та сумісності між компонентами. Модель тестового макету зображено на рисунку 3.6.

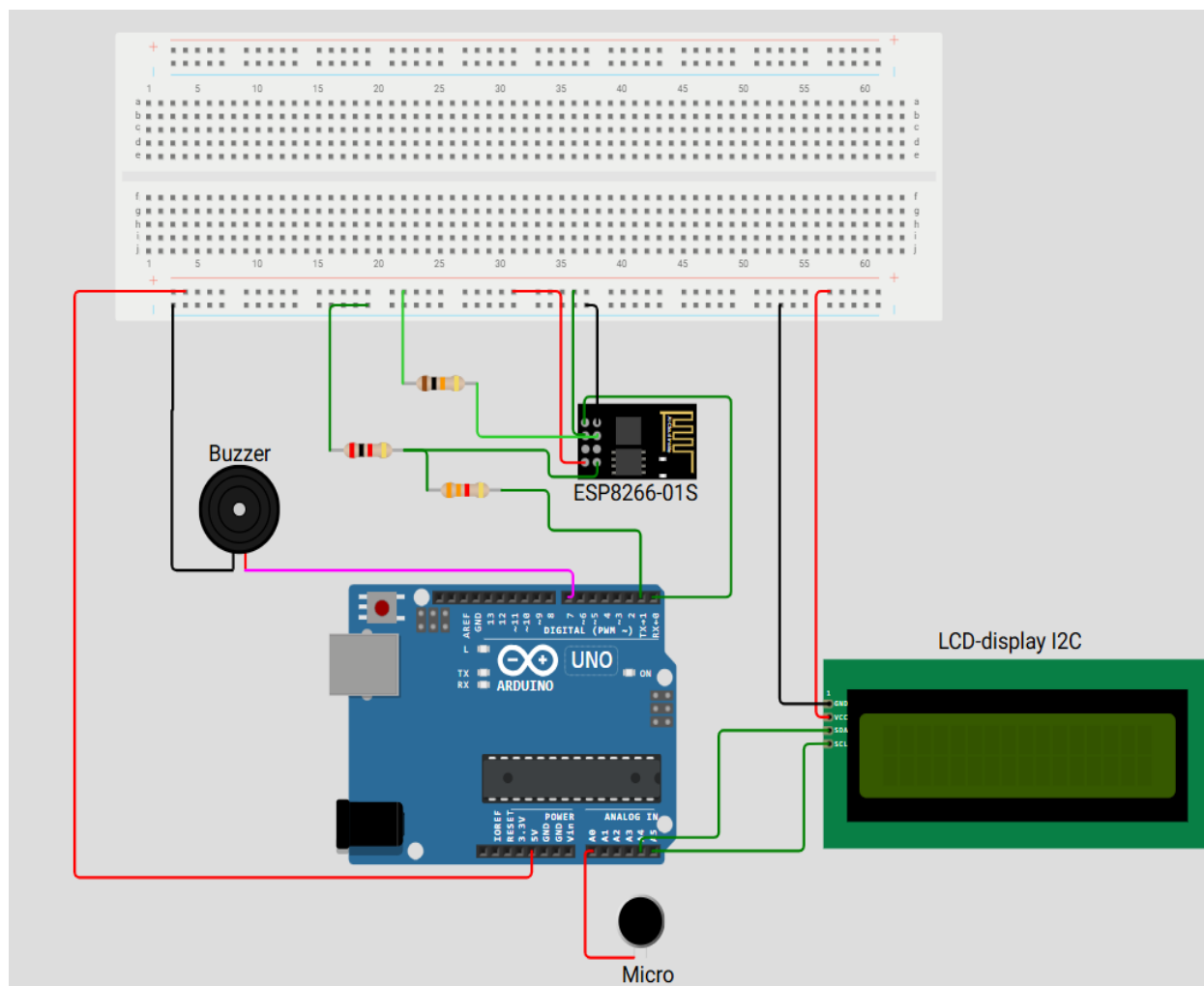


Рисунок 3.6 – Модель тестового макету

Отже, розроблена модель забезпечує ефективну взаємодію між усіма модулями системи, що є критично важливим для стабільної роботи системи вимірювання рівня шуму та подальшої передачі даних для їх аналізу та моніторингу.

3.3 Розробка програмного забезпечення

3.3.1 Написання коду для функціонування системи

Для розробки програмного забезпечення даного проекту було імпортовано декілька ключових бібліотек (рис. 3.7), які надають необхідний функціонал:

- `Arduino.h` відповідає за основні функції платформи Arduino, включаючи взаємодію з введенням-виведенням та керування апаратними ресурсами;
- `LiquidCrystal_I2C.h` використовується для роботи з LCD-дисплеєм через інтерфейс I2C, що дозволяє відображати інформацію про рівень шуму та статус з'єднання;
- `SoftwareSerial.h` дає змогу створити додатковий програмний послідовний порт для комунікації з модулем ESP-01, що працює як Wi-Fi інтерфейс;
- `Wire.h` забезпечує підтримку протоколу I2C, необхідного для взаємодії з LCD-дисплеєм.

```
1  #include <Arduino.h>
2  #include <LiquidCrystal_I2C.h>
3  #include <SoftwareSerial.h>
4  #include <Wire.h>
```

Рисунок 3.7 – Лістинг коду для імпорту бібліотек

Код на рисунку 3.8 відповідає за початкову ініціалізацію системи та підключення до Wi-Fi мережі. У функції `setup()` налаштовується контакт динаміка як вихідний та вимикається (логічний рівень LOW), ініціалізується LCD дисплей через шину I2C, а також активуються серійні порти. Це включає в себе стандартний UART для моніторингу через ПК (Serial) і програмний UART для зв'язку з ESP-01 (`espSerial`). На дисплеї виводиться повідомлення, що сигналізує про початок підключення до Wi-Fi, після чого викликається функція `connectWiFi()`, яка здійснює підключення до визначеної мережі. У випадку

					КС КРБ 123.241.00.00 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

успішного підключення на дисплей виводиться повідомлення «Wi-Fi: Connected».

```
29 void setup() {
30     // Налаштування піну динаміка
31     pinMode(buzzerPin, OUTPUT);
32     digitalWrite(buzzerPin, LOW);
33
34     // Ініціалізація LCD дисплея
35     lcd.init();
36     lcd.backlight();
37
38     // Ініціалізація серійних портів
39     Serial.begin(115200); // Стандартний UART (для монітору порту)
40     espSerial.begin(115200); // Програмний UART (для ESP)
41
42     // Виведення повідомлення про підключення до Wi-Fi
43     lcd.setCursor(0, 0);
44     lcd.print("Wi-Fi: Connecting...");
45
46     connectWiFi(); // Спроба з'єднання з Wi-Fi мережею
47
48     // Повідомлення про успішне з'єднання
49     lcd.setCursor(0, 0);
50     lcd.print("Wi-Fi: Connected ");
51 }
```

Рисунок 3.8 – Лістинг коду початкової ініціалізації та підключення до Wi-Fi

Після чого, програма розпочинає обробку сигналу, отриманого від мікрофона та обчислення умовного рівня шуму вираженого в децибелах (рис. 3.9). Дані з мікрофона, представлені аналоговим значенням, зчитується за допомогою функції `analogRead()`, яка видає показники в межах від 0 до 1023. Ці дані потім піддаються лінійному перетворенню функцією `map()`, що дозволяє перевести їх до приблизного діапазону гучності від 40 до 80 дБ.

```
54 // Зчитування аналогового значення з мікрофона (0-1023)
55 int micValue = analogRead(micPin);
56
57 // Перетворення значення в приблизний рівень шуму в дБ
58 float baseDB = map(micValue, 0, 1023, 40, 80);
```

Рисунок 3.9 – Лістинг коду перетворення сигналу з мікрофона у рівень шуму

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

Код на рисунку 3.10 реалізує передачу поточного значення рівня шуму, виміряного в децибелах на веб-сервер. Це відбувається через функцію `sendDataToServer()`. Після відправлення даних виконується затримка в 2000 мілісекунд (2 секунди). Ця пауза служить для встановлення інтервалу між кожним виміром та наступною передачею інформації, запобігаючи надмірному навантаженню на систему.

```

84      // Передача значення шуму на сервер
85      sendDataToServer(dB);
86
87      delay(2000); // Затримка 2 секунди між циклами

```

Рисунок 3.10 – Лістинг коду передачі значень рівня шуму на веб-сервер

Наступним кроком формується команда `AT+CWJAP` (рис. 3.11), яка містить назву (ssid) та пароль (password) Wi-Fi мережі для підключення модуля ESP-01. Виконання команди здійснюється функцією `sendCommand()` з таймаутом 5000 мс. Якщо у відповіді модуля присутній рядок "OK", підключення вважається успішним, і в серійному моніторі виводиться повідомлення про успіх. В іншому випадку на LCD-дисплей виводиться повідомлення про помилку підключення, а у серійному моніторі — відповідний запис.

```

95      // Команда для підключення до мережі Wi-Fi
96      String cmd = "AT+CWJAP=\"" + String(ssid) + "\",\"" + String(password) + "\"";
97      if (sendCommand(cmd.c_str(), 5000).indexOf("OK") != -1) {
98          Serial.println("WiFi connected");
99      } else {
100         lcd.clear();
101         lcd.print("WiFi Fail");
102         Serial.println("WiFi connection failed");
103     }

```

Рисунок 3.11 – Лістинг коду підключення до мережі Wi-Fi

Функція `sendCommand` (рис. 3.12) виконує відправку AT-команд до модуля ESP-01, використовуючи програмний послідовний порт, а також зчитування

					КС КРБ 123.241.00.00 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

відповіді від нього. Вона приймає два параметри: текст команди (cmd) та час очікування відповіді (timeout). Для надсилання команди до ESP застосовується `espSerial.println(cmd)`. Далі, протягом заданого часу, функція аналізує всі доступні символи з відповіді ESP і формує з них завершений рядок `response`. З метою діагностики та налагодження, зібрана відповідь виводиться у серійний монітор за допомогою `Serial.print(response)`.

```

106 // Функція відправки AT-команд до ESP і отримання відповіді
107 String sendCommand(const char* cmd, int timeout) {
108     String response = "";
109     espSerial.println(cmd); // Надсилання команди
110     long int time = millis();
111     while ((time + timeout) > millis()) {
112         while (espSerial.available()) {
113             char c = espSerial.read();
114             response += c;
115         }
116     }
117     Serial.print(response); // Виведення відповіді в монітор порту
118     return response;
119 }

```

Рисунок 3.12 – Лістинг коду функції відправки AT-команд до ESP

Функція `sendDataToServer(float noiseLevel)` встановлює TCP-з'єднання з сервером, куди буде відправлено HTTP-запит (рис. 3.17). Насамперед формується команда `AT+CIPSTART`, що запускає TCP-з'єднання з вказаною адресою та портом. У випадку, якщо відповідь модуля ESP не містить «OK», виводиться сповіщення про невдале з'єднання у серійному моніторі, і робота функції припиняється. Цей фрагмент є початковим кроком до передачі даних на сервер у вигляді GET-запиту.

```

121 // Функція надсилання HTTP-запиту до сервера з рівнем шуму
122 void sendDataToServer(float noiseLevel) {
123     String cmd;
124
125     // Встановлення TCP-з'єднання
126     cmd = "AT+CIPSTART=" + String(host) + ", " + String(port);
127     if (sendCommand(cmd.c_str(), 5000).indexOf("OK") == -1) {
128         Serial.println("Connection failed");
129         return;
130     }
131 }

```

Рисунок 3.13 – Лістинг коду встановлення TCP-з'єднання

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

Далі формується HTTP GET-запит, де параметр noise визначає рівень шуму. Після цього програма відправляє команду ESP: AT+CIPSEND (рис. 3.14). Ця команда інформує модуль про обсяг запиту, який буде передано. Якщо у відповіді модуля відсутній символ >, що свідчить про готовність до прийому даних, передавання припиняється, а у вікні серійного монітора виводиться повідомлення «CIPSEND failed».

Цей етап має вирішальне значення для підготовки ESP-модуля до відправки HTTP-запиту на сервер.

```

132 // Формування HTTP-запиту GET
133 String httpRequest = "GET /update?noise=" + String(noiseLevel) +
134 | | | | | | | | | | " HTTP/1.1\r\nHost: " + String(host) + "\r\nConnection: close\r\n\r\n";
135
136 // Повідомлення ESP про розмір запиту
137 cmd = "AT+CIPSEND=" + String(httpRequest.length());
138 if (sendCommand(cmd.c_str(), 2000).indexOf(">") == -1) {
139     Serial.println("CIPSEND failed");
140     return;
141 }

```

Рисунок 3.14 – Лістинг коду формування HTTP-запиту

Наступним кроком надсилається сформований HTTP-запит до вказаного хосту через ESP (espSerial.print(httpRequest);). Після цього відбувається виклик sendCommand("", 5000), що передбачає читання відповіді від сервера протягом визначеного періоду часу (рис. 3.15).

Далі здійснюється закриття TCP-з'єднання за допомогою команди AT+CIPCLOSE. У серійний монітор виводиться підтверджуюче повідомлення про успішну передачу значення рівня шуму (Serial.println("Data sent: ...")).

```

143 // Надсилання самого запиту
144 espSerial.print(httpRequest);
145
146 // Читання відповіді
147 sendCommand("", 5000);
148
149 // Закриття TCP-з'єднання
150 sendCommand("AT+CIPCLOSE", 2000);
151
152 Serial.println("Data sent: " + String(noiseLevel));

```

Рисунок 3.15 – Лістинг коду завершення циклу зв'язку з сервером

Цей код завершує цикл зв'язку із сервером, забезпечуючи відправку та коректне завершення з'єднання.

3.3.2 Створення веб-сервера для отримання даних

Для створення веб-сторінки було обрано середовище розробки Visual Studio Code, адже ця платформа безкоштовна, працює на різних операційних системах, та пропонує безліч додаткових модулів для веб-розробки. Програма забезпечує зручне редагування HTML, CSS і JavaScript, має автодоповнення коду та інтеграцію з системами контролю версій, такими як Git. Все це робить Visual Studio Code одним із найпопулярніших інструментів серед сучасних веб-розробників.

Першим кроком створюється папка в якій будуть три файли: index.html, script.js та style.css (рис. 3.16).

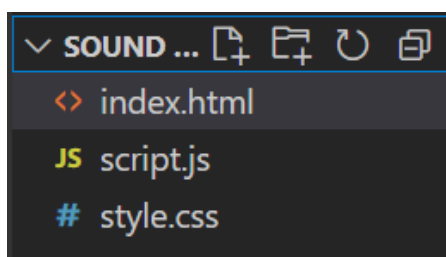


Рисунок 3.16 – Створення потрібних файлів

Першим реалізується фрагмент коду який задає початкову структуру HTML-документа для веб-сторінки моніторингу шуму (рис. 3.17). Він визначає тип документа як HTML5, що гарантує коректне відображення у сучасних браузерах, і встановлює мову сторінки українською для правильного відображення тексту. Усередині секції <head> налаштовується кодування символів UTF-8, що забезпечує підтримку українських та інших спеціальних символів, а також задається адаптивний вигляд сторінки для різних пристроїв за допомогою мета-тега viewport. Назва сторінки встановлена як «Моніторинг шуму», що відображається у вкладці браузера. Крім того, тут підключається зовнішній файл стилів style.css, який відповідає за візуальне оформлення сторінки. Таким чином, цей фрагмент створює базу для подальшого формування інтерфейсу користувача.

```

<> index.html > ...
1  <!DOCTYPE html>
2  <html lang="uk">
3  <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1" />
6      <title>Моніторинг шуму</title>
7      <link rel="stylesheet" href="style.css" />
8  </head>

```

Рисунок 3.17 – Лістинг коду початкової структури HTML-документа

Фрагмент коду на рисунку 3.18 отримує контекст малювання 2D з елемента <canvas> на веб-сторінці, який має ідентифікатор noiseChart. Цей контекст необхідний для подальшого створення і відображення графіка за допомогою бібліотеки Chart.js, оскільки він надає інтерфейс для роботи з графічним вмістом елемента canvas. Таким чином, цей рядок є ключовим для ініціалізації області, де буде відображено візуалізацію рівня шуму.

```

1 // Отримуємо контекст для малювання графіка з елемента canvas з id="noiseChart"
2 const ctx = document.getElementById('noiseChart').getContext('2d');

```

Рисунок 3.18 – Лістинг коду для отримання контексту елемента canvas

Функція `updateChart` (рис. 3.19) відповідає за оновлення графіка з новим значенням рівня шуму. Спочатку вона отримує поточний час у форматі рядка для позначення на осі часу. Потім додає цей час до масиву міток (`labels`) і нове значення шуму до набору даних графіка. Для збереження компактності відображення, якщо кількість точок перевищує 20, найстаріші дані та мітки видаляються з початку списку. Наприкінці функція викликає оновлення графіка, щоб відобразити внесені зміни. Таким чином, вона підтримує динамічне та актуальне відображення змін рівня шуму в реальному часі.

```

function updateChart(noiseValue) {
  const now = new Date().toLocaleTimeString();
  noiseChart.data.labels.push(now);
  noiseChart.data.datasets[0].data.push(noiseValue);

  if (noiseChart.data.labels.length > 20) {
    noiseChart.data.labels.shift();
    noiseChart.data.datasets[0].data.shift();
  }

  noiseChart.update();
}

```

Рисунок 3.19 – Лістинг коду що відповідає за оновлення графіка

Асинхронна функція `fetchNoiseData` (рис. 3.20) відповідає за отримання актуальних даних рівня шуму з веб-сервера за допомогою HTTP-запиту до маршруту `/data`. Спочатку вона робить запит `fetch`, після чого перевіряє статус відповіді — якщо відповідь неуспішна, виникає помилка. У разі успішного отримання даних, функція перетворює їх у формат JSON і викликає функцію `updateChart` для оновлення графіка з новим значенням шуму. Помилки при отриманні даних обробляються у блоці `catch` і виводяться в консоль. Виклик

setInterval(fetchNoiseData, 2000); забезпечує періодичне оновлення даних кожні 2 секунди, підтримуючи динамічне відображення шуму в реальному часі.

```

async function fetchNoiseData() {
  try {
    const response = await fetch('/data');
    if (!response.ok) throw new Error('Network response was not ok');
    const data = await response.json();
    updateChart(data.noise);
  } catch (error) {
    console.error('Помилка отримання даних:', error);
  }
}

```

Рисунок 3.20 – Лістинг коду функції отримання даних

У результаті реалізації веб-інтерфейсу у середовищі Visual Studio Code було створено функціональну веб-сторінку, яка дозволяє візуалізувати рівень шуму в режимі реального часу. Було розділено структуру проєкту на окремі файли HTML, CSS та JavaScript для забезпечення зручності розробки та підтримки коду. Основні фрагменти коду відповідають за відображення графіка, отримання даних із серверної частини та їх динамічне оновлення. Такий підхід дозволяє ефективно інтегрувати апаратну частину проєкту з веб-технологіями та забезпечує наочне представлення результатів вимірювання.

3.4 Тестування програми

Симуляція функціонування системи моніторингу рівня шуму дає змогу перевірити коректність роботи програмної частини без потреби у створенні фізичного прототипу. Для втілення цієї системи використовувалась платформа Wokwi, яка дала змогу змодельовати основні апаратні компоненти: мікроконтролер Arduino Uno, мікрофон (як джерело аналогового сигналу), LCD-дисплей (через інтерфейс I2C), п'єзодинамік для звукового оповіщення, а також модуль ESP-01, який виконує функцію передавання даних на веб-сервер. Такий

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

підхід забезпечує зручне тестування та візуалізацію роботи системи у віртуальному середовищі.

Після запуску програми на дисплей виводиться статус підключення до мережі Wi-Fi, та рівень шуму в децибелах (рис. 3.21).

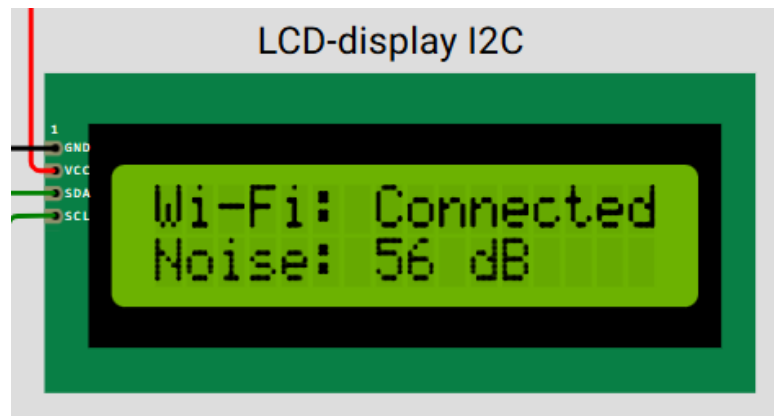


Рисунок 3.21 – Виведення даних на LCD-дисплей

У випадку перевищення допустимого рівня шуму, вмикається звукова сигналізація (рис. 3.22).

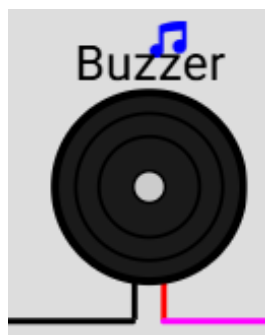


Рисунок 3.22 – Ввімкнення зумера

Дані одразу відправляються на веб-сторінку, на якій зручно спостерігати за рівнем шуму в громадських місцях, це зображено на рисунку 3.23.

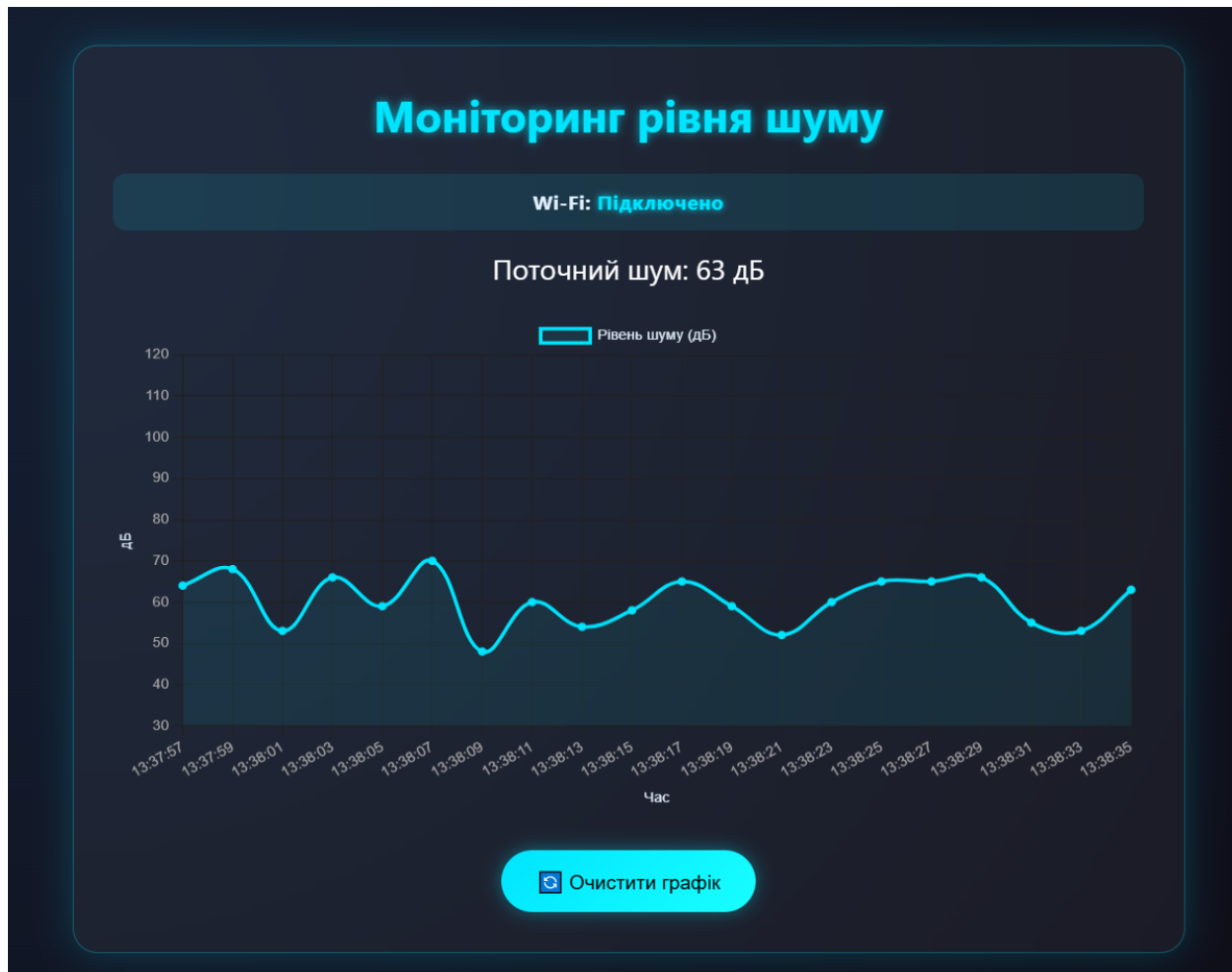


Рисунок 3.23 – Веб-сторінка для моніторингу рівня шуму

У випадку невдалого з'єднання до Wi-Fi, на LCD-дисплей виводиться повідомлення про помилку і програма не виконується (рис. 3.24).

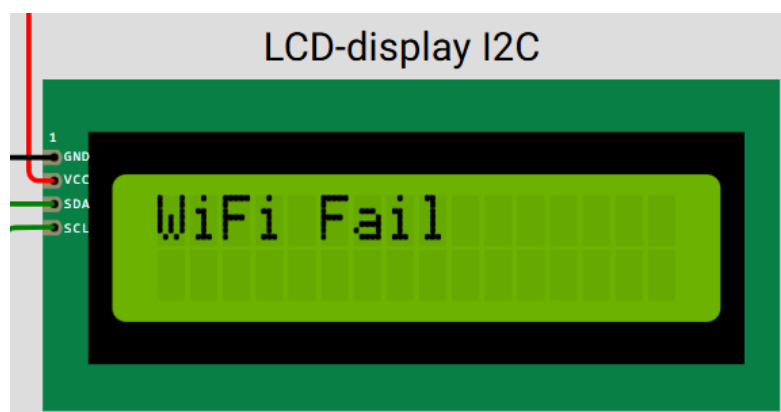


Рисунок 3.24 – Помилка з'єднання

Отже, фінальні підсумки тестової перевірки засвідчили, що розроблена система ефективно виконує всі покладені на неї завдання. Вона оперативно виявляє порушення шуму та дає змогу віддаленого моніторингу для відповідальних осіб, що демонструє її придатність до практичного застосування в реальних умовах.

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		54

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Вплив шуму на організм людини та розробка заходів щодо його зниженню до допустимих величин для обладнання.

Шум є одним із найпоширеніших та водночас недооцінених чинників впливу на здоров'я людини в умовах урбанізованого довкілля. У контексті розробки комп'ютеризованої системи контролю рівня шуму в громадських місцях надзвичайно важливо не лише виявляти перевищення припустимих параметрів, а й розуміти характер і наслідки дії шуму на фізіологічні системи людини, а також запропонувати дієві способи зниження цього впливу.

Фізіологічна дія шуму комплексна і багатofакторна. Насамперед, збільшений рівень шуму негативно впливає на центральну нервову систему. Спостерігається роздратованість, зниження концентрації уваги, стомлення, а у разі тривалого впливу — збільшення ризику розвитку неврозів та психосоматичних захворювань. За даними ВООЗ, систематичний вплив шуму більше ніж 55 децибел (дБ) протягом робочого дня вже вважається шкідливим для здоров'я [20].

Шум чинить тиск на серцево-судинну систему. У відповідь на інтенсивне акустичне навантаження активується симпатична частина вегетативної нервової системи, що зумовлює підвищення артеріального тиску, збільшення частоти серцевих скорочень і загальне навантаження на серце. За наявності хронічного впливу це може призвести до ризику розвитку гіпертонії та ішемічної хвороби серця [21].

У випадку громадських місць, таких як вокзали, торгові центри, вулиці міста чи освітні заклади, основним джерелом шуму є саме поведінка людей — зокрема скупчення натовпів, рух громадського транспорту або проведення

					КС КРБ 123.241.00.00 ПЗ						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Ханевич В.В.			Безпека життєдіяльності, основи охорони праці			Літ.	Арк.	Аркушів	
Перевір.		Шингера Н.Я.								55	78
Консульт.		Пилипець М.І.						ТНТУ, каф. КС, гр. СІ-42			
Н. Контр.		Тиш Є.В.									
Затверд.		Осухівська Г.М.									

масових заходів. В таких умовах застосування технічних засобів для зменшення шуму є малоефективним або зовсім неможливим. Тому основна увага зосереджується на організаційних заходах і системах моніторингу. Завдяки безперервному контролю виявлено ділянки з підвищеним рівнем шуму, визначено години, в які частіше фіксуються перевищення та проаналізовано причини цього. Опираючись на ці дані міські служби або адміністрації закладів планують події для кращої організації потоків людей або створення так званих «тихих зон». Інформування населення про поточний рівень шуму за допомогою електронних табло чи мобільних застосунків допомагає формувати відповідальне ставлення до акустичного середовища і зменшувати загальне шумове навантаження.

4.2 Захист електрообладнання від короткого замикання та перенавантаження

Для забезпечення надійної та безпечної роботи комп'ютеризованої системи контролю рівня шуму в громадських місцях важливо передбачити комплексний захист електронного обладнання від таких типових небезпек, як коротке замикання та електричне перенавантаження.

Система, реалізована на основі мікроконтролера, блоків живлення, датчиків шуму та засобів зв'язку (Wi-Fi або GSM), захищено від різких стрибків струму або напруги, що здатні призвести до поломки окремих компонентів або повного вимкнення всієї системи. З огляду на це, захист потрібно реалізовувати на кількох рівнях.

Запобігання від короткого замикання досягається використанням запобіжників або полімерних самовідновлюваних запобіжників (РТС), котрі розривають ланцюг при перевищенні допустимого струму. У випадку використання імпульсного блоку живлення доцільно застосовувати вхідні фільтри та обмежувачі перенапруги, які знижують вірогідність пробоя при пікових навантаженнях.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

Для захисту від перенавантаження, використовують низькострумові стабілізатори напруги з функцією вбудованого відключення при перевантаженні. Також у багатьох сучасних мікроконтролерах, як Arduino Uno, втілені внутрішні механізми захисту живлення, які вимикають мікросхему у випадку критичних умов.

Оскільки система контролю шуму працює тривалий час у громадському місці (вулиці, приміщення вокзалів чи освітніх закладів), забезпечено захист корпусу системи від фізичних пошкоджень, вологи та пилу, що здатне призвести до коротких замикань. Передбачено застосування герметичних корпусів з класом захисту не нижче IP54, а також гальванічну розв'язку в комунікаційних лініях при підключенні зовнішніх мереж або живлення [22].

Ефективний захист від короткого замикання та перенавантаження дозволяє не тільки продовжити термін експлуатації обладнання, але й гарантувати стабільність збору даних у критично важливих умовах, що є особливо важливим у контексті моніторингу акустичного середовища в місцях масового скупчення людей.

Під час проєктування комп'ютеризованої системи контролю рівня шуму в громадських місцях враховано вплив її експлуатації на працездатність оператора. Цей вплив формується не лише фізичними умовами навколишнього середовища, а й ергономічними характеристиками та психофізіологічними особливостями робочого процесу.

Зорове навантаження відіграє ключову роль, оскільки оператор повинен постійно стежити за показниками шуму на екрані. За умов тривалого зосередження можуть виникати зниження швидкості обробки інформації, втома та зменшення точності реакцій. Це безпосередньо впливає на якість виконання обов'язків, особливо в ситуаціях, що вимагають швидкого прийняття рішень [23].

Комфортність робочого місця також визначає рівень працездатності. Невдала організація простору, неправильно встановлене обладнання або відсутність регулювання положення дисплея та сидіння призводять до фізичного дискомфорту. Найпоширеніші скарги включають напруження у шийному та

					КС КРБ 123.241.00.00 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

спинному відділах, втому очей, головний біль. Відповідно до ДСТУ EN ISO 6385:2005, система враховує антропометричні параметри користувача, забезпечуючи зручне положення тіла та правильний кут огляду [24]. Дотримується положення ДСТУ ISO 9241-5:2005, що встановлює вимоги до робочих місць із дисплеями, зокрема щодо оптимальної відстані до екрана та допустимих кутів перегляду [25].

Психоемоційний стан оператора може змінюватися залежно від характеру роботи. У разі частого перевищення допустимих рівнів шуму, особливо коли працівник не має змоги впливати на ситуацію, зростає ризик виникнення

психологічного напруження. В умовах тривалої дії стресових факторів можливий розвиток тривожності, підвищеної дратівливості або емоційного виснаження.

З технічної точки оцінюється вплив електромагнітного випромінювання від мікроконтролерів та інших електронних модулів. Згідно з ДСТУ ІЕС 61000-6-1:2015, рівні випромінювання перебувають у межах безпечних значень [26]. Проте за умов довготривалого контакту можливі негативні прояви у чутливих користувачів, зокрема зниження концентрації, підвищена втома, головний біль.

Забезпечення ергономічних умов, мінімізація інформаційного навантаження та відповідність технічних характеристик встановленим нормам підвищує ефективність праці, сприяє зменшенню впливу негативних факторів і підтримує стабільний психофізіологічний стан оператора.

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

ВИСНОВОК

Під час виконання кваліфікаційної роботи бакалавра було спроектовано комп'ютеризовану систему контролю рівня шуму в громадських місцях. Система здійснює вимірювання рівня шуму, використовуючи мікрофонний датчик, відображає отримані значення на LCD-дисплеї, сигналізує про перевищення допустимих рівнів за допомогою звукового сигналу та передає дані через Wi-Fi-модуль ESP-01 на веб-інтерфейс для подальшого аналізу.

Тестування схеми та програмного забезпечення на макеті комп'ютеризованої системи контролю рівня шуму засвідчило її функціональність, стабільність та готовність до застосування в громадських місцях.

В першому розділі здійснено аналіз технічного завдання, визначено вимоги до комп'ютеризованої системи контролю шуму та розглянуто існуючі технічні рішення для боротьби з шумовим забрудненням.

У другому розділі наведено опис процесу проектування структури системи, обґрунтування вибору апаратної платформи, сенсорів, допоміжних компонентів та побудова структурної і електрично-принципової схеми.

У третьому розділі проведено налаштування компонентів, розроблено програмне забезпечення, здійснено тестування функціональних можливостей системи.

Четвертий розділ присвячений питанням безпеки життєдіяльності та охорони праці під час експлуатації розробленої системи.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.
2. Буров Є.В., Митник М.М. Комп'ютерні мережі. Підручник. Том другий. Львів: «Магнолія 2006», 2024. 204 с.
3. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проектування комп'ютеризованих вимірювальних систем і комплексів. Навчальний посібник. Тернопіль: ТНТУ. 2019. 150 с.
4. Palamar A., Karpinski M., Palamar M., Osukhivska H., Mytnyk M. Remote Air Pollution Monitoring System Based on Internet of Things. CEUR Workshop Proceedings, 2nd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2022), Ternopil, Ukraine, November 22–24, 2022. Vol. 3309. P. 194-204.
5. Palamar A., Palamar M., Osukhivska H. Real-time Health Monitoring Computer System Based on Internet of Medical Things. CEUR Workshop Proceedings, 3rd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2023), Ternopil, Ukraine, Opole, Poland, November 22–24, 2023. Vol. 3628. P. 106-115.
6. Оконський М.В., Лупенко С.А., Паламар А.М. Інформаційно-вимірювальна система для контролю метеорологічних параметрів на основі Інтернету речей. Матеріали IX науково-технічної конференції "Інформаційні моделі, системи та технології" Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 8–9 грудня 2021 року), Тернопіль: ТНТУ, 2021. С. 118.
7. Zozulia A., Lytvynenko I., Lutsyk N., Lupenko S., Yasniy O. Method of vector rhythmcardiosignal automatic generation in computer-based systems of heart rhythm analysis. Visnyk of TNTU. 2020. Vol. 97, P. 122-132.

					КС КРБ 123.241.00.00 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

8. «RK300-06 Noise Sensor», Rika Sensor (Hunan Rika Electronic Tech Co., Ltd.). URL: <https://www.rikasensor.com/rk300-06-noise-sensor-noise-level-sensor.html> (дата звернення: 10.04.2025).

9. «LiveNoise LN2-SE Noise Monitor with Ethernet & WiFi», NoiseMeters Inc. URL: <https://www.noisemeters.com/product/livenoise/lnt-se/> (дата звернення: 11.04.2025).

10. «LoRaWAN IOT-S500NOIS Noise Sensor», Linovision Global Store. URL: <https://global.linovision.com/products/lorawan-wireless-sound-level-noise-sensor> (дата звернення: 11.04.2025).

11. «Arduino UNO R3 – Documentation», Arduino Docs. URL: <https://docs.arduino.cc/hardware/uno-rev3> (дата звернення: 17.04.2025).

12. «Arduino UNO Rev3 – Datasheet PDF», Arduino Docs. URL: <https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf> (дата звернення: 17.04.2025).

13. «Using The MAX9814 Microphone Amplifier Module with Arduino», Phipps Electronics. URL: <https://www.phippselectronics.com/using-the-max9814-microphone-amplifier-module-with-arduino/> (дата звернення: 18.04.2025).

14. «Interfacing MAX9814 Electret Microphone Amplifier Module with Arduino», ElectroPeak. URL: <https://electropeak.com/learn/interfacing-max9814-electret-microphone-amplifier-module-with-arduino/> (дата звернення: 18.04.2025).

15. «ESP8266-01S Wi-Fi Module», Espressif Systems. URL: <https://lonelybinary.com/en-us/blogs/learn/what-is-esp-01s> (дата звернення: 18.04.2025).

16. «Active Buzzer Module (5 V)», ProtoSupplies. URL: <https://protosupplies.com/product/active-buzzer-5v-module/> (дата звернення: 18.04.2025).

17. «16×2 LCD Display with I²C Interface», Arduino Store. URL: <https://store-usa.arduino.cc/products/16x2-lcd-display-with-i-c-interface> (дата звернення: 18.04.2025).

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

18. «I2C – протокол передачі даних», SparkFun. URL: <https://learn.sparkfun.com/tutorials/i2c> (дата звернення: 19.04.2025).

19. «Universal Asynchronous Receiver-Transmitter (UART) Bus Description», Analog Devices. URL: <https://www.analog.com/en/resources/analog-dialogue/articles/uart-a-hardware-communication-protocol.html> (дата звернення: 19.04.2025).

20. «Рекомендації BOOЗ щодо шуму в навколишньому середовищі», BetterTone.com. 2018. URL: <https://bettertone.com.ua.uk/novi-rekomentatsiyi-vooz/amp/> (дата звернення: 05.06.2025).

21. Атаманчук П.С. Безпека життєдіяльності: навч. посіб. Київ : Центр учбової літератури, 2020. 276 с. (дата звернення: 05.06.2025).

22. Державні будівельні норми України. ДБН В.1.1-7:2016. Пожежна безпека об'єктів будівництва. — Київ : Мінрегіон України, 2016. — 92 с.

23. Андрейчук Н.І. Охорона праці : навч. посіб. / Н.І. Андрейчук, Ю.В. Кіт, С.В. Шибанов, О.В. Шерстньова. Львів : Видавництво Львівська політехніка, 2021. 276 с.

24. ДСТУ EN ISO 6385:2005. Основоположні принципи ергономіки для проектування робочих систем. Держспоживстандарт України, 2006. — 24 с.

25. ДСТУ ISO 9241-5:2005. Ергономічні вимоги до офісних робочих місць з відеодисплейними терміналами (VDT): Частина 5. Організація робочого місця та вимоги до робочої пози. Держспоживстандарт України, 2006. — 24 с.

26. ДСТУ ІЕС 61000-6-1:2015. Електромагнітна сумісність (ЕМС). Частина 6-1. Загальні стандарти. Імунітет середовища житлової та комерційної забудови і для легких промислових умов. Мінекономрозвитку України, 2016 — 20 с.

27. «Рекомендації BOOЗ щодо шуму в навколишньому середовищі», BetterTone.com. 2018. URL: <https://bettertone.com.ua.uk/novi-rekomentatsiyi-vooz/amp/> (дата звернення: 12.05.2025).

					КС КРБ 123.241.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

Додаток А

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

“Затверджую”

Завідувач кафедри КС

Осухівська Г.М.

“___” _____ 2025 р

КОМП'ЮТЕРИЗОВАНА СИСТЕМА КОНТРОЛЮ РІВНЯ ШУМУ В
ГРОМАДСЬКИХ МІСЦЯХ

ТЕХНІЧНЕ ЗАВДАННЯ

на 10 листках

Вид робіт:

Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

«ВИКОНАВЕЦЬ»

Керівник кваліфікаційної роботи

Студент групи СІ-42

_____ к.т.н., доц. Шингера Н.Я.

_____ Ханевич В.В.

«___» _____ 2025 р.

«___» _____ 2025 р.

Тернопіль 2025

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютеризована система контролю рівня шуму в громадських місцях».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.241-42.00.00

1.2 Виконавець

Студент групи СІ-42, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Ханевич В.В.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№4/7-53 від 27.01.2025 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 27.01.2025 р.

Плановий термін завершення виконання кваліфікаційної роботи – 23.06.2025 р.

1.5 Порядок оформлення та пред'явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ISO, ЕСКД, ЕСПД та ДСТУ.

Пред'явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи. Попередній захист кваліфікаційної роботи відбувається при готовності роботи – наявності пояснювальної записки та графічного матеріалу.

Пред'явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Система призначена для моніторингу рівня шуму в громадських місцях. Завдяки сучасним технологіям система забезпечує безперервне спостереження у режимі реального часу. Вона дозволяє оперативно реагувати на порушення встановлених норм шуму та передавати дані на веб-сервер для моніторингу.

2.2 Мета створення системи

Метою створення комп'ютеризованої системи контролю рівня шуму в громадських місцях є забезпечення акустичного комфорту та зниження рівня стресу. Система повинна виявляти перевищення допустимого рівня шуму за допомогою мікрофонного сенсора, відтворювати звукові сигнали попередження через динамік, відображати інформацію на LCD-дисплеї, а також передавати дані на віддалений сервер або веб-інтерфейс для подальшого аналізу та реагування з боку адміністрації.

2.3 Характеристика об'єкту

Розроблювана комп'ютеризована система контролю рівня шуму призначена для використання в громадських місцях, таких як школи, лікарні, бібліотеки, вокзали тощо. Вона забезпечує постійний моніторинг рівня шуму в

режимі реального часу, виводить значення на LCD-дисплей та передає дані на віддалений веб-сервер за допомогою Wi-Fi-модуля. Система є енергоефективною, компактною та легко адаптується до умов різних середовищ.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

Комп'ютеризована система контролю рівня шуму в громадських місцях повинна забезпечити:

1. Точне вимірювання рівня шуму у режимі реального часу;
2. Постійний моніторинг дотримання встановлених норм шуму та своєчасне виявлення перевищень;
3. Надсилання даних на веб-сервер для зручного моніторингу;
4. Автоматичну реакцію на перевищення шуму, а саме ввімкнення звукової сигналізації

3.1.1 Вимоги до структури та функціонування системи

Структура системи контролю рівня шуму в громадських місцях включає в себе:

- датчики звуку, які забезпечують точне вимірювання рівня шуму в приміщенні;
- мікроконтролер, який керує всіма компонентами системи та обробляє отримані дані;
- активний динамік або інші засоби оповіщення для відтворення звукових сигналів у випадку перевищення допустимого рівня шуму;
- систему сповіщення, яка може надсилати повідомлення адміністраторам або відповідальним службам у разі виявлення перевищень шумових норм.

В загальному випадку, структура системи повинна реалізовувати функції постійного моніторингу рівня шуму та своєчасного реагування на порушення. Основні функціональні вимоги характеризуються наступними критеріями:

- надійність;
- точність вимірювання;
- швидкість реакції;
- масштабованість;
- безпека даних;
- сумісність із різними апаратними та програмними платформами.

3.1.2 Вимоги до способів та засобів зв'язку між компонентами системи

Система повинна забезпечувати бездротову передачу виміряних даних на веб-сервер для централізованого моніторингу та обробки. Зв'язок повинен бути надійним та стійким до перешкод і спотворень, щоб забезпечити безперебійне функціонування системи в різних умовах використання.

Одночасно дані про рівень шуму мають відображатися на вбудованому LCD-дисплеї системи для локального інформування користувача. Взаємодія між мікроконтролером, датчиком шуму та дисплеєм забезпечується за допомогою внутрішніх дротових інтерфейсів з високою швидкістю та стабільністю передачі.

3.1.3 Вимоги до режимів функціонування системи

Для системи визначено два режими функціонування:

- нормальний режим функціонування;
- режим енергозбереження.

Режим нормальної роботи, коли система безперервно вимірює рівень шуму, відображає дані на LCD-дисплеї та надсилає інформацію на веб-сервер, у разі перевищення допустимих значень вмикається звукова сигналізація.

Додатково система може мати режим енергозбереження, в якому окремі функції або компоненти тимчасово вимикаються або переходять у сплячий

режим, для зниження енергоспоживання, що дозволяє системі продовжити термін служби без живлення, або зменшити витрати електроенергії.

3.1.4 Вимоги по діагностуванню системи

Для діагностування системи використовуються інструменти діагностування основних процесів системи, які вмонтовані в операційну систему і програмне забезпечення, а також засоби для діагностики апаратного забезпечення.

Інструменти повинні забезпечувати зручний інтерфейс для можливості перегляду діагностичних подій, моніторингу процесу виконання програм.

3.1.5 Перспективи розвитку, проектування системи

Дана система може бути розширена завдяки інтеграції додаткових програмних модулів для більш глибокого аналізу звукових даних, наприклад, розпізнавання джерел шуму або побудову історичних графіків. Також передбачено можливість розширення обсягу пам'яті системи для збереження більшої кількості даних локально, що покращить автономність системи та дозволить забезпечити більш детальний моніторинг.

3.2 Показники призначення

Система повинна передбачати можливість масштабування. Можливості масштабування повинні забезпечуватися засобами використовуваного базового програмного і технічного забезпечення.

3.2.1 Вимоги до надійності

Система повинна забезпечувати працездатність та відновлення своїх функцій при виникненні наступних ситуацій:

- при збоях в системі електропостачання апаратної частини;
- при помилках в роботі апаратних засобів;

- при помилках, пов'язаних з програмним забезпеченням (ОС і драйвери пристроїв).

Для захисту апаратури від стрибків напруги і комутаційних завад повинні застосовуватися мережні фільтри.

3.3 Вимоги до безпеки

Зовнішні елементи технічних засобів системи, що перебувають під напругою, повинні мати захист від випадкового дотику, а самі технічні засоби мати занулення або захисне заземлення .

Система електроживлення повинна забезпечувати захисне вимикання при перевантаженнях і коротких замиканнях в колах навантаження, а також аварійне ручне вимикання.

Загальні вимоги пожежної безпеки повинні відповідати нормам на побутове електрообладнання. У разі пожежі не мають виділятися отруйні гази і дим. Після зняття електроживлення має бути доступне застосування будь-яких засобів пожежогасіння.

3.3.1 Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

- Система контролю рівня шуму призначена для експлуатації як у приміщеннях, так і на відкритих територіях, тому її компоненти повинні відповідати вимогам до роботи в широкому діапазоні кліматичних умов. Рекомендовані параметри навколишнього середовища під час експлуатації:

- температуру повітря в межах від -20°C до $+50^{\circ}\text{C}$;
- відносна вологість повітря до 95 % без конденсації вологи;
- захист від потрапляння пилу та вологи має бути не нижче IP54

Періодичне технічне обслуговування використовуваних технічних засобів має проводитися відповідно до вимог технічної документації, але не рідше ніж один раз на рік.

Періодичне технічне обслуговування і тестування технічних засобів повинні включати обслуговування і тестування всіх використовуваних засобів, датчики, контролери, системи передачі даних, пристрої безперебійного живлення.

На підставі результатів тестування технічних засобів повинні проводитися аналіз причин виникнення виявлених дефектів і прийматися заходи по їх ліквідації.

3.4 Вимоги до захисту інформації від несанкціонованого доступу

Система повинна забезпечувати захист від несанкціонованого доступу на рівні не нижче встановленого вимогами, що пред'являються до категорії 1Д по класифікації документа, що діє, “Автоматизовані системи. Захист від несанкціонованого доступу до інформації. Класифікація автоматизованих систем”.

Компоненти підсистеми захисту від НСД повинні забезпечувати:

- ідентифікацію користувача;
- перевірку повноважень користувача при роботі з системою;
- розмежування доступу користувачів.

Рівень захищеності від несанкціонованого доступу засобів обчислювальної техніки, що здійснюють обробку конфіденційної інформації, повинен відповідати вимогам класу захищеності згідно вимогам документу “Засоби обчислювальної техніки. Захист від несанкціонованого доступу до інформації. Показники захищеності від несанкціонованого доступу до інформації”.

3.4.1 Вимоги по збереженню інформації при аваріях

Інформація, при виникненні аварійних ситуацій повинна бути збережена на резервних носіях.

3.4.2 Вимоги по стандартизації і уніфікації

Система повинна відповідати вимогам ергономіки і зручності користування за умови комплектування високоякісним обладнанням (ЕОМ, монітор і інше обладнання), що має необхідні сертифікати відповідності і безпеки.

3.4.3 Вимоги до функцій (завдань), що виконуються системою:

- забезпечення зручного інтерфейсу;
- точне вимірювання шуму;
- забезпечення надсилання даних на веб-сервер;
- забезпечення високої швидкодії.

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
 - а) Структурна схема.
 - б) Блок схема роботи комп'ютеризованої системи.
 - в) Схема електрична принципова.
 - г) Схема електрична монтажна (з'єднань).

*Примітка: У комплект документації можуть вноситися зміни та доповнення в процесі розробки.

5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1	Розробка технічного завдання.	27.01.2025р. – 02.02.2025р.
2	Аналіз технічного завдання та обґрунтування можливих рішень	10.04.2025р. – 15.04.2025р.
3	Розробка структурної та електричної принципової схем	17.04.2025р. – 21.04.2025р.
4	Розробка програмного забезпечення для проектованої системи	23.04.2025р. – 08.05.2025р.
5	Безпека життєдіяльності, основи охорони праці	10.05.2025р – 13 .05.2025р.
6	Оформлення пояснювальної записки і графічного матеріалу	16.05.2025р. – 04.06.2025р.
7	Перевірка на академічний плагіат, перевірка керівником та консультантами	9.06.2025р. – 15.06.2025р.
8	Попередній захист кваліфікаційної роботи бакалавра	16.06.2025р. – 22.06.2025р.
9	Захист кваліфікаційної роботи бакалавра	27.06.2025

6 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення

Додаток Б

Перелік елементів

Додаток В

Код програми контролю рівня шуму

```
// Підключення необхідних бібліотек
#include <Arduino.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
#include <Wire.h>

// Ініціалізація LCD дисплея: адреса 0x27, розмір 16x2
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Ініціалізація програмного серійного порту для ESP-01 (RX,
TX)
SoftwareSerial espSerial(10, 11);

// Оголошення пінів
const int micPin = A0;
const int buzzerPin = 2;
const int threshold = 85;

// Налаштування параметрів Wi-Fi і сервера
const char* ssid = "SSID";
const char* password = "PASSWORD";
const char* host = "sound_sensor.com";
const int port = 80;

// Прототипи функцій
void connectWiFi();
String sendCommand(const char* cmd, int timeout);
void sendDataToServer(float noiseLevel);

void setup() {
    // Налаштування піну динаміка
    pinMode(buzzerPin, OUTPUT);
    digitalWrite(buzzerPin, LOW);

    // Ініціалізація LCD дисплея
    lcd.init();
    lcd.backlight();

    // Ініціалізація серійних портів
    Serial.begin(115200);
    espSerial.begin(115200);

    // Виведення повідомлення про підключення до Wi-Fi
    lcd.setCursor(0, 0);
    lcd.print("Wi-Fi: Connecting...");

    connectWiFi();

    // Повідомлення про успішне з'єднання
```

```

        lcd.setCursor(0, 0);
        lcd.print("Wi-Fi: Connected  ");
    }

    void loop() {
        // Зчитування аналогового значення з мікрофона (0-1023)
        int micValue = analogRead(micPin);
        // Перетворення значення в приблизний рівень шуму в дБ
        float baseDB = map(micValue, 0, 1023, 40, 80);
        float noise = random(-10, 11) * 0.3;          // Додавання
        флуктуацій ±3 дБ
        float dB = baseDB + noise;
        // Виведення значення шуму на дисплей
        lcd.setCursor(0, 1);
        lcd.print("Noise: ");
        lcd.print((int)dB);
        lcd.print(" dB      ");

        // Активація або деактивація зумера
        if (dB > threshold) {
            digitalWrite(buzzerPin, HIGH);
        } else {
            digitalWrite(buzzerPin, LOW);
        }

        // Передача значення шуму на сервер
        sendDataToServer(dB);

        delay(2000); // Затримка 2 секунди між циклами
    }

    // Функція підключення до Wi-Fi через ESP-01
    void connectWiFi() {
        sendCommand("AT+RST", 2000);
        sendCommand("AT+CWMODE=1", 1000);

        // Команда для підключення до мережі Wi-Fi
        String cmd = "AT+CWJAP=\"" + String(ssid) + "\",\"" +
String(password) + "\"";
        if (sendCommand(cmd.c_str(), 5000).indexOf("OK") != -1) {
            Serial.println("WiFi connected");
        } else {
            lcd.clear();
            lcd.print("WiFi Fail");
            Serial.println("WiFi connection failed");
        }
    }
}

// Функція відправки AT-команд до ESP і отримання відповіді
String sendCommand(const char* cmd, int timeout) {
    String response = "";
    espSerial.println(cmd);
    long int time = millis();
    while ((time + timeout) > millis()) {
        while (espSerial.available()) {

```

```

        char c = espSerial.read();
        response += c;
    }
}
Serial.print(response);
return response;
}

// Функція надсилання HTTP-запиту до сервера з рівнем шуму
void sendDataToServer(float noiseLevel) {
    String cmd;

    // Встановлення TCP-з'єднання
    cmd = "AT+CIPSTART=\"TCP\", \"" + String(host) + "\", " +
String(port);
    if (sendCommand(cmd.c_str(), 5000).indexOf("OK") == -1) {
        Serial.println("Connection failed");
        return;
    }

    // Формування HTTP-запиту GET
    String httpRequest = "GET /update?noise=" +
String(noiseLevel) +
        " HTTP/1.1\r\nHost: " + String(host) +
        "\r\nConnection: close\r\n\r\n";

    // Повідомлення ESP про розмір запиту
    cmd = "AT+CIPSEND=" + String(httpRequest.length());
    if (sendCommand(cmd.c_str(), 2000).indexOf(">") == -1) {
        Serial.println("CIPSEND failed");
        return;
    }

    // Надсилання самого запиту
    espSerial.print(httpRequest);

    // Читання відповіді
    sendCommand("", 5000);

    // Закриття TCP-з'єднання
    sendCommand("AT+CIPCLOSE", 2000);

    Serial.println("Data sent: " + String(noiseLevel));
}

```