

КВАЛІФІКАЦІЙНА РОБОТА

Кіберфізична система "Розумна каналізаційна система"

Назва теми

Рівень вищої освіти перший (бакалаврський)

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія та програмування»

Назва

Шифр КвРКІ. 302134.23.10.10 ПЗ

Виконав здобувач IV курсу, група КІ2с-22-1


Підпис

Роман НАУМЕНКО
Ініціали, прізвище

Керівник


Підпис

Катерина
БЕРЕЗЬКА
Ініціали, прізвище

Науковий ступінь, учене звання


Підпис

Сергій ЛИСЕНКО
Ініціали, прізвище

Нормоконтролер

Науковий ступінь, учене звання


Підпис

Ольга ПАВЛОВА
Ініціали, прізвище

До захисту допускаю:
завідувач кафедри КІС

«01» червня 2026 р.

дата

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Рівень вищої освіти ПЕРШОЇ (БАКАЛАВРСЬКИЙ)

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Завідувачка кафедри КІПС


Ольга ПАВЛОВА

“ 10 ” 01 2026 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Науменку Роману Сергійовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Кіберфізична система “Розумна каналізаційна система”

Керівник проекту (роботи) Березька Катерина Миколаївна, к.т.н., доц.

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

Затверджена наказом ректора університету від 01.03.2026 р. № 5

2. Термін подання здобувачем роботи на кафедру 01.06.2026 р.

3. Вихідні дані до роботи Завдання на кваліфікаційну роботу

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

Аналіз предметної області

Проектування та розроблення кіберфізичної системи

Практична реалізація та перевірка роботи кіберфізичної системи

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

Схема структурна, схема функційна

Схема електрична принципова

Інтерфейс ПЗ

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання « 10 » 01 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	10.01.2026	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2026	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2026	виконано
4	Робота над розділом 2 – вибір компонентів для проектування системи керування якістю повітря в приміщеннях на основі мультисенсорних вимірювань	01.04.2026	виконано
5	Робота над розділом 3 – проектування системи керування якістю повітря в приміщеннях на основі мультисенсорних вимірювань	29.04.2026	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2026	виконано
7	Попередній захист ВКР	26.05.2025	виконано
8	Захист ВКР на засіданні ЕК	Червень 2026 року	

Здобувач

Підпис

Роман НАУМЕНКО

Імя, ПРІЗВИЩЕ

Керівник кваліфікаційної роботи

Підпис

Катерина БЕРЕЗЬКА

Імя, ПРІЗВИЩЕ

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Кіберфізична система “Розумна каналізаційна система”».

Автор роботи: Роман НАУМЕНКО.

Керівник роботи: Катерина БЕРЕЗЬКА.

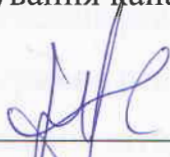
Пояснювальна записка: 73 с., 6 рис., 4 табл., 3 дод., 50 джерел.

Графічна частина: 3 креслення.

КІБЕРФІЗИЧНА СИСТЕМА, КОНДИЦІОНУВАННЯ, МІКРОКОНТРОЛЕР ESP32, МУЛЬТИСЕНСОРНІ ВИМІРЮВАННЯ, ОЧИЩЕННЯ ПОВІТРЯ, СЕНСОРИ, ЯКІСТЬ ВОДИ.

Кваліфікаційна робота бакалавра присвячена розробці кіберфізичної системи “Розумна каналізаційна система”.

Метою роботи є проектування, реалізація та перевірка функціонування програмно-апаратної кіберфізичної системи, що забезпечує збір, передавання, оброблення, візуалізацію та використання мультисенсорних даних для функціонування каналізаційної системи.







Підпис здобувача

30.05.2026

Дата

ЗМІСТ

Вступ	3
1 аналіз предметної області	4
1.1 Аналіз задачі розумної каналізаційної системи	4
1.2 кіберфізична система «розумна каналізаційна система»	6
1.3 висновок.....	10
2 Проєктування та розроблення кіберфізичної системи	11
2.1 Загальний опис складу кіберфізичної системи	11
2.2 Апаратні компоненти сенсорного вузла	12
2.3 Апаратні компоненти мережевого рівня.....	24
2.4 Серверна інфраструктура	25
2.5 Програмні засоби та середовища розробки	27
2.6 Висновки до розділу 2.....	34
3 Практична реалізація та перевірка роботи кіберфізичної системи	36
3.1 Озробка структурної схеми системи	36
3.2 Розробка функціональної схеми сенсорного вузла.....	38
3.3 Розробка принципової електричної схеми сенсорного вузла	42
3.4 Програмна реалізація прошивки сенсорного вузла	47
3.5 Програмна реалізація серверної частини	55
3.6 Реалізація інтерфейсу моніторингу	65
3.7 Фізичне складання та комплексне тестування системи	68
3.8 Висновки до розділу 3.....	73
Висновки.....	75
Перелік джерел посилань.....	77
Додаток А Копія креслення «Схема структурна, схема функційна».....	84
Додаток Б Копія креслення «Схема електрична принципова».....	85
Додаток В Копія креслення «Інтерфейс ПЗ».....	86

КвРКІ. 302134.23.10.10 ПЗ							
Вм.	Арк.	№докум.	Підпис	Кіберфізична система “Розумна каналізаційна система”. Пояснювальна записка	Літера	Аркуші	Аркушів
Виконав		Роман НАУМЕНКО			у		
Перевід.		Катерина БЕРЕЗЬКА			у	2	73
Н.контр.		Сергій ЛИСЕНКО			ХНУ КІ2С-23-1		
Ватвед.		Ольга ПАВЛОВА					

ВСТУП

У сучасних умовах урбанізації та промислового зростання стан міських інженерних мереж стає критичним фактором екологічної безпеки. Традиційні методи контролю стічних вод, що базуються на періодичному ручному відборі проб, не здатні забезпечити оперативне реагування на короточасні, але токсичні скиди промислових підприємств. Впровадження кіберфізичної системи (КФС) «Розумна каналізаційна система» дозволяє трансформувати пасивну інфраструктуру в інтелектуальну мережу, здатну до самодіагностики та миттєвого виявлення порушень екологічних норм [1-3].

Об'єкт дослідження – процес моніторингу та управління якістю стічних вод у міських каналізаційних мережах.

Предмет дослідження – архітектура, алгоритми та сенсорні технології кіберфізичної системи, що інтегрує фізичні процеси фільтрації/транспортування води з обчислювальними потужностями для аналізу хімічного складу в реальному часі.

Мета роботи – автоматизована ідентифікація джерел незаконних викидів та прогнозування поширення забруднюючих речовин шляхом розроблення кіберфізичної системи «Розумна каналізаційна система».

Для досягнення мети поставлено такі завдання:

- проаналізувати сучасний стан моніторингу стічних вод та виявити основні проблеми ідентифікації незаконних викидів;
- визначити оптимальний набір хімічних та фізичних параметрів (рН, електропровідність, наявність важких металів тощо) для безперервного контролю;
- описати механізм взаємодії між фізичним рівнем (датчиками) та кібернетичним рівнем (аналітичними хмарами) для обробки великих даних (Big Data);
- оцінити економічний та екологічний ефект від впровадження системи.

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						3
Зм.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз задачі розумної каналізаційної системи

Аналіз задачі побудови кіберфізичної системи «Розумна каналізаційна система» починається з детального вивчення архітектурних особливостей міської інфраструктури, яка зазвичай є закритою та важкодоступною для постійного нагляду [4-5].

Основною проблемою є ретроспективний характер сучасного екологічного моніторингу, коли факт скидання небезпечних речовин виявляється лише після потрапляння стічних вод на очисні споруди або у відкриті водойми.

Це створює ситуацію, за якої ідентифікація конкретного джерела забруднення стає майже неможливою через змішування потоків від різних підприємств у спільній мережі.

Отже, першочерговим завданням аналізу є визначення критичних точок розгалуження колекторів, де встановлення датчиків забезпечить максимальну інформативність щодо походження стічних мас [6-8].

Технічна складність задачі полягає у виборі сенсорного обладнання, яке здатне тривалий час функціонувати в агресивному середовищі з високою вологістю та змінною концентрацією газів.

Датчики повинні не просто фіксувати наявність сторонніх домішок, а проводити багатопараметричний аналіз хімічного складу, включаючи рівень кислотності, електропровідність та специфічні маркери промислових відходів. Важливо враховувати, що кожен тип виробництва залишає свій унікальний хімічний слід, тому система аналізу має розпізнавати патерни, характерні для металургійних, хімічних або харчових підприємств. Це вимагає розробки складної математичної моделі, яка буде відсіювати природні коливання складу стічних вод від цілеспрямованих незаконних викидів [9-11].

Кібернетична складова системи повинна вирішувати проблему передачі даних із підземних комунікацій, де проходження радіосигналів суттєво обмежене товщею бетону та ґрунту. Аналіз задачі передбачає вибір протоколів

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						4
Зм.	Арк.	№ док.ум.	Підпис	Дата		

зв'язку з низьким енергоспоживанням та високою проникаючою здатністю, що дозволить датчикам працювати автономно протягом тривалого часу без заміни елементів живлення. Отримана інформація має надходити в єдиний аналітичний центр у режимі реального часу, забезпечуючи візуалізацію стану мережі на цифровій карті міста. Це дозволить оператору миттєво реагувати на аномалії та направляти інспекційні групи безпосередньо до місця порушення ще до того, як речовина розчиниться у загальному потоці.

Важливим аспектом аналізу є створення алгоритмів обробки великих даних, які здатні корелювати показники датчиків із графіками роботи підприємств та погодними умовами. Наприклад, під час сильних злив концентрація забруднювачів може знижуватися через розведення дощовою водою, що потребує динамічного коригування порогів спрацювання сигналізації.

Система має враховувати гідравлічні моделі руху рідини в трубах, щоб точно розраховувати час добігання забрудненої плями від точки скиду до контрольного створу [12-15].

Таким чином, аналітичний блок стає «мозком» системи, що перетворює сирі дані від сенсорів на юридично значущі докази екологічного правопорушення.

Крім технічних параметрів, аналіз задачі охоплює питання кібербезпеки, оскільки доступ до управління каналізаційною системою є критично важливим для життєдіяльності міста. Будь-яке втручання в роботу датчиків або фальсифікація даних можуть призвести до приховування реальних екологічних злочинів або створення паніки через хибні тривоги. Необхідно розробити протоколи шифрування та багаторівневу систему автентифікації для захисту цілісності інформації.

Кінцевим результатом аналізу є формування цілісного бачення системи, яка не лише фіксує порушення, а й виступає превентивним інструментом, змушуючи підприємства дотримуватися екологічних стандартів під загрозою неминучого виявлення.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 5
Зм.	Арк.	№ докум.	Підпис	Дата		

1.2 Кіберфізична система «Розумна каналізаційна система»

Обґрунтування актуальності розробки кіберфізичної системи «Розумна каналізаційна система» впливає з критичного стану глобальної екологічної безпеки та зростаючого навантаження на міські екосистеми.

Традиційні методи контролю якості стічних вод, що базуються на вибірковому ручному відборі проб, безнадійно застаріли та не відповідають динаміці сучасного промислового виробництва [16-19].

Відсутність постійного моніторингу створює умови для безкарного скидання високотоксичних відходів, оскільки ймовірність фіксації порушення в момент його здійснення залишається вкрай низькою. Це призводить до систематичного отруєння водних ресурсів, що має довготривалі негативні наслідки для здоров'я населення та біорізноманіття регіону.

Впровадження інтелектуальних датчиків аналізу хімічного складу в реальному часі дозволяє вирішити проблему ідентифікації джерела забруднення, яка раніше була головною перешкодою для притягнення порушників до відповідальності. Кіберфізична система автоматично фіксує «хімічний відбиток» скиду, що робить процес моніторингу прозорим та позбавленим людського фактора чи корупційної складової.

Актуальність розробки підсилюється необхідністю захисту дороговартісного обладнання міських очисних споруд, яке часто виходить з ладу через непередбачувані пікові концентрації агресивних хімікатів. Своєчасне виявлення небезпечних сполук дозволяє автоматично перенаправляти або блокувати забруднені потоки, запобігаючи техногенним аваріям масштабного характеру.

Економічний аспект актуальності полягає у значній оптимізації витрат на ліквідацію наслідків екологічних катастроф та експлуатацію муніципальних мереж. Використання прогнозної аналітики на основі зібраних даних дає змогу виявляти незаконні підключення до каналізації та точніше розраховувати навантаження на систему під час інтенсивних опадів. Це перетворює каналізацію

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						6
Зм.	Арк.	№ докум.	Підпис	Дата		

з пасивного об'єкта інфраструктури на активне джерело даних, що сприяє концепції «Розумного міста».

Цифровізація підземних комунікацій стає необхідним кроком для модернізації міського господарства відповідно до європейських стандартів сталого розвитку [20-23].

Соціальна значущість проекту обумовлена зростаючим запитом суспільства на екологічну справедливість та відкритість інформації про стан довкілля.

Реалізація такої системи створює дієвий механізм громадського та державного контролю, де кожен виявлений випадок незаконного викиду стає зафіксованим цифровим фактом.

Це стимулює промислові підприємства інвестувати у власні локальні очисні споруди, замість того щоб використовувати міську мережу як безкоштовний резервуар для відходів. Таким чином, розробка виконує важливу превентивну функцію, змінюючи модель поведінки суб'єктів господарювання на більш екологічно відповідальну.

Інтеграція технологій штучного інтелекту та Інтернету речей у сферу водовідведення є логічною відповіддю на виклики четвертої промислової революції. Створення цифрових двійників каналізаційних мереж дозволяє моделювати різні сценарії поширення забруднень та оперативно розробляти плани реагування на надзвичайні ситуації.

Актуальність роботи підтверджується також глобальними змінами клімату, які вимагають більш гнучкого та точного управління водними ресурсами [24-27].

Загалом, розробка кіберфізичної системи «Розумна каналізаційна система» є стратегічно важливим кроком, що об'єднує технологічний прогрес із фундаментальними потребами збереження природного середовища для майбутніх поколінь.

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						7
Зм.	Арк.	№ док.ум.	Підпис	Дата		

Архітектура кіберфізичної системи «Розумна каналізаційна система» будується на принципі багаторівневої інтеграції, де фізичні об'єкти мережі водовідведення нерозривно пов'язані з обчислювальними алгоритмами. Перший, найнижчий рівень – це рівень фізичного сприйняття, що складається з мережі інтелектуальних сенсорних вузлів, розташованих у стратегічних точках колекторів. Кожен такий вузол оснащений набором датчиків для вимірювання електропровідності, рН, температури, а також специфічних іоноселективних електродів для виявлення важких металів. Важливою особливістю архітектури є те, що первинна обробка сигналу відбувається безпосередньо в місці заміру, що дозволяє відфільтровувати шуми та передавати лише значущі зміни показників.

Другий рівень архітектури відповідає за передачу даних і є одним із найскладніших через специфіку підземних комунікацій [28-30].

Для забезпечення стабільного зв'язку використовується комбінована топологія, де датчики об'єднуються в локальні кластери, передаючи інформацію на шлюзи за допомогою енергоефективних протоколів типу LoRaWAN або NB-IoT. Ці шлюзи розташовуються в наземних спорудах або оглядових люках, де рівень сигналу дозволяє зв'язатися з базовими станціями мобільного оператора або міською мережею Wi-Fi. Таким чином створюється надійна магістраль для безперервного потоку телеметрії від підземних сенсорів до центрального сховища.

Третій рівень – це рівень обробки та аналітики, що базується на хмарних технологіях та великих даних.

Тут архітектура передбачає використання серверів, де накопичується історична інформація про склад стічних вод, що дозволяє системі навчатися та розпізнавати «нормальний» стан мережі.

На цьому етапі залучаються алгоритми машинного навчання, які здатні зіставляти дані від різних датчиків у часі та просторі. Якщо декілька сенсорів послідовно фіксують проходження плями забруднення, система автоматично

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

розраховує траєкторію та швидкість її руху, будуючи віртуальну модель інциденту.

Наступний рівень архітектури – це рівень прийняття рішень та управління, який виступає інтерфейсом взаємодії з оператором. Він включає в себе цифрову панель управління (dashboard) з інтерактивною картою міста, де в реальному часі відображається статус кожного вузла та рівень екологічної небезпеки. Архітектура передбачає дворівневу систему оповіщення: автоматичне формування звіту про пікові навантаження та миттєву тривогу у разі виявлення критичних концентрацій токсинів. Це дозволяє диспетчерським службам миттєво координувати дії з екологічними інспекціями та аварійними бригадами водоканалу [31-34].

Важливим компонентом є блок інтеграції із зовнішніми інформаційними системами, такими як бази даних підприємств, метеорологічні сервіси та гідравлічні моделі міста. Це дозволяє архітектурі бути адаптивною: наприклад, при прогнозуванні зливи система автоматично змінює чутливість датчиків, щоб уникнути помилкових спрацювань через розведення стоків дощовою водою. Також архітектурно закладається можливість зворотного зв'язку, де КФС може керувати фізичними виконавчими механізмами, наприклад, запірною арматурою для локалізації забрудненого сегмента.

Завершальним елементом архітектурної побудови є контур кібербезпеки, що пронизує всі рівні від датчика до хмари.

Кожен пакет даних шифрується на етапі створення, а доступ до управління системою суворо регламентується за допомогою блокчейн-технологій або багаторівневої ідентифікації. Це гарантує цілісність даних, які в майбутньому можуть бути використані як доказова база в суді проти підприємств-порушників. Така комплексна архітектура перетворює розрізнені сенсори на єдиний живий організм, здатний до самостійного моніторингу та захисту екологічного простору міста.

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						9
Зм.	Арк.	№ докum.	Підпис	Дата		

1.3 Висновок

Було здійснено теоретичний аналіз, вивчено недоліки існуючих рішень та сформовано задачу на дослідження шляхів їх вирішення.

На першому етапі роботи відбувся аналіз задачі розумної каналізаційної системи. На другому етапі дослідження проведено огляд кіберфізичної системи «Роумна каналізаційна система».

На основі проведених досліджень визначено основну задачу подальшого дослідження – спроектувати та розробити кіберфізичну систему «Розумна каналізаційна система», яка передбачатиме автоматизовану ідентифікацію джерел незаконних викидів та прогнозування поширення забруднюючих речовин.

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

2 ПРОЄКТУВАННЯ ТА РОЗРОБЛЕННЯ КІБЕРФІЗИЧНОЇ СИСТЕМИ

2.1 Загальний опис складу кіберфізичної системи

Кіберфізична система «Розумна каналізаційна система» являє собою багаторівневий програмно-апаратний комплекс, призначений для безперервного моніторингу хімічного складу стічних вод у каналізаційній мережі міста. Головною метою системи є оперативне виявлення несанкціонованих промислових скидів шляхом автоматичного аналізу фізико-хімічних параметрів стічних вод у режимі реального часу. Для досягнення цієї мети система об'єднує сенсорне обладнання, засоби передачі даних, серверну інфраструктуру обробки та візуалізації інформації в єдиний інтегрований комплекс.

Архітектура розроблюваної системи складається з чотирьох основних рівнів, кожен з яких виконує специфічні функції у загальному процесі моніторингу та реагування.

Перший сенсорний рівень представлений розподіленою мережею автономних сенсорних вузлів, що встановлюються безпосередньо у каналізаційних колодязях, на ділянках колекторної мережі поблизу точок підключення промислових підприємств. Кожен сенсорний вузол обладнаний набором датчиків хімічного складу води (рН, електропровідність, каламутність, розчинений кисень, окислювально-відновний потенціал, температура), мікроконтролером для збору та попередньої обробки даних, а також модулем бездротового зв'язку для передачі інформації на наступний рівень. Вузли працюють в автономному режимі від акумуляторного живлення та здатні функціонувати у складних умовах підземних каналізаційних комунікацій.

Другий мережевий рівень забезпечує надійну передачу даних від сенсорних вузлів до серверної інфраструктури. Він включає шлюзи, які приймають дані від групи сенсорних вузлів по бездротових каналах зв'язку малої потужності (LoRa, Wi-Fi) та ретранслюють їх на сервер через мережу Інтернет.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 11
Зм.	Арк.	№ докум.	Підпис	Дата		

Шлюзи розміщуються на наземних об'єктах міської інфраструктури і мають стаціонарне електроживлення.

Третій серверний рівень передбачає основну логіку системи: брокер повідомлень MQTT для прийому та маршрутизації потоків даних від усіх вузлів, база даних часових рядів для довготривалого зберігання вимірювань, програмні модулі аналізу та виявлення аномалій, а також сервіс оповіщень, який формує та надсилає тривожні повідомлення відповідальним особам. Серверна частина розгортається у контейнеризованому середовищі на виділеному сервері або у хмарній інфраструктурі, що забезпечує гнучкість масштабування та спрощує адміністрування.

Четвертий рівень візуалізації та взаємодії з користувачем надає операторам та екологічним інспекторам зручний інтерфейс для спостереження за станом каналізаційної мережі. Веб-панель моніторингу на базі Grafana відображає поточні та історичні показання датчиків у вигляді графіків, карту розташування вузлів із кольоровою індикацією стану, журнал тривог та інцидентів. Додатково система забезпечує оперативне інформування через електронну пошту або SMS.

Потік даних у системі є переважно висхідним: дані рухаються від датчиків через мікроконтролер і шлюз до сервера та далі до інтерфейсу користувача. Водночас передбачено низхідний потік команд для дистанційного налаштування параметрів роботи.

2.2 Апаратні компоненти сенсорного вузла

Мікроконтролер є центральним елементом кожного сенсорного вузла. Він виконує функції збору аналогових та цифрових сигналів від датчиків, їх попередньої обробки (фільтрація, калібрувальні перетворення), формування пакетів даних та керування бездротовою передачею. Від вибору мікроконтролерної платформи залежать ключові характеристики вузла: кількість

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 12
Зм.	Арк.	№ докум.	Підпис	Дата		

підключених датчиків, точність вимірювань, енергоспоживання, можливості зв'язку та вартість.

Для обґрунтованого вибору було проаналізовано чотири найбільш поширені платформи, що використовуються в IoT-проектах: ESP32 (Espressif Systems), STM32F103C8T6 (STMicroelectronics), Arduino Mega 2560 (Microchip/Atmel) та Raspberry Pi Pico (Raspberry Pi Foundation). Порівняння проведено за критеріями, що є визначальними для задачі побудови автономного сенсорного вузла [35-38].

Таблиця 2.1 — Порівняльна характеристика мікроконтролерних платформ

Параметр	ESP32	STM32F103	Arduino Mega	RPi Pico
Процесор	Xtensa LX6, 2 ядра, 240 МГц	ARM Cortex-M3, 72 МГц	ATmega2560, 16 МГц	ARM Cortex-M0+, 2 ядра, 133 МГц
RAM / Flash	520 КБ / 4 МБ	20 КБ / 64 КБ	8 КБ / 256 КБ	264 КБ / 2 МБ
Канали АЦП	18 (12-біт)	10 (12-біт)	16 (10-біт)	3 (12-біт)
Wi-Fi / BLE	Так / Так	Ні / Ні	Ні / Ні	Ні / Ні
Інтерфейси	I ² C, SPI, UART, 1-Wire	I ² C, SPI, UART	I ² C, SPI, UART	I ² C, SPI, UART
Deep sleep	~10 мкА	~2 мкА (standby)	Не підтримує	~1,3 мА (dormant)
Орієнтовна ціна	3–5 USD	2–4 USD	8–12 USD	4–6 USD

За результатами порівняльного аналізу для реалізації сенсорного вузла обрано модуль ESP32 (ESP-WROOM-32) [39]. Основними перевагами цієї платформи, що визначили вибір, є такі. По-перше, наявність вбудованих модулів

Wi-Fi 802.11 b/g/n та Bluetooth 4.2/BLE, що дозволяє реалізувати бездротову передачу даних без додаткових зовнішніх модулів для вузлів, розташованих у зоні покриття Wi-Fi. По-друге, двоядерний процесор з тактовою частотою до 240 МГц забезпечує достатню обчислювальну потужність для цифрової фільтрації сигналів, калібрувальних перетворень та формування пакетів даних. По-третє, 18 каналів 12-бітного АЦП дозволяють підключити всі необхідні аналогові датчики без зовнішнього мультиплексора. По-четверте, наднизьке споживання в режимі глибокого сну (близько 10 мкА) є критично важливим для автономної роботи вузла від акумулятора. По-п'яте, широка підтримка спільноту, розвинена екосистема бібліотек Arduino та сумісність із середовищем PlatformIO значно прискорюють розробку (рис. 2.1) [40].

Модуль STM32F103, хоча і має ще нижче енергоспоживання у сплячому режимі, позбавлений вбудованого бездротового зв'язку та має значно менший обсяг оперативної пам'яті. Arduino Mega характеризується надмірною кількістю GPIO при занадто низькій обчислювальній потужності та відсутності режиму глибокого сну. Raspberry Pi Pico має лише три канали АЦП, що є недостатнім, та також не має вбудованого Wi-Fi.

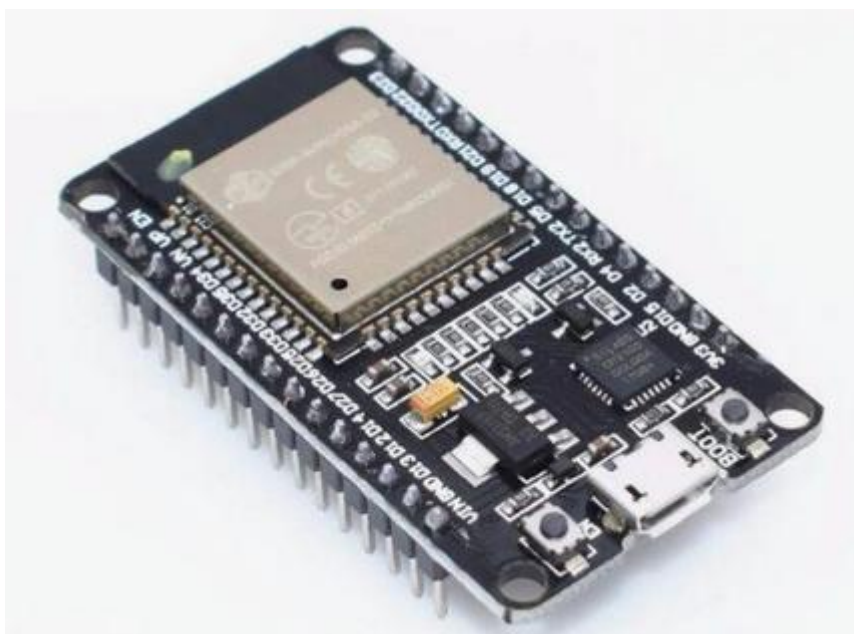


Рисунок 2.1 - Модуль ESP-WROOM-32 [41]

Вибір датчиків є ключовим етапом проєктування, оскільки саме від їхніх характеристик залежить здатність системи виявляти різні типи промислових забруднень. На підставі аналізу нормативних документів та типових хімічних сигнатур промислових скидів визначено перелік параметрів, що підлягають контролю: водневий показник (pH), загальна мінералізація / електропровідність (TDS/EC), каламутність (NTU), вміст розчиненого кисню (DO), окислювально-відновний потенціал (ORP) та температура [42-44].

Датчик водневого показника (pH). Водневий показник є одним з найважливіших індикаторів якості стічних вод та одним із перших параметрів, що реагує на промисловий скид. Різке зниження pH (нижче 6,5) свідчить про потрапляння кислотних стоків (гальванічні виробництва, хімічні підприємства), а підвищення (вище 9,0) про лужні скиди (целюлозно-паперова промисловість, виробництво миючих засобів). Для вимірювання pH обрано аналоговий датчик DFRobot Gravity Analog pH Sensor V2 (SEN0161-V2). Принцип його роботи базується на потенціометричному методі: скляний електрод занурюється у розчин, і різниця електричних потенціалів між внутрішнім буферним розчином та досліджуванним середовищем вимірюється відносно референтного електрода. Ця різниця потенціалів пропорційна значенню pH середовища і описується рівнянням Нернста.

Модуль датчика включає pH-електрод із BNC-роз'ємом та плату підсилювача-перетворювача сигналу, яка перетворює мілівольтний сигнал електрода в напругу 0–5 В, зручну для зчитування аналого-цифровим перетворювачем мікроконтролера. Діапазон вимірювань складає 0–14 pH із точністю $\pm 0,1$ pH, час відгуку менше 1 хвилини. Датчик потребує періодичного калібрування за двома точками (буферні розчини pH 4,0 та pH 7,0) для забезпечення точності вимірювань. Калібрувальні коефіцієнти зберігаються в енергонезалежній пам'яті мікроконтролера.

Датчик електропровідності загальної мінералізації (TDS/EC). Електропровідність стічних вод визначається концентрацією розчинених іонів і

є важливим індикатором загального рівня мінералізації. Різке зростання TDS (Total Dissolved Solids) може свідчити про скид концентрованих розчинів солей, кислот або лугів. Для вимірювання цього параметра обрано модуль DFRobot Gravity Analog TDS Sensor (SEN0244). Принцип дії кондуктометричний: два електроди занурюються у розчин, між ними пропускається змінний струм, а за вимірним опором розчину розраховується його електропровідність та еквівалентне значення TDS.

Діапазон вимірювань становить 0–1000 ppm (мг/л) із точністю $\pm 10\%$. Аналоговий вихід модуля (0–2,3 В) підключається до каналу АЦП мікроконтролера. Важливою особливістю є значна залежність електропровідності від температури розчину: зростання температури на 1 °С призводить до збільшення провідності приблизно на 2%. Тому для отримання коректних результатів у прошивці мікроконтролера реалізовано алгоритм автоматичної температурної компенсації, що використовує показання датчика температури DS18B20.

Розглянемо датчик каламутності. Каламутність характеризує вміст у воді завислих частинок (глина, мул, органічні речовини, промислові суспензії) і вимірюється в нефелометричних одиницях каламутності. Різке зростання каламутності є ознакою надходження механічних домішок, що часто супроводжує промислові скиди. Обрано датчик DFRobot Gravity Analog Turbidity Sensor (SEN0189), який працює за нефелометричним принципом: інфрачервоний світлодіод випромінює промінь у досліджувану рідину, а фотоприймач, розташований під кутом 90°, реєструє інтенсивність розсіяного завислими частинками світла. Чим більше частинок, тим більше розсіяного світла та вища каламутність.

Вихідна напруга датчика знаходиться в діапазоні 0–4,5 В та має обернену залежність від каламутності: чиста вода дає напругу близько 4,2 В, а каламутна менше 2,5 В. Перетворення напруги в значення NTU здійснюється за поліноміальною апроксимацією, коефіцієнти якої визначаються при

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 16
Зм.	Арк.	№ докум.	Підпис	Дата		

калібруванні. Діапазон вимірювань від 0 до 3000 NTU, що є достатнім для фіксації як незначних, так і критичних змін каламутності стічних вод.

Розглянемо датчик розчиненого кисню (DO). Концентрація розчиненого кисню у стічних водах є важливим показником, що відображає інтенсивність біохімічних процесів. У нормальних умовах стічні води містять 2–6 мг/л розчиненого кисню. Різде зниження цього показника свідчить про надходження великої кількості органічних речовин, які інтенсивно окислюються мікроорганізмами, споживаючи кисень. Це характерно для скидів підприємств харчової промисловості, молокозаводів, м'ясопереробних підприємств.

Для вимірювання обрано датчик DFRobot Gravity Analog Dissolved Oxygen Sensor (SEN0237-A), що працює за гальванічним принципом: кисень, проникаючи через мембрану датчика, вступає в електрохімічну реакцію на катоді, генеруючи електричний струм, пропорційний парціальному тиску кисню у розчині (рис. 2.2). Діапазон вимірювань складає 0–20 мг/л із точністю $\pm 0,3$ мг/л. Для підвищення точності зчитування аналогового сигналу цей датчик підключається через зовнішній 16-бітний АЦП ADS1115, що забезпечує значно кращу роздільну здатність порівняно зі вбудованим 12-бітним АЦП мікроконтролера ESP32.



Рисунок 2.2 - Модуль SEN0237-A [45]

Розглянемо датчик окислювально-відновного потенціалу (ORP). Окислювально-відновний потенціал характеризує здатність середовища окислювати або відновлювати хімічні речовини та вимірюється в мілівольтах. Для стічних вод нормальний діапазон ORP складає +150...+400 мВ. Різке зниження ORP свідчить про надходження відновників (сульфідів, органічних речовин, деяких важких металів), а різке підвищення — про надходження сильних окисників (хлору, хроматів, перманганатів). Вимірювання ORP разом з рН дозволяє значно точніше ідентифікувати тип забруднювача.

Обрано модуль DFRobot Gravity Analog ORP Sensor (SEN0165), що складається з платинового ORP-електрода та плати підсилювача. Діапазон вимірювань становить -2000...+2000 мВ. Вихідний сигнал — аналогова напруга, що подається на АЦП мікроконтролера. Калібрування здійснюється за стандартним розчином з відомим значенням ORP.



Рисунок 2.3 - Модуль SEN0165 [46]

Розглянемо датчик температури. Температура стічних вод є допоміжним, проте надзвичайно важливим параметром. По-перше, вона необхідна для температурної компенсації показань інших датчиків (рН, TDS, DO), оскільки всі електрохімічні датчики мають значну температурну залежність. По-друге, аномально висока температура стоків (вище 40 °С) сама по собі є ознакою промислового скиду, наприклад, від теплоенергетичних підприємств або технологічних процесів з нагріванням [47-49].

Обрано цифровий датчик температури DS18B20 у водонепроникному виконанні (зонд у металевій гільзі з кабелем довжиною 1 м). Датчик працює за інтерфейсом 1-Wire, що дозволяє підключити його до мікроконтролера за допомогою лише одного цифрового виводу GPIO. Діапазон вимірювань складає $-55...+125$ °C із точністю $\pm 0,5$ °C у діапазоні $-10...+85$ °C, роздільна здатність до $0,0625$ °C (у 12-бітному режимі). Протокол 1-Wire дозволяє підключати кілька датчиків на одну шину (кожен має унікальний 64-бітний ідентифікатор), що може бути корисним для контролю температури в різних точках вузла.

Таблиця 2.2 – Зведена характеристика обраних датчиків (рис. 2.4-2.8)

Параметр	Модель	Діапазон	Точність	Вихід	Живлення	Ціна, USD
pH	DFRobot SEN0161-V2	0–14 pH	$\pm 0,1$ pH	Аналог.	5 В	~30
TDS/EC	DFRobot SEN0244	0–1000 ppm	$\pm 10\%$	Аналог.	3,3–5 В	~12
Каламутність	DFRobot SEN0189	0–3000 NTU	$\pm 5\%$	Аналог.	5 В	~10
DO	DFRobot SEN0237-A	0–20 мг/л	$\pm 0,3$ мг/л	Аналог.	3,3–5 В	~35
ORP	DFRobot SEN0165	± 2000 мВ	± 5 мВ	Аналог.	5 В	~40
Темп.	DS18B20 (водон.)	$-55...+125$ °C	$\pm 0,5$ °C	1-Wire	3–5,5 В	~3

Загальна вартість комплекту датчиків для одного сенсорного вузла складає приблизно 130 USD. Усі обрані датчики виробництва DFRobot мають

уніфікований формат підключення, що спрощує фізичне з'єднання з мікроконтролером та зменшує ймовірність помилок при монтажі.

Слід зазначити, що електрохімічні датчики (pH, ORP, DO) мають обмежений ресурс електродів і схильні до дрейфу показань з часом, особливо в агресивному середовищі стічних вод. Тому для підтримання точності вимірювань передбачено регламент регулярного калібрування (рекомендовано раз на 2–4 тижні) та заміни витратних елементів (електродів, мембран) відповідно до специфікацій виробника.

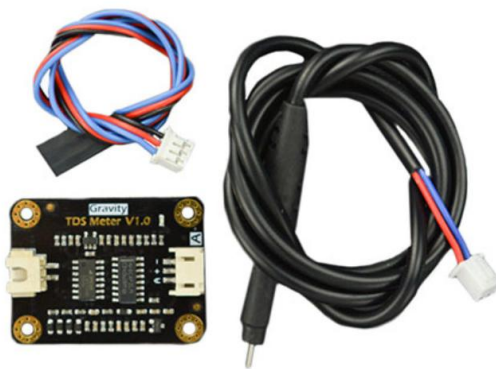


Рисунок 2.4 – Модуль DFRobot SEN0244 [50]

Розглянемо зовнішній аналого-цифровий перетворювач ADS1115. Вбудований АЦП мікроконтролера ESP32, хоча й має 12-бітну роздільну здатність, характеризується значною нелінійністю на краях робочого діапазону при напругах, близьких до 0 В та 3,3 В, що знижує точність вимірювань для чутливих аналогових датчиків. Для вирішення цієї проблеми до складу вузла включено зовнішній АЦП ADS1115 16-бітний чотириканальний перетворювач з інтерфейсом I²C. Його роздільна здатність 65536 рівнів проти 4096 у вбудованого АЦП забезпечує значно точніше перетворення аналогових сигналів. Модуль підключається до шини I²C мікроконтролера та використовується для зчитування найбільш критичних датчиків. Програмований коефіцієнт

підсилення дозволяє адаптувати діапазон вхідної напруги під конкретний датчик.

Розглянемо модуль годинника реального часу DS3231. Точна прив'язка кожного вимірювання до часу є критичною для роботи алгоритму локалізації джерела забруднення, який базується на аналізі часових затримок між спрацюванням послідовних вузлів. Вбудований таймер мікроконтролера ESP32 не зберігає абсолютний час при переході у глибокий сон та має значний дрейф.

Розглянемо модуль MicroSD-карти. У реальних умовах експлуатації в підземних каналізаційних колодязях можливі ситуації тимчасової втрати бездротового зв'язку перешкоди, несправність шлюзу, відключення електроживлення шлюзу. Для запобігання втраті даних у такі періоди кожен сенсорний вузол оснащено модулем для читання та запису MicroSD-карт. мність навіть мінімальної MicroSD-карти (2 ГБ) є достатньою для зберігання мільйонів вимірювань.

Розглянемо силові MOSFET-ключі. Для зниження загального енергоспоживання вузла в режимі глибокого сну передбачено повне відключення живлення датчиків за допомогою MOSFET-транзистора, наприклад, IRLZ44N є N-канальний з логічним рівнем керування. Затвор транзистора підключається до GPIO-виводу мікроконтролера через резистор 10 кОм. Коли мікроконтролер переходить у режим глибокого сну, він встановлює низький рівень на виході GPIO, що закриває транзистор та повністю знімає живлення з датчиків. При пробудженні живлення подається знову, після чого передбачено паузу 30–60 секунд для стабілізації показань електрохімічних датчиків.

Розглянемо пасивні компоненти та елементи захисту. Для надійної роботи схеми необхідні підтягуючі резистори: 4,7 кОм на лінії 1-Wire (для DS18B20) та 10 кОм на лініях SDA/SCL шини I²C. Фільтрація живлення забезпечується керамічними конденсаторами 100 нФ поблизу виводу живлення кожного модуля

та електролітичним конденсатором 100–470 мкФ на спільній шині живлення. Для захисту від зворотної полярності живлення використовується діод Шоттки.

Забезпечення надійного бездротового зв'язку між сенсорними вузлами, розташованими в підземних каналізаційних колодязях, та наземним шлюзом є одним із найскладніших інженерних завдань проєкту. Підземне розташування створює значне затухання радіосигналу: залізобетонні конструкції колодязів, ґрунт та металеві елементи каналізаційної інфраструктури суттєво послаблюють електромагнітні хвилі.

Для вирішення цього завдання передбачено використання двох технологій залежно від умов розташування конкретного вузла. Для вузлів, що знаходяться у зоні покриття наявної Wi-Fi мережі (наприклад, поблизу будівель із зовнішніми точками доступу або муніципальним Wi-Fi), використовується вбудований Wi-Fi модуль мікроконтролера ESP32. Це найпростіше та найдешевше рішення, що не потребує додаткових апаратних компонентів.

Для віддалених вузлів, де Wi-Fi недоступний, передбачено використання модуля LoRa на базі трансивера Semtech SX1278. Модуль SX1278 працює в неліцензійному діапазоні частот 433 або 868 МГц, підключається до мікроконтролера по інтерфейсу SPI і потребує антени (дротова чвертьхвильова або зовнішня на SMA-роз'ємі). Швидкість передачі даних є невисокою (0,3–50 кбіт/с), проте цілком достатньою для передачі компактних JSON-пакетів з результатами вимірювань, що мають об'єм порядку 200–300 байт.

Автономність роботи сенсорного вузла є критичним параметром, оскільки каналізаційні колодязі, як правило, не мають підключення до електромережі. Для живлення обрано літій-іонний акумулятор формату 18650 із номінальною напругою 3,7 В та ємністю 3400 мАг (наприклад, Samsung INR18650-35E або аналог). Акумулятор підключається через модуль зарядки TP4056 із вбудованим захистом від перезарядки, перерозрядки та короткого замикання. Для можливості підзарядки у місцях з доступом до денного світла (деякі оглядові

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 22
Зм.	Арк.	№ докум.	Підпис	Дата		

колодязі мають ґратчасті кришки) передбачено підключення мініатюрної сонячної панелі (5 В, 1 Вт) до входу модуля TP4056.

Стабілізація напруги здійснюється понижуючим DC-DC перетворювачем на базі мікросхеми MP1584 (вхід 4,5–28 В, вихід 3,3 В, ККД до 92%) для живлення мікроконтролера та логічних схем, а також лінійним стабілізатором AMS1117-5.0 для формування напруги 5 В, необхідної деяким датчикам. Використання імпульсного DC-DC перетворювача замість лінійного стабілізатора для основної лінії 3,3 В дозволяє суттєво знизити втрати енергії на регулювання напруги.

Для оцінки часу автономної роботи складено енергетичний баланс вузла. В активному режимі (вимірювання та передача) вузол споживає приблизно 180–250 мА протягом 30–60 секунд. У режимі глибокого сну (з вимкненими датчиками через MOSFET-ключ) споживання знижується до 10–50 мкА. За циклу роботи «вимірювання 45 с кожні 5 хвилин» середній струм споживання складає приблизно 3–5 мА. Це забезпечує час автономної роботи від одного акумулятора 3400 мАг приблизно 28–47 діб, що є прийнятним для регулярного обслуговування.

Сенсорний вузол експлуатується в надзвичайно складних умовах каналізаційного колодязя: висока вологість (до 100%), агресивні гази (сірководень H_2S , аміак NH_3 , метан CH_4), можливість тимчасового затоплення, механічні впливи. Тому до конструктиву висуваються жорсткі вимоги.

Корпус вузла виготовляється з ABS-пластику зі ступенем захисту IP67 (повний захист від пилу, захист від короткочасного занурення у воду на глибину до 1 м). Для виведення зондів датчиків та антени використовуються кабельні гермовводи типу PG7 або PG9, які забезпечують герметичність при проходженні кабелів через стінку корпусу. Всередині корпусу електронні компоненти додатково захищаються конформним покриттям (лаком) друкованої плати від конденсату. Кріплення корпусу у колодязі здійснюється за допомогою нержавіючих кронштейнів та анкерних болтів. Зонди датчиків виводяться через

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 23
Зм.	Арк.	№ докум.	Підпис	Дата		

нижню частину корпусу та занурюються у потік стічних вод на глибину 10–20 см від поверхні.

2.3 Апаратні компоненти мережевого рівня

Шлюз виконує функцію посередника між розподіленою мережею сенсорних вузлів та серверною інфраструктурою. Його основне завдання приймати дані від сенсорних вузлів по каналах зв'язку малої дальності та ретранслювати їх на сервер через високошвидкісне підключення до мережі Інтернет. Один шлюз здатний обслуговувати десятки сенсорних вузлів у радіусі до кількох кілометрів.

В якості апаратної бази шлюзу обрано одноплатний комп'ютер Raspberry Pi 4 Model B потужну та універсальну платформу, що широко використовується для побудови IoT-шлюзів. Основні характеристики Raspberry Pi 4: процесор Broadcom BCM2711 (чотириядерний ARM Cortex-A72, 1,5 ГГц), оперативна пам'ять 2 або 4 ГБ LPDDR4, гігабітний Ethernet-порт, двохдіапазонний Wi-Fi 802.11 ac (2,4 / 5 ГГц), Bluetooth 5.0, два порти USB 3.0, слот для MicroSD-карти, інтерфейс GPIO 40-pin для підключення периферійних модулів. Операційна система Raspberry Pi OS (на базі Debian Linux).

Для прийому даних від сенсорних вузлів по LoRa до Raspberry Pi підключається модуль-концентратор LoRa. Цей модуль встановлюється безпосередньо на GPIO-роз'єм Raspberry Pi та містить мікросхему Semtech SX1301, яка здатна одночасно приймати пакети на 8 каналах і з різними параметрами модуляції.

На програмному рівні шлюз виконує такі функції: приймання LoRa-пакетів через драйвер packet forwarder, декодування отриманих даних, публікація повідомлень у MQTT-брокер на сервері через Інтернет (Ethernet або Wi-Fi), локальне буферизування даних у разі тимчасової втрати Інтернет-з'єднання, а

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 24
Зм.	Арк.	№ докум.	Підпис	Дата		

також моніторинг власного стану (температура процесора, навантаження, доступність Інтернету) та відправка діагностичних повідомлень.

Живлення шлюзу здійснюється від мережі 220 В через блок живлення 5В / 3А (USB Type-C). Для забезпечення безперебійної роботи рекомендовано використання міні-UPS для Raspberry Pi, що забезпечує автономну роботу протягом 2–4 годин при відключенні електроенергії.

Шлюз розміщується на наземному об'єкті міської інфраструктури: в технічній шафі на опорі освітлення, на даху або фасаді будівлі, у приміщенні насосної станції. Антена LoRa (зовнішня, на базі фіброскляного щтиря з коефіцієнтом підсилення 5–8 дБі) встановлюється на максимально можливій висоті для забезпечення найкращого радіопокриття зони розташування сенсорних вузлів. Корпус шлюзу має ступінь захисту IP65 для зовнішнього встановлення.

Кількість шлюзів у системі визначається площею охоплення та щільністю розміщення сенсорних вузлів. Для типового міста середнього розміру один шлюз здатний обслуговувати зону радіусом 2–3 км у міській забудові, що відповідає десяткам каналізаційних колодязів. У разі масштабування системи на більшу територію додаються нові шлюзи; серверна частина не потребує при цьому суттєвих змін, оскільки кожен шлюз є незалежним і публікує дані в той самий MQTT-брокер.

2.4 Серверна інфраструктура

Серверна інфраструктура є центральним обчислювальним ядром кіберфізичної системи «Розумна каналізаційна система». Саме на серверному рівні відбуваються прийом, збереження, аналіз усіх даних, що надходять від розподіленої мережі сенсорних вузлів, а також генерація тривожних сповіщень та надання інтерфейсу моніторингу для операторів. Від правильного вибору

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 25
Зм.	Арк.	№ докум.	Підпис	Дата		

серверної платформи та архітектури розгортання залежить продуктивність, надійність та масштабованість системи в цілому.

Для розгортання серверної частини системи було розглянуто три основні варіанти: локальний фізичний сервер, хмарна IoT-платформа та віртуальний приватний сервер. Кожен варіант має свої переваги та обмеження, які було проаналізовано з урахуванням специфіки розроблюваної системи.

Локальний сервер передбачає розміщення обладнання на об'єкті замовника, наприклад, у серверній кімнаті водоканалу або екологічної інспекції. Перевагами є повний контроль над даними (що важливо для конфіденційної інформації про промислові підприємства), відсутність щомісячних платежів за хмарні сервіси та незалежність від якості Інтернет-з'єднання на ділянці між сервером і шлюзами (якщо вони в одній локальній мережі). Недоліками є необхідність придбання та обслуговування фізичного обладнання, організації резервного живлення та охолодження, а також складність масштабування.

Хмарні IoT-платформи пропонують готову інфраструктуру для прийому, зберігання та аналізу IoT-даних із вбудованими засобами масштабування, резервного копіювання та безпеки. Однак використання таких платформ пов'язане зі значними фінансовими витратами (щомісячна абонентська плата, яка зростає пропорційно обсягу даних та кількості пристроїв), залежністю від зовнішнього провайдера та необхідністю стабільного широкосмугового Інтернет-з'єднання. Крім того, розміщення даних про стан міської каналізаційної інфраструктури на серверах іноземних компаній може бути неприйнятним з міркувань інформаційної безпеки.

Віртуальний приватний сервер (VPS) є компромісним рішенням, що поєднує переваги обох підходів. VPS-хостинг надає віртуальну машину з виділеними ресурсами, повним адміністративним доступом, статичною IP-адресою та гарантованою доступністю. При цьому вартість є значно нижчою порівняно з хмарними IoT-платформами, а адміністрування є простішим порівняно з локальним сервером.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 26
Зм.	Арк.	№ докum.	Підпис	Дата		

Для етапу прототипування та початкового впровадження обрано варіант VPS із можливістю подальшої міграції на локальний сервер замовника. Мінімальні вимоги до серверних ресурсів для обслуговування прототипної мережі з 10–20 сенсорних вузлів визначено таким чином: процесор 2 віртуальні ядра (vCPU), оперативна пам'ять 4 ГБ, дисковий простір 50 ГБ SSD (з можливістю розширення при накопиченні історичних даних), мережевий канал 100 Мбіт/с. Операційна система Ubuntu Server 22.04 LTS, обрана завдяки довготривалій підтримці (до 2027 року), широкій сумісності з усіма необхідними програмними компонентами та великій базі документації.

Ключовим архітектурним рішенням серверної частини є контейнеризація усіх сервісів за допомогою Docker та оркестрація за допомогою Docker Compose. Кожен програмний компонент системи розгортається як окремий ізольований контейнер зі своїми залежностями. Мережева конфігурація Docker передбачає створення внутрішньої віртуальної мережі, в якій усі контейнери спілкуються між собою за іменами сервісів. Назовні відкриваються лише необхідні порти: порт 1883 для MQTT-брокера, порт 3000 для веб-інтерфейсу Grafana та порт 8000 для REST API.

2.5 Програмні засоби та середовища розробки

Розробка вбудованого програмного забезпечення для мікроконтролера ESP32 потребує спеціалізованого середовища, що підтримує крос-компіляцію, завантаження прошивки у мікроконтролер та засоби налагодження.

Для реалізації проєкту обрано середовище розробки Arduino IDE 2.x вільно поширюване інтегроване середовище розробки, що є стандартом де-факто для початкового та середнього рівня розробки вбудованих систем. Arduino IDE підтримує мову програмування C/C++ з використанням фреймворку Arduino, який надає високорівневий програмний інтерфейс (API) для роботи з периферією мікроконтролера: цифровими та аналоговими входами-виходами, послідовними інтерфейсами, таймерами та іншими апаратними ресурсами.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 27
Зм.	Арк.	№ докum.	Підпис	Дата		

Для забезпечення підтримки мікроконтролера ESP32 в Arduino IDE необхідно встановити пакет підтримки плат Espressif Arduino Core через менеджер плат. Цей пакет включає компілятор для архітектури Xtensa, бібліотеки для роботи з периферією ESP32 та утиліти для завантаження прошивки через USB або по бездротовому каналу.

Основними перевагами Arduino IDE, що визначили вибір, є: простий та інтуїтивно зрозумілий інтерфейс, що не потребує тривалого вивчення; вбудований менеджер бібліотек, який дозволяє у кілька клацань встановити будь-яку з тисяч доступних бібліотек для роботи з датчиками, протоколами зв'язку та іншими компонентами; вбудований серійний монітор для відлагодження через порт UART; величезна спільнота розробників та обширна база прикладів коду для ESP32.

Бібліотека WiFi.h входить до складу ядра Espressif Arduino Core і не потребує окремого встановлення. Вона надає функції для сканування доступних мереж, підключення до точки доступу із зазначенням SSID та пароля, моніторингу стану з'єднання та автоматичного перепідключення у разі втрати зв'язку. У прошивці сенсорного вузла використовується режим Station (STA), в якому мікроконтролер виступає клієнтом існуючої Wi-Fi мережі.

Бібліотеки OneWire.h та DallasTemperature.h працюють у зв'язці для обслуговування датчика температури DS18B20. OneWire.h реалізує низькорівневий протокол обміну по однопровідній шині, а DallasTemperature.h надає зручний інтерфейс для запиту температури в градусах Цельсія, управління роздільною здатністю (9–12 біт) та адресації конкретних датчиків на шині за їхніми унікальними ROM-кодами.

Бібліотека ArduinoJson.h є реалізацією JSON-серіалізатора/десеріалізатора, оптимізованою для вбудованих систем з обмеженими ресурсами пам'яті.

Таблиця 2.3 — Програмні бібліотеки для прошивки сенсорного вузла

Бібліотека	Компонент / протокол	Призначення
WiFi.h	Wi-Fi модуль ESP32	Підключення мікроконтролера до бездротової мережі Wi-Fi, управління з'єднанням, отримання IP-адреси через DHCP
PubSubClient.h	Протокол MQTT	Реалізація MQTT-клієнта: підключення до брокера, публікація повідомлень у топіки, підписка на топіки для прийому команд
OneWire.h	DS18B20	Реалізація протоколу 1-Wire для обміну даними з цифровими датчиками на однопровідній шині
DallasTemperature.h	DS18B20	Високорівнева бібліотека для зчитування температури з датчиків серії DS18B20; працює поверх OneWire.h
Wire.h	I ² C шина	Реалізація протоколу I ² C (Inter-Integrated Circuit) для зв'язку з ADS1115, DS3231 та іншими I ² C-пристроями
Adafruit_ADS1X15.h	АЦП ADS1115	Драйвер для зовнішнього 16-бітного АЦП ADS1115: налаштування каналів, коефіцієнта підсилення (PGA), зчитування значень

Кінець таблиці 2.3

RTClib.h	RTC DS3231	Робота з модулем годинника реального часу: зчитування та встановлення дати/часу, доступ до вбудованого датчика температури
SD.h, SPI.h	MicroSD модуль	Робота з файловою системою FAT на картках MicroSD через інтерфейс SPI: створення, читання, запис файлів
LoRa.h	LoRa SX1278	Бібліотека Sandeep Mistry для трансивера LoRa: ініціалізація, налаштування частоти та потужності, відправка та прийом пакетів
ArduinoJson.h	Формат даних	Серіалізація та десеріалізація даних у форматі JSON: формування пакетів для передачі та парсинг конфігураційних команд
ArduinoOTA.h	OTA-оновлення	Підтримка оновлення прошивки мікроконтролера «по повітрю» (Over-The-Air) без фізичного підключення USB-кабелю

У проєкті вона використовується для формування JSON-пакетів з результатами вимірювань перед відправкою на сервер, а також для парсингу конфігураційних повідомлень, що надходять від сервера (наприклад, зміна інтервалу вимірювань або порогових значень). Бібліотека підтримує як статичне (на стеку), так і динамічне (на купі) виділення пам'яті під JSON-документ, що дозволяє оптимізувати використання обмеженої RAM мікроконтролера.

Бібліотека ArduinoOTA.h забезпечує можливість оновлення прошивки мікроконтролера дистанційно через Wi-Fi мережу, що є критично важливим для

обслуговування вузлів, встановлених у важкодоступних каналізаційних колодязях. Процес OTA-оновлення включає: виявлення пристрою в мережі, перевірку автентичності (захист паролем), передачу бінарного образу нової прошивки, перевірку контрольної суми та автоматичне перезавантаження мікроконтролера із новою прошивкою.

Серверна частина системи побудована на стеку відкритих технологій, що забезпечує відсутність ліцензійних витрат, широкі можливості кастомізації та активну підтримку спільнотою розробників.

MQTT-брокер Eclipse Mosquitto. Eclipse Mosquitto є легковагим, високопродуктивним брокером повідомлень із відкритим вихідним кодом, що реалізує протокол MQTT. У розроблюваній системі Mosquitto виконує роль центральної точки збору даних: усі сенсорні вузли публікують свої вимірювання в брокер, а серверні сервіси підписуються на відповідні топіки та отримують повідомлення в реальному часі.

Серверна логіка системи реалізована мовою Python. Python є оптимальним вибором для задач обробки та аналізу даних завдяки наявності потужних бібліотек для наукових обчислень та машинного навчання. Основні бібліотеки Python, що використовуються у серверній частині системи, наведено в таблиці 2.4.

Для зберігання сенсорних даних обрано спеціалізовану базу даних часових рядів, розроблену компанією InfluxData. На відміну від реляційних баз даних загального призначення, InfluxDB спеціально оптимізована для прийому, зберігання та швидкого запиту великих обсягів даних, прив'язаних до часових міток, саме такого типу, що генерують сенсорні вузли. InfluxDB забезпечує надзвичайно високу швидкість запису (до сотень тисяч точок даних на секунду), ефективне стиснення даних на диску, потужну мову запитів Flux для складних аналітичних операцій (агрегування, фільтрація, обчислення ковзних вікон), а також вбудований механізм retention policy для автоматичного видалення застарілих даних.

Таблиця 2.4 — Основні бібліотеки Python для серверної частини системи

Бібліотека	Призначення у проєкті
raho-mqtt	Клієнт протоколу MQTT: підключення до брокера Mosquitto, підписка на топіки сенсорних вузлів, отримання повідомлень у режимі реального часу
influxdb-client	Клієнт для бази даних InfluxDB: запис вимірювань у базу, виконання запитів для отримання історичних та агрегованих даних
FastAPI	Сучасний веб-фреймворк для створення REST API: обробка HTTP-запитів від веб-інтерфейсу, автоматична генерація документації Swagger/OpenAPI
uvicorn	Асинхронний ASGI-сервер для запуску FastAPI-застосунку; забезпечує високу продуктивність обробки запитів
pandas	Бібліотека для аналізу та маніпуляції табличними даними: обчислення ковзних статистик, агрегування вимірювань за часовими інтервалами
numpy	Бібліотека для числових обчислень: математичні операції над масивами даних, обчислення статистичних показників (середнє, стандартне відхилення)
scikit-learn	Бібліотека машинного навчання: алгоритми виявлення аномалій (Isolation Forest, One-Class SVM), класифікація типу забруднювача
python-telegram-bot	Бібліотека для роботи з Telegram Bot API: відправка тривожних сповіщень у Telegram-чат або групу відповідальних осіб
python-dotenv	Завантаження конфігураційних змінних (паролі, токени, адреси) із файлу .env для безпечного зберігання секретів

Grafana – це провідна платформа з відкритим вихідним кодом для візуалізації та моніторингу даних часових рядів. У розроблюваній системі Grafana виконує роль основного інтерфейсу користувача для операторів та екологічних інспекторів. Grafana підключається безпосередньо до InfluxDB як джерело даних і дозволяє створювати інтерактивні дашборди без необхідності написання коду. Основні можливості Grafana, що використовуються у проєкті: різноманітні типи панелей візуалізації (графіки часових рядів, гістограми, калібрувальні шкали, таблиці, текстові панелі, географічні карти); гнучка система запитів до InfluxDB з можливістю побудови складних аналітичних виразів; вбудована система сповіщень, яка дозволяє налаштовувати правила тривоги безпосередньо у дашбордах; підтримка множини каналів оповіщення: email, Telegram, Slack, webhook; управління доступом.

Docker – це платформа для контейнеризації застосунків, що дозволяє пакувати програму разом з усіма її залежностями в ізольований контейнер. Docker Compose – це інструмент для визначення та запуску багатоконтейнерних застосунків за допомогою єдиного конфігураційного файлу `docker-compose.yml` у форматі YAML. У проєкті Docker Compose описує чотири основні сервіси: `mosquitto` (MQTT-брокер), `influxdb` (база даних), `grafana` (веб-інтерфейс) та `app` (Python-застосунок з логікою збору даних, аналізу та оповіщень). Розгортання всієї серверної інфраструктури виконується однією командою: `docker-compose up -d`.

Основним веб-інтерфейсом системи є дашборди Grafana, що не потребують окремої розробки фронтенд застосунку. Проте для реалізації додаткових функцій, що виходять за межі стандартних можливостей Grafana, передбачено можливість створення додаткового веб-застосунку на базі фреймворку React.js.

React.js – це JavaScript-бібліотека для побудови інтерактивних користувацьких інтерфейсів, що базується на компонентному підході та використовує віртуальний DOM для ефективного оновлення сторінки. Для

візуалізації даних у React-застосунку передбачено використання бібліотеки Recharts та Leaflet.js. Комунікація з серверною частиною здійснюється через REST API за допомогою HTTP-запитів.

Крім основних програмних засобів, у процесі проєктування та розробки кіберфізичної системи використовується ряд допоміжних інструментів, які суттєво підвищують ефективність та якість роботи.

Fritzing – програмний засіб для проєктування електронних схем. Fritzing використовується для створення наочних макетних схем, що ілюструють фізичне підключення компонентів, та для генерації принципів електричних схем.

MQTT Explorer – графічний клієнт для протоколу MQTT, що дозволяє підключатись до брокера Mosquitto, переглядати дерево топиків у реальному часі, спостерігати за повідомленнями, що публікуються сенсорними вузлами, та вручну публікувати тестові повідомлення.

Надійний та ефективний обмін даними між компонентами кіберфізичної системи забезпечується сукупністю протоколів різних рівнів, кожен з яких оптимізований для вирішення конкретних комунікаційних задач. У цьому підрозділі наведено детальний опис протоколів, що використовуються у розроблюваній системі: від низькорівневих апаратних інтерфейсів між компонентами сенсорного вузла до високорівневих мережевих протоколів передачі даних на сервер.

2.6 Висновки до розділу 2

У другому розділі кваліфікаційної роботи виконано повний та обґрунтований вибір компонентної бази та інструментальних засобів, необхідних для реалізації кіберфізичної системи «Розумна каналізаційна система».

В якості центрального мікроконтролера сенсорного вузла обрано ESP32 (ESP-WROOM-32), що забезпечує оптимальне поєднання обчислювальної потужності, кількості каналів АЦП, вбудованого Wi-Fi/BLE, наднизького

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 34
Зм.	Арк.	№ докум.	Підпис	Дата		

енергоспоживання у сплячому режимі та доступної вартості. Визначено набір з шести датчиків (рН, TDS, каламутність, розчинений кисень, ORP, температура), що охоплюють основні фізико-хімічні параметри стічних вод, необхідні для виявлення різних типів промислових забруднень. Загальна вартість комплекту датчиків для одного вузла не перевищує 130 USD.

Для бездротового зв'язку обрано комбінацію технологій Wi-Fi та LoRa (модуль SX1278) з протоколом MQTT як прикладним рівнем передачі. Апаратною базою шлюзу обрано Raspberry Pi 4 з модулем-концентратором LoRa RAK2245. Серверна частина розгортається у контейнеризованому середовищі Docker на VPS-сервері та включає MQTT-брокер Mosquitto, базу даних часових рядів InfluxDB 2.x, платформу візуалізації Grafana та Python-застосунок з логікою аналізу та оповіщень.

Обрані програмні засоби розробки (Arduino IDE, PlatformIO, Python, Docker, Git, Fritzing) є вільно поширюваними, що виключає ліцензійні витрати та забезпечує доступність проєкту для відтворення та подальшого розвитку. Детально описано набір бібліотек із зазначенням призначення кожної у проєкті.

Визначено та описано усі протоколи обміну даними, що використовуються у системі: MQTT для передачі сенсорних даних, HTTP/REST для взаємодії з веб-інтерфейсом, а також апаратні інтерфейси I²C, SPI, 1-Wire та UART для зв'язку між компонентами сенсорного вузла.

Таким чином, сформовано повний перелік апаратних компонентів, програмних засобів та протоколів, що становлять технологічну основу для переходу до етапу практичної реалізації системи, описаного у наступному розділі.

					КвРКІ. 302134.23.10.10 ПЗ	Арк. 35
Зм.	Арк.	№ докум.	Підпис	Дата		

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ПЕРЕВІРКА РОБОТИ КІБЕРФІЗИЧНОЇ СИСТЕМИ

3.1 Розробка структурної схеми системи

Проектування кіберфізичної системи «Розумна каналізаційна система» починається з розробки структурної схеми, яка відображає загальну організацію системи, визначає основні функціональні блоки та встановлює інформаційні зв'язки між ними. Структурна схема є верхнім рівнем абстракції проекту і слугує основою для подальшої деталізації кожного блоку у вигляді функціональних та принципівих схем.

Розроблена структурна схема кіберфізичної системи включає чотири основні функціональні рівні, з'єднані інформаційними каналами різного типу. На нижньому рівні розташовані сенсорні вузли автономні програмно-апаратні модулі, кожен з яких здійснює вимірювання фізико-хімічних параметрів стічних вод у конкретній точці каналізаційної мережі. Кількість сенсорних вузлів у системі є змінною та визначається протяжністю контрольованої ділянки каналізаційної мережі та кількістю точок підключення промислових підприємств. Для прототипної реалізації передбачено від 3 до 10 вузлів (рис .3.1).

Кожен сенсорний вузол на структурній схемі представлений як єдиний блок, що виконує три основні функції: збір даних (зчитування показань шести датчиків), попередня обробка (цифрова фільтрація, калібрувальні перетворення, формування пакета) та передача даних. Вихідним інформаційним потоком сенсорного вузла є JSON-пакет, що містить ідентифікатор вузла, часову мітку та значення всіх вимірюваних параметрів. Цей потік спрямовується до наступного рівня мережевого.

Мережевий рівень представлений шлюзами, кожен з яких обслуговує групу сенсорних вузлів у визначеній географічній зоні. На структурній схемі шлюз зображено як блок з двома інтерфейсами: з боку сенсорних вузлів бездротовий приймач LoRa та Wi-Fi точка прийому для ближніх вузлів; з боку

сервера підключення до мережі Інтернет через Ethernet або 4G. Зв'язок між сенсорними вузлами та шлюзом позначено на схемі як бездротовий канал (пунктирна лінія) з вказівкою технології (LoRa 868 МГц або Wi-Fi 2,4 ГГц) та протоколу прикладного рівня (MQTT). Зв'язок між шлюзом та сервером як дротовий або бездротовий канал Інтернет (суцільна лінія) з вказівкою протоколу MQTT поверх TCP/IP.

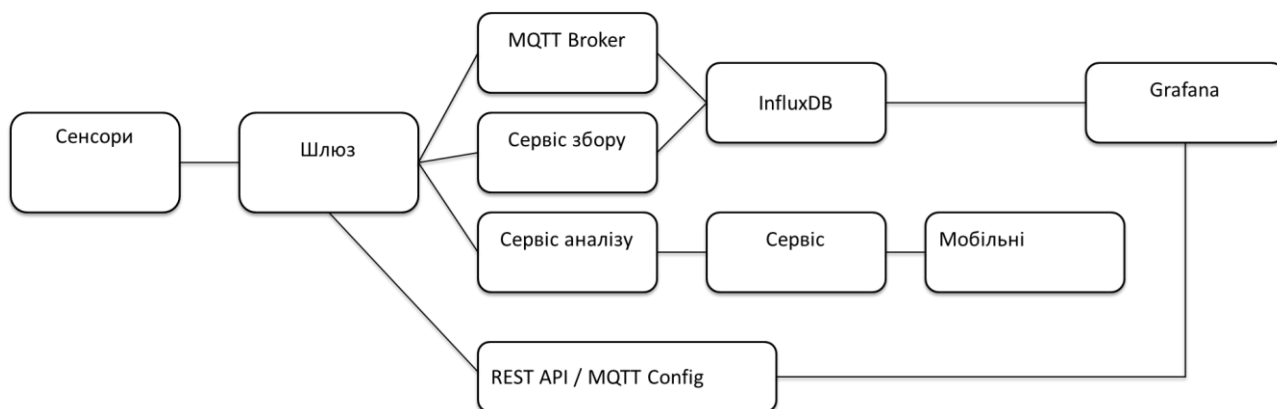


Рисунок 3.1 - Структурна схема системи

Серверний рівень на структурній схемі деталізований у вигляді кількох взаємопов'язаних функціональних блоків, розгорнутих у контейнеризованому середовищі Docker. Першим блоком є MQTT-брокер Mosquitto, який приймає повідомлення від усіх шлюзів та маршрутизує їх до підписників. До MQTT-брокера підключені три сервісні блоки: сервіс збору та збереження даних, який отримує повідомлення з топиків sewage/+/data та записує їх у базу даних; сервіс аналізу та виявлення аномалій, який в реальному часі аналізує вхідні дані та при виявленні відхилень публікує тривожне повідомлення у топик sewage/alerts; сервіс оповіщень, який підписаний на топик sewage/alerts і надсилає сповіщення відповідальним особам через e-mail або SMS.

Окремим блоком серверного рівня є база даних часових рядів InfluxDB, яка зберігає всі вимірювання та є джерелом даних для візуалізації. REST API забезпечує програмний інтерфейс для взаємодії зовнішніх застосунків з даними системи.

Верхній рівень структурної схеми – це рівень візуалізації та взаємодії з користувачем, який включає блок веб-інтерфейсу Grafana, який підключається до InfluxDB та REST API для відображення дашбордів моніторингу, а також блок мобільних сповіщень, який забезпечує оперативне інформування операторів незалежно від їхнього місцезнаходження.

На структурній схемі також позначено зворотний (низхідний) потік інформації. Цей потік використовується для дистанційного налаштування параметрів роботи вузлів: зміни інтервалу вимірювань, оновлення порогових значень, примусового позачергового вимірювання та оновлення прошивки (OTA). Команди від оператора надходять через веб-інтерфейс або REST API на MQTT-брокер, який публікує їх у відповідний топик `sewage/{node_id}/config`, звідки вони доставляються на цільовий сенсорний вузол.

Таким чином, структурна схема наочно демонструє модульну архітектуру системи, де кожен функціональний блок має чітко визначену зону відповідальності та взаємодіє з іншими блоками через стандартизовані інтерфейси (MQTT, HTTP, SQL-подібні запити Flux). Така архітектура забезпечує незалежність компонентів, можливість їхньої окремої заміни або модернізації, а також горизонтальне масштабування шляхом додавання нових сенсорних вузлів та шлюзів без змін серверної частини.

3.2 Розробка функціональної схеми сенсорного вузла

Функціональна схема сенсорного вузла деталізує внутрішню структуру центрального елемента кіберфізичної системи, розкриваючи склад підсистем, їхні функції та інформаційні зв'язки між ними (рис. 3.2). На відміну від структурної схеми всієї системи, де сенсорний вузол представлений єдиним блоком, функціональна схема відображає кожен апаратний та програмний компонент вузла як окремий функціональний елемент.

Розроблена функціональна схема сенсорного вузла включає шість основних підсистем: підсистему сенсорів, підсистему перетворення та обробки

сигналів, підсистему керування, підсистему зв'язку, підсистему локального зберігання даних та підсистему електроживлення. Кожна підсистема виконує специфічну роль у загальному циклі роботи вузла.

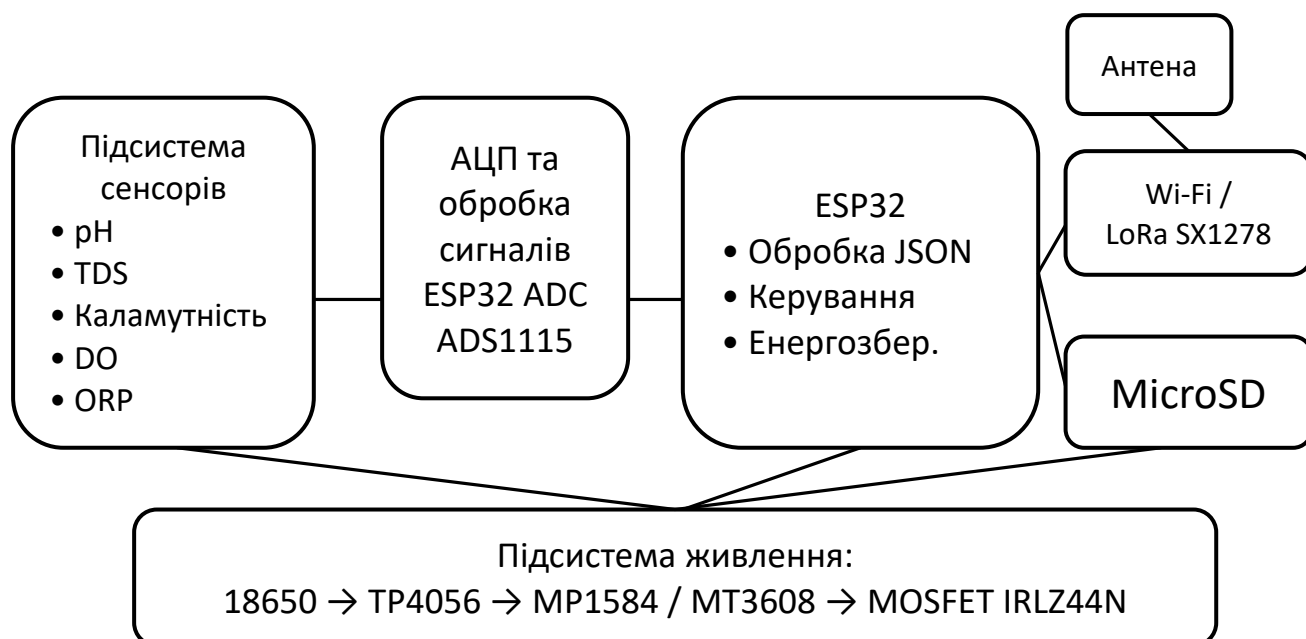


Рисунок 3.2 – Функціональна схема сенсорного вузла

Підсистема сенсорів розташована на вході інформаційного потоку та включає шість вимірювальних елементів: рН-електрод (DFRobot SEN0161-V2), датчик електропровідності (DFRobot SEN0244), датчик каламутності (DFRobot SEN0189), датчик розчиненого кисню (DFRobot SEN0237-A), ORP-електрод (DFRobot SEN0165) та цифровий датчик температури (DS18B20). Кожен сенсор безпосередньо контактує з досліджуваним середовищем, стічними водами, і перетворює вимірюваний фізико-хімічний параметр у відповідний електричний сигнал. П'ять з шести датчиків (рН, TDS, каламутність, DO, ORP) формують аналогові сигнали у вигляді напруги постійного струму, а датчик температури DS18B20 видає дані безпосередньо в цифровій формі по інтерфейсу 1-Wire.

Підсистема перетворення та обробки сигналів забезпечує перетворення аналогових сигналів від датчиків у цифрову форму, придатну для обробки мікроконтролером. Ця підсистема включає два типи аналого-цифрових

перетворювачів: вбудований 12-бітний АЦП мікроконтролера ESP32, де використовується для датчиків рН, TDS та каламутності, які мають достатній рівень вихідного сигналу, та зовнішній 16-бітний АЦП ADS1115, підключений по шині I²C, де використовується для датчиків DO та ORP, які потребують вищої роздільної здатності перетворення. На функціональній схемі підсистема перетворення позначена як проміжний блок між сенсорами та мікроконтролером, з відповідними позначками типу інтерфейсу. Крім того, кожен аналоговий датчик має власну плату підсилювача-перетворювача сигналу, яка входить до комплекту датчика DFRobot і виконує функції підсилення, фільтрації та узгодження рівнів напруги.

Підсистема керування є центральним елементом функціональної схеми і представлена мікроконтролером ESP32 (ESP-WROOM-32). Мікроконтролер виконує такі функції: керування послідовністю операцій: ініціалізація, вимірювання, обробка, передача, сон; зчитування цифрових значень з вбудованого та зовнішнього АЦП; отримання температури по шині 1-Wire; попередня цифрова обробка даних; формування JSON-пакета з результатами вимірювань; керування модулем бездротового зв'язку для передачі пакета; керування MOSFET-ключем для увімкнення та вимкнення живлення датчиків; отримання поточного часу від модуля RTC DS3231 по шині I²C; запис даних на MicroSD-карту у разі відсутності зв'язку; керування режимами енергозбереження.

На функціональній схемі від блоку мікроконтролера відходять численні інформаційні зв'язки до всіх інших підсистем. Ці зв'язки згруповані за типом інтерфейсу: аналогові входи (GPIO34, GPIO35, GPIO32, GPIO33 до вбудованого АЦП); шина I²C (GPIO21/SDA та GPIO22/SCL до ADS1115 та DS3231); шина 1-Wire (GPIO4 до DS18B20); шина SPI (GPIO18/SCK, GPIO19/MISO, GPIO23/MOSI до модуля MicroSD та модуля LoRa з окремими лініями вибору CS); керуючий вихід (GPIO25 до MOSFET-ключа живлення датчиків).

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 40
Зм.	Арк.	№ док.ум.	Підпис	Дата		

Підсистема зв'язку забезпечує бездротову передачу даних від сенсорного вузла до шлюзу або безпосередньо до MQTT-брокера на сервері. На функціональній схемі ця підсистема представлена двома альтернативними каналами: вбудований Wi-Fi модуль ESP32 використовується, коли вузол знаходиться в зоні покриття Wi-Fi мережі та зовнішній модуль LoRa SX1278, підключений по шині SPI використовується для віддалених вузлів. Вибір каналу зв'язку є конфігураційним параметром, що визначається при встановленні вузла та може бути змінений дистанційно. На схемі обидва канали зображені як паралельні гілки, що виходять з блоку мікроконтролера і мають на виході символ антени.

Підсистема локального зберігання представлена модулем MicroSD-карти, підключеним до мікроконтролера по шині SPI з окремою лінією CS на GPIO5. На функціональній схемі цей блок розташований паралельно до підсистеми зв'язку і виконує функцію резервного збереження даних. Двонаправлена стрілка між мікроконтролером та MicroSD відображає операції як запису збереження вимірювань при відсутності зв'язку, так і читання повторна передача збережених даних після відновлення зв'язку.

Підсистема електроживлення на функціональній схемі охоплює весь ланцюг від джерела енергії до споживачів: літій-іонний акумулятор 18650 → модуль зарядки TP4056 (з опціональним підключенням сонячної панелі) → DC-DC перетворювач MP1584 (вихід 3,3 В для мікроконтролера та логічних модулів) та лінійний стабілізатор AMS1117-5.0 (вихід 5 В для датчиків) → MOSFET-ключ IRLZ44N (програмне управління живленням датчиків від мікроконтролера). На схемі лінії живлення позначено іншим кольором або типом лінії (товста суцільна) для візуального відокремлення від інформаційних зв'язків.

Інформаційний потік через функціональну схему сенсорного вузла можна описати послідовністю: фізико-хімічний параметр середовища → датчик (перетворення у електричний сигнал) → плата підсилювача (узгодження рівнів) → АЦП (перетворення в цифрову форму) → мікроконтролер (обробка,

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 41
Зм.	Арк.	№ докum.	Підпис	Дата		

калібрування, фільтрація) → JSON-пакет → модуль зв'язку (Wi-Fi або LoRa) → антена → радіоканал до шлюзу. Паралельно JSON-пакет може бути записаний на MicroSD-карту для резервного зберігання. Увесь цикл, від подачі живлення на датчики до переходу мікроконтролера у глибокий сон, займає приблизно 45–60 секунд.

3.3 Розробка принципової електричної схеми сенсорного вузла

Принципова електрична схема є ключовим проєктним документом, що визначає повний склад електронних компонентів сенсорного вузла та всі електричні з'єднання між ними. На підставі цієї схеми виконується фізичний монтаж прототипу та, при необхідності, проєктування друкованої плати для серійного виготовлення. Принципова схема розроблена у середовищі Fritzing та містить три функціональні частини: схему підключення датчиків до мікроконтролера, схему ланцюга живлення та схему підключення модулів зв'язку та зберігання (рис. 3.1).

Центральним елементом принципової схеми є модуль ESP32 DevKit V1, що має дві паралельні гребінки по 19 виводів з кожного боку. Модуль живиться напругою 3,3 В на виводі 3V3 або 5 В на виводі VIN з вбудованим стабілізатором. Для даного проєкту живлення подається на вивід 3V3 від зовнішнього DC-DC перетворювача, що забезпечує стабільну напругу 3,3 В з високим ККД.

Підключення аналогових датчиків здійснюється до каналів першого блоку АЦП мікроконтролера (ADC1), оскільки другий блок (ADC2) не може використовуватися одночасно з Wi-Fi модулем. Вихід аналогового сигналу плати підсилювача рН-датчика, позначений як P₀ на платі DFRobot, з'єднується з виводом GPIO34 мікроконтролера, який відповідає каналу ADC1_CH6. Живлення плати підсилювача рН — від шини 5 В (вивід VCC) та спільної землі (GND). Аналогічно підключаються інші аналогові датчики: вихід модуля TDS (аналоговий вихід A) до GPIO35 (ADC1_CH7); вихід модуля каламутності (аналоговий вихід) до GPIO32 (ADC1_CH4); вихід модуля ORP (аналоговий

					КвРКІ. 302134.23.10.10 ПЗ	Арк. 42
Зм.	Арк.	№ докum.	Підпис	Дата		

вихід) до GPIO33 (ADC1_CH5). Усі плати підсилювачів датчиків живляться від шини 5 В та мають спільну землю з мікроконтролером.

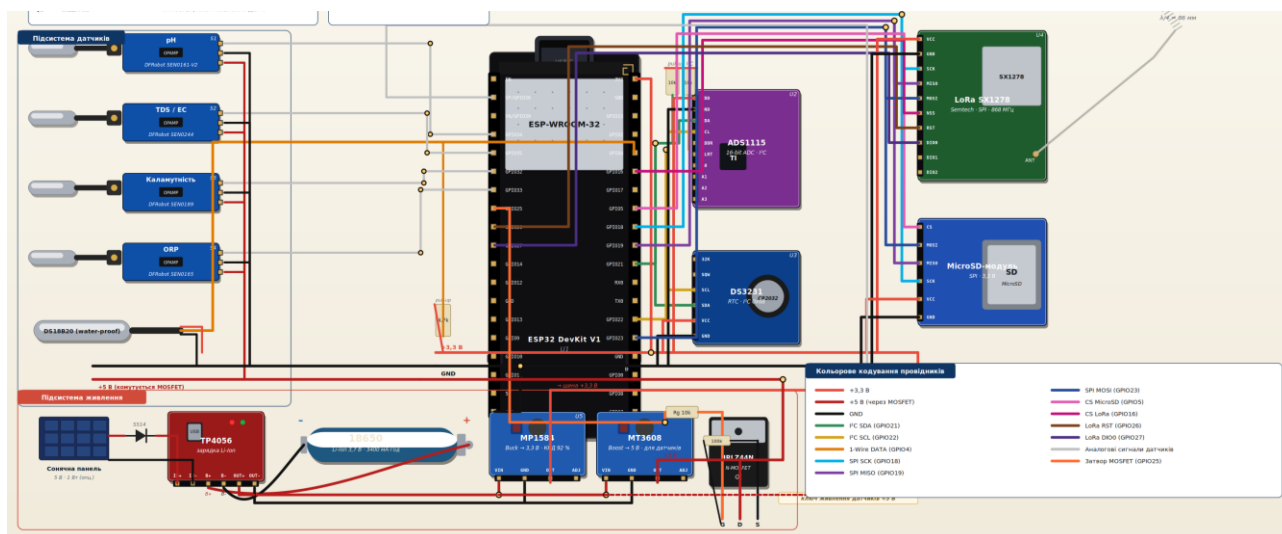


Рисунок 3.1 – Схема принципова електрична сенсорного вузла

Для зведення інформації про розподіл виводів мікроконтролера складено таблицю 3.1, яка є обов’язковим додатком до принципової схеми та використовується при фізичному монтажі.

Зовнішній АЦП ADS1115 підключається до шини I²C на виводах GPIO21 (SDA) та GPIO22 (SCL). Лінії шини I²C підтягуються до напруги 3,3 В через резистори 10 кОм. Адреса ADS1115 на шині I²C визначається станом виводу ADDR: при з’єднанні з GND адреса дорівнює 0x48 (використано за замовчуванням). Аналоговий вихід датчика розчиненого кисню підключається до каналу AIN0 модуля ADS1115, а вихід ORP-датчика (дублюючий канал для підвищеної точності) — до каналу AIN1. Живлення ADS1115 — від шини 3,3 В.

На тій самій шині I²C розташований модуль годинника реального часу DS3231 з фіксованою адресою 0x68. Модуль має власну батарею CR2032 для збереження часу при відключенні основного живлення. Живлення логічної частини DS3231 — від шини 3,3 В.

Датчик температури DS18B20 у водонепроникному виконанні підключається трьома проводами: червоний (VCC) — до шини 3,3 В, чорний

(GND) — до спільної землі, жовтий (DATA) — до GPIO4. Між виводами DATA та VCC встановлюється підтягуючий резистор номіналом 4,7 кОм, який є обов'язковим для коректної роботи протоколу 1-Wire та забезпечує повернення лінії даних до високого рівня між імпульсами.

Таблиця 3.1 — Розподіл виводів GPIO мікроконтролера ESP32

GPIO	Функція	Компонент	Примітка
GPIO34	ADC1_CH6	pH датчик	Тільки вхід, без підтяжки
GPIO35	ADC1_CH7	TDS датчик	Тільки вхід, без підтяжки
GPIO32	ADC1_CH4	Каламутність	Аналоговий вхід
GPIO33	ADC1_CH5	ORP датчик	Аналоговий вхід
GPIO21	I ² C SDA	ADS1115, DS3231	Підтяжка 10 кОм до 3,3 В
GPIO22	I ² C SCL	ADS1115, DS3231	Підтяжка 10 кОм до 3,3 В
GPIO4	1-Wire Data	DS18B20	Підтяжка 4,7 кОм до 3,3 В
GPIO5	SPI CS (SD)	MicroSD модуль	Chip Select для SD-карти
GPIO18	SPI SCK	MicroSD, LoRa	Спільна тактова лінія SPI
GPIO19	SPI MISO	MicroSD, LoRa	Спільна лінія даних (вхід)
GPIO23	SPI MOSI	MicroSD, LoRa	Спільна лінія даних (вихід)
GPIO16	SPI CS (LoRa)	LoRa SX1278	Chip Select для LoRa
GPIO26	LoRa RST	LoRa SX1278	Лінія апаратного скидання
GPIO27	LoRa DIO0	LoRa SX1278	Переривання при прийомі/передачі
GPIO25	Digital OUT	MOSFET IRLZ44N	Керування живленням датчиків

Зм.	Арк.	№ докум.	Підпис	Дата

Поблизу виводів живлення кожного модуля (ADS1115, DS3231, плати підсилювачів датчиків, модуль LoRa, модуль MicroSD) встановлюються блокувальні керамічні конденсатори ємністю 100 нФ. Ці конденсатори фільтрують високочастотні перешкоди по шині живлення, які можуть виникати при комутації цифрових сигналів та негативно впливати на точність аналогових вимірювань. Додатково на головній шині живлення 3,3 В та 5 В встановлюються електролітичні конденсатори ємністю 220–470 мкФ для згладжування більш низькочастотних пульсацій.

Ланцюг живлення сенсорного вузла на принциповій схемі починається від літій-іонного акумулятора формату 18650 з номінальною напругою 3,7 В. Позитивний вивід акумулятора підключається до входу V+ модуля зарядки TP4056, негативний до V-. Модуль TP4056 має вбудовану мікросхему захисту DW01A з двома MOSFET-транзисторами, яка захищає акумулятор від перезарядки обмеження напруги 4,2 В, перерозрядки відключення при напрузі нижче 2,5 В, надмірного струму та короткого замикання. Вхід зарядки Micro-USB або контактні площадки IN+/IN- підключається до опціональної сонячної панелі 5 В / 1 Вт через діод Шоттки (наприклад, SS14 або 1N5819), який запобігає зворотному струму від акумулятора до панелі у темну пору доби.

Вихід модуля TP4056 контакти OUT+/OUT- подає напругу акумулятора (3,0–4,2 В залежно від рівня заряду) на два паралельні стабілізатори. Понижуючий DC-DC перетворювач на базі мікросхеми MP1584EN (вхідний діапазон 4,5–28 В, вихідний регульований, встановлено 3,3 В підбором резисторного дільника на платі) формує стабільну напругу 3,3 В для живлення мікроконтролера ESP32, модулів ADS1115, DS3231, MicroSD та LoRa. Коефіцієнт корисної дії DC-DC перетворювача MP1584 складає до 92%, що значно ефективніше порівняно з лінійним стабілізатором (де ККД визначається співвідношенням вхідної та вихідної напруг). Другий стабілізатор – це підвищуючий DC-DC перетворювач на базі мікросхеми MT3608 формує напругу

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 45
Зм.	Арк.	№ докум.	Підпис	Дата		

5 В від акумуляторної батареї для живлення плат підсилювачів аналогових датчиків, які потребують саме 5 В для коректної роботи.

Ключовим елементом схеми живлення є N-канальний MOSFET-транзистор IRLZ44N, встановлений у ланцюг шини живлення 5 В датчиків. Цей транзистор керується мікроконтролером через вивід GPIO25. Стік транзистора підключається до від'ємного виводу навантаження (спільна земля датчиків), витік до загальної землі схеми, затвор до GPIO25 через резистор 10 кОм. Між затвором та витіком встановлено резистор 100 кОм для забезпечення гарантованого закритого стану транзистора при невизначеному стані GPIO. Коли мікроконтролер встановлює високий рівень (3,3 В) на GPIO25, транзистор відкривається і живлення подається на датчики. При низькому рівні або переході мікроконтролера в режим глибокого сну транзистор закривається і живлення датчиків повністю припиняється, що знижує споживання вузла до одиниць мікроампер.

Для моніторингу рівня заряду акумулятора на принциповій схемі передбачено резисторний дільник напруги, підключений між виходом акумулятора (OUT+ модуля TP4056) та аналоговим входом мікроконтролера GPIO36 (ADC1_CH0). Дільник складається з двох резисторів: $R1 = 100$ кОм (верхнє плече) та $R2 = 100$ кОм, що забезпечує коефіцієнт ділення 0,5.

Модуль LoRa на базі трансивера SX1278 підключається до мікроконтролера ESP32 по шині SPI. Відповідно до принципової схеми, з'єднання виконано наступним чином: вивід NSS (Chip Select) модуля SX1278 до GPIO16 мікроконтролера; вивід MOSI до GPIO23; вивід MISO до GPIO19; вивід SCK до GPIO18; вивід RESET до GPIO26; вивід DIO0 (переривання при завершенні передачі або прийомі пакета) до GPIO27. Живлення модуля SX1278 від шини 3,3 В. До виводу антени (ANT) модуля підключається дротова чвертьхвильова антена довжиною 86 мм (для частоти 868 МГц) або зовнішня антена через SMA-роз'єм.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 46
Зм.	Арк.	№ докум.	Підпис	Дата		

Модуль MicroSD-карти підключається до тієї ж шини SPI, що й модуль LoRa, але з окремою лінією Chip Select: вивід CS модуля MicroSD до GPIO5 мікроконтролера. Лінії MOSI, MISO та SCK є спільними для обох SPI-пристроїв. Протокол SPI передбачає, що в будь-який момент часу активним може бути лише один пристрій (той, на лінії CS якого встановлено низький рівень), тому спільне використання шини є коректним і не створює конфліктів. Живлення модуля MicroSD від шини 3,3 В. Модуль має вбудований стабілізатор напруги та адаптер рівнів логічних сигналів, що забезпечує сумісність із 3,3-вольтовою логікою ESP32.

На принциповій схемі усі з'єднання позначені відповідно до стандартів оформлення електричних схем: кожен компонент має позиційне позначення (U1-ESP32, U2 - ADS1115, U3 - DS3231, U4 - SX1278, Q1 - IRLZ44N тощо); вузли з'єднань позначені точками; перетини ліній без з'єднання — без точок; шини живлення та землі позначені відповідними символами. Повна принципова схема у форматі Fritzing наведена у Додатку А до кваліфікаційної роботи.

3.4 Програмна реалізація прошивки сенсорного вузла

Прошивка сенсорного вузла реалізує циклічний алгоритм роботи, оптимізований для мінімізації енергоспоживання при збереженні необхідної періодичності вимірювань. Загальний алгоритм описується блок-схемою, яка визначає послідовність операцій від моменту пробудження мікроконтролера до переходу в режим глибокого сну.

Після пробудження мікроконтролера за сигналом від вбудованого RTC-таймера або при першому увімкненні живлення виконується етап ініціалізації. На цьому етапі послідовно виконуються такі операції: ініціалізація послідовного порту UART на швидкості 115200 бод для виведення налагоджувальних повідомлень; ініціалізація шини I²C та перевірка наявності підключених пристроїв за їхніми адресами; ініціалізація шини 1-Wire та пошук датчика DS18B20; ініціалізація шини SPI та перевірка наявності MicroSD-карти;

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 47
Зм.	Арк.	№ докum.	Підпис	Дата		

зчитування поточного часу з модуля RTC DS3231; завантаження калібрувальних коефіцієнтів та конфігураційних параметрів інтервалу вимірювань, порогових значень з енергонезалежної пам'яті мікроконтролера ESP32.

Після успішної ініціалізації мікроконтролер подає команду на відкриття MOSFET-ключа (встановлює високий рівень на GPIO25), що подає живлення на датчики. Далі витримується пауза тривалістю 30–60 секунд, необхідна для стабілізації показань електрохімічних датчиків (особливо рН-електрода та ORP-електрода), які потребують часу для встановлення рівноважного потенціалу після подачі живлення.

Етап зчитування показань є ключовим у циклі роботи. Мікроконтролер послідовно опитує кожен датчик, зчитуючи по 10 значень з кожного з інтервалом 50 мс між зчитуваннями. Множинне зчитування необхідне для подальшої медіанної фільтрації, яка ефективно усуває випадкові артефактні вимірювання, спричинені електромагнітними перешкодами або нестабільністю контакту електрода з середовищем. Послідовність опитування: спочатку зчитується температура (DS18B20), оскільки її значення необхідне для температурної компенсації інших датчиків; потім рН, TDS, каламутність через вбудований АЦП; далі DO та ORP через зовнішній АЦП ADS1115; нарешті, напруга акумулятора через резисторний дільник.

Етап попередньої обробки включає три послідовні операції для кожного масиву з 10 зчитувань. Перша операція – це медіанна фільтрація, де масив з 10 значень сортується за зростанням, і вибирається медіана, тобто середнє з двох центральних елементів для парної кількості). Медіана є стійкою до одиничних викидів, на відміну від арифметичного середнього. Друга операція – це калібрувальне перетворення, тобто відфільтроване значення напруги (або коду АЦП) перетворюється у фізичну величину за калібрувальною формулою. Для рН використовується лінійна залежність: $pH = k \times voltage + b$, де коефіцієнти k та b визначаються при двоточковому калібруванні (буферні розчини рН 4,0 та рН 7,0). Для TDS перетворення напруги у концентрацію в ppm за поліноміальною

залежністю. Для каламутності перетворення напруги у NTU за зворотною поліноміальною залежністю. Третя операція температурна компенсація: для датчиків рН та TDS вносяться поправки на відхилення температури від стандартної (25 °C).

Після обробки мікроконтролер формує JSON-пакет, що містить ідентифікатор вузла (`node_id`), часову мітку від RTC у форматі ISO 8601, значення всіх шести вимірюваних параметрів та напругу акумулятора. Типовий JSON-пакет має такий вигляд:

```
{
  "node_id": "NODE_01",
  "timestamp": "2026-05-04T14:23:05Z",
  "data": {
    "pH": 7.24,
    "tds": 485,
    "turbidity": 120.5,
    "orp": 215,
    "do": 6.8,
    "temp": 18.3,
    "battery": 3.82
  }
}
```

Розмір такого пакета складає приблизно 200–250 байт, що є цілком прийнятним для передачі як по Wi-Fi, так і по LoRa.

На етапі передачі мікроконтролер намагається встановити з'єднання з MQTT-брокером. Якщо вузол сконфігурований для роботи через Wi-Fi, виконується підключення до точки доступу (з таймаутом 10 с), а потім підключення до MQTT-брокера (з таймаутом 5 с). Якщо обидва підключення успішні, JSON-пакет публікується в топик `sewage/{node_id}/data` з рівнем QoS 1 з підтвердженням доставки. Якщо вузол працює через LoRa, пакет безпосередньо відправляється як LoRa-фрейм на налаштованій частоті.

У разі невдалої спроби передачі JSON-пакет записується на MicroSD-карту у файл з ім'ям, що відповідає поточній даті. При наступному успішному підключенні мікроконтролер перевіряє наявність незвітованих записів на

					КвРКІ. 302134.23.10.10 ПЗ	Арк. 49
Зм.	Арк.	№ докум.	Підпис	Дата		

MicroSD та послідовно відправляє їх на сервер, реалізуючи механізм store-and-forward. Це гарантує, що жодне вимірювання не буде втрачено навіть при тривалих перебоях зв'язку.

Після завершення передачі мікроконтролер вимикає живлення датчиків встановлює низький рівень на GPIO25, закриваючи MOSFET-ключ, відключається від Wi-Fi, виконує MQTT disconnect та переходить у режим глибокого сну. Перед переходом у сон задається інтервал пробудження: у нормальному режимі 5 хвилин (300 секунд); в режимі тривоги (якщо попереднє вимірювання показало перевищення будь-якого порогу) 30 секунд для забезпечення частішого моніторингу аномальної ситуації. Значення інтервалу зберігається в RTC-пам'яті мікроконтролера.

Модуль зчитування датчиків реалізований як набір функцій мовою C/C++, кожна з яких інкапсулює логіку роботи з конкретним датчиком та повертає відфільтроване та калібрувальне значення вимірюваного параметра.

Функція readMedian є базовою утилітою, що використовується усіма функціями зчитування аналогових датчиків. Вона приймає номер GPIO-піна та кількість зчитувань (за замовчуванням 10), виконує серію послідовних зчитувань analogRead з інтервалом 50 мс, сортує отриманий масив та повертає медіанне значення. Алгоритм сортування — вставками (insertion sort), який є оптимальним для масивів малого розміру (до 20 елементів). Медіанна фільтрація ефективно усуває одиничні аномальні зчитування, спричинені електромагнітними перешкодами, нестабільністю живлення або іншими артефактами.

Функція readTemperature здійснює запит до датчика DS18B20 по протоколу 1-Wire. Спочатку викликається метод requestTemperatures бібліотеки DallasTemperature, який ініціює вимірювання температури датчиком (процес займає до 750 мс у 12-бітному режимі). Потім зчитується результат методом getTempCByIndex(0), де 0 — індекс датчика на шині. Функція перевіряє отримане значення на валідність: якщо повернено $-85\text{ }^{\circ}\text{C}$ (код помилки 1-Wire) або значення виходить за фізично обґрунтований діапазон ($-10\dots+60\text{ }^{\circ}\text{C}$ для

стічних вод), повертається код помилки та активується відповідний прапорець у діагностичному полі пакета.

Функція readPH зчитує медіанне значення напруги з GPIO34, перетворює код АЦП у напругу ($voltage = adcValue \times 3,3 / 4095$) та застосовує лінійну калібрувальну формулу: $pH = pH_{Slope} \times voltage + pH_{Offset}$. Калібрувальні коефіцієнти pH_Slope та pH_Offset зберігаються в NVS-пам'яті мікроконтролера та оновлюються при калібруванні. Після калібрувального перетворення вносяться поправки на температуру: різниця відповіді рН-електрода при поточній температурі t та стандартній $25\text{ }^{\circ}\text{C}$ описується рівнянням Нернста і становить приблизно $0,003\text{ рН}/^{\circ}\text{C}$ для кожної одиниці відхилення від нейтрального рН 7,0.

Функція readTDS зчитує медіанне значення напруги з GPIO35 та виконує температурну компенсацію за формулою: $compensatedVoltage = rawVoltage / (1,0 + 0,02 \times (temperature - 25,0))$, де temperature — значення, отримане від функції readTemperature, а коефіцієнт 0,02 відповідає приблизно 2% зміни провідності на кожен градус Цельсія. Скомпенсована напруга потім перетворюється у значення TDS (ppm) за поліноміальною залежністю другого порядку: $tds = (133,42 \times V^3 - 255,86 \times V^2 + 857,39 \times V) \times 0,5$, де V — компенсована напруга, а коефіцієнт 0,5 враховує подвоєння від перетворювача.

Функція readTurbidity зчитує медіанне значення з GPIO32. Перетворення напруги у значення NTU здійснюється за зворотною залежністю: чистій воді відповідає максимальна напруга ($\sim 4,2\text{ В}$), а зростанню каламутності — зниження напруги. Використовується кусково-лінійна апроксимація, калібрувальні точки якої визначаються експериментально з використанням еталонних розчинів каламутності. Для грубої оцінки у прототипі використовується спрощена формула: $turbidity = \sum 74,04 \times V^2 + 662,27 \times V - 780,56$, де V — це виміряна напруга.

Функція readDO використовує зовнішній АЦП ADS1115 для підвищеної точності. Зчитування здійснюється через бібліотеку Adafruit_ADS1X15: виклик методу ads.readADC_SingleEnded(0) повертає 16-бітний код АЦП, який

перетворюється в напругу з урахуванням встановленого коефіцієнта підсилення PGA. Калібрувальне перетворення напруги в концентрацію розчиненого кисню (мг/л) здійснюється за двоточною лінійною моделлю, де точками калібрування є: нульовий розчин (розчин Na₂SO₃, DO = 0 мг/л) та насичений киснем розчин (вода при інтенсивному аерації, DO відповідає табличному значенню для даної температури).

Функція readORP зчитує значення з каналу AIN1 модуля ADS1115 та перетворює код АЦП у мілівольти. Калібрування виконується за однотоочною моделлю з використанням стандартного розчину з відомим значенням ORP, наприклад, розчин Зобелла, ORP = +220 мВ при 25 °С. Функція повертає значення ORP у мілівольтах зі знаком.

Функція readBattery зчитує напругу з GPIO36 через резисторний дільник. Код АЦП перетворюється у напругу, яка подвоюється, коефіцієнт дільника 0,5, для отримання фактичної напруги акумулятора. Значення напруги включається у JSON-паке́т для моніторингу стану живлення кожного вузла на серверному рівні та генерації сповіщення при зниженні рівня заряду нижче критичного порогу (наприклад, 3,3 В, що відповідає приблизно 10% залишкового заряду Li-Ion акумулятора).

Модуль передачі даних відповідає за надійну доставку JSON-паке́тів з результатами вимірювань від сенсорного вузла до MQTT-брокера на сервері. Реалізація модуля розрізняється залежно від сконфігурованого каналу зв'язку (Wi-Fi або LoRa), але має єдиний зовнішній інтерфейс: функція sendData, яка приймає сформований JSON-рядок та повертає булеве значення результат спроби передачі.

У разі використання Wi-Fi модуль послідовно виконує три операції: підключення до Wi-Fi мережі, підключення до MQTT-брокера та публікацію повідомлення. Підключення до Wi-Fi здійснюється функцією WiFi.begin з передачею SSID та пароля мережі, збережених у NVS-пам'яті. Для підключення встановлюється таймаут 10 секунд: якщо протягом цього часу з'єднання не

встановлено, функція повертає помилку. Після успішного підключення до Wi-Fi ініціалізується MQTT-клієнт бібліотеки PubSubClient: задається адреса та порт брокера, де за замовчуванням 1883 для незашифрованого з'єднання, встановлюється ідентифікатор клієнта (client ID, рівний node_id), реєструється LWT-повідомлення (топик sewage/{node_id}/status, payload — "offline", QoS 1, retain = true).

Підключення до MQTT-брокера виконується функцією client.connect з передачею ідентифікатора клієнта, логіна та пароля (якщо автентифікація увімкнена). Реалізовано механізм повторних спроб з експоненційною затримкою (exponential backoff): перша повторна спроба через 1 секунду, друга через 2 секунди, третя через 4 секунди, після чого відмова та перехід до збереження на MicroSD. Такий підхід запобігає надмірному навантаженню на брокер у разі його тимчасової недоступності.

Публікація JSON-пакета здійснюється викликом client.publish із зазначенням топика (sewage/{node_id}/data), корисного навантаження (JSON-рядок) та рівня QoS 1. При QoS 1 брокер надсилає підтвердження (PUBACK) після отримання повідомлення, і бібліотека PubSubClient автоматично повторює передачу, якщо підтвердження не отримано протягом визначеного таймауту. Після успішної публікації вузол також публікує діагностичне повідомлення у топик sewage/{node_id}/status з payload "online" та інформацією про рівень сигналу Wi-Fi (RSSI) та напругу акумулятора.

У разі використання LoRa передача здійснюється через бібліотеку LoRa.h. Функція LoRa.beginPacket ініціює формування пакета, LoRa.print записує JSON-рядок у буфер передачі, LoRa.endPacket виконує фактичну передачу по радіоканалу. Параметри модуляції LoRa (spreading factor = 7, bandwidth = 125 кГц, coding rate = 4/5) забезпечують оптимальний баланс між дальністю, швидкістю та стійкістю до перешкод. На рівні LoRa немає вбудованого механізму підтвердження доставки (на відміну від LoRaWAN), тому для

підвищення надійності кожне повідомлення передається двічі з інтервалом 500 мс.

Механізм store-and-forward реалізований для забезпечення збереження даних при тимчасовій втраті зв'язку. Якщо функція sendData повернула помилку (передача не відбулася), JSON-пакет записується на MicroSD-карту за допомогою бібліотеки SD.h. Дані додаються у файл з ім'ям, що відповідає поточній даті у форматі YYYYMMDD.csv, у кінець файлу. Кожен запис – це один рядок, що містить повний JSON-пакет. При наступному успішному підключенні до MQTT-брокера мікроконтролер перевіряє наявність файлів з незвітованими даними на MicroSD, послідовно зчитує рядки та публікує їх у MQTT з позначкою retained = false. Після успішної відправки всіх накопичених записів файл видаляється з MicroSD-карти.

Енергозбереження є критично важливим аспектом прошивки, оскільки від нього безпосередньо залежить час автономної роботи сенсорного вузла від акумулятора. Мікроконтролер ESP32 підтримує кілька режимів зниженого енергоспоживання, з яких у проєкті використовується режим глибокого сну як найбільш ефективний.

У режимі глибокого сну вимикаються обидва процесорні ядра, більша частина оперативної пам'яті, Wi-Fi та Bluetooth модулі, цифрові периферійні пристрої. Залишається активним лише RTC-контролер з невеликим обсягом RTC-пам'яті (8 КБ) та RTC-таймер, який відраховує час до наступного пробудження. Споживання в цьому режимі складає приблизно 10 мкА, що у десятки тисяч разів менше за активний режим.

Перед переходом у режим глибокого сну прошивка виконує підготовчу послідовність дій: відключення від Wi-Fi; відключення від MQTT-брокера; закриття MOSFET-ключа живлення датчиків; збереження критичних змінних у RTC-пам'яті. RTC-пам'ять зберігає дані протягом deep sleep і доступна після пробудження, на відміну від звичайної RAM, яка повністю очищується.

Інтервал пробудження задається функцією `esp_sleep_enable_timer_wakeup`, яка приймає значення у мікросекундах. У прошивці реалізовано адаптивний інтервал: якщо попереднє вимірювання не виявило аномалій встановлюється стандартний інтервал 5 хвилин (300 000 000 мкс); якщо будь-який параметр перевищив пороговий рівень інтервал скорочується до 30 секунд для забезпечення оперативного відстеження динаміки аномалії. Прапорець аномалії зберігається в RTC-пам'яті та скидається, коли три послідовні вимірювання показують нормальні значення після цього вузол повертається до стандартного 5-хвилинного інтервалу.

Після виклику `esp_deep_sleep_start` мікроконтролер миттєво переходить у режим глибокого сну. Пробудження відбувається аналогічно апаратному скиданню: виконання програми починається з початку функції `setup()`, тому прошивка перевіряє причину пробудження та відновлює контекст з RTC-пам'яті.

3.5 Програмна реалізація серверної частини

Серверна частина кіберфізичної системи «Розумна каналізаційна система» побудована за мікросервісною архітектурою, де кожен функціональний компонент виконується як окремий ізольований контейнер Docker. Оркестрація контейнерів здійснюється за допомогою Docker Compose, що дозволяє описати всю серверну інфраструктуру єдиним конфігураційним файлом `docker-compose.yml` та розгорнути її однією командою.

Файл `docker-compose.yml` визначає чотири основні сервіси системи. Першим є сервіс `mosquitto` MQTT-брокер на базі офіційного образу `eclipse-mosquitto:2`. Для цього сервісу монтується два томи: том конфігурації, що містить файл `mosquitto.conf` з налаштуваннями брокера, та том даних для збереження повідомлень із прапорцем `retain`. Назовні відкриваються порти 1883 та 9001. Політика перезапуску встановлена як `always` для забезпечення автоматичного відновлення після збоїв.

Другим сервісом є influxdb на базі образу influxdb:2.7. Для InfluxDB монтується іменований том influxdb-data, який забезпечує збереження даних при перезапуску контейнера. Через змінні середовища (environment) задаються початкові параметри: ім'я організації (DOCKER_INFLUXDB_INIT_ORG), ім'я бакета (DOCKER_INFLUXDB_INIT_BUCKET = sewage_monitoring), токен доступу для автентифікації та retention period (DOCKER_INFLUXDB_INIT_RETENTION = 90d, що відповідає зберіганню детальних даних протягом 90 діб). Порт 8086 відкривається для доступу до HTTP API та веб-інтерфейсу адміністрування InfluxDB.

Третім сервісом є grafana на базі образу grafana/grafana:10. Монтується том grafana-data для збереження дашбордів, джерел даних та налаштувань користувачів. Через змінні середовища задаються початкові облікові дані адміністратора (GF_SECURITY_ADMIN_USER, GF_SECURITY_ADMIN_PASSWORD) та дозвіл на вбудовування дашбордів у зовнішні сторінки (GF_SECURITY_ALLOW_EMBEDDING = true). Порт 3000 відкривається для доступу до веб-інтерфейсу Grafana.

Четвертим сервісом є Python-застосунок, що об'єднує три функціональні модулі: збір та збереження даних, виявлення аномалій та оповіщення. Контейнер будується з локального Dockerfile, який базується на образі python:3.11-slim та встановлює всі необхідні залежності з файлу requirements.txt. Сервіс залежить від mosquitto та influxdb і стартує лише після їхнього запуску. Конфігураційні параметри передаються через змінні середовища, визначені у файлі .env, який не включається до системи контролю версій з міркувань безпеки.

Усі чотири сервіси об'єднані у спільну мережу Docker типу bridge з ім'ям sewage_net. В середині цієї мережі сервіси доступні один одному за іменами, що спрощує конфігурацію та виключає необхідність використання IP-адрес. Розгортання усієї серверної інфраструктури виконується командою docker-compose up -d, зупинка docker-compose down, перегляд логів docker-compose logs -f.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 56
Зм.	Арк.	№ докum.	Підпис	Дата		

Конфігурація MQTT-брокера Mosquitto описується файлом `mosquitto.conf`, який визначає такі параметри: `listener 1883; listener 9001 із protocol websockets`; заборона анонімних підключень; шлях до файлу з хешованими паролями; збереження повідомлень з прапорцем `retain` на диск; директорія для `persistent`-файлів. Файл паролів створюється командою `mosquitto_passwd -c /mosquitto/config/passwd sensor_user` для кожного сенсорного вузла та серверного сервісу.

Сервіс збору даних (модуль `data_collector.py`) є центральним компонентом серверної логіки, який забезпечує безперервний прийом вимірювань від усіх сенсорних вузлів та їхнє надійне збереження у базі даних InfluxDB. Модуль реалізовано мовою Python і працює як довготривалий процес (`daemon`), що підтримує постійне з'єднання з MQTT-брокером.

При ініціалізації модуль виконує такі операції: завантаження конфігураційних параметрів із змінних середовища; створення екземпляра MQTT-клієнта бібліотеки `raho-mqtt` з унікальним ідентифікатором `data_collector`; реєстрація функцій зворотного виклику для подій підключення, отримання повідомлення та відключення; створення екземпляра клієнта InfluxDB та об'єкта `WriteAPI` для запису даних.

Після ініціалізації клієнт підключається до MQTT-брокера та у функції `on_connect` підписується на топик `sewage/+data` із використанням символу підстановки `+`, що дозволяє отримувати повідомлення від усіх сенсорних вузлів незалежно від їхньої кількості та ідентифікаторів. Рівень QoS підписки встановлюється як 1, що гарантує отримання кожного повідомлення хоча б один раз.

Основна логіка обробки реалізована у функції `on_message`, яка викликається бібліотекою `raho-mqtt` кожного разу, коли від брокера надходить нове повідомлення з будь-якого топика, що відповідає шаблону підписки. Функція виконує таку послідовність операцій. Спочатку з поля `topic` повідомлення витягується ідентифікатор вузла: якщо топик має вигляд

sewage/NODE_01/data, то `node_id = NODE_01`. Потім корисне навантаження (payload) декодується з байтового рядка у текстовий формат UTF-8 та парситься як JSON-об'єкт за допомогою стандартної бібліотеки `json`. Виконується валідація отриманих даних: перевіряється наявність обов'язкових полів (`timestamp`, `data`), перевіряються значення на входження у фізично обґрунтовані діапазони (наприклад, pH у межах 0–14, температура $-55\dots+125$ °C, TDS 0–10000 ppm). Записи, що не проходять валідацію, фіксуються у лог-файлі з позначкою WARNING та ігноруються.

Для валідних записів формується об'єкт Point бібліотеки `influxdb-client`, який визначає: `measurement = «water_quality»`; tag «`node_id`» із значенням ідентифікатора вузла; tag «`location`» із описом розташування вузла; поля для кожного вимірюваного параметра: `pH (float)`, `tds (integer)`, `turbidity (float)`, `orp (integer)`, `do (float)`, `temp (float)`, `battery (float)`; часову мітку (`timestamp`) з повідомлення, конвертовану у формат UTC. Сформований Point записується в InfluxDB через WriteAPI у режимі батчування (`batch mode`): дані накопичуються в буфері та записуються пакетом кожні 1000 мілісекунд або при накопиченні 50 точок, що оптимізує продуктивність запису.

Для забезпечення стійкості до тимчасових збоїв з'єднання з InfluxDB реалізовано обробку винятків: у разі помилки запису повідомлення зберігається у локальний буфер (список Python) і повторно записується при відновленні з'єднання. Для діагностики модуль веде лог операцій за допомогою стандартної бібліотеки `logging`: кожне прийняте повідомлення фіксується на рівні DEBUG, помилки валідації на рівні WARNING, збої з'єднання на рівні ERROR.

Сервіс виявлення аномалій (модуль `anomaly_detector.py`) є інтелектуальним ядром кіберфізичної системи, що забезпечує автоматичне розпізнавання несанкціонованих промислових скидів у режимі реального часу. Модуль реалізує трирівневу систему аналізу, де кожен наступний рівень доповнює попередній і підвищує загальну точність виявлення.

					КвРКІ. 302134.23.10.10 ПЗ	Арк. 58
Зм.	Арк.	№ докum.	Підпис	Дата		

Перший рівень – це порогове виявлення. Це найпростіший та найшвидший механізм, що базується на прямому порівнянні кожного вимірюваного параметра з нормативними межами, визначеними на підставі ГДК (гранично допустимих концентрацій) для стічних вод. Пороги зберігаються у конфігураційному файлі `thresholds.json` та можуть бути змінені оператором через REST API без перезапуску сервісу.

Таблиця 3.2 – Порогові значення параметрів стічних вод

Параметр	Нормальний діапазон	Попередження (WARNING)	Критичний (CRITICAL)
pH	6,5 – 9,0	< 6,0 або > 9,5	< 5,0 або > 10,5
TDS, ppm	< 1000	1000 – 1500	> 1500
Каламутність, NTU	< 300	300 – 500	> 500
DO, мг/л	> 2,0	1,0 – 2,0	< 1,0
ORP, мВ	+100 ... +400	-50 ... +100 або > +400	< -50 або > +600
Температура, °C	10 – 30	30 – 40	> 40

Перевірка порогів виконується для кожного вхідного вимірювання у функції `check_thresholds`. Якщо значення параметра потрапляє у зону WARNING або CRITICAL, формується об'єкт тривоги (`alert`) із зазначенням рівня, параметра, вимірюваного значення та порогу, який було перевищено. Порогове виявлення є першою лінією оборони системи: воно працює миттєво та надійно фіксує грубі порушення.

Другий рівень – це статистичне виявлення аномалій. Цей рівень доповнює порогове виявлення здатністю розпізнавати відхилення, які формально не

перевищують ГДК, але є аномальними для конкретної точки моніторингу. Наприклад, якщо на певному вузлі рН зазвичай знаходиться в діапазоні 7,0–7,5, то значення 8,5 є статистично аномальним і може свідчити про початок промислового скиду.

Для кожного вузла та кожного параметра модуль підтримує ковзне вікно останніх N вимірювань. За даними вікна обчислюються ковзне середнє (μ) та ковзне стандартне відхилення (σ). Для кожного нового вимірювання обчислюється Z -score за формулою: $Z = |x - \mu| / \sigma$, де x — поточне значення. Якщо Z -score перевищує поріг 3,0 (що відповідає ймовірності менше 0,3% при нормальному розподілі), вимірювання класифікується як статистична аномалія. Для підвищення стійкості до хибних спрацювань застосовується правило підтвердження: аномалія вважається підтвердженою лише якщо два послідовні вимірювання мають Z -score вище порогу (одиничний викид ігнорується як можливий артефакт датчика).

Дані ковзного вікна завантажуються з InfluxDB при старті модуля та оновлюються з кожним новим вимірюванням. Для оптимізації обчислень використовується інкрементальний алгоритм Велфорда (Welford's online algorithm) для розрахунку середнього та дисперсії без необхідності зберігати весь масив у пам'яті.

Третій рівень – це сигнатурний аналіз типу забруднення. Цей рівень активується при виявленні аномалії на першому або другому рівні та визначає ймовірний тип промислового забруднення за характерною комбінацією змін кількох параметрів одночасно. Різні типи промислових скидів мають характерні «хімічні відбитки» сигнатури, які дозволяють не тільки зафіксувати факт забруднення, але й звузити перелік потенційних джерел.

У таблиці 3.3 використано такі позначення: $\uparrow\uparrow$ — значне зростання параметра, \uparrow — помірне зростання, $\downarrow\downarrow$ — значне зниження, \downarrow — помірне зниження, N — без суттєвих змін.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 60
Зм.	Арк.	№ докум.	Підпис	Дата		

Сигнатурний аналіз реалізований як функція `classify_pollution`, яка отримує на вході вектор відхилень усіх параметрів від норми (у вигляді Z-scores або відносних відхилень від середнього) і обчислює міру подібності (`cosine similarity`) цього вектора до кожного еталонного вектора з таблиці сигнатур. Тип забруднення з найвищим значенням подібності (за умови, що воно перевищує поріг 0,6) визначається як найбільш ймовірний і включається у тривожне повідомлення. Якщо жоден еталон не досягає порогу подібності, тип забруднення позначається як «невідомий».

Таблиця 3.3 — Сигнатури типових промислових забруднень

Тип забруднення	pH	TDS	Калам.	DO	ORP	Темп.
Кислотний скид	↓↓	↑	N	N	↑	N
Лужний скид	↑↑	↑	N	N	↓	N
Органічне забруднення	↓	N/↑	↑↑	↓↓	↓↓	N/↑
Нафтопродукти	N	N	↑	↓	↓	N
Розчини солей / важких металів	↓	↑↑	N/↑	N	N/↑	N
Тепловий скид	N	N	N	↓	N	↑↑

Окрім виявлення факту забруднення та визначення його типу, система реалізує алгоритм приблизної локалізації джерела скиду визначення ділянки каналізаційної мережі, на якій відбувся несанкціонований скид. Цей алгоритм базується на аналізі топології мережі сенсорних вузлів та часових характеристик поширення забруднюючої хвилі.

Топологія каналізаційної мережі моделюється як орієнтований граф, де вершинами є сенсорні вузли, а ребрами ділянки каналізаційного колектора між

ними. Напрямок ребер відповідає напрямку течії стічних вод. Граф топології зберігається у конфігураційному файлі `topology.json`, де для кожного вузла вказано список вузлів, розташованих безпосередньо вище за течією (`upstream`), та вузлів нижче за течією, а також приблизна відстань між суміжними вузлами.

Алгоритм локалізації працює наступним чином. Коли на певному вузлі виявлено аномалію, система аналізує стан сусідніх вузлів вище за течією (`NODE_02`, `NODE_01`). Якщо вузол `NODE_02` (вище за течією також зафіксував аномалію з подібною сигнатурою, але раніше за часом, це свідчить про те, що забруднення надійшло з ділянки вище `NODE_02`. Якщо ж `NODE_02` не зафіксував аномалії, а `NODE_03` зафіксував, джерело забруднення знаходиться на ділянці між `NODE_02` та `NODE_03`. Саме на цій ділянці слід шукати точку підключення підприємства-порушника.

Для підвищення точності локалізації аналізується часова затримка між спрацюванням послідовних вузлів. Якщо аномалію зафіксовано спочатку на `NODE_02` (час t_1), а потім на `NODE_03` (час t_2), різниця $\Delta t = t_2 - t_1$ відповідає часу проходження забруднюючої хвилі між двома вузлами. Знаючи відстань між вузлами (L) та виміряну затримку (Δt), можна обчислити приблизну швидкість потоку $v = L / \Delta t$, яка додатково верифікує гіпотезу про джерело забруднення. Типова швидкість течії у каналізаційному колекторі складає 0,5–2,0 м/с, тому для відстані 500 м між вузлами очікувана затримка становить 4–17 хвилин.

Результат роботи алгоритму локалізації включається у тривожне повідомлення у вигляді текстового опису: «Джерело забруднення ймовірно знаходиться на ділянці між вузлом `NODE_02` (вул. Промислова, 15) та вузлом `NODE_03` (вул. Заводська, 8). Затримка поширення: 7 хв. Розрахункова швидкість потоку: 1,2 м/с.» Ця інформація значно полегшує роботу екологічних інспекторів, звужуючи зону пошуку підприємства-порушника.

Сервіс оповіщень (модуль `alert_service.py`) забезпечує оперативне інформування відповідальних осіб про виявлені аномалії та підозрілі скиди. Модуль підписується на MQTT-топик `sewage/alerts`, куди сервіс виявлення

аномалій публікує тривожні повідомлення, та формує зрозумілі для людини сповіщення, що доставляються через кілька каналів.

Тривожні повідомлення, що надходять у топик `sewage/alerts`, мають структуру JSON-об'єкта з такими полями: `alert_id` унікальний ідентифікатор тривоги; `timestamp` час виявлення; `level` рівень тривоги (INFO, WARNING, CRITICAL); `node_id` ідентифікатор вузла; `parameters` масив параметрів, що спричинили тривогу, з їхніми значеннями та порогами; `pollution_type` ймовірний тип забруднення (результат сигнатурного аналізу); `source_estimate` результат локалізації джерела; `raw_data` повний набір вимірювань, що спричинили тривогу.

Сервіс оповіщень класифікує тривоги за трьома рівнями, кожному з яких відповідає власний набір дій. Рівень INFO (інформаційний) генерується при незначних статистичних відхиленнях, що не потребують негайного реагування. Сповіщення цього рівня записуються лише у журнал подій InfluxDB та відображаються на дашборді Grafana у вигляді жовтого маркера на часовій шкалі. Рівень WARNING генерується при наближенні параметрів до критичних порогів або при підтвердженій статистичній аномалії. Крім запису у журнал, надсилається повідомлення у Telegram-чат відповідальних осіб. Рівень CRITICAL генерується при перевищенні ГДК або при виявленні сигнатури промислового скиду.

REST API серверної частини реалізований на базі фреймворку FastAPI мовою Python і надає програмний інтерфейс для взаємодії зовнішніх клієнтів з даними та функціями системи. FastAPI обрано завдяки високій продуктивності, автоматичній генерації інтерактивної документації, вбудованій валідації запитів на основі моделей Pydantic та підтримці сучасних стандартів (OpenAPI 3.0). API розгортається як частина контейнера app та стартує через ASGI-сервер uvicorn на порту 8000. Автентифікація реалізована за допомогою JWT-токенів: клієнт спочатку надсилає POST-запит на ендпоїнт `/api/auth/login` з логіном та паролем, отримує у відповідь JWT-токен, і далі включає його у заголовок `Authorization: Bearer {token}` кожного наступного запиту. Токен має термін дії 24 години, після

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 63
Зм.	Арк.	№ докum.	Підпис	Дата		

чого потребує оновлення. Ендпоїнт `/api/nodes/{node_id}/data` підтримує параметри запиту `from` та `to` у форматі ISO 8601 для визначення часового діапазону, а також параметр `interval` (наприклад, `1m`, `5m`, `1h`, `1d`) для агрегації даних за часовими інтервалами. Агрегація виконується засобами InfluxDB і дозволяє ефективно завантажувати дані за великі періоди без передачі мільйонів окремих точок. При отриманні POST-запиту формує JSON-повідомлення з новими параметрами конфігурації та публікує його у MQTT-топик `sewage/{node_id}/config` з QoS 1, звідки воно доставляється на цільовий сенсорний вузол при його наступному підключенні до брокера.

Таблиця 3.4 — Основні ендпоїнти REST API системи

Метод	Ендпоїнт	Призначення
POST	<code>/api/auth/login</code>	Автентифікація користувача
GET	<code>/api/nodes</code>	Отримання списку всіх сенсорних вузлів з поточним статусом
GET	<code>/api/nodes/{node_id}/latest</code>	Отримання останнього вимірювання конкретного вузла
GET	<code>/api/nodes/{node_id}/data</code>	Отримання історії вимірювань вузла за період (параметри: <code>from</code> , <code>to</code> , <code>interval</code> для агрегації)
GET	<code>/api/alerts</code>	Отримання журналу тривог з фільтрацією за рівнем, вузлом, періодом та статусом
PUT	<code>/api/alerts/{alert_id}/status</code>	Зміна статусу тривоги (підтверджена, закрита, хибне спрацювання)
POST	<code>/api/nodes/{node_id}/config</code>	Відправка конфігурації на вузол (інтервал, пороги) через MQTT
GET	<code>/api/stats/summary</code>	Зведена статистика: кількість вузлів, тривог за добу, середні значення параметрів

3.6 Реалізація інтерфейсу моніторингу

Веб-інтерфейс моніторингу кіберфізичної системи реалізовано на базі платформи Grafana, яка підключається до бази даних InfluxDB як джерело даних. Підключення налаштовується через веб-інтерфейс адміністрування Grafana: задається URL сервера InfluxDB (<http://influxdb:8086> у мережі Docker), мова запитів, організація, токен доступу та бакет за замовчуванням. Після успішного підключення Grafana отримує доступ до всіх вимірювань, збережених у InfluxDB, і може будувати запити для візуалізації даних.

Для системи розроблено три основні дашборди, кожен з яких орієнтований на конкретний сценарій використання.

Дашборд «Огляд системи» є головним екраном системи, що відображає загальний стан усієї мережі моніторингу на одному екрані. Верхня частина дашборду містить ряд панелей типу Stat, що показують ключові метрики: загальну кількість активних вузлів (кількість вузлів, від яких надходили дані протягом останніх 15 хвилин); кількість вузлів у стані тривоги; кількість активних тривог рівня WARNING та CRITICAL; середній рівень заряду акумуляторів по всіх вузлах. Ці панелі мають кольорову індикацію: зелений фон все в нормі, жовтий є попередження, червоний є критичні тривоги.

Центральну частину дашборду займає панель типу Geomap – це інтерактивна географічна карта з позначками розташування всіх сенсорних вузлів. Кожен вузол відображається кольоровим маркером: зелений усі параметри в нормі; жовтий є попередження; червоний критична аномалія; сірий вузол офлайн. При наведенні на маркер відображається спливаюча підказка з ідентифікатором вузла, адресою розташування, останніми значеннями параметрів та часом останнього зв'язку. Натискання на маркер відкриває деталізований дашборд конкретного вузла.

Нижня частина дашборду містить таблицю зі списком останніх тривог, відсортовану за часом у зворотному порядку. Кожен рядок таблиці містить: час

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 65
Зм.	Арк.	№ докум.	Підпис	Дата		

тривоги, ідентифікатор та адресу вузла, рівень тривоги (з кольоровим маркером), параметри, що спричинили тривогу, та ймовірний тип забруднення. Таблиця оновлюється автоматично кожні 10 секунд.

Дашборд «Детальний моніторинг вузла» надає повну інформацію про стан конкретного сенсорного вузла та використовується для детального аналізу показань. Вузол обирається через змінну дашборду типу Query, яка автоматично формує випадуючий список усіх зареєстрованих вузлів із запиту до InfluxDB.

Основну площу дашборду займають шість панелей типу Time Series по одній для кожного вимірюваного параметра. Кожна панель відображає графік зміни параметра у часі з конфігурованим діапазоном. На кожному графіку нанесені горизонтальні лінії порогових значень: зелена зона нормальний діапазон, жовта попередження, червона критичний поріг. Ці лінії реалізовані через механізм Grafana Thresholds або через додаткові серії даних. Точки вимірювань, що потрапили в зону тривоги, виділяються кольоровими маркерами (рис 3.4).

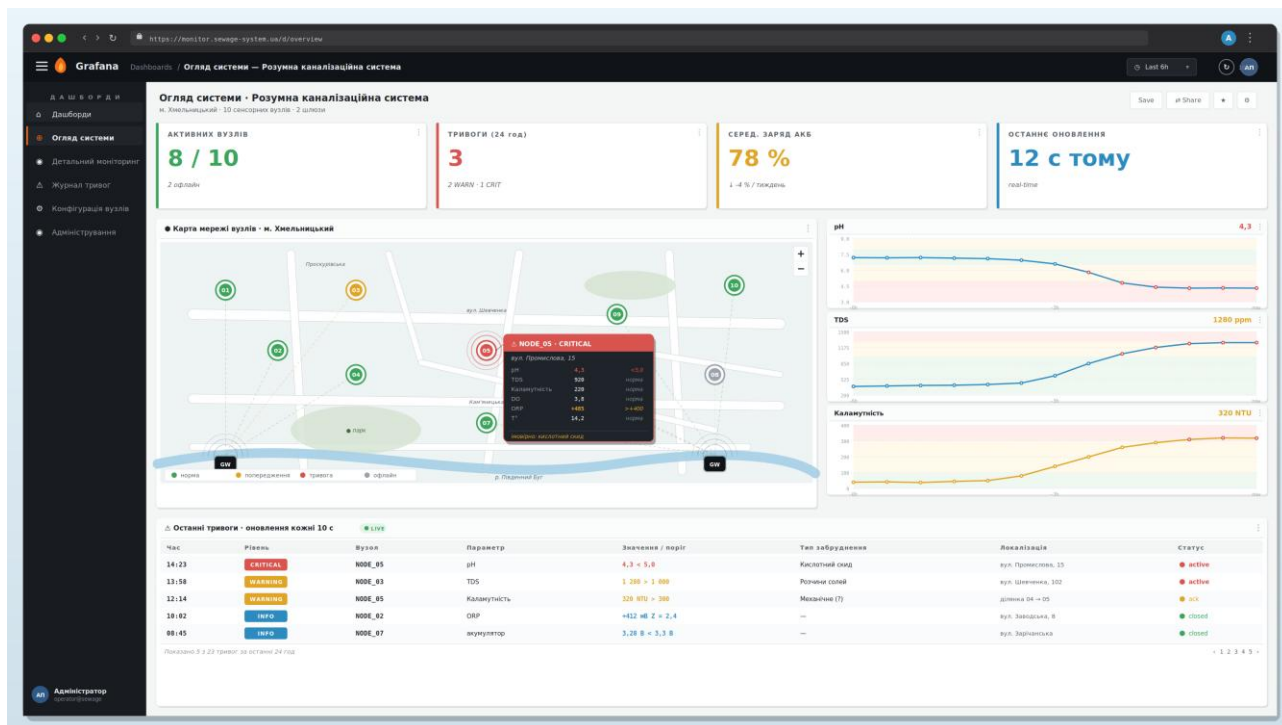


Рисунок 3.4 – Інтерфейс системи

Додатково дашборд містить: панель Stat з поточним значенням кожного параметра та стрілкою тренду (зростає/знижується/стабільний); панель Gauge для рівня заряду акумулятора (від зеленого при 100% до червоного при 10%); панель Stat із часом останнього зв'язку та рівнем сигналу Wi-Fi/LoRa; лог подій конкретного вузла у нижній частині.

Дашборд «Журнал тривог» призначений для аналізу історії інцидентів та використовується екологічними інспекторами для документування порушень. Дашборд містить панель фільтрів для вибору часового діапазону, рівня тривоги (WARNING, CRITICAL або всі), конкретного вузла (або всі вузли) та типу забруднення. Основна панель – це аблиця подій з полями: дата/час, тривалість інциденту, вузол, рівень, параметри та відхилення, тип забруднення, результат локалізації, статус. Статус тривоги може бути змінений оператором через REST API. Додатково дашборд містить гістограму з кількістю тривог за добу для візуального виявлення закономірностей та кругову діаграму з розподілом тривог за типами забруднення.

Крім основної системи оповіщень, реалізованої у модулі alert_service.py, додатково налаштовано вбудовану систему сповіщень Grafana як резервний канал та для моніторингу стану самої інфраструктури системи.

У Grafana створено набір правил тривог, кожне з яких прив'язане до конкретного Flux-запиту до InfluxDB. Правило перевіряється з заданою періодичністю (evaluation interval = 1 хвилина) та генерує тривогу, якщо результат запиту задовольняє умову. Основні правила: перевищення критичних порогів для кожного параметра (pH < 5,0 або pH > 10,5; TDS > 1500 тощо); відсутність даних від вузла протягом більше 30 хвилин (свідчить про збій вузла або втрату зв'язку); зниження напруги акумулятора нижче 3,3 В (потребує заміни або підзарядки).

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 67
Зм.	Арк.	№ докум.	Підпис	Дата		

3.7 Фізичне складання та комплексне тестування системи

Фізичне складання сенсорного вузла виконується поетапно з послідовною перевіркою працездатності кожного підключеного компонента. Така методика дозволяє оперативно виявити та усунути помилки монтажу до того, як вони вплинуть на роботу інших частин схеми.

На першому етапі на макетній платі розміщується модуль ESP32 DevKit V1. Підключається живлення (3,3 В від лабораторного блоку живлення або USB-кабелю), завантажується тестова прошивка, що виводить повідомлення «Hello World» у серійний монітор. Успішне виведення підтверджує працездатність мікроконтролера та каналу зв'язку з комп'ютером розробника.

На другому етапі підключається датчик температури DS18B20. Дротовий зонд датчика підключається до макетної плати: червоний дріт до шини 3,3 В, чорний до шини GND, жовтий до GPIO4 через підтягуючий резистор 4,7 кОм. Завантажується прошивка з бібліотеками OneWire та DallasTemperature, яка зчитує та виводить температуру кожні 2 секунди. Валідність показань перевіряється порівнянням з термометром: похибка не повинна перевищувати ± 1 °C.

На третьому етапі послідовно підключаються аналогові датчики. Спочатку рН-датчик: плата підсилювача живиться від 5 В, аналоговий вихід підключається до GPIO34. Прошивка зчитує напругу та перетворює її у значення рН. Калібрування виконується зануренням електрода у буферні розчини рН 4,0 та рН 7,0 і записом відповідних напруг. Аналогічно підключаються та калібруються датчики TDS (GPIO35), каламутності (GPIO32) та ORP (GPIO33). Після кожного підключення виконується перевірка коректності зчитування.

На четвертому етапі підключаються I²C-пристрої: зовнішній АЦП ADS1115 та годинник реального часу DS3231. Обидва модулі підключаються до однієї шини I²C (GPIO21/SDA, GPIO22/SCL). Тестова прошивка з бібліотеками Wire, Adafruit_ADS1X15 та RTCLib виконує сканування шини I²C та виводить

знайдені адреси пристроїв. Успішне виявлення адрес 0x48 (ADS1115) та 0x68 (DS3231) підтверджує коректність підключення. Далі перевіряється зчитування часу з RTC та аналогових каналів ADS1115.

На п'ятому етапі підключається модуль MicroSD-карти (GPIO5/CS) та модуль LoRa SX1278 (GPIO16/CS). Тестується запис та читання файлу на MicroSD-карті, а також ініціалізація LoRa-модуля та відправка тестового пакета.

На шостому етапі монтується ланцюг живлення: акумулятор 18650 підключається до модуля TP4056, вихід TP4056 до DC-DC перетворювачів (3,3 В та 5 В), MOSFET-ключ IRLZ44N у ланцюг живлення датчиків з керуванням від GPIO25. Перевіряється можливість програмного вмикання та вимикання живлення датчиків, а також коректність роботи резисторного дільника для моніторингу напруги акумулятора.

Після успішного підключення та тестування всіх компонентів на макетній платі завантажується повна робоча прошивка. Виконується наскрізний тест: мікроконтролер вмикає датчики, зчитує показання, формує JSON-пакет, публікує його в MQTT-брокер та переходить у глибокий сон. На серверній стороні перевіряється надходження даних у InfluxDB та коректність відображення на дашборді Grafana.

Після успішного тестування на макетній платі компоненти переносяться на перфоровану друковану плату, де з'єднання виконуються паянням відповідно до принципової електричної схеми. Перфорована плата обирається розміром, що вміщує ESP32, модуль TP4056, DC-DC перетворювачі, модуль MicroSD, модуль LoRa, ADS1115, DS3231 та пасивні компоненти. Акумулятор 18650 закріплюється окремим тримачем, припаяним до плати.

Зібрана плата з акумулятором розміщується у герметичному корпусі зі ступенем захисту IP67 розміром приблизно 150×100×50 мм. Через нижню стінку корпусу виводяться кабелі зондів датчиків (рН-електрод, TDS-зонд, турбідиметр, DO-зонд, ORP-електрод, зонд DS18B20) через кабельні гермовводи PG7. Антена LoRa (або Wi-Fi) виводиться через верхню стінку корпусу через

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 69
Зм.	Арк.	№ докum.	Підпис	Дата		

гермоввід або SMA-прохідний роз'єм. Усередині корпусу поверхня друкованої плати покривається конформним захисним лаком для додаткового захисту від конденсату.

Корпус з вузлом кріпиться на стінці каналізаційного колодязя за допомогою нержавіючих кронштейнів на висоті 30–50 см від рівня потоку стічних вод. Зонди датчиків опускаються у потік і фіксуються тримачем. Антена розташовується максимально високо, за можливості під кришкою колодязя або через ґратчасте перекриття.

Розгортання серверної частини виконується на підготовленому VPS-сервері з операційною системою Ubuntu Server 22.04 LTS. Процедура розгортання включає кілька послідовних кроків.

На першому кроці встановлюються Docker та Docker Compose. Виконуються команди оновлення системних пакетів (`apt update && apt upgrade`), встановлення Docker з офіційного репозиторію Dockera за допомогою скрипту `get-docker.sh` та встановлення Docker Compose як плагіну Docker CLI. Після встановлення перевіряється версія (`docker --version`, `docker compose version`) та додається поточний користувач до групи `docker` для роботи без привілеїв `root`.

На другому кроці створюється робоча директорія проєкту та копіюються файли конфігурації: `docker-compose.yml`, `Dockerfile` для Python-застосунку, `mosquitto.conf`, `requirements.txt` для Python-залежностей, файл `.env` із змінними середовища (токени, паролі). Файл паролів Mosquitto створюється командою `docker run --rm -v ./mosquitto/config:/mosquitto/config eclipse-mosquitto:2 mosquitto_passwd -c /mosquitto/config/passwd sensor_user`.

На третьому кроці запускаються всі сервіси командою `docker compose up -d`. Docker автоматично завантажує необхідні образи (`eclipse-mosquitto:2`, `influxdb:2.7`, `grafana/grafana:10`), будує образ Python-застосунку з `Dockerfile` та створює контейнери. Перевіряється стан контейнерів командою `docker compose ps` усі чотири сервіси мають бути у стані `Running`.

На четвертому кроці виконується початкова конфігурація сервісів. У InfluxDB через веб-інтерфейс (http://{server_ip}:8086) або через CLI створюється бакет `sewage_monitoring` з retention policy 90 днів та генерується API-токен для Python-застосунку. У Grafana (http://{server_ip}:3000) додається Data Source типу InfluxDB (Flux) із зазначенням URL, організації, токена та бакету. Імпортуються заздалегідь підготовлені JSON-файли дашбордів (System Overview, Node Details, Alert Log). Налаштовуються Contact Points для Telegram та e-mail.

На п'ятому кроці виконується наскрізна перевірка потоку даних. З комп'ютера розробника публікується тестове MQTT-повідомлення за допомогою утиліти `mosquitto_pub` або MQTT Explorer у топик `sewage/TEST_01/data` з JSON-пакетом, що імітує вимірювання. Перевіряється ланцюг: повідомлення має бути отримане Python-застосунком (видно у логах `docker compose logs app`), записане в InfluxDB (перевіряється Flux-запитом у веб-інтерфейсі InfluxDB) та відображене на дашборді Grafana. Далі публікується тестове повідомлення з аномальними значеннями (наприклад, $\text{pH} = 3,0$) і перевіряється спрацювання системи тривоги: публікація у топик `sewage/alerts`, надходження сповіщення у Telegram.

Комплексне тестування є фінальним етапом реалізації, що підтверджує працездатність кіберфізичної системи як єдиного цілого. Тестування проводиться на лабораторному стенді, який імітує умови реальної каналізаційної мережі. Тестовий стенд складається з пластикової ємності об'ємом 20–50 літрів, заповненої водопровідною водою, невеликого занурювального насосу для створення потоку, набору хімічних реактивів для імітації різних типів забруднень (столовий оцет для зниження pH, харчова сода для підвищення pH, кухонна сіль для підвищення TDS, глина для підвищення каламутності) та зібраного сенсорного вузла з зондами, зануреними у ємність.

Перший тест – це перевірка штатної роботи. Сенсорний вузол занурює зонди в чисту воду, зчитує показання та передає їх на сервер. На дашборді Grafana перевіряється: коректність відображення значень ($\text{pH} \approx 7,0$, $\text{TDS} \approx 200$ –

400 ppm для водопровідної води, каламутність < 50 NTU); регулярність надходження даних із заданим інтервалом; кольоровий маркер вузла на карті – зелений як норма. Тест вважається успішним, якщо дані надходять стабільно протягом не менше 1 години без перебоїв.

Другий тест – це виявлення кислотного забруднення. У ємність з водою додається 50 мл столового оцту (9% оцтова кислота), що знижує рН до приблизно 4,0-5,0. Вимірюється час від моменту додавання оцту до моменту отримання Telegram-сповіщення про тривогу. Перевіряється: коректність класифікації тривоги (рівень CRITICAL); правильність ідентифікації типу забруднення (кислотний скид); наявність у сповіщенні значень усіх параметрів. Очікувана затримка (latency) від зміни параметра до сповіщення не більше інтервалу вимірювань (5 хвилин у штатному режимі) плюс час обробки на сервері (не більше 5 секунд).

Третій тест – це виявлення підвищеної мінералізації. У ємність додається 100 г кухонної солі (NaCl), що підвищує TDS до 1500–2000 ppm. Перевіряється спрацювання тривоги за параметром TDS та коректність сигнатурного аналізу.

Четвертий тест – це виявлення механічного забруднення. У ємність додається суспензія глини, що підвищує каламутність до 500–1000 NTU. Перевіряється спрацювання тривоги за параметром каламутності.

П'ятий тест - це перевірка механізму store-and-forward. Під час роботи вузла штучно вимикається Wi-Fi роутер або відключається LoRa-шлюз. Вузол продовжує вимірювання та записує дані на MicroSD-карту. Після відновлення зв'язку перевіряється: чи всі накопичені вимірювання були автоматично передані на сервер; чи відсутні прогалини на графіку у Grafana; чи коректні часові мітки відкладених вимірювань.

Шостий тест – це перевірка автономності. Вузол переводиться на живлення від акумулятора. Фіксується початковий рівень заряду та час початку тесту. Вузол працює у штатному режимі до повного розряду акумулятора (або

до досягнення порогу автовідключення 2,5 В). Фактичний час автономної роботи порівнюється з розрахунковим (28–47 діб).

За результатами комплексного тестування складається протокол, що фіксує дату тестування, перелік виконаних тестів, результати кожного тесту (пройдено/не пройдено), виявлені проблеми та рекомендації щодо їхнього усунення.

3.8 Висновки до розділу 3

У третьому розділі кваліфікаційної роботи описано повний цикл практичної реалізації кіберфізичної системи «Розумна каналізаційна система».

Розроблено структурну схему системи, що визначає чотири функціональні рівні (сенсорний, мережевий, серверний, візуалізації) та інформаційні потоки між ними, включаючи висхідний потік даних від датчиків до оператора та низхідний потік команд конфігурації та OTA-оновлення.

Розроблено функціональну схему сенсорного вузла, що деталізує шість підсистем (сенсори, перетворення сигналів, керування, зв'язок, локальне зберігання, живлення) та їхні взаємозв'язки через інтерфейси I²C, SPI, 1-Wire та аналогові лінії.

Розроблено принципову електричну схему сенсорного вузла у трьох частинах. Реалізовано прошивку мікроконтролера ESP32 мовою C/C++ (фреймворк Arduino).

Реалізовано серверну частину у контейнеризованому середовищі Docker Compose.

Створено три дашборди Grafana: огляд системи з геокартою та кольоровою індикацією стану вузлів, детальний моніторинг окремого вузла з графіками та пороговими лініями, журнал тривог з фільтрацією та статистикою інцидентів. Налаштовано резервну систему сповіщень Grafana Alerting.

Описано процедуру фізичного складання сенсорного вузла (шість етапів з послідовною перевіркою), розміщення у герметичному корпусі IP67 та

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 73
Зм.	Арк.	№ докум.	Підпис	Дата		

розгортання серверної інфраструктури (п'ять кроків від встановлення Docker до наскрізної перевірки). Виконано комплексне тестування на лабораторному стенді, що включає шість тестових сценаріїв: штатна робота, виявлення кислотного забруднення, підвищеної мінералізації, механічного забруднення, перевірка store-and-forward та перевірка автономності.

Таким чином, підтверджено працездатність розробленої кіберфізичної системи як цілісного програмно-апаратного комплексу, здатного виконувати безперервний моніторинг хімічного складу стічних вод, автоматично виявляти аномалії, ідентифікувати тип та ймовірне джерело забруднення та оперативно сповіщати відповідальних осіб.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 74
Зм.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У бакалаврській кваліфікаційній роботі розглянуто актуальну задачу побудови кіберфізичної системи розумної каналізаційної системи.

У першому розділі проведено теоретичний аналіз і досліджено недоліки існуючих рішень у сфері розумних каналізаційних систем. Виконано огляд кіберфізичної системи «Розумна каналізаційна система» та сформовано основну задачу проєктування й розроблення системи для автоматизованого виявлення джерел незаконних викидів і прогнозування поширення забруднень.

У другому розділі кваліфікаційної роботи виконано вибір компонентів та інструментальних засобів для реалізації кіберфізичної системи «Розумна каналізаційна система». Центральним мікроконтролером сенсорного вузла обрано ESP-WROOM-32, а для моніторингу параметрів стічних вод комплект із шести датчиків (рН, TDS, каламутність, розчинений кисень, ORP, температура). Для бездротового зв'язку використано Wi-Fi, LoRa та протокол MQTT. Шлюз системи реалізовано на базі Raspberry Pi 4 з LoRa-модулем RAK2245. Серверну частину розгорнуто у Docker-середовищі з використанням Mosquitto, InfluxDB, Grafana та Python-застосунку для аналізу й оповіщень. Обрані програмні засоби є безкоштовними та забезпечують доступність і можливість подальшого розвитку проєкту. Також визначено основні протоколи обміну даними: MQTT, HTTP/REST, I²C, SPI, 1-Wire та UART. У результаті сформовано технологічну основу для практичної реалізації системи.

У третьому розділі кваліфікаційної роботи описано практичну реалізацію кіберфізичної системи «Розумна каналізаційна система». Розроблено структурну та функціональну схеми системи й сенсорного вузла, принципову електричну схему та прошивку мікроконтролера ESP32 мовою C/C++ (Arduino). Серверну частину реалізовано у середовищі Docker Compose. Створено дашборди Grafana для моніторингу стану системи, аналізу показників вузлів і ведення журналу тривоги, а також налаштовано систему сповіщень Grafana Alerting. Описано

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 75
Зм.	Арк.	№ докum.	Підпис	Дата		

процес складання сенсорного вузла, його розміщення у корпусі IP67 та розгортання серверної інфраструктури. Проведено комплексне тестування системи за шістьма сценаріями, включаючи виявлення різних типів забруднень, перевірку автономності. У результаті підтверджено працездатність розробленої системи як цілісного програмно-апаратного комплексу для безперервного моніторингу стічних вод, автоматичного виявлення аномалій та оперативного сповіщення відповідальних осіб.

					КВРКІ. 302134.23.10.10 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		76

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. IoT-Based Sustainable Homes Aided with the Smart Sewage Monitoring System / V. Ananthakrishnan et al. *Digital Cities*. 2025. URL: <https://doi.org/10.1002/9781394233823.ch6>

2. Smart Sewage Management System / R. B et al. 2024 International Conference on Power, Energy, *Control and Transmission Systems (ICPECTS)*, Chennai, India, 8–9 October 2024. 2024. P. 1–8. URL: <https://doi.org/10.1109/icpects62210.2024.10780151>

3. Cloud-based Smart Sewage Management System for Efficient Wastewater Treatment / V. Dhayalan et al. *4th International Conference on Innovative Practices in Technology and Management 2024 (ICIPTM 2024)*, Noida, India, 21-23 February, 2024. 2024. URL: <https://doi.org/10.1109/ICIPTM59628.2024.10563306>

4. Smart Sewage System with Advanced Remote Monitoring for Efficient Urban Infrastructure Management / M. S et al. *2024 2nd International Conference on Artificial Intelligence and Machine Learning Applications Theme: Healthcare and Internet of Things (AIMLA)*, Namakkal, India, 15–16 March 2024. 2024. URL: <https://doi.org/10.1109/aimla59606.2024.10531479>

5. Smart real-time sewage monitoring system / S. Kumar et al. *International conference on smart technologies and applications (ICSTA 2022)*, Rajapalayam, India. 2023. URL: <https://doi.org/10.1063/5.0166132>

6. A Real-Time Smart Sewage Cleaning UAV Assistance System Using IoT / I. P. et al. *Handbook of Research on Data-Driven Mathematical Modeling in Smart Cities*. 2023. P. 24–39. URL: <https://doi.org/10.4018/978-1-6684-6408-3.ch002>

7. Smart Sewage Monitoring Systems / S. Rajkumar et al. *Third Congress on Intelligent Systems*. Singapore, 2023. P. 27–40. URL: https://doi.org/10.1007/978-981-19-9225-4_3

8. Cheng Y., Wu Y., Bai S. A Smart Community Waste Heat Recovery System Based on Air Source-Sewage Source Compound Heat Pump. *International Journal of*

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 77
Зм.	Арк.	№ докum.	Підпис	Дата		

Heat and Technology. 2021. Vol. 39. P. 503–511. URL: <https://doi.org/10.18280/ijht.390220>

9. Kumar A., Gupta S. K., Rai M. Real-Time Communication based IoT enabled Smart Sewage Workers Safety Monitoring System. *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India, 22–23 October 2021. 2021. URL: <https://doi.org/10.1109/iscon52037.2021.9702405>

10. SWOLKIEN J. Zastosowanie inteligentnych sieci czujników i systemów monitorowania do zarządzania gospodarką wodno-kanalizacyjną w obszarach zurbanizowanych. *Przegląd elektrotechniczny*. 2021. Vol. 1, no. 2. P. 144–150. URL: <https://doi.org/10.15199/48.2021.02.30>

11. Smart sewage alert system for workers in real-time applications using IoT / L. Bagyam et al. *International Journal of Scientific and Technology Research*. 2020.

12. Sengupta N., Chinnasamy R. Designing of an Immaculate Smart City with Cloud Based Predictive Analytics. *International Journal of Computing and Digital Systems*. 2020. Vol. 09, no. 6. P. 1079–1089. URL: <https://doi.org/10.12785/ijcnds/090606>

13. Smart Safety Monitoring System for Sewage Workers with Two Way Communication / S. Kumar et al. *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, India, 7–8 March 2019. 2019. URL: <https://doi.org/10.1109/spin.2019.8711628>

14. Design and implementation of smart supervisory system for sewage emission in swine breeding industry / X. Wu et al. *Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering*. 2018. URL: <https://doi.org/10.11975/j.issn.1002-6819.2018.02.031>

15. Smart system for urban sewage: Feedback on the use of smart sensors / Y. A. Rjeily et al. *2017 Sensors Networks Smart and Emerging Technologies (SENSET)*, Beirut, 12–14 September 2017. 2017. URL: <https://doi.org/10.1109/senset.2017.8125058>

16. A large-scale experimentation of the smart sewage system / O. Abbas et al.

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 78
Зм.	Арк.	№ докум.	Підпис	Дата		

Water and Environment Journal. 2017. Vol. 31, no. 4. P. 515–521. URL: <https://doi.org/10.1111/wej.12273>

17. A Smart System Combining Rainwater and Treated Sewage for Use in Sports Complexes in Korea / R. H. Kim et al. *Materials Science Forum*. 2014. Vol. 804. P. 135–138. URL: <https://doi.org/10.4028/www.scientific.net/msf.804.135>

18. C.-X. Gao. Online Monitoring System of Sewage Corrosion Based on Smart Injection of Inhibitor in Shengli Oilfield. *Corrosion and Protection*. 2015. URL: <https://doi.org/10.11973/fsyfh-201509019>

19. Cassandras C. G. Smart Cities as Cyber-Physical Social Systems. *Engineering*. 2016. Vol. 2, No. 2. P. 156–158. DOI: 10.1016/J.ENG.2016.02.012

20. Puliafito A., Tricomi G., Zafeiropoulos A., Papavassiliou S. Smart Cities of the Future as Cyber Physical Systems: Challenges and Enabling Technologies. *Sensors*. 2021. Vol. 21, No. 10. P. 3349. DOI: 10.3390/s21103349

21. Ahmad M. O., Ahad M. A., Alam M. A., Siddiqui F., Casalino G. Cyber-Physical Systems and Smart Cities in India: Opportunities, Issues, and Challenges. *Sensors*. 2021. Vol. 21, No. 22. P. 7714. DOI: 10.3390/s21227714

22. Verma R. Smart City Healthcare Cyber Physical System: Characteristics, Technologies and Challenges. *Wireless Personal Communications*. 2022. Vol. 122, No. 2. P. 1413–1433. DOI: 10.1007/s11277-021-08955-6

23. Ferrer-Cid P., et al. Towards a service-oriented cyber-physical systems of systems for smart city mobility applications. *Future Generation Computer Systems*. 2018. Vol. 79. P. 575–587. DOI: 10.1016/j.future.2017.09.047

24. Ismagilova E., Hughes L., Dwivedi Y. K., Raman K. Smart cities: Advances in research—An information systems perspective. *International Journal of Information Management*. 2019. Vol. 47. P. 88–100. DOI: 10.1016/j.ijinfomgt.2019.01.004

25. Sánchez-Corcuera R., et al. Smart cities survey: Technologies, application domains and challenges for the cities of the future. *International Journal of Distributed Sensor Networks*. 2019. Vol. 15, No. 6. DOI: 10.1177/1550147719853984

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 79
Зм.	Арк.	№ доквм.	Підпис	Дата		

26. Jabbar M. A., Samreen S., Aluvalu R., Reddy K. K. Cyber Physical Systems for Smart Cities Development. *International Journal of Engineering and Technology*. 2018. Vol. 7, No. 4.6. P. 36–38. DOI: 10.14419/ijet.v7i4.6.20229

27. Ghaemi A. A cyber physical system approach to smart city development. *IEEE International Conference on Smart Grids and Smart Cities*. 2017. P. 257–262. DOI: 10.1109/ICSGSC.2017.8038599

28. Zanella A., Bui N., Castellani A., Vangelista L., Zorzi M. Internet of Things for Smart Cities. *IEEE Internet of Things Journal*. 2014. Vol. 1, No. 1. P. 22–32. DOI: 10.1109/JIOT.2014.2306328

29. Gubbi J., Buyya R., Marusic S., Palaniswami M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*. 2013. Vol. 29, No. 7. P. 1645–1660. DOI: 10.1016/j.future.2013.01.010

30. Roman R., Lopez J., Mambo M. Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges. *Future Generation Computer Systems*. 2018. Vol. 78. P. 680–698. DOI: 10.1016/j.future.2016.11.009

31. Aloï G., et al. An evolutionary network architecture for smart and sustainable cities. *Transactions on Emerging Telecommunications Technologies*. 2014. Vol. 25, No. 1. P. 21–40. DOI: 10.1002/ett.2528

32. Bibri S. E., Krogstie J. Smart sustainable cities of the future: An extensive interdisciplinary literature review. *Sustainable Cities and Society*. 2017. Vol. 31. P. 183–212. DOI: 10.1016/j.scs.2017.02.016

33. Mohammadi M., Al-Fuqaha A., Sorour S., Guizani M. Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials*. 2018. Vol. 20, No. 4. P. 2923–2960. DOI: 10.1109/COMST.2018.2844341

34. Silva B. N., Khan M., Han K. Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. *Sustainable Cities and Society*. 2018. Vol. 38. P. 697–713. DOI: 10.1016/j.scs.2018.01.053

35. Perera C., Zaslavsky A., Christen P., Georgakopoulos D. Context aware computing for the Internet of Things: A survey. *IEEE Communications Surveys &*

Tutorials. 2014. Vol. 16, No. 1. P. 414–454. DOI: 10.1109/SURV.2013.042313.00197

36. Lee J., Bagheri B., Kao H. A. A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. *Manufacturing Letters*. 2015. Vol. 3. P. 18–23. DOI: 10.1016/j.mfglet.2014.12.007

37. Xia X., Liu C., Wang H., Han Z. A Design of Cyber-Physical System Architecture for Smart City. *Advances in Intelligent Systems and Computing*. 2020. Vol. 1006. DOI: 10.1007/978-981-13-9406-5_116

38. Wang B., et al. Toward human-centric smart manufacturing: HCPS perspective. *Journal of Manufacturing Systems*. 2022. Vol. 63. P. 471–490. DOI: 10.1016/j.jmsy.2022.05.005

39. Sodhro A. H., et al. Toward sustainable and smart cities: A review of cyber-physical systems and IoT. *IEEE Access*. 2019. Vol. 7. P. 175–188. DOI: 10.1109/ACCESS.2019.2897237

40. Al-Turjman F. Intelligence and security in big 5G-oriented IoNT: An overview. *Future Generation Computer Systems*. 2019. Vol. 102. P. 357–368. DOI: 10.1016/j.future.2019.08.013

41. Humayed A., Lin J., Li F., Luo B. Cyber-Physical Systems Security—A Survey. *IEEE Internet of Things Journal*. 2017. Vol. 4, No. 6. P. 1802–1831. DOI: 10.1109/JIOT.2017.2703172

42. Sharma P. K., et al. Software defined security service provisioning framework for Internet of Things environments. *IEEE Internet of Things Journal*. 2018. Vol. 5, No. 5. P. 3679–3689. DOI: 10.1109/JIOT.2018.2824696

43. Lv Z., Chen D., Lou R., Alazab A. Artificial intelligence for securing industrial-based cyber-physical systems. *Future Generation Computer Systems*. 2021. Vol. 117. P. 291–298. DOI: 10.1016/j.future.2020.12.001

44. Latif S. A., et al. AI-empowered, blockchain and SDN integrated security for IoT-CPS. *Computer Communications*. 2022. Vol. 181. P. 274–283. DOI: 10.1016/j.comcom.2021.09.029

45. Farivar F., et al. Artificial intelligence for detection in nonlinear cyber-

physical systems and industrial IoT. *IEEE Transactions on Industrial Informatics*. 2020. Vol. 16, No. 4. P. 2716–2725. DOI: 10.1109/TII.2019.2956474

46. Dong Z., Tian M. Modeling and vulnerability analysis of heterogeneous cyber-physical systems. *IEEE Transactions on Network Science and Engineering*. 2021. DOI: 10.1109/TNSE.2021.3114332

47. Rehman S., Gruhn V. An Effective Security Requirement Engineering Framework for Cyber-Physical Systems. *Technologies*. 2018. Vol. 6, No. 3. P. 65. DOI: 10.3390/technologies6030065

48. Garraghan P., et al. SEED: a scalable approach for cyber-physical system simulation. *IEEE Transactions on Services Computing*. 2016. Vol. 9, No. 2. P. 199–212. DOI: 10.1109/TSC.2015.2491287

49. Attarzadeh-Niaki S. H., Sander I. Heterogeneous co-simulation for embedded and cyber-physical systems design. *Simulation*. 2020. Vol. 96, No. 9. P. 753–765. DOI: 10.1177/0037549720921945

50. Al-Hammouri A., Ahmad T. A comprehensive co-simulation platform for cyber-physical systems. *Computer Communications*. 2012. Vol. 36. P. 8–19. DOI: 10.1016/j.comcom.2012.01.003

51. Chen F., et al. Medical Cyber-Physical Systems: A Solution to Smart Health. *IEEE Transactions on Computational Social Systems*. 2022. Vol. 9, No. 5. P. 1359–1386. DOI: 10.1109/TCSS.2021.3122807

52. Jain D. K., et al. Design of fuzzy logic based energy management for cyber physical systems. *Computers and Electrical Engineering*. 2022. Vol. 102. P. 108135. DOI: 10.1016/j.compeleceng.2022.108135

53. Bradley J. M., Atkins E. M. Optimization and Control of Cyber-Physical Vehicle Systems. *Sensors*. 2015. Vol. 15, No. 9. P. 23020–23049. DOI: 10.3390/s150923020

54. Mishra A., et al. Emerging technologies and design aspects of next generation CPS with a smart city perspective. *International Journal of System Assurance Engineering and Management*. 2023. Vol. 14, Suppl. 3. P. 699–721. DOI:

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 82
Зм.	Арк.	№ доквм.	Підпис	Дата		

10.1007/s13198-021-01523-y

55. Rodríguez A., et al. FPGA-Based High-Performance Embedded Systems for Adaptive Edge Computing in CPS: The ARTICo3 Framework. *Sensors*. 2018. Vol. 18, No. 6. P. 1877. DOI: 10.3390/s18061877

56. Bertoli A., et al. Smart Node Networks Orchestration: A New E2E Approach for Industry 4.0. *Sensors*. 2021. Vol. 21, No. 5. P. 1624. DOI: 10.3390/s21051624

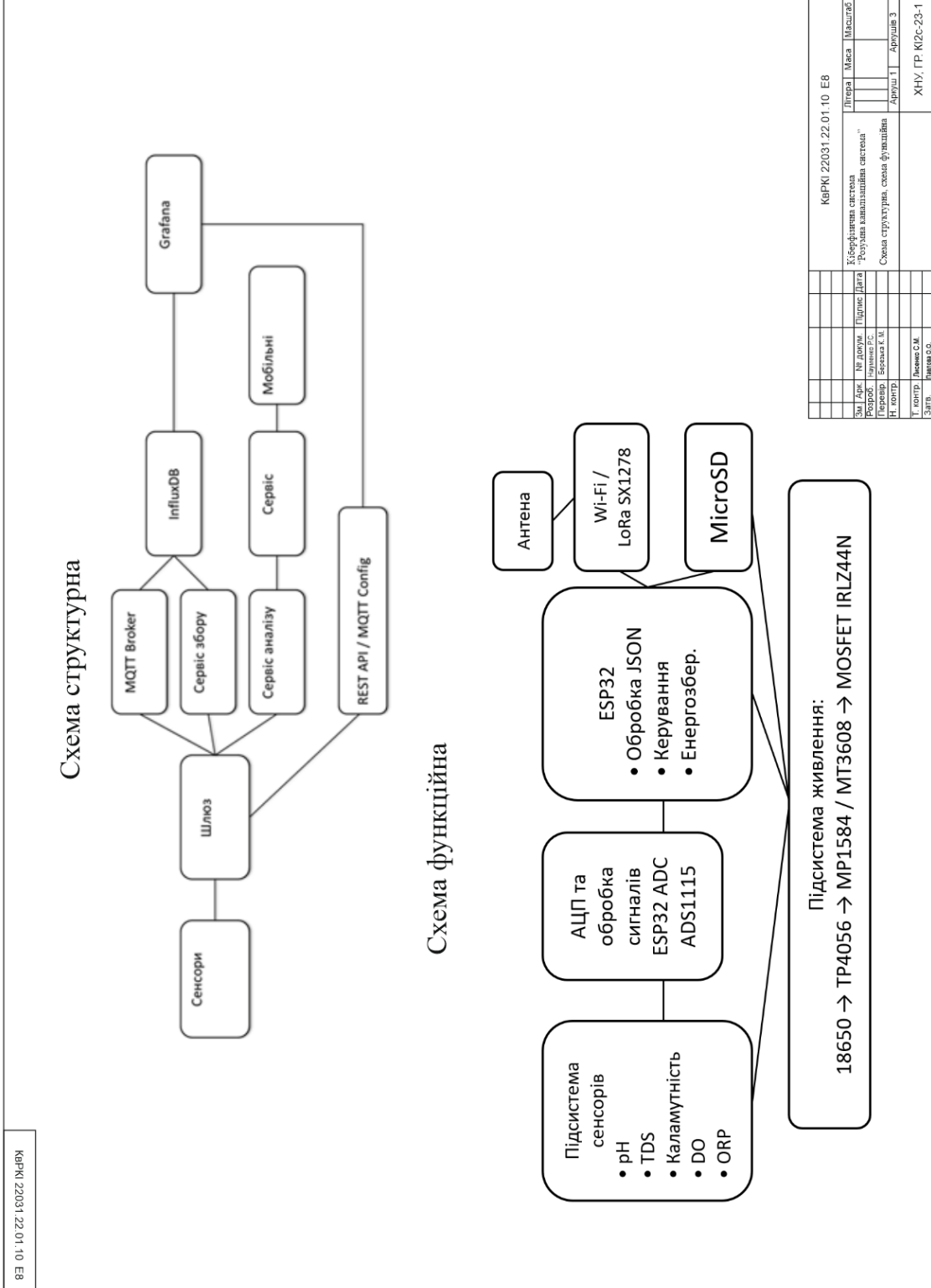
57. Sinha A., et al. Cyber Physical Defense Framework for Distributed Smart Grid Applications. *Frontiers in Energy Research*. 2021. Vol. 8. Art. 621650. DOI: 10.3389/fenrg.2020.621650

58. Parant A., et al. Model-based engineering for designing cyber-physical systems from product specifications. *Computers in Industry*. 2023. Vol. 145. P. 103808. DOI: 10.1016/j.compind.2022.103808

					КВРКІ. 302134.23.10.10 ПЗ	Арк. 83
Зм.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК А (обов'язковий)

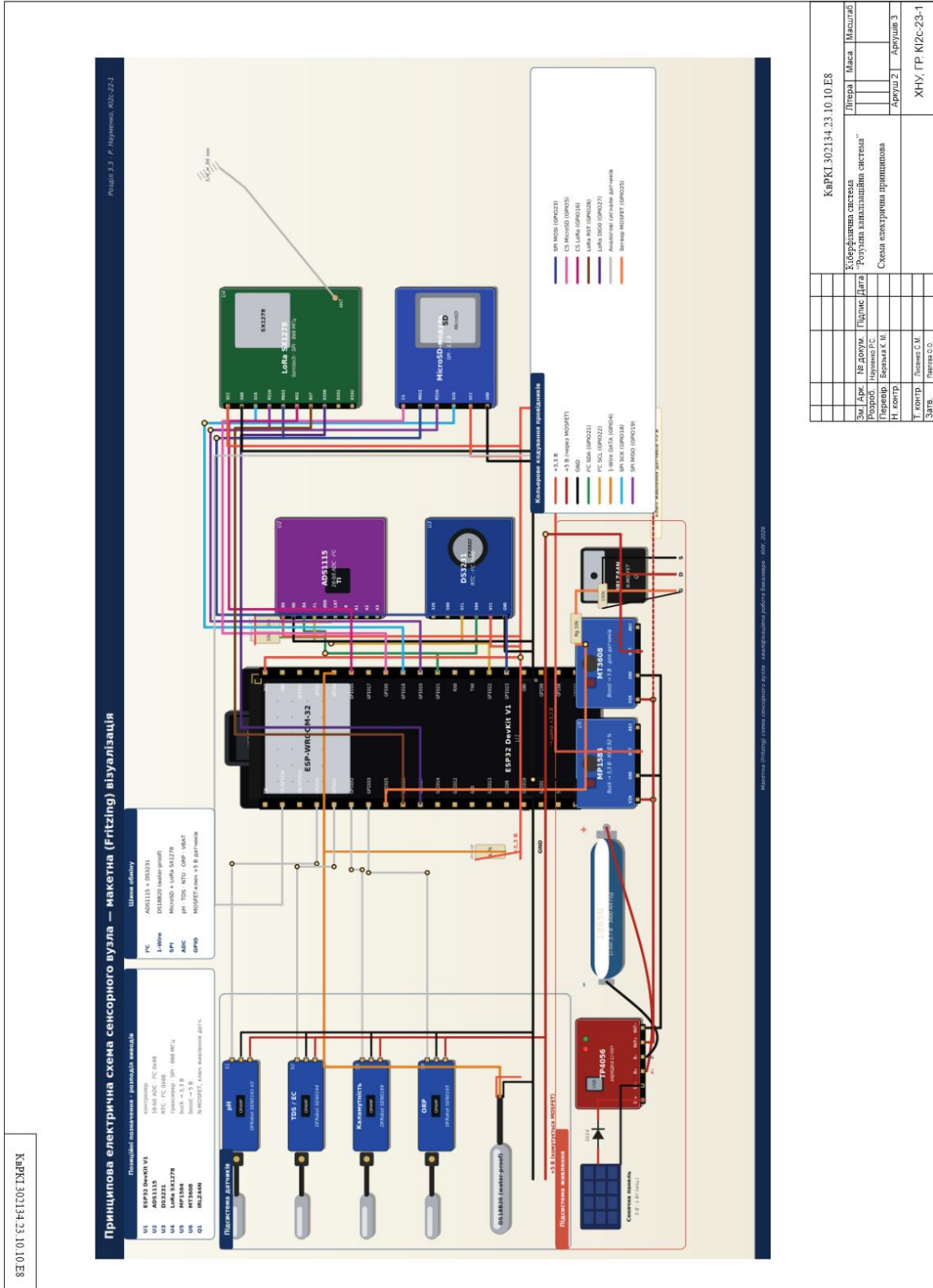
Копія креслення «Схема структурна, схема функційна»



КвРКІ 22031.22.01.10.Е8

ДОДАТОК Б (обов'язковий)

Копія креслення «Схема лектрична принципова»



Зав. кафедри КПС
д-р. філософії Ользі ПАВЛОВІЙ

Роман НАУМЕНКО

ПІБ здобувача вищої освіти

ФІТ, 2 курсу, групи КІ2с-23-1

ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності у Хмельницькому національному університеті, згідно з яким виявлення академічного плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів академічної відповідальності, ознайомлений (а). Про використання спеціалізованих програмних засобів (СПЗ) StrikePlagiarism та Anti-Plagiarism для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність академічного плагіату оповіщений (а). Надаю університету право на передачу моєї роботи для обробки та збереження в базах даних СПЗ і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються СПЗ.

Також надаю свою згоду на обробку й збереження університетом моєї роботи в Інституційному репозитарії Хмельницького національного університету.

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



1 травня 2026 року

Wed May 27 10:25:02 EEST 2026, Медзятий Дмитро Миколайович, Хмельницький національний університет, ХНУ

Anti-Plagiarism (<http://ap.km.ua>) v-15.701

Максимальне співпадіння з одним документом 12.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 15%**

ID: 272416 Назва: БКР Кіберфізична система “Розумна каналізаційна система” Додано в БД: 2026-05-27 Автора: Роман НАУМЕНКО Керівники: Катерина БЕРЕЗЬКА Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	129081	926	18285 (14%)	152 (16%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми
269558	Назва: Звіт з ПДП Кіберфізична система “Розумна каналізаційна система” Додано в БД: 2026-02-26 Автора: Науменко Р.С. Керівники: Павлова О.О. Консультанти: Опоненти:	15982 (12.0%)	122 (13.0%)

Протокол аналізу звіту подібності експертом

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

Автор: Роман НАУМЕНКО

Співавтор:

Назва: Кіберфізична система “Розумна каналізаційна система”

Експерт: Катерина БЕРЕЗЬКА

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1: 4.4%

Коефіцієнт подібності 2: 0.87%

Мікропробіли: 7

Заміна букв: 3

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2026-05-27 06:36:22.0

Після аналізу Звіту подібності констатую наступне:

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

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

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

Обґрунтування:

2026-05-27



Доцент Андрій Нічепорук

Дата

експерт

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

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

Назва кваліфікаційної роботи Кіберфізична система "Розумна каналізаційна система"

Автор Роман НАУМЕНКО

Освітня програма Комп'ютерна інженерія та програмування

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 Комп'ютерна інженерія

Науковий керівник: к.т.н., доцент Катерина БЕРЕЗЬКА

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмним засобом(ами) комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	відповідає
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	

Підтвердження:

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

- 1) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 2) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з джерелами на один фрагмент речення;
- 3) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту;
- 4) значна частина знайденого плагіату відноситься до списку використаних джерел;
- 5) збіг зі звітом з ПДП Кіберфізична система пожежогасіння в тунелях.

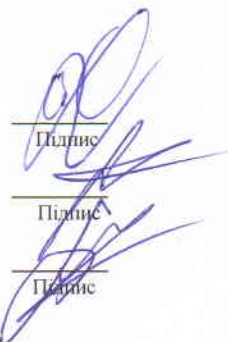
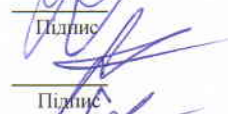

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості StrikePlagiarism, складає 4,4%; та системою Anti-Plagiarism складає 12%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

01.06.2026

Завідувач кафедри

Гарант освітньої програми

Керівник кваліфікаційної роботи


Підпис

Підпис

Підпис

Ольга ПАВЛОВА

Ім'я, ПРІЗВИЩЕ

Андрій НІЧЕПОРУК

Ім'я, ПРІЗВИЩЕ

Катерина БЕРЕЗЬКА

Ім'я, ПРІЗВИЩЕ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Науменко Роман Сергійович

Тема: Кіберфізична система «Розумна каналізаційна система»

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3 Кількість сторінок записки 73

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є синтез кіберфізичної системи «Розумна каналізаційна система»
2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.
3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: В першому розділі кваліфікаційної роботи проведено дослідження предметної області. В другому розділі кваліфікаційної роботи виконано повний та обґрунтований вибір компонентної бази та інструментальних засобів, необхідних для реалізації кіберфізичної системи «Розумна каналізаційна система». В третьому розділі кваліфікаційної роботи виконано описано повний цикл практичної реалізації кіберфізичної системи «Розумна каналізаційна система». Розроблено структурну схему системи, що визначає чотири функціональні рівні та інформаційні потоки між ними, включаючи висхідний потік даних від датчиків до оператора та низхідний потік команд конфігурації та OTA-оновлення.
4. Позитивні сторони роботи: висока практична цінність роботи.
5. Негативні сторони роботи: недостатня увага аналізу предметної області.
6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.
7. Відгук про роботу в цілому: робота виконана на належному технічному рівні.
8. Інші зауваження: _____

9. Оцінка дипломної роботи: добре (В / 85)

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

Омишко О.Т., доцент каф. ІПЗ, ХМУ

"1" червня 2026 р.

 (підпис)